



NEO 2 Interface Developer's Guide

80139403-002 Rev. 101.1
2 August 2021

ID TECH
10721 Walker Street, Cypress, CA 90630
Voice: (714) 761-6368 Fax: (714) 761-8880
idtechproducts.com

Copyright © 2021 ID TECH. All rights reserved.

ID TECH
10721 Walker St.
Cypress, CA 90630

This document, as well as the software and hardware described in it, is furnished under license and may be used or copied online in accordance with the terms of such license. The content of this document is furnished for information use only, is subject to change without notice, and should not be construed as a commitment by ID TECH. Reasonable effort has been made to ensure the accuracy of information provided herein. However, ID TECH assumes no responsibility or liability for any unintentional errors or inaccuracies that may appear in this document.

Except as permitted by such license, no part of this publication may be reproduced or transmitted by electronic, mechanical, recording, or otherwise, or translated into any language form without the express written consent of ID TECH. ID TECH and ViVOpay are trademarks or registered trademarks of ID TECH.

Warranty Disclaimer: The services and hardware are provided "as is" and "as-available" and the use of the services and hardware is at its own risk. ID TECH does not make, and hereby disclaims, any and all other express or implied warranties, including, but not limited to, warranties of merchantability, fitness for a particular purpose, title, and any warranties arising from a course of dealing, usage, or trade practice. ID TECH does not warrant that the services or hardware will be uninterrupted, error-free, or completely secure.

Table of Contents

| | |
|---|-----------|
| 1. INTRODUCTION | 18 |
| 1.1 TERMINOLOGY..... | 18 |
| 1.2 READER MODES..... | 18 |
| 1.3 UNIVERSAL SDK | 18 |
| 1.4 ORGANIZATION OF THIS GUIDE..... | 19 |
| 1.5 NOTATIONAL CONVENTIONS | 19 |
| 1.6 READER INTERFACE CAPABILITIES..... | 20 |
| 2. QUICK REFERENCE | 21 |
| 2.1 COMMAND TABLES | 21 |
| 2.1.1 <i>Commands Sorted by Command Name</i> | 21 |
| 2.1.2 <i>Commands Sorted by Command Number</i> | 27 |
| 2.1.3 <i>Pass-Through Command Table</i> | 33 |
| 2.1.4 <i>EMV Key Manager Command Tables</i> | 34 |
| 2.1.5 <i>Status Codes</i> | 35 |
| 2.2 ERROR CODES..... | 37 |
| 2.3 RF STATE CODES..... | 41 |
| 2.4 EMV RESULT CODES (TAG DF EE 25) | 43 |
| 2.5 SERIAL COMMUNICATION INTERFACES..... | 50 |
| 2.6 BLUETOOTH INTERFACE..... | 50 |
| 2.6.1 <i>Set Bluetooth Parameters (77-81)</i> | 50 |
| 2.6.2 <i>Get Bluetooth Name (77-82)</i> | 51 |
| 2.6.3 <i>Get Bluetooth Local Information (77-83)</i> | 52 |
| 2.6.4 <i>Set Bluetooth Parameters (77-84)</i> | 52 |
| 2.7 RS-232 SERIAL INTERFACE..... | 53 |
| 2.7.1 <i>Port Settings</i> | 53 |
| 2.7.2 <i>Basic Communication</i> | 53 |
| 2.7.3 <i>Timeouts</i> | 54 |
| 2.8 USB HID INTERFACE..... | 54 |
| 2.8.1 <i>HID Report Format</i> | 54 |
| 2.8.2 <i>Sample Single Report Command and Response</i> | 55 |
| 2.8.3 <i>Data Frames</i> | 56 |
| 2.8.4 <i>Sample Single Report Command with Multiple Report Response</i> | 56 |
| 2.8.5 <i>Error Handling at Report Level</i> | 57 |
| 2.8.6 <i>Error Handling at Command Level</i> | 58 |
| 3. VIVOPAY COMMUNICATION PROTOCOLS | 59 |
| 3.1 COMMAND AND RESPONSE FRAMES | 59 |
| 3.1.1 <i>Command Frame Format</i> | 59 |
| 3.1.2 <i>Response Frame Format</i> | 59 |
| 3.2 PASS-THROUGH MODE..... | 59 |
| 3.2.1 <i>Basic Pass-Through Operation</i> | 60 |
| 3.2.2 <i>Pass-Through Command Frame Format</i> | 61 |
| 3.2.3 <i>Pass-Through Response Frame Format</i> | 61 |
| 3.2.4 <i>Suggested Sequence for Pass-Through Commands</i> | 61 |
| 3.2.5 <i>Auto-Switch to Pass-Through Mode</i> | 62 |
| 3.2.6 <i>RF On/Off States for Pass-through Commands</i> | 65 |
| 3.3 BURST MODE..... | 65 |

| | |
|---|------------|
| 3.3.1 ViVOPay Burst Mode Frames | 66 |
| 3.4 CRC CALCULATION..... | 75 |
| 4. TAG AND DATA SET CONFIGURATION | 78 |
| 4.1 CONFIGURABLE AIDs AND GROUPS..... | 79 |
| 4.1.1 System AIDs..... | 81 |
| 4.1.2 User AIDs..... | 82 |
| 4.1.3 Card Brands and AIDs by Group..... | 82 |
| 4.1.4 Reader Default TLV Group..... | 82 |
| 4.1.5 PayPass Default Group..... | 83 |
| 4.1.6 User-Defined TLV Groups | 83 |
| 4.1.7 Configurable AID Reader Memory Requirement..... | 85 |
| 4.1.8 ViVOPay Proprietary TLVs..... | 85 |
| 4.2 CARD APPLICATION PROPRIETARY TAG LIST (FF69 OR FFEE 69) | 85 |
| 4.3 CONFIGURATION TAG TABLES..... | 86 |
| 4.3.1 Group Configuration Tags..... | 86 |
| 4.3.2 Master Group Configuration TLVs | 95 |
| 4.3.3 American Express Group Configuration TLVs | 99 |
| 4.3.4 Discover Group Configuration TLVs..... | 102 |
| 4.3.5 Interac Group Configuration TLVs..... | 105 |
| 4.3.6 AID Configuration Tags | 109 |
| 4.4 2-BYTE TAGS VS. 3-BYTE TAGS..... | 115 |
| 5. CARD APPLICATION SELECTION | 118 |
| 5.1 COMBINED SELECTION | 118 |
| 5.1.1 Selection Features (FFE3 or DFEE4D)..... | 118 |
| 5.2 PARTIAL SELECTION (FFE1 OR DFEE4B) | 119 |
| 5.3 AID PARTICIPATION IN SELECTION PROCESSES (FFE8 OR DFEE53) | 119 |
| 5.4 TERMINAL AID LIST (DFEF2C) | 120 |
| 6. CARD APPLICATION SPECIFIC BEHAVIOR | 121 |
| 6.1 MASTERCARD PAYPASS M/CHIP | 121 |
| 6.1.1 PayPass Default Group..... | 121 |
| 6.1.2 Balance Read Function..... | 121 |
| 6.1.3 Torn Transaction Recovery..... | 122 |
| 6.1.4 EMV Certificate Revocation List..... | 122 |
| 6.1.5 Stop Transaction Command | 122 |
| 6.1.6 Proprietary Tag List..... | 122 |
| 6.1.7 PayPass Personalization Limits..... | 122 |
| 7. PROTOCOL COMMAND REFERENCE: PROTOCOL 2 | 124 |
| 7.1 RTC (REAL TIME CLOCK) COMMANDS | 124 |
| 7.1.1 Get RTC Date and Time (25-06)..... | 124 |
| 7.1.2 Set/Get 24-hr Self-Check Time (25-07) | 125 |
| 7.2 GENERAL COMMANDS..... | 126 |
| 7.2.1 Ping (18-01)..... | 126 |
| 7.2.2 Get DRS Info (C7-3A)..... | 127 |
| 7.2.3 Set Poll Mode (01-01) | 129 |
| 7.2.4 Control User Interface (01-02) | 130 |
| 7.2.5 Set/Get Source for RTC/LCD/Buzzer/LED (01-05)..... | 132 |
| 7.2.6 Set Configuration Defaults (04-09)..... | 135 |

| | |
|---|-----|
| 7.2.7 Set Configuration Defaults and Keep Encryption Key Command (04-0A)..... | 136 |
| 7.2.8 Set Configuration (04-00) | 136 |
| 7.2.9 Get Configuration (03-02)..... | 137 |
| 7.2.10 Get Version: Protocol 2 (29-00)..... | 138 |
| 1.1.2. Get Product Full Information (29-01)..... | 139 |
| 7.2.11 Get USB Boot Loader Version (29-04)..... | 140 |
| 7.2.12 Get Contact EMV L2 Kernel Version (29-06)..... | 140 |
| 7.2.13 Get Contact EMV L2 Kernel Version Detail (29-07)..... | 141 |
| 7.2.14 Get Contact EMV L2 Kernel Checksum (29-08)..... | 141 |
| 7.2.15 Get Contact EMV L2 Terminal Configuration Checksum (29-09)..... | 142 |
| 7.2.16 Get UID of MCU (29-17)..... | 143 |
| 7.2.17 Get TMS (29-20)..... | 143 |
| 7.2.18 Get Health Data from SCRP (29-30)..... | 145 |
| 7.2.19 Set Function/Hardware Enable/Disable (29-40)..... | 146 |
| 7.2.20 Get Function/Hardware Enable/Disable (29-41)..... | 147 |
| 7.2.21 Set Baud Rate (30-01)..... | 148 |
| 7.2.22 Set Temporary Baud Rate (30-02)..... | 149 |
| 7.2.23 Set Baud Rate and Audio Level (30-03)..... | 151 |
| 7.2.24 Set Cable Type (32-02)..... | 152 |
| 7.2.25 Get Cable Type (32-01)..... | 153 |
| 7.2.26 Get Serial Number (12-01)..... | 153 |
| 7.2.27 Get MAC Address (D1-1B)..... | 154 |
| 7.2.28 Get Ethernet MAC Address (D1-11)..... | 154 |
| 7.2.29 Set Network Configuration (D1-13)..... | 155 |
| 7.2.30 Get Network Configuration (D1-15)..... | 156 |
| 7.2.31 Set Wireless Work Mode (D1-19)..... | 156 |
| 7.2.32 Get Wireless Work Mode (D1-1A)..... | 157 |
| 7.2.33 Set Wi-Fi Configuration (D1-17)..... | 157 |
| 7.2.34 Get Wi-Fi Configuration (D1-18)..... | 159 |
| 7.2.35 Set System Language (D1-27)..... | 159 |
| 7.2.36 Set USB Mode Convert and Attribution (D1-30)..... | 160 |
| 7.2.37 Get USB Mode Convert and Attribution (D1-31)..... | 161 |
| 7.2.38 Wi-Fi SSL Connection Request (D1-1C, PCI Only)..... | 162 |
| 7.2.39 Set Ethernet SSL Request Configuration (D2-01, PCI only)..... | 163 |
| 7.2.40 Get Ethernet SSL Request Configuration (D2-02, PCI only)..... | 164 |
| 7.2.41 Enable/Disable Communication Port (D2-03)..... | 165 |
| 7.2.42 Bootup Notification Command (14-01)..... | 165 |
| 7.2.43 Set Model Number (90-15)..... | 167 |
| 7.3 CONFIGURABLE AID AND GROUP COMMANDS..... | 168 |
| 7.3.1 Set Configurable AID (04-02)..... | 168 |
| 7.3.2 Set Configurable Group (04-03)..... | 170 |
| 7.3.3 Get Configurable AID (03-04)..... | 171 |
| 7.3.4 Get Configurable Group (03-06)..... | 172 |
| 7.3.5 Delete Configurable AID (04-04)..... | 173 |
| 7.3.6 Delete Configurable Group (04-05)..... | 174 |
| 7.3.7 Get All AIDs (03-05)..... | 175 |
| 7.3.8 Get All Groups (03-07)..... | 175 |

| | |
|--|-----|
| 7.4 TRANSACTION RELATED COMMANDS: CONTACT | 176 |
| 7.4.1 Setting transaction parameters..... | 176 |
| 7.4.2 Contact EMV L2 Transaction Flow | 177 |
| 7.4.3 Contact Retrieve Application Data (60-01) | 178 |
| 7.4.4 Contact Remove Application Data (60-02) | 178 |
| 7.4.5 Contact Set Application Data (60-03) | 179 |
| 7.4.6 Contact Retrieve Terminal Data (60-04) | 180 |
| 7.4.7 Contact Remove Terminal Data (60-05) | 181 |
| 7.4.8 Contact Set Terminal Data (60-06) | 182 |
| 7.4.9 Contact Retrieve AID List (60-07) | 189 |
| 7.4.10 Contact Retrieve CA Public Key (60-08) | 190 |
| 7.4.11 Contact Remove CA Public Key (60-09) | 191 |
| 7.4.12 Contact Set CA Public Key (60-0A) | 191 |
| 7.4.13 Contact Retrieve CA Public Key List (60-0B) | 192 |
| 7.4.14 Contact Retrieve Certification Revocation List (60-0C) | 193 |
| 7.4.15 Contact Remove Certification Revocation List (60-0D) | 194 |
| 7.4.16 Contact Set Certification Revocation List (60-0E) | 194 |
| 7.4.17 Contact Remove Transaction Amount Log (60-0F) | 195 |
| 7.4.18 Contact Start Transaction (60-10) | 195 |
| 7.4.19 Contact Authenticate Transaction (60-11) | 203 |
| 7.4.20 Contact Apply Host Response (60-12) – Complete Transaction | 207 |
| 7.4.21 Contact Retrieve Transaction Result (60-13) | 211 |
| 7.4.22 Contact Get Reader Status (60-14) | 213 |
| 7.4.23 Contact Get ICS Identification (60-15) | 214 |
| 7.4.24 Contact Set ICS Identification (60-16) | 214 |
| 7.4.25 Contact LCD Display Control (61-01) (Reader sends to Host) | 216 |
| 7.4.26 Contact Get PIN Control (61-02) (Reader or SCRP send to Host) | 218 |
| 7.4.27 Get Random Number (19-01) | 221 |
| 7.4.28 Refresh Enablement Token (19-02) | 222 |
| 7.4.29 Reserve Commands for SCRP (19-10, 19-11, 19-12, 19-13) | 223 |
| 7.4.30 CVMApp Get Nonce from SCRP for Refresh Token (19-15) | 223 |
| 7.4.31 Refresh Enablement Token for SCRP (19-16) | 224 |
| 7.4.32 Contact Get MSR Data Control (Reader sends to Host)(61-03) | 225 |
| 7.5 TRANSACTION RELATED COMMANDS: CONTACTLESS | 227 |
| 7.5.1 Activate Transaction Command (02-01 and 02-40) | 227 |
| 7.5.2 Activate Transaction Response Frame Data Format | 247 |
| 7.5.3 Get Transaction Result (03-00 and 03-40) | 252 |
| 7.5.4 Update Balance Command (03-03) | 258 |
| 7.5.5 Cancel Transaction Command/Reset Transaction Status (05-01) | 260 |
| 7.6 MASTERCARD M/CHIP 3.0 TRANSACTION COMMANDS | 261 |
| 7.6.1 Stop Transaction (05-02) | 261 |
| 7.6.2 Reset Torn Transaction Log (84-0E) | 262 |
| 7.6.3 Clean Torn Transaction Log (84-0F) Command | 262 |
| 7.6.4 Data Exchange Request (02-58) and Data Exchange Response (02-09) | 264 |
| 7.7 KEY MANAGEMENT COMMANDS | 264 |
| 7.7.1 Get CA Public Key (D0-01) | 265 |
| 7.7.2 Get CA Public Key Hash (D0-02) | 266 |

| | |
|--|-----|
| 7.7.3 Set CA Public Key (D0-03)..... | 266 |
| 7.7.4 Delete CA Public Key (D0-04)..... | 267 |
| 7.7.5 Delete All CA Public Keys (D0-05)..... | 268 |
| 7.7.6 Get All CA Public RIDs (D0-06)..... | 268 |
| 7.7.7 List CA Public Key IDs or RID (D0-07)..... | 269 |
| 7.8 MODULE VERSIONING..... | 269 |
| 7.8.1 Get Product Type (09-01)..... | 271 |
| 7.8.2 Get Processor Type (09-02)..... | 272 |
| 7.8.3 Get Main Firmware Version (09-03)..... | 273 |
| 7.8.4 Get Hardware Information (09-14)..... | 274 |
| 7.8.5 Get System Info (09-15)..... | 275 |
| 7.8.6 Get Module Version Information (09-20)..... | 277 |
| 7.8.7 Get TM4 Firmware Version Information (C7-3F)..... | 279 |
| 7.9 INTERNATIONAL LANGUAGE SUPPORT..... | 280 |
| 7.9.1 Other Languages..... | 280 |
| 7.9.2 Bitmap Conversion Completed by POS..... | 281 |
| 7.9.3 ILM Header Format..... | 282 |
| 7.9.4 Language Version Information..... | 282 |
| 7.10 EMV CERTIFICATE REVOCATION LIST COMMANDS..... | 284 |
| 7.10.1 Get EMV Revocation Log Status (84-03)..... | 284 |
| 7.10.2 Add Entry to EMV Revocation List (84-04)..... | 285 |
| 7.10.3 Delete All Entries for Single Index in EMV Revocation List (84-05)..... | 285 |
| 7.10.4 Delete All Entries from EMV Revocation List (84-06)..... | 286 |
| 7.10.5 Get EMV Revocation List (84-07)..... | 286 |
| 7.10.6 Delete an Entry from EMV Revocation List (84-0D)..... | 288 |
| 7.11 EMV EXCEPTION LOG LIST COMMANDS..... | 289 |
| 7.11.1 Get EMV Exception Log Status (84-08)..... | 289 |
| 7.11.2 Add Entry to EMV Exception List (84-09)..... | 290 |
| 7.11.3 Delete Entry from EMV Exception List (84-0A)..... | 291 |
| 7.11.4 Delete All Entries from EMV Exception List (84-0B)..... | 291 |
| 7.11.5 Get EMV Exception List (84-0C)..... | 292 |
| 7.12 GENERIC PASS-THROUGH COMMANDS..... | 293 |
| 7.12.1 Pass-Through Mode Start/Stop (2C-01)..... | 294 |
| 7.12.2 Poll for Token (2C-02)..... | 295 |
| 7.12.3 Get PCD and PICC Parameters (2C-05)..... | 297 |
| 7.12.4 High-level Halt Command (2C-09)..... | 299 |
| 7.12.5 Enhanced Pass-Through Command (2C-0B)..... | 299 |
| 7.12.6 Enhanced Poll for Token (2C-0C)..... | 306 |
| 7.12.7 Get ATR (2C-12)..... | 309 |
| 7.12.8 Contact Transaction Power Off (2C-18)..... | 310 |
| 7.12.9 Set ICC Voltage Option (2C-19)..... | 311 |
| 7.12.10 Get ICC Voltage Option (2C-1A)..... | 312 |
| 7.12.11 Set ICC Reader Type Option (2C-1B)..... | 313 |
| 7.12.12 Get ICC Reader Type Option (2C-1C)..... | 313 |
| 7.12.13 Antenna Control (28-01)..... | 314 |
| 7.13 PASS-THROUGH UI CONTROL..... | 315 |
| 7.13.1 LED Control (0A-02)..... | 315 |

| | |
|---|-----|
| 7.13.2 Front LED Control (0A-03)..... | 316 |
| 7.13.3 Buzzer Control (0B-xx)..... | 317 |
| 7.14 PASS-THROUGH DATA EXCHANGE..... | 318 |
| 7.14.1 Exchange Contactless Data (2C-03)..... | 318 |
| 7.14.2 PCD Single Command Exchange (2C-04) Protocol 2..... | 320 |
| 7.14.3 Exchange APDU Data (2C-13)..... | 325 |
| 7.15 HIGH-LEVEL PASS-THROUGH COMMANDS FOR MIFARE CARDS..... | 328 |
| 7.15.1 Mifare Authenticate Block (2C-06)..... | 328 |
| 7.15.2 Mifare Read Blocks (2C-07)..... | 329 |
| 7.15.3 Mifare Write Blocks (2C-08)..... | 332 |
| 7.15.4 Mifare ePurse Command (2C-0A)..... | 334 |
| 7.16 HIGH-LEVEL PASS-THROUGH COMMANDS FOR NFC CARDS..... | 341 |
| 7.16.1 NFC Commands (2C-40)..... | 341 |
| 7.17 HIGH-LEVEL PASS-THROUGH COMMANDS FOR FELICA CARDS..... | 343 |
| 7.17.1 Felica Commands (2C-41)..... | 343 |
| 7.17.2 Felica Lite/Lite-S Authenticate (2C-42)..... | 344 |
| 7.17.3 Felica Lite/Lite-S Read/Write Blocks with MAC (2C-43)..... | 345 |
| 7.17.4 Pass-Through Package with Felica Card (2C-44)..... | 346 |
| 7.17.5 Felica Get Code (2C-45)..... | 348 |
| 7.17.6 Cancel Felica Get Code (2C-46)..... | 350 |
| 7.18 PASS-THROUGH COMMANDS FOR MOTOR CONTROL..... | 351 |
| 7.18.1 Bypass Data to Motor (2C-70)..... | 351 |
| 7.18.2 Entry Motor Boot Mode (2C-71)..... | 352 |
| 7.19 SECURE PASS-THROUGH FUNCTION..... | 354 |
| 7.19.1 General Introduction..... | 354 |
| 7.19.2 Secure Pass-Thru Whitelist..... | 354 |
| 7.19.3 Handling Sensitive Financial Data..... | 355 |
| 7.19.4 Accessing SAMs in Pass- Thru Mode..... | 355 |
| 7.19.5 Secure Pass-Thru Whitelist..... | 356 |
| 7.19.6 Get Whitelist (2C-51)..... | 356 |
| 7.19.7 Get Discretionary Data Whitelist (2C-53)..... | 356 |
| 7.20 MSR FUNCTION CHECK..... | 358 |
| 7.20.1 Check MSR Swipe Function (2C-20)..... | 358 |
| 7.21 LOGGER COMMANDS..... | 359 |
| 7.21.1 Initialize/De-initialize Logger (4C-01)..... | 359 |
| 7.21.2 Set Logger Level (4C-02)..... | 360 |
| 7.21.3 Get/Set Logger Size (4C-03)..... | 361 |
| 7.21.4 Enable/Disable Logger (4C-04)..... | 362 |
| 7.21.5 Save/Delete Log (4C-05)..... | 362 |
| 7.21.6 Retrieve Log (4C-06)..... | 363 |
| 7.22 SECURE COMMUNICATION..... | 364 |
| 7.22.1 Burst Mode..... | 364 |
| 7.22.2 Data Output..... | 364 |
| 7.22.3 Encryption Algorithms..... | 364 |
| 7.22.4 Secure Data Exchange..... | 365 |
| 7.22.5 Padding of Data Fields..... | 365 |
| 7.22.6 Get Data Encryption Key Variant Type (C 7-30)..... | 365 |

| | |
|---|-----|
| 7.22.7 Set DUKPT Key Encryption Type (C7-32)..... | 366 |
| 7.22.8 Get DUKPT Key Encryption Type (C7-33)..... | 366 |
| 7.22.9 Set Data Encryption Enable Flag (C7-36)..... | 367 |
| 7.22.10 Get Data Encryption Enable Flag (C7-37)..... | 369 |
| 7.22.11 Set MSR Secure Parameters (C7-38)..... | 370 |
| 7.22.12 Get MSR Secure Parameters (C7-39)..... | 371 |
| 7.22.13 Set MSR Tracks (C7-44)..... | 372 |
| 7.22.14 Get MSR Tracks (C7-45)..... | 373 |
| 7.22.15 Load Certificate for TransArmor Encryption (C7-50)..... | 373 |
| 7.22.16 Set TransArmor TID (C7-51)..... | 374 |
| 7.22.17 Get TransArmor TID (C7-52)..... | 374 |
| 7.22.18 Get TransArmor Certificate Status (C7-53)..... | 375 |
| 7.22.19 Get TransArmor Certificate (C7-54)..... | 375 |
| 7.22.20 Set DUKPT PEK Attribution based on KeySlot (C7-A0)..... | 376 |
| 7.22.21 Get DUKPT PEK Attribution based on KeySlot (C7-A1)..... | 377 |
| 7.22.22 Set DUKPT DEK Attribution based on KeySlot (C7-A2)..... | 378 |
| 7.22.23 Get DUKPT DEK Attribution based on KeySlot (C7-A3)..... | 381 |
| 7.22.24 Set KeySlot for DUKPT PEK and DUKPT DEK (C7-A4)..... | 382 |
| 7.22.25 Get KeySlot for DUKPT PEK and DUKPT DEK (C7-A5)..... | 384 |
| 7.23 KEY INJECTION AND RELATED COMMANDS..... | 385 |
| 7.23.1 Get DUKPT Key KSN Extended (81-0B)..... | 385 |
| 7.23.2 Set and Get MAC Verification Output Option (81-40)..... | 386 |
| 7.24 PIN PAD (KEYPAD) COMMANDS..... | 387 |
| 7.24.1 Secondary Response Mechanism..... | 387 |
| 7.24.2 Status and State Codes in Secondary Response..... | 387 |
| 7.24.3 Cancel PIN or Key Entry (62-00)..... | 388 |
| 7.24.4 Display Message and Get Encrypted PIN (62-01)..... | 388 |
| 7.24.5 Get Function Key (62-02)..... | 392 |
| 7.24.6 Display Message and Get Numeric Key (62-03)..... | 394 |
| 7.24.7 Display Message and Get Amount (62-04)..... | 396 |
| 7.24.8 Display Multi Line Message and Get Numeric Key (62-05)..... | 398 |
| 7.24.9 Display Specific Message and Get Numeric Key (62-06)..... | 400 |
| 7.24.10 Get PAN (83-41)..... | 402 |
| 7.25 IMPROVED COLLISION DETECTION..... | 405 |
| 7.25.1 Issues with Standard Collision Detection..... | 405 |
| 7.25.2 Collision Detection Modes..... | 406 |
| 7.26 FIRMWARE DOWNLOADER COMMANDS..... | 408 |
| 7.26.1 Contact Get EMV L2 Kernel Link Type (60-17)..... | 408 |
| 7.26.2 Contact Set EMV L2 Kernel Link Type (60-18)..... | 409 |
| 7.26.3 Enter Boot Loader Process from Main Application (C7-41)..... | 409 |
| 7.26.4 Retrieve Bootloader Related Information (C7-42)..... | 410 |
| 7.26.5 Implement Bootloader loading and verification (C7-43)..... | 411 |
| 7.26.6 Firmware Downloader Command Processing Flow..... | 412 |
| 7.26.7 Start Update Kernel Process (C7-11)..... | 414 |
| 7.26.8 Send Kernel Firmware Data (C7-14)..... | 415 |
| 7.26.9 End Update Kernel Process (C7-15)..... | 416 |
| 7.27 APPLICATION CONTROL COMMANDS..... | 416 |

| | |
|---|------------|
| 7.27.1 ADF Application Control Command (C7-C7) | 416 |
| 7.28 POWER MANAGEMENT COMMANDS..... | 418 |
| 7.28.1 Enter Low Power Mode (F0-03) | 418 |
| 7.28.2 Set Low Power Consumption Configuration (F0-04)..... | 419 |
| 7.28.3 Get Low Power Consumption Configuration (F0-05) | 421 |
| 7.28.4 Get Battery Level (F0-02) | 422 |
| 7.28.5 Set USB Sleep Configuration (F0-06)..... | 422 |
| 7.28.6 Get USB Sleep Configuration (F0-07) | 423 |
| 7.28.7 Wake-Up Notification (F1-00)..... | 423 |
| 7.29 USB POWER MODE SUSPEND CONFIGURATION..... | 424 |
| 7.29.1 Set USB Power Mode Suspend Configuration (F0-08)..... | 424 |
| 7.29.2 Get USB Power Mode Suspend Configuration (F0-09) | 424 |
| 7.30 MISCELLANEOUS NEO 2 PLATFORM COMMANDS..... | 426 |
| 7.30.1 Front Removal Configuration (04-22)..... | 426 |
| 7.30.2 Retrieve Key Info (81-0C)..... | 426 |
| 7.30.3 Retrieve Detailed Key Info (81-0D)..... | 427 |
| 7.30.4 Get Certificate Info (81-0E) | 428 |
| 7.30.5 Get Certificate (81-0F) | 429 |
| 7.30.6 Get Device Status (32-03)..... | 430 |
| 7.30.7 Switches Notifications (F2-00)..... | 430 |
| 7.30.8 Set Switches Notifications Configuration (01-07) | 431 |
| 7.30.9 Get Switches Notifications Configuration (01-08)..... | 432 |
| 7.30.10 Set Key Press Event (01-0E)..... | 432 |
| 7.30.11 Get Key Press Event (01-0F)..... | 433 |
| 7.30.12 Key Press Event (61-04) (Reader sends to Host)..... | 434 |
| 7.31 SPECIAL FUNCTION OR FEATURE CONFIGURATION COMMANDS..... | 435 |
| 7.31.1 Set Special Function or Feature Configuration (C7-82) | 435 |
| 7.31.2 Get Special Customer Configuration (C7-83)..... | 436 |
| 7.31.3 Function/Feature ID Table | 436 |
| 8. PROTOCOL COMMAND REFERENCE: PROTOCOL 3 AND PROTOCOL 5 | 438 |
| 8.1 PROTOCOL 5 COMMAND AND RESPONSE FRAMES..... | 438 |
| 8.2 SUPPORTED COMMANDS..... | 438 |
| 8.2.1 Contact LCD Display Control (61-01) (K81 sends to Android Chip)..... | 438 |
| 9. LOYALTY FUNCTIONS..... | 439 |
| 9.1 APPLE VAS & GOOGLE PAY SMART TAP 2.1 WITH ACT COMMAND | 439 |
| 9.2 APPLEPAY VAS FUNCTIONALITY..... | 442 |
| 9.2.1 Overview..... | 442 |
| 9.2.2 ACT Parameters for ApplePay VAS..... | 443 |
| 9.2.3 ECP 2.0 support | 443 |
| 9.2.4 Tag 9F26 ApplePay Terminal Capabilities Information..... | 444 |
| 9.2.5 Tag DFED3D Terminal Type for ECP 2.0 Format Version '01' | 445 |
| 9.2.6 Tag DFED57 Apple VAS Terminal Type and Data for ECP 2.0 Format Version '02' | 445 |
| 9.2.7 Set Merchant Record (04-11)..... | 450 |
| 9.2.8 Get Merchant Record (03-11)..... | 452 |
| 9.3 GOOGLE PAY SMART TAP 2.1..... | 453 |
| 10. PEER TO PEER FUNCTIONALITY | 454 |
| 10.1 PEER TO PEER SEND A MESSAGE (C7-9A) | 454 |

| | |
|--|------------|
| 10.2 PEER TO PEER RECEIVE A MESSAGE (C7-9B)..... | 454 |
| 11. MSR EQUIVALENT DATA FUNCTION | 456 |
| 11.1 TAG DFEF4B..... | 456 |
| 11.2 SENTINELS..... | 458 |
| 11.3 TAG DFEF4C..... | 458 |
| 11.4 TAG DFEF4D..... | 458 |
| 12. SAMPLE SCENARIOS AND FRAME FLOW | 460 |
| 12.1 CONTACTLESS MAGSTRIPE TRANSACTIONS IN AUTO POLL MODE..... | 460 |
| 12.2 CONTACTLESS MAGSTRIPE TRANSACTIONS IN POLL ON DEMAND MODE | 462 |
| 12.3 EMV (M/CHIP) TRANSACTION IN POLL ON DEMAND MODE..... | 464 |
| 13. APPENDIX A.1: USER EXPERIENCE ILLUSTRATION..... | 467 |
| 14. APPENDIX A.2: AUDIBLE USER INTERFACE | 469 |
| 15. APPENDIX A.3: CONFIGURABLE AID USAGE EXAMPLES | 470 |
| 16. APPENDIX A.4: PREPARING BITMAPS FOR USE WITH ILM | 473 |
| 17. APPENDIX A.5: DEFAULT CONFIGURATION | 477 |
| 17.1 DEFAULT MESSAGE INDEX..... | 479 |
| 18. APPENDIX A.6: VP5300 COMMANDS | 487 |
| 18.1 INITIATE PINPAD SYMMETRIC KEY LOADING (63-01) | 487 |
| 18.2 PINPAD PASS-THROUGH MODE (63-02) | 487 |
| 18.3 SET SESSION KEY (63-03) | 488 |
| 18.4 SET KEY ID (63-04) | 489 |
| 18.5 GET KEY ID (63-05)..... | 489 |
| 18.6 GENERATE MAC FOR HOST (63-06) | 490 |
| 18.7 GET L100 FIRMWARE VERSION (70-03)..... | 490 |
| 19. APPENDIX A.7: VP6800 COMMANDS | 492 |
| 19.1 VP6800 DISPLAY ELEMENTS..... | 492 |
| 19.2 LCD COORDINATION..... | 492 |
| 19.2.1 LCD UI Control Function..... | 493 |
| 19.3 VP6800-SPECIFIC COMMANDS..... | 494 |
| 19.3.1 Show Screen (61-05)..... | 494 |
| 19.3.2 Get Button Event (61-06)..... | 495 |
| 19.3.3 Add Object to Screen (61-07)..... | 495 |
| 19.3.4 Update Object to Screen (61-08)..... | 501 |
| 19.3.5 Add Object to Scroll Container (61-18)..... | 505 |
| 19.3.6 Get List of All Objects on Scrollable Container (61-19) | 507 |
| 19.3.7 Object Event Call Back (61-FF) | 509 |
| 19.3.8 Remove Object from Screen (61-09) | 509 |
| 19.3.9 Create Screen (61-0A)..... | 510 |
| 19.3.10 Destroy Screen (61-0B)..... | 512 |
| 19.3.11 Get Current Screen Name (61-0C) | 513 |
| 19.3.12 Clone Screen (61-0D)..... | 514 |
| 19.3.13 Store Screen Info (61-0E)..... | 515 |
| 19.3.14 Load Screen Info (61-0F)..... | 515 |
| 19.3.15 Clear Screen Info (61-10)..... | 516 |
| 19.3.16 Get All Screens (61-11)..... | 516 |
| 19.3.17 Get List of All Objects on Screen (61-12)..... | 517 |
| 19.3.18 Query Screen by Name (61-13)..... | 518 |

| | |
|--|-----|
| 19.3.19 Query Object By Name (61-14)..... | 519 |
| 19.3.20 Query Screen by ID (61-15)..... | 520 |
| 19.3.21 Query Object by ID (61-16)..... | 521 |
| 19.3.22 Display system screen (61-17)..... | 522 |
| 19.3.23 Turn On or Off Visual LEDs on LCD (61-1A)..... | 522 |
| 19.3.24 Link UI with Transaction Message ID (61-1B)..... | 523 |
| 19.3.25 Get RT1050 Firmware Version (29-80)..... | 524 |
| 19.3.26 Set Backlight (83-2F)..... | 524 |
| 19.3.27 Delete File (83-1F)..... | 525 |
| 19.3.28 Transfer File (83-24)..... | 526 |
| 19.3.29 List Directory (83-22)..... | 527 |
| 19.3.30 Delete Directory (83-26)..... | 529 |
| 19.3.31 Enable/Disable Light Sensor (83-30)..... | 530 |
| 19.3.32 Query File (83-32)..... | 530 |
| 19.3.33 Transfer File to SD Card (83-33)..... | 531 |
| 19.3.34 Delete File from SD Card (83-34)..... | 533 |
| 19.3.35 List Directory on SD Card (83-35)..... | 534 |
| 19.3.36 Query File on SD Card (83-36)..... | 535 |
| 19.3.37 Get SD Card State (83-37)..... | 536 |
| 19.3.38 Read Out File from SD Card (83-39)..... | 537 |
| 19.3.39 Get Light Sensor Value (83-50)..... | 538 |
| 19.3.40 Get External Chip Status (83-51)..... | 539 |
| 19.3.41 Get Battery Level (F0-02)..... | 540 |
| 19.3.42 Get MAC Address (D1-1B)..... | 540 |
| 19.3.43 Append Detailed Information (4C-47)..... | 542 |
| 19.3.44 Start Removal Detection Activation Sequence (61-20)..... | 543 |
| 19.3.45 Reserved for VP6800 Use (61-21)..... | 543 |
| 19.3.46 Start Screen Saver (61-50)..... | 543 |
| 19.3.47 Stop Audio (61-30)..... | 544 |
| 19.3.48 Play Audio (61-31)..... | 544 |
| 19.3.49 Set/Get Audio Volume (61-32)..... | 545 |
| 19.3.50 Set Cancel Button On/Off (61-33)..... | 546 |
| 19.3.51 Get Cancel Button On/Off (61-34)..... | 546 |
| 19.3.52 QR Code Scan Start (61-41)..... | 547 |
| 19.3.53 QR Code Scan Stop (61-42)..... | 550 |
| 19.3.54 Camera Take Photo Start (61-43)..... | 550 |
| 19.3.55 Camera Take Photo Stop (61-44)..... | 553 |
| 19.3.56 Camera Take Photo Once (61-45)..... | 553 |
| 19.4 ADDITIONAL VP6800 INFORMATION..... | 554 |
| 19.4.1 Settings Menu..... | 554 |
| 19.4.2 Press-and-hold Events..... | 554 |
| 19.4.3 Supported special characters..... | 554 |
| 19.4.4 Numeric Entry..... | 555 |
| 19.4.5 Extended Status Codes..... | 555 |
| 19.5 TUTORIAL: CREATING SCREENS FOR THE VP6800 | 556 |
| 19.5.1 Creating Screen Style 1 | 556 |
| 19.5.2 Creating Screen Style 2 | 561 |

| | |
|---|------------|
| 19.5.3 Additional Information About VP6800 Screens..... | 564 |
| 19.6 VP6800 UI LIMITATIONS..... | 564 |
| 19.6.1 VP6800 Supported Fonts | 566 |
| 19.6.2 VP6800 Supported Special Characters..... | 567 |
| 19.7 VP6800-SPECIFIC HARDWARE INTEGRATION COMMANDS: WATCHDOG FOR HOSTS..... | 568 |
| 19.7.1 Set Configure Watchdog (57-00)..... | 568 |
| 19.7.2 Read Configure Watchdog (57-01)..... | 569 |
| 19.7.3 Enable/Disable Watchdog (57-02)..... | 570 |
| 19.7.4 Read Watchdog Enable/Disable Information (57-03)..... | 571 |
| 19.7.5 Feed Watchdog Timer (57-04)..... | 572 |
| 19.7.6 Erase Watchdog Reset Counter and Log (57-05)..... | 572 |
| 19.7.7 Read Watchdog Reset Counter and Log (57-06)..... | 573 |
| 19.8 VP6800 MEMORY-RELATED COMMANDS..... | 574 |
| 19.8.1 Get External Chip Memory Usage Information (09-3 1)..... | 574 |
| 19.9 VP6800 WI-FI CONFIGURATION..... | 576 |
| 19.9.1 VP6800 Wi-Fi Commands..... | 576 |
| 19.9.2 Using VP6800 Wi-Fi Commands..... | 577 |
| 20. APPENDIX A.9: NEO 2 CONTACTLESS GROUPS AND DEFAULT TLVS | 578 |
| 21. APPENDIX A.10: NEO 2 DEFAULT SYSTEM AIDS | 586 |
| 22. APPENDIX A.11: NEO 2 DEFAULT CONFIGURATION TLVS | 587 |
| 23. APPENDIX A.12: CARD IN NOTIFICATION BEHAVIOR | 592 |
| 24. APPENDIX A.13: LED 0 BEHAVIOR STATUS..... | 593 |
| 25. APPENDIX A.14: ENHANCED ENCRYPTED MSR DATA OUTPUT FORMAT FOR FINANCIAL AND NON-FINANCIAL CARDS | 594 |
| 25.1 SRED OUTPUT FORMATS..... | 594 |
| 25.2 NON-SRED OUTPUT FORMATS..... | 595 |
| 26. APPENDIX A.15: COMMON AID | 597 |
| 26.1 COMMON AID PRIORITY EXAMPLE..... | 597 |
| 27. APPENDIX A.16: COMMON AID SELECTION | 600 |
| 27.1 OVERVIEW..... | 600 |
| 27.2 NETWORKS..... | 600 |
| 27.2.1 For Debit Transactions | 600 |
| 27.2.2 Forcing No CVM..... | 601 |
| 27.2.3 Selectable Kernel..... | 601 |
| 27.2.4 Selectable Kernel Tag Fallback | 602 |
| 27.2.5 US Common Debit AIDs | 603 |
| 27.2.6 Global Debit AIDs..... | 603 |
| 27.2.7 Contactless vs Contact AID Selection Behavior | 603 |
| 27.3 US COMMON DEBIT AID SELECTION SOLUTION..... | 604 |
| 27.3.1 Common Debit Preferring: DFED48 – 1 byte..... | 604 |
| 27.3.2 Selectable Kernel Limit: DFED53 - 6 bytes..... | 604 |
| 27.3.3 Selectable Kernel Configuration ICS: DFED54 - 1 byte | 605 |
| 27.3.4 Credit/Debit/Cashback options: DFED55 – 1 byte..... | 605 |
| 27.3.5 Cashback Menu Values: DFED56 - var bytes..... | 605 |
| 27.3.6 Selectable Kernel Tags - FFEF01..... | 605 |
| 27.4 EXAMPLE TRANSACTION FLOW | 606 |
| 27.5 EXAMPLE FCI FROM US VISA GLOBAL..... | 607 |

| | |
|--|------------|
| 27.6 EXAMPLE FCI FROM US COMMON DEBIT | 607 |
| 27.7 CONTACTLESS TRANSACTION FLOW: REQUIREMENTS FOR COMMON AID PREFERRING CONFIGURATION..... | 608 |
| 27.8 CONTACT TRANSACTION FLOW, NO CREDIT/DEBIT/CASHBACK OPTION SELECTABLE KERNEL CONFIGURATION = NO CVM KERNEL | 609 |
| 27.9 CONTACT TRANSACTION FLOW, CREDIT/DEBIT/CASHBACK OPTIONS SELECTABLE KERNEL CONFIGURATION = NO CVM KERNEL | 610 |
| 27.10 SELECTABLE KERNEL FLOWCHART – CONTACT CARD SELECTABLE KERNEL CONFIGURATION - ANY CONFIGURATION | 611 |
| APPENDIX A.17: FASTEMV/DEFERRED APPROVAL | 612 |
| 27.11 AUTO COMPLETE FLOW | 612 |
| 27.12 DISPLAY MESSAGE ON LCD | 612 |
| 27.13 ASSUMPTIONS..... | 613 |
| 27.14 REQUIRED TAGS..... | 613 |
| 28. APPENDIX A.18: VP5300M COMMANDS | 615 |
| 28.1 MOTOR EJECT CARD (60-19) | 615 |
| 28.2 MOTOR CLEAN (60-1A) | 615 |
| 29. APPENDIX A.19: NEW SECURITY LOG EVENT IDS | 617 |
| 30. APPENDIX A.20: SREDKEY2 COMMANDS | 620 |
| 30.1 MSR AND MANUAL PAN ENTRY CONFIGURATION RELATED COMMANDS..... | 620 |
| 30.1.1 Set MSR / Manual PAN Entry Configuration with TLV (C7-80)..... | 620 |
| 30.1.2 Get MSR / Manual PAN Entry Configuration with Tag (C7-81) | 621 |
| 30.2 TLV FORMAT FOR MSR AND MANUAL PAN ENTRY | 622 |
| 30.2.1 Separator & Terminate..... | 622 |
| 30.2.2 Track Prefix & Suffix..... | 622 |
| 30.2.3 Preamble & Postamble..... | 622 |
| 30.2.4 Track StartChar & EndChar | 623 |
| 30.2.5 Sentinel & Account Number Control | 623 |
| 30.2.6 Decoding Direction..... | 623 |
| 30.2.7 Include Mod10 check digit..... | 623 |
| 30.2.8 Sync Bit Check..... | 624 |
| 30.2.9 Clear / Mask Data Option..... | 624 |
| 30.2.10 Track 3 Expiration Date Position Offset..... | 624 |
| 30.2.11 Non-Financial Card Encryption Options | 624 |
| 30.2.12 MSR Output Data Options | 624 |
| 30.2.13 Manual PAN Entry Output Data Options | 624 |
| 30.2.14 Manual PAN Entry Base Configuration | 625 |
| 30.2.15 32.3.15 XML output control | 625 |
| 31. APPENDIX A.21: TRANSACT CREDENTIALS OVER ECP2.0 | 627 |
| 31.1 KEY LOADING..... | 627 |
| 31.1.1 Set Card Keys (C7-67)..... | 627 |
| 31.1.2 Delete Keys (C7-69)..... | 628 |
| 31.1.3 Get Hash of Keys (C7-6A)..... | 628 |
| 31.1.4 Get the Next DEK KSN (C7-6B)..... | 629 |
| 31.1.5 Set KTK Private Key (C7-6C) | 630 |
| 31.2 OUTPUT CONFIGURATION..... | 631 |
| 31.2.1 Tag DFEF4B..... | 631 |
| 31.2.2 Tag DFEF4C..... | 633 |

| | |
|--|------------|
| 31.2.3 Tag DFEF4D..... | 633 |
| 31.2.4 Tag DFEF7B..... | 633 |
| 31.2.5 VAS Encryption..... | 633 |
| 32. APPENDIX B: GLOSSARY | 635 |
| 33. APPENDIX C: REVISION HISTORY..... | 637 |

List of Tables

| | |
|--|-------------------------------------|
| Table 1: Hardware Cross Reference..... | 20 |
| Table 2: Commands Sorted by Command Name | 21 |
| Table 3: Commands Sorted by Command Number..... | 27 |
| Table 4: Pass-Through Command Table..... | 33 |
| Table 5: EMV Key Management – Protocol 2..... | 34 |
| Table 6: Status Codes for Protocol 2..... | 35 |
| Table 7: Protocol 1 Status Codes | Error! Bookmark not defined. |
| Table 8: Error Codes..... | 37 |
| Table 9: RF State Codes..... | 41 |
| Table 10: EMV Result Codes..... | 43 |
| Table 11: Serial Port Settings..... | 53 |
| Table 12: Poll for Token Data Field for Response Frame (Status Code is OK)..... | 64 |
| Table 13: Burst Mode Frames | 66 |
| Table 14: Payload Frame with Cryptogram Data Format and Content When Status OK..... | 68 |
| Table 15: Asynchronous UI Message Event..... | 73 |
| Table 16: Asynchronous UI Message Event Status | 73 |
| Table 17: Asynchronous UI message Event Application Type..... | 74 |
| Table 18: System AIDs..... | 81 |
| Table 19: Group Configuration TLVs | 86 |
| Table 20: PayPass Default Group Configuration TLVs..... | 95 |
| Table 21: Phone Message Table – Hard-Coded Default Value in Kernel..... | 99 |
| Table 22: American Express Default Group A0 Configuration TLVs..... | 99 |
| Table 23: Discover Default Group B0 Configuration TLVs | 103 |
| Table 24: Interac Default Group C0 Configuration TLVs | 106 |
| Table 25: AID Configuration TLVs..... | 109 |
| Table 26: System AID Default Configuration TLVs..... | 112 |
| Table 27: Get Full Track Data Error Codes | Error! Bookmark not defined. |
| Table 28: EMV Key Management Commands Error Codes – Protocol 1 | Error! Bookmark not defined. |
| Table 29: Set CA Public Key Data Field..... | Error! Bookmark not defined. |
| Table 30: Control User Interface Data..... | 131 |
| Table 31: Activate Transaction Command Frame Data Format | 229 |
| Table 32: Activate Command TLVs | 229 |
| Table 33: Activate Transaction Response Frame Data Format: 02-01 Command..... | 231 |
| Table 34: Activate Transaction Response TLVs | 232 |
| Table 35: Activate Transaction Clearing Record TLVs..... | 238 |
| Table 36: Activate Transaction Cause of Failure When Not Requesting Online Authorization..... | 239 |
| Table 37: Activate Transaction Cause of Failure When Requesting Online Authorization | 240 |
| Table 38: Activate Transaction Response Frame Format, Failed Transaction: 02-01 Command | 240 |
| Table 39: Success Transaction--Plaintext data field format (02-01)..... | 248 |
| Table 40: Successful Transaction -- Plaintext and Encrypted data field format for Contactless transaction (02-40) | 249 |
| Table 41: Success Transaction -- Plaintext and Encrypted data field format for MSR card (02-40) | 250 |
| Table 42: Failed Transaction--Plaintext data field format (02-01) | 250 |
| Table 43: Failed Transaction-- Plaintext and Encrypted data field format (02-40) | 251 |
| Table 44: Get Transaction Result Format and Content: Plaintext data field format (03-00)..... | 253 |
| Table 45: Get Transaction Result format for contactless transaction: Plaintext and Encrypted data field format (03-40) | 255 |
| Table 46: Get Transaction Result format for MSR card: Encrypted | 256 |
| Table 47: Update Balance Format and Contents | 258 |
| Table 48: Update Balance Format and Contents When Status OK..... | 259 |
| Table 49: Update Balance Format and Contents When Status Not OK..... | 259 |

| | |
|--|-----|
| Table 50: EMV Key Manager Status Codes – Protocol 2 | 265 |
| Table 51: Language Version Information..... | 282 |
| Table 52: Exception List Record Format | 292 |
| Table 53: Get PCD and PICC Parameters Data Field..... | 298 |
| Table 54: Poll for Token Data Field for Command Frame..... | 296 |
| Table 55: Poll for Token Timeout | 296 |
| Table 56: Poll for Token Data Field for Response Frame (Status Code is OK)..... | 297 |
| Table 57: Enhanced Poll for Token Data Field for Command Frame | 307 |
| Table 58: Enhanced Poll for Token Timeout | 308 |
| Table 59: Enhanced Poll for Token Data Field for Response Frame | 308 |
| Table 60: LED Control Data Field..... | 315 |
| Table 61: Buzzer Control Data Field..... | 317 |
| Table 62: PCD Single Command Exchange Data Field Protocol 2 | 320 |
| Table 63: PCD Commands Protocol 2 | 321 |
| Table 64: PCD Channel Redundancy Register Protocol 2 | 321 |
| Table 65: PCD Single Command Exchange Data Field for Response | 322 |
| Table 66: Halt Command Exchange Between Terminal/PC and Reader | 324 |
| Table 67: Enhanced Pass-Through Data Field | 300 |
| Table 68: Mifare Authentication Block Data Field..... | 328 |
| Table 69: Mifare Read Block Data Field..... | 330 |
| Table 70: Mifare Write Block Data Field | 333 |
| Table 71: ePurse Value Block Format..... | 334 |
| Table 72: Mifare ePurse Command Data Field..... | 336 |
| Table 73: Mifare ePurse Data Field for Debit/Credit Function Block | 336 |
| Table 74: Mifare ePurse Data Field for Backup Function Block | 337 |
| Table 75: NFC Command Set List..... | 341 |
| Table 76: NFC Command Set Response Data List..... | 342 |
| Table 77: Felica Command Set List..... | 343 |
| Table 78: Felica Lite/Lite-S Read/Write Blocks with MAC Data Field..... | 345 |
| Table 79: Response Data Field..... | 392 |
| Table 81: Summary of LCD Messages | 467 |
| Table 84: NEO2 Group 0 default tags | 578 |
| Table 85: NEO 2 Group 80 (Master) default tags | 579 |
| Table 86: NEO 2 Group 90 (Visa) default tags..... | 580 |
| Table 87: NEO 2 Group A0 (Amex) default tags..... | 580 |
| Table 88: NEO 2 Group B0 (Discover) default tags..... | 581 |
| Table 89: NEO 2 Group C0 (Interac) default tags..... | 582 |
| Table 90: NEO 2 Group D0 (JCB EMV) default tags | 582 |
| Table 91: NEO 2 Default System AIDs..... | 586 |
| Table 92: NEO 2 Default Configuration TLVs..... | 587 |
| Table 93: LED 0 Behavior status | 593 |
| Table 94: Enhanced Encrypted MSR Data Output Format for Bank Card..... | 594 |
| Table 95: Enhanced Encrypted MSR Data Output Format for Non-Bank Card (TTK not support)..... | 594 |

1. Introduction

This document provides application developers and integrators with the detailed information necessary to integrate ID TECH ViVOpay readers with point of sale terminals (hosts). It specifies the low-level (firmware) commands and responses that host systems can use to communicate with a ViVOpay reader to carry out contactless EMV transactions. The underlying firmware is ID TECH's proprietary NEO 2 firmware, which runs on the real-time operating system of the reader in question.

A separate guide (similar to this one) exists for NEO 1 readers. Consult that guide for information about those products.

1.1 Terminology

Throughout this manual, the terms "host" refers to the host device (a POS terminal, a tablet, phone, or other machine) to which a ViVOpay reader is connected. Assume that the "host" controls the card "reader," which refers to a ViVOpay credit card reader. The host device is assumed to run a payment app that sends commands to the reader and retrieves data.

This manual describes the commands and responses sent directly to (or received directly from) the reader as raw bytes traveling over a USB, RS-232, Bluetooth, or audio jack connection. The raw bytes are described in terms of hexadecimal values throughout this guide (for example, 09h or 0x09) unless otherwise noted.

1.2 Reader Modes

Before the introduction of contactless EMV, ViVOpay readers usually worked in standalone mode, which does require a host device to initiate a transaction. In this mode, the reader detects and reads cards (and outputs transaction data) independently and autonomously. This mode is commonly referred to as "Auto Poll Mode."

ViVOpay readers can also function in a command/response mode ("Poll on Demand") to provide EMV functionality and fast processing of contactless EMV cards. This approach minimizes the time a cardholder needs to hold a contactless EMV card in front of a reader. However, it requires that the controlling application initiate all interactions with the reader.

1.3 Universal SDK

Developers who require high-level-language support should consider using ID TECH's Universal SDK when integrating NEO 2 readers into an EMV L3-certifiable solution. The Universal SDK is available for Android, iOS, Windows, MacOS, and Linux, with libraries and sample code in a variety of high-level languages (varying by platform).

To obtain the Universal SDK, go to the [ID TECH Knowledge Base](#), select the product of interest, and click through to the appropriate tools. The Universal SDK is available free of charge to all ID TECH customers and is available on demand.

1.4 Organization of this Guide

This document provides the details of how to communicate with ViVOpay readers, including the physical connections, the ViVOpay command protocols, and the actual serial commands required for interacting with the reader. The document is organized into major sections that contain increasing levels of detail:

- The [Quick Reference](#) section includes tables of commands, plus error and status codes. It acts as a quick index into the Protocol Command Reference sections and a quick reference for decoding serial commands and responses.
- The [Serial Communication Interfaces](#) section discusses the available serial interfaces.
- The [ViVOpay Communication Protocols](#) section provides information on the various protocols and modes of communication and describes the frame formats used by each of the protocols.
- The [Tag and Data Set Configuration](#) describes the method for configuring AIDs and groups (parameter/data sets).
- The [Card Application Selection](#) section describes the method for selecting a particular card application and how selection of a particular AID may be controlled.
- The section on [Card Application Specific Behavior](#) describes information specific to particular card applications and the ViVOpay implementation.
- The [Protocol Command Reference](#) section describes the available commands, their frame formats, and the response formats.
- The [Special Reader Features](#) section describes additional features that may optionally be used in conjunction with ViVOpay readers. Some features are specific to a particular ViVOpay reader hardware platform.
- The appendices in the back of this guide contain many useful examples of serial communication flows as well as examples of how to parse data payloads received during transactions.

1.5 Notational Conventions

Many of the tables used in this document describe data objects as TLV (tag, length, value) elements. The details of how TLVs are encoded and explained in the BER-TLV rules found in [EMV 4.2 Book 3, Annex B](#).

Note that although ViVOpay readers support most EMVCo-defined (industry standard) TLVs, they also utilize a number of proprietary, custom TLVs. Some of those TLVs are described in this document. For a comprehensive listing and description of ID TECH's proprietary tags, download the *P/N 80000503-001 TLV Tag Reference Guide*, available free of charge on the [ID TECH Knowledge Base](#).

In this guide, hexadecimal numbers are expressed in one of two ways:

- With an "h" after the number; for example, 2Ah
- With a "0x" preceding the number; for example, 0x2A

When a long stream of numbers is shown, assume that the digits are hex nibbles unless otherwise noted.

NOTE: In TLVs, when a two-byte data value appears, assume the value is big-endian (MSB followed by LSB) unless otherwise noted.

1.6 Reader Interface Capabilities

ViVOpay readers are categorized by their capability to interact with the host device. ViVOpay readers fall into one of the following categories depending on the available transaction interfaces:

- Contactless Only
- Contactless and MSR
- Contactless and LCD Display
- Contactless, MSR, and LCD Display
- Contactless, MSR, and Line Display

The following table categorizes ViVOpay readers by available interfaces.

Hardware Cross Reference

| Reader | Contactless | MSR | LCD Display | Line Display |
|--------|-------------|-----|-------------|--------------|
| VP3600 | • | • | | • |
| VP5300 | • * | • | | • ‡ |
| VP6300 | • | • | • | |
| VP6800 | • | • | • | |

*optional, with additional hardware

‡optional PIN pad

2. Quick Reference

This section contains tables for looking up commands, status codes, and error codes.

2.1 Command Tables

The tables in this section organize the commands by their names and by their command number.

2.1.1 Commands Sorted by Command Name

Commands Sorted by Command Name

| Command | C'less or C'less + MSR | CMD | SUB CMD | Notes |
|--|---------------------------|-----|---------|-------|
| Activate Transaction Command | √ | 02 | 01 | |
| Activate Transaction Command | √ | 02 | 40 | |
| Add Entry to EMV Exception List | √ | 84 | 09 | |
| Add Entry to EMV Revocation List | √ | 84 | 04 | |
| Add Object to Scroll Container | | 61 | 18 | |
| ADF Application Control Command | | C7 | C7 | |
| Antenna Control | √ | 28 | 01 | |
| Boot up Notification | √ | 14 | 01 | |
| Buzzer Control Long | √ | 0B | 02 | |
| Buzzer Control Short | √ | 0B | 01 | |
| Camera Take Photo Once | | 61 | 45 | |
| Camera Take Photo Start | | 61 | 43 | |
| Camera Take Photo Stop | | 61 | 44 | |
| Cancel Felica Get Code | | 2C | 46 | |
| Cancel Transaction Command | √ | 05 | 01 | |
| Check DUKPT Key | √ | 81 | 04 | |
| Check DUKPT Keys | √ | 81 | 02 | |
| Check MSR Swipe Function | | 2C | 20 | |
| Clean Torn Transaction Log | √ | 84 | 0F | |
| Contact Apply Host Response | | 60 | 12 | |
| Contact Authenticate Transaction | | 60 | 11 | |
| Contact Get MSR Data Control (Reader send to Host) | | 61 | 03 | |
| Contact Get PIN Control (Reader send to Host) | | 61 | 02 | |
| Contact Get Reader Status | | 60 | 14 | |
| Contact LCD Display Control (Reader send to Host) | | 61 | 01 | |
| Contact Remove Application Data | | 60 | 02 | a |
| Contact Remove CA Public Key | | 60 | 09 | a |
| Contact Remove Certification Revocation List | | 60 | 0D | a |
| Contact Remove Terminal Data | | 60 | 05 | a |
| Contact Retrieve AID List | | 60 | 07 | a |
| Contact Retrieve Application Data | | 60 | 01 | a |
| Contact Retrieve CA Public Key | | 60 | 0A | a |
| Contact Retrieve CA Public Key List | | 60 | 0B | a |
| Contact Retrieve Certification Revocation List | | 60 | 0C | a |
| Contact Retrieve Terminal Data | | 60 | 04 | a |
| Contact Retrieve Transaction Result | | 60 | 13 | a |
| Contact Set Application Data | | 60 | 03 | a |
| Contact Set CA Public Key | | 60 | 0A | a |

| Command | C'less or C'less + MSR | CMD | SUB CMD | Notes |
|--|------------------------|-----|---------|-------|
| Contact Set Certification Revocation List | | 60 | 0E | a |
| Contact Set Terminal Data | | 60 | 06 | a |
| Contact Start Transaction | | 60 | 10 | a |
| Contact Get ICS Identification | | 60 | 15 | a |
| Contact Remove Transaction Amount Log | | 60 | 0F | a |
| Contact Set ICS Identification | | 60 | 16 | a |
| Control User Interface | | 01 | 02 | |
| Data Exchange Request | √ | 02 | 58 | |
| Data Exchange Response | √ | 02 | 09 | |
| Delete All CA Public Keys Protocol 1 | √ | 24 | 03 | |
| Delete All CA Public Keys Protocol 2 | √ | D0 | 05 | |
| Delete All Entries for Single Index in EMV Revocation List | √ | 84 | 05 | |
| Delete All Entries from EMV Exception List | √ | 84 | 0B | |
| Delete All Entries from EMV Revocation List | √ | 84 | 06 | |
| Delete CA Public Key Protocol 1 | √ | 24 | 02 | |
| Delete CA Public Key Protocol 2 | √ | D0 | 04 | |
| Delete Configurable AID | √ | 04 | 04 | b |
| Delete Configurable Group (DCG) | √ | 04 | 05 | b |
| Delete Entry from EMV Exception List | √ | 84 | 0A | |
| Delete File from SD Card | | 83 | 34 | |
| Delete Keys | | C7 | 69 | |
| Disable Blue LED Sequence | | F0 | F6 | |
| Display Multi Line Message and Get Numeric Key | | 62 | 05 | |
| Display Specific Message and Get Numeric Key | | 62 | 06 | |
| Enable/Disable Logger | | 4C | 04 | |
| Enable/Disable Watchdog | | 57 | 02 | |
| Enable/Disable Communication Port | | D2 | 03 | |
| Enable/Disable Light Sensor | | 83 | 30 | |
| Enhanced Pass-Through Command | √ | 2C | 0B | |
| Enhanced Poll for Token | √ | 2C | 0C | |
| Erase Watchdog Reset Counter and Log | | 57 | 05 | |
| Exchange APDU Data | √ | 2C | 13 | |
| Exchange Contactless Data | √ | 2C | 03 | |
| Feed Watchdog Timer | | 57 | 04 | |
| Felica Commands | √ | 2C | 41 | |
| Felica Get Code | | 2C | 45 | |
| Felica Lite/Lite-S Authentication | √ | 2C | 42 | |
| Felica Lite/Lite-S Read/Write Blocks with MAC | √ | 2C | 43 | |
| Flush Track Data | √ | 17 | 02 | |
| Front Removal Configuration | | 04 | 22 | |
| Get/Set 24-hr Self-Check Time | √ | 25 | 07 | |
| Get/Set Logger Size | | 4C | 03 | |
| Get Data encryption Key Encryption Type | √ | C7 | 33 | |
| Get All AIDs | √ | 03 | 05 | b |
| Get All CA Public RIDs Protocol 2 | √ | D0 | 06 | |
| Get All Groups (GAG) | √ | 03 | 07 | b |
| Get ALL Reader Variables | √ | 09 | 00 | |
| Get ATR | √ | 2C | 12 | |

| Command | C'less or C'less + MSR | CMD | SUB CMD | Notes |
|--|------------------------|-----|---------|-------|
| Get Bluetooth Name | | 77 | 82 | |
| Get Bluetooth Local Information | | 77 | 83 | |
| Get Cable Type | √ | 32 | 01 | |
| Get CA Public Key Hash Protocol 2 | √ | D0 | 02 | |
| Get CA Public Key Protocol 2 | √ | D0 | 01 | |
| Get Cancel Button On/Off | | 61 | 34 | |
| Get Configurable AID | √ | 03 | 04 | b |
| Get Configurable Group | √ | 03 | 06 | b |
| Get Configuration | √ | 03 | 02 | |
| Get Contact EMV L2 Kernel Checksum | √ | 29 | 08 | |
| Get Contact EMV L2 Kernel Version | √ | 29 | 06 | |
| Get Contact EMV L2 Kernel Version Detail | √ | 29 | 07 | |
| Get Contact EMV L2 Terminal Configuration Checksum | √ | 29 | 09 | |
| Get External Chip Memory Usage Information | | 09 | 31 | |
| Get Discretionary Data Whitelist | √ | 2C | 53 | |
| Get DUKPT Key Serial Number Extended | √ | 81 | 0B | |
| Get EMV Exception List | √ | 84 | 0C | |
| Get EMV Exception Log Status | √ | 84 | 08 | |
| Get EMV Revocation List | √ | 84 | 07 | |
| Get EMV Revocation Log Status | √ | 84 | 03 | |
| Get Encrypted PIN | | 62 | 01 | |
| Set Ethernet SSL Request Configuration | | D2 | 02 | |
| Get Firmware Full Version | √ | 29 | 00 | |
| Get Full Track Data | √ | 17 | CD | |
| Get Function/Hardware Enable/Disable | | 29 | 41 | |
| Get Hardware Information | √ | 09 | 14 | |
| Get Hash of Keys | | C7 | 6A | |
| Get Data Encryption Enable Flag | | C7 | 37 | |
| Get Data Encryption Key Variant Type | | C7 | 30 | |
| Get DUKPT DEK Attribution based on KeySlot | | C7 | A3 | |
| Get DUKPT PEK Attribution based on KeySlot | | C7 | A1 | |
| Get Key Press Event | | 01 | 0F | |
| Get KeySlot for DUKPT PEK and DUKPT DEK | | C7 | A5 | |
| Get List of All Objects on Scrollable Container | | 61 | 19 | |
| Get Main Firmware Version | | 09 | 03 | |
| Get Merchant Record | √ | 03 | 11 | |
| Get Module Version Information | √ | 09 | 20 | |
| Get MSR / Manual PAN Entry Configuration with TLV | | C7 | 81 | |
| Get the Next DEK KSN | | C7 | 6B | |
| Get PCD and PICC Parameters | √ | 2C | 05 | |
| Get Processor Type | √ | 09 | 02 | |
| Get Product full information | | 29 | 01 | |
| Get Product Type | | 09 | 01 | |
| Get RTC Date and Time | | 25 | 06 | |
| Get SD Card State | | 83 | 37 | |
| Get MSR Secure Parameters | | C7 | 39 | |
| Get Serial Number | √ | 12 | 01 | |
| Get Special Function or Feature Configuration | | C7 | 83 | |

| Command | C'less or C'less + MSR | CMD | SUB CMD | Notes |
|--|------------------------|-----|---------|-------|
| Get Switches Notifications Configuration | | 01 | 08 | |
| Get System Info | | 09 | 15 | |
| Get TMS | | 29 | 20 | |
| Get TM4 Firmware Version Information | | C7 | 3F | |
| Get Transaction Result | √ | 03 | 00 | |
| Get Transaction Result | √ | 03 | 40 | |
| Get UID of MCU | √ | 29 | 17 | |
| Get USB Boot Loader Version | √ | 29 | 04 | d |
| Get USB Mode Convert and Attribution | | D1 | 31 | |
| Get USB Power Mode Suspend Configuration | | F0 | 09 | d |
| Get Whitelist | √ | 2C | 51 | |
| High-level Halt Command | √ | 2C | 09 | |
| Initialize/De-initialize Logger | | 4C | 01 | |
| Key Press Event (Reader sends to Host) | | 61 | 04 | |
| LED Control | √ | 0A | 02 | |
| Link UI with Transaction Message ID | | 61 | 1B | |
| List CA Public Key IDs or RID Protocol 2 | √ | D0 | 07 | |
| List Directory on SD Card | | 83 | 35 | |
| Mifare Authenticate Block | √ | 2C | 06 | |
| Mifare ePurse Command | √ | 2C | 0A | |
| Mifare Read Blocks | √ | 2C | 07 | |
| Mifare Write Blocks | √ | 2C | 08 | |
| Motor Clean | | 60 | 1A | |
| Motor Eject Card | | 60 | 19 | |
| NFC Commands | √ | 2C | 08 | |
| Pass-through Mode Start/Stop | √ | 2C | 01 | |
| Pass-through Mode (VP5300) | | 63 | 02 | |
| Pass-Through Package with Felica Card | | 2C | 44 | |
| PCD Single Command Exchange | √ | 2C | 04 | |
| Peer To Peer Send A Message | √ | C7 | 9A | |
| Peer To Peer Receive A Message | √ | C7 | 9B | |
| Play Audio | | 61 | 30 | |
| QR Code Scan Start | | 61 | 41 | |
| QR Code Scan Stop | | 61 | 42 | |
| Query File on SD Card | | 83 | 36 | |
| Read Configure Watchdog | | 57 | 01 | |
| Read Out File from SD Card | | 83 | 39 | |
| Read Watchdog Enable/Disable Information | | 57 | 03 | |
| Read Watchdog Reset Counter and Log | | 57 | 06 | |
| Reset Torn Transaction Log | √ | 84 | 0E | |
| Retrieve Log | | 4C | 06 | |
| RTC Get Date | √ | 25 | 04 | c |
| RTC Get Time | √ | 25 | 02 | c |
| RTC Set Date | √ | 25 | 03 | c |
| RTC Set Time | √ | 25 | 01 | c |
| Save/Delete Log | | 4C | 05 | |
| Set Data encryption Key Encryption Type | √ | C7 | 32 | |
| Set/Get Audio Volume | | 61 | 32 | |

| Command | C'less or C'less + MSR | CMD | SUB CMD | Notes |
|--|---------------------------|-----|---------|-------|
| Set Baud Rate | √ | 30 | 01 | |
| Set Bluetooth Parameters | | 77 | 81 | |
| Set Bluetooth Parameters | | 77 | 84 | |
| Set CA Public Key Protocol 1 | √ | 24 | 01 | |
| Set CA Public key Protocol 2 | √ | D0 | 03 | |
| Set Cable Type | √ | 32 | 02 | |
| Set Cancel Button On/Off | | 61 | 33 | |
| Set Card Keys | | C7 | 67 | |
| Set Configurable AID | √ | 04 | 02 | d |
| Set Configurable Group | √ | 04 | 03 | d |
| Set Configuration | √ | 04 | 00 | |
| Set Configuration Defaults and Keep Encryption Key | √ | 04 | 0A | |
| Set Configure Watchdog | | 57 | 00 | |
| Set Data Encryption Enable Flag | √ | C7 | 36 | |
| Set Data Encryption Key Variant Type | √ | C7 | 2F | |
| Set DUKPT DEK Attribution based on KeySlot | | C7 | A0 | |
| Set DUKPT PEK Attribution based on KeySlot | | C7 | A2 | |
| Set Ethernet SSL Request Configuration | | D2 | 01 | |
| Set Function/Hardware Enable/Disable | | 29 | 40 | |
| Set Key Press Event | | 01 | 0F | |
| Set KeySlot for DUKPT PEK and DUKPT DEK | | C7 | A4 | |
| Set KTK Private Key | | C7 | 6C | |
| Set Logger Level | | 4C | 02 | |
| Set Merchant Record | √ | 04 | 11 | |
| Set Model Number | | 90 | 15 | |
| Set MSR / Manual PAN Entry Configuration with TLV | | C7 | 80 | |
| Set MSR Secure Parameters | | C7 | 38 | |
| Set Parameter Defaults | | 04 | 09 | |
| Set Poll Mode | √ | 01 | 01 | |
| Set RF Error Reporting | √ | 17 | 03 | |
| Set Serial Number | √ | 12 | 02 | |
| Set Special Function or Feature Configuration | | C7 | 82 | |
| Set Switches Notifications Configuration | | 01 | 07 | |
| Set system language | | D1 | 27 | |
| Set/Get Source for RTC/LCD/Buzzer/LED | √ | 01 | 05 | |
| Set Temporary Baud Rate | √ | 30 | 02 | |
| Set USB Mode Convert and Attribution | | D1 | 30 | |
| Set USB Power Mode Suspend Configuration | | F0 | 08 | d |
| Stop Audio | | 61 | 30 | |
| Stop Transaction | √ | 05 | 02 | |
| Switches Notifications | | F2 | 00 | |
| Transfer File to SD Card | | 83 | 33 | |
| Turn On or Off Visual LEDs on LCD | | 61 | 1A | |
| Update Balance Command | √ | 03 | 03 | |
| Wake-Up Notification | | F1 | 00 | |
| Wi-Fi SSL Connection Request | | D1 | 1C | |

a Contact EMV products only

b Not in Global Reader Lite (GRL)

c Real Time Clock only

d Only applies to readers with USB

2.1.2 Commands Sorted by Command Number

Commands Sorted by Command Number

| CMD | SUB CMD | Command | C'less or C'less + MSR | Notes |
|-----|---------|--|------------------------|-------|
| 01 | 01 | Set Poll Mode | √ | |
| 01 | 02 | Control User Interface | | |
| 01 | 05 | Set/Get Source for RTC/LCD/Buzzer/LED | √ | |
| 01 | 07 | Set Switches Notifications Configuration | | |
| 01 | 08 | Get Switches Notifications Configuration | | |
| 01 | 0E | Set Key Press Event | | |
| 01 | 0F | Get Key Press Event | | |
| 02 | 01 | Activate Transaction Command | √ | |
| 02 | 09 | Data Exchange Response | √ | |
| 02 | 40 | Activate Transaction Command | √ | |
| 02 | 58 | Data Exchange Request | √ | |
| 03 | 00 | Get Transaction Result | √ | |
| 03 | 40 | Get Transaction Result | √ | |
| 03 | 02 | Get Configuration | √ | |
| 03 | 03 | Update Balance Command | √ | |
| 03 | 04 | Get Configurable AID | √ | a |
| 03 | 05 | Get All AIDs | √ | a |
| 03 | 06 | Get Configurable Group | √ | a |
| 03 | 07 | Get All Groups | √ | a |
| 03 | 11 | Get Merchant Record | √ | |
| 03 | 40 | Get Transaction Result | √ | |
| 04 | 00 | Set Configuration | √ | |
| 04 | 02 | Set Configurable AID | √ | a |
| 04 | 03 | Set Configurable Group | √ | a |
| 04 | 04 | Delete Configurable AID | √ | a |
| 04 | 05 | Delete Configurable Group | √ | a |
| 04 | 09 | Set Parameter Defaults | | |
| 04 | 0A | Set Configuration Defaults and Keep Encryption Key | √ | |
| 04 | 11 | Set Merchant Record | √ | |
| 04 | 22 | Front Removal Configuration | | |
| 05 | 01 | Cancel Transaction Command | √ | |
| 05 | 02 | Stop Transaction | √ | |
| 09 | 00 | Get ALL Reader Variables | √ | |
| 09 | 01 | Get Product Type | | |
| 09 | 02 | Get Processor Type | √ | |
| 09 | 03 | Get Main Firmware Version | √ | |
| 09 | 14 | Get Hardware Information | √ | |
| 09 | 15 | Get System Info | | |
| 09 | 20 | Get Module Version Information | √ | |
| 09 | 31 | Get External Chip Memory Usage Information | | |
| 0A | 02 | LED Control | √ | |
| 0B | 01 | Buzzer Control Short | √ | |
| 0B | 02 | Buzzer Control Long | √ | |
| 12 | 01 | Get Serial Number | √ | |
| 12 | 02 | Set Serial Number | √ | |

| CMD | SUB CMD | Command | C'less or C'less + MSR | Notes |
|-----|---------|--|------------------------|-------|
| 14 | 01 | Boot up Notification | √ | |
| 17 | 02 | Flush Track Data | √ | |
| 17 | 03 | Set RF Error Reporting | √ | |
| 17 | CD | Get Full Track Data | √ | |
| 18 | 01 | Ping | √ | |
| 24 | 01 | Set CA Public Key | √ | |
| 24 | 02 | Delete CA Public Key | √ | |
| 24 | 03 | Delete All CA Public Keys | √ | |
| 25 | 06 | Get RTC Date and Time | | |
| 25 | 07 | Get/Set 24-hr Self-Check Time | √ | b |
| 28 | 01 | Antenna Control | √ | |
| 29 | 00 | Get Version Protocol 2 | √ | |
| 29 | 00 | Get Firmware Full Version | √ | |
| 29 | 01 | Get product full information | | |
| 29 | 04 | Get USB Boot Loader Version | √ | c |
| 29 | 06 | Get Contact EMV L2 Kernel Version | √ | |
| 29 | 07 | Get Contact EMV L2 Kernel Version Detail | √ | |
| 29 | 08 | Get Contact EMV L2 Kernel Checksum | √ | |
| 29 | 09 | Get Contact EMV L2 Terminal Configuration Checksum | √ | |
| 29 | 17 | Get UID of MCU | √ | |
| 29 | 20 | Get TMS | | |
| 29 | 40 | Set Function/Hardware Enable/Disable | | |
| 29 | 41 | Get Function/Hardware Enable/Disable | | |
| 32 | 01 | Get Cable Type | √ | |
| 32 | 02 | Set Cable Type | √ | |
| 2C | 01 | Pass-Through Mode Start/Stop | √ | |
| 2C | 02 | Poll for Token | √ | |
| 2C | 03 | Exchange Contactless Data | √ | |
| 2C | 04 | PCD Single Command Exchange | √ | |
| 2C | 05 | Get PCD and PICC Parameters | √ | |
| 2C | 06 | Mifare Authenticate Block | √ | |
| 2C | 07 | Mifare Read Blocks | √ | |
| 2C | 08 | Mifare Write Blocks | √ | |
| 2C | 09 | High-level Halt Command | √ | |
| 2C | 0A | Mifare ePurse Command | √ | |
| 2C | 0B | Enhanced Pass-Through Command | √ | |
| 2C | 0C | Enhanced Poll for Token | √ | |
| 2C | 12 | Get ATR | √ | |
| 2C | 13 | Exchange APDU Data | √ | |
| 2C | 18 | Contact Transaction Power Off | | |
| 2C | 40 | NFC Commands | √ | |
| 2C | 41 | Felica Commands | √ | |
| 2C | 42 | Felica Lite/Lite-S Authentication | √ | |
| 2C | 43 | Felica Lite/Lite-S Read/Write Blocks with MAC | √ | |
| 2C | 44 | Pass-Through Package with Felica Card | | |
| 2C | 45 | Felica Get Code | | |
| 2C | 46 | Cancel Felica Get Code | | |
| 2C | 51 | Get Whitelist | √ | |

| CMD | SUB CMD | Command | C'less or C'less + MSR | Notes |
|-----|---------|--|------------------------|-------|
| 2C | 53 | Get Discretionary Data Whitelist | √ | |
| 2C | 20 | Check MSR Swipe Function | | |
| 30 | 01 | Set Baud Rate | √ | |
| 30 | 02 | Set Temporary Baud Rate | √ | |
| 4C | 01 | Initialize/De-initialize Logger | | |
| 4C | 02 | Set Logger Level | | |
| 4C | 03 | Get/Set Logger Size | | |
| 4C | 04 | Enable/Disable Logger | | |
| 4C | 05 | Save/Delete Log | | |
| 4C | 06 | Retrieve Log | | |
| 57 | 00 | Set Configure Watchdog | | |
| 57 | 01 | Read Configure Watchdog | | |
| 57 | 02 | Enable/Disable Watchdog | | |
| 57 | 03 | Read Watchdog Enable/Disable Information | | |
| 57 | 04 | Feed Watchdog Timer | | |
| 57 | 05 | Erase Watchdog Reset Counter and Log | | |
| 57 | 06 | Read Watchdog Reset Counter and Log | | |
| 60 | 01 | Contact Retrieve Application Data | | |
| 60 | 02 | Contact Remove Application Data | | |
| 60 | 03 | Contact Set Application Data | | |
| 60 | 04 | Contact Retrieve Terminal Data | | |
| 60 | 05 | Contact Remove Terminal Data | | |
| 60 | 06 | Contact Set Terminal Data | | |
| 60 | 07 | Contact Retrieve AID List | | |
| 60 | 08 | Contact Retrieve CA Public Key | | |
| 60 | 09 | Contact Remove CA Public Key | | |
| 60 | 0A | Contact Set CA Public Key | | |
| 60 | 0B | Contact Retrieve CA Public Key List | | |
| 60 | 0C | Contact Retrieve Certification Revocation List | | |
| 60 | 0D | Contact Remove Certification Revocation List | | |
| 60 | 0E | Contact Set Certification Revocation List | | |
| 60 | 0F | Contact Remove Transaction Amount Log | | |
| 60 | 10 | Contact Start Transaction | | |
| 60 | 11 | Contact Authenticate Transaction | | |
| 60 | 12 | Contact Apply Host Response | | |
| 60 | 13 | Contact Retrieve Transaction Result | | |
| 60 | 14 | Get Contact Reader Status | | |
| 60 | 15 | Contact Get ICS Identification | | |
| 60 | 16 | Contact Set ICS Identification | | |
| 60 | 19 | Motor Eject Card | | |
| 60 | 1A | Motor Clean | | |
| 61 | 01 | Contact LCD Display Control (Reader send to Host) | | |
| 61 | 02 | Contact Get PIN Control (Reader send to Host) | | |
| 61 | 03 | Contact Get MSR Data Control (Reader send to Host) | | |
| 61 | 04 | Key Press Event (Reader send to Host) | | |
| 61 | 18 | Add Object to Scroll Container | | |
| 61 | 19 | Get List of All Objects on Scrollable Container | | |
| 61 | 1A | Turn On or Off Visual LEDs on LCD | | |

| CMD | SUB CMD | Command | C'less or C'less + MSR | Notes |
|-----|---------|---|------------------------|-------|
| 61 | 1B | Link UI with Transaction Message ID | | |
| 61 | 30 | Stop Audio | | |
| 61 | 31 | Play Audio | | |
| 61 | 32 | Set/Get Audio Volume | | |
| 61 | 33 | Set Cancel Button On/Off | | |
| 61 | 34 | Get Cancel Button On/Off | | |
| 61 | 41 | QR Code Scan Start | | |
| 61 | 42 | QR Code Scan Stop | | |
| 61 | 43 | Camera Take Photo Start | | |
| 61 | 44 | Camera Take Photo Stop | | |
| 61 | 45 | Camera Take Photo Once | | |
| 62 | 00 | Cancel PIN or Key Entry | | |
| 62 | 01 | Display Message and Get Encrypted PIN (Host send to Reader) | | |
| 62 | 02 | Get Function Key (Host send to Reader) | | |
| 62 | 03 | Display Message and Get Numeric Key (Host send to Reader) | | |
| 62 | 04 | Display Message and Get Amount (Host send to Reader) | | |
| 62 | 05 | Display Multi Line Message and Get Numeric Key | | |
| 62 | 06 | Display Specific Message and Get Numeric Key | | |
| 77 | 81 | Set Bluetooth Parameters | | |
| 77 | 82 | Get Bluetooth Name | | |
| 77 | 83 | Get Bluetooth Local Information | | |
| 77 | 84 | Set Bluetooth Parameters | | |
| 81 | 02 | Check DUKPT Keys | √ | |
| 81 | 04 | Check DUKPT Key | √ | |
| 81 | 0A | Get DUKPT Key Serial Number (KSN) | √ | |
| 81 | 0B | Get DUKPT Key Serial Number Extended | | |
| 81 | 40 | Get and Set MAC Verification Option (for data output) | | |
| 83 | 30` | Enable/Disable Light Sensor | | |
| 83 | 33 | Transfer File to SD Card | | |
| 83 | 34 | Delete File from SD Card | | |
| 83 | 35 | List Directory on SD Card | | |
| 83 | 36 | Query File on SD Card | | |
| 83 | 37 | Get SD Card State | | |
| 83 | 39 | Read Out File from SD Card | | |
| 84 | 03 | Get EMV Revocation Log Status | √ | |
| 84 | 04 | Add Entry to EMV Revocation List | √ | |
| 84 | 05 | Delete All Entries for Single Index in EMV Revocation List | √ | |
| 84 | 06 | Delete All Entries from EMV Revocation List | √ | |
| 84 | 07 | Get EMV Revocation List | √ | |
| 84 | 08 | Get EMV Exception Log Status | √ | |
| 84 | 09 | Add Entry to EMV Exception List | √ | |
| 84 | 0A | Delete Entry from EMV Exception List | √ | |
| 84 | 0B | Delete All Entries from EMV Exception List | √ | |
| 84 | 0C | Get EMV Exception List | √ | |
| 84 | 0D | Delete an Entry from EMV Revocation List | √ | |
| 84 | 0E | Reset Torn Transaction Log | √ | |
| 84 | 0F | Clean Torn Transaction Log | √ | |
| 90 | 15 | Set Model Number | | |

| CMD | SUB CMD | Command | C'less or C'less + MSR | Notes |
|-----|---------|---|------------------------|-------|
| C7 | A0 | Set DUKPT PEK Attribution based on KeySlot | | |
| C7 | A1 | Get DUKPT PEK Attribution based on KeySlot | | |
| C7 | A2 | Set DUKPT DEK Attribution based on KeySlot | | |
| C7 | A3 | Get DUKPT DEK Attribution based on KeySlot | | |
| C7 | A4 | Set KeySlot for DUKPT PEK and DUKPT DEK | | |
| C7 | A5 | Get KeySlot for DUKPT PEK and DUKPT DEK | | |
| C7 | 2F | Set Data Encryption Key Variant Type | √ | |
| C7 | 30 | Get Data Encryption Key Variant Type | √ | |
| C7 | 32 | Set Data encryption Key Encryption Type | √ | |
| C7 | 33 | Get Data encryption Key Encryption Type | √ | |
| C7 | 36 | Set Data Encryption Enable Flag | √ | |
| C7 | 37 | Get Data Encryption Enable Flag | √ | |
| C7 | 38 | Set MSR Secure Parameters | | |
| C7 | 39 | Get MSR Secure Parameters | | |
| C7 | 3F | Get TM4 Firmware Version Information | | |
| C7 | 67 | Set Card Keys | | |
| C7 | 69 | Delete Keys | | |
| C7 | 6A | Get Hash of Keys | | |
| C7 | 6B | Get the Next DEK KSN | | |
| C7 | 6C | Set KTK Private Key | | |
| C7 | 9A | Peer To Peer Send A Message | √ | |
| C7 | 9B | Peer To Peer Receive A Message | √ | |
| C7 | 80 | Set MSR / Manual PAN Entry Configuration with TLV | | |
| C7 | 81 | Get MSR / Manual PAN Entry Configuration with TLV | | |
| C7 | 82 | Set Special Customer Configuration | | |
| C7 | 83 | Get Special Customer Configuration | | |
| C7 | C7 | ADF Application Control Command | | |
| D0 | 01 | Get CA Public Key | √ | |
| D0 | 02 | Get CA Public Key Hash | √ | |
| D0 | 03 | Set CA Public Key | √ | |
| D0 | 04 | Delete CA Public Key | √ | |
| D0 | 05 | Delete All CA Public Keys | √ | |
| D0 | 06 | Get All CA Public RIDs | √ | |
| D0 | 07 | List CA Public Key IDs or RID Protocol 2 | √ | |
| D1 | 27 | Set System Language | | |
| D1 | 1C | Wi-Fi SSL Connection Request | | |
| D1 | 30 | Set USB Mode Convert and Attribution | | |
| D1 | 31 | Get USB Mode Convert and Attribution | | |
| D2 | 01 | Set Ethernet SSL Request Configuration | | |
| D2 | 02 | Get Ethernet SSL Request Configuration | | |
| D2 | 03 | Enable/Disable Communication Port | | |
| F0 | 00 | Set PMC Status | √ | |
| F0 | 01 | Get PMC Status | √ | |
| F0 | 08 | Set USB Power Mode Suspend Configuration | | c |
| F0 | 09 | Get USB Power Mode Suspend Configuration | | c |
| F0 | 0F | Shut Off the Power | √ | |
| F1 | 00 | Wake-Up Notification | | |
| F2 | 00 | Switches Notification | | |

- a Not in Global Reader Lite (GRL)
- b Real Time Clock only
- e Only applies to readers with USB

VP6300-specific Commands

| Cmd | Description |
|-----------------|---|
| F6 F6 F6 | Disable Blue Attraction LEDs to indicate reader is disabled |
| F7 F7 F7 | Enable Blue Attraction LEDs to indicate reader is enabled |
| F9 F9 F9 | Clear LCD display and returns cursor to home position |
| FA FA FA | Disable Yellow LED to indicate vend session is NOT active |
| FB FB FB | Enable Flashing of Yellow END Button LED to indicate vend session is active |
| FE FE FE | BEEP Buzzer one time |
| 80 | LCD Line 1 |
| C0 | LCD Line 2 |
| 37 32 04 | Model Request |
| 37 30 04 | Serial Number Request |

2.1.3 Pass-Through Command Table**Pass-Through Command Table**

| Command | C'less or C'less + MSR | LCD | Line | US | EMV | Protocol | CMD | SUB CMD |
|---|---------------------------|-----|------|----|-----|----------|-----|------------|
| Antenna Control | √ | √ | √ | √ | √ | 2 | 28 | 01 |
| Buzzer Control Long | √ | √ | √ | √ | √ | 2 | 0B | 02 |
| Buzzer Control Short | √ | √ | √ | √ | √ | 2 | 0B | 01 |
| Enhanced Pass-Through Command | √ | √ | √ | √ | √ | 2 | 2C | 0B |
| Enhanced Poll for Token | √ | √ | √ | √ | √ | 2 | 2C | 0C |
| Exchange Contactless Data | √ | √ | √ | √ | √ | 2 | 2C | 03 |
| Get PCD and PICC Parameters | √ | √ | √ | √ | √ | 2 | 2C | 05 |
| High-level Halt Command | √ | √ | √ | √ | √ | 2 | 2C | 09 |
| LED Control | √ | √ | √ | √ | √ | 2 | 0A | 02 |
| Mifare Authenticate Block | √ | √ | √ | √ | √ | 2 | 2C | 06 |
| Mifare ePurse Command | √ | √ | √ | √ | √ | 2 | 2C | 0A |
| Mifare Read Blocks | √ | √ | √ | √ | √ | 2 | 2C | 07 |
| Mifare Write Blocks | √ | √ | √ | √ | √ | 2 | 2C | 08 |
| Pass-Through Mode Start/Stop | √ | √ | √ | √ | √ | 2 | 2C | 01 |
| PCD Single Command Exchange | √ | √ | √ | √ | √ | 2 | 2C | 04 |
| Poll for Token | √ | √ | √ | √ | √ | 2 | 2C | 02 |
| Felica Commands | √ | √ | √ | √ | √ | 2 | 2C | 41 |
| Felica Lite/Lite-S Authentication | √ | √ | √ | √ | √ | 2 | 2C | 42 |
| Felica Lite/Lite-S Read/Write Blocks with MAC | √ | √ | √ | √ | √ | 2 | 2C | 43 |
| Get Whitelist | √ | √ | √ | √ | √ | 2 | 2C | 51 |
| Get Discretionary Data Whitelist | √ | √ | √ | √ | √ | 2 | 2C | 53 |

2.1.4 EMV Key Manager Command Tables

The preferred method of accessing the Certificate Authority public keys is to use the following commands.

EMV Key Management – Protocol 2

| Command | C'less or C'less + MSR | LCD | Line | US | EMV | Protocol | CMD | SUB CMD | Notes |
|--------------------------------|---------------------------|-----|------|----|-----|----------|-----|------------|-------|
| Get CA Public Key | √ | √ | √ | | √ | 2 | D0 | 01 | |
| Get CA Public Key Hash | √ | √ | √ | | √ | 2 | D0 | 02 | |
| Set CA Public Key | √ | √ | √ | | √ | 2 | D0 | 03 | |
| Delete CA Public Key | √ | √ | √ | | √ | 2 | D0 | 04 | |
| Delete All CA Public Keys | √ | √ | √ | | √ | 2 | D0 | 05 | |
| Get All CA Public RIDs | √ | √ | √ | | √ | 2 | D0 | 06 | |
| List CA Public Key IDs for RID | √ | √ | √ | | √ | 2 | D0 | 07 | |

2.1.5 Status Codes

Status Codes for Protocol 2

| Status Code | Status |
|---|--|
| 00h | OK |
| 01h | Incorrect Header Tag |
| 02h | Unknown Command |
| 03h | Unknown Sub-Command |
| 04h | CRC Error in Frame |
| 05h | Incorrect Parameter |
| 06h | Parameter Not Supported |
| 07h | Mal-formatted Data |
| 08h | Timeout |
| 0Ah | Failed / NACK |
| 0Bh | Command not Allowed |
| 0Ch | Sub-Command not Allowed |
| 0Dh | Buffer Overflow (Data Length too large for reader buffer) |
| 0Eh | User Interface Event |
| 10h | Need clear firmware (apply in boot loader only) |
| 11h | Communication type not supported, VT-1, burst, and so on. |
| | Need encrypted firmware (apply in boot loader only) |
| 12h | Secure interface is not functional or is in an intermediate state. |
| 13h | Data field is not mod 8 |
| 14h | Pad 0x80 not found where expected |
| 15h | Specified key type is invalid |
| 16h | Could not retrieve key from the SAM (InitSecureComm) |
| 17h | Hash code problem |
| 18h | Could not store the key into the SAM (InstallKey) |
| 19h | Frame is too large |
| 1Ah | Unit powered up in authentication state but POS must resend the InitSecureComm command |
| 1Bh | The EEPROM may not be initialized because SecCommInterface does not make sense |
| 1Ch | Problem encoding APDU |
| 1Dh | Tamper occurred |
| 1Eh | Card not removed |
| Module-Specific Status Codes ¹ | |
| 20h | Unsupported Index (ILM) SAM Transceiver error – problem communicating with the SAM (Key Mgr) |
| 21h | Unexpected Sequence Counter in multiple frames for single bitmap (ILM) Length error in data returned from the SAM (Key Mgr) |
| 22h | Improper bit map (ILM) |
| 23h | Request Online Authorization |
| 24h | ViVOCard3 raw data read successful |
| 25h | Message index not available (ILM) ViVOcomm activate transaction card type (ViVOcomm) |

¹ Status codes in this range are "module-specific" so their values can be re-used by different modules. The meaning of these codes may depend on which command is being issued. An exception is 23h, which is used generally.

| Status Code | Status |
|-------------|---|
| 26h | Version Information Mismatch (ILM) |
| 27h | Not sending commands in correct index message index (ILM) |
| 28h | Time out or next expected message not received (ILM) |
| 29h | ILM languages not available for viewing (ILM) |
| 2Ah | Other language not supported (ILM) |
| 2Bh | Device is not ready state |
| 2Ch | Command Cancel Operation |
| 2Dh | Press Key Cancel Operation |
| 2Eh | Removal Detection is disabled |
| 30h | No SCRP Default Pairing Key |
| 31h | Status Error |
| 32h | SCRP Pairing Key Error |
| 33h | CVMApp Enc RSA Key Error |
| 34h | CVMApp Auth RSA Key Error |
| 35h | CVMApp Token RSA Key Error |
| 36h | SCRP Get Random Failed Or RNG Error |
| 37h | CVMApp RSA Key Sign Data Error Or Verify Signature Error |
| 38h | CVMApp RSA Key Encrypt Data Error |
| 39h | Data Shared Key Failed |
| 3Ah | PIN Shared Key Failed |
| 3Bh | Safety Access Not Allow (Verify Signature Error Times Overflow) |
| 3Ch | No Refresh Token – Don't implement transaction |
| 41h – 4Fh | Module-specific errors for Key Manager |
| 50h | Auto-Switch OK |
| 51h | Auto-Switch failed |
| 57h | No payment occurred (VAS transaction) |
| 5Ah | VAS container missing for VAS-only transaction. |
| 60h | Data not exist |
| 61h | Data Full |
| 62h | Write Flash Error |
| 63h | Ok and have next command |
| 64h | One Kernel was or some Kernels were disabled for absent or self-test error |
| 70h | Antenna Error |
| 80h | Use another card |
| 81h | Insert or swipe card |
| 90h | Data encryption Key does not exist |
| 91h | Data encryption Key KSN exhausted |
| 92h | MAC DUKPT Key does not exist (if enable MAC verification) |
| 93h | MAC DUKPT Key KSN exhausted (if enable MAC verification) |
| 94h | No DEK / Desjardins Data Key / TransArmor RSA Cert |
| 95h | Encryption Switch is Off |
| A0h | The screen or object already exists |
| A1h | The SD card does NOT exist |
| A2h | Error decoding QR code |
| A3h | Error code for card insertion |
| A4h | Error code for card insertion before sending a transaction command (for products with motors) |
| A5h | LCD control is rejected, due to LCD is not in working state |

2.2 Error Codes

Error Codes

| Error Code | Description | Reason for Error and Suggested Error Handling |
|------------|----------------------------|--|
| 00h | No Error | None. |
| 01h | Out of Sequence Command | Reader did not receive commands in the correct order. Correct the host application to send serial commands in the correct sequence. |
| 02h | Go to Contact Interface | <p>The contactless transaction failed.</p> <p><i>If the reader supports a contact interface, advise the user to complete the transaction on the contact interface.</i></p> <p>Previously, this error code was used if the reader supported another interface besides the contact interface. Integrators should use error code 04h Go to Other Interface instead. The previous use has been deprecated.</p> |
| 03h | Transaction Amount is Zero | If the transaction amount is zero and the host device is "an offline only terminal," the reader needs to terminate the transaction. |
| 04h | Go To Other Interface | <p>The transaction has failed.</p> <p><i>If the reader supports another interface, advise the user to complete the transaction on the other interface.</i></p> |
| 05h | Go To Nearby Interface | <p>The transaction has failed.</p> <p><i>If there is another nearby contact interface, advise the user to complete the transaction on the nearby contact interface. This situation might be a case where there are multiple pay stations but only one of them has a contact interface.</i></p> |
| 06h | Go To MagStripe Interface | <p>The transaction has failed.</p> <p><i>If the reader has a magstripe interface, advise the user to complete the transaction using the magstripe interface.</i></p> |
| 08h | Transaction Timeout | The transaction timed out. |
| 20h | Card returned Error Status | <p>Card returned SW1SW2 not equal to 9000 hex. The value of the SW1SW2 bytes from the card is returned in the data portion of the response frame. Details of what the SW1SW2 codes mean for each RF State are card-dependent and are out of the scope of this document.</p> <p>How the host device handles this error depends on when the error occurs in the transaction flow. The RF State Code indicates the transaction state when the error occurred. Suggested error handling for individual RF State Codes is given below:</p> <p>RF State Code = PPSE: If RF State Code = SELECT: If RF State Code = GPO: If RF State Code = READ RECORD: If RF State Code = GET DATA (Ticket): If RF State Code = GET DATA (Ticketing Profile): If RF State Code = GET DATA (Balance):</p> |

| Error Code | Description | Reason for Error and Suggested Error Handling |
|------------|------------------------------|--|
| | | <p>If RF State Code = PUT DATA (Ticket): The host device can retry the transaction or abandon it.</p> <p>If RF State Code = GEN AC: For credit transactions: The host device can retry the transaction or abandon it.</p> |
| 21h | Collision Error | There was more than one contactless transaction in the reader's range. |
| 22h | Amount Over Maximum Limit | The transaction amount is greater than host device contactless transaction limit (FFF1). |
| 23h | Request Online Authorization | If the transaction amount is greater than the balance on the card but is less than the host device contactless transaction limit (FFF1), the reader sends this error code back to the host device along with other information needed by the acquirer to format an online authorization request. |
| 24h | Card Communication Error | A communication error occurred while interacting with the card. An example might be the card was removed from the field. |
| 25h | Card Blocked | This error code is sent to the host device if the reader does not support the card according to the value of parameter application capability (FFF3). |
| 26h | Card Expired | This error code is sent to the host device if the current date of the reader is greater than the expiration date of the card. This status code is only valid for qVSDC cards. |
| 27h | Unsupported Card | Card presented to the reader is of a type not supported by the reader. This could be due to presenting a card with an AID that is not recognized by the reader. |
| 2Ch | Operation Exited | The 8341 operation was exited by the Cancel Transaction command. |
| 2Dh | Operation Exited | he 8341 operation was exited by the user pressing the cancel key. |

| Error Code | Description | Reason for Error and Suggested Error Handling |
|------------|---|--|
| 30h | Card did not respond | <p>Card was removed from the field or there was a communication error preventing the card response from reaching the reader. How the host device handles this error depends on when the error occurred in the transaction. The RF State Code indicates the transaction state when the error occurred. Suggested error handling for each RF State Code is given below:</p> <p>RF State Code = PPSE: The host device can retry the transaction or abandon it.</p> <p>If RF State Code = SELECT: The host device can retry the transaction or abandon it.</p> <p>If RF State Code = GPO: The host device can retry the transaction or abandon it.</p> <p>If RF State Code = READ RECORD: The host device can retry the transaction or abandon it.</p> <p>If RF State Code = GEN AC: For Credit transactions: The host device can retry the transaction or abandon it.</p> <p>If RF State Code = GET DATA (Ticket): If RF State Code = GET DATA (Ticketing Profile): If RF State Code = GET DATA (Balance): If RF State Code = PUT DATA (Ticket):</p> |
| 40h | Unknown Data Element Error | The data element is unknown. |
| 41h | Data Element Missing | A mandatory/required data element was missing from the card. |
| 42h | Card Generated AAC | The card declined the transaction by sending an AAC instead of a TC. |
| 43h | Card Generated ARQC | This error code is returned if the card generated an ARQC and the host/reader was configured as "Offline Only"; therefore the card was DECLINED. |
| 44h | SDA/DDA Failed (Not Supported by Card) | Card did not indicate support for the correct authentication method and date authentication failed. For Visa, when DDA is required, the card must indicate support for DDA in AIP. If this support is not indicated then the transaction fails and this error code is returned. |
| 50h | SDA/DDA/CDDA Failed (CA Public Key) | Data authentication failed due to missing CA public key. Retrying the transaction does not correct the error until the missing CA public key problem is corrected via key management commands. |
| 51h | SDA/DDA/CDDA Failed (Issuer Public Key) | <p>Data authentication failed due to a problem in recovering the issuer public key from the card data. Data on the card may be incorrect or the reader has the wrong CA public key.</p> <p>The transaction continues to fail until the issuer public key and the CA public key are correct.</p> |
| 52h | SDA Failed (SSAD) | Data authentication failed during SSAD. Retrying the transaction does not correct the error. |
| 53h | DDA/CDDA Failed (ICC Public Key) | Data authentication failed during attempted recovery of ICC public key. Retrying the transaction does not correct the error. |

| Error Code | Description | Reason for Error and Suggested Error Handling |
|------------|--|--|
| 54h | DDA/CDDA Failed (Dynamic Signature Verification) | Data authentication failed during dynamic signature verification. Retrying the transaction does not correct the error. At this point, the amount has been deducted from the card balance. |
| 55h | Processing Restrictions Failed | The processing restrictions step as defined in EMV specifications failed. This could be due to incorrectly set configuration. Retrying the transaction does not correct the error until the EMV configuration is corrected. |
| 56h | Terminal Risk Management (TRM) Failed | The terminal risk management step as defined in EMV specifications failed. This could be due to incorrectly set configuration. Retrying the transaction does not correct the error until the EMV configuration is corrected. |
| 57h | Cardholder Verification Failed | The cardholder verification step as defined in EMV specifications failed. This could be due to incorrectly set configuration. Retrying the transaction does not correct the error until the EMV configuration is corrected. |
| 58h | Terminal Action Analysis (TAA) Failed | The terminal action analysis step as defined in EMV specifications failed. This could be due to incorrectly set configuration. Retrying the transaction does not correct the error until the EMV configuration is corrected. |
| 59h | Card Expired (Interac) | This error code is sent to the host if the current date of the reader is greater than the expiration date of the card. This status code is only valid for qVSDC cards. (Terminated) |
| 5Ah | Cardholder Verification Failed (Interac) | The cardholder verification step as defined in EMV specifications failed. This could be due to incorrectly set configuration. Retrying the transaction does not correct the error until the EMV configuration is corrected. (Terminated) |
| 61h | SD Memory Error | This error is reported only when trying to retrieve transaction logs. This error is never reported during a transaction. |
| 62h | Generic Error | This is a generic or general error that is reported when a more specific reason for the error is not known. |
| 63h | Card returned Data Element Error | Returned card data element is invalid. |
| 73h | Torn Transaction Log Error | An error occurred while attempting to clean the torn transaction log. This might occur if the reader could not read the time and date from the real time clock. |
| 80h | No Merchants have been configured | This error usually occurs while MerchantID is empty. |
| 81h | TLV Parse Failure | This error usually occurs when failing to TLV parsing card response data. |
| 82h | Merchant Data Error | This error usually occurs when no merchant data is returned from the card. |
| 83h | System Memory Error | This error usually occurs when failing to read or write system memory. |
| 84h | Application Skip Error | This error usually occurs when configuration isn't consistent on whether or not to skip payment application. |
| 85h | Application Version Error | This error usually occurs when the application version number is incorrect. |
| 8Bh | LTPK Error | This error code indicates the LTPK (long term private key) hasn't been loaded. |

If an error occurs during a transaction and the host determines that the reader must perform exception processing, then the host must retry the transaction until the transaction has been completed successfully or the host decides to abort. The retries must be continued even if successive transactions fail with conditions that do not require exception processing. This must be done to allow the reader to complete exception processing (even if there are failures during exception processing).

Under certain conditions, such as when a customer walks away or there is a problem with the card, the host may want to abort the retries even if the reader has not been able to complete exception processing. How and when the host stops retrying is out of the scope of this document.

2.3 RF State Codes

For some Error Codes, the RF State Code indicates the exact reader-card command that failed. This helps determine the exact place where the failure occurred.

RF State Codes

| State Code | RF State | Description |
|------------|---------------------------|--|
| 00h | None | RF State Code not available |
| 01h | PPSE | Error occurred during PPSE command |
| 02h | SELECT | Error occurred during SELECT command |
| 03h | GPO | Error occurred during GET PROCESSING OPTIONS command |
| 04h | READ RECORD | Error occurred during READ RECORD command |
| 05h | GEN AC | Error occurred during GEN AC command |
| 06h | CCC | Error occurred during CCC command |
| 07h | IA | Error occurred during IA command |
| 08h | SDA | Error occurred during SDA processing |
| 09h | DDA | Error occurred during DDA processing |
| 0ah | CDA | Error occurred during CDA processing |
| 0bh | TAA | Error occurred during TAA processing |
| 0ch | UPDATE RECORD | Error occurred during UPDATE RECORD command |
| 0eh | NEGOTIATE | Error occurred during a Smart Tap transaction |
| 10h | GET DATA (Ticket) | Error occurred during GET DATA command to retrieve the ticket |
| 11h | GET DATA (Ticketing Prof) | Error occurred during GET DATA command to retrieve the ticketing profile |
| 12h | GET DATA (Balance) | Error occurred during GET DATA command to retrieve the balance |

| State Code | RF State | Description |
|------------|---------------------------|---|
| 13h | GET DATA (All) | Error occurred during GET DATA command to retrieve all data |
| 20h | PUT DATA (Ticket) | Error occurred during PUT DATA command to retrieve the ticket |
| 21h | CARD READ COMPLETE | Reader-card command response status |
| 22h | PROCESSING RESTRICTION | Reader-card command response status |
| 23h | OFFLINE DATA AUTHENTICATE | Reader-card command response status |
| 24h | CARDHOLDER VERIFICATION | Reader-card command response status |
| 25h | ONLINE PROCESSING | Reader-card command response status |
| 26h | COMPLETION | Reader-card command response status |
| 30h | OSE | Error occurred during OSE command |

2.4 EMV Result Codes (Tag DFEE25)

For contact EMV transactions, a two-byte result code may be returned in DFEE25. Check the contents of tag DFEE25, when it occurs, and interpret the results as follows.

NOTE: The result code is always two bytes long. The bottom two bits of the first byte are flags. If the zero bit is set, it means there was "advice." If the 1 bit was set, it means there was a "reversal." For example, 0203 means there was a reversal (02) and the result was 03 (declined).

EMV Result Codes

| Result | Code | Description |
|--|--------|--|
| EMV_RESULT_CODE_APPROVED_OFFLINE | 0 | Offline approved |
| EMV_RESULT_CODE_DECLINED_OFFLINE | 1 | Offline declined |
| EMV_RESULT_CODE_APPROVED | 2 | Online approved |
| EMV_RESULT_CODE_DECLINED | 3 | Online declined |
| EMV_RESULT_CODE_GO_ONLINE | 4 | Transaction will be remotely authorized |
| EMV_RESULT_CODE_NOT_ACCEPTED | 6 | The application cannot be selected |
| EMV_RESULT_CODE_FALLBACK_TO_MSR | 7 | Return MSR data from fallback |
| EMV_RESULT_CODE_TIMEOUT | 8 | Timed out waiting for next command (authentication or complete) |
| EMV_RESULT_CODE_GO_ONLINE_CTLS | 9 | |
| EMV_RESULT_CODE_AUTHENTICATE_TRANSACTION | 0x0010 | First ICC transaction command (Start Transaction) complete and waiting for authentication command |
| EMV_RESULT_CODE_TRANSACTION_CANCELED | 0x0012 | |
| EMV_RESULT_CODE_CARD_IN_NOTIFICATION | 0x0013 | |
| EMV_RESULT_CODE_SWIPE_NON_ICC | 0x11 | Return MSR data from direct swipe on transaction commands |
| EMV_RESULT_CODE_CTLS_TWO_CARDS | 0x7A | |
| EMV_RESULT_CODE_CTLS_TERMINATE | 0x7E | |
| EMV_RESULT_CODE_CTLS_TERMINATE_TRY_ANOTHER | 0x7D | |
| EMV_RESULT_CODE_UNABLE_TO_REACH_HOST | 0xFF | |
| EMV_RESULT_CODE_FILE_ARG_INVALID | 0x1001 | Transaction canceled by cancel command |
| EMV_RESULT_CODE_MEMORY_NOT_ENOUGH | 0x2001 | Failed to retrieve host data after transaction started |
| EMV_RESULT_CODE_SMARTCARD_OK | 0x3001 | |
| EMV_RESULT_CODE_SMARTCARD_FAIL | 0x3002 | |
| EMV_RESULT_CODE_SMARTCARD_INIT_FAILED | 0x3003 | |
| EMV_RESULT_CODE_FALLBACK_SITUATION | 0x3004 | The transaction would fall back to MSR, but the command parameter set to "Do not support fallback" |
| EMV_RESULT_CODE_SMARTCARD_ABSENT | 0x3005 | |
| EMV_RESULT_CODE_SMARTCARD_TIMEOUT | 0x3006 | |
| EMV_RESULT_CODE_MSR_CARD_ERROR | 0x3007 | |
| EMV_RESULT_CODE_MSR_CARD_ERROR_FALLBACK | 0x3012 | |
| EMV_RESULT_CODE_TIMEOUT_FOR_WAITING_ICC_INSERT_OR_MSR_SWIPE_FALLBACK | 0x3013 | |
| EMV_RESULT_CODE_CARD_INSERTION_FAILED | 0x3014 | VP5300M-specific error code. When the motor detects an operation error, it replies with this error code. |
| EMV_RESULT_CODE_PARSING_TAGS_FAILED | 0x5001 | Failed to parse tags |

| Result | Code | Description |
|--|--------|--|
| EMV_RESULT_CODE_CARD_DATA_ELEMENT_DUPLICATE | 0x5002 | Tag duplicated from TLV string |
| EMV_RESULT_CODE_DATA_FORMAT_INCORRECT | 0x5003 | Incorrect data format for tag |
| EMV_RESULT_CODE_APP_NO_TERM | 0x5004 | Failed to retrieve host data and application data |
| EMV_RESULT_CODE_APP_NO_MATCHING | 0x5005 | Host device does not support applications in card |
| EMV_RESULT_CODE_MANDATORY_OBJECT_MISSING | 0x5006 | Missing mandatory data from ICCR-APDU |
| EMV_RESULT_CODE_CARD_REJECTED | 0x5009 | EMV transaction blocked |
| EMV_RESULT_CODE_AIP_NOT_RECEIVED | 0x5010 | Application Interchange Profile(AIP) not received |
| EMV_RESULT_CODE_AFL_NOT_RECEIVED | 0x5011 | File Locator (AFL) not received |
| EMV_RESULT_CODE_AFL_LEN_OUT_OF_RANGE | 0x5012 | Length of AFL less than 4 bytes |
| EMV_RESULT_CODE_SFI_OUT_OF_RANGE | 0x5013 | Out of range of SFI: 0..31 |
| EMV_RESULT_CODE_AFL_INCORRECT | 0x5014 | AFL data incorrect |
| EMV_RESULT_CODE_EXP_DATE_INCORRECT | 0x5015 | Expiration date out of range |
| EMV_RESULT_CODE_EFF_DATE_INCORRECT | 0x5016 | Effective Date out of range |
| EMV_RESULT_CODE_ISS_COD_TBL_OUT_OF_RANGE | 0x5017 | |
| EMV_RESULT_CODE_CRYPTOGAM_TYPE_INCORRECT | 0x5018 | GAC (Generate AC) response data incorrect |
| EMV_RESULT_CODE_SERVICE_NOT_ALLOWED | 0x5021 | Service not allowed in GAC response |
| EMV_RESULT_CODE_CARD_BLOCKED | 0x5023 | Card blocked |
| EMV_RESULT_CODE_LEN_INCORRECT | 0x5024 | Incorrect length of R-APDU from card |
| EMV_RESULT_CODE_CARD_COM_ERROR | 0x5025 | Incorrect communication with L1 |
| EMV_RESULT_CODE_TSC_NOT_INCREASED | 0x5026 | |
| EMV_RESULT_CODE_HASH_INCORRECT | 0x5027 | |
| EMV_RESULT_CODE_ARC_NOT_PRESENCE | 0x5028 | No ARC in transaction completion command in which the ONLINE FLAG indicated no error |
| EMV_RESULT_CODE_TRAN_TYPE_INCORRECT | 0x5031 | Transaction type not recognized |
| EMV_RESULT_CODE_APP_NO_SUPPORT | 0x5032 | |
| EMV_RESULT_CODE_APP_NOT_SELECT | 0x5033 | |
| EMV_RESULT_CODE_LANG_NOT_SELECT | 0x5034 | |
| EMV_RESULT_CODE_TERM_DATA_NOT_PRESENCE | 0x5035 | |
| EMV_RESULT_CODE_APP_BLOCK_AID | 0x5036 | The selected application blocked |
| EMV_RESULT_CODE_PAN_INVALID | 0x5037 | PAN data in ICC incorrect |
| EMV_RESULT_CODE_TRAN_TYPE_NOT_SUPPORT | 0x5038 | Transaction type not supported |
| EMV_RESULT_CODE_PIN_ENTRY_TIMEOUT | 0x5039 | |
| EMV_RESULT_CODE_CVM_TYPE_UNKNOWN | 0x6001 | |
| EMV_RESULT_CODE_CVM_AIP_NOT_SUPPORTED | 0x6002 | Transaction should complete if 8E missing |
| EMV_RESULT_CODE_CVM_TAG_8E_FORMAT_ERROR | 0x6004 | The format of CVM list incorrect |
| EMV_RESULT_CODE_CVM_CODE_IS_NOT_SUPPORTED | 0x6005 | |
| EMV_RESULT_CODE_CVM_COND_CODE_IS_NOT_SUPPORTED | 0x6006 | |
| EMV_RESULT_CODE_CVM_NO_MORE | 0x6007 | |
| EMV_RESULT_CODE_PIN_BYPASSED_BEFORE | 0x6008 | |
| EMV_RESULT_CODE_UNKONWN | 0xFFFF | |
| RETURN_CODE_DO_SUCCESS | 0 | No error, starting task |
| RETURN_CODE_ERR_DISCONNECT | | No response from reader |
| RETURN_CODE_ERR_CMD_RESPONSE | | Invalid response data |
| RETURN_CODE_ERR_TIMEOUT | | Task or CMD timed out |
| RETURN_CODE_ERR_INVALID_PARAMETER | | Wrong parameter |
| RETURN_CODE_SDK_BUSY_MSR | | SDK doing MSR or ICC task |
| RETURN_CODE_SDK_BUSY_PINPAD | | SDK doing PINPad task |
| RETURN_CODE_SDK_BUSY_CTLS | | SDK doing CTLS task |

| Result | Code | Description |
|--|--------|---|
| RETURN_CODE_SDK_BUSY_EMV | | SDK doing EMV task |
| RETURN_CODE_ERR_OTHER | | SDK doing other task |
| RETURN_CODE_FAILED | | err response or data |
| RETURN_CODE_NOT_ATTACHED | | No reader attached |
| RETURN_CODE_MONO_AUDIO | | Mono audio enabled |
| RETURN_CODE_CONNECTED | | Connected |
| RETURN_CODE_LOW_VOLUME | | Audio volume too low |
| RETURN_CODE_CANCELED | | Task or CMD canceled |
| RETURN_CODE_INVALID_STR | | UF wrong string format |
| RETURN_CODE_NO_FILE | | UF file not found |
| RETURN_CODE_INVALID_FILE | | UF wrong file format |
| RETURN_CODE_HOST_UNREACHABLE | | Attempt to contact online host failed |
| RETURN_CODE_RKI_FAILURE | | Attempt to perform RKI failed |
| RETURN_CODE_MISSING_DLL | | DLL missing for the target reader |
| RETURN_CODE_ERR_BUF_NOT_ENOUGH | | Not enough memory for the buffer |
| RETURN_CODE_ERR_CMD_NOT_SUPPORTED | | Command not supported |
| RETURN_CODE_ERR_CMD_NOT_ALLOWED | | Command not allowed |
| RETURN_CODE_FW_INIT_SUCCESS | | Firmware initialized successfully |
| RETURN_CODE_BLOCK_TRANSFER_SUCCESS | 0x2B | Firmware write block successfully |
| RETURN_CODE_EMV_PIN | 0x31 | |
| RETURN_CODE_EMV_SIGNATURE | 0x32 | |
| RETURN_CODE_P2_DECLINED | 0X36 | No advice or reversal required (Declined) |
| RETURN_CODE_RKI_TIMEOUT | 0x101 | |
| RETURN_CODE_RKI_INITIALIZATION_ERROR | 0x0102 | |
| RETURN_CODE_RKI_AUTHENTICATION_ERROR | 0x0103 | |
| RETURN_CODE_RKI_TR31_ERROR | 0x0104 | |
| RETURN_CODE_P1_INCORRECT_FRAME_TAG | 0X0C01 | Incorrect frame tag |
| RETURN_CODE_P1_INCORRECT_FRAME_TYPE | 0X0C02 | Incorrect frame type |
| RETURN_CODE_P1_UNKNOWN_FRAME_TYPE | 0X0C03 | Unknown frame type |
| RETURN_CODE_P1_UNKNOWN_COMMAND | 0X0C04 | Unknown command |
| RETURN_CODE_P1_UNKNOWN_SUB_COMMAND | 0X0C05 | Unknown sub-command |
| RETURN_CODE_P1_CRC_ERROR | 0X0C06 | CRC error |
| RETURN_CODE_P1_FAILED | 0X0C07 | Failed |
| RETURN_CODE_P1_TIMEOUT | 0X0C08 | Timeout |
| RETURN_CODE_P1_INCORRECT_PARAMETER | 0X0C0A | Incorrect parameter |
| RETURN_CODE_P1_COMMAND_NOT_SUPPORTED | 0X0C0B | Command not supported |
| RETURN_CODE_P1_SUB_COMMAND_NOT_SUPPORTED | 0X0C0C | Sub-command not supported |
| RETURN_CODE_P1_STATUS_ABORT_COMMAND | 0X0C0D | Parameter not supported / status abort command |
| RETURN_CODE_P1_COMMAND_NOT_ALLOWED | 0X0C0F | Sub-command not allowed |
| RETURN_CODE_P2_ | 0X0D01 | Incorrect header tag |
| RETURN_CODE_P2_UNKNOWN_COMMAND | 0X0D02 | Unknown command |
| RETURN_CODE_P2_UNKNOWN_SUB_COMMAND | 0X0D03 | Unknown sub-command |
| RETURN_CODE_P2_CRC_ERROR | 0X0D04 | CRC error in frame |
| RETURN_CODE_P2_INCORRECT_PARAMETER | 0X0D05 | Incorrect parameter |
| RETURN_CODE_P2_PARAMETER_NOT_SUPPORTED | 0X0D06 | Parameter not supported |
| RETURN_CODE_P2_MAL_FORMATTED_DATA | 0X0D07 | Mal-formatted data |
| RETURN_CODE_P2_TIMEOUT | 0X0D08 | Timeout |
| RETURN_CODE_P2_FAILED | 0X0D0A | Failed / NACK |
| RETURN_CODE_P2_COMMAND_NOT_ALLOWED | 0X0D0B | Command not allowed |
| RETURN_CODE_P2_SUB_COMMAND_NOT_ALLOWED | 0X0D0C | Sub-command not allowed |
| RETURN_CODE_P2_BUFFER_OVERFLOW | 0X0D0D | Buffer overflow (data length too large for reader buffer) |
| RETURN_CODE_P2_USER_INTERFACE_EVENT | 0X0D0E | User interface event |

| Result | Code | Description |
|--|--------|---|
| RETURN_CODE_P2_COMM_TYPE_NOT_SUPPORTED | 0X0D11 | Communication type not supported, VT-1, burst, etc. |
| RETURN_CODE_P2_SECURE_INTERFACE_NOT_FUNCTIONAL | 0X0D12 | Secure interface not functional or in an intermediate state |
| RETURN_CODE_P2_DATA_FIELD_NOT_MOD_8 | 0X0D13 | Data field is not mod 8 |
| RETURN_CODE_P2_PADDING_UNEXPECTED | 0X0D14 | Pad 0x80 not found where expected |
| RETURN_CODE_P2_KEY_TYPE_INVALID | 0X0D15 | Specified key type is invalid |
| RETURN_CODE_P2_COULD_NOT_RETRIEVE_KEY | 0X0D16 | Could not retrieve key from the SAM (InitSecureComm) |
| RETURN_CODE_P2_HASH_CODE_ERROR | 0X0D17 | Hash code problem |
| RETURN_CODE_P2_COUNDTOT_STORE_KEY | 0X0D18 | Could not store the key into the SAM (InstallKey) |
| RETURN_CODE_P2_FRAME_TOO_LARGE | 0X0D19 | Frame is too large |
| RETURN_CODE_P2_RESEND_INITSECURECOMM_COMMAND | 0X0D1A | Unit powered up in authentication state but POS must resend the InitSecureComm command |
| RETURN_CODE_P2_EEPROM_NOT_INITIALIZED | 0X0D1B | The EEPROM may not be initialized because SecCommInterface does not make sense |
| RETURN_CODE_P2_APDU_ENCODING_ERROR | 0X0D1C | Problem encoding APDU |
| RETURN_CODE_P2_SAM_COMM_ERROR | 0X0D20 | Unsupported Index(ILM)SAM Transceiver error; problem communicating with the SAM (Key Mgr) |
| RETURN_CODE_P2_SEQUENCE_COUNTER_ERROR | 0X0D21 | Unexpected sequence counter in multiple frames for single bitmap(ILM); Length error in data returned from the SAM (Key Mgr) |
| RETURN_CODE_P2_IMPROPER_BITMAP | 0X0D22 | Improper bit map(ILM) |
| RETURN_CODE_P2_REQUEST_ONLINE_AUTHORIZATION | 0X0D23 | Request online authorization |
| RETURN_CODE_P2_RAW_DATA_READ_SUCCESSFUL | 0X0D24 | ViVOCard3 raw data read successful |
| RETURN_CODE_P2_MESSAGE_INDEX_NOT_AVAILABLE | 0X0D25 | Message index not available (ILM); ViVOComm activate transaction card type (ViVOComm) |
| RETURN_CODE_P2_VERSION_INFORMATION_MISMATCH | 0X0D26 | Version Information Mismatch (ILM) |
| RETURN_CODE_P2_INCORRECT_MESSAGE_INDEX | 0X0D27 | Not sending commands in correct message index (ILM) |
| RETURN_CODE_P2_NEXT_MESSAGE_NOT_RECEIVED | 0X0D28 | Time out or next expected message not received (ILM) |
| RETURN_CODE_P2_ILM_LANGUAGE_NOT_AVAILABLE | 0X0D29 | ILM languages not available for viewing (ILM) |
| RETURN_CODE_P2_OTHER_LANGUAGE_NOT_SUPPORTED | 0X0D2A | Other language not supported (ILM) |
| RETURN_CODE_UNKNOWN_ERROR_FROM_SAM | 0X0D41 | Unknown Error from SAM |
| RETURN_CODE_INVALID_DATA_DETECTED_BY_SAM | 0X0D42 | Invalid data detected by SAM |
| RETURN_CODE_INCOMPLETE_DATA_DETECTED_BY_SAM | 0X0D43 | Incomplete data detected by SAM |
| RETURN_CODE_RESERVED | 0X0D44 | Reserved |
| RETURN_CODE_INVALID_KEY_HASH_ALGORITHM | 0X0D45 | Invalid key hash algorithm |
| RETURN_CODE_INVALID_KEY_ENCRYPTION_ALGORITHM | 0X0D46 | Invalid key encryption algorithm |
| RETURN_CODE_INVALID_MODULUS_LENGTH | 0X0D47 | Invalid modulus length |
| RETURN_CODE_INVALID_EXPONENT | 0X0D48 | Invalid exponent |
| RETURN_CODE_KEY_ALREADY_EXISTS | 0X0D49 | Key already exists |
| RETURN_CODE_NO_SPACE_FOR_NEW_RID | 0X0D4A | No space for new RID |
| RETURN_CODE_KEY_NOT_FOUND | 0X0D4B | Key not found |
| RETURN_CODE_CRYPTOTOT_RESPONDING | 0X0D4C | Crypto not responding |
| RETURN_CODE_CRYPTOTOT_COMMUNICATION_ERROR | 0X0D4D | Crypto communication error |

| Result | Code | Description |
|--|--------|---|
| RETURN_CODE_P2_KEY_MANAGER_ERROR_4E | 0X0D4E | Module-specific error for key manager |
| RETURN_CODE_ALL_KEY_SLOTS_FULL | 0X0D4F | All key slots are full (maximum um number of keys has been installed) |
| RETURN_CODE_P2_AUTO_SWITCH_OK | 0X0D50 | Auto-switch OK |
| RETURN_CODE_P2_AUTO_SWITCH_FAILED | 0X0D51 | Auto-switch failed |
| RETURN_CODE_P2_DATA_DOES_NOT_EXIST | 0X0D60 | Data doesn't exist |
| RETURN_CODE_P2_DATA_FULL | 0X0D61 | Data Full |
| RETURN_CODE_P2_WRITE_FLASH_ERROR | 0X0D62 | Write flash error |
| RETURN_CODE_P2_OK_AND_HAVE_NEXT_COMMAND | 0X0D63 | OK and have next command |
| RETURN_CODE_P2_ACCOUNT_DUKPT_KWY_NOT_EXIST | 0X0D90 | Account DUKPT Key does not exist |
| RETURN_CODE_P2_ACCOUNT_DUKPT_KEY_KSN_EXHAUSTED | 0X0D91 | Account DUKPT Key KSN exhausted |
| RETURN_CODE_UNKNOWN_COMMAND | 0X9031 | Unknown command |
| RETURN_CODE_WRONG_PARAMETER | 0X9032 | Wrong parameter (such as the length of the command is incorrect) |
| RETURN_CODE_WAIT | 0X9038 | Wait (the command couldn't be finished in BWT) |
| RETURN_CODE_BUSY | 0X9039 | Busy (a previously command has not been finished) |
| RETURN_CODE_RETRIES_OVER_LIMIT | 0X903A | Number of retries over limit |
| RETURN_CODE_TIMEOUT | 0X8100 | Time out |
| RETURN_CODE_INVALID_MAN_SYSTEM_DATA | 0X9040 | Invalid manufacturing system data |
| RETURN_CODE_NOT_AUTHENTICATED | 0X9041 | Not authenticated |
| RETURN_CODE_INVALID_MASTER_DUKPT_KEY | 0X9042 | Invalid master DUKPT key |
| RETURN_CODE_INVALID_MAC_KEY | 0X9043 | Invalid MAC key |
| RETURN_CODE_RESERVED_FOR_FUTURE_USE | 0X9044 | Reserved for future use |
| RETURN_CODE_RESERVED_FOR_FUTURE_USE_ | 0X9045 | Reserved for future use |
| RETURN_CODE_INVALID_DATA_DUKPT_KEY | 0X9046 | Invalid DATA DUKPT key |
| RETURN_CODE_INVALID_PIN_PAIRING_DUKPT_KEY | 0X9047 | Invalid PIN pairing DUKPT key |
| RETURN_CODE_INVALID_DATA_PAIRING_DUKPT_KEY | 0X9048 | Invalid DATA pairing DUKPT key |
| RETURN_CODE_NO_NONCE_GENERATED | 0X9049 | No nonce generated |
| RETURN_CODE_NO_GUID_AVAILABLE | 0X9949 | No GUID available; perform getVersion first |
| RETURN_CODE_NO_MAC_CALCULATION | 0X9950 | MAC calculation unsuccessful; check BDK value |
| RETURN_CODE_NOT_READY | 0X904A | Not ready |
| RETURN_CODE_MSR_DATA_FAILED | 0X904B | MSR data failed |
| RETURN_CODE_INVALID_CERTIFICATE | 0X9050 | Invalid certificate |
| RETURN_CODE_DUPLICATE_KEY_DETECTED | 0X9051 | Duplicate key detected |
| RETURN_CODE_AT_CHECKS_FAILED | 0X9052 | AT checks failed |
| RETURN_CODE_TR34_CHECKS_FAILED | 0X9053 | TR34 checks failed |
| RETURN_CODE_TR31_CHECKS_FAILED | 0X9054 | TR31 checks failed |
| RETURN_CODE_AMC_CHECKS_FAILED | 0X9055 | MAC checks failed |
| RETURN_CODE_FIRMWARE_DOWNLOAD_FAILED | 0X9056 | Firmware download failed |
| RETURN_CODE_LOG_IS_FULL | 0X9060 | Log is full |
| RETURN_CODE_REMOVAL_SENSOR_UNENGAGED | 0X9061 | Removal sensor unengaged |
| RETURN_CODE_HARDWARE_PROBLEM | 0X9062 | Any hardware problems |
| RETURN_CODE_ICC_COMMUNICATION_TIMEOUT | 0X9070 | ICC communication timeout |
| RETURN_CODE_IFF_DATA_ERROR | 0X9071 | ICC data error (such check sum error) |
| RETURN_CODE_SMART_CARD_NOT_POWERED_UP | 0X9072 | Smart card not powered up |
| RETURN_CODE_NO_AID | 0xF200 | No AID or no application data |
| RETURN_CODE_NO_TERMINAL_DATA | 0xF201 | No host data |
| RETURN_CODE_WRONG_TLV_FORMAT | 0xF202 | Wrong TLV format |
| RETURN_CODE_AID_LIST_FULL | 0xF203 | AID list is full, maximum um is 16 |
| RETURN_CODE_NO_CA_KEY | 0xF204 | No CA Key |

| Result | Code | Description |
|---|--------|--|
| RETURN_CODE_NO_CA_KEY_RID | 0xF205 | No CA KeyRID |
| RETURN_CODE_NO_CA_KEY_INDEX | 0xF206 | No CA KeyIndex |
| RETURN_CODE_CA_KEY_LIST_FULL | 0xF207 | CA Keylist is full, maximum um is 96 |
| RETURN_CODE_CA_KEY_HASH_ERROR | 0xF208 | Wrong CA key hash |
| RETURN_CODE_COMMAND_FORMAT_ERROR | 0xF209 | Wrong transaction command format |
| RETURN_CODE_UNEXPECTED_COMMAND | 0xF20A | Unexpected command |
| RETURN_CODE_NO_CRL | 0xF20B | No CRL |
| RETURN_CODE_CRL_LIST_FULL | 0xF20C | CRL list is full, maximum is 30 |
| RETURN_CODE_MISSING_REQUIRED_PARAMETERS | 0xF20D | No amount, other amount and transaction type in transaction command |
| RETURN_CODE_CA_INCORRECT_HASH_ALGORITHM | 0xF20E | Wrong CA hash and encryption algorithm |
| RETURN_CODE_EMV_AUTHORIZATION_ACCEPTED | 0x0E00 | Authorization Accepted |
| RETURN_CODE_EMV_AUTHORIZATION_UNABLE_TO_GO_ONLINE | 0x0E01 | Unable to go online |
| RETURN_CODE_EMV_AUTHORIZATION_TECHNICAL_ISSUE | 0x0E02 | Technical issue |
| RETURN_CODE_EMV_AUTHORIZATION_DECLINED | 0x0E03 | Declined |
| RETURN_CODE_EMV_AUTHORIZATION_ISSUER_REFERRAL | 0x0E04 | Issuer referral transaction |
| RETURN_CODE_EMV_APPROVED | 0x0F00 | Online transaction accepted |
| RETURN_CODE_EMV_DECLINED | 0x0F01 | Online transaction declined |
| RETURN_CODE_EMV_GO_ONLINE | 0x0F02 | Request to go online |
| RETURN_CODE_EMV_FAILED | 0x0F03 | Transaction terminated |
| RETURN_CODE_EMV_SYSTEM_ERROR | 0x0F05 | Application was not selected by kernel or ICC format error or ICC missing data error |
| RETURN_CODE_EMV_NOT_ACCEPTED | 0x0F07 | ICC didn't accept transaction |
| RETURN_CODE_EMV_FALLBACK | 0x0F0A | Application may fallback to magstripe technology |
| RETURN_CODE_EMV_CANCEL | 0x0F0C | Transaction cancelled |
| RETURN_CODE_EMV_TIMEOUT | 0x0F0D | Timeout |
| RETURN_CODE_EMV_OTHER_ERROR | 0x0F0F | Other EMV error |
| RETURN_CODE_EMV_OFFLINE_APPROVED | 0x0F10 | Offline transaction approved |
| RETURN_CODE_EMV_OFFLINE_DECLINED | 0x0F11 | Offline transaction declined |
| RETURN_CODE_EMV_NEW_SELECTION | 0x0F21 | ICC detected but the conditions of use are not satisfied |
| RETURN_CODE_EMV_NO_AVAILABLE_APPS | 0x0F22 | No apps found on card matching host configuration |
| RETURN_CODE_EMV_NO_TERMINAL_FILE | 0x0F23 | Terminal file does not exist |
| RETURN_CODE_EMV_NO_CAPK_FILE | 0x0F24 | CAPK file does not exist |
| RETURN_CODE_EMV_NO_CRL_ENTRY | 0x0F25 | CRL entry does not exist |
| RETURN_CODE_BLOCKING_DISABLED | 0x0FFE | Return code when blocking is disabled |
| RETURN_CODE_CM100_WITHOUT_ERROR | 0xBBE0 | CM100 success |
| RETURN_CODE_CM100_PARAMETER | 0xBBE1 | CM100 parameter error |
| RETURN_CODE_CM100_LOWOUTBUFFER | 0xBBE2 | CM100 low output buffer |
| RETURN_CODE_CM100_CARD_NOT_FOUND | 0xBBE3 | CM100 card not found |
| RETURN_CODE_CM100_COLLISION_CARD_EXIST | 0xBBE4 | CM100 collision card exists |
| RETURN_CODE_CM100_TOO_MANY_CARDS_EXIST | 0xBBE5 | CM100 too many cards exist |
| RETURN_CODE_CM100_SAVED_DATA_NOT_EXIST | 0xBBE6 | CM100 saved data does not exist |
| RETURN_CODE_CM100_NO_DATA_AVAILABLE | 0xBBE8 | CM100 no data available |
| RETURN_CODE_CM100_INVALID_CID_RETURNED | 0xBBE9 | CM100 invalid cid returned |
| RETURN_CODE_CM100_INVALID_CARD_EXIST | 0xBBEA | CM100 invalid card exists |
| RETURN_CODE_CM100_COMMAND_UNSUPPORTED | 0xBBEC | CM100 command unsupported |
| RETURN_CODE_CM100_COMMAND_PROCESS | 0xBBED | CM100 error in command process |
| RETURN_CODE_CM100_INVALID_COMMAND | 0xBBEE | CM100 invalid command |

| Result | Code | Description |
|--|--------|---|
| RETURN_CODE_COMMAND_UNAVAILABLE | 0x0FFF | Command not applicable on the selected reader |
| IDG_P2_STATUS_CODE_DO_SUCCESS | 0 | |
| IDG_P2_STATUS_CODE_INCORRECT_HEADER_TAG | 0x01 | |
| IDG_P2_STATUS_CODE_UNKNOWN_COMMAND | 0x02 | |
| IDG_P2_STATUS_CODE_UNKNOWN_SUB_COMMAND | 0x03 | |
| IDG_P2_STATUS_CODE_CRC_ERROR_IN_FRAME | 0x04 | |
| IDG_P2_STATUS_CODE_INCORRECT_PARAMETER | 0x05 | |
| IDG_P2_STATUS_CODE_PARAMETER_NOT_SUPPORTED | 0x06 | |
| IDG_P2_STATUS_CODE_MAL_FORMATTED_DATA | 0x07 | |
| IDG_P2_STATUS_CODE_TIMEOUT | 0x08 | |
| IDG_P2_STATUS_CODE_FAILED_NACK | 0x0A | |
| IDG_P2_STATUS_CODE_COMMAND_NOT_ALLOWED | 0x0B | |
| IDG_P2_STATUS_CODE_SUB_COMMAND_NOT_ALLOWED | 0x0C | |
| IDG_P2_STATUS_CODE_BUFFER_OVERFLOW | 0x0D | |
| IDG_P2_STATUS_CODE_USER_INTERFACE_EVENT | 0x0E | |
| IDG_P2_STATUS_CODE_NEED_CLEAR_FIRMWARE | 0x10 | |
| IDG_P2_STATUS_CODE_COMMUNICATION_TYPE_NOT_SUPPORTED | 0x11 | |
| IDG_P2_STATUS_CODE_SECURE_INTERFACE_NOT_FUNCTIONAL | 0x12 | |
| IDG_P2_STATUS_CODE_DATA_FIELD_NOT_MOD_8 | 0x13 | |
| IDG_P2_STATUS_CODE_PAD_0x80_NOT_FOUND | 0x14 | |
| IDG_P2_STATUS_CODE_SPECIFIED_KEY_TYPE_INVALID | 0x15 | |
| IDG_P2_STATUS_CODE_FAIL_RETRIEVE_KEY_FROM_SAM | 0x16 | |
| IDG_P2_STATUS_CODE_HASH_CODE_PROBLEM | 0x17 | |
| IDG_P2_STATUS_CODE_FAIL_STORE_KEY_INTO_SAM | 0x18 | |
| IDG_P2_STATUS_CODE_FRAME_TOO_LARGE | 0x19 | |
| IDG_P2_STATUS_CODE_POS_RESEND_INITSECURECOMM | 0x1A | |
| IDG_P2_STATUS_CODE_EEPROM_NOT_INITIALIZED | 0x1B | |
| IDG_P2_STATUS_CODE_PROBLEM_ENCODING_APDU | 0x1C | |
| IDG_P2_STATUS_CODE_UNSUPPORTED_INDEX_SAM | 0x20 | |
| IDG_P2_STATUS_CODE_UNEXPECTED_SEQUENCE_COUNTER | 0x21 | |
| IDG_P2_STATUS_CODE_IMPROPER_BIT_MAP | 0x22 | |
| IDG_P2_STATUS_CODE_REQUEST_ONLINE_AUTHORIZATION | 0x23 | |
| IDG_P2_STATUS_CODE_VIVOCARD3_RAW_DATA_READ_SUCCESSFUL | 0x24 | |
| IDG_P2_STATUS_CODE_MESSAGE_INDEX_NOT_AVAILABLE | 0x25 | |
| IDG_P2_STATUS_CODE_VERSION_INFORMATION_MISMATCH | 0x26 | |
| IDG_P2_STATUS_CODE_NOT_SENDING_COMMANDS_IN_CORRECT_INDEX_MESSAGE | 0x27 | |
| IDG_P2_STATUS_CODE_EXPECTED_MESSAGE_NOT_RECEIVED | 0x28 | |
| IDG_P2_STATUS_CODE_ILM_LANGUAGES_NOT_AVAILABLE | 0x29 | |
| IDG_P2_STATUS_CODE_OTHER_LANGUAGE_NOT_SUPPORTED | 0x2A | |
| IDG_P2_STATUS_CODE_MODULE_SPECIFIC_ERROR_FOR_KEY_MANAGER_1 | 0x41 | |
| IDG_P2_STATUS_CODE_MODULE_SPECIFIC_ERROR_FOR_KEY_MANAGER_2 | 0x42 | |
| IDG_P2_STATUS_CODE_MODULE_SPECIFIC_ERROR_FOR_KEY_MANAGER_3 | 0x43 | |
| IDG_P2_STATUS_CODE_MODULE_SPECIFIC_ERROR_FOR_KEY_MANAGER_4 | 0x44 | |
| IDG_P2_STATUS_CODE_MODULE_SPECIFIC_ERROR_FOR_KEY_MANAGER_5 | 0x45 | |
| IDG_P2_STATUS_CODE_MODULE_SPECIFIC_ERROR_FOR_KEY_MANAGER_6 | 0x46 | |
| IDG_P2_STATUS_CODE_MODULE_SPECIFIC_ERROR_FOR_KEY_MANAGER_7 | 0x47 | |
| IDG_P2_STATUS_CODE_MODULE_SPECIFIC_ERROR_FOR_KEY_MANAGER_8 | 0x48 | |
| IDG_P2_STATUS_CODE_MODULE_SPECIFIC_ERROR_FOR_KEY_MANAGER_9 | 0x49 | |
| IDG_P2_STATUS_CODE_MODULE_SPECIFIC_ERROR_FOR_KEY_MANAGER_10 | 0x4A | |
| IDG_P2_STATUS_CODE_MODULE_SPECIFIC_ERROR_FOR_KEY_MANAGER_11 | 0x4B | |
| IDG_P2_STATUS_CODE_MODULE_SPECIFIC_ERROR_FOR_KEY_MANAGER_12 | 0x4C | |
| IDG_P2_STATUS_CODE_MODULE_SPECIFIC_ERROR_FOR_KEY_MANAGER_13 | 0x4D | |
| IDG_P2_STATUS_CODE_MODULE_SPECIFIC_ERROR_FOR_KEY_MANAGER_14 | 0x4E | |
| IDG_P2_STATUS_CODE_MODULE_SPECIFIC_ERROR_FOR_KEY_MANAGER_15 | 0x4F | |

| Result | Code | Description |
|--|------|-------------------------------|
| IDG_P2_STATUS_CODE_AUTO_SWITCH_OK | 0x50 | |
| IDG_P2_STATUS_CODE_AUTO_SWITCH_FAILED | 0x51 | |
| IDG_P2_STATUS_CODE_OK_AND_HAVE_NEXT_COMMAND | 0x63 | |
| IDG_P2_STATUS_CODE_ANTENNA_ERROR | 0x70 | |
| IDG_P2_STATUS_CODE_USE_ANOTHER_CARD | 0x80 | |
| IDG_P2_STATUS_CODE_INSERT_OR_SWIPE_CARD | 0x81 | |
| IDG_P2_STATUS_CODE_DATA_ENCRYPTION_KEY_NOT_EXIST | 0x90 | |
| IDG_P2_STATUS_CODE_DATA_ENCRYPTION_KEY_KSN_EXHAUSTED | 0x91 | |
| IDG_P2_STATUS_CODE_BUFFER_NOT_ENOUGH | 0x92 | |
| IDG_P2_STATUS_CODE_DISCONNECTED | 0x93 | |
| IDG_P2_STATUS_CODE_SDK_BUSY_MSR | 0x94 | |
| IDG_P2_STATUS_CODE_SDK_BUSY_PINPAD | 0x95 | |
| IDG_P2_STATUS_CODE_SDK_BUSY_EMV | 0x96 | |
| IDG_P2_STATUS_CODE_SDK_BUSY_CTL5 | 0x97 | |
| IDG_P2_STATUS_CODE_FW_INIT_SUCCESS | 0x98 | |
| IDG_P2_STATUS_CODE_NO_DRIVE_SPACE | 0x99 | |
| IDG_P2_USER_NOT_VALIDATED_TO_PERFORM_THIS_FUNCTION | 0x9A | |
| IDG_P2_SERVER_NOT_RESPONDING | 0x9B | |
| IDG_P2_INVALID_URL | 0x9C | |
| IDG_P2_UNABLE_TO_VALIDATE_WITH_SERVER | 0x9D | |
| IDG_P2_FILE_NOT_FOUND_ON_SERVER | 0x9E | |
| IDG_P2_FILE_NOT_FOUND_ON_PATH | 0x9F | |
| IDG_P2_STATUS_CODE_CDCVM | 0xA0 | Error code for CDCVM |
| IDG_P2_STATUS_CODE_CARD_INSERTED | 0xA3 | Error code for card insertion |

2.5 Serial Communication Interfaces

This section discusses the physical interfaces through which the host device communicates with the ViVOpay reader. All of the readers have either an RS232 or USB Serial Interface.

Note: Do not plug in/out serial communication interfaces during the reader power-on cycle, as this might cause unstable behavior.

2.6 Bluetooth Interface

Some ViVOpay readers support Bluetooth communication. Use the following commands to set Bluetooth parameters, if applicable.

