IDTECH®

Value through Innovation

ViVOpay®

NEO 2 Interface Developer's Guide

80139403-002 Rev. 88.1

10 October, 2019

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

ID TECH
10721 Walker St.
Cypress, CA 90630

Table of Contents

List of Tables

1. Introduction

This document provides application developers and integrators with the detailed information necessary to integrate ID TECH's ViVOpay readers with point of sale terminals (POS). 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 device in question.

A separate guide (similar to this one) exists for NEO 1 devices, such as Kiosk III and Vendi. Consult that guide for information about those products.

The Linux-based VP8800 uses AR 3 firmware, which is not covered in this manual. Contact your ID TECH representative to obtain the Interface Developer's Guide for AR 3.

1.1. Terminology

Throughout this manual, the terms "host," "terminal," and "host terminal" all refer to the host computer (or 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 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. Device Modes

Before the introduction of contactless EMV, ViVOpay readers usually worked in standalone mode, which does require a terminal 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 device.

1.3. Universal SDK

Developers who require high-level-language support should consider using ID TECH's Universal SDK when integrating NEO 2 devices 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 device. 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 (Protocol 1 and Protocol 2), 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 sections (Protocol 1 and Protocol 2) describe each of the commands available, 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 devices 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 terminal. 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	
Antenna Control	√	28	01	
Boot up Notification	√	14	01	
Buzzer Control Long	√	0B	02	
Buzzer Control Short	√	0B	01	
Buzzer On/Off Command		F0	FE	a
Camera Take Photo Start		61	43	
Camera Take Photo Stop		61	44	
Cancel Transaction Command	√	05	01	
Check DUKPT Key	√	81	04	
Check DUKPT Keys	√	81	02	
Clean Torn Transaction Log	√	84	0F	
Configure Buttons Command		F0	F4	a
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	b
Contact Remove CA Public Key		60	09	b
Contact Remove Certification Revocation List		60	0D	b
Contact Remove Terminal Data		60	05	b
Contact Retrieve AID List		60	07	b
Contact Retrieve Application Data		60	01	b
Contact Retrieve CA Public Key		60	0A	b
Contact Retrieve CA Public Key List		60	0B	b
Contact Retrieve Certification Revocation List		60	0C	b
Contact Retrieve Terminal Data		60	04	b
Contact Retrieve Transaction Result		60	13	b
Contact Set Application Data		60	03	b
Contact Set CA Public Key		60	0A	b
Contact Set Certification Revocation List		60	0E	b

Command	C'less or C'less + MSR	CMD	SUB CMD	Notes
Contact Set Terminal Data		60	06	b
Contact Start Transaction		60	10	b
Contact Get ICS Identification		60	15	b
Contact Remove Transaction Amount Log		60	0F	b
Contact Set ICS Identification		60	16	b
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	c
Delete Configurable Group (DCG)	√	04	05	c
Delete Entry from EMV Exception List	√	84	0A	
Disable Blue LED Sequence		F0	F6	
Enable/Disable Logger		4C	04	
Enable Blue LED Sequence Command		F0	F7	a
Enhanced Pass-Through Command	√	2C	0B	
Enhanced Poll for Token	√	2C	0C	
Exchange APDU Data	√	2C	13	
Exchange Contactless Data	√	2C	03	
Felica Commands	√	2C	41	
Felica Lite/Lite-S Authentication	√	2C	42	
Felica Lite/Lite-S Read/Write Blocks with MAC	√	2C	43	
Flush Track Data	√	17	02	
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	c
Get All CA Public RIDs Protocol 2	√	D0	06	
Get All Groups (GAG)	√	03	07	c
Get ALL Reader Variables	√	09	00	
Get ATR	√	2C	12	
Get Button Configuration Command		F0	F5	c
Get Cable Type	√	32	01	
Get CA Public Key Hash Protocol 2	√	D0	02	
Get CA Public Key Protocol 2	√	D0	01	
Get Cash Transaction Reader Risk Parameters	√	03	0C	
Get Cashback Transaction Reader Risk Parameters	√	03	0D	
Get Configurable AID	√	03	04	c
Get Configurable Group	√	03	06	c
Get Configuration	√	03	02	
Get Contact EMV L2 Kernel Checksum	√	29	08	
Get Contact EMV L2 Kernel Version	√	29	06	

Command	C'less or C'less + MSR	CMD	SUB CMD	Notes
Get Contact EMV L2 Kernel Version Detail	√	29	07	
Get Contact EMV L2 Terminal Configuration Checksum	√	29	09	
Get Discretionary Data White List	√	2C	53	
Get DRL Reader Risk Parameters	√	03	0E	
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	
Get Firmware Full Version	√	29	00	
Get Full Track Data	√	17	CD	
Get Hardware Information	√	09	14	
Get Data Encryption Enable Flag		C7	37	
Get Data Encryption Key Variant Type		C7	30	
Get Merchant Record	√	03	11	
Get Module Version Information	√	09	20	
Get Main Firmware Version	√	09	03	a
Get MSR Secure Parameters	√	C7	39	a
Get PCD and PICC Parameters	√	2C	05	
Get Processor Type	√	09	02	
Get Product full information		29	01	
Get Product Type	√	09	01	a
Get Serial Number	√	12	01	
Get Switches Notifications Configuration		01	08	
Get TMS		29	20	
Get Transaction Result	√	03	00	
Get Transaction Result	√	03	40	
Get UID of MCU	√	29	17	
Get USB Boot Loader Version	√	29	04	e
Get USB Power Mode Suspend Configuration		F0	09	e
Get White List	√	2C	51	
High-level Halt Command	√	2C	09	
Initialize/De-initialize Logger		4C	01	
LCD Display Clear Command		F0	F9	a
LCD Display Line 1 Message Command		F0	FC	a
LCD Display Line 2 Message Command		F0	FD	a
LED Control	√	0A	02	
List CA Public Key IDs or RID Protocol 2	√	D0	07	
Mifare Authenticate Block	√	2C	06	
Mifare ePurse Command	√	2C	0A	
Mifare Read Blocks	√	2C	07	
Mifare Write Blocks	√	2C	08	
NFC Commands	√	2C	08	
Pass-through Mode Start/Stop	√	2C	01	
Pass-through Mode (VP5300)		63	02	
PCD Single Command Exchange	√	2C	04	
Peer To Peer Send A Message	√	C7	9A	

Command	C'less or C'less + MSR	CMD	SUB CMD	Notes
Peer To Peer Receive A Message	√	C7	9B	
Play Audio		61	30	
Poll for Token with ATS		2C	0E	
QR Code Scan Start		61	41	
QR Code Scan Stop		61	42	
Reset Torn Transaction Log	√	84	0E	
Retrieve Log		4C	06	
RTC Get Date	√	25	04	d
RTC Get Time	√	25	02	d
RTC Set Date	√	25	03	d
RTC Set Time	√	25	01	d
Save/Delete Log		4C	05	
Set Data encryption Key Encryption Type	√	C7	32	
Set Baud Rate	√	30	01	
Set CA Public Key Protocol 1	√	24	01	
Set CA Public key Protocol 2	√	D0	03	
Set Cable Type	√	32	02	
Set Configurable AID	√	04	02	c
Set Configurable Group	√	04	03	c
Set Configuration	√	04	00	
Set Configuration Defaults and Keep Encryption Key	√	04	0A	
Set Data Encryption Enable Flag	√	C7	36	
Set Data Encryption Key Variant Type	√	C7	2F	
Set Logger Level		4C	02	
Set Merchant Record	√	04	11	
Set Model Number		90	15	
Set MSR Secure Parameters	√	C7	38	a
Set Parameter Defaults		04	09	
Set Poll Mode	√	01	01	
Set RF Error Reporting	√	17	03	
Set Serial Number	√	12	02	
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 Power Mode Suspend Configuration		F0	08	e
Stop Audio		61	30	
Stop Transaction	√	05	02	
Switches Notifications		F2	00	
Turn Off Yellow LED Command		F0	FA	a
Turn On Yellow LED Command		F0	FB	a
Update Balance Command	√	03	03	
Wake-Up Notification		F1	00	

a ViVOpay Vendi reader only
b Contact EMV products only
c Not in Global Reader Lite (GRL)
d Real Time Clock only
e Only applies to devices 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 Conifiguration		
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	√	c
03	05	Get All AIDs	√	c
03	06	Get Configurable Group	√	c
03	07	Get All Groups	√	c
03	0C	Get Cash Transaction Reader Risk Parameters	√	
03	0D	Get Cashback Transaction Reader Risk Parameters	√	
03	0E	Get DRL Reader Risk Parameters	√	
03	11	Get Merchant Record	√	
03	40	Get Transaction Result	√	
04	00	Set Configuration	√	
04	02	Set Configurable AID	√	c
04	03	Set Configurable Group	√	c
04	04	Delete Configurable AID	√	c
04	05	Delete Configurable Group	√	c
04	09	Set Parameter Defaults		
04	0A	Set Configuration Defaults and Keep Encryption Key	√	
04	11	Set Merchant Record	√	
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	20	Get Module Version 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	√	
14	01	Boot up Notification	√	

CMD	SUB CMD	Command	C'less or C'less + MSR	Notes
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	01	RTC Set Time	√	d
25	02	RTC Get Time	√	d
25	03	RTC Set Date	√	d
25	04	RTC Get Date	√	d
25	07	Get/Set 24-hr Self-Check Time	√	d
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	√	e
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		
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	0E	Poll for Token with ATS		
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	51	Get White List	√	
2C	53	Get Discretionary Data White List	√	
30	01	Set Baud Rate	√	

CMD	SUB CMD	Command	C'less or C'less + MSR	Notes
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		
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		
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	30	Stop Audio		
61	31	Play Audio		
61	41	QR Code Scan Start		
61	42	QR Code Scan Stop		
61	43	Camera Take Photo Start		
61	44	Camera Take Photo Stop		
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)		
77	81	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		

CMD	SUB CMD	Command	C'less or C'less + MSR	Notes
81	40	Get and Set MAC Verification Option (for data output)		
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		
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	3D	Read Log		
C7	9A	Peer To Peer Send A Message	√	
C7	9B	Peer To Peer Receive A Message	√	
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		
F0	00	Set PMC Status	√	
F0	01	Get PMC Status	√	
F0	02	Get Battery Level	√	
F0	08	Set USB Power Mode Suspend Configuration		e
F0	09	Get USB Power Mode Suspend Configuration		e
F0	0F	Shut Off the Power	√	
F0	F4	Configure Buttons Command		a
F0	F5	Get Button Configuration Command		a
F0	F6	Disable Blue LED Sequence		a
F0	F7	Enable Blue LED Sequence Command		a
F0	F9	LCD Display Clear Command		a
F0	FA	Turn Off Yellow LED Command		a
F0	FB	Turn On Yellow LED Command		a

CMD	SUB CMD	Command	C'less or C'less + MSR	Notes
F0	FC	LCD Display Line 1 Message Command		a
F0	FD	LCD Display Line 2 Message Command		a
F0	FE	Buzzer On/Off Command		a
F1	00	Wake-Up Notification		
F2	00	Switches Notification		

a ViVOpay Vendi reader only
c Not in Global Reader Lite (GRL)
d Real Time Clock only
e Only applies to devices 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

Poll for Token with ATS						2	2C	0E
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 White List	√	√	√	√	√	2	2C	51
Get Discretionary Data White List	√	√	√	√	√	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
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)

[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
	ViVOcomm activate transaction card type (ViVOcomm)
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)
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
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
A0h	The screen or object already exists
A1h	The SD card does NOT exist
A2h	Error decoding QR code

2.1.6. Status Codes for Protocol 1

Protocol 1 Status Codes

Status Code	Status
00h	OK
01h	Incorrect Frame Tag
02h	Incorrect Frame Type
03h	Unknown Frame Type
04h	Unknown Command
05h	Unknown Sub-Command
06h	CRC Error
07h	Failed
08h	Timeout
0Ah	Incorrect Parameter
0Bh	Command Not Supported
0Ch	Sub-Command Not Supported
0Dh	Parameter Not Supported / Status Abort Command
0Eh	Command not Allowed
0Fh	Sub-Command Not Allowed
57h	No payment occurred (VAS transaction)

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 terminal application to send serial commands in the correct sequence.
02h	Go to Contact Interface	The contactless transaction failed. *If the reader supports a contact interface,* advise the user to complete the transaction on the contact interface. 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.
03h	Transaction Amount is Zero	If the transaction amount is zero and the terminal is "an offline only terminal," the reader needs to terminate the transaction.
04h	Go To Other Interface	The transaction has failed. *If the reader supports another interface*, advise the user to complete the transaction on the other interface.
05h	Go To Nearby Interface	The transaction has failed. *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.
06h	Go To MagStripe Interface	The transaction has failed. *If the reader has a magstripe interface*, advise the user to complete the transaction using the magstripe interface.
20h	Card returned Error Status	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. How the terminal 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: **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):**

Error Code	Description	Reason for Error and Suggested Error Handling
		If RF State Code = PUT DATA (Ticket): The terminal can retry the transaction or abandon it. **If RF State Code = GEN AC:** For credit transactions: The terminal can retry the transaction or abandon it.
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 terminal 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 terminal contactless transaction limit (FFF1), the reader sends this error code back to the terminal 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 terminal 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 terminal 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.
30h	Card did not respond	Card was removed from the field or there was a communication error preventing the card response from reaching the reader. How the terminal 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: **RF State Code = PPSE:** The terminal can retry the transaction or abandon it. **If RF State Code = SELECT:** The terminal can retry the transaction or abandon it. **If RF State Code = GPO:** The terminal can retry the transaction or abandon it. **If RF State Code = READ RECORD:** The terminal can retry the transaction or abandon it. **If RF State Code = GEN AC:** For Credit transactions: The terminal can retry the transaction or abandon it. **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):**
41h	Data Element Missing	A mandatory/required data element was missing from the card.

Error Code	Description	Reason for Error and Suggested Error Handling
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 terminal/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)	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. The transaction continues to fail until the issuer public key and the CA public key are correct.
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.
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 terminal 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.

Error Code	Description	Reason for Error and Suggested Error Handling
62h	Generic Error	This is a generic or general error that is reported when a more specific reason for the error is not known.
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.
88h	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 terminal determines that the reader must perform exception processing, then the terminal must retry the transaction until the transaction has been completed successfully or the terminal 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 terminal may want to abort the retries even if the reader has not been able to complete exception processing. How and when the terminal 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

State Code	RF State	Description
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
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

Message	Code
EMV_RESULT_CODE_APPROVED_OFFLINE	0x0000
EMV_RESULT_CODE_DECLINED_OFFLINE	0x0001
EMV_RESULT_CODE_APPROVED	0x0002
EMV_RESULT_CODE_DECLINED	0x0003
EMV_RESULT_CODE_GO_ONLINE	0x0004
EMV_RESULT_CODE_CALL_YOUR_BANK	0x0005
EMV_RESULT_CODE_NOT_ACCEPTED	0x0006
EMV_RESULT_CODE_FALLBACK_TO_MSR (EMV_RESULT_CODE_USE_MSR)	0x0007
EMV_RESULT_CODE_TIMEOUT	0x0008
EMV_RESULT_CODE_AUTHENTICATE_TRANSACTION	0x0010
EMV_RESULT_CODE_UNABLE_TO_REACH_HOST	0x00FF
EMV_RESULT_CODE_FILE_ARG_INVALID	0x1001
EMV_RESULT_CODE_MEMORY_NOT_ENOUGH	0x2001
EMV_RESULT_CODE_FALLBACK_SITUATION	0x3004
EMV_RESULT_CODE_MSR_CARD_ERROR_FALLBACK	0x3012
EMV_RESULT_CODE_TIMEOUT_FOR_WAITING_ICC_INSERT_OR_MSR_SWIPE_FALLBACK	0x3013
EMV_RESULT_CODE_PARSING_TAGS_FAILED	0x5001
EMV_RESULT_CODE_CARD_DATA_ELEMENT_DUPLICATE	0x5002
EMV_RESULT_CODE_DATA_FORMAT_INCORRECT	0x5003
EMV_RESULT_CODE_APP_NO_TERM	0x5004
EMV_RESULT_CODE_APP_NO_MATCHING	0x5005
EMV_RESULT_CODE_MANDATORY_OBJECT_MISSING	0x5006
EMV_RESULT_CODE_APP_SELECTION_RETRY	0x5007
EMV_RESULT_CODE_AMOUNT_ERROR_GET	0x5008
EMV_RESULT_CODE_CARD_REJECTED	0x5009
EMV_RESULT_CODE_AIP_NOT_RECEIVED	0x5010
EMV_RESULT_CODE_AFL_NOT_RECEIVED	0x5011
EMV_RESULT_CODE_AFL_LEN_OUT_OF_RANGE	0x5012
EMV_RESULT_CODE_SFI_OUT_OF_RANGE	0x5013
EMV_RESULT_CODE_AFL_INCORRECT	0x5014

EMV_RESULT_CODE_EXP_DATE_INCORRECT	0x5015
EMV_RESULT_CODE_EFF_DATE_INCORRECT	0x5016
EMV_RESULT_CODE_CRYPTOGRAM_TYPE_INCORRECT	0x5018
EMV_RESULT_CODE_PSE_BY_CARD_NOT_SUPPORTED	0x5019
EMV_RESULT_CODE_USER_LANGUAGE_SELECTED	0x5020
EMV_RESULT_CODE_SERVICE_NOT_ALLOWED	0x5021
EMV_RESULT_CODE_NO_TAG_FOUND	0x5022
EMV_RESULT_CODE_CARD_BLOCKED	0x5023
EMV_RESULT_CODE_LEN_INCORRECT	0x5024
EMV_RESULT_CODE_CARD_COM_ERROR	0x5025
EMV_RESULT_CODE_ARC_NOT_PRESENCED	0x5028
EMV_RESULT_CODE_COMM_NO_ONLINE	0x5030
EMV_RESULT_CODE_TRAN_TYPE_INCORRECT	0x5031
EMV_RESULT_CODE_APP_BLOCK_AID	0x5036
EMV_RESULT_CODE_PAN_INVALID	0x5037
EMV_RESULT_CODE_TRANS_TYPE_NO_SUPPORT	0x5038
EMV_RESULT_CODE_CVM_TAG_8E_MISSING	0x6003
EMV_RESULT_CODE_CVM_TAG_8E_FORMAT_ERROR	0x6004

2.5. Serial Communication Interfaces

This section discusses the physical interfaces through which the terminal 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 device power on cycle, as this might cause unstable behavior.

2.6. Bluetooth Interface

Some ViVOpay devices support Bluetooth communication. Use the following command to set Bluetooth parameters, if applicable.

2.6.1. Set Bluetooth Parameters (77-81)

This 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)
ViVOtech2\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)

This 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	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	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)

This 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)
ViVOtech2\0	77h	83h	var		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)
ViVOtech2\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

2.7. 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 POS terminal communicate by exchanging command-response frames. The terminal always initiates communication by sending a command frame and ViVOpay 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 ViVOpay reader has received a command, the time required to respond to the terminal 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 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.

2.8.1.1. 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 device receives a report ID 1 frame, it should parse the report data to extract the command or response.

2.8.1.2. 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.

2.8.1.3. 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.

2.8.1.4. 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 device 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 device 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 Terminal.	CRC LSB if from ViVOpay. MSB if from Terminal.
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 Response Frame.

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 terminal/POS and a ViVOpay reader. Protocol 1 is retained for backward compatibility with older terminal/POS 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 ViVOpay reader:

- Pass-through mode allows the terminal/POS 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 device 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. Terminals can use Pass-Through Mode to communicate directly with a card. This feature allows a terminal 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 terminal 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 terminal 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 terminal must periodically instruct the ViVOpay reader to poll for cards. Whenever the ViVOpay 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 terminal 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 terminal to use low-level features provided by the ViVOpay 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 ViVOpay reader to poll for cards by sending the **Poll for Token** command. If no card is found within the time specified, the ViVOpay 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 ViVOpay 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.

3.2.4.1. 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.

3.2.4.2. For ISO I4443 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.

3.2.4.3. 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.

When done communicating with the card, the terminal is responsible for handling the termination sequence. The terminal may use the Antenna Disable/Antenna Enable commands to turn the RF field off and then on again.

The terminal 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 terminal communicates with the card in Pass-Through Mode and finds that it does not support the card, then the terminal is responsible for handling the termination sequence with the card. The terminal may keep sending Poll for Token commands to the ViVOpay 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 terminal may choose to terminate Pass-Through Mode as soon as it is reading is complete.
Care should be taken to ensure that the ViVOpay 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 terminal 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 DF7C 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 brand new transaction.

3.2.5.1. 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 DF7C TLV.

3.2.5.2. 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 DF7C 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 POS/Terminal in Pass-through mode.

3.3. Burst Mode

In Burst Mode (which requires auto poll mode to be set first, and is useful only for MSR data; see note below), a data frame is sent from the ViVOpay reader to the terminal each time a card is read successfully. The ViVOpay keeps polling for the supported RF cards. Whenever the ViVOpay reader detects a card in the RF field, it tries to read the card data.

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

Note: The reader must be in auto poll mode for Burst Mode to be used successfully. 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.

Burst Mode is enabled using 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 ViVOpay reader. In the command-response mode, the terminal sends a command to the ViVOpay reader and the ViVOpay reader responds in a pre-defined manner. These commands allow a terminal to use features provided by the ViVOpay reader, such as checking for the presence of the ViVOpay 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.

Note: Burst Mode is disabled for SRED devices.

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

3.3.1.1. Payload Frame (On Successful Read)

On successful read ViVOpay sends a card payload frame to the terminal 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>
```

3.3.1.2. 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)
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.

Data Item	Length (bytes)	Description
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
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

Data Item	Length (bytes)	Description
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 PayPass Third Party Data	F: b 32 T: '9F6E' L: 4-bytes F: b T '9F6E' L: 5-32 bytes	Indicates the form factor of the consumer payment device 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. 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
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.

3.3.1.3. NACK Frame

If the terminal 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 terminal. 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: ViVOpay receives NACK frame from terminal within 500ms after sending the original payload data, ViVOpay resends the card payload data.

Terminal **ViVOpay Reader**

Original Payload

◀---

| Time: 0

|

NACK sent at 400ms (within 500ms) |

---▶

| Time: 400ms

|

Resend Payload |

◀---

|

▼

Original Payload:

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

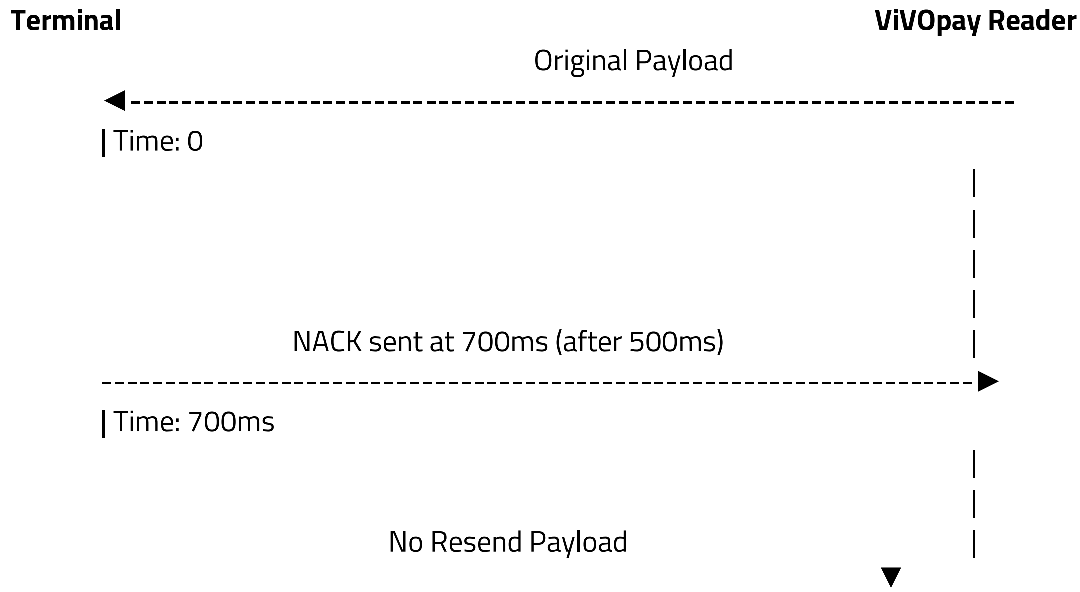
```
[01][00][01]%B5325350000623567^840SMITH/JOHN^0508501149256389247 3?
;5325350000623567=05081019492993892483? <CRC1><CRC2>
```

Resent payload:

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

```
[01][00][01]%B5325350000623567^840SMITH/JOHN^0508501149256389247 3?
;5325350000623567=05081019492993892483? <CRC1><CRC2>
```

Example 2: Reader receives NACK frame from terminal 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>`

3.3.1.4. Asynchronous UI Message Event

The asynchronous UI message event is used by the reader to indicate specific events to the terminal. 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 terminal to specify the required user experience on the terminal.

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.

3.3.1.5. 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	x16 + x12 + x5 + 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,
```

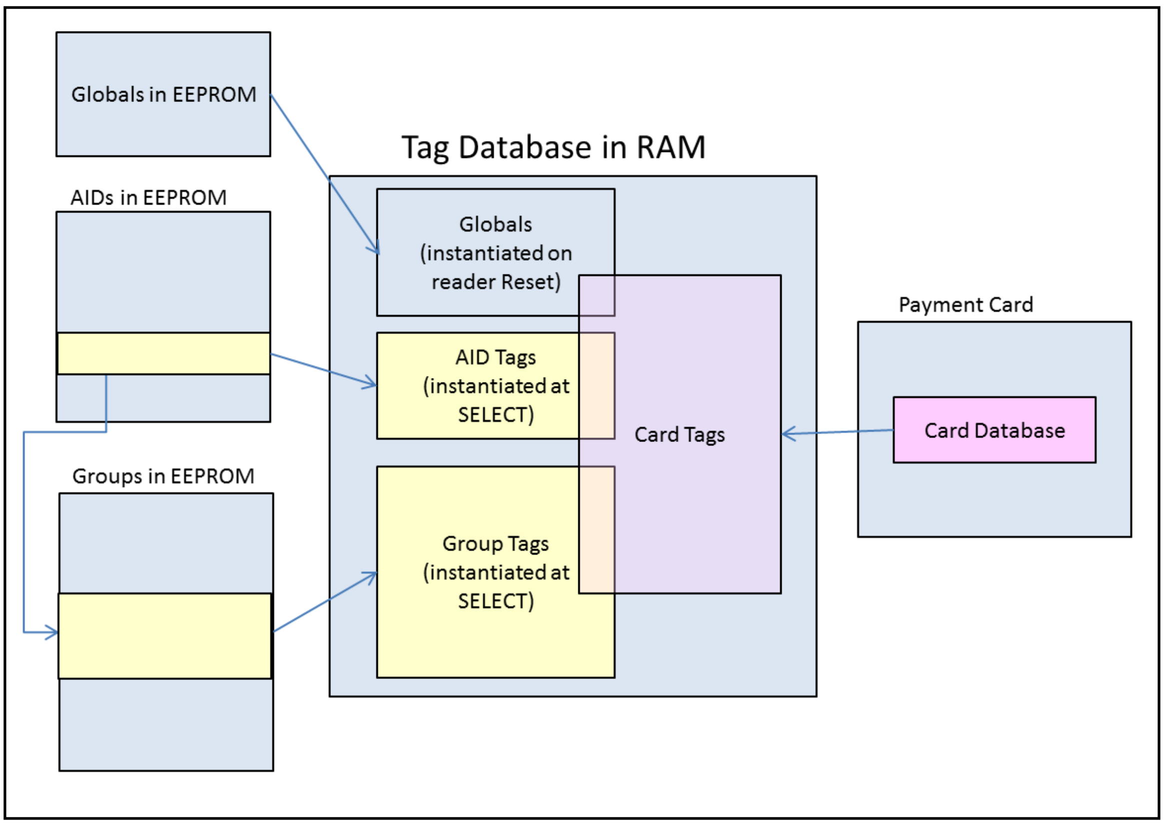
```
0x9DE8, 0x8DC9, 0x7C26, 0x6C07, 0x5C64, 0x4C45, 0x3CA2, 0x2C83, 0x1CE0, 0x0CC1,
0xEF1F, 0xFF3E, 0xCF5D, 0xDF7C, 0xAF9B, 0xBFBA, 0x8FD9, 0x9FF8, 0x6E17, 0x7E36,
0x4E55, 0x5E74, 0x2E93, 0x3EB2, 0x0ED1, 0x1EF0
};


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

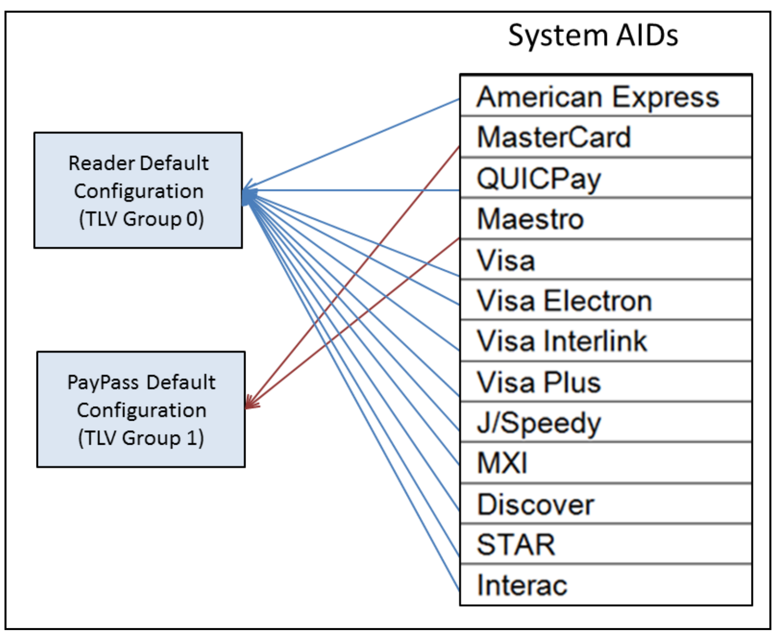
This section explains how to can create and modify application identifiers (AIDs) and associate them with TLV Groups in the reader's memory for specific types of transaction handling. Detailed descriptions of the configurable application identifier (CONAID) commands are also included.

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 1) 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 configurable application identifiers feature of the ViVOpay readers 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.

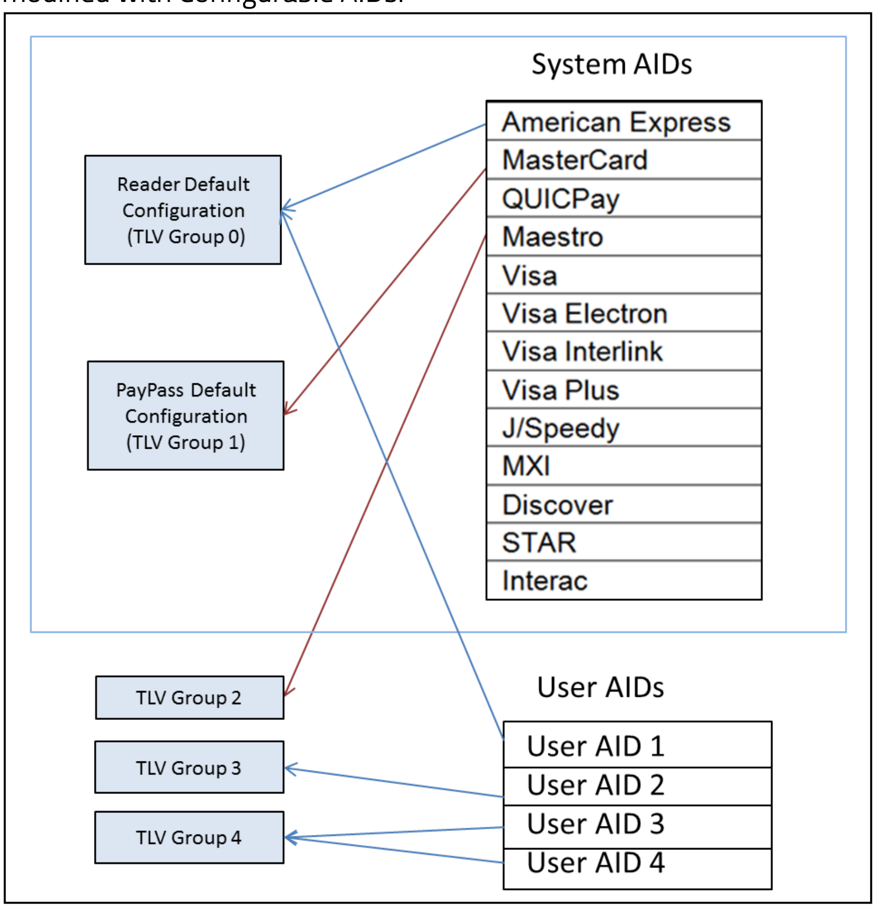
To create a new configurable AID, send to the reader both:

- 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
SmartTap2.1	A0 00 00 04 76 D0 00 01 11
AppleVas	4F 53 45 2E 56 41 53 2E 30 31

The terminal:

- 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 terminal:

- 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 Dpas	A0 00 00 01 52 30 10	B0
Interac	A0 00 00 02 77 10 10	C0
J/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 of the 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 1.

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 structure[2].

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 1. 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 1 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 1 through 7.
- SHOULD NEVER include the TDOL TLV if its length = zero (that is, only include the TDOL if it has a value).

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.

---

[2] PayPass specific tags are an exception to this rule. Those are maintained in Group 1.

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 regard 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 Terminal 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 1st 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
9F15	**Merchant Category Code** Classifies the type of business being done by the merchant, see ISO 8583:1993.	n4	2	00 00
9F1A	**Terminal Country Code** Indicates the country code of the terminal, represented according to ISO 3166.	n3	2	08 40
9F40	**Additional Terminal Capabilities** Indicates the data input and output capabilities of the terminal	b	5	60 00 00 30 00

Tag	Data Object Name and Description	Format	Length (Bytes)	Default Value
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	**Terminal 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 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
DFEE28	**Terminal Capabilities – No CVM Required** M/Chip v2.0 element indicating the terminal capabilities to be used when amount, authorized < cvm required limit. Formatted as terminal capabilities (tag '9F 33') Only byte 2 of this tag is actually used. The other terminal capabilities are configured using tag 9F33. DF28 Byte: 2 — PayPass Tag Equivalent: DF8119	b	3	00 08 E8
DFEE29	**Terminal Capabilities – CVM Required** M/Chip v2.0 element indicating the terminal capabilities to be used when amount, authorized >= cvm required limit. Formatted as terminal capabilities (tag '9F 33') Only byte 2 of this tag is actually used. The other terminal capabilities are configured using tag 9F33. DF29 Byte: 2 — PayPass Tag Equivalent: DF8118	b	3	00 68 E8
DFEE34[1]	**Terminal Contactless Transaction Limit** Indicates the terminal limit for this AID for contactless transactions.	n12	6	00 00 00 01 00 00

<table>
<tr>
<td>DFEE35[1]</td>
<td>
Visa Reader Risk Flags
Byte 1
<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>Status Check</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>-</td><td>RFU</td></tr>
</table>
Byte 2:
<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>Transaction Limit Check</td></tr>
<tr><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>X</td><td>-</td><td>CVM Required Limit Test</td></tr>
<tr><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>X</td><td>-</td><td>-</td><td>Terminal Floor Limit Check</td></tr>
<tr><td>-</td><td>-</td><td>-</td><td>-</td><td>X</td><td>-</td><td>-</td><td>-</td><td>Cash Transaction Reader Risk (RR)</td></tr>
<tr><td>-</td><td>-</td><td>-</td><td>X</td><td>-</td><td>-</td><td>-</td><td>-</td><td>Cashback Reader Risk (RR)</td></tr>
<tr><td>-</td><td>-</td><td>X</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>DRL (Dynamic Reader Limits) RR</td></tr>
<tr><td>-</td><td>X</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>Processing Restriction for Manual Cash</td></tr>
<tr><td>X</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>Processing Restriction for Cashback</td></tr>
</table>
Byte 3
<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>1 = online cryptogram required for zero amount (only used with zero</td></tr>
</table>
</td>
<td>b</td>
<td>3</td>
<td>00 06 01</td>
</tr>
</table>

<table>
<tr><th>Tag</th><th>Data Object Name and Description</th><th>Format</th><th>Length (Bytes)</th><th>Default Value</th></tr>
<tr><td></td><td>
<table>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>amount check enabled)</td></tr>
<tr><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>X</td><td>-</td><td>1 = perform zero amount check</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>
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</td><td></td><td></td><td></td></tr>
<tr><td>DFEE36[1]</td><td>CVM Required Limit
Indicates the CVM required limit in the terminal for the associated AIDs.</td><td>n12</td><td>6</td><td>00 00 00 00 80 00</td></tr>
<tr><td>DFEE37</td><td>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 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.</td><td>b</td><td>1</td><td>03</td></tr>
<tr><td>DFEE38[1]</td><td>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</td><td>b</td><td>1</td><td>00</td></tr>
<tr><td>DF8122[1]</td><td>Terminal Action Code (Online)
Reflect the acquirer-selected action to be taken upon analysis of the TVR.</td><td>b</td><td>5</td><td>F8 50 AC F8 00</td></tr>
<tr><td>DF8120[1]</td><td>Terminal Action Code (Default)
Reflect the acquirer-selected action to be taken upon analysis of the TVR.</td><td>b</td><td>5</td><td>F8 50 AC A0 00</td></tr>
<tr><td>DF8121[1]</td><td>Terminal Action Code (Denial)
Reflect the acquirer-selected action to be taken upon analysis of the TVR.</td><td>b</td><td>5</td><td>00 00 00 00 00</td></tr>
</table>

<table>
<tr><th>Tag</th><th>Data Object Name and Description</th><th>Format</th><th>Length (Bytes)</th><th>Default Value</th></tr>
<tr><td>DFEE44[1]</td><td>Application Capability (1:Support,0:Not Support)
Byte 1: (Leftmost)
<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>X</td><td>Normal J/Speedy support</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>X</td><td>ViVOpay Mifare for NFC</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>X</td><td>Interac support</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>X</td><td>CUP support</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>X</td><td>Google SmartTap</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>X</td><td>X</td><td>X</td><td>X</td><td>RFU</td><td></td><td></td><td></td><td></td></tr>
</table>
Byte 2:
<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>
Example: 0009 means reader support both MasterCard and Mobile J/Speedy applications</td><td>b</td><td>2</td><td>95 FF</td></tr>
<tr><td>DFEE5C</td><td>RF Deactivate Period</td><td>b</td><td>4</td><td>01 50 00 00</td></tr>
<tr><td>DFEE6B</td><td>Terminal IFD</td><td>an</td><td>8</td><td>00 00 00 00
00 00 00 00</td></tr>
<tr><td>DFEE70</td><td>Loyalty Program ID</td><td>b</td><td>1</td><td>00</td></tr>
<tr><td>DFEE71</td><td>Value Added Tax 1</td><td>b</td><td>2</td><td>00 00</td></tr>
<tr><td>DFEE72</td><td>Value Added Tax 2</td><td>b</td><td>2</td><td>00 00</td></tr>
</table>

Tag	Data Object Name and Description	Format	Length (Bytes)	Default Value
DFEE73	**Merchant Category Code**	b	2	00 00
DFEE74	**Discover Optional Features**	b	1	00
DFED18	**Poll Mode** Value = 00: Auto-Poll Value = 01: Poll on Demand	b	1	01
DFEE7E[1]	**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.	b	1	00
DFEE1C[1] [2] [3]	**LCD Font Size** Value = 02: Large Value = 03: Extra Large (default)	b	1	03
DFEF29[1] [2]	**LCD delay time (ms)** Default is 1000ms. If the device has no LCD, then the value will be 0.	b	2	00 00
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	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

[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 terminal 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.

[3] These objects only work on the Vendi.

4.3.2. Master Group Configuration TLVs

Master use 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 American Express 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. 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.	n2	1	80
5F57	**Account Type**	B	1	
9F01	**Acquirer Identifier**	n12	6	
9F40	**Additional Terminal Capabilities** Indicates the data input and output capabilities of the terminal.	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	**Terminal Capabilities - CVM Required**	b	1	60
DF8119	**Terminal 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 *of the terminal/POS*. It is configured by the POS and is unique to the terminal.	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 terminal/reader in the case of a mag-stripe mode transaction when the *amount, authorized (numeric)* is greater than the *reader CVM required limit*.	b	1	10
DF812C	**MagStripe No CVM Required Capability** Indicates the CVM capability of the terminal/reader in the case of a mag-stripe mode transaction when the *amount, authorized (numeric)* is less than or equal to the *reader CVM required limit*.	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 terminal *for the associated AIDs*. Default = $10.00	n12	6	00 00 00 00 10 00
DF811F	**Security Capabilities**	b	1	08
DF8120 [1]	**Terminal 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]	**Terminal 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]	**Terminal Action Code (Online)** Reflect the acquirer-selected action to be taken upon analysis of the TVR.	b	5	00 00 00 00 00
9F33	**Terminal Capabilities** Indicates the card data input, CVM, and security capabilities of the terminal. This tag (9F33) only configures bytes 1 and 3 of the terminal capabilities. Byte 2 of the terminal capabilities actually comes from DF28 or DF29 during the transaction. Refer to tags DF28 and DF29 for details. **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.	b	3	
9F1A	**Terminal Country Code** Indicates the country code of the terminal, represented according to ISO 3166.	n3	2	08 40
9F1C	**Terminal 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	**Terminal Type** Indicates the environment of the terminal, 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. For PayPass M/Chip, this value should be set to 03 (EMEA). It defaults to ViVOpay for backward compatibility with MagStripe applications.	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 [3]	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 terminal 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.

[3] These objects only work on the Vendi.

Phone Message Table – Hard-Coded Default Value in Kernel

PCI Mask	PCII 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)

4.3.1. 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.

Vendi 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 a 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 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 1st 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. Make sure to 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) 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	**Terminal Country Code** Indicates the country code of the terminal, represented according to ISO 3166. Default = US	n3	2	08 40
9F33	**Terminal Capabilities** Indicates the card data input, CVM, and security capabilities of the terminal Default = ▪ No CVM required ▪ SDA supported ▪ DDA supported ▪ Card Capture ▪ CDA supported	b	3	00 08 E8

<table>
<tr><th>Tag</th><th>Description</th><th>Format</th><th>Length</th><th>Default Value in Group 2</th></tr>
<tr><td>9F35</td><td>Terminal Type
Indicates the environment of the terminal, 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
Online only
Offline with online capability
Offline only</td><td>
11
12

13</td><td>
21
22

23</td><td></td></tr>
<tr><td>Unattended
Online only
Offline with online capability
Offline only</td><td>
14
15

16</td><td>
24
25

26</td><td>
34
35

36</td></tr>
</table></td><td>n2</td><td>1</td><td>25</td></tr>
<tr><td>9F40</td><td>Additional Terminal Capabilities
Indicates the data input and output capabilities of the terminal.</td><td>b</td><td>5</td><td>60 00 00 30 00</td></tr>
<tr><td>9F6D</td><td>Contactless Reader Capabilities</td><td>n2</td><td>1</td><td>C0</td></tr>
<tr><td>9F6E</td><td>Enhanced Contactless Reader Capabilities</td><td>b</td><td>4</td><td>58 80 00 00</td></tr>
<tr><td>DFEE34</td><td>Terminal Contactless Transaction Limit
Indicates the terminal limit for this AID for Contactless transactions.</td><td>n12</td><td>6</td><td>00 00 00 01 00 00</td></tr>
<tr><td>DFEE5C</td><td>RF Deactivate Period</td><td>b</td><td>4</td><td>01 50 00 00</td></tr>
<tr><td>DF8126 [1]</td><td>CVM Required Limit
Indicates the CVM required limit in the terminal for the associated AIDs.</td><td>n12</td><td>6</td><td>00 00 00 00 80 00</td></tr>
<tr><td>DF8123</td><td>Reader Contactless Floor Limit</td><td>n12</td><td>6</td><td>00 00 00 00 60 00</td></tr>
<tr><td>DF8122 [1]</td><td>Terminal Action Code (Online)
Reflect the acquirer-selected action to be taken upon analysis of the TVR.</td><td>b</td><td>5</td><td>F8 50 AC F8 00</td></tr>
<tr><td>DF8120 [1]</td><td>Terminal Action Code (Default)
Reflect the acquirer-selected action to be taken upon analysis of the TVR.</td><td>b</td><td>5</td><td>F8 50 AC A0 00</td></tr>
<tr><td>DF8121 [1]</td><td>Terminal Action Code (Denial)
Reflect the acquirer-selected action to be taken upon analysis of the TVR.</td><td>b</td><td>5</td><td>00 00 00 00 00</td></tr>
</table>

[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 terminal 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.

4.3.2. 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.

Vendi 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 regard as not present.

Vendi 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. 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.	n2	1	B0

<table>
<tr><th>Tag</th><th>Description</th><th>Format</th><th>Length</th><th>Default Value in Group 2</th></tr>
<tr><td>9F02</td><td>Amount Authorized (Numeric)</td><td>n12</td><td>6</td><td>00 00 00 00 00 01</td></tr>
<tr><td>9F03</td><td>Amount Other (Numeric)</td><td>n12</td><td>6</td><td>00 00 00 00 00 00</td></tr>
<tr><td>9F09</td><td>Application Version Number</td><td>b</td><td>2</td><td>01 00</td></tr>
<tr><td>9F1A</td><td>Terminal Country Code
Indicates the country code of the terminal, represented according to ISO 3166. Default = US</td><td>n3</td><td>2</td><td>08 40</td></tr>
<tr><td>9F1B</td><td>Terminal Floor Limit
Indicates the floor limit in the terminal for the AID(s) associated with this group.
Note: The value is the decimal limit amount given in binary represented in Hex in the command/response. (60 limit = 6000 decimal = 1770h).</td><td>b</td><td>4</td><td>00 00 17 70</td></tr>
<tr><td>9F33</td><td>Terminal Capabilities
Indicates the card data input, CVM, and security capabilities of the terminal
Default =
▪ No CVM required
▪ SDA supported
▪ DDA supported
▪ Card Capture
▪ CDA supported</td><td>b</td><td>3</td><td>00 08 E8</td></tr>
<tr><td>9F35</td><td>Terminal Type
Indicates the environment of the terminal, 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
Online only
Offline with online capability
Offline only</td><td>
11
12

13</td><td>
21
22

23</td><td></td></tr>
<tr><td>Unattended
Online only
Offline with online capability
Offline only</td><td>
14
15

16</td><td>
24
25

26</td><td>
34
35

36</td></tr>
</table></td><td>n2</td><td>1</td><td>25</td></tr>
<tr><td>9F66</td><td>Terminal Transaction Qualifier (TTQ)</td><td>b</td><td>4</td><td>80 00 40 00</td></tr>
</table>

Tag	Description	Format	Length	Default Value in Group 2
5F2A	**Transaction Currency Code** Indicates the currency code of the transaction according to ISO 4217. Make sure to 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
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
DF8122 [1]	**Terminal 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]	**Terminal 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]	**Terminal Action Code (Denial)** Reflect the acquirer-selected action to be taken upon analysis of the TVR.	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-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 terminal *for the associated AIDs.*	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 terminal 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.

4.3.3. 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.

Vendi rule for Inetarc 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 1st 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. Make sure you use the same Transaction Currency Code for all configurable AIDs (default = US Dollars).	n3	2	01 24
9F09	**Application Version Number**	b	2	00 02
9F1A	**Terminal Country Code** Indicates the country code of the terminal, represented according to ISO 3166. Default = US	n3	2	01 24

<table>
<tr><th>Tag</th><th>Description</th><th>Format</th><th>Length</th><th>Default Value in Group 4</th></tr>
<tr><td>9F35</td><td>Terminal Type
Indicates the environment of the terminal, 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
Online only
Offline with online capability
Offline only</td><td>
11
12

13</td><td>
21
22

23</td><td></td></tr>
<tr><td>Unattended
Online only
Offline with online capability
Offline only</td><td>
14
15

16</td><td>
24
25

26</td><td>
34
35

36</td></tr>
</table></td><td>n2</td><td>1</td><td>22</td></tr>
<tr><td>9F58</td><td>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).</td><td>n1</td><td>1</td><td>03</td></tr>
<tr><td>9F59</td><td>Terminal Transaction Information (TTI)
Provides terminal transaction information for the current transaction (Interac).</td><td>b</td><td>3</td><td>DC 87 00</td></tr>
<tr><td>9F5A</td><td>Terminal Transaction Type (Interac)
▪ 0x00 = Purchase
▪ 0x01 = Refund</td><td>b</td><td>1</td><td>00</td></tr>
<tr><td>9F5D</td><td>Terminal Contactless Receipt Required Limit
Limit amount used to compare against transaction amount to automatically print a transaction record (Interac).</td><td>n12</td><td>6</td><td>00 00 00 00 50 00</td></tr>
<tr><td>9F5E</td><td>Terminal Option Status
Options supported by the terminal (Interac).</td><td>b</td><td>2</td><td>E0 00</td></tr>
<tr><td>9F5F</td><td>Terminal (Reader) Contactless Floor Limit
Floor limit amount used to compare against transaction amount (Interac).</td><td>n12</td><td>6</td><td>00 00 00 00 50 00</td></tr>
<tr><td>9F1B</td><td>Terminal Floor Limit
Indicates the floor limit in the terminal for the AID(s) associated with this group.

Note: The value is the decimal limit amount given in binary represented in hex in the command/response. (60 limit = 6000 decimal = 1770h).</td><td>b</td><td>4</td><td>00 00 00 00</td></tr>
</table>

Tag	Description	Format	Length	Default Value in Group 4
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 terminal *for the associated AIDs.*	n12	6	00 00 00 00 60 00
DF8122 [1]	**Terminal 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]	**Terminal 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]	**Terminal 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
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

Tag	Description	Format	Length	Default Value in Group 4
9F33	**Terminal Capabilities** Indicates the card data input, CVM, and security capabilities of the terminal Default = ▪ 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 *Note:* 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 terminal 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.

4.3.4. 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

<table>
<tr><th>Tag</th><th>Data Object Name</th><th>Usage</th><th>Description</th><th>Format</th><th>Length</th></tr>
<tr><td>9F06</td><td>Application Identifier (AID)</td><td>MAND</td><td>Identifies the application as described in ISO/IEC 7816-5.
This must be the 2nd TLV in the data field.</td><td>b</td><td>5 – 16</td></tr>
<tr><td>DF7C</td><td>Auto-Switch</td><td>OPT</td><td>Automatically switch to Pass-Through Mode when PICC is unknown.
00h = disabled (default)
01h = enabled</td><td>b</td><td>1</td></tr>
<tr><td>FFE0[1]</td><td>Registered Application Provider Identifier (RID)</td><td>Sys = NEVER
User = OPT</td><td>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.</td><td>b</td><td>5</td></tr>
<tr><td>FFE1[1]</td><td>Partial Selection Allowed</td><td>OPT (Visa MAND)</td><td>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.</td><td>b</td><td>1</td></tr>
<tr><td>FFE2[1]</td><td>Application Flow</td><td>Sys = NEVER

User = MAND</td><td>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) –SmartTap</td><td>b</td><td>1</td></tr>
<tr><td>FFE3</td><td>Selection Features</td><td>OPT</td><td>Enables or disables selection features. For M/Chip 3.0, this value will default to 74h.
Refer to the Selection Features section for a detailed description of this tag.

<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></td><td>b</td><td>1</td></tr>
</table>

<table>
<tr>
<td>FFE4[1]</td>
<td>TLV Group Number</td>
<td>MAND</td>
<td>The TLV Group number that contains the characteristics for this AID
This must be the 1st TLV in Data Field.
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.</td>
<td>n2</td>
<td>1</td>
</tr>
<tr>
<td>FFE5[1]</td>
<td>Maximum AID Length</td>
<td>DEP</td>
<td>This value must be <= 16.
For Visa application flow, this value is set to 16 and cannot be changed.
Note: This tag must be included if the FFE1 Partial Select TLV is included.</td>
<td>b</td>
<td>1</td>
</tr>
<tr>
<td>FFE6[1]</td>
<td>AID Disabled</td>
<td>OPT</td>
<td>Used to disable a System AID (has no effect on a User AID). 80h = disabled and 00h = enabled</td>
<td>b</td>
<td>1</td>
</tr>
<tr>
<td>FFE8</td>
<td>Exclude from Processing</td>
<td>OPT</td>
<td>This byte is formatted as follows:
<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>
</td>
<td>b</td>
<td>1</td>
</tr>
<tr>
<td>FFE9</td>
<td>Transaction Type List</td>
<td>OPT</td>
<td>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:
<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>
Group 0 may not be used in this list.</td>
<td>b</td>
<td>Variable <= 24</td>
</tr>
</table>

<table>
<tr><td>FFEA</td><td>Configurable Kernel Identifier</td><td>OPT</td><td>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:
<table><tr><td>Card Application</td><td>Default KID</td></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>
This tag is equivalent to MasterCard tag DF810C.</td><td>b</td><td>1</td></tr>
<tr><td>DFEF2C</td><td>Terminal AID List</td><td>OPT</td><td>Tells the reader to allow Terminal AID List support during the initial select process. 01 = Allowed, 00 = Disabled</td><td>b</td><td>1</td></tr>
</table>

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	

Name	Tag	Length (Hex)	Value (Hex)	Application Name
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	
Max AID Length	FFE5/DFEE2E	01	10	
APPLICATION	FFE2/DFEE4C	01	06	
Group	FFE4/DFEE2D	01	90	VISA Electron
AID	9F06	07	A0 00 00 00 03 20 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	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	

Name	Tag	Length (Hex)	Value (Hex)	Application Name
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	
Group	FFE4/DFEE2D	01	B0	Discover Expresspay (ZIP)
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	
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	**SmartTap**
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	

Name	Tag	Length (Hex)	Value (Hex)	Application Name
Exclude from Processing	FFE8/DFEE53	01	04	
APPLICATION	FFE2/DFEE4C	01	23	

4.4. 2-byte tags vs. 3-byte tags

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 device 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 device, 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 device 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	Terminal contactless transaction limit, never used for PayPass.
FFF2	9F1E	Terminal 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	Terminal Action Code (Online).
FFFE	DF8120	Terminal Action Code (Default).
FFFF	DF8121	Terminal 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 terminal 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	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. 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.
-----x--	Cardholder Confirmation Not Supported	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.
----x---	API Required	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.
---x----	Invalid AID Allowed	0 = any invalid AID will cause this AID to terminate the transaction. 1 = any invalid AID will be ignored as related to this AID.
--x-----	Duplicate AID Allowed	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.
-x------	Enable Kernel ID	1 = allow the evaluation of the Kernel ID. 0 = if a Kernel ID is provided by the card it is ignored.
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.
xxxxxx--	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 matched by any Terminal AID.

One by one, the terminal will check each DFEF2C tag associated with the Terminal AID. If the Tag DFEF2C value is '01', the terminal 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 = Terminal will not send out Terminal AID data 1 = Terminal 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 1.

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. It 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. The DF68 tag is used to enable or disable the **Stop** command. The default value for this tag is "disabled" (0).

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.

6.1.7.1. 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)

6.1.7.2. 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 1

NOTE: Protocol 1 is deprecated. This section is provided for reference only and will eventually be eliminated. Protocol 2 is now the supported default protocol. For currently supported commands in Protocol 2, see Command Reference: Protocol 2.

7.1. Transaction Related Commands (Protocol 1)

7.1.1. Flush Track Data (17-02)

This command allows the POS application to instruct the reader to flush any track data that was previously read from a card but has not yet been transferred to the POS. On receiving this command, the reader clears any pending card data.

Command Frame from PC to ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Sub-Command	Data1	Data2	CRC (LSB)	CRC (MSB)
ViVOtech\0	'C'	17h	02h	00	00		

ACK Frame from ViVOpay Reader (or NACK)

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Status	Data1	Data2	CRC (MSB)	CRC (LSB)
ViVOtech\0	'A'	17h	Status=OK	Unused	XX		

7.1.2. Get Full Track Data (17-CD)

Use this command to return full track data from a ViVOpay reader. If a card has been swiped at the magnetic stripe reader or presented to a reader in Auto Poll mode, the reader sends back an ACK frame followed by a data frame containing track data. If no card has been swiped or presented, the reader returns an ACK frame and no data frame. If the reader returns both Track 1 and Track 2 data, the data frame contains the Track 1 data, followed by a NULL character (00h) marking the end of Track 1 data, followed by Track 2 data.

If a card has been swiped, but an error occurred, then ViVOpay just sends an ACK frame with a **Failed** status.

Command Frame from PC to ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Sub-Command	Data1	Data2	CRC (LSB)	CRC (MSB)
ViVOtech\0	'C'	17h	CDh	00	00		

Response Frame (ACK)

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Status	Data1	Data2	CRC (MSB)	CRC (LSB)
ViVOtech\0	'A'	17h	Status	Tracks/ Error Code	DataLen		

Byte 12 is used for tracks or error codes, depending on the value of the status in Byte 1. When the status is **OK**, Byte 12 is used to store tracks. When the status is **Failed**, Byte 12 is used to store the error codes from ViVOpay.

Note: These Error Codes are valid only when the RF Error Code Reporting is enabled through Set RF Error Reporting command, see Set RF Error Reporting.

Get Full Track Data Error Codes

Status	Tracks/Error Code
OK	Examples: Bit 0 = Track 1 Track = 00h => No Track Data Bit 1 = Track 2 Track = 01h => Track 1 Data Only Bit 3 = Track 3 Track = 02h => Track 2 Data Only Track = 03h => Track 1 & Track 2 Data and so on. Bit 7 = Card Type Card Type = 0 => Contactless Transaction Card Type = 1 => Swiped Magnetic Stripe
Failed	Error Code = 01h Card Removed Error Code = 02h Communication Error Error Code = 03h Protocol Error Error Code = 04h Multiple Cards Detected Error Code = 05h Card Not Accepted Error Code = 06h Bad Data Error Code = FFh Unknown Error
Other (See Status Code)	N/A

DataLen

This is the number of data bytes in the data frame to follow. This does not include the frame tag, frame type, or CRC bytes.

Data Frame from ViVOpay Reader to PC (If the reader sent an ACK and track data is available)

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)	CRC (LSB)
ViVOtech\0	'D'	Data	Data	…	Data		

If the host fails to receive the track data, it can send a NACK frame to request the reader to resend the track data. To ensure that the reader resends the track data, the NACK frame must be received within 500ms after it sends the original track data. If the reader receives the NACK Frame within that time period, it first resends the ACK Frame followed by the data frame to PC. If the reader receives the NACK frame after 500ms of sending out the original track data, or if a new card has been detected, the reader sends an ACK or NACK frame to the host and does not resend the track data. Each data payload is only re-sent one time.

Response Frame NACK Frame from PC to the ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Status	Data1	Data2	CRC (MSB)	CRC (LSB)
ViVOtech\0	'N'	17h	00h	00	00		

ACK Frame from the ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Status	Data1	Data2	CRC (MSB)	CRC (LSB)
ViVOtech\0	'A'	17h	Status	Tracks	DataLen		

Status: OK (see Status Code Protocol 1)

Tracks	**Example**
Bit 0 = Track 1	Track = 00h => No Track Data
Bit 1 = Track 2	Track = 01h => Track 1 Data Only
Bit 3 = Track 3	Track = 02h => Track 2 Data Only

Track = 03h => Track 1 & Track 2 Data and so on.

DataLen: Number of data bytes in the data frame to follow. This does not include the frame tag, frame type and checksum bytes.

Data Frame from the ViVOpay Reader to PC (If reader sent an ACK and track data is available)

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).	CRC (LSB).
ViVOtech\0	'D'	Data	Data	...	Data		

An example of the data frame sent in response to the **Get Full Track Data** command is as follows:

```
56 69 56 4F 74 65 63 68 00 44 42 35 33 32 35 33
35 30 30 30 30 36 32 33 35 36 37 5E 53 4D 49 54
48 2F 4A 4F 48 4E 5E 30 35 30 38 35 30 31 31 30
30 36 32 36 30 30 34 34 35 30 30 30 30 30 37 38
36 32 30 39 33 33 00 35 33 32 35 33 35 30 30 30
30 36 32 33 35 36 37 3D 30 35 30 38 31 30 31 39
34 34 35 39 39 37 38 36 30 36 32 33 DF 03
```

Annotated

`56 69 56 4F 74 65 63 68 00` – FRAME TAG

`44` – FRAME TYPE 'D'

TRACK 1 DATA

```
42 35 33 32 35 33 35 30 30 30 30 36 32 33 35 36
37 5E 53 4D 49 54 48 2F 4A 4F 48 4E 5E 30 35 30
38 35 30 31 31 30 30 36 32 36 30 30 34 34 35 30
30 30 30 30 37 38 36 32 30 39 33 33
```

`00` – END OF TRACK 1 START OF TRACK 2

TRACK 2 DATA