2.6.1 Set Bluetooth Parameters (77-81)

The **Set Bluetooth Parameters** command sets a Bluetooth name and password.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|---------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViV0tech2\0 | 77h | 81h | var | | var | | |

Command Data Field

| Data Field | Length (bytes) | Description |
|--------------|----------------|---------------------------|
| Name | Var to 26 | ASCII, must end with 0x00 |
| Old Password | 6 | Hex |
| New Password | 6 | Hex |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 77h | See Status Code Table | 00h | 00h | | |

2.6.2 Get Bluetooth Name (77-82)

The **Get Bluetooth Name** command retrieves a Bluetooth name.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|---------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 77h | 82h | 00h | 00h | null | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 77h | See status code table | var | | ASCII (<26 bytes) | | |

The Bluetooth name of the response data must be less than 26 bytes.

2.6.3 Get Bluetooth Local Information (77-83)

The **Get Bluetooth Local Information** command retrieves Bluetooth local information for the module version and Bluetooth address.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|---------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViV0tech2\0 | 77h | 83h | var | | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | 77h | See status code table | 00h | 0Dh | See below data table | | |

Data Table: Get Bluetooth Local information

| Data Item | Length (bytes) | Description |
|-----------------|----------------|------------------------------|
| Event | 2 | Reserved |
| Version | 4 | Firmware version (BCD) |
| Address | 6 | Ascii code MAC Address (LSB) |
| Identify | 1 | 0x01 – BM71 |

2.6.4 Set Bluetooth Parameters (77-84)

The **Set Bluetooth Parameters** command is a generic command to set Bluetooth parameters.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 – Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|---------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data Field | CRC (LSB) | CRC (MSB) |
| ViV0tech2\0 | 77h | 84h | Var | | | | |

Data Field

| Byte 14 | Byte 15-16 | Byte 17 – Byte 16+m |
|----------------------------|------------|--|
| Parameter | Length (m) | Parameter Description |
| 0x00: Advertising Interval | 2 | Bluetooth advertising interval in milliseconds Little endian, range is [1, 65535] ms <ul style="list-style-type: none"> VP3600 Advertising interval range is [20, 10240]ms VP3320 Advertising interval range is [1, 6250]ms |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 77h | See status code table | 00h | 00h | | |

2.7 RS-232 Serial Interface

2.7.1 Port Settings

To communicate with applicable ViVOpay readers using RS-232, set the host's serial communication parameters to the values listed below.

Serial Port Settings

| Parameter | Value |
|---------------------|-----------------------------------|
| Baud Rate | 115200 unless otherwise specified |
| Data Bits | 8 |
| Stop Bits | 1 |
| Parity | None |
| Out CTS Flow | Disabled |
| Out DSR Flow | Disabled |
| DTR Control | Disabled |
| RTS Control | Disabled |
| XON/XOFF | Disabled |
| Flow Control | None |

2.7.2 Basic Communication

The ViVOpay reader and the host device communicate by exchanging command-response frames. The host always initiates communication by sending a command frame and the reader responds by sending a response frame. The frames that are exchanged depend on the command and the protocol used. There are two command/response protocols. Protocol 1, now deprecated, uses separate data frames and ACK/NACK responses. Protocol 2 is simplified by including data within the command/response frames.

2.7.3 Timeouts

The ViVOpay reader does not timeout while trying to receive a command. There is no maximum inter-character delay. As a result, command frames with length errors may appear to “hang.” A subsequent command that does not contain a length error can be received successfully.

After the reader has received a command, the time required to respond to the host varies from command to command, depending on what processing is required.

During a transaction, the **Activate Transaction** command may specify a timeout value. The reader will continue to poll until it starts to process a transaction, or the specified timeout period has elapsed. The transaction may not complete within the timeout period.

2.8 USB HID Interface

All ViVOpay commands sent over the USB HID interface are encapsulated in the following protocol.

Note: The maximum length for any command or response is 1500 bytes, which is the size of the command FIFO buffer.

2.8.1 HID Report Format

All HID reports sent to or received from the ViVOpay reader are in frames that are 64 bytes long. The first byte of the frame is a single byte report ID number. The remaining 63 bytes carry the report payload. Any reports with less than 63 bytes of command or response data are padded with NULL bytes (00h) to make them 63 bytes long.

ViVOpay commands and responses are sent over the USB bus in 63-byte frames. Byte ordering in the USB frame is the same as if the command were sent over the serial port. In other words, the “ViVOTech2” command tag always starts in the second byte of the first report containing the command, just after the Report ID.

There are four defined report IDs used in this protocol: 1, 2, 3, and 4. Undefined report IDs are silently ignored.

1.1.1.1. Report ID 1

ID 1 frames are used when a complete command or response is 63 bytes or less. As soon as the host or reader receives a report ID 1 frame, it should parse the report data to extract the command or response.

1.1.1.2. Report ID 2

ID 2 frames are used when a complete command or response is more than 63 bytes long and cannot fit in a single report. The report ID 2 frame contains the first 63 bytes of the command. So the "ViVOtech2" command tag is only present in report IDs 1 or 2. The report ID 2 frames always contain 63 bytes of valid data with no padding bytes since the command is more than 63 bytes long.

1.1.1.3. Report ID 3

ID 3 frames are continuation frames. For any command or response that is more than 126 bytes long, the middle frames of the response are sent with a report ID of 3. Any frame received with a report ID 3 is ignored unless it is preceded by a report with an ID of 2 or 3. The report ID 3 frames should always contain 63 bytes of valid data with no padding bytes.

1.1.1.4. Report ID 4

ID 4 frames mark the end of multi-report commands. Any padding needed to make the command a multiple of 63 bytes should be placed in this report. Any frame received with a report ID 4 is ignored unless it is preceded by a report with an ID of 2 or 3. As soon as the host or reader receives a valid report ID 4 frame, it should parse the report data to extract the command or response.

The exception to the rule of only adding pad bytes to reports with IDs of 1 or 4 is debug test frames. Surrounding a command with pad bytes to make the command span multiple reports is valid for testing the multi-report handling of the host and reader software. Avoided doing this in deployed code because it slows command processing times.

2.8.2 Sample Single Report Command and Response

Ping Command Report

| | | | | | | | | | | | | | | | |
|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|
| 01 | "V" | "i" | "V" | "O" | "t" | "e" | "c" | "h" | "2" | 00 | 18 | 01 | 00 | 00 | B3 |
| CD | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |

Ping Response Report

| | | | | | | | | | | | | | | | |
|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|
| 01 | "V" | "I" | "V" | "O" | "t" | "e" | "c" | "h" | "2" | 00 | 18 | 00 | 00 | 00 | FA |
| 83 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |

The serial port version of this command and response would be (data bytes in hex format):

Command: 56 69 56 4F 74 65 63 68 32 00 18 01 00 00 B3 CD

Response: 56 69 56 4F 74 65 63 68 32 00 18 00 00 00 FA 8

2.8.3 Data Frames

| Byte 0-8 | Byte 9 | Byte 10 | Byte 11 | ... | Byte n+10 | Byte n+11 | Byte n+12 |
|------------|------------|---------|---------|-----|-----------|--|--|
| Frame Tag | Frame Type | Data 0 | Data 1 | ... | Data n | CRC MSB if from ViVOpay. LSB if from Host. | CRC LSB if from ViVOpay. MSB if from Host. |
| ViVOtech\0 | 'D' | | | ... | | | |

Direction: Both ways (depending on command). Variable length (n = 1 ... 244).

2.8.4 Sample Single Report Command with Multiple Report Response

Get Configuration Command Report

| | | | | | | | | | | | | | | | |
|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|
| 01 | "V" | "I" | "V" | "O" | "t" | "e" | "c" | "h" | "2" | 00 | 03 | 02 | 00 | 00 | 5B |
| 91 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |

Get Configuration response reports

1st Response Report

| | | | | | | | | | | | | | | | |
|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|
| 02 | "V" | "I" | "V" | "O" | "t" | "e" | "c" | "h" | "2" | 00 | 03 | 00 | 00 | C2 | FF |
| E4 | 01 | 00 | 9F | 02 | 06 | 00 | 00 | 00 | 00 | 00 | 01 | 9F | 03 | 06 | 00 |
| 00 | 00 | 00 | 00 | 00 | DF | 63 | 01 | 00 | DF | 64 | 01 | 01 | DF | 65 | 01 |
| 00 | DF | 66 | 01 | 00 | FF | F0 | 03 | 00 | 00 | 00 | FF | F2 | 08 | 30 | 30 |

2nd Response Report

| | | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 03 | 30 | 30 | 30 | 30 | 30 | 30 | FF | F3 | 02 | 03 | FF | FF | F7 | 01 | 02 |
| FF | F9 | 01 | 03 | FF | FA | 02 | 03 | E8 | 9A | 03 | 00 | 01 | 04 | 9F | 21 |
| 03 | 05 | 13 | 54 | 9C | 01 | 00 | 5F | 2A | 02 | 08 | 40 | 9F | 09 | 02 | 00 |
| 02 | 9F | 1A | 02 | 08 | 40 | 9F | 1B | 04 | 00 | 00 | 17 | 70 | 9F | 33 | 03 |

3rd Response Report

| | | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 03 | 00 | 08 | E8 | 9F | 35 | 01 | 22 | 9F | 40 | 05 | 60 | 00 | 00 | 30 | 00 |
| 9F | 66 | 04 | A0 | 00 | 00 | 00 | FF | F1 | 06 | 00 | 00 | 00 | 01 | 00 | 00 |
| FF | F4 | 03 | 01 | 00 | 01 | FF | F5 | 06 | 00 | 00 | 00 | 00 | 80 | 00 | FF |
| F8 | 01 | 00 | FF | FB | 01 | 00 | FF | FC | 01 | 00 | FF | FD | 05 | F8 | 50 |

4th and Final Response Report

| | | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 04 | AC | F8 | 00 | FF | FE | 05 | F8 | 50 | AC | A0 | 00 | FF | FF | 05 | 00 |
| 00 | 00 | 00 | 00 | 72 | 56 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |

Note: The response to this command changes each time the command is sent since it includes the current time and date.

The serial port version of this command and response would be (data bytes in hex format):

Command: 56 69 56 4F 74 65 63 68 32 00 03 02 00 00 5B 91
Response: 56 69 56 4F 74 65 63 68 32 00 03 00 00 C2 FF E4 01 00
 9F 02 06 00 00 00 00 00 01 9F 03 06 00 00 00 00 00 00 DF 63 01 00 DF
 64 01 01 DF 65 01 00 DF 66 01 00 FF F0 03 00 00 00 FF F2 08 30 30 30 30
 30 30 30 30 FF F3 02 03 FF FF F7 01 02 FF F9 01 03 FF FA 02 03 E8 9A 03
 00 01 04 9F 21 03 05 13 54 9C 01 00 5F 2A 02 08 40 9F 09 02 00 02 9F 1A
 02 08 40 9F 1B 04 00 00 17 70 9F 33 03 00 08 E8 9F 35 01 22 9F 40 05 60
 00 00 30 00 9F 66 04 A0 00 00 00 FF F1 06 00 00 00 01 00 00 FF F4 03 01
 00 01 FF F5 06 00 00 00 00 80 00 FF F8 01 00 FF FB 01 00 FF FC 01 00 FF
 FD 05 F8 50 AC F8 00 FF FE 05 F8 50 AC A0 00 FF FF 05 00 00 00 00 00 72
 56

2.8.5 Error Handling at Report Level

1. Any report with ID of 1 is processed as soon as it is received. All other unprocessed reports are discarded.
2. Any report with an ID of 2 causes all other unprocessed reports to be discarded.
3. Any report with an ID of 3 is discarded unless the previous report had an ID of 2 or 3. If the previous report ID 3 was discarded then this report also is discarded.
4. Any report with an ID of 4 is discarded unless the previous report had an ID of 2 or 3. If the previous report had an ID of 3 and was discarded then this report also is discarded. If the report ID 4 frame is retained, then all retained reports are processed.

Processing of reports means passing the concatenated data frames contained in the reports to the command handler. The report ID bytes must be discarded when concatenating the report data frames.

An alternate way to handle the rules for report IDs 3 and 4 is to set a flag when a report with an ID of 2 is received and reset the flag when a report with an ID of 1 is received or an ID of 4 is finished processing. Reports with IDs of 3 or 4 are only kept when the flag is set.

2.8.6 Error Handling at Command Level

The error handling at the command level remains as it is currently implemented for serial port commands.

Incomplete commands are silently ignored when the reception times out. This does not occur for commands received over the USB HID interface unless a complete report is dropped, resulting in missing data for the command. The normal USB handshaking is expected to prevent this.

A bad CRC value for the encapsulated command returns a bad CRC response to the command. An unknown command or subcommand code results in an unknown command or unknown subcommand ResponseFrame.

If the host does not receive any response to a command it should retry the command. If the host receives a bad CRC response to a command it should retry the command. This is not expected to occur when using USB because it includes a layer of error handling.

3. ViVOpay Communication Protocols

There are two main types of protocols: Protocol 1 (now deprecated), and Protocol 2. Protocol 2 is the preferred method of communicating between the host device and a ViVOpay reader. Protocol 1 is retained for backward compatibility with older host applications.

In addition to the two main protocols, there are modes of communication that are extensions of the protocols. These modes provide flexibility in the control of the reader:

- Pass-through mode allows the host application to interact directly with contactless cards.
- Burst Mode is a legacy mode intended for use with magstripe cards.

3.1 Command and Response Frames

Request and response interactions with the reader via serial connection involve two types of frames: command frames and response frames. The general format of these frames is given below.

3.1.1 Command Frame Format

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTECH2\0 | | | | | | | |

3.1.2 Response Frame Format

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-------------|----------------------|----------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTECH2\0 | | | | | | | |

3.2 Pass-Through Mode

Pass-through is actually a special mode of ViVOpay Protocol 2. Host devices can use Pass-Through Mode to communicate directly with a card. This feature allows a host to add support for RF cards that are not directly supported by ViVOpay firmware.

This section describes the Pass-Through protocol and frames for the ViVOpay serial interface protocol.

Note: Pass-through commands could only be executed in Pass-Through Mode. Other commands (non-Pass-Through) will return an error in Pass-Through Mode with the exception of **Ping**, **Get Version**, and **Get Serial Number** commands.

3.2.1 Basic Pass-Through Operation

Pass-through mode allows a host device to communicate directly with an ISO 14443 Type A or Type B proximity integrated circuit card (PICC) without the ViVOpay firmware knowing the specifics of the application or data present on the PICC. Pass-through mode supports a set of basic commands that allow polling and selection of a PICC and sending and receiving low-level information to and from the PICC. This allows a host to communicate with (and support) cards with applications and data that are not supported by a system AID. Individual Pass-Through commands are described below.

Pass-through mode subcommands are grouped into three categories:

- **General Pass-Through set up commands**

These commands have to be used regardless of using high-level communication with the PICC or low-level communication. These commands include:

- **Pass-Through Mode Start/Stop**
- **Poll for Token**

- **High-level PICC communication commands**

If a PICC supports ISO 14443-4 Protocol, high-level commands can be used to send application-level APDUs to the PICC and receive PICC responses. The send/receive commands must always be used in pairs unless the send command returns an error. High-level commands include [Exchange Contactless Data](#) and commands to interact with Mifare cards.

- **Low-level PICC communication commands**

Low-level commands can be used to send raw ISO 14443-3 data to the PICC and receive PICC responses. The send/receive commands must always be used in pairs unless the send command returns an error. In addition, these commands can also get and set some PCD and PICC parameters. Low-level commands include.

- **PCD Single Command Exchange**
- **Get PCD and PICC Parameters**

The host device must periodically instruct the reader to poll for cards. Whenever the reader detects a card in the RF field, it tries to carry out ISO 14443 Layer 3 and Layer 4 negotiations and report the card type found. In Pass-Through Mode, a ViVOpay reader does not attempt to check if the card application is one that it supports.

After a card is detected, the host may use one of the Pass-Through commands to communicate with the card at the application level and read the data.

Additional Pass-Through commands allow a host to use low-level features provided by the reader, such as controlling the RF antenna (field).

3.2.2 Pass-Through Command Frame Format

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|----------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | See Individual Commands | See Individual Commands | See Individual Commands | See Individual Commands | See Individual Commands | | |

Note: Within each Pass-Through Frame Type, the CRC is stored as big-endian number; that is, higher byte (MSB) first.

3.2.3 Pass-Through Response Frame Format

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | See Individual Commands | See Individual Commands | See Individual Commands | See Individual Commands | See Individual Commands | | |

3.2.4 Suggested Sequence for Pass-Through Commands

1. Put the ViVOpay reader in Pass-Through Mode by sending a [Start Pass-Through Mode](#) command.
2. Periodically request the reader to poll for cards by sending the **Poll for Token** command. If no card is found within the time specified, the reader indicates this with a timeout error. If a card is found, it returns the card type and serial number.
3. At this point, the reader has already gone through the anti-collision, selection, and activation sequence (if required) as per ISO 14443 A/B, and the card is ready for communication. Depending on the card type, use the appropriate Pass-Through commands to communicate with the card. Card types and the applicable commands are listed below.

1.1.1.5. For ISO 14443-4 Compliant Type A or Type B Cards

Use the [Exchange Contactless Data](#) command to communicate with the Card at the application level.

1.1.1.6. For ISO 14443 Type A or Type B Cards that are not ISO 14443-4 Compliant (such as ISO 14443-3 Compliant Cards), Mifare Type A, and Mifare Ultralight Type A

Low-level commands: Use the [PCD Single Command Exchange](#) command to communicate with the card. If required, use the [Get PCD and PICC Parameters](#) command for greater control.

High-level commands (for Mifare cards only): Or use the [Mifare Authenticate Block](#), [Mifare Read Blocks](#), and [Mifare Write Blocks](#) commands to communicate with a Mifare Standard (1K) or Mifare Ultralight Card.

1.1.1.7. For Card Type None

The card has either been removed from the field, or there was an error in trying to connect to the card, or the card is not ISO 14443-3 or 14443-4 compliant. There is no need to communicate with the card.

The host device is responsible for handling the termination sequence when it is finished communicating with the card. The host may use the [Antenna Disable/Antenna Enable](#) commands to turn the RF field off and then on again.

The host device can instruct the ViVOpay reader to terminate Pass-Through Mode and start normal polling for cards by sending a [Stop Pass-Through Mode](#) command.

Note: If the host device communicates with the card in Pass-Through Mode and finds that it does not support the card, then the host is responsible for handling the termination sequence with the card. The host may keep sending [Poll for Token](#) commands to the reader until the card has been removed from the field, replaced by another card (different serial number), or a timeout has occurred before it terminates Pass-Through Mode. The host may choose to terminate Pass-Through Mode as soon as its reading is complete.

Care should be taken to ensure that the reader is operating in the correct mode (auto-poll or poll on demand) when returning from Pass-Through Mode. If the card is not removed from the field fast enough, and the reader is in auto poll mode, the host may end up doing multiple reads of the same card.

3.2.5 Auto-Switch to Pass-Through Mode

The reader can be set to switch automatically out of polling (either poll on demand or auto-poll) and enter Pass-Through Mode. This allows the POS application to send Pass-Through Mode commands directly to the card APDU without explicitly setting the reader in Pass-Through Mode. Auto-switch can be enabled globally and for configurable user AIDs. This feature is not supported for system AIDs.

If the auto-switch feature is enabled, the reader switches to Pass-Through Mode under the following conditions:

- Card application is not recognized – global auto-switch is enabled
- Card AID is not recognized – global auto-switch is enabled
- Mifare card is recognized but fails – global auto-switch is enabled
- DesFire card is recognized but fails – global auto-switch is enabled
- Card AID is recognized – user AID auto-switch is enabled

There are two ways to use the auto-switch feature: global auto-switch or user AID auto-switch. The DFEE7C TLV sets the feature globally using the **Set Configuration** command (global auto-switch) and the **Set Configurable AID** command sets the feature for user AIDs (user AID auto-switch). Use both commands at the same time if desired, but they do different things so do not confuse the two. In general, one is used for MiFare, DesFire, or unrecognized cards. The second is ONLY used for a specific User AID. The auto-switch setting in a user AID overrides the global auto-switch setting.

After auto-switch is activated, the POS application must handle error recovery and exit Pass-Through Mode with the **Pass-Through Mode Start/Stop** command (2C-01) when done. The reader returns to previous polling mode or idle state. For example, if exiting Pass-Through Mode and resuming auto poll mode, the POS must make sure the PICC has left the field before terminating Pass-Through Mode or else Auto Poll will start and the reader will read the PICC again as a new transaction.

1.1.1.8. Global Auto-Switch

Use Global Auto-Switch to process:

- an unrecognized MiFare PICC
- an unrecognized DesFire PICC
- a completely unrecognized PICC (failed MiFare, DesFire, PPSE, Trial & Error)

Auto-switch is invoked if global auto-switch is enabled AND one of the above cards is tapped on the reader during a transaction.

If successful, the reader returns a response frame containing some of the following items:

- Error or Status condition
- UID
- PICC card type detected

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|------------|---|----------------------|----------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTEch2\0 | 50h or 51h | See status code table | | | | | |

Poll for Token Data Field for Response Frame (Status Code is OK)

| Data Field | Length (bytes) | Description |
|------------------|-------------------|--|
| Card | 1 | Type of Card Found (or No Card Found). 00h None (Card Not Detected or Could not Activate) 01h ISO 14443 Type A (Supports ISO 14443-4 Protocol) 02h ISO 14443 Type B (Supports ISO 14443-4 Protocol) 03h Mifare Type A (Standard) 04h Mifare Type A (Ultralight) 05h ISO 14443 Type A (Does not support ISO 14443-4 Protocol) 06h ISO 14443 Type B (Does not support ISO 14443-4 Protocol) 07h ISO 14443 Type A and Mifare (NFC phone) |
| Serial Number | 0 or Variable | Serial Number (or the UID) of the PICC. Length depends on the card detected. If no card was detected, then a serial number is not returned. |

The response frame is returned asynchronously if the reader is in auto poll.

After auto-switch is invoked the reader remains in Pass-Through Mode with the RF antenna on. The POS application must handle error recovery and exit Pass-Through Mode when done with the **Pass-Through Mode Start/Stop** command (2C-01). The reader returns to previous polling mode or idle state.

To enable global auto-switch, send the **Set Configuration** command (04-00) with a 01h value for the DFEE7C TLV.

1.1.1.9. User AID Auto-Switch

Use user aid auto-switch to process:

- a recognized User AID that is selected during PPSE
- a recognized User AID that is selected during trial & error

Auto-switch is invoked if user aid auto switch is enabled for an AID and a PICC is initiating a transaction with this AID selected.

If successful, the reader returns a response frame containing some of the following items:

- Error or Status condition
- AID
- PICC card type detected

The response frame is returned asynchronously if the reader is in auto poll.

After auto-switch is invoked the reader remains in Pass-Through Mode with the RF antenna on. The POS application must handle error recovery and exit Pass-Through Mode when done with the **Pass-Through Mode Start/Stop** command (2C-01). The reader returns to previous polling mode or idle state.

To enable global auto-switch, send the **Set Configurable AID** command (04-02) with a 01h value for the DFEE7C TLV.

3.2.6 RF On/Off States for Pass-through Commands

Sending a Stop Pass-through command will turn off the RF Antenna. Otherwise, the antenna is under the direct control of the host device in Pass-through mode.

3.3 Burst Mode

In Burst Mode (which requires the reader to be in Auto Poll Mode and is useful only for MSR data; see note below), the ViVOpay reader sends a data frame to the host device each time it successfully reads a card. The ViVOpay continues polling for supported RF cards. Whenever the reader detects a card in the RF field, it tries to read the card data.

If the read operation is successful, the reader sends a "card payload" frame that contains the status, application type, card data, and CRC to the host through its serial port. Detailed information on the frame format is given in the sections ahead. The host does not have to send any command or data to the reader.

Note: The reader must be in auto poll mode for to use Burst Mode. Setting Burst Mode on for other configurations can lead to unexpected results.
Burst Mode is intended to be used with magnetic stripe card data only.

Enable Burst Mode with the **Set Configuration** command and the FFF7 tag. There are two options for Burst Mode: **Always On** (FFF7 = 01) and **Auto Exit** (FFF7 = 02). When the reader is in **Burst Mode Always On**, it ignores **Activate Transaction** and **Get Full Track Data** commands and remains in Burst Mode. When **Burst Mode Auto Exit** is enabled, the reader ends Burst Mode (FFF7 = 00) and processes these commands. Burst Mode then remains off until it is reactivated with a new **Set Configuration** command with tag FFF7 set to 01 or 02.

When the Burst Mode is enabled, the standard ViVOpay serial interface is not disabled entirely. Commands not related to transactions, such as **Ping**, can still be sent to the reader. In the command-response mode, the host device sends a command to the reader and the reader responds in a pre-defined manner. These commands allow a host to use features provided by

the reader, such as checking for the presence of the reader by pinging it, or retrieving the firmware version number.

Burst Mode is not allowed when MSR/MSD or EMV encryption is enabled and a data encryption key exists.

When MSR/MSD or EMV encryption is enabled and a data encryption key exists, Burst Mode is always OFF. In this condition, the reader will turn Burst Mode OFF automatically. If Burst Mode is set to ON/AUTO EXIT through the **Set Configuration (04-00)** command, the reader will keep Burst Mode OFF and respond with the status code "Command Not Supported (0x0B)."

Note: Burst Mode is disabled for SRED readers.

3.3.1 ViVOpay Burst Mode Frames

The table below describes the Burst Mode frame types. The frame type appears in Byte 0 of a Burst Mode packet.

Burst Mode Frames

| Frame type | Description |
|------------|---|
| 01h | Payload Frame |
| 02h | Status Frame |
| 03h | Payload Frame for VISA MSD 202 CVN17 type transaction |
| 55h | NACK |
| 0Eh | Asynchronous Event Frame |

1.1.1.10. Payload Frame (On Successful Read)

On a successful read, the ViVOpay reader sends a card payload frame to the host device that always contains frame type, status, and application type. The status always shows **Success (=00)**. The application type can have any of the values defined in [Data Definitions](#). This is followed by the track data. Only those tracks the reader was able to read from the card are sent. Each track begins and ends with its start and end sentinel. After the track data, the reader sends two CRC bytes. The details of the CRC algorithm used are given in [CRC Calculation](#).

| Byte 0 | Byte 1 | Byte 2 | | | | Byte n-1 | Byte n |
|--------------------|----------------|---------------------|--------------------------------|--------------------------------|--------------------------------|--------------|--------------|
| Frame Type =01h | Status =00h | Application Type | Track 1 Field (if found) | Track 2 Field (if found) | Track 3 Field (if found) | CRC (MSB) | CRC (LSB) |

Example 1: Payload, Card Read Successfully, Application Type Visa, Both Track 1 and Track 2 Present

```
[01] [00] [02]
%B123456789^ABCDEF^12345678?;123456=12345?<CRC1><CRC2>
```

Example 2: Payload, Card Read Successfully, Application Type MasterCard, Only Track 2

Present

```
[01] [00] [01] ;123456=12345?<CRC1><CRC2>
```

Example 3: Payload, Card Read Successfully, Application Type AmEx, Only Track 1 Present

```
[01] [00] [03] %B1234567^ABCDEF^12345678? <CRC1><CRC2>
```

Example 4: Payload, Card Read Successfully, Application Type Unknown, Both Track 1 and

Track 2 Present

```
[01] [00] [00] %B123456789^ABCDEF^12345678?;123456=12345?
<CRC1><CRC2>
```

Note: Only equivalent Track 1 and 2 Tag data from the CL card will be used to combine Track 1 and 2 Field of the burst data frame. Track 3 Field will only be presented in the Application Type 0x0C.

1.1.1.11. Payload Frame for CVN17 Enabled Readers

For MSD-only readers that require an online cryptogram (that is, TTQ = '80 80 00 00') MSD v1.4.2 and v2.0.2 transactions return the Burst Mode payload frame is described as follows (refer to Visa Contactless Payment V. 2.0.2 Including Additions and Clarifications 3.0 – August 2007):

| Byte 0 | Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte5 ... Byte 5+n-1 | Byte 5+n | Byte6+n |
|------------|-----------------|--------------------------------------|-------------------|-------------------|-------------------------|-----------|-----------|
| Frame Type | Status Code | Application Type | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| 03h | See table below | See Application Type | | | See Data Tables | | |

If the status code is OK the format and contents of the data field in the response frame are given in the following table. All TLV lengths in the TLV include the tag and length bytes.

Payload Frame with Cryptogram Data Format and Content When Status OK

| Data Item | Length (bytes) | Description |
|---|--------------------|---|
| Track 1 Length | 1 | If Track 1 is available, then this field gives the length of the Track 1 data that follows. If Track 1 is not available, then a Length of 00h is returned. Format: Binary |
| Track 1 Data (MagStripe card) | Variable | Track 1 data (if available). Format: ASCII (no null terminator) |
| Track 2 Length | 1 | If Track 2 is available, then this field gives the length of the Track 2 data that follows. If Track 2 is not available, then a length of 00h is returned. Format: Binary |
| Track 2 Data (MagStripe card) | Variable | Track 2 Data (if available). Format: ASCII (no null terminator) |
| DE055 (Clearing Record) Present | 1 | If a Clearing Record (DE 055) field is available, then this field is 01h. If there is no Clearing Record (DE 055) field, then this field is 00h. |
| TLV DE 055 (Clearing Record) | Variable | DE 055 data (if available) as a TLV data object encoded with Tag 'E1'. The ISO-8583 DE 055 data is the same data as is included in the Clearing Record. Tag: E1 Format: variable length group tag. |
| TLV App PAN | Variable, up to 12 | Application primary account number (PAN) as a TLV object. This field is present only if the DE 055 object is present. Tag: 5A Format: cn variable length up to 19 (10 bytes) |
| TLV PAN Seq Number | 4 | PAN sequence number as a TLV object. This field is present only if the DE 055 object is present. Tag: Format: n2, BCD encoded on 1 bytes |
| TLV Application Expiration Date | 6 | Application expiration date as a TLV object. This field is present only if the DE 055 object is present. Tag: 5F24 Format: n6, BCD encoded on 3 bytes (YYMMDD) |
| TLV Application Label | Variable, up to 18 | Application label as a TLV object. This field is present only if the DE 055 object is present. Tag: 50 Format: an variable length up to 16 bytes |
| TLV CVM Results | 6 | Cardholder verification method (CVM) results as a TLV object. This field is present only if the DE 055 object is present. Tag: 9F34 Format: b3 |
| TLV Data Authentication Code | 5 | Data authentication code as a TLV object. This field is present only if the DE 055 object is present. Tag: 9F45 Format: b2 |
| TLV ICC Dynamic Number | 11 | ICC dynamic number as a TLV object. This field is present only if the DE 055 object is present. Tag: 9F4C Format: b8 |
| TLV Track 1 Equivalent Data (M/Chip card) | 81 | Track 1 equivalent data as a TLV object. This field is present only if the DE 055 object is present. Tag: 56 Format: b79 |
| TLV Transaction Status Information | 4 | Transaction status information as a TLV object. This field is present only if the DE 055 object is present. Tag: 9B Format: b2 |

| Data Item | Length (bytes) | Description |
|---|------------------------------------|---|
| Cardholder Name | 29 | Cardholder name as a TLV object. This field is present only if the DE 055 object is present. Tag: 5F20 Format: b26 |
| Application Usage Control | 5 | Application usage control as a TLV object. This field is present only if the DE 055 object is present. Tag: 9F07 Format: b2 |
| Issuer Action Code(Default) | 8 | Issuer action code (Online) as a TLV object. This field is present only if the DE 055 object is present. Tag: 9F0D Format: b5 |
| Issuer Action Code(Denial) | 8 | Issuer action code (Denial) as a TLV object. This field is present only if the DE 055 object is present. Tag: 9F0E Format: b5 |
| Issuer Action Code(Online) | 8 | Issuer action code (Default) as a TLV object. This field is present only if the DE 055 object is present. Tag: 9F0F Format: b5 |
| TLV Auth Code | 9 | Authorization code as a TLV object Tag: E300 Format: b8 |
| TLV Track 2 Equivalent Data | 21 | Track 2 equivalent data as a TLV object. This field is present only if the DE 055 object is present or authorization code is present. Tag: 57 Format: b19 |
| VLP Issuer Auth Code | 9 | VLP Issuer authorization code as a TLV object Tag: 9F74 Format: b6 |
| Application Identifier | Variable up to 19 | AID as a TLV object Tag: 9F06 Format: variable b5...16 |
| Available Offline Spending Amount (Balance) | 9 | Available offline spending amount as a TLV object Tag: 9F5D Format: variable b6 |
| TLV Application Effective Date | 6 | Application effective date as a TLV object. Tag: 5F25 Format: n6, BCD encoded on 3 bytes (YYMMDD) |
| Form Factor Indicator | F: b 32 T: '9F6E' L: 4-bytes | Indicates the form factor of the consumer payment reader and the type of contactless interface over which the transaction was conducted. The Form Factor Indicator is both an implementation and Issuer option. Inclusion of the form factor indicator in online messages (and clearing records for offline capable readers) is an option for qVSDC and MSD readers. |
| PayPass Third Party Data | F: b T: '9F6E' L: 5-32 bytes | Priority information from a third party in the following format: Country Code according to ISO 3166-1 n3, 2 bytes Unique ID assigned by MasterCard b, 2 bytes Proprietary Data b 1 to 28 bytes |

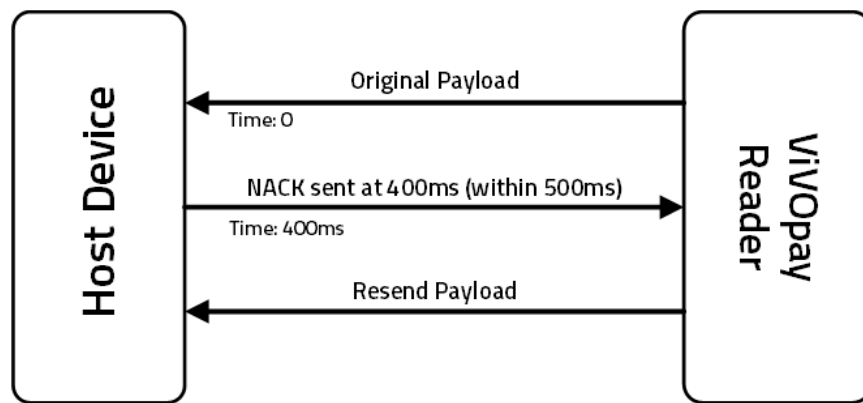
| Data Item | Length (bytes) | Description |
|-------------------------------|---|---|
| Customer Exclusive Data (CED) | F: b T: '9F7C' L: Var. up to 32-bytes | Contains data for transmission to the Issuer in MSD transactions with a cryptogram. Customer exclusive data is both an implementation and Issuer option. Inclusion of the customer exclusive data in online messages is an option for MSD readers compliant to this specification. Customer Exclusive Data shall be updateable via an Issuer script command. |

1.1.1.12. NACK Frame

If the host device fails to receive the card payload data, it can send a NACK frame and request the ViVOPay reader to resend the card payload data. To ensure that the reader resends the card payload data, the NACK frame must be received by the reader within 500ms after it sends the original card payload. If the reader receives the NACK frame within this time period, it resends the card payload data to the host. If the reader receives the NACK Frame after 500ms of sending the original card payload, or if a new card has been detected, the reader ignores the NACK frame and does not resend the payload data. Each payload data is only resent one time. The NACK frame is a 1-Byte code with value of 0x55.

| Byte 0 |
|-------------------|
| Frame Type =0x55h |

Example 1: The ViVOPay reader receives NACK frame from the host device within 500ms after sending the original payload data; the reader then resends the card payload data.

**Original Payload:**

Payload, Card Read Successfully, Application Type Master Card, Both Track 1 and Track 2 Present

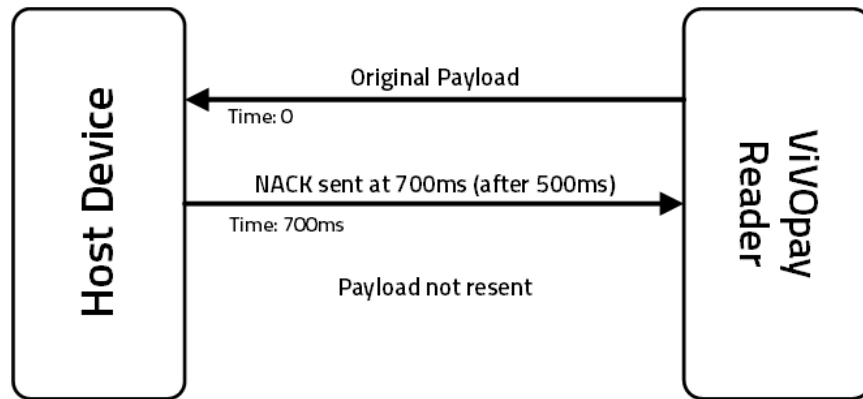
```
[01] [00] [01] %B5325350000623567^840SMITH/JOHN^05085011492563892473?;5325350000623567=05081019492993892483? <CRC1><CRC2>
```

Resent payload:

Payload, Card Read Successfully, Application Type Master Card, Both Track 1 and Track 2 Present

```
[01] [00] [01] %B5325350000623567^840SMITH/JOHN^05085011492563892473?;5325350000623567=05081019492993892483? <CRC1><CRC2>
```

Example 2: The reader receives NACK frame from the host device after 500ms of sending the original payload data; the reader does not resend the card payload data.



Original card payload data (no resent payload data):

Payload, Card Read Successfully, Application Type American Express, Both Track 1 and Track 2 Present

[01] [00] [03] %B379013539021002^TEST/CARD001^0604718000877840?; 379013539021002=060471800087784000102? <CRC1><CRC2>

1.1.1.13. Asynchronous UI Message Event

The asynchronous UI message event is used by the reader to indicate specific events to the host device. These frames are only sent when LCD and LED are to be sent to an external source.

In synchronizing with the transaction, the reader can send an asynchronous user interface (UI) message event to the host device to specify the required user experience on the host.

Asynchronous UI Message Event format:

| Byte 0 | Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte 5 & 6 | ... | Bn-3 | Bn-2 | Bn-1 | Bn |
|----------------------|----------------|--------------------|---|----------------|-----------------------|-----------------------------------|-------------------|-------------------|--------------|-----------|
| Frame Type 45h, E | Status =00h | Event Type 55h, | UI Scheme, defined on tag 'FF F8' | Length Byte | 2 Byte UI Event | Additional 2 Byte UI Events | Null Character | Null Character | CRC (MSB) | CRC (LSB) |

Byte 3 is the UI Scheme # that allows the user to have different user interfaces (LCD display message table, and buzzer/LED profiles).

Byte 4 is the length of the remainder of the frame, less CRC.

Bytes 5 & 6 are the UI Event consisting of component (LCD, LED, or Buzzer) and acts as defined below.

Asynchronous UI Message Event

| Component | UI Type | UI Status Definition |
|-----------|---------|--|
| LED | 01h | Higher nibble: LED # 00: LED0 01: LED1 02: LED2 03: LED3 FF: all Lower nibble: 00: Off 01: On 11: No change |
| Buzzer | 02h | Higher nibble: 1: short beeps 2: long beeps Lower nibble, short beep: 0: No change 1: Single beep 2: Double beep 3: Triple beep Lower nibble, long beep: 0: 200ms 1: 400ms 2: 600ms |
| LCD | 03h | LCD message index |

List of messages and the message flow for one user experience are given in [Appendix A.1](#).

1.1.1.14. Data Definitions**Status**

Status is a 1-Byte code that indicates success or contains an error code. This can have any value from 0 to 255.

Asynchronous UI Message Event Status

| Status | Value | Description |
|-----------------------------|-------|--|
| STATUS_OK | 00h | Card read completed successfully |
| STATUS_EC_CARD_REMOVED | 01h | A timeout occurred, card no longer present |
| STATUS_EC_COMM_ERROR | 02h | Some communication error occurred |
| STATUS_EC_PROTOCOL_ERROR | 03h | Protocol not respected |
| STATUS_EC_MULTIPLE_CARDS | 04h | Collisions were detected |
| STATUS_EC_CARD_NOT_ACCEPTED | 05h | Errors found in card information |
| STATUS_EC_BAD_DATA | 06h | Errors found in card information format |
| STATUS_EC_UNKNOWN_ERROR | FFh | Internal error |

Status never has a value that matches the Track 1 and Track 2 start/end sentinels.

Application Type

Application type is a 1 byte code that indicates the application type detected. This can have any value from 0 to 255.

Asynchronous UI message Event Application Type

| Application Type | Value |
|--|-------|
| Unknown | 00h |
| MasterCard | 01h |
| Visa | 02h |
| American Express | 03h |
| Discover | 04h |
| SpeedPass | 05h |
| Gift Card | 06h |
| Diners Club | 07h |
| EnRoute | 08h |
| JCB | 09h |
| ViVOCARD Diagnostic | 0Ah |
| HID card | 0Bh |
| MSR – Physical MSR, Application type unknown | 0Ch |
| Reserved for future use | 0Dh |
| DesFire (ViVOCARD3) Track Data | 0Eh |
| DesFire (ViVOCARD3) Raw Data | 0Fh |
| RBS | 11h |
| ViVOcomm | 14h |

Application type never has a value that matches the Track 1 and Track 2 start/end sentinels.

Track 1 Field

This is a variable length field consisting of Track 1 data as ASCII characters. This field starts with the Track 1 start sentinel '%' and ends with the Track 1 end sentinel '?'. If any Track 1 data is available, it is present between the start and end sentinel. For example:

```
%B123456789^ABCDEF^12345678?
```

Track 2 Field

This is a variable length field consisting of Track 2 data as ASCII characters. This field starts with the Track 2 start sentinel ';' and ends with the Track 2 end sentinel '?'. If any Track 2 data is available, it is present between the start and end sentinel. For example:

```
;12345678=12345?
```

Sample Output

```

45 00 55 00 04 03 03 00 00 E8 DD <-- LCD Event
45 00 55 00 04 01 11 00 00 28 B6 <-- LED Event
45 00 55 00 04 01 10 00 00 1F 86 <-- LED Event
45 00 55 00 04 02 20 00 00 41 FF <-- Buzzer Event
45 00 55 00 04 01 11 00 00 28 B6 <-- LED Event
45 00 55 00 04 01 21 00 00 ED 13 <-- LED Event
45 00 55 00 04 01 31 00 00 AE 70 <-- LED Event

01 00 0A 25 42 36 32 37 39 32 35 37 37 34 39 31
33 32 33 34 33 5E 54 45 53 54 20 43 41 52 44 2F
56 49 56 4F 54 45 43 48 5E 31 30 31 32 38 31 33
30 30 37 32 31 30 34 33 35 30 30 30 30 3F 3B 36
32 37 39 32 35 37 37 34 39 31 33 32 33 34 33 3D
31 30 31 32 38 31 33 30 30 37 32 31 30 34 33 35
30 30 30 30 3F B5 DC <-- Burst Mode Payload Frame

45 00 55 00 04 03 04 00 00 6D 4D <-- LCD Event
45 00 55 00 0C 01 30 00 00 01 20
00 00 01 10 00 00 53 78 <-- Three LED Events
45 00 55 00 04 03 01 00 00 86 BD <-- LCD Event

```

3.4 CRC Calculation

The 16-bit CRC value is based on CRC-16/CCITT and calculated based on the following parameter set.

Width: 16-bits
Polynomial $x^{16} + x^{12} + x^5 + 1$
Truncated Polynomial: 1021 hex
Initial Value: FFFF hex
Input Data: Not Reflected
Output CRC: Not Reflected
XOR of Output CRC: Not Done

The CRC-16 is calculated for the entire frame inclusive of frame tags, unused bytes, and so on.

For Protocol 1 (deprecated) and Protocol 2: The CRC of the command frame is little-endian; that is, lower byte first (LSB).
 The CRC of the response frame is big-endian; that is, higher byte first (MSB).
 For Pass-Through Frames, both command and response frames have the CRC stored in big-endian order (MSB first).

Some test values that can be used to test an implementation of this algorithm are given below.

Data String (ASCII Text): 123456789

CRC: 29B1h

Data (Hex): [01h] [02h] [03h] [04h] [05h]

CRC: 9304h

Data (Hex): [56] [69] [56] [4F] [74] [65] [63] [68] [00] [43] [18] [00] [00] [00]

CRC: A1F5h

Implementation Example

The following code snippet is an example of the CRC Calculation. The returned CRC would be stored in big-endian or little-endian form, depending on whether the Protocol 1, Protocol 2 or Pass-Through Mode was being used. This code was written in Microsoft Visual C++ 6.0.

```
// -----
// ID TECH
// ID TECH reserves the right to make changes without notice at any
// time. ID TECH makes no
// warranty, expressed, implied or statutory, including but not limited
// to any implied
// warranty of merchantability or fitness for any particular purpose, or
// that the use will
// not infringe any third party patent, copyright or trademark. ID TECH
// must not be liable
// for any loss or damage arising from its use.
// -----
```

```
static const unsigned short CrcTable[ 256 ] = {
0x0000, 0x1021, 0x2042, 0x3063, 0x4084, 0x50A5, 0x60C6, 0x70E7, 0x8108, 0x9129,
0xA14A, 0xB16B, 0xC18C, 0xD1AD, 0xE1CE, 0xF1EF, 0x1231, 0x0210, 0x3273, 0x2252,
0x52B5, 0x4294, 0x72F7, 0x62D6, 0x9339, 0x8318, 0xB37B, 0xA35A, 0xD3BD, 0xC39C,
0xF3FF, 0xE3DE, 0x2462, 0x3443, 0x0420, 0x1401, 0x64E6, 0x74C7, 0x44A4, 0x5485,
0xA56A, 0xB54B, 0x8528, 0x9509, 0xE5EE, 0xF5CF, 0xC5AC, 0xD58D, 0x3653, 0x2672,
0x1611, 0x0630, 0x76D7, 0x66F6, 0x5695, 0x46B4, 0xB75B, 0xA77A, 0x9719, 0x8738,
0xF7DF, 0xE7FE, 0xD79D, 0xC7BC, 0x48C4, 0x58E5, 0x6886, 0x78A7, 0x0840, 0x1861,
0x2802, 0x3823, 0xC9CC, 0xD9ED, 0xE98E, 0xF9AF, 0x8948, 0x9969, 0xA90A, 0xB92B,
0x5AF5, 0x4AD4, 0x7AB7, 0x6A96, 0x1A71, 0x0A50, 0x3A33, 0x2A12, 0xDBFD, 0xCBDC,
0xFBBF, 0xEB9E, 0x9B79, 0x8B58, 0xBB3B, 0xAB1A, 0x6CA6, 0x7C87, 0x4CE4, 0x5CC5,
0x2C22, 0x3C03, 0x0C60, 0x1C41, 0xEDAE, 0xFD8F, 0xCDEC, 0xDDCD, 0xAD2A, 0xBD0B,
0x8D68, 0x9D49, 0x7E97, 0x6EB6, 0x5ED5, 0x4EF4, 0x3E13, 0x2E32, 0x1E51, 0x0E70,
0xFF9F, 0xEFBE, 0xDFDD, 0xCFFC, 0xBF1B, 0xAF3A, 0x9F59, 0x8F78, 0x9188, 0x81A9,
0xB1CA, 0xA1EB, 0xD10C, 0xC12D, 0xF14E, 0xE16F, 0x1080, 0x00A1, 0x30C2, 0x20E3,
0x5004, 0x4025, 0x7046, 0x6067, 0x83B9, 0x9398, 0xA3FB, 0xB3DA, 0xC33D, 0xD31C,
0xE37F, 0xF35E, 0x02B1, 0x1290, 0x22F3, 0x32D2, 0x4235, 0x5214, 0x6277, 0x7256,
0xB5EA, 0xA5CB, 0x95A8, 0x8589, 0xF56E, 0xE54F, 0xD52C, 0xC50D, 0x34E2, 0x24C3,
0x14A0, 0x0481, 0x7466, 0x6447, 0x5424, 0x4405, 0xA7DB, 0xB7FA, 0x8799, 0x97B8,
0xE75F, 0xF77E, 0xC71D, 0xD73C, 0x26D3, 0x36F2, 0x0691, 0x16B0, 0x6657, 0x7676,
0x4615, 0x5634, 0xD94C, 0xC96D, 0xF90E, 0xE92F, 0x99C8, 0x89E9, 0xB98A, 0xA9AB,
0x5844, 0x4865, 0x7806, 0x6827, 0x18C0, 0x08E1, 0x3882, 0x28A3, 0xCB7D, 0xDB5C,
0xEB3F, 0xFB1E, 0x8BF9, 0x9BD8, 0xABBB, 0xBB9A, 0x4A75, 0x5A54, 0x6A37, 0x7A16,
0x0AF1, 0x1AD0, 0x2AB3, 0x3A92, 0xFD2E, 0xED0F, 0xDD6C, 0xCD4D, 0xBDAA, 0xAD8B,
```

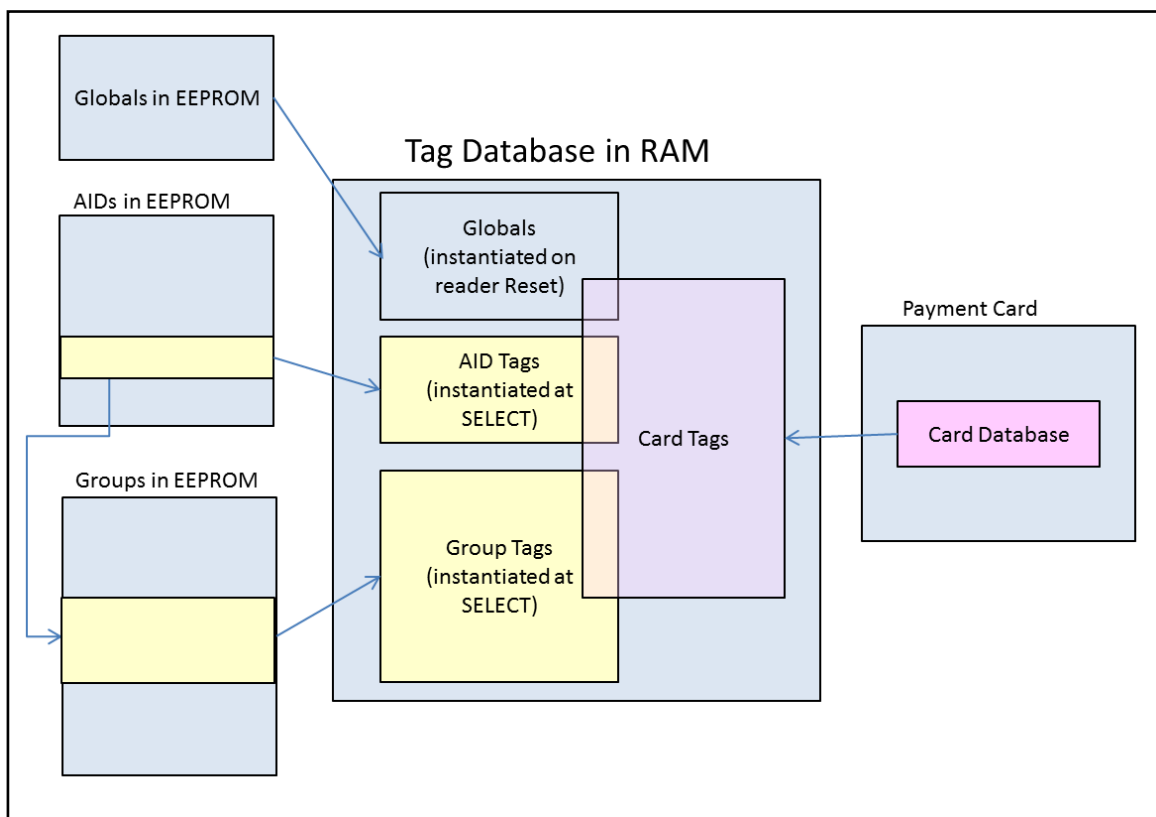
```
0x9DE8, 0x8DC9, 0x7C26, 0x6C07, 0x5C64, 0x4C45, 0x3CA2, 0x2C83, 0x1CE0, 0x0CC1,  
0xEF1F, 0xFF3E, 0xCF5D, 0xDF7C, 0xAF9B, 0xBFBA, 0x8FD9, 0x9FF8, 0x6E17, 0x7E36,  
0x4E55, 0x5E74, 0x2E93, 0x3EB2, 0x0ED1, 0x1EF0  
};
```

```
unsigned short CalculateCRC ( unsigned char *Buffer, unsigned int Len )  
{  
    unsigned short Crc = 0xffff;  
    while (Len--)  
    {  
        Crc = CrcTable[ ((Crc >> 8) ^ *Buffer++) ] ^ (Crc << 8);  
    }  
    return(Crc);  
}
```

4. Tag and Data Set Configuration

Tags are configured in the ViVOpay reader ahead of time so that when a card is selected, a "data set" (or group) may be instantiated for use in a transaction.

The following illustration shows the basic approach to instantiating the tag database for a transaction. Global variables (configured through the **Set Configuration** command) are instantiated when the reader is reset or powered up. When a card is in the field, an AID is selected and its tags are added to the database. The selection of the AID causes a "data set" (group) to be selected and its tags are added to the database. As the transaction proceeds, card tags are added to the database, possibly overwriting or updating some tags that were previously added to the database.



4.1 Configurable AIDs and Groups

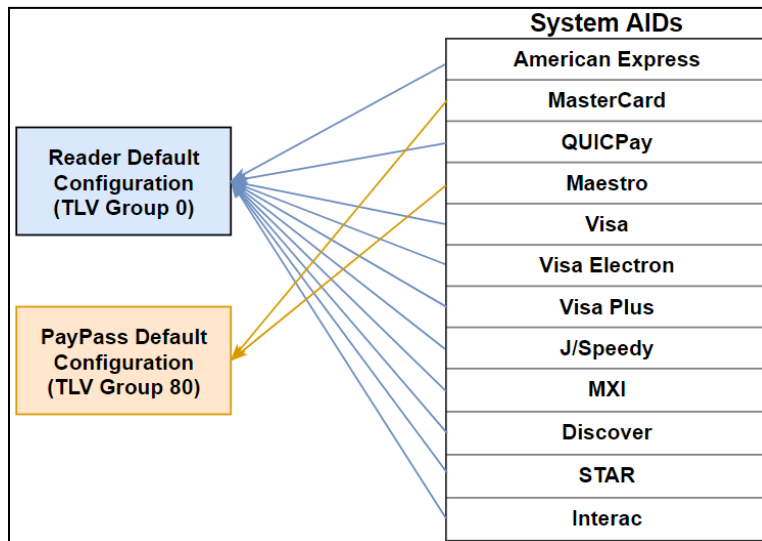
This section explains how to create and modify application identifiers (AIDs) and associate them with TLV Groups in the reader's memory for specific types of transaction handling. Detailed descriptions of the configurable application identifier (CONAID) commands are also included.

Each AID uniquely identifies a payment application. The reader has *default* AIDs that are preconfigured to support common payment applications. These AIDs are called **System AIDs** and they can be modified or disabled, but not deleted. Readers also support up to eight user-defined AIDs called **User AIDs**.

Each AID must be associated with a **TLV Group** that defines transaction processing for that payment application. The System AIDs are initially associated with a default TLV Group (called Group 0), which can be modified but not deleted. User AIDs can be associated to the default TLV Group or any of seven other user-defined TLV Groups.

With the implementation of M/Chip 3.0, an additional *default* TLV Group (Group 80) has been added. M/Chip 3.0 *does not* use the Reader Default Group (Group 0).

All AIDs must be unique. The reader's default configuration is based on System AIDs and two default groups. All of the System AIDs (except PayPass AIDs, as noted above) initially refer to the default TLV Group 0. The diagram below shows the default reader AID configuration.



The ViVOpay reader's configurable application identifiers feature allows users to create and customize AIDs and the TLV Groups associated with them. Each AID may have characteristics that are unique and different from the reader's default System AIDs and TLV Group configuration.

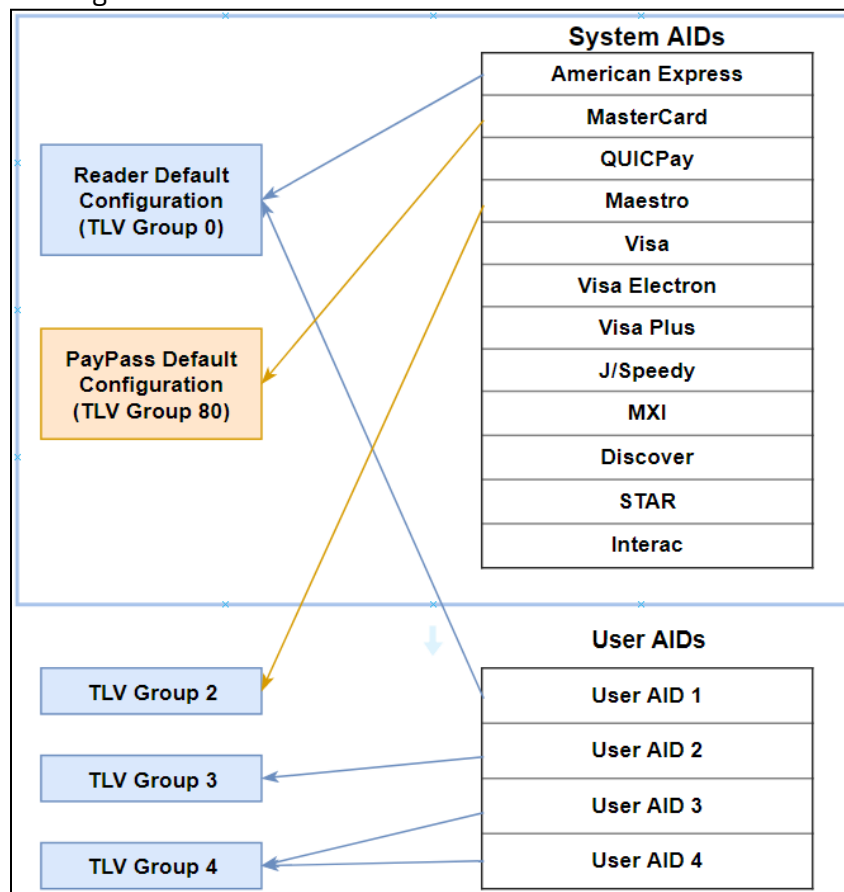
To create a new configurable AID, send to the reader both:

- the AID in question.
- the TLV Group to associate with it.

If the AID already exists in the reader's memory, it will modify the AID accordingly. When sent a new AID, the reader creates and saves the new AID. Multiple AIDs can be associated with the same TLV Group, or they can map to unique TLV Groups. Users may also redefine the functionality for an *existing* AID by linking it to a new configuration Group, or they may disable an AID if to prevent the reader processing transactions from that payment application. Users may *delete* an AID by communicating to the reader the AID number with no parameters.

As users add or modify AIDs and TLV Groups, the reader remembers all changes on subsequent boot up.

The diagram below shows an example of a reader's AID configuration after it has been modified with Configurable AIDs.



In the above example:

- Ten System AIDs have been disabled and four User AIDs and three new TLV Groups have been configured.
- The new AID User AID – 1 has been linked to the Reader Default Configuration (TLV Group 0) so that it functions as the other System AID 1 functions.
- The Maestro AID has been linked to the user-defined TLV Group – 2.
- User AID – 2 functions as defined in the new TLV Group – 3.
- Both User AID – 3 and User AID – 4 point to the new TLV Group – 4 and function accordingly.

Also notice that the other System AIDs have been disabled by removing their link to a configuration group.

Use the configurable AID commands to create new AIDs or change configuration values for an AID. Use the Configurable Group commands to create new groups or configuration values for a group.

4.1.1 System AIDs

A System AID is an AID preloaded for a specific application using a known AID value. Examples include MasterCard, American Express, and Visa. The table below shows all the System AIDs.

System AIDs

| Application Name | Application Identifier |
|------------------|-------------------------------|
| American Express | A0 00 00 00 25 01 |
| MasterCard | A0 00 00 00 04 10 10 |
| Maestro | A0 00 00 00 04 30 60 |
| Visa | A0 00 00 00 03 10 10 |
| Visa Electron | A0 00 00 00 03 20 10 |
| Visa Interlink | A0 00 00 00 03 30 10 |
| Visa Plus | A0 00 00 00 03 80 10 |
| J/Speedy | A0 00 00 00 65 10 10 |
| Discover | A0 00 00 03 24 10 10 |
| Discover | A0 00 00 01 52 30 10 |
| Interac | A0 00 00 02 77 10 10 |
| Smart Tap2.1 | A0 00 00 04 76 D0 00 01 11 |
| AppleVas | 4F 53 45 2E 56 41 53 2E 30 31 |

The host device:

- may disable a System AID.
- may ONLY modify some of the System AID properties.
- may NOT delete a System AID.

4.1.2 User AIDs

A User AID is an optional AID that is added and/or configured by the user. These AIDs are used for servicing transactions that are not defined by one of the System AIDs. This determination needs to be made by the integrator.

The host device:

- may modify ANY User AID property.
- may delete a User AID.

There is no equivalent to the System AID disable; the User AID either exists, and it is used for its associated transactions, or the User AID is not present.

4.1.3 Card Brands and AIDs by Group

| Application | AID | Group |
|-----------------------|----------------------|-------|
| | System Configuration | 0 |
| MasterCard | A0 00 00 00 04 10 10 | 80 |
| Maestro | A0 00 00 00 04 30 60 | 80 |
| Visa | A0 00 00 00 03 10 10 | 90 |
| Visa Electron | A0 00 00 00 03 20 10 | 90 |
| Visa Interlink | A0 00 00 00 03 30 10 | 90 |
| Visa Plus | A0 00 00 00 03 80 10 | 90 |
| Amex | A0 00 00 00 25 01 | A0 |
| Discover ZIP | A0 00 00 03 24 10 10 | B0 |
| Discover D-PAS | A0 00 00 01 52 30 10 | B0 |
| Interac | A0 00 00 02 77 10 10 | C0 |
| I/Speedy | A0 00 00 00 65 10 10 | D0 |

4.1.4 Reader Default TLV Group

The reader is provided with a default TLV Group (Group 0) that defines all the properties (with TLVs) required for a basic transaction. By default, all System AIDs except PayPass System AIDs (MasterCard and Maestro) use TLV Group 0 to define their transaction processing. By default, MasterCard PayPass System AIDs will use Group 80.

NOTE: For an exhaustive list of Groups and their TLV defaults, see the Appendix called [NEO 2 Contactless Groups and Default TLVs](#).

When configuring Groups, remember that configurations:

- MUST ALWAYS include the Group Number TLV as the FIRST TLV in the **Set Configurable Group** message.
- MUST define AT LEAST ONE TLV in addition to the Group Number TLV (in a [Set Configurable Group](#) command).
- May modify ANY TLVs in TLV Group 0.
- May NEVER delete TLV Group 0.

Unlike all other groups, the TLVs in the Default TLV Group (TLV Group 0) are constant. The reader ALWAYS uses the latest copy of the TLV. If a user issues a [Set Configurable Group](#) command that only updates some TLVs in TLV Group 0, the reader continues to use older versions of the TLVs that were not updated.

After each transaction, the reader reloads the default values from TLV Group 0 prior to the next transaction. For this reason, TLV Group 0 maintains a copy of ALL TLVs that can be entered into a TLV Group structure².

Warning: Changing values in TLV Group 0 should be done with **EXTREME CAUTION**, because this affects the default configuration that most (except PayPass) transactions use.

4.1.5 PayPass Default Group

The PayPass default group is Group 80. PayPass M/Chip 3.0 *does not use Group 0* tag definitions (not even for default values).

The process of instantiating a PayPass database is slightly different from other applications:

- *Group 0 tags are not loaded.*
- 28 default tags defined in the *EMV Contactless Book C-2 Kernel 2 Spec v2.3* are initialized with their specified default values. See [PayPass Group Configuration TLVs with Hard-Coded Values in Kernel](#).
- PayPass Group tags are loaded. (Group 80 is the default group for PayPass applications).
- Tags sent in the Activate Command are loaded into the database.

4.1.6 User-Defined TLV Groups

There are seven undefined TLV Groups in the reader at startup. These groups can be used for any purpose.

The user:

- MUST ALWAYS include the Group Number TLV as the FIRST TLV in the message.
- MUST include AT LEAST ONE TLV other than the Group Number TLV (in a [Set Configurable Group](#) command).
- May modify ANY TLV in the TLV Group.
- May ALWAYS delete a TLV Group 80 through 7.
- SHOULD NEVER include the TDOL TLV if its length = zero (that is, only include the TDOL if it has a value).

² PayPass specific tags are an exception to this rule. Those are maintained in Group 80.

User-defined TLV Groups differ from the default TLV Group 0 in two important ways. First, these groups only need to contain TLVs that are different than the TLVs in the default TLV Group 0. Thus they are normally a sub-set of the TLVs in the default group.

For American Express:

- Transaction limit (FFF1)
- CVM limit (FFF5)
- Floor limit (9F1B)

For Discover:

- Transaction limit (FFF1)
- CVM limit (FFF5)
- Floor limit (9F1B)
- Risk flags (FFF4)

If user-defined TLV Group is used, user should set these above TLVs in the user-defined Group. If not set, these above TLVs will be regarded as not existing.

For American Express, **Terminal Capabilities (9F33)** and Enhanced Expresspay **Terminal Capabilities (9F6E)** are expected to be set consistently. For example, if Byte 2 bit 7 of '9F6E' is set to 1(b) to indicate 'Online PIN Supported' then '9F33' byte 2 bit 7 should also be set to 1(b).

Secondly, the TLVs in TLV Groups 1 through 7 are not permanent. If a user configures a TLV Group and then issues a second [Set Configurable Group](#) command on the same TLV Group, the second Set Configurable Group command overwrites EVERY change to the TLV Group made by the first command.

Warning: Changing values in TLV Groups 1 through 7 overwrites all content in the TLV Group, including deleting TLVs not in the update.

Except for MasterCard PayPass transactions, when one of these user-defined TLV Groups is selected during a transaction, the reader uses the TLVs included in the group AND any other TLVs required for the transaction are taken from the default Group 0. After the reader has finished transaction processing, it reloads TLV Group 0 values for all TLVs. It is now ready to commence the next transaction.

There are some guidelines for setting and deleting TLV Groups listed below. Most of these guidelines are intuitive (for example, a user *may not* delete a TLV Group if an AID exists that currently uses it).

4.1.7 Configurable AID Reader Memory Requirement

The Configurable AIDs feature requires memory to store TLV groups and User AIDs. ViVOpay readers use 64K flash memory to support the Configurable AID feature. Refer to the reader's user documentation for more information on reader memory.

4.1.8 ViVOpay Proprietary TLVs

TLVs may be either standard TLVs or proprietary TLVs. *Standard TLVs* are defined by EMV and the Payment Association Requirements and recognized by everyone. *Proprietary TLVs* are created by individual payment associations and reader manufacturers for specific functions. Proprietary TLVs must be handled in a manner that isolates them from other proprietary TLVs. For a complete listing of all ID TECH proprietary tags and their meanings, you should consult the document P/N 80000503-001, *ID TECH TLV Tag Reference Guide.pdf*, available on the [ID TECH Knowledge Base](#).

ViVOpay proprietary TLVs can be present with standard TLVs without encapsulation when the command is processed exclusively by ViVOpay firmware or software. If the TLVs will be processed by other devices, ViVOpay proprietary TLVs must be encapsulated to prevent conflicts with proprietary TLVs from other organizations.

ViVOpay TLV Group Tag FFEE01 is used to encapsulate ViVOpay proprietary TLVs.

Example

The following example is for an encapsulated "Terminal Capabilities – CVM Required" TLV. The TLV string "FFEE0106DF29030101" is broken down as follows:

FFEE01 ViVOpay TLV Group Tag
06 Length of all encapsulated TLVs
DF29 Tag Host Capabilities – CVM Required – ViVOpay Proprietary
03 Length of Transaction CVM
00 01 00 Value: Actual Transaction CVM

4.2 Card Application Proprietary Tag List (FF69 or FFEE69)

Some applications may require list of defined proprietary tags that may be returned in Data Object Lists (DOLs). To accomplish this, the reader allows each user-defined Group (except Group 0) to define a list of proprietary tags that can be inserted into the tag database. The maximum size of this list is 32 bytes. The new tag used for encapsulating the proprietary tag list is FF69 (or FFEE69).

A tag in this list may be configured in one of two ways:

- **Constant Value:** the TLV contains a non-zero length and a value. The reader will not modify this value, but it can be provided when requested (as in a DOL).

- **Updateable:** the TLV contains a length of zero and no value. The tag is then "defined" but has no value, so it may be updated during the transaction. At the end of a transaction, the reader will send any updated proprietary tags back in the activate response frame.

4.3 Configuration Tag Tables

4.3.1 Group Configuration Tags

The following table contains TLVs that are configurable using the **Set Configuration (04-00)** command. These TLVs are global within in the reader.

If a Group does not define some of these TLVs, the values in Group 0 will be used.

Group Configuration TLVs

| Tag | Data Object Name and Description | Format | Length (Bytes) | Default Value |
|---------------|--|--------|----------------|-------------------------|
| DFEE2D | Group Number The group number assigned to this group of parameters. AIDs may be associated with the group number. This tag is mandatory when getting or setting group parameters and it must be the 1 st TLV in Data Field. It is used as the "key" for the group parameter set. | n2 | 1 | 00 |
| 9C | Transaction Type Indicates the type of financial transaction, represented by the first two digits of ISO 8583:1987 Processing Code (default = purchase goods or services). | n2 | 1 | 00 |
| 5F2A | Transaction Currency Code Indicates the currency code of the transaction according to ISO 4217. Make sure you use the same Transaction Currency Code for all configurable AIDs (default = US Dollars). | n3 | 2 | 08 40 |
| 5F36 | Transaction Currency Exponent Indicates the implied position of the decimal point from the right of the transaction amount represented according to ISO 4217 (decimal is two places from right of the transaction amount). | n1 | 1 | 02 |
| 9F02 | Amount Authorized (Numeric) | n12 | 6 | 00 00 00 00 00 01 |
| 9F03 | Amount Other (Numeric) | n12 | 6 | 00 00 00 00 00 00 |
| 9F09 | Application Version Number PayPass M/Chip (Value = 00 02) D-PAS (Value = 00 02) Interac (Value = 00 02) Amex (Value = 00 01) | b | 2 | 00 02 |

| Tag | Data Object Name and Description | Format | Length (Bytes) | Default Value |
|---------------|---|--------|----------------|--|
| 9F15 | Merchant Category Code Classifies the type of business being done by the merchant, see ISO 8583:1993. | n4 | 2 | 00 00 |
| 9F1A | Host Country Code Indicates the country code of the host, represented according to ISO 3166. | n3 | 2 | 08 40 |
| 9F40 | Additional Host Capabilities Indicates the data input and output capabilities of the host device. | b | 5 | 60 00 00 30 00 |
| 9F53 | Transaction Category Code This is a data object defined by MasterCard which indicates the type of transaction being performed, and which may be used in card risk management. | an | 1 | 00 |
| 9F66 | Host Transaction Qualifier (TTQ) Determine the type of transaction (MSD, qVSDC, and Contactless VSDC) and whether online processing is supported. | b | 4 | 30 00 40 00 |
| 9F6D | Application Version Number (MagStripe) PayPass magstripe (Value = 00 01) | b | 2 | 00 01 |
| 9F7C | Merchant Custom Data | b | 20 | 00 |
| DFED59 | First Response Control Controls the first response 0x63 when starting a contact transaction. 0: Disable (Send First Response) 1: Enable (Don't Send First Response) | b | 1 | N/A |

| | | | | | | | | | |
|--|------------------|------|-----------|-----------|-------|----|----|-----------------------------------|----------------------|
| DFED5A | UI Event Control | | | | | | | | |
| | TAG | Len | B1 | B2 | B3-B8 | | | | |
| | DFED5A | 0X08 | see below | see below | RFU | | | | |
| | Byte 1: | | | | | | | | |
| | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 | Description |
| | | | | | | | | x | "Authorizing" (0) |
| | | | | | | | x | | "\$1.00" (1) |
| | | | | | | x | | | "Timeout" (2) |
| | | | | | x | | | | "Cancelled" (3) |
| | | | | x | | | | | "Enter PIN" (4) |
| | | x | | | | | | "Beep on CT completion" (5) | |
| | x | | | | | | | "See PINPAD" (6) | |
| x | | | | | | | | "Declined" (7) | |
| Byte 1: (0) ALL LCD, during 62-01 get online PIN 0: do not display "Authorizing.." 1: display "Authorizing." (1) ALL LCD 0: do not hide \$1.00 1: hide \$1.00 (2) ALL LCD 0: do not hide "Timeout" 1: hide "Timeout"ut" (3) ALL LCD 0: do not hide "Cancelled" 1: hide "Cancelled" (4) Host LCD only 0: do not notify Host "Enter PIN " 1: notify Host "Enter PIN"" (5) When CT transaction is completed; Not currently available 0: do not beep 1: beep (6) Host LCD only 0: do not notify Host "See PINPAD" 1: notify Host "See PINPAD" | | | | | | | | | b |

| | <p>(7) ALL LCD, CT only, FastEMV/Quick Chip; online decline only 0: do not hide "Declined" 1: hide "Declined"</p> <p>Byte 2:</p> <table><tr><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th><th>Description</th></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>x</td><td>"Welcome" (0)</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td><td>x</td><td></td><td>"Welcome" (1)</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td>x</td><td></td><td></td><td>"Welcome" (2)</td></tr><tr><td></td><td></td><td></td><td></td><td>x</td><td></td><td></td><td></td><td>PINBYPASS (3)</td></tr><tr><td></td><td></td><td></td><td>x</td><td></td><td></td><td></td><td></td><td>Thousand Separator (4)</td></tr><tr><td></td><td></td><td>x</td><td></td><td></td><td></td><td></td><td></td><td>"Authorizing.." (5)</td></tr><tr><td></td><td>x</td><td></td><td></td><td></td><td></td><td></td><td></td><td>"Authorizing.." (6)</td></tr><tr><td>x</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>"Do Not Remove Card" (7)</td></tr></table> | | | | | | | | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 | Description | | | | | | | | x | "Welcome" (0) | | | | | | | x | | "Welcome" (1) | | | | | | x | | | "Welcome" (2) | | | | | x | | | | PINBYPASS (3) | | | | x | | | | | Thousand Separator (4) | | | x | | | | | | "Authorizing.." (5) | | x | | | | | | | "Authorizing.." (6) | x | | | | | | | | "Do Not Remove Card" (7) | | | |
|----|--|----|----|----|----|----|----|--------------------------------|----|----|----|----|----|----|----|----|-------------|--|--|--|--|--|--|--|---|------------------|--|--|--|--|--|--|---|--|------------------|--|--|--|--|--|---|--|--|------------------|--|--|--|--|---|--|--|--|------------------|--|--|--|---|--|--|--|--|------------------------------|--|--|---|--|--|--|--|--|------------------------|--|---|--|--|--|--|--|--|------------------------|---|--|--|--|--|--|--|--|--------------------------------|--|--|--|
| b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | x | "Welcome" (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | x | | "Welcome" (1) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | x | | | "Welcome" (2) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | x | | | | PINBYPASS (3) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | x | | | | | Thousand Separator (4) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | x | | | | | | "Authorizing.." (5) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | x | | | | | | | "Authorizing.." (6) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | | | | | | | | "Do Not Remove Card" (7) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | <p>Byte 2: (0) Host LCD only 0: do not hide "Welcome" 1: hide "Welcome" (1) Aux LCD, L100, RT1050 0: do not hide "Welcome" 1: hide "Welcome" (2) Internal LCD, 0: do not hide "Welcome" 1: hide "Welcome" (3) PINPAD 0: allow press "Enter" w/o any input 1: do not allow press "Enter" w/o any input (4) Thousand Separator in Amount String 0: do not display thousand separator 1: display thousand separator (5) After Swiping a non-financial card 0: do not hide "Authorizing..." 1: hide "Authorizing..." (6) After Swiping a financial card 0: do not hide "Authorizing..." 1: hide "Authorizing..." (7) When displaying "Processing..." during EMV CT 0: don't display "Do Not Remove Card"</p> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Tag | Data Object Name and Description | Format | Length (Bytes) | Default Value | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------------|---|-----------|------------------------|-------------------------|--------|----|----|---------------------|----|-------------|--|--|--|--|--|--|--|---|---------------------|--|--|--|--|--|--|---|--|-------------------|---|---|---|---|---|---|--|--|-----|--|--|--|
| | <p>1: display "Do Not Remove Card"</p> <p>Byte 3:</p> <table><tr><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th><th>Description</th></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>x</td><td>"Amount String" (0)</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td><td>x</td><td></td><td>"Bootup Beep" (1)</td></tr><tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td></td><td></td><td>RFU</td></tr></table> <p>Byte 3: (0) For Amount string from tag 9F02 0: do not hide the amountstring 1: hide the amountstring (1) Buzzer 0: Beep during bootup as normal 1: Mute bootup beep</p> <p>Byte 4 - Byte 8: RFU</p> | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 | Description | | | | | | | | x | "Amount String" (0) | | | | | | | x | | "Bootup Beep" (1) | x | x | x | x | x | x | | | RFU | | | |
| b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | x | "Amount String" (0) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | x | | "Bootup Beep" (1) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | x | x | x | x | x | | | RFU | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DFEE28 | <p>Host Capabilities – No CVM Required M/Chip v2.0 element indicating the host capabilities to be used when amount, authorized < cvm required limit. Formatted as host capabilities (tag '9F 33')</p> <p>Only byte 2 of this tag is actually used. The other host capabilities are configured using tag 9F33.</p> <table><tr><th>DF28 Byte</th><th>PayPass Tag Equivalent</th></tr><tr><td>2</td><td>DF8119</td></tr></table> | DF28 Byte | PayPass Tag Equivalent | 2 | DF8119 | b | 3 | 00 08 E8 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DF28 Byte | PayPass Tag Equivalent | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2 | DF8119 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DFEE29 | <p>Host Capabilities – CVM Required M/Chip v2.0 element indicating the host capabilities to be used when amount, authorized >= cvm required limit. Formatted as host capabilities (tag '9F 33')</p> <p>Only byte 2 of this tag is actually used. The other host capabilities are configured using tag 9F33.</p> <table><tr><th>DF29 Byte</th><th>PayPass Tag Equivalent</th></tr><tr><td>2</td><td>DF8118</td></tr></table> | DF29 Byte | PayPass Tag Equivalent | 2 | DF8118 | b | 3 | 00 68 E8 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DF29 Byte | PayPass Tag Equivalent | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2 | DF8118 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DFEE34 ^[1] | <p>Host Contactless Transaction Limit Indicates the host limit for this AID for contactless transactions.</p> | n12 | 6 | 00 00 00 01 00 00 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| | | | | | | | | | | | | |
|-----------------------|------------------------|----|----|----|----|----|----|----|---|---|-------------|--|
| DFEE35 ^[1] | Visa Reader Risk Flags | | | | | | | | b | 3 | 00 06 01 | |
| | Byte 1 | | | | | | | | | | | |
| | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | | | | Meaning (0 = disable, 1 = enable) |
| | - | - | - | - | - | - | - | X | | | | Status Check |
| | X | X | X | X | X | X | X | - | | | | RFU |
| | Byte 2: | | | | | | | | | | | |
| | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | | | | Meaning (0 = disable, 1 = enable) |
| | - | - | - | - | - | - | - | X | | | | Transaction Limit Check |
| | - | - | - | - | - | - | X | - | | | | CVM Required Limit Test |
| | - | - | - | - | - | X | - | - | | | | Host Floor Limit Check |
| | - | - | - | - | X | - | - | - | | | | Cash Transaction Reader Risk (RR) |
| | - | - | - | X | - | - | - | - | | | | Cashback Reader Risk (RR) |
| | - | - | X | - | - | - | - | - | | | | DRL (Dynamic Reader Limits) RR |
| | - | X | - | - | - | - | - | - | | | | Processing Restriction for Manual Cash |
| | X | - | - | - | - | - | - | - | | | | Processing Restriction for Cashback |
| | Byte 3 | | | | | | | | | | | |
| | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | | | | Meaning (0 = disable, 1 = enable) |
| | - | - | - | - | - | - | - | X | | | | 1 = online cryptogram required for zero amount (only used with zero amount |

| Tag | Data Object Name and Description | | | | | | | | | Format | Length (Bytes) | Default Value | |
|-----------------------|---|---|---|---|---|---|---|---|-------------------------------|--------|----------------|---------------|-------------------------|
| | | | | | | | | | check enabled) | | | | |
| | - | - | - | - | - | - | X | - | 1 = perform zero amount check | | | | |
| | X | X | X | X | X | X | - | - | RFU | | | | |
| | For example: 0x00 = Zero Amount check disabled. 0x01 = Zero Amount check is disabled and online cryptogram required bit will not be checked. (default) 0x02 = Zero Amount check enabled. 0x03 = Zero Amount check enabled and Option 1, online cryptogram required | | | | | | | | | | | | |
| DFEE36 ^[1] | CVM Required Limit Indicates the CVM required limit in the host <i>for the associated AIDs</i> . | | | | | | | | | | n12 | 6 | 00 00 00 00 80 00 |
| DFEE37 | UI Scheme Value = 00:ViVOpay User Interface (default) Value = 02:Visa Wave User Interface Value = 03:EMEA User Interface <i>Note:</i> LCD Messages need to be configured separately. Warning: EMEA UI is intended for use in the EMV or European environment, where the reader vend is not allowed to poll continuously (for example, to operate in Auto Poll Mode). The reader vend does NOT support Auto Poll while in EMEA UI mode. The reader is not certified to work properly in this situation. | | | | | | | | | | b | 1 | 03 |
| DFEE38 ^[1] | Language Option for LCD display Value = 00: English only display (default) Value = 01: Chinese only display ^[2] Value = 02: English & Chinese display ^[2] Value = 03: French only display Value = 04: Other Language (if ILM present) ^[2] Value = 05: English & French display ^[4] Value = 06: Japanese Value = 07: Spanish Value = 08: Turkish Note: The supported value may vary in different products; status code 0x06 is returned for unsupported values. | | | | | | | | | | b | 1 | 00 |
| DF8122 ^[1] | Host Action Code (Online) Reflect the acquirer-selected action to be taken upon analysis of the TVR. | | | | | | | | | | b | 5 | F8 50 AC F8 00 |
| DF8120 ^[1] | Host Action Code (Default) Reflect the acquirer-selected action to be taken upon analysis of the TVR. | | | | | | | | | | b | 5 | F8 50 AC A0 00 |

| Tag | Data Object Name and Description | Format | Length (Bytes) | Default Value | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|--------|----------------|----------------------------------|----|----|----|-------------------------|-----------------------------------|-----------------------------------|---|---|---|---|---|---|---|---|---------------------------|---|---|---|---|---|---|---|---|--------------------------|---|---|---|---|---|---|---|---|-----------------|---|---|---|---|---|---|---|---|-------------------------|---|---|---|---|---|---|---|---|------------------------------|---|---|---|---|---|---|---|---|-------------|---|---|---|---|---|---|---|---|-------------------------|---|---|-------|---|---|---|---|---|---|---|---|------------------|
| DF8121 ^[1] | Host Action Code (Denial) Reflect the acquirer-selected action to be taken upon analysis of the TVR. | b | 5 | 00 00 00 00 00 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DFEE44 ^[1] | Application Capability (1:Support,0:Not Support) <i>Byte 1: (Leftmost)</i> <table><tr><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>Meaning (0 = disable, 1 = enable)</th></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>X</td><td>Normal J/Speedy support</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td><td>X</td><td></td><td>ViVOpay Mifare for NFC</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td>X</td><td></td><td></td><td>Interac support</td></tr><tr><td></td><td></td><td></td><td></td><td>X</td><td></td><td></td><td></td><td>CUP support</td></tr><tr><td></td><td></td><td></td><td>X</td><td></td><td></td><td></td><td></td><td>Google Pay Smart Tap support</td></tr><tr><td>X</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>MIR Support</td></tr><tr><td></td><td>X</td><td>X</td><td>X</td><td></td><td></td><td></td><td></td><td>RFU</td></tr></table> | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning (0 = disable, 1 = enable) | | | | | | | | X | Normal J/Speedy support | | | | | | | X | | ViVOpay Mifare for NFC | | | | | | X | | | Interac support | | | | | X | | | | CUP support | | | | X | | | | | Google Pay Smart Tap support | X | | | | | | | | MIR Support | | X | X | X | | | | | RFU | b | 2 | 95 FF | | | | | | | | | |
| | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning (0 = disable, 1 = enable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | X | Normal J/Speedy support | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | X | | ViVOpay Mifare for NFC | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | X | | | Interac support | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | X | | | | CUP support | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | X | | | | | Google Pay Smart Tap support | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | X | | | | | | | | MIR Support | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | X | X | X | | | | | RFU | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | <i>Byte 2:</i> <table><tr><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>Meaning (0 = disable, 1 = enable)</th></tr><tr><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>X</td><td>MasterCard Credit support</td></tr><tr><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>X</td><td>-</td><td>American Express support</td></tr><tr><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>X</td><td>-</td><td>-</td><td>Visa support</td></tr><tr><td>-</td><td>-</td><td>-</td><td>-</td><td>X</td><td>-</td><td>-</td><td>-</td><td>Mobile J/Speedy support</td></tr><tr><td>-</td><td>-</td><td>-</td><td>X</td><td>-</td><td>-</td><td>-</td><td>-</td><td>ViVOWallet support</td></tr><tr><td>-</td><td>-</td><td>X</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>RBS support</td></tr><tr><td>-</td><td>X</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>MasterCard Cash support</td></tr><tr><td>X</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>Discover support</td></tr></table> | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning (0 = disable, 1 = enable) | - | - | - | - | - | - | - | X | MasterCard Credit support | - | - | - | - | - | - | X | - | American Express support | - | - | - | - | - | X | - | - | Visa support | - | - | - | - | X | - | - | - | Mobile J/Speedy support | - | - | - | X | - | - | - | - | ViVOWallet support | - | - | X | - | - | - | - | - | RBS support | - | X | - | - | - | - | - | - | MasterCard Cash support | | | | X | - | - | - | - | - | - | - | Discover support |
| | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning (0 = disable, 1 = enable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | - | - | - | - | - | - | - | X | MasterCard Credit support | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | - | - | - | - | - | - | X | - | American Express support | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | - | - | - | - | - | X | - | - | Visa support | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | - | - | - | - | X | - | - | - | Mobile J/Speedy support | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| - | - | - | X | - | - | - | - | ViVOWallet support | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| - | - | X | - | - | - | - | - | RBS support | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| - | X | - | - | - | - | - | - | MasterCard Cash support | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| X | - | - | - | - | - | - | - | Discover support | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Example: 0009 means reader support both MasterCard and Mobile J/Speedy applications | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DFEE5C | RF Deactivate Period | b | 4 | 01 50 00 00 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DFEE6B | Host IFD | an | 8 | 00 00 00 00 00 00 00 00 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DFEE70 | Loyalty Program ID | b | 1 | 00 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DFEE71 | Value Added Tax 1 | b | 2 | 00 00 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DFEE72 | Value Added Tax 2 | b | 2 | 00 00 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DFEE73 | Merchant Category Code | b | 2 | 00 00 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

[illegible]

[1] These objects use proprietary tags. The use of these tags should be restricted to the serial interface. After the reader has returned an OK response frame, the host application should dispose of the tags to avoid conflicts with other proprietary TLVs. Note that two-byte tags beginning in 'FF' have largely been superseded by 3-byte tags beginning in 'DF'; See the discussion under [2-Byte Tags vs. 3-Byte Tags](#).

^[2] These objects only work on the ViVOpay graphic reader.

4.3.2 Master Group Configuration TLVs

Master uses Group 80 as default Group. The [Set Configurable Group](#) command should be used to set the TLVs in this section. The following table in this section shows TLVs in the PayPass Configuration Group.

PayPass Default Group Configuration TLVs

| Tag | Description | Format | Length | Default Value in Group 80 |
|---------------|---|--------|--------|---------------------------|
| DFEE2D | Group Number The group number assigned to this group of parameters. AIDs may be associated with the group number. <div style="border: 2px solid red; background-color: yellow; padding: 5px; margin-top: 5px;"> This tag is mandatory when getting or setting group parameters and it must be the 1st TLV in Data Field. It is used as the "key" for the group parameter set. </div> | n2 | 1 | 80 |
| 5F57 | Account Type | B | 1 | |
| 9F01 | Acquirer Identifier | n12 | 6 | |
| 9F40 | Additional Host Capabilities Indicates the data input and output capabilities of the host device. | b | 5 | 00 00 00 00 00 |
| 9F09 | Application Version Number (M/Chip) PayPass M/Chip 3.0(Value = 00 02) Amex (Value = 00 01) | b | 2 | 00 02 |
| DF8117 | Card Data Input Capability | b | 1 | 00 |
| DF8118 | Host Capabilities - CVM Required | b | 1 | 60 |
| DF8119 | Host Capabilities - No CVM Required | b | 1 | 08 |
| DF811A | Default UDOL Used for calculating the CCC if no UDOL is present in the card. The default is the tag and length of the "unpredictable number." | b | 3 | 9F 6A 04 |
| DF8108 | DS AC Type | b | 1 | |
| DF60 | DS Input (Card) | b | 1 | |
| DF8109 | DS Input (Term) | b | 1 | |
| DF62 | DS ODS Info | b | 1 | |
| DF810A | DS ODS Info For Reader | b | 1 | |
| DF63 | DS ODS Term | b | 1 | |
| DF810D | DSVN Term | b | 1 | |

| Tag | Description | Format | Length | Default Value in Group 80 |
|---------------|--|--------|--------|-------------------------------|
| 9F1E | Interface Device Serial Number This is intended to be the serial number <i>of the host</i> . It is configured by the POS and is unique to the host device. | an | 8 | 30 30 30 30 30 30 30 30 |
| DF811B | Kernel Configuration | b | 1 | 20 |
| 9F6D | Application Version Number (MagStripe) | b | 2 | 00 01 |
| DF811E | MagStripe CVM Required Capability Indicates the CVM capability of the host/reader in the case of a MagStripe mode transaction when the <i>amount, authorized (numeric)</i> is greater than the <i>reader CVM required limit</i> . | b | 1 | 10 |
| DF812C | MagStripe No CVM Required Capability Indicates the CVM capability of the host/reader in the case of a MagStripe mode transaction when the <i>amount, authorized (numeric)</i> is less than or equal to the <i>reader CVM required limit</i> . | b | 1 | 00 |
| DF811C | Maximum Lifetime of Torn Transaction Record This is the maximum time a torn record can exist in the log before it expires. It is expressed in seconds. While the transaction log is global to the reader, the MasterCard application is the only application that supports it. | b | 2 | 00 00 |
| DF811D | Maximum Number of Torn Transaction Records Due to storage limitations, the maximum number of records that may be configured is 2. There may be 0, 1, or 2 torn transaction records configured. If 0 records are configured, the torn transaction recovery facility is effectively disabled. While the transaction log is global to the reader, the MasterCard application is the only application that supports it. | b | 1 | 00 |
| 9F15 | Merchant Category Code Classifies the type of business being done by the merchant, see ISO 8583:1993. | n4 | 2 | 11 11 |
| 9F16 | Merchant Identifier | ans | 15 | |
| 9F4E | Merchant Name and Location Allows the reader to be configured with the merchant's name and location (VCPS 2.1.1 and M/Chip 3.0) | ASCII | ≤30 | |
| 9F7E | Mobile Support Indication (PayPass only) | b | 1 | |
| DF8123 | Reader Contactless Floor Limit | n12 | 6 | 00 00 00 01 00 00 |

| Tag | Description | Format | Length | Default Value in Group 80 |
|----------------------|---|--------|--------|---------------------------|
| DF8124 | Reader Contactless Transaction Limit - No On-Device CVM When there is no on-device CVM available (for example, with a phone), then this is the transaction limit that will be used. Default = \$300.00 | n12 | 6 | 00 00 00 03 00 00 |
| DF8125 | Reader Contactless Transaction Limit - On-Device CVM When on-device CVM is available (for example, with a phone) then this is the transaction limit that will be used. | n12 | 6 | 00 00 00 05 00 00 |
| DF8126 (1) | CVM Required Limit Indicates the CVM required limit in the host <i>for the associated AIDs</i> . Default = \$10.00 | n12 | 6 | 00 00 00 00 10 00 |
| DF811F | Security Capabilities | b | 1 | 08 |
| DF8120 (1) | Host Action Code (Default) Reflect the acquirer-selected action to be taken upon analysis of the TVR. | b | 5 | 00 00 00 00 00 |
| DF8121 (1) | Host Action Code (Denial) Reflect the acquirer-selected action to be taken upon analysis of the TVR. | b | 5 | 00 00 00 00 00 |
| DF8122 (1) | Host Action Code (Online) Reflect the acquirer-selected action to be taken upon analysis of the TVR. | b | 5 | 00 00 00 00 00 |
| 9F33 | Host Capabilities Indicates the card data input, CVM, and security capabilities of the host device. <div style="border: 2px solid red; padding: 5px; margin: 10px 0;"> <p>This tag (9F33) only configures bytes 1 and 3 of the host capabilities. Byte 2 of the host capabilities actually comes from DF28 or DF29 during the transaction.</p> <p>Refer to tags DF28 and DF29 for details.</p> <p>Note: Byte 1 of 9F33 is the same as DF8117 card data input capability defined in PayPass 3.0.2 and Byte 3 of 9F33 is the same as DF811F security capability.</p> </div> | b | 3 | |
| 9F1A | Host Country Code Indicates the country code of the host device, represented according to ISO 3166. | n3 | 2 | 08 40 |
| 9F1C | Host Identification | an | 8 | |
| 9F1D | Terminal risk management | b | 8 | 6C FF 00 00 00 00 00 00 |

| Tag | Description | Format | Length | Default Value in Group 80 |
|----------------------|--|--------|--------|---------------------------|
| 9F35 | Host Type Indicates the environment of the host device, its communications capability, and its operational control. | n2 | 1 | 22 |
| DFEE37 [1] | UI Scheme Value = 00:ViVOpay User Interface (default) Value = 02:Visa Wave User Interface Value = 03:EMEA User Interface Note: LCD Messages need to be configured separately. Warning: EMEA UI is intended for use in the EMV or European environment, where the reader is not allowed to poll continuously (for example, to operate in Auto Poll Mode). The reader does NOT support Auto Poll while in EMEA UI mode. The reader is not certified to work properly in this situation. <div style="border: 2px solid red; background-color: yellow; padding: 5px;"> For PayPass M/Chip, this value should be set to 03 (EMEA). It defaults to ViVOpay for backward compatibility with MagStripe applications. </div> | b | 1 | 03 |
| DFEE38 [1] | Language Option for LCD display Value = 00: English only display (default) Value = 01: Chinese only display [2] Value = 02: English & Chinese display [2] Value = 03: French only display Value = 04: Other Language (if ILM present) [2] Value = 05: Reserved Value = 06: Japanese Value = 07: Spanish Value = 08: Turkish Note: The supported value may vary in different products; status code 0x06 is returned for unsupported values. | b | 1 | 00 |
| DFEE0F | Revocation List | b | 1 | 01 |

[1] These objects use proprietary tags. The use of these tags should be restricted to the serial interface. After the reader has returned an OK response frame, the host application should dispose of the tags to avoid conflicts with other proprietary TLVs. **IMPORTANT NOTE:** Many two-bytes tags that begin 'FF' have been superseded with BER-legal three-byte tags. See [2-byte Tags vs. 3-byte Tags](#) further below.

[2] These objects only work on the ViVOpay graphic reader.

Phone Message Table – Hard-Coded Default Value in Kernel

| PCI Mask | PCI Value | Message Identifier | Status |
|----------|-----------|--------------------|----------------|
| 000800 | 000800 | 20 (SEE PHONE) | 00 (NOT READY) |
| 000400 | 000400 | 20 (SEE PHONE) | 00 (NOT READY) |
| 000100 | 000100 | 20 (SEE PHONE) | 00 (NOT READY) |
| 000200 | 000200 | 20 (SEE PHONE) | 00 (NOT READY) |
| 000000 | 000000 | 20 (SEE PHONE) | 00 (NOT READY) |

4.3.3 American Express Group Configuration TLVs

American Express use Group A0 as default Group. The [Set Configurable Group](#) command should be used to set the TLVs in this section. The following table in this section shows TLVs in American Express Configuration Group.

Rule for American Express Configuration Group:

The following American Express Group TLVs should not be configured for Group 0. The default American Express group is Group A0. That is, when the reader is configured from the factory, the American Express System AIDs will be associated with Group A0. If there are tags in an American Express group that should be set, they must all be set explicitly, because the absent values *are not* filled in with Group 0 defaults.

American Express Group tags are instantiated a little differently than other groups. Group 0 is never used as a default. Refer to the section on the [American Express Default Group](#) for an explanation of how the tag database is instantiated for American Express.

American Express Default Group A0 Configuration TLVs

| Tag | Description | Format | Length | Default Value in Group 2 |
|--------|---|--------|--------|--------------------------|
| DFEE2D | Group Number The group number assigned to this group of parameters. AIDs may be associated with the group number. This tag is mandatory when getting or setting group parameters and it must be the 1 st TLV in Data Field. It is used as the "key" for the group parameter set. | n2 | 1 | A0 |
| 9C | Transaction Type Indicates the type of financial transaction, represented by the first two digits of ISO 8583:1987 processing code (default = purchase goods or services). | n2 | 1 | 00 |

| Tag | Description | Format | Length | Default Value in Group 2 |
|------|--|--------|--------|--------------------------|
| 5F2A | Transaction Currency Code Indicates the currency code of the transaction according to ISO 4217. <div style="border: 2px solid red; background-color: yellow; padding: 5px; margin-top: 10px;"> Make sure to use the same Transaction Currency Code for all configurable AIDs (default = US Dollars). </div> | n3 | 2 | 08 40 |
| 5F36 | Transaction Currency Exponent Indicates the implied position of the decimal point from the right of the transaction amount represented according to ISO 4217 (decimal is two places from right of the transaction amount). | n1 | 1 | 02 |
| 9F02 | Amount Authorized (Numeric) | n12 | 6 | 00 00 00 00 00 01 |
| 9F03 | Amount Other (Numeric) | n12 | 6 | 00 00 00 00 00 00 |
| 9F09 | Application Version Number PayPass M/Chip (Value = 00 02) Amex (Value = 00 01) | b | 2 | 00 02 |
| 9F15 | Merchant Category Code Classifies the type of business being done by the merchant, see ISO 8583:1993 | n4 | 2 | 00 00 |
| 9F1A | Host Country Code Indicates the country code of the host device, represented according to ISO 3166. Default = US | n3 | 2 | 08 40 |
| 9F33 | Host Capabilities Indicates the card data input, CVM, and security capabilities of the host device Default = <ul style="list-style-type: none"> ▪ No CVM required ▪ SDA supported ▪ DDA supported ▪ Card Capture ▪ CDA supported | b | 3 | 00 08 E8 |

| Tag | Description | Format | Length | Default Value in Group 2 | | | |
|--------------------------|---|--------|--------|--------------------------|----------------------------------|----------|------------|
| 9F35 | Host Type Indicates the environment of the host device, its communications capability, and its operational control. | n2 | 1 | 25 | | | |
| | Environment | | | | Operational Control Provided By: | | |
| | | | | | Financial Institution | Merchant | Cardholder |
| | Attended | | | | | | |
| | Online only | | | | 11 | 21 | |
| | Offline with online capability | | | | 12 | 22 | |
| | Offline only | | | | 13 | 23 | |
| | Unattended | | | | | | |
| | Online only | | | | 14 | 24 | 34 |
| | Offline with online capability | | | | 15 | 25 | 35 |
| Offline only | 16 | 26 | 36 | | | | |
| 9F40 | Additional Host Capabilities Indicates the data input and output capabilities of the host device. | b | 5 | 60 00 00 30 00 | | | |
| 9F6D | Contactless Reader Capabilities | n2 | 1 | C0 | | | |
| 9F6E | Enhanced Contactless Reader Capabilities | b | 4 | 58 80 00 00 | | | |
| DFEE34 | Host Contactless Transaction Limit Indicates the host device limit for this AID for Contactless transactions. | n12 | 6 | 00 00 00 01 00 00 | | | |
| DFEE5C | RF Deactivate Period | b | 4 | 01 50 00 00 | | | |
| DF8126 ^[1] | CVM Required Limit Indicates the CVM required limit in the host device <i>for the associated AIDs</i> . | n12 | 6 | 00 00 00 00 80 00 | | | |
| DF8123 | Reader Contactless Floor Limit | n12 | 6 | 00 00 00 00 60 00 | | | |
| DF8122 ^[1] | Host Action Code (Online) Reflect the acquirer-selected action to be taken upon analysis of the TVR. | b | 5 | F8 50 AC F8 00 | | | |
| DF8120 ^[1] | Host Action Code (Default) Reflect the acquirer-selected action to be taken upon analysis of the TVR. | b | 5 | F8 50 AC A0 00 | | | |
| DF8121 ^[1] | Host Action Code (Denial) Reflect the acquirer-selected action to be taken upon analysis of the TVR. | b | 5 | 00 00 00 00 00 | | | |

^[1] These objects use proprietary tags. The use of these tags should be restricted to the serial interface. After the reader has returned an OK response frame, the host application should dispose of the tags to avoid conflicts with other proprietary TLVs.

IMPORTANT NOTE: Many two-bytes tags that begin 'FF' have been superseded with BER-legal three-byte tags. See [2-byte Tags vs. 3-byte Tags](#) further below.

4.3.4 Discover Group Configuration TLVs

Discover uses Group B0 as default Group. The [Set Configurable Group](#) command should be used to set the TLVs in this section. The following table in this section shows TLVs in Discover Configuration Group.

Rule for Discover Configuration Group:

If a TLV value is absent in Discover Group, reader will use the value in Group 0, except 3 TLVs below:

For transaction limit (DFEE34), CVM limit (DFEE36), floor limit (9F1B): if user-defined TLV Group is used, user should set these above TLVs in the user-defined Group. If not set, these above TLVs will be regarded as not present.

Rule for Discover Configuration Group:

The following Discover Group TLVs should not be configured for Group 0. The default Discover group is Group B0. That is, when the reader is configured from the factory, the Discover System AIDs will be associated with Group B0.

If there are tags in a Discover group that should be set, they must all be set explicitly, because the absent values *are not* filled in with Group 0 defaults.

Discover Group tags are instantiated differently than other groups. Group 0 is never used as a default. Refer to the section on [Discover Default Group](#) for an explanation of how the tag database is instantiated for Discover.

Discover Default Group B0 Configuration TLVs

| Tag | Description | Format | Length | Default Value in Group 2 |
|--------|---|--------|--------|--------------------------|
| DFEE2D | Group Number The group number assigned to this group of parameters. AIDs may be associated with the group number. <div style="border: 2px solid red; background-color: yellow; padding: 5px;"> This tag is mandatory when getting or setting group parameters and it must be the 1st TLV in data field. It is used as the "key" for the group parameter set. </div> | n2 | 1 | B0 |
| 9F02 | Amount Authorized (Numeric) | n12 | 6 | 00 00 00 00 00 01 |
| 9F03 | Amount Other (Numeric) | n12 | 6 | 00 00 00 00 00 00 |
| 9F09 | Application Version Number | b | 2 | 01 00 |
| 9F1A | Host Country Code Indicates the country code of the host device, represented according to ISO 3166. Default = US | n3 | 2 | 08 40 |
| 9F1B | Host Floor Limit Indicates the floor limit in the host device <i>for the AID(s) associated with this group.</i> Note: The value is the decimal limit amount given in binary represented in Hex in the command/response. (60 limit = 6000 decimal = 1770h). | b | 4 | 00 00 17 70 |
| 9F33 | Host Capabilities Indicates the card data input, CVM, and security capabilities of the host device Default = <ul style="list-style-type: none"> ▪ No CVM required ▪ SDA supported ▪ DDA supported ▪ Card Capture ▪ CDA supported | b | 3 | 00 08 E8 |

| Tag | Description | Format | Length | Default Value in Group 2 | | | |
|------------------------------|--|--------|--------|--------------------------|----------------------------------|----------|------------|
| 9F35 | Host Type Indicates the environment of the host device, its communications capability, and its operational control | n2 | 1 | 25 | | | |
| | Environment | | | | Operational Control Provided By: | | |
| | | | | | Financial Institution | Merchant | Cardholder |
| | Attended | | | | | | |
| | Online only | | | | 11 | 21 | |
| | Offline with online capability | | | | 12 | 22 | |
| | Offline only | | | | 13 | 23 | |
| | Unattended | | | | | | |
| | Online only | | | | 14 | 24 | 34 |
| | Offline with online capability | | | | 15 | 25 | 35 |
| Offline only | 16 | 26 | 36 | | | | |
| 9F66 | Host Transaction Qualifier (TTQ) | b | 4 | 80 00 40 00 | | | |
| 5F2A | Transaction Currency Code Indicates the currency code of the transaction according to ISO 4217. <div>Make sure to use the same transaction currency code for all configurable AIDs (default = US Dollars).</div> | n3 | 2 | 08 40 | | | |
| 5F36 | Transaction Currency Exponent Indicates the implied position of the decimal point from the right of the transaction amount represented according to ISO 4217 (decimal is two places from right of the transaction amount). | n1 | 1 | 02 | | | |
| 9C | Transaction Type Indicates the type of financial transaction, represented by the first two digits of ISO 8583:1987 processing code (default = purchase <i>goods or services</i>). | n2 | 1 | 00 | | | |
| DF8122 <small>[1]</small> | Host Action Code (Online) Reflect the acquirer-selected action to be taken upon analysis of the TVR. | b | 5 | 00 00 00 00 00 | | | |
| DF8120 <small>[1]</small> | Host Action Code (Default) Reflect the acquirer-selected action to be taken upon analysis of the TVR. | b | 5 | 00 00 00 00 00 | | | |
| DF8121 <small>[1]</small> | Host Action Code (Denial) Reflect the acquirer-selected action to be taken upon analysis of the TVR. | b | 5 | 00 00 00 00 00 | | | |

| Tag | Description | Format | Length | Default Value in Group 2 |
|-----------------------|--|--------|--------|--------------------------|
| DF8123 | Reader Contactless Floor Limit | n12 | 6 | 00 00 00 01 50 00 |
| DF8124 | Reader Contactless Transaction Limit - No On-Device CVM When there is no on-device CVM available (for example, with a phone), then this is the transaction limit that will be used. Default = \$300.00 | n12 | 6 | 00 00 00 03 00 00 |
| DF8126 ^[1] | CVM Required Limit Indicates the CVM required limit in the host device <i>for the associated AIDs</i> . | n12 | 6 | 00 00 00 00 20 00 |
| DFEE35 | Visa Reader Risk Flags | b | 3 | 01 00 01 |

^[1] These objects use proprietary tags. The use of these tags should be restricted to the serial interface. After the reader has returned an OK response frame, the host application should dispose of the tags to avoid conflicts with other proprietary TLVs.

NOTE: Certain older 2-byte proprietary tags beginning with 'FF' are not properly formed BER-TLV tags. These "illegal" tags have since been replaced with properly formed 3-byte equivalents as defined in the table at [2-byte Tags versus 3-byte Tags](#).

4.3.5 Interac Group Configuration TLVs

Interac use Group C0 as default Group. The [Set Configurable Group](#) command should be used to set the TLVs in this section. The following table in this section shows TLVs in Interac Configuration Group.

Rule for Interac Configuration Group:

The following Interac Group TLVs should not be configured for Group 0. The default Interac group is Group C0. That is, when the reader is configured from the factory, the Interac System AIDs will be associated with Group C0.

If there are tags in a Interac group that should be set, they must all be set explicitly, because the absent values *are not* filled in with Group 0 defaults.

Interac Group tags are instantiated differently than other groups. Group 0 is never used as a default. Refer to the section on [Interac Default Group](#) for an explanation of how the tag database is instantiated for Interac.

Interac Default Group C0 Configuration TLVs

| Tag | Description | Format | Length | Default Value in Group 4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------------------------|--|-------------|----------------------------------|--------------------------|--|-----------------------|----------|------------|-----------------|--|--|--|-------------|----|----|--|--------------------------------|----|----|--|--------------|----|----|--|-------------------|--|--|--|-------------|----|----|----|--------------------------------|----|----|----|--------------|----|----|----|----|---|----|
| DFEE2D | Group Number The group number assigned to this group of parameters. AIDs may be associated with the group number. This tag is mandatory when getting or setting group parameters and it must be the 1 st TLV in DATA FIELD. It is used as the “key” for the group parameter set. | n2 | 1 | C0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 5F2A | Transaction Currency Code Indicates the currency code of the transaction according to ISO 4217. <div>Make sure you use the same Transaction Currency Code for all configurable AIDs (default = US Dollars).</div> | n3 | 2 | 01 24 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 9F09 | Application Version Number | b | 2 | 00 02 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 9F1A | Host Country Code Indicates the country code of the host device, represented according to ISO 3166. Default = US | n3 | 2 | 01 24 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 9F35 | Host Type Indicates the environment of the host device, its communications capability, and its operational control. <table><tr><td rowspan="2">Environment</td><td colspan="3">Operational Control Provided By:</td></tr><tr><td>Financial Institution</td><td>Merchant</td><td>Cardholder</td></tr><tr><td>Attended</td><td></td><td></td><td></td></tr><tr><td>Online only</td><td>11</td><td>21</td><td></td></tr><tr><td>Offline with online capability</td><td>12</td><td>22</td><td></td></tr><tr><td>Offline only</td><td>13</td><td>23</td><td></td></tr><tr><td>Unattended</td><td></td><td></td><td></td></tr><tr><td>Online only</td><td>14</td><td>24</td><td>34</td></tr><tr><td>Offline with online capability</td><td>15</td><td>25</td><td>35</td></tr><tr><td>Offline only</td><td>16</td><td>26</td><td>36</td></tr></table> | Environment | Operational Control Provided By: | | | Financial Institution | Merchant | Cardholder | Attended | | | | Online only | 11 | 21 | | Offline with online capability | 12 | 22 | | Offline only | 13 | 23 | | Unattended | | | | Online only | 14 | 24 | 34 | Offline with online capability | 15 | 25 | 35 | Offline only | 16 | 26 | 36 | n2 | 1 | 22 |
| Environment | Operational Control Provided By: | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Financial Institution | Merchant | Cardholder | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Attended | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Online only | 11 | 21 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Offline with online capability | 12 | 22 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Offline only | 13 | 23 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Unattended | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Online only | 14 | 24 | 34 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Offline with online capability | 15 | 25 | 35 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Offline only | 16 | 26 | 36 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 9F58 | Merchant Type Indicator Provides the merchant type indicator used by the card for risk management. Five values are valid: 01, 02, 03, 04 and 05 (Interac). | n1 | 1 | 03 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 9F59 | Host Transaction Information (TTI) Provides host transaction information for the current transaction (Interac). | b | 3 | DC 87 00 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Tag | Description | Format | Length | Default Value in Group 4 |
|-----------------------|--|--------|--------|--------------------------|
| 9F5A | Host Transaction Type (Interac) <ul style="list-style-type: none"> 0x00 = Purchase 0x01 = Refund | b | 1 | 00 |
| 9F5D | Host Contactless Receipt Required Limit Limit amount used to compare against transaction amount to automatically print a transaction record (Interac). | n12 | 6 | 00 00 00 00 50 00 |
| 9F5E | Host Option Status Options supported by the host device (Interac). | b | 2 | E0 00 |
| 9F5F | Host (Reader) Contactless Floor Limit Floor limit amount used to compare against transaction amount (Interac). | n12 | 6 | 00 00 00 00 50 00 |
| 9F1B | Host Floor Limit Indicates the floor limit in the host device <i>for the AID(s) associated with this group.</i> Note: The value is the decimal limit amount given in binary represented in hex in the command/response. (60 limit = 6000 decimal = 1770h). | b | 4 | 00 00 00 00 |
| DF8124 | Reader Contactless Transaction Limit - No On-Device CVM When there is no on-device CVM available (for example, with a phone), then this is the transaction limit that will be used. Default = \$300.00 | n12 | 6 | 00 00 00 01 00 00 |
| DF8126 ^[1] | CVM Required Limit Indicates the CVM required limit in the host device <i>for the associated AIDs.</i> | n12 | 6 | 00 00 00 00 60 00 |
| DF8122 ^[1] | Host Action Code (Online) Reflect the acquirer-selected action to be taken upon analysis of the TVR. | b | 5 | 00 00 00 00 00 |
| DF8120 ^[1] | Host Action Code (Default) Reflect the acquirer-selected action to be taken upon analysis of the TVR. | b | 5 | 00 00 00 00 00 |
| DF8121 ^[1] | Host Action Code (Denial) Reflect the acquirer-selected action to be taken upon analysis of the TVR. | b | 5 | 00 00 00 00 00 |
| DFEE2B | Maximum Target Percentage for Biased Random Selection Value used in terminal risk management for random transaction selection. | b | 1 | 32 |
| DFEE2C | Target Percentage for Random Selection Value used in terminal risk management for random transaction selection (Interac). | b | 1 | 0A |

| Tag | Description | Format | Length | Default Value in Group 4 |
|--------|--|--------|--------|--------------------------|
| DFEE2A | Threshold Value for Biased Random Selection Value used in terminal risk management for random transaction selection (Interac). | n12 | 6 | 00 00 00 00 50 00 |
| DFEF56 | Retry Limit | b | 1 | 03 |
| 9F02 | Amount Authorized (Numeric) | n12 | 6 | 00 00 00 00 15 00 |
| 9F03 | Amount Other (Numeric) | n12 | 6 | 00 00 00 00 00 00 |
| 5F36 | Transaction Currency Exponent Indicates the implied position of the decimal point from the right of the transaction amount represented according to ISO 4217 (decimal is two places from right of the transaction amount). | n1 | 1 | 02 |
| 9C | Transaction Type Indicates the type of financial transaction, represented by the first two digits of ISO 8583:1987 processing code (default = purchase goods or services). | n2 | 1 | 00 |
| 9F33 | Host Capabilities Indicates the card data input, CVM, and security capabilities of the host device Default = <ul style="list-style-type: none"> ▪ No CVM required ▪ SDA supported ▪ DDA supported ▪ Card Capture ▪ CDA supported | b | 3 | 00 48 E8 |
| DFEE37 | UI Scheme Value = 00:ViVOpay User Interface (default) Value = 02:Visa Wave User Interface Value = 03:EMEA User Interface <i>Note:</i> LCD Messages need to be configured separately. Warning: EMEA UI is intended for use in the EMV or European environment, where the reader Vend is not allowed to poll continuously (for example, to operate in Auto Poll Mode). The reader Vend does NOT support Auto Poll while in EMEA UI mode. The reader is not certified to work properly in this situation. | b | 1 | 03 |

^[1] These objects use proprietary tags. The use of these tags should be restricted to the serial interface. After the reader has returned an OK response frame, the host application should dispose of the tags to avoid conflicts with other proprietary TLVs.

NOTE: Certain older 2-byte proprietary tags beginning with 'FF' are not properly formed BER-TLV tags. These "illegal" tags have since been replaced with properly formed 3-byte equivalents as defined in the table at [2-byte Tags versus 3-byte Tags](#).

4.3.6 AID Configuration Tags

In this table, the "Usage" column indicates when the tag is used. In some cases, the use may depend on whether a system AID or a user AID is being configured. The possible usages are:

- MAND – this is a mandatory tag when configuring an AID.
- OPT – this is an optional tag when configuring an AID.
- NEVER – this tag should never be used for configuring this type of AID (for example, "System").
- DEP – this tag is Mandatory depending on how another tag is configured.

For default values of the AID configuration TLVs for each System AID, refer to the [System AID Default Configuration TLVs](#) table.

AID Configuration TLVs

| Tag | Data Object Name | Usage | Description | Format | Length |
|---------------------|--|----------------------------|---|--------|--------|
| 9F06 | Application Identifier (AID) | MAND | Identifies the application as described in ISO/IEC 7816-5. This must be the 2 nd TLV in the data field. | b | 5 – 16 |
| DFEE7C | Auto-Switch | OPT | Automatically switch to Pass-Through Mode when PICC is unknown. 00h = disabled (default) 01h = enabled | b | 1 |
| FFE0 ^[1] | Registered Application Provider Identifier (RID) | Sys = NEVER User = OPT | Identifies the payment system to which the certification authority public key is associated. if this tag is not provided the first five bytes from the AID are used. | b | 5 |
| FFE1 ^[1] | Partial Selection Allowed | OPT (Visa MAND) | Tells the reader to allow partial selection during the initial select process. 01 = Allowed, 00 = Disabled Note: Required for Visa application flow, this value is set to 01 Allowed and cannot be changed. | b | 1 |
| FFE2 ^[1] | Application Flow | Sys = NEVER User = MAND | 1 (01h) – MasterCard MagStripe Application 2 (02h) – American Express Application 3 (03h) – MasterCard PayPass Application 6 (06h) – Visa Application. 13 (0Dh) – Discover Application 14 (0Eh) – JCB QuicPay Application 15 (0Fh) – STAR Application 21 (15h) – Interac Application 23 (17h) – Smart Tap | b | 1 |

| FFE3 | Selection Features | OPT | <p>Enables or disables selection features. <i>For M/Chip 3.0, this value will default to 74h.</i> Refer to the Selection Features section for a detailed description of this tag.</p> <table><tr><th>87654321</th><th>Selection Feature</th></tr><tr><td>-----x</td><td>Deprecated / RFU</td></tr><tr><td>-----x-</td><td>Extended Selection Supported</td></tr><tr><td>-----x--</td><td>Cardholder Confirmation Not Supported</td></tr><tr><td>----x---</td><td>API (application priority indicator) required</td></tr><tr><td>---x----</td><td>Invalid AID Allowed</td></tr><tr><td>--x-----</td><td>Duplicate AID Allowed</td></tr><tr><td>-x-----</td><td>Enable Kernel ID</td></tr><tr><td>x-----</td><td>RFU</td></tr></table> | 87654321 | Selection Feature | -----x | Deprecated / RFU | -----x- | Extended Selection Supported | -----x-- | Cardholder Confirmation Not Supported | ----x--- | API (application priority indicator) required | ---x---- | Invalid AID Allowed | --x----- | Duplicate AID Allowed | -x----- | Enable Kernel ID | x----- | RFU | b | 1 |
|---------------------|--|------|--|----------|-----------------------------------|--------|---|---------|--|----------|---------------------------------------|----------|---|----------|---------------------|----------|-----------------------|---------|------------------|--------|-----|---|---|
| 87654321 | Selection Feature | | | | | | | | | | | | | | | | | | | | | | |
| -----x | Deprecated / RFU | | | | | | | | | | | | | | | | | | | | | | |
| -----x- | Extended Selection Supported | | | | | | | | | | | | | | | | | | | | | | |
| -----x-- | Cardholder Confirmation Not Supported | | | | | | | | | | | | | | | | | | | | | | |
| ----x--- | API (application priority indicator) required | | | | | | | | | | | | | | | | | | | | | | |
| ---x---- | Invalid AID Allowed | | | | | | | | | | | | | | | | | | | | | | |
| --x----- | Duplicate AID Allowed | | | | | | | | | | | | | | | | | | | | | | |
| -x----- | Enable Kernel ID | | | | | | | | | | | | | | | | | | | | | | |
| x----- | RFU | | | | | | | | | | | | | | | | | | | | | | |
| FFE4 ^[1] | TLV Group Number | MAND | <p>The TLV Group number that contains the characteristics for this AID This must be the 1st TLV in Data Field.</p> <p><i>For MasterCard PayPass and any applications that use the combined selection feature, this tag represents the fallback group if the TLV FFE9 transaction type list is empty, or the Kernel ID is disabled (see tag FFE3). For MasterCard PayPass, this tag may NOT be Group 0.</i></p> | n2 | 1 | | | | | | | | | | | | | | | | | | |
| FFE5 ^[1] | Maximum AID Length | DEP | <p>This value must be <= 16. For Visa application flow, this value is set to 16 and cannot be changed.</p> <p>Note: This tag must be included if the FFE1 Partial Select TLV is included.</p> | b | 1 | | | | | | | | | | | | | | | | | | |
| FFE6 ^[1] | AID Disabled | OPT | <p>Used to disable a System AID (has no effect on a User AID). 80h = disabled and 00h = enabled</p> | b | 1 | | | | | | | | | | | | | | | | | | |
| FFE8 | Exclude from Processing | OPT | <p>This byte is formatted as follows:</p> <table><tr><th>87654321</th><th>Meaning (0 = disable, 1 = enable)</th></tr><tr><td>-----x</td><td>Exclude from PPSE processing. 1 = This AID will not be added to the candidate list during PPSE.</td></tr><tr><td>-----x-</td><td>Exclude from Trial and Error processing. 1 = This AID may not be added to the candidate list during Trial and Error (sometime referred to as “List of AIDs” processing).</td></tr><tr><td>xxxxxx--</td><td>RFU</td></tr></table> | 87654321 | Meaning (0 = disable, 1 = enable) | -----x | Exclude from PPSE processing. 1 = This AID will not be added to the candidate list during PPSE. | -----x- | Exclude from Trial and Error processing. 1 = This AID may not be added to the candidate list during Trial and Error (sometime referred to as “List of AIDs” processing). | xxxxxx-- | RFU | b | 1 | | | | | | | | | | |
| 87654321 | Meaning (0 = disable, 1 = enable) | | | | | | | | | | | | | | | | | | | | | | |
| -----x | Exclude from PPSE processing. 1 = This AID will not be added to the candidate list during PPSE. | | | | | | | | | | | | | | | | | | | | | | |
| -----x- | Exclude from Trial and Error processing. 1 = This AID may not be added to the candidate list during Trial and Error (sometime referred to as “List of AIDs” processing). | | | | | | | | | | | | | | | | | | | | | | |
| xxxxxx-- | RFU | | | | | | | | | | | | | | | | | | | | | | |

| FFE9 | Transaction Type List | OPT | <p>This list defines 3-byte triplets, where the Kernel ID and transaction type may be used to identify the group that will be used to instantiate the dataset for the transaction. A maximum of 8 entries may appear in this list. The format of each triplet entry is as follows:</p> <table><tr><th>Byte</th><th>Description</th></tr><tr><td>1- Kernel ID</td><td>Kernel ID as defined by EMV first byte only.</td></tr><tr><td>2- Transaction Type</td><td>Supported transaction types may be: Payment(00), Cash(01), Cashback(09) or Refund(20)</td></tr><tr><td>3- Group Number</td><td>The group that should be used for this transaction and Kernel ID.</td></tr></table> <p><i>Group 0 may not be used in this list.</i></p> | Byte | Description | 1- Kernel ID | Kernel ID as defined by EMV first byte only. | 2- Transaction Type | Supported transaction types may be: Payment(00), Cash(01), Cashback(09) or Refund(20) | 3- Group Number | The group that should be used for this transaction and Kernel ID. | b | Variable ≤ 24 | | | | |
|---------------------|---|-----|---|------------------|-------------|--------------|--|---------------------|---|------------------|---|--------------|------------------|------------|---|---|---|
| Byte | Description | | | | | | | | | | | | | | | | |
| 1- Kernel ID | Kernel ID as defined by EMV first byte only. | | | | | | | | | | | | | | | | |
| 2- Transaction Type | Supported transaction types may be: Payment(00), Cash(01), Cashback(09) or Refund(20) | | | | | | | | | | | | | | | | |
| 3- Group Number | The group that should be used for this transaction and Kernel ID. | | | | | | | | | | | | | | | | |
| FFEA | Configurable Kernel Identifier | OPT | <p>This kernel identifier will be used if the card does not supply a valid Kernel ID (9F2A). If this tag is not provided, then default kernel identifiers will be used depending on the application:</p> <table><tr><th>Card Application</th><th>Default KID</th></tr><tr><td>MasterCard</td><td>2</td></tr><tr><td>Visa</td><td>3</td></tr><tr><td>American Express</td><td>4</td></tr><tr><td>JCB QuickPay</td><td>5</td></tr><tr><td>All others</td><td>0</td></tr></table> <p><i>This tag is equivalent to MasterCard tag DF810C.</i></p> | Card Application | Default KID | MasterCard | 2 | Visa | 3 | American Express | 4 | JCB QuickPay | 5 | All others | 0 | b | 1 |
| Card Application | Default KID | | | | | | | | | | | | | | | | |
| MasterCard | 2 | | | | | | | | | | | | | | | | |
| Visa | 3 | | | | | | | | | | | | | | | | |
| American Express | 4 | | | | | | | | | | | | | | | | |
| JCB QuickPay | 5 | | | | | | | | | | | | | | | | |
| All others | 0 | | | | | | | | | | | | | | | | |
| DFEF2C | Terminal AID List | OPT | <p>Tells the reader to allow Terminal AID List support during the initial select process. 01 = Allowed, 00 = Disabled</p> | b | 1 | | | | | | | | | | | | |

Setting the FFE6 tag may disable an AID. However, the preferred method to disable an AID is to issue a "Delete Configurable AID" command (04-04). For a system AID, the command will set the disable bit in FFE6.

The following table lists the System AIDs and the default values for their TLVs.

System AID Default Configuration TLVs

| Name | Tag | Length (Hex) | Value (Hex) | Application Name |
|---------------------------------------|-------------|--------------|--|----------------------------------|
| Group | FFE4/DFEE2D | 01 | A0 | American Express |
| AID | 9F06 | 06 | A0 00 00 00 25 01 | |
| Partial Selection | FFE1/DFEE4B | 01 | 01 | |
| Selection Features | FFE3/DFEE4D | 01 | 54 | |
| Max AID Length | FFE5/DFEE2E | 01 | 10 | |
| Kernel ID Transaction Type Group List | FFE9/DFEE54 | 0C | 04 00 A0 04 01 A0 04 09 A0 04 20 A0 | |
| Configurable Kernel Identifier | FFEA/DFEE59 | 01 | 04 | |
| APPLICATION | FFE2/DFEE4C | 01 | 02 | |
| Group | FFE4/DFEE2D | 01 | 80 | MasterCard Credit/Debit (Global) |
| AID | 9F06 | 07 | A0 00 00 00 04 10 10 | |
| Partial Selection | FFE1/DFEE4B | 01 | 01 | |
| Selection Features | FFE3/DFEE4D | 01 | 74 | |
| Max AID Length | FFE5/DFEE2E | 01 | 10 | |
| Exclude from Processing | FFE8/DFEE53 | 01 | 02 | |
| Kernel ID Transaction Type Group List | FFE9/DFEE54 | 0F | 02 00 80 02 01 80 02 09 80 02 17 80 02 20 80 | |
| Configurable Kernel Identifier | FFEA/DFEE59 | 01 | 02 | |
| APPLICATION | FFE2/DFEE4C | 01 | 03 | |
| Group | FFE4/DFEE2D | 01 | 80 | Maestro (Debit) |
| AID | 9F06 | 07 | A0 00 00 00 04 30 60 | |
| Partial Selection | FFE1/DFEE4B | 01 | 01 | |
| Selection Features | FFE3/DFEE4D | 01 | 74 | |
| Max AID Length | FFE5/DFEE2E | 01 | 10 | |
| Exclude from Processing | FFE8/DFEE53 | 01 | 02 | |
| Kernel ID Transaction Type Group List | FFE9/DFEE54 | 0F | 02 00 80 02 01 80 02 09 80 02 17 80 02 20 80 | |
| Configurable Kernel Identifier | FFEA/DFEE59 | 01 | 02 | |
| APPLICATION | FFE2/DFEE4C | 01 | 03 | |
| Group | FFE4/DFEE2D | 01 | 90 | VISA Debit/Credit (Classic) |
| AID | 9F06 | 07 | A0 00 00 00 03 10 10 | |
| Partial Selection | FFE1/DFEE4B | 01 | 01 | |
| Selection Features | FFE3/DFEE4D | 01 | 14 | |

| Name | Tag | Length (Hex) | Value (Hex) | Application Name |
|---------------------------------------|-------------|--------------|-------------------------------------|---------------------------|
| Max AID Length | FFE5/DFEE2E | 01 | 10 | |
| APPLICATION | FFE2/DFEE4C | 01 | 06 | |
| Group | FFE4/DFEE2D | 01 | 90 | |
| AID | 9F06 | 07 | A0 00 00 00 03 20 10 | VISA Electron |
| Partial Selection | FFE1/DFEE4B | 01 | 01 | |
| Selection Features | FFE3/DFEE4D | 01 | 14 | |
| Max AID Length | FFE5/DFEE2E | 01 | 10 | |
| APPLICATION | FFE2/DFEE4C | 01 | 06 | |
| Group | FFE4/DFEE2D | 01 | 90 | VISA Interlink |
| AID | 9F06 | 07 | A0 00 00 00 03 30 10 | |
| Partial Selection | FFE1/DFEE4B | 01 | 01 | |
| Selection Features | FFE3/DFEE4D | 01 | 14 | |
| Max AID Length | FFE5/DFEE2E | 01 | 10 | |
| APPLICATION | FFE2/DFEE4C | 01 | 06 | |
| Group | FFE4/DFEE2D | 01 | D0 | JCB J Smart Credit |
| AID | 9F06 | 07 | A0 00 00 00 65 10 10 | |
| Partial Selection | FFE1/DFEE4B | 01 | 01 | |
| Selection Features | FFE3/DFEE4D | 01 | 54 | |
| Max AID Length | FFE5/DFEE2E | 01 | 10 | |
| Kernel ID Transaction Type Group List | FFE9/DFEE54 | 0C | 05 00 D0 05 01 D0 05 09 D0 05 20 D0 | |
| Configurable Kernel Identifier | FFEA/DFEE59 | 01 | 05 | |
| APPLICATION | FFE2/DFEE4C | 01 | 14 | Discover Expresspay (ZIP) |
| Group | FFE4/DFEE2D | 01 | B0 | |
| AID | 9F06 | 07 | A0 00 00 03 24 10 10 | |
| Partial Selection | FFE1/DFEE4B | 01 | 01 | |
| Max AID Length | FFE5/DFEE2E | 01 | 10 | |
| APPLICATION | FFE2/DFEE4C | 01 | 13 | |
| Group | FFE4/DFEE2D | 01 | C0 | INTERAC |
| AID | 9F06 | 07 | A0 00 00 02 77 10 10 | |
| Partial Selection | FFE1/DFEE4B | 01 | 01 | |
| Max AID Length | FFE5/DFEE2E | 01 | 10 | |
| Exclude from Processing | FFE8/DFEE53 | 01 | 02 | |
| APPLICATION | FFE2/DFEE4C | 01 | 21 | |
| Group | FFE4/DFEE2D | 01 | B0 | Discover, Pulse D Pas |
| AID | 9F06 | 07 | A0 00 00 01 52 30 10 | |
| Partial Selection | FFE1/DFEE4B | 01 | 01 | |

| Name | Tag | Length (Hex) | Value (Hex) | Application Name |
|-------------------------|-------------|--------------|-------------------------------|------------------|
| Max AID Length | FFE5/DFEE2E | 01 | 10 | |
| APPLICATION | FFE2/DFEE4C | 01 | 13 | |
| Group | FFE4/DFEE2D | 01 | 90 | VISA Plus |
| AID | 9F06 | 07 | A0 00 00 00 03 80 10 | |
| Partial Selection | FFE1/DFEE4B | 01 | 01 | |
| Selection Features | FFE3/DFEE4D | 01 | 14 | |
| Max AID Length | FFE5/DFEE2E | 01 | 10 | |
| APPLICATION | FFE2/DFEE4C | 01 | 06 | |
| Group | FFE4/DFEE2D | 01 | 00 | AppleVas |
| AID | 9F06 | 0A | 4F 53 45 2E 56 41 53 2E 30 31 | |
| Partial Selection | FFE1/DFEE4B | 01 | 00 | |
| Max AID Length | FFE5/DFEE2E | 01 | 10 | |
| Exclude from Processing | FFE8/DFEE53 | 01 | 06 | |
| APPLICATION | FFE2/DFEE4C | 01 | 24 | |
| Group | FFE4/DFEE2D | 01 | 8E | Smart Tap |
| AID | 9F06 | 09 | A0 00 00 04 76 D0 00 01 11 | |
| Partial Selection | FFE1/DFEE4B | 01 | 00 | |
| Max AID Length | FFE5/DFEE2E | 01 | 10 | |
| Exclude from Processing | FFE8/DFEE53 | 01 | 04 | |
| APPLICATION | FFE2/DFEE4C | 01 | 23 | |

4.4 2-byte Tags vs. 3-byte Tags

Note: All NEO 2 firmware only supports strictly compliant BER TLV tags (3-byte tags that begin with DF). Consequently, the feature described below has been deprecated and is only included for reference. DFEF3D is disabled in 2-byte mode and reserved in 3-byte mode.

Some older versions of NEO firmware support the use of certain tags that are not compliant with BER-TLV format (the non-compliant tags are 2-byte tags that begin with 'FF'). It is possible to specify whether a reader outputs old-style (illegal) tags or strictly compliant BER-TLV tags (the latter are 3-byte tags that begin with 'DF'). Configuration tag DFEF3D (used in conjunction with command 04-00) controls the output choice. A data value of 0x01 in this tag forces the use of old/illegal tags, while 0x00 forces the use of the new 3-byte tags (which is the default in VP5300, VP6300, and subsequent products).

Example:

To switch to NEW (BER-compliant) tags:

Command: 04 00

Data: DF ED 11 01 00

Complete command: 56 69 56 4F 74 65 63 68 32 00 04 00 00 05 DF ED 11 01 00 32 E6

This feature only controls the *output* of old versus new tags (all tags are stored in the reader, internally, as the fully compliant BER-TLV tags). The configuration switch is mainly a backwards-compatibility feature for customers who need to support the original, non-compliant tags. If a user's reader outputs illegal (non-standard) 2-byte tags and they want it to output the newer 3-byte tags shown below, set the DFEF3D configuration tag as described above; if that doesn't work, update the firmware and try again.

Below is the list of old/illegal tags, and corresponding new 3-byte tags.

| 2-byte tag | 3-byte tag | Description |
|------------|------------|---|
| FFF0 | DFEE67 | Specific feature switch, used with Visa VCPS 2.1.1/2.1.2. It controls Visa CVN17 support and Track 1 & 2 data in the transaction response. |
| FFF1 | DFEE34 | Host device contactless transaction limit, never used for PayPass. |
| FFF2 | 9F1E | Host device IFD serial number. |
| FFF3 | DFEE44 | Application capability. |
| FFF4 | DFEE35 | Visa reader risk flags. |
| FFF5 | DF8126 | CVM required limit. |
| FFF7 | DFEE7E | Enable/Disable Burst Mode: Value = 00: Disable Burst Mode Value = 01: Enable Burst Mode Value = 02: Burst Mode auto exit; Burst Mode is turned off as soon as a transaction command is received. |

| 2-byte tag | 3-byte tag | Description |
|------------|------------|---|
| FFF8 | DFEE37 | UI Scheme: Value = 00: ViVOtech User Interface Value = 02: Visa Wave User Interface Value = 03: EMEA User Interface. |
| FFF9 | DFEE1C | LCD Font Size. |
| FFFA | DFEF29 | LCD delay time(ms). |
| FFFB | DFEE38 | Language Option for LCD display. |
| FFFD | DF8122 | Host Action Code (Online). |
| FFFE | DF8120 | Host Action Code (Default). |
| FFFF | DF8121 | Host Action Code (Denial). |
| FFE0 | DFEE4A | Registered application provider identifier (RID): Sys =NEVER, User = OPT. Identifies the payment system to which the certification authority public key is associated. If this tag is not provided the first five bytes from the AID are used. |
| FFE1 | DFEE4B | Partial selection allowed, OPT, (Visa MAND). Tells the reader to allow partial selection during the initial select process: 01 = Allowed, 00 = Disabled Note: Required for Visa application flow, this value is set to 01 Allowed and cannot be changed. |
| FFE2 | DFEE4C | 0 = APPLICATION_NONE 1 = MC_APPLICATION 2 = AMX_APPLICATION 3 = MC_MSTRIPE_APPLICATION 4 = AMX_MIDCAP_APPLICATION 5 = CJ_KEYMAN_APPLICATION 6 = VISA_APPLICATION 7 = MIFARE_DEMO_APPLICATION 8 = VIVOWALLET_APPLICATION 9 = AMX_ID_APPLICATION 10= MC_MAESTRO_APPLICATION 11= MC_TEST_APPLICATION 12= LOWES_VIVOWALLET_APPLICATION 13= DISCOVER_APPLICATION 14= JCB_QUICPAY_APPLICATION 15= STAR_APPLICATION 16= MC_MXI_APPLICATION 17= RBS_APPLICATION 18= DESFIRE_APPLICATION_TRACK 19= DESFIRE_APPLICATION_RAW 20= VIVO_COMM_APPLICATION 21= INTERAC_APPLICATION 22= GMAD_APPLICATION 23= ANDROID_PAY_APPLICATION 24= APPLEPAY_VAS_APPLICATION |
| FFE3 | DFEE4D | Selection Features. Flags for Selection Features specific to this AID. |
| FFE4 | DFEE2D | The Group number associated with the configurable AID functions. |
| FFE5 | DFEE2E | Maximum AID length, CON, 16 bytes. Note: This is a MANDATORY TLV if the FFE1 partial select TLV is included. For Visa application flow, this value is set to 16 and cannot be changed. |
| FFE6 | DFEE2F | AID Disabled, OPT. Used to disable a System AID (has no effect on a User AID). 80h = disabled and 00h = enabled |
| FFE8 | DFEE53 | Exclude from Processing. Indicates that this AID should not be considered as a host supported AID for the specified selection methods. |

| 2-byte tag | 3-byte tag | Description |
|------------|------------|--|
| FFE9 | DFEE54 | Kernel ID transaction type Group List. Defines a Kernel ID, a transaction type and a group number (triplet). This TLV uses variable length data, three bytes at a time up to 24 bytes. DO NOT PAD. |
| FFEA | DFEE59 | Configurable Kernel Identifier. Used to define the Kernel ID to use if the card does not provide a Kernel ID or provides one that is length 0. |

5. Card Application Selection

5.1 Combined Selection

The selection of a card application may be based on a Kernel ID, transaction type, and other requirements. The Selection Features tag (FFE3 or DFEE4D) directs the flow through the selection logic in the firmware. In addition, the selection of an AID may be based on a list of transaction types that it supports. Depending on the transaction type, the AID may be mapped to a different Configurable Group. This entire process is referred to as "Combined Selection."

5.1.1 Selection Features (FFE3 or DFEE4D)

The following table defines each of the bits in the Selection Features tag (FFE3 or DFEE4D) and describes how they control the logic flow.

| 87654321 | Feature Name | Description |
|----------|---------------------------------------|--|
| -----x | Deprecated/RFU | Reserved for future use. |
| -----x- | Extended Selection Supported | <p>Allow the Extended Selection value optionally provided by the card in PPSE to be added to the AID for final selection. The resulting AID must be less than or equal to 16 bytes or the candidate will not be added to the candidate list.</p> <p>0 = the Extended Selection value, if provided by the card, will not be appended to the AID value for final selection. 1 = the Extended Selection value, if provided by the card, will be appended to the AID value, if it fits (AID + ES <= 16), for use in final selection.</p> |
| -----x-- | Cardholder Confirmation Not Supported | <p>0 = Cardholder Confirmation is allowed for this AID. If API bit 8 (Cardholder Confirmation) is true, the application will not be added to the candidate list. 1 = Customer Cardholder Confirmation is not allowed for this AID, API bit 8 (Cardholder Confirmation) will be ignored.</p> |
| ----x--- | API Required | <p>0 = the API is not required for this AID and the application may be added to the candidate list if the API is missing. 1 = the API is required for this AID; the application will not be added to the candidate list if the API is missing.</p> |
| ---x---- | Invalid AID Allowed | <p>0 = any invalid AID will cause this AID to terminate the transaction. 1 = any invalid AID will be ignored as related to this AID.</p> |
| --x----- | Duplicate AID Allowed | <p>0 = a duplicate AID, whether extended or not, is not allowed and will not be added to the candidate list. 1 = a duplicate AID, whether extended or not, is allowed and may be added to the candidate list.</p> |
| -x----- | Enable Kernel ID | <p>1 = allow the evaluation of the Kernel ID. 0 = if a Kernel ID is provided by the card it is ignored.</p> |
| x----- | RFU | Reserved for future use. |

Refer to the [System AID Defaults](#) for the configuration of selection features for each of the AIDs. If no Selection Features tag (FFE3) is specified, then no selection features are specified.

5.2 Partial Selection (FFE1 or DFEE4B)

For some applications, an AID on the card may be longer than an AID configured in the reader. If partial selection is allowed, the AID will be considered a match if all of the AID configured in the reader matches the first portion of an AID in the card.

Table 1: Partial Selection (FFE1)

| 87654321 | Value | Description |
|----------|------------------------------|---|
| -----x | Partial selection is allowed | 0 = This AID does not participate in partial selection 1 = This AID will participate in partial selection (default) For M/Chip 3.0, this value should be set to "allowed." |
| xxxxxxx- | RFU | Reserved for future use. |

Historically, Partial Selection has been a separate tag. However, it is an integral part of the selection process, and may be used in conjunction with combined selection features.

5.3 AID Participation in Selection Processes (FFE8 or DFEE53)

In some cases, applications or AIDs may not be able to participate in some of the selection processes. For example, some cards or applications do not support PPSE. The following table describes the bits in tag FFE8 that may be used to exclude an AID from selection processes.

Table 2: Exclude from Processing (FFE8)

| 87654321 | Process to be Excluded | Description |
|-----------|---------------------------------------|---|
| -----x | Exclude from PPSE Processing | 0 = This AID may be added to the candidate list during PPSE. 1 = This AID will not be added to the candidate list during PPSE. |
| -----x- | Exclude from Trial & Error Processing | 0 = This AID may be added to the candidate list during T&E. 1 = This AID will not be added to the candidate list during T&E. |
| -----x-- | Exclude Payment | 0 = This AID may be added to the candidate list for payment. 1 = This AID will not be included for payment. |
| xxxxxxx-- | RFU | Reserved for future use. |

"Trial & Error" is sometimes referred to as "List of AIDs." It is a process by which the reader will attempt to select an AID by going through its list, hoping for a successful selection. For M/Chip 3.0, tag FFE8 should be set to 0x00.

5.4 Terminal AID List (DFEF2C)

The Terminal AID List Tag DFEF2C is associated with each Terminal AID set by the **Load AID** (04-02) command. This tag can control which Terminal AID is sent to the card if:

- the Select AID Command (with PPSE) returns a "Select AID Failed" (response SW1/SW2 not 9000 or format error)
- the AID list returned by the card cannot be matched by any Terminal AID.

One by one, the host device will check each DFEF2C tag associated with the Terminal AID. If the Tag DFEF2C value is '01', the host sends out that Terminal AID data to the card and it can continue the contactless transaction if the card conditions match the Terminal AID. If the Tag DFEF2C value is '00', that Terminal AID data is not sent.

Table 3: Terminal AID List (DFEF2C)

| 87654321 | Value | Description |
|----------|------------------------------|---|
| -----x | Terminal AID List is allowed | 0 = Host will not send out Terminal AID data 1 = Host will send out Terminal AID data Note: The Terminal AIDs can be modified by (Load AID) 04-02 command. |
| xxxxxxx- | RFU | Reserved for future use |

6. Card Application Specific Behavior

This section contains information specific to a particular card application.

6.1 MasterCard PayPass M/Chip

The implementation of MasterCard M/Chip 3.0 follows the EMV mode transaction flow. This includes support for magstripe, but does not include data exchange functionality.

The M/Chip 3.0 implementation incorporates new functionality for:

- Balance reading before and after GenAC
- Recovery of torn transactions
- Support for certificate revocation list functions
- STOP command processing
- Support for defining proprietary tags that are not otherwise handled or defined in the tag database

6.1.1 PayPass Default Group

The PayPass implementation required a new data model in which data objects could be "not present" or "not defined." As a result, the previous method of using Group 0 to define default tags could not be used.

Group 0 is no longer used by the PayPass application. The default group for PayPass applications is Group 80.

In addition to the PayPass default group, the PayPass Kernel also keeps hard-coded values for a sub-set of the group parameters that are essential for a transaction. If one of these data items is not available via **Activate Transaction** or via **Get Configurable Group**, then the kernel uses its own hard-coded value for the 'Not Present' data item.

6.1.2 Balance Read Function

The balance may be read from cards that support balance reading. The balance may be read before or after the Generate AC process. In order to enable balance reading, the tags for balance read must be defined in the tag database. Accomplished this via the **Set Configurable Group** command or by including the Balance TLVs (DF8104, DF8105) in the **Activate** command.

For example, if DF8104 is included in the **Activate** command and the card supports balance reading, the balance read prior to Generate AC will be returned in the DF8104 TLV in the **Activate** response.