```
35 33 32 35 33 35 30 30 30 30 36 32 33 35 36 37
3D 30 35 30 38 31 30 31 39 34 34 35 39 39 37 38
36 30 36 32 33
```

`DF 03` – CRC

7.1.3. Get ViVOpay Firmware Version (29-00)

Use this command to retrieve the ViVOpay reader's firmware version number. This is the Protocol 1 version of the command given in Get Version: Protocol 2 (29-00). The ViVOpay reader returns an ACK frame containing the length of the version data. This is followed by a data frame containing the firmware version information.

Command Frame from PC to ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Sub-Command	Data1	Data2	CRC (LSB)	CRC (MSB)
ViVOtech\0	'C'	29h	00h	00	00		

Response Frame from ViVOpay Reader (ACK or NACK)

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Status	Data1	Data2	CRC (MSB)	CRC (LSB)
ViVOtech\0	'A'	29h	Status=OK	Unused	DataLen		

Status: OK (see Status Code Protocol 1)

DataLen: Number of data bytes in the data frame to follow. This does not include the frame tag, frame type and checksum bytes.

Data Frame from ViVOpay Reader to PC (If the reader sent an ACK)

Byte 0-8	Byte 9	Byte 10 … Byte n+10	Byte n+11	Byte n+12
Frame Tag	Frame Type	Data 0 … Data n	CRC (MSB)	CRC (LSB)
ViVOtech\0	'D'	ViVOpay Version (Null Terminated ASCII String)		

7.2. Key Manager Commands Protocol 1

Note: The following commands use Protocol 1 frame formats. Whenever possible, instead use the Key Manager commands in Protocol 2.

The Key Management Protocol 1 commands are retained for compatibility purposes and are not to be used when doing secure communication.

Some ViVOpay firmware versions that support EMV security features provide an EMV Key Management interface that a terminal can use to add or delete CA public keys and related data. These firmware versions also provide a Real Time Clock set up interface and an EMV ViVOpay Terminal set up interface.

This document describes the ViVOpay Serial Interface—specifically the EMV Key Management commands, Real Time Clock setup commands, and EMV ViVOpay Terminal setup commands. It describes the communication parameters, the ViVOpay Serial Interface Protocol and the command-specific details.

Warning: DO NOT mix the two Key Management formats. Manage the keys using Protocol 1 or Protocol 2, but not both.

ViVOpay provides a secure storage environment on its Crypto Chip for storing the certification authority public keys. It allows for storage of up to a maximum of 30 keys which are uniquely identified as a key index in each payment scheme (RID). The basic key management functions provided are setting of a new public key based on a Unique <RID, Key Index> Pair and deletion of a key. After a key has been stored in the Crypto Chip, it does not allow retrieval of the key. All authentication/decryption functions that require the key take place inside the Crypto Chip.

The ViVOpay reader periodically checks for Command Frames. After it starts receiving a Command Frame, it expects each successive byte to arrive within the inter-byte timeout. If the

ViVOpay reader is receiving multiple data frames it expects each successive byte to arrive within the inter-frame timeout (see table below). If the data is not received within the timeout period listed, the ViVOpay reader times out and a timeout failure response is sent.

Serial Commands	Inter-Byte	Inter-Frame
Delete CA Public Key	200ms	5 sec
Set CA Public Key	200ms	5 sec
Set RTC Date	200ms	5 sec
Set RTC Time	200ms	5 sec
Pass-Through APDU Exchange (multiple Data Frames)	200ms	5 sec
Pass-Through PCD Single Command (multiple Data Frames)	200ms	5 sec
All other Commands	200ms	200ms

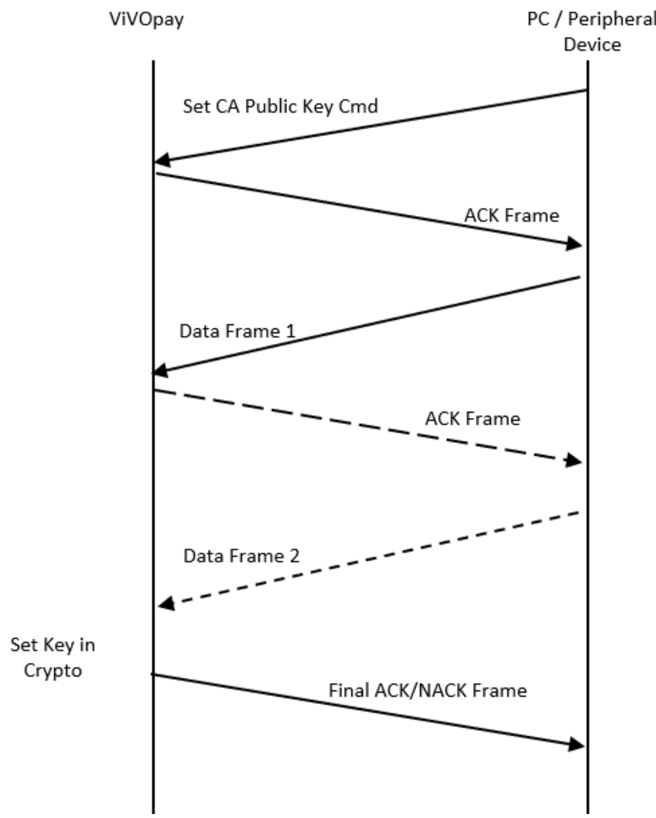
After the ViVOpay reader has received a command, the time in which it starts sending a response back to the terminal varies from command to command, depending on what kind of processing is required before a response can be sent back to the terminal.

EMV Key Management Commands Error Codes – Protocol 1

Error Code	Description
00h	No Error
01h	Unknown Error
02h	Invalid Data
03h	Incomplete Data
04h	Invalid Key Index
05h	Invalid CA Hash Algorithm Indicator
06h	Invalid CA Public Key Algorithm Indicator
07h	Invalid CA Public Key Modulus Length
08h	Invalid CA Public Key Exponent
09h	Key already Exists (try to set key after deleting existing key)
0Ah	No Space for New Rid
0Bh	Key not Found
0Ch	Crypto Chip not Responding
0Dh	Crypto Chip Communication Error
0Eh	RID Key Slots Full
0Fh	No Free Key Slots Available

7.2.1. Set CA Public Key (24-01) Protocol 1

Use this command to send data related to a CA public key to the ViVOpay reader for storing in a secure environment (Crypto Chip memory). The public key is uniquely identified by the <RID, Key Index> pair. If the total length of the key-related data sent is more than 244 bytes, it can be broken down into two data frames.



Flow of frames between ViVOpay reader and an external device

Command Frame from Terminal to ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Sub-Command	Data1	Data2	CRC (LSB)	CRC (MSB)
ViVOtech\0	'C'	24h	01h	DataLen2	DataLen		

DataLen, DataLen2

- If the key data is being sent in a single data frame, then DataLen contains the length of the one and only Data Frame to follow and DataLen2 is 0.
- If the key data is being sent in two data frames, then DataLen contains the length of the first data frame and DataLen2 contain the length of the second data frame. The length of either data frame must not exceed 244 (0xF4) bytes.
- DataLen > 0, DataLen2 >= 0

ACK Frame from ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Status	Data1	Data2	CRC (MSB)	CRC (LSB)
ViVOtech\0	'A'	24h	Status	00	00		

Status: OK (or see Status Code Protocol 1)

NACK from ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Status	Data1	Data2	CRC (MSB)	CRC (LSB)
ViVOtech\0	'N'	24h	FAILED	Error Code	Unused		

Error Code: See EMV Key Management Error Codes Table

If at any time ViVOpay sends back a NACK Frame with status set to **Failed**, the error code field indicates the reason for failure.

First Data Frame from terminal to ViVOpay Reader (If reader sent an ACK)

Byte 0-8	Byte 9	Byte 10-13	Byte 14	…	Byte 10+(n-1)	Byte 10+n	Byte 10+(n+1)
Frame Tag	Frame Type	Data 0	Data 1	…	Data (n-1)	CRC (LSB)	CRC (MSB)
ViVOtech\0	'D'	Data	Data	…	Data		

Where n = length of the data field.

the data field in the first data frame contains the complete or partial CA public key related data. The complete contents and format of the key data are given in the following table. The data portion of data frame 1 and data frame 2 (if present) when stripped of the frame overhead and concatenated, provides the data as given in the following table.

Set CA Public Key Data Field

Data Byte	Name	Length (bytes)	Format	Notes
0-4	RID	5	Binary	Registered Identifier. Necessary for Unique Identification.
5	CaPublicKey Index[1]	1	Binary	Index of the CA Public Key for this RID. Necessary for Unique Identification.
6	CaHashAlgoIndicator[1]	1	Binary	CA Hash Algorithm to produce Hash-Result in digital signature scheme. Valid values: 01h: SHA-1

7	CaPublicKeyAlgoIndicator[1]	1	Binary	Digital signature algorithm to be used with CA public key. Valid values: 01h: RSA
8-27	CaPublicKeyChecksum[1]	20	Binary	CA public key checksum
28-31	CaPublicKeyExponent[1]	4 (PICC-based Length may be 1 or 3)	Binary	CA public key exponent. Value can be 3 (Len=1 Byte) or 2^{16}+1=65537=010001h (Len=3 Bytes). We consider it as a 32-bit (4-Byte) big-endian number for the serial interface and crypto storage. The PICC may consider it as a 1-Byte or 3-byte number.
32,33	CaPublicKeyModulusLen	2	Binary	CA public key (modulus) length stored as a big-endian number. Aka N_{CA}
34	CaPublicKeyModulus[1]	Variable (max 256)	Binary	CA public key (modulus) with length=N_{CA}

[1]: Fields specified by EMV that need to be stored in Terminal Memory (See EMV2000, Book 2, Section 11.2.2 Table 23)

ACK Frame from ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Status	Data1	Data2	CRC (MSB)	CRC (LSB)
ViVOtech\0	'A'	24h	Status=OK	00	00		

Status: OK (or see Status Code Protocol 1)

NACK from ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Status	Data1	Data2	CRC (MSB)	CRC (LSB)
ViVOtech\0	'N'	24h	FAILED	Error Code	Unused		

Error Code: see EMV Key Management Error Codes Table

Second Data Frame from Terminal to ViVOpay Reader (If the reader sent an ACK, and Data remains to be sent)

Byte 0-8	Byte 9	Byte 10-13	Byte 14	...	Byte 10+(p-1)	Byte 10+p	Byte 10+(p+1)
Frame Tag	Frame Type	Data 0	Data 1	...	Data (p-1)	CRC (LSB)	CRC (MSB)
ViVOtech\0	'D'	Data	Data	...	Data		

Where p = DataLen2 > 0

If the second data frame is sent, then the data field in this frame contains the remaining CA public key related data.

On receiving valid data, the reader sends it to the Crypto Chip for secure storage. The Crypto Chip checks the data and stores it in its memory. If the CA public key is stored successfully in the Crypto Chip memory, the reader returns an ACK frame. If for any reason the CA public key is not stored, the reader returns a NACK frame.

Final ACK Frame from ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Status	Data1	Data2	CRC (MSB)	CRC (LSB)
ViVOtech\0	'A'	24h	Status=OK	00	00		

Status: OK (or see Status Code Protocol 1)

Final Nack Frame from ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Status	Data1	Data2	CRC (MSB)	CRC (LSB)
ViVOtech\0	'N'	24h	FAILED	Error Code	Unused		

Error Code: See EMV Key Management Error Codes Table

7.2.2. Delete CA Public Key (24-02) Protocol 1

Use this command to instruct the ViVOpay reader to delete a previously set CA public key from within secure storage in the Crypto Chip. The key is uniquely identified by the <RID, Key Index> pair.

When this command is received, the reader waits for a data frame containing the RID and key index. It then instructs the Crypto Chip to delete the specified CA public key. Depending on the result of this operation, the reader returns an ACK or NACK Frame.

Command Frame from Terminal to ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Sub-Command	Data1	Data2	CRC (LSB)	CRC (MSB)
ViVOtech\0	'C'	24h	02h	00	DataLen=6		

ACK Frame from ViVOpay Reader (or NACK)

ACK Frame

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Status	Data1	Data2	CRC (MSB)	CRC (LSB)
ViVOtech\0	'A'	24h	Status=OK	00	00		

Status: OK (or see Status Code Protocol 1)

NACK Frame

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Status	Data1	Data2	CRC (MSB)	CRC (LSB)
ViVOtech\0	'N'	24h	FAILED	Error Code	Unused		

Error Code: See EMV Key Management Error Codes Table

Data Frame from Terminal to ViVOpay Reader (If reader sent an ACK)

Byte 0-8	Byte 9	Byte 10	...	Byte 14	Byte 15	Byte 16	Byte 17
Frame Tag	Frame Type	Data 0	...	Data 4	Data 5	CRC (LSB)	CRC (MSB)
ViVOtech\0	'D'	RID [0]	...	RID [4]	Key Index		

RID: Registered Identifier (5 Bytes)
Key Index: Key Index (1 Byte)

The RID, together with the key index, specifies a unique key stored in ViVOpay secure memory.

Final ACK Frame from ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Status	Data1	Data2	CRC (MSB)	CRC (LSB)
ViVOtech\0	'A'	24h	Status=OK	00	00		

Status: OK (or see Status Code Protocol 1)

Final NACK Frame from ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Status	Data1	Data2	CRC (MSB)	CRC (LSB)
ViVOtech\0	'N'	24h	FAILED	Error Code	Unused		

Error Code: See EMV Key Management Error Codes

7.2.3. Delete All CA Public Keys (24-03) Protocol 1

Use this command to instruct the ViVOpay reader to delete all previously set CA public keys from within secure storage in the Crypto Chip. The keys are deleted regardless of the <RID, Key Index> pair.

When the reader receives this command, it instructs the Crypto Chip to delete all CA public keys. Depending on the result of this operation, the reader returns an ACK or NACK Frame.

Command Frame from Terminal to ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Sub-Command	Data1	Data2	CRC (LSB)	CRC (MSB)
ViVOtech\0	'C'	24h	03h	00	00		

ACK Frame from ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Status	Data1	Data2	CRC (MSB)	CRC (LSB)
ViVOtech\0	'A'	24h	Status=OK	00	00		

Status: OK (or see Status Code Protocol 1)

NACK Frame from ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Status	Data1	Data2	CRC (MSB)	CRC (LSB)
ViVOtech\0	'N'	24h	FAILED	Error Code	Unused		

Error Code: See EMV Key Management Error Codes Table

7.3. Miscellaneous Protocol 1 Commands

7.3.1. Set RF Error Reporting (17-03)

This command allows the POS application to enable or disable RF error code reporting for the **Get Full Track Data** command. When RF error code reporting is enabled, any RF error codes are reported to the POS application through the ACK frame for the Get Full Track Data command.

Command Frame from PC to the ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Sub-Command	Data1	Data2	CRC (LSB)	CRC (MSB)
ViVOtech\0	'C'	17h	03h	Operation Code	XX		

Operation Code:

00h: Disable RF Error Code Reporting
01h: Enable RF Error Code Reporting
02h or others: No change

ACK Frame from the ViVOpay Reader

Byte 0-8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Frame Tag	Frame Type	Command	Status	Data1	Data2	CRC (MSB)	CRC (LSB)
ViVOtech\0	'A'	17h	Status=OK	RF Error Code Reporting Status	XX		

RF Error Code Reporting Status (only for ACK frame):

00h: RF Error Code Reporting disabled
01h: RF Error Code Reporting enabled

8. Protocol Command Reference: Protocol 2

8.1. RTC (Real Time Clock) Commands

8.1.1. 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 device 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)
ViVOtech2\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 (UTC+8) – 8hrs = 18:30 (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 (UTC-7) + 7 hrs = 10:30 (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 (LSB)	CRC (MSB)
ViVOtech2\0	25h	See Status Code Table	00h	02h	BCD format	BCD format		

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

8.2. General Commands

8.2.1. Ping (18-01)

The **Ping** command checks if the ViVOpay reader is connected to the terminal. If the ViVOpay 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

8.2.2. 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 0xFE: 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):

```
5669564f746563683200c70000210001000101000201ff1001001101001201001301001401001501001601001f01ff780f
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
```

8.2.3. Set Poll Mode (01-01)

The **Set Poll Mode** command allows the terminal to set the ViVOpay reader polling mode. The ViVOpay reader functions in one of two polling modes: Auto Poll or Poll on Demand. The value is saved in nonvolatile memory so you only need to send this command when you want to change the mode.

ViVOpay products operate in Poll on Demand Mode by default. Use the Poll on Demand Mode when you want the reader to poll for cards only when requested to by the terminal. In this mode the ViVOpay 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 terminal 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 terminal 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 ViVOpay unit 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 ViVOpay reader is connected to a POS terminal via the terminal'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.

8.2.4. 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.

There are three 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 terminal does not trigger these messages.
- **Indexes 08h to 0Bh** are messages triggered by the terminal.
- **Index FFh** indicates that the command is only setting the buzzer and/or LEDs. The terminal does not display a message.

After completing a successful transaction, the "Thank You" message remains on the LCD until the terminal 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	**NOTE:** For the complete list of possible messages, see Appendix A-10. Messages 00-07 are normally controlled by the reader. 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) Messages 08-FF are controlled by the terminal 08: Insert or Swipe card (Use Chip & PIN) 09: Try Again(Tap Again) 0A: Tells the customer to present only one card (Present 1 card only) 0B: Tells the customer to wait for authentication/authorization (Wait) FF indicates the command is setting the LED/Buzzer only.
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

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.

8.2.5. 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: ViVOpay readers may not support all these 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 ViVOpay reader and the date and time from the RTC chip is used. When the reader is configured to use external time, the terminal needs to set up the RTC inside the ARM processor of the ViVOpay reader with the Set Configuration (04-00) command.

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

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

If configured to use internal power LEDs, the ViVOpay 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 terminal 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.

8.2.6. 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 (MSB)	CRC (LSB)
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	AEh	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)

8.2.7. 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	AEh	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)

8.2.8. 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		

8.2.9. 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.

8.2.10. 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		

8.2.11. 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	Byte14	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 (LSB)	CRC (MSB)
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	PCD-ID	01	N	80152100 Rev.A
2	PCD-Hardware-ID	02	N	80152110 Rev.A
3	PCD-Software-ID	03	N	80152120 Rev.A
4	Reader Tech Name and Version	04	N	80152100 Rev.A

5	Application Type	05	N	Mastercard Contactless Reader Specification, v3.1.1
6	Application Selection Mechanism	06	N	PPSE Module v1.2
7	Master Card CL Kernel	07	N	MasterCard PayPass M/Chip 3.1 v1.3
8	Operation System	08	N	uCOS3 v1.31
9	Firmware Version	09	N	VP5300 FW v1.00.S
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

8.2.12. Get USB Boot Loader Version (29-04)

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

Note: VP3300C not supported because that reader does not support contact EMV.

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		

8.2.13. 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"

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

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

Note: VP3300C not supported.

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"

8.2.15. 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

8.2.16. 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.

Note: VP3300C not supported.

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"

8.2.17. 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"

8.2.18. Get TMS (29-20)

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

Note: Not all devices support TMS, but if the device 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	Device 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	Device Uptime since Last Reboot	0D	4	Time (Seconds)
14	Device 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

8.2.19. 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:

1. 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
2. Use Data_Shared_Key and AES-128 CBC algorithm to Encrypt the Raw Data to be 96 bytes Enc_Health_Data.

8.2.20. Set Baud Rate (30-01)

The **Set Baud Rate** command instructs the reader to change its baud rate to the specified value. If the command frame is valid and the ViVOpay 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 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	01h	00h	01h	Baud Rate Code		

Baud Rate Code	Baud Rate
01h	9600 baud
02h	19200 baud
03h	38400 baud
04h	57600 baud
05h	115200 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.

8.2.21. Set Temporary Baud Rate (30-02)

This command instructs the 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 ViVOpay 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.

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	02h	00h	01h	Baud Rate Code		

Baud Rate Code	Baud Rate
01h	9600 baud
02h	19200 baud
03h	38400 baud
04h	57600 baud
05h	115200 baud

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

8.2.22. Set Baud Rate and Audio Level (30-03)

The **Set Baud Rate and Audio Level** command instructs the VP3300AJ (and UniPay 1.5) to change baud rate to the specified value. If the command frame is valid and the ViVOpay 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 terminal, 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.

8.2.23. 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		

8.2.24. 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 (LSB)	CRC (MSB)
ViVOtech2\0	32h	00h	00h	01h	Cable Type		

8.2.25. 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 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		

8.2.26. 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 (LSB)	CRC (MSB)
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`.

8.2.27. 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 (LSB)	CRC (MSB)
ViVOtech2/0	D1h	See Status Code Table	00h	06h	Ethernet MAC Address(hex)	Varies	Varies

8.2.28. 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)
ViVOtech2/0	D1h	15h	00h	00h	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 (LSB)	CRC (MSB)
ViVOtech2/0	D1h	See Status Code Table	00h	varible	Data (See D1-13 command)	Varies	Varies

8.2.29. Set Network Configuration (D1-13)

The **Set Network Configuration** command sets the static network DHCP for the device. 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.

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. Maxminum length is 16 bytes including null terminator.
Variable	Net mask, zero-terminated ASCII. Maxminum length is 16 bytes including null terminator.
Variable	Gateway, zero-terminated ASCII. Maxminum length is 16 bytes including null terminator.
Variable	DNS, zero-terminated ASCII. Maxminum 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 (LSB)	CRC (MSB)
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

8.2.30. 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	Byte14-15	Byte 16	Byte 17
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	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 (LSB)	CRC (MSB)
ViVOtech2/0	D1h	See Status Code Table	00h	00h	Varies	Varies

Example:

Set Wi-Fi TCP Server Mode

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

8.2.31. 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	18h	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 (LSB)	CRC (MSB)
ViVOtech2/0	D1h	See Status Code Table	00h	Variable	Data (See D1-19 command)	Varies	Varies

8.2.32. 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	13h	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. Maxminum length is 32 bytes including null terminator.
Variable	Password of the target AP, zero-terminated ASCII. Maxminum length is 32 bytes including null terminator.
Variable	IP address, zero-terminated ASCII. Maxminum length is 16 bytes including null terminator.
Variable	Net mask, zero-terminated ASCII. Maxminum length is 16 bytes including null terminator.
Variable	Gateway, zero-terminated ASCII. Maxminum 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 (LSB)	CRC (MSB)
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

8.2.33. 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 (LSB)	CRC (MSB)
ViVOtech2/0	D1h	See Status Code Table	00h	varible	Data (See D1-17 command)	Varies	Varies

8.2.34. 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		

8.2.35. 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

8.2.36. 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	'90'	'15'	'00'	'0X'	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		

8.3. Configurable AID and Group Commands

8.3.1. Set Configurable AID (04-02)

This 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 group that has been created containing all of the characteristics desired for this AID. Setting and configuring the TLV Group Number is explained below. The TLV Group Number must be configured first. If an AID is communicated referring to a non-existing group, that AID is rejected.

Registered Application Provider Identifier (RID) – The parameter is optional. If it is provided, this number is used to reference the CA Public Key payment system. If it is not provided the first five bytes of the AID are used.

For 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.

There are System AIDs in the reader. These can be disabled but cannot be deleted.

For 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 a TLV Group (in the TLV Group Number TLV) that already exists
- 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		

The TLV Data Objects that can be set using this command are defined in the AID Configuration Tags.

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

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		

8.3.2. Set Configurable Group (04-03)

This command creates or modifies a TLV Group. You configure a specific TLV Group by passing the TLVs with the desired functionality and a unique TLV Group Number to the reader. The TLVs that can be associated with a TLV Group are listed below. 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.

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		

If you are changing TLVs in Group 0 (the Default Group) the reader retains and uses the old values of any TLVs not included in the Set Configurable Group command. If you are changing any other Group, 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, simply pass on the desired values in the TLV. To disable the default TDOL, send a TDOL TLV with Length set to zero and no Value field included. This instructs the reader to delete any existing TDOL list for this group.

The TLV Data Objects that can be set using this command are given in the Configurable Group Tags Table (for non- PayPass applications) or the 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		

8.3.3. Get Configurable AID (03-04)

This command returns the configurable (User) AID parameters. The user MUST send an AID TLV in the command, 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) – terminal	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 TLV that is returned in the data section of the **Response Frame** 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.

8.3.4. Get Configurable Group (03-06)

Use this command to return 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 TLV that is returned in the data section of the **Response Frame** are given in the Group TLV Objects Table.

If the user requests a Group that is illegal, an error response is sent back.

If the user requests a valid Group number but the Group does NOT exist, then 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 then it will not be present in the data returned by this command. However, this does not mean that there is no value for this particular TLV in order to perform a transaction. It is possible that the PayPass Kernel may have a hard-coded value for this TLV which will be used 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 given in in the section on PayPass Group Configuration TLVs with Hard-Coded Values in Kernel.

8.3.5. Delete Configurable AID (04-04)

This command deletes a configurable AID. It is MANDATORY to include the AID TLV of the AID to be removed. No other TLVs should be included.

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 can be set using this command is below.

9F06	Application Identifier (AID) – terminal	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 it continues to reside in the reader's database).

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

8.3.6. Delete Configurable Group (04-05)

Use this command to delete a configurable Group. This means that this Group can no longer be used to 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 Terminal 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.

8.3.7. Get All AIDs (03-05)

Use this command to return all AIDs in the reader. This command may be used to verify configured AIDs or to determine what 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 only Data Objects that should be returned from the GAA command are AID Tags. The reader sends out ALL TLV 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.

8.3.8. Get All Groups (03-07)

This command returns all Groups in the reader. This command may be used 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 only Data Objects that should be returned from the GAG command are Group Tags. These are the same as those itemized in the Group Configuration Tags table.

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.

8.4. Transaction Related Commands: Contact

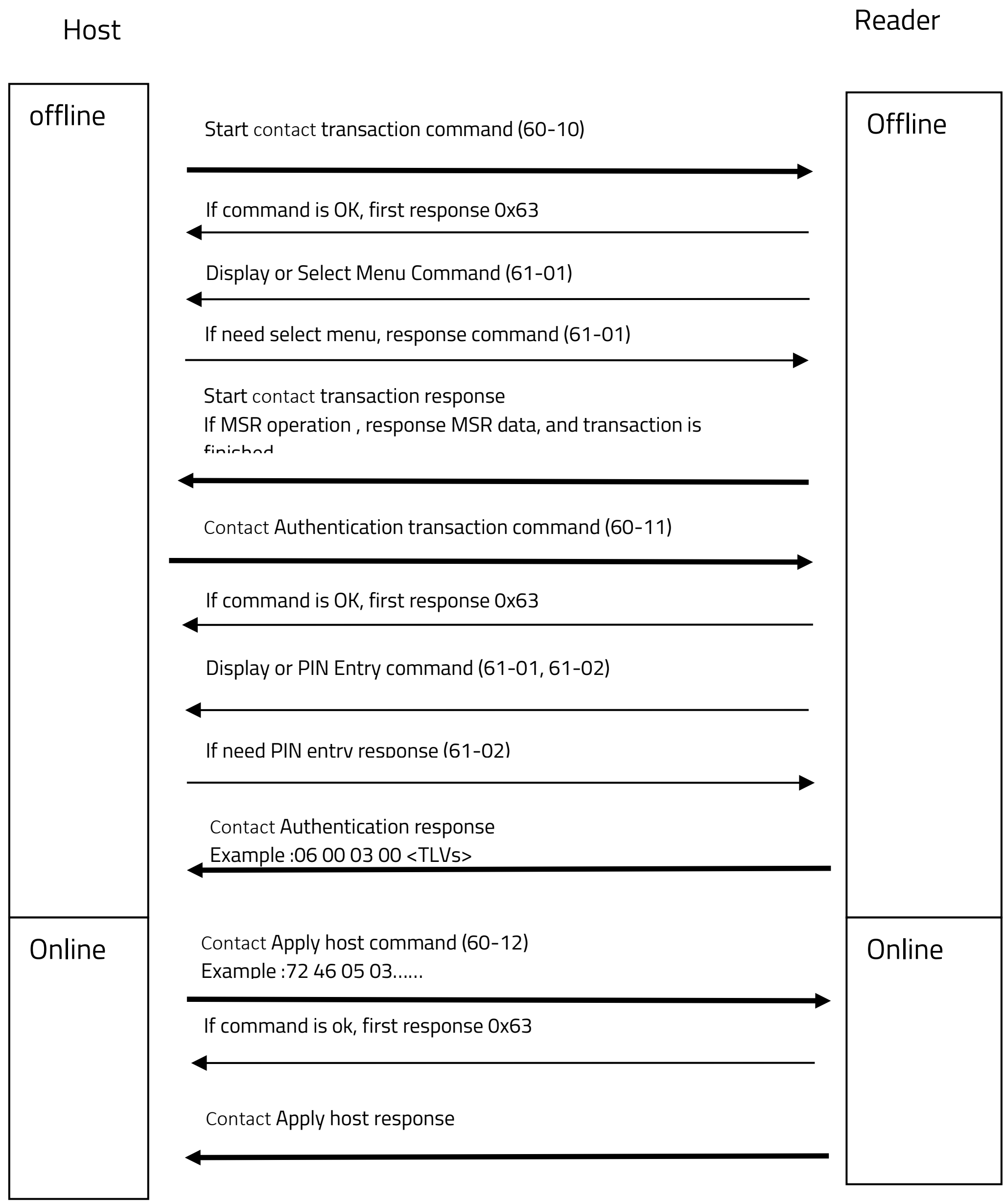
This section describes contact EMV commands applicable to NEO products that support contact EMV. For contactless transactions, see the chapter following this one.
Note that VP3300-series products should generally use the 02-40 Activate Transaction command in production. (The older 02-01 command is supported only for non-encrypted readers.)

8.4.1. Setting transaction parameters

The following commands can be used to set transaction parameters.

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

8.4.2. Contact EMV L2 Transaction Flow



1. Send Start Transaction Command. This could be 60-10 (for contact EMV in VP3300 products) or 02-40 (all modes, all products). Device will respond with
2. Send Authenticate Transaction Command.
3. If received online request from terminal, send Apply Host Response command.
4. When a transaction is completed, interface will output a TLV format data list, and transaction result (approve, or decline, or…) will display on the LCD.
5. If you want to review the transaction result, you can send Retrieve Transaction Result Command. It will output 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.

8.4.3. Contact Retrieve Application Data (60-01)

This command returns the User AID parameters. The host must send 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 AID List / Application Data does not exist, status code is 0x60.

8.4.4. Contact Remove Application Data (60-02)

This 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 length is 00 then remove 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 AID List / Application Data not exist, status code is 0x60.
- If format was error, status code is 0x05.

8.4.5. Contact Set Application Data (60-03)

This command creates a new AID configuration. Maximum is 16 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 a <TLV> format was error, status code is 0x05.
- If AID List has full (MAX is 16), 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	Terminal application version number	2	00 96
5	5F36	Transaction Currency Exponent	1	02
6	9F1B	Terminal 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

8.4.6. Contact Retrieve Terminal Data (60-04)

Use this command to return all TLVs containing terminal data (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	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 Terminal Data does not exist, status code is 0x60.

8.4.7. Contact Remove Terminal Data (60-05)

Use this command to delete terminal 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 Terminal Data does not exist, status code is 0x60.

8.4.8. Contact Set Terminal Data (60-06)

This command creates new terminal data according to the TLVs passed in. The terminal data is mandatory and seldom changes. It represents configuration data that usually gets set one time, in pre-production, and never changes thereafter.

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, status code is 0x05. If flash error, status code is 0x62.

Terminal Data List Example

Data ID	Tag	Value name	Length (Byte)	Data
1	5F36	Transaction Currency Exponent	1	02
2	9F1A	Terminal County Code	2	08 40
3	9F35	Terminal Type	1	21
4	9F33	Terminal Capability	3	60 28 C8
5	9F40	Additional Terminal 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	Terminal 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	Terminal 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	Terminal Support ASI	1	01
16	DFEE16	Terminal Encrypt Mode	1	00
17	DFEE17	Terminal Entry Mode for ICC	1	07
18	DFEE18	Terminal Encrypt Mode for MSR	1	80
19	DFEE1E	Contact Terminal 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 PIn. byte 3 -> MSR	3	32 3C 3C

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 cannot, in general, 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 device 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	Terminal 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	

x	x	x	x	x	x	0	x	RFU	
x	x	x	x	X	X	x	0	RFU	

9F35

Terminal Type

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

9F40

Additional Terminal Capabilities

Byte 1

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	

Byte 2

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	

Byte 3

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	

5

	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 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	
DF11	Transaction Log Support (Default: Enable) (Major) 0 → Disable 1 → Enable	1
DF26	Revocation List Support (Default: Enable) (Major) 0 → Disable 1 → Enable	1
DF27	Exception File Support (Default: Disable) (Major) 0 → Disable 1 → Enable	1
DFEE1E	Contact Terminal 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 | 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 | 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	Terminal action code support	Major
x	1	x	x	x	x	x	x	Terminal action code can be change	Major
x	x	1	x	x	x	x	x	Terminal 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	

8.4.9. Contact Retrieve AID List (60-07)

Use this command to return all AIDs list in the reader. This command may be used to verify configured AIDs in 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	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 Number of AID Blocks.
- <AID Block> format is <LenL> <LenH> <Several bytes AID>

Note: If AID List does not exist, status code is 0x60.

8.4.10. Contact Retrieve CA Public Key (60-08)

This command can get assign CA Public key form 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	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 this key is used. Currently support only one type: RSA. The value is set to 0x01.
- <HashValue>: Which is 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, status code is 0x60.
- If CA Key Index does not exist, status code is 0x60.

8.4.11. Contact Remove CA Public Key (60-09)

This command allows the host 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 length is 00 then remove all CA Public Key.

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, status code is 0x60.
- If CA Key Index does not exist, status code is 0x60.

8.4.12. Contact Set CA Public Key (60-0A)

This command adds a new key to reader. Maximum is 16 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	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:

- <Hash Algorithm>: The only algorithm supported is SHA-1.The value is set to 0x01
- <Encryption Algorithm>: The encryption algorithm in which this key is used. Currently support only one type: RSA. The value is set to 0x01.
- <HashValue>: Which is calculated using SHA-1 over the following fields: RID & Index & Modulus & Exponent
- <Public Key Exponent>: Actually, 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>** 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.

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, status code is 0x61.
- If CA Key Hash Data is an error, status code is 0x17.

8.4.13. Contact Retrieve CA Public Key List (60-0B)

This command can get all RID and key index for 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	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 any CA Key does not exist, status code is 0x60.

8.4.14. Contact Retrieve Certification Revocation List (60-0C)

This 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 have no CRL exist, status code is 0x60.

8.4.15. Contact Remove Certification Revocation List (60-0D)

This 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 Length is 00 then remove all Certification Revocation List.

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			

8.4.16. Contact Set Certification Revocation List (60-0E)

This command adds a new entry to the revocation list. The new entry is added at the end of the 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	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> (Option): <Low byte of length> <High byte of length> →
Fix is 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			

Note: Supported max CRL number is 90. If exceed max number, status code is 0x61.

8.4.17. Contact Remove Transaction Amount Log (60-0F)

This command can delete transaction amount log in reader. (When EMV transaction is offline approved, or online, transaction amount log saves to 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	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			

8.4.18. Contact Start Transaction (60-10)

Start a new contact EMV L2 transaction (ICC + MSR) or start 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

Note: Not supported in VP3300C because the device does not support contact EMV.

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 indicate it support FallBack to MSR, 0x00 indicate it not support FallBack.
- <TimeOut1> (2 bytes, unit is Second). Timeout for Card is seated.

- <TimeOut2> (2 bytes, unit is Second). Waiting time till "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	2
4	9A	Transaction Date. **Deprecated.** **NOTE: Do not attempt to use this tag in SRED devices. It may still work in non-SRED older products, but support for this tag in Activate Transaction will be dropped in future versions of firmware.**	3
5	9F21	Transaction Time **Deprecated.** **NOTE: Do not attempt to use this tag in SRED devices. It may still work in non-SRED older products, but support for this tag in Activate Transaction will be dropped in future versions of firmware.**	3
6	9F03	Amount, Other(Numeric)	6
7	DFEE1A	Output Tag List (Options)	N
8	DFEF1F	Auto Authenticate (Options) If auto authenticate set to 1, the reader will do command 60-11 automatic. Byte 1: Auto Authenticate enable 1: enable, 0 disable Byte 2: Force Online 1: enable, 0 disable	2

DFEE1A usage rules

- If DFEE1A in terminal data, all 3 transaction commands response same output tags
- If DFEE1A in transaction commands, it should replace the DFEE1A in terminal data, and response different tags.
- If DFEE1A is in transaction commands and terminal data, it should replace DFEE1A in the terminal data with the transaction command setting and respond to different tags.
- If DFEE1A is not exist in terminal data and not exist in transaction commands, the response output tags should be default output tags.

If command parse is successful and ICC transaction starts, response contains first command and status code is 0x63.

If command parse fails, response is other status code and end transaction.
"No terminal data" response 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 often needs to display messages and allow key-in.

When transaction needs display or key-in, device sends command 61-01 to host.
If host response is needed, host will respond to command 61-01 and send result to reader.
On Start Transaction success, send second response and 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	If encryption is not enabled, this tag is not present.
Contact Response	Variable	See Contact Response Code

Data Item	Length (bytes)	Description
Code (DFEE25)		
TLV Data	Variable	Output default Start Transaction 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
TLV Encrypt Information (DFEE26)	Variable	Encryption and Data Capture Information. Length: 1 or 2 (variable) NEOII the length must be 2 bytes 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

Data Item	Length (bytes)	Description
		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

MSR 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 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
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

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: 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 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 Encrypt Information (DFEE26)	Variable	Encryption and Data Capture Information. Length: 1 or 2 (variable) Values:

Data Item	Length (bytes)	Description
		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 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

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

8.4.19. Contact Authenticate Transaction (60-11)

This function should be called after the Start Transaction function.
The function will continue to perform 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 response code is 00 04, then need go online process and send Apply Host Response command.
Note: VP3300C does not support this because VP3300C does not support Contact EMV.

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 Second).means terminal 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 in terminal data, all 3 transaction commands response same output tags
- If DFEE1A in transaction commands, it should replace the DFEE1A in terminal data, and response different tags.
- If DFEE1A is in transaction commands and terminal data, it should replace DFEE1A in the terminal data with the transaction command setting and respond to different tags.
- If DFEE1A is not exist in terminal data and not exist in transaction commands, the response output tags should be default output tags.

If command parse is successful and ICC transactions continue, response is first command and status code is 0x63.
If command parse fails, response is some other status code and end of 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 display or Get PIN, reader sends command 61-01, 61-02 to host.
If host response is needed, host should respond with command 61-01, 61-02 and send result to reader.

Authenticate Transaction success: send second response and 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		

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: 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 0 1 0 1 = Verifone FPE

Data Item	Length (bytes)	Description
		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	If encrypt disable, this tag is not present.
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.
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	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 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	Terminal 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	Terminal Country Code	2
9F1C	Terminal 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	Terminal Capabilities	3
9F34	Cardholder Verification Method(CVM) Results	3
9F35	Terminal 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	PIN KSN	10 or 12

8.4.20. Contact Apply Host Response (60-12) – Complete Transaction

This function is the last step in EMV transaction flow. It is the Completion step.
The function will send acquirer data (if online) to the card and notify that the transaction is complete. This function will do the following process upon the transaction type (may or may not perform each step, depending upon acquirer's requirement and response):

(1) External Authenticate
(2) Script Processing
(3) 2nd Gen AC
(4) Completion

Note: VP3300C does not support contact EMV, hence it does not support this 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	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 went online with host,0x00 indicates unable to go online.
- Data in [ ] indicate these data are 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 command parse is successful, and ICC transaction continues, response is first command and status code is 0x63.

If command parse has failed, 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, device sends command 61-01 to host. If the device needs a host response, host should respond with command 61-01 and result to 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
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 0 0 0 1 = AES 0 0 1 0 = TransArmor Algorithm 0 0 1 1 = Voltage Algorithm 0 1 0 0 = Visa FPE

Data Item	Length (bytes)	Description
		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	If encryption is not enabled, this tag is not present.
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	Terminal 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

8.4.21. Contact Retrieve Transaction Result (60-13)

After a contact EMV transaction is completed, you can use this command 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 will return TLV format. Supported tags refer to "Option Data List" Table in Section "Reference Data List".

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: 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

Data Item	Length (bytes)	Description
		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 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	If encrypt disable, this tag is not present.
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						

8.4.22. Contact Get Reader Status (60-14)

This command can get ICC card power and card seat status.

Note: VP3300C does not support this 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	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

8.4.23. Contact Get ICS Identification (60-15)

Contact Common EMV L2 approved configurations of certification are 1C, 2C, 3C, and 4C. This command can get identification of ICS terminal configuration 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	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

8.4.24. Contact Set ICS Identification (60-16)

Contact Command EMV L2 includes 4 approved configurations of certification. This command can set identification of ICS terminal configuration to reader (1C, 2C, 3C, or 4C). This command affects command Contact Set Terminal Data (60-06). Generally, you will use 60-16 first, then 60-06.

Examples:

Set 3C configuration: `5669564f746563683200601600010392ed`

Reader responds with: `5669564f7465636832006000000003d35` (Success)

Now set terminal data (TLVs) with config values appropriate to 3C:

```
5669564f74656368320060060c818005f3601029f1a0208409f3501259f330360
d8c89f40056000f050019f1e085465726d696e616c9f150212349f160f30303030
30303030303030303030309f1c0838373635343332319f4e2231303732312057 61
6c6b65722053742e2043797072657373732c204341202c5553412edf260101df1008
656e667265737a68df110101df270100dfee150101dfee160100dfee170107dfee
180180dfee1e08f0dc24f0c20e1400dfee1f0180dfee1b083030303135313030df
ee20013cdfee21010adfee2203323c3caa88
```

Reader responds with: `5669564f7465636832006000000003d35` (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			

8.4.25. Contact LCD Display Control (61-01) (Reader sends to Host)

Some readers do not incorporate LCD and keypad, so this command is sent to host. The command requests display or key-in by host. If reader requests key-in, host response should be 61-01 command and result sent to reader. This command is used in 60-10, 60-11, and 60-12 commands. When in those commands, command 61-01 is sent to host automatically.

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

<table>
<tr><th>Data Item</th><th>Length (bytes)</th><th>Description</th></tr>
<tr><td>Display Mode</td><td>1</td><td>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 rest of 'Data Item', see below:
<table>
<tr><td>Data Item</td><td>Length (bytes)</td><td>Description</td></tr>
<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>
</table>
0x21: Update LED State (Do not receive input data)
For the rest of 'Data Item', see below:
<table>
<tr><td>Data Item</td><td>Length (bytes)</td><td>Description</td></tr>
<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>
</table>
0x22: Show Key Pad (Do not receive input data)
(If Clear Screen, don't need rod end below filed)</td></tr>
<tr><td>Timeout Length</td><td>2</td><td>Format: <Length L><Length H> (Little-endian)
If Display without key input. The value is 00 00.</td></tr>
<tr><td>Timeout Value</td><td>Variable</td><td>Format: <Value L><Value H> (Little-endian)
If Normal Display or Menu Display, Total timeout for keypad entry, in second. Default is 30 seconds.</td></tr>
</table>

Data Item	Length (bytes)	Description
		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. ▪ The most significant bit is to indicate following message is a Message String or Message ID. ▪ 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: ▪ "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 key pad 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", don't need to send below field.
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))

8.4.26. Contact Get PIN Control (61-02) (Reader or SCRP send to Host)

Some readers or SCRP do not incorporate a PIN Pad, so it's sometimes necessary to send this command to the host or PIN Pad Device or Phone. The command requests Get PIN on host or PIN Pad Device or Phone. If the reader or SCRP is requesting to "Get PIN", the host or PIN Pad Device or Phone responds with 61-02 command to reader or SCRP with PIN Block.

This command used in "60-11" command only in SRED. When in that command, command "61-02" will be sent to the host or PIN Pad Device automatically.

This command used in "60-11" command or "62-01" command in SCRP. When in these commands, command "61-02" will be sent to Phone automatically.

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	- 0x00 – Cancel (cancel through command) - 0x01 – Online PIN DUKPT - 0x02 – Online PIN MKSK - 0x03 – Offline PIN - 0x10 – SCRP Generic - 0x11 – SCRP 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, Device 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[SCRP PIN Conveyance Key]	2	This field is available for Enc [SCRP PIN Conveyance Key] Length when Mode is 0x10 Format: <Length L><Length H> (Little-endian)
Enc [SCRP PIN Conveyance Key]	16	This field is available for Enc [SCRP PIN Conveyance Key] Length when Mode is 0x10 [SCRP PIN Conveyance Key] is random generated by SCRP. Enc [SCRP PIN Conveyance Key] is encrypted [SCRP 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

Data Item	Length (bytes)	Description
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)
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 – SCRP Generic - 0x11 – SCRP MagicCube (If device 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: - if device does not implement Pairing function, Offline PIN is Plaintext. - If Mode is SCRP Generic/MagicCube

Data Item	Length (bytes)	Description
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: - if device does not implement Pairing function, Offline PIN is Plaintext. - If Mode is SCRP Generic/MagicCube
PIN Payload Length	2	Format: <Length L><Length H> (Little-endian)
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 device does not implement Pairing function, Offline PIN is Plaintext)

Note: Upon finishing this command, [SCRP PIN Conveyance Key] is erased.

8.4.27. Get Random Number (19-01)

Use this command to get random numbers from ViVOpay reader. The suggested length of random number is 256 bytes.

In SCRP (Secure Card Reader for PIN), this command could be used as a first step of refresh 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 (MSB)	CRC (LSB)
ViVOtech2\0	19h	See Status Code Table	00h	00h		

8.4.28. Refresh Enablement Token (19-02)

Use this command in SCRP (Secure Card Reader for PIN) environments only, to refresh the secure enablement token of ViVOpay reader.

The 19-01 command (above) shall be called previously to get 16 bytes random number first, then generate the signature by using the following algorithm:

1. Calculate 32 bytes Hash for 16 bytes random number;
2. Use RSAPSS algorithm to calculate the Hash to be 256 bytes Raw Data;
3. Use Private Key of Secure Message Certificate to sign the Raw Data to be 256 bytes signature

When SCRP reader receives this command, it may verify the signature with the previously saved 16 bytes random number, and the saved 16 bytes random number will be cleared whatever the result. After verification is successful, the enablement token is refreshed and SCRP is enabled to accept consumer cards for another 5 mins, no more than 10 mins (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 (MSB)	CRC (LSB)
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		

8.4.29. 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.

8.4.30. CVMApp Get Nonce from SCRP for Refresh Token (19-15)

Use this command to get random numbers from ViVOpay SCRP.
In SCRP, this command could be used as a first step of refresh 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	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]

8.4.31. Refresh Enablement Token for SCRP (19-16)

Use this command in SCRP only to refresh the secure enablement token of SCRP.

CVMApp previously (19-15 command) get 16 bytes random number from SCRP, then generate the signature by using the following algorithm

When SCRP receives this command, it may verify the signature with the previously saved 16 bytes random number, and the saved 16 bytes random number will be cleared whatever the result. After verification is successful, the enablement token is refreshed and SCRP is enabled to accept consumer cards for another 5 mins, no more than 10 mins (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 (MSB)	CRC (LSB)
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		

8.4.32. Contact Get MSR Data Control (Reader sends to Host)(61-03)

If reader is not MSR device, it can use this command to connect with 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.

Data Item	Length (bytes)	Description
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. ▪ The MSB is to indicate following message is a Message String or Message ID. ▪ 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: ▪ "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 ... 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: MSR Data for external MSR device max is 640 bytes. Tag DFEE23 data format will follow 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.

8.5. Transaction Related Commands: Contactless

8.5.1. Activate Transaction Command (02-01 and 02-40)

NOTE: 02-01 is a legacy command, applicable to non-encrypted transactions only. When EMV mode encryption is ON or MSR/MSD encryption is ON, if 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 if a key is present (unit is injected), encryption will occur.

Use the Activate Transaction command when the ViVOpay 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. When a valid Activate Transaction command is sent to the ViVOpay reader, it starts polling for cards.

If the ViVOpay reader does not find a supported card (an AID that matches one of the configured AIDs in the reader) for the specified time duration, it times out and ends the transaction. If the ViVOpay 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, only a Cancel or a Stop command may be sent. 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 version device, response format for Activate Transaction Command is according to "Set Data Encryption Enable Flag (C7-36)" setting and Data encryption Key. When Data Encryption is disabled, device responds with plaintext data format. When Data Encryption is enabled: (1) When Data encryption Key exists and is valid, device responds with encrypted data format. (2) When Data encryption Key does not exist, device responds with plaintext data format (3) When Data encryption Key exhausts, device responds status code 0x91 and no data.
For SRED version device, response format for Activate Transaction Command is according to Data encryption Key. When Data encryption Key exists and is valid, device responds with encrypted data format. When Data encryption Key exhausts or does not exist, 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 kick off a transaction in any desired mode, with or without fallback support. To control the behavior, use the DFEF37 and DFEF3C tags, as follows:

Tag	Length	Description	Example
DF EF 37	01	Define the type of interface to be activated with 02-40 Interface Select: • 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 • 0x00: not support fallback • 0x01: support fallback Byte 2~3: Timeout for next command (Unit: Sec) (Hex format) • 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:
`5669564f746563683200024000221e9c01009f020600000000001009f0306000000000000` **`dfef370107dfef3c03010060`** `18d1`

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	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. Note that if a card is found, the transaction may not complete within the timeout period. This field must be present in the Activate Command. Format: Binary
TLV Data	varies	See Activate Command TLVs below.

Activate Command TLVs

Tag	Description	Format	Length (in bytes)
9A	**Deprecated.** **NOTE: Do not attempt to use this tag in SRED devices. It may still work in non-SRED older products, but support for this tag in Activate Transaction will be dropped in future versions of firmware.** 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. Both date (9A) and time (9F21) tags must be present if either one is specified. The terminal/POS is responsible for ensuring that the date is valid: ▫ Year <=99 ▫ Month <=12 ▫ Day <=31 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.	n6 (YYMMDD)	3
9C	Transaction Type. Indicates the type of financial transaction, represented by the first two digits of ISO 8583:1993 Processing Code: ▫ 0x00 – Purchase Goods/Services ▫ 0x20 – Refund	n2	1
5F2A	Transaction Currency Code. Indicates the currency code of the transaction, in accordance with ISO 4217.	n3	2
5F36	Transaction Currency Exponent Indicates the implied position of the decimal point from the right of the transaction amount, according to ISO 4217.	n1	1
9F02	Amount, Authorized	n12	6
9F03	Amount Other	n12	6

Tag	Description	Format	Length (in bytes)
9F1A	Terminal Country Code. (Typically, this has been configured in the reader – refer to Group Configuration Tags.)	n	2
9F21	**Deprecated.** **NOTE: Do not attempt to use this tag in SRED devices. It may still work in non-SRED older products, but support for this tag in Activate Transaction will be dropped in future versions of firmware.** 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. Both date (9A) and time (9F21) tags must be present if either one is specified. The terminal/POS is responsible for ensuring that the time is valid.	n6 (HHMMSS)	3
9F5A	Terminal Transaction Type (Interac) ▫ 0x00 = Purchase ▫ 0x01 = Refund	b	1
DFEF7A	Determine Apple Pay or Google Pay transactions. If the value is set to 1, ater reading the card the output data will show tag DFEF7B to indicate whether the transaction is Apple Pay or Google Pay.		
FFEE01	ViVOtech TLV Group Tag	b	variable

For EMV transactions, if the terminal has already set up one or more of these data items using the Set Configuration or Set Configurable Group command, then the terminal need not include those data items in the **Command Frame**. If the terminal 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 ViVOpay 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" then 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 Track2 data are encapsulated in tags according to the MasterCard specification.

Activate Transaction Response TLVs

Tag	Description	Format	Length (in bytes)
DFEF17 (FFEE13)	Track 1 Data Contactless Transaction Transaction Result (response) Added for DiscoverZip, Visa MSD, Amex, PBOC	ans	Card
DFEF18 (FFEE14)	Track 2 Data Contactless Transaction Transaction Result (response) Added for DiscoverZip, Visa MSD, Amex, PBOC	ans	Card
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 *Track 1 Data* may be present in the file read using the READ RECORD command during a mag-stripe 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	Terminal Verification Results. Status of various functions from the terminal perspective.	b	5

Tag	Description	Format	Length (in bytes)
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
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 *Application PAN*.	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	Terminal Country Code. (Typically, this has been configured in the reader – refer to Group Configuration Tags.)	n	2

Tag	Description	Format	Length (in bytes)
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
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	Terminal Capabilities. Indicates the card data input, CVM, and security capabilities of the Terminal and Reader.	b	3
9F34	Cardholder Verification Method (CVM) Results. Indicates result of last CVM performed.	b	3
9F35	Terminal 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) Or Terminal Transaction Type (Interac)	an b	<=5 1
9F5B	Membership Scheme Number of Points (Amex).	an	<=29
	Issuer Script Result(UICS)	b	<=128

Tag	Description	Format	Length (in bytes)
9F5D	Available Offline Spending Amount (Balance).	b	6
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	Mag-Stripe Application Version Number (Reader). Version number assigned by the payment system for the specific mag-stripe 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.	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	Terminal 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

Tag	Description	Format	Length (in bytes)
DF8115	Error Indication. (MasterCard) Flags defining the error conditions from the transaction. Refer to the M/Chip PayPass specification.	b	6
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	b	1

<table>
<tr><th>Tag</th><th>Description</th><th>Format</th><th>Length (in bytes)</th></tr>
<tr><td></td><td>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</td><td></td><td></td></tr>
<tr><td>DFEF73</td><td>CUP Output Message Info (CUP)
Byte 1
<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>Terminal 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></td><td>b</td><td>1</td></tr>
<tr><td>DFEF7B</td><td>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.

If tag DFEF7A is not set, the output data will not output 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</td><td></td><td></td></tr>
<tr><td>FF8105</td><td>Data Record (MasterCard, container).
Contains the data from the transaction. Refer to the M/Chip PayPass specification.</td><td>b</td><td>varies</td></tr>
<tr><td>FF8106</td><td>Discretionary Data (MasterCard, container).
Contains the discretionary data from the transaction. Refer to the M/Chip PayPass specification.</td><td>b</td><td>varies</td></tr>
<tr><td>FFEE01</td><td>ViVOPay Group Tag. (container)
This three-byte Group Tag was created to contain ViVOpay proprietary Tags. See tags below.</td><td>b</td><td><=76</td></tr>
</table>

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	Terminal Verification Results: Indicates results of various transaction processes.	b 5	Reader
9A	Transaction Date	n6	Terminal
9B	Transaction Status Information	b2	Terminal
9C	Transaction Type ▫ 0x00 = Purchase Goods/Services ▫ 0x20 = refund	n2	Reader
5F2A	Transaction Currency Code	n3	Terminal
5F2D	Language Preference	an2-8 var	Card
5F34	Application Primary Account Number Sequence Number	n2	Card
9F02	Amount, Authorized (Numeric)	n12	Terminal
9F03	Amount, Other (Numeric, Visa only)	n12	Terminal
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	Terminal Country Code	n2	Terminal
9F21	Transaction Time	n6	Terminal
9F26	Application Cryptogram	b8	Card
9F27	Cryptogram Information Data	b1	Card
9F33	Terminal Capabilities	b3	Terminal
9F34	Cardholder Verification Method (CVM) Results	b3	Terminal
9F35	Terminal Type (Interac)	b1	Terminal
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	Terminal Transaction Information (Interac)	b3	Terminal
9F5A	Terminal Transaction Type (Interac)	b1	Terminal
9F66	Visa TTQ	b4	Reader
9F6C	Card Transaction Qualifiers (Visa only)	b2	Card

Tag	Data Element Name	Format	Origin
9F6E	Form Factor Indicator or PayPass Third Party Data	b4 b5-32 var	Reader Reader
9F74	VLP Issuer Authentication Code	b	Terminal
9F7C	Customer Exclusive Data	b1-32 var	Reader
DF30	Track Data Source ▫ 0x0C – swiped mag-stripe card ▫ 0x00 – contactless transaction	b1	Reader
DF33	Receipt Required (Interac) – calculated by checking the Interac Terminal Receipt Required Limit (9F5F).	b1	Reader
DF52	Transaction CVM: ▫ 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	Terminal

If the Status Code being returned in the **Response Frame** is "Failed" and the Error Code is not "Request Online Authorization", then the contents of 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 being returned in the **Response Frame** is "Failed" and the Error Code is "Request Online Authorization", then the contents of 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

Data Field	Length (bytes)	Description
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" then the format and contents of the data field in the **Response Frame** are given in the following table:

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 reponse 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)

8.5.2. Special TLV for Discover D-PAS and SmartTap

For D-PAS, a new ViVOtech proprietary tag "TLV Special Flow" is added to the Activate Transaction command data as shown below.

Command Data

<table>
<tr><th>Data Item</th><th>Length (Bytes)</th><th>Description</th></tr>
<tr><td>ViVOtech TLV Group Tag FFEE01</td><td>Variable</td><td>
<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>
</td></tr>
</table>

Special Flow Record Format

<table>
<tr><th>Byte #</th><th>Field</th></tr>
<tr><td>1</td><td>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 not use ViVOtech2 "Set Configurable AID" such as SmartTap.</td></tr>
<tr><td>2</td><td>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></td></tr>
<tr><td>3</td><td>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></td></tr>
<tr><td>4</td><td>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></td></tr>
</table>

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 be 1
2. Use "Activate Transaction Command (02-01)" to do normal Discover D-PAS transaction and reader will response transaction result and data.
3. After host receives Online Request from reader, host can send "Activate Transaction Command (02-01)" to reader with Issuer Scripts(DF51 TLV) in FFEE01 TLV. Then reader will run Issuer Update Processing and response 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 ViVOpay 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

(1) MSG Signals

MSG Signals are used by other processes to send the User Interface Request Data to Process D. Process D manages the User Interface Requests as defined in reference [3] and displays a message and/or a status. The User Interface Request Data is defined in reference [1] as tag DF8116 and holds twenty-two bytes of data as shown in the table below:

Data Field	Length
Message Identifier	1
Status	1
Hold Time	3
Language Preference	8
Value Qualifier	1
Value	6
Currency Code	2

(2) OUT Signals

OUT Signals are used by the kernel (Process K) to indicate the outcome of a transaction. According to reference [1] the OUT signal may comprise the following objects.

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 to be 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
Terminal Capabilities	9F33	3
Terminal Country Code	9F1A	2
Terminal Type	9F35	1
Terminal 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 Mag-Stripe 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
Mag-stripe 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 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
Terminal Capabilities	9F33	3
Terminal Country Code	9F1A	2
Terminal Type	9F35	1
Terminal 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 Mag-Stripe 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

(3) Signal Data TLV – FFEE04

A proprietary TLV is defined to hold the intermediate signal data during a transaction. It will be populated with one or more signal objects as the UI_MSG_Signal function in UserInterface.c receives MSG Signals and the UI_OUT_Signal function in UserInterface.c receives OUT Signals.

This tag must be included in the ACT command to enable the signal data capture feature during the transaction. If it is received in the ACT command this feature is enabled and the Signal Data Handler will add signal data to the buffer. If it is not received, the Signal Data Handler will do nothing and the length of this TLV will remain 0 and nothing will be returned.

A second proprietary tag is defined (FFEE05) which is used to separate and identify each individual signal entry added to the buffer, whether it is a MSG signal or an OUT signal. A new tag (DF8914) which includes Activate Response TLVs (Table 29) may be included in FFEE05.

When the transaction is complete, if tag FFEE04 was received in the ACT command and the Signal Data tag is not empty, it will be added to the ACT response. The signal data buffer (the "contents" of FFEE04) is cleared when an ACT command is received.

8.5.3. Activate Transaction Response Frame Data Format

For additional information on how data portions are encrypted, see "80000502-001, *ID Tech Encrypted Data Output Formats*" for details; available for download on the ID TECH Knowledge Base.

Mode/Transaction Output Format Matrix—Non-SRED Device Contactless transaction

Note: When MSR/MSD or EMV Encryption is enabled and Data encryption Key exists, Burst Mode is disabled (always OFF)

When encryption is enabled and Data encryption Key exists, **Encrypt Mode** is ON. Otherwise (encryption is disabled or Data encryption Key is absent), **Encrypt Mode** is OFF.

<table>
<tr><th>Poll Mode</th><th>Encrypt Mode</th><th>Burst Mode</th><th>Command</th><th>Output Format</th></tr>
<tr><td rowspan="4">Poll On Demond</td><td rowspan="2">OFF</td><td>NA</td><td>02-01</td><td>02-01 plaintext</td></tr>
<tr><td>NA</td><td>02-40</td><td>02-40 plaintext</td></tr>
<tr><td rowspan="2">ON</td><td rowspan="2">NA</td><td>02-01</td><td>Not Allow</td></tr>
<tr><td>02-40</td><td>02-40 encrypted</td></tr>
<tr><td rowspan="14">Auto Poll</td><td rowspan="10">OFF</td><td rowspan="3">ON</td><td>No Cmd</td><td>Burst Format</td></tr>
<tr><td>02-01</td><td>Burst Format</td></tr>
<tr><td>02-40</td><td>Burst Format</td></tr>
<tr><td rowspan="3">AutoExit</td><td>No Cmd</td><td>Burst Format</td></tr>
<tr><td>02-01</td><td>02-01 plaintext</td></tr>
<tr><td>02-40</td><td>02-40 plaintext</td></tr>
<tr><td rowspan="4">OFF</td><td>03-00</td><td>02-01 plaintext</td></tr>
<tr><td>02-01</td><td>02-01 plaintext</td></tr>
<tr><td>03-40</td><td>02-40 plaintext</td></tr>
<tr><td>02-40</td><td>02-40 plaintext</td></tr>
<tr><td rowspan="4">ON</td><td rowspan="4">OFF</td><td>03-00</td><td>Not Allow</td></tr>
<tr><td>02-01</td><td>Not Allow</td></tr>
<tr><td>03-40</td><td>02-40 encrypted</td></tr>
<tr><td>02-40</td><td>02-40 encrypted</td></tr>
</table>

Mode/Transaction Output Format Matrix—SRED Device Contactless transaction

Note: For SRED device, encryption is always enabled and Burst Mode is disabled (always OFF)

Poll Mode	Encrypt Mode	Burst Mode	Command	Output Format
Poll On Demond	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 If encrypt disable or MSR data, this tag is not present.
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: FFEE13 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: FFEE14 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)

<table>
<tr><th>Data Item</th><th>Length (bytes)</th><th>Description</th></tr>
<tr><td>Attribution</td><td>1</td><td>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</td></tr>
<tr><td>MSR TLV</td><td>Variable</td><td>MSR TLV
Data length compose of data length indicator (1 byte) and actual data length byte.
<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>
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</td></tr>
<tr><td>TLV Data</td><td>Variable</td><td>See Activate Response TLVs</td></tr>
</table>

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 If encrypt disable or MSR data, this tag is not present.
TLV Contactless Error Code Table (FFEE1F)	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.

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 If encrypt disable or MSR data, this tag is not present.
TLV Contactless Error Code Table (FFEE1F)	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.

8.5.4. Get Transaction Result (03-00 and 03-40)

NOTE: 03-00 is a legacy command, applicable to non-encrypted transactions only.When EMV encryption ON or MSR/MSD encryption ON, if Data encryption Key is loaded, 03-00 will be disabled. 03-40 is a unified command for both non-encrypted and encrypted transaction.

Use this command when the ViVOpay 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 it detects a card, it carries out a transaction with the card. If the card is a supported contactless MagStripe card, the reader does not need any parameters from the terminal. If the card is a supported contactless EMV Card, then the reader uses the default terminal parameters (Group 0 TLVs) in the reader. If some terminal parameters had been set by using the Set Configuration command, then the reader 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 terminal 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 terminal.

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 since these transactions require interaction with the Terminal during the transaction itself.

Note: For Non-SRED version device, response format for Activate Transaction Command is according to "Set Data Encryption Enable Flag (C7-36)" setting and Data encryption Key. When Data Encryption is disabled, device responds with plaintext data format. When Data Encryption is enabled: (1) When Data encryption Key exists and is valid, device responds with encrypted data format. (2) When Data encryption Key does not exist, device responds with plaintext data format (3) When Data encryption Key exhausts, device responds status code 0x91 and no data.
For SRED version device, response format for Activate Transaction Command is according to Data encryption Key. When Data encryption Key exists and is valid, device responds with encrypted data format. When Data encryption Key exhausts or does not exist, device responds 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 ViVOpay 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 indicates "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 Status Code is OK, then 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.

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)
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/Trakc2 and VLP Issuer Auth code
VSDC offline and qVSDC	Chip data E1 and some other tags

A Status Code 23 (request online authorization) can be returned for some cards (qVSDC & M/Chip) with a fully populated data field.

This command never returns a status code of "Failed". If any status code other than OK or status code '23' (request online authorization) is returned, the data field is empty.

The above description is 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	If encryption is not enabled, this tag is not present. 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: FFEE13 Format: ASCII (no null terminator) For MasterCard transactions, this field is not present. If track data is present, it is contained in the MasterCard TLVs.

Data Item	Length (bytes)	Description
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: FFEE14 Format: ASCII (no null terminator) For MasterCard transactions, this field is not present. If track data is present, it is contained in the MasterCard TLVs.
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

<table>
<tr><th>Data Item</th><th>Length (bytes)</th><th>Description</th></tr>
<tr><td>TLV MSR Data (DFEE23)</td><td>Variable</td><td>Refer to "Enhanced Encrypted MSR Data Output Format" section of document 80000502-001, Encrypted Data Output Formats. 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></td></tr>
<tr><td>TLV Data</td><td>Variable</td><td>TLV data list</td></tr>
<tr><td>TLV POS Entry Mode (9F39)</td><td>Variable</td><td>90h = Magnetic Stripe Reader Swipe
91h = Contactless MSD
05h = Contact EMV
07h = Contactless EMV
80h = Contact Fallback to Magnetic Stripe</td></tr>
<tr><td>TLV (DF30) (ViVOpay proprietary)</td><td>Variable</td><td>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</td></tr>
<tr><td>TLV Encrypt Information (DFEE26)</td><td>Variable</td><td>Encryption and Data Capture Information.
Length: 1 or 2 (variable)
NEOII the length must be 2 bytes
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.</td></tr>
</table>

Data Item	Length (bytes)	Description
		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 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

8.5.5. Update Balance Command (03-03)

Use this command when the ViVOpay reader has been put in "Poll on Demand" mode and after the reader sends an online request to the issuer. This command is the authorization response sent by the issuer to the terminal including the Authorization Status (OK or NOT OK).

This command is also being 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 terminal.

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	**Deprecated.** 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	**Deprecated.** 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 then 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", then the contents of 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.

For any other Status Code the data field is empty.

8.5.6. Cancel Transaction Command/Reset Transaction Status (05-01)

Use this command to stop reader/card communication after the Activate Transaction command or Update Balance command has been sent to the reader.

After the terminal has issued the Cancel Transaction command, the terminal 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 cancel transaction, data length (LSB) is 00h, data object is not required.
For reset transaction, data length (LSB) is 01h, 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		

8.6. MasterCard M/Chip 3.0 Transaction Commands

This section describes commands that are specific to MasterCard M/Chip 3.0 transaction behavior.

8.6.1. Stop Transaction (05-02)

The Stop Transaction command is similar to the Cancel command. However, the transaction will exit at whatever phase it was currently in. Depending on timing, the transaction could exit with an Activate Response. In that case, the Stop command was received too late to stop the transaction. Receipt of any response other than the Stop response is proof that the Stop command did not execute.

The Stop command is currently only used by the MasterCard M/Chip 3.0 application.

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		

The following information is returned in the data field of a successful Stop command:

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	

8.6.2. Reset Torn Transaction Log (84-0E)

The Reset Torn Transaction Log effectively erases the content of the torn transaction log and sets it back to an "empty" state.

Normally, this function will only be 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)
ViVOtech2\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 (LSB)	CRC (MSB)
ViVOtech2\0	84h	Refer to standard status values	00h	00h	Varies	Varies

This command, when sent, will restore the Torn Transaction Log back to its original pristine state, as if a power up had just occurred.

8.6.3. Clean Torn Transaction Log (84-0F) Command

This command is used to remove Torn Transaction Log entries that have exceeded 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)
ViVOtech2\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 (LSB)	CRC (MSB)
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 terminal should execute the CLEAN command repeatedly, until no more torn records are sent back to it. 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.

8.6.3.1. Torn Transaction Log Timer

The reader keeps 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 terminal/POS and the Reader is a command/response interface, cleaning the torn transaction log must be initiated by the terminal/POS.

Periodically, the POS should initiate a cleaning cycle and repeatedly issue the "Clean" command (84-0F) at that point until the reader reports that the Torn Log has been successfully purged.

How the POS accomplishes this is beyond the scope of the interface and this document.

8.6.4. Data Exchange Request (02-58) Command and Data Exchange Response (02-09)

These commands are used to do data exchange for MasterCard (MCL).

Data Exchange Request command (02-58) is sent by device to the host, while Data Exchange Response command (02-09) is sent by the host to the device. This process is different from other commands. It applies in certain MasterCard scenarios only.

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 (LSB)	CRC (MSB)
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
```

8.7. 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)

8.7.1. Get CA Public Key (D0-01)

This command retrieves all of the 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 (LSB)	CRC (MSB)
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)

8.7.2. Get CA Public Key Hash (D0-02)

This 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 (LSB)	CRC (MSB)
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

8.7.3. Set CA Public Key (D0-03)

This command adds a new key in 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 (LSB)	CRC (MSB)
ViVOtech2\0	D0h	See Key Manager status codes	00h	00h	Calculated	Calculated

8.7.4. Delete CA Public Key (D0-04)

This 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 (LSB)	CRC (MSB)
ViVOtech2\0	D0h	See Key Manager status codes	00h	00h	Calculated	Calculated

8.7.5. Delete All CA Public Keys (D0-05)

This command deletes all of the CA public keys.

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 (LSB)	CRC (MSB)
ViVOtech2\0	D0h	See Key Manager status codes	00h	00h	Calculated	Calculated

8.7.6. 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 (LSB)	CRC (MSB)
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.

8.7.7. List CA Public Key IDs or RID (D0-07)

The following 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 (LSB)	CRC (MSB)
ViVOtech2\0	D0h	See Key Manager status codes	00h	varies	varies	varies	Varies	Varies

Status – 00h = successful index retrieval

8.8. 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[3]

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 that is 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 not identified by the "application ID" (Example: SmartTap 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.

[3] 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.

8.8.1. Get Product Type (09-01)

This 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
42 37 00	ViVOpay 5000
43 33 00	ViVOpay 4500
43 35 00	ViVOpay Vend
43 36 00	Vendi (NEO 1)
43 37 00	ViVOpay Kiosk1 (ATM1)
43 38 00	Kiosk2
43 39 00	Kiosk3 (NEO)
55 31 00	UniPay 1.5 (NEO)

55 33 00	UniPay III (NEO)
55 33 31	VP3300, VP3300 OEM (NEO) (iBase/Cake same code)
55 33 32	VP3300E(NEO)
55 33 33	VP3300C(NEO)
55 33 34	BTPay Mini (NEO) (UniPayIII + BLE)
56 41 00	VP3320
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
56 36 00	VP8300
56 37 00	VP8310
56 38 00	VP8800
56 39 00	VP8810
56 40 00	VP9000
44 30 00	QX120
44 31 00	Mx8Series
44 32 00	NETs
44 33 00	Magtek
44 35 00	ICP

8.8.2. Get Processor Type (09-02)

This 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: Get Processor Type: `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`

8.8.3. Get Main Firmware Version (09-03)

This 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`

8.8.4. Get Hardware Information (09-14)

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).