6.1.3 Torn Transaction Recovery

A method exists for saving data from a torn transaction and matching up that transaction with a card that re-appears in the field to complete the transaction. Due to space limitations, a reader can retain a maximum number of two torn transaction records. New tags (configurable in a PayPass Group) control the size and use of the torn transaction log:

| Tag | Name | Description |
|--------|--|---|
| DF811C | Max Lifetime of Torn Transaction Record | Controls how long a torn transaction record can exist in the log before it expires. It is expressed in seconds. |
| DF811D | Maximum Number of Torn Transaction Records | The only possible values are 0, 1, or 2. If this tag is set to 0, then the torn transaction recovery is effectively disabled. |

These tags are configurable for PayPass groups. If they are configured for other groups, they will not be used.

6.1.4 EMV Certificate Revocation List

The MasterCard application can make use of the EMV Certificate Revocation List features if they are enabled. The DF26 tag is used to enable or disable the Certificate Revocation List function. The default value for this tag is "enabled" (1).

6.1.5 Stop Transaction Command

M/Chip 3.02 includes a **Stop Transaction** command that functions similar to a **Cancel** command.

6.1.6 Proprietary Tag List

The proprietary tag list feature is not specific to MasterCard. Refer to the [Card Application Proprietary Tag List](#) section for details.

6.1.7 PayPass Personalization Limits

To guarantee the successful completion of PayPass transactions using CDA or SDA, size restrictions noted in this section apply.

1.1.1.15. CDA Transactions

The combined length of the following data objects personalized to the card cannot exceed 2400 bytes.

- ICC public key certificate
- ICC public key exponent
- ICC public key remainder
- Static data used in data authentication (2048 bytes maximum)

1.1.1.16. SDA Transactions

The combined length of the following data objects personalized to the card cannot be greater than 2400 bytes.

- Signed static application data
- Static data used in data authentication (2048 bytes maximum)

7. Protocol Command Reference: Protocol 2

7.1 RTC (Real Time Clock) Commands

7.1.1 Get RTC Date and Time (25-06)

The **Get RTC Date and Time command** retrieves the RTC date and time from the reader.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 25h | 06h | 00h | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 - Byte 19 | Byte 20 | Byte 21 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 25h | See status code table | 06h | | See Data Field | | |

Response Data Field

| Data Field | Length (bytes) | Description |
|------------|----------------|----------------------|
| Year | 1 | 15~99, 00 (BCD Code) |
| Month | 1 | 01~12 (BCD Code) |
| Date | 1 | 01~31 (BCD Code) |
| Hour | 1 | 00~23 (BCD Code) |
| Minute | 1 | 00~59 (BCD Code) |
| Second | 1 | 00~59 (BCD Code) |

If the current date and time is 2019/08/23 15:24:59, the **<Date Field>** should be **0x19 0x08 0x23 0x15 0x24 0x59** (BCD Code).

7.1.2 Set/Get 24-hr Self-Check Time (25-07)

The **Set/Get 24-hr Self-Check Time** command instructs the ViVOpay reader to set a specific time for 24-hr self-check in Coordinated Universal Time.

NOTE: If this command sets the 24-hr self-check time successfully, the reader reboots after responding.

Command Frame (Set 24hrs Self-Check Time)

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 | Byte 17 |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Hours | Minutes | CRC (LSB) | CRC (MSB) |
| ViVOtech 2\0 | 25h | 07h | 00h | 02h | BCD format, range 00-23h | BCD format, range 00-59h | | |

Examples:

1. Beijing Time, UTC+8, desired 24-hr self-check time is 2:30 in the morning,
 $2:30 \text{ (UTC+8)} - 8\text{hrs} = 18:30 \text{ (UTC)}$
 Hours = 18h,
 Minutes = 30h,
2. Pacific Daylight Time (PDT), UTC-7, desired 24-hr self-check time is 3:30 in the morning,
 $3:30 \text{ (UTC-7)} + 7 \text{ hrs} = 10:30 \text{ (UTC)}$
 Hours = 10h
 Minutes = 30h

Response Frame (Set 24-hr Self-Check Time)

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 25h | See status code table | 00h | 00h | | |

If the status code does not return **OK**, the command failed.

Command Frame (Get 24-hr Self-Check Time)

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 25h | 07h | 00h | 00h | | |

Response Frame (Get 24hrs Self-Check Time)

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 | Byte 17 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|------------|------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Hours | Minutes | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 25h | See status code table | 00h | 02h | BCD format | BCD format | | |

If the status code does not return **OK**, the command failed.

7.2 General Commands**7.2.1 Ping (18-01)**

The **Ping** command checks if the ViVOpay reader is connected to the host device. If the reader is connected, it responds with a valid response frame; otherwise there is no response.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 18h | 01h | 00h | 00h | B3 | CD |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 18h | 00h | 00h | 00h | FA | 83 |

7.2.2 Get DRS Info (C7-3A)

The **Get DRS Info** command retrieves information about hardware-related abnormal status and "tamper state" causes.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|----------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | 3Ah | 00h | 00h | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|----------------------------------|---------|---|----------------------|----------------------|----------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00 | Var | Data Object | | |

Data Object: <SourceBlk1>... [<SourceBlkN>]

- <SourceBlkX> is n bytes, Format is <SourceID><SourceLen><SourceData>
- <SourceID> is 1 byte
- <SourceLen> is 1 byte, it is the length of <SourceData>

The updated return data object follows:

- 0x00, no error
- 0x01, error is found
- 0xFE: the FW can't get the status.
- 0xFF: the byte is reserved.

| Item | SourceID | SourceLen | SourceData |
|--------------------------------|----------|-----------|---|
| K81 Self check Error | 00 | 1 | 00: Normal; 01: Error; 0xFE: Case tampered, do not check self check |
| TM4 Check Value Error | 0x01 | 1 | 00: Normal; 01: Error; 0xFE(TM4): Command not successful, possibly be due to TM4 not working, or tampering |
| TM4 Slave Check Value Error | 0x02 | 1 | 00: Normal; 01: Error; 0xFF: TM4 slave not present |
| Battery | 0x10 | 1 | 00: Normal 01: Battery low error 0xFF: Not checked |

| Item | SourceID | SourceLen | SourceData |
|--------------------|----------|-----------|--|
| Tamper Switch | 0x11 | 1 | Bit 0: Tamper Switch 0 (0-Normal, 1-Error) Bit 1: Tamper Switch 1 (0- Normal, 1-Error) Bit 2: Tamper Switch 2 (0- Normal, 1-Error) Bit 3: Tamper Switch 3 (0- Normal, 1-Error) Bit 4: Tamper Switch 4 (0- Normal, 1-Error) Bit 5: Tamper Switch 5 (0- Normal, 1-Error) Bit 6: Tamper Switch 6 (0- Normal; 1-Error) Bit 7: Tamper Switch 7 (0- Normal; 1-Error) 0xFE: Not checked Note: Most NEO 2.0 products have 4 pairs of tamper switches: Pair 1: Switch 0 and 2; Pair 2: Switch 1 and 3; Pair 3: Switch 4 and 6; Pair 4: Switch 5 and 7; But VP3600 only has 3 pairs: Pair 1: Switch 4 and 5; Pair 2: Switch 2 and 3; Pair 3: Switch 6 and 7; Value of Switch 0 and 1 can be ignored. |
| Temperature | 0x12 | 1 | 00: Normal 01: Temperature High or Low 0xFE: Not checked |
| Voltage | 0x 13 | 1 | 00: Normal 01: Voltage High or Low 0xFE: Not checked |
| TM4 removal sensor | 0x 14 | 1 | 00: Normal 01: TM4 sensor remove error 0xFE: Not checked |
| K81 RTC TOF | 0x 15 | 1 | 00: Normal; 01: RTC time over flow error 0xFE: Not checked |
| K81 clock CTF | 0x 16 | 1 | 00: Normal; 01: Clock frequency out of range 0xFE: Not checked |
| Other(Reserved2) | 0x 1F | 1 | 0xFF: (Not used) |

Example test data:**Command sent:** 5669564f746563683200c73a00000a84**Data received (no error):**

```

5669564f746563683200c70000210001000101000201ff10010011010012010013
01001401001501001601001f01ff780f
00 01 00
01 01 00
02 01 FF
10 01 00
11 01 00
12 01 00
13 01 00
14 01 00
15 01 00
16 01 00
1F 01 FF

```

Data received (with battery low error):

```

00 01 FE
01 01 00
02 01 FF
10 01 01
11 01 00
12 01 00
13 01 00
14 01 00
15 01 00
16 01 00
1F 01 FF

```

7.2.3 Set Poll Mode (01-01)

The **Set Poll Mode** command allows the host device to set the ViVOpay reader's polling mode. ViVOPay readers function in one of two polling modes: Auto Poll or Poll on Demand. The value is saved in nonvolatile memory, so developers must only send this command to change between modes.

ViVOpay products operate in Poll on Demand Mode by default. Use the Poll on Demand Mode when to have the reader to poll for cards only when the host device makes requests. In this mode the reader remains in the idle state with the RF field off until it receives an [Activate Transaction \(02-01 and 02-40\) command](#). After the transaction is completed (or the reader times out while polling) the reader returns to the idle state. This mode allows the host to send data to the reader before the card data is read, as required for EMV transactions.

In Auto Poll Mode, the RF field is always active and the reader continuously polls for the presence of a contactless transaction. There is no requirement for the host device to initiate a

transaction. When a supported contactless MAGSTRIPE card is detected, the track data can be sent out on the magstripe interface (if the reader supports it) or retrieved using the [Get Transaction Result \(03-00 and 03-40\) command](#). The Auto Poll Mode is required in environments where the reader is connected to a host via the host's magstripe interface.

Warning: EMEA UI is intended for use in the EMV or European environment, where the reader is not allowed to poll continuously (for example, to operate in Auto Poll Mode). The reader does NOT support Auto Poll while in EMEA UI mode and has the potential for aberrant or unstable behavior. The reader is not certified to work properly in this situation.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVotech2\0 | 01h | 01h | 00h | 01h | Poll Mode | | |

Poll Mode:

00h = Auto Poll

01h = Poll on Demand

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVotech2\0 | 01h | See status code table | 00h | 00h | | |

The Poll Mode has been set to the requested mode only if the response frame contains an **OK** status code. No data is returned in the response.

7.2.4 Control User Interface (01-02)

The **Control User Interface** command instructs the reader to display a message, change LED behavior, and beep. Each action is controlled independently by values in the data command's field. This allows users to instruct the reader to beep when it displays a message. To display messages, the reader must be in Poll on Demand Mode. Readers without a display use this command to control the buzzer and LEDs.

Note: In order to support backward compatibility, only VP6300 series products can fully support this command. Other products may support it partially.

There are three use cases depending on the LCD Message index number:

- **Indexes 00h to 07h** correspond to messages that automatically display on the reader. In most cases, the host device does not trigger these messages.
- **Indexes 08h to 0Bh** are messages triggered by the host.
- **Index FFh** indicates that the command is only setting the buzzer and/or LEDs. The host does not display a message.

After completing a successful transaction, the "Thank You" message remains on the LCD until the host sends a new **Control User Interface (01-02)** command.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|--------------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 01h | 02h | 00h | 04h | See Data Table | | |

The format and contents of the data field in the command frame are described in the following table.

Control User Interface Data

| Data Item | Length (bytes) | Description |
|-------------------|-------------------|--|
| LCD Message Index | 1 | <p>NOTE: For the complete list of possible messages, see the Default Message Index.</p> <p>Messages 00-07 are normally controlled by the reader.</p> <p>00: Idle Message (Welcome)</p> <p>01: Present card (Please Present Card)</p> <p>02: Time Out or Transaction cancel (No Card)</p> <p>03: Transaction between reader and card is in the middle (Processing...)</p> <p>04: Transaction Pass (Thank You)</p> <p>05: Transaction Fail (Fail)</p> <p>06: Amount (Amount \$ 0.00 Tap Card)</p> <p>07: Balance or Offline Available funds (Balance \$ 0.00)</p> <p>Messages 08-FF are controlled by the host device</p> <p>08: Insert or Swipe card (Use Chip & PIN)</p> <p>09: Try Again(Tap Again)</p> <p>0A: Tells the customer to present only one card (Present 1 card only)</p> <p>0B: Tells the customer to wait for authentication/authorization (Wait)</p> <p>FF indicates the command is setting the LED/Buzzer only.</p> |
| Beep Indicator | 1 | <p>00h: No beep</p> <p>01h: Single beep</p> <p>02h: Double beep</p> <p>03h: Three short beeps</p> <p>04h: Four short beeps</p> <p>05h: One long beep of 200 ms</p> <p>06h: One long beep of 400 ms</p> |

| Data Item | Length (bytes) | Description |
|------------|----------------|---|
| | | 07h: One long beep of 600 ms 08h: One long beep of 800 ms |
| LED Number | 1 | 00h: LED 0 (Power LED) 01h: LED 1 02h: LED 2 03h: LED 3 FFh: All LEDs Where the LEDs are numbered 0, 1, 2, 3 counting from the left. Note: Users can attempt to control the Power LED (LED 0) but other UI behavior takes control, so attempting to manipulate the Power LED in non-Pass-Through Mode has no effect. |
| LED Status | 1 | 00h: LED Off 01h: LED On |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 01h | See status code table | 00h | 00h | | |

If the status code does not return **OK**, the command failed.

7.2.5 Set/Get Source for RTC/LCD/Buzzer/LED (01-05)

The **Set/Get Source for RTC/LCD/Buzzer/LED** command sets up or retrieves the source for RTC/LCD/Buzzer/LED on the ViVOpay reader. The reader can be configured to use internal source or external source for RTC/Buzzer/LED control. If necessary, users can configure the reader to use both an internal and external source except for the RTC.

Note: Not all ViVOpay readers support all options. Pay careful attention to these details.

When the data length is 02h, the command is used to set up the source configuration for RTC/LCD/Buzzer/LED; when the data length is 0, the current source configuration is returned in the response frame.

Command Frame (Set Source)

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 | Byte 17 |
|-------------------------------|---------|-------------|-------------------|-------------------|-------------------------------|------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data Byte1 | Data Byte2 | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 01h | 05h | 00h | 02h | Bitmap for RTC/LCD/Buzzer/LED | | | |

Response Frame (Set Source)

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 01h | See status code table | 00h | 00h | | |

Command Frame (Get Source)

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 01h | 05h | 00h | 00h | | |

Response Frame (Get Source)

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 | Byte 17 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-------------------------------|------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data Byte1 | Data Byte2 | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 01h | See status code table | 00h | 02h | Bitmap for RTC/LCD/Buzzer/LED | | | |

Data Byte1 Definition:

| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|----------|----------|------|------|------|------|--------|------|
| Reserved | Reserved | RTC | | LCD | | Buzzer | |

| Bit1 Bit0 | Description |
|-----------|---|
| 00 | Don't use buzzer |
| 01 | Use buzzer from ViVOpay reader |
| 10 | Use buzzer from external source |
| 11 | Use buzzer from both reader and external source |

| Bit3 Bit2 | Description |
|-----------|--|
| 00 | Don't use LCD |
| 01 | Use LCD from ViVOPay reader |
| 10 | Use LCD from external source |
| 11 | Use LCD from both reader and external source |

| Bit5 Bit4 | Description |
|-----------|-----------------------------|
| 00 | Don't use RTC |
| 01 | Use RTC from ViVOPay reader |
| 10 | Not allowed |
| 11 | Not allowed |

Data Byte2 Definition:

| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|----------|------|----------|------|-----------|------|-----------------|------|
| Reserved | | Reserved | | Power LED | | Transaction LED | |

| Bit1 Bit0 | Description |
|-----------|--|
| 00 | Don't use transaction LED |
| 01 | Use transaction LED from ViVOPay reader |
| 10 | Use transaction LED from external source |
| 11 | Use transaction LED from both reader and external source |

| Bit3 Bit2 | Description |
|-----------|--|
| 00 | Don't use power LED |
| 01 | Use power LED from ViVOPay reader |
| 10 | Use power LED from external source |
| 11 | Use power LED from both reader and external source |

The Date/Time can be configured to use internal or external Date/Time, depending on the reader configuration. When the reader is configured to use internal time, the RTC (Real Time Clock) chip is needed on the reader and the date and time from the RTC chip is used. When the reader is configured to use external time, the host device needs to set up the RTC inside the ARM processor of the reader with the [Set Configuration](#) (04-00) command.

If configured to use the internal buzzer, the reader's buzzer is used to indicate the transaction progress. If configured to use external buzzer, an external buzzer is used and not the reader's buzzer.

If configured to use internal transaction LEDs, the reader's transaction LEDs are used to indicate the transaction progress. If configured to use external transaction LEDs, the transaction LEDs on the host device are used and not the reader's transaction LEDs.

If configured to use internal power LEDs, the reader's power LEDs are used to indicate the reader's power state. If configured to use external power LEDs, the power LEDs on the host are used.

Note: Users can attempt to control the Power LED, but other UI behavior takes control, so attempting to manipulate the Power LED in non-Pass-Through Mode has no effect. This means "Don't use power LED" and "Use power LED from external source" won't turn off the Power LED.

7.2.6 Set Configuration Defaults (04-09)

The **Set Configuration Defaults** command provides an external method for resetting parameters in non-volatile memory (NVM) to their default values. Group configurations are reset to factory defaults. Contact (ICC L2) CAPKs are erased.

When the reader receives this serial command an NVM Initialization function is called and the display shows the message.

Initializing
Please Wait

After completing initialization the display returns to the ready state message.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 04h | 09h | 00h | 00h | 87h | 30h |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 04 | 00h | 00h | 00h | A Eh | 16h |

This command does not modify the following data objects:

- Merchant Name and Location (tag 9F 4E)
- Transaction Category code (tag 9F 53)
- Terminal IFD Serial Number (tag FF F2)
- Application Capability (tag FF F3)
- Enable/Disable Burst Mode (tag FF F7)
- LCD font size (tag FF F9)
- LCD delay time (tag FF FA)
- Poll mode (tag DF 89 1B)

- Baud rate (tag FE 02)
- Boot up Message Enable (tag FE 03)
- Serial Number (tag DF 89 1A)
- UI Source Config (tag FE 05)
- Analog parameters (tag FE FE)

7.2.7 Set Configuration Defaults and Keep Encryption Key Command (04-0A)

The **Set Configuration Defaults and Keep Encryption Key** command provides an external method for resetting parameters in non-volatile memory (NVM) to their default values. When the reader receives this serial command it erases EEPROM (but retains encryption keys). After completing initialization, the reader reboots.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 04h | 0Ah | 00h | 00h | F7h | 46h |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 04 | 00h | 00h | 00h | A Eh | 16h |

This command does not modify the following data objects:

- RKI-KEK
- Data encryption Key
- Encrypt type
- Encrypt Status
- Serial Number (tag DF 89 1A)
- UI Source Config (tag FE 05)
- Analog parameters (tag FE FE)
- Baud rate (tag FE 02)

7.2.8 Set Configuration (04-00)

The **Set Configuration** command sets or changes the values of specified Tag Length Value (TLV) data objects in the reader. It can set parameters for Auto Poll and Poll on Demand Modes.

When the reader receives this command, it extracts the TLV encoded parameters from the data portion of the command and saves them to the default TLV Group in non-volatile

memory. If a TLV data object is incorrectly formatted, the reader stops processing the object. A single command may contain more than one TLV data object.

The **Set Configuration (04-00)** command is the only mechanism for setting the values of global configuration parameters.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 04h | 00h | | | TLV Data Objects | | |

The TLV data objects that this command can set are defined in the [Global Configuration Tags](#) table.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 04h | See status code table | 00h | 00h | | |

7.2.9 Get Configuration (03-02)

The **Get Configuration** command retrieves the values of the TLV global data objects and default group data objects (TLV Group 0) in the reader from the reader's nonvolatile memory. After sending 03-02, wait 800ms **after** the response is received before continuing.

Note: If a reader supports configurable application identifier (AIDs), the following applies:

1. The **Get Configuration** command may be used to retrieve global configuration tags and Group 0 configuration tags.
2. The **Get Configuration** command produces the same result as a **Get Configurable Group** command for group 0 (default). **Get Configuration** cannot return TLVs from other TLV Groups.
3. The **Get Configuration** command cannot return PayPass Group tags because PayPass does not use group 0.

When the reader receives this command, it returns the current values for all the parameters that can be set using the Set Configuration command. Each parameter is returned as a TLV data object. Floor limits for different AIDs are preceded by the TLV of the specific AID associated with that object.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 03h | 02h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 03h | See status code table | | | TLV Data Objects | | |

Refer to [Group Configuration Tags](#) for definitions of tags that this command can return.

7.2.10 Get Version: Protocol 2 (29-00)

The **Get Version: Protocol 2** command retrieves the ViVOpay firmware version number from the reader. The reader returns a response frame containing the firmware version information.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 29h | 00h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 29h | See status code table | 00h | String length | Version string | | |

1.1.2. Get Product Full Information (29-01)

The **Get Product Full Information** command retrieves all module information for NEO 2.0 products, including Contact and Contactless card modules.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Parameter | CRC (LSB) | CRC (MSB) |
| ViVOPayV2\0 | '29' | '01' | '00' | '01' | See data table | | |

Data Table

| Data Item | Length | Description |
|-----------------------------|--------|--|
| Card type of product module | 1 | 0: To show Contact Card module information 1: To show Contactless Card module information |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14... Byte 14+n+1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-----------------------|-------------------|-------------------|---------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOPayV2\0 | '29' | See status code table | '00' | 'XX' | `data format | | |

Response Frame Data Format: <Block 1><Block 2> ... <Block n>

Where:

Block format is: 1 byte Tag, 1 byte Length, and several bytes Value

If Contact card module information is returned, the block list is shown below:

| Block Item | Description | Tag | Length | Value (ASCII code) |
|------------|-----------------|-----|--------|--------------------|
| 1 | IFM-ID | 01 | N | 80173101 Rev.A |
| 2 | IFM-Hardware-ID | 02 | N | 80173111 Rev.A |
| 3 | IFM-Software-ID | 03 | N | 80173111 Rev.A |

If Contactless card module information is returned, the block list is shown below:

| Block Item | Description | Tag | Length | Value (example value for VP5300) (ASCII code) |
|------------|------------------------------|-----|--------|---|
| 1 | Firmware Version | 01 | N | VP5300 FW v1.00.S |
| 2 | PCD-ID | 02 | N | 80152100 Rev.A |
| 3 | PCD-Hardware-ID | 03 | N | 80152110 Rev.A |
| 4 | PCD-Software-ID | 04 | N | 80152120 Rev.A |
| 5 | Reader Tech Name and Version | 05 | N | 80152100 Rev.A |

| | | | | |
|----|---------------------------------|----|---|---|
| 6 | Application Type | 06 | N | Mastercard Contactless Reader Specification, v3.1.1 |
| 7 | Application Selection Mechanism | 07 | N | PPSE Module v1.2 |
| 8 | Master Card CL Kernel | 08 | N | MasterCard PayPass M/Chip 3.1 v1.3 |
| 9 | Operation System | 09 | N | uCOS3 v3.06 |
| 10 | Test Application | 0A | N | Java Lab Tool v2.07.08 |
| 11 | Test Configuration Data Sets | 0B | N | PayPass 3.1.1 Configurations v1.00 |
| 12 | Test Environment Interface | 0C | N | USB v2.0 |

7.2.11 Get USB Boot Loader Version (29-04)

The **Get USB Boot Loader Version** command retrieves the version of the USB Boot Loader.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 29h | 04h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 29h | See status code table | 00h | String length | Version string | | |

7.2.12 Get Contact EMV L2 Kernel Version (29-06)

The **Get Contact EMV L2 Kernel Version** command retrieves the Contact EMV L2 kernel library major version.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 29h | 06h | 00h | 00h | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 29h | See status code table | | | See Response Frame | | |

Response Frame Example: "EMV Common L2 V1.10"

7.2.13 Get Contact EMV L2 Kernel Version Detail (29-07)

The **Get Contact EMV L2 Kernel Version Detail** command retrieves the contact EMV L2 kernel library major and minor version.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 29h | 07h | 00h | 00h | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 29h | See status code table | | | See Response Frame | | |

Response Frame Example: "EMV Common L2 V1.10.037"

7.2.14 Get Contact EMV L2 Kernel Checksum (29-08)

The **Get Contact EMV L2 Kernel Checksum** command retrieves contact EMV L2 kernel library checksum. The checksum is based on SHA-1. Pass a data value (and a data length of 01) corresponding to the kernel for which a checksum is desired, as shown below; or else, pass no data (and a data length of 00) to get the Contact EMV L2 kernel checksum only.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 29h | 08h | 00h | 01h | Kernel | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 29h | See status code table | | | See Response Frame | | |

Response Frame Example: "8D 93 95 C5 88 F2 DE C9 AE 9D C8 BB 7B 98 41 A3 52 57 11 C6"

Kernel

| Index | Kernel |
|-------|---------------------|
| 0 | Contact EMV L2 |
| 1 | Contactless VISA |
| 2 | Contactless Master |
| 3 | Contactless DPays |
| 4 | Contactless Amex |
| 5 | Contactless Interac |

7.2.15 Get Contact EMV L2 Terminal Configuration Checksum (29-09)

The **Get Contact EMV L2 Terminal Configuration Checksum** command retrieves contact EMV L2 terminal data checksum. (Calculate tags 9F33, 9F35, 9F40, DF11, DF26, DF27, DFEE1E). The checksum uses SHA-1.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 29h | 09h | 00h | 00h | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 29h | See status code table | | | See Response Frame | | |

Response Frame Example: "B4 30 CE B9 FE 85 8B 00 B8 B2 7D 6F 56 EF 40 D4 2F 0E AF 8A"

7.2.16 Get UID of MCU (29-17)

The **Get UID of MCU** command retrieves the 16 byte UID of MCU.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 29h | 17h | 00h | 00h | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 29h | See status code table | | | See Response Frame | | |

Response Frame Data Example: "00 00 1D 00 61 26 1C 6B 75 31 45 4E 05 00 01 10"

7.2.17 Get TMS (29-20)

The **Get TMS** command retrieves information about TMS functions for the reader.

Note: Not all readers support TMS, but if the reader does support TMS, it will accept the command.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 29h | 20h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|--------------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 29h | See status code table | 00h | variable | See Response Frame Data Format | | |

Response Frame Data Format: <Block 1><Block 2> ... <Block n>

Where:

Block format is: 1 byte Tag, 1 byte Length, Multiple bytes Value

| Block Item | Details | Tag | Length (L means dynamic) | Value (format) |
|------------|---|-----|--------------------------------|---|
| 1 | RTC/UTC Date time | 01 | 6 | Y M D H M S |
| 2 | Reader Model Number (TS will set Model Number) | 02 | var. (<=32) | Ascii Code |
| 3 | Firmware Version | 03 | var. | |
| 4 | Hardware Version | 04 | var. | |
| 5 | Serial Number (TS should set SN) | 05 | 10 | Ascii Code |
| 6 | Last swipe UTC/RTC | 06 | 6 | Y M D H M S |
| 7 | Life time total swipe (# of Swipe) | 07 | 4 | Counter |
| 8 | Last Dip (ICC) UTC/RTC | 08 | 6 | Y M D H M S |
| 9 | Lifetime Total Dip (ICC) (# of Dip) | 09 | 4 | Counter |
| 10 | Last Tap (CL) UTC/RTC | 0A | 6 | Y M D H M S |
| 11 | Lifetime Total Tap (CL) (# of Dip) | 0B | 4 | Counter |
| 12 | Reboot Count | 0C | 4 | Counter |
| 13 | Reader Uptime since Last Reboot | 0D | 4 | Time (Seconds) |
| 14 | Reader Lifetime Uptime (Only register Power On Time) | 0E | 4 | Time (Hours) |
| 15 | Tamper Status | 0F | var. | Same as the response of Get DRS command |
| 16 | Insert / Remove ICC Card Counter | 10 | 8 | First 4: |

| | | | | |
|--|--|--|--|--------------------------------|
| | | | | Insert Card Counter |
| | | | | Last 4: Remove Card Counter |

7.2.18 Get Health Data from SCRP (29-30)

The **Get Health Data from SCRP** command retrieves Health Data from SCRP Only. This command is always valid in SCRP.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 29h | 30h | 0000h | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 – Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 29h | See status code table | 0062h | | See Data table | | |

| Data Item | Length (bytes) | Description |
|------------------------|----------------|--|
| Length Enc_Health_Data | 2 | Length of Enc_Health_Data |
| Enc_Health_Data | 96 | Enc [16 bytes Random Block_SN Block_FW_Version Block_HW_Version Block_SRED_Status Block_MSR_Availability n bytes Random] |

Note:

- Create 96 bytes Raw data: Enc [16 bytes Random | Block_SN | Block_FW_Version | Block_HW_Version | Block_SRED_Status | Block_MSR_Availability | n bytes Random]:
 - Block data format: 2 bytes length (MSB & LSB) + m bytes data
 - SN data is 10 bytes
 - FW_Version data: "SCRP Vx.yy.zzz.aaaa.S Demo" or "SCRP Vx.yy.zzz.aaaa.S Prod"
 - HW_Version data is same as response of 09-14 command.
 - SRED_Status data is 2 bytes
 - MSR_Availability data is 1 byte
- Use Data_Shared_Key and AES-128 CBC algorithm to Encrypt the Raw Data to be 96 bytes Enc_Health_Data.

7.2.19 Set Function/Hardware Enable/Disable (29-40)

The **Set Function/Hardware Enable/Disable** command enables and disables functions or hardware (such as EMV CT, EMV CL, MSR, Manual PAN Entry, PINpad, LCD, LED, and so on).

FunID (ITP Protocol) is **1Ah** (Expand MSR only to all related).

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|---------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOPayV2\0 | 29h | 40h | xx | xx | See send data field | | |

Data Table

| Data Item | Length | Description |
|-------------------------------|-----------|---|
| Function / Hardware ID Length | 2 | Length of Customer ID. Format: <Length H><Length L> (Big-endian) Current value is 00h 01h |
| Function / Hardware ID | n | ID from 0# |
| Requirement Length | 2 | Length of Requirement. Format: <Length H><Length L> (Big-endian) |
| Requirement | None or n | See Note |

Note:

| Function or Hardware ID | Requirement |
|-------------------------|---|
| 0# | MSR. Requirement Length is 00h 01h Requirement (1 byte): <ul style="list-style-type: none"> 00h – Disable 01h – Enable |
| 1# | EMV CT. Requirement Length is 00h 01h Requirement (1 byte): <ul style="list-style-type: none"> 00h – Disable 01h – Enable |
| 2# | EMV CL. Requirement Length is 00h 01h Requirement (1 byte): <ul style="list-style-type: none"> 00h – Disable 01h – Enable |
| 3# | Manual PAN Entry. Requirement Length is 00h 01h Requirement (1 byte): <ul style="list-style-type: none"> 00h – Disable (Only valid in Poll On Demand Mode) 01h – Enable |
| RFU | TBD |

Examples:

1. Data to Enable MSR: 00 01 00 00 01 01
2. Data to Disable Manual PAN Entry: 00 01 03 00 01 00
3. If the Function ID is 18# and the requirement is 2 bytes (data is 0F 11) the data is: 00 01 12 00 02 0F 11

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14- Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|----------------------|----------------------|-----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | 29h | See status code table | 00h | 00h | See Data table | | |

7.2.20 Get Function/Hardware Enable/Disable (29-41)

The **Get Function/Hardware Enable/Disable** command retrieves the current enabled or disabled status for a function or piece of hardware.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14- Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|------------------------|--------------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | 29h | 41h | xx | xx | See send data field | | |

Data Table

| Data Item | Length | Description |
|----------------------------------|--------|---|
| Function / Hardware ID Length | 2 | Length of Customer ID. Format: <Length H><Length L> (Big-endian) Current value is 00h 01h |
| Function / Hardware ID | n | ID from Note of 29-40 command |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14- Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|----------------------|----------------------|-----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | 29h | See status code table | xx | xx | See Data table | | |

Data Table

| Data Item | Length | Description |
|--------------------|-----------|---|
| Requirement Length | 2 | Length of Requirement. Format: <Length H><Length L> (Big-endian) |
| Requirement | None or n | See Note of 29-40 command |

7.2.21 Set Baud Rate (30-01)

The **Set Baud Rate** command instructs the ViVOpay reader to change its baud rate to the specified value. If the command frame is valid and the reader supports the specified baud rate, it returns an OK response and then switches to the specified baud rate. If the command frame is not valid, or an invalid baud rate parameter is specified, the reader returns an error response frame. The new baud rate is retained over power cycles.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTEch2\0 | 30h | 01h | 00h | 01h | Baud Rate Code | | |

Baud Rate Code

| Baud Rate Code | Baud Rate |
|------------------|-------------|
| 01h | 9600 baud |
| 02h | 19200 baud |
| 03h | 38400 baud |
| 04h | 57600 baud |
| 05h | 115200 baud |
| 06h ³ | 128000 baud |
| 07h ³ | 230400 baud |
| 08h ³ | 460800 baud |
| 09h ³ | 521000 baud |
| 0Ah ³ | 921600 baud |

Important: All other values for Baud Rate Code are invalid and should not be accepted by reader.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 30h | See status code table | 00h | 00h | | |

The reader switches baud rate only if the response frame contains an **OK** status code. No data is returned in the response.

7.2.22 Set Temporary Baud Rate (30-02)

This command instructs the ViVOPay reader to change its baud rate to the specified value temporarily. **After power up, the baud rate will return to the previous value.** If the command frame is valid and the reader supports the specified baud rate, it returns an OK response and then switches to the specified baud rate. If the command frame is not valid, or an invalid baud rate parameter is specified, the reader returns an error response frame.

Note: If the new baud rate is set by sending the **Set Temporary Baud Rate** command, then after power up, the baud rate will return to the previous value. If the new baud rate is set by sending the **Set Baud Rate** command the baud rate will still be the new value then power up.

³ Only some readers support 06h-0Ah, others do not accept 06h-0Ah and return a 0x05 error code.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 30h | 02h | 00h | 01h | Baud Rate Code | | |

Baud Rate Code

| Baud Rate Code | Baud Rate |
|------------------|-------------|
| 01h | 9600 baud |
| 02h | 19200 baud |
| 03h | 38400 baud |
| 04h | 57600 baud |
| 05h | 115200 baud |
| 06h ⁴ | 128000 baud |
| 07h ⁴ | 230400 baud |
| 08h ⁴ | 460800 baud |
| 09h ⁴ | 521000 baud |
| 0Ah ⁴ | 921600 baud |

Important: All other values for baud rate code are invalid and not accepted by readers.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 30h | See status code table | 00h | 00h | | |

The reader switches baud rate only if the response frame contains an **OK** status code. No data is returned in the response.

⁴ Only some readers support 06h-0Ah, others do not accept 06h-0Ah and return a 0x05 error code.

7.2.23 Set Baud Rate and Audio Level (30-03)

The **Set Baud Rate and Audio Level** command instructs a UniPay 1.5 to change baud rate to the specified value. If the command frame is valid and the reader supports the specified baud rate, it returns an **OK** response and switches to the specified baud rate. If the command frame is not valid, or an invalid baud rate parameter is specified then the reader returns an error response frame. The new baud rate is retained over power cycles.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 30h | 03h | 00h | 01h | Baud Rate Code | | |

| Baud Rate Code | Baud Rate | Note |
|----------------|-----------------------------|--|
| 01h | 9600 baud | Send data 0x55 0x55 0x66 before response |
| 02h | 9600 baud | Do not send data 0x55 0x55 0x66 before response |
| 03h | 2400 baud and level minimum | This value is used only for the Android SDK |
| 04h | 4800 baud and level minimum | This value is used only for the Android SDK |
| 05h | 9600 baud and level minimum | This value is used only for the Android SDK |
| 33h | 2400 baud and level 1 | This value is used only for the Android SDK |
| 34h | 4800 baud and level 1 | This value is used only for the Android SDK |
| 35h | 9600 baud and level 1 | This value is used only for the Android SDK |

The baud rate code 03h, 04h, 05h, 33h, 34h, and 35h only adjust baud rate and audio level when reader sends response to the host device and sets baud rate and level to 9600 and L1 after transmission is finished.

Important: All other values for baud rate code are invalid and should not be accepted by ViVOpay readers.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 30h | See status code table | 00h | 00h | | |

The reader switches baud rate only if the response frame contains an **OK** status code. No data is returned in the response.

7.2.24 Set Cable Type (32-02)

The **Set Cable Type** command sets the type of cable the reader uses; some readers have two types of cable, a so-called "long cable" and a "short cable." Some parameter calibrations differ according to cable type. The user can set the cable type accordingly.

If the cable type is not set, the reader considers the cable type as long by default.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 32h | 02h | 00h | 01h | Cable Type | | |

| Cable Type | Value |
|----------------------|-------|
| Long cable (default) | 0 |
| Short cable | 1 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 32h | See status code table | 00h | 00h | | |

7.2.25 Get Cable Type (32-01)

The **Get Cable Type** command returns the cable type set for the ViVOpay reader. If the cable type is not set, the reader considers the cable type as "long" by default.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 32h | 01h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 32h | 00h | 00h | 01h | Cable Type | | |

7.2.26 Get Serial Number (12-01)

Note: This command can only be used after the reader has completed a **Set Serial Number** command.

The **Get Serial Number** command instructs the ViVOpay reader to return the 15-digit serial number stored in its non-volatile memory. If a serial number has not been previously set in the reader, this command fails with a **Command Not Allowed** error status. If the command frame is not valid, the reader returns an error response frame.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 12h | 01h | 00h | 0Fh | | |

The serial number will be returned tail-padded with 0x00 to a length of 15.

Example:

TX: 56 69 56 4F 74 65 63 68 32 00 12 01 00 00 18 A5

RX: 56 69 56 4F 74 65 63 68 32 00 12 00 00 0F **36 33 30 5A 30 30 30 30 30 31 00 00 00 00 94 BC**

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 12h | See status code table | 00h | 0Fh | 15-digit Serial Number | | |

7.2.27 Get MAC Address (D1-1B)

The **Get MAC Address** command retrieves the ViVOpay reader's Ethernet / Wi-Fi / BLE MAC Address.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|--------------------------|---------|----------------|---------------|---------------|-------------------------|--------------|--------------|
| Header Tag & Protocol | Command | Sub Command | Length MSB | Length LSB | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2/0 | D1h | 1Bh | 00h | 01h | See Below Data Table | Varies | Varies |

| Length | Description |
|--------|-----------------------|
| 00 | Get Ethernet Address |
| 01 | Get WI-FI Address |
| 02 | Get Bluetooth Address |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-15 | Byte 16 | Byte 17 |
|--------------------------|---------|---|---------------|---------------|---------------------|--------------|--------------|
| Header Tag & Protocol | Command | Status | Length MSB | Length LSB | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2/0 | D1h | See status code table | 00h | 06h | MAC Address(hex) | Varies | Varies |

Example: If the reader receives a **Get Ethernet MAC Address** command and its address is 00:11:22:33:CD:AB, the response data is 00 11 22 33 CD AB.

7.2.28 Get Ethernet MAC Address (D1-11)

The **Get Ethernet MAC Address** command retrieves the Ethernet MAC address.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|--------------------------|---------|----------------|---------------|---------------|---------|--------------|--------------|
| Header Tag & Protocol | Command | Sub Command | Length MSB | Length LSB | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2/0 | D1h | 11h | 00h | 01h | 00 | Varies | Varies |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14- Byte 19 | Byte 14+n | Byte 14+n+1 |
|-----------------------|---------|---------------------------------------|---------------|---------------|---------------------------------|--------------|--------------|
| Header Tag & Protocol | Command | Status | Length MSB | Length LSB | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2/0 | D1h | See status code table | 00h | 06h | Ethernet MAC Address(hex) | Varies | Varies |

7.2.29 Set Network Configuration (D1-13)

The **Set Network Configuration** command sets the static network DHCP for the reader. If set static, the network configuration is saved. If set DHCP, reader will function as a DHCP client to get the network configuration from the host device.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14- Byte 13+n | Byte 19 | Byte 20 |
|-----------------------|---------|----------------|---------------|---------------|-----------------------|--------------|--------------|
| Header Tag & Protocol | Command | Sub Command | Length MSB | Length LSB | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2/0 | D1h | 13h | 00h | variable | See blow | Varies | Varies |

| Length | Description |
|----------|--|
| 2 | Determine to set static or DHCP. Zero-terminated ASCII value. 30h - Set DHCP 31h - Set static IP network configuration |
| Variable | IP address, zero-terminated ASCII. Maximum length is 16 bytes including null terminator. |
| Variable | Net mask, zero-terminated ASCII. Maximum length is 16 bytes including null terminator. |
| Variable | Gateway, zero-terminated ASCII. Maximum length is 16 bytes including null terminator. |
| Variable | DNS, zero-terminated ASCII. Maximum length is 16 bytes including null terminator. |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-----------------------|---------|---------------------------------------|---------------|---------------|--------------|--------------|
| Header Tag & Protocol | Command | Status | Length MSB | Length LSB | CRC (MSB) | CRC (LSB) |
| ViVOtech2/0 | D1h | See status code table | 00h | 00h | Varies | Varies |

Example:

Set ethernet configuration as:

IP address: 192.168.6.99

Net mask: 255.255.255.0

Gateway: 192.168.6.1

DNS: 192.168.6.17

TX: 56 69 56 4F 74 65 63 68 32 00 D1 13 00 36 31 00 31 39 32 2E 31 36 38 2E 36 2E 39 39 00 32 35 35 2E 32 35 35 2E 32 35 35 2E 30 00 31 39 32 2E 31 36 38 2E 36 2E 31 00 31 39 32 2E 31 36 38 2E 36 2E 31 37 00 04 8B

RX: 56 69 56 4F 74 65 63 68 32 00 D1 00 00 00 50 BA

7.2.30 Get Network Configuration (D1-15)

The **Get Network Configuration** command retrieves all network configurations for the ViVOpay reader.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-----------------------|---------|-------------|------------|------------|---------|-----------|-----------|
| Header Tag & Protocol | Command | Sub Command | Length MSB | Length LSB | Data | CRC (LSB) | CRC (MSB) |
| ViVOpay2/0 | D1h | 15h | 00h | 01h | 00 | Varies | Varies |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-Byte 19 | Byte 14+n | Byte 14+n+1 |
|-----------------------|---------|---------------------------------------|------------|------------|--------------------------|-----------|-------------|
| Header Tag & Protocol | Command | Status | Length MSB | Length LSB | Data | CRC (MSB) | CRC (LSB) |
| ViVOpay2/0 | D1h | See status code table | 00h | variable | Data (See D1-13 command) | Varies | Varies |

7.2.31 Set Wireless Work Mode (D1-19)

The **Set Wireless Work Mode** command sets the ViVOpay reader's wireless work mode (Wi-Fi or BLE). The specified configuration is saved in flash.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-15 | Byte 16 | Byte 17 |
|-----------------------|---------|-------------|------------|------------|------------|-----------|-----------|
| Header Tag & Protocol | Command | Sub Command | Length MSB | Length LSB | Data | CRC (LSB) | CRC (MSB) |
| ViVOpay2/0 | D1h | 19h | 00h | variable | See below | Varies | Varies |

| Length | Description |
|--------|--|
| 2 | Wireless work mode, zero-terminated ASCII value 30h – Set Wi-Fi TCP Server Mode 31h – Set Wi-Fi SSL Server Mode 32h – Set BLE Server Mode |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-----------------------|---------|---------------------------------------|------------|------------|-----------|-----------|
| Header Tag & Protocol | Command | Status | Length MSB | Length LSB | CRC (MSB) | CRC (LSB) |
| ViVOTech2/0 | D1h | See status code table | 00h | 00h | Varies | Varies |

7.2.32 Get Wireless Work Mode (D1-1A)

The **Get Wireless Work Mode** command retrieves the ViVOPay reader's wireless work mode.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-----------------------|---------|-------------|------------|------------|---------|-----------|-----------|
| Header Tag & Protocol | Command | Sub Command | Length MSB | Length LSB | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2/0 | D1h | 1Ah | 00h | 01h | 00 | Varies | Varies |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-15 | Byte 16 | Byte 17 |
|-----------------------|---------|---------------------------------------|------------|------------|--------------------------|-----------|-----------|
| Header Tag & Protocol | Command | Status | Length MSB | Length LSB | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2/0 | D1h | See status code table | 00h | Variable | Data (See D1-19 command) | Varies | Varies |

7.2.33 Set Wi-Fi Configuration (D1-17)

The **Set Wi-Fi Configuration** command to set the ViVOPay reader's Wi-Fi configuration. The specified configuration is saved in flash.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-Byte 13+n | Byte 19 | Byte 20 |
|-----------------------|---------|-------------|------------|------------|-------------------|-----------|-----------|
| Header Tag & Protocol | Command | Sub Command | Length MSB | Length LSB | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2/0 | D1h | 17h | 00h | variable | See blow | Varies | Varies |

| Length | Description |
|----------|---|
| 2 | Wi-Fi work mode, zero-terminated ASCII value. 30h – Set NULL Mode, Wi-Fi RF will be disabled 31h - Set Station Mode Default 32h – Set SoftAP Mode 33h – Set SoftAP+Station Mode (Default) |
| Variable | SSID of the target AP, zero-terminated ASCII. Maximum length is 32 bytes including null terminator. |
| Variable | Password of the target AP, zero-terminated ASCII. |

| | |
|----------|---|
| | Maximum length is 32 bytes including null terminator. |
| Variable | IP address, zero-terminated ASCII. Maximum length is 16 bytes including null terminator. |
| Variable | Net mask, zero-terminated ASCII. Maximum length is 16 bytes including null terminator. |
| Variable | Gateway, zero-terminated ASCII. Maximum length is 16 bytes including null terminator. |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-----------------------|---------|---------------------------------------|------------|------------|-----------|-----------|
| Header Tag & Protocol | Command | Status | Length MSB | Length LSB | CRC (MSB) | CRC (LSB) |
| ViVOTEch2/0 | D1h | See status code table | 00h | 00h | Varies | Varies |

Notes:

Wi-Fi works as a TCP Server, port number 1025.

Example:

Set Wi-Fi configuration as:

- AP SSID: idtech-ShangHai-1
- AP Password: id64954196
- IP Address: 192.168.6.119
- Net mask: 255.255.255.0
- Gateway: 192.168.6.1

TX: 56 69 56 4F 74 65 63 68 32 00 D1 17 00 28 33 00 69 64 74 65 63
68 37 00 69 64 36 34 39 35 34 31 39 36 00 31 39 32 2E 31 36 38 2E
31 2E 31 30 31 00 35 30 30 30 00 4F 22

RX: 56 69 56 4F 74 65 63 68 32 00 D1 00 00 00 BA 50

7.2.34 Get Wi-Fi Configuration (D1-18)

The **Get Wi-Fi Configuration** command retrieves all network configurations for the ViVOpay reader.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-----------------------|---------|-------------|------------|------------|---------|-----------|-----------|
| Header Tag & Protocol | Command | Sub Command | Length MSB | Length LSB | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | D1h | 18h | 00h | 01h | 00 | Varies | Varies |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14- Byte 19 | Byte 14+n | Byte 14+n+1 |
|-----------------------|---------|---------------------------------------|------------|------------|--------------------------|-----------|-------------|
| Header Tag & Protocol | Command | Status | Length MSB | Length LSB | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | D1h | See status code table | 00h | variable | Data (See D1-17 command) | Varies | Varies |

7.2.35 Set System Language (D1-27)

The **Set System Language** command sets the system language for NEO 2 products and currently supports English and Japanese.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 - 17 | Byte 18 | Byte 19 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | 'D1' | '27' | '00' | '02' | Data Table | | |

Data Table

| Data Item | Length | Description |
|---------------|--------|--|
| Language type | 2 | ASCII code: "EN" is English display message "JP" is Japanese display message |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-----------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | 'D1' | See status code table | 00 | 00 | | |

7.2.36 Set USB Mode Convert and Attribution (D1-30)

The **Set USB Mode Convert and Attribution** command to switch a reader's USB mode between USB-KB and USB-HID (ID TECH or SREDKey2 only) and set the related attribution. **FunID** (ITP Protocol) is 23h, 14h, AEh, 25h, 5Dh, and 24h.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14- Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------------------|--------------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | D1h | 30h | 00h | 01h or 05h | See send data field | | |

Data Table

| Data Item | Length | Description |
|---------------------------------|--------|--|
| USB mode | 1 | 0: USB-HID (NEO2 format) 1: USB-KB 2: USB-HID (SREDKey2 format) |
| Polling interval | 1 | Range: 1~255 Only valid when USB mode is '0' or '1' If Data length is 01h, this area does not exist. |
| USB-HID Suspend Function Switch | 1 | 0: Disable 1: Enable If Data length is 01h, this area does not exist. |
| USB-KB Upper Switch | 1 | 0: 'a' ~ 'z' 1: 'A' ~ 'Z' If Data length is 01h, this area does not exist. |
| USB-KB Foreign Keyboard ID | 1 | Range: 0x30~0x3A If Data length is 01h, this area does not exist. |

Note:

1. AppleVAS data length is 1.
2. SREDKey2 data length is 5.
3. When the USB mode or Polling interval is changed successfully, the reader automatically restarts.
4. If another three items are changed successfully, the reader does not restart.

5. Interface behavior difference:

| | Mode | NEO2 Products | SREDKey2 |
|---------|----------------|--|--|
| USB-KB | Auto Poll | Automatic Output On: CMD/RES: transferred in Control PIPE Data: output in Endpoint 1 automatically | CMD/RES: transferred in Control PIPE Data: output in Endpoint 1 automatically |
| | | Automatic Output Off: CMD/RES(Data): transferred in Control PIPE | |
| | Poll On Demand | CMD/RES(Data): transferred in Control PIPE | CMD/RES(Data): transferred in Control PIPE |
| USB-HID | Auto Poll | Automatic Output On: CMD/RES: transferred in Endpoint 1 Data: output in Endpoint 1 automatically | CMD/RES: transferred in Control PIPE Data: output in Endpoint 1 automatically |
| | | Automatic Output Off: CMD/RES(Data): transferred in Endpoint 1 | |
| | Poll On Demand | CMD/RES(Data): transferred in Endpoint 1 | CMD/RES(Data): transferred in Control PIPE |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | D1h | See status code table | 00h | 00h | | |

7.2.37 Get USB Mode Convert and Attribution (D1-31)

The **Get USB Mode Convert and Attribution** command gets the current USB Mode and attribution for a reader.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | D1h | 31h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14- Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|----------------------|----------------------|-----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | D1h | See status code table | 00h | 01h or 05h | See Data table | | |

Data Table

| Data Item | Length | Description |
|------------------------------------|--------|--|
| USB mode | 1 | 0: USB-HID (NEO2 format) 1: USB-KB 2: USB-HID (SREDKey2 format) |
| Polling interval | 1 | Range: 1~255 If Data length is 01h, this area does not exist. |
| USB-HID Suspend Function Switch | 1 | 0: Disable 1: Enable If Data length is 01h, this area does not exist. |
| USB-KB Upper Switch | 1 | 0: 'a' ~ 'z' 1: 'A' ~ 'Z' If Data length is 01h, this area does not exist. |
| USB-KB Foreign Keyboard ID | 1 | Range: 0x30~0x3A If Data length is 01h, this area does not exist. |

7.2.38 Wi-Fi SSL Connection Request (D1-1C, PCI Only)

The **Wi-Fi SSL Connection Request** command sends an SSL request to the SSL server.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14- 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|------------------------|--------------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | D1h | 1Ch | variable | variable | See send data field | | |

Command Data Field

| Length | Description |
|----------|--|
| Variable | Server IP address, zero terminated ASCII. Maximum length is 16 bytes including null terminator |
| 2 | Port number, HEX value, Big Endian |
| 1 | 00 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | D1h | See status code table | 00h | 00h | | |

Example: Server IP 192.168.6.126, port: 1443

OUT: 5669564f746563683200d11c00113139322e3136382e362e3132360005a300ebc9

IN: 5669564f746563683200d1000000ba50

7.2.39 Set Ethernet SSL Request Configuration (D2-01, PCI only)

The **Set Ethernet SSL Request Configuration** command sets the reader's SSL-request-to-SSL-client configuration.

This command saves the server IP and port number in the client's flash memory. When the reader powers on it connects automatically to the remote server.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|---------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOpayV3\0 | D2h | 01h | variable | variable | See send data field | | |

Command Data Field

| Length | Description |
|----------|--|
| Variable | Server IP address, zero terminated ASCII. Maximum length is 16 bytes including null terminator |
| 2 | Port number, HEX value, Big Endian |
| 1 | 00 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | D2h | See status code table | 00h | 00h | | |

Example: Server IP 192.168.6.126, port: 1443

OUT: 5669564f746563683200d20100113139322e3136382e362e31323600 05a300 4592

IN: 5669564f746563683200d2000000218c

7.2.40 Get Ethernet SSL Request Configuration (D2-02, PCI only)

The **Get Ethernet SSL Request Configuration** command retrieves the server's IP address and port number from flash memory.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOpayV3\0 | D2h | 02h | 00 | 01 | 00 | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-Byte 19 | Byte 14+n | Byte 14+n+1 |
|-----------------------|---------|---------------------------------------|------------|------------|--------------------------|-----------|-------------|
| Header Tag & Protocol | Command | Status | Length MSB | Length LSB | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2/0 | D2h | See status code table | 00h | variable | Data (See D2-01 command) | Varies | Varies |

Example: Server IP 192.168.6.126, port: 1443

7.2.41 Enable/Disable Communication Port (D2-03)

The **Enable/Disable Communication Port** command enables or disables a communication port. The reader must support the communication port; otherwise the command returns a Parameter Not Supported (0x06) response.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-Byte 15 | Byte 16 | Byte 17 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data Filed | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | D2h | 03h | 00h | 02h | | | |

Data Field

| Byte 1 | Byte 2 |
|---|-------------------------|
| 0x00: LPUART0, 0x01: LPUART1, 0x02: LPUART2, 0x03: LPUART3, 0x04: UsbHID, 0x05: ENET, 0x06: BluetoothBT, 0x07: MSR_LPUART, 0x08: PINPAD_LPUART, 0x09: SPI, 0x0a: DAEMON, 0x0b: WIFI, | 0x00: Off; 0x01: On; |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | D2h | See status code table | 00h | 00h | | |

7.2.42 Bootup Notification Command (14-01)

ViVOpay firmware has the ability to spontaneously transmit its version information over the serial port during bootup. In this way the reader can notify the POS that it is ready to communicate when the bootup process has finished.

The **Bootup Notification Command** toggles this feature on or off. The default status is **Disabled**.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-----------------------|---------|-------------|------------|------------|--------------|-----------|-----------|
| Header Tag & Protocol | Command | Sub Command | Length MSB | Length LSB | Data Objects | CRC (LSB) | CRC (MSB) |
| ViVOtech2/0 | 14h | 01h | 00h | 01h | See below | | |

Data Objects

| Data Field | Length (bytes) | Description |
|------------|----------------|--|
| Status | 1 | 0x00: Bootup notification is disabled. 0x01: Bootup notification (raw) is enabled. 0x02: Bootup notification (frame) is enabled. |

If bootup notification is enabled, the firmware transmits its platform info string one time at the end of bootup.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-----------------------|---------|---------------------------------------|------------|------------|-----------|-----------|
| Header Tag & Protocol | Command | Status | Length MSB | Length LSB | CRC (MSB) | CRC (LSB) |
| ViVOtech2/0 | 14h | See status code table | 00h | 00h | | |

Bootup Notification (Raw)

Platform Info: "NEO 2.0 v1.00"

Bootup Notification (Frame)

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14- Byte 27 | Byte 28 | Byte 29 |
|-----------------------|---------|-------------|------------|------------|---------------------|-----------|-----------|
| Header Tag & Protocol | Command | Sub Command | Length MSB | Length LSB | Platform Info | CRC (MSB) | CRC (LSB) |
| ViVOtech2/0 | 14h | 01h | 00h | 0Eh | | Varies | Varies |

7.2.43 Set Model Number (90-15)

The **Set Model Number** command sets the model number for NEO 2 products.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 – 17 | Byte 18 | Byte 19 |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOPayV2\0 | 90h | 15h | 00h | 0Xh | Data Table | | |

Data Table

| Data Item | Length | Description |
|------------------------|--------|--|
| Length of Model Number | 1 | Shows the length of the model number data. Cannot exceed 32 bytes. |
| Model Number | X | Model number data. |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOPayV2\0 | '90' | See status code table | 00 | 00 | | |

7.3 Configurable AID and Group Commands

7.3.1 Set Configurable AID (04-02)

The **Set Configurable AID** command creates or selects an AID for configuration or deletion. There are eight TLVs that can be included in this command, some of which are mandatory.

TLV Group Number: This number refers to the created group that contains the characteristics desired for this AID. Setting and configuring the TLV Group Number is explained below. The TLV Group Number must be configured first. A communicated AID is rejected if it refers to a non-existing group.

Registered Application Provider Identifier (RID): Optional; if provided, the RID number is used to reference the CA public key payment system. The reader uses the first five bytes of the AID if there is no RID provided.

System AIDs:

- must always include the TLV Group Number TLV as the FIRST TLV in the message.
- must always include the AID TLV as the SECOND TLV in the message.
- must never include the Application Flow TLV (DFEE4C) in the message
- must never include the RID TLV in the message

The FOUR remaining TLVs are all optional.

The reader contains System AIDs. These can be disabled but cannot be deleted.

User AIDs:

- must always include the TLV Group Number TLV as the FIRST TLV in the message.
- must always include the AID TLV as the SECOND TLV in the message.
- must always include the Application Flow TLV in the message

The FIVE remaining TLVs are all optional.

The DISABLE AID tag is ignored if included in a USER AID.

There are eight User AIDs in the system. These can be added (set) or deleted at the user's discretion.

- No User AID can have the same exact AID as a System AID.

In addition to the above requirements:

- All AIDs must reference an existing TLV Group (in the TLV Group Number TLV).
- Any AID with a Partial Select TLV must also include the Max AID Length TLV.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 04h | 02h | | | TLV Data Objects | | |

This command sets TLV Data Objects defined in the [AID Configuration Tags](#).

To set Configurable AID tags, the Application Identifier (9F06) and Group Number (FFE4) tags are mandatory.

Note: At present, the preferred means of disabling a System AID is **NOT** to include the FFE6 TLV. Instead, just issue a Delete AID command for a particular AID. This deletes a User AID OR disables a System AID.

If a **Set Configurable AID** command is sent without an FFE6 TLV, the reader enables the AID if it is not already enabled.

Finally, a **Set Configurable AID** command used for a User AID can include a FFE6 Disable AID Tag, but it is ignored. This tag is only used to set System AIDs.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|----------------------------------|---------|---------------------------------------|----------------------|----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 04h | See status code table | 00h | 00h | | |

7.3.2 Set Configurable Group (04-03)

The **Set Configurable Group** command creates or modifies a TLV Group. Configure a specific TLV Group by passing the TLVs with the desired functionality and a unique TLV Group Number to the reader. See below for TLVs that can be associated with a TLV Group. A TLV Group Number and at least one other TLV is required. The reader uses TLVs in the default TLV Group 0 for any TLVs not defined in the user-defined TLV Group.

The length of TLV Data can only be written up to 1024 bytes.

For M/Chip 3.0, Group 0 is not used. If you are configuring a group for M/Chip PayPass, then you should refer to the [PayPass Group Tags](#). Otherwise, refer to the [Group Configuration Tags](#).

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTECH2\0 | 04h | 03h | | | TLV Data Objects | | |

When the TLVs in Group 0 (the Default Group) are changed, the reader retains and uses the old values of any TLVs not included in the **Set Configurable Group** command. When any other Groups are changed, the reader discards existing TLVs not in the current **Set Configurable Group** command.

The implication of the statement above is that if you are configuring a group for PayPass, you must configure all of the necessary tags, as they will not default to Group 0 tags.

To set the TDOL TLV, pass on the desired values in the TLV. To disable the default TDOL, send a TDOL TLV with the Length set to zero and no Value field included. This instructs the reader to delete any existing TDOL list for this group.

The [Configurable Group Tags](#) table (for non-PayPass applications) lists the TLV Data Objects possible to set with this command; also see [PayPass Group Tags](#).

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 04h | See status code table | 00h | 00h | | |

7.3.3 Get Configurable AID (03-04)

The **Get Configurable AID** command returns the configurable parameters for a User AID. The user **MUST** send an AID TLV both in the command and as the first TLV in the command. The reader then returns all tags associated with that User AID in the response.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 03h | 04h | | | TLV Data Objects | | |

The command **MUST** include the TLV below. The command should **NOT** include any other TLV.

| | | | | | |
|------|--|-----------|---|---|--------|
| 9F06 | Application Identifier (AID) – host device | MANDATORY | Identifies the application as described in ISO/IEC 7816-5. Note: This is the ONLY TLV in this command. | b | 5 – 16 |
|------|--|-----------|---|---|--------|

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 03h | See status code table | | | TLV Data Objects | | |

The optional data objects encoded as TLVs returned in the response frame's data section are the same as those listed in the [AID Configuration Tags](#). The reader returns ALL TLV associated with this AID in its response.

If an AID is requested and the reader fails to find it in its database, the reader returns the AID TLV itself and NO additional arguments. This indicates that the command was correct with the

proper argument, but there was no match in the reader's database. The reader does NOT indicate an error situation.

If the user requests a System AID that is currently disabled, the reader returns the AID TLVs, but appends the FFE6 TLV, showing that the AID is currently disabled.

7.3.4 Get Configurable Group (03-06)

The **Get Configurable Group** command returns all TLVs associated the specified Configurable Group. A configurable Group Tag must be included as the ONLY TLV in this command. The response should contain all of the Tags associated with this configurable Group.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 03h | 06h | | | TLV Data Objects | | |

The following TLV MUST be encoded in the command; it is the ONLY tag included in the command.

| | | | | | |
|---------------------|--------------|------|---|----|---|
| FFE4 ^[1] | Group Number | MAND | The group that contains the properties for this AID. Note: This must be the ONLY TLV in Data Field. | n2 | 1 |
|---------------------|--------------|------|---|----|---|

^[1] These objects use proprietary tags. The use of these tags should be restricted to the serial interface. After the reader has received these values and saved them in memory, it should dispose of the tags (and not keep them associated with these two values).

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 03h | See status code table | | | TLV Data Objects | | |

The data objects encoded as TLVs returned in the response frame's data section are listed in the [Group TLV Objects Table](#).

The reader returns an error if the user requests an illegal Group.

If the user requests a valid Group number but the Group does NOT exist, the reader returns the regular response but only includes the Group Number TLV (no other TLV is included). This signifies that the user has requested a valid number but no Group has been assigned to it.

Note: For a PayPass group, if a TLV data item is not present in the Group it will not be present in the data this command returns. However, this does not mean that particular TLV has no value in order to perform a transaction; the PayPass Kernel may have a hard-coded value for the TLV, which it uses in a transaction if the TLV is not present in the Group or in Activate Transaction data. The data items for which the PayPass kernel has hard-coded default values are listed in in the section on [PayPass Group Configuration TLVs with Hard-Coded Values in Kernel](#).

7.3.5 Delete Configurable AID (04-04)

The **Delete Configurable AID** command deletes a configurable AID. It is MANDATORY to include the AID TLV of the AID to remove. Do not include other TLVs.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVotech2\0 | 04h | 04h | | | TLV Data Objects | | |

The Data Object encoded as TLV that this command can set is below.

| | | | | | |
|------|--|-----------|--|---|--------|
| 9F06 | Application Identifier (AID) – host device | MANDATORY | Identifies the application as described in ISO/IEC 7816-5. Note: This is the ONLY TLV in this command. | b | 5 – 16 |
|------|--|-----------|--|---|--------|

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVotech2\0 | 04h | See status code table | 00h | 00h | | |

The user may NOT delete a System AID. If this command is used on a System AID, the reader disables that System AID but does not delete it. That System AID can be restored at any point

by using the Set AID command on it. Until that point it does not function but continues to reside in the reader's database.

When deleting an AID, the reader returns an OK response for successful operations. If it failed to find a matching AID, it returns 60h (data does not exist).

7.3.6 Delete Configurable Group (04-05)

The **Delete Configurable Group** command deletes a configurable Group. A deleted Group can no longer load the parameters for a transaction.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 04h | 05h | | | TLV Data Objects | | |

It is MANDATORY to include the Group Number TLV of the Group the user wishes to delete. No other TLVs should be included.

| | | | | | |
|---------------------|-----------------|------|---|----|---|
| FFE4 ^[1] | Group Number | MAND | The group that contains the properties for AID Note: This must be the ONLY TLV in Data Field. | n2 | 1 |
|---------------------|-----------------|------|---|----|---|

^[1] These objects use proprietary tags. The use of these tags should be restricted to the serial interface. After the reader has received these values and saved them in memory, the Host device should dispose of the tags.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|----------------------------------|---------|---------------------------------------|-------------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 04h | See status code table | 00h | 00h | | |

Do NOT delete the Default Group 0. The reader does not allow this command, and does NOT disable Group 0; instead it returns an error.

If the Group is not a valid Group Number this returns an error.

Finally, if the reader has ANY AID that references this Group, it does NOT delete the Group. It returns an error. That is, ONLY Groups that are NOT referenced by existing AID can be deleted. In this situation, the user must first delete or modify these AIDs, and then delete the Group.

7.3.7 Get All AIDs (03-05)

The **Get All AIDs** command returns all AIDs in the reader. Use this command to verify configured AIDs or to determine which System AIDs are in the reader.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 03h | 05h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 03h | See status code table | | | TLV Data Objects | | |

The GAA command should only return AID Tags and no other data objects. The reader sends out ALL TLVs associated with each AID.

The reader sends one or more frames with all the AID TLVs in it. Each AID grouping begins with the Group Number TLV that this AID uses. The user can use this fact to parse between the AID groups passed back to the POS.

7.3.8 Get All Groups (03-07)

The **Get All Groups** command returns all Groups in the reader. Use this command to verify all configured Groups in the reader.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 03h | 07h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 03h | See status code table | | | TLV Data Objects | | |

The GAG command should only return Group Tags and no other data objects. See the [Group Configuration Tags](#) table for details.

The reader sends one or more frames with all the Group TLVs in it. Each Group begins with the Group Number TLV for the Group in question. The user can use this fact to parse between the Groups passed back to the POS.

7.4 Transaction Related Commands: Contact

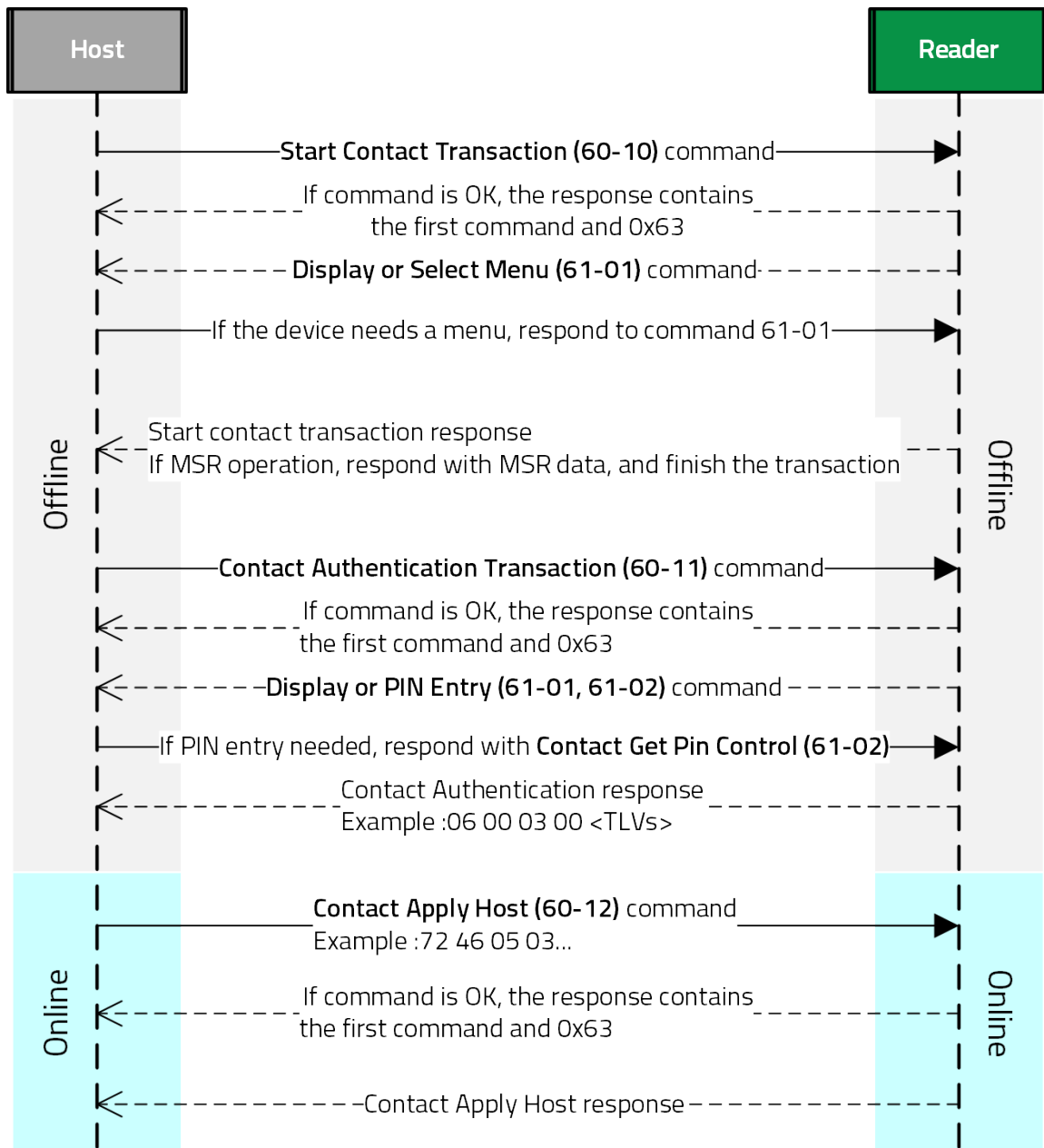
This section describes contact EMV commands applicable to NEO products that support contact EMV.

7.4.1 Setting transaction parameters

Use the following commands to set transaction parameters.

- 1. Send Contact Set Application Data (60-03)**
- 2. Send Contact Set Terminal Data (60-06)**
- 3. Send Contact Set CA Public Key (60-08)**
- 4. Send Contact Set Certification Revocation List (60-0E)**

7.4.2 Contact EMV L2 Transaction Flow



1. The host device sends a **Start Transaction** command to the reader.
2. The reader responds with a **Send Authenticate Transaction** command.
3. If the reader received an online request from the host, send a **Apply Host Response** command.
4. When a transaction is completed, the interface outputs a TLV format data list and the transaction result (approve, or decline, or...) will display on the LCD.
5. To review the transaction result, send a **Retrieve Transaction Result** command. The reader outputs the TLV format data; refer to the Option Data list.

Retrieve EMV Level Two Version Number command can be sent at any time.

Cancel Transaction command (05-01) can be sent at any time if you want to terminate a transaction.

7.4.3 Contact Retrieve Application Data (60-01)

The **Contact Retrieve Application Data** command returns the User AID parameters. The host device must send the AID in the command. The reader then returns all tags associated with that User AID in response.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTEch2\0 | 60h | 01h | | | Data Objects | | |

Data Objects: <5~16 bytes AID>

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|--------------------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTEch2\0 | 60h | See status code table | | | See Response Frame Data Format | | |

Response Frame Data Format: <TagCounterL><TagCounterH><TLV1><TLV2>...<TLVn>

Where: <TagCounterL><TagCounterH> is the number of <TLV> tags.

Note: If the AID List or Application Data does not exist, the response status code is 0x60.

7.4.4 Contact Remove Application Data (60-02)

The **Contact Remove Application Data** command deletes a configurable AID. It is mandatory to include the AID TLV of the AID to be removed.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTEch2\0 | 60h | 02h | | | Data Objects | | |

Data Objects: <5~16 bytes AID>

If the length is 00, the request removes all Application Data.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|--------------------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | See status code table | | | See Response Frame Data Format | | |

Note:

- If the AID List or Application Data does not exist, the response status code is 0x60.
- If there is a format error, the response status code is 0x05.

7.4.5 Contact Set Application Data (60-03)

The **Contact Set Application Data** command creates a new AID configuration, up to a maximum of 20 sets.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 60h | 03h | | | Data Objects | | |

Data Objects: <AID_LenL><AID_LenH><5~16 bytes

AID><TagCounterL><TagCounterH><TLV1><TLV2>...<TLVn>.

Where: <TagCounterL><TagCounterH> is the Number of <TLV>.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|--------------------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | See status code table | | | See Response Frame Data Format | | |

Note:

- If there was a <TLV> format error, the response status code is 0x05.
- If AID List is full (MAX is 20), the response status code is 0x61.

Application Data List Example

| Data ID | Tag | Value name | Length (Byte) | Data |
|---------|--------|--|---------------|-------------------|
| 1 | 9F01 | Acquirer identifier | 6 | 56 49 53 41 30 30 |
| 2 | 5F57 | Account type | 1 | 00 |
| 3 | 5F2A | Transaction Currency | 2 | 08 40 |
| 4 | 9F09 | Host device application version number | 2 | 00 96 |
| 5 | 5F36 | Transaction currency exponent | 1 | 02 |
| 6 | 9F1B | Host device floor limit | 4 | 00 00 3A 98 |
| 7 | DF25 | Default DDOL | Var | 9F 37 04 |
| 8 | DF28 | Default TDOL | Var | 9F 08 02 |
| 9 | DFEE15 | ASI | 1 | 01 |
| 10 | DF13 | TAC-Default | 5 | 00 00 00 00 00 |
| 11 | DF14 | TAC-Denial | 5 | 00 00 00 00 00 |
| 12 | DF15 | TAC-Online | 5 | 00 00 00 00 00 |
| 13 | DF18 | Target percentage for random transaction selection | 1 | 00 |
| 14 | DF17 | Threshold value for biased random selection | 4 | 00 00 27 10 |
| 15 | DF19 | Maximum target percentage for random transaction selection | 1 | 00 |

7.4.6 Contact Retrieve Terminal Data (60-04)

The **Contact Retrieve Terminal Data** command returns all TLVs containing host device configuration data.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 60h | 04h | 00h | 00h | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|--------------------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 60h | See status code table | | | See Response Frame Data Format | | |

Response Frame Data Format: <TagCounterL><TagCounterH><TLV1><TLV2>...<TLVn>

Note:

- <TagCounterL><TagCounterH> is the number of <TLV> tags.
- If host data does not exist, the response status code is 0x60.

7.4.7 Contact Remove Terminal Data (60-05)

The **Contact Remove Terminal Data** command deletes host device data.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 60h | 05h | 00h | 00h | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|--------------------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 60h | See status code table | | | See Response Frame Data Format | | |

If the host data does not exist, the response status code is 0x60.

7.4.8 Contact Set Terminal Data (60-06)

The **Contact Set Terminal Data** command creates new host device data according to the TLVs passed in. Host data is mandatory and seldom changes. It represents configuration data that usually gets set one time in pre-production and never changes after.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 60h | 06h | | | Data Objects | | |

Data Objects: <TagCounterL><TagCounterH><TLV1><TLV2>...<TLVn>.

Where: <TagCounterL><TagCounterH>: the Number of <TLV> tags.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|--------------------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | See status code table | | | See Response Frame Data Format | | |

Note:

If a <TLV> format was bad, the response status code is 0x05. If flash error, the response status code is 0x62.

Host Data List Example

| Data ID | Tag | Value name | Length (Byte) | Data |
|---------|--------|---|----------------|--|
| 1 | 5F36 | Transaction currency exponent | 1 | 02 |
| 2 | 9F1A | Host device county code | 2 | 08 40 |
| 3 | 9F35 | Host device type | 1 | 21 |
| 4 | 9F33 | Host device capability | 3 | 60 28 C8 |
| 5 | 9F40 | Additional host device capability | 5 | F0 00 F0 A0 01 |
| 6 | 9F1E | IFD serial number | 8 | 54 65 72 6D 69 6E 61 6C |
| 7 | 9F15 | Merchant category code | 2 | 12 34 |
| 8 | 9F16 | Merchant identifier | 15 | 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 |
| 9 | 9F1C | Host device identification | 8 | 38 37 36 35 34 33 32 31 |
| 10 | 9F4E | Merchant name and location | <=64 | 31 30 37 32 31 20 57 61 6C 6B 65 72 20 53 74 2E 20 43 79 70 72 65 73 73 2C 20 43 41 20 2C 55 53 41 2E |
| 11 | DF26 | Host device supports CRL | 1 | 01 |
| 12 | DF10 | Language | Var. up to 128 | 65 6E 66 72 65 73 7A 68 |
| 13 | DF11 | Support transaction log | 1 | 00 |
| 14 | DF27 | support exception file | 1 | 00 |
| 15 | DFEE15 | Host device support ASI | 1 | 01 |
| 16 | DFEE16 | Host device encrypt mode | 1 | 00 |
| 17 | DFEE17 | Host device entry mode for ICC | 1 | 07 |
| 18 | DFEE18 | Host device encrypt mode for MSR | 1 | 80 |
| 19 | DFEE1E | Contact host device configuration | 8 | D0 DC 20 D0 C4 1E 16 00 |
| 20 | DFEE1F | Issuer script limit | 1 | 80 |
| 21 | DFEE1B | ARC define | 8 | 30 30 30 31 35 31 30 30 |
| 22 | DFEE20 | ICC power on detect waiting time | 1 | 3C |
| 23 | DFEE21 | ICC L1 waiting time | 1 | 0A |
| 24 | DFEE22 | Driver waiting time. byte 1 -> Menu. byte 2 -> Get Pln. byte 3 -> MSR | 3 | 32 3C 3C |

The Contact Common EMV L2 comes with four approved configurations as shown below (1C, 2C, 3C, 4C); these correspond to the scenarios defined by EMVCo in tag 9F35. Parameters marked as Major Parameters usually cannot be changed without causing a checksum error, although certain flag bits (see tables below) can be changed. The Major Parameters are considered read-only because these are the settings the reader was certified with for EMV L2 certification.

| Terminal configuration | | | | | |
|------------------------|--------|----------------------------|----------------------------|----------------------------|----------------------------|
| Identification | Tag | 1C | 2C | 3C | 4C |
| Major parameters | 9F33 | 60 F8 C8 | 60 28 C8 | 60 D8 C8 | 60 08 C8 |
| | 9F35 | 22 | 21 | 25 | 25 |
| | 9F40 | F0 00 F0 A0 01 | F0 00 F0 A0 01 | 60 00 F0 50 01 | 60 00 F0 50 01 |
| | DF11 | 01 | 00 | 01 | 01 |
| | DF26 | 01 | 01 | 01 | 01 |
| | DF27 | 00 | 00 | 00 | 00 |
| | DFEE1E | F0 DC 3C F0 C2 9E 96 00 | D0 DC 20 D0 C4 1E 16 00 | F0 DC 24 F0 C2 0E 16 00 | D0 9C 20 F0 C2 0E 16 00 |

| Tag | Description | | | | | | | | | | Length |
|------|--------------------------|----|----|----|----|----|----|----|---|--------|--------|
| 9F33 | Host device Capabilities | | | | | | | | | | 3 |
| | Byte 1 | | | | | | | | | | |
| | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning | Change | |
| | 1 | x | x | x | x | x | x | x | Manual key entry | | |
| | x | 1 | x | x | x | x | x | x | Magnetic stripe | | |
| | x | x | 1 | x | x | x | x | x | IC with contacts | Major | |
| | x | x | x | 0 | x | x | x | x | RFU | | |
| | x | x | x | x | 0 | x | x | x | RFU | | |
| | x | x | x | x | x | 0 | x | x | RFU | | |
| | x | x | x | x | x | x | 0 | x | RFU | | |
| | x | x | x | x | x | x | x | 0 | RFU | | |
| | Byte 2 | | | | | | | | | | |
| | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning | Change | |
| | 1 | x | x | x | x | x | x | x | Plaintext PIN for IC verification | Major | |
| | x | 1 | x | x | x | x | X | x | Enciphered PIN for online verification | Major | |
| | x | x | 1 | x | x | x | X | x | Signature(paper) | Major | |
| | x | x | x | 1 | x | x | X | x | Enciphered PIN for offline verification | Major | |
| | x | x | x | x | 1 | x | X | x | No CVM Required | Major | |
| | x | x | x | x | x | 0 | x | x | RFU | | |
| | x | x | x | x | x | x | 0 | x | RFU | | |
| | x | x | x | x | x | x | X | 0 | RFU | | |
| | Byte 3 | | | | | | | | | | |
| | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning | Change | |
| | 1 | x | x | x | x | x | x | x | SDA | Major | |
| | X | 1 | x | x | x | x | x | x | DDA | Major | |
| | X | x | 1 | x | x | x | x | x | Card capture | | |
| | x | x | x | 0 | x | x | x | x | RFU | | |
| | x | x | x | x | 1 | x | x | X | CDA | Major | |
| | x | x | x | x | x | 0 | x | x | RFU | | |

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------------------------|---|-------------|-----------------------|----------|------------|--------|-----------------|--|--------|---------|--------|-------------|----|----|----|---|--------------------------------|----|----|------|-------|--------------|----|----|----|---|-------------------|---|---|-------|-------|-------------|----|----|----|---|--------------------------------|----|----|----------|-------|--------------|----|----|----|---|---|---|---|----------|-------|---|---|---|---|---|---|---|---|---------|--|---|---|---|---|---|---|---|---|----------|--|---|---|---|---|---|---|---|---|---------|--|---|---|---|---|---|---|---|---|----------------|--|----|----|----|----|----|----|----|----|---------|--------|---|---|---|---|---|---|---|---|--------------|--|---|---|---|---|---|---|---|---|-----|--|---|---|---|---|---|---|---|---|-----|--|---|---|---|---|---|---|---|---|-----|--|---|---|---|---|---|---|---|---|-----|--|---|---|---|---|---|---|---|---|-----|--|---|---|---|---|---|---|---|---|-----|--|---|---|---|---|---|---|---|---|-----|--|----|----|----|----|----|----|----|----|---------|--------|---|---|---|---|---|---|---|---|--------------|--|---|---|---|---|---|---|---|---|--|--|---|---|---|---|---|---|---|---|--------------|--|---|---|---|---|---|---|---|---|---------------|--|---|---|---|---|---|---|---|---|-----|--|---|
| | <table><tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>x</td><td>RFU</td><td></td></tr><tr><td>x</td><td>x</td><td>x</td><td>x</td><td>X</td><td>X</td><td>x</td><td>0</td><td>RFU</td><td></td></tr></table> | x | x | x | x | x | x | 0 | x | RFU | | x | x | x | x | X | X | x | 0 | RFU | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | x | x | x | x | x | 0 | x | RFU | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | x | x | x | X | X | x | 0 | RFU | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 9F35 | <div>Host device Type</div> <table><tr><td>Environment</td><td>Financial Institution</td><td>Merchant</td><td>Cardholder</td><td>Change</td></tr><tr><td>Attended</td><td></td><td></td><td></td><td>Major</td></tr><tr><td>Online only</td><td>11</td><td>21</td><td>--</td><td></td></tr><tr><td>Offline with online capability</td><td>12</td><td>22</td><td>--</td><td></td></tr><tr><td>Offline only</td><td>13</td><td>23</td><td>--</td><td></td></tr><tr><td>Unattended</td><td></td><td></td><td></td><td>Major</td></tr><tr><td>Online only</td><td>14</td><td>24</td><td>34</td><td></td></tr><tr><td>Offline with online capability</td><td>15</td><td>25</td><td>35</td><td></td></tr><tr><td>Offline only</td><td>16</td><td>26</td><td>36</td><td></td></tr></table> | Environment | Financial Institution | Merchant | Cardholder | Change | Attended | | | | Major | Online only | 11 | 21 | -- | | Offline with online capability | 12 | 22 | -- | | Offline only | 13 | 23 | -- | | Unattended | | | | Major | Online only | 14 | 24 | 34 | | Offline with online capability | 15 | 25 | 35 | | Offline only | 16 | 26 | 36 | | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Environment | Financial Institution | Merchant | Cardholder | Change | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Attended | | | | Major | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Online only | 11 | 21 | -- | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Offline with online capability | 12 | 22 | -- | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Offline only | 13 | 23 | -- | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Unattended | | | | Major | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Online only | 14 | 24 | 34 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Offline with online capability | 15 | 25 | 35 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Offline only | 16 | 26 | 36 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 9F40 | <div>Additional Host device Capabilities</div> <div>Byte 1</div> <table><tr><td>b1</td><td>b2</td><td>b3</td><td>b4</td><td>b5</td><td>b6</td><td>b7</td><td>b8</td><td>Meaning</td><td>Change</td></tr><tr><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Cash</td><td>Major</td></tr><tr><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Goods</td><td>Major</td></tr><tr><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Services</td><td>Major</td></tr><tr><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Cashback</td><td>Major</td></tr><tr><td>x</td><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>Inquiry</td><td></td></tr><tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>Transfer</td><td></td></tr><tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>Payment</td><td></td></tr><tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>1</td><td>Administrative</td><td></td></tr></table> <div>Byte 2</div> <table><tr><td>b8</td><td>b7</td><td>b6</td><td>b5</td><td>b4</td><td>b3</td><td>b2</td><td>b1</td><td>Meaning</td><td>Change</td></tr><tr><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Cash Deposit</td><td></td></tr><tr><td>x</td><td>0</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>RFU</td><td></td></tr><tr><td>x</td><td>x</td><td>0</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>RFU</td><td></td></tr><tr><td>x</td><td>x</td><td>x</td><td>0</td><td>x</td><td>x</td><td>x</td><td>x</td><td>RFU</td><td></td></tr><tr><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>x</td><td>x</td><td>x</td><td>RFU</td><td></td></tr><tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>x</td><td>x</td><td>RFU</td><td></td></tr><tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>x</td><td>RFU</td><td></td></tr><tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>RFU</td><td></td></tr></table> <div>Byte 3</div> <table><tr><td>b8</td><td>b7</td><td>b6</td><td>b5</td><td>b4</td><td>b3</td><td>b2</td><td>b1</td><td>Meaning</td><td>Change</td></tr><tr><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Numeric keys</td><td></td></tr><tr><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Alphabetic and special characters keys</td><td></td></tr><tr><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Command keys</td><td></td></tr><tr><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Function Keys</td><td></td></tr><tr><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>x</td><td>x</td><td>x</td><td>RFU</td><td></td></tr></table> | b1 | b2 | b3 | b4 | b5 | b6 | b7 | b8 | Meaning | Change | 1 | x | x | x | x | x | x | x | Cash | Major | x | 1 | x | x | x | x | x | x | Goods | Major | x | x | 1 | x | x | x | x | x | Services | Major | x | x | x | 1 | x | x | x | x | Cashback | Major | x | x | x | x | 1 | x | x | x | Inquiry | | x | x | x | x | x | 1 | x | x | Transfer | | x | x | x | x | x | x | 1 | x | Payment | | x | x | x | x | x | x | x | 1 | Administrative | | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning | Change | 1 | x | x | x | x | x | x | x | Cash Deposit | | x | 0 | x | x | x | x | x | x | RFU | | x | x | 0 | x | x | x | x | x | RFU | | x | x | x | 0 | x | x | x | x | RFU | | x | x | x | x | 0 | x | x | x | RFU | | x | x | x | x | x | 0 | x | x | RFU | | x | x | x | x | x | x | 0 | x | RFU | | x | x | x | x | x | x | x | 0 | RFU | | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning | Change | 1 | x | x | x | x | x | x | x | Numeric keys | | x | 1 | x | x | x | x | x | x | Alphabetic and special characters keys | | x | x | 1 | x | x | x | x | x | Command keys | | x | x | x | 1 | x | x | x | x | Function Keys | | x | x | x | x | 0 | x | x | x | RFU | | 5 |
| b1 | b2 | b3 | b4 | b5 | b6 | b7 | b8 | Meaning | Change | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | x | x | x | x | x | x | x | Cash | Major | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | 1 | x | x | x | x | x | x | Goods | Major | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | x | 1 | x | x | x | x | x | Services | Major | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | x | x | 1 | x | x | x | x | Cashback | Major | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | x | x | x | 1 | x | x | x | Inquiry | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | x | x | x | x | 1 | x | x | Transfer | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | x | x | x | x | x | 1 | x | Payment | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | x | x | x | x | x | x | 1 | Administrative | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning | Change | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | x | x | x | x | x | x | x | Cash Deposit | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | 0 | x | x | x | x | x | x | RFU | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | x | 0 | x | x | x | x | x | RFU | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | x | x | 0 | x | x | x | x | RFU | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | x | x | x | 0 | x | x | x | RFU | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | x | x | x | x | 0 | x | x | RFU | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | x | x | x | x | x | 0 | x | RFU | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | x | x | x | x | x | x | 0 | RFU | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning | Change | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | x | x | x | x | x | x | x | Numeric keys | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | 1 | x | x | x | x | x | x | Alphabetic and special characters keys | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | x | 1 | x | x | x | x | x | Command keys | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | x | x | 1 | x | x | x | x | Function Keys | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x | x | x | x | 0 | x | x | x | RFU | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| | | | | | | | | | | |
|--------|--|----|----|----|----|----|----|----|---------------------------|--------|
| | x | x | x | x | x | 0 | x | x | RFU | |
| | x | x | x | x | x | x | 0 | x | RFU | |
| | x | x | x | x | x | x | x | 0 | RFU | |
| | Byte 4 | | | | | | | | | |
| | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning | Change |
| | 1 | x | x | x | x | x | x | x | Print, attendant | |
| | x | 1 | x | x | x | x | x | x | Print, cardholder | |
| | x | x | 1 | x | x | x | x | x | Display, attendant | |
| | x | x | x | 1 | x | x | x | x | Display, cardholder | |
| | x | x | x | x | 0 | x | x | x | RFU | |
| | x | x | x | x | x | 0 | x | x | RFU | |
| | x | x | x | x | x | x | 1 | x | Code table 10 | Major |
| | x | x | x | x | x | x | x | 1 | Code table 9 | Major |
| | Byte 5 | | | | | | | | | |
| | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning | Change |
| | 1 | x | x | x | x | x | x | x | Code table 8 | Major |
| | x | 1 | x | x | x | x | x | x | Code table 7 | Major |
| | x | x | 1 | x | x | x | x | x | Code table 6 | Major |
| | x | x | x | 1 | x | x | x | x | Code table 5 | Major |
| | x | x | x | x | 1 | x | x | x | Code table 4 | Major |
| DF11 | x | x | x | x | x | 1 | x | x | Code table 3 | Major |
| | x | x | x | x | x | x | 1 | x | Code table 2 | Major |
| | x | x | x | x | x | x | x | 1 | Code table 1 | Major |
| | Transaction Log Support (Default: Enable) (Major) | | | | | | | | | |
| | 0 → Disable | | | | | | | | | |
| | 1 → Enable | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| DF26 | Revocation List Support (Default: Enable) (Major) | | | | | | | | | |
| | 0 → Disable | | | | | | | | | |
| | 1 → Enable | | | | | | | | | |
| DF27 | Exception File Support (Default: Disable) (Major) | | | | | | | | | |
| | 0 → Disable | | | | | | | | | |
| | 1 → Enable | | | | | | | | | |
| DFEE1E | Contact Host device Configuration (Default: F0 DC 3C F0 C2 9E 94 00) | | | | | | | | | |
| | Byte 1 | | | | | | | | | |
| | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning | Change |
| | 1 | x | x | x | x | x | x | x | Key Pad support | |
| | x | 1 | x | x | x | x | x | x | LCD support | |
| | x | x | 1 | x | x | x | x | x | PIN Pad support | |
| | x | x | x | 1 | x | x | x | x | Print Support | |
| | x | x | x | x | 1 | x | x | x | Confirm Amount is enabled | |
| | x | x | x | x | x | 0 | x | x | RFU | |
| | x | x | x | x | x | x | 0 | x | RFU | |
| | x | x | x | x | x | x | x | 0 | RFU | |

Byte 2

| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning | Change |
|----|----|----|----|----|----|----|----|-------------------------------|--------|
| 1 | x | x | x | x | x | x | x | PSE support | Major |
| x | 1 | x | x | x | x | x | x | Cardholder confirmation | Major |
| x | x | 1 | x | x | x | x | x | Preferred display order | |
| x | x | x | 1 | x | x | x | x | Multi language | |
| x | x | x | x | 1 | x | x | x | EMV language selection method | |
| x | x | x | x | x | 1 | x | x | Default DDOL | Major |
| x | x | x | x | x | x | 0 | x | RFU | |
| x | x | x | x | x | x | x | 0 | RFU | |

Byte 3

| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning | Change |
|----|----|----|----|----|----|----|----|---|--------|
| 0 | x | x | x | x | x | x | x | RFU (Revocation of Issuer Public Key Certificate (DF26)) | Major |
| x | 1 | x | x | x | x | x | x | Manual action when CA PK loading fails | Major |
| x | x | 1 | x | x | x | x | x | CA PK verified with check sum | Major |
| x | x | x | 1 | x | x | x | x | Bypass PIN Entry | Major |
| x | x | x | x | 1 | x | x | x | Subsequent bypass PIN Entry | Major |
| x | x | x | x | x | 1 | x | x | Get data for pin try counter | Major |
| x | x | x | x | x | x | 0 | x | RFU | |
| x | x | x | x | x | x | x | 0 | RFU | |

Byte 4

| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning | Change |
|----|----|----|----|----|----|----|----|---------------------------------|--------|
| 1 | x | x | x | x | x | x | x | Amount before CVM processing | Major |
| x | 1 | x | x | x | x | x | x | Floor limit checking | Major |
| x | x | 1 | x | x | x | x | x | Random transaction selection | Major |
| x | x | x | 1 | x | x | x | x | Velocity checking | Major |
| x | x | x | x | 0 | x | x | x | RFU (Transaction Log (DF11)) | Major |
| x | x | x | x | x | 0 | x | x | RFU (Exception File (DF27)) | Major |
| x | x | x | x | x | x | 0 | x | RFU | |
| x | x | x | x | x | x | x | 0 | RFU | |

Byte 5

| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning | Change |
|----|----|----|----|----|----|----|----|--------------------------|--------|
| 1 | x | x | x | x | x | x | x | Host action code support | Major |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|--|-------|
| x | 1 | x | x | x | x | x | x | Host action code can be change | Major |
| x | x | 1 | x | x | x | x | x | Host action code can be deleted or disable | Major |
| x | x | x | 1 | x | x | x | x | Default Action code processing before 1st GAC | Major |
| x | x | x | x | 1 | x | x | x | Default Action code processing after 1st GAC | Major |
| x | x | x | x | x | 1 | x | x | TAC/IAC default process when unable to go online (Skipped) | Major |
| x | x | x | x | x | x | 1 | x | TAC/IAC default process when unable to go online (Normal) | Major |
| x | x | x | x | x | x | x | 0 | RFU | |

Byte 6

| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning | Change |
|----|----|----|----|----|----|----|----|---------------------------|--------|
| 1 | x | x | x | x | x | x | x | Forced Online support | Major |
| x | 1 | x | x | x | x | x | x | Forced acceptance support | Major |
| x | x | 1 | x | x | x | x | x | Advices support | Major |
| x | x | x | 1 | x | x | x | x | Issuer referrals support | Major |
| X | x | x | x | 1 | x | x | x | Batch data capture | Major |
| x | x | x | x | x | 1 | x | x | Online data capture | Major |
| X | x | x | x | x | x | 1 | x | Default TDOL | Major |
| X | x | x | x | x | x | x | 0 | RFU | |

Byte 7

| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning | Change |
|----|----|----|----|----|----|----|----|---|--------|
| 1 | x | x | x | x | x | x | x | amount and pin entered on the same keypad | |
| x | 1 | x | x | x | x | x | x | ICC/Magstripe reader combined | |
| x | x | 1 | x | x | x | x | x | Magstripe read first | |
| x | x | x | 1 | x | x | x | x | Support account type selection | |
| x | x | x | x | 1 | x | x | x | On fly script processing | |
| x | x | x | x | x | 1 | x | x | Internal date management | |
| x | x | x | x | x | x | 1 | x | Reversal Mode (1)Unable go online (2) ARC Error 0: (3) Online Approved but reader not approved. 1: (3) Online Approved but card response AAC. | |
| x | x | x | x | x | x | x | 0 | RFU | |

| | | | | | | | | | | | |
|--|--------|----|----|----|----|----|----|----|---------|--------|--|
| | Byte 8 | | | | | | | | | | |
| | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Meaning | Change | |
| | x | x | x | x | x | x | x | x | RFU | | |

7.4.9 Contact Retrieve AID List (60-07)

The **Contact Retrieve AID List** command returns a list of all AIDs in the reader. Use this command to verify the reader's configured AIDs.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 60h | 07h | 00h | 00h | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|--------------------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | See status code table | | | See Response Frame Data Format | | |

Response Frame Data Format: <NumberL><NumberH><AID Block 1><AID Block 2> ... <AID Block N>.

Where:

- <NumberL><NumberH> is the number of AID Blocks.
- <AID Block> format is <LenL><LenH><Several bytes AID>

Note: If the AID List does not exist, the response status code is 0x60.

7.4.10 Contact Retrieve CA Public Key (60-08)

The **Contact Retrieve CA Public Key** command retrieves the reader's assigned CA public key.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 60h | 08h | | | Data Objects | | |

Data Objects: <5 bytes RID><1 byte Index>.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|--------------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 60h | See status code table | | | See Response Frame Data Format | | |

Response Frame Data Format: <5 bytes RID><1 byte Index><1 byte Hash Algorithm><1 byte Encryption Algorithm><20 bytes HashValue><4 bytes Public Key Exponent><2 bytes Modulus Length><Variable bytes Modulus>

Where:

- <Hash Algorithm>: The only algorithm supported is SHA-1. The value is set to 0x01.
- <Encryption Algorithm>: The encryption algorithm in which the key is used. Only one type is currently supported: RSA. The value is set to 0x01.
- <HashValue>: Calculated using SHA-1 over the following fields: RID & Index & Modulus & Exponent.
- <Public Key Exponent>: The length of the exponent is either one byte or 3 bytes. It can have two values: 3, or 65537.
- <Modulus Length>: **<LenL><LenH>** Indicated the length of the next field.
- <Modulus>: This is the modulus field of the public key. Its length is specified in the field above.

Note:

- If CA Key RID does not exist, the response status code is 0x60.
- If CA Key Index does not exist, the response status code is 0x60.

7.4.11 Contact Remove CA Public Key (60-09)

The **Contact Remove CA Public Key** command allows the host device to delete a specific key.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 60h | 09h | | | Data Objects | | |

Data Objects: <5 bytes RID><1 byte Index>.

If the length is 00 then remove all CA public keys.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | See status code table | 00h | 00h | | | |

Note:

- If CA Key RID does not exist, the response status code is 0x60.
- If CA Key Index does not exist, the response status code is 0x60.

7.4.12 Contact Set CA Public Key (60-0A)

The **Contact Set CA Public Key** command adds a new key to the reader.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 60h | 0Ah | | | Data Objects | | |

Data Objects: <5 bytes RID><1 byte Index><1 byte Hash Algorithm><1 byte Encryption
Algorithm><20 bytes HashValue><4 bytes Public Key Exponent><2 bytes Modulus
Length><Variable bytes Modulus>

Where:

- <HashAlgorithm>: The only algorithm supported is SHA-1. The value is set to 0x01.
- <Encryption Algorithm>: The encryption algorithm in which this key is used. Only one type is currently supported: RSA. The value is set to 0x01.
- <HashValue>: Calculated using SHA-1 over the following fields: RID & Index & Modulus & Exponent.
- <Public Key Exponent>: The real length of the exponent is either one byte or 3 bytes. It can have two values: 3, or 65537.
- <Modulus Length>: <LenL><LenH> indicate the length of the next field.
- <Modulus>: This is the modulus field of the public key. Its length is specified in the previous field.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | See status code table | 00h | 00h | | | |

Note:

- If key slots are full, the response status code is 0x61.
- If CA key hash data is an error, the response status code is 0x17.

7.4.13 Contact Retrieve CA Public Key List (60-0B)

The **Contact Retrieve CA Public Key List** command retrieves a list of all RID and the CA public key index.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 60h | 0Bh | 00h | 00h | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | See status code table | | | Data Objects | | |

Data Objects: <5Bytes RID1><1 byte RID1 Index><5Bytes RID2><1 byte RID2 Index>.....
<5Bytes RIDN><1 byte RIDN Index>.

Note: If no CA keys exist, the response status code is 0x60.

7.4.14 Contact Retrieve Certification Revocation List (60-0C)

The **Contact Retrieve Certification Revocation List** command retrieves a sequence of consecutive records from the EMV revocation list.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 60h | 0Ch | 00h | 00h | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | See status code table | | | Data Objects | | |

Data Objects: <CRL Length L><CRL Length H><CRL1><CRL2>...<CRLn>.
<CRL>format is <5Bytes RID><1Byte CA public key Index><3Bytes Certificate Serial Number>.

Note: If no CRLs exist, the response status code is 0x60.

7.4.15 Contact Remove Certification Revocation List (60-0D)

The **Contact Remove Certification Revocation List** command deletes a specific entry from the EMV revocation list. Unlike the commands described previously, this command deletes the specific entry that matches the RID, the key index, and the certificate serial number.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 60h | 0Dh | | | Data Objects | | |

Data Objects: <CRL1><CRL2>...<CRLn>.

<CRL>format is <5Bytes RID><1Byte CA public key Index><3Bytes Certificate Serial Number>.

If the Length is 00, the command removes all Certification Revocation Lists.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | See status code table | 00h | 00h | | | |

7.4.16 Contact Set Certification Revocation List (60-0E)

The **Contact Set Certification Revocation List** command adds a new entry to the revocation list. The new entry is added at the list's end.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 60h | 0Eh | | | Data Objects | | |

Data Objects

<CRL Total Length L><CRL Total Length H><CRL1><CRL2>...<CRLn><2 Bytes MAC Length>.
 <CRL>format is <5Bytes RID><1Byte CA public key Index><3Bytes Certificate Serial Number>
 <2Byte Length of CRL>

Optional: <Low byte of length><High byte of length> fixed to 0x00 0x00.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | See status code table | 00h | 00h | | | |

7.4.17 Contact Remove Transaction Amount Log (60-0F)

The **Contact Remove Transaction Amount Log** command deletes the reader's transaction amount log. When an EMV transaction is approved offline or online, the reader saves a transaction amount log.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 60h | 0Fh | 00h | 00h | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | See status code table | 00h | 00h | | | |

7.4.18 Contact Start Transaction (60-10)

The **Contact Start Transaction** command starts a new contact EMV L2 transaction (ICC + MSR) or an MSR-only transaction.

This function will go through below processes at the EMV card inserted at the reader:

1. Card power on
2. Card activation
3. Application selection
4. Initiate application processing
5. Get process options
6. Read records

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 60h | 10h | | | Data Objects | | |

Data Objects: <FallBack><TimeOut1><TimeOut2><App Data>.

Where:

- <FallBack> (1byte). 0x01 indicates FallBack to MSR support, 0x00 indicates FallBack is unsupported.
- <TimeOut1> (2 bytes, unit is Second). Timeout for card is seated.
- <TimeOut2> (2 bytes, unit is Second). Wait time until "Authenticate Transaction" command.
- <App Data> format is <TLV1><TLV2> ... <TLVn>. Refer to Transaction Data.

Transaction Data List (start command parameters)

| Data ID | Tag | Value name | Length (Byte) |
|---------|--------|--|---------------|
| 1 | 9F02 | Amount, Authorized (Numeric) | 6 |
| 2 | 9C | Transaction Type | 1 |
| 3 | 5F2A | Transaction Currency Code (Optional) | 2 |
| 4 | 9A | Transaction Date (Optional) | 3 |
| 5 | 9F21 | Transaction Time (Optional) | 3 |
| 6 | 9F03 | Amount, Other (Numeric) (Optional) | 6 |
| 7 | DFEE1A | Output Tag List (Optional) | N |
| 8 | DFEF1F | Auto Authenticate (Optional) If auto authenticate set to 1, the reader automatically does command 60-11 . Byte 1: Auto Authenticate enable 1: enable, 0 disable Byte 2: Force Online 1: enable, 0 disable | 2 |
| 9 | DFEE27 | MSR Control (Optional) MSR-only transactions must set this tag. This tag is used only in Unipay 1.5. | 1 |

DFEE1A usage rules

- If DFEE1A is in host device data, all three transaction commands respond with the same output tags.
- If DFEE1A is in transaction commands, it should replace the DFEE1A in host device data and the responses have different tags.
- If DFEE1A is in both transaction commands and host device data, it should replace DFEE1A in the host data with the transaction command setting and the response have different tags.
- If DFEE1A is not in host device data and not in transaction commands, the response output tags should be the default output tags.

If the command parse is successful and the ICC transaction starts, the response contains the first command and the status code is 0x63.

If the command parse fails, the response is another status code and end transaction.
The "No terminal data" response is 0x60.

First Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | See status code table | 00h | 00h | | | |

Because some readers don't support LCD or key pad, the external host device often needs to display messages and allow key-in. When a transaction needs to display message or key-in, the reader sends command 61-01 to the host.

If a host device response is needed, the host responds to command 61-01 and sends the result to reader. On **Start Transaction** success, it sends a second response and the result to host.

Second Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-------------|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | 00h | | | Data Objects | | |

Contact EMV L2 Data Objects

| Data Item | Length (bytes) | Description |
|--|----------------|---|
| Attribution | 1 | Bit 4/3/0: Captured Data Type 0 0 0 = Contact Transaction 0 0 1 = Contactless Transaction / EMV 1 0 1 = Contactless Transaction / MSD 0 1 x = MSR Transaction Bit 2/1: Encryption Mode 0 0 = TDES 0 1 = AES 1 x = Reserved Bit 5: Reserved for Attribution Byte Extension. Bit 6/7: Encryption Status 0 0 = MSR/MSD off, EMV off 0 1 = MSR/MSD off, EMV on 1 0 = MSR/MSD on, EMV off 1 1 = MSR/MSD on, EMV on |
| TLV KSN (DFEE12) | Variable | This Tag only appears when the transaction response contains encrypted data. |
| Contact Response Code (DFEE25) | Variable | See Contact Response Code. |
| TLV Data | Variable | Output default Start Transaction Output TLV data list. If tag DFEE1A is set, the TLV data will listed. |
| TLV (DF30) (ViVOpay proprietary) | Variable | Track Data Source. This tag is embedded in the ViVOpay Group tag. It specifies whether the track data came from a swipe or contactless transaction. 0Ch for swiped MagStripe 00h for a contactless transaction. 01h for a contact transaction. This tag is included in the ViVOpay proprietary information (FFEE01). Ex: FFEE0104DF30010C |
| TLV Encrypt Information (DFEE26) | Variable | Encryption and Data Capture Information. Length: 1 or 2 (variable) <div style="border: 2px solid red; background-color: yellow; padding: 2px;">NEO 2 the length must be 2 bytes</div> Values: |

| Data Item | Length (bytes) | Description |
|-----------|----------------|--|
| | | <p>Byte 1: Same as "Attribution" byte definition. Namely:</p> <p>Bits 4/3/0: Captured Data Type</p> <p>0 0 0 = Contact Transaction</p> <p>0 0 1 = Contactless Transaction / EMV</p> <p>1 0 1 = Contactless Transaction / MSD</p> <p>0 1 x = MSR Card</p> <p>Bits 2/1: Encryption Mode</p> <p>0 0 = TDES</p> <p>0 1 = AES</p> <p>1 x = Refer to "Extended Encryption Mode" further below.</p> <p>Bit 5: Attribution Byte Extension.</p> <p>0 = Length of tag is 1 byte</p> <p>1 = Length of tag is 2 bytes</p> <p>Bits 6/7: Encryption Status</p> <p>0 0 = EMV off, MSR/MSD off</p> <p>0 1 = EMV off, MSR/MSD on</p> <p>1 0 = EMV on, MSR/MSD off</p> <p>1 1 = EMV on, MSR/MSD on</p> <p>Byte 2: (When applicable; that is, when Length of TLV is 2)</p> <p>Bits 3/2/1/0: Extended Encryption Mode</p> <p>0 0 0 0 = TDES</p> <p>0 0 0 1 = AES</p> <p>0 0 1 0 = TransArmor Algorithm</p> <p>0 0 1 1 = Voltage Algorithm</p> <p>0 1 0 0 = Visa FPE</p> <p>0 1 0 1 = Verifone FPE</p> <p>0 1 1 0 = TransArmor TDES-DUKPT</p> <p>Bits 4 to 6: Reserved</p> <p>Bit 7:</p> <p>0 = No MAC Verification Data</p> <p>1 = Has MAC Verification Data</p> |

MSR Data Objects

| Data Item | Length (bytes) | Description |
|-------------|----------------|---|
| Attribution | 1 | <p>Bit 4/3/0: Captured Data Type</p> <p>0 0 0 = Contact Transaction</p> <p>0 0 1 = Contactless Transaction / EMV</p> <p>1 0 1 = Contactless Transaction / MSD</p> <p>0 1 x = MSR Card</p> <p>Bit 2/1: Encryption Mode</p> <p>0 0 = TDES</p> |

| Data Item | Length (bytes) | Description |
|----------------------------------|----------------|---|
| | | 0 1 = AES 1 x = Reserved Bit 5: Reserved for Attribution Byte Extension. Bit 6/7: Encryption Status 0 0 = MSR/MSD off, EMV off 0 1 = MSR/MSD off, EMV on 1 0 = MSR/MSD on, EMV off 1 1 = MSR/MSD on, EMV on |
| Contact Response Code (DFEE25) | Variable | See Contact Response Code. (00h 11h). |
| TLV Data | Variable | TLV data list |
| MSR Tag (DFEE23) | Variable | Data follow Enhanced Encrypted MSR FIELD DATA. Refer to 80000502-001 ID TECH Encrypted Data Output Formats.pdf |
| TLV POS Entry Mode (9F39) | Variable | 90h = Magnetic Stripe Reader Swipe 91h = Contactless MSD 05h = Contact EMV 07h = Contactless EMV 80h = Contact Fallback to Magnetic Stripe |
| TLV (DF30) (ViVOpay proprietary) | Variable | Track Data Source. This tag is embedded in the ViVOpay Group tag. It specifies whether the track data came from a swipe or RFID transaction. 0Ch for swiped MagStripe 00h for a contactless transaction. 01h for a contact transaction. This tag include in ViVOpay proprietary (FFEE01) Ex: FFEE0104DF30010C |
| TLV Encrypt Information (DFEE26) | Variable | Encryption and Data Capture Information. Length: 1 or 2 (variable) Values: Byte 1: Same as "Attribution" byte definition. Namely: Bits 4/3/0: Captured Data Type 0 0 0 = Contact Transaction 0 0 1 = Contactless Transaction / EMV 1 0 1 = Contactless Transaction / MSD 0 1 x = MSR Card Bits 2/1: Encryption Mode 0 0 = TDES 0 1 = AES 1 x = Refer to "Extended Encryption Mode" further below. |

| Data Item | Length (bytes) | Description |
|----------------------------------|----------------|---|
| | | <p>Bit 5: Attribution Byte Extension. 0 = Length of tag is 1 byte 1 = Length of tag is 2 bytes Bits 6/7: Encryption Status 0 0 = EMV off, MSR/MSD off 0 1 = EMV off , MSR/MSD on 1 0 = EMV on, MSR/MSD off 1 1 = EMV on, MSR/MSD on</p> <p>Byte 2: (When applicable; that is, when Length of TLV is 2)</p> <p>Bits 3/2/1/0: Extended Encryption Mode 0 0 0 0 = TDES 0 0 0 1 = AES 0 0 1 0 = TransArmor Algorithm 0 0 1 1 = Voltage Algorithm 0 1 0 0 = Visa FPE 0 1 0 1 = Verifone FPE 0 1 1 0 = TransArmor TDES-DUKPT</p> <p>Bits 4 to 6: Reserved</p> <p>Bit 7: 0 = No MAC Verification Data 1 = Has MAC Verification Data</p> |
| TLV Encrypt Information (DFEE26) | Variable | <p>Encryption and Data Capture Information. Length: 1 or 2 (variable)</p> <p>Values:</p> <p>Byte 1: Same as "Attribution" byte definition. Namely:</p> <p>Bits 4/3/0: Captured Data Type 0 0 0 = Contact Transaction 0 0 1 = Contactless Transaction / EMV 1 0 1 = Contactless Transaction / MSD 0 1 x = MSR Card</p> <p>Bits 2/1: Encryption Mode 0 0 = TDES 0 1 = AES 1 x = Refer to "Extended Encryption Mode" further below.</p> <p>Bit 5: Attribution Byte Extension. 0 = Length of tag is 1 byte 1 = Length of tag is 2 bytes Bits 6/7: Encryption Status 0 0 = EMV off, MSR/MSD off</p> |

| Data Item | Length (bytes) | Description |
|-----------|----------------|---|
| | | <p>0 1 = EMV off , MSR/MSD on 1 0 = EMV on, MSR/MSD off 1 1 = EMV on, MSR/MSD on</p> <p>Byte 2: (When applicable; that is, when Length of TLV is 2)</p> <p>Bits 3/2/1/0: Extended Encryption Mode 0 0 0 0 = TDES 0 0 0 1 = AES 0 0 1 0 = TransArmor Algorithm 0 0 1 1 = Voltage Algorithm 0 1 0 0 = Visa FPE 0 1 0 1 = Verifone FPE 0 1 1 0 = TransArmor TDES-DUKPT</p> <p>Bits 4 to 6: Reserved</p> <p>Bit 7: 0 = No MAC Verification Data 1 = Has MAC Verification Data</p> |

1. For MSR <Response Code> is 00 11.
2. MSR data in tag DFEE23, format follows "Enhanced Encrypted MSR Data Output Format."
Refer to [80000502-001 ID TECH Encrypted Data Output Formats.pdf](#)

Default Start Transaction Output TLV Data List

| Data ID | Tag | Value name | Length (Byte) |
|---------|--------|--------------------------------------|---------------|
| 1 | 57 | Track2 Equivalent Data | <=19 |
| 2 | 5A | PAN | <=10 |
| 3 | 5F34 | PAN Sequence Number | 1 |
| 4 | 5F20 | Cardholder Name | 2~26 |
| 5 | 5F24 | Application Expire Date | 3 |
| 6 | 9F20 | Track2 Discretionary Data | Var. |
| 7 | 5F25 | Application Effective Date | 3 |
| 8 | 5F2D | Language Preference | 2~8 |
| 9 | 50 | Application Label | 1~16 |
| 10 | 84/4F | DF Name or ADF Name | 5~16 |
| 11 | DFEE23 | MSR Data | N |
| 12 | 9F39 | POS Entry Mode | 1 |
| 13 | 9F53 | Transaction Category Code | 1 |
| 14 | 9F1E | Interface Device (IFD) Serial Number | 8 |

7.4.19 Contact Authenticate Transaction (60-11)

The **Contact Authenticate Transaction** command authenticates a contact transaction. Call this function after the **Start Transaction** command. The function continues the EMV transaction flow:

- (1) Offline data auth
- (2) Processing restrictions
- (3) Cardholder Verification
- (4) Terminal risk management
- (5) TAA/CAA/1st Gen AC

If the response code is 00 04, the reader needs to go online to process the transaction and must send an **Apply Host Response** command.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 60h | 11h | | | Data Objects | | |

Data Objects: <GoOnline><TimeOut><Output Data List>-.

- <GoOnline>(1byte). 0x01 means force the transaction to go online, 0x00 means do not force online (allow normal processing to occur via Terminal action analysis).
- <TimeOut> (2 byte, unit is seconds) is the waiting time for host response when online.
- <Output Tag List DFEE1A> format is <TLV> (V is output tag list.)

Example: DFEE1A 06 95 5A 84 9F39 50

DFEE1A usage rules

- If DFEE1A is in host data, all three transaction commands respond with the same output tags.
- If DFEE1A is in transaction commands, it should replace the DFEE1A in host data and the responses have different tags.
- If DFEE1A is in both transaction commands and host data, it should replace DFEE1A in the host data with the transaction command setting and the response have different tags.
- If DFEE1A is not in host data and not in transaction commands, the response output tags should be the default output tags.

If command parse is successful and ICC transactions continue, response is the first command and the status code is 0x63.

If the command parse fails, the response is some other status code and end of the transaction.

First Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | See status code table | 00h | 00h | | | |

When transaction needs to display a message or Get PIN, the reader sends commands 61-01 and 61-02 to the host device. If a host response is needed, the host device responds with command 61-01 and 61-02 and sends the results to the reader.

Authenticate Transaction success sends a second response and result to the host device.

Second Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-------------|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | 00h | | | Data Objects | | |

Data Objects

| Data Item | Length (bytes) | Description |
|-------------|-------------------|---|
| Attribution | 1 | Bit 4/3/0: Captured Data Type 0 0 0 = Contact Card 0 0 1 = Contactless Card / EMV 1 0 1 = Contactless Card / MSD 0 1 x = MSR Card Bit 2/1: Encryption Mode 0 0 = TDES 0 1 = AES 1 x = Reserved Bit 5: Reserved for Attribution Byte Extension. Bit 6/7: Encryption Status 0 0 = MSR/MSD off, EMV off 0 1 = MSR/MSD off, EMV on 1 0 = MSR/MSD on, EMV off 1 1 = MSR/MSD on, EMV on |
| TLV Encrypt | Variable | Encryption and Data Capture Information. |

| Data Item | Length (bytes) | Description |
|--------------------------------|----------------|--|
| Information (DFEE26) | | <p>Length: 1 or 2 (variable)</p> <p>Values:</p> <p>Byte 1: Same as "Attribution" byte definition. Namely:</p> <p>Bits 4/3/0: Captured Data Type 0 0 0 = Contact Transaction 0 0 1 = Contactless Transaction / EMV 1 0 1 = Contactless Transaction / MSD 0 1 x = MSR Card</p> <p>Bits 2/1: Encryption Mode 0 0 = TDES 0 1 = AES 1 x = Refer to "Extended Encryption Mode" further below.</p> <p>Bit 5: Attribution Byte Extension. 0 = Length of tag is 1 byte 1 = Length of tag is 2 bytes</p> <p>Bits 6/7: Encryption Status 0 0 = EMV off, MSR/MSD off 0 1 = EMV off, MSR/MSD on 1 0 = EMV on, MSR/MSD off 1 1 = EMV on, MSR/MSD on</p> <p>Byte 2: (When applicable; that is, when Length of TLV is 2)</p> <p>Bits 3/2/1/0: Extended Encryption Mode 0 0 0 0 = TDES 0 0 0 1 = AES 0 0 1 0 = TransArmor Algorithm 0 0 1 1 = Voltage Algorithm 0 1 0 0 = Visa FPE 0 1 0 1 = Verifone FPE 0 1 1 0 = TransArmor TDES-DUKPT</p> <p>Bits 4 to 6: Reserved</p> <p>Bit 7: 0 = No MAC Verification Data 1 = Has MAC Verification Data</p> |
| TLV KSN (DFEE12) | Variable | This Tag only appears when the transaction response contains encrypted data. |
| Contact Response Code (DFEE25) | Variable | See Contact Response Code. If response code is 00 04, then need go online process and send Apply Host Response command. |

| Data Item | Length (bytes) | Description |
|-------------------------------------|----------------|---|
| TLV Data | Variable | Output default Authenticate Transaction Output TLV data list. If set tag DFEE1A, The TLV data will be follow list. |
| TLV (DF30) (ViVOpay proprietary) | Variable | <p>Track Data Source.</p> <p>This tag is embedded in the ViVOpay Group tag. It specifies whether the track data came from a swipe or contactless transaction.</p> <p>0Ch for swiped MagStripe.</p> <p>00h for a contactless transaction.</p> <p>01h for a contact transaction.</p> <p>This tag is included in ViVOpay proprietary (FFEE01) Ex: FFEE0104DF30010C</p> |

Default Authenticate Transaction Output TLV Data List

| Tag | Value name | Length (Byte) |
|------------------|---|---------------|
| 4F | ADF Name | 5 to 16 |
| 50 | Application Label | 1 to 16 |
| 57 | Track 2 Equivalent Data | Var. up to 19 |
| 5A | Application PAN | Var. up to 10 |
| 82 | Application Interchange Profile | 2 |
| 84 | DF Name | 5 to 16 |
| 8E | CVM List | 10 to 252 |
| 9A | Transaction Date | 3 |
| 95 | Host Verification Results | 5 |
| 99 | Transaction Personal Identification Number (PIN) Data. NOTE: This tag appears only when the CVM is "online PIN." | Var. |
| 9B | Transaction Status Information | 2 |
| 9C | Transaction Type | 1 |
| 5F20 | Cardholder Name | 2 to 26 |
| 5F24 | Application Expiration Date | 3 |
| 5F25 | Application Effective Date | 3 |
| 5F34 | Application PAN Sequence Number | 1 |
| 5F28 | Issuer Country Code | 2 |
| 5F2A | Transaction Currency Code | 2 |
| 5F2D | Language Preference | 2 to 8 |
| 9F02 | Amount, Authorized (Numeric) | 6 |
| 9F03 | Amount, Other (Numeric) | 6 |
| 9F07 | Application Usage Control | 2 |
| 9F08, 9F09 | Application Version Number (Kernel) | 2 |
| 9F0D, 9F0E, 9F0F | Issuer Action Codes | 5 |
| 9F0B | Cardholder Name (Extended) | 27 to 45 |
| 9F10 | Issuer Application Data | Var. up to 32 |
| 9F11 | Issuer Code Table Index | 1 |
| 9F12 | Application Preferred Name | 1 to 16 |
| 9F13 | Last Online Application Transaction Counter(ATC) Register | 2 |
| 9F15 | Merchant Category Code | 2 |
| 9F16 | Merchant Identifier | 16 |
| 9F1A | Host Country Code | 2 |

| | | |
|--------|---|----------|
| 9F1C | Host Identifier | 8 |
| 9F1E | Interface Device Serial Number | 8 |
| 9F20 | Track2 Discretionary Data | Var. |
| 9F21 | Transaction Time | 3 |
| 9F24 | Payment Account Reference | 24 |
| 9F26 | Application Cryptogram | 8 |
| 9F27 | Cryptogram Information Data | 1 |
| 9F33 | Host Capabilities | 3 |
| 9F34 | Cardholder Verification Method(CVM) Results | 3 |
| 9F35 | Host Type | 1 |
| 9F36 | Application Transaction Counter(ATC) | 2 |
| 9F37 | Unpredictable Number | 4 |
| 9F39 | POS Entry Mode | 1 |
| 9F53 | Transaction Category Code | 1 |
| 9F5B | Issuer script results | Var. |
| DF21 | Issuer script results | 1 |
| DFEE23 | MSR Data | Var. |
| DFEE51 | KSN of online PIN DUKPT | 10 or 12 |

7.4.20 Contact Apply Host Response (60-12) – Complete Transaction

This function is the “Completion” step in the EMV transaction flow. The command sends the acquirer data (if online) to the card and notifies the reader that the transaction is complete. Depending on the transaction type, this command performs the following process; the reader may or may not perform each step depending upon the acquirer's requirement and response:

- (1) External Authenticate
- (2) Script Processing
- (3) 2nd Gen AC
- (4) Completion

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTEch2\0 | 60h | 12h | | | Data Objects | | |

Data Objects: <1Byte ComFlag> [<Authorization Response Code (TLV, Tag 8A)>< Issuer Authentication Data (TLV, Tag 91)>< <Scripts (TLV, Tag 71/72)>] <Output Data List>

Where:

- <1Byte ComFlag>: 0x01 indicates the reader went online with the host device, 0x00 indicates unable to go online.
- Data in [] indicates that data is optional:
 - If ComFlag is 0x01, the data exists.
 - If ComFlag is 0x00, the data does not exist.

<Output Tag List DFEE1A> format is <TLV> (V is output tag list.)

Ex: DFEE1A 06 95 5A 84 9F39 50

If the command parse is successful, and ICC transaction continues, response is first command and status code is 0x63.

If the command parse fails, response will contain other status code and end transaction.

First Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | See status code table | 00h | 00h | | | |

When transaction needs LCD display or manual key in, the reader sends command 61-01 to the host device. If the reader needs a host response, the host device should respond with command 61-01 and the result to the reader. See command 61-01 for details.

Second Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-------------|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | 00h | | | Data Objects | | |

Data Objects

| Data Item | Length (bytes) | Description |
|----------------------------------|----------------|---|
| Attribution | 1 | Bit 4/3/0: Captured Data Type 0 0 0 = Contact Card 0 0 1 = Contactless Card / EMV 1 0 1 = Contactless Card / MSD 0 1 x = MSR Card Bit 2/1: Encryption Mode 0 0 = TDES 0 1 = AES 1 x = Reserved Bit 5: Reserved for Attribution Byte Extension. Bit 6/7: Encryption Status 0 0 = MSR/MSD off, EMV off 0 1 = MSR/MSD off, EMV on 1 0 = MSR/MSD on, EMV off 1 1 = MSR/MSD on, EMV on |
| TLV Encrypt Information (DFEE26) | Variable | Encryption and Data Capture Information. Length: 1 or 2 (variable) Values: Byte 1: Same as "Attribution" byte definition. Namely: Bits 4/3/0: Captured Data Type 0 0 0 = Contact Transaction 0 0 1 = Contactless Transaction / EMV 1 0 1 = Contactless Transaction / MSD 0 1 x = MSR Card Bits 2/1: Encryption Mode 0 0 = TDES 0 1 = AES 1 x = Refer to "Extended Encryption Mode" further below. Bit 5: Attribution Byte Extension. 0 = Length of tag is 1 byte 1 = Length of tag is 2 bytes Bits 6/7: Encryption Status 0 0 = EMV off, MSR/MSD off 0 1 = EMV off, MSR/MSD on 1 0 = EMV on, MSR/MSD off 1 1 = EMV on, MSR/MSD on Byte 2: (When applicable; that is, when Length of TLV is 2) Bits 3/2/1/0: Extended Encryption Mode 0 0 0 0 = TDES |

| Data Item | Length (bytes) | Description |
|----------------------------------|----------------|--|
| | | 0 0 0 1 = AES 0 0 1 0 = TransArmor Algorithm 0 0 1 1 = Voltage Algorithm 0 1 0 0 = Visa FPE 0 1 0 1 = Verifone FPE 0 1 1 0 = TransArmor TDES-DUKPT Bits 4 to 6: Reserved Bit 7: 0 = No MAC Verification Data 1 = Has MAC Verification Data |
| TLV KSN (DFEE12) | Variable | This Tag only appears when the transaction response contains encrypted data. |
| Contact Response Code (DFEE25) | Variable | See Contact Response Code. |
| TLV Data | Variable | Output default Apply Host Response Output TLV data list. If set tag DFEE1A, The TLV data will be follow list. |
| TLV (DF30) (ViVOpay proprietary) | Variable | Track Data Source. This tag is embedded in the ViVOpay Group tag. It specifies whether the track data came from a swipe or contactless transaction. 0Ch for swiped MagStripe 00h for a contactless transaction. 01h for a contact transaction. This tag is included in ViVOpay proprietary (FFEE01) Ex: FFEE0104DF30010C |

Default Apply Host Response Output TLV Data List

| Data ID | Tag | Value name | Length (Byte) |
|---------|--------|---|---------------|
| 1 | 8A | ARC | 2 |
| 2 | 9F10 | Issuer Application Data | Var. up to 32 |
| 3 | 9F26 | Application Cryptogram | 8 |
| 4 | 9F27 | Cryptogram Information Data | 1 |
| 5 | 9F36 | Application Transaction Counter(ATC) | 2 |
| 6 | 9F37 | Unpredictable Number | 4 |
| 7 | 9F02 | Amount, Authorized (Numeric) | 6 |
| 8 | 9F13 | Last Online Application Transaction Counter(ATC) Register | 2 |
| 9 | 95 | Host Verification Results | 5 |
| 10 | 9B | Transaction Status Information | 2 |
| 11 | 9F03 | Amount, Other (Numeric) | 6 |
| 12 | 9F34 | Cardholder Verification Method(CVM) Results | 3 |
| 13 | DFEE51 | PIN KSN | 10 or 12 |
| 14 | 99 | PIN Block | Var. |
| 15 | DF21 | Issuer script results | 5 |

| | | | |
|----|--------|---------------------------|--------|
| 16 | 9F39 | POS Entry Mode | 1 |
| 17 | 9F53 | Transaction Category Code | 1 |
| 18 | DFEE26 | Encrypt Information | 1 or 2 |

7.4.21 Contact Retrieve Transaction Result (60-13)

Use the **Contact Retrieve Transaction Result** command after a contact EMV transaction completes to get other TLVs.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 60h | 13h | | | Data Objects | | |

Data Objects: <Tags>.

<Tags> these tags return TLV format. Supported tags refer to the Reference Data List table below.

Tags Example:

9F029F36959F37 means total 4 tags (9F02, 9F36, 95, 9F37) requested to in response. Length is 7 bytes.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---------------------------------------|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | See status code table | | | Data Objects | | |

Data Objects

| Data Item | Length (bytes) | Description |
|--|-------------------|--|
| TLV Encrypt Information (DFEE26) | Variable | Encryption and Data Capture Information. Length: 1 or 2 (variable) Values: Byte 1: Same as "Attribution" byte definition. Namely: |

| Data Item | Length (bytes) | Description |
|------------------|----------------|---|
| | | <p>Bits 4/3/0: Captured Data Type</p> <p>0 0 0 = Contact Transaction</p> <p>0 0 1 = Contactless Transaction / EMV</p> <p>1 0 1 = Contactless Transaction / MSD</p> <p>0 1 x = MSR Card</p> <p>Bits 2/1: Encryption Mode</p> <p>0 0 = TDES</p> <p>0 1 = AES</p> <p>1 x = Refer to "Extended Encryption Mode" further below.</p> <p>Bit 5: Attribution Byte Extension.</p> <p>0 = Length of tag is 1 byte</p> <p>1 = Length of tag is 2 bytes</p> <p>Bits 6/7: Encryption Status</p> <p>0 0 = EMV off, MSR/MSD off</p> <p>0 1 = EMV off, MSR/MSD on</p> <p>1 0 = EMV on, MSR/MSD off</p> <p>1 1 = EMV on, MSR/MSD on</p> <p>Byte 2: (When applicable; that is, when Length of TLV is 2)</p> <p>Bits 3/2/1/0: Extended Encryption Mode</p> <p>0 0 0 0 = TDES</p> <p>0 0 0 1 = AES</p> <p>0 0 1 0 = TransArmor Algorithm</p> <p>0 0 1 1 = Voltage Algorithm</p> <p>0 1 0 0 = Visa FPE</p> <p>0 1 0 1 = Verifone FPE</p> <p>0 1 1 0 = TransArmor TDES-DUKPT</p> <p>Bits 4 to 6: Reserved</p> <p>Bit 7:</p> <p>0 = No MAC Verification Data</p> <p>1 = Has MAC Verification Data</p> |
| TLV KSN (DFEE12) | Variable | This Tag only appears when the transaction response contains encrypted data. |
| TLV Data | Variable | Output TLV data list. |

Reference Data List

| Tag | Tag | Tag | Tag | Tag | Tag | Tag |
|------|-----|------|------|------|------|-----|
| 5F2A | 98 | 9F02 | 9F16 | 9F21 | 9F39 | 4F |
| 5F36 | | 9F03 | 9F1A | 9F22 | 9F3A | 50 |
| 81 | 9A | 9F04 | 9F1B | 9F33 | 9F3C | 57 |
| 8A | 9B | 9F06 | 9F1C | 9F34 | 9F3D | 58 |
| 95 | 9C | 9F09 | 9F1D | 9F35 | 9F40 | 5A |

| | | | | | | |
|--------|--------|--------|--------|--------|--------|--------|
| 5F57 | 9F01 | 9F15 | 9F1E | 9F37 | 9F41 | 5F20 |
| 5F24 | 5F25 | 5F28 | 5F2D | 5F30 | 5F34 | 5F36 |
| 6F | 71 | 72 | 73 | 82 | 84 | 87 |
| 88 | 8C | 8D | 8E | 8F | 90 | 91 |
| 92 | 93 | 94 | 97 | 9F05 | 9F07 | 9F08 |
| 9F0D | 9F0E | 9F0F | 9F10 | 9F11 | 9F12 | 9F13 |
| 9F14 | 9F17 | 9F1F | 9F20 | 9F23 | 9F26 | 9F27 |
| 9F2D | 9F2E | 9F2F | 9F32 | 9F36 | 9F38 | 9F3B |
| 9F42 | 9F43 | 9F44 | 9F45 | 9F46 | 9F47 | 9F48 |
| 9F49 | 9F4A | 9F4B | 9F4C | 9F4D | BF0C | DF62 |
| 9F4E | 9F5B | DFEE15 | DFEE16 | DFEE17 | DFEE18 | DFEE19 |
| DFEE1A | DFEE1B | DF11 | DF14 | DF15 | DF17 | DF18 |
| DF19 | DF20 | DF21 | DF22 | DF25 | DF26 | DF27 |
| DF28 | DF10 | DF40 | DF41 | DF42 | DF43 | DFEE1E |
| DFEE1F | DFEE20 | DFEE21 | DFEE22 | DFEE23 | DFEE24 | 89 |
| DF13 | | | | | | |

7.4.22 Contact Get Reader Status (60-14)

The **Contact Get Reader Status** command retrieves the ICC card power and card seat status.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 60h | 14h | | | Command Data | | |

Command Data

| Data Item | Length (bytes) | Description/Example |
|-----------|----------------|---------------------|
| Interface | 1 | 20h = ICC |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | See status code table | | | Data Objects | | |

Data Objects (1 byte)

| Bit Position | Meaning if '0' | Meaning if '1' |
|--------------|---------------------------|-----------------------|
| 0 | ICC Power not ready | ICC Powered |
| 1 | Card not seated | Card seated |
| 2 | Front switch not detected | Front switch detected |
| 3~7 | RFU | RFU |

7.4.23 Contact Get ICS Identification (60-15)

The **Contact Get ICS Identification** command retrieves the reader's ICS host configuration number. Contact Common EMV L2 approved configurations of certification are 1C, 2C, 3C, and 4C.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 60h | 15h | 00h | 00h | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | See status code table | | | Response Data | | |

Response Data:

0x01 → Identification 1C

0x02 → Identification 2C

0x03 → Identification 3C

0x04 → Identification 4C

...

...

0x18 → Identification 24C

7.4.24 Contact Set ICS Identification (60-16)

The **Contact Set ICS Identification** command sets the reader's ICS host configuration number (Contact Command EMV L2 includes 4 approved configurations of certification: 1C, 2C, 3C, or 4C).

This command affects the **Contact Set Terminal Data (60-06)** command. Generally, readers call 60-16 first, then 60-06.

Examples:

Set 3C configuration: 5669564f746563683200601600010392ed

Reader responds with: 5669564f746563683200600000003d35 (Success)

Now set host data (TLVs) with config values appropriate to 3C:

5669564f746563683200600600c818005f3601029f1a0208409f3501259f330360d8c89f40056000f050019f1e085465726d696e616c9f150212349f160f3030303030303030303030309f1c0838373635343332319f4e2231303732312057616c6b65722053742e20437970726573732c204341202c5553412edf260101df1008656e667265737a68df110101df270100dfee150101dfee160100dfee170107dfee180180dfee1e08f0dc24f0c20e1400dfee1f0180dfee1b083030303135313030dfee20013cdf21010adfee2203323c3caa88

Reader responds with: 5669564f746563683200600000003d35 (Success)

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 60h | 16h | 00h | 01h | ICS Identification | | |

ICS Identification:

0x01 → Identification 1C

0x02 → Identification 2C (Default)

0x03 → Identification 3C

0x04 → Identification 4C

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 60h | See status code table | 00 | 00 | | | |

7.4.25 Contact LCD Display Control (61-01) (Reader sends to Host)

The **Contact LCD Display Control** command requests an LCD display or key-in by the host device. Because some readers do not incorporate an LCD screen or a keypad, if said reader requests a display or key-in from the host, the host device responds with this command and a result. This command is sent automatically in 60-10, 60-11, and 60-12 commands.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 01h | | | Data Objects | | |

Data Objects

| Data Item | Length (bytes) | Description | | | | | | | | | | | | | | | | | | | | | |
|----------------|-------------------|--|-----------|----------------|-------------|-------|---|--|---------------|---|---------------|---------------|---|---------------|-----------|----------------|-------------|--------|---|-------------|-----------|---|-----------------------|
| Display Mode | 1 | <p> 0x01: Menu Display 0x02: Normal Display get function key 0x03: Display without key input (Do not receive input data) 0x08: Language Menu Display 0x10: Clear Screen (Do not receive input data) 0x20: Notify Touch Panel Information (Do not receive input data) For the remaining Data Items, see below: </p> <table border="1"> <thead> <tr> <th>Data Item</th><th>Length (bytes)</th><th>Description</th></tr> </thead> <tbody> <tr> <td>Event</td><td>1</td><td> 0x00: no event 0x01: touch down 0x02: touch move 0x03: touch up </td></tr> <tr> <td>x- coordinate</td><td>2</td><td>Little endian</td></tr> <tr> <td>y- coordinate</td><td>2</td><td>Little endian</td></tr> </tbody> </table> <p> 0x21: Update LED State (Do not receive input data) For the remaining Data Items, see below: </p> <table border="1"> <thead> <tr> <th>Data Item</th><th>Length (bytes)</th><th>Description</th></tr> </thead> <tbody> <tr> <td>LED ID</td><td>1</td><td>ID of RFLED</td></tr> <tr> <td>LED state</td><td>1</td><td> 0x00: Off 0x01: On </td></tr> </tbody> </table> <p> 0x22: Show Key Pad (Do not receive input data) (If Clear Screen, don't need rod end below filed) </p> | Data Item | Length (bytes) | Description | Event | 1 | 0x00: no event 0x01: touch down 0x02: touch move 0x03: touch up | x- coordinate | 2 | Little endian | y- coordinate | 2 | Little endian | Data Item | Length (bytes) | Description | LED ID | 1 | ID of RFLED | LED state | 1 | 0x00: Off 0x01: On |
| Data Item | Length (bytes) | Description | | | | | | | | | | | | | | | | | | | | | |
| Event | 1 | 0x00: no event 0x01: touch down 0x02: touch move 0x03: touch up | | | | | | | | | | | | | | | | | | | | | |
| x- coordinate | 2 | Little endian | | | | | | | | | | | | | | | | | | | | | |
| y- coordinate | 2 | Little endian | | | | | | | | | | | | | | | | | | | | | |
| Data Item | Length (bytes) | Description | | | | | | | | | | | | | | | | | | | | | |
| LED ID | 1 | ID of RFLED | | | | | | | | | | | | | | | | | | | | | |
| LED state | 1 | 0x00: Off 0x01: On | | | | | | | | | | | | | | | | | | | | | |
| Timeout Length | 2 | <p>Format: <Length L><Length H> (Little-endian)</p> <p>If Display without key input. The value is 00 00.</p> | | | | | | | | | | | | | | | | | | | | | |
| Timeout Value | Variable | Format: <Value L><Value H> (Little-endian) | | | | | | | | | | | | | | | | | | | | | |

| Data Item | Length (bytes) | Description |
|---------------------------------|----------------|--|
| | | If Normal Display or Menu Display, Total timeout for keypad entry, in second. Default is 30 seconds. Note: Total timeout will cancel keypad entry and return error. |
| Display Message Language Length | 2 | Format: <Length L><Length H> (Little-endian) |
| Display Message Language | Variable | Display Message Language, 2 byte "EN" – English (default) "ES" – Spanish "ZH" – Chinese "FR" – French ... |
| Display Message Control Length | 2 | Format: <Length L><Length H> (Little-endian) |
| Display Message Control | Variable | Repeatable combination of <Line><Message><0x1C> <Line> - Display line number (1-First Line, n-nth Line), Maximum 16 lines. The lower 7 bits is for line number. <ul style="list-style-type: none"> The most significant bit is to indicate following message is a Message String or Message ID. <ul style="list-style-type: none"> MSB – 0: Message String. MSB – 1: Message ID. <Message> - Message String or Message ID. Message String: <ul style="list-style-type: none"> Maximum 16 characters, support regular and extended ASCII chart table (0x00-0xFF) "Menu Display": character in the range of 0x20 – 0x7f, Maximum 16 characters "Language Menu Display": 2 bytes Language ID "EN" – English (default) "ES" – Spanish "ZH" – Chinese "FR" – French "Show Key Pad": character in the range of 0x30 – 0x39, total 10 characters, the sequence of key pad display sequence. Message ID: 1 byte, check LCD Foreign Language Mapping Table . <0x1C> - separator |
| Back Light Timer Length | 2 | Format: <Length L><Length H> (Little-endian) |
| Back Light Timer Value | Variable | Format: <Value L><Value H> (Little-endian) Back Light On Timer Value in second. (all 0-Back Light Off, all 0xff-Back Light always On) |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-------------|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 01h | | | Data Objects | | |

Data Objects

| Data Item | Length (bytes) | Description |
|---------------------|-------------------|--|
| Display Mode | 1 | 0x00 - Cancel (user presses cancel key on the keypad for mode 1) 0x01 - Menu Display 0x02 - Normal Display get function key 0x08 - Language Menu Display If Mode byte is "Cancel" or "Display without key input", do not send the field below. |
| Display Menu Length | 2 | Format: <Length L><Length H> (Little-endian) (If Normal Display, Length of Key (Get Function)) |
| Display Menu | Variable | Menu value, sequence number of selected line, hex format (If Normal Display, ASCII format ('E' is Enter, 'C' is Cancel)) |

7.4.26 Contact Get PIN Control (61-02) (Reader or SCRP send to Host)

The **Contact Get PIN Control** command requests Get PIN on the host device or PIN pad device. Because some readers do not incorporate an LCD screen or a keypad, if said reader requests a Get PIN from the host device, the host responds with this command and a result. If the reader or SCRP is requesting a PIN, the host device or PIN pad device or phone responds to the reader or SCRP with a 61-02 command with the PIN block.

This command is used in 60-11 commands only on SRED readers. When included in 60-11, the reader sends the 61-02 command to the host device or PIN pad device automatically.

This command used in 60-11 or 62-01 commands in SCRP. When in these commands, the reader automatically sends command 61-02 back to the phone.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 02h | | | Data Objects | | |

Data Objects

| Data Item | Length (bytes) | Description |
|---|----------------|---|
| Mode | 1 | <ul style="list-style-type: none"> - 0x00 – Cancel (cancel through command) - 0x01 – Online PIN DUKPT - 0x02 – Online PIN MKSK - 0x03 – Offline PIN - 0x10 – SCRPP Generic - 0x11 – SCRPP MagicCube If Mode byte is "Cancel", don't need to send below field. |
| DATA PAIRING DUKPT KSN Length | 2 | This field is available only when Mode is 0x01 or 0x02 Format: <Length L><Length H> (Little-endian) If the value is 00 00, the reader does not implement Pairing Function. |
| DATA PAIRING DUKPT KSN Value | Variable | This field is available only when Mode is 0x01 or 0x02 If Length of DATA_PAIRING_DUKPT KSN is 00 00, the block does not exist |
| Encrypted Truncated PAN Length/PAN Token Length | 2 | This field is available for Encrypted Truncated PAN Length when Mode is 0x01 or 0x02, and is available for PAN Token Length when Mode is 0x10 Format: <Length L><Length H> (Little-endian) If Length of DATA_PAIRING_DUKPT KSN is 00 00, PAN in Plaintext. |
| Encrypted Truncated PAN/PAN Token | Variable | This field is available for Encrypted Truncated PAN when Mode is 0x01 or 0x02, and is available for PAN Token when Mode is 0x10 If Length of DATA_PAIRING_DUKPT KSN is 00 00, PAN in Plaintext. |
| Length of Enc[SCRPP PIN Conveyance Key] | 2 | This field is available for Enc [SCRPP PIN Conveyance Key] Length when Mode is 0x10 Format: <Length L><Length H> (Little-endian) |
| Enc [SCRPP PIN Conveyance Key] | 16 | This field is available for Enc [SCRPP PIN Conveyance Key] Length when Mode is 0x10 [SCRPP PIN Conveyance Key] is random generated by SCRPP. Enc [SCRPP PIN Conveyance Key] is encrypted [SCRPP PIN Conveyance Key] by PIN_Shared_Key. |
| Start PIN Input Timeout Length | 2 | Format: <Length L><Length H> (Little-endian) |
| Start PIN Input Timeout Value | Variable | Format: <Value L><Value H> (Little-endian) Unit: Second |
| PIN Entry Interval Length | 2 | Format: <Length L><Length H> (Little-endian) |
| PIN Entry Interval Value | Variable | Format: <Value L><Value H> (Little-endian) Unit: Second |
| Display Message Language Length | 2 | Format: <Length L><Length H> (Little-endian) |

| Data Item | Length (bytes) | Description |
|--------------------------|----------------|--|
| Display Message Language | Variable | "EN" – English (default) "ES" – Spanish "ZH" – Chinese "FR" – French ... |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-------------|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 02h | | | Data Objects | | |

Data Objects

| Data Item | Length (bytes) | Description |
|--------------------|-------------------|--|
| Mode | 1 | - 0x00 – Cancel (Can be cancel through command or user presses cancel key on the key pad) - 0x01 – Online PIN DUKPT. (PIN_KEY is PIN_DUKPT_KEY) - 0x02 – Online PIN MKSK. (PIN_KEY is PIN_SESSION_KEY) - 0x03 – Offline PIN. (PIN_KEY is PIN_PAIRING_DUKPT_KEY) - 0x10 – SCRIP Generic - 0x11 – SCRIP MagicCube (If the reader does not implement Pairing function, Offline PIN is Plaintext) If Mode byte is "Cancel", following field is not present. |
| KSN Length | 2 | Format: <Length L><Length H> (Little-endian) If Online PIN DUKPT, Length of PIN_DUKPT_KEY KSN If Offline PIN, Length of PIN_PAIRING_DUKPT_KEY KSN. The Length Value is 00 00 in the following cases: <ul style="list-style-type: none"> - If the reader does not implement Pairing function, Offline PIN is Plaintext. - If Mode is SCRIP Generic/MagicCube |
| KSN | Variable | If Online PIN DUKPT, PIN_DUKPT_KEY KSN – 10 bytes If Offline PIN, PIN_PAIRING_DUKPT_KEY KSN. This field is not present in the following cases: <ul style="list-style-type: none"> - If the reader does not implement Pairing function, Offline PIN is Plaintext. - If Mode is SCRIP Generic/MagicCube |
| PIN Payload Length | 2 | Format: <Length L><Length H> (Little-endian) |

| Data Item | Length (bytes) | Description |
|-------------|----------------|--|
| PIN Payload | Variable | If Online PIN DUKPT or Online PIN MKSK, Enciphered PIN If Offline PIN, Enciphered PIN. If SCRP Generic, Enciphered format 4 PIN block using [SCRP PIN Conveyance Key] If SCRP MagicCube, whole PIN payload (If the reader does not implement Pairing function, Offline PIN is Plaintext) |

Note: Upon finishing this command, [SCRP PIN Conveyance Key] is erased.