Command: Get Hardware Version Information: `56 69 56 4F 74 65 63 68 32 00 09 14 00 00 33 08`

Response: `56 69 56 4F 74 65 63 68 32 00 09 00 00 15 48 57 2C 56 50 56 65 6E 64 69 0D 0A 4B 32 31 46 20 52 65 76 39`

ASCII	Description
HW,VP3300 Audio Jack<CR><LF>K21F Rev9	Unipay III
HW,VPUnipay1.5<CR><LF>K21F Rev9	Unipay 1.5
HW,VPUniPay1.5TTK<CR><LF>K21F Rev9	UniPay 1.5 TTK
HW,VP3300 USB<CR><LF>K21F Rev9	VP3300 USB, VP3300 USB OEM (iBase/Cake same code)
HW,VP3300 USB-E<CR><LF>K21F Rev9	VP3300 USB-E
HW,VP3300 USB-C<CR><LF>K21F Rev9	VP3300 USB-C
HW,VPVP3300 Bluetooth<CR><LF>K21F Rev9	VP3300 Bluetooth
HW,.VP6300<CR><LF>K81F.Rev4	VP6300

8.8.5. Get Module Version Information (09-20)

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	20h	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 appropriate Status Code will be returned with an empty Data field (Data Length = 0000h).

The format for 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 and spec.
version>,[<implementation version>],<CRLF>
```

The following example shows the module version information subcommand and the information being 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 69 56 4F 74 65 63 68 32 00 09 00 01 2A 46 57 2C 56 65 6E 64 69
20 56 31 2E 30 30 2C 2C 0D 0A 20 46 53 26 44 42 20 56 31 2E 30 30
2C 2C 0D 0A 20 43 4C 20 41 70 70 53 65 6C 2C 50 50 53 45 20 4D 6F
64 75 6C 65 2C 20 76 31 2E 30 30 2C 2C 0D 0A 20 43 4C 20 41 49 44
2C 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 76 33 2E 30 2E 32 2C 20 56 65 6E 64 69 20 76 31 2E
30 2E 30 2C 2C 0D 0A 20 43 4C 20 41 49 44 2C 56 69 73 61 20 56 43
50 53 20 32 2E 31 2E 33 2C 20 76 30 2E 39 39 2C 2C 0D 0A 20 43 4C
20 41 49 44 2C 41 6D 65 78 20 45 78 70 72 65 73 73 50 61 79 20 33
2E 30 2C 20 76 31 2E 30 30 2C 2C 0D 0A 20 43 4C 20 41 49 44 2C 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 30 30 2C 2C 0D 0A 20 43 4C 20 41 49 44
2C 49 6E 74 65 72 61 63 20 31 2E 35 2C 20 76 31 2E 30 30 2C 2C 0D
0A 20 43 4C 20 4C 31 2C 45 4D 56 20 34 2E 33 20 4C 31 2C 20 76 31
2E 30 30 00 8C 33
```

ASCII translation of the data field:

```
FW,Vendi V1.00,,<CR><LF>
 FS&DB V1.00,,<CR><LF>
 CL AppSel,PPSE Module, v1.00,,<CR><LF>
 CL AID,MasterCard PayPass M/Chip v3.0.2, Vendi v1.0.0,,<CR><LF>
 CL AID,Visa VCPS 2.1.3, v0.99,,<CR><LF>
 CL AID,Amex ExpressPay 3.0, v1.00,,<CR><LF>
 CL AID,Discover DPAS 1.0 Zip 3.1.2, v1.00,,<CR><LF>
 CL AID,Interac 1.5, v1.00,,<CR><LF>
 CL L1,EMV 4.3 L1, v1.00<NUL>
```

8.9. International Language Support

The goal of this feature is to offer support for foreign languages, including those based on graphical fonts (like Chinese, Thai, Arabic, and others). This applies only to devices that have graphical output capability.

Four core language options are built in. 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 that have been 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 Appendix A.7 Preparing Bitmaps for Use by ILM for instructions on working with bitmaps.

Each reader keeps four "core" languages in its firmware. This ensures that 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. They always exist and are always available for use. At present, there are four (4) Core Languages available:

- 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.

8.9.1. Other Languages

In addition to the Core Languages, a fifth language called "Other Language" is available. This 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 for this language. The Other Language only needs a unique record for each distinct message it must display.

At present there are twenty-two (22) predefined message types that can be displayed.

Note: Each time you configure the reader for ILM, you must download messages for all 22 indices in consecutive order. You cannot change individual messages.

8.9.2. 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.

8.9.3. 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.

8.9.4. 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. This is done because there are a number of languages (English, French, Spanish, and others) shared by multiple nations but the reader is intended to be operate in a particular 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 02 03	Alpha2 (2 character) – "ES\0" Alpha3 (3 character) – "ESP\0" 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 (as is usually the case) the ASCII string does not occupy the entire data field, 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 the Language Version Information is used by the customer 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.

8.10. 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.

8.10.1. Get EMV Revocation Log Status (84-03)

This command returns information about the EMV revocation log. The information returned can be used by the terminal/POS 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 (LSB)	CRC (MSB)
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

8.10.2. Add Entry to EMV Revocation List (84-04)

This command adds a new entry to the revocation list. The new entry is added at the end of 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 (LSB)	CRC (MSB)
ViVOtech2\0	84h	See Status Code Table	00	00		

8.10.3. Delete All Entries for Single Index in EMV Revocation List (84-05)

This 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 (LSB)	CRC (MSB)
ViVOtech2\0	84h	See Status Code Table	00	00		

8.10.4. Delete All Entries from EMV Revocation List (84-06)

This 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 (LSB)	CRC (MSB)
ViVOtech2\0	84h	See Status Code Table	00	00		

8.10.5. Get EMV Revocation List (84-07)

This 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[4] (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 (LSB)	CRC (MSB)
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.

[4] For GR, this number must be less than 30. The first record is 0.

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

8.10.6. Delete an Entry from EMV Revocation List (84-0D)

This 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 ... 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	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 (LSB)	CRC (MSB)
ViVOtech2\0	84h	See Status Code Table	00	00		

8.11. EMV Exception Log List Commands

8.11.1. Get EMV Exception Log Status (84-08)

This command returns information about the EMV exception log. The version number, record size, and number of records contained in the file are returned.

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 (LSB)	CRC (MSB)
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

8.11.2. Add Entry to EMV Exception List (84-09)

This 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):
`084761739001010010FFFF010857627391010100 10FFFF000811223344556677 8F FFFF050A112233445566778899 0F00`

The above file has four entries:

- `08 4761739001010010FFFF 01` where PAN = `4761739001010010`
- `08 5762739101010010FFFF 00` where PAN = `5762739101010010`
- `08 112233445566778FFFFF 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 (LSB)	CRC (MSB)
ViVOtech2\0	84h	See Status Codes Table	00h	00h		

8.11.3. Delete Entry from EMV Exception List (84-0A)

This 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

Example Exception List File Data (in a non-null terminated ASCII text file):
`084761739001010010FFFF010857627391010100 10FFFF00081122334455667 78FFFFF050A11223344556677889 90F00`

The above file has four entries:

- `08 4761739001010010FFFF 01` where PAN = `4761739001010010`
- `08 5762739101010010FFFF 00` where PAN = `5762739101010010`
- `08 112233445566778FFFFF 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 (LSB)	CRC (MSB)
ViVOtech2\0	84h	See Status Codes Table	00h	00h		

8.11.4. Delete All Entries from EMV Exception List (84-0B)

This 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 (LSB)	CRC (MSB)
ViVOtech2\0	84h	See Status Code Table	00h	00h		

8.11.5. Get EMV Exception List (84-0C)

This 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 byte 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 (LSB)	CRC (MSB)
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**[5]	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

8.12. 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 firmware.

Note: Pass-Through commands can only be executed in Pass-Through Mode. Other commands (that is, non-Pass-Through commands) return an error in Pass-Through Mode with the exception of Ping, Get Version, and Get Serial Number commands, which will work in both modes.

If the reader is not in Pass-Through Mode, a "Pass-Through Mode Start" command must first be issued. Otherwise, the commands in this section will result in an error unless otherwise specified.

The following commands always work regardless whether or not the reader is in Generic passthrough mode:

- 0x1201
- 0x1801
- 0x2801
- 0x2900
- 0x2C01

[5] 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.

- 0x6014

The following commands only work in Generic passthrough mode. To use them, first run 2C-01: Pass-Through Mode Start/Stop.

- 0x0A02
- 0x0A03
- 0x0B01
- 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

8.12.1. Pass-Through Mode Start/Stop (2C-01)

The Pass-Through Mode Start/Stop command is used to enter and exit Pass-Through Mode. The 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.

8.12.2. Get PCD and PICC Parameters (2C-05)

This command allows the terminal 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 (MSB)	CRC (LSB)
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 terminal, the ViVOpay 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.

8.12.3. Poll for Token (2C-02)

After Pass-Through Mode is started, ViVOpay will not poll for any cards until the "Poll for Token" command is received. This command tells ViVOpay 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 a PICC is detected within the specified time limit, ViVOpay activates it and responds back to the terminal with card related data such as the Serial Number.

If no PICC is detected within the specified time limit, ViVOpay stops polling and responds back indicating that no card was found. No card related data is returned in this case.

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	02h	00h	02h	See Below		

Poll for Token Data Field for Command Frame

<table>
<tr><th>Data Field</th><th>Length (bytes)</th><th>Description</th></tr>
<tr><td>Timeout1</td><td>1</td><td>Time in Seconds Timeout1 cannot be zero seconds if Timeout2 is Zero.</td></tr>
<tr><td>Timeout2</td><td>1</td><td>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></td></tr>
</table>

Together Timeout1 and Timeout2 are used by the ViVOpay reader 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 Apple VAS 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.

8.12.4. Enhanced Poll for Token (2C-0C)

After Pass-Through Mode is started, ViVOpay waits until the **Poll for Token** command is received. This command tells ViVOpay 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 a PICC is detected within the speified time limit, ViVOpay activates it and responds back to the terminal with card related data, such as the Card Type and Serial Number (UID).
If no PICC is detected within the specified time limit, ViVOpay stops polling and responds back indicating that no card was found. No card related data is returned in this case.

8.12.4.1. 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.

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 (MSB)	CRC (LSB)
ViVOtech2\0	2Ch	0Ch	00h	04h	See Table Below		

Enhanced Poll for Token Data Field for Command Frame

<table>
<tr><th>Data Field</th><th>Length (bytes)</th><th>Description</th></tr>
<tr><td>Timeout1</td><td>1</td><td>Time in Seconds Timeout1 cannot be zero seconds if Timeout2 is Zero.</td></tr>
<tr><td>Timeout2</td><td>1</td><td>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></td></tr>
<tr><td>Transaction Type</td><td>2</td><td>Initiate a transaction based upon the following masks (more than 1 can be active):
<table>
<tr><td>Mask (hex)</td><td>Action</td></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></td></tr>
</table>

Together Timeout1 & Timeout2 are used by the ViVOpay reader 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)
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.

8.12.4.2. Enhanced Poll for Token Usage

This command can be substituted for the standard **Poll for Token** in any transaction taking place in Pass-Through Mode. Its differences in operation are noted as above; its other arguments should be identical to those of the standard **Poll for Token** that it replaces.

Note: If you use an **Enhanced Poll for Token** command but have set all values in the Transaction Type field to zero, then the command performs a standard **Poll for Token** instead.

8.12.5. Get ATR (2C-12)

This Pass-Through command can be used to get the ATR received by the reader from the ICC and SAM when a Level 1 session was 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		

8.12.6. Terminate ISO Session (2C-16)

Traditionally, Pass-Through sessions have been terminated by dropping the RF field. However, the RCTIF test specification requires a graceful termination of a session, using either a DISC (Bprime) or a DESELECT (A and B).

The TerminateISOSession command send the appropriate command to disconnect the session. It will wait for the requisite timeout period, reset the field and then return a status.

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	2Ch	16h	00h	00h		

The **Response Frame** is a simple confirmation that the session has been terminated. The field is left in the desired state.

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	varies	00h	00h		

The status returned indicates:
0 = any existing session was terminated.
1 = no session was active (however the field will always be reset)

8.12.7. Antenna Control (28-01)

This 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 (MSB)	CRC (LSB)
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 on use of 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 ViVOpay 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.

8.13. Pass-through UI Control

8.13.1. LED Control (0A-02)

This command switches the specified ViVOpay LEDs off or on *only* when ViVOpay 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 (MSB)	CRC (LSB)
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)

Note: the setting of Tri-color LED(02h~04h) is valid for VP3300 Audio Jack, VP3300 USB, and VP3300BT.

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		

8.14. Front LED Control (0A-03)

This command controls the front LED of VP5300.

Note: This command can only be performed while in Passthrough 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		

8.14.1. Buzzer Control (0B-xx)

This command can be used to sound the ViVOpay 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 (MSB)	CRC (LSB)
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		

8.15. Pass-through Data Exchange

8.15.1. Exchange Contactless Data (2C-03)

This command allows the terminal to send, via ViVOpay, application-level APDUs to a PICC that supports ISO 14443-4 Protocol. The PICC response is sent back by ViVOpay to the Terminal.

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	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 terminal, ViVOpay sends the APDU data to the PICC and receives its response. ViVOpay treats the PICC response as unknown data and does not try to interpret it. If the operation was successful, ViVOpay 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 occurred during communication with the PICC, then ViVOpay sends a **Response Frame** with an appropriate Status. No Data is returned in this case.

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 that was 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 device, the APDU data being received from the card/device by the reader will be 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 White List, the Command will return a Parameter Not Supported error (0x06).

8.15.2. PCD Single Command Exchange (2C-04) Protocol 2

This command allows the terminal 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 PICC response is sent back by the reader to the terminal.

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	2Ch	04h	Variable	Variable	See Table below		

PCD Single Command Exchange Data Field Protocol 2

Data Field	Length (bytes)	Description
PCD Command	1	This is the command that is 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	This is 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 ViVOpay 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 ViVOpay to determine what PCD function is to be carried 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 terminal 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 definition of bit 5 is only to keep compatible with Kiosk II, it doesn't work, but this bit is occupied, and is not allowed for other usage.

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.
3	Rx CRC Enable	1	Last bytes of received data are interpreted as CRC bytes. **Note**: The CRC is not sent back to the terminal 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 a valid **Command Frame** is received from the terminal, ViVOpay sends the data to the PICC (or carries out the appropriate action) and receives the PICC response. The ViVOpay 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), then the reader sends a **Response Frame** with an appropriate Status. No Data is returned in this case.

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

<table>
<tr><th>Data Field</th><th>Length (bytes)</th><th>Description</th></tr>
<tr><td>PCD Status</td><td>1</td><td>This field contains the result of the PCD Command. Possible values are given in the following table.
<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>
<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>
</table>
</td></tr>
</table>

		-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 device, the Raw Data being received from the card/device by the reader will be 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 White List, the Command will return a Parameter Not Supported error (0x06).

8.15.2.1. Example: Sending a HALT Command to a Type A PICC

Assuming that ViVOpay has been put into Pass-Through Mode, a Type A PICC has been detected using the Poll for Token command, and the terminal application has completed the transaction with the card, an ISO 14443 HALT command can be sent to the PICC using the PCD Single Command Exchange command. Given below is a log of the command and data that the terminal would send to ViVOpay and also the responses that may be received from ViVOpay.

The following serial data may be exchanged between a terminal/PC and a ViVOpay reader:

Halt Command Exchange Between Terminal/PC 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>

8.15.3. High-level Halt Command (2C-09)

This command instructs the ViVOpay reader to send a HALT command to the card and can be used for 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 (MSB)	CRC (LSB)
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		

8.15.4. Enhanced Pass-Through Command (2C-0B)

This command instructs the reader to carry out several tasks while in Pass-Through Mode. This command is ONLY enabled in Pass-Through Mode. If the reader is not in Pass-Through Mode, the reader ignores this command.

Note: SAM interface is only supported in SRED devices. It is not supported in non-SRED version.

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:

- 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

If this command is only used to set the Buzzer/LED, it works on the all ID TECH readers.

There are three cases depending 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 terminal.
- Index FFh indicates the terminal 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 format and contents of the data field in the **Command Frame** are given in the following table.

Enhanced Pass-Through Data Field

Data Item	Length (bytes)	Description
Single-Shot Commands	1	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. 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 For more information see Single-Shot Commands below.
LCD Message Index (for readers with a display only)	1	**NOTE:** For the complete list of possible messages, see Appendix A-10. 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) 08-0B is controlled by the terminal through this command

<table>
<tr><th>Data Item</th><th>Length (bytes)</th><th>Description</th></tr>
<tr><td></td><td></td><td>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)

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.

FF indicates not to set the LCD message which allows terminal to set LED/Buzzer only.

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")</td></tr>
<tr><td>Beep Indicator</td><td>1</td><td>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</td></tr>
<tr><td>LED Number</td><td>1</td><td>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.</td></tr>
<tr><td>LED Status</td><td>1</td><td>00h: LED Off
01h: LED On</td></tr>
<tr><td>Timeout1</td><td>1</td><td>Time in Seconds. Timeout1 cannot be zero seconds if Timeout2 is Zero.</td></tr>
<tr><td>Timeout2</td><td>1</td><td>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></td></tr>
<tr><td>LCD String1 Message</td><td>X</td><td>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</td></tr>
</table>

Data Item	Length (bytes)	Description
		and LCD String2 Message are included, then the reserved field must be included, with LCD String1 Message appearing immediately after it. **Note:** The string must be null terminated (00) to indicate the end of the string.
LCD String2 Message	Y	The field is Y bytes long and consists of a simple character string. It contains NO formatting information, ONLY text characters. **Note:** The string must be null terminated (00) to indicate the end of the string.
Selected Interface	Z	This field is present when the Single-Shot Commands Byte, Specified Interface bit = 1: It is 1 byte long. Allowed Interface Values are: 00h = Contactless 20h = RFU 21h = SAM1 (SRED version only) 22h = SAM2 (SRED version only) **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.

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		

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 Transaction, the response is as follows: If Poll for Token bit is disabled, no data is returned in the response. It is the same for the beep indicator. If Poll for Token bit is enabled, the response is the same as Poll for Token (2C-02) command..

8.15.5. 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 indicated discrete operation is carried out. If the bit is cleared, the associated command is NOT carried out.

If the Poll for Token bit is enabled, the reader checks the value of the Use Specified Interface bit.

If Use Specified Interface bit = 0, reader polls for a PICC card.

If Use Specified Interface bit = 1, reader attempts a Poll for Token operation on the interface specified in the Selected Interface byte.

8.15.5.1. 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

8.15.5.2. 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.

8.15.5.3. 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

8.15.5.4. 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).

8.15.5.5. 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).

8.15.5.6. 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

8.15.5.7. 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)

8.15.1. Poll for Token with ATS (2C-0E)

After Pass-Through Mode is started, the device will not poll for any cards until it receives the **Poll for Token** command. This command tells the device 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 device detects a PICC is detected within the specified time limit, it activates the PICC and returns card related data such as the Serial Number to the terminal.

If the device does not detect a PICC within the specified time limit, it stops polling and responds to the terminal that no card was found. No card related data is returned in this case.

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	0Eh	00h	02h	See Below		

Data Fields for the Command Frame

Table 4: Poll for Token Data Field for Command Frame

<table>
<tr><th>Data Field</th><th>Length (bytes)</th><th>Description</th></tr>
<tr><td>Timeout1</td><td>1</td><td>Time in Seconds Timeout1 cannot be zero seconds if Timeout2 is Zero.</td></tr>
<tr><td>Timeout2</td><td>1</td><td>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></td></tr>
</table>

The device uses Timeout1 and Timeout2 together to calculate the Timeout (that is, the time to wait for a PICC).

Example:

Table 5: 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.

Table 6:
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) 08h Felica 09h RCTIF Type B Prim
Serial Number Len	1	Serial Number Length
Serial Number	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.
ATS Len	1	ATS data length
ATS Data	Variable	ATS data

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 will soon become available. All of these card types are handled by this command.

8.15.2. Exchange APDU Data (2C-13)

This command allows exchange of application level APDU's 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 this pass-through command can be used, a Level 1 session must have been established 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 device, if the interface is Contactless, the APDU data being received from the card/device by the reader will be 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 White List, the Command will return a Parameter Not Supported error (0x06).

8.15.3. Contact Transaction Power Off (2C-18)

This command can close contact transaction power. When transaction is success, the card needs power off before remove 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			

8.15.4. Set ICC Voltage Option (2C-19)

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			

8.15.5. Get ICC Voltage Option (2C-1A)

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)		

8.15.6. Set ICC Reader Type Option (2C-1B)

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		

8.15.7. Get ICC Reader Type Option (2C-1C)

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)		

8.16. High-level Pass-Through Commands for Mifare Cards

This section contains serial commands that implement higher level functionality for the Mifare Cards. These commands can only be used after the reader has been put in Pass-Through mode and the "Poll for Token" command has indicated that a Mifare Card is present. These commands do not work for non-Mifare cards.

8.16.1. Mifare Authenticate Block (2C-06)

This command allows the terminal to instruct the ViVOpay reader to authenticate the Mifare Card sector containing the specified block of data. The Key to be used is also specified by the terminal. This command is applicable only for 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 (MSB)	CRC (LSB)
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 the authentication. It sets the key to be used for the subsequent authentication. The actual authentication will be performed before the next read or write operation. If a sector boundary is crossed, the reader will attempt to authenticate using the key that was established with this command.

After receiving the **Command Frame**, the ViVOpay reader verifies the data and if the data is valid, it interacts with the Mifare card to authenticate the sector containing the specified block. If this operation is successful, the ViVOpay 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 Mifare card, a 0Ch (Sub-command not allowed) will be returned in the status code.

8.16.2. Mifare Read Blocks (2C-07)

Use this command to read data from one or more blocks on the Mifare Card. The terminal 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. For multi-block reads, the sector trailer will be skipped. Sector trailer's may be read (except that the keys will not be visible) using a single block read.

If the card specified is a Mifare Standard card, then the terminal must have successfully sent 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 use by the reader as it performs reads and writes.

If the card specified is a Mifare Standard card and the read command specifies a single block read, then the reader tries to read the data regardless of whether the block is a sector trailer block.

If the card specified is a Mifare Standard card, and the read is a multi-block read, then the reader skips reading the sector trailer blocks that contain the Keys (because the Keys cannot be read).

Skipped blocks are not included in the block count. While reading blocks in a Mifare Standard Card, if the read requires access to the next sector, then the ViVOpay reader carries out authentication for this block/sector automatically by using the Key Type and Key Value that were set in the Mifare Authenticate Block command 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 14,15	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	07h	00h	02h	See Table below		

Mifare Read Block Data Field

Data Field	Length (bytes)	Description
Card & Block Count	1	Card Type: [Bit 7..4] This can only indicate the following cards Mifare Type A (Standard) Mifare Type A (Ultralight)

		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 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.
Start Block	1	This is the card block number from which the reader starts reading.

After receiving the **Command Frame** the ViVOpay reader verifies the parameters. If the parameters are valid, then it reads the data from the card. If this operation is successful, the ViVOpay reader sends a **Response Frame** containing a Status of OK and the data that was read. 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

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.

For SRED device, the data being received from the card/device by the reader will be 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 White List, the Command will return a Parameter Not Supported error (0x06).

8.16.2.1. Reading Mifare Ultralight Cards

Mifare Ultralight cards differ from Mifare Cards. For Mifare Standard Cards the block size is 16 bytes. However, for Mifare Ultralight Cards the 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 a read is requested starting at Block 3 and Block Count is 1 then 16 bytes of data are returned consisting of Page # 3, 4, 5 & 6. And if a read is requested starting at Block 3 and Block Count is 2 then 16*2=32 bytes of data are returned consisting of Page # 3, 4, 5, 6, 7, 8, 9, and 10.

Typically, Mifare Ultralight Cards are read by the ViVOpay reader, but are not written. This is because they are typically used for disposable applications such as ticketing.

8.16.3. Mifare Write Blocks (2C-08)

Use this command to instruct the ViVOpay reader to write data to one or more blocks on the Mifare Card. The terminal can instruct ViVOpay 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. For Mifare Standard Cards the block size is 16 bytes. For Mifare Ultralight Cards the block size is 4 bytes.

If the card specified is a Mifare Standard card, then the terminal must have successfully sent 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 for use by the reader as it performs reads and writes.

If the card specified is a Mifare Standard card and the write command is a single block write, the reader tries to write the data regardless of whether the block is a sector trailer block or not.

If the card specified is a Mifare Standard card, and the write is a multi-block write, then the reader skips writing to the sector trailer blocks that contain the Keys. Skipped blocks are not included in the block count. While writing blocks to a Mifare Standard Card, if the write requires access to the next sector, then the ViVOpay reader carries out authentication for this block/sector automatically by using the Key Type and Key Value that were used by the terminal 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 (MSB)	CRC (LSB)
ViVOtech2\0	2Ch	08h	Variable	Variable	See Table below		

Mifare Write Block Data Field

Data Field	Length (bytes)	Description
Card & Block Count	1	Card Type: [Bit 7..4] This can only indicate the following cards Mifare Type A (Standard) Mifare Type A (Ultralight) 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.
Start Block	1	This is the card block number from which the reader starts writing.
Data to Write	Variable (multiple of block size)	Data to write to the Card. The length of the data to be written to the card depends on the number of blocks to be written and the card type.

After receiving the **Command Frame** the ViVOpay reader verifies the parameters. If the parameters are valid, it writes the data to the card. If this operation is successful, the ViVOpay reader sends a **Response Frame** with a Status of OK.

If the **Command Frame** is invalid or the write operation fails then 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)
ViVOtech2\0	2Ch	See Status Code Table	00h	00h		

8.16.4. Mifare ePurse Command (2C-0A)

Use this command to instruct the ViVOpay reader to carry out Debit, Credit and Backup operations on value blocks in a Mifare card. These functions require that the related data blocks be 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				$\overline{\text{Value}}$				Value				Adr	$\overline{\text{Adr}}$	Adr	$\overline{\text{Adr}}$

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 Purse 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 Purse 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 Purse 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 terminal must have successfully sent at least one Mifare Authenticate Block command to the reader for the first block.

Warning: Multiple 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 (MSB)	CRC (LSB)
ViVOtech2\0	2Ch	0Ah	Variable	Variable	See Table below		

Mifare ePurse Command Data Field

<table>
<tr><th>Data Field</th><th>Length (bytes)</th><th colspan="8">Description</th></tr>
<tr><td rowspan="2">Mode, Card Type & Operation Count</td><td rowspan="2">1</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>1 = Inc
0 = Dec</td><td colspan="3">Card Type</td><td colspan="4">Operation Count</td></tr>
<tr><td></td><td></td><td colspan="8">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.</td></tr>
<tr><td>Purse Function Blocks</td><td>Variable [1]</td><td colspan="8">Series of any combination of supported Purse Function blocks (Debit/Credit, Backup). Refer to the description of each individual Command Frame below.</td></tr>
</table>

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 ViVOpay reader verifies the parameters. If the parameters are valid, it performs the operations specified in the order in which they appear within the Purse 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 Purse 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 ViVOpay reader sends a **Response Frame** with a Status of OK.

If the **Command Frame** is invalid or any of the operations fail then the reader sends a **Response Frame** with an appropriate Status.

Response Frame

Byte 0-9	Byte 10	Byte 11	Byte 12	Byte 13	Byte14-15	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 Purse 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 Purse 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 Purse 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 Purse 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						

8.17. 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.

8.17.1. 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)
ViVOtech2\0	2Ch	40	00	00	See Below		

Individial commands in NFC command set are distingushed 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)
ViVOtech2\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 or Could not Active) 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) 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 device, if the command isn't "Poll for a NFC Tag", the data being received from the card/device by the reader will be 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 White List, the Command will return a Parameter Not Supported error (0x06).

8.18. 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.

8.18.1. 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 a 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)
ViVOtech2\0	2Ch	41	Variable	Variable	See Below		

Individial commands in Felica command set are distinguished as to 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)
ViVOtech2\0	2Ch	See Status Code Table	00	00	Returned data from card		

For details on these data field, refer to the relevant Felica Specifications.

For SRED device, the data being received from the card/device by the reader will be 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 White List, the Command will return a Parameter Not Supported error (0x06).

8.18.2. Felica Lite/Lite-S Authenticate (2C-42)

This command allows the terminal to instruct the ViVOpay reader to authenticate the Felica Lite/Lite-S card. The Key to be used is also specified by the terminal. 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 ViVOpay 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 ViVOpay 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, a 0Ch (Sub-command not allowed) will be returned in the status code.

8.18.3. 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 terminal 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 terminal 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 ViVOpay 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 ViVOpay reader sends a Response Frame containing a Status of OK and the data that was read. If the write operation is successful, the ViVOpay 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		

For SRED device, the data being received from the card/device by the reader will be 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 White List, the Command will return a Parameter Not Supported error (0x06).

8.18.4. Poll Felica Card (2C-44)

Special Poll command for polling 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`

8.19. Pass-Through Commands for Motor control

This section contains serial commands apply specifically to the VP5300M reader.

8.19.1. 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

8.19.2. 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-Though mode open**
   IDG command: 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**
   IDG command: 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**
   IDG command: 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**

   IDG command: `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**

   IDG command: `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`

8.20. Secure Pass-Through Function

Note: In SRED device, Pass-Through Mode is called secure Pass-Through Mode.

8.20.1. 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 White List. An AID or BIN added to the White List 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 White List, 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 White List

8.20.2. Secure Pass-Thru White List

The White List is loaded with MAC verification. The White List 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 White List, 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 White List, 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 White List, 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 White List 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 White List 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 White List.

8.20.3. 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.

8.20.4. 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

8.20.5. Secure Pass-Thru White List

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 White List.

White List related commands can only be used when Pass-Through mode is started.

8.20.6. Get White List (2C-51)

This command retrieves the White 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	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 White List above for the White List format and options.

Data Field	Length (bytes)	Description
White List	Var	The actual white list. See format of White List in Set White List section above. Tag: FFEE0C Format: Constructed

8.20.7. Get Discretionary Data White List (2C-53)

This command retrieves the Discretionary Data White List.

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)
ViVOtech2\0	2Ch	Status	Variable	Variable	Response Data		

Response Data - See tables for Set White List above for the White List format and options.

Data Field	Length (bytes)	Description
White List	Var	The actual white list. See format of White List in Set White List section above. Tag: FFEE0C Format: Constructed

Example Response:

```
FFEE0C70FFEE0A14DFEF2106373037363030DFEF2106373037363939FFEE0A14DF
EF2106373737373030DFEF2106373737373939FFEE0A14DFEF2106363232303236
DFEF2106363232303237DFEF2106333038333737DFEF2106363030363439DFEF21
06363031303536DFEF2106363239383539
```

8.21. Logger commands

The NEO II 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.

8.21.1. 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 (MSB)	CRC (LSB)
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;

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		

8.21.2. 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 (MSB)	CRC (LSB)
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		

8.21.3. Get/Set Logger Size (4C-03)

Use this command to get/set max count of logger ring buffer.

8.21.3.1. 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 (MSB)	CRC (LSB)
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.

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		

8.21.3.2. 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 (MSB)	CRC (LSB)
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		

8.21.4. Enable/Disable Logger (4C-04)

Use this command to enable or disable 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 (MSB)	CRC (LSB)
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		

8.21.5. Save/Delete Log (4C-05)

Use this command to save the log from ring buffer to a file or delete the current log file.

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	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		

8.21.6. 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 (MSB)	CRC (LSB)
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 whole log from file and 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.

8.22. Secure Communication

Special Considerations for Secure Communications

Take time to familiarize yourself with certain key differences in device usage that come into play when secure communications are required (as described below).

8.22.1. 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 devices, because SRED devices are encryption-enabled.

8.22.2. 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 (including actual data in the two output styles), see the appendix called TDES Data Encryption Examples, and/or consult the appendix on Enhanced Encrypted MSR Data Output Format.

8.22.3. Encryption Algorithms

The reader uses TDES encryption by default. During the authentication phase, the reader will use TDES in ECB mode. After the reader and terminal are authenticated, the data field in the command/Response Frames is encrypted with Cipher Block Chaining (TDES-CBC).
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.

8.22.4. Secure Data Exchange

Data is encrypted using TDES-CBC. After a session is established, the initial vector 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.

8.22.5. 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.
2. Blockwise encrypt, using CBC.

The order of operations for receiving frames:

1. Decrypt using CBC.
2. Remove pads.

8.22.6. Get Data Encryption Key Variant Type (C7-30)

Command Frame

Byte 0-9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte15
Header Tag & Protocol Version	Command	Sub-Command	Data length (MSB)	Data length (LSB)	CRC (MSB)	CRC (LSB)
ViVOtech2\0	C7h	30h	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	Key Variant Type		

Encryption Type	Description
0	Data Variant
1	Pin Variant

8.22.7. Set DUKPT Key Encryption Type (C7-32)

This command exists to specify the encryption type of Data encryption Key, and MUST be used before the initial loading of the Data encryption Key into the device. The encryption type CANNOT be changed after the Data encryption Key is present. It must remain either TDES or AES.