7.4.27 Get Random Number (19-01)

The **Get Random Number** command to gets random numbers from a ViVOpay reader. The suggested random number length is 256 bytes.

In SCRP (Secure Card Reader for PIN), this command can be used as the first step in refreshing a secure enablement token.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 19h | 01h | 01h or 03h | | Data Objects | | |

Data Objects

| Data Item | Length (bytes) | Description |
|-------------------------|----------------|--|
| Mode | 1 | - 0x00 – Generic - 0x01 – SCRP If Mode is SCRP, the following field is not present, and a fixed 16-byte random number will be returned in response and saved in SCRP. The saved random number may be refreshed after another 19-01 is called, or cleared after 19-02 is called. |
| Length of Random Number | 2 | This field is available only when Mode is Generic Format: <Length H><Length L> (Big-endian) Valid value range is 0x0001 to 0x0100 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 19h | See status code table | 00h | 00h | | |

7.4.28 Refresh Enablement Token (19-02)

The **Refresh Enablement Token** command refreshes the secure enablement token of SCRP (Secure Card Reader for PIN) environments (and only SCRP environments).

Readers must first call the 19-01 command (above) to get a 16-byte random number, then generate a signature using the following algorithm:

1. Calculate a 32-byte hash for a 16-byte random number.
2. Use a RSAPSS algorithm to calculate the hash to 256 bytes of raw data.
3. Use the Secure Message Certificate private key to sign the raw data to be a 256-byte signature.

When an SCRP reader receives this command, it may verify the signature with the previously-saved 16-byte random number and clear the saved 16-byte random number, regardless of the result. After successful verification, the enablement token is refreshed and SCRP is enabled to accept consumer cards for another five minutes, up to a maximum of 10 minutes (as a grace period).

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14~269 | Byte 270 | Byte 271 |
|-------------------------------|---------|-------------|-------------------|-------------------|-------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Signature | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 19h | 02h | 01h | 00h | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 19h | See status code table | 00h | 00h | | |

7.4.29 Reserve Commands for SCRP (19-10, 19-11, 19-12, 19-13)

Commands 19-10 through 19-13 are used to Create Secure Channels for SCRP.

7.4.30 CVMApp Get Nonce from SCRP for Refresh Token (19-15)

The **CVMApp Get Nonce from SCRP for Refresh Token** command gets random numbers from a ViVOpay SCRP. In SCRP, this command is used as the first step in refreshing a secure enablement token.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 19h | 15h | 0001h | | Data Objects | | |

| Data Item | Length (bytes) | Description |
|-----------|----------------|-------------------------|
| Mode | 1 | 0x00 – Generic for SCRP |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 – Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|--------------------------|--------------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 19h | See status code table | 0012h | | See Data table | | |

| Data Item | Length (bytes) | Description |
|-------------------|----------------|------------------------|
| Length DeviceRN_4 | 2 | Length of [DeviceRN_4] |
| DeviceRN_4 | 16 | 16 bytes [DeviceRN_4] |

7.4.31 Refresh Enablement Token for SCRP (19-16)

The **Refresh Enablement Token for SCRP** command refreshes an SCRP secure enablement token in SCRP environments (and only SCRP environments).

First, use the **CVMApp Get Nonce from SCRP for Refresh Token (19-15)** command to get a 16-byte random number from SCRP, then generate a signature using the following algorithm:

1. Calculate a 32-byte hash for a 16-byte random number.
2. Use a RSAPSS algorithm to calculate the hash to 256 bytes of raw data.
3. Use the Secure Message Certificate private key to sign the raw data to be a 256-byte signature.

When SCRP receives this command, it may verify the signature with the previously saved 16-byte random number, and clear the saved 16-byte random number, regardless of the result. After successful verification, the enablement token is refreshed and SCRP is enabled to accept consumer cards for another five minutes, up to a maximum of 10 minutes (as a grace period).

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 – Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 19h | 16h | 0100h | | See Data Table | | |

| Data Item | Length (bytes) | Description |
|--------------------|----------------|-----------------------------------|
| Signed_MonitorRN_1 | 256 | Signed_DeviceRN_4 (Cert-S, Token) |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|----------------------------------|---------|---|----------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 19h | See status code table | 00h | 00h | | |

7.4.32 Contact Get MSR Data Control (Reader sends to Host)(61-03)

The **Contact Get MSR Data Control (Reader sends to Host)** command to connect a non-MSR reader to an external MSR device.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 03h | | | Data Objects | | |

Data Objects

| Data Item | Length (bytes) | Description |
|------------------------------------|-------------------|---|
| Timeout Length | 2 | Format: <Length L><Length H> (Little-endian) Length of Total timeout for Swipe MSR Card. |
| Timeout Value | Variable | Format: <Value L><Value H> (Little-endian). Unit:Second Total timeout for Swipe MSR Card, in second, default is 60 seconds. |
| Display Message Language Length | 2 | Format: <Length L><Length H> (Little-endian) Length of Display Message Language. |
| Display Message Language | Variable | Display Message Language, 2 byte "EN" – English (default) "ES" – Spanish "ZH" – Chinese "FR" – French ... |
| Display Message Control Length | 2 | Format: <Length L><Length H> (Little-endian) Length Display Message Control – 2 bytes |
| Display Message Control | Variable | repeatable combination of <Line><Message><0x1C> <Line> - Display line number (1-First Line, n-nth Line), Maximum 16 lines. The lower 7 bits is for line number. <ul style="list-style-type: none"> The MSB is to indicate following message is a Message String or Message ID. <ul style="list-style-type: none"> MSB – 0: Message String. (It is valid for "Menu Display" and "Language Menu Display") MSB – 1: Message ID. (It is only valid for "Menu Display") <Message> - Message String or Message ID. Message String: <ul style="list-style-type: none"> "Menu Display": character in the range of 0x20 – 0x7f, Maximum 16 characters "Language Menu Display": 2 bytes Language ID "EN" – English (default) "ES" – Spanish "ZH" – Chinese |

| Data Item | Length (bytes) | Description |
|-----------|----------------|--|
| | | "FR" – French ... Message ID: 1 byte, check LCD Foreign Language Mapping Table <0x1C> - separator |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-------------|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | | | MSR Data | | |

Note: The maximum size for MSR data from an external MSR device is 640 bytes. Tag DFEE23 data format follows the external MSR format (see ID TECH document P/N 80000502-001 for information on the Enhanced Encrypted MSR format, available for download on the [ID TECH Knowledge Base](#)).

7.5 Transaction Related Commands: Contactless

The following section describes commands related to Contactless transactions.

7.5.1 Activate Transaction Command (02-01 and 02-40)

NOTE: 02-01 is a legacy command, applicable only to non-encrypted transactions. When EMV mode encryption is ON or MSR/MSD encryption is ON, and a data encryption key is loaded, **02-01** will be disabled. Use **02-40** for production; **02-40** is a unified command for both non-encrypted and encrypted transactions, but encryption will occur if a key is present (that is, the unit is injected), encryption.

Use the **Activate Transaction** command when the reader is in Poll on Demand mode to begin a contactless EMV or contactless MagStripe Card transaction. When the reader is in "Poll on Demand" mode, the RF is turned on only after receiving an **Activate Transaction** command. The reader starts polling for cards when it receives a valid **Activate Transaction** command.

If the reader does not find a supported card (an AID that matches one of the configured AIDs in the reader) during the specified time duration, it times out and ends the transaction. If the reader finds a card within the specified time interval, it attempts to carry out the transaction. The transaction flow between the reader and the card depends on the type of card detected.

If the transaction is successful, the reader returns the data in the response data. If the transaction is not successful, yet it proceeded into the transaction state machine, the reader returns a Failed Transaction Record in the response data. The presence and format of the Clearing Record, Track Data, and Failed Transaction record depends on the type of card that was detected.

Note: While an **Activate** command is in progress, readers only accept a **Cancel** or **Stop** command. Do not send other commands until **Activate Transaction** has completed because the reader will interpret these as a **Cancel Transaction** command.

Note: For Non-SRED readers, the response format for the **Activate Transaction Command** is set according to the **Set Data Encryption Enable Flag (C7-36)** command and data encryption key. When Data Encryption is disabled, the reader responds in plaintext data format. When Data Encryption is enabled:

1. When a data encryption key exists and is valid, the reader responds with encrypted data format.
2. When a data encryption key does not exist, the reader responds in plaintext data format.
3. When a data encryption key is exhausted, the reader responds with status code 0x91 and no data.

For SRED readers, the response format for the **Activate Transaction Command** is set according to the data encryption Key. When a data encryption key exists and is valid, the device responds in encrypted data format. When a data encryption key is exhausted or does not exist, the device responds status code 0x91/0x90 and no data.

Note that in products that support multiple presentation modes (dip, tap, swipe), the **02-40** command can be used to initiate a transaction in any desired mode, with or without fallback support. To control the behavior, use the DFEF37 and DFEF3C tags.

Note that ONLY 02-40 supports DFEF37 and DFEF3C.

| Tag | Length | Description | Example |
|----------|--------|--|---|
| DF EF 37 | 01 | Define the type of interface to be activated with 02-40. Interface selection: <ul style="list-style-type: none"> • Bit 0: MSR • Bit 1: Contactless • Bit 2: Contact | DF EF 37 01 07 07 = 0000 0111 This activates transaction for all 3 interfaces. |
| DF EF 3C | 03 | Fallback support and Timeout value for waiting for the next command (mainly to support EMV workflow). Byte 1: Fallback support <ul style="list-style-type: none"> • 0x00: not support fallback • 0x01: support fallback Byte 2~3: Timeout for next command (Unit: Sec) (Hex format) <ul style="list-style-type: none"> • 00 0A = 10s • 01 00 = 256s | DF EF 3C 03 01 00 60 Fallback is supported, and the timeout is set to 96 seconds before the transaction times out. |

Example Usage:

```
5669564f746563683200024000221e9c01009f02060000000001009f0306000000
000000dfe370107dfe3c0301006018d1
```

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|---------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVotech2\0 | 02h | 40h | | | See Data Format below | | |

The format and contents of the data field in the command frame are given in the following table. *All length values include the Tag and Length bytes.*

Activate Transaction Command Frame Data Format

| Data Item | Length (bytes) | Description |
|-----------|----------------|---|
| Timeout | 1 | <p>Time in seconds that the reader waits for a card to be presented before timing out and returning an Error response. The reader will continue to poll for this amount of time if no card is found.</p> <p>Note that if a card is found, the transaction may not complete within the timeout period.</p> <p>This field must be present in the Activate command.</p> <p>If Timeout=0x00, the reader will wait forever.</p> <p>Format: Binary</p> |
| TLV Data | varies | See Activate Command TLVs below. |

Activate Command TLVs

| Tag | Description | Format | Length (in bytes) |
|------|---|----------------|-------------------|
| 9A | <p>Transaction Date. EMV data element. Local date that the transaction was authorized. If TLV 9A and 9F21 are not provided, then the reader's current date and time will be used.</p> <p>Both date (9A) and time (9F21) tags must be present if either one is specified.</p> <p>The host is responsible for ensuring that the date is valid:</p> <ul style="list-style-type: none"> ▫ Year <=99 ▫ Month <=12 ▫ Day <=31 <p>If the date value is set to 0xFF, 0xFF, 0xFF, then the date and time stamp will be taken from the reader's date and time will be used.</p> | n6 (YYMMDD) | 3 |
| 9C | <p>Transaction Type. Indicates the type of financial transaction, represented by the first two digits of ISO 8583:1993 Processing Code:</p> <ul style="list-style-type: none"> ▫ 0x00 – Purchase Goods/Services ▫ 0x20 – Refund | n2 | 1 |
| 5F2A | <p>Transaction Currency Code. Indicates the currency code of the transaction, in accordance with ISO 4217.</p> | n3 | 2 |
| 5F36 | <p>Transaction Currency Exponent Indicates the implied position of the decimal point from the right of the transaction amount, according to ISO 4217.</p> | n1 | 1 |

| Tag | Description | Format | Length (in bytes) |
|--------|--|----------------|----------------------|
| 9F02 | Amount, Authorized | n12 | 6 |
| 9F03 | Amount Other | n12 | 6 |
| 9F1A | Host Country Code. (Typically, this has been configured in the reader – refer to Group Configuration Tags .) | n | 2 |
| 9F21 | Transaction Time. Local time that the transaction was authorized. If TLV 9A and 9F21 are not provided, then the reader's current date and time will be used. <div style="border: 2px solid red; background-color: yellow; padding: 5px;">Both date (9A) and time (9F21) tags must be present if either one is specified.</div> The host is responsible for ensuring that the time is valid. | n6 (HHMMSS) | 3 |
| 9F5A | Host Transaction Type (Interac) <ul style="list-style-type: none"> ▫ 0x00 = Purchase ▫ 0x01 = Refund | b | 1 |
| DDEF7A | Determine Apple Pay or Google Pay transactions. If the value is set to 1, after reading the card the output data will show tag DDEF7B to indicate whether the transaction is Apple Pay or Google Pay. | | |
| FFEE01 | ViVOtech TLV Group Tag | b | variable |

For EMV transactions, if the host device has already set up one or more of these data items using the [Set Configuration](#) or [Set Configurable Group](#) command, then the host device need not include those data items in the command frame. If the host includes one or more values in the command frame, the reader uses the included values. If it does not, the reader just uses the default or previously set values.

The reader starts polling for cards when it receives this command. If it finds a card, it tries to complete a transaction with the card. If the card is a supported contactless EMV card, the reader uses the TLV fields in the command frame for the transactions. If the card is a contactless MagStripe card, the reader does not use the TLV objects for the transaction.

If the transaction is completed successfully, and the card supported contactless EMV, then the reader returns the clearing record in the response data. Otherwise, if the card does not support contactless EMV it is a contactless MagStripe card and the reader returns track information in the response data.

If the transaction cannot be completed successfully, the response contains an appropriate status code. The response frame contains more error information in the data field for certain status codes.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|--------------------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 02h | See status code table | | | See Response Frame Data Format | | |

Note: Specific TLV data may or may not be returned based on what was recovered from the card. Also, there is no implied sequence for returning the TLVs; the TLVs may or may not be returned in the order listed in the table based on what was recovered from the card.

Note: ViVOcomm and DesFire cards return raw track data only.

If the Status Code is OK or "Request Online Authorization," the format and contents of the data field in the response frame are given in the following table.

Some data objects may not be present depending on the card involved in the transaction and the presence or absence of a clearing record object (DE 055). All TLV lengths include the Tag and Length bytes.

Activate Transaction Response Frame Data Format: 02-01 Command

For 02-40, see below.

| Data Item | Length (bytes) | Description |
|----------------------------------|-------------------|--|
| Track 1 Length | 1 | If Track 1 is available, then this field gives the length of the Track 1 data that follows. If Track 1 is not available, then a Length of 00h is returned. Format: Binary |
| Track 1 Data (MagStripe Card) | Variable | Track 1 Data (if available). Format: ASCII (no null terminator) |
| Track 2 Length | 1 | If Track 2 is available, then this field gives the length of the Track 2 data that follows. If Track 2 is not available, then a Length of 00h is returned. Format: Binary |
| Track 2 Data (MagStripe Card) | Variable | Track 2 Data (if available). Format: ASCII (no null terminator) |
| Track 3 Length | 1 | If Track 3 is available, then this field gives the length of the Track 3 data that follows. If Track 3 is not available, then a Length of 00h is returned. Format: Binary |
| Track 3 Data (MagStripe Card) | Variable | Track 3 Data (if available). Format: ASCII (no null terminator) |

| Data Item | Length (bytes) | Description |
|-----------------------------------|----------------|---|
| DE055 (Clearing Record) Present.) | 1 | If a Clearing Record (DE 055) field is available, then this field is 01h. If there is no Clearing Record (DE 055) field, then this field is 00h. For MasterCard transactions, this field is always 0. |
| TLV DE 055 (Clearing Record) | Variable | DE 055 data (if available) as a TLV data object encoded with Tag 'E1'. The DE 055 data is the same data as is included in the Clearing Record. Refer to the Activate Transaction Clearing Record table. Tag: E1 BER-TLV group tag. |
| TLV Data | Variable | See Activate Response TLVs below. |

MasterCard transactions do not return a clearing record or the track data fields. Tags are returned in a format specified by M/Chip 3.0. Track 1 and Track 2 data are encapsulated in tags according to the MasterCard specification.

Activate Transaction Response TLVs

| Tag | Description | Format | Length (in bytes) |
|-----|--|----------------|--------------------|
| 50 | Application Label. Name associated with the AID, in accordance with ISO/IEC 7816-5. | an | <= 16 |
| 56 | Track 1 Equivalent Data. Contains the data objects of the track 1 according to [ISO/IEC 7813] Structure B, excluding start sentinel, end sentinel and LRC. The <i>Track 1 Data</i> may be present in the file read using the READ RECORD command during a MagStripe mode transaction. | ans | <=79 |
| 57 | Track 2 Equivalent Data. Contains the data objects of the track 2, in accordance with ISO/IEC 7813, excluding start sentinel, end sentinel, and LRC. | b | <=19 |
| 5A | Application Primary Account Number (PAN). The cardholder account number. | cn | <=19 (10 bytes) |
| 82 | Application Interchange Profile. Indicates the capabilities of the Card to support specific functions in the application. | b | 2 |
| 84 | DF Name. Identifies the name of the DF as described in ISO/IEC 7816-4. | b | 5..16 |
| 95 | Host Verification Results. Status of various functions from the host perspective. | b | 5 |
| 99 | Online PIN Block | n | 16 |
| 9A | Transaction Date. Local date that the transaction was performed. | n6 (YYMMDD) | 3 |
| 9B | Transaction Status Information. | b | 2 |

| Tag | Description | Format | Length (in bytes) |
|------|--|----------------|----------------------|
| 9C | Transaction Type. Indicates the type of financial transaction, represented by the first two digits of ISO 8583:1993 Processing Code. | n2 | 1 |
| 5F20 | Cardholder Name. | b | <=26 |
| 5F24 | Application Expiration Date. The date after which the card application has expired. | n6 (YYMMDD) | 3 |
| 5F25 | Application Effective Date. Date from which the card application may be used. | n6 (YYMMDD) | 3 |
| 5F2A | Transaction Currency Code. Indicates the currency code of the transaction, in accordance with ISO 4217. | n3 | 2 |
| 5F2D | Language Preference. 1-4 languages stored in order of preference, each represented by 2 alphabetical characters according to ISO 639. | an | 2..8 |
| 5F34 | PAN Sequence Number. Identifies and differentiates cards with the same <i>Application PAN</i> . | n2 (BCD) | 1 |
| 9F02 | Amount, Authorized | n12 | 6 |
| 9F03 | Amount Other | n12 | 6 |
| 9F06 | Application Identifier (AID). | b | 5..16 |
| 9F07 | Application Usage Control. | b | 2 |
| 9F09 | Application Version Number (Reader) Version number assigned by the payment system for the Kernel application. | b | 2 |
| 9F0D | Issuer Action Code (Default). | b | 5 |
| 9F0E | Issuer Action Code (Denial). | b | 5 |
| 9F0F | Issuer Action Code (Online). | b | 5 |
| 9F10 | Issuer Application Data. Contains proprietary application data for transmission to the issuer in an online transaction. | b | <=32 |
| 9F11 | Issuer Code Table Index. Indicates the code table according to ISO/IEC 8859 for displaying the Application Preferred Name. | n2 | 1 |
| 9F12 | Application Preferred Name. The preferred mnemonic associated with the AID. | ans | <=16 |
| 9F1A | Host Country Code. (Typically, this has been configured in the reader – refer to Group Configuration Tags .) | n | 2 |
| 9F1E | Interface Device Serial Number. Unique and permanent serial number assigned to the IFD by the manufacturer. (Typically, this has been configured in the reader – refer to Group Configuration Tags .) | an | 8 |

| Tag | Description | Format | Length (in bytes) |
|------|--|----------------|----------------------|
| 9F21 | Transaction Time. Local time that the transaction was performed. | n6 (HHMMSS) | 3 |
| 9F26 | Application Cryptogram. This is returned in the response to GenAC or RecoverAC. | b | 8 |
| 9F27 | Cryptogram Information Data Indicates the type of cryptogram and the actions to be performed. | b | 1 |
| 9F33 | Host Capabilities. Indicates the card data input, CVM, and security capabilities of the Host and Reader. | b | 3 |
| 9F34 | Cardholder Verification Method (CVM) Results. Indicates result of last CVM performed. | b | 3 |
| 9F35 | Host Type. (Typically, this has been configured in the reader – refer to Group Configuration Tags .) | n2 | 1 |
| 9F36 | Application Transaction Counter. Counter maintained by the application in the card. | b | 2 |
| 9F37 | Unpredictable Number. A challenge number used by the card to ensure uniqueness of the generated cryptogram. | b | 4 |
| 9F39 | Point of Service (POS) Entry Mode. Indicates the method by which the PAN was entered. Values: 90h = Magnetic Stripe Reader Swipe 91h = Contactless MSD 05h = Contact EMV 07h = Contactless EMV 80h = Contact Fallback to Magnetic Stripe | n2 | 1 |
| 9F42 | Application Currency Code. Indicates the currency in which the account is managed in accordance with ISO 4217. | n3 | 2 |
| 9F45 | Data Authentication Code. | b | 2 |
| 9F4C | ICC Dynamic Number. | b | 8 |
| 9F53 | Transaction Category Code. Indicates the type of transaction being performed, and which may be used in card risk management. | an | 1 |
| 9F5A | Membership Scheme - Account Number (Amex) | an | <=5 |
| | Or Host Transaction Type (Interac) | b | 1 |
| 9F5B | Membership Scheme Number of Points (Amex). | an | <=29 |
| | Issuer Script Result(UICS) | b | <=128 |
| 9F5D | Available Offline Spending Amount (Balance). | b | 6 |

| Tag | Description | Format | Length (in bytes) |
|----------------------------------|---|----------|----------------------|
| 9F6B | Track 2 Data. Contains the data objects of the track 2 according to ISO/IEC 7813, excluding start sentinel, end sentinel and LRC. | b | <=19 |
| 9F6C | Card Transaction Qualifiers (Visa transactions only). If card does not return this tag then a length of zero is returned. | b | 16 |
| 9F6D | MagStripe Application Version Number (Reader). Version number assigned by the payment system for the specific MagStripe mode functionality of the Kernel. | b | 2 |
| 9F6E | Third Party Data. Contains various information, possibly including information from a third party. | b | 5..32 |
| 9F74 | VLP Issuer Authorization | b | 6 |
| E300 | Authorization Code. | b | 8 |
| DF30 (ViVOPay proprietary) | Track Data Source. This tag is embedded in the ViVOPay Group tag. It specifies whether the track data came from a swipe or RFID transaction. 0Ch for swiped MagStripe 00h for a contactless transaction. 01h for a contact transaction. | b | 1 |
| DF33 (ViVOPay proprietary) | Receipt Requirement (Interac) This tag is embedded in the ViVOPay Group tag for Interac transaction responses. 00 = No receipt required 01 = Receipt required | b | 1 |
| DF5B (ViVOPay proprietary) | Host Entry Capability (Visa). For Visa Transactions, defines reader support for VSDC contact chip. Values: 05h = Reader supports VSDC contact chip 08h = Reader does not support VSDC contact chip | n2 | 1 |
| DF8104 | Balance Read Before GenAC. (MasterCard) Balance read from the card before the GenAC. | n12 | 6 |
| DF8105 | Balance Read After GenAC. (MasterCard) Balance read from the card after the GenAC. | n12 | 6 |
| DF812A | DD Card Track 1 (MagStripe Card) If Track 1 Data is present, then this tag contains a copy of the discretionary data field of Track 1 Data as returned by the card. | b | <= 56 |
| DF812B | DD Card Track 2 (MagStripe Card) If Track 2 Data is present, then this tag contains a copy of the discretionary data field of Track 2 Data as returned by the card. | b | <=8 |
| DF8115 | Error Indication. (MasterCard) Flags defining the error conditions from the transaction. Refer to the M/Chip PayPass specification. | b | 6 |

| Tag | Description | Format | Length (in bytes) |
|--------|--|--------|----------------------|
| DF8129 | Outcome Parameter Set (MasterCard). Contains the result of the transaction. Refer to the M/Chip PayPass specification. | b | 8 |
| DFEF7F | VISA ODA/Online indicator (within FFEE01) Bit 7 – ODA/Online performed: 1 performed, 0 Not performed Bit 6 – RFU Bit 5 – RFU Bit 4 – RFU Bit 3 – fDDA performed: 1 performed, 0 Not performed Bit 2 – fDDA failed: 1 Failed, 0 Succeed Bit 1 – SDA performed: 1 performed, 0 Not performed Bit 0 – SDA failed: 1 Failed, 0 Succeed | b | 1 |
| DFEE26 | Encryption and Data Capture Information. Length: 1 or 2 (variable) Values: Byte 1: Same as "Attribution" byte definition. Namely: Bits 4/3/0: Captured Data Type 0 0 0 = Contact Transaction 0 0 1 = Contactless Transaction / EMV 1 0 1 = Contactless Transaction / MSD 0 1 x = MSR Card Bits 2/1: Encryption Mode 0 0 = TDES 0 1 = AES 1 x = Refer to "Extended Encryption Mode" further below. Bit 5: Attribution Byte Extension. 0 = Length of tag is 1 byte 1 = Length of tag is 2 bytes Bits 6/7: Encryption Status 0 0 = EMV off, MSR/MSD off 0 1 = EMV off , MSR/MSD on 1 0 = EMV on, MSR/MSD off 1 1 = EMV on, MSR/MSD on Byte 2: (When applicable; that is, when Length of TLV is 2) Bits 3/2/1/0: Extended Encryption Mode 0 0 0 0 = TDES 0 0 0 1 = AES 0 0 1 0 = TransArmor Algorithm 0 0 1 1 = Voltage Algorithm 0 1 0 0 = Visa FPE | b | 1 |

| Tag | Description | Format | Length (in bytes) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------|---|--------|----------------------|----|----|----|----|------------------------|----|--|---|---|---|---|---|---|---|---|-----------------------|---|---|---|---|---|---|---|---|-----------------|---|---|---|---|---|---|---|---|------------------|---|---|---|---|---|---|---|---|------------------------|---|---|---|---|---|---|---|---|--------------|---|---|
| | <p>0 1 0 1 = Verifone FPE 0 1 1 0 = TransArmor TDES-DUKPT</p> <p>Bits 4 to 6: Reserved</p> <p>Bit 7: 0 = No MAC Verification Data 1 = Has MAC Verification Data</p> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DFEF73 | <p>CUP Output Message Info (CUP) Byte 1</p> <table><tr><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th></th></tr><tr><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>X</td><td>Authorisation Message</td></tr><tr><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>X</td><td>-</td><td>Confirm Message</td></tr><tr><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>X</td><td>-</td><td>-</td><td>Reversal Message</td></tr><tr><td>-</td><td>X</td><td>X</td><td>X</td><td>X</td><td>-</td><td>-</td><td>-</td><td>Host Floor Limit Check</td></tr><tr><td>X</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>Date Expired</td></tr></table> | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | | - | - | - | - | - | - | - | X | Authorisation Message | - | - | - | - | - | - | X | - | Confirm Message | - | - | - | - | - | X | - | - | Reversal Message | - | X | X | X | X | - | - | - | Host Floor Limit Check | X | - | - | - | - | - | - | - | Date Expired | b | 1 |
| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| - | - | - | - | - | - | - | X | Authorisation Message | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| - | - | - | - | - | - | X | - | Confirm Message | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| - | - | - | - | - | X | - | - | Reversal Message | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| - | X | X | X | X | - | - | - | Host Floor Limit Check | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| X | - | - | - | - | - | - | - | Date Expired | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| DFEF7B | <p>If tag DFEF7A value is set to 1, after reading the card the output data will show tag DFEF7B to indicate Apple Pay or Google Pay.</p> <p>If tag DFEF7A is not set, the output data will not output the tag.</p> <p>0 = Not Apple Pay/ Apple VAS or Google Pay/ Android VAS 1 = Apple Pay or Apple VAS 2 = Google Pay or Android VAS</p> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| FF8105 | <p>Data Record (MasterCard, container).</p> <p>Contains the data from the transaction. Refer to the M/Chip PayPass specification.</p> | b | varies | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| FF8106 | <p>Discretionary Data (MasterCard, container).</p> <p>Contains the discretionary data from the transaction. Refer to the M/Chip PayPass specification.</p> | b | varies | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| FFEE01 | <p>ViVOPay Group Tag. (container)</p> <p>This three-byte Group Tag was created to contain ViVopay proprietary Tags. See tags below.</p> | b | <=76 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

If a Clearing Record is returned, its potential TLVs are described in the following table.

Different card applications may have a slightly different format, or different TLVs, for the clearing record. MasterCard M/Chip 3.0 does not return a clearing record.

Activate Transaction Clearing Record TLVs

| Tag | Data Element Name | Format | Origin |
|------|--|---------------|--------|
| E1 | DE 055 Clearing Record (group tag) | Binary, var | |
| 50 | Application Label | ans1...16 var | Card |
| 82 | Application Interchange Profile | b2 | Card |
| 84 | DF name | b | Card |
| 95 | Host Verification Results: Indicates results of various transaction processes. | b 5 | Reader |
| 9A | Transaction Date | n6 | Host |
| 9B | Transaction Status Information | b2 | Host |
| 9C | Transaction Type <ul style="list-style-type: none"> ▫ 0x00 = Purchase Goods/Services ▫ 0x20 = refund | n2 | Reader |
| 5F2A | Transaction Currency Code | n3 | Host |
| 5F2D | Language Preference | an2-8 var | Card |
| 5F34 | Application Primary Account Number Sequence Number | n2 | Card |
| 9F02 | Amount, Authorized (Numeric) | n12 | Host |
| 9F03 | Amount, Other (Numeric, Visa only) | n12 | Host |
| 9F08 | Application Version Number | b2 | Card |
| 9F09 | Application Version Number | b2 | Card |
| 9F10 | Issuer Application Data | b1-32 var | Card |
| 9F11 | Issuer Code Table Index | n2 | Card |
| 9F12 | Application Preferred Name | ans1-16 var | Card |
| 9F1A | Host Country Code | n2 | Host |
| 9F21 | Transaction Time | n6 | Host |
| 9F26 | Application Cryptogram | b8 | Card |
| 9F27 | Cryptogram Information Data | b1 | Card |
| 9F33 | Host Capabilities | b3 | Host |
| 9F34 | Cardholder Verification Method (CVM) Results | b3 | Host |
| 9F35 | Host Type (Interac) | b1 | Host |
| 9F36 | Application Transaction Counter | b2 | Card |
| 9F37 | Unpredictable Number | b4 | Reader |
| 9F39 | POS Entry Mode | b1 | Reader |
| 9F45 | Data Authentication Code | b2 | Card |
| 9F4C | ICC Dynamic Number | b2-8 var | Card |
| 9F59 | Host Transaction Information (Interac) | b3 | Host |
| 9F5A | Host Transaction Type (Interac) | b1 | Host |
| 9F66 | Visa TTQ | b4 | Reader |
| 9F6C | Card Transaction Qualifiers (Visa only) | b2 | Card |

| Tag | Data Element Name | Format | Origin |
|--------|--|------------|--------|
| 9F6E | Form Factor Indicator | b4 | Reader |
| | or PayPass Third Party Data | b5-32 var | Reader |
| 9F74 | VLP Issuer Authentication Code | b | Host |
| 9F7C | Customer Exclusive Data | b1-32 var | Reader |
| DF30 | Track Data Source <ul style="list-style-type: none"> 0x0C – swiped MagStripe card 0x00 – contactless transaction 0x01 – contact transaction | b1 | Reader |
| DF33 | Receipt Required (Interac) – calculated by checking the Interac Host Receipt Required Limit (9F5F). | b1 | Reader |
| DF52 | Transaction CVM: <ul style="list-style-type: none"> 00 = for No CVM 01 = for Signature 02 = for Online PIN 03 = for Mobile CVM / Consumer Device CVM | b1 | Reader |
| DF76 | TVR Backup – value of TVR prior to GenAC | b5 | Reader |
| FFEE01 | ViVOPay Proprietary Group Tag | b variable | Host |

If the status code returned in the response frame is “Failed” and the error code is not “Request Online Authorization,” the data field contains further information on the cause of the failure and does not contain the track or clearing record information. In this case, the data field in the response frame has the following format:

Activate Transaction Cause of Failure When Not Requesting Online Authorization

| Data Field | Length (bytes) | Description |
|---------------|----------------|--|
| Error Code | 1 | Error Code giving the reason for the failure. See sub-section on Error Codes |
| SW1 | 1 | Value of SW1 returned by the Card (SW1SW2 is 0000 if SW1 SW2 not available) |
| SW2 | 1 | Value of SW2 returned by the Card (SW1SW2 is 0000 if SW1 SW2 not available) |
| RF State Code | 1 | RF State Code indicating exactly where the error occurred in the Reader-Card transaction flow. See sub-section on RF State Codes . |

If the status code returned in the response frame is “Failed” and the error code is “Request Online Authorization,” the data field contains further information on the cause of the failure and does not contain the track or clearing record information. In this case, the data field in the response frame has the following format:

Activate Transaction Cause of Failure When Requesting Online Authorization

| Data Field | Length (bytes) | Description |
|-----------------------------|-----------------------------|--|
| Error Code | 1 | Error Code giving the reason for the failure. See sub-section on Error Codes |
| SW1 | 1 | Value of SW1 returned by the Card (SW1SW2 is 0000 if SW1 SW2 not available) |
| SW2 | 1 | Value of SW2 returned by the Card (SW1SW2 is 0000 if SW1 SW2 not available) |
| RF State Code | 1 | RF State Code indicating exactly where the error occurred in the Reader-Card transaction flow. See sub-section on RF State Codes . |
| TLV Track 2 Equivalent Data | 21 (including Tag & Length) | Track 2 Equivalent Data as a TLV object. Tag: 57 Format: b19 |
| Amount Requested | 6 | Difference between the Terminal Contactless Transaction Limit (FFF1) and Balance. Format: n12 |

If the status code is "User Interface Event," the format and contents of the data field in the response frame are:

| Data Item | Length (bytes) | Description |
|--------------------|----------------|--|
| Transaction status | 1 | 01: The reader detects the card and initiates the transaction. |

For any other status code the data field is empty.

If the transaction failed, the response frame may have the following format. Invalid or inappropriate cards may result in no response frame.

Activate Transaction Response Frame Format, Failed Transaction: 02-01 Command

For 02-40, see further below.

| Data Item | Length (bytes) | Description |
|---------------|----------------|---|
| Error Code | 1 | Error Code giving the reason for the failure. See sub-section on Error Codes |
| SW1 | 1 | Value of SW1 returned by the Card (SW1SW2 is 0000 if SW1 SW2 not available) |
| SW2 | 1 | Value of SW2 returned by the Card (SW1SW2 is 0000 if SW1 SW2 not available) |
| RF State Code | 1 | RF State Code indicating exactly where the error occurred in the Reader-Card transaction flow. See RF State Codes . |
| TLV data | Varies | Refer to the Activate Response TLVs . |

If the status code being returned in the response frame is "Request Online PIN," the response data has the following format:

| Data Item | Length (bytes) | Description |
|---------------------|-------------------|--|
| TLV Application PAN | Variable up to 12 | Application Primary Account Number (PAN) as a TLV object. Tag: 5A Format: cn variable length up to 19(10 bytes) |

1.1.2.1. Special TLV for Discover D-PAS and Smart Tap

D-PAS uses a ViVOtech proprietary tag, "TLV Special Flow" in **Activate Transaction** command data.

Command Data

| Data Item | Length (Bytes) | Description | | | | | | |
|-------------------------------|--|---|---|--|--|-------------------|--|--|
| ViVOTech TLV Group Tag FFEE01 | Variable | <table><tr><td>TLV Special Flow</td><td>2 Byte Tag + 1 Byte Length + Variable data (max 40 bytes)</td><td>This TLV defines one or more special transaction flows for specific non-payment process. Tag: DF50 Format: The TLV value contains one or more 4-byte "Special Flow Record" entries. The format of a Special Flow Record is given in the next table. A Special Flow TLV cannot have more than 10 record entries.</td></tr><tr><td>TLV Issuer Script</td><td>2 Byte Tag + n-Byte Length + Variable Data</td><td>This TLV is for Discover D-PAS only. This TLV contains the Issuer Script that is to be sent to the card. The Issuer Script is defined by the Discover D-PAS specification. Tag: DF51 Format: Raw data</td></tr></table> | TLV Special Flow | 2 Byte Tag + 1 Byte Length + Variable data (max 40 bytes) | This TLV defines one or more special transaction flows for specific non-payment process. Tag: DF50 Format: The TLV value contains one or more 4-byte "Special Flow Record" entries. The format of a Special Flow Record is given in the next table. A Special Flow TLV cannot have more than 10 record entries. | TLV Issuer Script | 2 Byte Tag + n-Byte Length + Variable Data | This TLV is for Discover D-PAS only. This TLV contains the Issuer Script that is to be sent to the card. The Issuer Script is defined by the Discover D-PAS specification. Tag: DF51 Format: Raw data |
| | | TLV Special Flow | 2 Byte Tag + 1 Byte Length + Variable data (max 40 bytes) | This TLV defines one or more special transaction flows for specific non-payment process. Tag: DF50 Format: The TLV value contains one or more 4-byte "Special Flow Record" entries. The format of a Special Flow Record is given in the next table. A Special Flow TLV cannot have more than 10 record entries. | | | | |
| TLV Issuer Script | 2 Byte Tag + n-Byte Length + Variable Data | This TLV is for Discover D-PAS only. This TLV contains the Issuer Script that is to be sent to the card. The Issuer Script is defined by the Discover D-PAS specification. Tag: DF51 Format: Raw data | | | | | | |
| | | | | | | | | |

Special Flow Record Format

| Byte # | Field | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------|---|------|------|------|------|------|------|--|------|------------------|---|---|---|---|---|---|--|--|------------------|--|--|--|--|--|---|---|--|-------------------------------|--|--|--|--|--|--|---|---|------------------------------|--|--|--|--|--|--|--|---|--|
| 1 | Application Flow Code This is the card application for which this special flow entry is being defined. This can be any of the Application Flows defined for the ViVOTech2 “Set Configurable AID” such as Discover D-PAS. This also can be any special Application Flow that does not use ViVOTech2 “Set Configurable AID” such as Smart Tap. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2 | Special Transaction Flow Flags This byte defines the nature of the special flow and indicates whether the Application Flow defined in the previous byte occurs during pre-PPSE or post-PPSE processing. It also indicates whether a standard ISO Transaction will be performed in between pre-PPSE and post-PPSE processing or skipped. The flags within this byte are given below. <table><tr><th>B7</th><th>B6</th><th>B5</th><th>B4</th><th>B3</th><th>B2</th><th>B1</th><th>B0</th><th>Flag Description</th></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td></td><td></td><td></td><td>Unused. Set to 0</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td>1</td><td></td><td></td><td>Perform Post-PPSE Transaction</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td><td>1</td><td></td><td>Perform Pre-PPSE Transaction</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>1</td><td>Perform ISO Payment Transaction (PPSE+AID)</td></tr></table> | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | Flag Description | 0 | 0 | 0 | 0 | 0 | | | | Unused. Set to 0 | | | | | | 1 | | | Perform Post-PPSE Transaction | | | | | | | 1 | | Perform Pre-PPSE Transaction | | | | | | | | 1 | Perform ISO Payment Transaction (PPSE+AID) |
| B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | Flag Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | 0 | 0 | | | | Unused. Set to 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | 1 | | | Perform Post-PPSE Transaction | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | 1 | | Perform Pre-PPSE Transaction | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | 1 | Perform ISO Payment Transaction (PPSE+AID) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3 | Special Transaction Type Flags for Pre-PPSE Processing This byte indicates the type of special (non-payment) transaction that will be performed in the Pre-PPSE transaction processing (if any). The flags within this byte are given below. <table><tr><th>Bit7</th><th>Bit6</th><th>Bit5</th><th>Bit4</th><th>Bit3</th><th>Bit2</th><th>Bit1</th><th>Bit0</th><th>Flag Description</th></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td></td><td></td><td>Unused. Set to 0</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td><td>1</td><td></td><td>Write Transaction</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>1</td><td>Read Transaction</td></tr></table> | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | Flag Description | 0 | 0 | 0 | 0 | 0 | 0 | | | Unused. Set to 0 | | | | | | | 1 | | Write Transaction | | | | | | | | 1 | Read Transaction | | | | | | | | | |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | Flag Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | 0 | 0 | 0 | | | Unused. Set to 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | 1 | | Write Transaction | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | 1 | Read Transaction | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 4 | Special Transaction Type Flags for Post-PPSE Processing This byte indicates the type of special (non-payment) transaction that will be performed in the Post-PPSE transaction processing (if any). The flags within this byte are given below. <table><tr><th>Bit7</th><th>Bit6</th><th>Bit5</th><th>Bit4</th><th>Bit3</th><th>Bit2</th><th>Bit1</th><th>Bit0</th><th>Flag Description</th></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td></td><td></td><td>Unused. Set to 0</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td><td>1</td><td></td><td>Write Transaction</td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>1</td><td>Read Transaction</td></tr></table> | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | Flag Description | 0 | 0 | 0 | 0 | 0 | 0 | | | Unused. Set to 0 | | | | | | | 1 | | Write Transaction | | | | | | | | 1 | Read Transaction | | | | | | | | | |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | Flag Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | 0 | 0 | 0 | | | Unused. Set to 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | 1 | | Write Transaction | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | 1 | Read Transaction | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Special Flow for Discover D-PAS

For Discover D-PAS, Application Flow Code is 14 (0Dh).

For Discover D-PAS 2nd Presentment of Issuer Update Process, the value in the Special Flow TLV will be:

Pre-PPSE Write D-PAS On-Line Script Processing DF50 04 0D 02 02 00

The 0xFFEE01 TLV format for Discover D-PAS when input Issuer Script for Issuer Update Processing:

FFEE01 <len> DF50 04 0D020200 DF51 <len><Issuer Script>

<Issuer Script> is defined by D-PAS spec.

Steps to do Issuer Update Processing for Discover D-PAS:

1. Set TTQ Byte3 Bit8 "Issuer Update Processing supported" to **1**.
2. Use **Activate Transaction (02-01)** to do a normal Discover D-PAS transaction; the reader responds with the transaction result and data.
3. After the receiving the online request from the reader, the host device can send **Activate Transaction (02-01)** to the reader with Issuer Scripts (DF51 TLV) in the FFEE01 TLV. The reader then runs Issuer Update Processing and responds with the TVR and TSI.

Special Flow for UICS

For UICS, the special flow code, Issuer Script format, and usage are the same as D-PAS.

Special TLV for PayPass Application

This design connects the M/Chip signal handling to the existing user interface module ("Process D") to support MSG and OUT signals in readers with displays.

Reference files

- [1] EMV Contactless Book C-2, Kernel 2 Specification, v2.3
- [2] PayPass Test Cases for PayPass v3.0 Level 2 Reader Testing, Aug 2011
- [3] EMV Contactless Specifications for Payment Systems, Book A, Architecture and General Requirements, v2.3
- [4] PayPass M/Chip Reader Card Application Interface Specification v3.0.2
- [5] Engineering Specification, MChip 3.0 on GR, v1.6

MSG Signals

Other processes in the reader use MSG Signals to send User Interface Request Data to Process D. Process D manages the User Interface Requests as defined in reference [3] and displays a message or a status. The User Interface Request Data is defined in reference [1] as tag DF8116 and holds 22 bytes of data as shown below:

| Data Field | Length |
|---------------------|--------|
| Message Identifier | 1 |
| Status | 1 |
| Hold Time | 3 |
| Language Preference | 8 |
| Value Qualifier | 1 |
| Value | 6 |
| Currency Code | 2 |

OUT Signals

The Kernel (Process K) uses OUT Signals to indicate the a transaction's outcome. According to reference [1], the following objects may comprise the OUT signal:

| Data Field | Tag | Length |
|--------------------------------------|--------|--------|
| Outcome Parameter Set | DF8129 | 8 |
| Data Record (if any) | FF8105 | var. |
| Discretionary Data | FF8106 | var. |
| User Interface Request Data (if any) | DF8116 | 22 |

According to reference [1], all objects listed below are added to the output buffer if they are present.

The data record (FF8105) may contain the following objects for an EMV transaction, with a maximum length of 256 bytes (74 TL, 182 V).

| Data Field | Tag | Length |
|-------------------------------------|------|--------|
| Amount, Authorized (Numeric) | 9F02 | 6 |
| Amount, Other (Numeric) | 9F03 | 6 |
| Application Cryptogram | 9F26 | 8 |
| Application Expiration Date | 5F24 | 3 |
| Application Interchange Profile | 82 | 2 |
| Application Label | 50 | 16 |
| Application PAN | 5A | 16 |
| Application PAN Sequence Number | 5F34 | 1 |
| Application Preferred Name | 9F12 | 16 |
| Application Transaction Counter | 9F36 | 2 |
| Application Version Number (Reader) | 9F08 | 2 |
| Cryptogram Information Data | 9F27 | 1 |
| CVM Results | 9F34 | 3 |
| DF Name | 84 | 16 |
| Interface Device Serial Number | 9F1E | 8 |
| Issuer Application Data | 9F10 | 32 |
| Issuer Code Table Index | 9F11 | 1 |
| Host Capabilities | 9F33 | 3 |
| Host Country Code | 9F1A | 2 |
| Host Type | 9F35 | 1 |
| Host Verification Results | 95 | 5 |
| Track 2 Equivalent Data | 57 | 19 |
| Transaction Category Code | 9F53 | 1 |
| Transaction Currency Code | 5F2A | 2 |
| Transaction Date | 9A | 3 |
| Transaction Type | 9F21 | 3 |
| Unpredictable Number | 9F37 | 4 |

The data record may contain the following objects for a MagStripe transaction, with a possible maximum length of 182 bytes (20 TL, 162 V).

| Data Field | Tag | Length |
|--------------------------------------|------|--------|
| Application Label | 50 | 16 |
| Application PAN | 5A | 16 |
| Application Preferred Name | 9F12 | 16 |
| MagStripe Application Version Number | 9F6D | 2 |
| DF Name | 84 | 16 |
| Issuer Code Table Index | 9F11 | 1 |
| Track 1 Data | 56 | 76 |
| Track 2 Data | 9F6B | 19 |

Discretionary data is always included in an OUT signal. Discretionary data for an EMV transaction may include the following objects, with a possible maximum length of 1009 bytes (61 TL, 948 V), ignoring the data storage elements. Without the torn record the maximum size would only be 80 bytes.

| FF8106 | var | Discretionary Data | | |
|--------|-----|--------------------|-----|-----------------------------|
| | | 9F42 | 2 | Application Currency Code |
| | | DF8105 | 6 | Bal Read After Gen AC |
| | | DF8104 | 6 | Bal Read Before Gen AC |
| | | | | DS Summary 3 |
| | | | | DS Summary Status |
| | | DF8115 | 6 | Error Indication |
| | | DF810E | 1 | Post-Gen AC Put Data Status |
| | | DF8105 | 1 | Pre-Gen AC Put Data Status |
| | | 9F6E | 32 | Third Party Data |
| | | FF8101 | 894 | Torn Record |

A torn record (FF8101) contains the following objects and may have a maximum length of 894 bytes (66 TL, 828 V);

| Data Field | Tag | Length |
|---------------------------------|--------|--------|
| Amount, Authorized (Numeric) | 9F02 | 6 |
| Amount, Other (Numeric) | 9F03 | 6 |
| Application PAN | 5A | 16 |
| Application PAN Sequence Number | 5F34 | 1 |
| Balance Read Before Gen AC | DF8104 | 6 |
| CDOL1 Related Data | DF8107 | 252 |
| CVM Results | 9F34 | 3 |
| DRDOL Related Data | DF8113 | 252 |
| IDS Status | DF8128 | 1 |
| Interface Device Serial Number | 9F1E | 8 |
| PDOL Related Data | DF8111 | 252 |
| Reference Control Number | DF8114 | 1 |
| Host Capabilities | 9F33 | 3 |
| Host Country Code | 9F1A | 2 |

| | | |
|---------------------------|------|---|
| Host Type | 9F35 | 1 |
| Host Verification Results | 95 | 5 |
| Transaction Category Code | 9F53 | 1 |
| Transaction Currency Code | 5F2A | 2 |
| Transaction Date | 9A | 3 |
| Transaction Type | 9F21 | 3 |
| Unpredictable Number | 9F37 | 4 |

Discretionary data for a MagStripe transaction may include the following objects, with a possible maximum length of 117 bytes (15 TL, 102 V):

| FF8106 | var | Discretionary Data | | |
|--------|-----|--------------------|----|-------------------|
| | | DF812A | 56 | DD Card (Track 1) |
| | | DF812B | 8 | DD Card (Track 2) |
| | | DF8115 | 6 | Error Indication |
| | | 9F6E | 32 | Third Party Data |

The most typical Intermediate OUT Signal (for cases such as Error – Other Card and Try Again) are only required to include the outcome parameter set and the error indication and the L2 tests usually focus on verifying data within these objects.

| Tag | Len | Data Object | | |
|--------|-----|-----------------------|---|------------------|
| DF8129 | 8 | Outcome Parameter Set | | |
| FF8106 | 10 | Discretionary Data | | |
| | | DF8115 | 6 | Error Indication |

7.5.2 Activate Transaction Response Frame Data Format

For additional information on how data portions are encrypted, see 80000502-001, *ID Tech Encrypted Data Output Formats*, available for download on the [ID TECH Knowledge Base](#).

Mode/Transaction Output Format Matrix—Non-SRED Reader Contactless transaction

Note: When MSR/MSD or EMV encryption is enabled and a data encryption key exists, Burst Mode is disabled (always OFF).

When encryption is enabled and a data encryption key exists, **Encrypt Mode** is ON. Otherwise (encryption is disabled or a data encryption key is absent), **Encrypt Mode** is OFF.

| Poll Mode | Encrypt Mode | Burst Mode | Command | Output Format |
|----------------|--------------|------------|---------|-----------------|
| Poll On Demand | OFF | NA | 02-01 | 02-01 plaintext |
| | | NA | 02-40 | 02-40 plaintext |
| | ON | NA | 02-01 | Not Allow |
| | | | 02-40 | 02-40 encrypted |
| Auto Poll | OFF | ON | No Cmd | Burst Format |
| | | | 02-01 | Burst Format |
| | | | 02-40 | Burst Format |
| | | AutoExit | No Cmd | Burst Format |
| | | | 02-01 | 02-01 plaintext |
| | | | 02-40 | 02-40 plaintext |
| | | OFF | 03-00 | 02-01 plaintext |
| | | | 02-01 | 02-01 plaintext |
| | | | 03-40 | 02-40 plaintext |
| | | | 02-40 | 02-40 plaintext |
| | | | 02-40 | 02-40 plaintext |
| | | OFF | 03-00 | Not Allow |
| | | | 02-01 | Not Allow |
| | | | 03-40 | 02-40 encrypted |
| | | | 02-40 | 02-40 encrypted |

Mode/Transaction Output Format Matrix—SRED Reader Contactless transaction

Note: For SRED readers, encryption is always enabled and Burst Mode is disabled (always OFF).

| Poll Mode | Encrypt Mode | Burst Mode | Command | Output Format |
|----------------|--------------|------------|---------|-----------------|
| Poll On Demand | ON | OFF | 02-01 | Not Allow |
| | | | 02-40 | 02-40 encrypted |
| Auto Poll | ON | ON | 03-00 | Not Allow |
| | | | 02-01 | Not Allow |
| | | | 03-40 | 02-40 encrypted |
| | | | 02-40 | 02-40 encrypted |

Success Transaction--Plaintext data field format(02-01)

| Data Item | Length (bytes) | Description |
|-----------------------------------|----------------|--|
| Track 1 Length | 1 | If Track 1 is available, then this field gives the length of the Track 1 data that follows. If Track 1 is not available, then a Length of 00h is returned. Format: Binary |
| Track 1 Data (MagStripe Card) | Variable | Track 1 Data (if available). Format: ASCII (no null terminator) |
| Track 2 Length | 1 | If Track 2 is available, then this field gives the length of the Track 2 data that follows. If Track 2 is not available, then a Length of 00h is returned. Format: Binary |
| Track 2 Data (MagStripe Card) | Variable | Track 2 Data (if available). Format: ASCII (no null terminator) |
| Track 3 Length | 1 | If Track 3 is available, then this field gives the length of the Track 3 data that follows. If Track 3 is not available, then a Length of 00h is returned. Format: Binary |
| Track 3 Data (MagStripe Card) | Variable | Track 3 Data (if available). Format: ASCII (no null terminator) |
| DE055 (Clearing Record) Present.) | 1 | If a Clearing Record (DE 055) field is available, then this field is 01h. If there is no Clearing Record (DE 055) field, then this field is 00h. For MasterCard transactions, this field is always 0. |
| TLV DE 055 (Clearing Record) | Variable | DE 055 data (if available) as a TLV data object encoded with Tag 'E1'. The DE 055 data is the same data as is included in the Clearing Record. Refer to the Activate Transaction Clearing Record table. Tag: E1 Format: Variable (binary), group tag. |
| TLV Data | Variable | See Activate Response TLVs Not all of the tags will be present. |

Successful Transaction: Plaintext and Encrypted data field format for Contactless transaction (02-40)

| Data Item | Length (bytes) | Description |
|-----------------------------------|----------------|---|
| Attribution | 1 | Bit 4/3/0: Captured Data Type 0 0 0 = Contact Transaction 0 0 1 = Contactless Transaction / EMV 1 0 1 = Contactless Transaction / MSD 0 1 x = MSR Card Bit 2/1: Encryption Mode 0 0 = TDES 0 1 = AES 1 x = Reserved Bit 5: Reserved for Attribution Byte Extension. Bit 6/7: Encryption Status 0 0 = MSR/MSD off, EMV off 0 1 = MSR/MSD off, EMV on 1 0 = MSR/MSD on, EMV off 1 1 = MSR/MSD on, EMV on |
| TLV KSN (DFEE12) | Variable | KSN of DUKPT Account Key Tag: DFEE12 (was FFEE12) Format: Binary This Tag only appears when the transaction response contains encrypted data. |
| TLV Track 1 Data (MagStripe Card) | Variable | TDES/AES Encrypted Track 1 Data (if available) with Padding (0x00). If Track 1 is not available, this field is not present. Tag: DFEF17 Format: ASCII (no null terminator) |
| TLV Track 2 Data (MagStripe Card) | Variable | TDES/AES Encrypted Track 2 Data (if available) with Padding (0x00). If Track 2 is not available, this field is not present Tag: DFEF18 Format: ASCII (no null terminator) |
| TLV DE 055 (Clearing Record) | Variable | DE 055 data (if available) as a TLV data object encoded with Tag 'E1'. The DE 055 data is the same data as is included in the Clearing Record. Refer to the Activate Transaction Clearing Record table. Sensitive TLV will be TDES/AES encrypted with Padding (0x00) See "80000502-001, ID Tech Encrypted Data Output Formats" for details; available for download on the ID TECH Knowledge Base . Tag: E1 Format: Variable length. BER-TLV group tag. |
| TLV Data | Variable | See Activate Response TLVs Sensitive TLV will be TDES/AES encrypted with Padding (0x00) See "80000502-001, ID Tech Encrypted Data Output Formats" for details; available for download on the ID TECH Knowledge Base . Not all of the tags will be present. |

Success Transaction -- Plaintext and Encrypted data field format for MSR card (02-40)

| Data Item | Length (bytes) | Description | | | | | | | | | | | | | | | | |
|-------------------------------|----------------|---|-----------------------|--|----------------------------|------------------|-----------------|-------|---|---|-------------------------------|-------|----|---|------------------|-----|----|---|
| Attribution | 1 | Bit 4/3/0: Captured Data Type 0 0 0 = Contact Transaction 0 0 1 = Contactless Transaction / EMV 1 0 1 = Contactless Transaction / MSD 0 1 x = MSR Card Bit 2/1: Encryption Mode 0 0 = TDES 0 1 = AES Bit 6/5: Reserved Bit 7: Encryption Status 0 = Encryption OFF 1 = Encryption ON | | | | | | | | | | | | | | | | |
| MSR TLV | Variable | <div>MSR TLV Data length compose of data length indicator (1 byte) and actual data length byte.</div> <table><tr><th colspan="2">MSR FIELD DATA length</th><th>Data length Indicator byte</th><th>Data length byte</th></tr><tr><td>Data <128 bytes</td><td>01~7F</td><td>X</td><td>1</td></tr><tr><td>128 bytes <= Data <=255 bytes</td><td>80~FF</td><td>81</td><td>1</td></tr><tr><td>Data > 255 bytes</td><td>FF~</td><td>82</td><td>2</td></tr></table> <div>Data is follow Enhanced Encrypted MSR FIELD DATA. See “80000502-001, ID Tech Encrypted Data Output Formats” for details; available for download on the ID TECH Knowledge Base. Tag: DFEE23 Format: Binary</div> | MSR FIELD DATA length | | Data length Indicator byte | Data length byte | Data <128 bytes | 01~7F | X | 1 | 128 bytes <= Data <=255 bytes | 80~FF | 81 | 1 | Data > 255 bytes | FF~ | 82 | 2 |
| MSR FIELD DATA length | | Data length Indicator byte | Data length byte | | | | | | | | | | | | | | | |
| Data <128 bytes | 01~7F | X | 1 | | | | | | | | | | | | | | | |
| 128 bytes <= Data <=255 bytes | 80~FF | 81 | 1 | | | | | | | | | | | | | | | |
| Data > 255 bytes | FF~ | 82 | 2 | | | | | | | | | | | | | | | |
| TLV Data | Variable | See Activate Response TLVs | | | | | | | | | | | | | | | | |

Failed Transaction: Plaintext data field format (02-01)

| Data Item | Length (bytes) | Description |
|---------------|----------------|---|
| Error Code | 1 | Error Code giving the reason for the failure. See sub-section on Error Codes |
| SW1 | 1 | Value of SW1 returned by the Card (SW1SW2 is 0000 if SW1 SW2 not available) |
| SW2 | 1 | Value of SW2 returned by the Card (SW1SW2 is 0000 if SW1 SW2 not available) |
| RF State Code | 1 | RF State Code indicating exactly where the error occurred in the Reader-Card transaction flow. See RF State Codes . |
| TLV data | Varies | Refer to the Activate Response TLVs . |

Failed Transaction: Plaintext and Encrypted data field format (02-40)

| Data Item | Length (bytes) | Description |
|---|----------------|--|
| Attribution | 1 | Bit 4/3/0: Captured Data Type 0 0 0 = Contact Transaction 0 0 1 = Contactless Transaction / EMV 1 0 1 = Contactless Transaction / MSD 0 1 x = MSR Card Bit 2/1: Encryption Mode 0 0 = TDES 0 1 = AES 1 x = Reserved Bit 5: Reserved for Attribution Byte Extension. Bit 6/7: Encryption Status 0 0 = MSR/MSD off, EMV off 0 1 = MSR/MSD off, EMV on 1 0 = MSR/MSD on, EMV off 1 1 = MSR/MSD on, EMV on |
| TLV KSN (DFEE12) | Variable | KSN of DUKPT Account Key Tag: DFEE12 Format: Binary, 10 bytes This Tag only appears when the transaction response contains encrypted data. |
| TLV Contactless Error Code Table (DFEE02) | Variable | Byte 1: Error Code. (Error Code giving the reason for the failure.) Byte 2: SW1 (Value of SW1 returned by the Card (SW1SW2 is 0000 if SW1 SW2 not available)) Byte 3: SW2 (Value of SW2 returned by the Card (SW1SW2 is 0000 if SW1 SW2 not available)) Byte 4: RF State Code (RF State Code indicating exactly where the error occurred in the Reader-Card transaction flow.) |
| TLV Data | Variable | Refer to the Activate Response TLVs. Sensitive TLV will be TDES/AES encrypted with Padding (0x00) See "80000502-001, ID Tech Encrypted Data Output Formats" for details; available for download on the ID TECH Knowledge Base . |

7.5.3 Get Transaction Result (03-00 and 03-40)

NOTE: 03-00 is a legacy command, applicable to non-encrypted transactions only. When EMV encryption is ON or MSR/MSD encryption is ON, 03-00 will be disabled if a data encryption key is loaded. **03-40** is a unified command for both non-encrypted and encrypted transactions.

Use this command when the reader is functioning in Auto Poll Mode. In this mode the reader does not wait for an **Activate Transaction** command to start polling for a card—it is always in Auto Poll Mode. When the reader detects a card, it carries out a transaction. If the card is a supported contactless MagStripe card, the reader does not need any parameters from the host device. If the card is a supported contactless EMV Card, then the reader uses the default host parameters (Group 0 TLVs) in the reader. If the reader has any host parameters set by the **Set Configuration** command, it uses the new values for these parameters.

If the transaction is successful, the reader keeps the transaction data (Track or clearing record) in its memory. When it receives the **Get Transaction Result** command, it returns this data to the host device immediately and reset its data buffer. If the reader has not detected any card since power up or since the last **Get Transaction Result** command and this command is received, the reader responds back immediately indicating that it has no data for the host.

In Auto Poll Mode the reader can carry out only contactless MagStripe and contactless EMV transactions. It cannot carry out any ticketing or ePurse transactions because they require interaction with the host device during the transaction.

Note: For Non-SRED readers, the response format for **Activate Transaction** follows **Set Data Encryption Enable Flag (C7-36)** settings and the data encryption key.

When data encryption is disabled, the reader responds with plaintext data format. When data encryption is enabled:

1. When a data encryption key exists and is valid, the reader responds with the encrypted data format.
2. When a data encryption key does not exist, the reader responds with the plaintext data format
3. When a data encryption key exhausts, the reader responds status code 0x91 and no data.

For SRED readers, the response format for the **Activate Transaction** command is set according to the data encryption key. When a data encryption key exists and is valid, the reader responds with encrypted data format. When a data encryption key exhausts or does not exist, the reader responds with status code 0x91/0x90 and no data.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 03h | 00h | 00h | 00h | | |

On receiving this command, the reader returns one of the following:

- A response containing track data (Contactless MagStripe transaction)
- A response containing a clearing record (Contactless EMV transaction)
- A response containing no data (no transaction)

If the transaction cannot be completed successfully, the response indicates an OK status and "No Data."

Note: ViVOcomm and DesFire cards return raw track data only.

If there was an error in the command frame received then the response contains an appropriate status code.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 03h | See status code table | | | See Data Tables | | |

If the status code is OK, then the format and contents of the data field in the response frame are listed in the following table. Some data objects may not be present depending on the card involved in the transaction and the presence or absence of a clearing record object (DE 055). All TLV lengths include the Tag and Length bytes.

Get Transaction Result Format and Content: Plaintext data field format (03-00)

| Data Item | Length (bytes) | Description |
|-------------------------------|----------------|---|
| Track 1 Length | 1 | If Track 1 is available, then this field gives the length of the Track 1 data that follows. If Track 1 is not available, then a Length of 00h is returned. Format: Binary |
| Track 1 Data (MagStripe card) | Variable | Track 1 Data (if available). Format: ASCII (no null terminator) |

| Data Item | Length (bytes) | Description |
|---|----------------|--|
| Track 2 Length | 1 | If Track 2 is available, then this field gives the length of the Track 2 data that follows. If Track 2 is not available, then a Length of 00h is returned. Format: Binary |
| Track 2 Data (MagStripe card) | Variable | Track 2 Data (if available). Format: ASCII (no null terminator) |
| Track 3 Length | 1 | If Track 3 is available, then this field gives the length of the Track 3 data that follows. If Track 3 is not available, then a Length of 00h is returned. Format: Binary |
| Track 3 Data (MagStripe Card) | Variable | Track 3 Data (if available). Format: ASCII (no null terminator) |
| DE055 (Clearing Record) Present | 1 | If a Clearing Record (DE 055) field is available, then this field is 01h. If there is no Clearing Record (DE 055) field, then this field is 00h. |
| TLV DE 055 (Clearing Record) (see Clearing Record Format) | Variable | DE 055 data (if available) as a TLV data object encoded with Tag 'E1'. The DE 055 data is the same data as is included in the Activate Transaction Clearing Record. Refer to the Activate Transaction Clearing Record table. Tag: E1 Format: Variable length group tag. |
| TLV Data | Variable | Refer to Activate Response TLVs. |

If the status code is OK the response is different depending on the card application:

| Card Application | Return Data |
|-------------------------|--|
| PayPass MagStripe | Track1/Track2 |
| PayPass M/Chip | Chip data plus other tags |
| JCB QuicPay | TLV Auth code and Track2 Equivalent data |
| VSDC online application | Track1/Track2 and VLP Issuer Auth code |
| VSDC offline and qVSDC | Chip data E1 and some other tags |

Some cards (qVSDC & M/Chip) can return a status code or 23 (request online authorization) with a fully populated data field.

This command never returns a status code of "Failed." If any status code other than OK or 23 (request online authorization) is returned, the data field is empty.

The above description is a plaintext response. The encrypted data format is as follows below; see 80000502-001, *ID Tech Encrypted Data Output Formats* for details; available for download on the [ID TECH Knowledge Base](#).

Get Transaction Result format for contactless transaction: Plaintext and Encrypted data field format (03-40)

| Data Item | Length (bytes) | Description |
|---|----------------|--|
| Attribution | 1 | Bit 4/3/0: Captured Data Type 0 0 0 = Contact Transaction 0 0 1 = Contactless Transaction / EMV 1 0 1 = Contactless Transaction / MSD 0 1 x = MSR Card Bit 2/1: Encryption Mode 0 0 = TDES 0 1 = AES 1 x = Reserved Bit 5: Reserved for Attribution Byte Extension. Bit 6/7: Encryption Status 0 0 = MSR/MSD off, EMV off 0 1 = MSR/MSD off, EMV on 1 0 = MSR/MSD on, EMV off 1 1 = MSR/MSD on, EMV on |
| TLV KSN | 10 | This Tag only appears when the transaction response contains encrypted data. KSN of DUKPT Account Key Tag: DFEE12 Format: Binary |
| TLV Track 1 Data (MagStripe card) | Variable | TDES/AES Encrypted Track 1 Data (if available) with Padding (0x00). If Track 1 is not available, this field is not present. Tag: DFEF17 Format: ASCII (no null terminator) |
| TLV Track 2 Data (MagStripe card) | Variable | TDES/AES Encrypted Track 2 Data (if available) with Padding (0x00). If Track 2 is not available, this field is not present Tag: DFEF18 Format: ASCII (no null terminator) |
| TLV DE 055 (Clearing Record) (see Clearing Record Format) | Variable | DE 055 data (if available) as a TLV data object encoded with Tag 'E1'. The DE 055 data is the same data as is included in the Clearing Record. Refer to the Activate Transaction Clearing Record table. Sensitive TLV will be TDES/AES encrypted with Padding (0x00) See "80000502-001, ID Tech Encrypted Data Output Formats" for details; available for download on the ID TECH Knowledge Base . Tag: E1 Format: Variable (group tag) |
| TLV Data | Variable | See Activate Response TLVs Sensitive TLVs will be TDES/AES encrypted with Padding (0x00). See "80000502-001, ID Tech Encrypted Data Output Formats" for details; available for download on the ID TECH Knowledge Base . |

Get Transaction Result format for MSR card: Encrypted

| Data Item | Length (bytes) | Description | | | | | | | | | | | | | | | | |
|--|----------------|--|--|--|----------------------------|------------------|-----------------|-------|---|---|-------------------------------|-------|----|---|------------------|-----|----|---|
| Attribution | 1 | Bit 4/3/0: Captured Data Type 0 0 0 = Contact Transaction 0 0 1 = Contactless Transaction / EMV 1 0 1 = Contactless Transaction / MSD 0 1 x = MSR Card Bit 2/1: Encryption Mode 0 0 = TDES 0 1 = AES 1 x = Reserved Bit 5: Reserved for Attribution Byte Extension. Bit 6/7: Encryption Status 0 0 = MSR/MSD off, EMV off 0 1 = MSR/MSD off, EMV on 1 0 = MSR/MSD on, EMV off 1 1 = MSR/MSD on, EMV on | | | | | | | | | | | | | | | | |
| TLV (DFEE25) | Variable | Response Code TLV Tag: DFEE25 Format: Binary Response: 00h 11h | | | | | | | | | | | | | | | | |
| TLV MSR Data (DFEE23) | Variable | Refer to “Enhanced Encrypted MSR Data Output Format” section of document 80000502-001, <i>Encrypted Data Output Formats</i> . This tag will wrap an entire MSR data block. Consult “80000502-001, ID Tech Encrypted Data Output Formats” for details; available for download on the ID TECH Knowledge Base . <table><tr><th colspan="2">Enhanced encrypted MSR FIELD DATA length</th><th>Data length Indicator byte</th><th>Data length byte</th></tr><tr><td>Data <128 bytes</td><td>01~7F</td><td>X</td><td>1</td></tr><tr><td>128 bytes <= Data <=255 bytes</td><td>80~FF</td><td>81</td><td>1</td></tr><tr><td>Data > 255 bytes</td><td>FF~</td><td>82</td><td>2</td></tr></table> | Enhanced encrypted MSR FIELD DATA length | | Data length Indicator byte | Data length byte | Data <128 bytes | 01~7F | X | 1 | 128 bytes <= Data <=255 bytes | 80~FF | 81 | 1 | Data > 255 bytes | FF~ | 82 | 2 |
| Enhanced encrypted MSR FIELD DATA length | | Data length Indicator byte | Data length byte | | | | | | | | | | | | | | | |
| Data <128 bytes | 01~7F | X | 1 | | | | | | | | | | | | | | | |
| 128 bytes <= Data <=255 bytes | 80~FF | 81 | 1 | | | | | | | | | | | | | | | |
| Data > 255 bytes | FF~ | 82 | 2 | | | | | | | | | | | | | | | |
| TLV Data | Variable | TLV data list | | | | | | | | | | | | | | | | |
| TLV POS Entry Mode (9F39) | Variable | 90h = Magnetic Stripe Reader Swipe 91h = Contactless MSD 05h = Contact EMV 07h = Contactless EMV 80h = Contact Fallback to Magnetic Stripe | | | | | | | | | | | | | | | | |
| TLV (DF30) (ViVOpay proprietary) | Variable | Track Data Source. This tag is embedded in the ViVOpay Group tag. It specifies whether the track data came from a swipe or RFID transaction. 0Ch for swiped MagStripe 00h for a contactless transaction. 01h for a contact transaction. This tag include in ViVOpay proprietary (FFEE01) Ex: FFEE0104DF30010C | | | | | | | | | | | | | | | | |

| Data Item | Length (bytes) | Description |
|----------------------------------|----------------|---|
| TLV Encrypt Information (DFEE26) | Variable | <p>Encryption and Data Capture Information. Length: 1 or 2 (variable)</p> <p>The length must be 2 bytes for NEO 2 readers.</p> <p>Values:</p> <p>Byte 1: Same as "Attribution" byte definition. Namely:</p> <p>Bits 4/3/0: Captured Data Type 0 0 0 = Contact Transaction 0 0 1 = Contactless Transaction / EMV 1 0 1 = Contactless Transaction / MSD 0 1 x = MSR Card</p> <p>Bits 2/1: Encryption Mode 0 0 = TDES 0 1 = AES 1 x = Refer to "Extended Encryption Mode" further below.</p> <p>Bit 5: Attribution Byte Extension. 0 = Length of tag is 1 byte 1 = Length of tag is 2 bytes</p> <p>Bits 6/7: Encryption Status 0 0 = EMV off, MSR/MSD off 0 1 = EMV off, MSR/MSD on 1 0 = EMV on, MSR/MSD off 1 1 = EMV on, MSR/MSD on</p> <p>Byte 2: (When applicable; that is, when Length of TLV is 2)</p> <p>Bits 3/2/1/0: Extended Encryption Mode 0 0 0 0 = TDES 0 0 0 1 = AES 0 0 1 0 = TransArmor Algorithm 0 0 1 1 = Voltage Algorithm 0 1 0 0 = Visa FPE 0 1 0 1 = Verifone FPE 0 1 1 0 = TransArmor TDES-DUKPT</p> <p>Bits 4 to 6: Reserved</p> <p>Bit 7: 0 = No MAC Verification Data 1 = Has MAC Verification Data</p> |

7.5.4 Update Balance Command (03-03)

Update Balance Command is the authorization response sent by the issuer to the host device including the authorization status (OK or NOT OK). Send this command when the ViVOpay reader is in "Poll on Demand" mode and after the reader sends an online request to the issuer.

This command is also used in some implementations (for example, in EMEA) to communicate the results of issuer authentication to the reader in order to display the correct LCD messages.

With this command, the POS passes the authorization result (OK/NOT OK), and possibly the Authorization Code (Auth_Code)/Date/Time to the host device.

For a Visa transaction when the card supports Available Offline Spending Amount, the LCD displays the available amount.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 03h | 03h | | | See Data Table | | |

The format and contents of the data field in the command frame are given in the following table. All TLV lengths include the Tag and Length bytes.

Update Balance Format and Contents

| Data Item | Length (bytes) | Description |
|----------------------|-------------------|---|
| Status Code | 1 | 00: OK 01: NOT OK 02: (ARC response 89 for Interac) |
| TLV Auth_Code | 9 | Authorization Code as a TLV object. Tag: E300 Format: b8 |
| TLV Transaction Date | 5 | EMV data element "Transaction Date" as a TLV data object. Local date that the transaction was authorized. If this TLV is not provided, the transaction uses the reader's current date. Tag: 9A Format: n6 (YYMMDD) Note: The reader does not perform range checking on this value. The POS application should perform range checking on this value to ensure it is within acceptable limits. |
| TLV Transaction Time | 6 | EMV data element "Transaction Time" as a TLV data object. Local time that the transaction was authorized. If this TLV is not provided, the transaction uses the reader's current time. Tag: 9F21 Format: n6 (HHMMSS) |

| | | |
|--|--|--|
| | | Note: The reader does not perform range checking on this value. The POS application should perform range checking on this value to ensure it is within acceptable limits. |
|--|--|--|

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 03h | See status code table | | | See Data Tables | | |

If the status code is OK, the format and contents of the data field in the response frame are given in the following table. All TLV lengths include the Tag and Length bytes.

Update Balance Format and Contents When Status OK

| Data Item | Length (Bytes) | Description |
|--------------------------------|----------------|---|
| TLV Track 2 Equivalent Data | 21 | Track 2 Equivalent Data as a TLV object. Tag: 57 Format: b19 |
| TLV Auth_Code | 9 | Authorization Code as a TLV object Tag: E300 Format: b8 |

If the status code being returned in the response frame is "Failed," the data field contains further information on the cause of the failure and does not contain the authorization code and other information. In this case the data field in the response frame has the following format.

Update Balance Format and Contents When Status Not OK

| Data Field | Length (bytes) | Description |
|---------------|----------------|--|
| Error Code | 1 | Error Code giving the reason for the failure. See sub-section on Error Codes |
| SW1 | 1 | Value of SW1 returned by the Card (SW1SW2 is 0000 if SW1 SW2 not available) |
| SW2 | 1 | Value of SW2 returned by the Card (SW1SW2 is 0000 if SW1 SW2 not available) |
| RF State Code | 1 | RF State Code indicating exactly where the error occurred in the Reader-Card transaction flow. See sub-section on RF State Codes . |

The data field is empty for any other Status Code.

7.5.5 Cancel Transaction Command/Reset Transaction Status (05-01)

The **Cancel Transaction Command/Reset Transaction Status** command stops reader or card communication after an [Activate Transaction](#) or [Update Balance](#) command has been sent to the reader.

After the host device has issued the **Cancel Transaction** command, the host should not send any commands until it receives a response from the reader.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | N/A Byte 14 | Byte 14 Byte 15 | Byte 15 Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------|--------------------|--------------------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data Object | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 05h | 01h | 00h | 00h or 01h | | | |

For cancelling a transaction, the data length (LSB) is 00h; a data object is not required.

For resetting a transaction, the data length (LSB) is 01h; the following data object is required:

Data Object:

| Data Item | Length (bytes) | Description |
|-----------|----------------|--|
| Mode | 1 | -0x01: Reset Mode, reset transaction status without displaying anything; |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 05h | See status code table | 00h | 00h | | |

7.6 MasterCard M/Chip 3.0 Transaction Commands

This section describes commands specific to MasterCard M/Chip 3.0 transaction behavior.

7.6.1 Stop Transaction (05-02)

The **Stop Transaction** command is similar to the **Cancel** command. However, the transaction will exit regardless of its present phase. Depending on timing, the transaction could exit with an **Activate Response**. In that case, the **Stop Transaction** command was received too late to stop the transaction. Receipt of any response other than the **Stop Transaction** response is proof that the **Stop Transaction** command did not execute.

Only the MasterCard M/Chip 3.0 application uses the **Stop Transaction** command.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 05h | 02h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 05h | See status code table | 00h | 30h | See below | | |

A successful **Stop Transaction** command returns the following information in the data field:

| Data item | Tag | Tag Len | Notes |
|---------------------------|---------|---------|----------------------------|
| Outcome Parameter Set TLV | DF8129h | 12 | |
| Discretionary Data TLV | FF8106h | 10 | Encapsulates the Error TLV |
| Error Indication TLV | DF8115 | 6 | |

7.6.2 Reset Torn Transaction Log (84-0E)

The **Reset Torn Transaction Log** command effectively erases the content of the torn transaction log and sets it back to an “empty” state.

Normally, this function is only used in certification scenarios where the torn transaction log must be put into a known state before performing a test.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViV0tech2\0 | 84h | 0Eh | 00h | 00h | Varies | Varies |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | 84h | Refer to standard status values | 00h | 00h | Varies | Varies |

This command restores the Torn Transaction Log back to its original pristine state as if a power up had just occurred.

7.6.3 Clean Torn Transaction Log (84-0F) Command

The **Clean Torn Transaction Log** command removes Torn Transaction Log entries that exceed the allowed lifetime defined in tag DF811C (Maximum Lifetime of Torn Transaction Log Record).

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViV0tech2\0 | 84h | 0Fh | 00h | 00h | Varies | Varies |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-N | Byte N + 1 | Byte N + 2 |
|-------------------------------|---------|---------------------------------|-------------------|-------------------|--------------------|------------|------------|
| Header Tag & Protocol Version | Command | Status | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 84h | Refer to standard status values | XX | XX | List of Torn TLV's | Varies | Varies |

The response may contain expiring torn entries. These are returned inside a discretionary data tag, as shown below:

| Byte 0-2 | Byte 3 | Bytes 4-6 | Byte 7 | Byte 8-N |
|--------------------|------------|------------------|------------|-----------------------|
| Discretionary Data | Tag Length | Torn Transaction | Tag Length | TLV's for Torn Record |
| FF8106h | Varies | FF8101h | Varies | Varies |

NOTE: The host device should execute the CLEAN command repeatedly, until it receives no more torn records. In other words, in the final response, the value length of tag FF8106 shall be 0.

Refer to the M/Chip PayPass specification for the contents of the Torn Transaction Log Record.

1.1.2.2. Torn Transaction Log Timer

Readers keep track of how much time has elapsed for each torn transaction record. However, it *does not* take any action when this time has expired. Because the interface between the host and the reader is a command/response interface, the host must initiate cleaning the torn transaction log.

Periodically, the POS should initiate a cleaning cycle and repeatedly issue the **Clean Torn Transaction Log (84-0F)** command at that point until the reader reports that the Torn Transaction Log has been successfully purged.

How the POS accomplishes this is beyond the scope of the interface and this document.

7.6.4 Data Exchange Request (02-58) and Data Exchange Response (02-09)

The **Data Exchange Request** and **Data Exchange Response** commands exchange data for MasterCard (MCL).

The reader sends the **Data Exchange Request** command to the host device, and, conversely, the host sends the **Data Exchange Response** command to the reader. This process is different from other commands. It only applies to certain MasterCard scenarios.

Request Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-N | Byte N + 1 | Byte N + 2 |
|-------------------------------|---------|---------|-------------------|-------------------|---|------------|------------|
| Header Tag & Protocol Version | Command | Status | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 02h | 58h | XX | XX | 0Ah followed by TLV FF8104 and TLV DF8106 | Varies | Varies |

Example:

```
56 69 56 4F 74 65 63 68 32 00 02 58 xx xx 0A FF 81 04 xx ... DF 81
06 xx ... CRC
```

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-N | Byte N + 1 | Byte N + 2 |
|-------------------------------|---------|---------|-------------------|-------------------|------------|------------|------------|
| Header Tag & Protocol Version | Command | Status | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 02h | 09h | XX | XX | TLV FFEE1A | Varies | Varies |

Example:

```
56 69 56 4F 74 65 63 68 32 00 02 09 xx xx FF EE 1A xx ... CRC
```

7.7 Key Management Commands

Warning: DO NOT mix Protocol 1 (now deprecated) and Protocol 2 Key Management commands. The preferred method is to use the Protocol 2 commands.

The Key Management Protocol 2 commands are the preferred method. The Key Management Protocol 2 commands **MUST** be used when doing secure communication.

The following status codes may be generated in response to the CA Public Key commands.

The following status codes are specific to the Key Manager module. Their values may have different meanings when used with other commands.

EMV Key Manager Status Codes – Protocol 2

| Status Code | Status |
|-------------|---|
| 00h | Operation was successful |
| 20h | SAM Transceiver error – problem communicating with the SAM (see note below) |
| 21h | Length error in data returned from the SAM |
| 41h | Unknown Error from SAM |
| 42h | Invalid data detected by SAM |
| 43h | Incomplete data detected by SAM |
| 44h | Reserved |
| 45h | Invalid key hash algorithm |
| 46h | Invalid key encryption algorithm |
| 47h | Invalid modulus length |
| 48h | Invalid exponent |
| 49h | Key already exists |
| 4Ah | No space for new RID |
| 4Bh | Key not found |
| 4Ch | Crypto not responding |
| 4Dh | Crypto communication error |
| 4Fh | All key slots are full (maximum number of keys has been installed) |

7.7.1 Get CA Public Key (D0-01)

The **Get CA Public Key** command retrieves all information related to a specific key. It includes the key hash, the algorithms, and so forth; see the data definition below.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 – 18 | Byte 19 | Byte 20 | Byte 21 |
|-------------------------------|---------|---------|--------------|--------------|--------------|-----------|-----------|-----------|
| Header Tag & Protocol Version | Cmd | Sub Cmd | Length (MSB) | Length (LSB) | RID | Key Index | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | D0h | 01h | 00h | 06h | varies | varies | Varies | Varies |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 19 – n | Byte n+1 | Byte n+2 |
|-------------------------------|---------|--|--------------|--------------|-------------|-----------|-----------|
| Header Tag & Protocol Version | Cmd | Status | Length (MSB) | Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | D0h | See Key Manager status codes | 00h | varies | varies | Varies | Varies |

When the status is successful (00h), the data field contains:

- **Key Hash Algorithm** (1 Byte): 01h = SHA-1
- **Key Encryption Algorithm** (1 Byte): 01h = RSA
- **Checksum**: This Checksum is calculated with a concatenation of:
- **RID & KeyIndex & Modulus & Exponent**
 - where the exponent is either one byte or 3 bytes
- **Modulus Length** (2 bytes)
- **Modulus** (varies in length)

7.7.2 Get CA Public Key Hash (D0-02)

The **Get CA Public Key Hash** command returns only the "Checksum" portion of the key.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 – 18 | Byte 19 | Byte 20 | Byte 21 |
|-------------------------------|---------|---------|--------------|--------------|--------------|-----------|-----------|-----------|
| Header Tag & Protocol Version | Cmd | Sub Cmd | Length (MSB) | Length (LSB) | RID | Key Index | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | D0h | 02h | 00h | 06h | varies | varies | Varies | Varies |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 19 – n | Byte n+1 | Byte n+2 |
|-------------------------------|---------|--|--------------|--------------|-------------|-----------|-----------|
| Header Tag & Protocol Version | Cmd | status | Length (MSB) | Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | D0h | See Key Manager status codes | 00h | varies | varies | Varies | Varies |

Status = 00 if successful. When the status is successful, the data contains:

- Key Hash Algorithm (1 byte)
- Key Algorithm (1 byte)
- **Checksum** (20 bytes): calculated over the key information as previously described

7.7.3 Set CA Public Key (D0-03)

The **Set CA Public Key** command adds a new key to the reader.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 – 18 | Byte 19 | Bytes 19-n | Byte n+1 | Byte n+2 |
|-------------------------------|---------|---------|--------------|--------------|---------------|--------------------|------------|-----------|-----------|
| Header Tag & Protocol Version | Cmd | Sub Cmd | Length (MSB) | Length (LSB) | RID (5 bytes) | Key Index (1 byte) | Key Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | D0h | 03h | varies | varies | varies | varies | See below | Varies | Varies |

Key Data is as follows: (all binary)

| Byte | Name | Length (bytes) | Description |
|-------|----------------------|----------------|--|
| 0 | Hash Algorithm | 1 | The only algorithm supported is SHA-1. The value is set to 01h |
| 1 | Public Key Algorithm | 1 | The encryption algorithm in which this key is used. Currently support only one type: RSA. The value is set to 01h |
| 3-22 | Checksum/Hash | 20 | Checksum which is calculated using SHA-1 over the following fields: RID & KeyIndex & Modulus & Exponent where the exponent is either one byte or 3 bytes (although we store it in a 4 byte field) |
| 23-26 | Public Key Exponent | 4 | Actually, the real length of the exponent is either one byte or 3 bytes. It can have two values: 3, or 65537. |
| 27-28 | Modulus Length | 2 | Indicates the length of the next field, MSB followed by LSB. |
| 29-n | Modulus | Variable | This is the modulus field of the public key. Its length is specified in the field above. |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|--|--------------|--------------|------------|------------|
| Header Tag & Protocol Version | Cmd | status | Length (MSB) | Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | D0h | See Key Manager status codes | 00h | 00h | Calculated | Calculated |

7.7.4 Delete CA Public Key (D0-04)

The **Delete CA Public Key** command allows the POS to delete a specific key.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 – 18 | Byte 19 | Byte 20 | Byte 21 |
|-------------------------------|---------|---------|--------------|--------------|---------------|--------------------|-----------|-----------|
| Header Tag & Protocol Version | Cmd | Sub Cmd | Length (MSB) | Length (LSB) | RID (5 bytes) | Key Index (1 byte) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | D0h | 04h | 00h | 06h | varies | varies | Varies | Varies |

The RID and Key Index for the key being deleted must be specified in the frame.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|--|--------------|--------------|------------|------------|
| Header Tag & Protocol Version | Cmd | status | Length (MSB) | Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | D0h | See Key Manager status codes | 00h | 00h | Calculated | Calculated |

7.7.5 Delete All CA Public Keys (D0-05)

The **Delete All CA Public Keys** command deletes all CA public keys stored in the reader.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------|--------------|--------------|------------|------------|
| Header Tag & Protocol Version | Cmd | Sub Cmd | Length (MSB) | Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | D0h | 05h | 00h | 00h | Calculated | Calculated |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|--|--------------|--------------|------------|------------|
| Header Tag & Protocol Version | Cmd | status | Length (MSB) | Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | D0h | See Key Manager status codes | 00h | 00h | Calculated | Calculated |

7.7.6 Get All CA Public RIDs (D0-06)

The **Get All CA Public RIDs** command tells the reader to retrieve a list of all the RIDs.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------|--------------|--------------|------------|------------|
| Header Tag & Protocol Version | Cmd | Sub Cmd | Length (MSB) | Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | D0h | 06h | 00h | 00h | Calculated | Calculated |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 13-n | Byte n+1 | Byte n+2 |
|-------------------------------|---------|--|--------------|--------------|----------------------|------------|------------|
| Header Tag & Protocol Version | Cmd | status | Length (MSB) | Length (LSB) | RID(s) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | D0h | See Key Manager status codes | 00h | varies | Each RID is 5 bytes. | Calculated | Calculated |

Status 00h – RIDs returned with the number of RIDs = Length/5;

Note: If the length returned is 0, then the communication was good, but no RIDs are stored.

7.7.7 List CA Public Key IDs or RID (D0-07)

The **List CA Public Key IDs or RID** command retrieves a list of key indices that are installed for this RID.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 – 18 | Byte 19 | Byte 20 |
|-------------------------------|---------|---------|--------------|--------------|---------------|-----------|-----------|
| Header Tag & Protocol Version | Cmd | Sub Cmd | Length (MSB) | Length (LSB) | RID (5 bytes) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | D0h | 07h | 00h | 05h | varies | Varies | Varies |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-18 | Byte 19 – n | Byte n+1 | Byte n+2 |
|-------------------------------|---------|--|--------------|--------------|---------------|-----------------|-----------|-----------|
| Header Tag & Protocol Version | Cmd | status | Length (MSB) | Length (LSB) | RID (5 bytes) | List of Indices | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | D0h | See Key Manager status codes | 00h | varies | varies | varies | Varies | Varies |

Status – 00h = successful index retrieval

7.8 Module Versioning

The module versioning feature provides information about the firmware versions, and the specification versions for specific modules, interfaces, and hardware in the reader. The information is returned to the POS or the POS Simulator via the serial interface. Versioning of card applications may also facilitate the tracking of changes for certification purposes.

The implementation of this feature has been simplified for the following reasons:

- To align more closely with the behavior of the Advanced Reader firmware.
- To make the version strings more accessible for human readers.
- To facilitate maintenance of version strings.

The following subcommands are available for the Module Version command:

| Sub-command | Description |
|-------------|---|
| 02h | Get Processor Type |
| 14h | Get Hardware Information |
| 20h | Get Module Version Information ⁵ |

Note: All other sub-commands for the Module version command have been deprecated. However, a 0x00 in the sub-command field will return the same result as a 20h sub-command. All other commands will return an "unknown sub-command" error.

The table below shows the information that is available and the subcommand used to extract that information. The term "module" is used very loosely in the context of the firmware.

| Module Type | Sub-Command | Description | Format |
|-------------|-------------|--|------------|
| FW | 20h | The firmware version that is resident in the reader. | ASCII text |
| CL AppSel | 20h | Refers to the special application selection module and version. | ASCII text |
| CL AID | 20h | Contactless L2 Application specification/version (because L2 applications are identified by the "application ID," this type refers to an AID). | ASCII text |
| CL AppSpe | 20h | Contactless L2 Special Application specification/version that is not identified by the "application ID" (example: Smart Tap and ApplePay VAS). | ASCII text |
| CL L1 | 20h | L1 Interface specification/version. | ASCII text |
| UI | 20h | User Interface specification/version. | ASCII text |
| SAM | 14h | Secure Access Module version string. | ASCII text |
| HW | 14h | Hardware platform identifier. | ASCII text |
| EEPROM | 14h | The EEPROM version. | ASCII text |
| N/A | 02h | Returns the processor type in TLV format. | TLV |

The module types described above appear in the response packet for the respective sub-command. Refer to the examples in the response packet section.

⁵ Previously a subcommand "0x00" was supported. It is being deprecated. However, because some of the VIVOPay internal utilities used that command to determine if the reader was alive, a subcommand of 0x00 will behave exactly the same as a subcommand 0x20 and will not give an error.

7.8.1 Get Product Type (09-01)

The **Get Product Type** command returns a "product type" value in a proprietary TLV.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-----------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 09h | 01 | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-----------------------|---------|---------------------------------------|-------------------|-------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 09h | See status code table | | | See below | | |

The Get Product Type sub-command returns a TLV string as follows:

- **Tag:** 0xDF60
- **Length:** 0x03
- **Value:** 3-byte field representing the product type.

The following example shows the command and response.

Command: Get Product Type: 56 69 56 4F 74 65 63 68 32 00 09 01 00 00 A0 A0

Response: 56 69 56 4F 74 65 63 68 32 00 09 00 00 06 DF 60 03 43 36 00 DC 60

| Product Type (hex values) | Description |
|------------------------------|-----------------------|
| 43 37 00 | ViVOpay Kiosk1 (ATM1) |
| 43 38 00 | Kiosk2 |
| 56 41 00 | VP3320 |
| 56 42 00 | VP3320S |
| 56 31 00 | VP3600 |
| 56 31 01 | VP3600 CPR |
| 56 32 00 | VP5200 |
| 56 33 00 | VP5300 |
| 56 33 01 | VP5300M |

| | |
|----------|---------------|
| 56 34 00 | VP6300 |
| 56 34 01 | VP6300 USAT |
| 56 34 02 | VP6300 USAT-S |
| 56 35 00 | VP6800 |
| 44 32 00 | NETs |
| 44 33 00 | Magtek |
| 44 35 00 | ICP |

7.8.2 Get Processor Type (09-02)

The **Get Processor Type** command returns a processor type TLV.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-----------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 09h | 02 | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-----------------------|---------|---------------------------------------|-------------------|-------------------|--------------------------|--------------|--------------|
| Header Tag & Protocol | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 09h | See status code table | | | See below | | |

The **Get Processor Type** sub-command returns a TLV string as follows:

- **Tag:** 0xDF61
- **Length:** 0x02
- **Value:** a field representing the processor type.

The following types of processors may be identified in the **Value** field:

| Processor Type (hex values) | Description |
|--------------------------------|---------------------------|
| 45 00 | ARM7/ LPC21xx |
| 4D 00 | ARM Cortex-M4/ K21 Family |
| 4E 00 | ARM Cortex-M4/ K81 Family |

The following example shows the command and response.

Command: GetProcessorType: 56 69 56 4F 74 65 63 68 32 00 09 02 00 00 F0 F9

Response: 56 69 56 4F 74 65 63 68 32 00 09 00 00 05 DF 61 02 4D 00 AC 4D

1.1.3. Get Main Firmware Version (09-03)

The **Get Main Firmware Version** command returns main firmware version TLV.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-----------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 09h | 03 | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 0-9 |
|-----------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|-----------|
| Header Tag & Protocol | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 09h | See status code table | | | See below | | |

The Get Main Firmware Version sub-command returns a TLV string as follows:

Tag: 0xDF62

Length: Varies

Value: Varies field representing the main firmware version.

The following example shows the command and response.

Command: Get Main Firmware Version: 56 69 56 4F 74 65 63 68 32 00 09 03 00 00 C0 CE

Response: 56 69 56 4F 74 65 63 68 32 00 09 00 00 14 DF 62 11 43 72 61 6E 65 56 65 6E 64 69 5F 31 2E 30 2E 30 00 E1 5D

7.8.3 Get Hardware Information (09-14)

The **Get Hardware Information** command retrieves information about the reader's hardware.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-----------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 09h | 14h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-----------------------|---------|---------------------------------------|-------------------|-------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 09h | See status code table | | | See below | | |

The format for hardware module version information returned is "human readable", consisting of fields that are separated by commas, and lines separated by carriage return and line feed characters:

```
<module type>,<module name><CRLF>
<chip version>
```

The following example shows the hardware version information subcommand and the information being returned (in ASCII format).

For example, a command sent to a VP5300 would return the following:

```
5669564f7465636832000900001548572c205650353330300d0a4b383146205265
763477d5
```

In ASCII: HW, VP5300 <CR><LF>K81F.Rev4

7.8.4 Get System Info (09-15)

The **Get System Info** command retrieves system info for individually set items or an entire system. When retrieving all system info, this command returns only available information. Use Index byte **0xFF** to get all available system info.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|---------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2 10 | 09h | 15h | 00h | 01h | Index | | |

Data: Index byte is composed of chip and items as below.

| Bit7-Bit6 | Description | Bit5-Bit0 | Description |
|-----------|--|-----------|--|
| Chip | 00b: Secure Chip, (e.g., K81) 01b: App Chip, (e.g., RT1050) 10b: RFU 11b: RFU | Items | Valid range: 000001b~111110b 000000b and 111111b are RFU |

K81 Index: (0x01~0x3E)

| Index | b[7:6] | b[5:0] | Description |
|-------|--------|---------|-------------------------------|
| 0x01 | 00b | 000001b | K81 Flash Size |
| 0x02 | | 000010b | K81 Wi-Fi HW/FW Info |
| 0x03 | | 000011b | K81 BLE HW/FW Info |
| 0x04 | | 000100b | K81 LTE HW/FW Info |
| 0x05 | | 000101b | K81 Printer HW/FW Info |
| 0x06 | | 000110b | K81 LCD ID Info |
| 0x07 | | 000111b | K81 Audio CODEC HW/FW Info |
| 0x08 | | 001000b | K81 Touch IC HW/FW Info |
| 0x09 | | 001001b | K81 Ethernet HW/FW Info |
| 0x0A | | 001010b | K81 Light sensor HW/FW Info |
| 0x0B | | 001011b | K81 Camera HW/FW Info |
| 0x0C | | 001100b | K81 NFC controller HW/FW Info |

RT1050 Index: (0x41~0x7E)

| Index | b[7:6] | b[5:0] | Description |
|-------|--------|---------|-------------------------------|
| 0x41 | 01b | 000001b | RT1050 Flash Size |
| 0x42 | | 000010b | RT1050 Wi-Fi HW/FW Info |
| 0x43 | | 000011b | RT1050 BLE HW/FW Info |
| 0x44 | | 000100b | RT1050 LTE HW/FW Info |
| 0x45 | | 000101b | RT1050 Printer HW/FW Info |
| 0x46 | | 000110b | RT1050 LCD ID Info |
| 0x47 | | 000111b | RT1050 Audio CODEC HW/FW Info |
| 0x48 | | 001000b | RT1050 Touch IC HW/FW Info |
| 0x49 | | 001001b | RT1050 Ethernet HW/FW Info |

| Index | b[7:6] | b[5:0] | Description |
|-------|--------|---------|----------------------------------|
| 0x4A | | 001010b | RT1050 Light sensor HW/FW Info |
| 0x4B | | 001011b | RT1050 Camera HW/FW Info |
| 0x4C | | 001100b | RT1050 NFC controller HW/FW Info |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|--------------------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVotech2\0 | 09h | See status code table | 00h | variable | See Response Frame Data Format | | |

Response Frame Data Format:

1 byte index (same as index byte in command), 1 byte length and multiple bytes value

| Index (1 Byte) | | | Length (1 Byte) | Value (format) (Var Bytes) |
|----------------|---------|--------------------------------|--------------------|--|
| b[7:6] | b[5:0] | Description | | |
| Chip | 000001b | Chip Flash Size | var | Size in MB, e.g. 0x08: 8 MB 0x10: 16 MB 0x20: 32 MB 0x80: 128 MB |
| | 000010b | Chip Wi-Fi HW/FW Info | var | ASCII string |
| | 000011b | Chip BLE HW/FW Info | var | ASCII string |
| | 000100b | Chip LTE HW/FW Info | var | ASCII string |
| | 000101b | Chip Printer HW/FW Info | var | ASCII string |
| | 000110b | Chip LCD ID Info | var | ASCII string |
| | 000111b | Chip Audio CODEC HW/FW Info | var | ASCII string |
| | 001000b | Chip Touch IC HW/FW Info | var | ASCII string |
| | 001001b | Chip Ethernet HW/FW Info | var | ASCII string |
| | 001010b | Chip Light sensor HW/FW Info | var | ASCII string |
| | 001011b | Chip Camera HW/FW Info | var | ASCII string |
| | 001100b | Chip NFC controller HW/FW Info | var | ASCII string |

7.8.5 Get Module Version Information (09-20)

The **Get Module Version Information** command retrieves the reader's module information.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|--------------------------|---------|-----------------|-------------------------|-------------------------|--------------------------|--------------|--------------|
| Header Tag & Protocol | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 09h | 20h | 00h | 00h or 01h | See below | | |

Data: 0xFF is the only parameter accepted to retrieve the detailed implementation version, if supported.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|--------------------------|---------|---------------------------------------|-------------------------|-------------------------|--------------------------|--------------|--------------|
| Header Tag & Protocol | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 09h | See status code table | | | See below | | |

If there is an error, the reader returns the appropriate status code with an empty data field (Data Length = 0000h).

The format for module version information returned is "human readable," consisting of fields separated by commas and lines separated by carriage return and line feed characters:

```
<module type>,<module name and spec. version>,<implementation  
version><CR><LF>
```

Note: The example below shows the module version information subcommand and the information returned in ASCII format:

Command: Get Module Version Information: 56 69 56 4F 74 65 63 68 32 00 09
20 00 00 56 11

Response:

```

56 50 33 36 30 30 20 46 57 20 76 31 2E 30 30 2E 54 0D 0A 4E 45 4F
20 32 2E 30 20 76 31 2E 30 30 0D 0A 43 4C 20 41 49 44 2C 20 50 50
53 45 20 4D 6F 64 75 6C 65 2C 20 76 31 2E 32 0D 0A 43 4C 20 41 49
44 2C 20 56 69 73 61 20 56 43 50 53 20 32 2E 32 2C 20 76 31 2E 30
0D 0A 43 4C 20 41 49 44 2C 20 41 6D 65 78 20 45 78 70 72 65 73 73
50 61 79 20 33 2E 31 2C 20 76 31 2E 32 0D 0A 43 4C 20 41 49 44 2C
20 4D 61 73 74 65 72 43 61 72 64 20 50 61 79 50 61 73 73 20 4D 2F
43 68 69 70 20 33 2E 31 2C 20 76 31 2E 33 0D 0A 43 4C 20 41 49 44
2C 20 44 69 73 63 6F 76 65 72 20 44 50 41 53 20 31 2E 30 20 5A 69
70 20 33 2E 31 2E 32 2C 20 76 31 2E 31 0D 0A 43 4C 20 41 49 44 2C
20 43 55 50 20 71 55 49 43 53 20 31 2E 30 2E 32 2C 20 76 31 2E 31
0D 0A 43 4C 20 41 49 44 2C 20 49 6E 74 65 72 61 63 20 31 2E 35 2C
20 76 32 2E 30 30 0D 0A 43 4C 20 41 49 44 2C 20 4A 43 42 20 31 2E
34 2C 20 76 31 2E 30 0D 0A 43 4C 20 41 49 44 2C 20 41 70 70 6C 65
50 61 79 20 56 41 53 20 31 2E 30 2E 31 30 2C 20 76 31 2E 32 0D 0A
43 4C 20 41 49 44 2C 20 53 6D 61 72 74 54 61 70 20 32 2E 31 2C 20
76 31 2E 32 0D 0A 43 4C 20 4C 31 2C 20 45 4D 56 32 2E 36 62 2C 20
76 35 2E 30 32 0D 0A 43 54 20 4C 32 2C 20 45 4D 56 20 43 6F 6D 6D
6F 6E 20 4C 32 20 56 31 2E 31 30 2E 30 33 37 0D 0A 43 54 20 4C 31
2C 20 45 4D 56 34 2E 33 61 2C 20 76 32 2E 30 30 0D 0A

```

ASCII translation of the data field:

```

VP3600 FW v1.00.T<CR><LF>
NEO 2.0 v1.00<CR><LF>
CL AID, PPSE Module, v1.2<CR><LF>
CL AID, Visa VCPS 2.2, v1.0<CR><LF>
CL AID, Amex ExpressPay 3.1, v1.2<CR><LF>
CL AID, MasterCard PayPass M/Chip 3.1, v1.3<CR><LF>
CL AID, Discover DPAS 1.0 Zip 3.1.2, v1.1<CR><LF>
CL AID, CUP qUICS 1.0.2, v1.1<CR><LF>
CL AID, Interac 1.5, v2.00<CR><LF>
CL AID, JCB 1.4, v1.0<CR><LF>
CL AID, ApplePay VAS 1.0.10, v1.2
CL AID, SmartTap 2.1, v1.2<CR><LF>
CL L1, EMV2.6b, v5.02<CR><LF>
CT L2, EMV Common L2 V1.10.037<CR><LF>
CT L1, EMV4.3a, v2.00<CR><LF>

```

7.8.6 Get TM4 Firmware Version Information (C7-3F)

The **Get TM4 Firmware Version Information** command retrieves Trimag4 IC firmware version information.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | C7h | 3Fh | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... 14+ n - 1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-------------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | C7h | See status code table | 00h | xx | TM4 version string, see below | | |

If the TM4 IC is working correctly, the response string is:

"ID TECH TM4 for NEO2.0 RS232 Reader VX.YY" (ASCII format)

| Version number | TM4 application |
|----------------|-----------------|
| 1.XX | TM4 905 |
| 2.XX | TM4 905 JIS |
| 3.XX | TM4 905 ISO |
| 4.XX | TM4 907 |

7.9 International Language Support

The International Language Support feature offers support for foreign languages, including those based on graphical fonts, such as Chinese, Thai, Arabic, and others. This feature applies only to readers that have graphical output capability.

International Language Support has four built-in core language options. Some of these are font-based; the others are ideogram-based. The ideogram language options are stored in flash as bitmaps. The other languages use fonts stored in flash.

For example, an English message is composed of multiple small bitmaps that represent different characters (for example, a "Thank You" message is 8 bitmaps displayed together). The ideogram messages are usually a single bitmap (for example, the Chinese "Thank You" message is a single bitmap displayed on the screen). See [Preparing Bitmaps for Use by ILM](#) for instructions on working with bitmaps.

Each reader keeps four "core" languages in its firmware, ensuring the reader can be used in virtually any geographical area "as is." A core language may NOT be modified or deleted by customer/third party actions. Core languages always exist and are always available for use.

At present, International Language Support offers four (4) core languages:

- English (U.S., fonts)
- French (fonts)
- Chinese (ideograms)
- English + Chinese (fonts & ideograms)

See [Set Configuration](#) (04-00) for information on how to set a language preference.

7.9.1 Other Languages

In addition to the core Languages, International Language Support offers a fifth language called "Other Language," which is a slot in flash memory that can contain all the message bitmaps for a language.

Because this Other Language is stored in the ideogram bitmap format, it eliminates the need for fonts and other requirements. The Other Language only needs a unique record for each distinct message it must display.

At present there are twenty-two (22) displayable predefined message types.

Note: Each time a user configures the reader for International Language Support, they must download messages for all 22 indices in consecutive order. Users cannot change individual messages.

7.9.2 Bitmap Conversion Completed by POS

The reader expects to load a simplified version of the monochrome bitmap. While data is the same as in a standard bitmap, it must be converted to a format that the LCD hardware can use.

The standard 40-byte DIB bitmap header is discarded. It is replaced by a simplified ViVOpay header, described in [ILM Header Format](#).

The bitmap data produces an LCD image that is:

- Reversed in color (black is white, and vice versa).
- Upside down.

White space reduction

Large parts of the bitmap are empty background. The LCD does not need to save this white space, because it corresponds to off-pixel values (which are already turned off). This limited form of image compression makes the image much smaller.

7.9.3 ILM Header Format

Each bitmap loaded onto a reader is expected to contain a proprietary header instead of the standard DIB header.

This header format is shown in the following table, prefixed to the actual bitmap data:

| Bytes 0-1 | Bytes 2-3 | Bytes 4-5 | Bytes 6-7 | Bytes 8-9 | Bytes 10-11 | Bytes 12...n |
|------------------|---------------|------------------|--------------|--------------|-------------------------------|-----------------|
| Bitmap Length | Row Number | Column Number | Height | Width | Type (truncate or other info) | Bitmap data |

All variables in the header are 2 bytes long.

| Byte | Description |
|---------------|--|
| Bitmap Length | This data field contains the total number of bytes in the Bitmap Data Field. |
| Row Number | This data field contains the row offset that this image should start at. Value is in PIXELS. |
| Column Number | This data field contains the column offset that this image should start at. Value is in BYTES. |
| Height | This data field contains the number of rows (in pixels) that this image contains. |
| Width | This data field contains the number of columns (in bytes) that this image contains. |
| Type | Reserved; must be set to 00h. |
| Bitmap Data | This is the actual image data. |

7.9.4 Language Version Information

This block contains data used for version control of the ILM. It contains variables that identify the particular language module that is currently stored in the system.

The language module specifies both the language name and the Country Code for situations where a number of languages (English, French, Spanish, and others) are shared by multiple nations but the reader is intended to be operate in a specific country (such as Canada).

Language Version Information

| Variable | Length (bytes) | Description/Example |
|---------------|----------------|---|
| Language Name | 25 | ASCII String, Null Terminated – "Spanish\0" |
| Abbreviation | 4 | ASCII String, Null Terminated – "ES\0" (ASCII-2 character Country Code) or "ESP\0" (ASCII-3 character Country Code) or "724\0" (3 digit decimal number Country Code) |
| Format | 1 | Unsigned 8-bit Integer |
| Author | 10 | ASCII String, Null Terminated – "NCR\0" |
| Version | 6 | ASCII String, Null Terminated – "1.3\0" |
| ID | 4 | Unsigned 32-bit Integer |

Language Name is the name of the ILM language, using ASCII characters.

Abbreviation is the Country Code listed in ISO 3166. The format is either ASCII-2 character, ASCII-3 character or a 3 digit decimal number. Regardless of method, it is stored in ASCII alphanumeric characters.

The **Format** data field specifies the format of the Abbreviation field noted above. All follow the specification from ISO DOC 3166.

| Format Data Field | |
|-------------------|--------------------------------|
| Value | Description/Example |
| 01 | Alpha2 (2 character) – "ES\0" |
| 02 | Alpha3 (3 character) – "ESP\0" |
| 03 | Decimal (3 digit) – "724\0" |

Author is an ASCII string noting the customer that developed this language module.

Version is an ASCII string, also customer defined. It identifies this language module by version number.

ID is a value that is reserved for future use. It is currently **NOT** used.

All fields are the length indicated. If the ASCII string does not occupy the entire data field (as is usually the case), the remaining bytes **MUST** be padded with zeroes.

The Language Version Information area is provided to the customer as a way to track which language is currently loaded into the reader. It can be accessed and values are returned to the POS. The intent is to facilitate automated updating through the POS. The POS can examine the existing language module currently stored, and then make appropriate decisions as to its use (for example, updating the module).

How a customer uses the Language Version Information cannot be defined or enforced. It is only provided for identification and could be unused.

However, value of 00h in this area is interpreted as indicating that the ILM area of flash memory is empty. Therefore, if the customer does edit the ILM area, they **MUST** update this Version Information area as well, if only to write arbitrary non-zero values to it.

7.10 EMV Certificate Revocation List Commands

The Certificate Revocation List (CRL) contains entries that include the RID, key index, and certificate serial numbers for cards that should be rejected. The kernel checks the CRL for entries matching the index and serial number of the issuer public key certificate *provided by the card*. If it is found the card is rejected.

The CRL is maintained in non-volatile memory but a copy is kept in RAM to provide faster access during transactions. A tag (DF26h) is defined to enable or disable the entire revocation feature in the reader.

The M/Chip 3.0 application is the only application capable of using the Certificate Revocation List feature.

The firmware supports a maximum of 30 entries in the certificate revocation list.

7.10.1 Get EMV Revocation Log Status (84-03)

The **Get EMV Revocation Log Status** command returns information about the EMV revocation log. The host device can use the information to determine how to read the log.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 84h | 03h | 00 | 00 | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 84h | See status code table | | | TLV Data Objects | | |

If the command is successful, the following data is returned. All fields are encoded with MSB first.

| Offset | Length (bytes) | Data Description |
|--------|----------------|-------------------|
| 00h | 4 | Version number |
| 04h | 4 | Number of records |
| 08h | 4 | Size of records |

7.10.2 Add Entry to EMV Revocation List (84-04)

The **Add Entry to EMV Revocation List** command adds a new entry to the revocation list.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|-------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 84h | 04h | 00 | 09 | | | |

The data field contains the revocation list entry:

| Offset | Length (bytes) | Data Description | Example |
|--------|-------------------|---|----------------|
| 00h | 5 | RID (packed hex format) | A0 00 00 00 04 |
| 04h | 1 | Key Index (packed hex format) | F8 |
| 08h | 3 | Certificate serial number (packed hex format) | 00 10 00 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------------|---------|---|----------------------|----------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Status | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 84h | See status code table | 00 | 00 | | |

7.10.3 Delete All Entries for Single Index in EMV Revocation List (84-05)

The **Delete All Entries for Single Index in EMV Revocation List** command deletes all entries *that match a key index and RID* from the EMV revocation list.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|-------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 84h | 05h | 00 | 06 | | | |

The data field contains RID and key index for the records to be deleted.

| Offset | Length (bytes) | Data Description | Example |
|--------|-------------------|-------------------------------|----------------|
| 00h | 5 | RID (packed hex format) | A0 00 00 00 04 |
| 05h | 1 | Key Index (packed hex format) | F8 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVotech2\0 | 84h | See status code table | 00 | 00 | | |

7.10.4 Delete All Entries from EMV Revocation List (84-06)

The **Delete All Entries from EMV Revocation List** command deletes *all* entries from the EMV revocation list.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVotech2\0 | 84h | 06h | 00 | 00 | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVotech2\0 | 84h | See status code table | 00 | 00 | | |

7.10.5 Get EMV Revocation List (84-07)

The **Get EMV Revocation List** command retrieves a sequence of consecutive records from the EMV revocation list. The list may be retrieved in several command exchanges, depending on the size specified in the command and the number of entries in the list.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVotech2\0 | 84h | 07h | 00 | 00 | | | |

The data field specifies the following information:

| Offset | Length (bytes) | Data Description |
|--------|----------------|--|
| 00h | 2 | Size - maximum number of bytes to be retrieved (MSB first) |
| 02h | 2 | Starting record ⁶ (MSB first) |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|-------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 84h | 07h | 00 | 00 | | | |

The data field will contain the maximum number of records that can fit in the size provided in the command. No partial records are returned.

The data field is formatted as follows:

| Offset | Length (bytes) | Data Description |
|--------|----------------|---|
| 00h | 4 | Number of records returned |
| 04h | 4 | Number of records remaining in the file |
| 08h | 4 | Record size |
| 0Ch | varies | Revocation list records |

Each record is formatted as follows:

| Offset | Length (bytes) | Data Description | Example |
|--------|----------------|---|----------------|
| 00h | 5 | RID (packed hex format) | A0 00 00 00 03 |
| 05h | 1 | Key Index (packed hex format) | FE |
| 08h | 3 | Certificate Serial Number (packed hex format) | 00 10 00 |

⁶ For GR, this number must be less than 30. The first record is 0.

7.10.6 Delete an Entry from EMV Revocation List (84-0D)

The **Delete an Entry from EMV Revocation List** command deletes a specific entry from the EMV revocation list. Unlike the commands described previously, this command deletes the specific entry that matches *the RID, the key index, and the certificate serial number*.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|--------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViV0tech2\0 | 84h | 0Dh | 00 | 09 | | | |

The data field contains the information to select the EMV revocation list record:

| Offset | Length (bytes) | Data Description | Example |
|--------|-------------------|---|----------------|
| 00h | 5 | RID (packed hex format) | A0 00 00 00 04 |
| 05h | 1 | Key Index (packed hex format) | F8 |
| 06h | 3 | Certificate Serial Number (packed hex format) | 00 10 01 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------------|---------|---|-------------------------|-------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Status | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | 84h | See status code table | 00 | 00 | | |

7.11 EMV Exception Log List Commands

The EMV Exception Log List contains entries that include the PAN and Sequence number for cards that should be rejected. If enabled, the kernels check this list for entries matching the PAN and Sequence number *provided by the card*. If the supplied number is found the card is rejected.

Tag DFEE3E is defined to enable or disable the Exception Log List feature in the reader. The following commands deal with EMV Exception Log Lists.

7.11.1 Get EMV Exception Log Status (84-08)

The **Get EMV Exception Log Status** command returns information about the EMV exception log. The response includes the version number, record size, and number of records contained in the file.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 84h | 08 | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|--------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 84h | See status code table | 00h | 0Ch | | | |

The data returned in a successful command contains the following information:

| Offset | Length (bytes) | Description |
|--------|----------------|-------------------|
| 00h | 4 | Version number |
| 04h | 4 | Number of records |
| 08h | 4 | Size of records |

7.11.2 Add Entry to EMV Exception List (84-09)

The **Add Entry to EMV Exception List** command adds an entry to the EMV exception list.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-25 | Byte 26 | Byte 27 |
|-------------------------------|---------|-------------|-------------------|-------------------|------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 84h | 09 | 00h | 0Ch | See Below | | |

The exception list data is as follows:

| Offset | Length (bytes) | Description | Example |
|--------|----------------|---|-----------------------|
| 00h | 1 | PAN logical length in bytes (packed hex format). Must be <= 0Ah | 08h |
| 01h | 10 | PAN (packed hex format, padded with 'F' if required) | 5413339000001596FFFFh |
| 0Bh | 1 | Sequence number (packed hex format) | 00h |

Example Exception List File Data (in a non-null terminated ASCII text file):

```
084761739001010010FFFF01085762739101010010FFFF0008112233445566778F
FFFF050A1122334455667788990F00
```

The above file has four entries:

- 08 4761739001010010FFFF 01 where PAN = 4761739001010010
- 08 5762739101010010FFFF 00 where PAN = 5762739101010010
- 08 1122334455667788FFFF 05 where PAN = 112233445566778
- 0A 1122334455667788990F 00 where PAN = 1122334455667788990

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 84h | See status code table | 00h | 00h | | |

7.11.3 Delete Entry from EMV Exception List (84-0A)

The **Delete Entry from EMV Exception List** command deletes an entry from the EMV exception list.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-25 | Byte 26 | Byte 27 |
|-------------------------------|---------|-------------|-------------------|-------------------|------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 84h | 0A | 00h | 0Ch | See Below | | |

The exception list data is as follows:

| Offset | Length (bytes) | Description | Example |
|--------|----------------|---|-----------------------|
| 00h | 1 | PAN logical length in bytes (packed hex format). Must be <= 0Ah | 08h |
| 01h | 10 | PAN (packed hex format, padded with 'F' if required) | 5413339000001596FFFFh |
| 0Bh | 1 | Sequence number (packed hex format) | 00h |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 84h | See status code table | 00h | 00h | | |

7.11.4 Delete All Entries from EMV Exception List (84-0B)

The **Delete All Entries from EMV Exception List** command deletes all entries from the EMV exception list.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 84h | 0B | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 84h | See status code table | 00h | 00h | | |

7.11.5 Get EMV Exception List (84-0C)

The **Get EMV Exception List** command retrieves consecutive records from the EMV exception list.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Bytes 14-15 | Bytes 16-17 | Byte 18 | Byte 19 |
|-------------------------------|---------|-------------|-------------------|-------------------|---------------------|-----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Max number of bytes | Starting Record | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 84h | 0C | 00h | 04h | N | 0-65535 | | |

The maximum number of bytes is a 16-bit binary number, MSB first.

The starting record is the first record to be retrieved; expressed as a 16-bit binary number MSB first, value from 0-65535.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Bytes 14-13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|---------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 84h | See status code table | | | | | |

The data returned is the maximum number of transaction records that satisfy the command constrains. The number of bytes returned always is an integer multiple of the transaction record size (in other words, no partial records are returned) plus 12 decimal bytes. The maximum number of data bytes that can be returned in a single operation is limited to 4080 bytes.

| Offset | Size in Bytes MSB First | Description |
|--------|-------------------------|-------------------------------------|
| 0 | 4 | Number of records returned |
| 4 | 4 | Number of records remaining in file |
| 8 | 4 | Record size |
| C | n-C | Exception list records |

The format of an exception list record (as returned in the response data) is as follows:

Exception List Record Format

| Offset | Field | Length (bytes) |
|--------|---------------------------------------|----------------|
| 0 | PAN Length (Logical) | 1 |
| 1 | PAN (right-padded with F if required) | 10 |
| 11 | PAN Sequence Number | 1 |

An example of an exception list records returned by this serial command is given below (only the data section of the response is given), along with an explanation.

| Data Section of the Response Frame ⁷ | Explanation |
|--|----------------------------------|
| 08 47 61 73 90 01 01 00 10 FF FF 01 | Rec 1: PAN = 4761739001010010 |
| 08 57 62 73 91 01 01 00 10 FF FF 00 | Rec 2: PAN = 5762739101010010 |
| 08 11 22 33 44 55 66 77 8F FF FF 05 | Rec 3: PAN = 112233445566778 |
| 0A 11 22 33 44 55 66 77 88 99 0F 00 | Rec 4: PAN = 1122334455667788990 |

7.12 Generic Pass-through Commands

The commands in this section provide the basic capability to communicate directly with the chip on a chip card. They provide control of the polling process, and exchange of application protocol data units (APDU). These commands may thus be used to "extend" the capabilities of the ViVOpay reader to (for example) accept cards using application protocols that are not currently supported in the reader's firmware.

Note: Pass-Through commands can only be executed in Pass-Through Mode. Other commands (that is, non-Pass-Through commands) return an error if send while a reader is in Pass-Through Mode with the exception of **Ping**, **Get Version**, and **Get Serial Number** commands, which work in both modes.

If the reader is not in Pass-Through Mode, a "Pass-Through Mode Start" command must first be issued before sending Pass-Through commands. Otherwise, the commands in this section will result in an error unless otherwise specified.

The following commands always work regardless of whether or not the reader is in Generic Pass-Through mode:

- 0x1201
- 0x1801
- 0x2801
- 0x2900
- 0x2C01
- 0x6014

The following commands only work in Generic Pass-Through mode. To use them, first run the **Pass-Through Mode Start/Stop (2C-01)** command.

- 0x0A02
- 0x0A03
- 0x0B01

⁷ Spaces have been added in this example to increase readability; the space does not exist in the actual data. The data exists as hex bytes.

- 0x0B02
- 0x2C02
- 0x2C03
- 0x2C04
- 0x2C05
- 0x2C06
- 0x2C07
- 0x2C08
- 0x2C09
- 0x2C0A
- 0x2C0B
- 0x2C0C
- 0x2C11
- 0x2C12
- 0x2C13
- 0x2C18
- 0x2C19
- 0x2C1A
- 0x2C1B
- 0x2C1C
- 0x2C70
- 0x2C71
- 0x2C40
- 0x2C41
- 0x2C42
- 0x2C43

7.12.1 Pass-Through Mode Start/Stop (2C-01)

The **Pass-Through Mode Start/Stop** command is used to enter and exit Pass-Through Mode. A ViVOpay reader can only accept Pass-Through commands when it is in Pass-Through Mode.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|---------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | 01h | 00h | 01h | Mode | | |

Mode

0 = Stop Pass-Through

1 = Start Pass-Through

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | See status code table | 00h | 00h | | |

The **Pass-Through Mode Start** command *must* be used to enter Pass-Through Mode. The **Pass-Through Mode Stop** command can only be used in Pass-Through Mode. If the reader is not in Pass-Through mode when the **Pass-Through Mode Stop** command is issued, the reader will respond with an error.

7.12.2 Poll for Token (2C-02)

After starting Pass-Through Mode, the ViVOpay reader will not poll for any cards until receiving the **Poll for Token** command. This command tells the reader to start polling for a Type A or Type B PICC until it detects a PICC or times out.

This command automatically turns the RF Antenna on.

After detecting a PICC within the specified time limit, the reader activates it and responds back to the host device with card-related data, such as the serial number.

If the reader does not detect a PICC within the specified time limit, it stops polling and responds back indicating that it found no card and does not return any card-related data.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14,15 | Byte 16 | Byte 17 |
|-------------------------------|---------|-------------|-------------------|-------------------|------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 2Ch | 02h | 00h | 02h | See Below | | |

Poll for Token Data Field for Command Frame

| Data Field | Length (bytes) | Description | | | | | | | | | | | | |
|------------|----------------|---|----------|------------|---|---|---|----|---|----|---|---|-----|------|
| Timeout1 | 1 | Time in Seconds Timeout1 cannot be zero seconds if Timeout2 is Zero. | | | | | | | | | | | | |
| Timeout2 | 1 | Multiplier for Time in multiples of 10 milliseconds. <table><tr><th>Timeout2</th><th>Time in ms</th></tr><tr><td>0</td><td>0</td></tr><tr><td>1</td><td>10</td></tr><tr><td>2</td><td>20</td></tr><tr><td>:</td><td>:</td></tr><tr><td>255</td><td>2550</td></tr></table> | Timeout2 | Time in ms | 0 | 0 | 1 | 10 | 2 | 20 | : | : | 255 | 2550 |
| Timeout2 | Time in ms | | | | | | | | | | | | | |
| 0 | 0 | | | | | | | | | | | | | |
| 1 | 10 | | | | | | | | | | | | | |
| 2 | 20 | | | | | | | | | | | | | |
| : | : | | | | | | | | | | | | | |
| 255 | 2550 | | | | | | | | | | | | | |

The reader uses Timeout1 and Timeout2 together to calculate the Timeout (the time to wait for a PICC).

Example:**Poll for Token Timeout**

| Timeout1 | Timeout2 | Timeout |
|----------|----------|-------------------|
| 0 | 0 | Not Allowed |
| 0 | 20 | 0 Seconds, 200 ms |
| 0 | 50 | 0 Seconds, 500 ms |
| 0 | 100 | 1 Second |
| 1 | 0 | 1 Second |
| 1 | 20 | 1 Second, 200 ms |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | See status code table | 00h | Variable | below | | |

The Data field contains data only if the Status Code is OK.

Poll for Token Data Field for Response Frame (Status Code is OK)

| Data Field | Length (bytes) | Description |
|---------------|----------------|--|
| Card | 1 | Type of Card Found (or No Card Found). 00h None (Card Not Detected or Could not Activate) 01h ISO 14443-4 Type A 02h ISO 14443-4 Type B 03h Mifare Classic 04h Mifare Ultralight 05h ISO 14443-3 Type A 06h ISO 14443-3 Type B 07h ISO 14443 Type A and Mifare (NFC phone) 08h Felica 09h RCTIF Type B Prim 0Ah RCTIF Type A 0Bh RCTIF Type B 0Ch Mifare Plus 0Dh Mifare Mini 0Eh Mifare DESFire 0Fh Reserved 10h ISO 18092 11h ISO 15693 12h ISO 18000 |
| Serial Number | 0 or Variable | Serial Number (or the UID) of the PICC. Length depends on the Card Detected. If no card was detected, then a Serial Number is not returned. |

Note: Most cards use a 4-byte UID, so the data field of the response is five (5) bytes long. However, some cards with 7-byte UIDs exist and it is expected that cards with 10-byte UIDs will become available. All of these card types are handled by this command.

7.12.3 Get PCD and PICC Parameters (2C-05)

The **Get PCD and PICC Parameters** command allows the host device to retrieve PCD and PICC related parameters from the ViVOpay reader.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 2Ch | 05h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | See status code table | 00h | 00h or 0Fh | See Table below | | |

If a valid command frame is received from the host device, the reader retrieves the parameters from the PCD and PICC. If the parameters are retrieved successfully, the reader returns a response frame with OK Status and Data containing the parameters given below. For details on the parameters, refer to ISO 14443.

If the command frame contains any errors, or an error occurred while retrieving the parameters, then the reader sends a response frame with an appropriate Status. No data is returned in this case.

Data Fields for the response frame (if Status = OK)

Get PCD and PICC Parameters Data Field

| Data Byte | Name | Length (bytes) | Format | Notes |
|--------------|---------------------|-------------------|--------|--|
| 0-1 | Reader Buffer Size | 2 | Binary | Reader RF Buffer Size stored as a big-endian number. |
| 2-3 | Max PICC Frame Size | 2 | Binary | Maximum PICC Frame Size stored as a big-endian number. |
| 4 | CID | 1 | Binary | CID |
| 5 | Block | 1 | Binary | Block Number. |
| 6 | CID Supported | 1 | Binary | CID Supported |
| 7-10 | FWT | 4 | Binary | Frame Waiting Time in ETUs. It is stored as a big-endian number. |
| 11-14 | D-FWT | 4 | Binary | Delta FWT in ETUs. It is stored as a big-endian number. |

If you need to use these parameters, you should issue this command immediately after issuing a "Poll for Token" command. This command simply reads the last parameters out of the control block used to set parameters in the RF chip.

7.12.4 High-level Halt Command (2C-09)

The **High-level Halt** command instructs the ViVOpay reader to send a HALT command to any Type A or Type B card. This command can only be used after the reader has been put in Pass-Through mode and the **Poll for Token** command has indicated that a card is present.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 2Ch | 09h | 00h | 01h | Card Type | | |

Card Type:

- 0x01 = Type A
- 0x02 = Type B

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | See status code table | 00h | 00h | | |

7.12.5 Enhanced Pass-Through Command (2C-0B)

The **Enhanced Pass-Through** command instructs the reader to carry out several tasks while in Pass-Through Mode. This command is ONLY enabled in Pass-Through Mode. The reader ignores this command when not in Pass-Through Mode.

Notes:

- In order to support backward compatibility, only VP6300 series products can fully support this command. Other products may support it partially.
- SAM interface is only supported in SRED readers. Non-SRED readers do not support SAM.

This command is used in three basic situations:

1. To initiate a Pass-Through transaction.
2. To terminate a "successful" transaction.
3. To terminate a "failed" transaction.

This command is based in large part on the standard **Set Message LED/Buzz** command. It differs primarily in that it can also instruct a reader to perform the following tasks:

- Activation/Deactivation of interface
- Turn On/Off Antenna (PICC)
- Single-shot command processing
- Poll for Token (PICC/ICC/SAM)
- Sound the buzzer in various ways
- Turn on or off any of the LEDs
- Write text and amount messages to the display

When only used to set the Buzzer/LED, this command works on the all ID TECH readers.

Three use-cases depend on the LCD Message index number:

- Index 00h to 07h messages are directly display by the reader. Normally these messages are not set through this command.
- Index 08h to 0Bh messages can be set by the host device.
- Index FFh indicates the host device is setting LED/Buzzer only.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|---------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 2Ch | 0Bh | 07 + N OR 7 + X + Y | | See Data Table | | |

The table below describes the format and contents of the data field in the command frame.

Enhanced Pass-Through Data Field

| Data Item | Length (bytes) | Description |
|-------------------------|-------------------|---|
| Single-Shot Commands | 1 | <p>This bitmask contains the following commands. If a bit is set (1), the command is issued by the reader in the proper order. If the command mask is cleared (0), the command is NOT executed.</p> <p>0000 0001: Activate Interface (mutually exclusive) 0000 0010: Deactivate Interface (mutually exclusive) 0000 0100: Issue Poll For Token 0000 1000: Use Independent LED instead (mutually exclusive) 0001 0000: Use Independent Buzzer instead (mutually exclusive) 0010 0000: Use Specified Interface 0100 0000: Reserved 1000 0000: Reserved</p> <p>For more information see Single-Shot Commands below.</p> |

| Data Item | Length (bytes) | Description |
|---|----------------|--|
| LCD Message Index (for readers with a display only) | 1 | <p>NOTE: For the complete list of possible messages, see the Default Message Index.</p> <p>00-07 is controlled by the reader and normally not set by this command 00: Idle Message (Welcome) 01: Present card (Please Present Card) 02: Time Out or Transaction Cancel (No Card) 03: Transaction between reader and card is in the middle (Processing...) 04: Transaction Pass (Thank You) 05: Transaction Fail (Fail) 06: Amount (Amount \$ 0.00 Tap Card) 07: Balance or Offline Available funds (Balance \$ 0.00)</p> <p>08-0B is controlled by the host device through this command 08: Insert or Swipe card (Use Chip & PIN) 09: Try Again(Tap Again) 0A: Indicate the custom to present only one card (Present 1 card only) 0B: Indicate the custom to wait for authentication/authorization (Wait)</p> <p>80 MASK – indicates the User has included a String1 character string to be displayed with the standard message. If 0x80 is present, then the message index (in the lower portion of the byte) is displayed and LCD String1 and LCD String2 are only used for the Amount and Balance messages.</p> <p>FF indicates not to set the LCD message which allows host device to set LED/Buzzer only.</p> <p>EXAMPLE 1: Index = 00h, reader displays standard "Welcome" EXAMPLE 2: Index = 86h, reader displays standard 06h "Amount" message but also displays String1 (in this case String1 = "\$3.95")</p> |
| Beep Indicator | 1 | 00h: No beep 01h: Single beep 02h: Double beep 03h: Three short beeps 04h: Four short beeps 05h: One long beep of 200 ms 06h: One long beep of 400 ms 07h: One long beep of 600 ms 08h: One long beep of 800 ms |
| LED Number | 1 | 00h: LED 0 (Power LED) 01h: LED 1 02h: LED 2 03h: LED 3 FFh: All LEDs Where the LEDs are numbered 0, 1, 2, 3 counting from the left. Note: If you are using past-through mode to control the Power LED (LED 0), it is your responsibility to make sure that it behaves correctly. |
| LED Status | 1 | 00h: LED Off 01h: LED On |

| Data Item | Length (bytes) | Description | | | | | | | | | | | | |
|---------------------|----------------|--|----------|------------|---|---|---|----|---|----|---|---|-----|------|
| Timeout1 | 1 | Time in Seconds. Timeout1 cannot be zero seconds if Timeout2 is Zero. | | | | | | | | | | | | |
| Timeout2 | 1 | Multiplier for Time in multiples of 10 milliseconds. <table><tr><th>Timeout2</th><th>Time in ms</th></tr><tr><td>0</td><td>0</td></tr><tr><td>1</td><td>10</td></tr><tr><td>2</td><td>20</td></tr><tr><td>:</td><td>:</td></tr><tr><td>255</td><td>2550</td></tr></table> | Timeout2 | Time in ms | 0 | 0 | 1 | 10 | 2 | 20 | : | : | 255 | 2550 |
| Timeout2 | Time in ms | | | | | | | | | | | | | |
| 0 | 0 | | | | | | | | | | | | | |
| 1 | 10 | | | | | | | | | | | | | |
| 2 | 20 | | | | | | | | | | | | | |
| : | : | | | | | | | | | | | | | |
| 255 | 2550 | | | | | | | | | | | | | |
| LCD String1 Message | X | <p>This field is included when LCD Message Index AND 80h= True. The field is X bytes long and consists of a simple character string. It contains NO formatting information, ONLY text characters. If LCD String1 Message and LCD String2 Message are included, then the reserved field must be included, with LCD String1 Message appearing immediately after it.</p> <p>Note: The string must be null terminated (00) to indicate the end of the string.</p> | | | | | | | | | | | | |
| LCD String2 Message | Y | <p>The field is Y bytes long and consists of a simple character string. It contains NO formatting information, ONLY text characters.</p> <p>Note: The string must be null terminated (00) to indicate the end of the string.</p> | | | | | | | | | | | | |
| Selected Interface | Z | <p>This field is present when the Single-Shot Commands Byte, Specified Interface bit = 1:</p> <p>It is 1 byte long.</p> <p>Allowed Interface Values are:</p> <p>00h= Contactless</p> <p>20h= RFU</p> <p>21h= SAM1 (SRED version only)</p> <p>22h= SAM2 (SRED version only)</p> <p>Note: If this field is not present, the firmware will default to standard PICC behavior OR generate an error, depending upon which actions are indicated.</p> | | | | | | | | | | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | See status code table | 00h | Variable | below | | |

The message has been set to the requested message only if the response frame contains an OK Status Code.

If the interface is SAM, the response is ATR.

If the interface is Contactless, the response is as follows: If the Poll for Token bit is disabled, the response returns no data, and likewise for the beep indicator. If the Poll for Token bit is enabled, the response is the same as the **Poll for Token (2C-02)** command.

1.1.3.1. Single Shot Commands

The first data byte of the command lists several 'single-shot' commands. If the associated bit of this field is set, the command carries out the indicated discrete operation. If the bit is cleared, the command does NOT carry out the associated operation.

If the Poll for Token bit is enabled, the reader checks the value of the Use Specified Interface bit.

If the Use Specified Interface bit = 0, the reader polls for a PICC card.

If the Use Specified Interface bit = 1, the reader attempts a Poll for Token operation on the interface specified in the Selected Interface byte.

1.1.3.2. Activate Interface

This bit turns on the RF antenna for the contactless interface by default. If the "Use Specified Interface" bit in the Single-Shot Command byte is set, the specified interface will be activated:

- Contactless turns on the RF antenna
- SAM1, SAM2 activates the appropriate SAM

1.1.3.3. Deactivate Interface

This turns off the RF antenna for the contactless interface by default. If the Use Specified Interface bit in the Single-Shot Command byte is set, this bit turns off the indicated interface:

- Contactless turns off the RF antenna
- SAM1, SAM2 deactivates the appropriate SAM

Note: The interface to be deactivated MUST be the one that is currently active.

1.1.3.4. Poll for Token

This bit executes a Poll for Token on the Contactless interface by default. If the Use Specified Interface bit in the Single-Shot Command byte is set, this bit performs a Poll for Token on the indicated interface:

- Contactless performs a Poll for Token
- SAM1, SAM2 performs a Poll for Token on appropriate SAM slot

1.1.3.5. Use Independent Buzzer

If the Use Independent Buzzer bit is set, the reader checks the Beep Indicator byte and calls the standard Buzzer Command (0B 01).

If the bit is cleared, the reader checks the Buzzer byte and follows the Set Message Buzzer command (01 02).

1.1.3.6. Use Independent LED

If the Use Independent LED bit is set, the reader reads the included LED Number and Status bytes and then call the standard LED Command (0A 02).

If the bit is cleared, the reader reads the included LED bytes and then calls the Set Message LED command (01 02).

1.1.3.7. Mutual Exclusions

This command has several elements that are mutually exclusive; the command fails if both are enabled.

| | |
|-----------------------------|---|
| Antenna | You cannot enable both the Switch On and Switch Off options. |
| Independent LED Enabled | Reader uses the LED Bytes and calls the standard 0A 02 LED command |
| Independent LED Disabled | Reader uses the LED Bytes and follows the SetMsg 01 02 LED command |
| Independent Buzzer Enabled | Reader uses the Buzzer Byte and calls the standard 0B 01 Buzzer command |
| Independent Buzzer Disabled | Reader uses the Buzzer Byte and follows the SetMsg 01 02 Buzzer command |

LED, Buzzer and Message operations all occur simultaneously. The order in which processes are executed depends upon the situation:

If SWITCH ANTENNA ON is enabled:

- Switch Antenna On
- Perform any Message, LED, and Buzzer operations
- Poll for Token

If SWITCH ANTENNA OFF is enabled:

- Perform any Message, LED, and Buzzer operations
- Switch Antenna Off

1.1.3.8. Example Using Enhanced Pass-Through Commands

Currently, a typical command order used during a transaction flow might look like:

- Switch Antenna On
- Set Message LED Buzz
- Poll for Token

- Exchange APDU (Select)
- Exchange APDU (PPSE)
- Exchange APDU (Get Processing Option)
- Exchange APDU (Read Record 1.1)
- Exchange APDU (Read Record 2.1)
- Exchange APDU (Read Record 3.1)
- Exchange APDU (Read Record 3.2)
- Exchange APDU (Cryptogram)
- LED On
- Buzzer
- Switch Antenna Off

The Enhanced Pass-Through command order for the *same* transaction flow becomes:

- Enhanced Pass-Through Control
- Exchange APDU (Select)
- Exchange APDU (PPSE)
- Exchange APDU (Get Processing Option)
- Exchange APDU (Read Record 1.1)
- Exchange APDU (Read Record 2.1)
- Exchange APDU (Read Record 3.1)
- Exchange APDU (Read Record 3.2)
- Exchange APDU (Cryptogram)
- Enhanced Pass-Through Control

The first command would be formatted as follows:

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVotech2\0 | 2Ch | 0Bh | 00h | 07h | See Below | | |

| Byte 0 Single Shot | Byte 1 LCD Index | Byte 2 Beep | Byte 3 LED # | Byte 4 LED Status | Byte 5 Timeout1 | Byte 6 Timeout2 |
|-----------------------|---------------------|----------------|-----------------|----------------------|--------------------|--------------------|
| 05 | 01 | 00 | 00 | 01 | 01 | 01 |

In this first call of the Enhanced Pass-Through command:

- Byte 0 instructs the reader to single-shot the [Turn Antenna On](#) and [Poll for Token](#) commands
- Byte 1 instructs the reader to display message #1 on the display
- Byte 2 says that no buzzer is expected
- Byte 3 & 4 set the left-most LED on
- Byte 5 & 6 set the timeouts (see [Timeout1](#) and [Timeout2](#))

The second command would be formatted as follows:

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 2Ch | 0Bh | 00h | 07h | See Below | | |

| Byte 0 Single Shot | Byte 1 LCD Index | Byte 2 Beep | Byte 3 LED # | Byte 4 LED Status | Byte 5 Timeout1 | Byte 6 Timeout2 |
|-----------------------|---------------------|----------------|-----------------|----------------------|--------------------|--------------------|
| 02 | 04 | 01 | FF | 02 | 01 | 01 |

In this call of the Enhanced Pass-Through command:

- Byte 0 instructs the reader to single-shot the [Turn Antenna Off](#) command
- Byte 1 instructs the reader to display message #4 (Thank You) on the display
- Byte 2 says that give one short beep with the buzzer
- Bytes 3 & 4 set all 4 LEDs to blink on then off
- Bytes 5 & 6 set the timeouts (see [Timeout1](#) and [Timeout2](#))

7.12.6 Enhanced Poll for Token (2C-0C)

After Pass-Through Mode is started, the ViVOpay reader waits until it receives the **Poll for Token** command. This command tells the reader to start polling for a Type A or Type B PICC until a PICC is detected or a timeout occurs.

This command automatically turns the RF Antenna on.

If the reader detects a PICC within the specified time limit, it activates it and responds back to the host device with card-related data, such as the card type and serial number (UID).

If the reader does not detect a PICC within the specified time limit, it stops polling and responds back indicating that it found no card and does not return any card-related data.

1.1.3.9. Dual Application Cards

Some cards have more than one type of application stored on them. These are known as Dual Application cards. At present, all these cards have an ISO-APDU compliant application as well as a Mifare application.

To date, the only such supported dual application card is Card Type '07', supporting ISO 14443 Type A, Mifare.

For normal ViVOpay transactions (those not in Pass-Through Mode), these cards are automatically handled as ISO 14443 applications.

Note: The following behavior is only for compatibility with **Vendi**.

In Pass-Through Mode, the POS controls the polling mechanism. The POS can use a standard [Poll for Token \(2C-02\)](#), where Card Type '07' establishes a Mifare session.

Alternatively, the POS can issue an **Enhanced Poll for Token (02-0C)**, where Card Type '07' can establish an ISO 14443 session.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14,15 | Byte 16 | Byte 17 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 2Ch | 0Ch | 00h | 04h | See Table Below | | |

Enhanced Poll for Token Data Field for Command Frame

| Data Field | Length (bytes) | Description | | | | | | | | | | | | |
|-----------------------|--|--|------------|------------|-------|--|-------|---------------------------------|-------|---|-----------------------|-------------------------------|-----|------|
| Timeout1 | 1 | Time in Seconds Timeout1 cannot be zero seconds if Timeout2 is Zero. | | | | | | | | | | | | |
| Timeout2 | 1 | Multiplier for Time in multiples of 10 milliseconds. <table><tr><td>Timeout2</td><td>Time in ms</td></tr><tr><td>0</td><td>0</td></tr><tr><td>1</td><td>10</td></tr><tr><td>2</td><td>20</td></tr><tr><td>:</td><td>:</td></tr><tr><td>255</td><td>2550</td></tr></table> | Timeout2 | Time in ms | 0 | 0 | 1 | 10 | 2 | 20 | : | : | 255 | 2550 |
| Timeout2 | Time in ms | | | | | | | | | | | | | |
| 0 | 0 | | | | | | | | | | | | | |
| 1 | 10 | | | | | | | | | | | | | |
| 2 | 20 | | | | | | | | | | | | | |
| : | : | | | | | | | | | | | | | |
| 255 | 2550 | | | | | | | | | | | | | |
| Transaction Type | 2 | Initiate a transaction based upon the following masks (more than 1 can be active): <table><tr><th>Mask (hex)</th><th>Action</th></tr><tr><td>00 01</td><td>Expect Card Type '07' Force polling ISO 14443</td></tr><tr><td>00 02</td><td>PUPI Read deprecated by EMV 2.0</td></tr><tr><td>00 03</td><td>Force ISO 14443 Polling Set Single PUPI Read</td></tr><tr><td>00 04 ... 80 00</td><td>RFU – Reserved for future use</td></tr></table> | Mask (hex) | Action | 00 01 | Expect Card Type '07' Force polling ISO 14443 | 00 02 | PUPI Read deprecated by EMV 2.0 | 00 03 | Force ISO 14443 Polling Set Single PUPI Read | 00 04 ... 80 00 | RFU – Reserved for future use | | |
| Mask (hex) | Action | | | | | | | | | | | | | |
| 00 01 | Expect Card Type '07' Force polling ISO 14443 | | | | | | | | | | | | | |
| 00 02 | PUPI Read deprecated by EMV 2.0 | | | | | | | | | | | | | |
| 00 03 | Force ISO 14443 Polling Set Single PUPI Read | | | | | | | | | | | | | |
| 00 04 ... 80 00 | RFU – Reserved for future use | | | | | | | | | | | | | |

The reader uses Timeout1 and Timeout2 together to calculate the Timeout (the time to wait for a PICC).

Example:**Enhanced Poll for Token Timeout**

| Timeout1 | Timeout2 | Transaction Type | Timeout |
|----------|----------|------------------|---|
| 0 | 0 | 00 01 | Not Allowed Timeout error |
| 0 | 20 | 00 01 | 0 Seconds, 200 ms Force ISO 14443 polling |
| 0 | 50 | 00 02 | PUPI Read deprecated by EMV 2.0 |
| 0 | 100 | 00 00 | 1 Second |
| 1 | 0 | 00 04 | Not Allowed Transaction Type Error |
| 1 | 0 | 00 03 | 1 second Force ISO 14443 Polling Set Single PUPI Read |
| 1 | 20 | 00 01 | 1 second Force ISO 14443 Polling |

Multiple Transaction Types in an **Enhanced Poll for Token** command are supported. That is, it is possible to enable both Single PUPI Read and forced ISO 14443 polling simultaneously.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|----------------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | 00 - OK ?? – Fail | 00h | Variable | See Table Below | | |

The data field contains data only if the Status Code is OK (00h). For more information on the status, check the [Status Code Table](#).

Enhanced Poll for Token Data Field for Response Frame

| Data Field | Length (bytes) | Description |
|------------|-------------------|---|
| Card | 1 | Type of Card Found (or No Card Found). 00h None (Card not detected or could not activate) 01h ISO 14443 Type A (Supports ISO 14443-4 protocol) 02h ISO 14443 Type B (Supports ISO 14443-4 protocol) 03h Mifare Type A (Standard) 04h Mifare Type A (Ultralight) 05h ISO 14443 Type A (Does not support ISO 14443-4 protocol) 06h ISO 14443 Type B (Does not support ISO 14443-4 protocol) 07h ISO 14443 Type A and Mifare (NFC phone) |

| Data Field | Length (bytes) | Description |
|---------------|----------------|--|
| | | 08h Felica 09h RCTIF Type B Prim 0Ah RCTIF Type A 0Bh RCTIF Type B 0Ch Mifare Plus 0Dh Mifare Mini 0Eh Mifare DESFire 0Fh Reserved 10h ISO 18092 11h ISO 15693 12h ISO 18000 |
| Serial Number | 0 or Variable | Serial number (or the UID) of the PICC. Length depends on the card detected. If no card was detected, then a serial number is not returned. |

Note: Most cards use a 4-byte UID, so the data field of the response is five (5) bytes long. However, some cards with 7-byte UID's have entered the market (for example, ViVOcard3) and it is expected that cards with 10-byte UID's soon becomes available. All of these card types are handled by this command.

1.1.3.10. Enhanced Poll for Token Usage

Substitute the **Enhanced Poll for Token Usage** command for the standard **Poll for Token** in any transaction taking place in Pass-Through Mode. The differences in operation are noted as above; the other arguments should be identical to those of the standard **Poll for Token** that this command replaces.

Note: If an **Enhanced Poll for Token** command has set all values in the Transaction Type field set to zero, the command performs a standard **Poll for Token** instead.

7.12.7 Get ATR (2C-12)

Use the **Get ATR** command in Pass-Through mode to get the ATR the reader receives from the ICC and SAM when a Level 1 session is established.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 2Ch | 12h | 00h | 01h | Interface | | |

Command Data

| Data Item | Length (bytes) | Description/Example |
|-----------|----------------|---|
| Interface | 1 | Allowed interfaces for which to get the ATR. 20h = ICC 21h = SAM1 22h = SAM2 23h = SAM3 24h = SAM4 25h = SAM5 26h = SAM6 27h = SAM7 28h = SAM8 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 2Ch | See status code table | 00h | variable | ATR | | |

7.12.8 Contact Transaction Power Off (2C-18)

The **Contact Transaction Power Off** command can close contact transaction power. When a transaction is successful, the contact reader needs to power off before the user removes the card.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 2Ch | 18h | | | Command Data | | |

Command Data

| Data Item | Length (bytes) | Description/Example |
|-----------|----------------|---|
| Interface | 1 | 20h = ICC 21h = SAM1 22h = SAM2 23h = SAM3 24h = SAM4 25h = SAM5 26h = SAM6 27h = SAM7 |

| | | |
|--|--|------------|
| | | 28h = SAM8 |
|--|--|------------|

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | See status code table | 00h | 00h | | | |

7.12.9 Set ICC Voltage Option (2C-19)

The **Set ICC Voltage Option** sets the voltage used for the ICC slot.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 2Ch | 19h | 00h | 01h | Voltage Select | | |

ICC Voltage Select:

| Voltage | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---------|-------|-------|-------|-------|-------|-------|-------|-------|
| 5V | x | x | x | x | x | | | 1 |
| 3V | x | x | x | x | x | | 1 | |
| 1.8V | x | x | x | x | x | 1 | | |

NOTE:

- If Voltage Select value is 0, return value will contain an error code.
- Default power could be all voltages. (Pass a value of 0x07.)
- If do EMV L1 test, this value need change to 5V only. (0x01)

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | See status code table | 00 | 00 | | | |

7.12.10 Get ICC Voltage Option (2C-1A)

The **Get ICC Voltage Option** command retrieves the voltage currently used for the ICC slot.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 2Ch | 1Ah | 00h | 00h | N/A | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|---|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | See status code table | 00 | 01 | Voltage Select (see previous table) | | |

7.12.11 Set ICC Reader Type Option (2C-1B)

The **Set ICC Reader Type Option** command sets the reader type for the ICC slot.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|--------------------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 2Ch | 1Bh | 00h | 01h | Reader Type Select (see below) | | |

ICC Reader Type Select:

| ICC Reader Type | Value |
|-----------------|-------|
| EMV | 0 |
| ISO7816 | 1 |

Note: Default value is EMV (0).

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 2Ch | See status code table | 00 | 00 | N/A | | |

7.12.12 Get ICC Reader Type Option (2C-1C)

The **Get ICC Reader Type Option** command retrieves the currently reader type set for the ICC slot.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 2Ch | 1Ch | 00h | 00h | N/A | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|---|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 2Ch | See status code table | 00 | 01 | ICC Reader Type (0 for EMV, 1 for ISO-7816) | | |

7.12.13 Antenna Control (28-01)

The **Antenna Control** command turns the RF Antenna ON or OFF.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|---------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 28h | 01h | 00h | 01h | Mode | | |

Mode:

0 = Disable RF Antenna

1 = Enable RF Antenna

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|----------------------------------|---------|---|-------------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 28h | See status code table | 00h | 00h | | |

Warnings for the Antenna Off command:

- Turning off the antenna deactivates the RF field and returns the card to a Power Off state, terminating any existing connection. A new "Poll for Token" command would need to be issued to establish a new conversation with a card. Turning the antenna off and then turning it back on is a useful way to reset the card to its Idle State where it will respond to polling commands.
- When exiting Pass-Through Mode, if the reader is returning to Auto Poll mode, it is advised to issue an "Antenna Control Enable RF Antenna" command before exiting to ensure the antenna is in the right state.

7.13 Pass-through UI Control

7.13.1 LED Control (0A-02)

The **LED Control** command switches the specified LEDs on the ViVOpay reader off or on *only* when the reader is in Pass-Through Mode.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14,15 | Byte 16 | Byte 17 |
|-------------------------------|---------|-------------|-------------------|-------------------|------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 0Ah | 02h | 00h | 02h | below | | |

LED Control Data Field

| Data Field | Length (bytes) | Description |
|------------|----------------|--|
| LED# | 1 | 00h: LED 0 (Power LED) 01h: LED 1 02h: LED 2 03h: LED 3 FFh: All 4 LEDs Where the LEDs are numbered 0, 1, 2, 3 counting from the left. Note: If you are using Pass-Through Mode to control the Power LED (LED 0), it is your responsibility to make sure that it behaves correctly. |
| LED Status | 1 | 00h: LED Off 01h: LED On 02h: Green Color (Tri-Color LED) 03h: Red Color (Tri-Color LED) 04h: Amber Color (Tri-Color LED) |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 0Ah | See status code table | 00h | 00h | | |

7.13.2 Front LED Control (0A-03)

The **Front LED Control** command controls the front LED of the VP5300.

Note: This command can only be performed while in Pass-Through Mode.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 - Byte14+n-1 | Byte 14+n | Byte 15+n |
|----------------------------------|---------|-----------------|-------------------------|-------------------------|-------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 0Ah | 03h | 00h | 03h | Data Field | | |

| Data Item | Length (bytes) | Description |
|--------------|-------------------|--|
| LED Location | 1 | LED Location Selection - 0x00 - Back (not supported) - 0x01 - Front (always selected) |
| Red LED | 1 | Red LED Control Option - 0x00 – OFF - 0x01 – ON (default) - 0x02 – BLINK (500ms interval) |
| Green LED | 1 | Green LED Control Option - 0x00 – OFF (default) - 0x01 – ON - 0x02 – BLINK (500ms interval) |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|----------------------------------|---------|---------------------------------------|-------------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 0Ah | See status code table | 00h | 00h | | |

7.13.3 Buzzer Control (0B-xx)

The **BuzzerControl** command sounds the ViVOPay reader's buzzer.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 0Bh | See Below | 00h | 01h | Buzzer Parameter | | |

Sub-Command

- 01h: N short beeps
- 02h: Single long beep of specified duration

Buzzer Control Data Field

| Data Field | Length (bytes) | Description |
|------------------|----------------|---|
| Buzzer Parameter | 1 | If Sub-Command is Short Beeps ... Num Beeps = 01h One Short Beep = 02h Two Short Beeps = 03h Three Short Beeps = 04h Four Short Beeps If Sub-Command is Long Beep ... Duration = 00h 200 ms = 01h 400 ms = 02h 600 ms = 03h 800 ms |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 0Bh | See status code table | 00h | 00h | | |

7.14 Pass-through Data Exchange

7.14.1 Exchange Contactless Data (2C-03)

The **Exchange Contactless Data** command allows the host device to send, via the ViVOpay reader, application-level APDUs to a PICC that supports ISO 14443-4 Protocol. The reader sends the PICC response back to the host.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 2Ch | 03h | Variable | Variable | APDU Out | | |

APDU Out is the complete APDU that is to be sent to the PICC. The contents of the APDU depend on the application residing on the PICC and are out of the scope of this document.

If a valid command frame is received from the host device, the reader sends the APDU data to the PICC and receives its response. The reader treats the PICC response as unknown data and does not try to interpret it. If the operation was successful, the reader returns a response frame with an OK status and the response received from the PICC (APDU response).

If the command frame contains any errors, or an error occurs during communication with the PICC, the reader sends a response frame with an appropriate status and no data.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---------------------------------------|-------------------------|-------------------------|-----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | See status code table | Variable | Variable | APDU response Or None | | |

The data field contains data only if the status code is OK. In this case, the data consists of "APDU Response"—the response data received from the PICC. The contents of the response depend on the application residing on the PICC and are out of the scope of this document.

For SRED readers, the APDU data the reader receives from the card or device is checked for sensitive data elements using rules defined in "Secure Pass-Through Function." If the reader finds sensitive data elements, and the transaction does not belong to an AID or BIN on the Whitelist, the command returns a Parameter Not Supported error (0x06).

After enabling Output Encrypted R-APDU:

- If the Encryption Key is DEK, the data response from the PICC skips checking for sensitive data elements and follows Encrypted R-APDU Format rules.
- If the Encryption Key is TransArmor RSA Cert, the reader disables transactions and sends a **0x0A** response status code.

Encrypted R-APDU Format Rules:

<Encrypted R-APDU Length (MSB)> < Encrypted R-APDU Length (LSB)> <n bytes Encrypted R-APDU without Status Bytes> <SW1 SW2 > <DEK KSN Length (MSB)> < DEK KSN Length (LSB)> <DEK KSN> <MAC Value Length (MSB)> <MAC Value Length (LSB)> [<MAC Value>]

Where:

- <Encrypted R-APDU Length (MSB)> < Encrypted R-APDU Length (LSB)>: is length of <n bytes Encrypted R-APDU without Status Bytes>
 - if R-APDU is only SW1 SW2, it does not exist.
- <n bytes Encrypted R-APDU without Status Bytes>
 - if R-APDU is only SW1 SW2, it does not exist.
 - The format of n bytes Raw R-APDU without Status Bytes as below: 2 bytes valid R-APDU Length (MSB, LSB) + x bytes valid Raw R-APDU without Status Bytes + y bytes Padding.

Note:

- For TDES mode: 2+x should be a multiple of 8. If it is not a multiple of 8, the reader should pad with zero by y bytes automatically (2+x+y should be multiple of 8).
- For AES mode: 2+x should be a multiple of 16. If it is not a multiple of 16, the reader should pad with by zero y bytes automatically (2+x+y should be multiple of 16).
- SW1 SW2 (2 bytes)
- <DEK KSN Length (MSB)> < DEK KSN Length (LSB)>:
 - if R-APDU is only SW1 SW2, it does not exist.
- <DEK KSN>
 - if R-APDU is only SW1 SW2, it does not exist.
- <MAC Value Length (MSB)> <MAC Value Length (LSB)>:
 - If R-APDU is only SW1 SW2, it does not exist.
 - If there is encrypted R-APDU:
 - if MAC Verification Data is Switch On, it is MAC Value length.
 - If MAC Verification Data is Switch Off, it is <0x00> <0x00>.
- <MAC Value>:
 - if R-APDU is only SW1 SW2, it does not exist.
 - If there is encrypted R-APDU:
 - if MAC Verification Data is Switch On, it is MAC Value.
 - If MAC Verification Data is Switch Off, it does not exist.

7.14.2 PCD Single Command Exchange (2C-04) Protocol 2

The **PCD Single Command Exchange** command allows the host device to send, via the ViVOpay reader, raw data to an ISO 14443 PICC that does not support ISO 14443-4 Protocol (such as Mifare Standard or Mifare Ultralight). The reader sends the PICC response back to the host.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 2Ch | 04h | Variable | Variable | See Table below | | |

PCD Single Command Exchange Data Field Protocol 2

| Data Field | Length (bytes) | Description |
|-----------------------------|-------------------|---|
| PCD Command | 1 | The command sent to the PCD Reader IC on the ViVOpay board. It tells the PCD what to do with the data sent with the command. The PCD commands supported and their values are given in the "PCD Cmd" table below |
| PCD Timeout | 4 | The RF communication timeout in ETUs stored as a 4-byte big-endian number, where 1 ETU is 9.44 microseconds. The RF communication timeout guards the communication between the PCD reader IC and the PICC Card. The timeout is measured between the last bit sent to the PICC and the first bit received from the PICC. |
| PCD Command Flags | 1 | These flags allow greater control over the way the reader processes the command via the PCD Reader IC. Format of the PCD Command Flags byte is given in the "PCD Command Flags" table below. |
| Channel Redundancy Register | 1 | This value tells the PCD what data integrity checks to perform during communication with the PICC Card. The checks to perform at each stage are defined by the protocol (14443 Type A or B). The format of the PCD Command Flags byte is given in the "Channel Redundancy Register" Table below. |
| Raw Data Out | Variable | Raw data that is sent to the PICC or to the PCD. |

The command frame contains some PCD parameters and raw data. The PCD command parameter is used by the reader to determine what PCD function to carry out. The raw data is sent to the PICC for the Transceive command or is used for LoadKey/Authentication. The contents of the data depend on the PICC and PCD and are out of the scope of this document.

PCD Commands Protocol 2

| PCD Command | Value | Description |
|-------------------|-------|--|
| PCD LOADKEY | 19h | Used for Loading Mifare Key into PCD for Authentication |
| PCD AUTHENTICATE1 | 0Ch | Used for PCD-based Mifare Authentication. This command results in both Level 1 and Level 2 authentication being performed automatically. |
| PCD TRANSCEIVE | 1Eh | Used to Send/Receive raw Data to/from the PICC |

Note: The PCD LOADKEY and PCD AUTHENTICATE1 functions may also be performed by the host device directly by using the PCD Transceive Command.

PCD Command Flags Table

| Bit# | Flag | Value | Meaning |
|------|---------------------------------------|-------------|---|
| 0 | Disable DF (DF=Disturbance Filter) | 1 | When response from PICC Card has been received, end of response is signaled regardless of errors. |
| | | 0 | When response from PICC Card has been received, if there are errors then the data received is flushed and we continue to receive. If there are no errors then end of response is signaled. |
| 1 | Flush FIFO | 1 | PCD FIFO is flushed before starting this PCD command. |
| | | 0 | PCD FIFO is not flushed before starting this command. |
| 2-4 | TxLastBits | 000 ~111 | Used for transmission of bit oriented frames: TxLastBits defines the number of bits of the last byte that shall be transmitted. A 000 indicates that all bits of the last byte shall be transmitted. After transmission, TxLastBits is cleared automatically. |
| 5 | DISABLE RXMULTIPLE FILTER | 1 | Disable |
| | | 0 | Enable(Default) |
| 6-7 | RFU | 00 | Reserved for future use. |

Note: The setting in Bit 5 does not work; the bit is reserved for other use.

PCD Channel Redundancy Register Protocol 2

| Bit# | Flag | Value | Meaning |
|------|---------------|-------|--|
| 0 | Parity Enable | 1 | Parity bit inserted in transmitted data and expected in received data. |
| | | 0 | No parity bit inserted or expected. |
| 1 | Parity Odd | 1 | Odd parity. |
| | | 0 | Even parity. |
| 2 | Tx CRC Enable | 1 | CRC Bytes appended to transmitted data. |
| | | 0 | CRC Bytes not appended to transmitted data. |

| Bit# | Flag | Value | Meaning |
|------|---------------|-------|--|
| 3 | Rx CRC Enable | 1 | Last bytes of received data are interpreted as CRC bytes. Note: The CRC is not sent back to the host by ViVOpay. |
| | | 0 | No CRC expected. |
| 4 | CRC-8 | 1 | 8-Bit CRC calculated. |
| | | 0 | 16-Bit CRC calculated. |
| 5 | CRC 3309 | 1 | CRC-Calculation is done according to ISO /IEC3309 (ISO 14443B). |
| | | 0 | CRC-Calculation is done according to ISO 14443A. |
| 6 | RFU | 0 | Must always be zero. |
| 7 | RFU | 0 | Must always be zero. |

If the reader receives a valid command frame from the host device, it sends the data to the PICC (or carries out the appropriate action) and receives the PICC response. The reader treats the response as unknown data and does not try to interpret it. If there is no error, the reader returns a response frame with OK status and the data received from the PICC (if any). The response frame also contains the result of the PCD command (PCD Status). The PCD status may indicate success or an error code.

If the command frame contains any errors, or an error occurred during communication with the PICC (such as PICC removed from the field), the reader sends a response frame with an appropriate status and no data.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|-------------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTECH2\0 | 2Ch | See status code table | Variable | Variable | See Table below OR None | | |

PCD Single Command Exchange Data Field for Response

| Data Field | Length (bytes) | Description | | | | | | | | | | | | | | |
|------------|---------------------------------|--|------------|-------------|---|----|-----------|---------------------------------|-----------|-----------|-----------|-------|-----------|----------------------|-----------|--------------|
| PCD Status | 1 | <p>This field contains the result of the PCD Command. Possible values are given in the following table.</p> <table><tr><th>PCD Status</th><th>Description</th></tr><tr><td>0</td><td>OK</td></tr><tr><td>-1 (0xFF)</td><td>No Tag Error (Card not present)</td></tr><tr><td>-2 (0xFE)</td><td>CRC Error</td></tr><tr><td>-3 (0xFD)</td><td>Empty</td></tr><tr><td>-4 (0xFC)</td><td>Authentication Error</td></tr><tr><td>-5 (0xFB)</td><td>Parity Error</td></tr></table> | PCD Status | Description | 0 | OK | -1 (0xFF) | No Tag Error (Card not present) | -2 (0xFE) | CRC Error | -3 (0xFD) | Empty | -4 (0xFC) | Authentication Error | -5 (0xFB) | Parity Error |
| PCD Status | Description | | | | | | | | | | | | | | | |
| 0 | OK | | | | | | | | | | | | | | | |
| -1 (0xFF) | No Tag Error (Card not present) | | | | | | | | | | | | | | | |
| -2 (0xFE) | CRC Error | | | | | | | | | | | | | | | |
| -3 (0xFD) | Empty | | | | | | | | | | | | | | | |
| -4 (0xFC) | Authentication Error | | | | | | | | | | | | | | | |
| -5 (0xFB) | Parity Error | | | | | | | | | | | | | | | |

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------|--|---|-----------|------------|-----------|-----------------|-----------|---------------------|-----------|-----------|------------|--|------------|-----------------|------------|------------------|------------|------|------------|-------------|------------|------------|------------|---------------------|------------|---------------|------------|--------------|------------|-----------------|------------|-----------------|------------|-------------|------------|----------------|------------|--------------|------------|--------------------------------|-------------|-------------------------|
| | | <table><tr><td>-6 (0xFA)</td><td>Code Error</td></tr><tr><td>-7 (0xF9)</td><td>Card Type Error</td></tr><tr><td>-8 (0xF8)</td><td>Serial Number Error</td></tr><tr><td>-9 (0xF7)</td><td>Key Error</td></tr><tr><td>-10 (0xF6)</td><td>Authentication not carried out for this Sector</td></tr><tr><td>-11 (0xF5)</td><td>Bit Count Error</td></tr><tr><td>-12 (0xF4)</td><td>Byte Count Error</td></tr><tr><td>-13 (0xF3)</td><td>Idle</td></tr><tr><td>-15 (0xF1)</td><td>Write Error</td></tr><tr><td>-18 (0xED)</td><td>Read Error</td></tr><tr><td>-19 (0xEC)</td><td>FIFO Overflow Error</td></tr><tr><td>-21 (0xEB)</td><td>Framing Error</td></tr><tr><td>-22 (0xEA)</td><td>Access Error</td></tr><tr><td>-23 (0xE9)</td><td>Unknown Command</td></tr><tr><td>-24 (0xE8)</td><td>Collision Error</td></tr><tr><td>-25 (0xE7)</td><td>Reset Error</td></tr><tr><td>-27 (0xE5)</td><td>Access Timeout</td></tr><tr><td>-31 (0xE1)</td><td>Coding Error</td></tr><tr><td>-54 (0xCA)</td><td>Baud rate not supported by PCD</td></tr><tr><td>-112 (0x90)</td><td>Receive Buffer Overflow</td></tr></table> | -6 (0xFA) | Code Error | -7 (0xF9) | Card Type Error | -8 (0xF8) | Serial Number Error | -9 (0xF7) | Key Error | -10 (0xF6) | Authentication not carried out for this Sector | -11 (0xF5) | Bit Count Error | -12 (0xF4) | Byte Count Error | -13 (0xF3) | Idle | -15 (0xF1) | Write Error | -18 (0xED) | Read Error | -19 (0xEC) | FIFO Overflow Error | -21 (0xEB) | Framing Error | -22 (0xEA) | Access Error | -23 (0xE9) | Unknown Command | -24 (0xE8) | Collision Error | -25 (0xE7) | Reset Error | -27 (0xE5) | Access Timeout | -31 (0xE1) | Coding Error | -54 (0xCA) | Baud rate not supported by PCD | -112 (0x90) | Receive Buffer Overflow |
| -6 (0xFA) | Code Error | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -7 (0xF9) | Card Type Error | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -8 (0xF8) | Serial Number Error | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -9 (0xF7) | Key Error | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -10 (0xF6) | Authentication not carried out for this Sector | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -11 (0xF5) | Bit Count Error | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -12 (0xF4) | Byte Count Error | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -13 (0xF3) | Idle | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -15 (0xF1) | Write Error | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -18 (0xED) | Read Error | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -19 (0xEC) | FIFO Overflow Error | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -21 (0xEB) | Framing Error | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -22 (0xEA) | Access Error | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -23 (0xE9) | Unknown Command | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -24 (0xE8) | Collision Error | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -25 (0xE7) | Reset Error | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -27 (0xE5) | Access Timeout | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -31 (0xE1) | Coding Error | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -54 (0xCA) | Baud rate not supported by PCD | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -112 (0x90) | Receive Buffer Overflow | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| RcvdBits | 4 | Number of bits received (stored as a big-endian number) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Raw Data In | 0 or Variable | The response data that is received from the PICC. The contents of the response depend on the application residing on the PICC and are out of the scope of this document. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

For SRED readers, the reader checks the raw data received from the card or device for sensitive data elements using rules defined in "Secure Pass-Through Function." If the reader finds any sensitive data, and the transaction does not belong to an AID or BIN on the Whitelist, the command returns a Parameter Not Supported error (0x06).

After enabling Output Encrypted R-APDU:

- If the Encryption Key is DEK, and if the command is "PCD TRANSCEIVE", the data response from the PICC skips checking for sensitive data elements and follows the Encrypted R-APDU Format rules.
- If the Encryption Key is TransArmor RSA Cert, the reader disables transactions and sends a 0x0A response status code.

Encrypted R-APDU Format Rules:

- <Encrypted R-APDU Length (MSB)> <Encrypted R-APDU Length (LSB)> <n bytes Encrypted R-APDU> <DEK KSN Length (MSB)> <DEK KSN Length (LSB)> <DEK KSN> <MAC Value Length (MSB)> <MAC Value Length (LSB)> [<MAC Value>]
- Where:
- <Encrypted R-APDU Length (MSB)><Encrypted R-APDU Length (LSB)>: is length of <n bytes Encrypted R-APDU>
- <n bytes Encrypted R-APDU>

- The format of n bytes Raw R-APDU as below: 2 bytes valid R-APDU Length (MSB, LSB) + x bytes valid Raw R-APDU + y bytes Padding.

Note:


- For TDES mode: 2+x should be a multiple of 8. If it is not a multiple of 8, the reader should pad with zero by y bytes automatically (2+x+y should be multiple of 8).
- For AES mode: 2+x should be a multiple of 16. If it is not a multiple of 16, the reader should pad with zero by y bytes automatically (2+x+y should be multiple of 16).
- <DEK KSN Length (MSB)> < DEK KSN Length (LSB)>:
- <DEK KSN>
- <MAC Value Length (MSB)> <MAC Value Length (LSB)>:
 - If there is encrypted R-APDU:
 - if MAC Verification Data is Switch On, it is MAC Value length.
 - If MAC Verification Data is Switch Off, it is <0x00> <0x00>.
- <MAC Value>:
 - If there is encrypted R-APDU:
 - if MAC Verification Data is Switch On, it is MAC Value.
 - If MAC Verification Data is Switch Off, it does not exist.

1.1.3.11. Example: Sending a HALT Command to a Type A PICC

Assuming that the ViVOpay reader is in Pass-Through Mode, the Poll for Token command detects a Type A PICC, and the host application has completed the transaction with the card, the reader can send an ISO 14443 HALT to the PICC using the [PCD Single Command Exchange](#) command. The command log below describes the command and data that the host sends to the reader and the responses the reader may return.

A host device or PC and a ViVOpay reader may exchange the following serial data:

Halt Command Exchange Between Host and Reader

| Terminal | ViVOpay |
|--|--|
| Command Frame ("PCD Single Command Exchange", PcdTransceive, 106 ETUs, [FlushFIFO=0, DisableDF=0], ChanRedReg=07)  "ViVOTech2\0" 2Ch 04h 00h 09h 1Eh 00h 00h 00h 6Ah 00h 07h 50h 00h <CRC><CRC> | Response Frame (OK, NoTagError, RcvdBits=0) "ViVOTech2\0" 2Ch 00h 00h 05h FFh 00h 00h 00h 00h <CRC><CRC> |

7.14.3 Exchange APDU Data (2C-13)

The **Exchange APDU Data** command allows exchange of application level APDUs with the following:

- PICC (contactless transaction) that is ISO 14443-4 compliant
- SAMs
- Contact ICC

An application-level Command APDU meant for a card or SAM is sent to the reader in the command frame, along with the interface. The reader sends the APDU to the card/SAM. The response APDU received from the card/SAM is sent back by the reader in the response frame.

Before using this pass-through command, establish a Level 1 session with a card on the interface to be used, such as contactless (PICC) or contact (ICC, SAMs).

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVotech2\0 | 2Ch | 13h | Variable | | See Command Data Table | | |

Command Data

| Data Item | Length (bytes) | Description/Example |
|--------------|----------------|--|
| Interface | 1 | Allowed interfaces for which to get the ATR. 00h = Contactless 20h = ICC 21h = SAM1 22h = SAM2 23h = SAM3 24h = SAM4 25h = SAM5 26h = SAM6 27h = SAM7 28h = SAM8 |
| Command APDU | Variable | Command APDU data that will be sent to the card via the specified interface. For SAMs, any command/response pair can be passed. |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 2Ch | See status code table | 00h | variable | Response APDU | | |

For SRED readers, if the interface is Contactless, the reader checks the APDU data received from the card or reader for sensitive data elements using rules defined in "Secure Pass-Through Function." If the reader finds any sensitive data, and the transaction does not belong to an AID or BIN on the Whitelist, the command returns a Parameter Not Supported error (0x06).

After enabling Encrypted R-APDU Output:

- If the Encryption Key is DEK, the data response from the PICC skips checking for sensitive data elements and follows the Encrypted R-APDU Format rules.
- If the Encryption Key is TransArmor RSA Cert, the reader disables transactions and sends a 0x0A response status code.

Encrypted R-APDU Format Rules:

p<Encrypted R-APDU Length (MSB)> < Encrypted R-APDU Length (LSB)> <n bytes Encrypted R-APDU without Status Bytes> <SW1 SW2 > <DEK KSN Length (MSB)> < DEK KSN Length (LSB)> <DEK KSN> <MAC Value Length (MSB)> <MAC Value Length (LSB)> [<MAC Value>]

Where:

- <Encrypted R-APDU Length (MSB)>< Encrypted R-APDU Length (LSB)>: is length of <n bytes Encrypted R-APDU without Status Bytes>
 - if R-APDU is only SW1 SW2, it does not exist.
- <n bytes Encrypted R-APDU without Status Bytes>
 - if R-APDU is only SW1 SW2, it does not exist.
 - The format of n bytes Raw R-APDU without Status Bytes as below: 2 bytes valid R-APDU Length (MSB, LSB) + x bytes valid Raw R-APDU without Status Bytes + y bytes Padding.

Note:

- For TDES mode: 2+x should be a multiple of 8. If it is not a multiple of 8, the reader should pad with zero by y bytes automatically (2+x+y should be multiple of 8).
- For AES mode: 2+x should be a multiple of 16. If it is not a multiple of 16, the reader should pad with zero by y bytes automatically (2+x+y should be multiple of 16).
- SW1 SW2 (2 bytes)
- <DEK KSN Length (MSB)> < DEK KSN Length (LSB)>:

- if R-APDU is only SW1 SW2, it does not exist.
- <DEK KSN>
 - if R-APDU is only SW1 SW2, it does not exist.
- <MAC Value Length (MSB)> <MAC Value Length (LSB)>:
 - If R-APDU is only SW1 SW2, it does not exist.
 - If there is encrypted R-APDU:
 - if MAC Verification Data is Switch On, it is MAC Value length.
 - If MAC Verification Data is Switch Off, it is <0x00> <0x00>.
- <MAC Value>:
 - if R-APDU is only SW1 SW2, it does not exist.
 - If there is encrypted R-APDU:
 - if MAC Verification Data is Switch On, it is MAC Value.
 - If MAC Verification Data is Switch Off, it does not exist.

7.15 High-level Pass-Through Commands for Mifare Cards

This section contains serial commands that implement higher level functionality for Mifare cards. These commands require a reader to be in Pass-Through mode and that the **Poll for Token** command indicates that a Mifare Card is present. These commands do not work for non-Mifare cards.

7.15.1 Mifare Authenticate Block (2C-06)

The **Mifare Authenticate Block** command allows the host device to instruct the ViVOpay reader to authenticate the Mifare Card sector containing the specified block of data. The host device also specifies the key to use. This command only applies to Mifare Standard/Classic Cards.

This command is not applicable for Mifare Ultralight Cards.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-Byte 21 | Byte 22 | Byte 23 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 2Ch | 06h | 00h | 08h | See Table below | | |

Mifare Authentication Block Data Field

| Data Field | Length (bytes) | Description |
|------------|----------------|---|
| Block | 1 | Block Number in the Mifare Card for which the relevant sector must be authenticated. |
| Key Type | 1 | Specifies which type of key to use for authentication. It can have the following values. 01h: Key A 02h: Key B |
| Key | 6 | Value of the Key |

For details on these fields, refer to the relevant Mifare Specifications.

This command does not actually perform authentication. It sets the key to use for the subsequent authentication. The actual authentication is performed before the next read or write operation. If a sector boundary is crossed, the reader will attempt to authenticate using the key established with this command.

After receiving the **Command Frame**, the reader verifies the data, and if the data is valid interacts with the Mifare card to authenticate the sector containing the specified block. If this operation succeeds, the reader returns a response frame with an OK Status. If the operation fails or the data was invalid, the reader returns a response frame with an appropriate status.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | See status code table | 00h | 00h | | |

If the card in the field is not a Mifare card, the reader returns a 0Ch (sub-command not allowed) status code.

7.15.2 Mifare Read Blocks (2C-07)

The **Mifare Read Blocks** command reads data from one or more blocks on the Mifare Card. The host device can instruct the reader to read up to 15 blocks using this command. If more than one block is defined, then the reader automatically reads the starting block and the blocks that follow. The sector trailer is skipped for multi-block reads. Sector trailers may be read with a single block read (except the keys will not be visible).

If the specified card is a Mifare Standard card, the host device must successfully send at least one [Mifare Authenticate Block](#) command to the reader for the first block to read. This does not authenticate the block; it stores a key for the reader to use as it performs reads and writes.

If the card specified is a Mifare Standard card and the read command specifies a single block read, the reader tries to read the data regardless of whether or not the block is a sector trailer block.

If the card specified is a Mifare Standard card and the read is a multi-block read, the reader skips reading the sector trailer blocks that contain keys (because the keys cannot be read).

The block count does not include skipped blocks. While reading blocks in a Mifare Standard Card, if the read requires access to the next sector, the reader carries out authentication for this block/sector automatically by using the key type and key value set in the **Mifare Authenticate Block** command to authenticate the sector for the Starting Block.

Block reads and writes that span multiple sectors assume that the keys to authenticate those sectors are the same as the one that was set using the **Mifare Authenticate Block** command.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14, 15 | Byte 16 | Byte 17 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 2Ch | 07h | 00h | 02h | See Table below | | |

Mifare Read Block Data Field

| Data Field | Length (bytes) | Description |
|--------------------|----------------|---|
| Card & Block Count | 1 | <p>Card Type: [Bit 7..4] This can only indicate the following cards:</p> <ul style="list-style-type: none"> • Mifare Type A (Standard) • Mifare Type A (Ultralight) <p>The values for these card types are defined in the Poll for Token command (consider only the lower 4 bits).</p> <p>Block Count: [Bit 3..0] This is the number of 16-byte blocks that are read. The Block Count cannot be greater than 15. This count does not include the skipped blocks if the card is a Mifare Standard card.</p> |
| Start Block | 1 | This is the card block number from which the reader starts reading. |

After receiving the command frame, the reader verifies the parameters. If the parameters are valid, it reads the data from the card. If this operation succeeds, the reader sends a response frame containing an OK status and the read data. If the operation fails or one or more parameters were invalid, the reader sends a response frame containing an appropriate status and no data.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | See status code table | Variable | Variable | Data Read from Card OR None | | |

If the Status is OK, then the Data Length depends on the number of blocks read and the card type.
 $\text{DataLen} = \text{Blocks Read} * (\text{Bytes per Block for Card})$.

If there was an error or no data was read, the Data Length is zero.

For SRED readers, the data received from the card or reader by the reader is checked for sensitive data elements using rules described in [Secure Pass-Through Function](#). If the reader finds sensitive data, and has not established that the transaction belongs to an AID or BIN on the Whitelist, the command returns a Parameter Not Supported error (0x06).

After enabling Encrypted R-APDU Output:

- If the Encryption Key is DEK, the data response from the PICC skips checking for sensitive data elements and follows the Encrypted R-APDU Format rules.
- If the Encryption Key is TransArmor RSA Cert, the reader disables transactions and sends a 0x0A response status code.

Encrypted R-APDU Format Rules:

<Encrypted R-APDU Length (MSB)> < Encrypted R-APDU Length (LSB)> <n bytes Encrypted R-APDU> <DEK KSN Length (MSB)> < DEK KSN Length (LSB)> <DEK KSN> <MAC Value Length (MSB)> <MAC Value Length (LSB)> [<MAC Value>]

Where:

- <Encrypted R-APDU Length (MSB)>< Encrypted R-APDU Length (LSB)>: is length of <n bytes Encrypted R-APDU>
- <n bytes Encrypted R-APDU>
 - The format of n bytes Raw R-APDU as below: 2 bytes valid R-APDU Length (MSB, LSB) + x bytes valid Raw R-APDU + y bytes Padding.

Note:

- For TDES mode: 2+x should be a multiple of 8. If it is not a multiple of 8, the reader should pad with zero by y bytes automatically (2+x+y should be multiple of 8).
- For AES mode: 2+x should be a multiple of 16. If it is not a multiple of 16, the reader should pad with zero by y bytes automatically (2+x+y should be multiple of 16).
- <DEK KSN Length (MSB)> < DEK KSN Length (LSB)>:
- <DEK KSN>
- <MAC Value Length (MSB)> <MAC Value Length (LSB)>:
 - If there is encrypted R-APDU:
 - if MAC Verification Data is Switch On, it is MAC Value length.
 - If MAC Verification Data is Switch Off, it is <0x00> <0x00>.
- <MAC Value>:
 - If there is encrypted R-APDU:
 - if MAC Verification Data is Switch On, it is MAC Value.
 - If MAC Verification Data is Switch Off, it does not exist.

1.1.3.12. Reading Mifare Ultralight Cards

Mifare Ultralight cards differ from Mifare Cards. The Mifare Standard Card block size is 16 bytes. However, the Mifare Ultralight Cards block (page) size is 4 bytes. When reading Mifare Ultralight cards, *Block Count* is taken to mean the number of 16 byte blocks (each consisting of four 4-byte pages). However, for Mifare Ultralight cards, the *Start Block* represents a 4-byte page.

For example, if the card is a Mifare Ultralight card, and the host device requests a read starting at Block 3 and the Block Count is 1, the reader returns 16 bytes of data, consisting of Page #3, 4, 5, and 6. If the host device requests a read starting at Block 3 and the Block Count is 2, the reader returns $16 \times 2 = 32$ bytes of data, consisting of Page #3, 4, 5, 6, 7, 8, 9, and 10.

Typically, Mifare Ultralight Cards are read by the reader, but are not written because they are typically used for disposable applications such as ticketing.

7.15.3 Mifare Write Blocks (2C-08)

The **Mifare Write Blocks** command instructs a ViVOpay reader to write data to one or more blocks on the Mifare Card. The host device can instruct the reader to write up to 15 blocks of data using this command. If more than one block is defined, then the reader automatically writes to the starting block and the blocks that follow.

The block size depends on the type of Mifare card being accessed. The Mifare Standard Cards block size is 16 bytes. The Mifare Ultralight Cards block size is 4 bytes.

If the specified card is a Mifare Standard card, the host device must successfully send at least one [Mifare Authenticate Block](#) command to the reader for the first block to write. This does not authenticate the block; it stores a key the reader uses as it performs reads and writes.

If the specified card is a Mifare Standard card and the write command is a single block write, the reader tries to write the data regardless of whether or not the block is a sector trailer block.

If the specified card is a Mifare Standard card and the write is a multi-block write, the reader skips writing to the sector trailer blocks that contain the keys. The block count does not include skipped blocks. While writing blocks to a Mifare Standard Card, if the write requires access to the next sector, the reader carries out authentication for the block/sector automatically by using the key type and key value used by the host device to authenticate the sector for the Starting Block via the Mifare Authenticate Block command.

Block reads and writes that span multiple sectors assume that the keys to authenticate those sectors are the same as the one that was set using the **Mifare Authenticate Block** command.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 2Ch | 08h | Variable | Variable | See Table below | | |

Mifare Write Block Data Field

| Data Field | Length (bytes) | Description |
|--------------------|---|--|
| Card & Block Count | 1 | <p>Card Type: [Bit 7..4] This can only indicate the following cards:</p> <ul style="list-style-type: none"> • Mifare Type A (Standard) • Mifare Type A (Ultralight) <p>The values for these card types are defined in the "Poll for Token" command (consider only the lower 4 bits). Block Count: [Bit 3..0] This is the number of blocks that are written. The Block Count cannot be greater than 15. This count does not include the skipped blocks if the card is a Mifare Standard card.</p> |
| Start Block | 1 | This is the card block number from which the reader starts writing. |
| Data to Write | Variable (multiple of block size) | <p>Data to write to the Card.</p> <p>The length of the data to be written to the card depends on the number of blocks to be written and the card type.</p> |

After receiving the command frame the reader verifies the parameters. If the parameters are valid, it writes the data to the card. If this operation is successful, the reader sends a response frame with an OK status.

If the command frame is invalid or the write operation fails, the reader sends a response frame with an appropriate status.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | 2Ch | See status code table | 00h | 00h | | |

7.15.4 Mifare ePurse Command (2C-0A)

The **Mifare ePurse** command instructs the reader to carry out Debit, Credit, and Backup operations on value blocks in a Mifare card. These functions require that the related data blocks are formatted as a value block and that operations and keys used match the defined Access Conditions for that sector.

The following illustration shows the format of a value block:

ePurse Value Block Format

| Byte Number | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|-------------|-------|---|---|---|-------|---|---|---|-------|---|----|----|-----|-----|-----|-----|
| Description | Value | | | | Value | | | | Value | | | | Adr | Adr | Adr | Adr |

A Debit function subtracts a given amount from a Mifare value block and stores the result in the same block. A Credit function adds a given amount to a Mifare value block and stores the result in the same block. A Backup function reads a value block and stores a copy of it in another value block in the same authenticated sector.

This command is flexible in that it allows any number of Debit, Credit, or Backup function blocks to be embedded within one command frame in any order, with or without keys specified, as long as the total number of bytes is within the size capability of one Pass-Through command. Operations are performed in the order they are specified.

For instance, a **Mifare ePurse** command could simply contain one Debit function to debit a value block by a specified amount. If a key and key type is included they are used to authenticate the block and the debit function is performed. If no key information is included the key and key type used in the previous Mifare Authentication command is used.

In another case, the **Mifare ePurse** command could contain a Credit function to credit a value block by a specific amount and a Backup function to backup the resulting balance to another value block somewhere on the card. Each command could include a specific key for the block being addressed, or omit the key information and let the reader use the last known key.

Note: The default key and key type are overwritten each time a key is encountered while processing a **Mifare ePurse** command. The initial default values are those set when the [Mifare Authenticate Block](#) command is received. That key type and key are used until another key is encountered, at which point the new key becomes the default key for subsequent transactions. If purse commands are used without key information then the host device must have successfully sent at least one Mifare Authenticate Block command to the reader for the first block.

Warning: Multiple **Mifare ePurse** command blocks can be included in one command; each command contains a count of the number of command blocks included in the command.

If the count of command blocks specified is not equal to the actual number of command blocks included in the command, an error may or may not be returned to the user.

If the count of command blocks is greater than the actual number of command blocks specified, all command blocks available are acted upon and an error is returned.

If the count of command blocks is less than the actual number of command blocks in the command, only the number of commands specified in the count is acted upon but no error is returned.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 2Ch | 0Ah | Variable | Variable | See Table below | | |

Mifare ePurse Command Data Field

| Data Field | Length (bytes) | Description | | | | | | | |
|-----------------------------------|----------------|---|-----------|---|---|-----------------|---|---|---|
| Mode, Card Type & Operation Count | 1 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | | 1 = Inc 0 = Dec | Card Type | | | Operation Count | | | |
| | | Increment / Decrement Flag: [Bit 7] Set to 1 instructs reader to Add to (Credit) amount. Set to 0 instructs reader to Subtract from (Debit) amount. Card Type: [Bit 6..4] This can only indicate Mifare Type A (Standard) card (3, as defined in the "Poll for Token" command). Operation Count: [Bit 3..0] This is the number of operation command blocks contained within the rest of the Purse Function data area. | | | | | | | |
| Purse Function Blocks | Variable [1] | Series of any combination of supported Purse Function blocks (Debit/Credit, Backup). Refer to the description of each individual command frame below. | | | | | | | |

Debit / Credit Function Block (with Key specified)

| Byte 0 | Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte 5 | Byte 6 | Byte 7 | Byte 8 | Byte 9 | Byte 10 | Byte 11 | Byte 12 |
|--------------------|----------------|-----------------|--------|--------|--------|-----------|-----------|--------|--------|---------|---------|---------|
| Value Block Number | Command Length | Amount | | | | Key Type | Key | | | | | |
| | 0Bh | See Table below | | | | See Table | See Table | | | | | |

Debit / Credit Function Block (using default Key)

| Byte 0 | Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte 5 |
|--------------------|----------------|-----------|--------|--------|--------|
| Value Block Number | Command Length | Amount | | | |
| | 04h | See Table | | | |

Mifare ePurse Data Field for Debit/Credit Function Block

| Data Field | Length (bytes) | Description |
|------------|----------------|--|
| Amount | 4 | Amount to be added (Debit) or subtracted (Credit) in Little-Endian format. Mode of operation (+ or -) is specified by most significant bit of first data byte in Purse Command (Mode, Card Type and Operation Count) |
| Key Type | 1 | Specifies which type of key to use for authentication. It can have the following values. 01h: Key A 02h: Key B |
| Key | 6 | Value of the Key |

For details on these fields, refer to the relevant Mifare Specifications.

Backup Function Block (with Key specified)

| Byte 0 | Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte 5 | Byte 6 | Byte 7 | Byte 8 | Byte 9 |
|---------------------|----------------|----------------------|-----------------|-----------------|--------|--------|--------|--------|--------|
| Backup Block Number | Command Length | Primary Block Number | Key Type | Key | | | | | |
| See Table below | 08h | See Table below | See Table below | See Table below | | | | | |

Backup Function Block (using default Key)

| Byte 0 | Byte 1 | Byte 2 |
|---------------------|----------------|----------------------|
| Backup Block Number | Command Length | Primary Block Number |
| See Table below | 01h | See Table below |

Mifare ePurse backup commands are distinguished from Debit/Credit operations by the value in the Command Length field.

Mifare ePurse Data Field for Backup Function Block

| Data Field | Length (bytes) | Description |
|----------------------|----------------|---|
| Backup Block Number | 1 | Number of destination value block to be used for backup. |
| Command Length | 1 | Set to 01h or 08h, depending on whether a key type and key are supplied. |
| Primary Block Number | 1 | Number of source value block to be copied. |
| Key Type | 1 | Present only if Command Length = 08h Specifies which type of key to use for authentication. It can have the following values. 01h: Key A 02h: Key B |
| Key | 6 | Present only if Command Length = 08h Value of the Key |

For details on these fields, refer to the relevant Mifare Specifications.

After receiving the command frame, the reader verifies the parameters. If the parameters are valid, it performs the operations specified in the order in which they appear within the **Mifare ePurse** command data frame.

Note: Although it is possible to include multiple value operations (Debit or Credit) in one command, because there is only a single one-bit flag to specify the Debit or Credit mode, all value commands within one **Mifare ePurse** command must be either Debit or Credit functions. However, backup operations may be included because they are distinguished by the command length field.

If all operations are successful, the reader sends a response frame with an OK status.

If the command frame is invalid or any of the operations fail, the reader sends a response frame with an appropriate status.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-15 | Byte 16 | Byte 17 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Return Code | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | See status code table | 00h | 02h | Varies | Varies | Varies |

Note: The block number and error code appear in the return data only if the block operation fails. The remaining blocks will not be manipulated.

| Return Code | Length | Description |
|--------------|--------|--|
| Block Number | 1 | 00h-FFh |
| Error Code | 1 | 01h:Function block Authenticate fails 02h:Function block Read/Write fails 10h:Backup block Authenticate fails 20h:Backup block Read/Write fails |

Examples

Application: Perform a Debit operation. Subtract 2000 from value block number 20H using last key specified. Blue shaded area shows the Debit function block within the **Mifare ePurse** command frame.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------------------------|-------------|------------------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Mode, Card Type, Operation Count | Value Block | Debit Cmd Length |
| ViVOtech2\0 | 2Ch | 0Ah | 00h | 07h | 31h | 20h | 04h |
| Byte 17 | Byte 18 | Byte 19 | Byte 20 | Byte 21 | Byte 22 | | |
| Debit Amount | | | | CRC MSB | CRC LSB | | |
| D0h | 07h | 00h | 00h | | | | |

Application: Perform a Credit operation. Add 100 to value block number 20H specifying Key A. Blue shaded area shows the Credit function block within the **Mifare ePurse** command frame.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------------------------|-------------|-------------------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Mode, Card Type, Operation Count | Value Block | Credit Cmd Length |
| ViVOtech2\0 | 2Ch | 0Ah | 00h | 0Eh | B1h | 20h | 0Bh |
| Byte 17 | Byte 18 | Byte 19 | Byte 20 | Byte 21 | Byte 22 | Byte 23 | Byte 24 |
| Credit Amount | | | | Key Type | Key | | |
| 64 | 00 | 00 | 00 | 01 | Ka | Kb | Kc |
| Byte 25 | Byte 26 | Byte 27 | Byte 28 | Byte 29 | | | |
| Key | | | CRC MSB | CRC LSB | | | |
| Kd | Ke | Kf | | | | | |

Application: Perform a Debit operation with Backup. Subtract 300 from value block number 20H specifying Key A and backup the result to value block number 21H using the same key. Blue shaded area shows the Debit function block and yellow shaded area shows the Backup function block within the **Mifare ePurse** command frame.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------------------------|-------------|------------------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Mode, Card Type, Operation Count | Value Block | Debit Cmd Length |
| ViVOtech2\0 | 2Ch | 0Ah | 00h | 11h | 32h | 20h | 0Bh |
| Byte 17 | Byte 18 | Byte 19 | Byte 20 | Byte 21 | Byte 22 | Byte 23 | Byte 24 |
| Debit Amount | | | | Key Type | Key | | |
| 2Ch | 01 | 00 | 00 | 01 | Ka | Kb | Kc |
| Byte 25 | Byte 26 | Byte 27 | Byte 28 | Byte 29 | Byte 30 | Byte 31 | Byte 32 |
| Key | | | Backup Block | Backup Cmd Length | Primary Block | CRC MSB | CRC LSB |
| Kd | Ke | Kf | 21h | 01h | 20h | | |

Application: Perform a Backup (value copy) operation only. Copy the value amount from block 1CH to block 1DH specifying Key B. Yellow shaded area shows the Backup function block within the **Mifare ePurse** command frame.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|----------|-------------|-------------------|-------------------|----------------------------------|--------------|-------------------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Mode, Card Type, Operation Count | Backup Block | Backup Cmd Length |
| ViVOtech2\0 | 2Ch | 0Ah | 00h | 0Bh | 31h | 1Dh | 08h |
| Byte 17 | Byte 18 | Byte 19 | Byte 20 | Byte 21 | Byte 22 | Byte 23 | Byte 24 |
| Primary Block | Key Type | Key | | | | | |
| 1Ch | 02h | Ka | Kb | Kc | Kd | Ke | Kf |
| Byte 25 | Byte 26 | | | | | | |
| CRC MSB | CRC LSB | | | | | | |
| | | | | | | | |

7.16 High-level Pass-Through Commands for NFC Cards

This section contains serial commands that implement higher level functionality for the NFC Cards. These commands do not work for non-NFC cards.

7.16.1 NFC Commands (2C-40)

This command uses Data[0] in command data field to implement different functions. This command should be used in Pass-Through mode and command with "Poll for a NFC Tag" data should be used first. Command with other data can only be used after the "Poll for a NFC Tag" command has indicated that a NFC tag is present.

NFC Commands

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data length (MSB) | Data length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | 2Ch | 40 | 00 | 00 | See Below | | |

Individual commands in NFC command set are distinguished as to parameters in Data field.

NFC Command Set List

| Command | Data length | Command Data Field Description |
|-------------------------------|----------------|---|
| Poll for a NFC Tag | 2 | Data[0]: FFh Data[1]: Timeout (in second) |
| Tag1 Static Get All Data | 1 | Data[0]: 11h |
| Tag1 Static Read a Byte | 2 | Data[0]: 12h Data[1]: Address of the data |
| Tag1 Static Write a Byte | 3 | Data[0]: 13h Data[1]: Address of the data Data[2]: Data to be written |
| Tag1 Static Write a Byte NE | 3 | Data[0]: 14h Data[1]: Address of the data Data[2]: Data to be written |
| Tag1 Dynamic Read a Segment | 2 | Data[0]: 15h Data[1]: Address of the segment |
| Tag1 Dynamic Read 8 Bytes | 2 | Data[0]: 16h Data[1]: Address of the data |
| Tag1 Dynamic Write 8 Bytes | 10 | Data[0]: 17h Data[1]: Address of the data Data[2]~Data[9]: Data to be written |
| Tag1 Dynamic Write 8 Bytes NE | 10 | Data[0]: 18h Data[1]: Address of the data Data[2]~Data[9]: Data to be written |
| Tag2 Read Data (16 bytes) | 2 | Data[0]: 21h Data[1]: Address of the data |
| Tag2 Write Data (4 bytes) | 6 | Data[0]: 22h Data[1]: Address of the data Data[2]~Data[5]: Data to be written |

| | | |
|------------------|----------|---|
| Tag2 Select Sect | 2 | Data[0]: 23h Data[1]: Sect number |
| Tag3 Read Data | variable | Data[0]: 41h Data[1]: Number of services, value n. Data[2]~Data[2n+1]: Service code list Data[2n+2]: Number of blocks, value m. Data[2n+3....]: Block list, length is 2m~3m |
| Tag3 Write Data | variable | Data[0]: 42h Data[1]: Number of services, value n. Data[2]~Data[2n+1]: Service code list Data[2n+2]: Number of blocks, value m. Data[2n+3....]: Block list, length is 2m~3m Data[...]: Block data, length is 16m |
| Tag4 Command | variable | Data[0]: 0x81 Data[1]~Data[n]: data |

NFC Response

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | 2Ch | See status code table | 00 | 00 | See Below | | |

NFC Command Set Response Data List

| Command Response | Data length | Command Response Data Field Description |
|--------------------|-------------|---|
| Poll for a NFC Tag | variable | Data[0]: Card type 00h None (Card not detected, could not activate or not a Tag Card) 0Ah NFC Tag 1 0Bh NFC Tag 2 0Ch NFC Tag 3 0Dh NFC Tag 4 Data[1...]: Serial Number (or the UID) of the PICC. Length depends on the card detected. If no card was detected, then a Serial Number is not returned. |
| Others | variable | Returned data from card |

For details on these data field, refer to the relevant NFC Specifications.

For SRED readers, if the command isn't "Poll for a NFC Tag", the data the reader receives from the card or device is checked for sensitive data elements using rule in "Secure Pass-Through Function." If found, and it has not been established that this transaction belongs to an AID or BIN on the Whitelist, the Command will return a Parameter Not Supported error (0x06).

7.17 High-level Pass-Through Commands for Felica Cards

This section contains serial commands that implement higher level functionality for the Felica Cards. These commands do not work for non-Felica cards.

7.17.1 Felica Commands (2C-41)

This command uses Data[0] in command data field to implement different functions.

These commands can only be used after the reader has been put in Pass-Through mode and the "Poll for an NFC Tag" command has indicated that a Felica Card is present. These commands do not work for non-Felica cards.

NFC Commands

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data length (MSB) | Data length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | 2Ch | 41h | Variable | Variable | See Below | | |

Individual commands in Felica command set are distinguished as parameters in Data field.

Felica Command Set List

| Command | Data length | Command Data Field Description |
|--------------------------|-------------|---|
| Read without Encryption | variable | Data[0]: 06h Data[1]: Number of services, value n. Data[2]~Data[2n+1]: Service code list Data[2n+2]: Number of blocks, value m. Data[2n+3....]: Block list, length is 2m~3m |
| Write without Encryption | variable | Data[0]: 08h Data[1]: Number of services, value n. Data[2]~Data[2n+1]: Service code list Data[2n+2]: Number of blocks, value m. Data[2n+3....]: Block list, length is 2m~3m Data[...]: Block data, length is 16m |

Felica Response

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---------------------------------------|-------------------------|-------------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | 2Ch | See status code table | 00 | 00 | Returned data from card | | |

For details on these data field, refer to the relevant Felica Specifications.

7.17.2 Felica Lite/Lite-S Authenticate (2C-42)

This command allows the host device to instruct the ViVOpay reader to authenticate the Felica Lite/Lite-S card. The Key to be used is also specified by the host. This command is applicable only for Felica Lite/Lite-S card Cards.

This command is not applicable for Felica Standard Cards.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-Byte 21 | Byte 22 | Byte 23 |
|-------------------------------|---------|-------------|-------------------|-------------------|------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | 42h | 00h | 10h | Value of the Key | | |

For details on these fields, refer to the relevant Felica Lite/Lite-S Specifications.

After receiving the Command Frame, the reader verifies the data and if the data is valid, it interacts with the Felica Lite/Lite-S card to do authentication. If this operation is successful, the reader sends a response frame with an OK Status. If the operation fails or the data was invalid, then the reader returns a response frame with an appropriate Status.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | See status code table | 00h | 00h | | |

If the card in the field is not a Felica card, the reader returns a 0Ch (sub-command not allowed) status code.

7.17.3 Felica Lite/Lite-S Read/Write Blocks with MAC (2C-43)

This command uses Data[0] in command data field to implement different functions. Use this command to read data from one or more blocks on the Felica Lite/Lite-S Card with MAC or write data to one block on Felica Lite-S with MAC. The host device can instruct the reader to read up to 3 blocks using this command. The blocks to be read/written are specified by BlockList.

Before you use this command, the host device must have successfully sent at least one Felica Lite/Lite-S Authenticate command to the reader.

If the card specified is a Felica Lite card, Write Block with MAC is not valid.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14, 15 | Byte 16 | Byte 17 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | 43h | 00h | 02h | See Table below | | |

Felica Lite/Lite-S Read/Write Blocks with MAC Data Field

| Command | Length (bytes) | Description |
|----------------|----------------|--|
| Read With MAC | variable | Data[0]: 06h Data[1]: Number of blocks to be read, value n Data[2]~Data[2n+1]: Block List |
| Write With MAC | 18 | Data[0]: 08h Data[1]: Block number to be written Data[2]~Data[17]: Data to be written to Block |

After receiving the command frame the reader verifies the parameters. If the parameters are valid, then it reads/writes the data from/to the card. If the read operation is successful, the reader sends a response frame containing a Status of OK and the data that was read. If the write operation is successful, the reader sends a response frame containing a Status of OK. If the operation fails or one or more parameters were invalid, then the reader sends a response frame containing an appropriate Status, but no data.

Response Frame for Felica Lite/Lite-S Read Blocks with MAC

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|--------------------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | See status code table | Variable | Variable | Data Read from Card OR None | | |

If the Status is OK, then the Data Length depends on the number of blocks read and the card type.

DataLen = Blocks Read * (Bytes per Block for Card).

If there was an error or no data was read, then the Data Length is zero.

Response Frame for Felica Lite-S Write Blocks with MAC

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|----------------------------------|---------|---|----------------------|----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | See status code table | 00h | 00h | | |

7.17.4 Pass-Through Package with Felica Card (2C-44)

The **Pass-Through Package with Felica Card** command is a data-package-related command specifically used with Felica cards.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|----------------------------------|---------|-----------------|----------------------|----------------------|--------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 2Ch | 44h | | | See Below | | |

Data Field

| Item | Length (bytes) | Description |
|---------|----------------|--|
| Length | 1 | The length of payload. |
| Payload | N | Command code and command data (exclude CRC). |

Example:

Polling command data field: 05 00 00 03 01 03.

Felica Response

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14... Byte 13+n | Byte 14+n | Byte 15+n |
|----------------------------------|---------|---|-------------------------|-------------------------|-------------------------------|--------------|-----------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVotech2\0 | 2Ch | See status code table | 00 | 00 | Returned data from card | | |

Data Field

| Item | Length (bytes) | Description |
|---------|----------------|--|
| Payload | N | Response code and response data (exclude and CRC). |

Example:

Polling response data field: 01 01 02 03 04 05 06 07 08 10 0B 4D 42 84 85
D1 FF 00 03

After enabling Encrypted R-APDU Output:

- If the Encryption Key is DEK, and if the command is not "Poll for an NFC Tag", the data response from the PICC skips checking for sensitive data elements and follows the Encrypted R-APDU Format rules.
- If the Encryption Key is TransArmor RSA Cert, the reader disables transactions and sends a 0x0A response status code.

Encrypted R-APDU Format Rules:

<Encrypted R-APDU Length (MSB)> <Encrypted R-APDU Length (LSB)> <n bytes Encrypted R-APDU> <DEK KSN Length (MSB)> <DEK KSN Length (LSB)> <DEK KSN> <MAC Value Length (MSB)> <MAC Value Length (LSB)> [<MAC Value>]

Where:

- <Encrypted R-APDU Length (MSB)> <Encrypted R-APDU Length (LSB)>: is length of <n bytes Encrypted R-APDU>
- <n bytes Encrypted R-APDU>
 - The format of n bytes Raw R-APDU as below: 2 bytes valid R-APDU Length (MSB, LSB) + x bytes valid Raw R-APDU + y bytes Padding.

Note:

- For TDES mode: 2+x should be a multiple of 8. If it is not a multiple of 8, the reader should pad with zero by y bytes automatically (2+x+y should be multiple of 8).
- For AES mode: 2+x should be a multiple of 16. If it is not a multiple of 16, the reader should pad with zero by y bytes automatically (2+x+y should be multiple of 16).
- <DEK KSN Length (MSB)> < DEK KSN Length (LSB)>:
- <DEK KSN>
- <MAC Value Length (MSB)> <MAC Value Length (LSB)>:
 - If there is encrypted R-APDU:
 - if MAC Verification Data is Switch On, it is MAC Value length.
 - If MAC Verification Data is Switch Off, it is <0x00> <0x00>.
- <MAC Value>:
 - If there is encrypted R-APDU:
 - if MAC Verification Data is Switch On, it is MAC Value.
 - If MAC Verification Data is Switch Off, it does not exist.

7.17.5 Felica Get Code (2C-45)

The **Felica Get Code** command starts the Felica get code process. The timeout of the entire process is three minutes.

Note: This command only works in Pass-Through Mode.

After the reader receives a 2C-45 command, it immediately sends a first response back to the host to indicate whether 2C-45 is accepted or not. If the reader accepts the command, it begins the get code process and sends second response after completing the operation.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVotech2\0 | 2Ch | 45h | 00h | 00h | | |

First Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVotech2\0 | 2Ch | See status code table | 00h | 00h | | |

Second Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|--|--------------|--------------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | See status code table | 00h | 00h or 04h | 4 bytes data of entered text plain | | |

Examples

Example 1: Entered code data is "1234"

TX: 56 69 56 4F 74 65 63 68 32 00 2C 45 00 00 EA C6

RX: 56 69 56 4F 74 65 63 68 32 00 2C 00 00 00 1C 9B

RX: 56 69 56 4F 74 65 63 68 32 00 2C 00 00 04 31 32 33 34 82 D2

Example 2: Process timed out

TX: 56 69 56 4F 74 65 63 68 32 00 2C 45 00 00 EA C6

RX: 56 69 56 4F 74 65 63 68 32 00 2C 00 00 00 1C 9B

RX: 56 69 56 4F 74 65 63 68 32 00 2C 08 00 00 B5 3A

Example 3: Process canceled by user

TX: 56 69 56 4F 74 65 63 68 32 00 2C 45 00 00 EA C6

RX: 56 69 56 4F 74 65 63 68 32 00 2C 00 00 00 1C 9B

RX: 56 69 56 4F 74 65 63 68 32 00 2C 2D 00 00 D8 0C

7.17.6 Cancel Felica Get Code (2C-46)

The **Cancel Felica Get Code** command closes the Felica get code process immediately.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 2Ch | 46h | 00h | 00h | | |

Command Response

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | See status code table | 00 | 00 | | |

Example:

TX: 56 69 56 4F 74 65 63 68 32 00 2C 45 00 00 EA C6 // 2C-45

RX: 56 69 56 4F 74 65 63 68 32 00 2C 00 00 00 1C 9B // first response to 2C-45 (GUI handling for Felica Get Code)

TX: 56 69 56 4F 74 65 63 68 32 00 2C 46 00 00 B3 96 // 2C-46

RX: 56 69 56 4F 74 65 63 68 32 00 2C 00 00 00 1C 9B // response to 2C 46

RX: 56 69 56 4F 74 65 63 68 32 00 2C 2C 00 00 EF 3C // second response to 2C 45

7.18 Pass-Through Commands for Motor control

This section contains serial commands apply specifically to the VP5300M reader.

7.18.1 Bypass Data to Motor (2C-70)

This command is used to bypass motor commands to the motor board. (VP5300M Only)

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14... Byte14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|--------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 2Ch | 70h | Variable | Command Data | | | |

Command Data

| Data Item | Length (bytes) | Description/Example |
|-----------|----------------|----------------------------------|
| Mode | 1 | 30H: APP Mode 31H: Flash Mode |
| Command | Variable | Reference Motor Spec |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | See status code table | 00h | variable | Response Data | | |

Response Data

| Data Item | Length (bytes) | Description/Example |
|-----------|----------------|--------------------------------|
| Data | variable | Reference Sanwa Motor App Spec |

7.18.2 Entry Motor Boot Mode (2C-71)

This command is used to set the boot mode or application mode of the motor board.
(VP5300M Only)

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14... Byte14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|--------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 2Ch | 71h | Variable | | | | |

Command Data

| Data Item | Length (bytes) | Description/Example |
|-----------|----------------|--|
| Reset | 1 | Hardware reset for Sanwa Motorized Board |
| Mode | 1 | 30H: Application Mode. (Uart3 baud rate change to 9600) 31H: Boot Mode. (Uart3 baud rate change to 38400) |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---------------------------------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 2Ch | See status code table | 00h | variable | Response Data | | |

These commands can only be used after the reader has entered Pass-Through mode.
These commands control the motor through a specific setup flow.

Specific Setup Flow:

1. Pass-Through mode open

IDGcommand: 56 69 56 4F 74 65 63 68 32 00 2C 01 00 01 01 1D 19
Command return data: 56 69 56 4F 74 65 63 68 32 00 2C 00 00 00 1C 9B

2. Motor APP mode setting

IDGcommand: 56 69 56 4F 74 65 63 68 32 00 2C 71 00 02 00 30 28 9B
Command return data: 56 69 56 4F 74 65 63 68 32 00 2C 00 00 0A 02 08 0F 30 30 30 00 30 03 04 1D 08

3. Card Entry setting

IDGcommand: 56 69 56 4F 74 65 63 68 32 00 2C 70 00 06 30 02 03 33 03 33 A9 EE

Command return data: 56 69 56 4F 74 65 63 68 32 00 2C 00 00 09 02
07 33 31 30 30 06 03 00 F8 D2

4. Card Eject setting

IDGcommand: 56 69 56 4F 74 65 63 68 32 00 2C 70 00 08 30 02 05
80 00 E8 03 80 66 E3

Command return data: 56 69 56 4F 74 65 63 68 32 00 2C 00 00 09 02
07 80 34 30 30 01 03 B1 A3 4F

5. Card State Clean setting

IDGcommand: 56 69 56 4F 74 65 63 68 32 00 2C 70 00 06 30 02 03
40 03 40 05 21

Command return data: 56 69 56 4F 74 65 63 68 32 00 2C 00 00 09 02
07 40 30 30 30 01 03 75 4A 33

7.19 Secure Pass-Through Function

Note: In SRED readers, Pass-Through Mode is called secure Pass-Through Mode.

7.19.1 General Introduction

In Secure Pass-Thru mode the reader will not allow the exposure of sensitive financial data from the reader unless the data from the card provides an Application Identifier (AID) or a Bank Identification Number (BIN) that is listed in the Whitelist. An AID or BIN added to the Whitelist is a guarantee that the AID or BIN is not used for a financial transaction.

In Secure Pass-Thru mode, if the AID or BIN is not in the Whitelist, then all data returned from the card is parsed and analyzed to detect sensitive financial data before that data is provided in the transaction response. If any sensitive financial data is found only an error is returned and no data from the card is provided for the entire transaction.

Note: In Secure Pass-Through Function, use the entire PAN as input BIN to compare with the BIN in Whitelist

7.19.2 Secure Pass-Thru Whitelist

The Whitelist is loaded with MAC verification. The Whitelist is a list of AID's and BIN's that will be used to determine if data that appears to be financial data will be allowed and shown in clear text.

If an AID selected is on the Secure Pass-Thru Whitelist, all data for that transaction will be shown in clear text regardless of whether it is sensitive financial data or not. The Selected AID is not dependent on the occurrence of PPSE or List of AID's selection methods. If the AID selected is not on the Secure Pass-Thru Whitelist, all data received from PICC will be parsed searching for sensitive financial data. Any TLV or structure on the SRED list of protected financial data will cause the Pass-Thru command response to return a Parameter Not Supported error status (0x06) with no data.

If a BIN is found in a track data TLV or track data structure, that is also in the Whitelist, all data for that transaction will be shown in clear text regardless of whether it is sensitive financial data or not. If the BIN is not on the Whitelist the Pass-Thru command response will return a Parameter Not Supported error status (0x06) with no data. Because the BIN is embedded in track data, the fact that there was a BIN indicates sensitive data was found.

Note: In either the matching AID or matching BIN scenarios, if sensitive data is retrieved in a command prior to the command where the AID or BIN that matches the Whitelist is found, the transaction response will return a Parameter Not Supported status (0x06) error with no data.

WARNING – No Card Association AID's may be added to the white list. For example the AID's for Mastercard, Visa, Amex, or others may NOT be added to the Whitelist.

7.19.3 Handling Sensitive Financial Data

Any TLV or structure on the SRED list of protected financial data will cause the Pass-Thru command response to return a Parameter Not Supported status (0x06) with no data returned.

SRED List of Protected Financial Data

| TLV | Name | Find Method |
|------|-----------------------------|---|
| 56 | Track 1 Equivalent Data | Match EMV TLV |
| 57 | Track 2 Equivalent Data | Match EMV TLV |
| 5A | Application PAN | Match EMV TLV |
| 5F20 | Cardholder Name | Match EMV TLV |
| 5F24 | Application Expiration Date | Match EMV TLV |
| 5F30 | Service Code | Match EMV TLV |
| 9F27 | Cryptogram Information Data | Match EMV TLV |
| 9F60 | CVC3Track1 | Match EMV TLV |
| 9F61 | CVC3Track2 | Match EMV TLV |
| 9F6B | Track 2 Data | Match EMV TLV |
| | | |
| none | Track 1 | Matches ISO/IEC 7813 format for Track 1. See details in Track 1 Format Test below. |
| none | Track 2 | Matches ISO/IEC 7813 format for Track 2. See details in Track 2 Format Test below. |
| none | Track 3 | Matches ISO/IEC 4909:2006 format for Track 3. See details in Track 3 Format Test below. |

7.19.4 Accessing SAMs in Pass- Thru Mode

In Secure Pass-Thru SAM access is always clear data. The SAM will never contain sensitive financial data.

Pass-Thru Commands Needing to be parsed for sensitive financial data:

| Cmd-Sub | Name |
|---------|-----------------------------|
| 2C-03 | Exchange APDU |
| 2C-04 | PCD Single Command Exchange |
| 2C-07 | Read Mifare Block |
| 2C-13 | Exchange APDU |
| 2C-40 | NFC Commands |

7.19.5 Secure Pass-Thru Whitelist

WARNING: No Card Association AIDs may be added to the white list. For example the AIDs for Mastercard, Visa, Amex, or others CANNOT be added to the Whitelist.

Whitelist related commands can only be used when Pass-Through mode is started.

7.19.6 Get Whitelist (2C-51)

This command retrieves the Whitelist.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVotech2\0 | 2Ch | 51h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVotech2\0 | 2Ch | Status | Variable | Variable | Response Data | | |

Response Data: See tables for Set Whitelist above for the Whitelist format and options.

| Data Field | Length (bytes) | Description |
|------------|----------------|---|
| Whitelist | Var | The actual white list. See format of Whitelist in Set Whitelist section above. Tag: FFEE0C Format: Constructed |

7.19.7 Get Discretionary Data Whitelist (2C-53)

This command retrieves the Discretionary Data Whitelist.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14, 15 | Byte 16 | Byte 17 |
|-------------------------------|---------|-------------|-------------------|-------------------|---------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVotech2\0 | 2Ch | 53h | var | var | Response Data | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|----------------|-------------------------|-------------------------|--------------------------|--------------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | 2Ch | Status | Variable | Variable | Response Data | | |

Response Data - See tables for Set Whitelist above for the Whitelist format and options.

| Data Field | Length (bytes) | Description |
|------------|-------------------|---|
| Whitelist | Var | The actual white list. See format of Whitelist in Set Whitelist section above. Tag: FFEE0C Format: Constructed |

Example Response:

```
FFEE0C70FFEE0A14DFEF2106373037363030DFEF2106373037363939FFEE0A14DF
EF2106373737373030DFEF21063737373939FFEE0A14DFEF2106363232303236
DFEF2106363232303237DFEF2106333038333737DFEF2106363030363439DFEF21
06363031303536DFEF2106363239383539
```

7.20 MSR Function Check

The following commands deal with MagStripe Reader functionality.

7.20.1 Check MSR Swipe Function (2C-20)

The **Check MSR Swipe Function** command checks MSR swipe functionality (including Mag-head and TM4 chip). The command is used mostly during manufacturing, when the reader is not yet activated.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | 2Ch | 20h | 00h | 00h | | |

After sending this command, user should swipe MSR card in 20 seconds. If the MSR module works OK, the reader will send MSR result data after MSR is swiped successfully.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 – 15 | Byte 16 | Byte 17 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | 2Ch | See status code table | 00h | 00h or 02h | See data table | | |

Data Table

| Data Item | Length | Description |
|--------------|--------|---|
| Card type | 1 | 0x80: ISO-ABA card 0x81: AAMVA card 0x83: non-financial card 0x85, 0x86, 0x87: JIS card |
| Track status | 1 | Bit0: 1: track1 decode success, 0: track1 decode fail Bit1: 1: track2 decode success, 0: track2 decode fail Bit2: 1: track3 decode success, 0: track3 decode fail Bit3: 1: track1 sample data exist, 0: track1 has no data Bit4: 1: track2 sample data exist, 0: track2 has no data Bit5: 1: track3 sample data exist, 0: track3 has no data Bit6~Bit7: RFU |

For security, the reader doesn't output MSR card data; the only data displayed is the card type and the track status bytes.

7.21 Logger Commands

The NEO 2 logging system creates log information based on specified log levels. The logging system creates host utilities to download logs and convert them to a readable format. To reduce memory buffer size, the logging system saves only the log string address to the logger buffer; that address is used later to look up the string from the corresponding ELF created at build time.

Note that the total count of log entries depends on the memory allocated for the ring buffer used for log storage. Old logs are overwritten by the latest logs if they are not retrieved in time.

Only statically-declared log strings can be retrieved. Strings in the buffer declared as local variables cannot be retrieved.

The host utility neolog.exe supports only serial and USB communication.

7.21.1 Initialize/De-initialize Logger (4C-01)

Use this command to allocate a log count * 16 bytes memory for logger ring buffer.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14~16 | Byte 17 | Byte 18 |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVotech2\0 | 4Ch | 01h | 00h | 03h | Data Objects | | |

Data Objects

| Data Item | Length (bytes) | Description |
|-----------|----------------|---|
| Mode | 1 | -0x00: de-init; -0x01: init; |
| Log count | 2 | If mode is de-init, should be 0x0000; If mode is init, valid value: 0x0001~0xffff; |

Note:

- 0xffff is the theoretical maximum log count.
- Based on different memory allocated allowances, each product may have a different maximum log count.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 4Ch | See status code table | 00h | 00h | | |

7.21.2 Set Logger Level (4C-02)

Use this command to set the level that the log with equal or higher level shall be recorded in the logger ring buffer.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 4Ch | 02h | 00h | 01h | Data Objects | | |

Data Objects

| Data Item | Length (bytes) | Description |
|--------------|----------------|---|
| Logger Level | 1 | -0x00: Debug; -0x01: Info; -0x02: Warning; -0x03: Error; |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 4Ch | See status code table | 00h | 00h | | |

7.21.3 Get/Set Logger Size (4C-03)

Use this command to get/set max count of logger ring buffer.

1.1.3.13. Set Max Count

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14, 15 | Byte 16 | Byte 17 |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 4Ch | 03h | 00h | 02h | Data Objects | | |

Data Objects

| Data Item | Length (bytes) | Description |
|-----------|----------------|--|
| Log count | 2 | 0x0000~0xffff, 0x0000 means de-init logger. |

Note:

- 0xffff is the theoretical maximum log count.
- Based on different memory allocated allowances, each product may have a different maximum log count.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 4Ch | See status code table | 00h | 00h | | |

1.1.3.14. Get Max Count

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 4Ch | 03h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14, 15 | Byte 16 | Byte 17 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 4Ch | See status code table | 00h | 02h | Log count | | |

7.21.4 Enable/Disable Logger (4C-04)

Use this command to enable or disable the logger.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 4Ch | 04h | 00h | 01h | Data Objects | | |

Data Objects

| Data Item | Length (bytes) | Description |
|-----------|----------------|-----------------------------------|
| Mode | 1 | -0x00: disable; -0x01: enable; |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 4Ch | See status code table | 00h | 00h | | |

7.21.5 Save/Delete Log (4C-05)

Use this command to save the log from ring buffer to a file or delete the current log file. The newly saved log is appended to the end of the file.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 4Ch | 05h | 00h | 01h | Data Objects | | |

Data Objects

| Data Item | Length (bytes) | Description |
|-----------|----------------|--------------------------------|
| Mode | 1 | -0x00: Delete; -0x01: Save; |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 4Ch | See status code table | 00h | 00h | | |

7.21.6 Retrieve Log (4C-06)

Use this command to output log saved in file and ring buffer.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 4Ch | 06h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n | Byte 15 + n | Byte 16 + n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|--------------------------|----------------|----------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 4Ch | See status code table | | | Log data | | |

Note:

- Because the maximum output data length is 1024 bytes, you may call this command several times to retrieve the whole log from file first and then from the ring buffer.
- A response with 0 data length indicates that all log data has been output.
- You can try to retrieve a log from the beginning again after receiving a response 0 data length response.

7.22 Secure Communication

Special Considerations for Secure Communications

Take time to familiarize yourself with certain key differences in reader usage that come into play when secure communications are required (as described below).

7.22.1 Burst Mode

Burst Mode is not allowed when MSR/MSD or EMV encryption is enabled and Data encryption Key exists. Burst Mode is a legacy mode, supported in readers that do not have encryption turned on.

When MSR/MSD or EMV encryption is enabled and Data encryption Key exists, Burst Mode is always OFF. In this condition, reader will turn the Burst Mode to be OFF automatically. If user tries to set Burst Mode to be ON/AUTO EXIT through "Set Configuration (04-00)" command, reader will keep Burst Mode to be OFF.

Note: Burst Mode is disabled for SRED readers, because SRED readers are encryption-enabled.

7.22.2 Data Output

When encryption is enabled, all magstripe data output (MSR) will be encoded according to the rules described in P/N 80000502-001 *ID TECH Encrypted Data Output Formats.pdf*, available for download on the [ID TECH Knowledge Base](#). The same document also describes which TLVs (for EMV transactions) are encrypted, and how.

For further information, consult the appendix on [Enhanced Encrypted MSR Data Output Format](#).

7.22.3 Encryption Algorithms

The reader uses TDES encryption by default. The data field in the Command/Response Frames is encrypted with Cipher Block Chaining (TDES-CBC, IV is 8 bytes 0x00). If AES encryption setting is accepted, the data field in the Command/Response Frames is encrypted with Cipher Block Chaining (AES-CBC, IV is 16 bytes 0x00).

Only the data fields of the ViVOpay command/Response Frames are encrypted. The 14-byte preamble consisting of the command header, command, sub-command, and status fields, will not be encrypted.

7.22.4 Secure Data Exchange

Data is encrypted using TDES-CBC or AES-CBC. After a session is established, the initial vector (IV) will never be reset to its initial value until a new session is established. Thus, the chaining extends across packets and ensures the order of packets. The result is that a session is encrypted in a unique/per-instance non-repeatable way, to make replay attacks all but impossible.

7.22.5 Padding of Data Fields

Padding is usually required for the Cipher Block Chaining (CBC) algorithm, because Triple DES will require that data blocks be a multiple of 8 bytes long (whereas AES will require data blocks to be a multiple of 16 bytes). Padding consists of null bytes (0x00) added to the end of the data, to give data with a length that's a multiple of 8 (for TDES) or 16 (for AES).

The order of operations for sending frames:

1. Add padding so that data length is a multiple of 8 for TDES-CBC or a multiple of 16 for AES-CBC.
2. Blockwise encrypt, using CBC.

The order of operations for receiving frames:

1. Decrypt using CBC.
2. Remove pads.

7.22.6 Get Data Encryption Key Variant Type (C7-30)

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data length (MSB) | Data length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | 30h | 00 | 00 | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | Data1 | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00 | 01 | Key Variant Type | | |

| Encryption Type | Description |
|-----------------|--------------|
| 0 | Data Variant |
| 1 | Pin Variant |

7.22.7 Set DUKPT Key Encryption Type (C7-32)

The **Set DUKPT Key Encryption Type** command specifies the encryption type of the TDES DUKPT data encryption key. It allows the reader to switch between standard TDES and TransArmor TDES. The command will not work when the reader is configured to use a key that is not TDES DUKPT for data encryption.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data length (MSB) | Data length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | C7h | 32h | 00 | 01 | Encryption Type | | |

| Encryption Type | Description |
|-----------------|-----------------|
| 0 | TDES |
| 2 | TransArmor TDES |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | C7h | See status code table | 00 | 00 | | |

7.22.8 Get DUKPT Key Encryption Type (C7-33)

Note: This command is only supported in Non-SRED readers. In SRED devices, only TDES algorithm is used to encrypt transaction output sensitive data.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data length (MSB) | Data length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | C7h | 33h | 00 | 00 | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte14 | Byte 15 | Byte16 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | Data1 | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00 | 01 | Encryption Type | | |

| Encryption Type | Description |
|-----------------|-------------|
| 0 | TDES |
| 1 | AES |

Example data (top line: command; bottom line: response)

TDES:

```
56 69 56 4F 74 65 63 68 32 00 C7 33 00 00 1A 9B
56 69 56 4F 74 65 63 68 32 00 C7 00 00 01 00 AC 7F
```

AES:

```
56 69 56 4F 74 65 63 68 32 00 C7 33 00 00 1A 9B
56 69 56 4F 74 65 63 68 32 00 C7 00 00 01 01 BC 5E
```

7.22.9 Set Data Encryption Enable Flag (C7-36)

This command is meant to be used one time (only), to turn encryption ON permanently. It elevates the security status of the reader. *This is meant to be an irreversible event.*

Note: This command is supported only in non-SRED readers. In SRED readers, the reader is *always* encryption-enabled and this command is unsupported.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte16 |
|-------------------------------|---------|-------------|-------------------|-------------------|------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data length (MSB) | Data length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | 36h | 00 | 01 | Encryption Enable Flag | | |

| Encryption Type | Description |
|-----------------|---|
| Bit 0 | 0 -> CT/CL EMV Data Encryption Disable (default) 1 -> CT/CL EMV Data Encryption Enable |
| Bit 1 | 0 -> MSR/MSD Data Encryption Disable (default) 1 -> MSR/MSD Data Encryption Enable |
| Bit 2~7 | Reserved |

Note: MSR/MSD Encryption Disable and EMV Encryption Enable isn't allowed.
When MSR/MSD or EMV Encryption is enabled and Data encryption Key exists, Burst Mode is disabled (always OFF)

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00 | 00 | | |

When Data Encryption is disabled, the device will always respond with plaintext data.

When Data Encryption is enabled, the reader will output data as follows:

1. When a DEK does not exist, the commands below will respond in plaintext data for NSRED device, respond status code in below for SRED:
 - a. 0x2B and no data if the DEK is mandatory and exists.
 - b. 0x90 and no data if the DEK is not mandatory and exists.
2. When a DEK exists and is valid, the commands below will respond in encrypted data.
3. When a DEK exists and exhausted, the commands below will respond with status code 0x91 and no data.

Commands:

1. Activate Transaction Command (02-01/02-40/60-10)
2. Get Transaction Result Command (03-00/03-40)

Reader Current Status/Command Setting/Reader Operation/Output Matrix

(Data encryption Key exists and valid)

| Current Encryption Status | Command Encryption Setting | Reader Operation | Output after Setting |
|------------------------------|------------------------------|------------------|------------------------------------|
| MSR/MSD_OFF, EMV_OFF 0x00 | MSR/MSD_OFF, EMV_OFF (0x00) | Keep | MSR/MSD plaintext, EMV plaintext |
| | MSR/MSD _OFF, EMV _ON (0x01) | Refuse | MSR/MSD plaintext, EMV plaintext |
| | MSR/MSD ON, EMV_OFF (0x02) | Update | MSR/MSD encryption, EMV plaintext |
| | MSR/MSD _ON, EMV_ON (0x03) | Update | MSR/MSD encryption, EMV encryption |
| MSR/MSD_ON, EMV_OFF 0x02 | MSR/MSD_OFF, EMV_OFF (0x00) | Refuse | MSR/MSD encryption, EMV plaintext |
| | MSR/MSD _OFF, EMV _ON (0x01) | Refuse | MSR/MSD encryption, EMV plaintext |
| | MSR/MSD ON, EMV_OFF (0x02) | Keep | MSR/MSD encryption, EMV plaintext |
| | MSR/MSD _ON, EMV_ON (0x03) | Update | MSR/MSD encryption, EMV encryption |

| Current Encryption Status | Command Encryption Setting | Reader Operation | Output after Setting |
|-------------------------------|------------------------------|------------------|------------------------------------|
| MSR/MSD_ON, EMV_ON 0x03 | MSR/MSD_OFF, EMV_OFF (0x00) | Refuse | MSR/MSD encryption, EMV encryption |
| | MSR/MSD _OFF, EMV _ON (0x01) | Refuse | MSR/MSD encryption, EMV encryption |
| | MSR/MSD ON, EMV_OFF (0x02) | Refuse | MSR/MSD encryption, EMV encryption |
| | MSR/MSD _ON, EMV_ON (0x03) | Keep | MSR/MSD encryption, EMV encryption |

7.22.10 Get Data Encryption Enable Flag (C7-37)

Note: This command is only supported in Non-SRED readers; it is not supported in SRED readers.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data length (MSB) | Data length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | 37h | 00 | 00 | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte14 | Byte 15 | Byte16 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | Data1 | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00 | 01 | Encryption Enable Flag | | |

| Encryption Type | Description |
|-----------------|---|
| Bit 0 | 0 -> CT/CL EMV Data Encryption Disable (default) 1 -> CT/CL EMV Data Encryption Enable |
| Bit 1 | 0 -> MSR/MSD Data Encryption Disable (default) 1 -> MSR/MSD Data Encryption Enable |
| Bit 2~7 | Reserve |

7.22.11 Set MSR Secure Parameters (C7-38)

The **Set MSR Secure Parameters** command allows setting parameters that determine encrypted output from MSR sessions. Use it to force encrypted data output to include various kinds of data per Enhanced Encrypted MSR Data Output When Encryption is Turned On with the C7-38 Command. Consult the table in the [Enhanced Encrypted MSR Data Output Format for Financial and Non-Financial Cards appendix](#) to see the types of output that can occur.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|---------------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | 38h | 00h | 05h | MSR Secure Parameters TLV | | |

MSR Secure Parameters TLV objects

| Tag | Data Object Name | Description | Format | Length |
|--------|-----------------------|---|--------|--------|
| DFEF04 | MSR Encryption Option | Encryption Option (Forced encryption or not) Bit 0: T1 force encrypt Bit 1: T2 force encrypt Bit 2: T3 force encrypt Bit 3: RFU Bit 4: Hash option for NSRED products; 1 is on, 0 is off Default value is 0x10. | b | 1 |

Notes:

1. Bits 0-2 (force encrypt setting) only affect **NON-financial** card data. If these are not set, non-financial card data is sent in clear text. If set, the data is sent in encrypted format. There is no mask data for non-financial card.
2. Bit 3 is RFU (to be backward compatible with some old ID TECH products) because financial card (type 80) data is always encrypted if the data key is injected.
3. SRED products use all zeroes (0x00) for hash data.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------------------|-------------------------|-------------------------|------------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | C7h | See Status Code Table | | | | | |
| ViVOTech2\0 | C7h | 38h | 00h | 05h | MSR Secure Parameters TLV | | |

7.22.12 Get MSR Secure Parameters (C7-39)

The **Get MSR Secure Parameters** command retrieves parameters from flash settings.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|---------------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | C7h | 39h | 00h | 03h | MSR Secure Parameters TLV | | |

| Tag | Data Object Name | Description | Format | Length |
|-------|-----------------------|---|--------|--------|
| DFF04 | MSR Encryption Option | Encryption Option (Forced encryption or not) Bit 0: T1 force encrypt Bit 1: T2 force encrypt Bit 2: T3 force encrypt Bit 3: RFU Bit 4: Hash option for NSRED products; 1 is on, 0 is off Default value is 0x10. | b | 1 |

Notes:

1. Bits 0-2 (force encrypt setting) only affect NON-financial card data. If these are not set, non-financial card data is sent in clear text. If set, the data is sent in encrypted format. There is no mask data for non-financial card.
2. Bit 3 is RFU (to be backward compatible with some old ID TECH products) because financial card (type 80) data is always encrypted if the data key is injected.
3. SRED products use all zeroes (0x00) for hash data.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00h | 05h | TLV | | |

7.22.13 Set MSR Tracks (C7-44)

The **Set MSR Tracks** command can get parameters from flash settings.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | 44h | 00h | 01h | Track Select ID | | |

TrackSelectID

| Value | Description | Length |
|-----------------|---|--------|
| '0' ('0' ~ '9') | Any Track '0' -any; '1'-'7' – bit 0 enable/disable tk1, bit 1 enable/disable tk2; bit 2 enable/disable tk3 '8' – tk1-2 '9' – tk2-3 | 1 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00h | 00h | | |

7.22.14 Get MSR Tracks (C7-45)

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | 45h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00h | 01h | Track Select ID | | |

TrackSelect ID

| Value | Description | Length |
|-----------------|---|--------|
| '0' ('0' ~ '9') | Any Track '0' -any; '1'-'7' – bit 0 enable/disable tk1, bit 1 enable/disable tk2; bit 2 enable/disable tk3 '8' – tk1-2 '9' – tk2-3 | 1 |

7.22.15 Load Certificate for TransArmor Encryption (C7-50)

The **Load Certificate for TransArmor Encryption** command sets the TransArmor Cert data for the TransArmor RSA encryption function.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | 50h | | | TransArmor Certificate | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00h | 00h | | |

7.22.16 Set TransArmor TID (C7-51)

The **Set TransArmor TID** command sets the TransArmor ID. TID must be 8 bytes (0x20 – 0x7F)

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | 51h | 00h | 08h | TransArmor ID | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00h | 00h | | |

7.22.17 Get TransArmor TID (C7-52)

The **Get TransArmor TID** command retrieves the TransArmor ID data for TransArmor RSA encryption functionality.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | 52h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00h | 08h | TransArmor ID | | |

7.22.18 Get TransArmor Certificate Status (C7-53)**Command Frame**

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | 53h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|-------------------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00h | 01h | TransArmor Certificate Status | | |

TransArmor Certificate Status

| Data Object Name | Description | Length |
|----------------------------------|--|--------|
| TransArmor Certificate Status | Bit 0: Root CA Certificate loaded Bit 1: Intermediate CA Certificate loaded Bit 2: TA Key Certificate Loaded | 1 |

7.22.19 Get TransArmor Certificate (C7-54)**Command Frame**

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | 54h | 00h | 01h | TransArmor Certificate | | |

TransArmor Certificate

| Data Object Name | Description | Length |
|------------------------|--|--------|
| TransArmor Certificate | 0x00: Root CA Certificate 0x01: Intermediate CA Certificate 0x02: TA Key Certificate | 1 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | | | TransArmor Certificate | | |

7.22.20 Set DUKPT PEK Attribution based on KeySlot (C7-A0)

Currently, when PEK is managed by AES DUKPT, the derivation key is AES-128 and the working key is AES-128. The KBH Optional Block setting for each key can be always valid in the future.

Command settings can only be changed for each key one time. If the command settings change the DUKPT Management Method of an existing key, it will be rejected.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data length (MSB) | Data length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | A0h | 00h | 08h | See Data Table | | |

Data Table

| Data Item | Length | Description |
|---------------------------|--------|---|
| PEK KeySlot Length | 1 | Length of DEK KeySlot, it is 1. |
| PEK KeySlot | 1 | It is one byte, and value is 0-9 |
| PEK DUKPT Management Mode | 1 | 00h: TDES DUKPT Management 01h: AES DUKPT Management |

| Data Item | Length | Description |
|---|--------|---|
| TDES DUKPT Output Mode or AES DUKPT Working Key | 1 | If TDES DUKPT Management: 00h: TDES If AES DUKPT Management: 00h: TDES Working Key [#1] 01h: TDES3 Working Key [#1] 02h: AES-128 Working Key 03h: AES-192 Working Key [#1] 04h: AES-256 Working Key [#1] #1: Option is not currently implemented. |
| RFU | 4 | Currently, this field should be 4 bytes 00h |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------------|---------|---------------------------------------|-------------------------|----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00h | 00h | | |

Examples:

1. Set TDES PEK 0 with no used Working key: 01 00 00 00 00 00 00 00
2. Set TDES PEK 3 with no used Working key: 01 03 00 00 00 00 00 00
3. Set AES PEK 0 Working Key is AES-128: 01 00 01 03 00 00 00 00
4. Set AES PEK 9 Working Key is AES-128: 01 09 01 03 00 00 00 00

7.22.21 Get DUKPT PEK Attribution based on KeySlot (C7-A1)**Command Frame**

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data length (MSB) | Data length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | A1h | 00 | var | See Data Table | | |

Data Table

| Data Item | Length | Description |
|--------------------|--------|----------------------------------|
| PEK KeySlot Length | 1 | Length of PEK KeySlot, it is 1. |
| PEK KeySlot | 1 | It is one byte, and value is 0-9 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | C7h | See status code table | 00h | 06h | See Data Table | | |

Data Table

| Data Item | Length | Description |
|--|--------|---|
| PEK DUKPT Management Mode | 1 | 00h: TDES DUKPT Management & TDES 01h: AES DUKPT Management |
| TDES DUKPT Output Mode or AES DUKPT Working Key | 1 | If TDES DUKPT Management: 00h: TDES If AES DUKPT Management: 00h: TDES Working Key 01h: TDES3 Working Key 02h: AES-128 Working Key 03h: AES-192 Working Key 04h: AES-256 Working Key |
| RFU | 4 | 4 bytes 00h |

7.22.22 Set DUKPT DEK Attribution based on KeySlot (C7-A2)

Currently, DEK is managed only by TDES DUKPT for AES-128 key length and by AES-112 for the strong option. KBH Optional Block setting for each key can be always valid in the future.

Command settings can only be changed for each key one time. However, in TDES DUKPT Management, switching the Output mode (TDES and TransArmor TDES) is always valid.

If the command settings change the DUKPT Management Method of an existing key, it will be rejected.

There are two attributes to TDES DUKPT DEK management:

- Output mode:
 - TDES
 - AES-128 (Strong is AES-112)
 - TransArmor TDES
- Variant:
 - Data Key
 - PIN Key

Note:

- If DEK does not exist, the above attributes can be set (changed).

- If DEK already exists, only TDES and TransArmor TDES can be switched.

There are two attributes to AES DUKPT DEK management:

- Working Key:
 - TDES
 - TDES3
 - AES-128
 - AES-192
 - AES-256
- KeyUsage:
 - Data Key both ways (_Data-Encryption-both-ways_)
 - Data Key encrypt(_Data-Encryption-encrypt_)
 - Data Key decrypt(_Data-Encryption-decrypt_)
 - MAC Key both ways (_Message-Authentication-both-ways_)
 - MAC Key generation(_Message-Authentication-generation_)
 - MAC Key verification(_Message-Authentication-verification_)

Note:

- If DEK does not already exist, the above attributes can be set (changed)
- If DEK already exists, the above attributes cannot be set (changed).

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|----------------------|-------------------------|----------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data length (MSB) | Data length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | A2h | 00h | 08h | See Data Table | | |

Data Table

| Data Item | Length | Description |
|---|--------|--|
| DEK KeySlot Length | 1 | Length of DEK KeySlot, it is 1. |
| DEK KeySlot | 1 | It is one byte, and value is 0-9 |
| DEK DUKPT Management Mode | 1 | 00h: TDES DUKPT Management 01h: AES DUKPT Management [#1] #1: Option is not currently implemented. |
| TDES DUKPT Output Mode or AES DUKPT Working Key | 1 | If TDES DUKPT Management: 00h: TDES 01h: AES-128 (Strong is AES-112) 02h: TransArmor TDES If AES DUKPT Management: [#1] 00h: TDES Working Key 01h: TDES3 Working Key |

| Data Item | Length | Description |
|--|--------|--|
| | | 02h: AES-128 Working Key 03h: AES-192 Working Key 04h: AES-256 Working Key #1: Option is not currently implemented. |
| TDES DUKPT Variant or AES DUKPT KeyUsage | 1 | If TDES DUKPT Management: 00h: Data Key (Data Encryption, request or both ways) 01h: PIN Key (PIN Encryption) If AES DUKPT Management: [#1] 00h: Data Key both ways (_Data-Encryption-both-ways_) 01h: Data Key encrypt (_Data-Encryption-encrypt_) 02h: Data Key decrypt (_Data-Encryption-decrypt_) 03h: MAC Key both ways (_Message-Authentication-both-ways_) 04h: MAC Key generation (_Message-Authentication-generation_) 05h: MAC Key verification (_Message-Authentication-verification_) #1: Option is not currently implemented. |
| RFU | 3 | Currently, this field should be 3 bytes 00h |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00h | 00h | | |

Examples:

1. Set DEK 0 is TDES DUKPT management & TransArmor TDES & Data Key: 01 00 00 02 00 00 00 00
2. Set DEK 3 is TDES DUKPT management & TransArmor TDES & PIN Key: 01 03 00 02 01 00 00 00
3. Set DEK 9 is TDES DUKPT management & AES-128 & Data Key: 01 09 00 01 00 00 00 00
4. Set DEK 6 is AES DUKPT management & AES-128 & Data Key both ways: 01 06 01 02 00 00 00 00

7.22.23 Get DUKPT DEK Attribution based on KeySlot (C7-A3)

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data length (MSB) | Data length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | C7h | A3h | 00 | var | See Data Table | | |

Data Table

| Data Item | Length | Description |
|--------------------|--------|----------------------------------|
| DEK KeySlot Length | 1 | Length of DEK KeySlot, it is 1. |
| DEK KeySlot | 1 | It is one byte, and value is 0-9 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | C7h | See status code table | 00h | 06h | See Data Table | | |

Data Table

| Data Item | Length | Description |
|---|--------|--|
| DEK DUKPT Management Mode | 1 | 00h: TDES DUKPT Management 01h: AES DUKPT Management |
| TDES DUKPT Output Mode or AES DUKPT Working Key | 1 | If TDES DUKPT Management: 00h: TDES 01h: AES-128 (Strong is AES-112) 02h: TransArmor TDES If AES DUKPT Management: 00h: TDES Working Key 01h: TDES3 Working Key 02h: AES-128 Working Key 03h: AES-192 Working Key 04h: AES-256 Working Key |
| TDES DUKPT Variant or AES DUKPT KeyUsage | 1 | If TDES DUKPT Management: 00h: Data Key (Data Encryption, request or both ways) 01h: PIN Key (PIN Encryption) If AES DUKPT Management: 00h: Data Key both ways (_Data-Encryption-both-ways_) 01h: Data Key encrypt (_Data-Encryption-encrypt_) 02h: Data Key decrypt (_Data-Encryption-decrypt_) |

| Data Item | Length | Description |
|-----------|--------|--|
| | | 03h: MAC Key both ways (_ Message-Authentication-both-ways_) 04h: MAC Key generation (_ Message-Authentication-generation_) 05h: MAC Key verification (_ Message-Authentication-verification_) |
| RFU | 3 | 3 bytes 00h |

7.22.24 Set KeySlot for DUKPT PEK and DUKPT DEK (C7-A4)

The **Set KeySlot for PEK and DEK** command selects:

- one of the available PEKs for use in subsequent PIN encryption operations.
- one of the available DEKs for use in subsequent Data encryption operations.

The KeySlot number specifies the key this command will select. The KeySlot number will also be saved.

7.23.25.1. PEKs

If the following conditions are true, this command selects the PEK associated with the specified KeySlot as the current PEK:

- The KeySlot is a valid KeySlot for PEKs.
- The key exists and has not reached End of Life.
- There is no PEK KeySlot TLV in a transaction command, or there is no PEK KeySlot parameter in a Get PN command.

If the conditions above are not met, the command returns a failed status.

Readers that support this command can have 10 PEKs, starting from KeySlot number 0.

7.23.25.2. DEKs

The DEK associated with the specified KeySlot will be selected as the current DEK if the following conditions are true, otherwise the command will return with a failed status:

- The KeySlot is a valid KeySlot for DEKs.
- The key exists and has not reached End of Life.
- In transaction command, there is Not DEK KeySlot TLV.

If the conditions above are not met, the command returns a failed status.

Readers that support this command can have 10 DEKs, starting from KeySlot number 0.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data length (MSB) | Data length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | A4h | 00 | var | See Data Table | | |

Data Table

| Data Item | Length | Description |
|--------------------|--------------|---|
| PEK KeySlot Length | 1 | If select PEK, this field is 01h. If there is not PEK or don't select PEK, this field is 00h. |
| PEK KeySlot | None or 1 | If PEK KeySlot Length is 00h, this field is not present. If PEK KeySlot Length is 00h, this field is 0-9 |
| DEK KeySlot Length | 1 | If select DEK, this field is 01h. If there is not DEK or don't select DEK, this field is 00h. |
| DEK KeySlot | None or 1 | If DEK KeySlot Length is 00h, this field is not present. If DEK KeySlot Length is 00h, this field is 0-9 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------------|---------|---------------------------------------|-------------------------|----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00 | 00 | | |

Examples:

1. Set Device use PEK 0 & DEK 1: 01 00 01 01
2. Set Device use PEK 5 & DEK 3: 01 05 01 03
3. Set Device use DEK 3 only: 00 01 03
4. Set Device use PEK 3 only: 01 03 00

7.22.25 Get KeySlot for DUKPT PEK and DUKPT DEK (C7-A5)

The **Get KeySlot for PEK and DEK** command returns information for the currently-selected KeySlot.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data length (MSB) | Data length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | A5h | 00 | 00 | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00 | var | See Data Table | | |

Data Table

| Data Item | Length | Description |
|--------------------|--------|--|
| PEK KeySlot Length | 1 | This field is 01h. |
| PEK KeySlot | 1 | If there is not PEK or don't select PEK, this field is 00h. If PEK is selected, this field is 0-9 |
| DEK KeySlot Length | 1 | This field is 01h. |
| DEK KeySlot | 1 | If there is not DEK or don't select DEK, this field is 00h. If DEK is selected, this field is 0-9 |

7.23 Key Injection and Related Commands

7.23.1 Get DUKPT Key KSN Extended (81-0B)

Use this command to get DUKPT Key KSN (Extended) according to Key Name Index and Key Slot.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14,15 | Byte 16 | Byte 17 |
|-------------------------------|---------|-------------|-------------------|-------------------|------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 81h | 0Bh | 03h | | See Tables Below | | |

Command Data Field

| Data Field | Length (bytes) | Description |
|----------------|----------------|---|
| Key Name Index | 1 | 0x14 – LCL-KEK 0x01 – Pin encryption Key 0x02 – Data encryption Key 0x05 – MAC DUKPT Key 0x0A – PCI Pairing Key |
| Key Slot | 2 | Indicate different slots of a certain Key Name Example: slot=5 (0x00 0x05), slot=300 (0x01 0x2C) For BTPay380, slot is always 0 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 81h | See status code table | 01h or 0Bh | | See Tables Below | | |

Response Data Field

| Data Field | Length (bytes) | Description |
|------------|----------------|--|
| Key State | 1 | Key State for the DUKPT key associated with the slot. Possible values for the key state are: 00h: Unused (Slot is supported but no key injected) 01h: Valid (A valid key is available in this slot) 02h: End of life (The key on their slot has reached end of life) FFh: Not Available (This slot is not supported) |
| KSN Length | 1 | Value is 10 to indicate TDES PIN DUKPT Key, Value is 12 to indicate AES PIN DUKPT Key |
| KSN | 10 or 12 | DUKPT Key KSN (Hex) Length is 10 bytes Hex to indicate TDES PIN DUKPT Key, Length is 12 bytes Hex to indicate AES PIN DUKPT Key |

7.23.2 Set and Get MAC Verification Output Option (81-40)

Use this command to set and get MAC verification option (ON/OFF) for transaction output. If MAC verification option is ON. Reader will send out transaction result with MAC signature. If MAC verification option is OFF. Reader will send out transaction result without MAC signature.

Note: MAC verification option is defaulted to be OFF. If the option is ON, it can't be turned OFF.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14,15 | Byte 16 | Byte 17 |
|-------------------------------|---------|-------------|-------------------|-------------------|------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTEch2\0 | 81h | 40h | 01h | | See Tables Below | | |

Command Data Field

| Data Field | Length (bytes) | Description |
|---------------------------------------|----------------|---|
| Operation for MAC Verification Option | 1 | 0x00 – Just Get Current Option Status 0x01 – Set Option to be ON |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|--------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 81h | See status code table | 01h | | See Tables Below | | |

Response Data Field

| Data Field | Length (bytes) | Description |
|---|-------------------|---|
| Current MAC Verification Option Status | 1 | 0x00 – Current Option Status is OFF 0x01 - Current Option Status is ON |

7.24 PIN Pad (Keypad) Commands**7.24.1 Secondary Response Mechanism**

Except for **62-00**, all 62-xx commands use the secondary response mechanism.

When a reader receives a 62-XX command, it immediately sends an ACK response; the reader sends a second response after completing the command's operation or if it times out.

The reader only accepts the 62-00 during the operation (that is, the interval between the first and second response), which can cancel the 62-xx command that is running; other commands respond with the fail (0x0A) status code immediately.

7.24.2 Status and State Codes in Secondary Response

The second response's data field contains a state code that indicates the information detailed below. The data field is presented only when the status code is **0x00** or **0x0A**.

00h: Success
 01h: Fail, Key Pad Cancel
 02h: Fail, External Command Cancel
 03h: Fail, Invalid input parameters
 04h: Fail, PAN error
 05h: Fail, PEK (DUKPT / MKSK) is absent
 06h: Fail, PEK (DUKPT) is exhausted
 07h: Fail, Display message error
 08h: Fail, Removal Detection is disabled
 09h: Fail, PINs are not matched
 0Ah: Fail, PEK (MKSK) reached limit

0Fh: Fail, Feature is not supported

10h: Fail, Financial card is not allowed

If the status code is other than 0x00 or 0x0A, such as **0x08** (timeout) or **0x0C** (operation is not allowed), the data field is empty.

7.24.3 Cancel PIN or Key Entry (62-00)

The **Cancel PIN or Key Entry** command to cancels the following commands:

- **Display Message and Get Encrypted PIN (62-01)**
- **Get Function Key (62-02)**
- **Display Message and Get Numeric Key (62-03)**
- **Display Message and Get Amount (62-04)**
- **Display Multi Line Message and Get Numeric Key (62-05)**

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 62h | 00h | 00h | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 62h | See status code table | 00h | | | |

7.24.4 Display Message and Get Encrypted PIN (62-01)

The Display Message and Get Encrypted PIN command gets encrypted PINs from readers that allow PIN entry (such as VP3600).

NOTE: MKSK allows only one PIN attempt per 30 seconds.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 62h | 01h | var | | See Tables Below | | |

Command Data Field

| Data Field | Length (bytes) | Description |
|--------------------------------|----------------|--|
| PAN and Key Type | 1 | <p>00h: MKSK to encrypt PIN, Internal PAN (from MSR or Manual PAN Entry or Contactless EMV Transaction)</p> <p>01h: DUKPT to encrypt PIN, Internal PAN (from MSR or Manual PAN Entry or Contactless EMV Transaction)</p> <p>03h: DUKPT to encrypt PIN, Internal PAN (from MSR or Manual PAN Entry or Contactless EMV Transaction), with KeySlot Selection</p> <p>10h: MKSK to encrypt PIN, External Plaintext PAN</p> <p>11h: DUKPT to encrypt PIN, External Plaintext PAN</p> <p>13h: DUKPT to encrypt PIN, External Plaintext PAN, with KeySlot Selection</p> <p>20h: MKSK to encrypt PIN, External Ciphertext PAN (for PIN pad only)</p> <p>21h: DUKPT to encrypt PIN, External Ciphertext PAN (for PIN pad only)</p> <p>23h: DUKPT to encrypt PIN, External Ciphertext PAN (for PIN pad only) , with KeySlot Selection</p> <p>80h: MKSK to encrypt PIN, Internal PAN, Verify PIN (from MSR or Manual PAN Entry or Contactless EMV Transaction)</p> <p>81h: DUKPT to encrypt PIN, Internal PAN, Verify PIN (from MSR or Manual PAN Entry or Contactless EMV Transaction)</p> <p>83h: DUKPT to encrypt PIN, Internal PAN, Verify PIN (from MSR or Manual PAN Entry or Contactless EMV Transaction) , with KeySlot Selection</p> <p>90h: MKSK to encrypt PIN, External Plaintext PAN, Verify PIN</p> <p>91h: DUKPT to encrypt PIN, External Plaintext PAN, Verify PIN</p> <p>93h: DUKPT to encrypt PIN, External Plaintext PAN, Verify PIN, with KeySlot Selection</p> <p>Note: MKSK is either standard, or Desjardins-based, according to Key Loading. DUKPT is TDES or AES according to Key Loading. Verify PIN require user to enter PIN twice, and only matched PIN is processed.</p> |
| PAN length | 1 | <p>Length of PAN (If Internal PAN, this field is 00h)</p> <p>If TDES PIN DUKPT Key, value is 16 bytes digital</p> <p>If AES PIN DUKPT Key, value is 12~19 bytes digital</p> |
| PAN | var | <p>If PAN is 00h, this field is not present.</p> <p>ASCII code for digital (30h-39h).</p> <p>If TDES PIN DUKPT Key, length is 16 bytes.</p> <p>If AES PIN DUKPT Key, length is 12~19 bytes.</p> <p>The reader will check PAN length; if length is error, the reader will respond with fail state code.</p> |
| PIN Max Length | 1 | Max Len of PIN, valid range is $4 \leq \text{min} \leq \text{max} \leq 12$ |
| PIN Min Length | 1 | Min Len of PIN, valid range is $4 \leq \text{min} \leq \text{max} \leq 12$ |
| 1 st LCD Msg Length | 1 | <p>Length of 1st scenario LCD message, valid in 00h~21h (0~33).</p> <p>If no LCD message input, length is 00h, and display default msg "PLEASE ENTER PIN"</p> |

| Data Field | Length (bytes) | Description | | | | | | | | | |
|-------------------------------|---------------------|--|--|---------|----------|--------|--|---------------------|--------|---------------------|---------------------|
| 1 st LCD Message | var | <p>If 1st LCD Msg Length is 00h, this field is not present.</p> <p>ASCII code Display Message.</p> <p>Support two lines to display.</p> <p>Each line supports 1~16 characters.</p> <p>'\0' (0x00) necessary to separate the 1st and 2nd msg;</p> <table border="1"> <tr> <td></td><td>One msg</td><td>Two msgs</td></tr> <tr> <td>Line 1</td><td></td><td>1st msg</td></tr> <tr> <td>Line 2</td><td>1st msg</td><td>2nd msg</td></tr> </table> | | One msg | Two msgs | Line 1 | | 1 st msg | Line 2 | 1 st msg | 2 nd msg |
| | One msg | Two msgs | | | | | | | | | |
| Line 1 | | 1 st msg | | | | | | | | | |
| Line 2 | 1 st msg | 2 nd msg | | | | | | | | | |
| 2nd LCD Msg Length | 1 | <p>Length of 2nd Scenario LCD message. Valid in 00h~21h (0~33).</p> <p>This field is present only when PAN and Key Type has Verify PIN.</p> <p>If no LCD message input, length is 00h, and display default msg " ENTER PIN AGAIN"</p> | | | | | | | | | |
| 2nd LCD Message | var | <p>If 2nd LCD Msg Length is 0, this field is not present.</p> <p>ASCII code Display Message.</p> <p>Support two lines display.</p> <p>Each line supports 1~16 characters.</p> <p>'\0' (0x00) is necessary to separate the 1st and 2nd msg;</p> <table border="1"> <tr> <td></td><td>One msg</td><td>Two msgs</td></tr> <tr> <td>Line 1</td><td></td><td>1st msg</td></tr> <tr> <td>Line 2</td><td>1st msg</td><td>2nd msg</td></tr> </table> | | One msg | Two msgs | Line 1 | | 1 st msg | Line 2 | 1 st msg | 2 nd msg |
| | One msg | Two msgs | | | | | | | | | |
| Line 1 | | 1 st msg | | | | | | | | | |
| Line 2 | 1 st msg | 2 nd msg | | | | | | | | | |
| PEK KeySlot Length (Optional) | 1 | <p>This field is present only when PAN and Key Type has KeySlot Selection.</p> <p>00h: Use default KeySlot in device configuration, if device configuration is not available, KeySlot 0 shall be applied.</p> <p>01h: Use KeySlot in next field</p> | | | | | | | | | |
| PEK KeySlot (Optional) | 1 | <p>This field is present only when PEK KeySlot Length is available and not equal to 0.</p> <p>Available from 00h to 09h</p> <p>This KeySlot does NOT change KeySlot in device configuration</p> | | | | | | | | | |

First Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | 62h | See status code table | 00h | | | |

After the reader receives this command, it will send ACK as the first response. The reader will begin to wait for PIN entry. When a PIN is entered or timeout occurs, the reader will send a PIN Block or timeout as the second response. The reader only accepts the Cancel PIN or Key Entry command (62-00) between the first and second responses.

Note:

- Timeout for each enter pin is 3 minutes, thus overall timeout for Verify PIN is 6 minutes
- Per 20 Seconds, if the PIN length is not zero, the PIN is clear
- the default PIN Length is 4~12
- After the user presses a numeric key, "*" is displayed on LCD as follows:
 - 2 numeric keys are pressed: **
 - 12 numeric keys are pressed: *****
- When the user presses the backspace key, the last "*" on LCD is cleared (if any).
- When the user presses the cancel key is pressed, the LCD is cleared, and the operation is canceled with state code 01h (Fail, Key Pad Cancel) in the second response.

Second Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|----------------------|-------------------------|--------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 62h | See status code table | var | | See Tables Below | | |

- TDES DUKPT Key mode
 - Build Format 0 PIN Block with PAN. The PIN Block is encrypted using TDES algorithm and PEK (DUKPT).
- AES DUKPT Key mode:
 - Build Format 4 PIN Block with PAN. The PIN Block is encrypted using AES-128 algorithm and PEK (DUKPT).
- MKSK mode:
 - Build Format 0 PIN Block with PAN. The PIN Block is encrypted using TDES algorithm & PEK (MKSK – Session Key).

Response Data Field

| Data Field | Length (bytes) | Description |
|---------------------|----------------|--|
| State | 1 | State of get encrypted PIN process See "State Code in PIN and Key Entry Commands" |
| PIN DUKPT Key KSN | 20 or 24 | If State byte isn't 00h, this field is not present If MKSK mode, this field is not present Length is 20 bytes ASCII code to indicate TDES PIN DUKPT Key Length is 24 bytes ASCII code to indicate AES PIN DUKPT Key |
| Encrypted PIN Block | 16 or 32 | If State byte isn't 00h, this field is not present Length is 16 bytes ASCII code to indicate PIN Block is TDES encrypted by PEK (DUKPT / MKSK) Length is 32 bytes ASCII code to indicate PIN Block is AES-128 encrypted by PEK (DUKPT) |

7.24.5 Get Function Key (62-02)

Use this command to capture a single Keypress value from a reader.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 62h | 02h | 00h | | | |

First Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 62h | See status code table | 00h | | | |

After a reader receives this command, it will send ACK as first response. The reader will then begin to wait for Function Key. When Function Key is entered or timeout, the reader will send Function Key value or timeout as second response. Between first and second response, no command should be sent (other than Cancel).

Second Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|--------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 62h | See status code table | 02h or 03h | | See Tables Below | | |

Response Data Field

| Data Field | Length (bytes) | Description |
|--------------------|-------------------|---|
| State | 1 | State of get function key process See "State Code in PIN and Key Entry Commands" |
| Function Key Value | 1 or 2 | If State byte isn't 00h, this field is not present ASCII code Length of "B", "C", "E", "*" or "#" is 1 byte Length of "F1" ~ "F4" is 2 bytes |

Note:

- Timeout is 3 minutes.
- While you press Backspace key, the reader sends "B"
- While you press Cancel key, the reader sends "C"
- While you press Enter key, the reader sends "E"
- While you press '*' key, the reader sends "*"
- While you press '#' key, the reader sends "#"
- While you press 'F1' key, the reader sends "F1"
- While you press 'F2' key, the reader sends "F2"
- While you press 'F3' key, the reader sends "F3"
- While you press 'F4' key, the reader sends "F4"

7.24.6 Display Message and Get Numeric Key (62-03)

Use this command to make the reader display requested message and then get Numeric Key sequence from the reader.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|--------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 62h | 03h | var | | See Tables Below | | |

Command Data Field

| Data Field | Length (bytes) | Description |
|------------------------------|-------------------|---|
| Display Flag | 1 | 0 - Display numeric for numeric key on LCD 1 - Display "*" for numeric key on LCD |
| Key Max Length | 1 | The max length for numeric key, valid range is $1 \leq \text{min} \leq \text{max} \leq 16$ |
| Key Min Length | 1 | The min length for numeric key, valid range is $1 \leq \text{min} \leq \text{max} \leq 16$ |
| Display Message | 1~16 | Plaintext Display Message. ASCII code. |
| Display Message Signature | 256 | Display message signed by Private Key of Secure Message Certificate using RSAPSS algorithm. Algorithm: <ol style="list-style-type: none"> 4. Calculate 32 bytes Hash for "< Display Flag >< Key Max Length >< Key Min Length >< Display Message >" 5. Using RSAPSS algorithm calculate the Hash to be 256 bytes Raw Data 6. Using Private Key of Secure Message Certificate to sign the Raw Data to be 256 bytes signature |

First Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------------|---------|---------------------------------------|-------------------------|-------------------------|--------------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 62h | See status code table | 00h | | | |

After the reader receives this command, it will send ACK as first response. Then the reader will begin to wait for Numeric Key press. When Numeric Key is entered or timeout, the reader will send Numeric Key value or timeout as second response. Between first and second response, no command should be sent (other than Cancel).

Second Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|--------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 62h | See status code table | 1 or 17 | | See Tables Below | | |

Response Data Field

| Data Field | Length (bytes) | Description |
|-------------------|-------------------|---|
| State | 1 | State of get function key process See "State Code in PIN and Key Entry Commands" |
| Numeric Key Value | 16 | If State byte isn't 00h, this field is not present. Encoded as ASCII. |

After host device sends this command, the reader is waiting for entering Numeric Key.
16 bytes Numeric Key Value is ASCII Code, padding with 0x46 ('F').
<Key0><Key1><Key2>.....<Key15>.

Example: enter 7 numeric keys: 2 5 7 8 9 0 6, response data field is 16 bytes, padded with 0x46: 0x32 0x35 0x37 0x38 0x39 0x30 0x36 0x46 0x46 0x46 0x46 0x46 0x46 0x46 0x46 0x46.

Note:

- Timeout is 3 minutes.
- Per 20 Seconds, if the Numeric length was not 0, the Numeric would be clear and will display cursor on LCD.
- While you press numeric key, the reader will increase display numeric on LCD if key length is less than Key Max Length. Line 1 display:
 - If Enter 2 numeric (12): 12 or **
 - If Enter 16 numeric (1234567890123456): 1234567890123456 or

- While you press Backspace key, the reader will decrease display numeric on LCD if Total numeric length is not 0.
- While you press Cancel key, the reader will display cursor on LCD if Total numeric length is not 0, or the reader will quit the work state.

7.24.7 Display Message and Get Amount (62-04)

Use this command to make the reader display requested message and then get **amount** value from the reader.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|--------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 62h | 04h | var | | See Tables Below | | |

Command Data Field

| Data Field | Length (bytes) | Description |
|---------------------------|-------------------|---|
| Display Flag | 1 | reserved |
| Amount Max Length | 1 | The max length for amount, valid range is $1 \leq \text{min} \leq \text{max} \leq 15$ |
| Amount Min Length | 1 | The min length for amount, valid range is $1 \leq \text{min} \leq \text{max} \leq 15$ |
| Display Message | 1~16 | Plaintext Display Message. ASCII code. |
| Display Message Signature | 256 | Display message signed by Private Key of Secure Message Certificate using RSAPSS algorithm. Algorithm: <ol style="list-style-type: none"> 1. Calculate 32 bytes Hash for "< Display Flag >< Amount Max Length >< Amount Min Length >< Display Message >" 2. Using RSAPSS algorithm calculate the Hash to be 256 bytes Raw Data 3. Using Private Key of Secure Message Certificate to sign the Raw Data to be 256 bytes signature |

First Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------------|---------|---------------------------------------|-------------------------|-------------------------|--------------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 62h | See status code table | 00h | | | |

After the reader receives this command, it will send ACK as first response. Then the reader will begin to wait for Enter Amount. When Amount is entered or timeout, the reader will send Amount value or timeout as second response. Between first and second response, no command should be sent (other than Cancel).

Second Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|--------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 62h | See status code table | 1 or 16 | | See Tables Below | | |

Response Data Field

| Data Field | Length (bytes) | Description |
|--------------|-------------------|---|
| State | 1 | State of get function key process See "State Code in PIN and Key Entry Commands" |
| Amount Value | 15 | If State byte isn't 00h, this field is not present ASCII code |

After the host device sends this command, the reader is waiting for entering Numeric Key to indicate the amount.

16 bytes Numeric Key Value is ASCII Code, padding with 0x46.

<Key0><Key1><Key2>.....<Key14>.

Example: enter 7 numeric keys: 2 5 7 8 9 0 6, amount is 25789.06, response data field is 0x32 0x35 0x37 0x38 0x39 0x30 0x36 0x46 0x46 0x46 0x46 0x46 0x46 0x46 0x46 0x46.

Note:

- Timeout is 3 minutes.
- Per 20 Seconds, if the amount length was not 0, the amount would be clear and will display cursor on LCD.
- While you press numeric key, the reader will increase display amount on LCD if amount length is less than Amount Max Length. Line 1 display:
 - If Enter 2 numeric (12): 0.12
 - If Enter 15 numeric (12345678901245): 1234567890123.45
- While you press Backspace key, the reader will decrease display amount on LCD if Total amount length is not 0.
- While you press the Cancel key, the reader will display cursor on LCD if amount length is not 0, or the device will quit the work state.

7.24.8 Display Multi Line Message and Get Numeric Key (62-05)

Use this command to make the reader display requested message and then get Numeric Key sequence from the reader.

This command also allow to capture the key sequence by swiping a MSR non-financial card.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|--------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 62h | 05h | var | | See Tables Below | | |

Command Data Field

| Data Field | Length (bytes) | Description |
|---------------------------|-------------------|--|
| Function Flag | 1 | 0x00 - Display numeric for numeric key on LCD 0x01 - Display "*" for numeric key on LCD 0x02 - Display numeric first and then masked by "*" on LCD 0x10 - Enable MSR & Display numeric for numeric key on LCD 0x11 - Enable MSR & Display "*" for numeric key on LCD 0x12 - Enable MSR & Display numeric first and then masked by "*" on LCD Note: Only support Non-Financial Card (such as Employee Card) |
| Key Max Length | 1 | The max length for numeric key, valid range is $1 \leq \text{min} \leq \text{max} \leq 16$ |
| Key Min Length | 1 | The min length for numeric key, valid range is $1 \leq \text{min} \leq \text{max} \leq 16$ |
| Display Message Length | 1 | The length of Display Message, valid range is 1~33 |
| Display Message | 1~33 | Plaintext and supports two lines. Each line supports 1~16 characters; '\0' (0x00) is necessary to separate the 1 st and 2 nd line |
| Display Message Signature | 256 | Display message signed by Private Key of Secure Message Certificate using RSAPSS algorithm. Algorithm: <ol style="list-style-type: none"> Calculate 32 bytes Hash for "< Function Flag >< Key Max Length >< Key Min Length >< Display Message Length >< Display Message >" Using RSAPSS algorithm calculate the Hash to be 256 bytes Raw Data Using Private Key of Secure Message Certificate to sign the Raw Data to be 256 bytes signature |
| Timeout | 2 | In seconds, big-endian. If 00h, default 3 mins timeout is applied |

First Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 62h | See status code table | 00h | | | |

After the reader receives this command, it will send ACK as first response. Then the reader will begin to wait for Numeric Key entry, or MSR swipe if it is enable. Either Numeric Key is entered, MSR is swiped or timeout, the reader will send Numeric Key value, MSR data or timeout as second response. Between first and second response, no command should be sent (other than Cancel).

Second Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 62h | See status code table | var | | See Tables Below | | |

Response Data Field

| Data Field | Length (bytes) | Description |
|-------------|----------------|---|
| State | 1 | State of get function key process See "State Code in PIN and Key Entry Commands" |
| Data source | 1 | If State byte isn't 00h, this field is not present. 0x00 – Followed Tracks Data 0x01 – Followed Manual Entry |
| Value | n | If State byte isn't 00h, this field is not present. If Data source is 0x00, this field is Tracks Data of MSR card. If Data source is 0x01, this field is Manual Entry Data. |

After the host device sends this command, the reader is waiting for entering Numeric Key & Swiping MSR Card.

- If Swipe MSR Card, Keypad entry is disabled.
- If press "Enter" key, MSR is disabled.

If Swipe MSR Non-Financial Card (such as Employee Card), the Value Field will be Plaintext Tracks Data from the Card.

Example: If Micros POS Employee Card Data is ; : <>4274053177?>, the Value Field data is 0x3B 0x3A 0x3C 0x3E 0x34 0x32 0x37 0x34 0x30 0x 35 0x33 0x31 0x37 0x37 0x3F 0x3E.

If press Numeric Key OK, the Value Field will be any bytes Numeric Key Value is ASCII Code (Max is 16 or 20).

Example: If Manual entry 10 numeric keys: 4 2 7 4 0 5 3 1 7 7, response data field is 10 bytes: 0x34 0x32 0x37 0x34 0x30 0x35 0x33 0x31 0x37 0x37.

Note:

- Timeout can be changed without re-signing this command.
- Default 3 mins timeout is applied if it is 00h.
- Per 20 Seconds, if the Numeric length was not 0, the Numeric would be clear and will display cursor on LCD.
- While you press numeric key, the reader will increase display numeric on LCD if key length is less than Key Max Length. Line 1 display:
 - If Enter 2 numeric (12): 12 or **
 - If Enter 16 numeric (1234567890123456): 1234567890123456 or *****
- While you press Backspace key, the reader will decrease display numeric on LCD if Total numeric length is not 0.
- While you press Cancel key, the reader will display cursor on LCD if Total numeric length is not 0, or the reader will quit the work state.

7.24.9 Display Specific Message and Get Numeric Key (62-06)

The **Display Specific Message and Get Numeric Key** command displays a specific message and then retrieves the Numeric Key sequence from the the reader.

Note: This command is only valid for the VP6800.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|--------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 62h | 06h | var | | See Tables Below | | |

Command Data Field

| Data Field | Length (bytes) | Description |
|------------------------|----------------|---|
| Display Flag | 1 | 0: Display numeric for numeric key on LCD 1: Display "*" for numeric key on LCD according to pre-cleartext display and post-cleartext display. |
| Pre-Cleartext Display | 1 | Only valid when display flag is 1, valid range is $0 \leq \text{min} \leq \text{max} \leq 6$. |
| Post-Cleartext Display | 1 | Only valid when display flag is 1, valid range is $0 \leq \text{min} \leq \text{max} \leq 6$. |
| Key Max Length | 1 | The maximum length for numeric key. For Message ID 0~6, valid range is $1 \leq \text{min} \leq \text{max} \leq 20$. For Message ID 7, valid range is $1 \leq \text{min} \leq \text{max} \leq 64$. |
| Key Min Length | 1 | The minimum length for numeric key. For Message ID 0~6, valid range is $1 \leq \text{min} \leq \text{max} \leq 20$. For Message ID 7, valid range is $1 \leq \text{min} \leq \text{max} \leq 64$. |
| Display Language | 2 | ASCII code: "EN" is English display message. "JP" is Japanese display message. "ES" is Spanish display message. "FR" is French display message. "ZH" is Chinese display message. "PT" is Portuguese display message. |
| Scenario ID | 1 | 0x00: Enter Phone Number. 0x01: Enter IP Address. 0x02: Enter Subnet Mask. 0x03: Enter Default Gateway. 0x04: Enter Odometer Reading/Mileage. 0x05: Enter Employee ID. 0x06: Enter Password for Default Factory Setting. 0x07: Enter Email Address (Full keyboard). |
| Default String Length | 2 | Length of default string. Format: <Length H><Length L> (Big-endian) Valid range refer to 'Key Min Length' and 'Max Length'. |
| Default String | 1 to var | Default String on input field. No longer than Key Max Length + 1. ASCII string, terminated by 0x00. |
| Timeout | 2 | In seconds, big-endian. If 00h, default 3 mins timeout is applied. |

First Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 62h | See status code table | 00h | | | |

After the VP6800 receives this command, it sends ACK as the first response. The reader then waits for a numeric key press. When a numeric key is entered or times out, the reader sends

the numeric key value or timeout as a second response. Do not send any commands between the first and second response (other than Cancel).

Second Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|--------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | 62h | See status code table | 1 or n | | See Tables Below | | |

Response Data Field

| Data Field | Length (bytes) | Description |
|-------------------|-------------------|--|
| State | 1 | State of the get function key process. See "State Code in PIN and Key Entry Commands" under PIN Pad (Keypad) Commands: Status Codes . |
| Numeric Key Value | n | User input string info, encoded as ASCII. If the State byte isn't 00h, this field is not present. |

After the host device sends this command, the reader waits for numeric key entry.

Example: If a user enters seven numeric keys: 2 5 7 8 9 0 6, the response data is: 0x32 0x35 0x37 0x38 0x39 0x30 0x36 0x00.

7.24.10 Get PAN (83-41)

Issue this command to prompt the user to manually enter a card PAN and Expiry Date (and optionally CSC) from the keypad and return it to the POS. All cardholder data will be in plain text or encrypted form according to the reader encryption status. If a particular configuration is set in DFEE1D, some of the first or last digits of the PAN maybe returned in plain text. For the ExpiryDate, the default format of MM/YY will be used.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 – 17 | Byte 18 | Byte 19 |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|-----------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViV0payV3\0 | '83' | '41' | '00' | '04' | Data Table | | |

Data Table

| Data Item | Length | Description |
|-------------------------------|--------|---|
| Time Out | 2 | Number of seconds that the reader waits for the data entry session to complete, stored as a big-endian number. 0 = no timeout |
| Flags | 2 | See Flags Field Table below |
| DEK KeySlot Length (Optional) | 1 | This field is present only when Bit 6 of Byte 1 in the Flags is 1. 00h: Use the default KeySlot in the reader's device configuration; if the device configuration is not available, KeySlot 0 shall be applied. 01h: Use the KeySlot in the next field. |
| DEK KeySlot (Optional) | 1 | This field is present only when the DEK KeySlot Length is available and not equal to 0. Available from 00h to 09h. This KeySlot does NOT change the KeySlot in the reader's device configuration. |

Command Flags Field Table

| Byte No. | Bit No. | Flag | Value | Meaning |
|----------|---------|-------------|-------|----------------------------|
| 0 | 6 – 7 | RFU | 0 | Reserved for future use |
| | 5 | Mod10 Check | 0 | Do not request Mod10 Check |
| | | | 1 | Request Mod10 Check |
| | 4 | ZIP | 0 | Do not request ZIP |
| | | | 1 | Request ZIP |
| | 3 | ADR | 0 | Do not request ADR |
| | | | 1 | Request ADR |
| | 2 | Exp Date | 0 | Do not request Exp Date |
| | | | 1 | Request Exp Date |
| | 1 | CSC | 0 | Do not request CSC |
| | | | 1 | Request CSC |
| | 0 | PAN | 1 | Request PAN |

| Byte No. | Bit No. | Flag | Value | Meaning |
|----------|---------|-------------------------------|-------|---|
| 1 | 7 | Special processing flag. | 0 | Standard processing. |
| | | | 1 | Special for Customer: 1. Encrypted PAN Only 2. Entry PAN Only If Manual Entry PAN only, output encrypted PAN only. EXP will be 0000 or masked. If Manual Entry of PAN & EXP, output encrypted PAN only. EXP will be Entered data or masked. |
| | 6 | KeySlot Selection requirement | 0 | Without KeySlot Selection |
| | | | 1 | With KeySlot Selection |
| | 0 – 5 | RFU | 0 | Reserved for future use |

If Bit 7 of Byte 1 is 1, PAN Only will be allowed.

If Bit 7 of Byte 1 is 0, PAN Only will be NOT allowed.

Response Frame on Success

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 21 | Byte 22 |
|-------------------------------|---------|-------------|-------------------|-------------------|---------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOPayV3\0 | '83' | '41' | variable | variable | Data | | |

Response Data: Refer to 80000502-001 ID TECH Encrypted Data Output Formats.pdf, available on the [ID TECH Knowledge Base](#).

1.1.3.15. Notes for Using Get PAN (83-41) with SREDKey 2

Situation 1: There are no flags in the Command Data Table.

Behavior: Admin Key (Keypad) is valid at the start of the operation. All Keypad process are the same as Auto Poll mode.

Situation 2: There are flag in the Command Data Table.

Behavior: Admin Key (Keypad) is **Invalid**. All Keypad process are the same as on NEO2 Platform.

Important information: Byte 2 of the flags is **Invalid**.

7.25 Improved Collision Detection

7.25.1 Issues with Standard Collision Detection

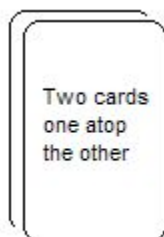
This firmware supports the EMV Contactless Communication Protocol Specification. While the EMV specification defines collision detection, there are often physical constraints which prevent collision detection resolving within the timing limits outlined in the specification. For instance, multiple cards in the field cannot always be detected reliably. If a particular card responds more quickly to RF polling, or if a card has a stronger antenna, then the signal will lock to that card. (This problem is not limited to ID TECH equipment.) Card geometry can also be a major factor in collision detection.

The following bullets explain some of the difficulties associated with multiple card presentation:

- When a single type A and single type B card are presented side-by-side the reader will detect collisions without much difficulty:



- When two cards are stacked on top of one another, the faster card or the card that is closer to the PCD will be activated. This is because the slower card or the card that is further from the PCD suffers insufficient power, interference from the other card, or timing that falls outside the boundaries defined in the EMV specification.



- Presenting two cards of the same type side by side (for example, A || A) will suffer from the same problems described in the previous bullet, because the RF power draw from one card can negatively impact the communication and/or power of the other card.



The six possible multiple card presentation scenarios are listed in the table below.

| Scenario | Card Type | Orientation |
|----------|-----------|-------------|
| 1 | A / B | Stacked |
| 2 | A / B | Fanned |
| 3 | A / A | Stacked |
| 4 | A / A | Fanned |
| 5 | B / B | Stacked |
| 6 | B / B | Fanned |

Note: Card Type B / B (scenarios 5 & 6) are inherently difficult to detect. When multiple type B cards are presented they should detune one another. If there is minimal detuning then no collision would be detected.

7.25.2 Collision Detection Modes

The firmware features two mutually-exclusive collision detection modes, Standard Collision Detection and Improved Collision Detection, which are described in the subsections below.

1.1.3.16. Standard Collision Detection

Standard Collision Detection, also known as EMEA Anti-Collision Detection, is enabled by default, as it maximizes the chances of completing a transaction. In this mode, if a collision event is detected, the reader will back off and resume polling.

Media removal events are handled as per the EMEA requirements for collision. When detecting a collision a media removal UI event is triggered (all LEDs off, 2 tone alert, and appropriate message for LCD equipped units), a small delay is then introduced before returning to polling.

Note: It is important to understand that the procedure outlined above will repeat if the collision is not resolved. Also, the media removal event only operates when the EMEA UI is enabled (tag 'FF F8' = '03'). Finally, this procedure occurs automatically without interaction from the integrated system and will operate for the duration of the Timeout period, as set by [Activate Transaction](#).

The reader will then continue to retry the transaction; until either, the collision issue has been resolved (and a transaction takes place), or until the transaction timeout expires. In the latter case, the timeout will contain the timeout status (0x08) and timeout cause in the data field (0x21, collision error).

Note: In this mode, if a collision is detected, the reader will interpret further 'Communication Error' or 'Card Not Present' events as being caused by collision.

1.1.3.17. Improved Collision Detection

Enabling this mode disables Standard Collision Detection mode and changes reader behavior:

1. While polling, the reader will attempt to find the PICC using the standard polling method.
2. If a collision is detected, the reader will abort the transaction and notify the POS.
3. It will report this with an error in the data field of its response (0x21, collision error).

Tag DF7F enables/disables Improved Collision Detection. When this tag is set to zero (default value), Improved Collision mode is disabled. When this tag is set to another value (1-255), Improved Collision mode is enabled.

When Improved Collision mode is enabled, the DF7F tag value defines the number of successful sequential polling attempts required for signal lock. For example, if tag DF7F = 3, then the reader must detect a card successfully three times in a row before the firmware decides this is a successful polling attempt. Given the same conditions, the reader must fail to detect a card three times in a row before the firmware decides this is a 'card not present' polling attempt.

To reiterate, Improved Collision Detection requires a specified number of polling attempts to complete without an EMV collision event before the RF signal is locked to a specific card. If an EMV collision event is reported, the transaction will end and return a collision status code.

The following table summarizes the tag-related information provided above.

| Feature | Tag | Length | Value | Notes |
|------------------------------|------|--------|-------------|--|
| Improved Collision Detection | DF7F | 1 byte | 0 = default | Improved Collision Detection Disabled (one successful polling attempt is sufficient for signal lock). |
| | | | 1 – 255 | Improved Collision Detection Enabled. Number of successful sequential polling attempts required for signal lock. |

In this new mode, the collision scenarios have been improved in the following manner:

1. Increased sensitivity to improve collision detection generally. Previously, silent card and garbled receive were not identified in the same manner as a standard collision.
2. If the transaction times out due to any of the collision methods described previously, the serial response will reflect this in its error state, with Status Code 0x08 (Time Out) and Error Code 0x21 (Collision Error).

Example

Assume a reader has enabled Improved Collision Detection with tag DF7F = 3. When two cards are placed within view of the reader, the following polling results are obtained.

| | | | |
|-----------------|----|----|-----|
| Polling Attempt | 1 | 2 | 3 |
| Polling Result | OK | OK | L1C |

These results show that the first and second polling attempts are successful; but, the third polling attempt reports an EMV L1 collision (for example, a slower or weaker signal). This collision detection would result in immediate termination of the transaction and the reader returning a collision status code. In contrast, if Standard Collision Detection mode was enabled instead, then the reader would accept the first attempt and the transaction would proceed with the first card detected.

7.26 Firmware Downloader Commands

7.26.1 Contact Get EMV L2 Kernel Link Type (60-17)

Contact Common EMV L2 kernel can be linked into product firmware, or separately loaded as an executable firmware module in the reader. This command can get the kernel link type in reader.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTEch2\0 | 60h | 17h | 00h | 00h | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTEch2\0 | 60h | See status code table | | | Response Data | | |

Response Data:

0x00 -- The EMV L2 kernel is internal.

0x01 -- The EMV L2 kernel is external (separately loaded).

7.26.2 Contact Set EMV L2 Kernel Link Type (60-18)

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 60h | 18h | 00h | 01h | Link Type | | |

Link Type:

0x00 -- The EMV L2 kernel is internal.

0x01 -- The EMV L2 kernel is external (separately loaded).

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 60h | See status code table | 00 | 00 | | | |

7.26.3 Enter Boot Loader Process from Main Application (C7-41)

The host device must use this command to let reader reboot into boot loader mode if reader is running in main application. No response is issued for this command; it just resets the reader immediately.