Note: This command is only supported in non-SRED devices. In SRED device, 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	Byte16
Header Tag & Protocol Version	Command	Sub-Command	Data length (MSB)	Data length (LSB)	Data	CRC (MSB)	CRC (LSB)
ViVOtech2\0	C7h	32h	00	01	Encryption Type		

Encryption Type	Description
0	TDES
1	AES
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		

8.22.8. Get DUKPT Key Encryption Type (C7-33)

Note: This command is only supported in NSRED device. In SRED device, 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	Byte15
Header Tag & Protocol Version	Command	Sub-Command	Data length (MSB)	Data length (LSB)	CRC (MSB)	CRC (LSB)
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
```

8.22.9. 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 device. *This is meant to be an irreversible event.*

Note: This command is supported only in non-SRED devices. In SRED devices, 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 (MSB)	CRC (LSB)
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, device will always respond plaintext data
When Data Encryption is enabled, device will output data as follows:

(1) When Data encryption Key does not exist, the commands below will respond plaintext data for NSRED device, respond status code 0x90 and no data for SRED device.
(2) When Data encryption Key exists and is valid, the commands below will respond encrypted data.
(3) When Data encryption Key exists and exhausted, the commands below will respond status code 0x91 and no data.

Commands:

(1) Activate Transaction Command (02-01/02-40)
(2) Get Transaction Result Command (03-00/03-40)

Device Current Status/Command Setting/Device Operation/Output Matrix （Data encryption Key exists and valid）

Current Encryption Status	Command Encryption Setting	Device 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
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

8.22.10. Get Data Encryption Enable Flag (C7-37)

Note: This command is only supported in Non-SRED version devices, not supported in SRED version devices.

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 (MSB)	CRC (LSB)
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

8.22.11. Set MSR Secure Parameters (C7-38)

This command allows setting parameters that determine encrypted output from MSR sessions. Use it to force encryped data output to include various kinds of data per Enhanced Encrypted MSR Data Output When Encryption is Turned On with C7-38 Command. Consult the table in Appendix (A.13) 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: T3 force encrypt when card type is 80 Bit 4: Hash option for non-PCI products; 1 is on, 0 is off Default value is 0x10.	b	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					
ViVOtech2\0	C7h	38h	00h	05h	MSR Secure Parameters TLV		

8.22.12. Get MSR Secure Parameters (C7-39)

This command can get parameters from flash setting.

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		

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: T3 force encrypt when card type is 80 Default value is 0x10.	b	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	00h	05h	TLV		

8.22.13. Set MSR Tracks (C7-44)

This command can get parameters from flash setting.

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		

Track Select ID

Value	Description	Length
'0' ('0' ~ '9')	Any Track '0' -any; '1'-'7' – bit 1 tk1, bit 2 tk2; bit 3 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		

8.22.14. 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 (MSB)	CRC (LSB)
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 (LSB)	CRC (MSB)
ViVOtech2\0	C7h	See Status Code Table	00h	01h	Track Select ID		

Track Select ID

Value	Description	Length
'0' ('0' ~ '9')	Any Track '0' -any; '1'-'7' – bit 1 tk1, bit 2 tk2; bit 3 tk3 '8' – tk1-2 '9' – tk2-3	1

8.22.15. Load Certificate for TransArmor Encryption (C7-50)

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		

8.22.16. Set TransArmor TID (C7-51)

Set 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		

8.22.17. Get TransArmor TID (C7-52)

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		

8.22.18. 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

8.22.19. 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		

8.23. Key Injection and Related Commands

8.23.1. 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 Codes	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

8.23.2. 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 Codes	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

8.24. PIN Pad (Keypad) Commands

8.24.1. Secondary Response Mechanism

Except for 62-00, all 62-xx commands use the secondary response mechanism.

Device sends a first ACK response immediately after 62-xx command is received, and a second response shall be sent after the operation of received command is completed or timed out.

During the operation (that is, the interval between the first and second response), only the 62-00 command is acceptable, which can be used to cancel the current 62-xx command; other commands shall respond with the fail (0x0A) status code immediately.

8.24.2. Status and State Codes in Secondary Response

Basically, the Data Field of the second response contains a state code which indicates the detail information as below, and 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
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 shall be empty.

8.24.3. Status Codes

Most commands have an immediate response, and a secondary response. If timeout occurs, the Second Response Status Code is 0x08 and the Data Field is empty. If the operation is not allowed, Second Response Status Code is 0x0C. If success or failure, Second Response Status Code is 0x00 or 0x0A, respectively, and Data Field contains "State" to indicate detail information. See below.

State Code in PIN and Key Entry Commands

00h – Success

01h – Fail, Key Pad Cancel

02h – Fail, External Command Cancel

03h – Fail, Invalid input parameters

04h – Fail, PAN error

05h – Fail, PIN DUKPT Key is absent

06h – Fail, PIN DUKPT Key is exhausted

07h – Fail, Display message error

8.24.4. Cancel PIN or Key Entry (62-00)

Use this command to cancel the below-listed 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 Codes	00h			

8.24.5. Display Message and Get Encrypted PIN (62-01)

For devices that allow PIN entry (such as VP3600), use this command to get encrypted PIN from device.

NOTE: only one PIN attempt is allowed per 30-seconds for MKSK.

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	00h – MKSK to encrypt PIN, Internal PAN (from MSR or Manual PAN Entry or Contactless EMV Transaction) 01h – DUKPT to encrypt PIN, Internal PAN (from MSR or Manual PAN Entry or Contactless EMV Transaction) 10h –MKSK to encrypt PIN, External Plaintext PAN 11h – DUKPT to encrypt PIN, External Plaintext PAN 20h –MKSK to encrypt PIN, External Ciphertext PAN (for PIN pad only) 21h – DUKPT to encrypt PIN, External Ciphertext PAN (for PIN pad only) **Note:** MKSK is either standard, or Desjardins-based, according to Key Loading. DUKPT is TDES or AES according to Key Loading.
PAN length	1	Length of PAN (If Internal PAN, this field is 00h) If TDES PIN DUKPT Key, value is 16 bytes digital If AES PIN DUKPT Key, value is 12~19 bytes digital
PAN	var	If PAN is 00h, this field is not present. ASCII code for digital (30h-39h). If TDES PIN DUKPT Key, length is 16 bytes. If AES PIN DUKPT Key, length is 12~19 bytes. Device will check PAN length, if length is error, device will respond with fail state code.
PIN Max Length	1	Max Len of PIN, valid range is 4 ≤ min ≤ max ≤ 12
PIN Min Length	1	Min Len of PIN, valid range is 4 ≤ min ≤ max ≤ 12
LCD Length	1	Length of LCD message. If no LCD message input, length is 00h.
LCD Message	1-16	If LCD Length is 00h, this field is not present. ASCII code Display Message.

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 Codes	00h			

After device receives this command, it will send ACK as the first response. Then device will begin to wait for PIN entry. When PIN is entered or timeout occurs, device will send PIN Block or timeout as second response. Between first and second response, no command should be sent (other than Cancel).

Note:

- Timeout is 3 minutes, the Pin Len default is 4~12
- Per 20 Seconds, if the PIN length was not zero, the PIN would be clear
- While you press numeric key, Device will increase display "*" on LCD if Total PIN length is smaller than 12. Line 1 display:
- If Enter 2 numeric: **
- If Enter 12 numeric: ************
- While you press Backspace key, Device will decrease display "*" on LCD if Total PIN length is not 0.
- While you press Cancel key, Device will display cursor on LCD if Total PIN length is not 0, or Device will quit the work state.

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 Codes	var		See Tables Below		

- TDES DUKPT Key mode
  - Build Format 0 PIN Block with PAN. PIN Block is encrypted using TDES algorithm & PEK (DUKPT).
- AES DUKPT Key mode:
  - Build Format 4 PIN Block with PAN. PIN Block is encrypted using AES-128 algorithm & PEK (DUKPT).
- MKSK mode:
  - Build Format 0 PIN Block with PAN. 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)

8.24.6. Get Function Key (62-02)

Use this command to capture a single Keypress value from device.

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 Codes	00h			

After device receives this command, it will send ACK as first response. Then device will begin to wait for Function Key. When Function Key is entered or timeout, device 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 Codes	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, Device sends "B"
- While you press Cancel key, Device sends "C"
- While you press Enter key, Device sends "E"
- While you press '*' key, Device sends "*"
- While you press '#' key, Device sends "#"
- While you press 'F1' key, Device sends "F1"
- While you press 'F2' key, Device sends "F2"
- While you press 'F3' key, Device sends "F3"
- While you press 'F4' key, Device sends "F4"

8.24.7. Display Message and Get Numeric Key (62-03)

Use this command to make device display requested message and then get Numeric Key sequence from device.

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 ≤ min ≤ max ≤16
Key Min Length	1	The min length for numeric key, valid range is 1 ≤ min ≤ max ≤16
Display Message	1~16	Plaintext Display Message. ASCII code.

Data Field	Length (bytes)	Description
Display Message Signature	256	Display message signed by Private Key of Secure Message Certificate using RSAPSS algorithm. Algorithm: 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 Codes	00h			

After device receives this command, it will send ACK as first response. Then device will begin to wait for Numeric Key press. When Numeric Key is entered or timeout, device 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 Codes	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 sends this command, device 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, Device 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, Device will decrease display numeric on LCD if Total numeric length is not 0.
- While you press Cancel key, Device will display cursor on LCD if Total numeric length is not 0, or Device will quit the work state.

8.24.8. Display Message and Get Amount (62-04)

Use this command to make device display requested message and then get **amount** value from device.

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 ≤ min ≤ max ≤15
Amount Min Length	1	The min length for amount, valid range is 1 ≤ min ≤ max ≤15
Display Message	1~16	Plaintext Display Message. ASCII code.

Data Field	Length (bytes)	Description
Display Message Signature	256	Display message signed by Private Key of Secure Message Certificate using RSAPSS algorithm. Algorithm: 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 Codes	00h			

After device receives this command, it will send ACK as first response. Then device will begin to wait for Enter Amount. When Amount is entered or timeout, device 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 Codes	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 host send this command, device 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, Device 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, Device will decrease display amount on LCD if Total amount length is not 0.
- While you press the Cancel key, Device will display cursor on LCD if amount length is not 0, or Device will quit the work state.

8.24.9. Display Multi Line Message and Get Numeric Key (62-05)

Use this command to make device display requested message and then get Numeric Key sequence from device.

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 ≤ min ≤ max ≤16
Key Min Length	1	The min length for numeric key, valid range is 1 ≤ min ≤ max ≤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: 1. Calculate 32 bytes Hash for "< Function Flag >< Key Max Length >< Key Min Length >< Display Message 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
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 Codes	00h			

After device receives this command, it will send ACK as first response. Then device 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, device 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 Codes	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 host sends this command, device 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, Device 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, Device will decrease display numeric on LCD if Total numeric length is not 0.
- While you press Cancel key, Device will display cursor on LCD if Total numeric length is not 0, or Device will quit the work state.

8.24.10. 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 device 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)
ViVOpayV3\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

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

Byte No.	Bit No.	Flag	Value	Meaning
	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
1	7	Special processing flag.	0 1	Standard processing. 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.
	0 – 6	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.

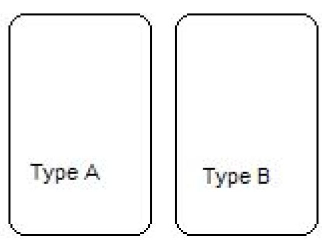
8.25. Improved Collision Detection

8.25.1. Issues with Standard Collision Detection

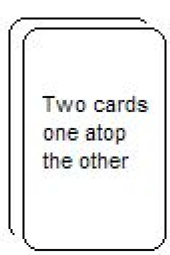
This firmware supports the EMV Contactless Communication Protocol Specification. While the EMV specification defines collision detection, there are often physical constraints which prevent collision detection resolving within the timing limits outlined in the specification.
For instance, multiple cards 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:

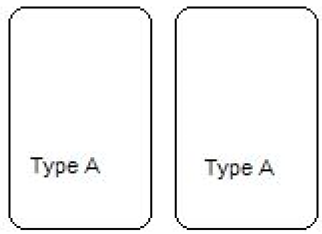
- When a single type A and single type B card are presented side-by-side the reader will detect collisions without much difficulty:



- 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.

8.25.2. 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.

8.25.2.1. 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.

8.25.2.2. 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.

8.26. Firmware Downloader Commands

8.26.1. 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 device. 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).

8.26.2. 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			

8.26.3. Enter Boot Loader Process from Main Application (C7-41)

Host 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		

8.26.4. 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

8.26.5. 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 (MSB)	CRC (LSB)
ViVOtech2\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

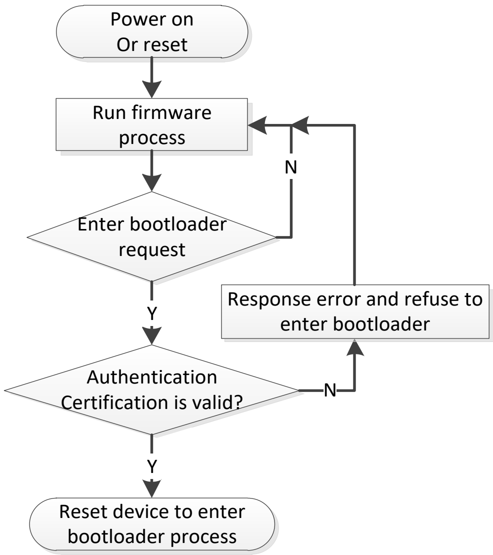
Byte 0-9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14 – Byte 14+n-1	Byte 14+n	Byte 15+n
Header Tag & Protocol Version	Command	Status Code	Data Length (MSB)	Data Length (LSB)	Data	CRC (MSB)	CRC (LSB)
ViVOtech2\0	C7h	See Status Code Table	00h	00h			

Note: refer to document"NEO_II Bootloader PDS_V52.doc" for detailed BootLoader specifications.

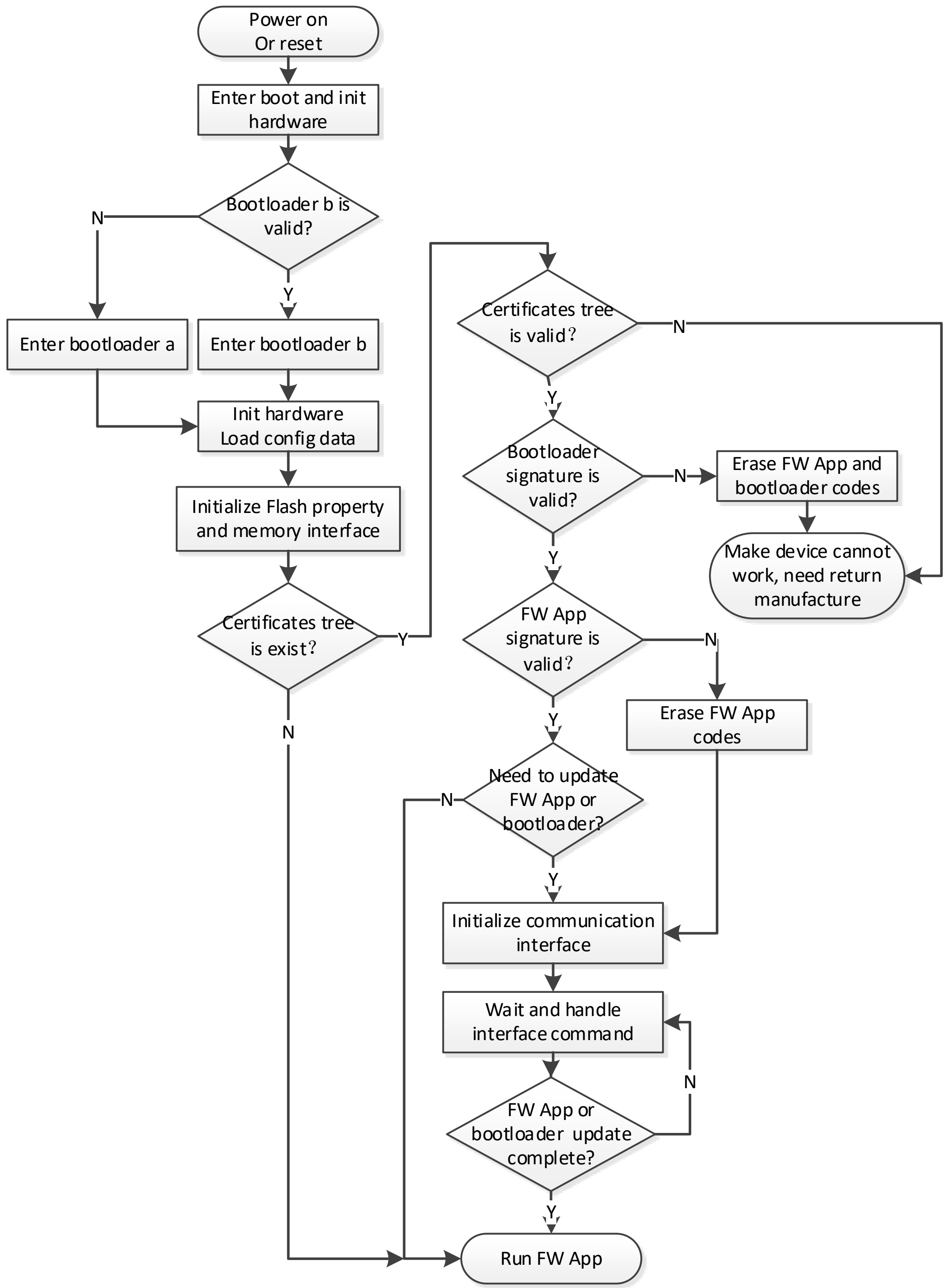
8.27. Firmware Downloader Command Processing Flow

Firmware downloader commands must be sent to device in sequence as below.

(1) Enter Bootloader process



(2) Bootloader process



8.27.1. Start Update Kernel Process (C7-11)

This is the first command sent by host to open a firmware update process.

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	11h	00h	0Ch	See Below	

In this command, data length must be 12 bytes:

The first one byte is Kernel ID.

Kernel Name	ID	Support
External EMV CTL2 Kernel	0	Yes
EMV CLL2 PPSE Kernel	1	Yes
EMV CLL2 VCPS Kernel (Visa)	2	Yes
EMV CLL2 MChip Kernel	3	Yes
EMV CLL2 DPass Kernel	4	Yes
EMV CLL2 Amex Kernel	5	Yes
EMV CLL2 Interac Kernel	6	Yes
EMV CLL2 CUP Kernel	7	Yes
EMV CLL2 JCB Kernel	8	No
APVAS	9	No
SmartTap	10	No
Reserved	11	NC

The next 3 bytes is Kernel Firmware data total length. (LSB format)
The other 8 bytes are reserved.
If the Kernel is Amex and total size is 72018 (0x011952), the data should be 05 52 19 01 xx xx xx xx xx xx xx xx.

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	C7h	See Status Code Table	00h	00h		

8.27.2. Send Kernel Firmware Data (C7-14)

This command is sent by host to program Kernel Firmware related zone according to ID. The Kernel Firmware data (Firmware Codes + Signature) should be m*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 (LSB)	CRC (MSB)
ViVOtech2\0	C7h	See Status Code Table	00h	00h		

8.27.3. End Update Kernel Process (C7-15)

This is the last command sent by host 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 (LSB)	CRC (MSB)
ViVOtech2\0	C7h	See Status Code Table	00h	00h		

8.28. Power Management Commands

8.28.1. 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.n

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`.

8.28.2. 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:

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 Standby 0
- `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			

8.28.3. 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.
1st lower power timer	2	
…		
4th 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`

8.28.4. 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

8.29. 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		

8.29.1. 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)

8.29.2. 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		

8.30. USB Power Mode Suspend Configuration

8.30.1. 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 devices.

Note: "USB Power Mode Suspend" is a special term used by the USB Implementers Forum (USBIF), which requires USB devices 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		

8.30.2. 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

8.31. Miscellaneous NEO II Platform Commands

8.31.1. Removal Tamper Configuration (04-22)

Use this command to enable removal tamper detection, *or* to get tamper status.

NOTE: Only TriMag IV (ID TECH ASIC) physical tamper features are used for this functionality.

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 tamper 02h: Get tamper 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 Codes 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

8.31.2. 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

8.31.3. 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
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)		

8.31.4. 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 Key ID 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_SEVER_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:

IDG command: `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 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.

8.31.5. Get Certificate (81-0F)

This command is used to retrieve individual certificate using ID Tech's certificate Key ID 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)
ViVOtech2\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)
ViVOtech2\0	81h	See Status Code Table	Var	Var	Certificate		

8.31.6. Save Security Data (C7-56)

This command is available only in certain devices that support EasyCard. Contact your ID TECH representative if you are unsure whether this applies to your device.

Use this command to store sensitive data of customer in a device security area that will be erased when the device is attacked.

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	C7h	56h	02h	01h	See Table below		

Data Field

Data Field	Length (bytes)	Description
Type	1	0x00: 512Bytes Others: Reserved
Security Data	512	Any value you want to save

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	00h			

8.31.7. Read Security Data (C7-57)

Use this command to retrieve sensitive data (previously set with C7-56) from a device security area.

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	C7h	57h	00h	01h	See Table below		

Data Field

Data Field	Length (bytes)	Description
Type	1	0x00: 512 bytes Others: Reserved

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	02h	00h	See Table below		

Data Field

Data Field	Length (bytes)	Description
Security Data	512	Arbitrary data previously stored with C7-56.

8.31.8. Control IO (01-06)

This command is used to control output PIO level to test PIO. This command is only valid before delivery.

Command Frame

Byte 0-9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14 – Byte 14+n-1	Byte 14	Byte 15
Header Tag & Protocol Version	Command	Sub-Command	Data Length (MSB)	Data Length (LSB)	Data	CRC (LSB)	CRC (MSB)
ViVOtech2\0	01h	06h	00h	04h	See Below Data Table		

Data Table: Control IO Command Data

Data Item	Length (bytes)	Description
PIO Group	1	Ascii code 'A' – PTA, 'B' – PTB, 'C' – PTC, 'D' – PTD, 'E' – PTE, 'F' – PTF
PIO Number	2	Ascii code '0' '0' – PT?0 '0' '1' – PT?1 … '1' '0' – PT?10 '1' '9' – PT?19 …
PIO status	1	Ascii code '0' – Output Low '1' – Output High

Example1: Control PTA1 output High, Command data is: 41 30 31 31
Example 2: Control PTC15 output Low, Command data is: 43 31 35 30

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	01h	See Status Code Table	00h	00h		

8.31.9. Get Device Status (32-03)

Use this command to return device 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

8.31.10. 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.

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

8.31.11. Set Switches Notifications Configuration (01-07)

This command is used to set the switches notifications 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	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

8.31.12. Get Switches Notifications Configuration (01-08)

This command returns the configuration for switches notifications.

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

8.31.13. Get DRL Reader Risk Parameters (03-0E)

This command returns the Index, Application Program ID, and reader risk parameters for the DRL 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	03h	0Eh	00h	01h	DRL Index (01-04)		

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		

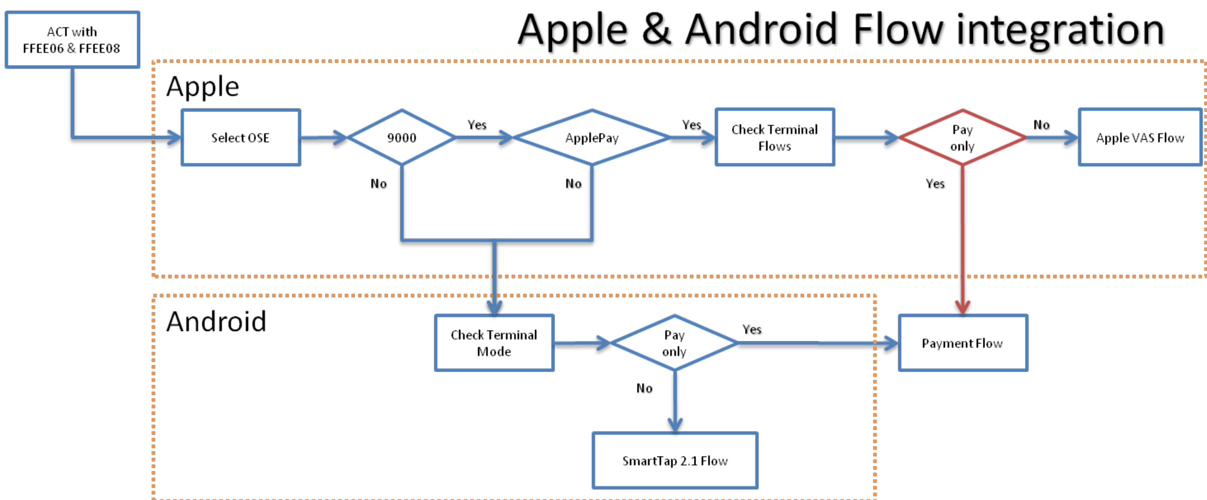
The response TLV Data Objects are formatted as shown in the table DRL Transaction TLVs (Index, 9F5A, FFF1, FFF4, FFF5, and 9F1B).

9. Loyalty Functions

NEO II firmware has built-in support for Apple Pay VAS (Value Added Services) and Google Smart Tap.

Support for Google SmartTap functionality is described in a separate document, *Google SmartTap 2.1 in ViVOpay Devices.pdf*, available for download on the ID TECH Knowledge Base. Consult that document for SmartTap-related functionality. It supersedes this IDG where SmartTap is concerned.

9.1. Apple VAS & SmartTap 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 SmartTap 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 00 00
00 00 00 E7 6F
```

Set all configurations for SmartTap 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 SmartTap 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 SmartTap 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 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 SmartTap 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 II 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 terminal's merchant records must first beset. This should be done using TLVs sent in the Set Merchant Record (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 device can store at most 6 Merchant Records for Apple VAS. This means the device 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.

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.

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: Terminal Type Byte 3: RFU Byte 4: Terminal Mode See table further below.
9F22	ApplePay Terminal Application Version Number	Required	Hard-defined as 01.00 for now.
9F2B	ApplePay VAS Filter	Optional	If not provided filtering will not be performed by the mobile.
DFEE01	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	Required	If not provided the following settings are used by default: • Full VAS protocol • No beeps for VAS • EMEA Communications Error Handling If provided the bits define the settings.
DFED49	ApplePay VAS Transaction Only 0: Transactions behave normally 1: VAS transactions only	Required	Sets transaction to VAS-only if enabled.

Tag 9F26 ApplePay Terminal Capabilities Information

Byte 1: Format

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 Terminal
-	-	-	-	0	0	0	1	Transit Terminal
-	-	-	-	0	0	1	0	Access Terminal
-	-	-	-	0	0	1	1	Wireless handoff Terminal
-	-	-	-	0	1	0	0	App Handoff Terminal
-	-	-	-	1	1	1	1	Other Terminal
-	-	-	-	-	-	-	-	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
x	x	x	x	x	x	x	x	Terninal in VAS App OR Payment Mode
								Terminal in VAS App AND Payment Mode
								Terminal in VAS App Only Mode
								Terminal in Payment Mode Only
								Bits b8-b3 shall be set to 0
								All other values are RFU

Byte 4: Terminal Mode polling Loop

ApplePay Terminal Capabilities Information	Card Filter Settings
00	PICC_POLL_TYPE_APPLE_VAS_OR_PAY, PICC_POLL_TYPE_A and PICC_POLL_TYPE_B
01	PICC_POLL_TYPE_APPLE_VAS_AND_PAY, PICC_POLL_TYPE_A and PICC_POLL_TYPE_B
10	PICC_POLL_TYPE_APPLE_VAS_ONLY, PICC_POLL_TYPE_A and PICC_POLL_TYPE_B
11	PICC_POLL_TYPE_APPLE_PAY_ONLY, PICC_POLL_TYPE_A and PICC_POLL_TYPE_B

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 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 terminal 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

Vendi 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
```

Vendi Response Breakdown:

- `56 69 56 4F 74 65 63 68 32 00` – ViVOTech2 header
- `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 ContainerViVOpay
    - `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 VAS Failure in Select

- 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 ContainerViVOpay
    - 9A nn – Date
    - 9F21 nn - Time
    - DF02 nn – ApplePay VAS Failure Report
- xx xx – CRC for entire response

ApplePay VAS Failure Report

- DF0204206A8002
- DF02 - ApplePay VAS Failure Report Tag
- 04 – Length of ApplePay VAS Failure Report
- 20 = Error Code, See IDG for Error Code Encoding
- 6A80 – SW1-SW2 Status from last APDU received
- 02 – RF State failure occurred in. 02 = Select

Transaction Response for VAS Failure in Get Data

- 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 ContainerViVOpay
    - 9A nn – Date
    - 9F21 nn - Time
    - 9F25 nn – Merchant ID **a**
    - DF02 nn – ApplePay VAS Failure Report for Merchant ID a
    - 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

9.2.2. 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	Byte15+n
Header Tag & Protocol Version	Command	Sub-Command	Data length (MSB)	Data length (LSB)	Data	CRC (MSB)	CRC (LSB)
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		

EXAMPLE:

Set Merchant Record command using the SDK:

```
idtVendi.device_sendDataCommand("041101013CC70ED89A9D4354BE9830AB5
8D89C6FE7E62BACA939D2A6851DFC602EA798F7000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000",
false,resDataStruct);
```

Set Merchant Record command via raw firmware commands:

```
56 69 56 4F 74 65 63 68 32 00 04 11 00 63 01 01 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 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 00 00
00 00 00 25 B1
```

Breakdown of command sent:

- `56 69 56 4F 74 65 63 68 32 00` – ViVOtech2\0 header
- `04` – Set Merchant Command
- `11` – Set Merchant Sub-Command
- `00 63` – Data Length
- `01` - Merchant Index number
- `01` - Merchant ID is enabled
- `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` - Merchant ID (this is the SHA-256 hash of the IDTech Pass having the name "pass.com.apple.wallet.vas.prodtest")
- `00` - Length of VAS URL. In this example, none was provided.
- `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 00 00 00 00` – URL if provided
- `25 B1` – CRC-16

Vendi Response:

```
56 69 56 4F 74 65 63 68 32 00 04 00 00 00 AE 16
```

Breakdown of Vendi Response:

- `56 69 56 4F 74 65 63 68 32 00` – ViVOtech2\0 Header
- `04` – Command
- `00` – Status (see table "Status Codes For Protocol 2")
- `00 00` – data
- `AE 16` - CRC

9.2.3. 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 (MSB)	CRC (LSB)
ViVOtech2\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)
ViVOtech2\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. SmartTap 2.1

ViVOpay readers running NEO II firmware have a built-in SmartTap kernel that provides support for Google SmartTap. 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 (MSB)	CRC (LSB)
ViVOtech2\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)
ViVOtech2\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 (MSB)	CRC (LSB)
ViVOtech2\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).

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 terminal configuration settings. Use Set Configuration (04-00) or Contact Set Terminal Data (60-06) to send the settings to your device as you normally would.

NOTE: If this tag does not exist in Terminal Settings, 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)

Sentinels

For any found data element of Track1, Track2 or Track3, 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"

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.

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 provider the length of Track and PAN in Tag DFEF4B

2) Tag DFEF4D providers 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 terminal.

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 Terminal 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	FBh
Track 2 Data 9607997242183	Clearing Record Not Present		

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 terminal.

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. Terminal has to 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^0508101335373336072222

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. Terminal has to issue an Activate command to allow the reader to poll for a card and carry out a transaction.

2. Command: Set Configuration (Terminal 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 Terminal Country Code	TLV Trans Currency Code		

Assuming the current terminal values is 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 terminal has to 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 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 Appendix A-10.

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.

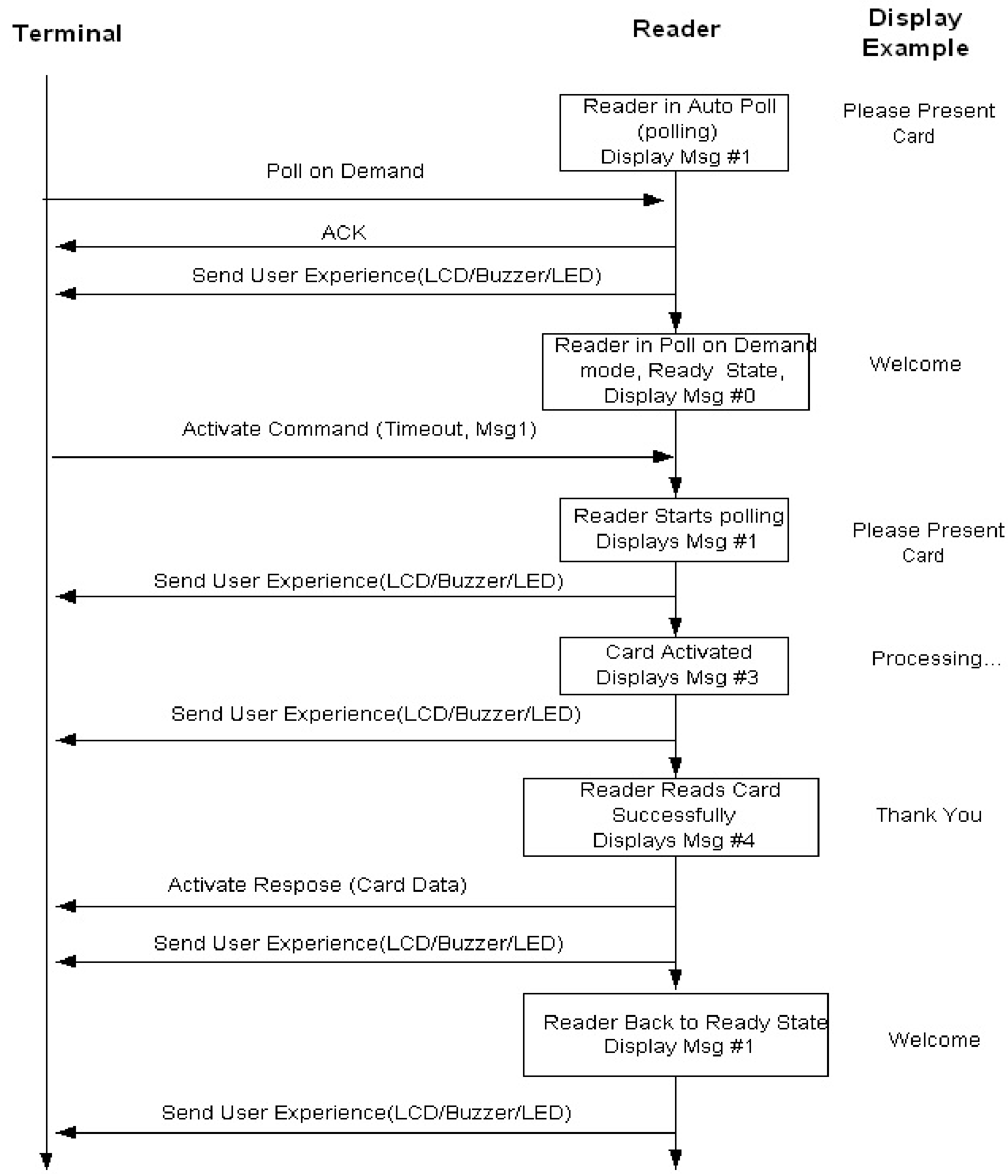


Figure 1a: Poll On Demand: Reader Reads Card Successfully

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

<table>
<tr><th>Card/ Interface Type</th><th>Tone</th><th>Reason for Tone</th></tr>
<tr><td rowspan="4">Contactless</td><td>Check Phone</td><td>• Consumer interaction required (user needs to do something on the phone, such as enter a PIN)</td></tr>
<tr><td>Alert</td><td>• Card Read Error
• Collision Detected
• Unsupported Card
• Application Error
• No Response after Restart (only for VCPS2.1.1 and ExpressPay 3.0)</td></tr>
<tr><td>Card Read Complete + Alert</td><td>• Card Read Complete and Card Error
• Card Read Complete & switch to another interface</td></tr>
<tr><td>Card Read Complete</td><td>• Transaction Approved, Offline
• Transaction Declined Offline (see status and error codes)
• Transaction Approved Online
• Transaction Declined Online</td></tr>
<tr><td rowspan="2">Contact</td><td>Alert</td><td>• Card Removed
• Transaction terminated with Error
• Switch from Contact to MagStripe Interface</td></tr>
<tr><td>Card Read Complete</td><td>• Transaction Approved, Offline
• Transaction Declined, Offline
• Transaction Approved, Online
• Transaction Declined, Online</td></tr>
<tr><td rowspan="2">MagStripe</td><td>Alert</td><td>• Card Swipe Error</td></tr>
<tr><td>Card Read Complete</td><td>• Card Swiped Successfully</td></tr>
</table>

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

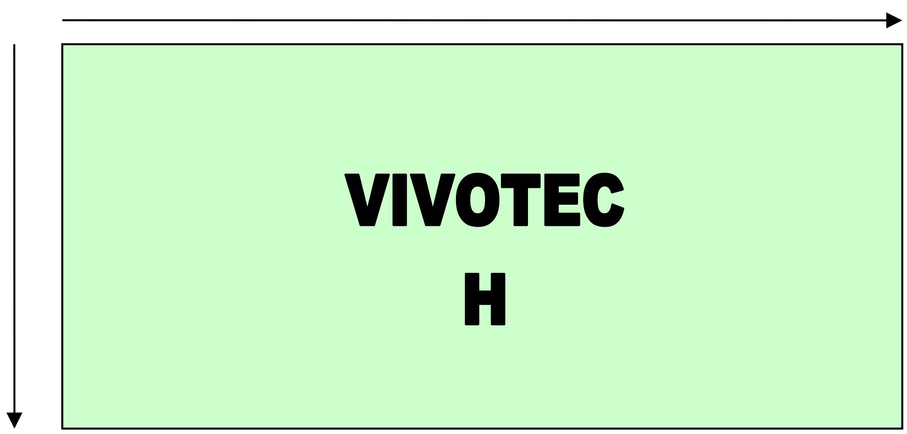
Compared to a regular monochrome bitmap, the image used with ILM commands has inverted color. White areas of the bitmap must be black and black areas white. To invert the color, each bit of a bitmap image must be reversed by performing a NOT operation on each byte of image data.

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.

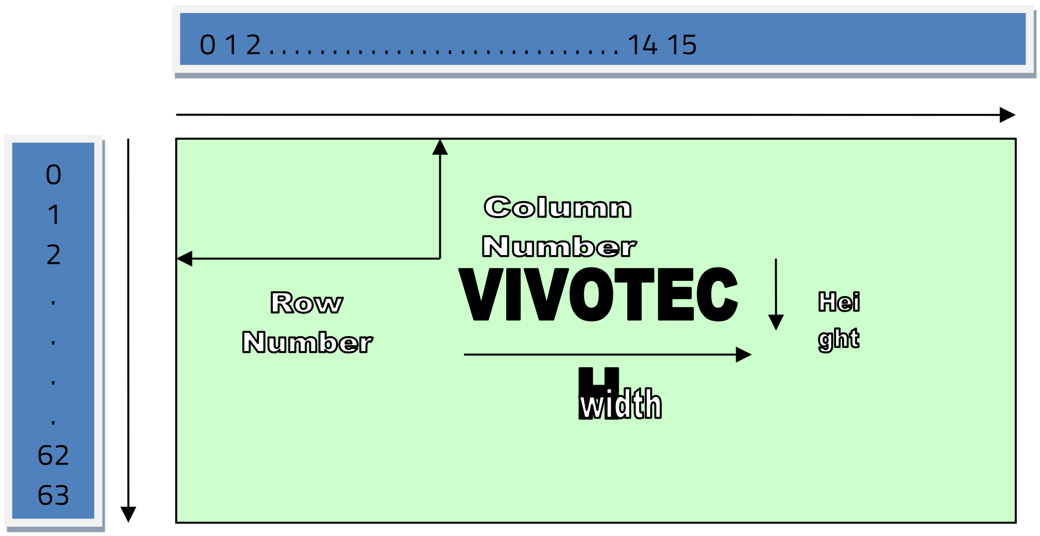
For example, here is a 128x64 pixel bitmap:



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. The 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 Bust 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

The 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 ViVOpay unit. 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 unit 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:	" Remove Card "	"PIN EntryRequire"	" Remove Card "
		L2:	" Retirez la Carte"	" Code exigé "	" Retirez la Carte"
0x0D			**APPROVED**	**AVAIL_OFFLINE_AMOUNT**	**APPROVED**
	ENG	L1:	" Approved "	" Offline Amount "	" Approved "
		L2:	" "	" "	" "
	FRA	L1:	" Approuvé "	" Montant "	" Approuvé "
		L2:	" "	" hors ligne "	" "
	ENG&FRA	L1:	" Approved "	" Offline Amount "	" Approved "
		L2:	" Approuvé "	" Mt hors ligne "	" Approuvé "
0x0E			**NOT_AUTHORIZED**	**ENTER_PIN**	**DECLINED**
	ENG	L1:	" Not "	" Please "	" Declined "
		L2:	" Authorized "	" Enter PIN "	" "
	FRA	L1:	" Non "	"Entrez votre "	" Refusé "
		L2:	" autorisé "	" code "	" "
	ENG&FRA	L1:	" Not Authorized "	"PIN EntryRequire"	" Declined "
		L2:	" Non autorisé "	" Code exigé "	" Refusé "
0x0F			**DECLINED**	**SIGNATURE_REQUIRED**	Reserved MSG16
	ENG	L1:	" Declined "	" Signature "	" "
		L2:	" "	" Required "	" "
	FRA	L1:	" Refusé "	" Signature "	" "
		L2:	" "	" Requise "	" "
	ENG&FRA	L1:	" Declined "	"SignatureRequire"	" "
		L2:	" Refusé "	"SignatureRequise"	" "
0x10			**TERMINATED**	**Reserved MSG17**	**Reserved MSG17**
	ENG	L1:	" Terminated "	" "	" "
		L2:	" "	" "	" "
	FRA	L1:	" Terminé "	" "	" "
		L2:	" "	" "	" "
	ENG&FRA	L1:	" Cannot Process "	" "	" "
		L2:	" Ne Peut Procés "	" "	" "
0x11			**TRY_OTHER_INTERFACE**	**TAP_OR_SWIPE_CARD_VISA**	**TRY_OTHER_INTERFACE**
	ENG	L1:	" Try Other "	" Present Card "	" Try Other "
		L2:	" Interface "	" "	" Interface "
	FRA	L1:	"Autre Interface "	" Présentez"	"Autre Interface "
		L2:	" "	"votre carte"	" "
	ENG&FRA	L1:	"AnotherInterface"	" Purchase/Achat "	"AnotherInterface"
		L2:	"Autre Interface "	" "	"Autre Interface "
0x12			**USE_OTHER_CARD**	**REMOVE_CARD**	**USE_OTHER_CARD**
	ENG	L1:	" Use Other Card "	" Please "	" Use Other Card "
		L2:	" "	" Remove Card "	" "
	FRA	L1:	" Use Other Card "	" Retirez la Carte "	" Use Other Card "
		L2:	" "		" "
	ENG&FRA	L1:	" Use Other Card "	" Remove Card "	" Use Other Card "
		L2:	" "	" Retirez la Carte"	" "
0x13			**TIMEOUT**	**PROCESSING**	**TIMEOUT**
	ENG	L1:	" Time Out "	" Processing "	" Time Out "
		L2:	" "	" "	" "
	FRA	L1:	" Pause "	" En cours "	" Pause "
		L2:	" "	" "	" "

	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 Procés "	" 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 recu " " "	" Pas de carte " " "	" Utilisez une " "autre carte VISA"
	ENG&FRA	L1: L2:	" Sign Receipt " " Signez le recu "	" 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 recu " " "
	ENG&FRA	L1: L2:	" Offline Amount " " Mt hors ligne "	" Try Again " " Ré-essayez "	" Sign Receipt " " Signez le recu "
0x2C			**SIGNATURE_REQUIRED**	**NOT_AUTHORIZED**	**ENTER_PIN**
	ENG	L1: L2:	" Signature " " Required "	" Not " " Authorized "	" Please " " Enter PIN "
	FRA	L1: L2:	" Signature " " Requise "	" Non " " autorisé "	"Entrez votre " " code "
	ENG&FRA	L1: L2:	"SignatureRequire" "SignatureRequise"	" 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 " " Requise "
	ENG&FRA	L1: L2:	" " " "	"AnotherInterface" "Autre Interface "	"SignatureRequire" "SignatureRequise"
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_REQUIRE**	**TERMINATE_PIN_REQUIRE**	**TERMINATE_PIN_REQUIRE**
	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)

Host can send the device 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 Passthrough 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 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	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)
ViVOtech2\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)
ViVOtech2\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)
ViVOtech2\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)
ViVOtech2\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:

31311C3931383B353135303731303230303130373936303D303930393130313430303030303335353F4511

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: 5669564f7465636832007003000 0c27f

IN:

5669564f74656368320070000049064944205445434820536d61727450494e204c3130
30204669726d77617265205631 2e30322e3030392e530d4d4158494d20436869702046
6f72204b3231204c3130302056312e30301 4ab

19. Appendix A.7: VP6800 Commands

A number of commands, outlined below, apply only to the VP6800.

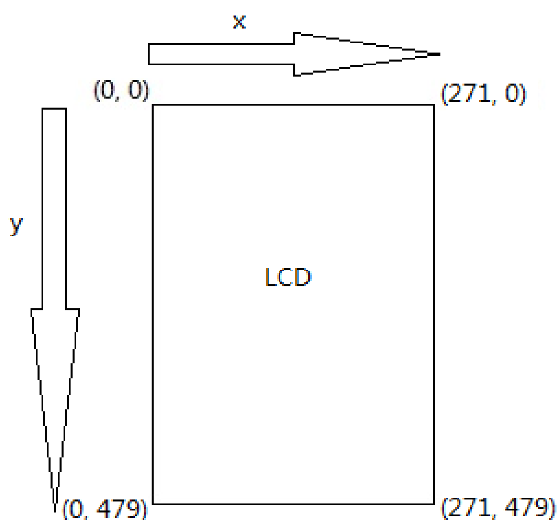
19.1. VP6800 Display Elements

- **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 0: Text Area 1: Large Button 2: Medium Button 3: Invisible Button (70px * 60px) 4: Numeric Entry (Cannot be added by user, *Note 1) 5: Ethernet Setting (Cannot be added by user, *Note 2) 6: LED Widget 7: Long Press (Cannot be added by user ,*Note 3) 8: Image. Only JPG file is supported 9. Video FF: Nothing
Byte 1	Object Number The object number for Numeric Entry is Special: 00: Enter of numeric entry is clicked 01: Cancel of numeric entry is clicked The object number for Long Press is Special: 00: top-left corner is long pressed 01: top-right corner is long pressed FF: Nothing

- ***Note 1:** Numeric entry will 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 device 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 device to send a Long Press event with a Long Press Object ID.

19.3. VP6800 UI Commands

19.3.1. Show Screen (61-05)

This command specifies the screen to display as live on the device.

Note: the Screen ID **MUST** be created before sending this command. Users can also add or update objects for the Screen, which is optional for the **Show Screen** 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.

19.3.1.1. Showing Predefined Screens

Predefined screens are hard coded in VP6800 firmware; users should not create these screens and cannot add or remove components to or from them.

Screen names are defined as follows:

Screen Name	Appearance
PreDefScreen_8100	

Screen Name	Appearance
PreDefScreen_8101	
PreDefScreen_8102 (IP Address) **PreDefScreen_8103** (Subnet Mask) **PreDefScreen_8104** (Default Gateway)	
PreDefScreen_8105	

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 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 item text area and leds widgt, 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: • Text area is 64bytes, • Large button is 25bytes • Medium button is 10bytes) • Filename of image/video (ASCII string, terminated by 0x00, the maximum string lens is 32bytes)

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 0xFFFFFF00, 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	

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
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	20
Large Button	8
Medium Button	16
Invisible Button	16
Led widget	1
Image	20

Video	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 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

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 below table

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. Object Event Call Back (61-FF)

The VP6800 immediately sends this command when a button on-click event happens.

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

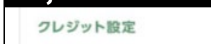
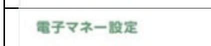
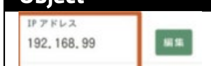
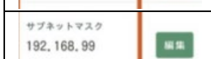
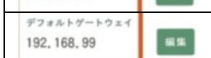
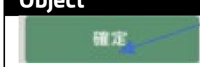
Item	Length (byte)	Description
Screen ID	2	Screen ID format; see LCD UI Table-1
Object ID	2	Object ID format; see LCD UI Table-2
Screen Name	Var.	Screen Name String No longer than 32 bytes ASCII string, terminated by 0x00
Object Name	Var.	Object Name String

		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

Note:

1. The button callback will be sent by previously communicated I/F.
2. Only buttons and numeric entries have call back events.

The table below describes detailed event information for predefined screens:

Screen ID	Screen Name	Object	Object ID	Object Name	Parameter
0x8100	PreDefScreen_8100		0x0001	Item1	N/A
			0x0002	Item2	N/A
			0x0003	Item3	N/A
			0x0004	Item4	N/A
			0x0005	Item5	N/A
			0x0000	Cancel	N/A
0x8101	PreDefScreen_8101		0x0001	Item1	N/A
			0x0002	Item2	N/A
			0x0003	Item3	N/A
			0x0000	Cancel	N/A
0x8102	PreDefScreen_8102		0x0000	OK	String of entered info
0x8103	PreDefScreen_8103		0x0001	Cancel	N/A
0x8104	PreDefScreen_8104				
0x8105	PreDefScreen_8105		0x0000	OK	String of entered info
			0x0001	Cancel	N/A

19.3.6. 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

19.3.7. 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 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

19.3.8. 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)
ViVOtech2\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)
ViVOtech2\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)

19.3.9. Get Current Screen Name (61-0C)

This command allows a user to check the currently live screen's name.

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 a default one, an error report would be 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 3E 43 75 72 72 65 6E 74 20 73 63 72 65 65 6E 20 69 73 20 6E 6F 74 20 75 73 65 72 2D 64 65 66 69 6E 65 64 2C 20 67 65 74 20 73 63 72 65 65 6E 20 69 64 20 69 73 20 72 65 6A 65 63 74 65 64 2E 08 3A (Current screen is not user-defined, get screen ID is rejected)

19.3.10. 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`

19.3.11. 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

19.3.12. 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

19.3.13. Clear Screen Info (61-10)

This command clears all current screen information in RAM and flash, then diplays 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

19.3.14. 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 (LSB)	CRC (MSB)
ViVOtech2\0	61h	00h			See below		

Data Format

<table>
<tr><th>Item</th><th>Length(byte)</th><th>Description</th></tr>
<tr><td>Number of Screens</td><td>1</td><td>Number of screens</td></tr>
<tr><td>Screen List</td><td>var.</td><td>List screens, each screen is shown as:
<table>
<tr><td>Screen ID</td><td>See Table-1</td></tr>
<tr><td>Screen Name String</td><td>Screen name string
ASCII string, each screen name string terminated by 0x00</td></tr>
</table></td></tr>
</table>

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 00 00~00 03

19.3.15. 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 (LSB)	CRC (MSB)
ViVOtech2\0	61h	00h			See Below		

Data Format

<table>
<tr><th>Item</th><th>Length(bytes)</th><th>Description</th></tr>
<tr><td>Number of Objects</td><td>1</td><td>Number of objects</td></tr>
<tr><td>Objects List</td><td>var.</td><td>List objects, each object is shown as:
<table>
<tr><th>Object ID</th><th>See Table-2</th></tr>
<tr><td>Object Name String</td><td>Object name string, ASCII string, each object name string is terminated by 0x00</td></tr>
</table></td></tr>
</table>

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

19.3.16. 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 (LSB)	CRC (MSB)
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

19.3.17. 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 (LSB)	CRC (MSB)
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

19.3.18. 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 (LSB)	CRC (MSB)
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

19.3.19. 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 (LSB)	CRC (MSB)
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

19.3.20. 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 Codes 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

19.3.21. 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		

19.3.22. Set Backlight (83-2F)

This command set's 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 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	83h	2Fh	00	01	Backlight percent		

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 01 64 93 FB
  RX: 56 69 56 4F 74 65 63 68 32 00 83 00 00 00 22 03

19.3.23. 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 Codes 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`

19.3.24. Transfer File (83-24)

This command transfers a data file to the reader. The maximum size of one file is 128KB; the maximum number of image files is 64.

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 Codes 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 61 62 32 B3`
  RX: `56 69 56 4F 74 65 63 68 32 00 83 00 00 00 0A 70`

19.3.25. 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 Codes 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:
```
d/,afile.txt:35,bfile.txt:38222,cfile.txt:38241
```

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
```

The response is a single file (testfile.x) of length 40 bytes:
```
56 69 56 4F 74 65 63 68 32 00 83 00 00 0E 74 65 73 74 66 69 6C 65 2E
78 3A 34 30 00 22 79
```

19.3.26. 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 Codes 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

19.3.27. 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 Codes 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

19.3.28. 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 Codes Table	00h	01	1050 current mode (see below)		

Data Format

Item	Length(byte)	Description
1050 current mode	1	0: None Device Mode 1: Restricted Mode 2: Self Check Mode 3: Customer Mode 4: System Mode 5: Boot Loader Mode

Example:

- Get RT1050 Status:
  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.

19.3.29. 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 Codes 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

19.3.30. 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 Codes Table	00h	06	MAC Address		

Data Format

Item	Length(byte)	Description
Interface	1	0x30: Ethernet

Example:

- Get MAC Address:
  TX: 56 69 56 4F 74 65 63 68 32 00 F0 02 00 01 00 2D 2A
  RX: 56 69 56 4F 74 65 63 68 32 00 F0 00 00 02 A0 0B 19 EE

19.3.31. VP6800 Specific TS Command (90-13)

This command performs specific TS commands on the VP6800.

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	90h	13h	00h	Variable	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 (LSB)	CRC (MSB)
ViVOtech2/0	93h	See Status Code Table	00h	00h		

Data Format

Test Item	Length(byte)	Test Command Data Field Description
Touch panel	1	Data[0]: 0Fh
USB mass storage	1	Data[0]: 0Ch
Audio	1	Data[0]: 0Ah
SD Card	1	Data[0]: 0Bh
LCD	2	Data[0]: 01h Data[1]: 02h
Bluetooth	2	Data[0]: 09h Data[1]: 00h
Wi-Fi	2	Data[0]: 10h Data[1]: 00h
Camera	1	Data[0]: 11h

19.3.32. 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	Byte14	Byte 15	Byte 16
Header Tag & Protocol	Command	Status	Length MSB	Length LSB	Data	CRC (LSB)	CRC (MSB)
ViVOtech2\0	4Ch	See Status Codes Table	00h	00h	See Below		

Data Format

Note: the following function codes enable or disable the log function or the error detail log function. Device 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 device
Disable adding detail after error code	1	20h	

19.3.33. Start Removal Detection Activation Sequence (61-20)

This command only works on production devices 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 Codes Table	00h	00		

19.3.34. Reserved for VP6800 Use (61-21)

Command 61-21 is reserved and used to distinguish between different customizations.

19.3.35. 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 device 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 (LSB)	CRC (MSB)
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)`

19.3.36. Stop Audio (61-30)

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	61h	30h	00h	00h		

19.3.37. Play Audio (61-31)

This 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

19.3.38. 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 Item	Length (bytes)	Description
Time Out	0 or 2	Timeout in seconds. • 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)
ViVOpayV3\0	61h	See Status Code Table					

This command uses the secondary response mechanism. The device 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.

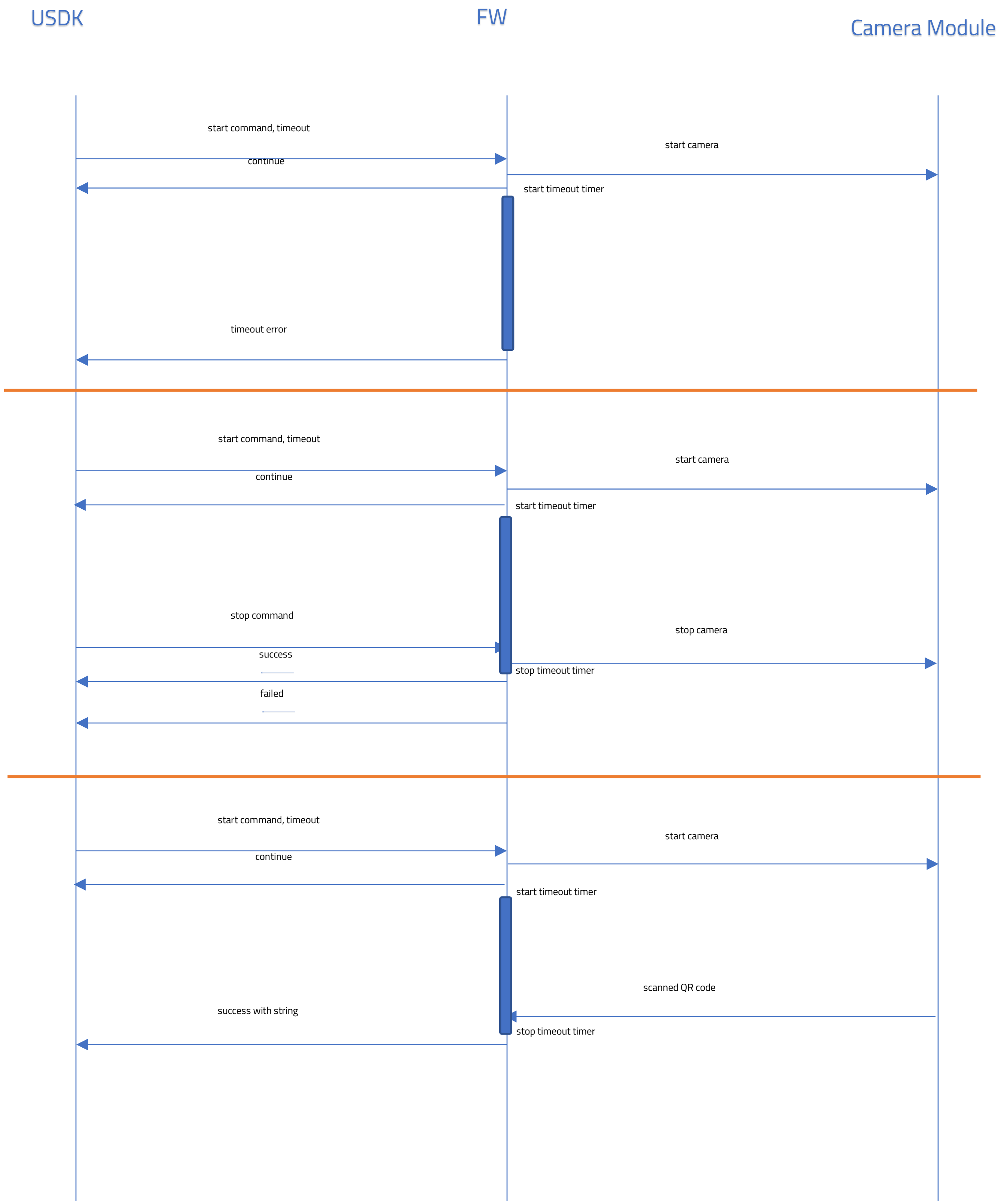
During the operation (that is, the interval between the first and second response), the device only accepts the **QR Code Scan Stop** command, which will cancel the current **QR Code Scan Start** operation.

The second response is the scanned QR code's data after the device completes reading it.

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 00 00 07 31 32 33 34 35 0A 0A E1 8B` – Second response

19.3.38.1. QR Code Reading Use Cases



19.3.39. 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		

19.3.40. Camera Take Photo Start (61-43)

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	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 use the secondary response mechanism. The device 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 device only accepts the **Camera Take Photo Stop** command, which will cancel the current **Camera Take Photo Start** operation.

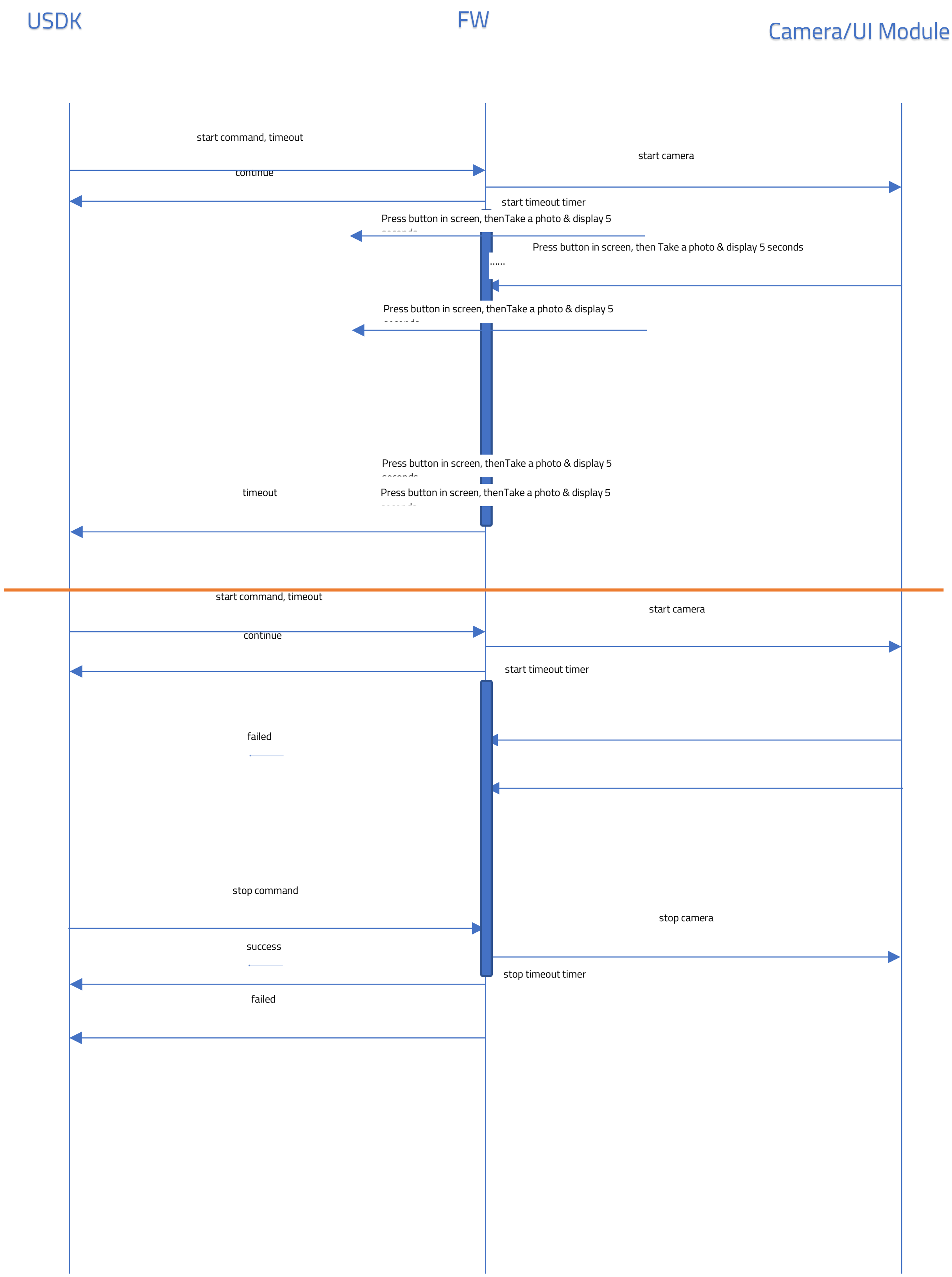
The device rejects the **Camera Take Photo Start** command if an SD card does not exist.

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 device takes a picture, which is stored in SD card and displayed on-screen for five seconds. The device stores a maximum of 20 pictures. Upon reaching capacity, the 21st picture replaces the 1st.

19.3.40.1. Use Cases for Taking Photos with the Camera



19.3.41. Camera Take Photo Stop (61-44)

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	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		

19.4. 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 device 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 device 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	

Symbol	Value	Note
&	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

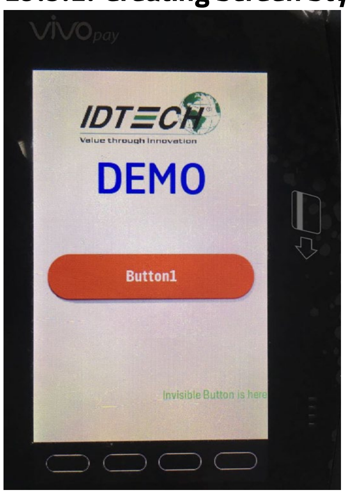
Status Code	Status
00h	OK
01h	Incorrect Header Tag
02h	Unknown Command
03h	Unknown Sub-Command
04h	CRC Error in Frame
05h	Incorrect Parameter
06h	Parameter Not Supported
07h	Mal-formatted Data
08h	Timeout
0Ah	Failed / NACK
0Bh	Command not Allowed
0Ch	Sub-Command not Allowed
0Dh	Buffer Overflow (Data 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.

Status Code	Status
	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

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 device 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 device 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
    ```

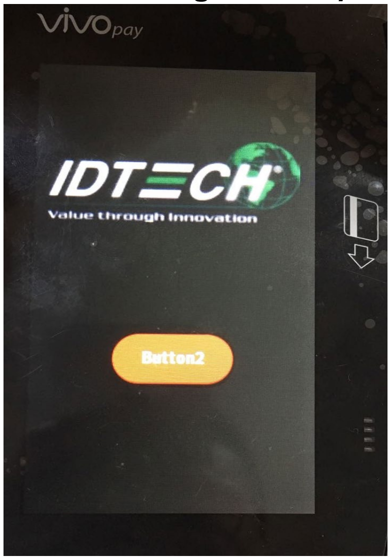
11. Publish and display Screen1 with the **Show Screen (61-05)** command.
    Command:
    ```
    56 69 56 4F 74 65 63 68 32 00 61 05 00 08 53 63 72 65 65 6E 31 00
    91 5C
    ```
    Response:
    ```
    56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81
    ```

12. Store all current screen information from RAM into flash with the **Store Screen Info (61-0E)** command to ensure the device does not lose any data after a reboot.

    Note: If the device 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 "Sceen2" result)
Object ID	02 00	Medium Button(if you do not know, refer 2.2.3 Add Medium Button To "Sceen2" 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 device does not lose any data after a reboot.

   Note: If the device 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.

19.5.3.1. 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.

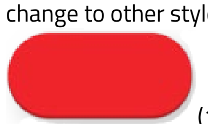
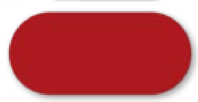
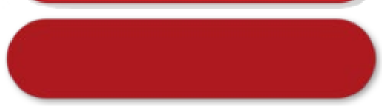
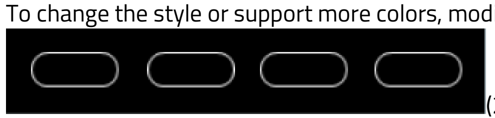
Command:
`56 69 56 4F 74 65 63 68 32 00 61 0F 00 00 B0 67`
Response:
`56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81`

19.5.3.2. Load the System UI

Use the Display System Screen (61-17) command to hide the customer-facing UI and show the system UI.