If reader is running in boot loader, this command is not needed.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|----------------------------------|---------|-------------|-------------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | 41h | 00h | 00h | | |

7.26.4 Retrieve Bootloader Related Information (C7-42)

Use this command to get the detailed information of Bootloader FW.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7 | 42h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 – Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00h | 1Eh | See Data | | |

Data:

| Data Item | Length (bytes) | Description |
|---------------------------|----------------|---|
| Fix String Data1 | 14 | 'Bootloader FW ' 14 bytes Ascii Code Data, Hex Value is 42 6F 6F 74 6C 6F 61 64 65 72 20 46 57 20 |
| BootLoader Indicator Flag | 1 | Indicator which BootLoader is running 'b': Current running in BootLoader 1 'a': Current running in BootLoader 2 |
| Fix String Data2 | 7 | 'Vx.yy' 7 bytes Ascii Code Data, Hex Value of 'V1.00' is 20 56 31 2E 30 30 20 |
| Package SizeLength | 2 | Length of Package Size (<Package Size_L><Package Size_H>) 1K size: 00 04 2K size: 00 08 |
| Request Package Num | 2 | Num of Request Package (<Request Package Num_L><Request Package Num_H>) Hex Value always 00 00 |
| Request Indicator | 4 | Request Indicator (Little Endian Mode) Hex Value always 00 00 00 00 |

7.26.5 Implement Bootloader loading and verification (C7-43)

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 – Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|--------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViV0tech2\0 | C7h | 43h | 00h | | See Below | | |

Data:

| Data Item | Length (bytes) | Description |
|------------|-------------------|---|
| Indicator | 4 | Indicator (Little Endian Mode) |
| Block Data | 256 or 1024 | Type1 – 256 bytes Block 1 Data (Indicator is 00 00 00 00) Type2 – 256 bytes Block 2 Data (Indicator is 01 00 00 00) Type3 – 1024 bytes Block 3 Data (Indicator is Flash Address). It will be m Blocks data Type4 – 256 bytes Block 4 Data (Indicator is FF FF FF FF) |

Response Frame

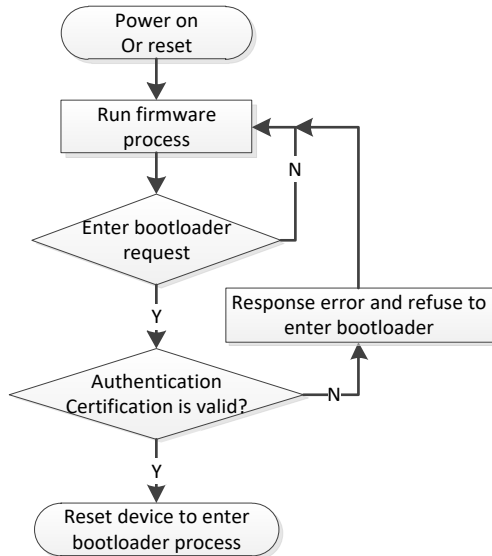
| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 – Byte 14+n-1 | Byte 14+n | Byte 15+n |
|----------------------------------|---------|---|----------------------|----------------------|--------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | C7h | See status code table | 00h | 00h | | | |

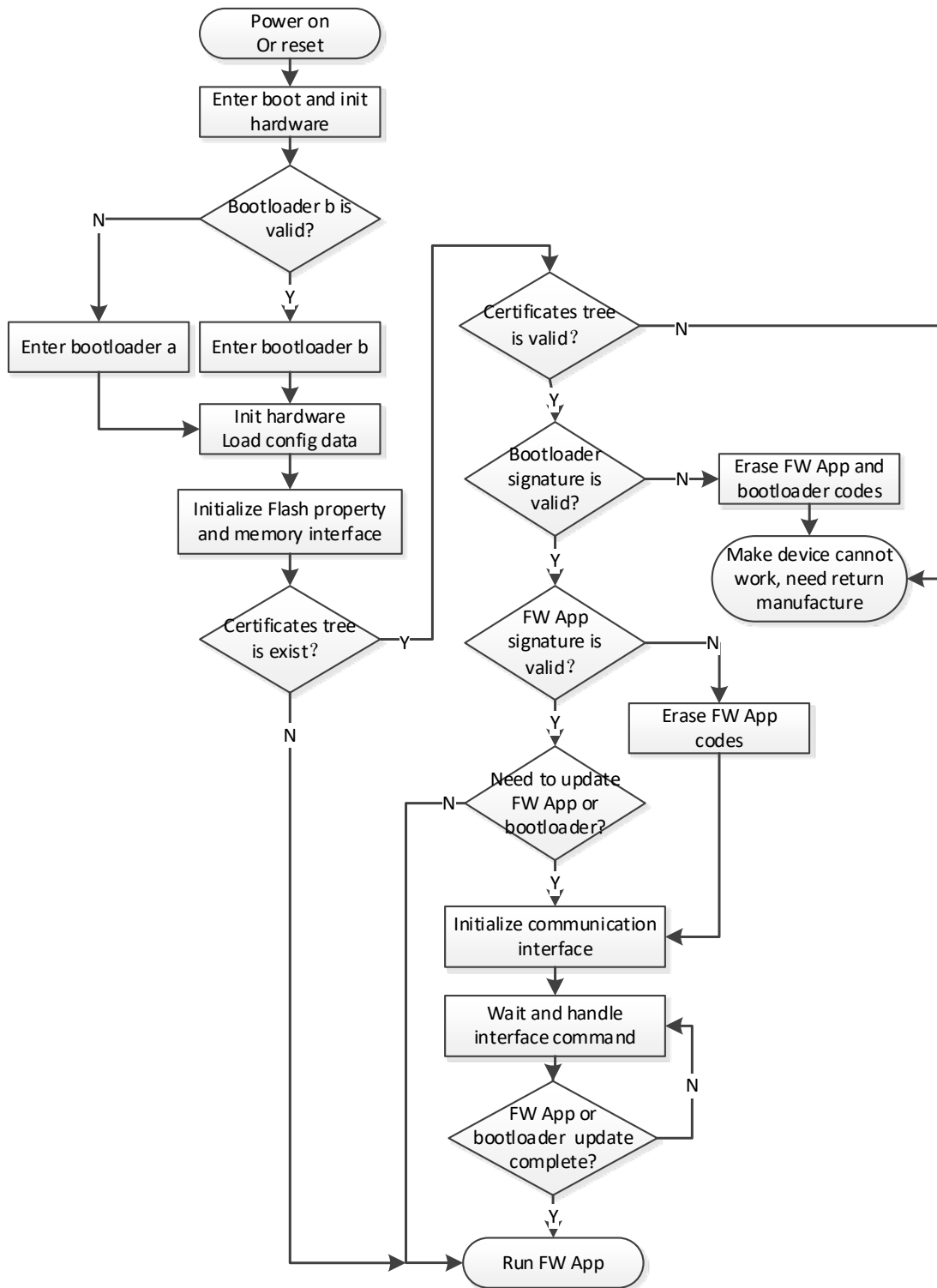
Note: refer to document “NEO_II Bootloader PDS_V52.doc” for detailed BootLoader specifications.

7.26.6 Firmware Downloader Command Processing Flow

Firmware downloader commands must be sent to reader in sequence as below.

(1) Enter Bootloader process



(2) Bootloader process

7.26.7 Start Update Kernel Process (C7-11)

The **Start Update Kernel Process** command is the first command a host device sends to start a kernel or app update process.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-25 | Byte 26 | Byte 27 |
|-------------------------------|---------|-------------|-------------------|-------------------|------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data Field | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | 11h | 00h | 0Ch | See Below | | |

Data Field

| Item | Length in bytes | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------------------|-----------------|---|-------------|----|---------|--------------------------|------|-----|----------------------|------|-----|-----------------------------|------|-----|-----------------------|------|-----|-----------------------|------|-----|----------------------|------|-----|-------------------------|------|-----|---------------------|------|-----|---------------------|------|-----|-------|------|-----|-----------|------|-----|-------|------|-------------------------|---------|------|-----------------------|---------|------|-----------------------|
| Kernel/App Index | 1 | <table> <tr> <th>Kernel Name</th><th>ID</th><th>Support</th></tr> <tr><td>External EMV CTL2 Kernel</td><td>0x00</td><td>Yes</td></tr> <tr><td>EMV CLL2 PPSE Kernel</td><td>0x01</td><td>Yes</td></tr> <tr><td>EMV CLL2 VCPS Kernel (Visa)</td><td>0x02</td><td>Yes</td></tr> <tr><td>EMV CLL2 MChip Kernel</td><td>0x03</td><td>Yes</td></tr> <tr><td>EMV CLL2 DPASS Kernel</td><td>0x04</td><td>Yes</td></tr> <tr><td>EMV CLL2 Amex Kernel</td><td>0x05</td><td>Yes</td></tr> <tr><td>EMV CLL2 Interac Kernel</td><td>0x06</td><td>Yes</td></tr> <tr><td>EMV CLL2 CUP Kernel</td><td>0x07</td><td>Yes</td></tr> <tr><td>EMV CLL2 JCB Kernel</td><td>0x08</td><td>Yes</td></tr> <tr><td>APVAS</td><td>0x09</td><td>Yes</td></tr> <tr><td>Smart Tap</td><td>0x0A</td><td>Yes</td></tr> <tr><td>SCRIP</td><td>0x0B</td><td>Only supported in SCRIP</td></tr> <tr><td>ADF SDK</td><td>0x0C</td><td>Only supported in ADF</td></tr> <tr><td>ADF App</td><td>0x0D</td><td>Only supported in ADF</td></tr> </table> | Kernel Name | ID | Support | External EMV CTL2 Kernel | 0x00 | Yes | EMV CLL2 PPSE Kernel | 0x01 | Yes | EMV CLL2 VCPS Kernel (Visa) | 0x02 | Yes | EMV CLL2 MChip Kernel | 0x03 | Yes | EMV CLL2 DPASS Kernel | 0x04 | Yes | EMV CLL2 Amex Kernel | 0x05 | Yes | EMV CLL2 Interac Kernel | 0x06 | Yes | EMV CLL2 CUP Kernel | 0x07 | Yes | EMV CLL2 JCB Kernel | 0x08 | Yes | APVAS | 0x09 | Yes | Smart Tap | 0x0A | Yes | SCRIP | 0x0B | Only supported in SCRIP | ADF SDK | 0x0C | Only supported in ADF | ADF App | 0x0D | Only supported in ADF |
| Kernel Name | ID | Support | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| External EMV CTL2 Kernel | 0x00 | Yes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| EMV CLL2 PPSE Kernel | 0x01 | Yes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| EMV CLL2 VCPS Kernel (Visa) | 0x02 | Yes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| EMV CLL2 MChip Kernel | 0x03 | Yes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| EMV CLL2 DPASS Kernel | 0x04 | Yes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| EMV CLL2 Amex Kernel | 0x05 | Yes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| EMV CLL2 Interac Kernel | 0x06 | Yes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| EMV CLL2 CUP Kernel | 0x07 | Yes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| EMV CLL2 JCB Kernel | 0x08 | Yes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| APVAS | 0x09 | Yes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Smart Tap | 0x0A | Yes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| SCRIP | 0x0B | Only supported in SCRIP | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ADF SDK | 0x0C | Only supported in ADF | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ADF App | 0x0D | Only supported in ADF | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Data Length | 3 | LSB format: Index 0x00-0D: Kernel or app firmware's total data length. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| App Start Address | 4 | Index 0x00-0B: Not valid, the address is pre-defined in firmware. Index 0x0C-0D: Actual start address. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| RFU | 4 | 0x00000000 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Example 1: Amex Kernel w/ total size of 72,018 bytes (0x011952)

| Byte 1 | Byte 2-4 | Byte 5-8 | Byte 9-12 |
|--------|----------|-------------|-------------|
| 05 | 52 19 01 | 00 00 00 00 | 00 00 00 00 |

Example 2: ADF App w/ total size of 174,036 bytes (0x02A7D4) and address of 0x0004b040

| Byte 1 | Byte 2-4 | Byte 5-8 | Byte 9-12 |
|--------|----------|-------------|-------------|
| 0d | d4 a7 02 | 40 b0 04 00 | 00 00 00 00 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00h | 00h | | |

7.26.8 Send Kernel Firmware Data (C7-14)

This command is sent by the host device to program Kernel Firmware related zone according to ID. The Kernel Firmware data (Firmware Codes + Signature) should be $m \times 2048$ bytes + n bytes.

The first command to m^{th} command are 2048 bytes Kernel Firmware data. The latest command is n bytes Kernel Firmware data.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 - Byte 2065 | Byte 2066 | Byte 2067 |
|-------------------------------|---------|-------------|-------------------|-------------------|---------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | 14h | | | See Below | | |

In this command, data length must be 2052 bytes (Front m commands) or $(n+4)$ bytes (the latest command):

For 2052 bytes (Front m commands):

- The first one byte is Kernel ID.
- The next 3 bytes are reserved.
- The other 2048 bytes are Kernel Firmware data.

For $(n+4)$ bytes (the latest command):

- The first one byte is Kernel ID.
- The next 3 bytes are reserved.
- The other n bytes are Kernel Firmware data.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00h | 00h | | |

7.26.9 End Update Kernel Process (C7-15)

This is the last command sent by host device to close a Kernel Firmware data update process.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 - Byte 21 | Byte 22 | Byte 23 |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | 15h | 00h | 01h | See Below | | |

In this command, data length must be 1 byte Kernel ID.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|----------------------------------|---------|---|-------------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | 00h | 00h | | |

7.27 Application Control Commands

The commands in this section control applications on readers.

7.27.1 ADF Application Control Command (C7-C7)

The **ADF Application Control** command controls ADF applications and starting, stopping, and erasing applications and the SDK area.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14- 15 | Byte 16 | Byte 17 |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data Field | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | C7h | 00h | 02h | See Below | | |

Data Field

| Byte 14 | Byte 15 | Description/Example |
|---------------------------------------|---------|---|
| 0x00: Application Control | 0x00 | Stop all applications |
| | 0x01 | Start all applications |
| | 0x02 | Start the first application only |
| | 0x03 | Start all applications except the first one |
| 0x01: Flash Control | 0x00 | Erase SDK flash |
| | 0x01 | Erase app flash |
| 0x02: Retrieve APP/SDK info list | 0x00 | Retrieve SDKs Info List |
| | 0x01 | Retrieve APPs Info List |
| 0x03: Retrieve APP/SDK first 64 bytes | 0x00 | Retrieve SDKs first 64 bytes |
| | 0x01 | Retrieve APPs first 64 bytes |
| 0xFF: Mode Control | 0x00 | Disable ADF Mode |
| | 0x01 | Enable ADF Mode |
| | 0xFF | Retrieve Mode Status |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVotech2\0 | C7h | See status code table | 00h | 00h | | |

Retrieve Mode Status Response

| Data Field | Length (bytes) | Description/Example |
|------------|----------------|---|
| Status | 2 | 0xFF 0x00: ADF Mode Disabled 0xFF 0x01: ADF Mode Enabled |

Retrieve SDKs/APPs Info List Response

| Data Field | Length (bytes) | Description/Example |
|------------------------|----------------|---|
| Total SDKs/APPs Number | 1 | Total SDKs/APPs Number marks as N |
| SDKs/APPs Info List | 12 * N | Info for each SDK/APP Format: 4bytes: types 4bytes: SDRAM address 4bytes: Runtime size |

Retrieve SDKs/APPs (First 64 Bytes) Response

| Data Field | Length (bytes) | Description/Example |
|------------------------|----------------|---------------------------------------|
| Total SDKs/APPs Number | 1 | Total SDKs/APPs Number marks as N |
| Data of SDK/APPs | 64 * N | First 64 bytes of each SDKs/APPs List |

The response data of other commands not listed in above list is 0 bytes.

7.28 Power Management Commands

7.28.1 Enter Low Power Mode (F0-03)

Note: This command cannot be used on VP5300 during USB mode. It is supported in RS-232 mode.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | F0h | 03h | 0x00 0x02 | | See Command Data Table | | |

Command Data

| Data Item | Length (bytes) | Description/Example |
|------------------|----------------|---|
| Low Power Mode | 1 | 0x00 – Sleep Mode (resume from last instruction) 0x01 – Stop Mode (POR required) |
| MSR / NFC Option | 1 | 0x00 – OFF 0x01 – ON (Swipe/Tap will wake from low power) |

Use 00 01 to put the reader in Sleep Mode. It will awaken upon card presentation. With 00 00, the reader will wake up with card *insertion*, but not with contactless (NFC) presentation. (The card-seated switch is always enabled; it cannot be disabled.) Wakeup from Stop Mode involves a reboot (which takes several seconds). Wakeup from Sleep Mode does not reboot.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | F0h | See status code table | 00h | 00h | | |

The reader will wake up automatically upon card presentation, or when a command is issued. On waking up (after being triggered by card presentation), the reader will output 5669564F746563683200F10000008D1E.

7.28.2 Set Low Power Consumption Configuration (F0-04)

This command is used to Enable/Disable some level Low Power Consumption function and the waiting timer of previous state.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | F0h | 04h | | | See Tables below | | |

| Data Field | Length (bytes) | Description |
|------------------------|----------------|---|
| The low power level ID | 1 | The range of low power ID is 1~4. The detailed supported level is defined by the PDS of the product |
| low power timer | 2 | Time (default 30s) 0000h = NOT support the level of low power consumption function (should be skipped) 0001h ~ 0E10h (1~3600 Seconds) = Support the level of low power consumption function. The value is defined the timer of previous state to 'The low power level ID' level. |

Note:

Default Setting for a product:

| ID | Current Name | Timeout |
|----|---------------|---------|
| 1 | Low speed run | 30s |
| 2 | Standby0 | 30s |
| 3 | Standby1 | 30s |
| 4 | Sleep | 30s |

IDLE -(30s)->LowSpeedRun -(30s)->Standby0 -(30s)->Standby1 -(30s)->Sleep

After send below commands:

- 01 00 1F – Enable Low Speed Run state and previous state (Idle) waiting timer is 31 seconds
- 02 00 00 – Disable Standby0
- 03 00 21 - Enable Standby 1 state and previous state (Low Speed Run) waiting timer is 33 seconds
- 04 00 22 - Enable Sleep state and previous state (Standby 1) waiting timer is 34 seconds

| ID | Current Name | Timeout |
|----|---------------|---------|
| 1 | Low speed run | 31s |
| 2 | Standby0 | 0 |
| 3 | Standby1 | 33s |
| 4 | Sleep | 34s |

IDLE -(31s)-> LowSpeedRun -(33s)-> Standby1 -(34s)-> Sleep

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | F0h | See status code table | 00h | 00h | | | |

7.28.3 Get Low Power Consumption Configuration (F0-05)

This command is used to Get Low Power Consumption function configuration.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | F0h | 05h | 00h | 00h | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | F0h | See status code table | 00h | 09h | See Tables below | | |

| Data Field | Length (bytes) | Description/Example |
|-----------------------------------|----------------|--|
| Modes number | 1 | Total number of supported low power modes. It will be 4. |
| 1 st lower power timer | 2 | |
| ... | | |
| 4 th lower power timer | 2 | |

Note:

If configuration of a product is:

| ID | Current Name | Timeout |
|----|---------------|---------|
| 1 | Low speed run | 31s |
| 2 | Standby0 | 0 |
| 3 | Standby1 | 33s |
| 4 | Sleep | 34s |

The data: 04 00 1F 00 00 00 21 00 22

7.28.4 Get Battery Level (F0-02)

This command is used to get the battery level.

Note: Only VP3600 supports this command.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | F0h | 02h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|---------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | F0h | See status code table | 00h | 01h | Level | | |

| Level | Content |
|---------|--------------------|
| >= 0xD2 | Battery power full |
| <= 0xC0 | Battery power low |

7.28.5 Set USB Sleep Configuration (F0-06)

This command is used to Enable/Disable USB Sleep Configuration

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|-------------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | F0h | 06h | 00h | 01h | 00h: OFF 01h: ON (default) | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | F0h | See status code table | 00h | 00h | | |

7.28.6 Get USB Sleep Configuration (F0-07)

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | F0h | 07h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | F0h | See status code table | 00h | 01h | USB Sleep Configuration | | |

Data:

00h: OFF

01h: ON (default)

7.28.7 Wake-Up Notification (F1-00)

The reader sends this data string if it wakes up from sleep/stop mode. This command is sent in RS232 mode only. The wake-up source must be the switch, NFC, or MSR.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | F1h | 00h | 00h | 00h | | |

7.29 USB Power Mode Suspend Configuration

7.29.1 Set USB Power Mode Suspend Configuration (F0-08)

This command enables or disables USB Power Mode Suspend configuration. Keep this feature turned off for RS232 readers.

Note: “USB Power Mode Suspend” is a special term used by the USB Implementers Forum (USBIF), which requires USB readers go to suspend (low-current) mode after power up, even without enumeration or configuration. The **Set USB Sleep Configuration (F0-06)** command should also be set to ON(default) in order to make this setting work properly.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|-------------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | F0h | 08h | 00h | 01h | 00h: OFF (default) 01h: ON | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | F0h | See status code table | 00h | 00h | | |

7.29.2 Get USB Power Mode Suspend Configuration (F0-09)

This command is used to read out USB Power Mode Suspend Configuration.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | F0h | 09h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------------|---------|---|-------------------------|-------------------------|---|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | F0h | See status code table | 00h | 01h | USB Power Mode Suspend Configuration | | |

Data:

- 00h: OFF (default)
- 01h: ON

7.30 Miscellaneous NEO 2 Platform Commands

7.30.1 Front Removal Configuration (04-22)

The **Front Removal Configuration** command enables front removal detection, retrieves the front removal status, or retrieves the front removal activation status.

NOTE: Only TriMag IV (ID TECHASIC) physical tamper features are used for this feature and command.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|--|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 04h | 22h | 00h | 01h | 01h: Enable front removal 02h: Get front removal status 03h: Get front removal activation status | Varies | Varies |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 04h | See status code table | 00h | 00h or 01h | See Response Data OR None | | |

Response Data (If Status = OK)

| Data Field | Length (bytes) | Description |
|---------------|----------------|---|
| Tamper Status | 1 | 00h: not enabled 01h: tampered 02h: enabled |

7.30.2 Retrieve Key Info (81-0C)

This command is used to retrieve basic key information. Each pair of three bytes represents one key's parameters (index and slot).

For example, 0x02 0x00 0x00 0x02 0x00 0x01 will represent

[KeyIndex=0x02, KeySlot=0x0000] and [KeyIndex=0x02, KeySlot=0x0001]

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 81h | 0Ch | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 81h | See status code table | Var | Var | See Data Table | | |

| Data Field | Length (bytes) | Description |
|------------|----------------|---------------------------------|
| Key Index | 1 | Index of the unique key |
| Key Slot | 2 | Specific Slot of the unique key |

7.30.3 Retrieve Detailed Key Info (81-0D)

This command will retrieve the Key Block Header of the specific key index/slot provided in the **Command Frame**. Key Block Header contains information such as type of encryption (TDES/AES), Key Variant, and so on.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14, 15 | Byte 16 | Byte 17 |
|-------------------------------|---------|-------------|-------------------|-------------------|------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 81h | 0Dh | 03h | | See Tables Below | | |

Command Data Field

| Data Field | Length (bytes) | Description |
|----------------|----------------|---|
| Key Name Index | 1 | 0x14 – LCL-KEK 0x01 – Pin encryption Key 0x02 – Data encryption Key 0x05 – MAC DUKPT Key 0x0A – PCI Pairing Key |

| Data Field | Length (bytes) | Description |
|------------|----------------|---|
| Key Slot | 2 | Indicate different slots of a certain Key Name Example: slot=5 (0x00 0x05), slot=300 (0x01 0x2C) For BTPay380, slot is always 0 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|----------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 81h | See status code table | 00h | 10h | Key Block Header (KBH) | | |

7.30.4 Get Certificate Info (81-0E)**Command Frame**

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|----------------------------------|---------|-------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 81h | 0Eh | 00h | 00h | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|----------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 81h | See status code table | 00 | Var | CertInfo | | |

CertInfo:

Byte 1 N Length of SN

Byte 2 ~ Byte N+1, SN in ASCII

Byte N+2: M Total downloaded Certificate number

Byte N+3 ~ byte 2+N+M CertKeyID-1~ CertKeyID-M

ID Tech certificate KeyID definition:

| | |
|---------------------------|------|
| #define ROOT_CA | 0x00 |
| #define DEVICE_CA | 0x0A |
| #define SECURE_CONTENT_CA | 0x0B |
| #define TMS_CA | 0x0C |
| #define DATA_KEY_KDH_CA | 0x0D |
| #define PIN_KEY_KDH_CA | 0x0E |

| | |
|----------------------------------|------|
| #define DEVICE_CERTIFICATE | 0x14 |
| #define FW_AUTH_CERTIFICATE | 0x15 |
| #define TMS_SERVER_CERTIFICATE | 0x18 |
| #define PIN_KDH_AUTH_CERTIFICATE | 0x1A |
| #define PIN_KDH_ENC_CERTIFICATE | 0x1D |
| #define MFG_COMMANDS_CERT | 0x16 |
| #define SECURE_MESSAGE_CERT | 0x17 |
| #define APPLICATION_AUTH_CERT | 0x1B |
| #define DATA_SKDH_AUTH_CERT | 0x19 |
| #define DATA_SKDH_ENC_CERT | 0x1C |
| #define DATA_NSKDH_AUTH_CERT | 0x1E |
| #define DATA_NSKDH_ENC_CERT | 0x1F |

Example:

IDGcommand:56 69 56 4F 74 65 63 68 32 00 81 0E 00 00 D4 6A.

Command return data:56 69 56 4F 74 65 63 68 32 00 81 00 00 1C 0A 36
33 30 5A 30 30 30 30 30 31 10 00 0A 0B 0C 0D 0E 14 15 16 17 18 19
1A 1B 1C 1D E7 4E

0A: Length of SN in Hex

36 33 30 5A 30 30 30 30 31: SN (630Z000006 in ASCII)

10: number of certificates in Hex

00 0A 0B 0C 0D 0E 14 15 16 17 18 19 1A 1B 1C 1D: certificate KeyID(Hex) list.

7.30.5 Get Certificate (81-0F)

This command is used to retrieve individual certificate using ID Tech's certificate KeyID definition.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|-------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViV0tech2\0 | 81h | 0Fh | 00h | 01h | Cert Key ID | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | 81h | See status code table | Var | Var | Certificate | | |

7.30.6 Get Device Status (32-03)

Use this command to return reader status. This command is only valid for VP6800 and VP3320.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-----------------------|---------|-------------|------------|------------|-----------|-----------|
| Header Tag & Protocol | Command | Sub Command | Length MSB | Length LSB | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 32h | 03h | 00h | 00h | Varies | Varies |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOPayV3\0 | '83' | '41' | variable | variable | See below | | |

Data:

| Data Item | Length (bytes) | Description |
|-------------------|----------------|------------------------------------|
| Activation Status | 1 | 0 – Deactivation 1 - Activation |
| Lock Status | 1 | 0 – Not Locked 1 - Activation |

7.30.7 Switches Notifications (F2-00)

Reader will send out this data string when there is a change in state for the switches. The configuration must be set in 01-07 command.

Note: This command is only available for Hybrid Insert Readers.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|--------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | F0h | See status code table | 00h | 01h | Data Objects | | |

Data Objects (1 byte)

| Bit Position | Meaning if '0' | Meaning if '1' |
|--------------|---------------------------|-----------------------|
| 0 | RFU | RFU |
| 1 | Card not seated | Card seated |
| 2 | Front switch not detected | Front switch detected |
| 3~7 | RFU | RFU |

7.30.8 Set Switches Notifications Configuration (01-07)

This command is used to set the switches notifications configuration.

Note: This command is only available for Hybrid Insert Readers.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 01h | 07h | 00h | 01h | Data Objects | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 01h | See status code table | 00h | 00h | | |

Data Objects (1 byte)

| Bit Position | Meaning if '1' |
|--------------|-----------------------------------|
| 0 | Card seated notification enabled |
| 1 | Front switch notification enabled |
| 2~7 | RFU |

7.30.9 Get Switches Notifications Configuration (01-08)

This command returns the configuration for switches notifications.

Note: This command is only available for Hybrid Insert Readers.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 01h | 08h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|--------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 01h | See status code table | 00h | 01h | Data Objects | | |

Data Objects (1 byte)

| Bit Position | Meaning if '1' |
|--------------|-----------------------------------|
| 0 | Card seated notification enabled |
| 1 | Front switch notification enabled |
| 2~7 | RFU |

7.30.10 Set Key Press Event (01-0E)

The **Set Key Press Event** command enables or disables the **Key Press Event (61-04)** command.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|-------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 01h | 0Eh | 00h | 01h | Data Object | | |

Data Object

| Value | Description |
|-------|-------------|
| 0x00 | Off |
| 0x01 | On |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 01h | See status code table | 00h | 00h | | |

7.30.11 Get Key Press Event (01-0F)

The **Get Key Press Event** command returns the configuration set with the **Key Press Event (61-04)** command.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 01h | 0Fh | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 01h | See status code table | 00h | 01h | Data Object | | |

Data Object

| Value | Description |
|-------|-------------|
| 0x00 | Off |
| 0x01 | On |

7.30.12 Key Press Event (61-04) (Reader sends to Host)

The **Key Press Event** command, sent by the reader to the host device, indicates to the host which key the user pressed; the host does not need to send a response.

For PIN entry, the the reader sends the '*' character instead of the key value. Turn this feature on or off with the **Set Key Press Event (01-0E)** command.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|-----------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 04h | 00h | 01h | Key Value | | |

| Data Item | Length (bytes) | Description |
|-----------|----------------|--|
| Key Value | 1 | '0~9' – Numeric digit '*' – Masked PIN digit 'E' – Enter 'B' – Back 'C' – Cancel |

7.31 Special Function or Feature Configuration Commands

7.31.1 Set Special Function or Feature Configuration (C7-82)

The Set Special Function or Feature Configuration command sets a specific special configuration.

Note: If the reader supports ITP Protocol, the FunID (ITP Protocol) is B0h and B1h.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|---------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | C7h | 82h | 00h | var | See send data field | | |

Data Table

| Data Item | Length | Description |
|-------------------------------|-----------|--|
| Function/Feature ID Length | 2 | Length of Function/Feature. Format: <Length H><Length L> (Big-endian) Current value is 00h 01h |
| Function/Feature ID | n | ID from 0# See Function/Feature ID Table |
| Additional Requirement Length | 2 | Length of Additional Requirement. Format: <Length H><Length L> (Big-endian) See Function/Feature ID Table |
| Additional Requirement | None or n | See Function/Feature ID Table |

Note:

| Customer ID | Requirement |
|-------------|--|
| 0# | ID TECH standard: None. Requirement Length is 00h. |
| 1# | Requirement Length is 00h 01h <Timeout (1 byte)>: range is 1~20 (0.5s ~ 10s). |
| RFU | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | C7h | See status code table | 00h | 00h | | |

7.31.2 Get Special Function or Feature Configuration (C7-83)

The **Get Special Function or Feature Configuration** command retrieves the reader's currently set special configuration.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | C7h | 83h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | C7h | See status code table | 00h | var | See Data table | | |

Data Table

| Data Item | Length | Description |
|-------------------------------|-----------|--|
| Function/Feature ID Length | 2 | Length of Function/Feature ID. Format: <Length H><Length L> (Big-endian) Current value is 00h 01h |
| Function/Feature ID | n | ID from 0# See Function/Feature ID Table |
| Additional Requirement Length | 2 | Length of Additional Requirement. Format: <Length H><Length L> (Big-endian) See Function/Feature ID Table |
| Additional Requirement | None or n | See Function/Feature ID Table |

7.31.3 Function/Feature ID Table

Function/Feature ID: 0# (ID-TECH) ~ 2#.

| ID | Requirement |
|----|--|
| 0# | Function/Feature 0# ID TECH standard: None. Additional Requirement Length is 00h 00h. |
| 1# | Function/Feature 1# Additional Requirement Length is 00h 01h <Timeout (1 byte)>: range is 1~20 (0.5s ~ 10s). Function / Feature indicates: 1. PAN display format on LCD screen is special: Each 4 digits has a separator; if the related PAN data should be masked, it will show clear text for about 1 second (Additional Requirement Setting) and then change to a mask character. |

| | |
|-----|--|
| | 2. Support PAN data that length is 11 digits, and will use special algorithm to check the LRC digit if PAN length is 11 digits. |
| 2# | Function/Feature 2#: None. Additional Requirement Length is 00h 00h. Function/Feature indicates: Whitelist setting does not matter, output encrypted R-APDU. |
| RFU | |

8. Protocol Command Reference: Protocol 3 and Protocol 5

Protocol 3 is an advanced version of Protocol 2. Protocol 3 changes the Header Tag & Protocol Version to ViVOpayV3\0 and includes a sub-command ID byte in the response frame format so the host device can more precisely identify multiple responses with same command ID. Currently, Protocol 3 is not supported in NEO 2.0 (but may be supported in the future).

Protocol 5 has the same design as Protocol 3 and is mostly used in the internal interaction between the K81 and Android chips in NEO 2.0 Android series readers.

8.1 Protocol 5 Command and Response Frames

Request and response interactions with the reader involve two types of frames: command frames and response frames. The general format of these frames is given below.

Command Frame Format

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | n Bytes | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|---------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOpayV5\0 | | | | | | | |

Response Frame Format

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | n Bytes | Byte 15+n | Byte 16+n |
|-------------------------------|---------|-------------|-------------|-------------------|-------------------|---------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOpayV5\0 | | | | | | | | |

8.2 Supported Commands

The following commands are supported for Protocol 5.

8.2.1 Contact LCD Display Control (61-01) (K81 sends to Android Chip)

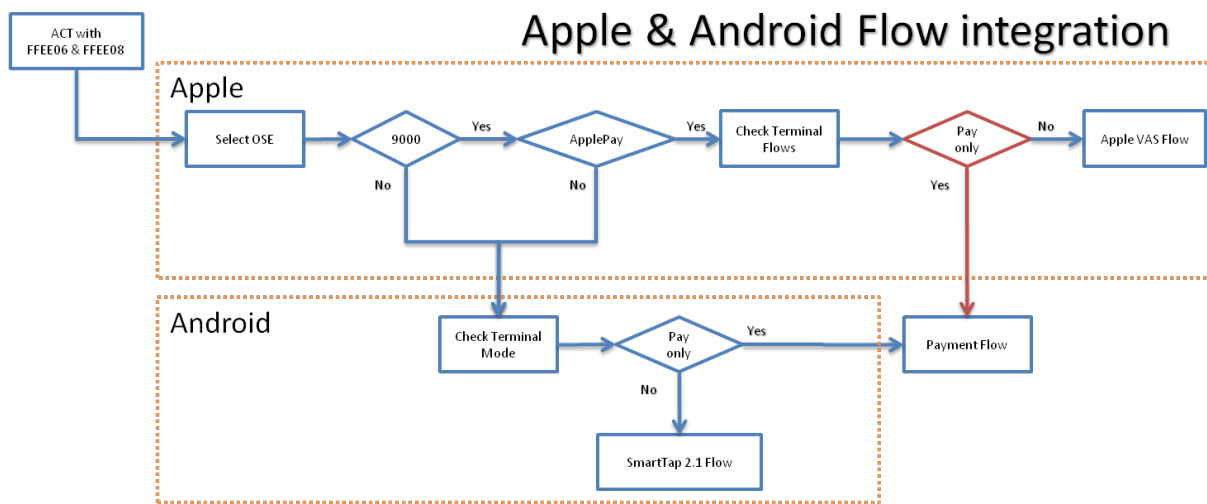
The [Contact LCD Display Control \(61-01\) command](#) functions as it does in Protocol 2 but with the Protocol 5 command and response frames provided above.

9. Loyalty Functions

NEO 2 firmware has built-in support for Apple Pay VAS (Value Added Services) and Google Smart Tap.

Support for Google Pay Smart Tap functionality is described in a separate document, *Google Pay Smart Tap 2.1 in ViVOPay Devices.pdf*, available for download on the [ID TECH Knowledge Base](#). Consult that document for Smart Tap-related functionality. It supersedes this IDG where Smart Tap is concerned.

9.1 Apple VAS & Google Pay Smart Tap 2.1 with ACT command



For non-payment applications, check the status word and Application Label from response of Select OSE. Via Select OSE, one can get 9000 and "Apple Pay" from iPhone or "Google Pay" from Android phone, and then continue to each transaction flow.

How to use ACT for Apple VAS and Google Pay Smart Tap 2.1:

Set all configurations for Apple VAS

Set the Merchant List

```

56 69 56 4F 74 65 63 68 32 00 04 11 00 63 04 01 06 41 3B 95 7A 52
59 98 3B 60 8C FC 89 CF B1 DA B9 0C E7 05 AD 8E FF 78 E9 DE 12 2C
CF 8D 2C BF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 E7 6F
  
```

Set all configurations for Google Pay Smart Tap 2.1**Set Long term private key**

```

56 69 56 4F 74 65 63 68 32 00 C7 62 01 20 A7 82 BE 58 B2 FB E0 51
D0 CE 53 48 88 D0 4A AB C5 22 37 D3 E2 A7 3A 41 C0 BB C1 BD 71 1B
B9 F6 AE 95 A6 14 45 8A 9D 11 88 00 5C B8 20 0F 56 8F ED C8 3B 1D
A9 1F BD 0F F4 92 4E 72 D7 1C A7 91 29 44 84 79 FE BA CE 55 DF 9A
2B AC 6A 40 BC 1A 1A 90 F2 18 1F 0D 5A 1E 04 0D CB 5D 90 33 3E 1A
6A 4A 36 6B FB D0 35 DC FB 95 52 F8 74 AB 47 FC 0F 99 64 96 1C 76
27 D1 AC 5F E7 7C 87 B2 02 A1 9A C2 3C 9C 8A 2C CA D6 3F 24 FF 56
52 1A 45 3B EB D7 C5 4A F5 7B D2 89 A3 E0 3B 19 CE 2A 9A E3 B9 8B
49 30 2F 97 95 FF B1 55 74 63 D7 6A A7 14 70 AD B0 A8 E8 4E 6A 74
6A 6C 29 22 B7 C6 C2 77 1E EC B5 D7 C4 E3 E2 6C 05 F1 43 B3 0A 29
62 F1 E5 49 26 63 E4 87 39 38 02 1A FE EE B7 48 70 B0 A4 FF 1A 93
10 5D CD 8E 74 5D 9B B7 34 C8 A4 7A 1C B2 79 48 D1 19 5F F5 97 04
06 3F 7D F4 BE 4D A0 27 1B D1 96 D3 C0 C4 58 CB 5F D6 D9 ED AB 12
8E B3 CD 24 BC 27 F2 DB E6 44 7C 75 5F 63 09 43 BE 36

```

Set Merchant ID, Store Location, Terminal ID, Merchant Name, Merchant Category, POS Capabilities Bitmap, Retry times, Select OSE support, Skip Second Select support, Stop Payment if Google Pay Smart Tap 2.1 failed support and Pre-Signed support.

```

56 69 56 4F 74 65 63 68 32 00 04 00 00 3A DF EE 3B 04 00 BC 61 4E
DF EE 3C 00 DF EE 3D 00 DF EE 25 00 DF ED 01 00 DF ED 02 05 00 00
00 00 01 DF ED 03 01 00 DF ED 04 01 01 DF ED 05 01 01 DF ED 06 01
00 DF ED 07 01 00 A9 AE

```

Send ACT command, including Apple VAS container (FFEE06) and Smart Tap 2.1 container (FFEE08):

```

56 69 56 4F 74 65 63 68 32 00 02 01 00 2F 0A 9F 02 06 00 00 00 00
00 01 9C 01 00 FF EE 06 18 9F 22 02 01 00 9F 26 04 00 00 00 00 9F
2B 05 01 00 00 00 00 00 DF 01 01 01 FF EE 08 02 02 00 57 47

```

Tapping Apple VAS Panera Bread Card

```

56 69 56 4F 74 65 63 68 32 00 02 57 00 8C FF EE 06 82 00 75 9A 03
14 08 10 9F 21 03 12 08 01 9F 25 20 06 41 3B 95 7A 52 59 98 3B 60
8C FC 89 CF B1 DA B9 0C E7 05 AD 8E FF 78 E9 DE 12 2C CF 8D 2C BF
9F 2A 00 9F 27 41 44 8D EC 4C 6B 2E 5B 06 FC 24 F1 5A 5D 5B B1 D4
41 54 CA D0 EF 34 AB 19 D5 BB 98 93 A7 B7 5F 23 CC 3A A2 F0 FC B4
AB 1B 71 21 D2 4A 3A 18 5C EE 6B 6B F7 23 7B D7 CE B1 63 F6 C7 F1
4E 69 85 85 C5 9F 39 01 07 FF EE 01 04 DF 30 01 00 DF EE 26 01 01
D2 E3

```

Tapping Google Pay Panera Bread Card

```

56 69 56 4F 74 65 63 68 32 00 02 57 00 69 00 00 00 FF EE 08 51 90
01 D4 03 49 61 73 76 94 01 06 69 04 02 71 79 79 71 54 03 39 63 75
73 94 03 11 63 69 64 04 E1 50 A7 94 A5 BD 36 E7 E8 94 42 05 52 9E
50 A6 19 01 03 03 54 63 70 6C 00 65 6E 54 03 11 63 75 74 04 95 3E

```



```
95 20 FC AD 8F 79 9C 65 E5 94 FE E4 2B 3E 9F 39 01 07 FF EE 01 04
DF 30 01 00 DF EE 26 01 01 42 65
```

Send ACT command include Apple VAS container (FFEE06) only:

```
56 69 56 4F 74 65 63 68 32 00 02 01 00 29 30 9F 02 06 00 00 00 00
00 01 9C 01 00 FF EE 06 18 9F 22 02 01 00 9F 26 04 00 00 00 02 9F
2B 05 01 00 00 00 00 DF 01 01 01 CF 40
```

Tapping Apple VAS Panera Bread Card

```
56 69 56 4F 74 65 63 68 32 00 02 57 00 8C FF EE 06 82 00 75 9A 03
14 08 10 9F 21 03 12 10 31 9F 25 20 06 41 3B 95 7A 52 59 98 3B 60
8C FC 89 CF B1 DA B9 0C E7 05 AD 8E FF 78 E9 DE 12 2C CF 8D 2C BF
9F 2A 00 9F 27 41 44 8D EC 4C 66 54 D0 F8 01 60 7C DC 06 BF 93 88
21 23 B4 F6 C9 6D 1B CC 63 4D C3 4D D9 BD 71 5F B1 C9 DC 60 2C AF
77 75 62 E5 16 E7 7F 8F 83 95 18 4F 1E EC 7D C8 8F DC 38 01 CE 99
03 98 0E AD 3E 9F 39 01 07 FF EE 01 04 DF 30 01 00 DF EE 26 01 01
5D 5C
```

Tapping Google Pay Panera Bread Card

```
56 69 56 4F 74 65 63 68 32 00 02 0A 00 29 FF EE 06 82 00 12 9A 03
14 08 10 9F 21 03 12 23 19 DF 02 04 24 90 00 02 9F 39 01 07 FF EE
01 04 DF 30 01 00 DF EE 26 01 01 65 58
```

Send ACT command include Smart Tap 2.1 container (FFEE08) only:

```
56 69 56 4F 74 65 63 68 32 00 02 01 00 15 30 9F 02 06 00 00 00 00
00 01 9C 01 00 FF EE 08 82 00 02 02 00 37 B6
```

Tapping Apple VAS Panera Bread Card

```
56 69 56 4F 74 65 63 68 32 00 02 0A 00 11 20 6A 82 01 FF EE 08 04
00 6A 82 00 DF EE 26 01 01 A0 69
```

Tapping Google Pay Panera Bread Card

```
56 69 56 4F 74 65 63 68 32 00 02 57 00 69 00 00 00 FF EE 08 51 90
01 D4 03 49 61 73 76 94 01 06 69 04 02 71 79 79 71 54 03 39 63 75
73 94 03 11 63 69 64 04 E1 50 A7 94 A5 BD 36 E7 E8 94 42 05 52 9E
50 A6 19 01 03 03 54 63 70 6C 00 65 6E 54 03 11 63 75 74 04 B4 A1
E4 92 85 F9 30 A7 AB A6 CF 36 F0 1E D7 B6 9F 39 01 07 FF EE 01 04
DF 30 01 00 DF EE 26 01 01 9B 63
```

9.2 ApplePay VAS Functionality

NEO 2 firmware has built-in support for Apple Value Added Services Protocol (hereinafter called Apple VAS). For detailed information on how Apple VAS works, consult [Apple's developer site](#). The following discussion assumes that you are familiar with the basic concepts behind ApplePay VAS.

9.2.1 Overview

In order to process an ApplePay VAS transaction, the host device's merchant records must first be set. This should be done using TLVs sent in the [SetMerchantRecord \(04-11\) command](#) (discussed further below).

A Merchant ID (of the form "Pass.com.apple.wallet.vas.prodtest") is generally associated with a hash value and (optionally) a URL. The hash is just a SHA-256 hash of the Merchant ID string itself. For example, the Merchant ID "pass.com.apple.wallet.vas.prodtest," in hex, is represented as 70 61 73 73 2e 63 6f 6d 2e 61 70 70 6c 65 2e 77 61 6c 6c 65 74 2e 76 61 73 2e 70 72 6f 64 74 65 73 74.

When a byte array containing those values is hashed, the resulting SHA-256 value is 3C C7 0E D8 9A 9D 43 54 BE 98 30 AB 58 D8 9C 6F E7 E6 2B AC A9 39 D2 A6 85 1D FC 60 2E A7 98 F7.

Limitations

Note that a NEO reader can store at most 6 Merchant Records for Apple VAS. This means the reader can natively support up to 6 loyalty programs (per transaction) at a time.

Note to Developers

If you are sending commands via the "Send NEO Command" in the Universal Demo, or using **device_sendDataCommand()** in the SDK, only the command, subcommand, and data fields need to be sent. The header, data length, and CRC will be dynamically calculated and inserted into the command string, automatically, by the SDK.

9.2.2 ACT Parameters for ApplePay VAS

The Activate Transaction (ACT) parameters required for the ApplePay VAS functionality are communicated via the ApplePay VAS Container TLV (tag FFEE06). The FFEE06 TLV must be provided in the ACT command (02-01 or 02-40) if an ApplePay VAS transaction is desired.

9.2.3 ECP 2.0 support

When using ECP 2.0 Format Version '01', include tag 9F22, tag DFED3D, and tag 9F26 in tag FFEE06.

- If TAG 9F22 is '0100', the reader will check for tag DFED3D. If DFED3D is missing, the reader uses tag 9F26 byte 2 to replace the terminal type. The reader will then implement ECP 2.0 Format Version '01'.
- If tag 9F22 and tag 9F26 are missing or produce an error, the reader returns an error.

When using ECP 2.0 Format Version '02', include tag 9F22 and tag DFED57 in tag FFEE06. If tag 9F22 is '0200', the reader checks for tag DFED57. If DFED57 is missing or produces an error, the reader returns an error; otherwise, the reader implements ECP 2.0 Format Version '02'.

| TLV | Name | Presence | Description |
|--------|---|----------|---|
| 9F26 | ApplePay Terminal Capabilities Information (see table below for more information) | Required | Determines how the reader handles the VAS and/or Payment flow. This data comprises 4 Bytes. Byte 1: RFU Byte 2: Host Type (ECP 2.0 Format Version '01' Support) (If tag DFED3D exists, this byte is not used.) Byte 3: RFU Byte 4: Host Mode See table further below. |
| 9F22 | ApplePay Terminal Application Version Number | Required | 0100: support ECP 2.0 Format Version '01' 0200: support ECP 2.0 Format Version '02' |
| 9F2B | ApplePay VAS Filter | Optional | If not provided filtering mobile device will not perform filtering. |
| DF01 | ApplePay VAS Protocol 87654321 -----0 URL VAS Protocol -----1 FULL VAS Protocol -----0- No VAS Beeps -----1- VAS Beeps -----0-- EMEA Comm Error -----1-- Silent Comm Error | Optional | If not provided the following settings are used by default: <ul style="list-style-type: none"> • Full VAS protocol • No beeps for VAS • EMEA Communications Error Handling If provided the bits define the settings. |
| DFED3D | Terminal Type | Optional | For ECP 2.0 Format Version '01' use. See table further below. |

| | | | |
|--------|--|----------|---|
| DFED57 | Apple VAS Terminal Type and Data | Required | For ECP 2.0 Format Version '02' use. See table further below. |
| DFED49 | ApplePay VAS Transaction Only 0: Transactions behave normally 1: VAS transactions only | Required | Sets transaction to VAS-only if enabled. |

9.2.4 Tag 9F26 ApplePay Terminal Capabilities Information

Byte 1: RFU

| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Description |
|----|----|----|----|----|----|----|----|-----------------------------------|
| x | x | x | x | x | x | x | x | RFU, Bits b8-b1 shall be set to 0 |

Byte 2: Terminal Type

| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Description |
|----|----|----|----|----|----|----|----|--|
| 0 | - | - | - | - | - | - | - | VAS Supported |
| 1 | - | - | - | - | - | - | - | VAS Not Supported |
| - | 0 | - | - | - | - | - | - | Touch ID Required |
| - | 1 | - | - | - | - | - | - | Touch ID not Required |
| - | - | - | - | 0 | 0 | 0 | 0 | Payment Host |
| - | - | - | - | 0 | 0 | 0 | 1 | Transit Host (Use Format Version '02') |
| - | - | - | - | 0 | 0 | 1 | 0 | Access Host (use Format Version '02') |
| - | - | - | - | 0 | 0 | 1 | 1 | Wireless handoff Host |
| - | - | - | - | 1 | 1 | 1 | 1 | Other Host |
| x | x | 0- | 0- | x | x | x | x | Bits b6 and b5 shall be set to 0 |
| - | - | - | - | - | - | - | - | All other values are RFU |

Byte 3: RFU

| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Description |
|----|----|----|----|----|----|----|----|-----------------------------------|
| x | x | x | x | x | x | x | x | RFU, Bits b8-b1 shall be set to 0 |

Byte 4: Terminal Mode

| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Description |
|----|----|----|----|----|----|----|----|------------------------------------|
| | | | | | | 0 | 0 | Host in VAS App OR Payment Mode |
| | | | | | | 0 | 1 | Host in VAS App AND Payment Mode |
| | | | | | | 1 | 0 | Host in VAS App Only Mode |
| 0 | | | | | | | | Last Get VAS DATA command |
| 1 | | | | | | | | More 'GET VAS DATA' commands forth |
| x | 0 | 0 | 0 | 0 | 0 | x | x | Bits b7-b3 shall be set to 0 |
| | | | | | | | | All other values are RFU |

9.2.5 Tag DFED3D Terminal Type for ECP 2.0 Format Version '01'

Terminal Type

| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Description |
|----|----|----|----|----|----|----|----|--|
| 0 | - | - | - | - | - | - | - | VAS Supported |
| 1 | - | - | - | - | - | - | - | VAS Not Supported |
| - | 0 | - | - | - | - | - | - | Touch ID Required |
| - | 1 | - | - | - | - | - | - | Touch ID not Required |
| - | - | - | - | 0 | 0 | 0 | 0 | Payment Host |
| - | - | - | - | 0 | 0 | 0 | 1 | Transit Host (use Format Version '02') |
| - | - | - | - | 0 | 0 | 1 | 0 | Access Host (use Format Version '02') |
| - | - | - | - | 0 | 0 | 1 | 1 | Wireless handoff Host |
| - | - | - | - | 1 | 1 | 1 | 1 | Other Host |
| x | x | 0- | 0- | x | x | x | x | Bits b6 and b5 shall be set to 0 |
| - | - | - | - | - | - | - | - | All other values are RFU |

9.2.6 Tag DFED57 Apple VAS Terminal Type and Data for ECP 2.0 Format Version '02'

Tag DFED57 Definition:

- **Tag:** DFED57
- **Length:** Var
- **Data:** Terminal- Info, Terminal-Type, Terminal-Type-Data
 - **Where:**
 - Terminal Info: 1 byte
 - Terminal Type: 2 bytes
 - Terminal Type Data: 0- 15 bytes

Terminal- Info: 1 byte

| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Description |
|----|----|----|----|----|----|----|----|---|
| 0 | - | 0 | 0 | - | - | - | - | VAS Supported |
| 1 | - | 0 | 0 | - | - | - | - | VAS Not Supported |
| - | 0 | 0 | 0 | - | - | - | - | User Auth Requested |
| - | 1 | 0 | 0 | - | - | - | - | User Auth Not Requested |
| - | - | 0 | 0 | L | L | L | L | Length of Terminal Type Data field (0-15) |
| - | - | 0 | 0 | - | - | - | - | Bits b6 and b5 shall be set to 0 |

Terminal- Type: 2 bytes**Byte: 1 (follow Tag 9F26 byte 4)**

| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Description |
|----|----|----|----|----|----|----|----|---|
| - | - | - | - | - | - | 0 | 0 | Host in VAS App OR Payment Mode |
| - | - | - | - | - | - | 0 | 1 | Host in VAS App AND Payment Mode |
| - | - | - | - | - | - | 1 | 0 | Host in VAS App Only Mode |
| 0 | - | - | - | - | - | - | - | Last Get VAS Data Command |
| 1 | - | - | - | - | - | - | - | More 'Get VAS DATA' command forthcoming |
| x | 0 | 0 | 0 | 0 | 0 | x | X | Bits b7-b3 shall be set to 0 |
| | | | | | | | | All other values are RFU |

Byte: 2 (Terminal Sub-Type)

| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Description |
|----|----|----|----|----|----|----|----|------------------|
| - | - | - | - | - | - | - | - | Provide by Apple |

Terminal-Type-Data:0-15 bytes

| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Description |
|----|----|----|----|----|----|----|----|---------------------------------|
| - | - | - | - | - | - | - | - | Terminal-Type-Data (0~15 bytes) |

EXAMPLES (Activate Transaction Command Examples for ApplePay VAS)**Command sent using the SDK (VAS-only transaction):**

```
idtDevice.ctls_startTransaction(10,0,0,60,Common.getByteArray("FFE
E06189F220201009F2604000000039F2B050100000000DF010101"));
```

Using firmware commands:

Note: Some firmware examples use command 02-01, but 02-40 (encryption-enabled Activate Transaction) may also be used.

VAS Only Activate Transaction (02-40)

```
56 69 56 4F 74 65 63 68 32 00 02 40 00 2D 30 9F 02 06 00 00 00 00
00 01 9C 01 00 FF EE 06 18 9F 22 02 01 00 9F 26 04 00 00 00 02 9F
2B 05 01 00 00 00 00 DF 01 01 01 DF ED 49 01 01 33 FE
```

Command Sent Breakdown:

- 56 69 56 4F 74 65 63 68 32 00 – ViVOTech2 header
- 02 40 – Start transaction command
- 00 29 – Data Length
- 30 – Time out
- 9F 02 06 00 00 00 00 01 – Transaction amount
- 9C 01 00 – Transaction Type
- FF EE 06 – ApplePay VAS Collective
- 18 - length of ApplePay VAS collective
- 9F 22 02 01 00 - ApplePay Terminal AVN
- 9F 26 04 00 00 00 02 - ApplePay host Capabilities - 02 = VAS only
- 9F 2B 05 01 00 00 00 00 - AppplePay VAS Filter (optional)
- DF 01 01 01 –
- DF ED 49 01 01 - Required for VAS-only transactions
- 33 FE – CRC-16

Response:

```
56 69 56 4F 74 65 63 68 32 00 02 57 00 FF D1 FF EE 12 0A 62 99 49
01 2C 00 04 60 01 AC FF EE 06 82 00 D9 9A 03 14 08 15 9F 21 03 17
32 53 9F 25 20 3C C7 0E D8 9A 9D 43 54 BE 98 30 AB 58 D8 9C 6F E7
E6 2B AC A9 39 D2 A6 85 1D FC 60 2E A7 98 F7 9F 2A 00 9F 27 3E 93
1C 1A 60 4A 46 09 9E 21 EC 88 6D EF CC 8C B8 8B CA 03 CC 4B C6 62
0C C1 8F 8C 10 5A 7A F1 4F 9B 3C D9 E3 36 4E 9C 8C BF E0 90 34 10
B1 58 3C 3D 63 AC 9F CC 48 9C A8 76 AE 8C B3 E5 62 9F 25 20 3C C7
0E D8 9A 9D 43 54 BE 98 30 AB 58 D8 9C 6F E7 E6 2B AC A9 39 D2 A6
85 1D FC 60 2E A7 98 F7 9F 2A 00 9F 27 3E 93 1C 1A 60 4A 46 09 9E
21 EC 88 6D EF CC 8C B8 8B CA 03 CC 4B C6 62 0C C1 8F 8C 10 5A 7A
F1 4F 9B 3C D9 E3 36 4E 9C 8C BF E0 90 34 10 B1 58 3C 3D 63 AC 9F
CC 48 9C A8 76 AE 8C B3 E5 62 9F 39 01 07 FF EE 01 04 DF 30 01 00
DF EE 26 01 D1 ED CA
```

Response Breakdown:

- 56 69 56 4F 74 65 63 68 32 00 – ViVOTech2header
- 02 – Act Command returned
- 57 – Status. 57 = no payment occurred
- 00 FF – Data length
- D1 – Attribution byte
- FF EE 12 0A 62 99 49 01 2C 00 04 60 01 12 – KSN tag (FFEE12), length 0x0A, KSN bytes. Transaction is encrypted.
- FF EE 06 – Apple Pay Vas container
- 00 82 – Data length
- D9 9A 03 14 08 15 9F 21 03 17 32 53 9F 25 20 3C C7 0E D8 9A 9D 43 54 BE 98 30 AB 58 D8 9C 6F E7 E6 2B AC A9 39 D2 A6 85 1D FC 60 2E A7 98 F7 9F 2A 00 9F 27 3E 93 1C 1A 60 4A 46 09 9E 21 EC 88 6D EF CC 8C B8 8B CA 03 CC 4B C6 62 0C C1 8F 8C 10 5A 7A F1 4F 9B 3C D9 E3 36 4E 9C 8C BF E0 90 34 10 B1 58 3C 3D 63 AC 9F CC 48 9C A8 76 AE 8C B3 E5 62 9F 25 20 3C C7 0E D8 9A 9D 43 54 BE 98 30 AB 58 D8 9C 6F E7 E6 2B AC A9 39 D2 A6 85 1D FC 60 2E A7 98 F7 9F 2A 00 9F 27 3E 93 1C 1A 60 4A 46 09 9E 21 EC 88 6D EF CC 8C B8 8B CA 03 CC 4B C6 62 0C C1 8F 8C 10 5A 7A F1 4F 9B 3C D9 E3 36 4E 9C 8C BF E0 90 34 10 B1 58 3C 3D 63 AC 9F CC 48 9C A8 76 AE 8C B3 E5 62 9F 39 01 07 FF EE 01 04 DF 30 01 00 DF EE 26 01 D1 –
- ED CA – CRC

VAS Or Pay Activate Transaction (02-01)

```
56 69 56 4F 74 65 63 68 32 00 02 01 00 29 30 9F 02 06 00 00 00 00
00 01 9C 01 00 FF EE 06 18 9F 22 02 01 00 9F 26 04 00 00 00 00 9F
2B 05 01 00 00 00 00 DF 01 01 01 09 CA
```

VAS And Pay Activate Transaction (02-01)

```
56 69 56 4F 74 65 63 68 32 00 02 01 00 29 30 9F 02 06 00 00 00 00
00 01 9C 01 00 FF EE 06 18 9F 22 02 01 00 9F 26 04 00 00 00 01 9F
2B 05 01 00 00 00 00 DF 01 01 01 6A 8F
```

VAS Only Activate Transaction (02-01)

```
56 69 56 4F 74 65 63 68 32 00 02 01 00 29 30 9F 02 06 00 00 00 00
00 01 9C 01 00 FF EE 06 18 9F 22 02 01 00 9F 26 04 00 00 00 02 9F
2B 05 01 00 00 00 00 DF 01 01 01 CF 40
```

Pay Only Activate Transaction (02-01)

```
56 69 56 4F 74 65 63 68 32 00 02 01 00 29 30 9F 02 06 00 00 00 00
00 01 9C 01 00 FF EE 06 18 9F 22 02 01 00 9F 26 04 00 00 00 03 9F
2B 05 01 00 00 00 00 DF 01 01 01 AC 05
```


Transaction Responses

Both the ApplePay VAS response and the normal payment transaction response will be provided in a single returned data record. Whether returned in response to a blocking ACT or a non-blocking ACT, it will be the same. As described above, there are ApplePay VAS scenarios where either the VAS transaction or the payment transaction may not be performed. In those scenarios you will only see the results of the transaction that was actually performed. Only when both VAS and payment transactions are performed will you see both transaction responses in the same returned data record.

The Payment transaction response will not change. The VAS transaction response will be embedded in the proprietary ApplePay VAS Container TLV (0xFFEE06). Each Merchant ID and its associated data will be shown in sequence.

Transaction Response for Combined Payment and VAS

- 56 69 56 4F 74 65 63 68 32 00 – Serial Command header
- 02 – Command
- 23 – Status – for the payment transaction only. In this example it indicates a Request for Online Authorization
- nn nn – length of entire response (VAS and Pay)
- xx – Start of payment response. Payment response format has not changed.
- xx – End of payment response.
- FFEE06 nn - ApplePay VAS Container
 - 9A nn – Date
 - 9F21 nn - Time
 - 9F25 nn – Merchant ID **a**
 - 9F2A nn – Mobile Token
 - 9F27 nn – VAS Data
 - 9F25 nn – Merchant ID **b**
 - 9F2A nn – Mobile Token
 - 9F27 nn – VAS Data
 -
 - 9F25 nn – Merchant ID **n**
 - 9F2A nn – Mobile Token
 - 9F27 nn – VAS Data
- xx xx – CRC for entire response

Transaction Response for VAS Only (No Payment)

- 56 69 56 4F 74 65 63 68 32 00 – Serial Command header
- 02 – Command
- 57 – Status for the payment transaction. 0x57 indicates there was no payment transaction.
- nn nn – length of entire response (VAS)

- FFEE06 nn - ApplePay VAS Container ViVOpay
 - 9A nn – Date
 - 9F21 nn - Time
 - 9F25 nn – Merchant ID **a**
 - 9F2A nn – Mobile Token
 - 9F27 nn – VAS Data
 - 9F25 nn – Merchant ID **b**
 - 9F2A nn – Mobile Token
 - 9F27 nn – VAS Data
 -
 - 9F25 nn – Merchant ID **n**
 - 9F2A nn – Mobile Token
 - 9F27 nn – VAS Data
- xx xx – CRC for entire response

9.2.7 Set Merchant Record (04-11)

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data length (MSB) | Data length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 04 | 11h | 63h | | See Data Format below | | |

Data Field for Command Frame

| Data Field | Length (bytes) | Description |
|-------------------------|----------------|--|
| Merchant Record Index | 1 | The valid value is 1--6. |
| ID Present | 1 | 1: The Merchant ID is valid, 0: The Merchant ID is not valid. |
| Merchant ID | 32 | The tag is 9F25. |
| Length of Merchant URL | 1 | Can be zero, if no URL is used. |
| Merchant URL (optional) | 64 | The tag is 9F29. |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------------|---------|---------------------------------------|----------------------|----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 04h | See status code table | 00 | 00 | | |

9.2.8 Get Merchant Record (03-11)

Note: NEO 2 readers do not have a default merchant record. Use the **Set Merchant Record (04-11)** command to set one before using **Get Merchant Record (03-11)**.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte16 |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data length (MSB) | Data length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViV0tech2\0 | 03 | 11h | 01 | | Merchant Record Index(1-6) | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | 03 | See Status Code Table | 63h | | See Data Format below | | |

Data Field for Response Frame

| Data Field | Length (bytes) | Description |
|------------------------|----------------|--|
| Merchant Record Index | 1 | The valid value is 1--6. |
| ID Present | 1 | 1: The Merchant ID is valid, 0: The Merchant ID is not valid. |
| Merchant ID | 32 | The tag is 9F25. |
| Length of Merchant URL | 1 | |
| Merchant URL | 64 | The tag is 9F29. |

9.3 Google Pay Smart Tap 2.1

ViVOpay readers running NEO 2 firmware have a built-in Smart Tap kernel that provides support for Google Pay Smart Tap. This functionality is described in a separate document. Refer to the ID TECH *Smart Tap Integration Guide* (publication date TBD).

10. Peer to Peer Functionality

Peer To Peer functionality (which can only be used in Pass-Through mode) allows the sending and receiving of NDEF messages to/from a mobile device.

See [NDEF \(NFC Data Exchange Format\)](#) for more information about NDEF payload composition.

10.1 Peer to Peer Send A Message (C7-9A)

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|----------------------------|-----------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data length (MSB) | Data length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViV0tech2\0 | C7h | 9Ah | Variable | | See Data Format below | | |

Peer To Peer Send A Message Data Field for Command Frame

| Data Field | Length (bytes) | Description |
|------------|----------------|--------------------------------------|
| Timeout | 1 | Time in Seconds. |
| Message | Variable | NDEF message to be sent to the phone |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------------|---------|---------------------------------------|----------------------|----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | C7h | See status code table | 00 | 00 | | |

10.2 Peer to Peer Receive A Message (C7-9B)

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte16 |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|---|-----------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data length (MSB) | Data length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViV0tech2\0 | C7 | 9Bh | 00h | 01h | Timeout (1 byte, time in seconds) | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | C7h | See status code table | Variable | | See Data description below | | |

If Status code is OK, Data Field for response frame is a message received from the phone.

Otherwise, Data Length is zero and no data for **Response Frame**.

Messages conform to NDEF (<https://learn.adafruit.com/adafruit-pn532-rfid-nfc/ndef>).

11. MSR Equivalent Data Function

Proprietary tags DFEF4B, DFEF4C, and DFEF4D provide a way for track data (and optionally, PAN data) to be supplied in conjunction with an EMV transaction, with or without sentinels, in a form similar to the form track data would take in a conventional MSR transaction.

DFEF4C and DFEF4D TLV will only appear in the response of 02-40/03-40, 60-10, 60-11, and 60-12 commands for successful transaction. Use DFEF4B (in Terminal Settings) to configure how DFEF4C and DFEF4D will behave (see next section).

11.1 Tag DFEF4B

Tag DFEF4B is a configuration tag. Use it to tell your reader which tracks you want to receive in tag DFEF4D, whether or not to use sentinels, and whether or not to include the PAN as a separate string.

Byte 1:

| 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | NOTES |
|---|---|---|---|---|---|---|---|---|
| - | - | - | - | - | - | - | X | 0 – Disable Track 3 Sentinels 1 – Enable Track 3 Sentinels |
| - | - | - | - | - | - | X | - | 0 – Disable Track 2 Sentinels 1 – Enable Track 2 Sentinels |
| - | - | - | - | - | X | - | - | 0 – Disable Track 1 Sentinels 1 – Enable Track 1 Sentinels |
| - | - | - | - | X | - | - | - | 0 – Disable Track 3 1 – Enable Track 3 |
| - | - | - | X | - | - | - | - | 0 – Disable Track 2 1 – Enable Track 2 |
| - | - | X | - | - | - | - | - | 0 – Disable Track 1 1 – Enable Track 1 |
| - | X | - | - | - | - | - | - | 0 – Disable PAN 1 – Enable PAN |
| X | - | - | - | - | - | - | - | 0 – All Data Elements Found 1 – Only First Element Found |

Byte 2: RFU

Byte 3: RFU

You can use the top bit of the first byte of DFEF4B to control search behavior: If the bit is ON, all data elements requested will be provided (if they exist). If the bit is OFF, only the first element found will be retrieved and placed in DFEF4D.

If you request multiple data items, they will be concatenated. To know the original lengths of the items, you must retrieve and inspect Tag DFEF4C (see below).

To use tag DFEF4B, add it (as a TLV) to your host configuration settings. Use [Set Configuration \(04-00\)](#) or [Contact Set Terminal Data \(60-06\)](#) to send the settings to your reader as you normally would.

NOTE: If this tag does not exist in TerminalSettings, tags DFEF4C and DFEF4D will not be generated.

The default value of this tag is 0x12 (Track 2 enabled, with Sentinels).

Data Search Order

When **"Only First Element Found" (bit 8 = 1)** is set in DFEF4B, Tag DFEF4D will be populated with a single data element according to the following search order

Track 2, Tag 57 (converted to alpha numeric format)
 Track 2, Tag 9F6B
 Track 2, Tag 5F22
 Track 1, Tag 56
 Track 1, Tag 5F21
 PAN, Tag 5A (converted to alpha numeric format)
 Track 3, Tag 58
 Track 3, Tag 5F23

Regardless of the original format, the data will be placed in the DFEF4D tag in alpha numeric format, such that after decryption (and with padding removed) the data will look similar to:

```
3b343736313733393030313031303031303d3135313232303131313433383738303839
3f
```

Which means that after rendering it as ASCII, it would look like:

```
;4761739001010010=15122011143878089?
```

When **"All Data Elements Found" (BIT 8)**, is specified in DFEF4B, Tag DFEF4D will be populated with a single instance of each requested data element, according to the following order:

Track 1 requested (bit 6 = 1). Includes first instance of:

Tag 56 = Track 1 Equivalent

Tag 5F21 = Track 1, identical to the data coded

Track 2 requested (bit 5 = 1). Includes first instance of:

Tag 57 = Track 2 Equivalent (converted to alpha numeric format)

Tag 9F6B = Track 2 Data

Tag 5F22 = Track 2, identical to the data coded

Track 3 requested (bit 4 = 1). Includes first instance of:

Tag 58 = Track 3 Equivalent

Tag 5F23 = Track 3, identical to the data coded

PAN requested (bit 7 = 1). Includes:

Tag 5A = PAN (converted to alpha numeric format)

11.2 Sentinels

For any found data element of Track 1, Track 2 or Track 3, sentinels will be included or not included according to the preferences set in bits 1, 2 and 3.

Compressed Numeric Elements

For any data element captured as compressed numeric, the following rules shall apply:

Padding (0xf) shall not be included

Center separators: 0xd shall be converted to 0x3d ("=")

Data shall be encoded as ASCII representation of binary data

example 0x123f = 0x313233 = "123" (ignore padding)

example 0x1234 = 0x31323334 = "1234"

example 0x123d456f = 0x3132333d343536 = "123=456"

11.3 Tag DFEF4C

If tag DFEF4B is set, tags DFEF4C will appear in transaction output.

This tag's 6-byte value provides the native lengths of tracks 1, 2, and 3, and the PAN (if applicable). Two bytes are reserved for future use.

<Track 1 Length><Track 2 Length><Track 3 length><PAN length><RFU><RFU>

A length of 0 indicates track disabled in DFEF4B or data not available. This tag also serves as an indicator of which data element was found first, when "Only First Element Found" is enabled in DFEF4B.

11.4 Tag DFEF4D

If tag DFEF4B is set, tags DFEF4D will appear in transaction output.

This variable-length tag contains track and/or PAN data, encrypted. The exact contents will vary depending on values supplied previously in DFEF4B (see above).

The track data will present track data if it is a MSD transaction, present track Equivalent data if it is a EMV transaction.

When TDES or AES encryption have been used in conjunction with traditional DUKPT, decrypt the data normally, using the 10-byte KSN found in tag DFEE12.

After one of Track1 Data, Track2 Data, and Tag 5A is presented:

- 1) Tag DFEF4C will provide the length of Track and PAN in Tag DFEF4B
- 2) Tag DFEF4D provides the data of Track and PAN in Tag DFEF4B

12. Sample Scenarios and Frame Flow

12.1 Contactless MagStripe Transactions in Auto Poll Mode

For a contactless MagStripe transaction, the reader does not require any setup data from the host device.

1. Command: Set Poll Mode (Auto Poll)

| Header | Cmd | Sub-Cmd | DLen (MSB) | DLen (LSB) | Data | CRC (LSB) | CRC (MSB) |
|-------------------------------|-----|---------|------------------|------------|----------------|-----------|-----------|
| 56 69 56 4F 74 65 63 68 32 00 | 01h | 01h | 00h | 01h | 00 | F6h | 24h |
| ViVOtech2\0 | | | DLen = 1 decimal | | Auto Poll Mode | | |

Response: OK

| Header | Cmd | Status Code | DLen (MSB) | DLen (LSB) | Data | CRC (MSB) | CRC (LSB) |
|-------------------------------|-----|-------------|------------------|------------|------|-----------|-----------|
| 56 69 56 4F 74 65 63 68 32 00 | 01h | 00 | 00h | 00h | | 12h | 53h |
| ViVOtech2\0 | | OK | DLen = 0 decimal | | None | | |

Reader starts polling for cards. The host device should keep checking for data from the reader. If a card has been read, data is available, otherwise there is no data. The [Get Transaction Result](#) command is for retrieving the data. This command is not required for the reader to poll for cards or to carry out a transaction.

2. Command: Get Transaction Result

| Header | Cmd | Sub-Cmd | DLen (MSB) | DLen (LSB) | Data | CRC (LSB) | CRC (MSB) |
|-------------------------------|-----|---------|------------------|------------|------|-----------|-----------|
| 56 69 56 4F 74 65 63 68 32 00 | 03h | 00h | 00h | 00h | | 3Bh | FFh |
| ViVOtech2\0 | | | DLen = 0 decimal | | None | | |

Response: OK, No Track Data, No Clearing Record (No Transaction)

| Header | Cmd | Status Code | DLen (MSB) | DLen (LSB) | Data | CRC (MSB) | CRC (LSB) |
|-------------------------------|-----|-------------|------------------|------------|---|-----------|-----------|
| 56 69 56 4F 74 65 63 68 32 00 | 03h | 00h | 00h | 03h | 00 00 00 | 8Dh | D0h |
| ViVOtech2\0 | | OK | DLen = 3 decimal | | T1 Len = 0, T2 Len = 0, Clearing Record Not Present | | |

Reader continues to poll for cards. No Card has been presented so far.

3. Command: Get Transaction Result

| Header | Cmd | Sub-Cmd | DLen (MSB) | DLen (LSB) | Data | CRC (LSB) | CRC (MSB) |
|-------------------------------|-----|---------|------------------|------------|------|-----------|-----------|
| 56 69 56 4F 74 65 63 68 32 00 | 03h | 00h | 00h | 00h | | 3Bh | FFh |
| ViVOtech2\0 | | | DLen = 0 decimal | | None | | |

Response: OK, No Track Data, No Clearing Record (No Transaction)

| Header | Cmd | Status Code | DLen (MSB) | DLen (LSB) | Data | CRC (MSB) | CRC (LSB) |
|-------------------------------|-----|-------------|------------------|------------|---|-----------|-----------|
| 56 69 56 4F 74 65 63 68 32 00 | 03h | 00h | 00h | 03h | 00 00 00 | 8Dh | D0h |
| ViVOtech2\0 | | OK | DLen = 3 decimal | | T1 Len = 0, T2 Len = 0, Clearing Record Not Present | | |

Reader continues to poll for cards. No Card has been presented so far.

Reader continues to poll for cards. Card presented and accepted by the reader.

4. Command: Get Transaction Result

| Header | Cmd | Sub-Cmd | DLen (MSB) | DLen (LSB) | Data | CRC (LSB) | CRC (MSB) |
|-------------------------------|-----|---------|------------------|------------|------|-----------|-----------|
| 56 69 56 4F 74 65 63 68 32 00 | 03h | 00h | 00h | 00h | | 3Bh | FFh |
| ViVOtech2\0 | | | DLen = 0 decimal | | None | | |

Response: OK, Track1, Track2 Data available

| Header | Cmd | Status Code | DLen (MSB) | DLen (LSB) | Data | |
|-------------------------------|-----|-------------|----------------|------------|-----------------|-------------------------------------|
| 56 69 56 4F 74 65 63 68 32 00 | 03h | 00 | 00h | 64h | 3Ch | 42 35 34 31 33 31 32 33 34 35 36 37 |
| ViVOtech2\0 | | OK | DLen = 100 dec | | T1Len= 60 (dec) | Track 1 Data "B54131234567" |

| Data | |
|---|--|
| 38 34 38 30 38 5E 53 4D 49 54 48 2F 4A 4F 48 4E 5E 30 35 30 38 31 30 31 33 33 35 33 37 33 33 33 36 30 37 32 32 32 | |
| Track 1 Data "84808^SMITH/JOHN^050810133537333607222" | |

| Data | |
|---|-----|
| 32 32 37 32 34 31 31 31 31 33 | 25h |
| 35 34 31 33 31 32 33 34 35 36 37 38 34 38 30 38 3D 30 35 30 38 31 30 31 | |

| | | |
|---------------------------|--------------------|--|
| Track1 Data 2272411113 | T2Len= 37 (dec) | Track 2 Data "5413123456784808=0508101" |
|---------------------------|--------------------|--|

| Data | CRC (MSB) | CRC (LSB) |
|--|--------------------------------|-----------|
| 39 36 30 37 39 39 37 32 34 32 31 38 33 | 00h | F1h |
| Track 2 Data 9607997242183 | Clearing Record Not Present | FBh |

Contactless MagStripe card was presented and accepted by the reader before the [Get Transaction Result](#) command. Track 1 and Track 2 data returned in response.

12.2 Contactless MagStripe Transactions in Poll on Demand Mode

For a contactless MagStripe transaction, the reader does not require any setup data from the host device.

1. Command: Set Poll Mode (Poll on Demand)

| Header | Cmd | Sub-Cmd | DLen (MSB) | DLen (LSB) | Data | CRC (LSB) | CRC (MSB) |
|----------------------------------|-----|---------|---------------------|------------|---------------------|-----------|-----------|
| 56 69 56 4F 74 65 63 68 32 00 | 01h | 01h | 00h | 01h | 01h | D7h | 34h |
| ViVotech2\0 | | | DLen = 1 decimal | | Poll on Demand Mode | | |

Response: OK

| Header | Cmd | Status Code | DLen (MSB) | DLen (LSB) | Data | CRC (MSB) | CRC (LSB) |
|----------------------------------|-----|-------------|---------------------|------------|------|-----------|-----------|
| 56 69 56 4F 74 65 63 68 32 00 | 01h | 00h | 00h | 00h | | 12h | 53h |
| ViVotech2\0 | | OK | DLen = 0 decimal | | None | | |

Reader stops polling for cards. The host device must issue an Activate command to allow the reader to poll for a card and carry out a transaction.

2. Command: Activate (MagStripe/EMV)

| Header | Cmd | Sub-Cmd | DLen (MSB) | DLen (LSB) | Data | CRC (LSB) | CRC (MSB) |
|----------------------------------|-----|---------|---------------------|------------|-----------------------------------|-----------|-----------|
| 56 69 56 4F 74 65 63 68 32 00 | 02h | 01h | 00h | 01h | 0Ah | 6Eh | 6Bh |
| ViVotech2\0 | | | DLen = 1 decimal | | Timeout = 10 Seconds (decimal) | | |

Reader starts polling for cards. No card is presented. Reader stops polling after 10 seconds and sends back a response indicating timeout.

Response: Error (Timeout); No Card Detected.

| Header | Cmd | Status Code | DLen (MSB) | DLen (LSB) | Data | CRC (MSB) | CRC (LSB) |
|-------------------------------|-----|-------------|------------------|------------|------|-----------|-----------|
| 56 69 56 4F 74 65 63 68 32 00 | 02h | 08h | 00h | 00h | | 20h | 2Eh |
| ViVOtech2\0 | | Time Out | DLen = 0 decimal | | None | | |

Reader is not polling for cards.

3. Command: Activate (MagStripe/EMV)

| Header | Cmd | Sub-Cmd | DLen (MSB) | DLen (LSB) | Data | CRC (LSB) | CRC (MSB) |
|-------------------------------|-----|---------|------------------|------------|--------------------------------|-----------|-----------|
| 56 69 56 4F 74 65 63 68 32 00 | 02h | 01h | 00h | 01h | 0Ah | 6Eh | 6Bh |
| ViVOtech2\0 | | | DLen = 1 decimal | | Timeout = 10 Seconds (decimal) | | |

Reader starts polling for cards. A contactless MagStripe card is presented within 10 seconds. Reader completes transaction, even if more than ten seconds pass since Activate command was received. After completing transaction the reader does not restart polling and just sends back the response containing the Track1 and Track2 data.

Response: OK, Track1, Track2 Data available

| Header | Cmd | Status Code | DLen (MSB) | DLen (LSB) | Data | |
|-------------------------------|-----|-------------|----------------|------------|-----------------|-------------------------------------|
| 56 69 56 4F 74 65 63 68 32 00 | 02h | 00 | 00h | 64h | 3Ch | 42 35 34 31 33 31 32 33 34 35 36 37 |
| ViVOtech2\0 | | OK | DLen = 100 dec | | T1Len= 60 (dec) | Track 1 Data "B54131234567" |

| Data | |
|---|--|
| 38 34 38 30 38 5E 53 4D 49 54 48 2F 4A 4F 48 4E 5E 30 35 30 38 31 30 31 33 33 35 33 37 33 33 33 36 30 37 32 32 32 | |
| Track 1 Data 84808^SMITH/JOHN^050810133537333607222 | |

| Data | |
|---|-----------------|
| 32 32 37 32 34 31 31 31 31 33 | 25h |
| 35 34 31 33 31 32 33 34 35 36 37 38 34 38 30 38 3D 30 35 30 38 31 30 31 | |
| Track1 Data 2272411113 | T2Len= 37 (dec) |
| Track 2 Data "5413123456784808=0508101" | |

| Data | | | | CRC (MSB) | CRC (LSB) |
|--|--------------------------------|--|--|-----------|-----------|
| 39 36 30 37 39 39 37 32 34 32 31 38 33 | 00h | | | F6h | 7Fh |
| Track 2 Data "607997242183 | Clearing Record Not Present | | | | |

12.3 EMV (M/Chip) Transaction in Poll on Demand Mode

The correct CA public keys required by the Cards that is read have already been set up using the Key Management Commands (refer to [Key Management](#)). This operation needs to be done only one time for each key. Keys are retained over power cycles by the reader.

1. Command: Set Poll Mode (Poll on Demand)

| Header | Cmd | Sub-Cmd | DLen (MSB) | DLen (LSB) | Data | CRC (LSB) | CRC (MSB) |
|-------------------------------|-----|---------|------------------|------------|---------------------|-----------|-----------|
| 56 69 56 4F 74 65 63 68 32 00 | 01h | 01h | 00h | 01h | 01h | D7h | 34h |
| ViVOtech2\0 | | | DLen = 1 decimal | | Poll on Demand Mode | | |

Response: OK

| Header | Cmd | Status Code | DLen (MSB) | DLen (LSB) | Data | CRC (MSB) | CRC (LSB) |
|-------------------------------|-----|-------------|------------------|------------|------|-----------|-----------|
| 56 69 56 4F 74 65 63 68 32 00 | 01h | 00 | 00h | 00h | | 12h | 53h |
| ViVOtech2\0 | | OK | DLen = 0 decimal | | None | | |

Reader stops polling for cards. The host device must issue an Activate command to allow the reader to poll for a card and carry out a transaction.

2. Command: Set Configuration (Host Country Code, Transaction Currency Code)

| Header | Cmd | Sub-Cmd | DLen (MSB) | DLen (LSB) | Data | | CRC (LSB) | CRC (MSB) |
|-------------------------------|-----|---------|-------------------|------------|-----------------------|-------------------------|-----------|-----------|
| 56 69 56 4F 74 65 63 68 32 00 | 04h | 00h | 00h | 0Ah | 9F 1A 02 00 56 | 5F 2A 02 09 78 | 69h | 03h |
| ViVOtech2\0 | | | DLen = 10 decimal | | TLV Host Country Code | TLV Trans Currency Code | | |

Assuming the current host values are used for all other parameters (unless specified otherwise in Activate command).

Response: OK

| Header | Cmd | Status Code | DLen (MSB) | DLen (LSB) | Data | CRC (MSB) | CRC (LSB) |
|-------------------------------|-----|-------------|------------------|------------|------|-----------|-----------|
| 56 69 56 4F 74 65 63 68 32 00 | 04h | 00 | 00h | 00h | | AEh | 16h |
| ViVOtech2\0 | | OK | DLen = 0 decimal | | None | | |

Reader is still not polling for cards.

Note: These parameter values may not apply to all cards. The host must make sure that correct values have been defined for the parameters based on card requirements otherwise a transaction fails.

3. Command: Activate (MagStripe/EMV)

| Header | Cmd | Sub-Cmd | DLen (MSB) | DLen (LSB) | Data | CRC (LSB) | CRC (MSB) |
|-------------------------------|-----|---------|------------------|------------|--------------------------------|----------------------|-----------|
| 56 69 56 4F 74 65 63 68 32 00 | 02h | 01h | 00h | 06h | 0Ah 9A 03 05 08 18 | 77h | 1Dh |
| ViVOtech2\0 | | | DLen = 1 decimal | | Timeout = 10 Seconds (decimal) | TLV Transaction Date | |

Reader starts polling for cards. A contactless EMV (M/Chip) card is presented within 10 seconds. Reader completes transaction, even if more than ten seconds pass since Activate command was received. After completing transaction the reader does not restart polling and just sends back the response containing the Clearing Record data.

Response: OK, Clearing Record and additional Data available

| Header | Cmd | Status Code | DLen (MSB) | DLen (LSB) | Data |
|-------------------------------|-----|-------------|----------------|------------|---|
| 56 69 56 4F 74 65 63 68 32 00 | 02h | 00 | 00h | ABh | 00h 00h 01h |
| ViVOtech2\0 | | OK | DLen = 171 dec | | T1Len = 0 (dec) T2Len = 0 (dec) Clearing Record Present |

| Data |
|---|
| E1 56 9F 1A 02 01 58 9F 02 06 00 00 00 00 00 01 5F 2A 02 09 01 9A 03 05 08 02 9C 01 00 95 05 00 00 00 00 00 9F 37 |
| Clearing Record (DE 055) |

| Data |
|--|
| 04 84 77 98 32 82 02 58 80 9F 26 08 02 BB 21 5D D9 06 94 01 9F 27 01 40 9F 10 12 02 10 90 08 01 22 30 00 00 00 00 |
| Clearing Record (DE 055) |

| Data | | | |
|-------------------------------------|----------------------------------|--------------------|----------------------------|
| 00 00 00 00 15 00 FF 9F 36 02 00 D0 | 5A 08 54 12 34 00 00 00 00 19 | 5F 34 01 00 | 5F 24 03 10 07 31 |
| Clearing Record (DE 055) | TLV App PAN | TLV PAN Seq Num | TLV App Expiration Date |

| Data | | | |
|--|----------------------|----------------|-------------------------------------|
| 50 0A 4D 61 73 74 65 72 43 61 72 64 | 9F 34 03 00 1F 03 | 9F 45 02 DA C0 | 9F 4C 08 00 00 00 00 00 00 00 00 |
| TLV Application Label | CVM Results | Data Auth Code | ICC Dynamic Number |

| Data | | |
|---|--------------------------------|-----------------------------------|
| 57 13 54 12 34 00 00 00 00 19 D1 00 72 01 14 43 14 31 00 00 0F | 56 00 | 9B 02 C8 00 |
| TLV Track 2 Equivalent Data | TLV Track 1 Equivalent Data | Transaction Status Information |

| Data | CRC (MSB) | CRC (LSB) |
|---|--------------|--------------|
| 5F 20 1A 53 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 | 27h | 60h |
| Cardholder Name | | |

13. Appendix A.1: User Experience Illustration

Following are list of messages and the message flow for one user experience.

Summary of LCD Messages

NOTE: For the complete list of possible messages, see the [Default Message Index](#).

| User Interface States | ViVOtech User Experience |
|--------------------------------|--|
| Idle | 00: Idle Message (Welcome) |
| Polling | 01: Present card (Please Present Card) |
| Time out or Transaction Cancel | 02: Time Out or Transaction cancel (No Card) |
| Transaction In Progress | 03: Transaction between reader and card is in the middle (Processing...) |
| Transaction Succeed | 04: Transaction Pass (Thank You) |
| Transaction Fail | 05: Transaction Fail (Fail) |
| Configurable messages | 06: Amount (Amount \$ 0.00 Tap Card) |
| | 07: Balance or Offline Available funds (Balance \$ 0.00) |
| | 08: Insert or Swipe card (Use Chip & PIN) |
| | 09: Try Again(Tap Again) |
| | 0A: Indicate the custom to present only one card (Present 1 card only) |
| | 0B: Indicate the custom to wait for authentication/authorization (Wait) |

The flow diagram below illustrates how an external UI may be controlled, using asynchronous UI events.

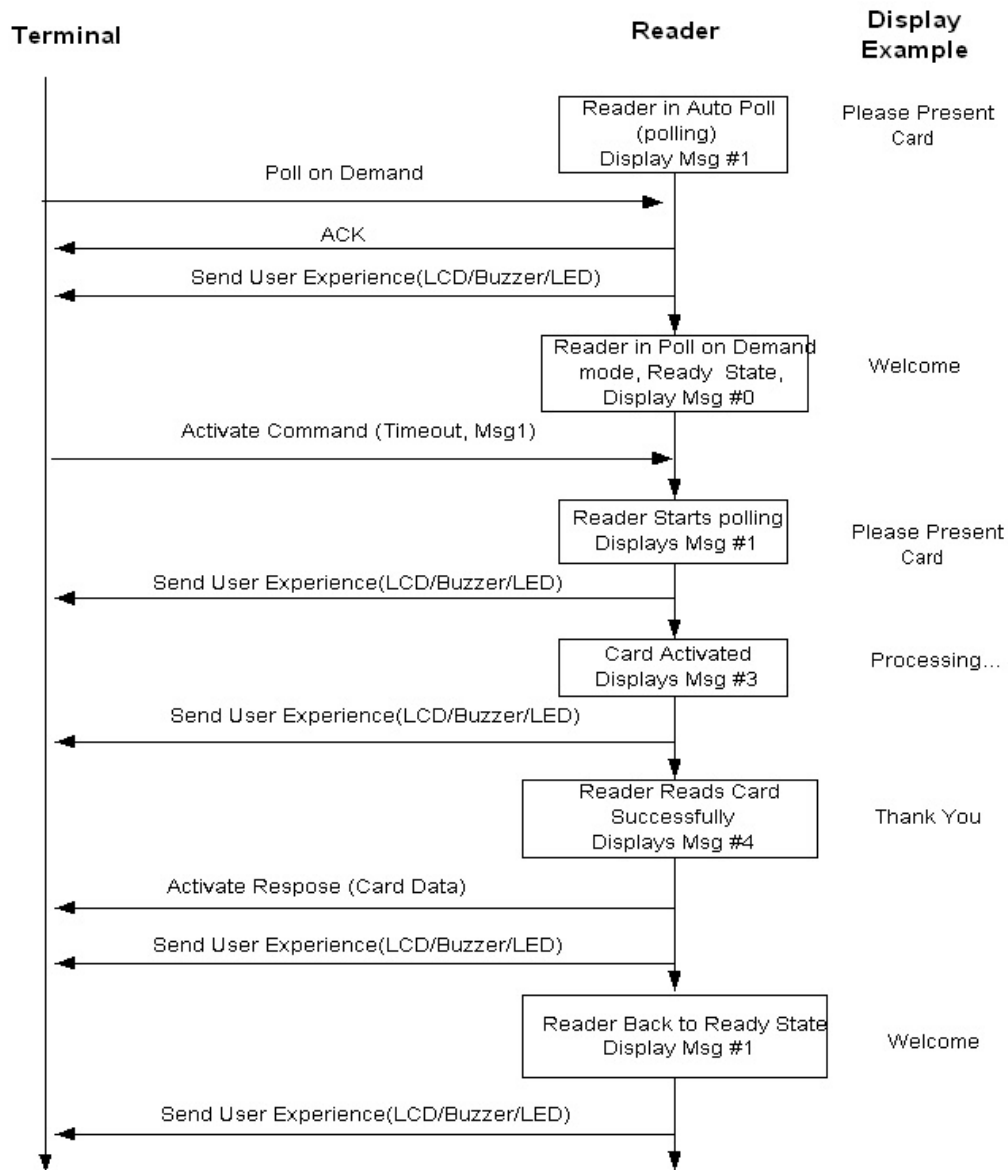


Figure 1a: Poll On Demand: Reader Reads Card Successfully

14. Appendix A.2: Audible User Interface

Some readers do not have an LCD display. In that case, audible tones and lights indicate the status and when an action must be taken.

The following table describes the audible user interface tones.

Table 82: Definition of Audible Tones

| Tone Name | Sound |
|--------------------|-------------------|
| Alert | Two short beeps |
| Card Read Complete | One long beep |
| Check Phone | Three short beeps |

The Alert tone is an indication to the card user that something unusual has occurred and some action must be taken (for example, insert a card, swipe a card, check your mobile phone, or use one card at a time).

The following table describes the audible tones emitted by the reader for each of the interfaces under various conditions:

Table 83: Meaning of Audible Tones by Interface

| Card/ Interface Type | Tone | Reason for Tone |
|----------------------|----------------------------|---|
| Contactless | Check Phone | <ul style="list-style-type: none"> Consumer interaction required (user needs to do something on the phone, such as enter a PIN) |
| | Alert | <ul style="list-style-type: none"> Card Read Error Collision Detected Unsupported Card Application Error No Response after Restart (only for VCP52.1.1 and ExpressPay 3.0) |
| | Card Read Complete + Alert | <ul style="list-style-type: none"> Card Read Complete and Card Error Card Read Complete & switch to another interface |
| | Card Read Complete | <ul style="list-style-type: none"> Transaction Approved, Offline Transaction Declined Offline (see status and error codes) Transaction Approved Online Transaction Declined Online |
| Contact | Alert | <ul style="list-style-type: none"> Card Removed Transaction terminated with Error Switch from Contact to MagStripe Interface |
| | Card Read Complete | <ul style="list-style-type: none"> Transaction Approved, Offline Transaction Declined, Offline Transaction Approved, Online Transaction Declined, Online |
| MagStripe | Alert | <ul style="list-style-type: none"> Card Swipe Error |
| | Card Read Complete | <ul style="list-style-type: none"> Card Swiped Successfully |

15. Appendix A.3: Configurable AID Usage Examples

These examples show the communications between a Configurable AID capable reader and an attached PC simulating a POS.

Disable System AID

From POS →

```
56 69 56 4F 74 65 63 68 32 00 04 04 00 0A 9F 06 07 A0 00 00 00 04
10 10 25 59
```

From Reader ←

```
56 69 56 4F 74 65 63 68 32 00 04 00 00 00 AE 16
```

Uses the DCA Command (Delete Configurable AID - Cmd 4, Sub Cmd 4)

9F06 07 A0 00 00 00 04 10 10 – Selects the AID Number

Enable System AID

From POS →

```
56 69 56 4F 74 65 63 68 32 00 04 02 00 0E FF E4 01 00 9F 06 07 A0
00 00 00 04 10 10 D2 A8
```

From Reader ←

```
56 69 56 4F 74 65 63 68 32 00 04 07 00 00 2B 86
```

Uses the SCA Command (Set Configurable AID - Cmd 4, Sub Cmd 2)

FFE4 01 00 – Selects Group 0

9F06 07 A0 00 00 00 04 10 10 – Selects the AID Number

Add a New Configurable AID

From POS →

```
56 69 56 4F 74 65 63 68 32 00 04 02 00 18 FF E4 01 00 9F 06 05 B0
12 34 56 78 FF E2 01 03 FF E1 01 01 FF E5 01 0A 09 AB
```

From Reader ←

```
56 69 56 4F 74 65 63 68 32 00 04 00 00 00 AE 16
```

Uses the SCA Command (Set Configurable AID - Cmd 4, Sub Cmd 2)

FFE4 01 00 – Selects Group 0

9F06 05 B0 12 34 56 78 – Selects the AID Number

FFE2 01 03 – Selects Application Flow

FFE1 01 01 – Enables Partial Selection

FFE5 01 0A – Specify Maximum Partial Selection Length

Delete a Configurable AID

From POS →

56 69 56 4F 74 65 63 68 32 00 04 04 00 08 9F 06 05 B0 12 34 56 78
DF 97

From Reader ←

56 69 56 4F 74 65 63 68 32 00 04 00 00 00 AE 16

Uses the DCA Command (Delete Configurable AID - Cmd 4, Sub Cmd 4)

9F06 05 B0 12 34 56 78 – Specifies the AID to delete.

Create a New Group

From POS →

56 69 56 4F 74 65 63 68 32 00 04 03 00 0D FF E4 01 01 FF F1 06 00
00 00 01 00 00 64 03

From Reader ←

56 69 56 4F 74 65 63 68 32 00 04 00 00 00 AE 16

Uses the SCG Command (Set Configurable Group - Cmd 4, Sub Cmd 3)

FFE4 01 01 – Specify the NEW group number 1.

FFF1 06 00 00 00 01 00 00 – Terminal Transaction Limit.

Connect Existing AID to a Different Group

From POS →

56 69 56 4F 74 65 63 68 32 00 04 02 00 18 FF E4 01 01 9F 06 05 B0
12 34 56 78 FF E2 01 03 FF E1 01 01 FF E5 01 0A FF 7E

From Reader ←

56 69 56 4F 74 65 63 68 32 00 04 00 00 00 AE 16

Uses the SCA Command (Set Configurable AID - Cmd 4, Sub Cmd 2)

FFE4 01 01 – Specify the NEW group number 1.

9F06 05 B0 12 34 56 78 – Specifies the AID.

FFE2 01 03 – Selects Application Flow

FFE1 01 01 – Enables Partial Selection

FFE5 01 0A – Specify Maximum Partial Selection Length

Return Existing AID to Group 0

From POS →

56 69 56 4F 74 65 63 68 32 00 04 02 00 18 FF E4 01 00 9F 06 05 B0
12 34 56 78 FF E2 01 03 FF E1 01 01 FF E5 01 0A 09 AB

From Reader ←

56 69 56 4F 74 65 63 68 32 00 04 00 00 00 AE 16

Uses the SCA Command (Set Configurable AID - Cmd 4, Sub Cmd 2)

FFE4 01 00 – Specify Group number 0.

9F06 05 B0 12 34 56 78 - Specifies the AID.

FFE2 01 03 – Selects Application Flow

FFE1 01 01 – Enables Partial Selection

FFE5 01 0A – Specify Maximum Partial Selection Length

Delete a Group

From POS →

56 69 56 4F 74 65 63 68 32 00 04 05 00 04 FF E4 01 01 0C 5D

From Reader ←

56 69 56 4F 74 65 63 68 32 00 04 00 00 00 AE 16

Uses the DCG Command (Delete Configurable Group - Cmd 4, Sub Cmd 5)

FF E4 01 01 – Specify Group number to delete.

16. Appendix A.4: Preparing Bitmaps for Use with ILM

The serial ILM commands for language support require bitmap images to display messages. In place of 22 text string messages, ILM commands use 22 bitmap images to display messages. These bitmaps are downloaded to the reader as described in [Download ILM Message Command](#).

The bit map images used for ILM support must be modified before they can be downloaded to the reader. You need to make the following changes:

- Replace the standard bmp header with a ViVOpay header
- Invert the image orientation (top to bottom)
- Invert the image color (black to white)
- Reduce image size by cropping unused pixels

All processing of regular monochrome bitmaps must be done before attempting to download the images to the reader. You cannot download color or grayscale images.

ViVOpay BMP Header

For each bitmap image, you must replace the standard bitmap header with a simplified ViVOpay header. The ViVOpay bitmap header is 12 bytes of data in the format shown in the following table, prefixed to the actual bitmap data:

| Bytes 0-1 | Bytes 2-3 | Bytes 4-5 | Bytes 6-7 | Bytes 8-9 | Bytes 10-11 | Bytes 12...n |
|------------------|--------------|------------------|--------------|--------------|----------------|-----------------|
| Bitmap Length | Row Number | Column Number | Height | Width | Reserved | Bitmap data |

All variables in the header are 2 bytes long.

| | |
|----------------------|--|
| Bitmap Length | This data field contains the total number of bytes in the Bitmap Data Field. |
| Row Number | This data field contains the row offset that this image should start at. Value is in PIXELS. |
| Column Number | This data field contains the column offset that this image should start at. Value is in BYTES. |
| Height | This data field contains the number of rows (in pixels) that this image contains. |
| Width | This data field contains the number of columns (in bytes) that this image contains. |
| Reserved | This data field is reserved for future image manipulation options. |
| Bitmap Data | This is the actual image data. |

Inverting the Image

The ViVOpay LCD expects each row of data in the opposite order than is in the bitmap. To invert the image you can employ row swapping. Assuming the bitmap is a 128 x 64 pixel image, each 16 bytes of data constitutes one "row" of 128 pixels (128 pixels / 8 pixels/byte = 16 bytes). Reversing the order of each 16 bytes of data in this case inverts the image.

Inverting the Color

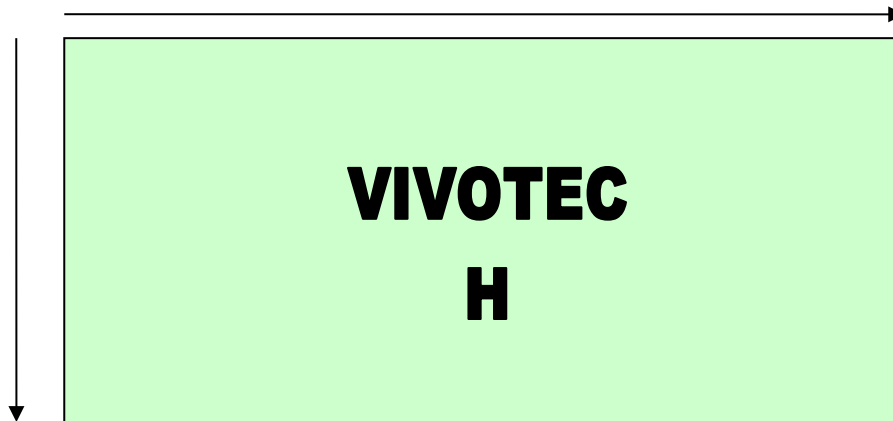
Compared to a regular monochrome bitmap, the image used with ILM commands has inverted color. White areas of the bitmap must be black and black areas white. To invert the color, each bit of a bitmap image must be reversed by performing a NOT operation on each byte of image data.

For example, suppose that 8 pixels were stored as 0x43 (0100 0011 in binary). This value must be reversed with the logical NOT to become 0xBC (1011 1100 in binary). Thus, 0xBC on an LCD matches 0x43 displayed on a PC monitor.

Image Cropping

Although message bitmaps can be sent at maximum screen size, cropping the images speeds the download and uses less memory. Cropping **MUST** be done after all other processing of the bitmap image. Other operations may be done in any order (as long as cropping is done last). Cropping the image requires that you include the row number, column number, height and width parameters in the ViVOpay header. The Column Number/Row Numbers define where the upper-left corner of the bitmap is positioned. The Height and Width parameters determine the area the bitmap takes up on the LCD screen.

For example, here is a 128x64 pixel bitmap:



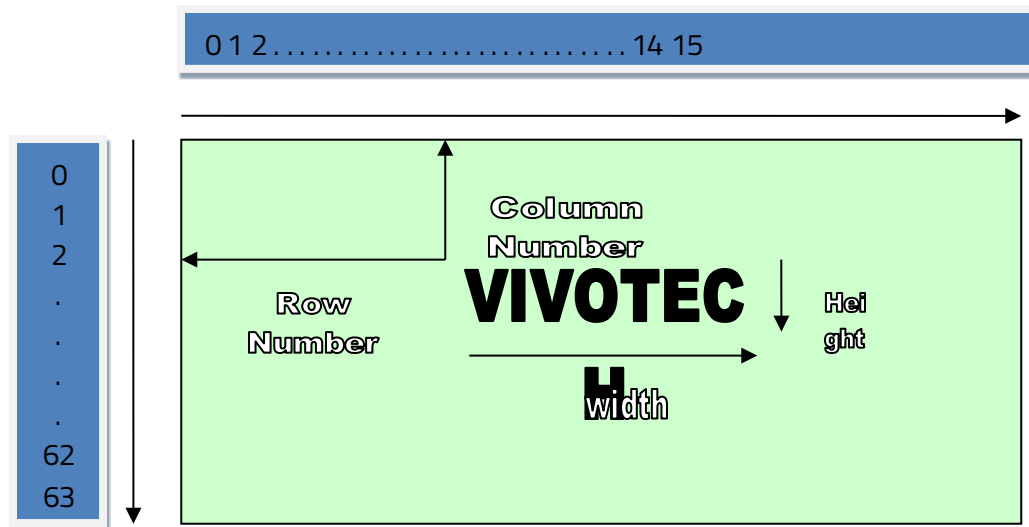
Here is the same image with unnecessary white space cropped out. It is now 50x10 pixels:



Cropping must be expressed in byte boundaries instead of pixels. For example, suppose you want to crop pixels 0 – 11 of a row of 128 pixels. Byte 0 (containing pixels 0–7) can be cropped but Byte 1 (containing pixels 8–15) cannot be cropped, because it contains bits of non-cropped

data. Thus, in this case, the cropping would begin at the Byte 1 (that is, the Column 1 boundary).

The following diagram shows how the header parameters are derived.



The Row Number refers to the number of rows, each of which is a row of pixels. The Column Number refers to a byte location. So Row varies from 0 to 63, and the Column varies from 0 to 15.

Example

The following is a simple example of a bitmap which measures 24x4 pixels (24 pixels per row, 4 rows). All bitmap values shown are arbitrary (and in hexadecimal).

Note: This example does **NOT** include image cropping.

ACTUAL BMP:

```
[14-byte BMP file header] + [40 byte DIB header] +
[Bitmap Data = (11 00 00 00) (22 00 01 00) (33 00 01 00) (44 00 00
00) ]
```

First remove the headers.

```
[Bitmap Data = (11 00 00 00) (22 00 01 00) (33 00 01 00) (44 00 00
00) ]
```

Then invert the image by swapping the rows. In this example, each row is 32 bits (4 bytes long). Row 1 is swapped with the last row (4), Row 2 is swapped with the next-to-last row (3) and so on.

```
[Bitmap Data = (44 00 00 00) (33 00 01 00) (22 00 01 00) (11 00 00
00) ]
```

Next reverse the bits.

```
[Bitmap Data = (BB FF FF FF) (CC FF FE FF) (DD FF FE FF) (EE FF FF  
FF) ]
```

Analyze the image. If the image needs to be cropped for white space reduction, do it now.

Calculate positioning parameters and generate the ILM header.

```
[12 byte ILM header] +  
[Bitmap Data = [BB FF FF FF] [CC FF FE FF] [DD FF FE FF] [EE FF FF  
FF]]
```

This modified image data is now ready to be displayed or stored on the ViVOpay LCD.

17. Appendix A.5: Default Configuration

ViVOpay readers are set to operate out of the box for many applications. This appendix describes the default operating mode and default TLV data object values that have not been previously covered.

Refer to the [Configuration Tag Tables](#) for default TLV values.

Communication Speed

The default baud rate for most units is 19200. The baud rate itself is a configurable parameter that can be set as described in the Interface Developer's Guide. ViVOpay readers can communicate on any COM port specified by the integrators application. Handshaking is disabled (8 N 1).

Polling Mode

The default configuration for the firmware is Auto Poll. While operating in Auto Poll, the reader is constantly polling for contactless transactions. When a card is presented to the field, reader completes the data exchange. Transaction data is obtained by the POS via GET TRANSACTION RESULT, READ FULL TRACK DATA, or a Burst Mode configuration. This poll mode is best suited for contactless-magstripe data transactions (for example, PayPass Mag Stripe & payWave MSD).

Auto Poll mode is not compatible with the EMEA User Interface configuration or M/Chip 3.0 applications. If you are using the EMEA UI or M/Chip 3.0, then you should configure the polling mode to be "Poll on Demand." Refer to the [Set Poll Mode \(01-01\)](#) command.

Burst Mode (Deprecated)

This MSD-centric mode is deprecated. The information shown below is primarily of historical interest. It will be deleted from this document eventually.

While operating in Burst Mode, any time a valid contactless transaction is presented to the reader and the transaction completes successfully, the Burst Mode Payload Packet is immediately transmitted to the POS. The payload packet primarily consists of Track 1 and Track 2 data.

The default configuration for the firmware is Burst Mode, Auto-Off. When configured in this fashion Burst Mode is fully active until receiving a transaction command (such as ACTIVATE TRANSACTION or GET TRANSACTION RESULT), at which time Burst Mode is disabled until the next power cycle.

Burst Mode is only valid for contactless applications returning magstripe Track 1 and Track 2 data, such as PayPass Magstripe & Visa payWave MSD.

To configure Burst Mode, refer to the [Global Configuration Tags](#) table, tag FFF7h. The magstripe reader itself always operates in Burst Mode, regardless of the Burst Mode settings defined in the EMV parameters, meaning that any swipe at the magstripe reader will result in a payload packet immediately being sent across the serial interface.

Burst Mode output from the magstripe reader can be disabled with the Burst Mode parameter. Refer to the [Global Configuration Tags](#) table, tag FFF7h.

RTC/LCD/Buzzer/LED Source

VIVOpay readers are designed with flexibility in regards to the source of the Real Time Clock (RTC), Liquid Crystal Display (LCD), Buzzer, and Light Emitting Diodes (LED). Each of these components can be set to use a source internal or external to the reader. These components can also be disabled by setting the source to "none." As a default the source of each of these components is set as follows:

| | |
|---------|---|
| RTC: | Internal for units with an RTC. External for units without an RTC |
| LCD: | Internal |
| Buzzer: | Internal |
| LED: | Internal |

To configure the RTC/LCD/Buzzer/LED source, refer to the [Set/Get Source for RTC/LCD/Buzzer/LED \(01-05\)](#) command.

17.1 Default Message Index

The LCD message set applied by the ViVOpay reader depends on the User Interface (UI), tag DFEE37 (or the deprecated FFF8). As previously defined, the default UI is ViVOpay, and when setting a new UI (Visa Wave or EMEA) the associated message index must be loaded.

The various UI message indexes are structured as follows.

| Message Index | Dot Matrix LCD Display Message | | | | |
|---------------|--------------------------------|------------|---|--|--|
| | Indication | | ViVotech UI Scheme = '00' | VisaWave UI Scheme = '02' | EMEA UI Scheme = '03' |
| | Language | Line | | | |
| 0x00 | | | WELCOME | WELCOME | WELCOME |
| | ENG | L1: L2: | " Welcome " " " | " Welcome " " " | " Welcome " " " |
| | FRA | L1: L2: | " Bienvenue " " " | " Bienvenue " " " | " Bienvenue " " " |
| | ENG&FRA | L1: L2: | " Welcome " " Bienvenue " | " Welcome " " Bienvenue " | " Welcome " " Bienvenue " |
| 0x01 | | | TAP_OR_SWIPE_CARD | THANK_YOU | TAP_OR_SWIPE_CARD |
| | ENG | L1: L2: | " Please Tap Or " " Swipe Card " | " Thank You " " " | " Please Tap Or " " Swipe Card " |
| | FRA | L1: L2: | " Présentez " " carte SVP " | " Merci " " " | " Présentez " " carte SVP " |
| | ENG&FRA | L1: L2: | " Tap Or Swipe " " Présentez Carte " | " Thank You " " Merci " | " Tap Or Swipe " " Présentez Carte " |
| 0x02 | | | NO_CARD | THANK_YOU | NO_CARD |
| | ENG | L1: L2: | " No Card " " " | " Thank You " " " | " No Card " " " |
| | FRA | L1: L2: | " Pas de carte " " " | " Merci " " " | " Pas de carte " " " |
| | ENG&FRA | L1: L2: | " No Card " " Pas de carte " | " Thank You " " Merci " | " No Card " " Pas de carte " |
| 0x03 | | | PROCESSING | TRANSACTION_COMPLETED | NOT_CONNECTED |
| | ENG | L1: L2: | " Processing " " " | " Transaction " " Completed " | " Not " " Connected " |
| | FRA | L1: L2: | " En cours " " " | " Transaction " " Terminée " | " Pas " " connecté " |
| | ENG&FRA | L1: L2: | " Processing " " En Cours " | "Transaction Done" Transaction Term" | " Not Connected " " Pas connecté " |
| 0x04 | | | THANK_YOU | USE_OTHER_VISA_CARD | CARD_READ_OK |
| | ENG | L1: L2: | " Thank You " " " | " Please Use " "Other VISA Card " | " Card Read OK " " Remove Card " |
| | FRA | L1: L2: | " Merci " " " | " Utilisez une " "autre carte VISA" | " Lecture Carte OK " "Retirez la carte" |
| | ENG&FRA | L1: L2: | " Thank You " " Merci " | "Use Other Card " "Autre carte VISA " | " Remove the card " "Retirez la carte" |
| 0x05 | | | CARD_FAIL | SWIPE_CARD | CARD_FAIL |
| | ENG | L1: | " Fail " | "Use Alternative " | " Fail " |

| | | | | | |
|------|---------|------------|--|--|--|
| | | L2: | " " | " Payment Method " | " " |
| | FRA | L1: L2: | " Échec " " " | " Utilisez une ou " " autre carte " | " Échec " " " |
| | ENG&FRA | L1: L2: | " Card Failure " " Échec Carte " | " Swipe Card " " Passez la carte " | " Card Failure " " Échec Carte " |
| 0x06 | | | AMOUNT | 1_CARD | AMOUNT_EMEA |
| | ENG | L1: L2: | " Amount: " " " | " Present One " " Card Only " | " Present Card: " " " |
| | FRA | L1: L2: | " Montant: " " " | " Présentez une " " seule carte " | "Présentez carte:" " " |
| | ENG&FRA | L1: L2: | "Amount/Montant: " " " | " Present 1 Card " "Présentez 1 Cart" | "Purchase/Achat: " " " |
| 0x07 | | | BALANCE | INTERNATIONAL_CARD | BALANCE_EMEA |
| | ENG | L1: L2: | " Balance: " " " | " International " " Card Only " | " Available: " " " |
| | FRA | L1: L2: | " Solde: " " " | " Internationale " " Carte Seulement" | " Disponible: " " " |
| | ENG&FRA | L1: L2: | " Balance/Solde: " " " | " Card/Carte " "International(e) " | " Available: " " Disponible: " |
| 0x08 | | | USE_CHIP_N_PIN | TRY_AGAIN | SWIPE_CARD |
| | ENG | L1: L2: | " Use Chip " " & PIN " | " Please " " Try Again " | "Use Alternative " " Payment Method " |
| | FRA | L1: L2: | " Insérez " " la carte " | " Ré-essayez " " " | " Utilisez une ou " " autre carte " |
| | ENG&FRA | L1: L2: | " Use Chip & PIN " " Insérez la carte " | "Please Try Again" " Ré-essayez " | " Swipe Card " "Passez la carte" |
| 0x09 | | | TRY_AGAIN | INTERNATIONAL_CARD | TRY_AGAIN EMEA |
| | ENG | L1: L2: | " Please " " Try Again " | " International " " Card Only " | " Try Again " " " |
| | FRA | L1: L2: | " Ré-essayez " " " | " Internationale " " Carte Seulement" | " Ré-essayez " " " |
| | ENG&FRA | L1: L2: | " Please Try Again " " Ré-essayez " | " Card/Carte " "International(e) " | " Try Again " " Ré-essayez " |
| 0x0A | | | 1_CARD | SIGN_RECEIPT | 1_CARD |
| | ENG | L1: L2: | " Present One " " Card Only " | " Please " " Sign Receipt " | " Present One " " Card Only " |
| | FRA | L1: L2: | " Présentez une " " seule carte " | " Signez le recu " " " | " Présentez une " " seule carte " |
| | ENG&FRA | L1: L2: | " Present 1 Card " "Présentez 1 Cart" | " Sign Receipt " " Signez le recu " | "Present 1 Card " "Présentez 1 Cart" |
| 0x0B | | | WAIT | SIGN_RECEIPT | WAIT_EMEA |
| | ENG | L1: L2: | " Please Wait " " " | " Please " " Sign Receipt " | " Please Wait " " " |
| | FRA | L1: L2: | " Attendez " " " | " Signez le recu " " " | " Attendez SVP " " " |
| | ENG&FRA | L1: L2: | " Please Wait " " Attendez " | " Sign Receipt " " Signez le recu " | " Please Wait " " Attendez SVP " |
| 0x0C | | | REMOVE_CARD | ENTER_PIN | REMOVE_CARD |
| | ENG | L1: L2: | " Remove Card " " " | " Please " " Enter PIN " | " Remove Card " " " |
| | FRA | L1: | " Retirez " | "Entrez votre " | " Retirez " |

| | | | | | |
|------|---------|------------|--|--|--|
| | | L2: | "votre carte " | "code " | "votre carte " |
| | ENG&FRA | L1: L2: | "Remove Card " "Retirez la Carte" | "PIN EntryRequire" "Code exigé " | "Remove Card " "Retirez la Carte" |
| 0x0D | | | APPROVED | AVAIL_OFFLINE_AMOUNT | APPROVED |
| | ENG | L1: L2: | "Approved " " " | "Offline Amount " " " | "Approved " " " |
| | FRA | L1: L2: | "Approuvé " " " | "Montant " "hors ligne " | "Approuvé " " " |
| | ENG&FRA | L1: L2: | "Approved " "Approuvé " | "Offline Amount " "Mt hors ligne " | "Approved " "Approuvé " |
| 0x0E | | | NOT_AUTHORIZED | ENTER_PIN | DECLINED |
| | ENG | L1: L2: | "Not " "Authorized " | "Please " "Enter PIN " | "Declined " " " |
| | FRA | L1: L2: | "Non " "autorisé " | "Entrez votre " "code " | "Refusé " " " |
| | ENG&FRA | L1: L2: | "Not Authorized " "Non autorisé " | "PIN EntryRequire" "Code exigé " | "Declined " "Refusé " |
| 0x0F | | | DECLINED | SIGNATURE_REQUIRED | Reserved MSG16 |
| | ENG | L1: L2: | "Declined " " " | "Signature " "Required " | " " " " |
| | FRA | L1: L2: | "Refusé " " " | "Signature " "Requise " | " " " " |
| | ENG&FRA | L1: L2: | "Declined " "Refusé " | "SignatureRequire" "SignatureRequise" | " " " " |
| 0x10 | | | TERMINATED | Reserved MSG17 | Reserved MSG17 |
| | ENG | L1: L2: | "Terminated " " " | " " " " | " " " " |
| | FRA | L1: L2: | "Terminé " " " | " " " " | " " " " |
| | ENG&FRA | L1: L2: | "Cannot Process " "Ne Peut Procés " | " " " " | " " " " |
| 0x11 | | | TRY_OTHER_INTERFACE | TAP_OR_SWIPE_CARD_VISA | TRY_OTHER_INTERFACE |
| | ENG | L1: L2: | "Try Other " "Interface " | "Present Card " " " | "Try Other " "Interface " |
| | FRA | L1: L2: | "Autre Interface " " " | "Présentez" "votre carte" | "Autre Interface " " " |
| | ENG&FRA | L1: L2: | "AnotherInterface" "Autre Interface " | "Purchase/Achat " " " | "AnotherInterface" "Autre Interface " |
| 0x12 | | | USE_OTHER_CARD | REMOVE_CARD | USE_OTHER_CARD |
| | ENG | L1: L2: | "Use Other Card " " " | "Please " "Remove Card " | "Use Other Card " " " |
| | FRA | L1: L2: | "Use Other Card " " " | "Retirez la Carte " | "Use Other Card " " " |
| | ENG&FRA | L1: L2: | "Use Other Card " " " | "Remove Card " "Retirez la Carte" | "Use Other Card " " " |
| 0x13 | | | TIMEOUT | PROCESSING | TIMEOUT |
| | ENG | L1: L2: | "Time Out " " " | "Processing " " " | "Time Out " " " |
| | FRA | L1: L2: | "Pause " " " | "En cours " " " | "Pause " " " |

| | | | | | |
|------|---------|------------|--|--|--|
| | ENG&FRA | L1: L2: | " Time Out " " Pause " | " Processing " " En Cours " | " Time Out " " Pause " |
| 0x14 | | | CANCEL | DECLINED | CANCEL |
| | ENG | L1: L2: | " Cancel " " " | " Declined " " " | " Cancel " " " |
| | FRA | L1: L2: | " Annuler " " " | " Refusé " " " | " Annuler " " " |
| | ENG&FRA | L1: L2: | " Cancel " " Annuler " | " Declined " " Refusé " | " Cancel " " Annuler " |
| 0x15 | | | ONLINE | TERMINATED | ONLINE |
| | ENG | L1: L2: | " Authorizing " " " | " Terminated " " " | " Authorizing " " " |
| | FRA | L1: L2: | " En Cours " " " | " Terminé " " " | " En Cours " " " |
| | ENG&FRA | L1: L2: | " Processing " " En Cours " | " Cannot Process " " Ne Peut Procès " | " Processing " " En Cours " |
| 0x16 | | | SEE_PHONE | SEE_PHONE | SEE_PHONE |
| | ENG | L1: L2: | "See Mobile Phone" " " | "See Mobile Phone" " " | "See Mobile Phone" " " |
| | FRA | L1: L2: | " Voir téléphone " " " | " Voir téléphone " " " | " Voir téléphone " " " |
| | ENG&FRA | L1: L2: | "See Mobile Phone" " Voir téléphone " | "See Mobile Phone" " Voir téléphone " | "See Mobile Phone" " Voir téléphone " |
| 0x17 | | | NOT_ACCEPTED | NOT_ACCEPTED | NOT_ACCEPTED |
| | ENG | L1: L2: | " Not Accepted " " " | " Not Accepted " " " | " Not Accepted " " " |
| | FRA | L1: L2: | " Pas accepté " " " | " Pas accepté " " " | " Pas accepté " " " |
| | ENG&FRA | L1: L2: | " Not Accepted " " Pas accepté " | " Not Accepted " " Pas accepté " | " Not Accepted " " Pas accepté " |
| 0x18 | | | INSERT_CARD | INSERT_CARD | INSERT_CARD |
| | ENG | L1: L2: | " Insert Card " " " | " Insert Card " " " | " Insert Card " " " |
| | FRA | L1: L2: | "Insérez la carte" " " | "Insérez la carte" " " | "Insérez la carte" " " |
| | ENG&FRA | L1: L2: | " Insert Card " "Insérez la carte" | " Insert Card " "Insérez la carte" | " Insert Card " "Insérez la carte" |
| 0x19 | | | REFUND | REFUND | REFUND |
| | ENG | L1: L2: | " Refund " " " | " Refund " " " | " Refund " " " |
| | FRA | L1: L2: | " Remboursement " " " | " Remboursement " " " | " Remboursement " " " |
| | ENG&FRA | L1: L2: | "Refund" "Remboursement " | "Refund" " Remboursement " | "Refund" "Remboursement" |
| 0x1A | | | STOP | STOP | STOP |
| | ENG | L1: L2: | " STOP " " " | " STOP " " " | " STOP " " " |
| | FRA | L1: L2: | " Arrêtez " " " | " Arrêtez " " " | " Arrêtez " " " |
| | ENG&FRA | L1: L2: | " STOP " " Arrêtez " | " STOP " " Arrêtez " | " STOP " " Arrêtez " |

| 0x1B | | | SWIPE_CARD | TAP_OR_SWIPE_CARD | TAP_OR_SWIPE_CARD_VISA |
|------|---------|------------|--|--|--|
| | ENG | L1: L2: | "Use Alternative " " Payment Method " | " Présentez " " carte SVP " | " Present Card " " " |
| | FRA | L1: L2: | " Utilisez une ou " " autre carte " | " Tap Or Swipe " " Présentez Carte " | " Présentez " "votre carte" |
| | ENG&FRA | L1: L2: | " Swipe Card " "Passez la carte" | " Please Tap Or " " Swipe Card " | " Purchase/Achat " " " |
| 0x1C | | | DONE | DONE | DONE |
| | ENG | L1: L2: | " DONE " " " | " DONE " " " | " DONE " " " |
| | FRA | L1: L2: | " Terminé " " " | " Terminé " " " | " Terminé " " " |
| | ENG&FRA | L1: L2: | " DONE " " Terminé " | " DONE " " Terminé " | " DONE " " Terminé " |
| 0x1D | | | TAP_OR_SWIPE_CARD_VISA | CARD_FAIL | THANK_YOU |
| | ENG | L1: L2: | " Present Card " " " | " Fail " " " | " Thank You " " " |
| | FRA | L1: L2: | " Présentez " "votre carte" | " Échec " " " | " Merci " " " |
| | ENG&FRA | L1: L2: | " Purchase/Achat " " " | " Card Failure " " Échec Carte " | " Thank You " " Merci " |
| 0x1E | | | APPROVED_EXTRA | APPROVED_EXTRA | APPROVED_EXTRA |
| | ENG | L1: L2: | " Approved, Bal: " " " | " Approved, Bal: " " " | " Approved, Bal: " " " |
| | FRA | L1: L2: | "Approuvé, Solde:" " " | "Approuvé, Solde:" " " | "Approuvé, Solde:" " " |
| | ENG&FRA | L1: L2: | " Approved, Bal: " "Approuvé, Solde:" | " Approved, Bal: " "Approuvé, Solde:" | " Approved, Bal: " "Approuvé, Solde:" |
| 0x1F | | | DECLINED_EXTRA | DECLINED_EXTRA | DECLINED_EXTRA |
| | ENG | L1: L2: | " Declined, Bal: " " " | " Declined, Bal: " " " | " Declined, Bal: " " " |
| | FRA | L1: L2: | " Refusé Solde: " " " | " Refusé Solde: " " " | " Refusé Solde: " " " |
| | ENG&FRA | L1: L2: | " Declined, Bal: " " Refusé Solde: " | " Declined, Bal: " " Refusé Solde: " | " Declined, Bal: " " Refusé Solde: " |
| 0x20 | | | WAIT_EMEA | WAIT_EMEA | WAIT |
| | ENG | L1: L2: | " Please Wait " " " | " Please Wait " " " | " Please Wait " " " |
| | FRA | L1: L2: | " Attendez SVP " " " | " Attendez SVP " " " | " Attendez " " " |
| | ENG&FRA | L1: L2: | " Please Wait " " Attendez SVP " | " Please Wait " " Attendez SVP " | " Please Wait " " Attendez " |
| 0x21 | | | NOT_CONNECTED | NOT_CONNECTED | TERMINATED |
| | ENG | L1: L2: | " Not " " Connected " | " Not " " Connected " | " Terminated " " " |
| | FRA | L1: L2: | " Pas " " connecté " | " Pas " " connecté " | " Terminé " " " |
| | ENG&FRA | L1: L2: | " Not Connected " " Pas connecté " | " Not Connected " " Pas connecté " | " Cannot Process " " Ne Peut Procés " |

| | | | | | |
|------|---------|------------|--|--|--|
| 0x22 | | | BALANCE_EMEA | BALANCE | BALANCE |
| | ENG | L1: L2: | " Available: " " " | " Balance: " " " | " Balance: " " " |
| | FRA | L1: L2: | " Disponible: " " " | " Solde: " " " | " Solde: " " " |
| | ENG&FRA | L1: L2: | " Available: " " Disponible: " | " Balance/Solde: " " " | " Balance/Solde: " " " |
| 0x23 | | | CARD_READ_OK | WAIT | AMOUNT |
| | ENG | L1: L2: | " Card Read OK " " Remove Card " | " Please Wait " " " | " Amount: " " " |
| | FRA | L1: L2: | "Lecture Carte OK" "Retirez la carte" | " Attendez " " " | " Montant: " " " |
| | ENG&FRA | L1: L2: | " Remove the card" "Retirez la carte" | " Please Wait " " Attendez " | "Amount/Montant: " " " |
| 0x24 | | | AMOUNT_EMEA | CARD_READ_OK | PROCESSING |
| | ENG | L1: L2: | " Present Card: " " " | " Card Read OK " " Remove Card " | " Processing " " " |
| | FRA | L1: L2: | "Présentez carte:" " " | "Lecture Carte OK" "Retirez la carte" | " En cours " " " |
| | ENG&FRA | L1: L2: | "Purchase/Achat: " " " | " Remove the card" "Retirez la carte" | " Processing " " En Cours " |
| 0x25 | | | TRY_AGAIN_EMEA | APPROVED | USE_CHIP_N_PIN |
| | ENG | L1: L2: | " Try Again " " " | " Approved " " " | " Use Chip " " & PIN " |
| | FRA | L1: L2: | " Ré-essayez " " " | " Approuvé " " " | " Insérez " " la carte " |
| | ENG&FRA | L1: L2: | " Try Again " " Ré-essayez " | " Approved " " Approuvé " | " Use Chip & PIN " " Insérez la carte " |
| 0x26 | | | TRANSACTION_COMPLETED | BALANCE_EMEA | TRY_AGAIN |
| | ENG | L1: L2: | " Transaction " " Completed " | " Available: " " " | " Please " " Try Again " |
| | FRA | L1: L2: | " Transaction " " Terminée " | " Disponible: " " " | " Ré-essayez " " " |
| | ENG&FRA | L1: L2: | "Transaction Done" Transaction Term" | " Available: " " Disponible: " | "Please Try Again" " Ré-essayez " |
| 0x27 | | | USE_OTHER_VISA_CARD | AMOUNT | NOT_AUTHORIZED |
| | ENG | L1: L2: | " Please Use " " Other VISA Card" | " Amount: " " " | " Not " " Authorized " |
| | FRA | L1: L2: | " Utilisez une " "autre carte VISA" | " Montant: " " " | " Non " " autorisé " |
| | ENG&FRA | L1: L2: | " Use Other Card " "Autre carte VISA" | "Amount/Montant: " " " | " Not Authorized " " Non autorisé " |
| 0x28 | | | INTERNATIONAL_CARD | AMOUNT_EMEA | TRANSACTION_COMPLETED |
| | ENG | L1: L2: | " International " " Card Only " | " Present Card: " " " | " Transaction " " Completed " |
| | FRA | L1: L2: | " Internationale " " Carte Seulement" | "Présentez carte:" " " | " Transaction " " Terminée " |
| | ENG&FRA | L1: L2: | " Card/Carte " "International(e)" | "Purchase/Achat: " " " | "Transaction Done" Transaction Term" |

| | | | | | |
|------|---------|------------|--|--|--|
| 0x29 | | | SIGN_RECEIPT | NO_CARD | USE_OTHER_VISA_CARD |
| | ENG | L1: L2: | " Please " " Sign Receipt " | " No Card " " " | " Please Use " " Other VISA Card" |
| | FRA | L1: L2: | " Signez le reçu " " " | " Pas de carte " " " | " Utilisez une " "autre carte VISA" |
| | ENG&FRA | L1: L2: | " Sign Receipt " " Signez le reçu " | " No Card " " Pas de carte " | " Use Other Card " "Autre carte VISA" |
| 0x2A | | | ENTER_PIN | USE_CHIP_N_PIN | INTERNATIONAL_CARD |
| | ENG | L1: L2: | " Please " " Enter PIN " | " Use Chip " " & PIN " | " International " " Card Only " |
| | FRA | L1: L2: | "Entrez votre " " code " | " Utilisez " " la puce " | " Internationale " " Carte Seulement" |
| | ENG&FRA | L1: L2: | "PIN EntryRequire" " Code exigé " | " Use Chip & PIN " "Utilizer la puce" | " Card/Carte " "International(e)" |
| 0x2B | | | AVAIL_OFFLINE_AMOUNT | TRY_AGAIN_EMEA | SIGN_RECEIPT |
| | ENG | L1: L2: | " Offline Amount " " " | " Try Again " " " | " Please " " Sign Receipt " |
| | FRA | L1: L2: | " Montant " " hors ligne " | " Ré-essayez " " " | " Signez le reçu " " " |
| | ENG&FRA | L1: L2: | " Offline Amount " " Mt hors ligne " | " Try Again " " Ré-essayez " | " Sign Receipt " " Signez le reçu " |
| 0x2C | | | SIGNATURE_REQUIRED | NOT_AUTHORIZED | ENTER_PIN |
| | ENG | L1: L2: | " Signature " " Required " | " Not " " Authorized " | " Please " " Enter PIN " |
| | FRA | L1: L2: | " Signature " " Requisite " | " Non " " autorisé " | "Entrez votre " " code " |
| | ENG&FRA | L1: L2: | "SignatureRequire" "SignatureRequire" | " Not Authorized " " Non autorisé " | "PIN EntryRequire" " Code exigé " |
| 0x2D | | | INITIALIZING | INITIALIZING | AVAIL_OFFLINE_AMOUNT |
| | ENG | L1: L2: | " Initializing " " Please wait... " | " Initializing " " Please wait... " | " Offline Amount " " " |
| | FRA | L1: L2: | " Initialisation " " Attendez SVP " | " Initialisation " " Attendez SVP " | " Montant " " hors ligne " |
| | ENG&FRA | L1: L2: | " Initializing " " Initialisation " | " Initializing " " Initialisation " | " Offline Amount " " Mt hors ligne " |
| 0x2E | | | NO_MESSAGE | TRY_OTHER_INTERFACE | SIGNATURE_REQUIRED |
| | ENG | L1: L2: | " " " " | " Try Other " " Interface " | " Signature " " Required " |
| | FRA | L1: L2: | " " " " | "Autre Interface " " " | " Signature " " Requisite " |
| | ENG&FRA | L1: L2: | " " " " | "AnotherInterface" "Autre Interface " | "SignatureRequire" "SignatureRequire" |
| 0x2F | | | NO_MESSAGE | USE_OTHER_CARD | INITIALIZING |
| | ENG | L1: L2: | " " " " | " Use Other Card " " " | " Initializing " " Please wait... " |
| | FRA | L1: L2: | " " " " | " Use Other Card " " " | " Initialisation " " Attendez SVP " |
| | ENG&FRA | L1: | " " | " Use Other Card " | " Initializing " |

| | | | | | |
|------|---------|------------|--|--|--|
| | | L2: | " " | " " | " Initialisation " |
| 0x30 | | | NO_MESSAGE | TIMEOUT | NO_MESSAGE |
| | ENG | L1: L2: | " " " | " Time Out " " | " " " |
| | FRA | L1: L2: | " " " | " Pause " " | " " " |
| | ENG&FRA | L1: L2: | " " " | " Time Out " " Pause " | " " " |
| 0x31 | | | NO_MESSAGE | CANCEL | NO_MESSAGE |
| | ENG | L1: L2: | " " " | " Cancel " " | " " " |
| | FRA | L1: L2: | " " " | " Annuler " " | " " " |
| | ENG&FRA | L1: L2: | " " " | " Cancel " " Annuler " | " " " |
| 0x32 | | | NO_MESSAGE | ONLINE | NO_MESSAGE |
| | ENG | L1: L2: | " " " | " Authorizing " " | " " " |
| | FRA | L1: L2: | " " " | " En Cours " " | " " " |
| | ENG&FRA | L1: L2: | " " " | " Processing " " En Cours " | " " " |
| 0x33 | | | TOO_MANY_TAP | TOO_MANY_TAP | TOO_MANY_TAP |
| | ENG | L1: L2: | " Too Many Taps " " | " Too Many Taps " " | " Too Many Taps " " |
| | FRA | L1: L2: | " Trop d'essais " " | " Trop d'essais " " | " Trop d'essais " " |
| | ENG&FRA | L1: L2: | " Too Many Taps " " Trop d'essais " | " Too Many Taps " " Trop d'essais " | " Too Many Taps " " Trop d'essais " |
| 0x34 | | | TERMINATE_PIN_REQUIRED | TERMINATE_PIN_REQUIRED | TERMINATE_PIN_REQUIRED |
| | ENG | L1: L2: | " PIN Required " " | " PIN Required " " | " PIN Required " " |
| | FRA | L1: L2: | " Code Requis " " | " Code Requis " " | " Code Requis " " |
| | ENG&FRA | L1: L2: | " PIN Required " " Code Requis " | " PIN Required " " Code Requis " | " PIN Required " " Code Requis " |

18. Appendix A.6: VP5300 Commands

A number of commands, outlined below, apply specifically to the VP5300 reader.

18.1 Initiate PINPAD Symmetric key loading (63-01)

The host device can send the reader this message to initiate "PIN pad pairing" (assuming a compatible PIN pad is connected).

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 63h | 01h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 63h | See status code table | 00h | 00h | | |

18.2 PINPAD Pass-Through Mode (63-02)

This command can be issued any time.

For this mode, Reader passes all the commands to PINPAD.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|---|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 63h | 02h | 00h | 01h | Pass-Through Mode: 0- PT mode off (Default) 1- PT mode on | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 63h | See status code table | 00h | 00h | | |

18.3 Set Session Key (63-03)

Load Session Keys into reader

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 – Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 63h | 03h | 00h | | See Data Table 2 | | |

Data Table 2: Set Session Key Command Data

| Data Item | Length (bytes) | Description |
|----------------------------|----------------|--|
| Length of Session Key Type | 2 | Session Key Type Length |
| Session Key Type | 1 | Session Key Type: 0 – Data, 1 – MAC, 2 – PIN |
| Length of Session Key | 2 | Session Key Length |
| Session Key | 16 | Session Key is encrypted with Master Key |
| Length of Check Digits | 2 | Check Digits Length |
| Check Digits | 2 | Check digits are gathered from the clear Session Key |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 63h | See status code table | 00h | 00h | | |

18.4 Set Key ID (63-04)

Set Key ID associated with a Master Key set

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 – Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|----------------------|----------------------|-----------------------------|--------------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViV0tech2\0 | 63h | 04h | 00h | 08h | Key ID | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|----------------------------------|---------|---------------------------------------|----------------------|----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | 63h | See status code table | 00h | | | |

Note: Key ID is 8-byte alphanumeric and can only be loaded one time.

18.5 Get Key ID (63-05)

Get Key ID associated with a Master Key set

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------------|---------|-----------------|----------------------|----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViV0tech2\0 | 63h | 05h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 – Byte 14+n-1 | Byte 14+n | Byte 15+n |
|----------------------------------|---------|---------------------------------------|----------------------|----------------------|--------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | 63h | See status code table | 00h | 08h | Key ID | | |

18.6 Generate MAC for Host (63-06)

This command allows creating a MAC-authenticated message for Online authorization.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 - Byte14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 63h | 06h | 00h | | Input Stream | | |

Note: Input stream is in ASCII. Use symbol 0xC0, 0xD0 for clear track 2 data in input stream

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte14- Byte14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---------------------------------------|----------------------|----------------------|-----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 63h | See status code table | 00h | | MAC Data | | |

Example:

Input Stream: 31311C393138C0D04511

Track2:

3B353135303731303230303130373936303D3039303931303134303030303335353F

Input Stream to be MAC:

31311C3931383B353135303731303230303130373936303D3039303931303134303030303335353F4511

MAC Data is 8 bytes

The algorithm is defined in ISO 16609 as TDES CBC MAC

18.7 Get L100 Firmware Version (70-03)

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 70h | 03h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte14- Byte14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|----------------------|----------------------|-----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\ 0 | 63h | See status code table | | | L100 Firmware | | |

Example

OUT: 5669564f74656368320070030000c27f

IN:

5669564f74656368320070000049064944205445434820536d61727450494e204c3130
30204669726d776172652056312e30322e3030392e530d4d4158494d20436869702046
6f72204b3231204c3130302056312e303014ab

19. Appendix A.7: VP6800 Commands

A number of commands, outlined below, apply only to the VP6800.

All commands listed in this chapter are allowed in Pass-Through mode.

Note that Wi-Fi commands are not currently supported for the VP6800; future use cases are forthcoming.

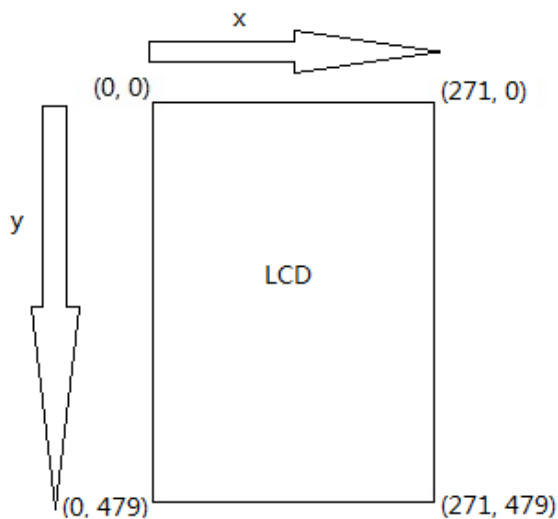
19.1 VP6800 Display Elements

- **Screen**
 - Pre-defined by user
- **Customizable Features**
 - **Text Area**
 - Position
 - Width
 - Height
 - Font
 - Label
 - Buttons
 - Position
 - Font Color
 - Label

Note: Component IDs are counted from 1 in the sequence they are created

19.2 LCD Coordination

The VP6800's LCD is 272*480; refer to the figure below for the x/y coordinates.



19.2.1 LCD UI Control Function

LCD UI Table-1 Screen ID Format

| Byte | Description |
|--------|--|
| Byte 0 | Screen theme ID (Reserved, should be 0x00); screen theme should be re-defined in firmware. The VP6800 and host have pre-loaded screen theme IDs. |
| Byte 1 | Number |

LCD UI Table-2 Object ID Format

| Object ID | Description |
|-----------|---|
| Byte 0 | Component ID 00h: Text Area (*Note 4) 01h: Large Button (*Note 4) 02h: Medium Button (*Note 4) 03h: Invisible Button (70px * 60px (*Note 4) 04h: Numeric Entry (Cannot be added by user, *Note 1) 05h: Ethernet Setting (Cannot be added by user, *Note 2) 06h: LED Widget 07h: Long Press (Cannot be added by user, *Note 3) 08h: Image. Only JPG file is supported (*Note 4) 09h: Video 0Ah: Numeric Button (*Note 4) 0Bh: Scroll Container (The Component ID for the Scroll Container component is 0x80 Component ID) 0Ch: Image Container 0Dh: Extended video FFh: Nothing |
| Byte 1 | Object Number The object number for Numeric Entry is Special: 00h: Enter of numeric entry is clicked 01h: Cancel of numeric entry is clicked The object number for Long Press is Special: 00h: top-left corner is long pressed 01h: top-right corner is long pressed FFh: Nothing |

- ***Note 1:** Numeric entry is only displayed when a K81 requires it. If the numeric entry screen is shown when user UI is live, a numeric entry is kept until the user updates the UI. If the numeric entry screen is shown on the system UI while it is live, a numeric entry is closed after the end of the numeric entry session.
- ***Note 2:** If the reader does not display a screen after a reset, quickly press the top-right corner of the screen, then do a long press in the top-left corner to show the Settings menu.
- ***Note 3:** The area that detects press-and-hold events is 70px*70px and always at the top-left and top-right screen corners. Pressing and holding for one second causes the reader to send a Long Press event with a Long Press Object ID.
- ***Note 4:** These objects can also be added to the scroll container.

19.3 VP6800-Specific Commands

19.3.1 Show Screen (61-05)

This command specifies the screen to display as live on the reader.

Note: The Screen ID **MUST** be created before sending this command. Users can also add or update objects for the Screen, by calling the **Add/Update Object to Screen (61-07/08)** command.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|--------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 61h | 05h | | | See Below | | |

Data Field

| Item | Length (bytes) | Description |
|-------------|----------------|---|
| Screen Name | var. | Screen Name String No longer than 32 bytes ASCII string, terminated by 0x00 |

Use the **Create Screen** command (61-0A) to set the screen name.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|----------------------------------|---------|-------------|----------------------|----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 61h | 00h | 00h | 00h | | |

Example:

- Show Screen "Screen1"
TX: 56 69 56 4F 74 65 63 68 32 00 61 05 00 08 53 63 72 65 65 6E 31
00 91 5C
RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81
- Show Screen "Screen2"
TX: 56 69 56 4F 74 65 63 68 32 00 61 05 00 08 53 63 72 65 65 6E 32
00 C2 09
RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81
- Show Screen "Screen4" (not created yet)
TX: 56 69 56 4F 74 65 63 68 32 00 61 05 00 08 53 63 72 65 65 6E 34
00 C2 09
RX: 56 69 56 4F 74 65 63 68 32 00 61 05 00 00 A0 71 (error code 0x05)

19.3.2 Get Button Event (61-06)

This command allows the host device to check button click events.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 06h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | 00h | var. | See below | | |

Data Format

| Item | Length(byte) | Description |
|-------------|--------------|---|
| Screen ID | 2 | See LCD UI Table-1 |
| Object ID | 2 | See LCD UI Table-2 |
| Screen Name | var. | Screen Name String No longer than 32 bytes ASCII string, terminated by 0x00 |
| Object Name | var. | Object Name String No longer than 32 bytes ASCII string, terminated by 0x00 |

Example:

- No Button clicked:

TX: 56 69 56 4F 74 65 63 68 32 00 61 06 00 00 21 F9

RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 04 00 01 FF FF DF D0

- Object 'Button1' on Screen 'Screen1' is clicked

TX: 56 69 56 4F 74 65 63 68 32 00 61 06 00 00 21 F9

RX: 56 69 56 4F 74 65 63 68 32 00 61 FF 00 14 00 00 01 00 53 63 72
65 65 6E 31 00 42 75 74 74 6F 6E 31 00 32 B9

19.3.3 Add Object to Screen (61-07)

This command adds an object to the screen. However, the screen does not display new objects until calling **Show Screen (61-05)** to update the screen.

Note: Before sending this command, first create the screen with the **Create Screen (61-0A)** command.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14... Byte 13+n | Byte 14+n | Byte 15+n |
|----------------------------------|---------|-------------|-------------------------|-------------------------|-------------------------|--------------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 07h | | | See below | | |

Data Field

| Data Item | Length (bytes) | Description |
|--------------|----------------|---|
| Screen Name | var. | Screen Name String No longer than 32 bytes ASCII string, terminated by 0x00 |
| Object Name | var. | Object Name String No longer than 32 bytes ASCII string, terminated by 0x00 |
| Component ID | 1 | Refer to LCD UI Table-2 byte0 |
| Alignment | 1 | 0: Display object at the horizon center of specified y, while x ignored 1: Display object at specified x and y 2: Display object at center of screen, x, y are both ignored 3: Display object at left of the screen of specified y, while x ignored 4: Display object at right of the screen of specified y, while x ignored |
| x-coordinate | 2 | x-coordinate for text area/button Little endian, range of x-coordinate is [0, 271] |
| y-coordinate | 2 | y-coordinate for text area/button Little endian, range of x-coordinate is [0, 479] |
| features | var. | This field only valid for the following items: text area, LEDs widget, numeric button, scroll container, images container, and extended video; for details, see table below. This field can be omitted for item button. |
| String | var. | Label to show on text area or button (ASCII String, Terminated by 0x00, the Maximum string lens are: <ul style="list-style-type: none"> Text area is 32 bytes, Large button is 25 bytes Medium button is 10 bytes) Filename of image/video (ASCII string, terminated by 0x00, the maximum string lens is 32bytes) Filenames list of image container (maximum number of files 5, maximum string len is 32 bytes each, ASCII string, terminated by 0x00) Filenames list of extended video (maximum number of files is 4, maximum string len is 64 bytes ASCII string, including terminated by 0x00) |

Note:

- **For images:** image files are read from NANDFlash. Only JPEG files without orientation Exif info and width ≤ 272 , height ≤ 480 are supported; otherwise the VP6800 rejects this command.
- **Exif** (exchangeable image file format): a standard that specifies the formats for images, sound, and ancillary tags used by digital cameras (including smartphones), scanners, and other systems that handle image and sound files recorded by digital cameras.
- **For videos:** video files are read from the SD Card and only videos with a width ≤ 272 and height ≤ 480 are supported. The video will be shown +90 degree, so the video source should be rotated -90 degree. Videos must be MJPEG encoded; ffmpeg is a free, third-party tool that can generate the necessary target file.

Text Area Feature:

| Features | Length (bytes) | Description |
|----------|----------------|---|
| Width | 2 | Reserved |
| Height | 2 | Reserved |
| Font ID | 1 | See below table |
| Color | 4 | Byte 0 – B Byte 1 – G Byte 2 – R Byte 3 – Reserved For example: 0xFF000000, blue 0x00FF0000, green 0x0000FF00, red 0x00000000, white 0xFFFF0000, black |

LEDs Widget Feature

| Features | Length (bytes) | Description |
|------------|----------------|---|
| LED0 state | 1 | 00: Off 01: Green 02: Yellow 03: Red |
| LED1 state | 1 | |
| LED2 state | 1 | |
| LED3 state | 1 | |

Numeric Button Feature

| Features | Length (bytes) | Description |
|--------------------------|----------------|--|
| Width | 2 | Width of numeric button |
| Value | 2 | The initial numeric value |
| Font ID | 1 | See below table |
| Outline Color | 4 | Byte 0 – B Byte 1 – G Byte 2 – R |
| Numeric background Color | 4 | |

| | | |
|------------------------|---|---|
| Minus background Color | 4 | Byte 3 – Reserved For example: 0xFF000000, blue 0x00FF0000, green 0x0000FF00, red 0x00000000, white 0xFFFFFFFF00, black |
| Plus background Color | 4 | |
| Numeric text Color | 4 | |
| Minus text Color | 4 | |
| Plus text Color | 4 | |

Scroll Container Feature

| Features | Length (bytes) | Description |
|----------|----------------|----------------------------|
| Width | 2 | Width of scroll container |
| Height | 2 | Height of scroll container |

Image Container Feature

| Features | Length (bytes) | Description |
|----------|----------------|----------------------------|
| Width | 2 | Width of image container |
| Height | 2 | Height of scroll container |

Extended Video Feature

| Features | Length (bytes) | Description |
|------------------|----------------|--|
| Loop | 1 | 0: Do not Loop 1: Loop If this value is 0, when all videos play end, an Object Event Call Back will be sent out by device, refer to Object Event Call Back (61-FF) command . |
| Number of videos | 1 | Vaild value: 1~4 |

Font List

| Font ID | Typography Name | Font | Size |
|---------|-----------------------|------------------------|------|
| 0 | RoundBold_12 | RoundBold.ttf | 12 |
| 1 | RoundBold_18 | RoundBold.ttf | 18 |
| 2 | RoundBold_24 | RoundBold.ttf | 24 |
| 3 | RoundBold_36 | RoundBold.ttf | 36 |
| 4 | RoundBold_48 | RoundBold.ttf | 48 |
| 5 | RoundBold_60 | RoundBold.ttf | 60 |
| 6 | RoundBold_72 | RoundBold.ttf | 72 |
| 7 | RoundCondensedBold_12 | RoundCondensedBold.ttf | 12 |
| 8 | RoundCondensedBold_18 | RoundCondensedBold.ttf | 18 |
| 9 | RoundCondensedBold_24 | RoundCondensedBold.ttf | 24 |
| 10 | RoundCondensedBold_36 | RoundCondensedBold.ttf | 36 |
| 11 | RoundCondensedBold_48 | RoundCondensedBold.ttf | 48 |
| 12 | RoundCondensedBold_60 | RoundCondensedBold.ttf | 60 |
| 13 | RoundCondensedBold_72 | RoundCondensedBold.ttf | 72 |

| Font ID | Typography Name | Font | Size |
|---------|---------------------------|----------------------------|------|
| 14 | RoundCondensedMedium_12 | RoundCondensedMedium_0.ttf | 12 |
| 15 | RoundCondensedMedium_18 | RoundCondensedMedium_0.ttf | 18 |
| 16 | RoundCondensedMedium_24 | RoundCondensedMedium_0.ttf | 24 |
| 17 | RoundCondensedMedium_36 | RoundCondensedMedium_0.ttf | 36 |
| 18 | RoundCondensedMedium_48 | RoundCondensedMedium_0.ttf | 48 |
| 19 | RoundCondensedMedium_60 | RoundCondensedMedium_0.ttf | 60 |
| 20 | RoundCondensedMedium_72 | RoundCondensedMedium_0.ttf | 72 |
| 21 | RoundCondensedSemibold_12 | RoundCondensedSemibold.ttf | 12 |
| 22 | RoundCondensedSemibold_18 | RoundCondensedSemibold.ttf | 18 |
| 23 | RoundCondensedSemibold_24 | RoundCondensedSemibold.ttf | 24 |
| 24 | RoundCondensedSemibold_36 | RoundCondensedSemibold.ttf | 36 |
| 25 | RoundCondensedSemibold_48 | RoundCondensedSemibold.ttf | 48 |
| 26 | RoundCondensedSemibold_60 | RoundCondensedSemibold.ttf | 60 |
| 27 | RoundCondensedSemibold_72 | RoundCondensedSemibold.ttf | 72 |
| 28 | RoundMedium_12 | RoundMedium.ttf | 12 |
| 29 | RoundMedium_18 | RoundMedium.ttf | 18 |
| 30 | RoundMedium_24 | RoundMedium.ttf | 24 |
| 31 | RoundMedium_36 | RoundMedium.ttf | 36 |
| 32 | RoundMedium_48 | RoundMedium.ttf | 48 |
| 33 | RoundMedium_60 | RoundMedium.ttf | 60 |
| 34 | RoundMedium_72 | RoundMedium.ttf | 72 |
| 35 | RoundSemibold_12 | RoundSemibold.ttf | 12 |
| 36 | RoundSemibold_18 | RoundSemibold.ttf | 18 |
| 37 | RoundSemibold_24 | RoundSemibold.ttf | 24 |
| 38 | RoundSemibold_36 | RoundSemibold.ttf | 36 |
| 39 | RoundSemibold_48 | RoundSemibold.ttf | 48 |
| 40 | RoundSemibold_60 | RoundSemibold.ttf | 60 |
| 41 | RoundSemibold_72 | RoundSemibold.ttf | 72 |

The maximum number of each component that can per screen page be created is listed below:

| Item | Maximum can be created for each screen |
|----------------------|--|
| Text Area | 30 |
| Large Button | 8 |
| Medium Button | 16 |
| Invisible Button | 3 |
| Numeric Button | 20 |
| Led widget | 1 |
| Image | 20 |
| Video/Extended Video | 1 |
| Scroll Container | 1 |
| Image Container | 1 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte14-25 | Byte 26 | Byte 27 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | 00h | 0Ch | See below | | |

Data Format

| Item | Length (bytes) | Description |
|----------------|----------------|--|
| Screen ID | 2 | Screen ID format; see LCD UI Table-1 |
| Object ID | 2 | Object ID format; see LCD UI Table-2 |
| Top-Left x | 2 | top-left x-coordinate of object |
| Top-Left y | 2 | top-left y-coordinate of object |
| Bottom-Right x | 2 | bottom-right x-coordinate of object |
| Bottom-Right y | 2 | bottom-right y-coordinate of object |

Example:

- Add "Text1" to "Screen1": "WELCOME TO SONIC0000000 00011"

TX: 56 69 56 4F 74 65 63 68 32 00 61 07 00 3D 53 63 72 65 65 6E 31
 00 54 65 78 74 31 00 00 01 01 00 1E 00 00 00 00 00 09 00 00 00 00
 57 45 4C 43 4F 4D 45 20 54 4F 20 53 4F 4E 49 43 30 30 30 30 30 30
 30 20 30 30 30 0D 0A 31 31 00 EF C9
 RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 0C 00 00 00 00 01 00 1E
 00 11 01 72 00 0C 55

- Add "Button1" to "Screen1": "CHECK BALANCE0000"

TX: 56 69 56 4F 74 65 63 68 32 00 61 07 00 28 53 63 72 65 65 6E 31
 00 42 75 74 74 6F 6E 31 00 01 00 00 00 90 01 43 48 45 43 4B 20 42
 41 4C 41 4E 43 45 30 30 30 30 00 A2 39
 RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 0C 00 00 01 00 0B 00 90
 01 05 01 CC 01 F0 CB

- Add "Button1" to "Screen1": "CHECK BALANCE0000" again

TX: 56 69 56 4F 74 65 63 68 32 00 61 07 00 28 53 63 72 65 65 6E 31
 00 42 75 74 74 6F 6E 31 00 01 00 00 00 90 01 43 48 45 43 4B 20 42
 41 4C 41 4E 43 45 30 30 30 30 00 A2 39
 RX: 56 69 56 4F 74 65 63 68 32 00 61 A0 00 00 F6 1D (error: for this object
 has already existed in that place)

- Add "Medium Button1" to "Screen1": "Medium1"

TX: 56 69 56 4F 74 65 63 68 32 00 61 07 00 25 53 63 72 65 65 6E 31
 00 4D 65 64 69 75 6D 20 42 75 74 74 6F 6E 31 00 02 02 00 00 00 00
 4D 65 64 69 75 6D 31 00 23 6C

```
RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 0C 00 00 02 00 4C 00 D2
00 C4 00 0E 01 B9 DB
```

- Add "Invisible1" to "Screen1"

```
TX: 56 69 56 4F 74 65 63 68 32 00 61 07 00 19 53 63 72 65 65 6E 31
00 49 6E 76 69 73 69 62 6C 65 31 00 03 01 00 00 00 00 15 53
RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 0C 53 63 03 00 00 00 00
00 46 00 3C 00 5D 8B
```

- Add "LEDs" to "Screen1"

```
TX: 56 69 56 4F 74 65 63 68 32 00 61 07 00 17 53 63 72 65 65 6E 31
00 4C 45 44 73 00 06 01 00 00 00 00 01 02 03 66 0B
RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 0C 53 63 06 00 00 00 00
00 10 01 32 00 D9 66
```

- Add "pic.JPG" to "Screen1"; to complete this command, first transfer the file "pic.JPG" (file name: "UTITLED1.JPG"):

```
TX: 56 69 56 4F 74 65 63 68 32 00 61 07 00 24 53 63 72 65 65 6E 31
00 70 69 63 2E 4A 50 47 00 08 01 00 00 00 00 55 4E 54 49 54 4C 45
44 31 2E 4A 50 47 00 CA 95
RX: 56 61 00 00 0C 00 00 08 01 00 00 00 00 3C 00 49 00 E9 84
```

- Add "ScrollContainer_0000" to "Screen1"; to complete this command

```
TX: 56 69 56 4F 74 65 63 68 32 00 61 07 00 27 53 63 72 65 65 6E 31
00 53 63 72 6F 6C 6C 43 6F 6E 74 61 69 6E 65 72 5F 30 30 30 30 00
0B 01 00 00 1E 00 10 01 10 01 CB 7B
```

- Add "Exvideo" to "Screen1"

```
TX: 56 69 56 4F 74 65 63 68 32 00 61 07 00 2B 53 63 72 65 65 6E 31
00 45 78 76 69 64 65 6F 00 0D 02 1E 00 00 00 00 02 76 69 64 65 6F
31 2E 61 76 69 00 6E 66 63 2E 61 76 69 00 E0 CF
RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 0C 00 00 0D 00 01 00 8C
00 0F 01 54 01 BD CC
```

This command generates an error if the picture is larger than the screen resolution.

19.3.4 Update Object to Screen (61-08)

This command updates a screen object. However, the screen does not display updated objects until calling **Show Screen (61-05)** to update the screen.

Note: Before sending this command, first create the screen with the **Create Screen (61-0A)** command and add an object with the **Add Object to Screen (61-07)** command.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\ 0 | 61h | 08h | | | See below | | |

Data Format

| Data Item | Length (bytes) | Description |
|-------------|----------------|---|
| Screen Name | var | Screen Name String No longer than 32 bytes ASCII string, terminated by 0x00 |
| Object Name | var | Object Name String No longer than 32 bytes ASCII string, terminated by 0x00 |
| Item ID | 1 | 0 – Label 1 – Color 2 – Position |
| features | var | See table below |

The table below describes the attributes affected by the **features** field depending on the Component ID (which is byte 0 of the Object ID).

| Item ID | Description of feature field |
|----------|--|
| Label | String, variable length |
| Color | Color value for label, 4 bytes Byte 0 – B Byte 1 – G Byte 2 – R Byte 3 – Reserved Color value for LEDs, 4 bytes Byte 0 – LED0 color index (0:Off, 1: Green, 2: Yellow, 3: Red) Byte 1 – LED1 color index (0:Off, 1: Green, 2: Yellow, 3: Red) Byte 2 – LED2 color index (0:Off, 1: Green, 2: Yellow, 3: Red) Byte 3 – LED3 color index (0:Off, 1: Green, 2: Yellow, 3: Red) |
| Position | Alignment: 1byte 0: Display object at the horizon center of specified y, while x is ignored 1: Display object at specified x and y 2: Display object at center of screen, x, y are both ignored 3: Display object at left of the screen, while x is ignored 4: Display object at right of the screen, while x is ignored x-coordinate: 2 bytes y-coordinate: 2 bytes |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | 00h | 08h | See below | | |

Data Format

| Features | Length (bytes) | Description |
|-----------------------------|----------------|--|
| Position and Label for Text | 8 | Top-left x-coordinate: 2bytes Top-left y-coordinate: 2bytes Bottom-right x-coordinate: 2bytes bottom-right y-coordinate: 2bytes |

Example:

- Update "Text1" String on "Screen1":Text1

TX: 56 69 56 4F 74 65 63 68 32 00 61 08 00 1B 53 63 72 65 65 6E 31
 00 54 65 78 74 31 00 00 54 65 78 74 31 20 4C 61 62 65 6C 00 F7 C6
 RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 08 01 00 1E 00 6F 00 3A
 00 A0 8E

- "Text1" Position on "Screen1":Center

TX: 56 69 56 4F 74 65 63 68 32 00 61 08 00 14 53 63 72 65 65 6E 31
 00 54 65 78 74 31 00 02 02 00 00 00 00 D8 CB
 RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 08 51 00 E2 00 BF 00 FE
 00 83 26

- "Text1" color on "Screen1":RED

TX: 56 69 56 4F 74 65 63 68 32 00 61 08 00 13 53 63 72 65 65 6E 31
 00 54 65 78 74 31 00 01 00 00 FF FF FC 43
 RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81

19.3.5 Add Object to Scroll Container (61-18)

The **Add Object to Scroll Container** command adds an object to the specified container. The Scroll Container name and ID is predefined for each screen. Follow the steps below to add a scroll container to the screen:

1. Complete the **Create Screen (61-0A)** command to get the Screen ID; for example, a ScreenID of **000A**.
2. Complete the **Add Object to Scroll Container (61-18)** command. A Scroll Container name is "ScrollContainer_" plus the Screen ID; for example, **ScrollContainer_000A**.
3. Complete the **Add Object to Screen (61-07)** command to add the Scroll Container to the screen.
4. Complete the **Show Screen (61-05)** command to display the screen.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|----------------------------------|---------|-------------|-------------------------|-------------------------|--------------------------|--------------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 18h | | | See below | | |

Data Field

| Data Item | Length (bytes) | Description |
|-----------------------|----------------|--|
| Scroll Container Name | 21 | "ScrollContainer_XXXX" (XXXX: Screen ID) ASCII string, terminated by 0x00 |
| Object Name | var. | Object Name string No longer than 32 bytes ASCII string, terminated by 0x00 |
| Component ID | 1 | Refer to LCD UI Table-2 byte 0 |
| Alignment | 1 | 0: Display object at the horizon center of specified y, while x ignored 1: Display object at specified x and y 3: Display object at left of the screen of specified y, while x ignored 4: Display object at right of the screen of specified y, while x ignored |
| x-coordinate | 2 | x-coordinate for text area/button Little endian, range of x-coordinate is [0, 271] |
| y-coordinate | 2 | y-coordinate for text area/button Little endian, range of x-coordinate is [0, 479] |
| features | var. | This field is valid only for the text area or numeric button; for details, see table below This field can be omitted for item button |
| String | var. | Label to show on text area or button ASCII String, terminated by 0x00 Maximum string lens are: <ul style="list-style-type: none"> • Text area is 64 bytes, • Large button is 25 bytes • Medium button is 10 bytes |

| Data Item | Length (bytes) | Description |
|-----------|----------------|--|
| | | <ul style="list-style-type: none"> Filename of image/video <ul style="list-style-type: none"> ASCII string, terminated by 0x00 Maximum string lens is 32 bytes |

Note:

- For images:** image files are read from NANDFlash. Only JPEG files without orientation Exif info and width <= 272, height <= 480 are supported; otherwise the VP6800 rejects this command.
- Exif** (exchangeable image file format): a standard that specifies the formats for images, sound, and ancillary tags used by digital cameras (including smartphones), scanners, and other systems that handle image and sound files recorded by digital cameras.

Text Area Feature:

| Features | Length (bytes) | Description |
|----------|----------------|---|
| Width | 2 | Reserved |
| Height | 2 | Reserved |
| Font ID | 1 | See below table |
| Color | 4 | Byte 0: B Byte 1: G Byte 2: R Byte 3: Reserved For example: 0xFF000000, blue 0x00FF0000, green 0x0000FF00, red 0x00000000, white 0xFFFF0000, black |

LEDs Widget Feature

| Features | Length (bytes) | Description |
|------------|----------------|---|
| LED0 state | 1 | 00: Off 01: Green 02: Yellow 03: Red |
| LED1 state | 1 | |
| LED2 state | 1 | |
| LED3 state | 1 | |

Numeric Button Feature

| Features | Length (bytes) | Description |
|--------------------------|----------------|---|
| Width | 2 | Width of numeric button3 |
| Value | 2 | The initial numeric value |
| Font ID | 1 | See below table |
| Outline Color | 4 | Byte 0: B Byte 1: G Byte 2: R Byte 3: Reserved For example: 0xFF000000, blue 0x00FF0000, green 0x0000FF00, red 0x00000000, white 0xFFFF0000, black |
| Numeric background Color | 4 | |
| Minus background Color | 4 | |
| Plus background Color | 4 | |
| Numeric text Color | 4 | |
| Minus text Color | 4 | |
| Plus text Color | 4 | |
| | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte14-25 | Byte 26 | Byte 27 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | 00h | 0Ch | See below | | |

Data Format

| Item | Length (bytes) | Description |
|----------------|----------------|---|
| Container ID | 2 | Container ID format; see LCD UI Table-2 |
| Object ID | 2 | Object ID format; see LCD UI Table-2 |
| Top-Left x | 2 | top-left x-coordinate of object |
| Top-Left y | 2 | top-left y-coordinate of object |
| Bottom-Right x | 2 | bottom-right x-coordinate of object |
| Bottom-Right y | 2 | bottom-right y-coordinate of object |

19.3.6 Get List of All Objects on Scrollable Container (61-19)

The **Get List of All Objects on Scrollable Container** command retrieves a string list of all created object names on a specified scrollable container.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|----------------------|----------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 19h | | | See Below | | |

Data Field

| Item | Length (bytes) | Description |
|------------------------------|----------------|---|
| Scrollable Container Name | var. | Scrollable Container name string No longer than 32 bytes ASCII string, terminated by 0x00 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-------------|----------------------|----------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | | | See Below | | |

Data Format

| Item | Length(bytes) | Description |
|-------------------|---------------|---|
| Number of Objects | 1 | Number of objects |
| Objects List | var. | List objects, each object is shown as: |
| | | Object ID |
| | | Object Name String |
| | | See Table-2 |
| | | Object name string, ASCII string, each object name string is terminated by 0x00 |

Example:

- **List the objects in "Screen1":**

TX: 56 69 56 4F 74 65 63 68 32 00 61 19 00 15 53 63 72 6F 6C 6C 43
6F 6E 74 61 69 6E 65 72 5F 30 30 30 30 00 53 8D

RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 09 02 08 00 32 00 00 00
31 00 8E EB

19.3.7 Object Event Call Back (61-FF)

The VP6800 immediately sends this command when a button on-click event happens or video playback ends.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14... Byte 13+n | Byte 14+n | Byte 15+n |
|----------------------------------|---------|-------------|----------------------|-------------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | FFh | | | See below | | |

Data Format

| Item | Length (byte) | Description |
|-------------|---------------|---|
| Screen ID | 2 | Screen ID format; see LCD UI Table-1 |
| Object ID | 2 | Object ID format; see LCD UI Table-2 |
| Screen Name | Var. | Screen Name String No longer than 32 bytes ASCII string, terminated by 0x00 |
| Object Name | Var. | Object Name String No longer than 32 bytes ASCII string, terminated by 0x00 |
| Parameter | Var. | This field only exists for invisible button: 00: short press/ 01:long press Numeric Button: current number of numeric button |

Note:

1. The button callback will be sent by previously communicated I/F.
2. Only buttons and numeric entries have call back events.

19.3.8 Remove Object from Screen (61-09)

This command removes an object from a screen. However, the screen does not display removed objects until calling **Show Screen (61-05)** to update the screen.

Command Frames

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-------------|-------------------------|-------------------------|-------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 09h | | | See below | | |

Data Format

| Item | Length(byte) | Description |
|-------------|---------------|---|
| Screen Name | Var. | Screen name string No longer than 32 bytes ASCII string, terminated by 0x00 |
| Object Name | Var. | Object name string No longer than 32 bytes ASCII string, terminated by 0x00 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | 00h | 00h | | |

Example:

- Remove "Text1" from "Screen1":
- TX: 56 69 56 4F 74 65 63 68 32 00 61 09 00 0E 53 63 72 65 65 6E 31 00 54 65 78 74 31 00 3B 5C
- RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81

19.3.9 Create Screen (61-0A)

This command allows the firmware to create a new screen instance based on a screen theme and send out a Screen ID accordingly.

After the screen is created, user can send a **Show Screen (61-05)** command or **Add/Update Object to Screen (61-07/08)** command using the created screen's name.

Screen themes should be pre-defined in firmware. The VP6800 and host device should already have the screen theme IDs.

Currently, Screen IDs are limited to 50 per screen theme.

Command Frames

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ~ N | Byte N+1 | Byte N+2 |
|-------------------------------|---------|-------------|-------------------|-------------------|---|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 0Ah | 00 | Var. | Screen Name String ASCII String, Terminated by 0x00, Maximum string len is 32bytes | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-15 | Byte 16 | Byte 17 |
|-------------------------------|---------|-------------|-------------------|-------------------|------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | 00h | 00h | Screen ID | | |

For example:

- Create Screen "Screen1", return Screen ID 0x0000:
TX: 56 69 56 4F 74 65 63 68 32 00 61 0A 00 08 53 63 72 65 65 6E 31
00 A2 D7
RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 02 00 00 3E FF
- Create Screen "Screen2", return Screen ID 0x0001:
TX: 56 69 56 4F 74 65 63 68 32 00 61 0A 00 08 53 63 72 65 65 6E 32
00 F1 82
RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 02 00 01 2E DE
- Create Screen "Screen3", return Screen ID 0x0001:
TX: 56 69 56 4F 74 65 63 68 32 00 61 0A 00 08 53 63 72 65 65 6E 33
00 C0 B1
RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 02 00 02 1E BD
- Create Screen "Screen1" again, return error code due to duplicated name:
TX: 56 69 56 4F 74 65 63 68 32 00 61 0A 00 08 53 63 72 65 65 6E 31
00 A2 D7
RX: 56 69 56 4F 74 65 63 68 32 00 61 A0 00 00 8C 40

19.3.10 Destroy Screen (61-0B)

This command removes a Screen ID from firmware. After removing the screen, the user cannot operate that Screen ID anymore without getting the Screen ID again by sending a **Create Screen (61-0A)** command.

Note: A Screen ID cannot be removed when it is live.

Command Frames

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ~ N | Byte N+1 | Byte N+2 |
|-------------------------------|---------|-------------|-------------------|-------------------|---|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViV0tech2\0 | 61h | 0Bh | 00 | Var. | Screen Name String ASCII String, Terminated by 0x00 | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | 61h | 00h | 00h | 00h | | |

Example:

- Remove "Screen1":

TX: 56 69 56 4F 74 65 63 68 32 00 61 0B 00 08 53 63 72 65 65 6E 31 00 EB 0F

RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81

- Remove "Screen4", which does not exist:

TX: 56 69 56 4F 74 65 63 68 32 00 61 0B 00 08 53 63 72 65 65 6E 34 00 1E F0

RX: 56 69 56 4F 74 65 63 68 32 00 61 0A 00 3B 52 65 6D 6F 76 65 20 73 63 72 65 65 6E 20 66 72 6F 6D 20 73 74 6F 72 61 67 65 20 66 61 69 6C 2C 20 72 65 6D 6F 76 65 20 73 63 72 65 65 6E 20 69 73 20 72 65 6A 65 63 74 65 64 2E 4C 7A (destroy failed)

19.3.11 Get Current Screen Name (61-0C)

The **Get Current Screen Name** command allows a user to check the name of the currently live screen.

Command Frames

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 0Ch | 00 | 00 | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | | | See Below | | |

Data Field

| Data Item | Length (bytes) | Description |
|-------------|----------------|---|
| Screen ID | 2 | Screen ID format; see LCD UI Table-1 |
| Screen Name | var | Screen name string No longer than 32 bytes ASCII string, terminated by 0x00 |

Example:

- Get Cur Screen ("Screen1", ID=0x0000 is live)

TX: 56 69 56 4F 74 65 63 68 32 00 61 0C 00 00 E0 3E

RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 0A 00 00 53 63 72 65 65 6E 31 00 10 B5

- If the current screen is an ID TECH system screen, an error report is returned.

TX: 56 69 56 4F 74 65 63 68 32 00 61 0C 00 00 E0 3E

RX: 56 69 56 4F 74 65 63 68 32 00 61 0A 00 20 43 75 72 72 65 6E 74 20 73 63 72 65 65 6E 20 69 73 20 73 79 73 74 65 6D 20 73 63 72 65 65 6E 00 B9 53 (Current screen is system screen)

- If the current screen is an ID TECH pop-up screen while a customer screen shown, an error report is returned.

TX: 56 69 56 4F 74 65 63 68 32 00 61 0C 00 00 E0 3E

RX: 56 69 56 4F 74 65 63 68 32 00 61 0A 00 20 43 75 72 72 65 6E 74 20 73 63 72 65 65 6E 20 69 73 20 70 6F 70 2D 75 70 20 73 63 72 65 65 6E 00 05 65 (Current screen is pop-up screen)

19.3.12 Clone Screen (61-0D)

This command creates a new screen instance in the firmware that is cloned from the specified screen and sends out a screen name accordingly.

After the screen is cloned, the user can send a **Show Screen (61-05)** command or **Add/Update Object to Screen (61-07/08)** command using the cloned screen's name.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|----------------------|----------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 0Dh | 00 | var | | | |

Data Format

| Item | Length (byte) | Description |
|-------------|---------------|---|
| Screen Name | Var. | Screen Name String (copy from) No longer than 32 bytes ASCII string, terminated by 0x00 |
| Screen Name | Var. | Screen Name String (New) No longer than 32 bytes ASCII string, terminated by 0x00 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-15 | Byte 16 | Byte 17 |
|-------------------------------------|---------|-------------|----------------------|----------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | 00h | 02h | Screen ID(New) | | |

Example:

- Clone "Screen1" to "Screen3":

TX: 56 69 56 4F 74 65 63 68 32 00 61 0D 00 10 53 63 72 65 65 6E 31
00 53 63 72 65 65 6E 33 00 FA 5E

RX: 56 69 56 4F 74 65 63 68 32 00 61 0A 00 00 8C 40

19.3.13 Store Screen Info (61-0E)

This command stores all current screen information from RAM into flash.

Command Frames

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 0Eh | 00 | 00 | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | 00h | 00h | | |

Example:

- Store Screen Info:

TX: 56 69 56 4F 74 65 63 68 32 00 61 0E 00 00 80 50

RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81

19.3.14 Load Screen Info (61-0F)

This command clears all current screen information in RAM and loads the screen information from flash into RAM.

Command Frames

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 0Fh | 00 | 00 | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | 00h | 00h | | |

Example:

- Load Screen Info:

TX: 56 69 56 4F 74 65 63 68 32 00 61 0F 00 00 B0 67

RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81

19.3.15 Clear Screen Info (61-10)

This command clears all current screen information in RAM and flash, then displays a power-on screen.

Command Frames

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 10h | 00 | 00 | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | 00h | 00h | | |

Example:

- Clear Screen Info:

TX: 56 69 56 4F 74 65 63 68 32 00 61 10 00 00 E2 08

RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81

19.3.16 Get All Screens (61-11)

This command retrieves a string list of names for all created screens.

Command Frames

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 11h | 00 | 00 | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-------------|----------------------|----------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | | | See below | | |

Data Format

| Item | Length(byte) | Description | |
|-------------------|--------------|--|--|
| Number of Screens | 1 | Number of screens | |
| Screen List | var. | List screens, each screen is shown as: | |
| | | Screen ID | See Table-1 |
| | | Screen Name String | Screen name string ASCII string, each screen name string terminated by 0x00 |

Example:

- List screens:

TX: 56 69 56 4F 74 65 63 68 32 00 61 11 00 00 D2 3F

RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 29 04 00 00 53 63 72 65
 65 6E 31 00 00 01 53 63 72 65 65 6E 32 00 00 02 53 63 72 65 65 6E
 33 00 00 03 53 63 72 65 65 6E 34 00 CF AD

The response includes four screens, numbered from 0000~00 03

19.3.17 Get List of All Objects on Screen (61-12)

This command retrieves a string list of all created object names on a specified screen.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|----------------------|----------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 12h | | | See Below | | |

Data Field

| Item | Length (bytes) | Description |
|-------------|----------------|---|
| Screen Name | var. | Screen name string No longer than 32 bytes ASCII string, terminated by 0x00 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-------------|----------------------|----------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | | | See Below | | |

Data Format

| Item | Length(bytes) | Description |
|-------------------|---------------|---|
| Number of Objects | 1 | Number of objects |
| Objects List | var. | List objects, each object is shown as: |
| | | Object ID |
| | | Object Name String |
| | | See Table-2 |
| | | Object name string, ASCII string, each object name string is terminated by 0x00 |

Example:

- List the objects in "Screen1":

TX: 56 69 56 4F 74 65 63 68 32 00 61 12 00 08 53 63 72 65 65 6E 31
00 11 20

RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 38 05 00 00 54 65 78 74
31 00 01 00 42 75 74 74 6F 6E 31 00 02 00 4D 65 64 69 75 6D 20 42
75 74 74 6F 6E 31 00 03 00 49 6E 76 69 73 69 62 6C 65 31 00 06 00
4C 45 44 73 00 CC 6A

19.3.18 Query Screen by Name (61-13)

This command checks if the specified screen exists.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|----------------------|----------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 13h | | | See Below | | |

Data Field

| Item | Length (bytes) | Description |
|-------------|----------------|---|
| Screen Name | var. | Screen name string No longer than 32 bytes ASCII string, terminated by 0x00 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-------------|----------------------|----------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | 00h | 01 | See Below | | |

Data Format

| Item | Length (bytes) | Description |
|--------------|----------------|-----------------------------|
| Exist or Not | 1 | 1: Exist 0: Do not exist |

Example:

Query screen "Screen1"

TX: 56 69 56 4F 74 65 63 68 32 00 61 13 00 08 53 63 72 65 65 6E 31 00 58 F8

RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 01 01 5B BF

Query screen "Screen4", which does not exist:

TX: 56 69 56 4F 74 65 63 68 32 00 61 13 00 08 53 63 72 65 65 6E 34 00 AD 07

RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 01 00 4B 9E

19.3.19 Query Object By Name (61-14)**Command Frame**

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|----------------------|----------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 14h | | | See Below | | |

Data Format

| Item | Length(bytes) | Description |
|--------------------|---------------|---|
| Object Name String | Var. | Object name string No longer than 32 bytes ASCII string, terminated by 0x00 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-------------|----------------------|----------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | 00h | var | See Below | | |

Data Format

| Item | Length(bytes) | Description |
|-------------------|-------------------|--|
| Number of Objects | 1 | |
| Screen Names | Var. | List of screen name strings ASCII String, each screen name terminated by 0x00 |

Example:

Query object "Text1":

TX: 56 69 56 4F 74 65 63 68 32 00 61 14 00 06 54 65 78 74 31 00 63 F6

RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 09 01 53 63 72 65 65 6E 31 00 E0 9A

19.3.20 Query Screen by ID (61-15)**Command Frame**

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 Byte 15 | Byte 16 | Byte 17 |
|-------------------------------------|---------|-----------------|----------------------|----------------------|-----------------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 15h | 00 | 02 | Screen ID (See LCD UI Table-1) | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-------------|----------------------|----------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | | | See Below | | |

Data Format

| Item | Length (bytes) | Description |
|--------------|----------------|--|
| Exist or Not | 1 | 1: Exist 0: Do not exist |
| Screen Name | Var. | Screen name strings if the screen exists ASCII string, terminated by 0x00 |

Example:

- Query Screen 00 00:

TX: 56 69 56 4F 74 65 63 68 32 00 61 15 00 02 00 00 F2 19

RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 09 01 53 63 72 65 65 6E
31 00 E0 9A

- Query Screen 00 04 (error):

TX: 56 69 56 4F 74 65 63 68 32 00 61 15 00 02 00 04 76 59

RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 01 00 4B 9E

19.3.21 Query Object by ID (61-16)

This command checks by ID if the specified object exists.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 Byte 15 | Byte 16 | Byte 17 |
|-------------------------------------|---------|-----------------|----------------------|----------------------|-----------------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 16h | 00 | 02 | Object ID (See LCD UI Table-1) | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ~ N | Byte N+1 | Byte N+2 |
|----------------------------------|---------|-------------|----------------------|-------------------------|----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | 00h | var | See Below | | |

Data Format

| Item | Length (bytes) | Description |
|-------------------|----------------|--|
| Number of Objects | 1 | |
| Screen Names | Var. | List Screen Name Strings ASCII String, each screen name string is terminated by 0x00 |

Example:

- QueryObjectID

TX: 56 69 56 4F 74 65 63 68 32 00 61 16 00 02 01 00 11 C4

RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 09 01 53 63 72 65 65 6E
31 00 E0 9A

19.3.22 Display system screen (61-17)

This command hides the customer UI and shows the L2 UI.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 17h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | See status code table | 00h | 00 | | |

Example:

- Display system screen:

TX: 56 69 56 4F 74 65 63 68 32 00 61 17 00 00 72 8D

RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81

19.3.23 Turn On or Off Visual LEDs on LCD (61-1A)

The **Turn On or Off Visual LEDs on LCD** command turns the Visual LEDs on a VP6800's LCD screen when displaying the IDTECH L2 UI. The reader retains this setting even after a device reset.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | On/Off | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 1Ah | 00h | 01h | 0: off 1: on | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 83h | See status code table | 00h | 00h | | |

Example:

- Disable Visual LEDs:

TX: 56 69 56 4F 74 65 63 68 32 00 61 1A 00 01 00 38 92

RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81

19.3.24 Link UI with Transaction Message ID (61-1B)

The **Link UI with Transaction Message ID** command links an exist Customer UI ID to a specified a transaction message ID. The linked System UI is replaced by the linked Customer UI during transactions.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-13+n | Byte 14+n+m | Byte 15+n+m |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------|-------------|-------------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 1Bh | 00h | | See Below | | |

Data Field

| Data Item | Length (bytes) | Description |
|------------------------|----------------|---|
| Transaction Message ID | 1 | Transaction Message ID Refer to <i>EMV Display Message ID Assignment.pdf</i> |
| Screen Name | var | Screen Name String No longer than 32 bytes ASCII string, terminated by 0x00 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | 00h | 00h | 00 | | |

19.3.25 Get RT1050 Firmware Version (29-80)

This command retrieves the reader's ViVOpay firmware version number. The reader returns a response frame containing the ViVOpay firmware version information.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 29h | 80h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 29h | See status code table | 00h | String length | Version string | | |

19.3.26 Set Backlight (83-2F)

This command sets the VP6800's backlight level. If the level is set to >100%, the command will be rejected; if the level is set to <10%, backlight level will be set to 10%.

Command Frames

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14- Byte17 | Byte 18 | Byte 19 |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | '83' | '2F' | '00' | '04' | See below | | |

Data Format

| Item | Length (bytes) | Description |
|------------|----------------|-------------------------------------|
| Status | 1 byte | 0: backlight off 1: backlight on |
| Brightness | 1 byte | 0~100 |
| Reserved | 2 bytes | Fixed 0x00, 0x00 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 83h | 00h | 00h | 00h | | |

Example:

- Set Backlight 100:

TX: 56 69 56 4F 74 65 63 68 32 00 83 2F 00 04 01 32 00 00 1E 33

RX: 56 69 56 4F 74 65 63 68 32 00 83 00 00 00 22 03

19.3.27 Delete File (83-1F)

This command deletes a data file from the reader.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Bytes 14-13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|---------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 83h | 1Fh | 00h | Variable | | | |

Command Data

The data for this command is a zero-terminated ASCII string with the name of the file to be deleted.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Bytes 14 – 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Error | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 83h | See status code table | 00h | 0 – n | | | |

If the command returns an error status, the data portion may contain a 4-byte extended status giving more detailed error information (in which case, see the Appendix on [Extended Status Codes](#)).

If the data length is non-zero, a descriptive, zero-terminated ASCII error string is contained in the data portion of the response.

Example:

- Delete testfile.x:

TX: 56 69 56 4F 74 65 63 68 32 00 83 1F 00 0B 74 65 73 74 66 69 6C
65 2E 78 00 75 CC

RX: 56 69 56 4F 74 65 63 68 32 00 83 00 00 00 22 03

19.3.28 Transfer File (83-24)

This command transfers a data file to the reader. The maximum size of one file is 64KB; the maximum number of image files is 256.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-13+n | Byte 14+n+m | Byte 15+n+m |
|-------------------------------|---------|-------------|-------------------|-------------------|------------------------|-------------|-------------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | parameters (see below) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 83h | 24h | 00h | n | | | |

The table below defines the file transfer command parameters. If a file is larger than 1000 bytes, it must be sent in sections of 1000 bytes or less. Each section must be transmitted in sequential order, and the last section must be flagged to indicate the file transfer is complete.

The file size is only used to check the available space before the transfer. Sending an incorrect file size may result in an error during the download.

| Length (bytes) | Description |
|----------------|---|
| 2-128 | File name including subdirectory, zero-terminated ASCII |
| 1-9 | Number of total file data bytes, zero-terminated ASCII |
| 2 | Flag to indicate the end points of file transmission, zero-terminated ASCII 0 = Neither first or last packet 1 = Last packet: commits all file transfers received since packet with first packet flag. 2 = First packet: all incomplete transfers are destroyed. 3 = First and Last packet: set for one packet <= 1000 bytes. |
| Variable | Binary file contents up to 1000 bytes |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|--------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Response | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 83h | See status code table | 00h | Variable | | | |

If the command returns an error status, the data portion may contain a 4-byte extended status giving more detailed error information (in which case, see the Appendix on [Extended Status Codes](#)).

A successful response contains a Status Code of zero and a data length of zero.

Example:

- Download a 20-byte file named **testfile.x**. Each command contains 10 bytes of the file contents. Set the flag to ASCII 2 to indicate first segment of file:

```
TX: 56 69 56 4F 74 65 63 68 32 00 83 24 00 1A 74 65 73 74 66 69 6C
    65 2E 78 00 32 30 00 32 00 61 61 61 61 61 61 61 61 62 32 B3
RX: 56 69 56 4F 74 65 63 68 32 00 83 00 00 00 0A 70
```

19.3.29 List Directory (83-22)

This command retrieves a directory listing of user-accessible files from the reader.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 83h | 22h | 00h | variable | | | |

Command Data

The command data is a zero-terminated ASCII string specifying the directory and contents to be listed. To retrieve all sub-directories and files below the specified directory, add the recursive directory option before the directory name.

| Length (bytes) | Description |
|----------------|---|
| 3 | Zero-terminated ASCII hex flags Allowed values: 30 30 00 – List top directory files only 31 30 00 – Recursively list all files and directories below the top directory |
| 2-128 | Zero-terminated directory name |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Bytes 14-13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Listing Response String | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 83h | See status code table | | | | | |

| Length (bytes) | Description |
|----------------|---|
| 1-9 | Number of total file data bytes, zero-terminated ASCII |
| 2 | Flag to indicate the end points of file transmission, zero-terminated ASCII 0 = Neither first or last packet 1 = Last packet: commits all file transfers received since packet with first packet flag. 2 = First packet: all incomplete transfers are destroyed. 3 = First and Last packet: set for one packet <= 4000 bytes. |
| Variable | the requested directory list as a zero-terminated ASCII string containing the file names. The files are listed sequentially separated by commas. Subdirectories are shown with a slash (/) suffix. |

Example 1:

No File

List the contents of the root custom directory, recursive flag is set:

56 69 56 4F 74 65 63 68 32 00 83 22 00 04 31 30 00 00 9D 9B

Response:

56 69 56 4F 74 65 63 68 32 00 83 00 00 04 30 00 33 00 76 5F

Example 2:

List the contents of the root custom directory, recursive flag is set:

56 69 56 4F 74 65 63 68 32 00 83 22 00 04 31 30 00 00 9D 9B

Response:

SCR000.JPG, SCR100.JPG

56 69 56 4F 74 65 63 68 32 00 83 00 00 1B 32 32 00 33 00 53 43 52 30
30 30 2E 4A 50 47 2C 53 43 52 31 30 30 2E 4A 50 47 00 47 60

19.3.30 Delete Directory (83-26)

This command deletes a directory. The directory must be empty for this command to succeed.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Bytes 14-13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|---------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 83h | 26h | 00 | Variable | | | |

Command Data

The data for this command is a zero-terminated ASCII string with the complete path and directory name to delete. The command does not require the root directory to be specified. Indicate subdirectories with a forward slash (/). The last directory in the string is deleted.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Bytes 14 – 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Extended Status Code | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 83h | See status code table | 00h | 00h or 04h | | | |

If the response status code is not Success (00h), this command may return a 4-byte extended status code in the data portion of the response (see [Extended Status Codes](#)).

Example:

- Delete directory **dork** from the custom root directory:

TX: 56 69 56 4F 74 65 63 68 32 00 83 26 00 05 64 6F 72 6B 00 17 13

RX: 56 69 56 4F 74 65 63 68 32 00 83 00 00 00 22 03

19.3.31 Enable/Disable Light Sensor (83-30)

The **Enable/Disable Light Sensor** command allows the user enable or disable the light sensor. If disabled, the backlight does not adjust automatically.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 83h | 30h | 00h | 01h | 0: disable 1: enable | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 83h | See status code table | 00h | 00 | | |

19.3.32 Query File (83-32)

The **Query File** command checks if the specified file exists in NAND Flash.

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | 83h | 32h | variable | variable | See below | | |

Data Format

| Item | Length (bytes) | Description |
|-----------------------|----------------|--|
| Directory Name String | Var. | Directory name string No longer than 32 bytes ASCII string, terminated by 0x00 |
| File Name String | Var. | File name string No longer than 32 bytes ASCII string, terminated by 0x00 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (MSB) |
| ViVOtech2\0 | 83h | 00h | 00h | var | See Below | | |

Data Format

| Item | Length (bytes) | Description |
|--------------------------|----------------|--------------------------------|
| Exists or Does not exist | 1 | 1: Exists 0: Does not exist |
| Latest Time stamp | 0 or 6 | BCD code |
| File Size | 0 or var | Zero-terminated ASCII string |

Example:

Query for file "01img.jpg", which does not exist:

Exists, latest time stamp: November 25, 19 at 8:50 minutes and 53 seconds, file size is 28513 bytes.

TX: 56 69 56 4F 74 65 63 68 32 00 83 32 00 0B 00 30 31 69 6D 67 2E 6A 70 67 00 1C 59

RX: 56 69 56 4F 74 65 63 68 32 00 83 00 00 0D 01 19 11 25 08 50 53 32 38 35 31 33 00 9B 07

19.3.33 Transfer File to SD Card (83-33)

The **Transfer File to SD Card** command transfers a data file to the reader's SD card.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14- 13+n | Byte 14+n+m | Byte 15+n+m |
|-------------------------------|---------|-------------|-------------------|-------------------|------------------------|-------------|-------------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Parameters (see below) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 83h | 33h | 00h | n | | | |

The table below defines the file transfer command parameters. File larger than 1000 bytes must be sent in sections of 1000 bytes or less. Each section must be transmitted in sequential order, and the last section must be flagged to indicate the file transfer is complete.

The file size is only used to check the available space before the transfer. Sending an incorrect file size may result in an error during the download.

Data Field

| Length (bytes) | Description |
|----------------|---|
| 2-128 | File name including subdirectory, zero-terminated ASCII |
| 1-9 | Number of total file data bytes, zero-terminated ASCII |
| 2 | Flag to indicate the end points of file transmission, zero-terminated ASCII 0 = Neither first or last packet 1 = Last packet: commits all file transfers received since packet with first packet flag. 2 = First packet: all incomplete transfers are destroyed. 3 = First and Last packet: set for one packet <= 1000 bytes. |
| Variable | Binary file contents up to 1000 bytes |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|--------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Response | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 83h | See status code table | 00h | Variable | | | |

If the command returns an error status, the data portion may contain a 4-byte extended status giving more detailed error information (in which case, see the Appendix on [Extended Status Codes](#)).

A successful response contains a status code of zero and a data length of zero.

Example:

- The download in this example is a 20-byte file named **testfile.x**. Each command contains 10 bytes of the file contents. Set the flag to **ASCII 2** to indicate first segment of file:

TX: 56 69 56 4F 74 65 63 68 32 00 83 33 00 1A 74 65 73 74 66 69 6C
65 2E 78 00 31 30 00 33 00 61 61 61 61 61 61 61 61 61 62 5E 1D

RX: 56 69 56 4F 74 65 63 68 32 00 83 00 00 00 22 03

19.3.34 Delete File from SD Card (83-34)

The **Delete File from SD Card** command deletes a data file from the reader's SD card.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Bytes 14-13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|---------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 83h | 34h | 00h | Variable | | | |

Command Data

The data for this command is a zero-terminated ASCII string with the name of the file to delete.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Bytes 14 – 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Error | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 83h | See status code table | 00h | 0 – n | | | |

If the command returns an error status, the data portion may contain a 4-byte extended status giving more detailed error information (in which case, see the Appendix on [Extended Status Codes](#)).

If the data length is non-zero, the data portion of the response contains a descriptive, zero-terminated ASCII error string.

Example:

- Delete **testfile.x**:

TX: 56 69 56 4F 74 65 63 68 32 00 83 34 00 0B 74 65 73 74 66 69 6C
65 2E 78 00 37 72

RX: 56 69 56 4F 74 65 63 68 32 00 83 00 00 00 22 03

19.3.35 List Directory on SD Card (83-35)

The **List Directory on SD Card** command retrieves a directory listing of user-accessible files from the reader's SD card.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 83h | 35h | 00h | variable | | | |

Command Data

The command data is a zero-terminated ASCII string specifying the directory and contents to be listed. To retrieve all sub-directories and files below the specified directory, add the recursive directory option before the directory name.

| Length (bytes) | Description |
|----------------|---|
| 3 | Zero-terminated ASCII hex flags Allowed values: 30 30 00 – List top directory files only 31 30 00 – Recursively list all files and directories below the top directory |
| 2-128 | Zero-terminated directory name |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Bytes 14-13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Listing Response String | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 83h | See status code table | | | | | |

If the command returns an error status, the data portion may contain a 4-byte extended status giving more detailed error information (in which case, see the Appendix on [Extended Status Codes](#)).

A successful response contains a status code of zero and the requested directory list as a zero-terminated ASCII string containing the file names and file sizes in bytes. The files are listed sequentially separated by commas. The file size of each file immediately follows the file name and is separated by a colon. Subdirectories are shown with a slash (/) suffix.

Example 1:

List the contents of the "folder" directory, recursive flag is set:

TX: 56 69 56 4F 74 65 63 68 32 00 83 35 00 0A 31 30 00 66 6F 6C 64 65 72 00 12 60

The response is three files (afile.txt, bfile.txt, cfile.txt):

RX: 56 69 56 4F 74 65 63 68 32 00 83 00 00 38 35 31 00 33 00 66 6F 6C 64 65 72 2F 61 66 69 6C 65 2E 74 78 74 2C 66 6F 6C 64 65 72 2F 62 66 69 6C 65 2E 74 78 74 2C 66 6F 6C 64 65 72 2F 63 66 69 6C 65 2E 74 78 74 00 40 0D

Example 2:

List the contents of the root custom directory, recursive flag is clear:

TX: 56 69 56 4F 74 65 63 68 32 00 83 35 00 0A 30 30 00 66 6F 6C 64 65 72 00 7D 25

The response is a 0 file:

RX: 56 69 56 4F 74 65 63 68 32 00 83 00 00 04 30 00 33 00 76 5F

19.3.36 Query File on SD Card (83-36)

The **Query File on SD Card** command checks if the specified file exists in the reader's SD card.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | 83h | 36h | variable | variable | See below | | |

Data Format

| Item | Length (bytes) | Description |
|-----------------------|----------------|--|
| Directory Name String | Var. | Directory name string No longer than 32 bytes ASCII string, terminated by 0x00 |
| File Name String | Var. | File name string No longer than 32 bytes ASCII string, terminated by 0x00 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 83h | 00h | 00h | var | See Below | | |

Data Format

| Item | Length (bytes) | Description |
|--------------------------|----------------|--------------------------------|
| Exists or Does not exist | 1 | 1: Exists 0: Does not exist |
| Latest Time stamp | 0 or 6 | BCD code |
| File Size | 0 or var | Zero-terminated ASCII string |

Example:

Query file **01img.jpg**, which does not exist:

Exists, latest time stamp: November 25, 19 at 8:50 minutes and 53 seconds, file size is 28513 bytes.

TX: 56 69 56 4F 74 65 63 68 32 00 83 36 00 0B 00 30 31 69 6D 67 2E 6A 70 67 00 E1 FC

RX: 56 69 56 4F 74 65 63 68 32 00 83 00 00 0D 01 19 11 25 08 50 53 32 38 35 31 33 00 9B 07

19.3.37 Get SD Card State (83-37)

The **Get SD Card State** command retrieves a VP6800's SD card state.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 83h | 37h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|------------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | SD Card State | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | 83h | See status code table | 00h | 01h | 0: Not Ready yet 1: Ready | | |

Example:

- Get SD card State (Not Ready yet)

TX: 56 69 56 4F 74 65 63 68 32 00 83 37 00 00 62 36

RX: 56 69 56 4F 74 65 63 68 32 00 83 00 00 01 00 34 11

19.3.38 Read Out File from SD Card (83-39)

The **Read Out File from SD Card** command retrieves a specified file data from the VP6800's SD card.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte14-13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|-------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViV0payV2\0 | 83h | 39h | variable | variable | See below | | |

Data Format

| Item | Length (bytes) | Description |
|-----------------------|----------------|--|
| Directory Name String | Var. | Directory name string No longer than 32 bytes ASCII string, terminated by 0x00 |
| File Name String | Var. | File name string No longer than 32 bytes ASCII string, terminated by 0x00 |
| Starting offset | Var. | Starting offset in the file to retrieve No longer than 9 bytes ASCII string, terminated by 0x00 |
| Number of Bytes | Var. | Number of bytes of file data to retrieve No longer than 9 bytes ASCII string, terminated by 0x00 |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Bytes 14-13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Listing Response String | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 83h | See status code table | | | | | |

| Length (bytes) | Description |
|----------------|---|
| 1-9 | Number of total file data bytes, zero-terminated ASCII |
| 2 | Flag to indicate the end points of file transmission, zero-terminated ASCII 0 = Neither first or last packet 1 = Last packet: commits all file transfers received since packet with first packet flag. 2 = First packet: all incomplete transfers are destroyed. 3 = First and Last packet: set for one packet <= 4000 bytes. |
| Variable | The file data |

19.3.39 Get Light Sensor Value (83-50)

This command retrieves the light sensor detected value.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 83h | 50h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Bytes 14...13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Light Value | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 83h | See status code table | 00h | 02 | lower byte first | | |

Example:

- Get Light Sensor:

TX: 56 69 56 4F 74 65 63 68 32 00 83 50 00 00 CD 7C

RX: 56 69 56 4F 74 65 63 68 32 00 83 00 00 02 E1 01 0D 25

19.3.40 Get External Chip Status (83-51)

This command retrieves the RT1050 status.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 83h | 51h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Bytes 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-------------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 83h | See status code table | 00h | 01 | 1050 current mode (see below) | | |

Data Format

| Item | Length(byte) | Description |
|-------------------|--------------|--|
| 1050 current mode | 1 | 0: No Reader Mode 1: Restricted Mode 2: Self Check Mode 3: Customer Mode 4: System Mode 5: Boot Loader Mode |

Example:

- Get RT1050 Status(customer mode):

TX: 56 69 56 4F 74 65 63 68 32 00 83 51 00 00 FD 4B

RX: 56 69 56 4F 74 65 63 68 32 00 83 00 00 01 03 04 72

The highlighted value indicates the RT1050 is currently in customer mode.

19.3.41 Get Battery Level (F0-02)

This command retrieves the current battery level.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | See Below | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | F0h | 02h | | | | | |

Data Format

| Item | Length (bytes) | Description |
|-----------------------------------|----------------|--|
| Get Li Battery Voltage Percentage | 0 | |
| Get Battery Voltage Value | 1 | 0x00: Coin Battery Voltage Value 0x01: Li Battery Voltage Value |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | F0h | See status code table | | | See Below | | |

Response Data Format

| Item | Length (bytes) | Description |
|----------------------------|----------------|-------------------------------|
| Li Battery Percentage | 1 | 0x05 ~ 0x64 (5% ~ 100%) |
| Coin Battery Voltage Value | 2 | Low byte is first, unit is mV |
| Li Battery Voltage Value | 2 | Low byte is first, unit is mV |

19.3.42 Get MAC Address (D1-1B)

This command gets the VP6800's MAC address.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | D1h | 1Bh | 00h | 01h | Interface (see below) | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Bytes 14 – 19 | Byte 20 | Byte 21 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|---------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | D1h | See status code table | 00h | 06 | MAC Address | | |

Data Format

| Item | Length(byte) | Description |
|-----------|--------------|----------------|
| Interface | 1 | 0x30: Ethernet |

Example:

- Get MAC Address (00:11:22:33:CD:AB):
TX: 56 69 56 4F 74 65 63 68 32 00 D1 1B 00 01 00 60 18
RX: 56 69 56 4F 74 65 63 68 32 00 D1 00 00 06 00 11 22 33 CD AB 45 E8

19.3.43 Append Detailed Information (4C-47)

This command sets how the VP6800 handles log files.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Bytes 14-13+n | Byte 14+n | Byte 15+n |
|-----------------------|---------|-------------|------------|------------|---------------|-----------|-----------|
| Header Tag & Protocol | Command | Sub Command | Length MSB | Length LSB | Data | CRC (LSB) | CRC (MSB) |
| ViVotech2\0 | 4Ch | 47h | 00h | 01h | see below | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-----------------------|---------|---------------------------------------|------------|------------|-----------|-----------|-----------|
| Header Tag & Protocol | Command | Status | Length MSB | Length LSB | Data | CRC (LSB) | CRC (MSB) |
| ViVotech2\0 | 4Ch | See status code table | 00h | 00h | See Below | | |

Data Format

Note: the following function codes enable or disable the log function or the error detail log function. Reader operations and errors are logged only when these functions are enabled.

| Function | Length (bytes) | Test Command Data Field Description | Comment |
|--|----------------|-------------------------------------|--|
| Read Log | 1 | 00h | To enable log file function, an SD Card must be inserted . Log file function is disabled when the unit powers on; use parameter 11h to enable logging. |
| Clear Log | 1 | 01h | |
| Disable log function | 1 | 10h | |
| Enable log function | 1 | 11h | |
| Enable adding detail after error code | 1 | 21h | This setting will be available even reset reader |
| Disable adding detail after error code | 1 | 20h | |

19.3.44 Start Removal Detection Activation Sequence (61-20)

This command only works on production readers and starts the dual control removal detection activation sequence.

Note: the default password1 is **12345678** and password2 is **87654321**.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOTech2\0 | 61h | 20h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 61h | See status code table | 00h | 00 | | |

19.3.45 Reserved for VP6800 Use (61-21)

Command 61-21 is reserved and used to distinguish between different customizations.

19.3.46 Start Screen Saver (61-50)

The command starts the screen saver, which is disabled when the touchpad is touched, the customize screen is shown, or a transaction is started. The reader reads the video from the inserted SD Card; the video format must be MJPEG with a maximum frame width of 480px and maximum frame height of 272px. Note that the video displays rotated +90 degrees.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Bytes 14... 13+n | Byte 14+n | Byte 15+n |
|-----------------------|---------|-------------|------------|------------|------------------|-----------|-----------|
| Header Tag & Protocol | Command | Sub Command | Length MSB | Length LSB | Data | CRC (LSB) | CRC (MSB) |
| ViVOTech2/0 | 61h | 50h | | | see below | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-----------------------|---------|---------------------------------------|------------|------------|-----------|-----------|
| Header Tag & Protocol | Command | Status | Length MSB | Length LSB | CRC (MSB) | CRC (LSB) |
| ViVOTech2/0 | 4Ch | See status code table | 00h | 00h | | |

Data Format

| Item | Length(byte) | Data Field Description |
|-----------------|--------------|---|
| Video File Name | Variable | Video name string (no longer than 30 bytes) ASCII string, terminated by 0x00 |

Example:

TX: 56 69 56 4F 74 65 63 68 32 00 61 50 00 08 6E 66 63 2E 61 76 69 00 FE A8 (play the file "nfc.avi" from the SD card)

RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81 (request successful, the video plays on the screen)

TX: 56 69 56 4F 74 65 63 68 32 00 61 50 00 08 6E 67 63 2E 61 76 69 00 9F 10 (play the file "nfc.avi" from the SD card)

RX: 56 69 56 4F 74 65 63 68 32 00 61 0A 00 00 8C 40 (error, the file "ngc.avi" does not exist)

19.3.47 Stop Audio (61-30)

The **Stop Audio** command stops the audio the reader is playing.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 30h | 00h | 00h | | |

19.3.48 Play Audio (61-31)

The **Play Audio** command plays an audio file loaded from the inserted SD card. The VP6800 supports 16bit PCM format .WAV files.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 31h | | | Data Objects | | |

Data Objects

| Data Item | Length (bytes) | Description |
|------------------|----------------|---|
| File Name | Variable | ASCII String, Null Terminated, Maximum file name length is 30 |
| File Search from | 1 | 0: Flash (the Maximum audio file in Flash is 5M and only 2 audio file is supported) 1: SD Card |

19.3.49 Set/Get Audio Volume (61-32)

The **Set/Get Audio Volume** command sets and retrieves the reader's audio volume.

Command Frame (Set Volume)

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOPayV2\0 | 61h | 32h | 00h | 01h | See below | | |

Data Format

| Item | Length (bytes) | Description |
|--------------------|----------------|---|
| Set Volume Percent | 1 | 0~20; other values will be rejected. Each integer increases volume by 5%. |

Response Frame (Set Volume)

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 61h | See status code table | 00h | 00h | | |

Command Frame (Get Volume)

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOPayV2\0 | 61h | 32h | 00h | 00h | | |

Response Frame (Get Volume)

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOTech2\0 | 61h | See status code table | 00h | 01h | See Below | | |

Data Format

| Item | Length (bytes) | Description |
|--------------------|----------------|--|
| Get Volume Percent | 1 | 0~20; each integer increases volume by 5% (0~100%) |

19.3.50 Set Cancel Button On/Off (61-33)

The **Set Cancel Button On/Off** command determines whether or not to display the cancel button during transactions.

Note: The value returns to the default OFF setting each time the reader boots up.

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | 61h | 33h | 00h | 01 | See below | | |

Data Format

| Item | Length (bytes) | Description |
|----------------------|----------------|-----------------|
| Cancel Button On/Off | 1 | 0: Off 1: On |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | See status code table | 00h | 00h | | |

19.3.51 Get Cancel Button On/Off (61-34)

The **Get Cancel Button On/Off** command checks during a transaction for the cancel button setting from the **Set Cancel Button On/Off (61-33)** command.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | 61h | 34h | 00h | 00 | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOtech2\0 | 61h | See status code table | 00h | Variable | See Below | | |

Response Data

| Item | Length (bytes) | Description |
|----------------------|----------------|-----------------|
| Cancel Button On/Off | 1 | 0: Off 1: On |

19.3.52 QR Code Scan Start (61-41)

This command is only available for a VP6800 with a camera (PCI Standard version).

After the camera is turned on and then off, the original screen (shown on-screen before the camera is turned on) will be lost. Call 61-xx commands to change the screen.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-----------------------|---------|-------------|------------|------------|-----------------------|-----------|-----------|
| Header Tag & Protocol | Command | Sub Command | Length MSB | Length LSB | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2/0 | 61h | 41h | 00h | Variable | See below | Varies | Varies |

Data Format

| Data Item | Length (bytes) | Description | | | | | | | | | | | | | | | |
|-------------------------------|----------------|---|-----------|--------|-------------|--------------|---|--|--------------|---|--|-------|---|--|--------|---|--|
| Length of Time Out | 2 | 0 or 2 | | | | | | | | | | | | | | | |
| Time Out | 0 or 2 | Timeout in seconds. <ul style="list-style-type: none"> 0 – Default 30s 2 – Data Value Range 30s~65536s | | | | | | | | | | | | | | | |
| Length of Display Window Info | 2 | 0 or 8 | | | | | | | | | | | | | | | |
| Display Window Info | 0 or 8 | Camera UI Display Window Info <ul style="list-style-type: none"> 0 – Default x = 8, y = 110, width = 256, height = 256 8 bytes detail: <table> <tr> <th>Data Item</th><th>Length</th><th>See Table-2</th></tr> <tr> <td>x-coordinate</td><td>2</td><td>Little endian, range of x-coordinate is [0, 271]</td></tr> <tr> <td>y-coordinate</td><td>2</td><td>Little endian, range of x-coordinate is [0, 479]</td></tr> <tr> <td>Width</td><td>2</td><td>Little endian, range of x-coordinate is [0, 271]</td></tr> <tr> <td>height</td><td>2</td><td>Little endian, range of x-coordinate is [0, 479]</td></tr> </table> | Data Item | Length | See Table-2 | x-coordinate | 2 | Little endian, range of x-coordinate is [0, 271] | y-coordinate | 2 | Little endian, range of x-coordinate is [0, 479] | Width | 2 | Little endian, range of x-coordinate is [0, 271] | height | 2 | Little endian, range of x-coordinate is [0, 479] |
| Data Item | Length | See Table-2 | | | | | | | | | | | | | | | |
| x-coordinate | 2 | Little endian, range of x-coordinate is [0, 271] | | | | | | | | | | | | | | | |
| y-coordinate | 2 | Little endian, range of x-coordinate is [0, 479] | | | | | | | | | | | | | | | |
| Width | 2 | Little endian, range of x-coordinate is [0, 271] | | | | | | | | | | | | | | | |
| height | 2 | Little endian, range of x-coordinate is [0, 479] | | | | | | | | | | | | | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|----------------------|----------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOpayV3\0 | 61h | See status code table | | | | | |

This command uses the secondary response mechanism. The reader sends a first ACK response immediately after it receives the command; it then sends a second response after the QR Code is completed read, or timed out, or it receives a **QR Code Scan Stop** command.

During the operation (that is, the interval between the first and second response), the reader only accepts the **QR Code Scan Stop** command, which will cancel the current **QR Code Scan Start** operation.

The second response is the scanned QR code's data after the reader completes reading it.

Example:

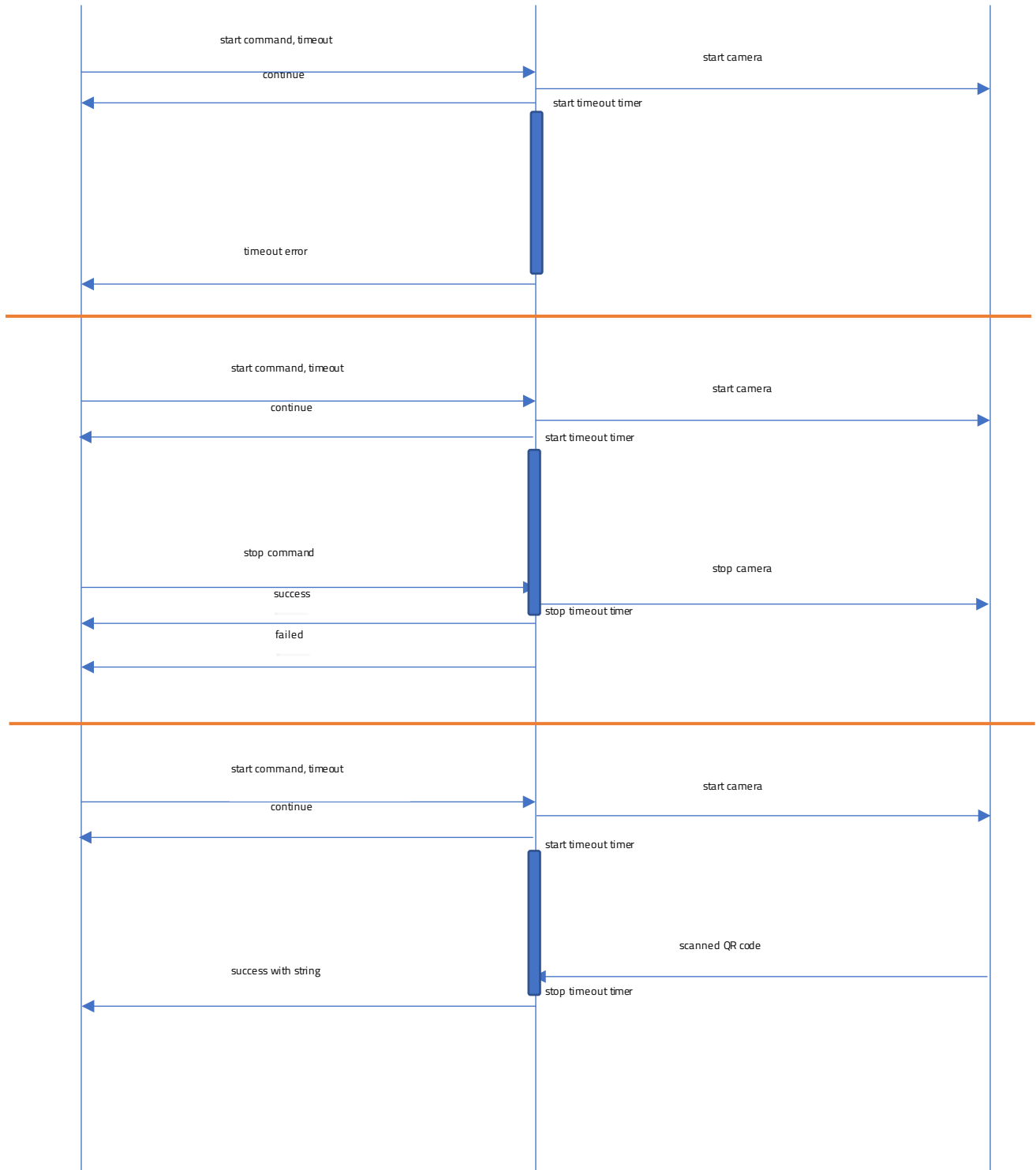
- TX: 56 69 56 4F 74 65 63 68 32 00 61 41 00 04 00 00 00 00 90 62
(default setting)
- RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81 – First response
- RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 07 31 32 33 34 35 0A 0A
E1 8B – Second response

1.1.3.18. QR Code Reading Use Cases

USDK

FW

Camera Module



19.3.53 QR Code Scan Stop (61-42)

This command is only available for a VP6800 with a camera (PCI Standard version).

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-----------------------|---------|-------------|------------|------------|-----------|-----------|
| Header Tag & Protocol | Command | Sub Command | Length MSB | Length LSB | CRC (LSB) | CRC (MSB) |
| ViVOtech2/0 | 61h | 42h | 00h | 00h | Varies | Varies |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOPayV3\0 | 61h | See status code table | 01h | 00h | See below | | |

19.3.54 Camera Take Photo Start (61-43)

The **Camera Take Photo Start** command is only available for a VP6800 with a camera (PCI Standard version).

After the camera is turned on and then off, the original screen (shown on-screen before the camera is turned on) will be lost. Call 61-xx commands to change the screen.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-----------------------|---------|-------------|------------|------------|--------------------------|-----------|-----------|
| Header Tag & Protocol | Command | Sub Command | Length MSB | Length LSB | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2/0 | 61h | 43h | 00h | Variable | See below | Varies | Varies |

| Data Item | Length (bytes) | Description |
|-----------|----------------|---|
| Time Out | 0 or 2 | Unit is second. 0 – Default 30s 2 – Data Value Range 30s~65536s |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|----------------------|----------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | 61h | See status code table | | | | | |

This command uses the secondary response mechanism. The reader sends a first ACK response immediately after it receives the command; it then sends a second response after the operation times out or it receives a **Camera Take Photo Stop** command.

During the operation (that is, the interval between the first and second response), the reader only accepts the **Camera Take Photo Stop** command, which will cancel the current **Camera Take Photo Start** operation.

The reader rejects the **Camera Take Photo Start** command if an SD card does not exist.

Example:

- TX: 56 69 56 4F 74 65 63 68 32 00 61 40 00 01 01 35 23
- RX: 56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81 – First response
- RX: 56 69 56 4F 74 65 63 68 32 00 61 08 24 46 – Second response (Timeout)

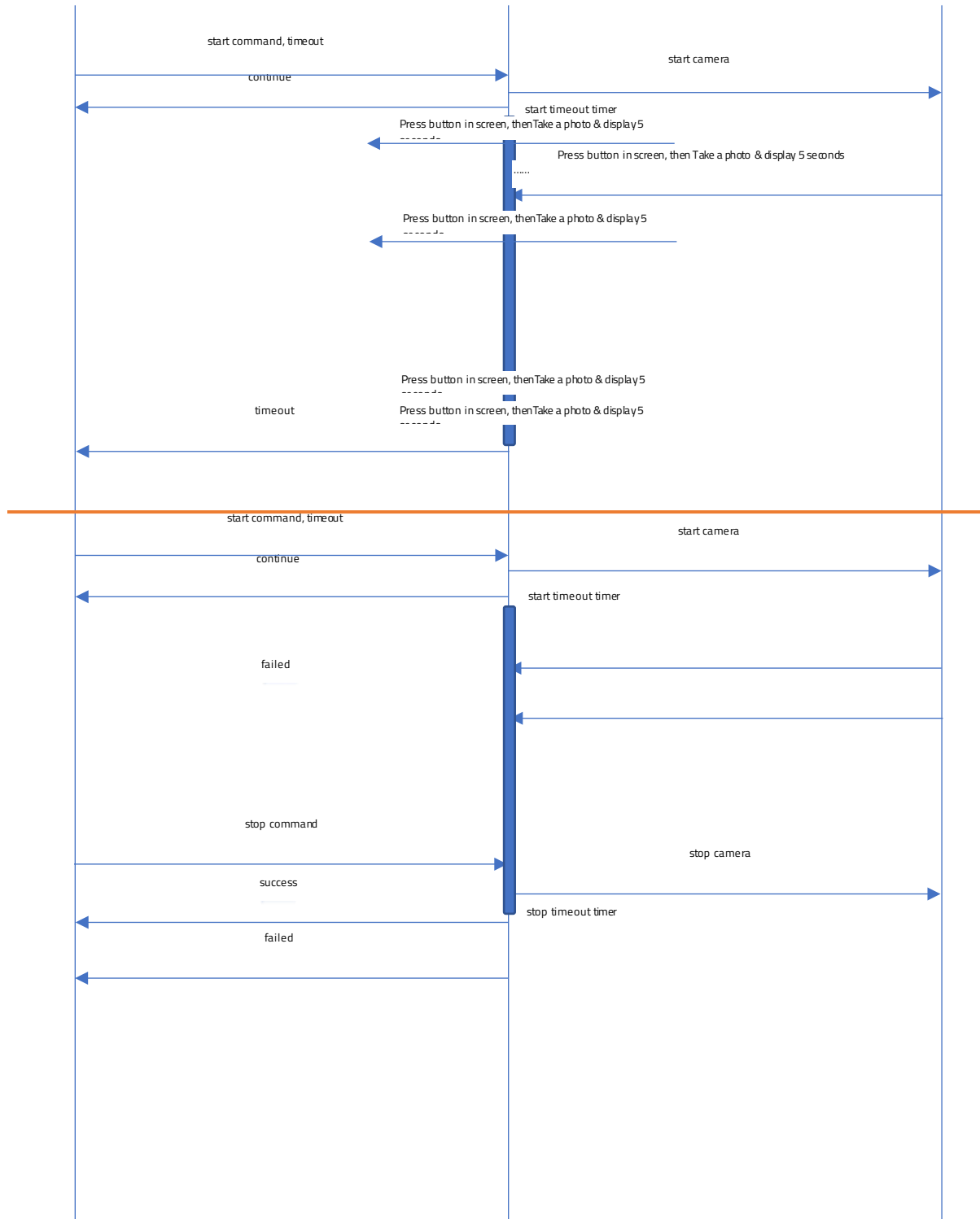
When the camera is turned on, a “take a picture” button appears on the display. If the button is pressed, the reader takes a picture, which is stored in SD card and displayed on-screen for five seconds. The reader stores a maximum of 20 pictures. Upon reaching capacity, the 21st picture replaces the 1st.

1.1.3.19. Use Cases for Taking Photos with the Camera

USDK

FW

Camera/UI Module



19.3.55 Camera Take Photo Stop (61-44)

The **Camera Take Photo Stop** command is only available for a VP6800 with a camera (PCI Standard version).

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-----------------------|---------|-------------|------------|------------|-----------|-----------|
| Header Tag & Protocol | Command | Sub Command | Length MSB | Length LSB | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 44h | 00h | 00h | Varies | Varies |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | 61h | See status code table | 01h | 00h | | |

19.3.56 Camera Take Photo Once (61-45)

The **Camera Take Photo** command is only available for a VP6800 with a camera (PCI Standard version).

Before using this command, send the **Camera Take Photo Start (61-43)** command to activate the camera.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-----------------------|---------|-------------|------------|------------|-----------|-----------|
| Header Tag & Protocol | Command | Sub Command | Length MSB | Length LSB | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | 61h | 45h | 00h | 00h | Varies | Varies |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | 61h | See status code table | 00h | 00h | | |

19.4 Additional VP6800 Information

19.4.1 Settings Menu

To access the settings menu:

1. Reset the VP6800.
2. Before the screen loads, quickly press the top-right corner of the screen, then do a long press in the top-left corner.

The reader loads the settings menu.

19.4.2 Press-and-hold Events

The area that detects press-and-hold events is 70px*70px and always at the top-left and top-right screen corners.

If a user presses-and-holds for 1 second, the reader registers and sends an Object Event Call Back for a long press event (refer to 3.5).

19.4.3 Supported special characters

| Symbol | Value | Note |
|--------|-------|--------------------|
| ® | 0xAE | |
| © | 0xA9 | |
| ™ | 0xF0 | |
| † | 0xF1 | |
| ē | 0xF2 | Not yet supported. |
| è | 0xE8 | |
| é | 0xE9 | |
| ~ | 0x7E | |
| SM | 0xF3 | |
| & | 0x26 | |
| ! | 0x21 | |
| ¡ | 0xA1 | |
| ? | 0x3F | |
| ¿ | 0xBF | |
| * | 0x2A | |
| + | 0x2B | |
| § | 0xA7 | |
| ¢ | 0xA2 | |
| \$ | 0x24 | |
| ▪ | 0xB7 | |
| / | 0x2F | |
| \ | 0x5C | |
| # | 0x23 | |
| - | 0x2D | |
| % | 0x25 | |
| ½ | 0xBD | |
| ¼ | 0xBC | |

19.4.4 Numeric Entry

Only IDG commands requiring digital input can make numeric entries display. A user CANNOT call **Add Object to Screen (61-07)** to display a numeric entry widget.

If the user UI shows a numeric entry screen, the numeric entry is kept until the user updates the UI.

If the system UI shows a numeric entry screen, the numeric entry is closed after the end of the numeric entry session.

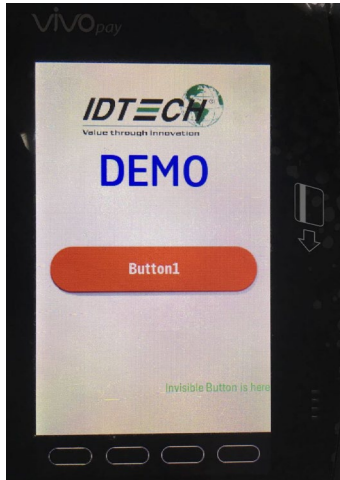
19.4.5 Extended Status Codes

| Status Code | Status |
|-------------|--|
| 00h | OK |
| 01h | Incorrect Header Tag |
| 02h | Unknown Command |
| 03h | Unknown Sub-Command |
| 04h | CRC Error in Frame |
| 05h | Incorrect Parameter |
| 06h | Parameter Not Supported |
| 07h | Mal-formatted Data |
| 08h | Timeout |
| 0Ah | Failed / NACK |
| 0Bh | Command not Allowed |
| 0Ch | Sub-Command not Allowed |
| 0Dh | Buffer Overflow (Data Length too large for reader buffer) |
| 0Eh | User Interface Event |
| 10h | Need clear firmware (apply in boot loader only) |
| 11h | Communication type not supported, VT-1, burst, and so on. Need encrypted firmware (apply in boot loader only) |
| 12h | Secure interface is not functional or is in an intermediate state. |
| 13h | Data field is not mod 8 |
| 14h | Pad 0x80 not found where expected |
| 15h | Specified key type is invalid |
| 16h | Could not retrieve key from the SAM (InitSecureComm) |
| 17h | Hash code problem |
| 18h | Could not store the key into the SAM (InstallKey) |
| 19h | Frame is too large |
| 1Ah | Unit powered up in authentication state but POS must resend the InitSecureComm command |
| 1Bh | The EEPROM may not be initialized because SecCommInterface does not make sense |
| 1Ch | Problem encoding APDU |
| A0h | Duplicated Name when create screen or object |
| A1h | SD card not inserted |
| A5h | LCD is not working |

19.5 Tutorial: Creating Screens for the VP6800

The following sections outline the steps used to create screens for the VP6800.

19.5.1 Creating Screen Style 1



1. Create **Screen1** with the **Create Screen (61-0A)** command.

Command:

```
56 69 56 4F 74 65 63 68 32 00 61 0A 00 08 53 63 72 65 65 6E 31 00
A2 D7
```

Response:

```
56 69 56 4F 74 65 63 68 32 00 61 00 00 02 00 00 3E FF
```

2. Add **Image1** to **Screen1** with the **Add Object to Screen (61-07)** command.

| Data Item | Code | Description |
|---------------------|-------------------------|---|
| Screen Name | 53 63 72 65 65 6E 31 | Screen1 |
| Object Name | 49 6D 61 67 65 31 | Image1 |
| Component ID | 08 | Image |
| Alignment | 00 | Display object at the horizon center of specified y, while x is ignored |
| x-coordinate | 00 00 | 0 |
| y-coordinate | 14 00 | 20 |
| String | 6C 6F 67 6F 2E 6A 70 67 | logo.jpg |

Command:

```
56 69 56 4F 74 65 63 68 32 00 61 07 00 1E 53 63 72 65 65 6E 31 00
49 6d 61 67 65 31 00 08 00 00 00 14 00 6C 6F 67 6F 2E 6A 70 67 00
```

Response:

```
56 69 56 4F 74 65 63 68 32 00 61 00 00 0C 00 00 08 00 20 00 14 00
65 00 E3 00 11 13
```

3. Add **Text1** to **Screen1** with the **Add Object to Screen (61-07)** command. The maximum number of text areas per page screen is 20.

| Data Item | Code | Description |
|--------------|----------------------|---|
| Screen Name | 53 63 72 65 65 6E 31 | Screen1 |
| Object Name | 54 65 78 74 31 | Text1 |
| Component ID | 00 | Text Area |
| Alignment | 00 | Display object at the horizon center of specified y, while x is ignored |
| x-coordinate | 00 00 | 0 |
| y-coordinate | 5E 00 | 94 |
| Font ID | 27 | RoundSemibold_48 |
| Font Color | FC 44 30 00 | blue B-252 G-68 R-48 |
| String | 44 45 4D 4F | DEMO |

Command:

```
56 69 56 4F 74 65 63 68 32 00 61 07 00 22 53 63 72 65 65 6E 31 00
54 65 78 74 31 00 00 00 00 00 5E 00 00 00 00 00 27 FC 44 30 00 44
45 4D 4F 00 EF C9
```

Response:

```
56 69 56 4F 74 65 63 68 32 00 61 00 00 0C 00 00 00 00 47 00 5E 00
C9 00 A0 00 00 31
```

4. Add a large button to **Screen1** with the **Add Object to Screen (61-07)** command. The maximum number of large buttons per page screen is 8.

| Data Item | Code | Description |
|--------------|-------------------------------------|---|
| Screen Name | 53 63 72 65 65 6E 31 | Screen1 |
| Object Name | 4C 61 72 67 65 20 42 75 74 74 6F 6E | Large Button |
| Component ID | 01 | Large Button |
| Alignment | 02 | Display object at center of screen, x, y are both ignored |
| String | 42 75 74 74 6F 6E 31 | Button1 |

Command:

```
56 69 56 4F 74 65 63 68 32 00 61 07 00 23 53 63 72 65 65 6E 31 00
4C 61 72 67 65 20 42 75 74 74 6F 6E 00 01 02 00 00 00 00 42 75 74
74 6F 6E 31 00 A2 39
```

Response:

```
56 69 56 4F 74 65 63 68 32 00 61 00 00 0C 00 00 01 00 0B 00 D2 00
05 01 0E 01 B2 26
```

5. Add an invisible button to **Screen1** with the **Add Object to Screen (61-07)** command. The maximum number of invisible buttons per page screen is 16.

| Data Item | Code | Description |
|---------------------|--|--|
| Screen Name | 53 63 72 65 65 6E 31 | Screen1 |
| Object Name | 49 6E 76 69 73 69 62 6C 65 20 42 75 74 74 6F 6E | Invisible Button |
| Component ID | 03 | Invisible Button |
| Alignment | 04 | Display object at left of the screen of specified y, while x ignored |
| x-coordinate | 00 00 | 0 |
| y-coordinate | 72 01 | 370 |

Command:

```
56 69 56 4F 74 65 63 68 32 00 61 07 00 1F 53 63 72 65 65 6E 31 00
49 6E 76 69 73 69 62 6C 65 20 42 75 74 74 6F 6E 00 03 04 00 00 72
01 15 53
```

Response:

```
56 69 56 4F 74 65 63 68 32 00 61 00 00 0C 00 00 03 00 C9 00 72 01
0F 01 AE 01 D6 F7
```

6. Add a get button event with the **Add Object to Screen (61-07)** command. **Be sure to click the button before sending the command.**

Note: See also the **Get Medium Button Call Back** step below for **Screen2**.

7. Command:

```
56 69 56 4F 74 65 63 68 32 00 61 06 00 00 21 F9
```

- a. If the large button (**Button1**) is implemented correctly and clicked, the reader sends this message:

```
56 69 56 4F 74 65 63 68 32 00 61 00 00 19 00 00 01 00 53 63
72 65 65 6E 31 00 4C 61 72 67 65 20 42 75 74 74 6F 6E 00 CC
B4
```

8. Add a get invisible button event with the **Add Object to Screen (61-07)** command. In order to verify the invisible button loaded successfully, locate of the button through **Text3** and get the button's click event.

| Data Item | Code | Description |
|---------------------|----------------------|--|
| Screen Name | 53 63 72 65 65 6E 31 | Screen1 |
| Object Name | 54 65 78 74 32 | Text2 |
| Component ID | 00 | Text Area |
| Alignment | 04 | Display object at the horizon center of specified y, while x ignored |
| x-coordinate | 00 00 | 0 |

| | | |
|---------------------|---|--------------------------|
| y-coordinate | 72 01 | 370 |
| Font ID | 0E | RoundCondensedMedium_12 |
| Font Color | 3A F1 2C 00 | green B-58 G-241 R-44 |
| String | 49 6E 76 69 73 69 62 6C 65 20 42 75 74 74 6F 6E 20 69 73 20 68 65 72 65 | Invisible Button is here |

Command:

```
56 69 56 4F 74 65 63 68 32 00 61 07 00 36 53 63 72 65 65 6E 31 00
54 65 78 74 32 00 00 04 00 00 72 01 00 00 00 00 0E 3A F1 2C 00 49
6E 76 69 73 69 62 6C 65 20 42 75 74 74 6F 6E 20 69 73 20 68 65 72
65 00 EF C9
```

- a. If the invisible button and its callback were implemented correctly, the reader sends this message:

```
56 69 56 4F 74 65 63 68 32 00 61 00 00 1E 00 00 03 00 08 53
63 72 65 65 6E 31 00 49 6E 76 69 73 69 62 6C 65 20 42 75 74
74 6F 6E 00 9B 55
```

9. Add an LED widget to **Screen1** with the **Add Object to Screen (61-07)** command. This command lasts for a maximum of 5 seconds. The maximum number of LED widgets per page screen is 16.

| Data Item | Code | Description |
|--------------|----------------------|--|
| Screen Name | 53 63 72 65 65 6E 31 | Screen1 |
| Object Name | 4C 45 44 31 | LED1 |
| Component ID | 06 | LED |
| Alignment | 01 | Display object at specified x and y |
| x-coordinate | 00 00 | 0 |
| y-coordinate | AE 01 | 430 |
| Features | 00 01 02 03 | LED0-Off;LED1-Green;LED2-Yellow;LED3-Red |

Command:

```
56 69 56 4F 74 65 63 68 32 00 61 07 00 17 53 63 72 65 65 6E 31 00
4C 45 44 31 00 06 01 00 00 AE 01 00 01 02 03 66 0B
```

Response:

```
56 69 56 4F 74 65 63 68 32 00 61 00 00 0C 00 00 06 00 00 00 AE 01
10 01 E0 01 08 02
```

10. Adjust the backlight with the **Set Backlight (83-2F)** command. This command lasts for a maximum of 5 seconds.

Command:

56 69 56 4F 74 65 63 68 32 00 83 2F 00 01 64 93 FB

Response:

56 69 56 4F 74 65 63 68 32 00 83 00 00 00 22 03

11. Publish and display Screen1 with the **Show Screen (61-05)** command.

Command:

56 69 56 4F 74 65 63 68 32 00 61 05 00 08 53 63 72 65 65 6E 31 00
91 5C

Response:

56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81

12. Store all current screen information from RAM into flash with the **Store Screen Info (61-0E)** command to ensure the reader does not lose any data after a reboot.

Note: If the reader requires a reboot, execute the **Load Screen Info (61-0F)** command to display **Screen1** again.

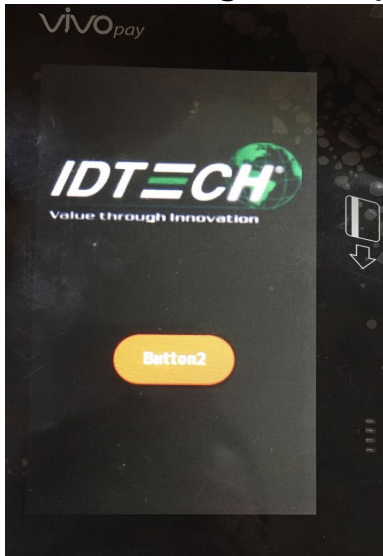
Command:

56 69 56 4F 74 65 63 68 32 00 61 0E 00 00 80 50

Response:

56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81

19.5.2 Creating Screen Style 2



1. Create **Screen2** with the **Create Screen (61-0A)** command.

Note: In the response below, **00 01** is the screen ID for **Screen2**. See the Create Screen (61-0A) command for more details.

Command:

```
56 69 56 4F 74 65 63 68 32 00 61 0A 00 08 53 63 72 65 65 6E 32 00
A2 D7
```

Response:

```
56 69 56 4F 74 65 63 68 32 00 61 00 00 02 00 01 2E DE
```

2. Add **Image2** To **Screen2** with the **Add Object to Screen (61-07)** command. The maximum number of images per page screen is 20.

Notes:

- Only JPEG files without orientation Exif info and width <= 272, height <= 480 are supported; otherwise the VP6800 rejects this command.
- **Exif** (exchangeable image file format): a standard that specifies the formats for images, sound, and ancillary tags used by digital cameras (including smartphones), scanners, and other systems that handle image and sound files recorded by digital cameras.

| Data Item | Code | Description |
|---------------------|--|-------------------------------------|
| Screen Name | 53 63 72 65 65 6E 32 | Screen2 |
| Object Name | 49 6D 61 67 65 32 | Image2 |
| Component ID | 08 | Image |
| Alignment | 02 | Display object at specified x and y |
| x-coordinate | 00 00 | 0 |
| y-coordinate | 00 00 | 0 |
| String | 6C 6F 67 6F 5F 62 6C 61 63 6B 2E 6A 70 67 | Logo_black.jpg |

Command:

```
56 69 56 4F 74 65 63 68 32 00 61 07 00 24 53 63 72 65 65 6E 32 00
49 6D 61 67 65 32 00 08 02 00 00 00 00 6C 6F 67 6F 5F 62 6C 61 63
6B 2E 6A 70 67 00
```

Response:

```
56 69 56 4F 74 65 63 68 32 00 61 00 00 0C 00 01 08 00 00 00 00 00
E0 01 10 01 AE E9
```

3. Add a medium button to **Screen2** with the **Add Object to Screen (61-07)** command. The maximum number of medium buttons per page screen is 20.

Note: In the result below, **02 00** is the object ID for the medium button. Refer to the Add Object to Screen (61-07) command for more details.

| Data Item | Code | Description |
|---------------------|--|--|
| Screen Name | 53 63 72 65 65 6E 32 | Screen2 |
| Object Name | 4C 61 72 67 65 20 42 75 74 74 6F 6E | Medium Button |
| Component ID | 02 | Medium Button |
| Alignment | 00 | Display object at the horizon center of specified y, while x ignored |
| x-coordinate | 00 00 | 0 |
| y-coordinate | 22 01 | 290 |
| String | 42 75 74 74 6F 6E 32 | Button2 |

Command:

```
56 69 56 4F 74 65 63 68 32 00 61 07 00 24 53 63 72 65 65 6E 32 00
4D 65 64 69 75 6D 20 42 75 74 74 6F 6E 00 02 00 00 00 22 01 42 75
74 74 6F 6E 32 00 A2 39
```

Response:

```
56 69 56 4F 74 65 63 68 32 00 61 00 00 0C 00 01 02 00 4C 00 22 01
C4 00 5E 01 2D 1A
```

4. Add a call back for the medium button with the **Object Event Call Back(61-FF)** command.

| Item | Code | Description |
|--------------------|-------------------------------------|--|
| Screen ID | 00 01 | Screen2(if you do not know, refer 2.2.1 Create "Screen2" result) |
| Object ID | 02 00 | Medium Button(if you do not know, refer 2.2.3 Add Medium Button To "Screen2" result) |
| Screen Name | 53 63 72 65 65 6E 32 | Screen2 |
| Object Name | 4C 61 72 67 65 20 42 75 74 74 6F 6E | Medium Button |

Command:

```
56 69 56 4F 74 65 63 68 32 00 61 FF 00 19 00 01 02 00 53 63 72 65
65 6E 32 00 4C 61 72 67 65 20 42 75 74 74 6F 6E 00
```

- a. If the medium button (Button2) and its call back are correctly implemented, the VP6800 sends the following message:

```
56 69 56 4F 74 65 63 68 32 00 61 FF 00 1A 00 01 02 00 53 63
72 65 65 6E 32 00 4D 65 64 69 75 6D 20 42 75 74 74 6F 6E 00
A9 BF
```

5. Publish and display Screen2 with the **Show Screen (61-05)** command.

Command:

```
56 69 56 4F 74 65 63 68 32 00 61 05 00 08 53 63 72 65 65 6E 32 00
91 5C
```

Response:

```
56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81
```

6. Store all current screen information from RAM into flash with the **Store Screen Info (61-0E)** command to ensure the reader does not lose any data after a reboot.

Note: If the reader requires a reboot, execute the **Load Screen Info (61-0F)** command to display **Screen2** again.

Command:

```
56 69 56 4F 74 65 63 68 32 00 61 0E 00 00 80 50
```

Response:

```
56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81
```

19.5.3 Additional Information About VP6800 Screens

Note the following additional information about VP6800 screens and commands.

1.1.3.20. Reloading Screens After Reboot

A VP6800 loads the default factory screen after a reboot. To display custom screens created in the process above, use the **Load Screen Info (61-10)** command.

Command:

```
56 69 56 4F 74 65 63 68 32 00 61 0F 00 00 B0 67
```

Response:

```
56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81
```

1.1.3.21. Load the System UI

Use the Display System Screen (61-17) command to hide the customer-facing UI and show the system UI.


Command:

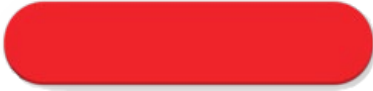


```
56 69 56 4F 74 65 63 68 32 00 61 17 00 00 72 8D
```

Response:

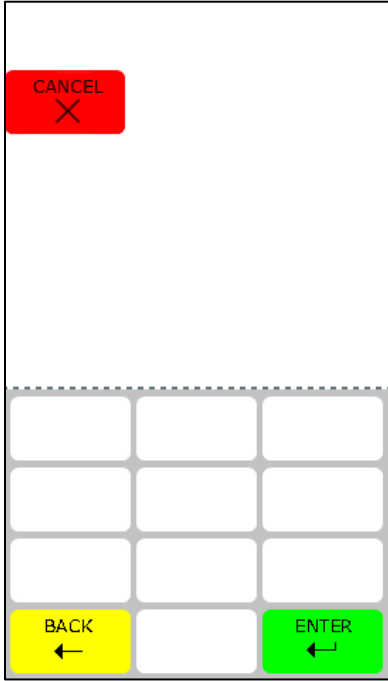
```
56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81
```

19.6 VP6800 UI Limitations

| Supported Customer-Used Components | Limitation |
|------------------------------------|---|
| Screen | <ul style="list-style-type: none"> • Maximum number of text components in each screen is 50. • Maximum characters of a screen name is 32 bytes. |
| Object | <ul style="list-style-type: none"> • Maximum characters of an object name is 32 bytes |
| Text | <ul style="list-style-type: none"> • Current supported fonts are listed in VP6800 Supported Fonts. • To support new fonts, modify RT1050 firmware. • Supported characters must be listed in the character library. To support new characters, modify RT1050 firmware. For the current supported special characters, see VP6800 Supported Special Characters. • Maximum number of text components in each screen is 20. • Maximum characters in each text component is 64 bytes. |
| Medium Button | <ul style="list-style-type: none"> • Current supported style and size is fixed to the following image. To support or change to other styles, modify RT1050 firmware. <div style="text-align: center;">  <p>(120px*60px unpressed)</p> <p>(120px*60px pressed)</p> </div> <ul style="list-style-type: none"> • Maximum number of medium button components in each screen is 16. • Maximum characters in each medium button component is 10 bytes. |
| Large Button | <ul style="list-style-type: none"> • Current supported style and size is fixed to the following image. To support or change to other styles, modify RT1050 firmware. |

| Supported Customer-Used Components | Limitation |
|------------------------------------|--|
| |  (250px*60px unpressed)  (250px*60px pressed) <ul style="list-style-type: none"> • Maximum number of large button components in each screen is 16. • Maximum characters in each large button component is 25 bytes. |
| Invisible Button | <ul style="list-style-type: none"> • Size is fixed to (70px * 60px) and long press timing is 1 second. To support other sizes or long press timing, modify RT1050 firmware. • Maximum number of Invisible Button components in each screen is 3. |
| LED Widget | <ul style="list-style-type: none"> • LEDs style and size is fixed to following image. The only supported colors are Green, Red, Yellow, and Off. To change the style or support more colors, modify RT1050 firmware.  (272px*50px) <ul style="list-style-type: none"> • Maximum number of LED Widget components in each screen is 1. |
| Image | <ul style="list-style-type: none"> • Only images stored in flash can be loaded to a screen. • Only JPEG files without orientation Exif(*) info and width <= 272, height <= 480 is are supported. • Maximum number of image components in each screen is 20. • Maximum characters of each image name is 32 bytes. |
| Video | <ul style="list-style-type: none"> • Only videos stored in the SD card can be loaded to a screen. • Only MJPEG files with width <= 272 and height <= 480 are supported. • Maximum number of image components in each screen is 1. • Maximum characters of each image name is 32 bytes. • The file size should < 3MB. • If a file name includes an underscore (_), only numbers can follow the underscore. For example: <ul style="list-style-type: none"> • Correct: video_1.avi • Incorrect: video_no1.avi |

***Exif** (exchangeable image file format): a standard that specifies the formats for images, sound, and ancillary tags used by digital cameras (including smartphones), scanners, and other systems that handle image and sound files recorded by digital cameras.

| System UI | Limitation |
|---------------|---|
| Numeric Entry | <p>Style and button position is fixed to the following image. To support special requirements, modify RT1050 and K81 firmware.</p>  |
| Text | <p>Font is fixed to:</p> <ul style="list-style-type: none"> • RoundCondensedBold_24 (English, French) • STZhongsong_24 (Chinese) • YuMincho_24 (Japanese) <p>To support special requirements, modify RT1050 firmware.</p> |

19.6.1 VP6800 Supported Fonts

| Font ID | Typography Name | Font | Size |
|---------|-------------------------|----------------------------|------|
| 0 | RoundBold_12 | RoundBold.ttf | 12 |
| 1 | RoundBold_18 | RoundBold.ttf | 18 |
| 2 | RoundBold_24 | RoundBold.ttf | 24 |
| 3 | RoundBold_36 | RoundBold.ttf | 36 |
| 4 | RoundBold_48 | RoundBold.ttf | 48 |
| 5 | RoundBold_60 | RoundBold.ttf | 60 |
| 6 | RoundBold_72 | RoundBold.ttf | 72 |
| 7 | RoundCondensedBold_12 | RoundCondensedBold.ttf | 12 |
| 8 | RoundCondensedBold_18 | RoundCondensedBold.ttf | 18 |
| 9 | RoundCondensedBold_24 | RoundCondensedBold.ttf | 24 |
| 10 | RoundCondensedBold_36 | RoundCondensedBold.ttf | 36 |
| 11 | RoundCondensedBold_48 | RoundCondensedBold.ttf | 48 |
| 12 | RoundCondensedBold_60 | RoundCondensedBold.ttf | 60 |
| 13 | RoundCondensedBold_72 | RoundCondensedBold.ttf | 72 |
| 14 | RoundCondensedMedium_12 | RoundCondensedMedium_0.ttf | 12 |
| 15 | RoundCondensedMedium_18 | RoundCondensedMedium_0.ttf | 18 |
| 16 | RoundCondensedMedium_24 | RoundCondensedMedium_0.ttf | 24 |
| 17 | RoundCondensedMedium_36 | RoundCondensedMedium_0.ttf | 36 |

| Font ID | Typography Name | Font | Size |
|---------|---------------------------|----------------------------|------|
| 18 | RoundCondensedMedium_48 | RoundCondensedMedium_0.ttf | 48 |
| 19 | RoundCondensedMedium_60 | RoundCondensedMedium_0.ttf | 60 |
| 20 | RoundCondensedMedium_72 | RoundCondensedMedium_0.ttf | 72 |
| 21 | RoundCondensedSemibold_12 | RoundCondensedSemibold.ttf | 12 |
| 22 | RoundCondensedSemibold_18 | RoundCondensedSemibold.ttf | 18 |
| 23 | RoundCondensedSemibold_24 | RoundCondensedSemibold.ttf | 24 |
| 24 | RoundCondensedSemibold_36 | RoundCondensedSemibold.ttf | 36 |
| 25 | RoundCondensedSemibold_48 | RoundCondensedSemibold.ttf | 48 |
| 26 | RoundCondensedSemibold_60 | RoundCondensedSemibold.ttf | 60 |
| 27 | RoundCondensedSemibold_72 | RoundCondensedSemibold.ttf | 72 |
| 28 | RoundMedium_12 | RoundMedium.ttf | 12 |
| 29 | RoundMedium_18 | RoundMedium.ttf | 18 |
| 30 | RoundMedium_24 | RoundMedium.ttf | 24 |
| 31 | RoundMedium_36 | RoundMedium.ttf | 36 |
| 32 | RoundMedium_48 | RoundMedium.ttf | 48 |
| 33 | RoundMedium_60 | RoundMedium.ttf | 60 |
| 34 | RoundMedium_72 | RoundMedium.ttf | 72 |
| 35 | RoundSemibold_12 | RoundSemibold.ttf | 12 |
| 36 | RoundSemibold_18 | RoundSemibold.ttf | 18 |
| 37 | RoundSemibold_24 | RoundSemibold.ttf | 24 |
| 38 | RoundSemibold_36 | RoundSemibold.ttf | 36 |
| 39 | RoundSemibold_48 | RoundSemibold.ttf | 48 |
| 40 | RoundSemibold_60 | RoundSemibold.ttf | 60 |
| 41 | RoundSemibold_72 | RoundSemibold.ttf | 72 |

19.6.2 VP6800 Supported Special Characters

| Symbol | Value | Note |
|--------|-------|-------------------|
| ® | 0xAE | |
| © | 0xA9 | |
| ™ | 0xF0 | |
| † | 0xF1 | |
| ē | 0xF2 | Not supported yet |
| è | 0xE8 | |
| é | 0xE9 | |
| ~ | 0x7E | |
| SM | 0xF3 | |
| & | 0x26 | |
| ! | 0x21 | |
| i | 0xA1 | |
| ? | 0x3F | |
| ¿ | 0xBF | |
| * | 0x2A | |
| + | 0x2B | |
| § | 0xA7 | |
| ç | 0xA2 | |
| \$ | 0x24 | |

| | | |
|---|------|--|
| ▪ | 0xB7 | |
| / | 0x2F | |
| \ | 0x5C | |
| # | 0x23 | |
| - | 0x2D | |
| % | 0x25 | |
| ½ | 0xBD | |
| ¼ | 0xBC | |

19.7 VP6800-Specific Hardware Integration Commands: Watchdog for Hosts

The commands listed in this section only work on VP6800 readers connected to specific hardware.

Sending these commands to a VP6800 not connected to that specific hardware results in status code 0x0A; sending these commands to any other NEO 2.0 reader results in status code 0x0B

19.7.1 Set Configure Watchdog (57-00)

The **Set Configure Watchdog** command configures a VP6800 to act as a watchdog for the connected host device.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOPayV2\0 | 57h | 00h | 00h | 06h | See Data Table Below | | |

Data Table

| Data Item | Length (bytes) | Description |
|--|----------------|--|
| Select Watchdog Hardware | 1 | 0 = Do not use watchdog functionality. 1 = Use VP6800 HW Definition. |
| Configure Watchdog Hardware | 1 | 0 = Reset is logic low; steady state, not pulse-based. 1 = Reset is logic high; steady state, not pulsed-based. |
| Configure Watchdog Timeout (Seconds) | 2 | Default is 0x0384 (900 Seconds – 15 Minutes). 1 ~ 65535 seconds (1092 Minutes). |
| Configure Power Down Pulse Duration (ms) | 2 | Default is 0x00C8 (200ms). 1~65535 ms . |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-14+n-1 | Byte 14 | Byte 15 |
|-------------------------------|---------|------------------|-------------------|-------------------|----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | 57h | See Status Codes | 00h | 00h | | | |

Status Codes

| Status Code | Description |
|-------------|--|
| 00h | Success |
| 05h | illegal value or out of range |
| 0Ah | Watchdog hardware not supported on this reader |

19.7.2 Read Configure Watchdog (57-01)

The **Read Configure Watchdog** command to retrieves the watchdog configuration set in the **Set Configure Watchdog (57-00)** command.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | 57h | 01h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|------------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | 57h | 00h | 00h | 06h | See Return Value Table Below | | |

Return Values

| Data Item | Length (bytes) | Description |
|--|----------------|--|
| Select Watchdog Hardware | 1 | 0 = Do not use watchdog functionality. 1 = Use VP6800 HW Definition. |
| Configure Watchdog Hardware | 1 | 0 = Reset is logic low; steady state, not pulse-based. 1 = Reset is logic high; steady state, not pulsed-based. |
| Configure Watchdog Timeout (Seconds) | 2 | Default is 0x0384 (900 Seconds – 15 Minutes). 1 ~ 65535 seconds (1092 Minutes). |
| Configure Power Down Pulse Duration (ms) | 2 | Default is 0x00C8 (200ms). 1~65535 ms . |

19.7.3 Enable/Disable Watchdog (57-02)

The **Enable/Disable Watchdog** command enables or disables a VP6800 unit as a host device watchdog.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | 57h | 02h | 00h | 01h | See Below Data Table | | |

Data Table

| Data Item | Length (bytes) | Description |
|--------------|----------------|---------------------------|
| Set Watchdog | 1 | 0 = Disable 1 = Enable |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-14+n-1 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | 57h | See status code table | 00h | 00h | | | |

Status Codes

| Status Code | Description |
|-------------|--|
| 00h | Success. Note: Always feed the WD timer (57-04). |
| 05h | Illegal value or out of range. Note: Always feed the WD timer (57-04). |
| 0Ah | Watchdog hardware not supported on this NEO 2.0 reader. Note: Always feed the WD timer (57-04). |

19.7.4 Read Watchdog Enable/Disable Information (57-03)

The **Read Watchdog Enable/Disable Information** command retrieves the enable/disable watchdog information set in the **Enable/Disable Watchdog (57-02)** command.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | 57h | 03h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|------------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | 57h | 00h | 00h | 01h | See Below Return Value Table | | |

Return Values

| Data Item | Length (bytes) | Description |
|--------------|----------------|---------------------------|
| Return Value | 1 | 0 = Disable 1 = Enable |

19.7.5 Feed Watchdog Timer (57-04)

The **Feed Watchdog Timer** command to increases ("feeds") the timer duration for the VP6800.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | 57h | 04h | 00h | 00h | | |

Note: Sending any other commands to the VP6800 is also treated as feeding.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-14+n-1 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | 57h | See status code table | 00h | 00h | | | |

Status Codes

| Status Code | Description |
|-------------|-------------------------------|
| 00h | Success (Feed WD Timer OK) |
| 05h | illegal value or out of range |

19.7.6 Erase Watchdog Reset Counter and Log (57-05)

The **Erase Watchdog Reset Counter and Log** command erases the watchdog reset counter and log.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | 57h | 05h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-14+n-1 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | 57h | See status code table | 00h | 00h | | | |

Status Codes

| Status Code | Description |
|-------------|--|
| 00h | Success (Sets the counter to zero and erases any data in the log). |
| 05h | Illegal value or out of range. |
| 0Ah | Watchdog hardware not supported on this NEO 2.0 reader. |

19.7.7 Read Watchdog Reset Counter and Log (57-06)

The **Read Watchdog Reset Counter and Log** command retrieves the watchdog reset counter and log.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | 57h | 06h | 00h | 01h | See Below Data Table | | |

Data Table

| Data Item | Length (bytes) | Description |
|-------------|----------------|---|
| Read Select | 1 | 0x01 = The request returns the counter for the number of resets triggered. 0x02 = The request returns the date and time of each triggered reset, based on RTC. |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|------------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | 57h | 00h | Variable | Variable | See Below Return Value Table | | |

Return Values

| Data Item | Length (bytes) | Description |
|--------------|----------------|--|
| Return Value | 1 or x | <p>If Data of command is 0x01:</p> <ul style="list-style-type: none"> Return 1-byte Counter Value (Max is 255). Maximum of 255. If there are more than 255 triggered resets, do not roll counter over to zero. Keep counter at 255 until it is reset. <p>If Data of command is 0x02:</p> <ul style="list-style-type: none"> Return n bytes date & time, format: n * 6 bytes (BCD code). [YY MM DD HH MM SS] (Year, Month, Day, Hour, Minute, Second) Maximum of 255. The output will be from newest to oldest. If there are more than 255 records, remove the oldest record first (queue). |

19.8 VP6800 Memory-Related Commands

The following commands provide information about VP6800 memory usage.

19.8.1 Get External Chip Memory Usage Information (09-31)

The **Get External Chip Memory Usage Information** command retrieves the reader's RT1050 memory usage information.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-----------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViVotech2\0 | 09h | 31h | 00h | 00h | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte12 | Byte 13 | Byte 14 ... Byte 13+n | Byte 14+n | Byte 15+n |
|-----------------------|---------|---------------------------------------|-------------------|-------------------|--------------------------|-----------|-----------|
| Header Tag & Protocol | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVotech2\0 | 09h | See status code table | | | See below | | |

If there is an error, the reader returns the appropriate status code with an empty data field (Data Length = 0000h).

If the command is successful, the data field (Length = 000Ch) contains the following information:

Free Heap Size: Available heap size.
Memory Not Freed Block Count: Memory in use block count.
Minimum Ever Free Heap Size: The lowest ever available heap size.

The data field (big endian) is formatted as follows:

| Offset | Length (bytes) | Data Description |
|--------|----------------|------------------------------|
| 00h | 4 | Free Heap Size |
| 04h | 4 | Memory Not Freed Block Count |
| 08h | 4 | Minimum Ever Free Heap Size |

Example:

Command: 56 69 56 4F 74 65 63 68 32 00 09 31 00 00 35 52

Response: 56 69 56 4F 74 65 63 68 32 00 09 00 00 0C 00 01 1D 70 00 00 00 11 00 00 F7 80 9D 30

Data field Length: 0x000C

Free Heap Size: 0x011D70

Memory Not Freed Block Count: 0x00000011

Minimum Ever Free Heap Size: 0x0000F780

19.9 VP6800 Wi-Fi Configuration

The following commands configure VP6800 Wi-Fi settings. See the General Commands section for each command's full details.

19.9.1 VP6800 Wi-Fi Commands

- **Set Wi-Fi Configuration (D1-17):** The **Set Wi-Fi Configuration** command sets the reader's Wi-Fi configuration; the configuration is saved in flash.
- **Get Wi-Fi Configuration (D1-18):** The **Get Wi-Fi Configuration** command returns all the reader's network configurations.
- **Set Wireless Work Mode (D1-19):** The **Set Wireless Work Mode** command sets the reader's wireless work mode (Wi-Fi/BLE); the configuration is saved in flash.
- **Get Wireless Work Mode (D1-1A):** The **Get Wireless Work Mode** command returns the reader's wireless work mode.
- **Wi-Fi SSL Connection Request (D1-1C):** The **Wi-Fi SSL Connection Request** command sends an SSL request to the SSL server.

19.9.2 Using VP6800 Wi-Fi Commands

To configure a VP6800, complete the commands above in the following sequence with the information specified below.

1. Complete a **Set Wifi Configuration (D1-17)** command to configure the Wi-Fi mode (station mode), SSID of AP, password of AP, IP address, net mask, and gateway.
2. Complete a **Set Wireless Work Mode (D1-19)** command to set the Wi-Fi SSL client mode.
3. Start a TLS server on the host PC using the server's IP and port number.
4. After the reader has been configured and TLS server is already running, use the **Wi-Fi SSL Connection Request (D1-1C)** command with the server's IP address and server port number to connect to the remote server.

Notes:

- Steps 1 to 3 only need to be completed once unless the Wi-Fi configuration changes.
- Every time the TLS connection is established or fails (in step 4), re-send the **Wi-Fi SSL Connection Request (D1-1C)** command.
- Restart the VP6800 if the reader's Wi-Fi connection fails multiple times.

20. Appendix A.9: NEO 2 Contactless Groups and Default TLVs

The following table contains tags that are preconfigured within Group 0.

Table 84: NEO2 Group 0 Default Tags

| Tag | Description | Format | Length | Default Value |
|--------|--|--------|--------|---|
| DFEE2D | Group Number / Fallback Group | n2 | 1 | 00 |
| 5F2A | Transaction Currency Code | n3 | 2 | 08 40 |
| 5F36 | Transaction Currency Exponent | n1 | 1 | 02 |
| 9C | Transaction Type | n2 | 1 | 00 |
| 9F02 | Amount, Authorized (Numeric) | n12 | 6 | 00 00 00 00 00 01 |
| 9F03 | Amount, Other (Numeric) | n12 | 6 | 00 00 00 00 00 00 |
| 9F09 | Application Version Number | b | 2 | 00 02 |
| 9F15 | Merchant Category Code | n4 | 2 | 00 00 |
| 9F1A | Host Country Code | n2 | 2 | 08 40 |
| 9F40 | Additional Host Capabilities | b | 5 | 60 00 00 30 00 |
| 9F4E | Merchant Name and Location | ans | 30 | 00 |
| 9F53 | Host Interchange Profile (dynamic) | an | 1 | 00 |
| 9F66 | TTQ | b | 4 | 30 00 40 00 |
| 9F6D | Kernel 4 Reader Capabilities | b | 2 | 00 01 |
| 9F7C | Merchant Custom Data | b | 20 | 00 |
| DF8120 | Host Action Code - Default | b | 5 | F8 50 AC A0 00 |
| DF8121 | Host Action Code - Denial | b | 5 | 00 00 00 00 00 |
| DF8122 | Host Action Code - Online | b | 5 | F8 50 AC F8 00 |
| DFED18 | Poll Mode | b | 1 | 01 |
| DFEE1C | LCD Font Size | b | 1 | 03 |
| DFEE28 | Host Capabilities - No CVM Required | b | 3 | 00 08 E8 |
| DFEE29 | Host Capabilities - CVM Required | b | 3 | 00 68 E8 |
| DFEE34 | Host Contactless Transaction Limit | n12 | 6 | 00 00 00 01 00 00 |
| DFEE35 | Visa Reader Risk Flags | b | 3 | 00 06 01 |
| DFEE36 | CVM Required Limit SEE COMMENTS | n12 | 6 | 00 00 00 00 80 00 |
| DFEE37 | UI Scheme | b | 1 | 03 |
| DFEE38 | Language Option for LCD (See NOTE further below) | n2 | 1 | 00 |
| DFEE44 | Application Capability | b | 2 | 9D FF |
| DFEE5C | RF Deactivate Period | b | 4 | 01 50 00 00 |
| DFEE6B | Host IFD | an | 8 | 00 00 00 00 00 00 00 00 |
| DFEE70 | Loyalty Program ID | b | 1 | 00 |
| DFEE71 | Value Added Tax 1 | b | 2 | 00 00 |
| DFEE72 | Value Added Tax 2 | b | 2 | 00 00 |
| DFEE73 | Merchant Category Code | b | 2 | 00 00 |
| DFEE74 | Discover Optional Features | b | 1 | 00 |

| Tag | Description | Format | Length | Default Value |
|--------|----------------|--------|--------|---------------|
| DFEE7E | Burst Mode | b | 1 | 00 |
| DFEF29 | LCD Delay Time | b | 2 | 00 00 |

Table 85: NEO 2 Group 80 (Master) Default Tags

| Tag | Description | Format | Length | Default Value |
|--------|---|--------|--------|----------------------------|
| DFEE2D | Group Number / Fallback Group | n2 | 1 | 80 |
| 5F57 | Account type | B | 0 | null |
| 9F01 | Acquirer Identifier | n12 | 0 | null |
| 9F09 | Application Version Number | b | 2 | 00 02 |
| 9F15 | Merchant Category Code | n4 | 2 | 11 11 |
| 9F16 | Merchant Identifier | ans | 0 | null |
| 9F1A | Host Country Code | n3 | 2 | 08 40 |
| 9F1C | Host ID | an | 0 | null |
| 9F1D | Terminal risk management | b | 8 | 6C FF 00 00 00 00 00 00 |
| 9F1E | Host IFD | an | 8 | 30 30 30 30 30 30 30 30 |
| 9F33 | Host Capabilities | b | 0 | null |
| 9F35 | Host Type | b | 1 | 22 |
| 9F40 | Additional Host Capabilities | b | 5 | 00 00 00 00 00 |
| 9F4E | Merchant Name & Location | ans | 0 | null |
| 9F53 | Transaction Category Code | an | 1 | 00 |
| 9F6D | Magstripe Application version number | b | 2 | 00 01 |
| 9F7E | Mobile Support Indicator | b | 0 | null |
| DF60 | DS Input (Card) | b | 0 | null |
| DF62 | DS ODS Info | b | 0 | null |
| DF63 | DS ODS Term | b | 0 | null |
| DF8108 | DS AC Type | b | 0 | null |
| DF8109 | DS Input (Term) | b | 0 | null |
| DF810A | DS ODS Info For Reader | b | 0 | null |
| DF810D | DSVN Term | b | 0 | null |
| DF8117 | Card Data Input Capability | b | 1 | 00 |
| DF8118 | Host Capabilities - CVM Req. | b | 1 | 60 |
| DF8119 | Host Capabilities - No CVM Required | b | 1 | 08 |
| DF811A | Default UDOL | b | 3 | 9F 6A 04 |
| DF811B | Kernel Configuration | b | 1 | 20 |
| DF811C | Max Lifetime of Torn Transaction Log Record | b | 2 | 00 00 |
| DF811D | Max Number of Torn Transaction Log Records | b | 1 | 00 |
| DF811E | Magstripe CVM Capability CVM Required | b | 1 | 10 |
| DF811F | Security Capabilities | b | 1 | 08 |
| DF8120 | TAC Default | b | 5 | 00 00 00 00 00 |
| DF8121 | TAC Denial | b | 5 | 00 00 00 00 00 |
| DF8122 | TAC Online | b | 5 | 00 00 00 00 00 |
| DF8123 | Reader Contactless Floor Limit | n12 | 6 | 00 00 00 01 00 00 |
| DF8124 | Reader Contactless Limit - No On-device CVM | n12 | 6 | 00 00 00 03 00 00 |
| DF8125 | Reader contactless Limit - On-device CVM | n12 | 6 | 00 00 00 05 00 00 |
| DF8126 | CVM Required Limit | n12 | 6 | 00 00 00 00 10 00 |

| Tag | Description | Format | Length | Default Value |
|--------|---|--------|--------|---------------|
| DF812C | Magstripe CVM Capability No CVM Required | b | 1 | 00 |
| DFEE0F | Revocation List | b | 1 | 01 |
| DFEE37 | UI Scheme | b | 1 | 03 |
| DFEE38 | LCD Language | b | 1 | 00 |
| DFEF46 | ACT Response DOL for ODA (online or offline) PASS | b | 2 | 9F 41 |
| DFEF4E | ACT Response DOL for MSD PASS | b | 2 | 9F 41 |
| DFEF4F | ACT Response DOL for Fail or Declined | b | 2 | 9F 41 |

Table 86: NEO 2 Group 90 (Visa) Default Tags

| Tag | Description | Format | Length | Default Value |
|--------|---|--------|--------|--|
| DFEE2D | Group Number / Fallback Group | n2 | 1 | 90 |
| 5F2A | Transaction Currency Code | n3 | 2 | 08 40 |
| 5F36 | Transaction Currency Exponent | n1 | 1 | 02 |
| 9C | Transaction Type | n2 | 1 | 00 |
| 9F02 | Amount, Authorized (Numeric) | n12 | 6 | 00 00 00 00 01 00 |
| 9F03 | Amount, Other (Numeric) | n12 | 6 | 00 00 00 00 00 00 |
| 9F1A | Host Country Code | n3 | 2 | 08 40 |
| 9F1B | Host Floor Limit | b | 4 | 00 00 17 70 |
| 9F1E | Host IFD | an | 8 | 00 00 00 00 00 00 00 00 |
| 9F33 | Host Capabilities | b | 3 | 00 08 E8 |
| 9F35 | Host Type | n2 | 1 | 25 |
| 9F41 | Transaction Sequence Counter | n4-8 | 4 | 00 00 00 00 |
| 9F66 | Host Transaction Qualifier (TTQ) | b | 4 | 30 00 40 00 |
| DF8120 | Host Action Code - Default | b | 5 | F8 50 AC A0 00 |
| DF8121 | Host Action Code - Denial | b | 5 | F8 50 AC F8 00 |
| DF8122 | Host Action Code - Online | b | 5 | F8 50 AC F8 00 |
| DF8126 | CVM Required Limit | b | 6 | 00 00 00 00 80 00 |
| DFEE34 | Host Contactless Transaction Limit | b | 6 | 00 00 00 01 00 00 |
| DFEE35 | Visa Reader Risk Flags | b | 3 | 01 00 01 |
| DFEE67 | Specific Feature Switch | b | 3 | 02 00 00 |
| DFEF46 | ACT Response DOL for ODA (online or offline) PASS | b | 6 | 5F 30 9F 19 9F 41 |
| DFEF4E | ACT Response DOL for MSD PASS | b | 10 | 5F 30 9F 19 9F 41 9F 6E 9F 7C |
| DFEF4F | ACT Response DOL for Fail or Declined | b | 6 | 5F 30 9F 19 9F 41 |
| FFEE13 | Cashback Reader Risk Record | b | 41 | 9F 1B 04 00 00 05 DC 9F 66 04 30 00 40 00 DF EE 34 06 00 00 00 00 20 00 DF EE 35 03 00 D7 00 DF 81 26 06 00 00 00 00 10 00 |

Table 87: NEO 2 Group A0 (Amex) Default Tags

| Tag | Description | Format | Length | Default Value |
|--------|-------------------------------|--------|--------|---------------|
| DFEE2D | Group Number / Fallback Group | n2 | 1 | A0 |
| 5F2A | Transaction Currency Code | n3 | 2 | 08 40 |

| Tag | Description | Format | Length | Default Value |
|--------|---|--------|--------|----------------------------|
| 5F36 | Transaction Currency Exponent | n1 | 1 | 02 |
| 9C | Transaction Type | n2 | 1 | 00 |
| 9F02 | Amount, Authorized (Numeric) | n12 | 6 | 00 00 00 00 00 01 |
| 9F03 | Amount, Other (Numeric) | n12 | 6 | 00 00 00 00 00 00 |
| 9F09 | Application Version Number | b | 2 | 00 02 |
| 9F15 | Merchant Category Code | n4 | 2 | 00 00 |
| 9F1A | Host Country Code | n3 | 2 | 08 40 |
| 9F1E | Host IFD | an | 8 | 00 00 00 00 00 00 00 00 |
| 9F33 | Host Capabilities | b | 3 | 00 08 E8 |
| 9F35 | Host Type | n2 | 1 | 25 |
| 9F40 | Additional Host Capabilities | b | 5 | 60 00 00 30 00 |
| 9F41 | Transaction Sequence Counter | n4-8 | 4 | 00 00 00 00 |
| 9F6D | Contactless Reader Capabilities | n2 | 1 | C0 |
| 9F6E | Enhanced Contactless Reader Capabilities | b | 4 | 58 80 00 00 |
| DF8120 | Host Action Code - Default | b | 5 | F8 50 AC A0 00 |
| DF8121 | Host Action Code - Denial | b | 5 | 00 00 00 00 00 |
| DF8122 | Host Action Code - Online | b | 5 | F8 50 AC F8 00 |
| DF8123 | Reader Contactless Floor Limit | b | 6 | 00 00 00 00 60 00 |
| DF8126 | CVM Required Limit | b | 6 | 00 00 00 00 80 00 |
| DFEE34 | Host Contactless Transaction Limit | n12 | 6 | 00 00 00 01 00 00 |
| DFEE5C | RF Deactivate Period | b | 4 | 01 50 00 00 |
| Dfef46 | ACT Response DOL for ODA (online or offline) PASS | b | 8 | 5F 24 5F 30 9F 41 9F 6E |
| Dfef4E | ACT Response DOL for MSD PASS | b | 8 | 5F24 5F 30 9F 41 9F 6E |
| Dfef4F | ACT Response DOL for Fail or Declined | b | 4 | 5F 30 9F 41 |

Table 88: NEO 2 Group B0 (Discover) Default Tags

| Tag | Description | Format | Length | Default Value |
|--------|----------------------------------|--------|--------|----------------------------|
| DFEE2D | Group Number / Fallback Group | n2 | 1 | B0 |
| 5F2A | Transaction Currency Code | n3 | 2 | 08 40 |
| 5F36 | Transaction Currency Exponent | n1 | 1 | 02 |
| 9C | Transaction Type | n2 | 1 | 00 |
| 9F02 | Amount, Authorized (Numeric) | n12 | 6 | 00 00 00 00 00 01 |
| 9F03 | Amount, Other (Numeric) | n12 | 6 | 00 00 00 00 00 00 |
| 9F09 | Application Version Number | b | 2 | 01 00 |
| 9F1A | Host Country Code | n3 | 2 | 08 40 |
| 9F1B | Host Floor Limit | b | 4 | 00 00 17 70 |
| 9F1E | Host IFD | an | 8 | 00 00 00 00 00 00 00 00 |
| 9F33 | Host Capabilities | b | 3 | 00 08 E8 |
| 9F35 | Host Type | n2 | 1 | 25 |
| 9F41 | Transaction Sequence Counter | n4-8 | 4 | 00 00 00 00 |
| 9F66 | Host Transaction Qualifier (TTQ) | b | 4 | 80 00 40 00 |
| DF8120 | Host Action Code - Default | b | 5 | 00 00 00 00 00 |
| DF8121 | Host Action Code - Denial | b | 5 | 00 00 00 00 00 |

| Tag | Description | Format | Length | Default Value |
|--------|--|--------|--------|--|
| DF8122 | Host Action Code - Online | b | 5 | 00 00 00 00 00 |
| DF8123 | Reader Contactless Floor Limit | N12 | 6 | 00 00 00 01 50 00 |
| DF8124 | Reader Contactless Transaction Limit (No On Dev-CVM) | N12 | 6 | 00 00 00 03 00 00 |
| DF8126 | CVM Required Limit | N12 | 6 | 00 00 00 00 20 00 |
| DFEE35 | Visa Reader Risk Flags | b | 3 | 01 00 01 |
| DFEF46 | ACT Response DOL for ODA (online or offline) PASS | b | 12 | 9F 19 9F 1E 9F 41 5F 24 9F 6E 9F 21 |
| DFEF4E | ACT Response DOL for MSD PASS | b | 12 | 9F 19 9F 1E 9F 41 5F 24 9F 6E 9F 21 |
| DFEF4F | ACT Response DOL for Fail or Declined | b | 12 | 9F 19 9F 1E 9F 41 5F 24 9F 6E 9F 21 |

Table 89: NEO 2 Group C0 (Interac) Default Tags

| Tag | Description | Format | Length | Default Value |
|--------|--|--------|--------|-------------------|
| DFEE2D | Group Number / Fallback Group | n2 | 1 | C0 |
| 5F2A | Transaction Currency Code | n12 | 6 | 01 24 |
| 5F36 | Transaction Currency Exponent | n1 | 1 | 02 |
| 9C | Transaction Type | b | 1 | 00 |
| 9F02 | Amount Authorized | n12 | 6 | 00 00 00 00 15 00 |
| 9F03 | Amount Other | n12 | 6 | 00 00 00 00 00 00 |
| 9F09 | Application Version Number | b | 2 | 00 02 |
| 9F1A | Host Country Code | n3 | 2 | 01 24 |
| 9F1B | Host Floor limit | b | 4 | 00 00 00 00 |
| 9F33 | Host Capabilities | b | 3 | 00 48 E8 |
| 9F35 | Host Type | n2 | 1 | 22 |
| 9F58 | Merchant Indicator | n1 | 1 | 03 |
| 9F59 | Host Transaction Information | b | 3 | DC 87 00 |
| 9F5A | Host Transaction Type | b | 1 | 00 |
| 9F5D | Host Contactless Receipt Requirement Limit | n12 | 6 | 00 00 00 00 50 00 |
| 9F5E | Host Option Status | b | 2 | E0 00 |
| 9F5F | Host Contactless Floor Limit | n12 | 6 | 00 00 00 00 50 00 |
| DF8120 | TAC - Default | b | 5 | 00 00 00 00 00 |
| DF8121 | TAC - Denial | b | 5 | 00 00 00 00 00 |
| DF8122 | TAC - Online | b | 5 | 00 00 00 00 00 |
| DF8124 | Transaction limit | n12 | 6 | 00 00 00 01 00 00 |
| DF8126 | CVM limit | n12 | 6 | 00 00 00 00 60 00 |
| DFEE2A | Threshold Value Biased Random Selection | n12 | 6 | 00 00 00 00 50 00 |
| DFEE2B | Max Target% Based Random Selection | b | 1 | 32 |
| DFEE2C | Target% Random Selection | b | 1 | 0A |
| DFEE37 | UI Scheme | b | 1 | 03 |
| DFEF56 | Retry Limit | b | 1 | 03 |

Table 90: NEO 2 Group D0 (JCB EMV) Default Tags

| Tag | Description | Format | Length | Default Value |
|--------|-------------------------------|--------|--------|---------------|
| DFEE2D | Group Number / Fallback Group | n2 | 1 | D0 |
| 5F2A | Transaction Currency Code | n3 | 2 | 03 92 |
| 5F36 | Transaction Currency Exponent | n1 | 1 | 02 |

| Tag | Description | Format | Length | Default Value |
|--------|--|--------|--------|---|
| 9C | Transaction Type | n2 | 1 | 00 |
| 9F01 | Acquirer Identifier | b | 6 | 00 00 00 00 00 10 |
| 9F15 | Merchant Category Code | b | 2 | 70 32 |
| 9F1A | Host Country Code | n3 | 2 | 03 92 |
| 9F35 | Host Type | n2 | 1 | 25 |
| 9F40 | Additional Host Capabilities | b | 5 | F0 00 00 00 00 |
| 9F4E | Merchant Name and Location | b | 23 | 58 58 20 4D 45 52 43 48 41 4E 54 20 59 59 20 4C 4F 43 41 54 49 4F 4E |
| 9F53 | JCB Host Interchange Profile (static) | b | 3 | 70 80 00 |
| DF8120 | Host Action Code - Default | b | 5 | 90 40 00 80 00 |
| DF8121 | Host Action Code - Denial | b | 5 | 04 10 00 00 00 |
| DF8122 | Host Action Code - Online | b | 5 | 90 60 00 90 00 |
| DF8123 | Reader Contactless Floor Limit | b | 6 | 00 00 00 00 45 00 |
| DF8126 | CVM Required Limit | b | 6 | 00 00 00 01 00 00 |
| DFEE2A | Threshold Value for Biased Random Selection | b | 6 | 00 00 00 00 20 00 |
| DFEE2B | Maximum Target Percentage to be Used for Biased Random Selection | b | 1 | 00 |
| DFEE2C | Target Percentage to be Used for Biased Random Selection | b | 1 | 00 |
| DFEE34 | Host Contactless Transaction Limit | b | 6 | 00 00 00 02 00 00 |
| DFEF19 | Unpredictable Number Range | b | 4 | 00 00 00 3C |
| DFEF44 | JCB Combination Option | b | 2 | 7B 00 |
| DFEF45 | JCB Removal Timeout (The time after which cardholder is asked to remove the card) | b | 2 | 10 00 |

Table 91: NEO 2 Group E0 (CUP) Default Tags

| Tag | Description | Format | Length | Default Value |
|--------|------------------------------------|--------|--------|--|
| DFEE2D | Group Number / Fallback Group | n2 | 1 | E0 |
| 5F2A | Transaction Currency Code | n3 | 2 | 01 56 |
| 9C | Transaction Type | n2 | 1 | 00 |
| 9F09 | Application version number | b | 2 | 00 30 |
| 9F16 | Merchant Identifier | ans | 15 | 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 |
| 9F1A | Host Country Code | n3 | 2 | 01 56 |
| 9F1B | Host Floor limit | n32 | 4 | 00 00 3A 98 |
| 9F33 | Host Capabilities | b24 | 3 | 00 68 C8 |
| 9F35 | Host Type | n2 | 1 | 25 |
| 9F4E | Merchant Name and Location | b | 20 | D2 F8 BC EC D6 D0 D0 C4 20 42 43 54 43 26 30 31 30 32 30 33 |
| 9F66 | Host Transaction Qualifiers | b32 | 4 | 37 00 40 80 |
| DF8126 | CVM Required Limit | b | 6 | 00 00 00 01 20 00 |
| DFEE34 | Host Contactless Transaction Limit | b | 6 | 00 00 00 03 00 00 |

Table 92: NEO 2 Group 8E (SmartTap) Default Tags

| Tag | Description | Format | Length | Default Value |
|--------|--|--------|--------|----------------|
| DFEE2D | Group Number / Fallback Group | n2 | 1 | 8E |
| DFEE3B | Collector ID | b | 0 | null |
| DFEE3C | Store Location | b | 0 | null |
| DFEE3D | Host ID | b | 0 | null |
| DFEF25 | Merchant Name | b | 0 | null |
| DFED01 | Merchant Category | b | 0 | null |
| DFED02 | POS Capabilities Bitmaps | b | 5 | 00 00 00 00 01 |
| DFED03 | Retry Times | b | 1 | 01 |
| DFED04 | Select OSE support | b | 1 | 01 |
| DFED05 | Skip Second Select support | b | 1 | 01 |
| DFED06 | Stop Payment if Google Pay SmartTap 2.1 failed support | b | 1 | 00 |
| DFED07 | Pre-Signed support | b | 1 | 00 |
| DFED27 | Delimiter for Service Objects | b | 1 | 0D |
| DFED3F | VAS encryption | b | 1 | 00 |
| DFEF77 | Enable/Disable Multiple Service Objects | b | 1 | 00 |

Table 93: NEO 2 Group 85 (Common AID) Default Tags

| Tag | Description | Format | Length | Default Value |
|--------|---------------------------------------|--------|--------|----------------------------|
| DFEE2D | Group Number / Fallback Group | n2 | 1 | 85 |
| 5F57 | Account Type | n2 | 0 | null |
| 9F01 | Acquirer Identifier | n6 | 0 | null |
| 9F09 | Application Version Number | b | 2 | 00 02 |
| 9F15 | Merchant Category Code | n4 | 2 | 11 11 |
| 9F16 | Merchant Identifier | ans15 | 0 | null |
| 9F1A | Terminal Country Code | n3 | 2 | 00 56 |
| 9F1C | Terminal Identification | an8 | 0 | null |
| 9F1D | Terminal Risk Management Data | b | 8 | 40 FF 80 00 00 00 00 00 |
| 9F1E | Interface Device (IFD) Serial Number | an8 | 8 | 30 30 30 30 30 30 30 30 |
| 9F33 | Terminal Capabilities | b | 0 | null |
| 9F35 | Terminal Type | n2 | 1 | 22 |
| 9F40 | Additional Terminal Capabilities | b | 5 | 00 00 00 00 00 |
| 9F4E | Merchant Name and Location | ans | 0 | null |
| 9F6D | Mag-stripe Application Version Number | b | 2 | 00 01 |
| 9F7E | Mobile Support Indicator | b | 1 | 00 |
| DF60 | DS Input (Card) | b | 0 | null |
| DF62 | DS ODS Info | b | 0 | null |
| DF63 | DS ODS Term | b | 0 | null |
| DF8108 | DS AC Type | b | 0 | null |
| DF8109 | DS Input (Term) | b | 0 | null |
| DF810A | DS ODS Info For Reader | b | 0 | null |
| DF810D | DSVN Term | b | 0 | null |
| DF8117 | Card Data Input Capability | b | 1 | 00 |
| DF8118 | CVM Capability - CVM Required | b | 1 | 40 |
| DF8119 | CVM Capability - No CVM Required | b | 1 | 08 |

| Tag | Description | Format | Length | Default Value |
|--------|---|--------|--------|-------------------|
| DF811A | Default UDOL | b | 3 | 9F 6A 04 |
| DF811B | Kernel Configuration | b | 1 | 30 |
| DF811C | Max Lifetime of Torn Transaction Log | b | 2 | 00 00 |
| DF811D | Max Number of Torn Transaction Log | b | 1 | 00 |
| DF811E | Mag-stripe CVM Capability - CVM | b | 1 | 10 |
| DF811F | Security Capability | b | 1 | 08 |
| DF8120 | Terminal Action Code - Default | b | 5 | F4 50 84 80 0C |
| DF8121 | Terminal Action Code - Denial | b | 5 | 00 00 00 00 00 |
| DF8122 | Terminal Action Code - Online | b | 5 | F4 50 84 80 0C |
| DF8123 | Reader Contactless Floor Limit | n12 | 6 | 00 00 00 00 80 00 |
| DF8124 | Reader Contactless Transaction Limit (No On-device CVM) | n12 | 6 | 00 00 00 03 00 00 |
| DF8125 | Reader Contactless Transaction Limit (On-device CVM) | n12 | 6 | 00 00 00 05 00 00 |
| DF8126 | Reader CVM Required Limit | n12 | 6 | 00 00 00 00 60 00 |
| DF812C | Mag-stripe CVM Capability - No CVM Required | b | 1 | 00 |
| DFEE0F | Enable Revocation List Processing | b | 1 | 01 |
| DF8132 | MasterCard MIN Relay Resistance Grace Period | b | 2 | 00 14 |
| DF8133 | MasterCard MAX Relay Resistance Grace Period | b | 2 | 09 60 |
| DF8134 | MasterCard TERM EXP TRANS Relay Resistance CAPDU | b | 2 | 0F 1A |
| DF8135 | MasterCard TERM EXP TRANS Relay Resistance RAPDU | b | 2 | 0F A0 |
| DF8136 | MasterCard Relay Resistance Accuracy Threshold | b | 2 | 0F A0 |
| DFEE37 | UI Scheme | b | 1 | 03 |
| DFEE38 | Language Option for LCD | n2 | 1 | 00 |
| DFEE44 | Application Capability | b | 2 | 00 01 |

21. Appendix A.10: NEO 2 Default System AIDs

The following table contains the default system AIDs for NEO 2.

Table 91: NEO 2 Default System AIDs

| Application Name | Application Identifier | Group |
|-----------------------|-------------------------------|-------|
| System Configuration | | 0 |
| MasterCard | A0 00 00 00 04 10 10 | 80 |
| Maestro | A0 00 00 00 04 30 60 | 80 |
| Visa | A0 00 00 00 03 10 10 | 90 |
| Visa Electron | A0 00 00 00 03 20 10 | 90 |
| Visa Interlink | A0 00 00 00 03 30 10 | 90 |
| Visa Plus | A0 00 00 00 03 80 10 | 90 |
| Amex | A0 00 00 00 25 01 | A0 |
| Discover ZIP | A0 00 00 03 24 10 10 | B0 |
| Discover D-PAS | A0 00 00 01 52 30 10 | B0 |
| Interac | A0 00 00 02 77 10 10 | C0 |
| JCB EMV | A0 00 00 00 65 10 10 | D0 |
| Common-AID VISA | A0 00 00 00 98 08 40 | 90 |
| Common-AID Discover | A0 00 00 01 52 40 10 | B0 |
| Common-AID CUP | A0 00 00 03 33 01 01 08 | E0 |
| Common-AID MasterCard | A0 00 00 00 04 22 03 | 85 |
| Google Pay Smart Tap | 00 00 04 76 D0 00 01 11 | 8E |
| Apple VAS | 4F 53 45 2E 56 41 53 2E 30 31 | 0 |

22. Appendix A.11: NEO 2 Default Configuration TLVs

The following table contains the default configuration TLVs for NEO 2.

Table 92: NEO 2 Default Configuration TLVs

| Name | Tag | Length (Hex) | Value (Hex) | Card Application |
|--------------------|----------|--------------|---|--------------------------------------|
| Group | DF EE 2D | 01 | A0 | American Express |
| AID | 9F 06 | 06 | A0 00 00 00 25 01 | |
| Partial Selection | DF EE 4B | 01 | 01 | |
| Selection Features | DF EE 4D | 01 | 54 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Transaction Type | DF EE 54 | 03 | 04 00 A0 | |
| List | | | | |
| Default Kernel ID | DF EE 59 | 01 | 04 | |
| Application Flow | DF EE 4C | 01 | 02 | |
| Group | DF EE 2D | 01 | 80 | MasterCard PayPass Application |
| AID | 9F 06 | 07 | A0 00 00 00 04 10 10 | |
| Partial Selection | DF EE 4B | 01 | 01 | |
| Selection Features | DF EE 4D | 01 | 74 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Exclude from | DF EE 53 | 01 | 02 | |
| Processing | | | | |
| Transaction Type | DF EE 54 | 0F | 02 00 80 02 01 80 02 09 80 02 17 80 02 20 80 | |
| List | | | | |
| Kernel ID | DF EE 59 | 01 | 02 | |
| Application Flow | DF EE 4C | 01 | 03 | |
| Group | DF EE 2D | 01 | 80 | MasterCard PayPass Application |
| AID | 9F 06 | 07 | A0 00 00 00 04 30 60 | |
| Partial Selection | DF EE 4B | 01 | 01 | |
| Selection Features | DF EE 4D | 01 | 74 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Exclude from | DF EE 53 | 01 | 02 | |
| Processing | | | | |
| Transaction Type | DF EE 54 | 0F | 02 00 80 02 01 80 02 09 80 02 17 80 02 20 80 | |
| List | | | | |
| Kernel ID | DF EE 59 | 01 | 02 | |
| Application Flow | DF EE 4C | 01 | 03 | |

| Name | Tag | Length (Hex) | Value (Hex) | Card Application |
|--------------------|----------|--------------|-------------------------|--------------------------------------|
| Group | DF EE 2D | 01 | 90 | VISA Application |
| AID | 9F 06 | 07 | A0 00 00 00 03 10 10 | |
| Partial Selection | DF EE 4B | 01 | 01 | |
| Selection Features | DF EE 4D | 01 | 14 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Application Flow | DF EE 4C | 01 | 06 | |
| Group | DF EE 2D | 01 | 90 | VISA Application (Visa Electron) |
| AID | 9F 06 | 07 | A0 00 00 00 03 20 10 | |
| Partial Selection | DF EE 4B | 01 | 01 | |
| Selection Features | DF EE 4D | 01 | 14 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Application Flow | DF EE 4C | 01 | 06 | |
| Group | DF EE 2D | 01 | 90 | VISA Application (Visa Interlink) |
| AID | 9F 06 | 07 | A0 00 00 00 03 30 10 | |
| Partial Selection | DF EE 4B | 01 | 01 | |
| Selection Features | DF EE 4D | 01 | 14 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Application Flow | DF EE 4C | 01 | 06 | |
| Group | DF EE 2D | 01 | D0 | JCB EMV Application |
| AID | 9F 06 | 07 | A0 00 00 00 65 10 10 | |
| Partial Selection | DF EE 4B | 01 | 01 | |
| Selection Features | DF EE 4D | 01 | 54 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Transaction Type | DF EE 54 | 0C | 05 00 D0 05 01 D0 05 09 | |
| List | | | D0 05 20 D0 | |
| Kernel ID | DF EE 59 | 01 | 05 | |
| Application Flow | DF EE 4C | 01 | 0E | |
| Group | DF EE 2D | 01 | B0 | Discover (ZIP) Application |
| AID | 9F 06 | 07 | A0 00 00 03 24 10 10 | |
| Partial Selection | DF EE 4B | 01 | 01 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Application Flow | DF EE 4C | 01 | 0D | |

| Name | Tag | Length (Hex) | Value (Hex) | Card Application |
|-------------------------|----------|--------------|----------------------------------|------------------------------|
| Group | DF EE 2D | 01 | C0 | Interac Application |
| AID | 9F 06 | 07 | A0 00 00 02 77 10 10 | |
| Partial Selection | DF EE 4B | 01 | 01 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Exclude from Processing | DF EE 53 | 01 | 02 | |
| Application Flow | DF EE 4C | 01 | 15 | |
| Group | DF EE 2D | 01 | B0 | Discover (D-PAS) Application |
| AID | 9F 06 | 07 | A0 00 00 01 52 30 10 | |
| Partial Selection | DF EE 4B | 01 | 01 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Application Flow | DF EE 4C | 01 | 0D | |
| Group | DF EE 2D | 01 | 90 | Visa Application (Visa Plus) |
| AID | 9F 06 | 07 | A0 00 00 00 03 80 10 | |
| Partial Selection | DF EE 4B | 01 | 01 | |
| Selection Features | DF EE 4D | 01 | 14 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Application Flow | DF EE 4C | 01 | 06 | |
| Group | DF EE 2D | 01 | E0 | CUP |
| AID | 9F 06 | 07 | A0 00 00 03 33 01 01 | |
| Partial Selection | DF EE 4B | 01 | 01 | |
| Selection Features | DF EE 4D | 01 | 54 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Transaction Type | DFEE54 | 0C | 07 00 E0 07 01 E0 07 09 E0 07 20 | |
| List | DFEE59 | 01 | E0 | |
| Kernel ID | DF EE 4C | 01 | 07 | |
| Application Flow | | | 11 | |
| Group | DF EE 2D | 01 | E0 | CUP Debit |
| AID | 9F 06 | 08 | A0 00 00 03 33 01 01 01 | |
| Partial Selection | DF EE 4B | 01 | 01 | |
| Selection Features | DF EE 4D | 01 | 54 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Transaction Type | DFEE54 | 0C | 07 00 E0 07 01 E0 07 09 E0 07 20 | |
| List | DFEE59 | 01 | E0 | |
| Kernel ID | DF EE 4C | 01 | 07 | |
| Application Flow | | | 11 | |

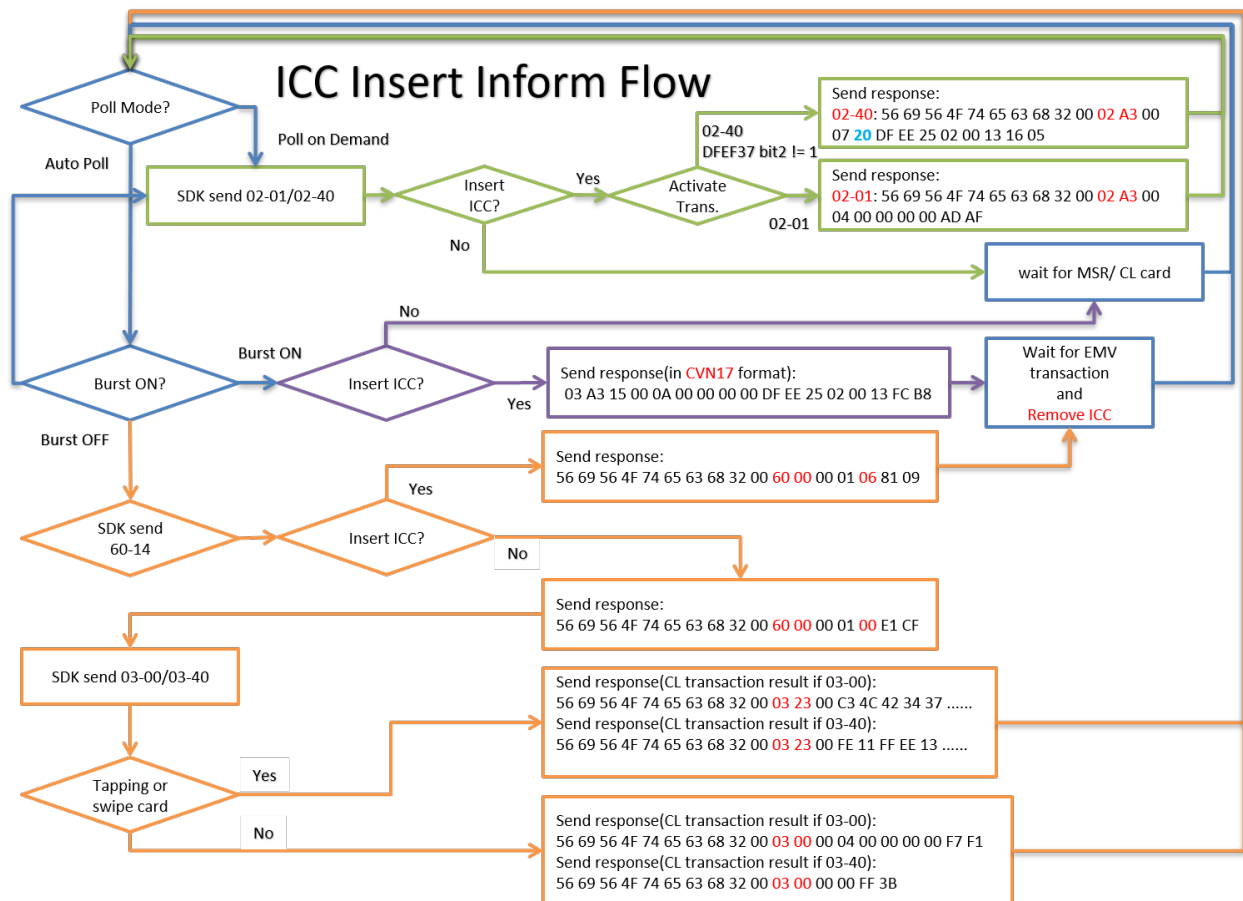
| Name | Tag | Length (Hex) | Value (Hex) | Card Application |
|-------------------------|----------|--------------|----------------------------------|----------------------|
| Group | DF EE 2D | 01 | E0 | CUP Credit |
| AID | 9F 06 | 08 | A0 00 00 03 33 01 01 02 | |
| Partial Selection | DF EE 4B | 01 | 01 | |
| Selection Features | DF EE 4D | 01 | 54 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Transaction Type | DFEE54 | 0C | 07 00 E0 07 01 E0 07 09 E0 07 20 | |
| List | DFEE59 | 01 | E0 | |
| Kernel ID | DF EE 4C | 01 | 07 | |
| Application Flow | | | 11 | |
| Group | DF EE 2D | 01 | 00 | Apple VAS |
| AID | 9F 06 | 0A | 4F 53 45 2e 56 41 53 2e 30 31 | |
| Partial Selection | DF EE 4B | 01 | 00 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Exclude from Processing | DF EE 53 | 01 | 06 | |
| Application Flow | DF EE 4C | 01 | 18 | |
| Group | DF EE 2D | 01 | 8E | Google Pay Smart Tap |
| AID | 9F 06 | 0A | 00 00 04 76 D0 00 01 11 | |
| Partial Selection | DF EE 4B | 01 | 00 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Exclude from Processing | DF EE 53 | 01 | 04 | |
| Application Flow | DF EE 4C | 01 | 18 | |
| Group | DF EE 2D | 01 | 90 | Common AID VISA |
| AID | 9F 06 | 07 | A0 00 00 00 98 08 40 | |
| Partial Selection | DF EE 4B | 01 | 01 | |
| Selection Features | DF EE 4D | 01 | 54 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Application Flow | DF EE 4C | 01 | 06 | |
| Transaction Type | DF EE 54 | 0C | 03 00 90 03 01 90 03 20 90 03 09 | |
| List | | | 90 | |
| Kernel ID | DF EE 59 | 01 | 03 | |

| Name | Tag | Length (Hex) | Value (Hex) | Card Application |
|-------------------------|----------|-----------------|----------------------------------|---------------------------|
| Group | DF EE 2D | 01 | 85 | Common AID Master Card |
| AID | 9F 06 | 07 | A0 00 00 00 04 22 03 | |
| Partial Selection | DF EE 4B | 01 | 01 | |
| Selection Features | DF EE 4D | 01 | 74 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Application Flow | DF EE 4C | 01 | 03 | |
| Transaction Type | DF EE 54 | 03 | 02 00 A9 | |
| List | | | | |
| Kernel ID | DF EE 59 | 01 | 02 | |
| Exclude from Processing | DF EE 53 | 01 | 02 | |
| Group | DF EE 2D | 01 | B0 | Common AID Discover |
| AID | 9F 06 | 07 | A0 00 00 01 52 40 10 | |
| Partial Selection | DF EE 4B | 01 | 01 | |
| Selection Features | DF EE 4D | 01 | 74 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Application Flow | DF EE 4C | 01 | 0D | |
| Transaction Type | DF EE 54 | 06 | 06 00 B0 06 09 B0 | |
| List | | | | |
| Kernel ID | DF EE 59 | 01 | 06 | |
| Exclude from Processing | DF EE 53 | 01 | 02 | |
| Group | DF EE 2D | 01 | E0 | Common AID CUP |
| AID | 9F 06 | 08 | A0 00 00 03 33 01 01 08 | |
| Partial Selection | DF EE 4B | 01 | 01 | |
| Selection Features | DF EE 4D | 01 | 54 | |
| Max AID Length | DF EE 2E | 01 | 10 | |
| Application Flow | DF EE 4C | 01 | 11 | |
| Transaction Type | DF EE 54 | 0C | 07 00 E0 07 01 E0 07 09 E0 07 20 | |
| List | | | E0 | |
| Kernel ID | DF EE 59 | 01 | 07 | |

23. Appendix A.12: Card in Notification Behavior

Some readers perform contact and contactless transactions, and reader can return status code A3h after Activate Transaction (02-01/02-40) and user presentation of ICC card.

Interface transaction flow of ACT is shown below:



24. Appendix A.13: LED 0 Behavior status

The following table describes the LED 0 status between the polling mode and the UI scheme.

Table 93: LED 0 Behavior status

| LED 0 Behavior | | Poll mode | |
|----------------|-----------|-------------------|-------------------|
| | | Auto poll | Poll on demand |
| UI scheme | ViVOtech | ON | ON |
| | Visa wave | Flash every 1 sec | Toggle |
| | EMEA | ON | Flash every 5 sec |

25. Appendix A.14: Enhanced Encrypted MSR Data Output Format for Financial and Non-Financial Cards

The following table describes how masking, encryption, and hash outputs are affected by Command C7-38. For details on the format, refer to document 80000502-001, *ID Tech Encrypted Data Output Formats*, available on the [ID TECH Knowledge Base](#) (no registration required).

25.1 SRED Output Formats

Table 94: Enhanced Encrypted MSR Data Output Format for Bank Card for SRED

| C7-38 Parameter Value | | Clear / Mask Data | | | Encrypted / Hash Data | | |
|------------------------------|------|-------------------|---------|---------|-----------------------|---------|---------|
| | | Track 3 | Track 2 | Track 1 | Track 3 | Track 2 | Track 1 |
| Force Encryption (Bank Card) | 0x17 | M | M | M | E+H | E+H | E+H |

C: Clear Data. M: Mask Data. E: Encrypted Data. H: Hash Data.

Table 95: Enhanced Encrypted MSR Data Output Format for Non-Bank Card for SRED

| C7-38 Parameter Value | | Clear / Mask Data | | | Encrypted / Hash Data | | |
|----------------------------------|---------------------------|-------------------|---------|---------|-----------------------|---------|---------|
| | | Track 3 | Track 2 | Track 1 | Track 3 | Track 2 | Track 1 |
| Force Encryption (non-Bank Card) | 0x00(0000) 0x10(10000) | C | C | C | | | |
| | 0x01(0001) 0x11(10001) | C | C | | | | E+H |
| | 0x02(0010) 0x12(10010) | C | | C | | E+H | |
| | 0x03(0011) 0x13(10011) | C | | | | E+H | E+H |
| | 0x04(0100) 0x14(10100) | | C | C | E+H | | |
| | 0x05(0101) 0x15(10101) | | C | | E+H | | E+H |
| | 0x06(0110) 0x16(10110) | | | C | E+H | E+H | |
| | 0x07(0111) 0x17(10111) | | | | E+H | E+H | E+H |

C: Clear Data. M: Mask Data. E: Encrypted Data. H: Hash Data.

If a bank card PAN is between whitelist settings, the output response remains in plaintext even if encryption is enabled and force encryption is set to ON.

SRED products use all zeroes (0x00) for hash data.

25.2 Non-SRED Output Formats

Table 96: Enhanced Encrypted MSR Data Output Format for Bank Card for NSRED

| C7-38 Parameter Value | | Clear / Mask Data | | | Encrypted/ Hash Data | | |
|---|-------------|-------------------|---------|---------|----------------------|---------|---------|
| | | Track 3 | Track 2 | Track 1 | Track 3 | Track 2 | Track 1 |
| Force Encryption (non-Bank Card) | 0x00(0000) | M | M | M | E | E | E |
| | 0x01(0001) | M | M | M | E | E | E |
| | 0x02(0010) | M | M | M | E | E | E |
| | 0x03(0011) | M | M | M | E | E | E |
| | 0x04(0100) | M | M | M | E | E | E |
| | 0x05(0101) | M | M | M | E | E | E |
| | 0x06(0110) | M | M | M | E | E | E |
| | 0x07(0111) | M | M | M | E | E | E |
| | 0x10(10000) | M | M | M | E+H | E+H | E+H |
| | 0x11(10001) | M | M | M | E+H | E+H | E+H |
| | 0x12(10010) | M | M | M | E+H | E+H | E+H |
| | 0x13(10011) | M | M | M | E+H | E+H | E+H |
| | 0x14(10100) | M | M | M | E+H | E+H | E+H |
| | 0x15(10101) | M | M | M | E+H | E+H | E+H |
| | 0x16(10110) | M | M | M | E+H | E+H | E+H |
| | 0x17(10111) | M | M | M | E+H | E+H | E+H |

C: Clear Data. M: Mask Data. E: Encrypted Data. H: Hash Data.

Table 97: Enhanced Encrypted MSR Data Output Format for Non-Bank Card for NSRED

| C7-38 Parameter Value | | Clear / Mask Data | | | Encrypted/ Hash Data | | |
|---|-------------|-------------------|---------|---------|----------------------|---------|---------|
| | | Track 3 | Track 2 | Track 1 | Track 3 | Track 2 | Track 1 |
| Force Encryption (non-Bank Card) | 0x00(0000) | C | C | C | | | |
| | 0x01(0001) | C | C | | | | E |
| | 0x02(0010) | C | | C | | E | |
| | 0x03(0011) | C | | | | E | E |
| | 0x04(0100) | | C | C | E | | |
| | 0x05(0101) | | C | | E | | E |
| | 0x06(0110) | | | C | E | E | |
| | 0x07(0111) | | | | E | E | E |
| | 0x10(10000) | C | C | C | | | |
| | 0x11(10001) | C | C | | | | E+H |
| | 0x12(10010) | C | | C | | E+H | |
| | 0x13(10011) | C | | | | E+H | E+H |

| | | | | | | | |
|--|-------------|--|---|---|-----|-----|-----|
| | 0x14(10100) | | C | C | E+H | | |
| | 0x15(10101) | | C | | E+H | | E+H |
| | 0x16(10110) | | | C | E+H | E+H | |
| | 0x17(10111) | | | | E+H | E+H | E+H |

C: Clear Data. M: Mask Data. E: Encrypted Data. H: Hash Data.

1. Bits 0-2 (force encrypt setting) only affect **NON-financial** card data. If not set, non-financial card data is sent in clear text. If set, the data is sent in encrypted format. There is no mask data for non-financial cards.
2. Bit 3 is RFU (to be backward compatible with some old ID TECH products) because financial card (type 80) data is always encrypted if the data key is injected.
3. Bit 4 is control hash for NSRED products.

26. Appendix A.15: Common AID

There currently are 5 US Common Debit AIDs:

- Visa: A00000000980840
- M/C: A0000000042203
- Discover: A0000001524010
- China Union Pay: A000000333010108
- Debit Network Alliance (DNA): A0000006200620

ID TECH readers define six new tags. These tags can exist either in host settings to apply to all transactions or in the AID configuration to apply to a specific card brand. When the values exist in both host settings and AID configuration, the AID configuration takes priority.

| Tag | Name | Length | Description |
|--------|---------------------------------------|--------|---|
| DFED48 | Common Debit Prioritizing | 1 | Bit 5-7: RFU Bit 4: Prefer US Common Debit DNA Bit 3: Prefer US Common Debit China Union Pay Bit 2: Prefer US Common Debit Discover Bit 1: Prefer US Common Debit MasterCard Bit 0: Prefer US Common Debit VISA |
| DFED53 | Selectable Kernel Limit | 6 | Range: 00 00 00 00 00 00 ~ 99 99 99 99 99 99 |
| DFED54 | Selectable Kernel Configuration - ICS | 1 | EMV Common L2 V1.10: Range 1~5 EMV Common L2 V1.20: Range 1~23 |
| DFED55 | Credit/Debit/Cashback options | 1 | Bit 2-7: RFU Bit 1: Prompt for Cashback (if kernel configuration supports it) Bit 0: Prompt for Debit/Credit |
| DFED56 | Cashback Menu Values | Var | The first selection in the Cashback Menu is "None" (or something similar) to indicate no cashback desired. The other menu values are taken from DFED56, which contains 6-byte values that populate 9F03 if selected. Example: DFED56 0C 000000002000 000000005000 |
| FFEF01 | Selectable Kernel Tags | Var | Must only contain the following major parameter tags that match the values of the ICS listed in DFED54: 0x9F33, 0x9F35, 0x9F40, 0xDF11, 0xDF26, 0xDF27, 0xDFEE1E |

26.1 Common AID Priority Example

Contact transaction: Common AID Prioritizing

1. Add Contact Application Data (Common Debit VISA)

```
TX: 56 69 56 4F 74 65 63 68 32 00 60 03 00 B7 07 00 A0 00 00 00
98 08 40 14 00 9F 01 06 56 49 53 41 30 30 5F 57 01 00 5F 2A 02
08 40 9F 09 02 00 96 5F 36 01 02 9F 1B 04 00 00 3A 98 DF 25 03
9F 37 04 DF 28 03 9F 08 02 DF EE 15 01 01 DF 13 05 00 00 00 00
00 DF 14 05 00 00 00 00 00 DF 15 05 00 00 00 00 00 00 DF 18 01 00
DF 17 04 00 00 27 10 DF 19 01 00 DF ED 53 06 00 00 00 00 50 00
DF ED 54 01 04 DF ED 55 01 00 DF ED 56 0C 00 00 00 00 20 00 00
00 00 00 50 00 FF EF 01 2A 9F 33 03 60 08 C8 9F 35 01 25 9F 40
05 60 00 F0 50 01 DF 11 01 01 DF 26 01 01 DF 27 01 00 DF EE 1E
08 D0 9C 20 F0 C2 0E 16 00 3B 5E
```

2. Set Contact Host Data (Common Debit Preferring)

TX: 56 69 56 4F 74 65 63 68 32 00 60 06 00 C3 17 00 9F 33 03 60
 08 C8 9F 35 01 25 9F 40 05 60 00 F0 50 01 DF 11 01 01 DF 26 01
 01 DF 27 01 00 DF EE 1E 08 D0 9C 20 F0 C2 0E 16 00 5F 36 01 02
 9F 1A 02 08 40 9F 1E 08 54 65 72 6D 69 6E 61 6C 9F 15 02 12 34
 9F 16 0F 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 9F 1C 08
 38 37 36 35 34 33 32 31 9F 4E 22 31 30 37 32 31 20 57 61 6C 6B
 65 72 20 53 74 2E 20 43 79 70 72 65 73 73 2C 20 43 41 20 2C 55
 53 41 2E DF EE 15 01 01 DF EE 16 01 00 DF EE 17 01 05 DF EE 18
 01 80 DF EE 1F 01 80 DF EE 1B 08 30 30 30 31 35 31 30 30 DF EE
 22 03 32 3C 3C DF 10 08 65 6E 66 72 65 73 7A 68 **DF ED 48 01 01**
 58 63

3. Start Contact transaction

TX: 56 69 56 4F 74 65 63 68 32 00 60 10 00 23 01 00 0A 00 0A 9C
 01 00 5F 57 01 00 9F 02 06 00 00 00 00 02 00 9F 03 06 00 00 00
 00 00 00 5F 2A 02 08 40 AA 34

RX: 56 69 56 4F 74 65 63 68 32 00 60 00 00 83 20 DF EE 25 02 00
 10 57 10 47 61 73 90 01 01 01 35 D1 91 22 01 19 55 92 58 5A 08
 47 61 73 90 01 01 01 35 5F 34 01 01 5F 20 0E 43 41 52 44 30 31
 2F 55 53 44 45 42 49 54 5F 24 03 19 12 31 9F 1E 08 30 4E 30 30
 30 30 30 31 9F 20 00 5F 25 00 5F 2D 02 65 6E 50 08 55 53 20 44
 45 42 49 54 4F 00 84 07 A0 00 00 00 98 08 40 DF EE 23 00 9F 39
 01 05 9F 53 00 DF EE 26 02 20 00 FF EE 01 05 DF EE 30 01 01 37
 67

TX: 56 69 56 4F 74 65 63 68 32 00 60 11 00 03 00 00 78 F3 46

RX: 56 69 56 4F 74 65 63 68 32 00 60 00 01 6F 20 DF EE 25 02 00
 04 4F 00 5A 08 47 61 73 90 01 01 01 35 50 08 55 53 20 44 45 42
 49 54 57 10 47 61 73 90 01 01 01 35 D1 91 22 01 19 55 92 58 82
 02 18 00 84 07 **A0 00 00 00 98 08 40** 8E 12 00 00 00 00 00 00 00
 00 02 01 00 04 02 05 42 00 1F 00 95 05 80 00 10 00 00 9A 03 19
 09 19 9B 02 48 00 9C 01 00 99 00 5F 20 0E 43 41 52 44 30 31 2F
 55 53 44 45 42 49 54 5F 24 03 19 12 31 5F 25 00 5F 28 02 08 40
 5F 2A 02 08 40 5F 2D 02 65 6E 5F 34 01 01 9F 02 06 00 00 00 00
 02 00 9F 03 06 00 00 00 00 00 00 9F 07 02 AB 80 9F 08 02 00 96
 9F 09 02 00 96 9F 0B 00 9F 0D 05 FC 50 AC 88 00 9F 0E 05 00 00
 00 00 00 9F 0F 05 FC 70 BC 98 00 9F 10 07 06 01 0A 03 00 00 00
 9F 11 01 00 9F 12 00 9F 13 00 9F 15 02 12 34 9F 16 0F 30 30 30
 30 30 30 30 30 30 30 30 30 30 30 30 9F 1A 02 08 40 9F 1C 08 38
 37 36 35 34 33 32 31 9F 1E 08 30 4E 30 30 30 30 30 31 9F 20 00
 9F 21 03 08 07 37 9F 24 00 9F 26 08 36 80 21 18 76 CE 99 5D 9F
 27 01 80 9F 33 03 60 08 C8 9F 34 03 1F 00 02 9F 35 01 25 9F 36
 02 01 01 9F 37 04 BE 4D 9F CF 9F 53 00 9F 5B 00 DF 21 00 DF EE
 23 00 9F 39 01 05 DF EE 51 00 DF EE 26 02 20 00 FF EE 01 05 DF
 EE 30 01 01 45 D5

TX: 56 69 56 4F 74 65 63 68 32 00 60 12 00 11 01 8A 02 30 30 91
 0A 11 22 33 44 55 66 77 88 30 30 F3 1C

```

RX: 56 69 56 4F 74 65 63 68 32 00 60 00 00 75 20 DF EE 25 02 00
02 8A 02 30 30 9F 10 07 06 01 0A 03 00 00 00 9F 26 08 36 80 21
18 76 CE 99 5D 9F 27 01 40 9F 36 02 01 01 9F 37 04 BE 4D 9F CF
9F 02 06 00 00 00 00 02 00 9F 13 00 95 05 80 00 10 00 00 9B 02
68 00 9F 03 06 00 00 00 00 00 9F 34 03 1F 00 02 DF EE 51 00
99 00 DF 21 00 9F 39 01 05 9F 53 00 DF EE 26 02 20 00 FF EE 01
05 DF EE 30 01 01 47 06

```

Contactless Transaction: Common AID Prioritizing

1. Add Contactless Configuration (Common Debit VISA)

```

56 69 56 4F 74 65 63 68 32 00 04 02 00 38 DF EE 2D 01 90 9F 06
07 A0 00 00 00 98 08 40 DF EE 4B 01 01 DF EE 4C 01 06 DF EE 4D
01 54 DF EE 2E 01 10 DF EE 54 0C 03 00 90 03 01 90 03 20 90 03
09 90 DF EE 59 01 03 9B 89

```

2. Set Common Debit Prioritizing

```

56 69 56 4F 74 65 63 68 32 00 04 00 00 05 DF ED 48 01 01 EC D0

```

3. Start Contactless Transaction

```

TX: 56 69 56 4F 74 65 63 68 32 00 02 40 00 15 0A 9F 02 06 00 00
00 00 67 89 9A 03 17 12 25 9F 21 03 11 30 05 8F CD
RX: 56 69 56 4F 74 65 63 68 32 00 02 23 01 74 21 50 08 55 53 20
44 45 42 49 54 57 13 47 61 73 90 01 01 01 35 D1 91 22 01 19 55
92 58 00 00 1F 82 02 00 20 84 07 A0 00 00 00 98 08 40 95 05 00
00 00 00 00 9A 03 17 12 25 9B 02 00 00 9C 01 00 5F 20 0F 43 41
52 44 48 4F 4C 44 45 52 2F 56 49 53 41 5F 2A 02 08 40 5F 2D 02
65 6E 5F 34 01 01 9F 02 06 00 00 00 00 67 89 9F 03 06 00 00 00
00 00 00 9F 06 07 A0 00 00 00 98 08 40 9F 07 02 80 80 9F 10 07
06 01 12 03 A0 00 00 9F 1A 02 08 40 9F 1B 04 00 00 00 00 9F 21
03 11 30 05 9F 26 08 B8 DA A9 3B 91 87 04 45 9F 27 01 80 9F 33
03 00 08 E8 9F 36 02 00 02 9F 37 04 3D 96 BB E0 9F 39 01 07 9F
45 02 00 00 9F 4C 08 00 00 00 00 00 00 00 00 9F 4E 1E 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 9F 66 04 A0 00 40 00 9F 6C 02 00 00 9F 6E 04
20 70 00 00 DF 81 26 06 00 00 00 00 80 00 FF EE 01 14 DF EE 30
01 00 DF EE 52 01 00 DF EE 5B 01 08 DF EF 7F 01 00 9F 34 03 3F
00 00 9F 1E 08 30 4E 30 30 30 30 30 31 DF EF 4C 06 00 27 00 00
00 00 DF EF 4D 27 3B 34 37 36 31 37 33 39 30 30 31 30 31 30 31
33 35 3D 31 39 31 32 32 30 31 31 39 35 35 39 32 35 38 30 30 30
30 31 3F DF EE 26 02 21 00 3D 1C

```

27. Appendix A.16: Common AID Selection

Or:

Debit AID Processing Global vs Common Process as Credit vs. Process as Debit Cashback Option
Selectable Kernel EMV Level 2 Kernels in ID TECH products

27.1 Overview

Debit cards issued in the United States come with two default AIDs: one AID provided by issuers that uses the issuers' own networks to route debit transactions and the US Common Debit AID that uses all available networks for routing.

Card issuers are not required to set the US Common Debit AID network by default and instead set their own networks as the higher priority for transactions, and for all "cardholder confirmation off" contact transactions, as well. Merchants, however, can choose to use the less-restrictive US Common Debit network if desired.

27.2 Networks

The US Common Debit AID supports multiple networks the merchant can access. Each network has the following characteristics:

- They will/will not process Online PIN transactions (debit transaction with or without cashback)
- They will/will not process Signature transactions (credit transactions)
- They will/will not process No CVM transactions (debit or credit transactions under a defined threshold)

27.2.1 For Debit Transactions

- If the transaction is **CVM = ONLINE_PIN**:
 - The merchant can get approval through a network that can process Online PIN transactions.
 - If no cashback:
 - The merchant can get approval through a network that can process NO_CVM if the transaction amount is under the Selectable Kernel limit and the Selectable Kernel is a NO_CVM kernel.
 - The merchant can get approval through a network that can process the transaction as a Signature transaction when the transaction amount is over the Selectable Kernel limit, but the network's approval conditions may still require the merchant to capture a Signature transaction.
- If matched **CVM = NO_CVM**, merchant can:
 - Get approval through a network that can process NO_CVM transactions.
 - Get approval through a network that can process the transaction as a Signature transaction.

27.2.2 Forcing No CVM

If the merchant knows that a network will accept debit (PIN transactions) without PIN entry when the transaction is under a certain amount, the merchant has no need to prompt for PIN entry. Readers can include a Selectable Kernel Limit tag, evaluated during runtime, that would bypass PIN entry if the transaction amount is under that limit and the Selectable Kernel tag is set to a NO_CVM kernel. This can be done by loading a kernel configuration, before attempting CVM, that only has the NO_CVM CVM.

27.2.3 Selectable Kernel

A transaction with a Debit AID can be processed as "Credit" or "Debit." In the context of a Debit AID, Credit refers to using a non-PIN CVM, while Debit refers to using a PIN CVM.

- **Credit** transactions on a Debit AID use a non-PIN CVM and are processed over a network that can process Signature transactions (or a network for NO CVM if under NO CVM limit).
- **Debit** transactions on a Debit AID use an Online PIN CVM and are processed over a network that can process PIN (or a network for NO CVM if the transaction is under the NO CVM limit and no cashback).

Presenting the Debit/Credit choice to the customer follows the logic below (see the flowchart at end of doc).

Note that because Contactless transactions don't support Menu selection by the customer, this flow is limited to Contact transactions.

1. The selected AID is a Debit AID (has IIN Tag 42).
2. Prompt the user for **Credit** or **Debit** (if enabled).
3. If the customer chooses **Credit**, load the Selectable Kernel Configuration, which should be a NO_CVM configuration; otherwise, proceed as **Debit**.
4. Prompt the user for **Cashback** (if enabled).
5. If the user selects a **Cashback** amount, set tag **9F03** and process as an Online PIN transaction.
6. If the user selects no cashback (**9F03** = 0), and the transaction amount (**9F02**) is under the Selectable Kernel Limit, load the Selectable Kernel Configuration, which should be a NO_CVM configuration.

For Non-Debit cards, in some cases a region prefers certain cards to process with one CVM and other cards with a different CVM. For example, Japan prefers to have all cards processed as Offline PIN transactions; however, they prefer China Union Pay cards processed as Online PIN transactions. In this case, define the CUP AID file to have a selectable kernel applicable in all cases (with no amount restriction) and do not define other cards with a selectable kernel; just use the default kernel configuration for all other cards.

27.2.4 Selectable Kernel Tag Fallback

The Selectable Kernel Logic has two outcomes: if the transaction is under a specified amount, change to another configuration; otherwise, use the default (starting) configuration.

This logic allows merchants to start with a PIN configuration and then, if the transaction is under a certain amount, change to a NO_CVM (or Signature) transaction configuration.

The limitation our integrators are running into is they would like to start with PIN configuration, but they don't want application selection, and they can't find any configuration that allows PIN but and bypasses application selection. So these integrators will define the Selectable Kernel Tag in the host settings that will allow them to start in a no-application selection kernel (like 4C), and then switch to a PIN kernel (like 3C) as soon as the kernel automatically selects an applications. They do this by setting the Selectable Kernel Tag to apply to all transaction by setting the amount to 0.00.

If a merchant wishes to start with a PIN configuration but does not wish to use application selection, they can define the Selectable Kernel Tag in the host settings. This allows the merchant to start in a no application selection kernel (such as 4C) and then switch to a PIN kernel (such as 3C) as soon as the kernel automatically selects an application. To do this, the merchant sets the Selectable Kernel Tag to apply to all transactions by setting the selectable kernel amount to 0.00.

Selectable Kernel Tag Fallback Logic Flow:

1. Start all transactions in 4C.
2. An application is auto-selected.
3. The Selectable Kernel Tag in Host Settings changes to 3C because the selectable kernel amount is 0.00.
4. The transaction proceeds as PIN_CVM, but the host bypassed application selection.

However, using this logic flow causes the Selectable Kernel Tag defined in the AID files to FAIL when the amount is over the specified limit. This is because the Selectable Kernel Tag expects transaction to start with the default configuration (PIN CVM 3C), and only change to another kernel configuration if under a specified amount.

For this reason, apply the following rule:

When both the AID and Host Settings have a Selectable Kernel Tag and the amount is OVER the amount limit in the AID tag, fallback to the Host Settings Tag and evaluate.

27.2.5 US Common Debit AIDs

Current US Common Debit AIDs:

| | |
|--------------------------------------|------------------|
| Visa: | A00000000980840 |
| M/C: | A0000000042203 |
| Discover: | A0000001524010 |
| China Union Pay: | A000000333010108 |
| Debit Network Alliance (DNA): | A0000006200620 |

27.2.6 Global Debit AIDs

To determine if an AID is a Debit Aid, check for tag 42—the IIN (Issuer Identification Number) in the PPSE/PSE or FCI (File Control Information, under Issuer Discretionary Data Tag BFOC). The IIN is the first 6 digits of the BIN, represented by 3 bytes. If the PPSE/PSE or FCI do not contain tag 42, the AID is not a debit AID.

To determine if US Common Debit AID can replace the Global Debit AID, the Global AID must have tag 42 (IIN) and must match tag 42 of the US Common Debit AID.

See the end of this appendix for examples of PPSE/PSE or FCI.

27.2.7 Contactless vs Contact AID Selection Behavior

Contactless Transactions:

- If a card has multiple AIDs, Contactless transactions always use the AID with the highest priority.

Contact Transactions:

- Use the AID with the highest priority when Cardholder confirmation is OFF.
- Allow the cardholder to select an AID (listed in ascending order with the lowest-priority number first) if Cardholder confirmation is ON and there is more than one AID in the Candidate list.

27.3 US Common Debit AID Selection Solution

Given the information described above in this appendix, ID TECH implements six new tags to allow a merchant to use the US Common Debit AID as the preferred AID. The tags can exist either in host settings to apply to all transactions or in the AID configuration for a specific card brand. If the values exist in both the host settings and the AID configuration, the AID configuration takes priority.

- **DFED48:** Common Debit Preferring, 1 byte
- **DFED53:** Selectable Kernel Limit, 6 bytes
- **DFED54:** Selectable Kernel Configuration - ICS, 1 byte
- **DFED55:** Credit/Debit/Cashback options
- **DFED56:** Cashback Menu Values
- **FFEF01:** Selectable Kernel Tags

27.3.1 Common Debit Preferring: DFED48 – 1 byte

Tag DFED48 describes which US Common AID to use to replace a US Global Debit AID if they both exist on the candidate list.

| Flag | Description |
|----------------|--|
| Bit 5-7 | RFU |
| Bit 4 | Prefer US Common Debit DNA |
| Bit 3 | Prefer US Common Debit China Union Pay |
| Bit 2 | Prefer US Common Debit Discover |
| Bit 1 | Prefer US Common Debit MasterCard |
| Bit 0 | Prefer US Common Debit VISA |

27.3.2 Selectable Kernel Limit: DFED53 - 6 bytes

If both DFED54 and FFEF01 exist, tag DFED53 defines the transaction amount ceiling for changing the kernel configuration to the Selectable Kernel configuration and if the transaction amount is under it. The format matches tag 9F02 (amount), which is 6 bytes.

This tag can be in either:

- the host settings to apply it to all transactions
- the AID of the issuer to apply it to only that transaction

Example: DFED53 06 000000005000

After the AID is selected, if the amount is under 50.00, before the CVM matching stage happens within the kernel, the Selectable Kernel Configuration ICS in DFEE54, and the Selectable Kernel Tags (Major Parameters) in FFEF01 are used for the remainder of the transaction.

If this tag is set to **0** or doesn't exist, but tags DFED54 and FFEF01 do exist, the Selectable Kernel Configuration will apply to all transactions.

27.3.3 Selectable Kernel Configuration ICS: DFED54 - 1 byte

Tag DFED54 contains the Selectable Kernel Configuration to temporarily use when the amount is under the No-CVM limit tag DFED53, or to use for all transactions if DFED53 = 0, or if DFED53 does not exist.

Example: DFED54 01 05

This sets the kernel configuration to 5C before CVM matching happens if the transaction amount is under the amount specified in DFED53, all transactions if DFED53 = 0, or if DFED53 does not exist. This requires host tags that are valid with the Selectable Kernel Configuration ICS to exist in FFEF01.

27.3.4 Credit/Debit/Cashback options: DFED55 – 1 byte

| Flag | Description |
|---------|---|
| Bit 2-7 | RFU |
| Bit 1 | Prompt for Cashback (if kernel configuration supports it) |
| Bit 0 | Prompt for Debit/Credit |

27.3.5 Cashback Menu Values: DFED56 - var bytes

The first selection in Cashback Menu is "None" or some other title to indicate no cashback desired. Other menu values come from DFED56, which contains 6-byte values to populate 9F03 if selected.

Example: DFED56 0C 000000002000 000000005000

This populates a menu with "None", "\$20.00", or "\$50.00." Because this is only for US Debit cards, we know the exponent is 2 and the currency symbol is "\$".

27.3.6 Selectable Kernel Tags - FFEF01

Tag FFEF01 must only contain the following Major Parameter Tags that match the values of the ICS listed in DFED54:

- 0x9F33
- 0x9F35
- 0x9F40
- 0xDF11
- 0xDF26
- 0xDF27
- 0xDFEE1E

27.4 Example Transaction Flow

1. Start of transaction.
2. Candidate List is populated.
 - a. Check tag DFED48 to see if it's necessary to check for US Common AID
 - b. If US Common AID is found, look for any US Global AIDs with Tag 42 (IIN) that match US Common AID. If found, remove it from the Candidate List.
 - c. Return the Candidate List to the Kernel.
3. The Kernel processes the Candidate List.
 - a. US Common AID is selected.
 - b. Check to see if the transaction amount is under the amount in tag DFED53 and if tag DFED54/FFEF01 exists.
 - c. If the transaction amount is under the DFED53 amount and no cashback, load the kernel configuration listed in tag DFED54, which for this process should contain a NO_CVM configuration.
 - d. Check for Debit/Credit option ON by checking tag DFED55, bit 0.
 - i. Present Debit/Credit choice to the customer.
 - ii. If the customer selects Credit, load the kernel configuration listed in tag DFED54 and the tags in FFEF01, which should be a NO_CVM configuration, then proceed with the EMV transaction flow.
 - e. Check for Cashback option ON by checking tag DFED55, bit 1.
 - i. Build the cashback menu from tag DFED56 values.
 - ii. If the customer chose cashback value:
 1. Set 9F03 with cashback value, then proceed with the EMV transaction flow.
 - f. If the transaction is No Credit, No Cashback, and the amount (9F02) is under the Selectable Kernel Configuration limit:
 - i. Load the kernel configuration listed in tag DFED54 and the tags from FFEF01, which should be a NO_CVM configuration, then proceed with the EMV transaction flow.
4. The kernel processes CVM.
5. The kernel produces 1st Gen AC results.
6. At the end of the transaction, restore the kernel configuration to its original value if changed.

27.5 Example FCI from US Visa Global

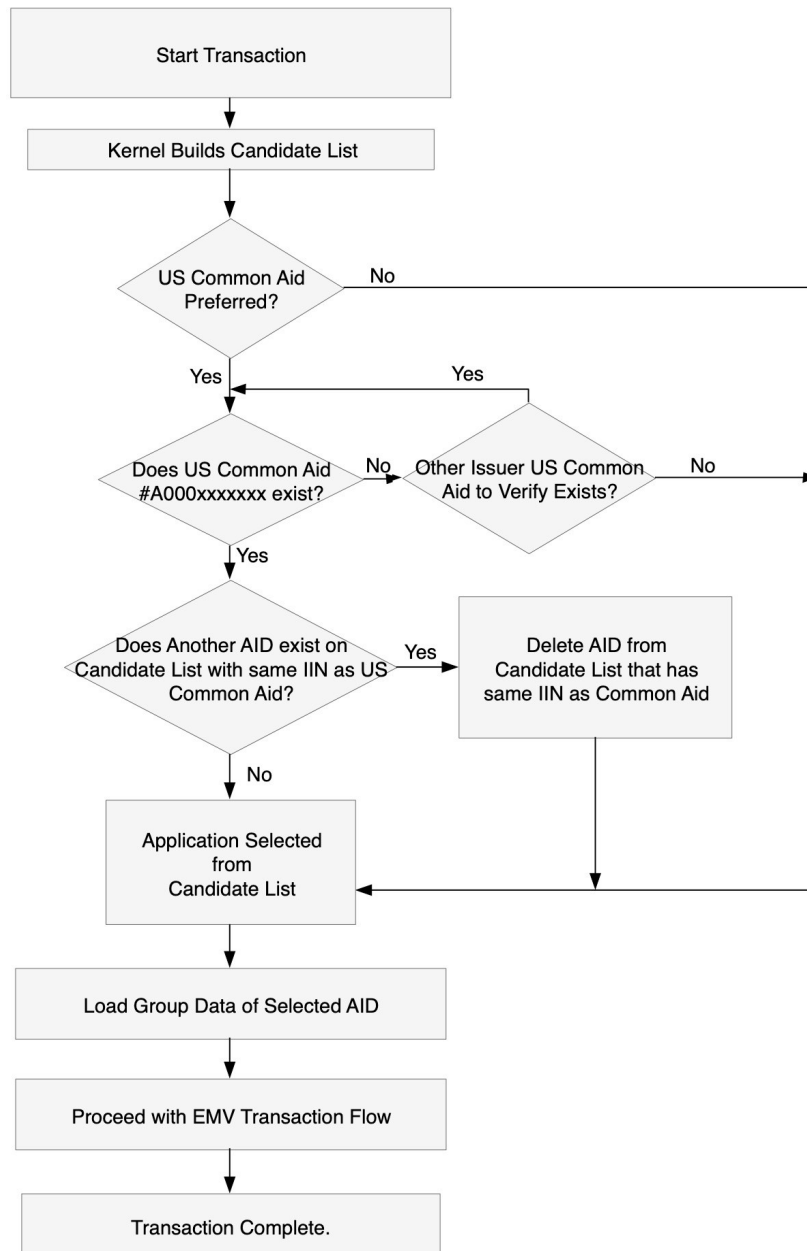
| | | |
|--|----|---|
| ▼ application #A000000031010 | | |
| ▼ answer to select | 63 | 6F3D8407A0000000031010A532500A564953412044454249548701015F2D02656E9F120A564953412044454249549F110101BF0C0A5F550255534203430665h |
| ▼ File Control Information (FCI) Template 6F | 61 | |
| Dedicated File (DF) Name 84 | 7 | > \240\000\000\000\003\020\020 (A0000000031010) |
| ▼ FCI Proprietary Template A5 | 50 | |
| Application Label 50 | 10 | > VISA DEBIT |
| Application Priority Indicator 87 | 1 | 01h |
| Language Preference 5F2D | 2 | > en |
| Application Preferred Name 9F12 | 10 | 56495341204445424954h |
| Issuer Code Table Index 9F11 | 1 | 01h |
| ▼ FCI Issuer Discretionary Data BFOC | 10 | |
| Issuer Country Code (alpha2 format) 5F55 | 2 | 5553h |
| Issuer authority 42 | 3 | 430665h |
| ▼ processing_options | 8 | 8006180008020400h |

27.6 Example FCI from US Common Debit

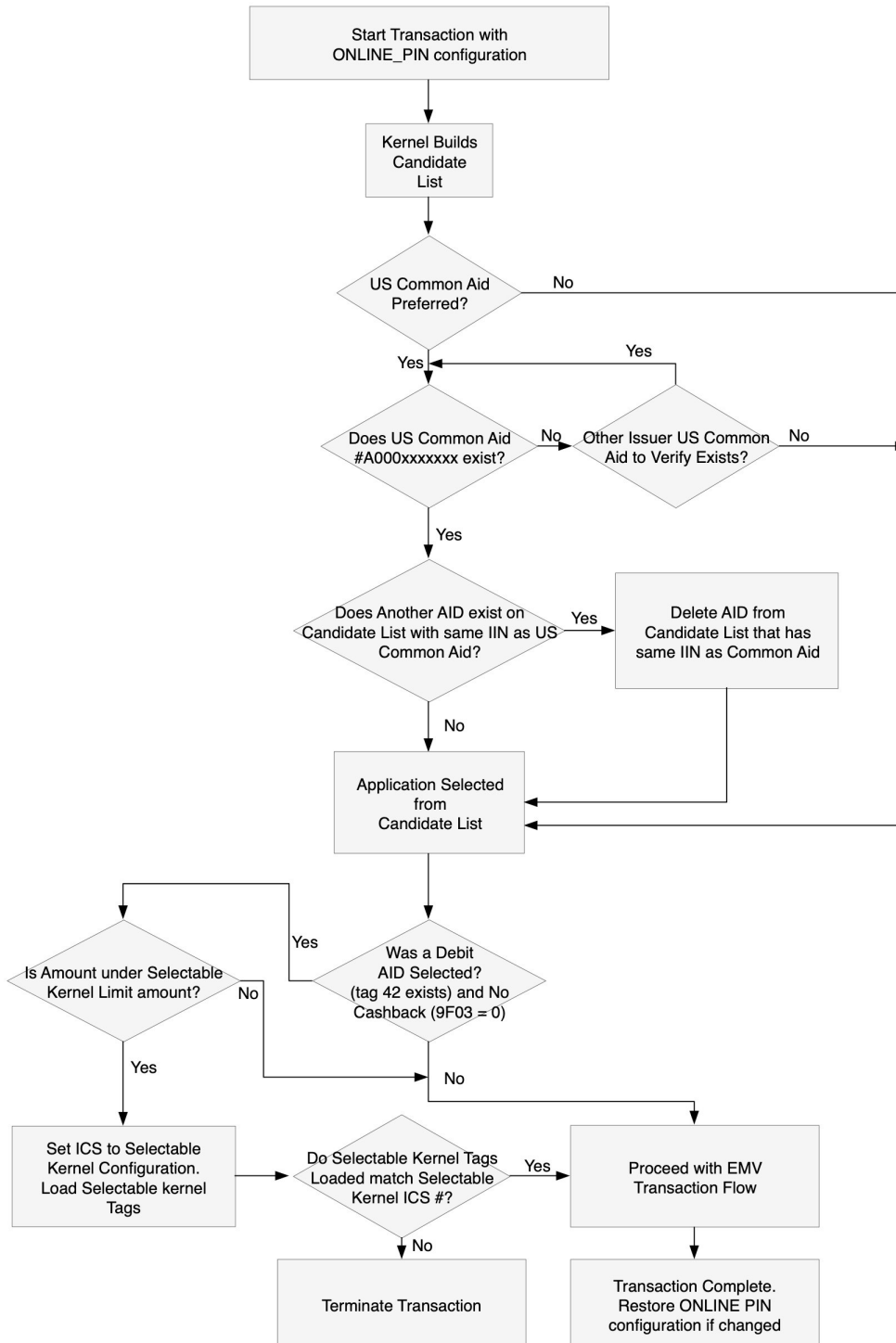
| | | |
|--|----|--|
| ▼ application #A0000000980840 | | |
| ▼ answer to select | 59 | 6F398407A00000000980840A52E500855532044454249548701025F2D02656E9F120855532044454249549F110101BF0C0A5F550255534203430665h |
| ▼ File Control Information (FCI) Template 6F | 57 | |
| Dedicated File (DF) Name 84 | 7 | > \240\000\000\000\230\0100 (A00000000980840) |
| ▼ FCI Proprietary Template A5 | 46 | |
| Application Label 50 | 8 | > US DEBIT |
| Application Priority Indicator 87 | 1 | 02h |
| Language Preference 5F2D | 2 | > en |
| Application Preferred Name 9F12 | 8 | 5553204445424954h |
| Issuer Code Table Index 9F11 | 1 | 01h |
| ▼ FCI Issuer Discretionary Data BFOC | 10 | |
| Issuer Country Code (alpha2 format) 5F55 | 2 | 5553h |
| Issuer authority 42 | 3 | 430665h |

27.7 Contactless Transaction Flow: Requirements for Common AID Preferring Configuration

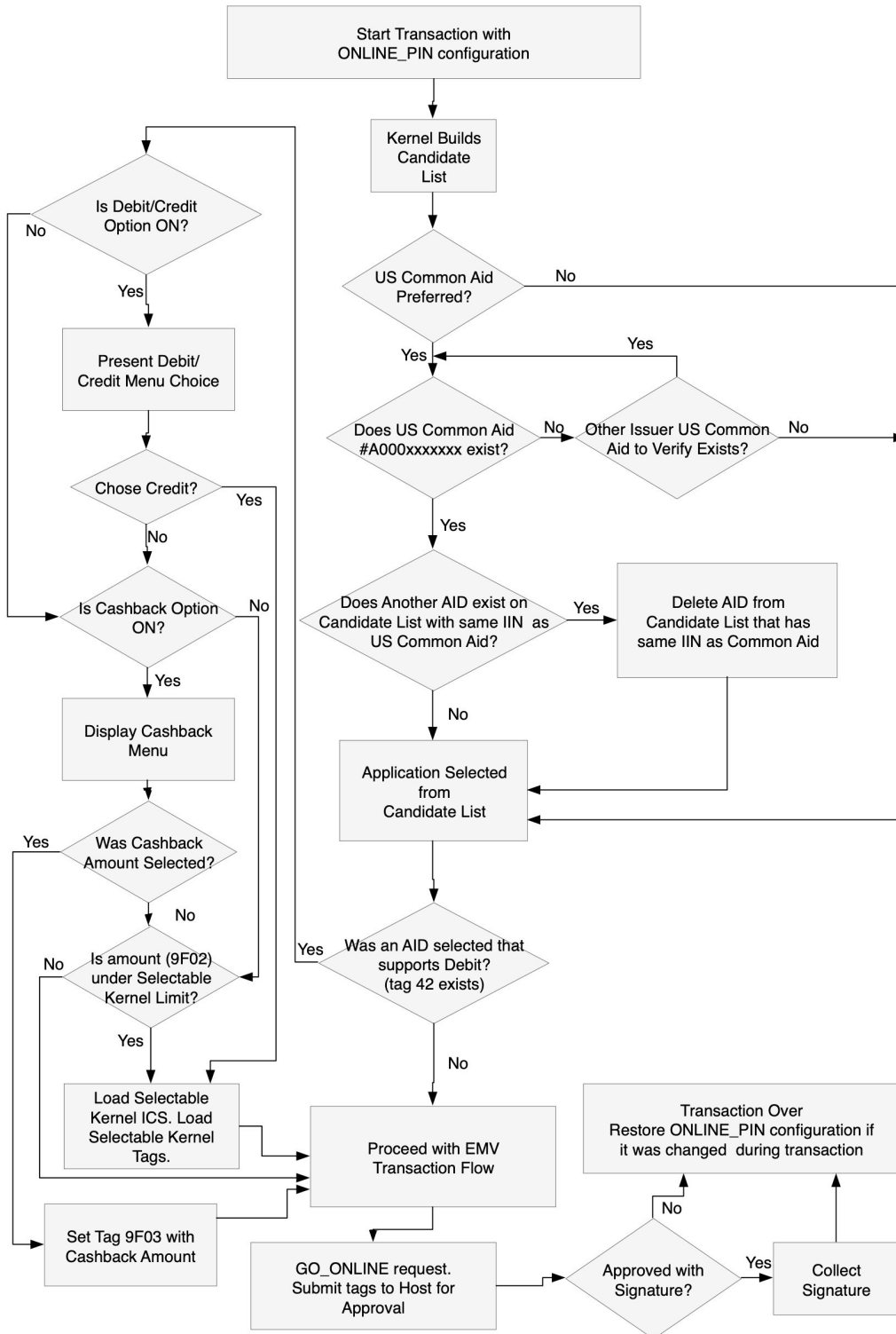
1. Define Configuration Group with Global AID Tags.
2. Load Global Aid to Reader, defined to use Global AID Configuration Group.
3. Define Configuration Group with Common AID Tags.
4. Load Common AID to Reader, defined to use Common AID Configuration Group.
5. Load Tag DFED48 to Configuration Group 0.



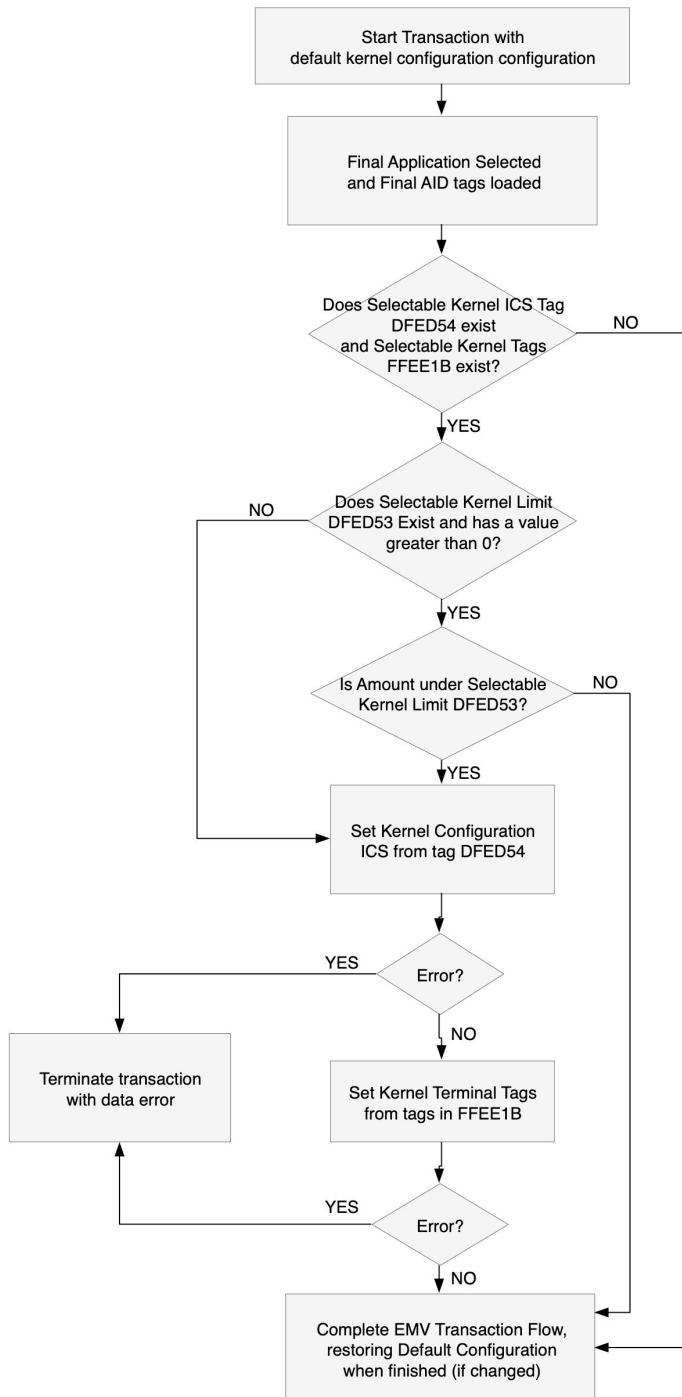
27.8 Contact Transaction Flow, No Credit/Debit/Cashback Option Selectable Kernel Configuration = NO CVM kernel



27.9 Contact Transaction Flow, Credit/Debit/Cashback Options Selectable Kernel Configuration = NO CVM kernel



27.10 Selectable Kernel Flowchart – Contact Card Selectable Kernel Configuration - Any Configuration



Appendix A.17: FastEMV/Deferred Approval

FastEMV speeds up checkout times on chip transactions at the point of sale and optimizes the consumer experience. Deferred Approval allows the host device to override the DECLINE message with APPROVAL at transaction conclusion.

ViVOpay readers use tag DFED46 to enable or disable the FastEMV/Deferred Approval function. The FastEMV/Deferred Approval function is composed of two items:

- Auto complete
- Display message on LCD

One-byte tag DFED46 definition

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|----------|----------|----------|----------|-----------------|-------|-------|---------------|
| Reserved | Reserved | Reserved | Reserved | Message Display | | | Auto complete |

Bit 0:

- 0 => normal operation
- 1 => Auto Complete

27.11 Auto Complete Flow

1. Auto Authenticate
2. Don't return 1st Gen AC GO_ONLINE result/tags to host device
3. Don't display GO_ONLINE on LCD
4. Auto Complete with "Z3"
5. Return 2nd Gen AC results/tags to host device

27.12 Display message on LCD

If bit 0 is equal to 1, then evaluate the bit1~bit3.

| | |
|--------|---|
| Bit 1: | 0 = Display "Authorizing, please wait" on the LCD until the reader receives another command from the POS to change the LCD message for final decision. 1 = Display "APPROVED" as the final result on the LCD, followed by "REMOVE CARD" if Bit2 is set; otherwise followed by the idle state message if Bit2 is not set. |
| Bit 2: | 0 = Don't display "REMOVE CARD" before the final result. 1 = Display "REMOVE CARD" until the card is removed, followed by the idle state message after the card is removed. |
| Bit 3: | 0 = Reader does not beep even if the card is inserted. 1 = Reader beeps until the card is removed. |

27.13 Assumptions

If Auto Complete is selected but the result is APPROVED OFFLINE or DECLINE OFFLINE, the result displays on the LCD after First Gen AC.

Set the DFED46 tag in Host Configuration, Start Transaction, or AID.

- If set in Host Configuration, DFED46 applies to all transaction.
- If set in Start Transaction, DFED46 overrides any setting in Host Configuration and is for that transaction only.
- If set in a matching AID file, DFED46 overrides any setting in Host Configuration and Start Transaction and is for that transaction only.

27.14 Required Tags

In compliance with EMV Specification Bulletin No. 220, section 8.5.4: Kernel Output, the kernel is capable of outputting the latest value for the following data for FastEMV:

- 9F02: Amount, Authorized (Numeric)
- 9F03: Amount, Other (Numeric)
- 9F26: Application Cryptogram
- 5F25: Application Effective Date
- 5F24: Application Expiration Date
- 82: Application Interchange Profile
- 50: Application Label
- 5A: Application PAN
- 5F34: Application PAN Sequence Number
- 9F12: Application Preferred Name
- 9F36: Application Transaction Counter
- 9F07: Application Usage Control
- 9F08, 9F09: Application Version Number (Kernel)
- 5F20: Cardholder Name
- 9F0B: Cardholder Name (Extended)
- 9F27: Cryptogram Information Data
- 8E: CVM List
- 9F34: CVM Results
- 84: DF Name
- 9F1E: Interface Device Serial Number
- 9F0D, 9F0E, 9F0F: Issuer Action Codes
- 9F10: Issuer Application Data
- 9F11: Issuer Code Table Index
- 5F28: Issuer Country Code
- 5F2D: Language Preference
- 9F15: Merchant Category Code
- 9F16: Merchant Identifier
- 9F24: Payment Account Reference

- 9F33: Host Capabilities
- 9F1A: Host Country Code
- 9F1C: Host Identifier
- 9F35: Host Type
- 95: Host Verification Results
- 57: Track 2 Equivalent Data
- 9F53: Transaction Category Code
- 5F2A: Transaction Currency Code
- 9A: Transaction Date
- 9F21: Transaction Time
- 9C: Transaction Type
- 9F37: Unpredictable Number

In addition to the tags specified by the bulletin, the following tags are also included in FastEMV output:

- 9F20: Track2 Discretionary Data
- 4F: ADF Name
- DFEE23: MSR Data
- 9F13: Last Online Application Transaction Counter(ATC) Register
- 9B: Transaction Status Information
- 99: PIN Block
- 9F5B: Issuer script results
- DF21: Issuer script results
- 9F39: POS Entry Mode
- 8A: Authorization Response Code
- DFEE51: PIN Block KSN

28. Appendix A.18: VP5300M Commands

The following commands are used exclusively with the VP5300M. These commands only work on motorized readers.

28.1 Motor Eject Card (60-19)

The **Motor Eject Card** command allows a motorized reader to eject the card at the end of a transaction.

Note: This command is exclusive to motorized readers.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-15 | Byte 16 | Byte 17 |
|-------------------------------|---------|-------------|-------------------|-------------------|------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViV0tech2\0 | 60h | 19 | 00 | | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | 60h | See status code table | 00 | | | | |

28.2 Motor Clean (60-1A)

The **Motor Clean** command allows a motorized reader to perform the roller cleaning function.

Note the following:

- This command must be used with a special cleaning card.
- This command is exclusive to readers with motor units.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-15 | Byte 16 | Byte 17 |
|-------------------------------|---------|-------------|-------------------|-------------------|------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViV0tech2\0 | 60h | 1A | 00 | | | | |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|-------------------------|-------------------------|----------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | 60h | See status code table | 00 | | | | |

29. Appendix A.19: New Security Log Event IDs

The security log system event classes and event type IDs are defined as follows:

Event Class

| Name | ID | Note |
|----------------------|-----|------|
| EVENT_CLASS_SECURITY | 0 | |
| EVENT_CLASS_TAMPER | 1 | |
| EVENT_CLASS_UMFG | 2 | |
| EVENT_CLASS_SMFG | 3 | |
| EVENT_CLASS_PKI | 4 | |
| EVENT_CLASS_SMSG | 5 | |
| EVENT_CLASS_APP | 6 | New |
| EVENT_CLASS_KERNEL | 7 | New |
| EVENT_CLASS_SYSTEM | 8 | New |
| EVENT_CLASS_ALL | 255 | |

Event Type

| Name | ID | Note |
|---|-----|--|
| EVENT_TYPE_SECURITY_CLEARED | 0 | |
| EVENT_TYPE_SECURITY_SENSOR_REMOVE_FLAG_CLEARED | 1 | |
| EVENT_TYPE_SECURITY_SENSOR_REMOVE_FLAG_SET | 2 | |
| EVENT_TYPE_SECURITY_MANUFACTURE_SYS_DATA_LOADED | 3 | |
| EVENT_TYPE_SECURITY_MAXQ_FW_WRITTEN | 4 | |
| EVENT_TYPE_SECURITY_K21_FW_WRITTEN | 5 | |
| EVENT_TYPE_SECURITY_ROOT_CA_RELOADED | 6 | |
| EVENT_TYPE_SECURITY_TM4_FW_WRITTEN | 7 | New |
| EVENT_TYPE_SECURITY_K81_FW_WRITTEN | 8 | New |
| EVENT_TYPE_SECURITY_ALL | 255 | New |
| EVENT_TYPE_TAMPER_ACTIVE | 0 | |
| EVENT_TYPE_TAMPER_DEACTIVE | 1 | |
| EVENT_TYPE_TAMPER_GENERIC | 2 | |
| EVENT_TYPE_TAMPER_ACK | 3 | |
| EVENT_TYPE_TAMPER_TIMEOVRF | 4 | |
| EVENT_TYPE_TAMPER_MONOTONICOVRF | 5 | |
| EVENT_TYPE_TAMPER_VOLT | 6 | |
| EVENT_TYPE_TAMPER_CLK | 7 | |
| EVENT_TYPE_TAMPER_TEMP | 8 | |
| EVENT_TYPE_TAMPER_FLASH | 9 | |
| EVENT_TYPE_TAMPER_TST | 10 | |
| EVENT_TYPE_TAMPER_PIN | 11 | Use event parameter[0] to indicate tamper pin number |
| EVENT_TYPE_TAMPER_BAT | 12 | |
| EVENT_TYPE_TAMPER_ALL | 255 | New |

| | | |
|---|-----|-----|
| EVENT_TYPE_UMFG_DEVICE_RESET | 0 | |
| EVENT_TYPE_UMFG_LOG_FILE_CREATED | 1 | |
| EVENT_TYPE_UMFG_SAFTY_FILE_CREATED | 2 | |
| EVENT_TYPE_UMFG_LOG_FILE_UNLOCKED | 3 | |
| EVENT_TYPE_UMFG_SAFTY_FILE_INITIALIZED | 4 | |
| EVENT_TYPE_UMFG_UTC_STARTED | 5 | |
| EVENT_TYPE_UMFG_RSA_PRIV_KEY_CREATED | 6 | |
| EVENT_TYPE_UMFG_RSA_PUB_KEY_CREATED | 7 | |
| EVENT_TYPE_UMFG_SN_COMMON_NAME_TO_CSR_STAMPED | 8 | |
| EVENT_TYPE_UMFG_DEVICE_ACTIVATED | 9 | |
| EVENT_TYPE_UMFG_DEVICE_LOCKED | 10 | |
| EVENT_TYPE_UMFG_DEVICE_UNLOCKED | 11 | |
| EVENT_TYPE_UMFG_FW_CHECK_VALUE | 12 | |
| EVENT_TYPE_UMFG_SET_DEK_DATA_KEY_VARIANT | 13 | |
| EVENT_TYPE_UMFG_SET_DEK_PIN_KEY_VARIANT | 14 | |
| EVENT_TYPE_UMFG_SET_SN | 15 | |
| EVENT_TYPE_UMFG_SET_SALT | 16 | |
| EVENT_TYPE_UMFG_SECURED_TASK_START | 17 | |
| EVENT_TYPE_UMFG_SECURED_TASK_END | 18 | |
| EVENT_TYPE_UMFG_CERT_ROOT_CA_LOADED | 19 | |
| EVENT_TYPE_UMFG_CERT_ROOT_CA_VALIDATED | 20 | |
| EVENT_TYPE_UMFG_CERT_ROOT_CA_CLEARED | 21 | |
| EVENT_TYPE_UMFG_ALL | 255 | New |
| EVENT_TYPE_SMFG_MASTER_DEVICE_RESET | 0 | |
| EVENT_TYPE_SMFG_DEVICE_UNLOCKED | 1 | |
| EVENT_TYPE_SMFG_SET_UTC | 2 | |
| EVENT_TYPE_SMFG_CERTIFICATE_LOAD | 3 | |
| EVENT_TYPE_SMFG_CERTIFICATE_DELETE | 4 | |
| EVENT_TYPE_SMFG_DELETE_ALL_KEYS | 5 | |
| EVENT_TYPE_SMFG_SET_CRL | 6 | |
| EVENT_TYPE_SMFG_REMOVE_CRL | 7 | |
| EVENT_TYPE_SMFG_REMOVE_ALL_KEYS | 8 | |
| EVENT_TYPE_SMFG_REMOVE_SENSOR | 9 | |
| EVENT_TYPE_SMFG_SET_SALT | 10 | |
| EVENT_TYPE_SMFG_SET_SN | 11 | |
| EVENT_TYPE_SMFG_SET_WHITE_LIST | 12 | |
| EVENT_TYPE_SMFG_CLR_WHITE_LIST | 13 | |
| EVENT_TYPE_SMFG_SET_KEY_PARAMETER | 14 | |
| EVENT_TYPE_SMFG_SET_PASS_THRU_WL | 15 | |
| EVENT_TYPE_SMFG_CLR_PASS_THRU_WL | 16 | |
| EVENT_TYPE_SMFG_SET_DEK_DATA_KEY_VARIANT | 17 | |
| EVENT_TYPE_SMFG_SET_DEK_PIN_KEY_VARIANT | 18 | |
| EVENT_TYPE_SMFG_CERTIFICATE_LOADED | 19 | |
| EVENT_TYPE_SMFG_CERTIFICATE_DELETED | 20 | |

| | | |
|---------------------------------------|-----|-----|
| EVENT_TYPE_SMFG_ALL | 255 | New |
| EVENT_TYPE_PKI_GET_DEVICE_CERTIFICATE | 0 | |
| EVENT_TYPE_PKI_PEDK_VALIDATED | 1 | |
| EVENT_TYPE_PKI_PEDK_KEY_REQUEST | 2 | |
| EVENT_TYPE_PKI_PEDK_KEY_LOADED | 3 | |
| EVENT_TYPE_PKI_PEDK_KEY_VERIFIED | 4 | |
| EVENT_TYPE_PKI_ALL | 255 | New |
| EVENT_TYPE_SMSG_ACTIVATE_REMOVAL | 0 | |
| EVENT_TYPE_SMSG_SET_WHITE_LIST | 1 | |
| EVENT_TYPE_SMSG_CLR_WHITE_LIST | 2 | |
| EVENT_TYPE_SMSG_SET_PASS_THRU_WL | 3 | |
| EVENT_TYPE_SMSG_CLR_PASS_THRU_WL | 4 | |
| EVENT_TYPE_SMSG_ALL | 255 | New |
| EVENT_TYPE_APP_UPGRADE_START | 0 | New |
| EVENT_TYPE_APP_UPGRADE_SUCCESS | 1 | New |
| EVENT_TYPE_APP_UPGRADE_ABORT | 2 | New |
| EVENT_TYPE_APP_UPGRADE_ERROR | 3 | New |
| EVENT_TYPE_APP_REMOVE_SUCCESS | 4 | New |
| EVENT_TYPE_APP_REMOVE_ERROR | 5 | New |
| EVENT_TYPE_APP_VERIFY_SUCCESS | 6 | New |
| EVENT_TYPE_APP_VERIFY_ERROR | 7 | New |
| EVENT_TYPE_APP_ALL | 255 | New |
| EVENT_TYPE_KERNEL_INSTALL | 0 | New |
| EVENT_TYPE_KERNEL_INSTALL_SUCCESS | 1 | New |
| EVENT_TYPE_KERNEL_INSTALL_ERROR | 2 | New |
| EVENT_TYPE_KERNEL_REMOVE | 3 | New |
| EVENT_TYPE_KERNEL_ALL | 255 | New |
| EVENT_TYPE_SYSTEM_FW_START | 0 | New |
| EVENT_TYPE_SYSTEM_FW_UPDATE_START | 1 | New |
| EVENT_TYPE_SYSTEM_FW_UPDATE_SUCCESS | 2 | New |
| EVENT_TYPE_SYSTEM_FW_UPDATE_ERROR | 3 | New |
| EVENT_TYPE_SYSTEM_BL_UPDATE_START | 4 | New |
| EVENT_TYPE_SYSTEM_BL_UPDATE_SUCCESS | 5 | New |
| EVENT_TYPE_SYSTEM_BL_UPDATE_ERROR | 6 | New |
| EVENT_TYPE_SYSTEM_ALL | 255 | New |

30. Appendix A.20: SREDKey2 Commands

IDG Command and FunID (ITP Protocol) Comparative Table

| IDG Command | FunID (ITP protocol) |
|-------------|---------------------------------------|
| D1-30/31 | 23h, 14h, AEh, 25h, 5Dh and 24h |
| C7-44/45 | 13h |
| C7-80/81 | Remainder FunIDs (TLV format setting) |
| C7-84/85 | B0h, B1h |

Tag and FunID (ITP Protocol) Comparative Table

| Tag | FunID (ITP protocol) | Comments |
|--------|-------------------------|--|
| DEEF04 | 84h | Set: C7-38 command Get: C7-39 command |
| DFEE7D | 6Fh | |
| DFEE1D | 49h, 4Ah, 4Bh, 50h, 60h | |
| DFEC09 | 17h, 21h | Separator & Terminate |
| DFEC01 | 34h~39h | Track Prefix & Suffix |
| DFED6F | D2h, D3h | Preamble & Postamble |
| DFED70 | 61h~6Eh | Track StartChar & EndChar |
| DFED71 | 19h | Sentinel & Account number control |
| DFED72 | 1Dh | Decoding Direction |
| DFED73 | 55h | Include Mod10 check digit |
| DFEC02 | 86h | Clear / mask data option |
| DFEC03 | 89h | Track 3 expiration date position offset |
| DFEC04 | 90h | Non-financial card encryption options |
| DFEC05 | 85h | MSR output data options |
| DFEC06 | 8Fh(bit0) | Manual PAN Entry output data options |
| DFEC07 | 8Dh, 8E, 8F | Manual PAN Entry base configuration |
| DFEC08 | 77h (Bit3) | XML Output Control |

30.1 MSR and Manual PAN Entry Configuration Related Commands

30.1.1 Set MSR / Manual PAN Entry Configuration with TLV (C7-80)

The **Set MSR / Manual PAN Entry Configuration with TLV** command sets a reader's MSR-related configuration. **FunID** (ITP Protocol) is defined by TLV.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14- Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|-----------------|-------------------------|-------------------------|------------------------|--------------|-----------|
| Header Tag & Protocol Version | Command | Sub- Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOpayV2\0 | C7h | 80h | xx | xx | See send data field | | |

Data Table

| Data Item | Length | Description |
|----------------|--------|--|
| Default or Set | 1 | 0: Default All 1: Set by TLV blocks |
| TLV blocks | n | If "Default or Set" == 0, this area does not exist If "Default or Set" == 1, this area exist, please see note |

Note: See below for TLV design information and Tag & FunID (ITP protocol) comparison tables.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------------------------------------|-------------------|-------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | CRC (MSB) | CRC (LSB) |
| ViVOPayV2\0 | C7h | See status code table | 00h | 00h | | |

30.1.2 Get MSR / Manual PAN Entry Configuration with Tag (C7-81)

The **Get MSR / Manual PAN Entry Configuration with Tag** command retrieves the reader's current MSR / Manual PAN entry configuration.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14-Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|---------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOPayV2\0 | C7h | 81h | xx | xx | See send data field | | |

Data Table

| Data Item | Length | Description |
|-----------|--------|-----------------|
| Tags | n | please see note |

Note: See below for TLV design information and Tag & FunID (ITP protocol) comparison tables.

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14- Byte 13+n | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---|----------------------|----------------------|-----------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViVOpayV2\0 | C7h | See status code table | xx | xx | See Data table | | |

Data Table

| Data Item | Length | Description |
|------------|--------|-----------------|
| TLV blocks | n | please see note |

Note: See below for TLV design information and Tag & FunID (ITP protocol) comparison tables.

30.2 TLV Format for MSR and Manual PAN Entry**30.2.1 Separator & Terminate**

| Tag | Length | Value |
|--------|--------------------|---|
| DFEC09 | 1 byte: value is 2 | 2 bytes |
| | | First byte: Separator CR for RS232, Enter for KB any character supported except 00 which means none Default: 0x0D=CR/Enter |
| | | Second byte: Terminate CR for RS232, Enter for KB any character supported except 00 which means none Default: 0x0D=CR/Enter |
| | | |

30.2.2 Track Prefix & Suffix

| Tag | Length | Value |
|--------|-------------------------|--|
| DFEC01 | 1 byte: value is 6 ~ 42 | 6 Blocks: <1 bytes Block Len> [<n bytes Block Data>] |
| | | Track 1 Prefix Length (1 byte): 0 or 1-6 (Default 0) |
| | | If Track 1 Prefix Length is not 0, the field exist: Track 1 Prefix (1~ 6 bytes) |
| | | Track 2 Prefix Length (1 byte): 0 or 1-6 (Default 0) |
| | | If Track 2 Prefix Length is not 0, the field exist: Track 2 Prefix (1~ 6 bytes) |
| | | Track 3 Prefix Length (1 byte): 0 or 1-6 (Default 0) |
| | | If Track 3 Prefix Length is not 0, the field exist: Track 3 Prefix (1~ 6 bytes) |
| | | Track 1 Suffix Length (1 byte): 0 or 1-6 (Default 0) |

30.2.3 Preamble & Postamble

| Tag | Length | Value |
|--------|-------------------------|--|
| DFED6F | 1 byte: value is 2 ~ 32 | 2 Blocks: <1 bytes Block Len> [<n bytes Block Data>] |
| | | Preamble Length (1 byte): 0 or 1-15 (Default 0) |

| | | |
|--|--|---|
| | | If Preamble Length is not 0, the field exist: Preamble (1~ 15 bytes) |
| | | Postamble Length (1 byte): 0, 1-15 (Default 0) |

30.2.4 Track StartChar & EndChar

| Tag | Length | Value |
|--------|---------------------|---|
| DFED70 | 1 byte: value is 12 | 12 bytes |
| | | Track 1 7 Bit Start Char '%' (any) |
| | | T15B Start ';' (any) |
| | | Track 2 7 Bit Start Char '%' (any) |
| | | T25BStart ';' (any) |
| | | Track 3 7 Bit Start Char '%' (any) |
| | | T35BStart ';' (any) |
| | | Track 1 End Sentinel '?' (any) |
| | | Track 2 End Sentinel '?' (any) |
| | | Track 3 End Sentinel '?' (any) |
| | | Track 1 error code '%' (any) |
| | | Track 2 error code ';' (any) |
| | | Track 3 error code '+' (any) |

30.2.5 Sentinel & Account Number Control

| Tag | Length | Value |
|--------|--------------------|--|
| DFED71 | 1 byte: value is 1 | 1 byte |
| | | Bit 0: 1: Send start/end sentinel, 0: Not send start/end sentinel |
| | | Bit1: 1: Only send account number on Track 2, 0: Send all data on Track 2 |
| | | Bit2: 1: Send error notification, 0: Not error notification |
| | | Bit3: 1: Alt Key Output, 0: Control Key Output |
| | | |

30.2.6 Decoding Direction

| Tag | Length | Value |
|--------|--------------------|---|
| DFED72 | 1 byte: value is 1 | 1 byte |
| | | 30h: Raw Data Decoding in Both Directions. |
| | | 0x31: Decode in Both directions. |
| | | 32h: Moving Stripe Along Head in Direction of Encoding |
| | | 33h: Moving Stripe Along Head Against Direction of Encoding |

30.2.7 Include Mod10 check digit

| Tag | Length | Value |
|--------|--------------------|--------|
| DFED73 | 1 byte: value is 1 | 1 byte |

| | | |
|--|--|----------------------------|
| | | 30h: Do not include mod10. |
| | | 31h: Display mod10 |
| | | 32h: Display wrong mod10. |

30.2.8 Sync Bit Check

Checking for track sync bits can allow readers to read poorly-encoded cards.

| Tag | Length | Value |
|--------|--------------------|--|
| FFEE1B | 1 byte: value is 1 | 1 byte |
| | | 30h: 13 bits. |
| | | 31h: 13 bits, but allow if valid through track LRC |
| | | 32h: 9 bits ABA; 13 bits IATA; 16 bits JIS. |

30.2.9 Clear / Mask Data Option

| Tag | Length | Value |
|--------|--------------------|---|
| DFEC02 | 1 byte: value is 1 | 1 byte |
| | | Bit 0: 1: send clear/mask tk1; 0: Do not send |
| | | Bit 1: 1: send clear/mask tk2; 0: Do not send |
| | | Bit 2: 1: send clear/mask tk3; 0: Do not send |
| | | |
| | | |

30.2.10 Track 3 Expiration Date Position Offset

| Tag | Length | Value |
|--------|--------------------|----------------------------|
| DFEC03 | 1 byte: value is 1 | 1 byte |
| | | 34h: 4th |
| | | 36h: 6 th |

30.2.11 Non-Financial Card Encryption Options

| Tag | Length | Value |
|--------|--------------------|---|
| DFEC04 | 1 byte: value is 1 | 1 byte |
| | | 30h: non-financial card output plaintext while DEK was loaded |
| | | 31h: non-financial card output as financial card while DEK was loaded |

30.2.12 MSR Output Data Options

| Tag | Length | Value |
|--------|--------------------|-----------------------------------|
| DFEC05 | 1 byte: value is 1 | 1 byte |
| | | 30h: ID TECH Original Data Format |
| | | 31h: Enhanced Data Format |

30.2.13 Manual PAN Entry Output Data Options

| Tag | Length | Value |
|-----|--------|-------|
|-----|--------|-------|

| | | |
|--------|--------------------|----------------------------------|
| DFEC06 | 1 byte: value is 1 | 1 byte |
| | | 30h: Original Data Format |
| | | 31h: Enhanced Data Format |

30.2.14 Manual PAN Entry Base Configuration

| Tag | Length | Value |
|--------|--------------------|---|
| DFEC07 | 1 byte: value is 3 | 3 bytes |
| | | Byte 1: Admin Level <ul style="list-style-type: none">• 0Bh: Admin 1• 15h: Admin 2• 1Fh: Admin 3• 29h: Admin 4• 33h: Admin 5• 3Dh: Admin 6 |
| | | Byte 2: Key Timeout Value Range: 1~225 |
| | | Byte 3: Manual PAN Entry Option |
| | | Bit 00: output in original keyed output 1: output in enhanced keyed-in output |
| | | Bit 10: allow empty CVV entry 1: require 3 or more CVV digits |
| | | Bit 20: allow empty ZIP entry 1: require 5 or more ZIP digits |
| | | Bit30: allow empty ADR entry 1: require 1 or more ADR digits |
| | | Bit40: do mod-10 check on keyed-in PAN 1: don't check PAN mod-10 |
| | | Bit 50: "Admin" enable 1: "Admin" disable |
| | | Bit 6-7 reserved all zero |

Note:

Admin 1: Card Number, Expiration Date Configuration

Admin 2: Card Number, Expiration Date, Zip Code Configuration

Admin 3: Card Number, Expiration Date, Street Number of the Address, Zip Code Configuration

Admin 4: Card Number / Expiration Date / Security Code / Zip Code

Admin 5: Card Number / Expiration Date / Security Code / Street Number / Zip Code

Admin 6: Card Number / Expiration Date / Security Code

30.2.15 32.3.15 XML output control

Set output format to XML, valid for both MSR and Manual PAN Entry

| Tag | Length | Value |
|--------|--------------------|---|
| DFEC08 | 1 byte: value is 1 | 1 byte |
| | | 30h: Close XMLL format, the actual output format is set by Tag FEE1F~FFEE23. |
| | | 31h: Set output to XML format, the Tag FFEE1F and FFEE23 setting is invalid. |

31. Appendix A.21: Transact Credentials over ECP2.0

31.1 Key Loading

31.1.1 Set Card Keys (C7-67)

The **Set Card Keys** command sets the keys to use for decryption.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------|---------|-------------|-------------------|-------------------|-------------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data length (MSB) | Data length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | C7h | 67h | Variable | Variable | See Data Objects below | | |

Data Objects

PublicKeyEncrypt (PlainKeySet) + SHA-256 (PlainKeySet)

PlainKeySet is composed of the following data,

<1-byte Amount>

<1-byte Key Index> <2-byte Key Length> < Variable bytes Key Value>

....

<1-byte Key Index> <1-byte Key Length> < Variable bytes Key Value>

| Data Field | Length (bytes) | Description |
|------------|----------------|--|
| Amount | 1 | The number of Keys |
| Key Index | 1 | The type of Key, refer to Key Table listed in Delete Keys (C7-6A) |
| Key Length | 2 | <LenMSB> <LenLSB> indicate the Key length |
| Key Value | var | The Key data |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------|-------------------|-------------------|----------|----------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | CRC(MSB) | CRC(LSB) |
| ViV0tech2\0 | C7h | 00h | 00 | 00 | 96h | 12h |

Note: If the Status Code does not return 00h, refer to the status code table.

31.1.2 Delete Keys (C7-69)

The **Delete Keys** command deletes the currently set decryption Keys.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|---------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | 69h | 00h | 01h | See Key Table below | | |

Key Table

| Item | Index | Description |
|-------------|-------|--|
| Card Key | 0 | All keys imported by command C7-67 |
| Private Key | 1 | The modulus and private exponent imported by command C7-6C |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------|---------|---------|-------------------|-------------------|----------|----------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | CRC(MSB) | CRC(LSB) |
| ViVOtech2\0 | C7h | 00h | 00 | 00 | 96h | 12h |

Note: If the Status Code does not return 00h, refer to the status code table.

31.1.3 Get Hash of Keys (C7-6A)

The **Get Hash of Keys** command gets the hash of the currently set keys.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 | Byte 16 |
|-------------------------------|---------|-------------|-------------------|-------------------|---------------------|-----------|-----------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | 6Ah | 00h | 01h | See Key Table below | | |

Key Table

| Item | Index | Description |
|-------------|-------|--|
| Card Key | 0 | All keys imported by command C7-67 |
| Private Key | 1 | The modulus and private exponent imported by command C7-6C |

Response Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte 14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|---------|----------------------|----------------------|----------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Status | Data length (MSB) | Data length (LSB) | Data | CRC (MSB) | CRC (LSB) |
| ViV0tech2\0 | C7h | 00h | 00h | 20h | SHA-256 | 96h | 12h |

Note: Status Code 4Bh indicates there is no key in the reader.

31.1.4 Get the Next DEK KSN (C7-6B)

The **Get the Next DEK KSN** command retrieves the next DEK KSN to use.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|-------------------------------------|---------|-------------|-------------------------|-------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
| ViV0tech2\0 | C7h | 6Bh | 00h | 00h | F4h | EDh |

Response

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte23 | Byte 24 | Byte 25 |
|-------------------------------------|---------|-------------|-------------------------|-------------------------|-----------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViV0tech2\0 | C7h | 00h | 00h | 0Ah | Key Serial Number | | |

Note: If the Status Code does not return 00h, refer to the status code table.

31.1.5 Set KTK Private Key (C7-6C)

The **Set KTK Private Key** command sets the KTK Private Key.

Command Frame

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 ... Byte14+n-1 | Byte 14+n | Byte 15+n |
|-------------------------------------|---------|----------------|-------------------------|-------------------------|------------------------------|--------------|--------------|
| Header Tag & Protocol Version | Command | Status Code | Data Length (MSB) | Data Length (LSB) | Data | CRC (LSB) | CRC (MSB) |
| ViVOtech2\0 | C7h | 6Ch | Variable | variable | See data objects below | | |

Data Objects

$KSN + SessionKeyEncrypt(PlainPrivateKey) + SHA-256(PlainPrivateKey)$

PlainPrivateKey is composed of the following data:

<1-byte Amount>

<1-byte Key Index> <2-byte Key Length> < Variable bytes Key Value>

<1-byte Key Index> <2-byte Key Length> < Variable bytes Key Value>

Note: The private key is composed of the modulus and private exponent. For safety, these two items must be included in a single command. In other words, the amount field is always two.

| Data Field | Length (bytes) | Description |
|------------|----------------|---|
| Amount | 1 | The number of keys |
| Key Index | 1 | The type of key: Index 0: The modulus <i>n</i> of RSA-2048 private key Index 1: The private exponent <i>d</i> of RSA-2048 private key |
| Key Length | 2 | <LenMSB> <LenLSB> indicate the length of the key |
| Key Value | 256 | The key data |
| KSN | 10 | The value from the command (C7-6B) to derive the session key |

The encrypted part of the data object is always encrypted with a session key derived from the IPEK and KSN. The encryption algorithm is AES-CBC. Padding consists of null bytes (0x00) added to the end of the data, to give data a length that is a multiple of 16.

Response

| Byte 0-9 | Byte 10 | Byte 11 | Byte 12 | Byte 13 | Byte 14 | Byte 15 |
|----------|---------|---------|---------|---------|---------|---------|
|----------|---------|---------|---------|---------|---------|---------|

| Header Tag & Protocol Version | Command | Sub-Command | Data Length (MSB) | Data Length (LSB) | CRC (LSB) | CRC (MSB) |
|-------------------------------|---------|-------------|-------------------|-------------------|-----------|-----------|
| ViVOtech2\0 | C7h | 00h | 00h | 00h | 96h | 12h |

Note: If the Status Code does not return 00h, refer to the status code table.

31.2 Output Configuration

31.2.1 Tag DFEF4B

Tag DFEF4B configures what the reader receives in tag DFEF4D. The Transact Credential card number is one of the options.

Byte 1:

| 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | NOTES |
|---|---|---|---|---|---|---|---|---|
| - | - | - | - | - | - | - | X | 0 – Disable Track 3 Sentinels 1 – Enable Track 3 Sentinels |
| - | - | - | - | - | - | X | - | 0 – Disable Track 2 Sentinels 1 – Enable Track 2 Sentinels |
| - | - | - | - | - | X | - | - | 0 – Disable Track 1 Sentinels 1 – Enable Track 1 Sentinels |
| - | - | - | - | X | - | - | - | 0 – Disable Track 3 1 – Enable Track 3 |
| - | - | - | X | - | - | - | - | 0 – Disable Track 2 1 – Enable Track 2 |
| - | - | X | - | - | - | - | - | 0 – Disable Track 1 1 – Enable Track 1 |
| - | X | - | - | - | - | - | - | 0 – Disable PAN 1 – Enable PAN |
| X | - | - | - | - | - | - | - | 0 – All Data Elements Found 1 – Only First Element Found |

Byte 2: RFU

Byte 3:

| 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | NOTES |
|---|---|---|---|---|---|---|---|---|
| - | - | - | - | - | - | - | X | 0 – Disable EMV / MSD card data format switch 1 – Enable EMV / MSD card data format switch |
| - | - | - | - | - | - | X | - | 0 – Disable Transact Card Number 1 – Enable Transact Card Number |
| X | X | X | X | X | X | - | - | RFU |

Use the top bit of the first byte of DFEF4B to control search behavior; if the bit is ON, all data elements requested will be provided (if they exist). If the bit is OFF, only the first element found will be retrieved and placed in DFEF4D.

When "**Only First Element Found**" (**bit 8 = 1**)" is set in DFEF4B, Tag DFEF4D will be populated with a single data element according to the following search order:

- Track 2, Tag 57 (converted to alpha numeric format)
- Track 2, Tag 9F6B
- Track 2, Tag 5F22
- Track 1, Tag 56
- Track 1, Tag 5F21
- PAN, Tag 5A (converted to alpha numeric format)
- Track 3, Tag 58
- Track 3, Tag 5F23
- Transact Credential Card Number, Tag DFED6E

If you request multiple data items, they will be concatenated. To determine the original lengths of the items, retrieve and inspect Tag DFEF4C (see below).

To use tag DFEF4B, add it (as a TLV) to your terminal configuration settings. Use the **Set Configuration (04-00)** command to send the settings to a device as usual.

For example:

```

56 69 56 4F 74 65 63 68 32 00 ViVOtech2\0
04 00 Set Configuration
00 07 Data Length
DF EF 4B 03 12 00 01 Enable Track 2 and Transact Credential Card Number
2F 79 CRC16

```


NOTE: If this tag does not exist in Terminal Settings, tags DFEF4C and DFEF4D will not be generated.

31.2.2 Tag DFEF4C

If Tag DFEF4B is set, Tag DFEF4C will appear in transaction output. This tag's 6-byte value provides the native lengths of tracks 1, 2, 3, the PAN (if applicable) and the Transact Credential card number. One byte is reserved for future use.

<Track 1 Length><Track 2 Length><Track 3 length><PAN length><Transact length><RFU>

A length of 0 indicates track disabled in DFEF4B or data not available. This tag also serves as an indicator of which data element was found first, when "Only First Element Found" is enabled in DFEF4B.

31.2.3 Tag DFEF4D

If tag DFEF4B is set, tags DFEF4D will appear in transaction output. This variable-length tag contains track and/or PAN data, encrypted. The exact contents will vary depending on values supplied previously in DFEF4B (see above).

The track data will present track data if it is an MSD transaction and present track-equivalent data if it is an EMV transaction.

When TDES or AES encryption has been used in conjunction with traditional DUKPT, decrypt the data normally, using the 10-byte KSN found in Tag DFEE12.

31.2.4 Tag DFEF7B

If Tag DFEF7A's value is set to 1, after reading the card the output data will show Tag DFEF7B to indicate whether it is Apple Pay, Google Pay, or Transact Credentials.

If Tag DFEF7A is not set, the output data will not include the tag.

0 = Not Apple Pay/ Apple VAS or Google Pay/ Android VAS

1 = Apple Pay or Apple VAS

2 = Google Pay or Android VAS

3 = Transact Credentials

31.2.5 VAS Encryption

DFED3F controls VAS encryption options. The tag is set to Group 0.

| | | |
|--------|----------|--|
| DFED3F | Optional | VAS encryption on/off flag Bit 0: Encrypt VAS data with device's data encryption key Bit 1: Decrypt Apple VAS data with Apple VAS private key Bit 2 to 7: RFU |
|--------|----------|--|

For example:

| | |
|--------------------------------------|---|
| 56 69 56 4F 74 65 63 68 32 00 | ViVOtech2\0 |
| 04 00 | Set configuration |
| 00 05 | Data length |
| DF ED 3F 01 01 | Enable both the encryption of Smart Tap and Apple VAS |
| BF 00 | CRC16 |

32. Appendix B: Glossary

The following terms are relevant to this document:

| Term | Definition |
|-------|---|
| AAC | Application Authentication Cryptogram |
| AEF | Application Elementary File (EMV) |
| AELx | Evaluation Assurance Level (1 ..7) |
| AFL | Application File Locator |
| AID | Application Identifier |
| AIP | Application Interchange Profile |
| API | Application Programming Interface or Application Priority Indicator (tag 87) |
| APDU | Application Protocol Data Unit |
| ARQC | Authorization Request Cryptogram |
| ASK | Amplitude Shift Key |
| ATC | Application Transaction Counter |
| ATR | Answer To Reset |
| AUC | Application Usage Control |
| BER | Basic Encoding Rules (ASN.1) |
| CAT | Cardholder Activated Terminals |
| CCC | Compute Cryptographic Checksum |
| CDA | Combined Dynamic Data Authentication/Application Cryptogram Generation (EMV) |
| CID | Cryptogram Information Data |
| CVC | Card Verification Code |
| CVM | Cardholder Verification Method, EMV Book 3, C3 |
| CVV | Card Verification Value (That's the 3 digit number on the back of cards) |
| DCVV | Dynamic CVV |
| DDA | Dynamic Data Authentication (EMV) |
| DF | Dedicated File (7816-4) |
| DOL | Data Object List |
| EF | Elementary File (7816-4) |
| EMV | Europay MasterCard Visa (EMVCo LLC) |
| IIN | Issuer Identification Number |
| IPC | Inter Process Communications |
| KSI | Key set index |
| LRC | Longitudinal Redundancy Check |
| MP | Master File (7816-4) |
| MSD | Magnetic Stripe Data |
| NFC | Near Field Communications |
| PAN | Primary Account Number |
| PCD | Proximity coupling device |
| PICC | Proximity card |
| PN511 | FeliCa Chip from Philips |
| POS | Point of Sale (host) |
| PPSE | Proximity Payment Selection (or System) Environment |
| PSE | Partial Selection something |
| PTS | Protocol Type Selection |
| PUPI | Pseudo Unique PICC Identifier |
| qVSDC | Quick Visa Smart / Debit Credit |
| RID | Registered Application Provider Identifier |

| Term | Definition |
|------|---|
| RN | Random Number |
| SAK | Select Acknowledge |
| SAM | Security Access Module, communicated via 7816-3 in T=0. |
| SDA | Static Data Authentication (EMV) |
| SFGI | Startup Frame Guard Interval (or time Integer) |
| SFI | Short File Identifier (EMV) |
| SID | SAM ID inside of reader |
| T=0 | Protocol Type, T=0 is the asynchronous half duplex character transmission protocol. |
| T=1 | Protocol Type, T=1 is the asynchronous half duplex block transmission protocol. |
| TAK | Terminal Authentication Key |
| TC | Transaction Certificate |
| TID | Terminal ID |
| TLV | Tag Length Value |
| TSI | Transaction Status Information, EMV Book 3, C7 |
| TTQ | Terminal Transaction Qualifier |
| TVR | Terminal Verification Results, EMV Book 3, C5 |
| UN | Unknown Number |
| XOR | Exclusive OR |

33. Appendix C: Revision History

| Revision | Date | Change |
|----------|------------|--|
| | | <ul style="list-style-type: none"> IDG is forked after NEO 1 rev. 142. This document is now for NEO 2. |
| Rev. 50 | 03/27/2018 | <ul style="list-style-type: none"> Add new table for Card Brands and AIDs by Group. Removed "If any command is sent, it will be responded to after second response" for 62-01, -02, -03 to fix NEO 2-1256. |
| Rev. 51 | 04/27/2018 | <ul style="list-style-type: none"> Added links, in tables that discuss LCD messages, to point to Appendix A-1, which contains new message codes and corrected messages for many previously existing codes. Fixes for NEO 2-1397, -1399, -1467. Changed the language about 62-series commands to say no command should be issued, other than Cancel, between responses. |
| | 05/03/2018 | <ul style="list-style-type: none"> Command 09-14: Response table updated to include VP6300 |
| Rev. 52 | 05/15/2018 | <ul style="list-style-type: none"> 2C-0A Response table is updated. New note added about block number and error code. |
| Rev. 53 | 05/16/2018 | <ul style="list-style-type: none"> Add description of tag DFEE26 bit5 -- page 178(Contact EMV L2 Data Objects), 180,181(MSR Data Objects), 210(table 38), 231(table 50) Add Track 3 data description -- page 40(table 15), 204(table 37), 221(table 43), 227(table 48) Delete MasterCard transaction Response Frame data description -- page 204(table 37), 221(table 43), 222(table 44), 227(table 48) Add default configuration groups (Master, Visa, Amex, Discover, Interac, JCB EMV) -- page 492(A.16) Add NEO 2 configuration group default tags -- page 498(A.17) Add NEO 2 Default System AIDs -- page 499(A.18) Add Card in notification behavior -- page 502(A.19) Add commands 81-0E and 81-0F. |
| | 05/18/2018 | <ul style="list-style-type: none"> Add commands 81-0E and 81-0F. |
| | 05/30/2018 | <ul style="list-style-type: none"> Change default value of 9F03 in Group 0 to all zeros in Table 74. |
| Rev. 54 | 05/30/2018 | <ul style="list-style-type: none"> Removed most references to Protocol 1. Removed Burst Mode discussion. |
| Rev. 55 | 05/31/2018 | <ul style="list-style-type: none"> Restored Burst Mode discussion Add TransArmor TDES encrypt option for command C7-32 Add Service code status mask option of tag DFEE1D Modify response data description of command 60-15 Update RF State codes Update Burst Mode payload frame Update Card in notification behavior Add C7-3A (Get DRS Info) Added Bit 5 info for DFEE26 to all tables. 9F1A default value changed to 08 40 for applicable config groups. (It is still 01 24 for Interac.) |
| | 06/01/2018 | |
| Rev. 56 | 06/4/2018 | <ul style="list-style-type: none"> Add command 25-07 (Set 24hr Self-Check Time) |
| Rev. 57 | 06/07/2018 | <ul style="list-style-type: none"> Add command 02-09, 02-58 (Used for MCL Data Exchange) |
| Rev. 58 | 06/07/2018 | <ul style="list-style-type: none"> Amend the Removal Tamper Configuration (04-22) information. Remove note from Command 60-10 to fix NEO 2-1358. |
| Rev. 59 | 06/22/2018 | <ul style="list-style-type: none"> Comprehensive copy edits, formatting adjustments. Remove several obsolete Appendices. Remove Smart Tap section and refer to external document. |
| Rev. 60 | 06/29/2018 | <ul style="list-style-type: none"> Add 60-17 and 60-18 commands. Delete DFEE44 from Group 80. |

| | | |
|---------|------------|---|
| Rev. 61 | 06/29/2018 | <ul style="list-style-type: none"> Change DFDE04 as DFEF04. Add Appendix of Enhanced Encrypted MSR Data Output Format for Financial and Non-Financial Cards. |
| | 07/03/2018 | <ul style="list-style-type: none"> JCB default value of DFEE4C is 0E. Remove max size limitation on 84-07 command. |
| Rev. 62 | 07/06/2018 | <ul style="list-style-type: none"> Add commands 2C-19, -1A, -1B, -1C. |
| Rev. 63 | 07/12/2018 | <ul style="list-style-type: none"> Add command 83-41. |
| | 07/16/2018 | <ul style="list-style-type: none"> Add additional items to response table for command 61-01. |
| Rev. 64 | 07/23/2018 | <ul style="list-style-type: none"> Add Set 24-hr Self-Check Time to command 25-07. |
| Rev. 65 | 07/24/2018 | <ul style="list-style-type: none"> Updated command information for 81-0B, 81-0C, and 81-0D |
| Rev. 66 | 07/30/2018 | <ul style="list-style-type: none"> Remove Vendi and Kiosk III tags and commands. Remove (for now) references to ADF. Clarify tag 'E1' usage. Remove DFEE39. Remove warning about key erasure in 04-09 discussion. Clarify DF811B. Update Gen PAN command to show bit 7 usage in byte 1. |
| | 08/10/2018 | |
| Rev. 67 | 08/17/2018 | <ul style="list-style-type: none"> Remove Visa commands 04-0C, -0D, -E and 03-0C, -D, -E. |
| Rev. 68 | 08/20/2018 | <ul style="list-style-type: none"> Remove commands 25-01, -02, -03, -04 (clock set/get time, set/get date). Modify bit 7 of byte 1 of 83-41 response to reflect encrypted PAN only. ACT TLVs table 36 updated to eliminate DF31, DF32 in favor of DF812A, DF812B. Also, DFEF7F is added. |
| | 08/26/2018 | |
| | 08/31/2018 | <ul style="list-style-type: none"> Remove warning about CAPK erasure in 04-09. For 2C-12, 2C-13, 2C-18, change the SAM support number to 8 Add New Pass Through Command for VP5300M Motor Control: 2C-70: Bypass Data to Motor 2C-71: Entry Motor Boot Mode |
| Rev. 69 | 09/07/2018 | <ul style="list-style-type: none"> Remove commands F0-00, -01, and -0F. Add note about VP3600 to F0-02. Add note about RS-232 mode for F0-03. |
| Rev. 70 | 09/11/2018 | <ul style="list-style-type: none"> Remove commands C7-10, -12, -13, -16, and -23. (Obsolete.) |
| Rev. 71 | 11/08/2018 | <ul style="list-style-type: none"> Remove C7-2F, per JX advice. |
| Rev. 72 | 11/30/2018 | <ul style="list-style-type: none"> Update commands 60-17, 60-18, 60-19. Update C7-11, -14, -15. Update C7-42, -43. Update 60-10, -11, -12. Add C7-17, -18-, 19-, 1A. Update 77-81. Add F0-04, -05 to power management commands. Update all PINPad Keypad commands, including the addition of 62-05. |
| Rev. 73 | 12/03/2018 | <ul style="list-style-type: none"> Add D1-1B. (Remove 90-14 for the 12/4/2018 revision.) |
| | 12/04/2018 | <ul style="list-style-type: none"> Update 83-41, C7-42. Add 01-06. Clarify AID not found error code and message. |
| Rev. 74 | 12/10/2018 | <ul style="list-style-type: none"> Add commands 77-82 and 77-83 for Bluetooth. |
| Rev. 75 | 12/17/2018 | <ul style="list-style-type: none"> Add 32-03. Clarify D1-1B, D1-16, 09-01, and 83-41. |
| Rev. 76 | 12/20/2018 | <ul style="list-style-type: none"> Add C7-44, C7-45, 70-03. Add 19-01, 19-02 for SCRP. Update 61-02 for SCRP. |

| | | |
|---------|-----------------------------|---|
| Rev. 77 | 12/26/2018 | <ul style="list-style-type: none"> Remove 81-02, 81-04, 81-0A, C7-2D, C7-2E. |
| Rev. 78 | 01/07/2019 | <ul style="list-style-type: none"> Add VP5300M and VP6300 USAT product types in 09-01 return values. Add tag 9F53 in default Start Transaction tags. Update Appendix A.11. |
| Rev. 79 | 01/16/2019 | <ul style="list-style-type: none"> Add commands F0-F6, F0-F7. Hex values added to "Table 67: PCD Single Command Exchange Data Field for Response" table. "After sending 03-02, please wait for 800ms after the response is received before continuing" added to 03-02. |
| Rev. 80 | 01/31/2019 | <ul style="list-style-type: none"> Add Cancel Transaction (05-01) to Contact commands section. |
| Rev. 81 | 02/19/2019 | <ul style="list-style-type: none"> Change DFED11 to DFEF3D. Add K81 family to 09-02 results chart. Clarify D0-03 length bytes as MSB/LSB. Add note to 12-01 to indicate that Serial Number is tail-padded with 0x00 to a length of 15. |
| Rev. 82 | 03/12/2019 | <ul style="list-style-type: none"> Update Bootup Notification Command 14-01 with additional detail. |
| Rev. 83 | 04/11/2019 05/3/2019 | <ul style="list-style-type: none"> Add DF ED 49 to VAS Only Activate Transaction (02-40). PayPass Group tag 9F53: Transaction Category Code default changed from 01 to 00. Add tag 9F53 to Default Authenticate Transaction Output TLV Data List. Add DFEF7A to Activate Command TLVs. Add DFEF7B to Activate Transaction Response TLVs. Add product type 56 34 02 (VP6300 USAT-S) to Command: Get Product Type. Add conditions for tag DFEF4D in MSR Equivalent Data Function. Updated messages 0x09, 0x0C, 0x12, and 0x15 in Default Message Index. Revised entry for Cancel Transaction Command/Reset Transaction Status (05-01). Add Get Discretionary Data Whitelist (2C-53). Removed 2C-50, 2C-52, and D1-16 commands; they are no longer supported in NEO2. 60-12: update Default Apply Host Response Output TLV Data List table. Global Config Tags: add Google Smart Tap to Byte 1 Bit 5 in app capability. Instances of "contact card" and "contactless card" have been changed to "contact/contactless transaction" throughout the document; functionality has not otherwise changed. F0-06 and F0-07: Corrected USB Sleep behavior default to "on". Added Wake-Up Notification (F1-00), Switches Notifications (F2-00), Set Switches Notifications Configuration (01-07), and Get Switches Notifications Configuration (01-08). |
| Rev.84 | 05/20/2019 | <ul style="list-style-type: none"> Global TLV settings: <ul style="list-style-type: none"> Added DFED58, "Support of List of AIDs." Get Network Configuration (D1-15): <ul style="list-style-type: none"> Corrected Byte 13 to 00h. Error codes, protocol 2: <ul style="list-style-type: none"> Added 1DH, "Tamper occurred." General Commands: <ul style="list-style-type: none"> Added Set System Language (D1-20). Added Set Model Number (90-15). Added Get Health Data from SCRP (29-30). Transaction Related Commands: <ul style="list-style-type: none"> Contact: Updated Contact Get PIN Control (61-02). |

| | | |
|-----------|---|---|
| | <p>05/21/2019</p> <p>05/31/2019</p> <p>06/04/2019</p> | <ul style="list-style-type: none"> ○ Added note that 19-10 through 19-13 are reserved and used to Create Secure Channels for SCRP. ○ Added 1CVMAApp Get Nonce from SCRP for Refresh Token (19-15). ○ Added Refresh Enablement Token for SCRP (19-16). • Configuration Tab Tables, FFF3: <ul style="list-style-type: none"> ○ Corrected Google Smart Tap support to Byte 1, Bit 5. • Secure Pass-Through Function: <ul style="list-style-type: none"> ○ Get Whitelist (2C-51): Corrected missing information in Command Frame. ○ Get Discretionary Data Whitelist (2C-53): Added missing information in command frame and example response. • Logger Commands: <ul style="list-style-type: none"> ○ Added new commands for the logging system: 4C-01, 4C-02, 4C-03, 4C-04, 4C-05, and 4C-06. • EMV Result Codes: <ul style="list-style-type: none"> ○ Removed several EMV result codes that are not supported by the NEO 2 IDG. • General Commands: <ul style="list-style-type: none"> ○ Added Get Product full information (29-01). • Global TLV settings: <ul style="list-style-type: none"> ○ Corrected tag value for DFED58 to 1 byte in length. • General commands: <ul style="list-style-type: none"> ○ D1-1B: Added data response table. • Miscellaneous NEO 2 Platform Commands: <ul style="list-style-type: none"> ○ 03-0E: Added description and tables. (DOC-109) ○ F2-00, 01-07, 01-08: Added Switches Notifications, Get Switches Notifications Configuration, and Set Switches Notifications Configuration commands. ○ 61-30, 61-31: Added Stop Audio and Play Audio commands for VP6800. • Power Management Commands: <ul style="list-style-type: none"> ○ F1-00: Added Wake-Up Notification command. • EMV Result Codes: <ul style="list-style-type: none"> ○ Corrected EMV_RESULT_CODE_APP_BLOCK_AID to 0x5036. • System AIDs: <ul style="list-style-type: none"> ○ Updated Table 18, System AIDs. ○ Updated Table 28, System AID Default Configuration TLVs. • Transaction Related Commands: Contactless: <ul style="list-style-type: none"> ○ DFEF3C: Updated description table. |
| Rev. 85.1 | | <ul style="list-style-type: none"> • General <ul style="list-style-type: none"> ○ Implemented new minor version numbering scheme. Each update will increment the minor version for the sake of easier version tracking in JIRA. • Miscellaneous NEO 2 Platform Commands <ul style="list-style-type: none"> ○ Added QR Code Scan Start (61-41) and QR Code Scan Stop (61-42). ○ Added Camera Take Photo Start (61-43) and Camera Take Photo Stop (61-44). • USB Power Mode Suspend Configuration <ul style="list-style-type: none"> ○ Added Set USB Power Mode Suspend Configuration (F0-08) and Get USB Power Mode Suspend Configuration (F0-09). • Secure Communication <ul style="list-style-type: none"> ○ Updated Set MSR Secure Parameters (C7-38). |

| | | |
|----------|------------|---|
| | | <ul style="list-style-type: none"> • Apple VAS Functionality: <ul style="list-style-type: none"> ◦ Added DFED49 to ACT Parameters for ApplePay VAS table. |
| Rev 85.2 | 07/26/2019 | <ul style="list-style-type: none"> • Peer To Peer Functionality <ul style="list-style-type: none"> ◦ Removed Detect Card Presence Command (02-05) as it is currently unsupported. • Contact Start Transaction (60-10): <ul style="list-style-type: none"> ◦ Added DFEE1F usage rules. ◦ Removed 9F39 tag and description. • Contact Authenticate Transaction (60-11): <ul style="list-style-type: none"> ◦ Added DFEE1A usage rules. ◦ Added tag DF30 supplement description. ◦ Updated 60-11 Default Authenticate Transaction Output TLV Data List to follow FastEMV Output format • Contact Apply Host Response (60-12): <ul style="list-style-type: none"> ◦ Added tag DF30 supplement description. • MSR Equivalent Data Function: <ul style="list-style-type: none"> ◦ Added 60-10, 60-11, and 60-12 commands to the list of responses in which DFEE4C and DFEE4D TLV tags can appear. ◦ Added 60-06 to the instructions for setting DFEE4B. • Get DRS Info (C7-3A): <ul style="list-style-type: none"> ◦ Added note to Tamper Switch (0x11) explaining the difference in paired switches found in VP3600. • Get TMS (29-20): <ul style="list-style-type: none"> ◦ Added Get TMS command. |
| Rev 86.1 | 08/02/2019 | <ul style="list-style-type: none"> • Updated TLV tags to meet current NEO 2 spec. <ul style="list-style-type: none"> ◦ Most Global Configuration Tags were removed ◦ PayPass Group Configuration TLVs are now in Master Group Configuration TLVs |
| Rev 87.1 | 08/16/2019 | <ul style="list-style-type: none"> • Get DRS Info (C7-3A): <ul style="list-style-type: none"> ◦ Changed Source ID 2 from "Reserved" to "TM4 Slave Check Value Error." • Pass-through Data Exchange Commands: <ul style="list-style-type: none"> ◦ Added Poll for Token with ATS (2C-0E). • Added Appendix A.7: VP6800 Commands <ul style="list-style-type: none"> ◦ Includes commands, limitations, and a screen creation tutorial for the VP6800. |
| Rev 87.2 | 08/23/2019 | <ul style="list-style-type: none"> • Response codes <ul style="list-style-type: none"> ◦ Added A0h, A1h, and A2h. • Appendix A.7: VP6800 Commands <ul style="list-style-type: none"> ◦ Get External Chip Status (83-51): <ul style="list-style-type: none"> ▪ Renamed from Get RT1050 Status. ◦ Get RT1050 Firmware Version (29-80): <ul style="list-style-type: none"> ▪ Renumbered command from (29-00) ◦ Updated several command and Response Frames in multiple commands. ◦ Added additional description for the response. ◦ Moved 61-30, 61-31, 61-41, 61-42, 61-43, and 61-44 from general commands to Appendix A.7 ◦ Added video support information to VP6800 display tables. |
| Rev 88.1 | 10/7/2019 | <ul style="list-style-type: none"> • General <ul style="list-style-type: none"> ◦ Format overhaul for current branding style. |

| | | |
|----------|------------|--|
| | | <ul style="list-style-type: none"> ○ Revised text for style, grammar, and clarity from the start of the document through section 8.2. • Commands <ul style="list-style-type: none"> ○ Set System Language: changed sub-command from D1-20 to D1-27. ○ Added Poll Felica Card (2C-44). ○ Added information about predefined screens for Show Screen (61-05) and Object Event Call Back (61-FF). • Removed Appendix A.8: Whitelist Commands for VP3300. |
| Rev 88.2 | 10/18/2019 | <ul style="list-style-type: none"> • General <ul style="list-style-type: none"> ○ Removed Save Security Data (C7-56) and Read Security Data (C7-57) commands. • Result Codes <ul style="list-style-type: none"> ○ Added documentation for numerous result codes previously not included in previous versions of the NEO 2 IDG. See EMV Result Codes (Tag DFEE25) for details. • RTC Commands <ul style="list-style-type: none"> ○ Added Get RTC Date and Time (25-06) command. • Firmware Downloader Commands <ul style="list-style-type: none"> ○ Updated Start Update Kernel Process (C7-11) command to include support for downloading ADF SDK and ADF App. • Appendix A.7: VP6800 commands <ul style="list-style-type: none"> ○ Added note that Wi-Fi is not currently supported; future use cases are forthcoming. |
| Rev 88.3 | 10/23/2019 | <ul style="list-style-type: none"> • General <ul style="list-style-type: none"> ○ Added link to LCD Foreign Language Mapping table in Contact LCD Display Control (61-01) and Contact Get MSR Data Control (61-03). ○ Added Appendix A.15: Common AIDs. ○ Renumbered Glossary to Appendix A.16. |
| Rev 88.4 | 11/8/2019 | <ul style="list-style-type: none"> • EMV Result Codes <ul style="list-style-type: none"> ○ Restored 0x3013: EMV_RESULT_CODE_TIMEOUT_FOR_WAITING_ICC_INSERT_OR_MS_SWIPE_FALLBACK to table after previous table revision. • General <ul style="list-style-type: none"> ○ Tags 9A: Transaction Date and 9F21: Transaction Time are no longer deprecated. ○ Revised text and formatting in sections 8.3 and 8.4 ○ Updated link to LCD Foreign Language Mapping Table |
| Rev 89.1 | 11/14/2019 | <ul style="list-style-type: none"> • Status Codes <ul style="list-style-type: none"> ○ Added 2Bh: Device not ready. • General <ul style="list-style-type: none"> ○ Renumbered Glossary to Appendix B ○ Renumbered Revision History to Appendix C ○ Added Appendix A.16: Common AID Selection ○ Added Appendix A.17: Fast EMV/Deferred Approval. |
| Rev 89.2 | 11/19/2019 | <ul style="list-style-type: none"> • General <ul style="list-style-type: none"> ○ Added Wi-Fi SSL Connection Request (D1-1C), Set Ethernet SSL Request Configuration (D2-01), and Get Ethernet SSL Request Configuration (D2-02) commands. ○ Corrected PayPass default group from Group 1 to Group 80. ○ Added notes that commands 01-07, 01-08, and F2-00 are only available for Hybrid Insert Readers. ○ Added status codes 2Ch and 2Dh. |

| | | |
|----------|------------|---|
| | | <ul style="list-style-type: none"> • Contact Start Transaction (60-10) <ul style="list-style-type: none"> ◦ Added tag 9F1E: Interface Device (IFD) Serial Number to default Start Transaction output TLV data list. |
| Rev 89.3 | 11/27/2019 | <ul style="list-style-type: none"> • VP6800 UI Commands <ul style="list-style-type: none"> ◦ Removed sections describing pre-defined screens. • PIN Pad (Keypad) Commands <ul style="list-style-type: none"> ◦ Added 8.24.10. Display Specific Message and Get Numeric Key (62-06) command. • Get Product Full Information (29-01) <ul style="list-style-type: none"> ◦ Revised block item numbers in the block list for when Contactless card modules are returned and the Operating System therein to uCOS3 v3.06. • General <ul style="list-style-type: none"> ◦ Revised diagrams in Configurable AIDs and Groups. |
| Rev 89.4 | 12/05/2019 | <ul style="list-style-type: none"> • Tag DFEE44 <ul style="list-style-type: none"> ◦ Corrected inaccurate Byte 1 table. • Error Codes <ul style="list-style-type: none"> ◦ Added 08h: Transaction Timeout, 40h Unknown Data Element Error, and 63h: Card returned Data Element Error. |
| Rev 90.1 | 12/06/2019 | <ul style="list-style-type: none"> • EMV Result Codes (Tag DFEE25) <ul style="list-style-type: none"> ◦ Added 0x3012: EMV_RESULT_CODE_MSR_CARD_ERROR_FALLBACK. (DOC-388) • Loyalty Functions <ul style="list-style-type: none"> ◦ Updated ACT Parameters for ApplePay VAS section. |
| Rev 90.2 | 12/13/2019 | <ul style="list-style-type: none"> • Get Full Product Information (29-01) <ul style="list-style-type: none"> ◦ Corrected error in tag numbers for Contactless card module return information. |
| Rev 90.3 | 12/20/2019 | <ul style="list-style-type: none"> • Contact Start Transaction (60-10) <ul style="list-style-type: none"> ◦ Updated Transaction Data List (start command parameters) table; added Tag DFEE27 and marked several tags as optional. • Felica Pass-Through Commands <ul style="list-style-type: none"> ◦ Updated Pass-Through Package with Felica Card (2C-44) command. • Appendix A.7: VP6800 Commands <ul style="list-style-type: none"> ◦ Added section for VP6800-Specific Hardware Integration Commands: Watchdog for Terminals, including the following commands: <ul style="list-style-type: none"> ▪ Set Configure Watchdog (57-00) ▪ Read Configure Watchdog (57-01) ▪ Enable/Disable Watchdog (57-02) ▪ Read Watchdog Enable/Disable Information (57-03) ▪ Feed Watchdog Timer (57-04) ▪ Erase Watchdog Reset Counter and Log (57-05) ▪ Read Watchdog Reset Counter and Log (57-06) |
| Rev 91.0 | 12/27/2019 | <ul style="list-style-type: none"> • Added Protocol Command Reference: Protocol 3 and Protocol 5. • Transaction Related Commands: Contactless <ul style="list-style-type: none"> ◦ Updated Tag FFEE1F to DFEE02 in Failed Transaction TLV Contactless Error Code Table. |
| Rev 91.1 | 01/03/2020 | <ul style="list-style-type: none"> • Appendix A.7: VP6800 Commands <ul style="list-style-type: none"> ◦ Added section detailing VP6800 Wi-Fi commands and setup. |
| Rev 91.2 | 01/09/2020 | <ul style="list-style-type: none"> • Status Codes <ul style="list-style-type: none"> ◦ Added A3h: Error code for card insertion. ◦ Added A4h: Error code for card insertion before sending a transaction command (for products with motors). |

| | | |
|-----------|------------|--|
| | | <ul style="list-style-type: none"> Appendix A.14: Enhanced Encrypted MSR Data Output Format for Financial and Non-Financial Cards. <ul style="list-style-type: none"> Retitled previously existing output format tables to SRED Output Formats. Added tables describing Non-SRED output formats. |
| Rev 91.3 | 01/10/2020 | <ul style="list-style-type: none"> Updated SetBacklight (83-2F) command frame and data format tables. |
| Rev 91.4 | 01/28/2020 | <ul style="list-style-type: none"> Card Application Specific Behavior <ul style="list-style-type: none"> Stop Transaction Command: Removed reference to Tag DF68. |
| Rev 91.5 | 02/12/2020 | <ul style="list-style-type: none"> General Commands <ul style="list-style-type: none"> Added Enable/Disable Communication Port (D2-03) command. (DOC-458) Miscellaneous Commands <ul style="list-style-type: none"> Added Set Key Press Event (01-0E), Get Key Press Event (01-0F), and Key Press Event (61-04). (DOC-457) |
| Rev 91.6 | 02/18/2020 | <ul style="list-style-type: none"> Added Appendix A.18: VP5300M Commands. <ul style="list-style-type: none"> Added Motor Eject Card (60-19) and Motor Clean (60-1A) commands. General <ul style="list-style-type: none"> Added notes to Control User Interface (01-02) and Enhanced Pass-Through Command (2C-0B) stating that only VP6300 devices fully support the commands. Removed the response frame from Key Press Event (61-04) command. |
| Rev 91.7 | 02/19/2020 | <ul style="list-style-type: none"> Appendix A.7: VP6800 Commands: Added the following commands: <ul style="list-style-type: none"> Set/Get Audio Volume (61-32) Add Object to Scroll Container (61-18) Get List of All Objects on Scrollable Container (61-19) Camera Take Photo Once (61-45) Added Scrollable Container data and object information to LCD UI Control Function to the Add Object to Screen (61-07) and Object Event Call Back (61-FF) commands. |
| Rev 91.8 | 02/20/2020 | <ul style="list-style-type: none"> Appendix A.7: VP6800 Commands <ul style="list-style-type: none"> Added Set Cancel Button On/Off (61-33) and Get Cancel Button On/Off (61-34) commands. |
| Rev 91.9 | 02/21/2020 | <ul style="list-style-type: none"> Loyalty Functions <ul style="list-style-type: none"> Restored Tag DFED49 to Apple VAS ECP 2.0 support table. |
| Rev 91.10 | 02/25/2020 | <ul style="list-style-type: none"> General <ul style="list-style-type: none"> Updated Description of Tag DFEE12 to "This Tag only appears when the transaction response contains encrypted data." Added a new section, MSRFunction Check, and a new command, Check MSRSwipe Function (2C-20). Added "1Eh: Card not removed" to Status Codes for Protocol 2. PIN Pad Commands <ul style="list-style-type: none"> Display Specific Message and Get Numeric Key (62-06) <ul style="list-style-type: none"> Updated Pre-Cleartext Display and Post-Cleartext Display parameters: the valid range is now $0 \leq \text{min} \leq \text{max} \leq 6$. |
| Rev 91.11 | 02/26/2020 | <ul style="list-style-type: none"> Transaction Related Commands: Contact <ul style="list-style-type: none"> Contact Apply Host Response (60-12): Added Attribution byte to the second response frame's Data Objects table. |
| Rev 91.12 | 03/05/2020 | <ul style="list-style-type: none"> Miscellaneous NEO 2 Platform Commands |

| | | |
|-----------|------------|--|
| | | <ul style="list-style-type: none"> ○ Renamed Removal Tamper Command (04-22) to Front Removal Configuration (04-22), updated description, and changed the description for the data in byte 14 of the command frame. (DOC-492) • General <ul style="list-style-type: none"> ○ Changed instances of "NEO II" to "NEO 2" for consistency. • Transaction Related Commands: Contact <ul style="list-style-type: none"> ○ Contact Apply Host Response (60-11): Added Attribution byte to the second response frame's Data Objects table. |
| Rev 91.13 | 03/11/2020 | <ul style="list-style-type: none"> • Contact LCD Display Control (61-01) command: <ul style="list-style-type: none"> ○ Updated Display Message Control data object description. • Configuration Tags: <ul style="list-style-type: none"> ○ Tag DFEE38: Added Values 06=Japanese, 07=Spanish, and 08=Turkish to description in various tables and pages. |
| Rev 92.1 | 03/12/2020 | <ul style="list-style-type: none"> • VP6800 Commands <ul style="list-style-type: none"> ○ Added new section, VP6800 Memory-Related Commands, with the following items: <ul style="list-style-type: none"> ▪ Get Device Memory Usage Information (09-30) ▪ Get External Chip Memory Usage Information (09-31) |
| Rev 92.2 | 03/13/2020 | <ul style="list-style-type: none"> • Updated description of TLV Tag DFEE51 to "KSN of online PIN DUKPT;" functionality remains the same. • Module Versioning commands <ul style="list-style-type: none"> ○ Added VP3320S to possible product types returned by the Get Product Type (09-01) command. • Appendix A.7: VP6800 Commands <ul style="list-style-type: none"> ○ Added Query File (83-32) command. |
| Rev 92.3 | 03/26/2020 | <ul style="list-style-type: none"> • Status Codes <ul style="list-style-type: none"> ○ Updated table of Module-Specific Status Codes. • Configuration Tags <ul style="list-style-type: none"> ○ Added Tag DFED5A: UI Event Control to Group Configuration Tags. |
| Rev 92.4 | 04/01/2020 | <ul style="list-style-type: none"> • Bluetooth Interface Commands <ul style="list-style-type: none"> ○ Added Set Bluetooth Parameters (77-84) command. • Tag DFEE44 <ul style="list-style-type: none"> ○ Added Google Smart Tap capability to Application Capability. • Get DRS Info (C7-3A) Command <ul style="list-style-type: none"> ○ Corrected 0xFE under Battery; the correct result for "Status not checked" is 0xFF. |
| Rev 92.5 | 04/14/2020 | <ul style="list-style-type: none"> • Appendix A.7 VP6800 Commands: <ul style="list-style-type: none"> ○ Renamed "VP6800 UI Commands" to "VP6800-Specific Commands." ○ Added the following commands: <ul style="list-style-type: none"> ▪ Transfer File to SD Card (83-33) ▪ Delete File from SD Card (83-34) ▪ List Directory on SD Card (83-35) ▪ Query File on SD Card (83-36) |
| Rev 93.1 | 04/17/2020 | <ul style="list-style-type: none"> • Major update to Appendix A.9: NEO 2 Contactless Groups and Default TLVs, including new groups Group E0 (CUP) default tags and Group 8E (Smart Tap) default tags. |
| Rev 93.2 | 04/24/2020 | <ul style="list-style-type: none"> • General <ul style="list-style-type: none"> ○ Updated "Google SmartTap" to "Google Pay Smart Tap." • Protocol 2 Status Codes <ul style="list-style-type: none"> ○ Added the following codes: <ul style="list-style-type: none"> ▪ 2Bh: Device is not ready state ▪ 2Ch: Command Cancel Operation |

| | | |
|----------|------------|--|
| | | <ul style="list-style-type: none"> ▪ 2Dh: Press Key Cancel Operation ▪ 2Eh: Removal Detection is disabled |
| Rev 93.3 | 05/08/2020 | <ul style="list-style-type: none"> • EMV Result Codes Tag (Tag DFEE25): <ul style="list-style-type: none"> ○ Added 0x0013: EMV_RESULT_CODE_CARD_IN_NOTIFICATION. ○ Added 0x3014: EMV_RESULT_CODE_CARD_INSERTION_FAILED. • Updated DFED5A bit5 Byte 1, "Beep on CT completion" description: setting is for VP3300-only. • Apple VAS <ul style="list-style-type: none"> ○ Removed information related to VAS Failure. • Secure Communication <ul style="list-style-type: none"> ○ Updated description in Set DUKPT Key Encryption Type (C7-32) command. |
| Rev 93.4 | 05/12/2020 | <ul style="list-style-type: none"> • Secure Communications <ul style="list-style-type: none"> ○ Added Get Security Log (C7-3D) command. ○ Added Appendix A.19 New Security Log Event IDs. • VP6800-Specific Commands <ul style="list-style-type: none"> ○ Updated Transfer File (83-24) command: maximum single file size is now 64KB, maximum number of image files is now 256. • Transaction Related Commands: Contact <ul style="list-style-type: none"> ○ Contact Set CA Public Key (60-0A) command: removed 16-key limit. |
| Rev 93.5 | 05/15/2020 | <ul style="list-style-type: none"> • PIN Pad (Keypad) Commands <ul style="list-style-type: none"> ○ Updated the following sections: <ul style="list-style-type: none"> ▪ Secondary Response Mechanism ▪ Status and State Codes in Secondary Response ▪ Cancel PIN or Key Entry (62-00) ▪ Display Message and Get Encrypted PIN (62-01) |
| Rev 94.1 | 05/21/2020 | <ul style="list-style-type: none"> • Contact Start Transaction (60-10): <ul style="list-style-type: none"> ○ Corrected Tag DFEF27 to DFEE27. • Activate Transaction Command (02-01 and 02-40): <ul style="list-style-type: none"> ○ Removed Tag FFEE04: Tag not supported in NEO 2. • VP6800 Commands <ul style="list-style-type: none"> ○ Added Enable/Disable Light Sensor (83-30) command. ○ Get Current Screen Name (61-0C): <ul style="list-style-type: none"> ▪ Added example with error report cases. ○ List Directory (83-22): <ul style="list-style-type: none"> ▪ Updated response information and examples. ○ List Directory on SD Card (83-35): <ul style="list-style-type: none"> ▪ Updated response information and examples. • General Commands: • Added the following commands: <ul style="list-style-type: none"> ○ Set Function/Hardware Enable/Disable (29-40) ○ Get Function/Hardware Enable/Disable (29-41) ○ Set USB Mode Convert and Attribution (D1-30) ○ Get USB Mode Convert and Attribution (D1-31) • Appendix A.20: SREDKey 2 Commands <ul style="list-style-type: none"> ○ Added new appendix. ○ Added new commands: <ul style="list-style-type: none"> ▪ Set MSR / Manual PAN Entry Configuration with TLV (C7-80) ▪ Get MSR / Manual PAN Entry Configuration with TLV (C7-81) ▪ Set Special Customer Configuration (C7-82) ▪ Get Special Customer Configuration (C7-83) ○ Added TLV Format for MSR and Manual PAN Entry section. |

| | | |
|----------|------------|---|
| Rev 95.1 | 06/12/2020 | <ul style="list-style-type: none"> Removed references in command tables to commands previously removed from this document. Removed text related to VP3300 products. Removed Get DRL Reader Risk Parameters (03-0E). Removed Appendix A.8: Whitelist Commands for VP3300 Family. Removed Control IO (01-06) command. Tag DFED5A: Added note to Byte 1 Bit 7: "online decline only." |
| Rev 95.2 | 06/16/2020 | <ul style="list-style-type: none"> EMV Status Codes <ul style="list-style-type: none"> Moved status codes 2Bh, 2Ch, 2Dh, and 2Eh from general status codes to module-specific status codes in the same table. Added status codes A0h, A1h, A2h, and A5h. VP6800-Specific Commands <ul style="list-style-type: none"> Updated examples for List Directory (83-22) command. Removed the following commands, which do not apply to NEO 2 products, and references to those commands: <ul style="list-style-type: none"> Get Product Type (09-01) Get Main Firmware Version (09-03) Set MSR Secure Parameters (C7-38) Get MSR Secure Parameters (C7-39) Configure Buttons Command (F0-F4) Get Button Configuration Command (F0-F5) Disable Blue LED Sequence (F0-F6) Enable Blue LED Sequence Command (F0-F7) Turn Off Yellow LED Command (F0-FA) Turn On Yellow LED Command (F0-FB) LCD Display Line 1 Message Command (F0-FC) LCD Display Line 2 Message Command (F0-FD) Buzzer On/Off Command (F0-FE) Group Configuration Tags <ul style="list-style-type: none"> Removed DFEE1C: LCD Font Size; tag does not apply to NEO 2. General Commands <ul style="list-style-type: none"> Get Network Configuration (D1-15): Updated data length to 00 01. |
| Rev 95.3 | 06/23/2020 | <ul style="list-style-type: none"> VP6800-Specific Commands <ul style="list-style-type: none"> Added Turn On or Off Visual LEDs on LCD (61-1A) and Get SD Card State (83-37) commands. |
| Rev 95.4 | 07/02/2020 | <ul style="list-style-type: none"> Restored the following commands: <ul style="list-style-type: none"> Get Product Type (09-01) Get Main Firmware Version (09-03) Set MSR Secure Parameters (C7-38) Get MSR Secure Parameters (C7-39) |
| Rev 95.5 | 07/17/2020 | <ul style="list-style-type: none"> Tag DF30 <ul style="list-style-type: none"> Added 0x01: Contact card for all descriptions of this tag. PIN Pad (Keypad) Commands <ul style="list-style-type: none"> Updated Notes in Display Message and Get Encrypted PIN (62-01). Logger Commands <ul style="list-style-type: none"> Added notes to Initialize/De-initialize Logger (4C-01) and Get/Set Logger Size (4C-03) noting theoretical maximum log count and caveat that the maximum size depends on device memory. |

| | | |
|-----------|------------|---|
| Rev 96.1 | 08/10/2020 | <ul style="list-style-type: none"> • General • Removed section for Protocol 1. <ul style="list-style-type: none"> ◦ Revised descriptions throughout the document to indicate that a ViVOPay device is a "reader" and that POS terminals, mobile devices, etc. are "hosts" where applicable. ◦ Made the use of MSB/LSB consistent in command and response frames throughout the document. • Appendix A.20: SREDKey 2 Commands <ul style="list-style-type: none"> ◦ Many existing tags were replaced, see 31.3. TLV Format for MSR and Manual PAN Entry for details. ◦ Added situational notes about using Get PAN (83-41) with SREDKey2 readers. See the command for details in 7.25. PIN Pad (Keypad) Commands. |
| Rev 96.2 | 09/11/2020 | <ul style="list-style-type: none"> • Added Select KeySlot for PEK and DEK (C7-E0) and Return KeySlot for PEK and DEK (C7-E1). • Added ADF Application Control Command (C7-C7). |
| Rev 96.3 | 09/18/2020 | <ul style="list-style-type: none"> • Special Customer Configuration Commands <ul style="list-style-type: none"> ◦ Updated Byte 13 length of C7-82 command frame and C7-83 response frame; byte can have a variable value, not a fixed length. |
| Rev. 96.4 | 09/28/2020 | <ul style="list-style-type: none"> • Transaction Related Commands <ul style="list-style-type: none"> ◦ Updated Contact Set Application Data (60-03) description to reflect that the command creates a maximum of 20 AID configurations. • VP6800 Commands <ul style="list-style-type: none"> ◦ Added Link UI with Transaction Message ID (61-1B) command. |
| Rev 96.5 | 10/9/2020 | <ul style="list-style-type: none"> • Removed the following commands not supported in NEO 2: <ul style="list-style-type: none"> ◦ Poll for Token with ATS (2C-0E) ◦ Terminate ISO Session (2C-16) • Generic Pass-Through Commands <ul style="list-style-type: none"> ◦ Changed 0Fh to "Reserved" in the response frame for Poll for Token (2C-02). |
| Rev 96.6 | 10/16/2020 | <ul style="list-style-type: none"> • Status and State Codes in Secondary Response <ul style="list-style-type: none"> ◦ Added codes 0Ah and 0Fh. • Error Codes <ul style="list-style-type: none"> ◦ Corrected error code for LTPK Error from 88h to 8Bh. • VP6800-Specific Commands <ul style="list-style-type: none"> ◦ Updated data format table for QR Code Scan Start (61-41). |
| Rev 97.1 | 10/23/2020 | <ul style="list-style-type: none"> • Burst Mode <ul style="list-style-type: none"> ◦ Added note that if Burst Mode is set to ON/AUTO EXIT through the Set Configuration (04-00) command, the reader will keep Burst Mode OFF and respond with the status code "Command Not Supported (0x0B)." • Set Configurable Group (04-03) <ul style="list-style-type: none"> ◦ The length of TLV Data can only be written up to 1024 bytes. • NEO II Group B0 (Discover) Default Tags <ul style="list-style-type: none"> ◦ Updated length and default values for DFEF46, DFEF4E, and DFEF4F. • Appendix A.10: NEO 2 Default System AIDs <ul style="list-style-type: none"> ◦ Added Common-AID VISA, Common-AID Discover, Common-AID CUP, and Common-AID MasterCard. • Appendix A.11: NEO 2 Default Configuration TLVs <ul style="list-style-type: none"> ◦ Added default config TLVs for Common AID Visa, Common AID MasterCard, Common AID Discover, and Common AID CUP. |
| Rev 97.2 | 10/28/2020 | <ul style="list-style-type: none"> • VP6800 Commands |

| | | |
|----------|------------|---|
| | | <ul style="list-style-type: none"> ○ Updated Add Object to Screen (61-07) with Extended Video parameters. |
| Rev 97.3 | 10/29/2020 | <ul style="list-style-type: none"> • VP6800 Commands <ul style="list-style-type: none"> ○ Removed VP6800 Specific TS Command (90-13). (DOC-809) • Add Object to Screen (61-07) <ul style="list-style-type: none"> ○ Updated Extended Video Feature parameters (removed width and height parameters). (DOC-810). • Object Event Call Back (61-FF) <ul style="list-style-type: none"> ○ Updated command description to refer to the end of video playback. (DOC-810) |
| Rev 97.4 | 12/04/2020 | <ul style="list-style-type: none"> • Appendix A.9: NEO 2 Contactless Groups and Default TLVs <ul style="list-style-type: none"> ○ NEO 2 Group B0 (Discover) default tags <ul style="list-style-type: none"> ▪ Updated default values for DFEF46 and DFEF4F. ○ Added NEO 2 Group 85 (Common AID) default tags. • Appendix A.10: NEO 2 Default System AIDs <ul style="list-style-type: none"> ○ Updated Default System AIDs table. • Appendix A.11: NEO 2 Default Configuration TLVs <ul style="list-style-type: none"> ○ Added configurations for CUP, CUP Debit, CUP Credit, Apple VAS, and Google Pay Smart Tap. |
| Rev 97.5 | 12/11/2020 | <ul style="list-style-type: none"> • Appendix A.9: NEO 2 Contactless Groups and Default TLVs <ul style="list-style-type: none"> ○ NEO 2 Group 0 Default Tags <ul style="list-style-type: none"> ▪ Updated description of Tag 9F66 from PUNTAC to TTQ. ○ NEO 2 Group B0 (Discover) Default Tags <ul style="list-style-type: none"> ▪ Updated default length for DFEF46 and DFEF4F to 12. |
| Rev 98.1 | 12/30/2020 | <ul style="list-style-type: none"> • New Secure Communication Commands <ul style="list-style-type: none"> ○ Set DUKPT PEK Attribution based on KeySlot (C7-A0) ○ Get DUKPT PEK Attribution based on KeySlot (C7-A1) ○ Set DUKPT DEK Attribution based on KeySlot (C7-A2) ○ Get DUKPT DEK Attribution based on KeySlot (C7-A3) ○ Set KeySlot for DUKPT PEK and DUKPT DEK (C7-A4) ○ Get KeySlot for DUKPT PEK and DUKPT DEK (C7-A5) • Updates to description text or command information in the following commands: <ul style="list-style-type: none"> ○ Add Entry to EMV Revocation List (84-04) ○ Enhanced Poll for Token (2C-0C) ○ Felica Commands (2C-41) ○ Felica Lite/Lite-S Read/Write Blocks with MAC (2C-43) ○ Save/Delete Log (4C-05) ○ Retrieve Log (4C-06) ○ Set Data Encryption Enable Flag (C7-36) ○ Set MSR Tracks (C7-44) ○ Get MSR Tracks (C7-45) • Updates to introductory text in the following chapter and sub-chapter headings: <ul style="list-style-type: none"> ○ EMV Exception Log List Commands ○ Secure Communication: Encryption Algorithms ○ Secure Communication: Secure Data Exchange ○ Secure Communication: Padding of Data Fields • General Changes |

| | | |
|----------|------------|---|
| | | <ul style="list-style-type: none"> ○ Configuration Tags: Updated Tag DFEE1E Byte 1 Bit 4 to "Confirm Amount is enabled") ○ EMV Result Codes: Added 0x5039: EMV_RESULT_CODE_PIN_ENTRY_TIMEOUT. ○ Commands in the Pass-through Commands section and sub-sections have been reorganized. ○ NFC Commands (2C-40): Removed irrelevant card types from NFC Command Set Response Data List table. • Secure Communication <ul style="list-style-type: none"> ○ Removed Get Security Log (C7-3D). ○ Removed Select KeySlot for PEK and DEK (C7-E0). ○ Removed Return KeySlot for PEK and DEK (C7-E1). |
| Rev 99.1 | 01/15/2021 | <ul style="list-style-type: none"> • VP6800 Commands <ul style="list-style-type: none"> ○ Added note that all commands in VP6800 Appendix are allowed in Passthrough Mode. ○ Removed Get Battery Level (F0-02) command. ○ Removed Get Device Memory Usage Information (09-30) command. ○ Extended Status Codes <ul style="list-style-type: none"> ▪ Added A1h: SD card not inserted ▪ Added A5h: LCD is not working ○ Updated description text, parameters, or examples for the following commands: <ul style="list-style-type: none"> ▪ Show Screen (61-05) ▪ Add Object to Screen (61-07) ▪ Add Object to Scroll Container (61-18) ▪ List Directory on SD Card (83-35) ▪ Get External Chip Status (83-51) ▪ QR Code Scan Start (61-41) • Module Versioning <ul style="list-style-type: none"> ○ Added Get TM4 Firmware Version Information (C7-3F) command. • General Commands <ul style="list-style-type: none"> ○ Updated Set Baud Rate (30-01) and Set Temporary Baud Rate (30-02) command frame and baud rate codes. |
| Rev 99.2 | 01/29/2021 | <ul style="list-style-type: none"> • Display Message and Get Encrypted PIN (62-01) <ul style="list-style-type: none"> ○ Updated possible PAN and Key Type parameters in Command Data Field. ○ Added optional PEK KeySlot Length and PEK KeySlot parameters in Command Data Field. • Get PAN (83-41) <ul style="list-style-type: none"> ○ Added optional DEK KeySlot Length and DEK KeySlot parameters in Command Frame Data Field. ○ Updated Command Flags Field Table. • ADF Application Control Command (C7-C7) <ul style="list-style-type: none"> ○ Updated Command Frame Data Field Table. • High-level Pass-Through Commands for Felica Cards <ul style="list-style-type: none"> ○ Added Felica Get Code (2C-45) and Cancel Felica Get Code (2C-46) commands. • Group Configuration Tags <ul style="list-style-type: none"> ○ DFEE44: Corrected Google Pay Smart Tap to Byte 1, Bit 5. • Module Versioning |

| | | |
|-----------|------------|--|
| | | <ul style="list-style-type: none"> ○ Updated the example for Get Hardware Information (09-14) command. |
| Rev 99.3 | 04/02/2021 | <ul style="list-style-type: none"> • Activate Transaction Response Frame Data Format and Get Transaction Result (03-00 and 03-40): <ul style="list-style-type: none"> ○ Updated TLV Tags; Track1 updated from FFEE13 to DFEF17, Track2 updated from FFEE14 to DFEF18. • Group Configuration Tags <ul style="list-style-type: none"> ○ Added Tag DFED59. ○ Added MIR Support as Byte 1, bit 8 to Tag DFEE44. |
| Rev 99.4 | 04/09/2021 | <ul style="list-style-type: none"> • Activate Transaction Command (02-01 and 02-40) <ul style="list-style-type: none"> ○ Added note to Timeout parameter in Activate Transaction Command Frame Data Format; If Timeout=0x00, the reader will wait forever. • Group Configuration Tags <ul style="list-style-type: none"> ○ Updated DFED5A Byte3, Bit0 and Bit1. • Secure Communication: Set KeySlot for DUKPT PEK and DUKPT DEK (C7-A4) <ul style="list-style-type: none"> ○ Altered description for conditions for PEK/DEK key selection; "If the key exists and has not reached End of Life". |
| Rev 100.1 | 04/23/2021 | <ul style="list-style-type: none"> • 2-Byte vs 3-Byte Tags <ul style="list-style-type: none"> ○ This feature is now deprecated; Tag DFEF3D is now disabled in 2-byte mode and reserved in 3-byte mode. The section describing this feature has been retained for reference. • TLV Tags <ul style="list-style-type: none"> ○ Auto-Switch Tag DF7C is now DFEE7C. |
| Rev 100.2 | 04/30/2021 | <ul style="list-style-type: none"> • Module Versioning Commands <ul style="list-style-type: none"> ○ Added Get System Info (09-15) command. • Bluetooth Interface <ul style="list-style-type: none"> ○ Set Bluetooth Parameters (77-84) <ul style="list-style-type: none"> ▪ Added specific advertising intervals for VP3600 and VP3320S. • Group Configuration Tags <ul style="list-style-type: none"> ○ DFED5A <ul style="list-style-type: none"> ▪ Updated Byte 2 Bit 7 from RFU to "Do Not Remove Card." |
| Rev 100.3 | 05/07/2021 | <ul style="list-style-type: none"> • Burst Mode: ViVOPay Burst Mode Frames <ul style="list-style-type: none"> ○ Added note regarding Track 1 and 2 data from CL cards in Burst Mode; removed Track 3 Length and Track 3 Data from Payload Frame for CVN17 Enabled Readers. • Get Merchant Record <ul style="list-style-type: none"> ○ Added a note that there is no default merchant record and that users must use the Set Merchant Record (04-11) command before using Get Merchant Record. • Added new Appendix: Transact Credentials over ECP2.0. |
| Rev 100.4 | 06/22/2021 | <ul style="list-style-type: none"> • Module Versioning <ul style="list-style-type: none"> ○ Updated command information and example for Get Module Version Information (09-20) command. |
| Rev 100.5 | 07/02/2021 | <ul style="list-style-type: none"> • Special Customer Configuration Commands <ul style="list-style-type: none"> ○ Moved Set Special Customer Configuration (C7-82) and Get Special Customer Configuration (C7-83) from SREDKey 2 Commands to Protocol 2 commands, as they apply to all devices. |
| Rev 100.6 | 07/23/2021 | <ul style="list-style-type: none"> • Tag DFEE38 <ul style="list-style-type: none"> ○ Added note related to DFEE38 that if a reader does not support a language value, it will return 0x06. |

| | | |
|-----------|------------|---|
| Rev 101.1 | 08/02/2021 | <ul style="list-style-type: none"> • Contact Set Certification Revocation List (60-0E) <ul style="list-style-type: none"> ○ This command is no longer limited to a maximum CRL number. • Activate Transaction Response Frame Data Format (02-01) <ul style="list-style-type: none"> ○ Removed note that MasterCard transactions do not return a clearing record or the track data fields. • Get Transaction Result (03-00 and 03-40) <ul style="list-style-type: none"> ○ Removed notes in "Get Transaction Result format for contactless transaction: Plaintext and Encrypted data field format" table for Track 1 and Track 2 TLV data that field is not present in MasterCard transactions. • ADF Application Control Command (C7-C7) <ul style="list-style-type: none"> ○ Updated parameters in data field; added additional response info for Retrieve Mode Status, Retrieve SDKs/APPs Info List, and Retrieve SDKs/APPs (First 64 Bytes). • Added Encrypted R-APDU Format Rules to the following commands: <ul style="list-style-type: none"> ○ Exchange Contactless Data (2C-03) ○ PCD Single Command Exchange (2C-04) Protocol 2 ○ Exchange APDU Data (2C-13) ○ Mifare Read Blocks (2C-07) ○ Pass-Through Package with Felica Card (2C-44) • Special Function Feature Configuration Commands <ul style="list-style-type: none"> ○ Updated sections and commands previously named "Special Customer Command" to "Special Function or Feature Command" (includes C7-82 and C7-83). ○ Set Special Function or Feature Configuration (C7-82) <ul style="list-style-type: none"> ▪ Updated Data Table, added note about ITP protocol. ○ Get Special Functions or Features Configuration (C7-83) <ul style="list-style-type: none"> ▪ Updated Data Table. ○ Added Function/Feature ID Table. • VP6800 Commands <ul style="list-style-type: none"> ○ Added Read Out File from SD Card (83-39) command. |
|-----------|------------|---|