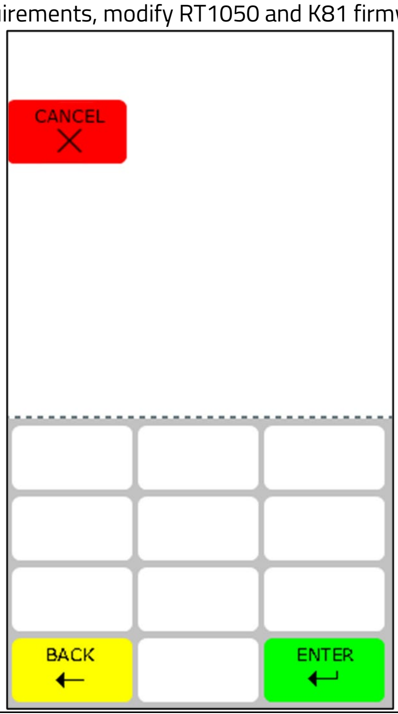
Command:
`56 69 56 4F 74 65 63 68 32 00 61 17 00 00 72 8D`
Response:
`56 69 56 4F 74 65 63 68 32 00 61 00 00 00 4B 81`

19.6. VP6800 UI Limitations

Supported Customer-Used Components	Limitation
Screen	• **Maximum number** of text components in each screen is 50. • **Maximum characters** of a screen name is 32 bytes.
Object	• **Maximum characters** of an object name is 32 bytes
Text	• 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	• Current supported **style and size** is fixed to the following image. To support or change to other styles, modify RT1050 firmware.  (120px*60px unpressed)  (120px*60px pressed) • **Maximum number** of medium button components in each screen is 16. • **Maximum characters** in each medium button component is 10 bytes.
Large Button	• Current supported **style and size** is fixed to the following image. To support or change to other styles, modify RT1050 firmware.  (250px*60px unpressed)  (250px*60px pressed) • **Maximum number** of large button components in each screen is 16. • **Maximum characters** in each large button component is 25 bytes.
Invisible Button	• 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	• 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) • **Maximum number** of LED Widget components in each screen is 1.
Image	• Only images stored in flash can be loaded to a screen.

Supported Customer-Used Components	Limitation
	• **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	• 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: • 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	Style and button position is fixed to the following image. To support special requirements, modify RT1050 and K81 firmware. 
Text	Font is fixed to: • RoundCondensedBold_24 **(English, French)** • STZhongsong_24 **(Chinese)** • YuMincho_24 **(Japanese)** To support special requirements, modify RT1050 firmware.

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
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	
¡	0xA1	
?	0x3F	
¿	0xBF	
*	0x2A	
+	0x2B	
§	0xA7	
¢	0xA2	
$	0x24	
▪	0xB7	
/	0x2F	
\	0x5C	
#	0x23	
-	0x2D	
%	0x25	
½	0xBD	
¼	0xBC	

20. Appendix A.8: White List Commands for VP3300 Family

For historical reasons, the VP3300 family (which includes the UniPay 1.5, UniPay III, and BTPay Mini, in addition to VP3300 series and VP8300 products) has its own White List commands, distinct from the 2C-50 command (and related commands) used in other NEO-firmware products. The relevant commands are discussed below.

Typically, these commands govern the management of white-listed magstripe gift cards (and other non-financial cards). The purpose of white listing is to allow *non-financial* card data to be processed in the clear (without encryption). On non-SRED devices (such as the VP3300 family of devices), these commands do not need to be authenticated. On a PCI-validated SRED product, such commands would be subject to authentication using PKI techniques and/or MAC hashing. The commands shown below do not require that a MAC be used, because VP3300 devices are not SRED-validated.

At transaction time, the reader will check the BIN range of a card (the first 8 digits of the PAN) against the white list to determine if the card can safely be considered a non-financial card. If so, the transaction proceeds in the clear. If not, normal encryption policy prevails.

Because most white-list cards are gift cards, and most gift cards are magnetic stripe cards (rather than ICC/EMV cards), command names shown below have "MSR" in them.

20.1.1. Set MSR White List (91-10)

This command can be used to set the MSR White 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	90h	10h			White List Format		

White List Format:

Data Item	Length (bytes)	Description
Command Data Length	2	<Length Whitelist ASN.1 BLK> is 2 bytes, format is LenL LenH, it is length of <Whitelist ASN.1 BLK>, less than 512
Command Data	Var	<Whitelist ASN.1 BLK> is N bytes, as an ASN.1 Block of data for White List for Gift Card. Format: Whitelist ASN.1 structure::= Sequence { Whitelist ASN.1 structure version = 1 (INTEGER)

Data Item	Length (bytes)	Description

		BIN Exclusion::= Set
		{
		bininfo::= Sequence
		{
		binSetName = (PRINTABLESTRING) -- 0 if no name
		LowBoundinfo::= Sequence
		{
		First 8 digits= (INTEGER) – bin
		Last 8 digits=(INTEGER) – bin
		}
		HighBoundinfo::= Sequence
		{
		First 8 digits= (INTEGER) – bin
		Last 8 digits=(INTEGER) – bin
		}
		}
		}
		}

Sign1-Device Length	2	For Non-PCI device, Not Exist keep: 0x00 0x00
Sign1-Device	256	

If a card's PAN is in the white list range, then output response will be in plaintext even when encryption is enabled. Take care not to include bank cards in the white-listable range.

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	90h	See Status Code Table	00	00		

Example

```
56 69 56 4F 74 65 63 68 32 00 91 10 00 8E 8A 00 30 81 87 02 01 01 31 81 81 30
29 13 0B 49 44 54 45 43 48 20 54 45 53 54 30 0C 02 04 05 F5 E0 FF 02 04 05 F5
B9 F0 30 0C 02 04 05 F5 E0 FF 02 04 05 F5 E0 FF 30 29 13 0B 49 44 54 45 43 48
20 54 45 53 54 30 0C 02 04 05 4C 56 38 02 04 05 4C 33 80 30 0C 02 04 05 4C 56
38 02 04 05 4C 56 38 30 29 13 0B 49 44 54 45 43 48 20 54 45 53 54 30 0C 02 04
02 9B E9 CD 02 04 02 69 B9 15 30 0C 02 04 02 9B E9 CD 02 04 02 69 B9 15 00 00
B1 C8
56 69 56 4F 74 65 63 68 32 00 91 00 00 00 D4 CC 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
```

Using the parsing tool at the following link to parse ASN.1 data: *ASN.1 JavaScript decoder* (see a complete example by following this link: https://goo.gl/Ujb4FE).

And we get:

```
SEQUENCE(2 elem)
        INTEGER1
        SET(3 elem)
                SEQUENCE(3 elem)
                        PrintableString         IDTECH TEST
                        SEQUENCE(2 elem)
                                INTEGER  99999999               -- first 8 digits of low bound
                                INTEGER  99990000               -- last 8 digits of low bound
                        SEQUENCE(2 elem)
                                INTEGER  99999999               -- first 8 digits of high bound
                                INTEGER  99999999               -- last 8 digits of high bound
                SEQUENCE(3 elem)
                        PrintableStringIDTECH TEST
                        SEQUENCE(2 elem)
                                INTEGER  88888888               -- first 8 digits of low bound
                                INTEGER  88880000               -- last 8 digits of low bound
                        SEQUENCE(2 elem)
                                INTEGER  88888888               -- first 8 digits of high bound
                                INTEGER  88888888               -- last 8 digits of high bound
                SEQUENCE(3 elem)
                        PrintableStringIDTECH TEST
                        SEQUENCE(2 elem)
                                INTEGER  43772365               -- first 8 digits of low bound
                                INTEGER  40483093               -- last 8 digits of low bound
                        SEQUENCE(2 elem)
                                INTEGER  43772365               -- first 8 digits of high bound
                                INTEGER  40483093               -- last 8 digits of high bound
```

If card PAN is between 9999 9999 9999 9999 to 9999 9999 9999 0000, 8888 8888 8888 8888 to 8888 8888 8888 0000, or 4377 2365 4048 3093 to 4377 2365 4048 3093, the response would keep in plaintext.

20.1.2. Clear MSR White List (91-11)

This command used to clear MSR White 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 (MSB)	CRC (LSB)
ViVOtech2\0	91h	11h				

Digital Signature x2: SMFG Clear White Listfor MSR data

Data Item	Length (bytes)	Description
Sign1-Device Length	2	Length of Sign1-Device: 0x00
Sign1-Device	256	

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	91h	See Status Code Table	00h	00h		

20.1.3. Retrieve Get MSR Whitelist (C7-A2)

This command used to set MSR White 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	C7h	A2h				

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 (LSB)	CRC (MSB)
ViVOtech2\0	C7h	00h			White List Format		

White List Fomat:

Data Item	Length (bytes)	Description
Command Data Length	2	<Length Whitelist ASN.1 BLK> is 2 bytes, format is LenL LenH, it is length of <Whitelist ASN.1 BLK>, less than 512

Command Data	Var	<Whitelist ASN.1 BLK> is N bytes, it is ASN.1 Block data for White List for Gift Card.
		Format:
		Whitelist ASN.1 structure::= Sequence
		{
		Whitelist ASN.1 structure version = 1 (INTEGER)
		BIN Exclusion::= Set
		{
		bininfo::= Sequence
		{
		binSetName = (PRINTABLESTRING) -- 0 if no name
		binLow = (INTEGER) -- bin >= binLow
		binHigh = (INTEGER) -- bin <= binHigh
		}
		}
		}

Sign1-Device Length	2	For Non-PCI device, Not Exist keep: 0x00 0x00
Sign1-Device	256	

21. 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
9C	Transaction Type	n2	1	00
5F2A	Transaction Currency Code	n3	2	08 40
5F36	Transaction Currency Exponent	n1	1	02
9F02	Amount, Authorised (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	Terminal Country Code	n2	2	08 40
9F40	Additional Terminal Capabilities	b	5	60 00 00 30 00
9F53	Terminal Interchange Profile (dynamic)	an	1	00
9F66	PUNATC (Track2)	b	4	30 00 40 00
9F6D	Kernel 4 Reader Capabilities	b	2	00 01
9F7C	Merchant Custom Data	b	20	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
DFEE28	Terminal Capabilities - No CVM Required	b	3	00 08 E8
DFEE29	Terminal Capabilities - CVM Required	b	3	00 68 E8
DFEE34	Terminal 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
DF8122	Terminal Action Code - Online	b	5	F8 50 AC F8 00
DF8120	Terminal Action Code - Default	b	5	F8 50 AC A0 00
DF8121	Terminal Action Code - Denial	b	5	00 00 00 00 00
DFEE44	Application Capability	b	2	95 FF
DFEE5C	RF Deactivate Period	b	4	01 50 00 00
DFEE6B	Terminal 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

Tag	Description	Format	Length	Default Value
DFEE73	Merchant Category Code	b	2	00 00
DFEE74	Discover Optional Features	b	1	00
DFED18	Poll Mode	b	1	01
DFEE7E	Burst Mode	b	1	00
DFEE1C	LCD Font Size	b	1	03
DFEF29	LCD Delay Time	b	2	00 00
9F4E	Merchant Name and Location	ans	30	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: DFEE38 can have the following values.

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
Value = 06 Japanese

Table 85: NEOII Group 80 (Master) default tags

Tag	Description	Format	Length	Default Value
DFEE2D	Group Number / Fallback Group	n2	1	80
5F57	Account type	B	1	
9F01	Acquirer Identifier	n12	6	
9F40	Additional Terminal Capabilities	b	5	00 00 00 00 00
9F09	Application Version Number	b	2	00 02
DF8117	Card Data Input Capability	b	1	00
DF8118	Terminal Capabilities - CVM Req.	b	1	60
DF8119	Terminal Capabilities - No CVM Required	b	1	08
DF811A	Default UDOL	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	
9F1E	Terminal IFD	an	8	30 30 30 30 30 30 30 30

Tag	Description	Format	Length	Default Value
DF811B	Kernel Configuration	b	1	20
9F6D	Magstripe Application version number	b	2	00 01
DF811E	Magstripe CVM Capability CVM Required	b	1	10
DF812C	Magstripe CVM Capability No CVM Required	b	1	00
DF811C	Max Lifetime of Torn Transaction Log Record	b	2	00 00
DF811D	Max Number of Torn Transaction Log Records	b	1	00
9F15	Merchant Category Code	n4	2	11 11
9F16	Merchant Identifier	ans	15	
9F4E	Merchant Name & Location	ans	<=30	
9F7E	Mobile Support Indicator	b	1	
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
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
9F33	Terminal Capabilities	b	3	
9F1A	Terminal Country Code	n3	2	08 40
9F1C	Terminal ID	an	8	
9F1D	Terminal Risk Management	b	8	6C FF 00 00 00 00 00 00
9F35	Terminal Type	b	1	22
DFEE37	UI Scheme	b	1	03
DFEE38	LCD Language	b	1	00
DFEE0F	Revocation List	b	1	01

Table 86: NEOII Group 90 (Visa) default tags

Tag	Description	Format	Length	Default Value
DFEE2D	Group Number / Fallback Group	n2	1	90
9F02	Amount, Authorised (Numeric)	n12	6	00 00 00 00 01 00
9F03	Amount, Other (Numeric)	n12	6	00 00 00 00 00 00
9F1A	Terminal Country Code	n3	2	08 40
9F1B	Terminal Floor Limit	b	4	00 00 17 70
9F33	Terminal Capabilities	b	3	00 08 E8
9F35	Terminal Type	n2	1	25
9F66	Terminal Transaction Qualifier (TTQ)	b	4	30 00 40 00
5F2A	Transaction Currency Code	n3	2	08 40
5F36	Transaction Currency Exponent	n1	1	02
9C	Transaction Type	n2	1	00
DF8122	Terminal Action Code - Online	b	5	F8 50 AC F8 00
DF8120	Terminal Action Code - Default	b	5	F8 50 AC A0 00
DF8121	Terminal Action Code - Denial	b	5	F8 50 AC F8 00
DF8126	CVM Required Limit	b	6	00 00 00 00 80 00

Tag	Description	Format	Length	Default Value
DFEE34	Terminal 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
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: NEOII Group A0 (Amex) default tags

Tag	Description	Format	Length	Default Value
DFEE2D	Group Number / Fallback Group	n2	1	A0
9C	Transaction Type	n2	1	00
5F2A	Transaction Currency Code	n3	2	08 40
5F36	Transaction Currency Exponent	n1	1	02
9F02	Amount, Authorised (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	Terminal Country Code	n3	2	08 40
9F33	Terminal Capabilities	b	3	00 08 E8
9F35	Terminal Type	n2	1	25
9F40	Additional Terminal Capabilities	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	Terminal Contactless Transaction Limit	n12	6	00 00 00 01 00 00
DFEE5C	RF Deactivate Period	b	4	01 50 00 00
DF8126	CVM Required Limit	b	6	00 00 00 00 80 00
DF8123	Reader Contactless Floor Limit	b	6	00 00 00 00 60 00
DF8122	Terminal Action Code - Online	b	5	F8 50 AC F8 00
DF8120	Terminal Action Code - Default	b	5	F8 50 AC A0 00
DF8121	Terminal Action Code - Denial	b	5	00 00 00 00 00

Table 88: NEOII Group B0 (Discover) default tags

Tag	Description	Format	Length	Default Value
DFEE2D	Group Number / Fallback Group	n2	1	B0
9F02	Amount, Authorised (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	Terminal Country Code	n3	2	08 40
9F1B	Terminal Floor Limit	b	4	00 00 17 70
9F33	Terminal Capabilities	b	3	00 08 E8

Tag	Description	Format	Length	Default Value
9F35	Terminal Type	n2	1	25
9F66	Terminal Transaction Qualifier (TTQ)	b	4	80 00 40 00
5F2A	Transaction Currency Code	n3	2	08 40
5F36	Transaction Currency Exponent	n1	1	02
9C	Transaction Type	n2	1	00
DF8122	Terminal Action Code - Online	b	5	00 00 00 00 00
DF8120	Terminal Action Code - Default	b	5	00 00 00 00 00
DF8121	Terminal Action Code - Denial	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

Table 89: NEOII Group C0 (Interac) default tags

Tag	Description	Format	Length	Default Value
DFEE2D	Group Number / Fallback Group	n2	1	C0
5F2A	Transaction Currently Code	n12	6	01 24
9F09	Application Version Number	b	2	00 02
9F1A	Terminal Country Code	n3	2	01 24
9F35	Terminal Type	n2	1	22
9F58	Merchant Indicator	n1	1	03
9F59	Terminal Transaction Information	b	3	DC 87 00
9F5A	Terminal Transaction Type	b	1	00
9F5D	Terminal Contactless Receipt Requirement Limit	n12	6	00 00 00 00 50 00
9F5E	Terminal Option Status	b	2	E0 00
9F5F	Terminal Contactless Floor Limit	n12	6	00 00 00 00 50 00
9F1B	Terminal Floor limit	b	4	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
DF8122	TAC - Online	b	5	00 00 00 00 00
DF8120	TAC - Default	b	5	00 00 00 00 00
DF8121	TAC - Denial	b	5	00 00 00 00 00
DFEE2B	Max Target% Based Random Selection	b	1	32
DFEE2C	Target% Random Selection	b	1	0A
DFEE2A	Threshold Value Biased Random Selection	n12	6	00 00 00 00 50 00
DFEF56	Retry Limit	b	1	03
9F02	Amount Authorized	n12	6	00 00 00 00 15 00
9F03	Amount Other	n12	6	00 00 00 00 00 00
5F36	Transaction Currency Exponent	n1	1	02
9C	Transaction Type	b	1	00
9F33	Terminal Capabilities	b	3	00 48 E8
DFEE37	UI Scheme	b	1	03

Table 90: NEOII Group D0 (JCB EMV) default tags

Tag	Description	Format	Length	Default Value
DFEE2D	Group Number / Fallback Group	n2	1	D0
9C	Transaction Type	n2	1	00
DFEF44	JCB Combination Option	b	2	7B 00
DF8123	Reader Contactless Floor Limit	b	6	00 00 00 00 45 00
DFEE34	Terminal Contactless Transaction Limit	b	6	00 00 00 02 00 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
DFEF45	JCB Removal Timeout (The time after which cardholder is asked to remove the card)	b	2	10 00
DF8120	Terminal Action Code - Default	b	5	90 40 00 80 00
DF8121	Terminal Action Code - Denial	b	5	04 10 00 00 00
DF8122	Terminal Action Code - Online	b	5	90 60 00 90 00
DFEF50	JCB Terminal Interchange Profile (static)	b	3	70 80 00
9F01	Acquirer Identifier	b	6	00 00 00 00 00 10
9F15	Merchant Category Code	b	2	70 32
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
9F1A	Terminal Country Code	n3	2	03 92
9F35	Terminal Type	n2	1	25
5F2A	Transaction Currency Code	n3	2	03 92
5F36	Transaction Currency Exponent	n1	1	02

22. Appendix A.10: NEO II Default System AIDs

The following table contains the default system AIDs for NEOII.

Table 91: NEOII 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 Dpas	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

23. Appendix A.11: NEO II Default Configuration TLVs

The following table contains the default configuration TLVs for NEO II.

Table 92: NEOII 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 List	DF EE 54	03	04 00 A0	
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 Processing	DF EE 53	01	02	
Transaction Type List	DF EE 54	0F	02 00 80 02 01 80 02 09 80 02 17 80 02 20 80	
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 Processing	DF EE 53	01	02	
Transaction Type List	DF EE 54	0F	02 00 80 02 01 80 02 09 80 02 17 80 02 20 80	
Kernel ID	DF EE 59	01	02	
Application Flow	DF EE 4C	01	03	

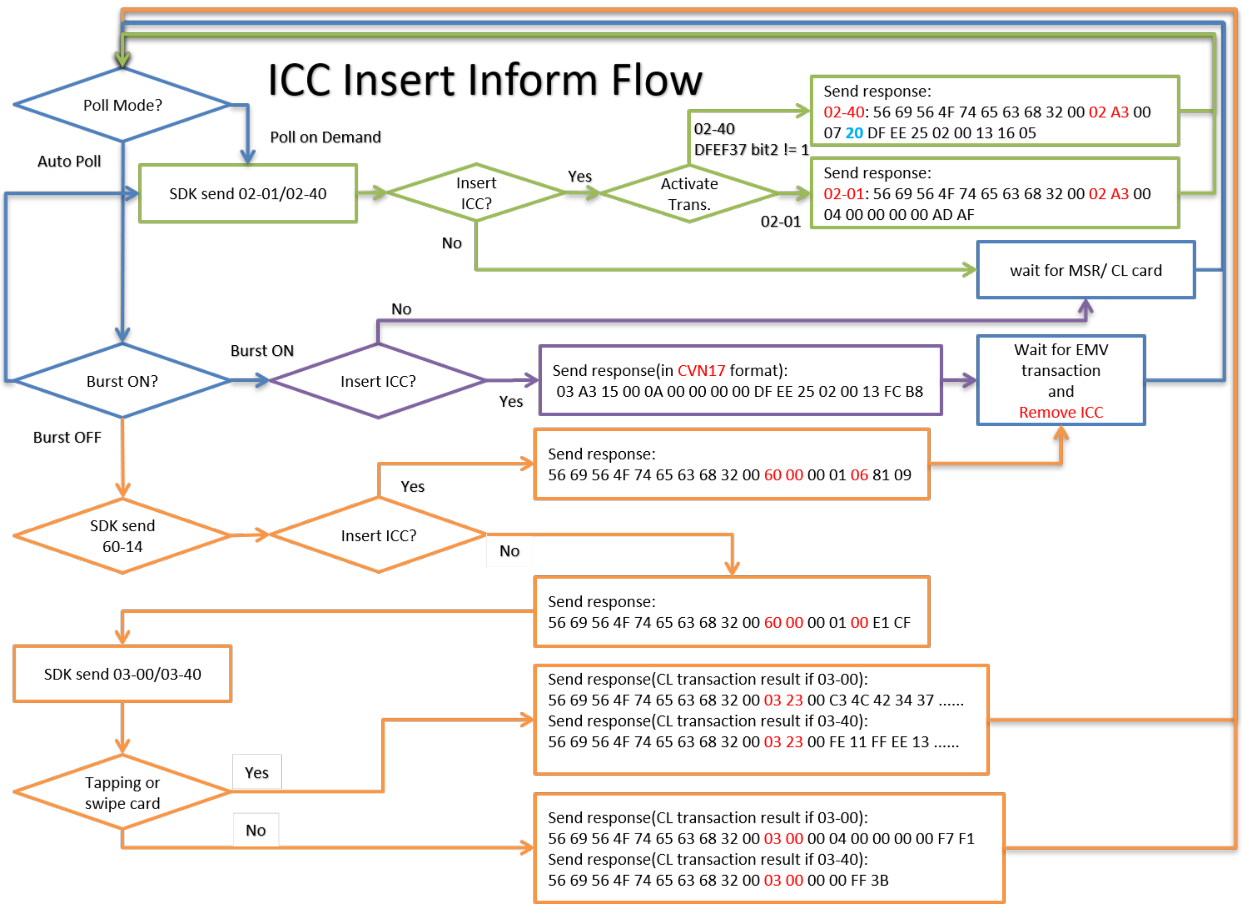
Name	Tag	Length (Hex)	Value (Hex)	Card Application
Group AID Partial Selection Selection Features Max AID Length Application Flow	DF EE 2D 9F 06 DF EE 4B DF EE 4D DF EE 2E DF EE 4C	01 07 01 01 01 01	90 A0 00 00 00 03 10 10 01 14 10 06	VISA Application
Group AID Partial Selection Selection Features Max AID Length Application Flow	DF EE 2D 9F 06 DF EE 4B DF EE 4D DF EE 2E DF EE 4C	01 07 01 01 01 01	90 A0 00 00 00 03 20 10 01 14 10 06	VISA Application (Visa Electron)
Group AID Partial Selection Selection Features Max AID Length Application Flow	DF EE 2D 9F 06 DF EE 4B DF EE 4D DF EE 2E DF EE 4C	01 07 01 01 01 01	90 A0 00 00 00 03 30 10 01 14 10 06	VISA Application (Visa Interlink)
Group AID Partial Selection Selection Features Max AID Length Transaction Type List Kernel ID Application Flow	DF EE 2D 9F 06 DF EE 4B DF EE 4D DF EE 2E DF EE 54 DF EE 59 DF EE 4C	01 07 01 01 01 0C 01 01	D0 A0 00 00 00 65 10 10 01 54 10 05 00 D0 05 01 D0 05 09 D0 05 20 D0 05 0E	JCB EMV Application
Group AID Partial Selection Max AID Length Application Flow	DF EE 2D 9F 06 DF EE 4B DF EE 2E DF EE 4C	01 07 01 01 01	B0 A0 00 00 03 24 10 10 01 10 0D	Discover (ZIP) Application

Name	Tag	Length (Hex)	Value (Hex)	Card Application
Group AID Partial Selection Max AID Length Exclude from Processing Application Flow	DF EE 2D 9F 06 DF EE 4B DF EE 2E DF EE 53 DF EE 4C	01 07 01 01 01 01	C0 A0 00 00 02 77 10 10 01 10 02 15	Interac Application
Group AID Partial Selection Max AID Length Application Flow	DF EE 2D 9F 06 DF EE 4B DF EE 2E DF EE 4C	01 07 01 01 01	B0 A0 00 00 01 52 30 10 01 10 0D	Discover (D-PAS) Application
Group AID Partial Selection Selection Features Max AID Length Application Flow	DF EE 2D 9F 06 DF EE 4B DF EE 4D DF EE 2E DF EE 4C	01 07 01 01 01 01	90 A0 00 00 00 03 80 10 01 14 10 06	Visa Application (Visa Plus)

24. 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:



25. 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 demane
UI scheme	ViVotech	ON	ON
	Visa wave	Flash every 1 sec	Toggle
	EMEA	ON	Flash every 5 sec

26. 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).

Table 94: Enhanced Encrypted MSR Data Output Format for Bank Card

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)	0xXX	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 (TTK not support)

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, then output response would keep plaintext even if encryption is enabled and force encryption is set to ON.

27. Appendix A.15: 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 (terminal)
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

28. Appendix B Revision History

Revision	Date	Change
		• IDG is forked after NEO 1 rev. 142. This document is now for NEO 2.
Rev. 50	03/27/2018	• 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 NEOII-1256.
Rev. 51	04/27/2018 05/03/2018	• 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 NEOII-1397, -1399, -1467. • Changed the language about 62-series commands to say no command should be issued, other than Cancel, between responses. • Command 09-14: Response table updated to include VP6300
Rev. 52	05/15/2018	• 2C-0A Response table is updated. New note added about block number and error code.
Rev. 53	05/16/2018 05/18/2018 05/30/2018	• 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 NEOII configuration group default tags -- page 498(A.17) • Add NEOII Default System AIDs -- page 499(A.18) • Add Card in notification behavior – page 502(A.19) • Add commands 81-0E and 81-0F. • Change default value of 9F03 in Group 0 to all zeros in Table 74.
Rev. 54	05/30/2018	• Removed most references to Protocol 1. • Removed Burst Mode discussion.
Rev. 55	05/31/2018 06/01/2018	• 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.)
Rev. 56	06/4/2018	• Add command 25-07 (Set 24hr Self-Check Time)

Rev. 57	06/07/2018	• Add command 02-09, 02-58 (Used for MCL Data Exchange)
Rev. 58	06/07/2018	• Amend the Removal Tamper Configuration (04-22) information. • Remove note from Command 60-10 to fix NEOII-1358.
Rev. 59	06/22/2018	• Comprehensive copy edits, formatting adjustments. • Remove several obsolete Appendices. • Remove Smart Tap section and refer to external document.
Rev. 60	06/29/2018	• Add 60-17 and 60-18 commands. • Delete DFEE44 from Group 80.
Rev. 61	06/29/2018 07/03/2018	• Change DFDE04 as DFEF04. • Add Appendix of Enhanced Encrypted MSR Data Output Format for Financial and Non-Financial Cards. • JCB default value of DFEE4C is 0E. • Remove max size limitation on 84-07 command.
Rev. 62	07/06/2018	• Add commands 2C-19, -1A, -1B, -1C.
Rev. 63	07/12/2018 07/16/2018	• Add command 83-41. • Add additional items to response table for command 61-01.
Rev. 64	07/23/2018	• Add Set 24-hr Self-Check Time to command 25-07.
Rev. 65	07/24/2018	• Updated command information for 81-0B, 81-0C, and 81-0D
Rev. 66	07/30/2018 08/10/2018	• 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.
Rev. 67	08/17/2018	• Remove Visa commands 04-0C, -0D, -E and 03-0C, -D, -E.
Rev. 68	08/20/2018 08/26/2018 08/31/2018	• 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. • 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	• Remove commands FO-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	• Remove commands C7-10, -12, -13, -16, and -23. (Obsolete.)
Rev. 71	11/08/2018	• Remove C7-2F, per JX advice.

Rev. 72	11/30/2018	• 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 12/04/2018	• Add D1-1B. (Remove 90-14 for the 12/4/2018 revision.) • Update 83-41, C7-42. • Add 01-06. • Clarify AID not found error code and message.
Rev. 74	12/10/2018	• Add commands 77-82 and 77-83 for Bluetooth.
Rev. 75	12/17/2018	• Add 32-03. • Clarify D1-1B, D1-16, 09-01, and 83-41.
Rev. 76	12/20/2018	• Add C7-44, C7-45, 70-03. • Add 19-01, 19-02 for SCRP. • Update 61-02 for SCRP.
Rev. 77	12/26/2018	• Remove 81-02, 81-04, 81-0A, C7-2D, C7-2E.
Rev. 78	01/07/2019	• 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	• 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	• Add Cancel Transaction (05-01) to Contact commands section.
Rev. 81	02/19/2019	• 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	• Update Bootup Notification Command 14-01 with additional detail.
Rev. 83	04/11/2019	• 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 White List (2C-53). • Removed 2C-50, 2C-52, and D1-16 commands; they are no longer supported in NEO2.

	05/3/2019	• 60-12: update Default Apply Host Response Output TLV Data List table. • Global Config Tags: add Google SmartTap 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
	
	05/21/2019
		• Global TLV settings:
		  ○ Added DFED58, "Support of List of AIDs."
		• Get Network Configuration (D1-15):
		  ○ Corrected Byte 13 to 00h.
		• Error codes, protocol 2:
		  ○ Added 1DH, "Tamper occurred."
		• General Commands:
		  ○ Added Set System Language (D1-20).
		  ○ Added Set Model Number (90-15).
		  ○ Added Get Health Data from SCRP (29-30).
		• Transaction Related Commands:
		  ○ Contact: Updated Contact Get PIN Control (61-02).
		  ○ Added note that 19-10 through 19-13 are reserved and used to Create Secure Channels for SCRP.
		  ○ Added 1CVMApp Get Nonce from SCRP for Refresh Token (19-15).
		  ○ Added Refresh Enablement Token for SCRP (19-16).
		• Configuration Tab Tables, FFF3:
		  ○ Corrected Google SmartTap support to Byte 1, Bit 5.
		• Secure Pass-Through Function:
		  ○ Get White List (2C-51): Corrected missing information in **Command Frame**.
		  ○ Get Discretionary Data White List (2C-53): Added missing information in **Command Frame** and example response.
		• Logger Commands:
		  ○ Added new commands for the logging system: 4C-01, 4C-02, 4C-03, 4C-04, 4C-05, and 4C-06.
		• EMV Result Codes:
		  ○ Removed several EMV result codes that are not supported by the NEO II IDG.
		• General Commands:
		  ○ Added Get Product full information (29-01).
		• Global TLV settings:
		  ○ Corrected tag value for DFED58 to 1 byte in length.

	05/31/2019
	
	06/04/2019
		• General commands:
		  ○ D1-1B: Added data response table.
		• Miscellaneous NEO II Platform Commands:
		  ○ 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:
		  ○ F1-00: Added Wake-Up Notification command.
		• EMV Result Codes:
		  ○ Corrected EMV_RESULT_CODE_APP_BLOCK_AID to 0x5036.
		• System AIDs:
		  ○ Updated Table 18, System AIDs.
		  ○ Updated Table 28, System AID Default Configuration TLVs.
		• Transation Related Commands: Contactless:
		  ○ DFEF3C: Updated description table.

Rev. 85.1		• General
		  ○ Implemented new minor version numbering scheme. Each update will increment the minor version for the sake of easier version tracking in JIRA.
		• Miscellaneous NEO II Platform Commands
		  ○ 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
		  ○ Added Set USB Power Mode Suspend Configuration (F0-08) and Get USB Power Mode Suspend Configuration (F0-09).
		• Secure Communication
		  ○ Updated Set MSR Secure Parameters (C7-38).
		• Apple VAS Functionality:
		  ○ Added DFED49 to ACT Parameters for ApplePay VAS table.

Rev 85.2	07/26/2019	• Peer To Peer Functionality
		  o Removed Detect Card Presence Command (02-05) as it is currently unsupported.
		• Contact Start Transaction (60-10):
		  o Added DFEF1F useage rules.
		  o Removed 9F39 tag and description.
		• Contact Authenticate Transaction (60-11):
		  o Added DFEE1A usage rules.
		  o Added tag DF30 supplement description.
		  o Updated 60-11 Default Authenticate Transaction Output TLV Data List to follow FastEMV Output format
		• Contact Apply Host Response (60-12):
		  o Added tag DF30 supplement description.
		• MSR Equivalent Data Function:
		  o Added 60-10, 60-11, and 60-12 commands to the list of responses in which DFEF4C and DFEF4D TLV tags can appear.
		  o Added 60-06 to the instructions for setting DFEF4B.
		• Get DRS Info (C7-3A):
		  o Added note to Tamper Switch (0x11) explaining the diference in paired switches found in VP3600.
		• Get TMS (29-20):
		  o Added Get TMS command.

Rev 86.1	08/02/2019	• Updated TLV tags to meet current NEO II spec.
		  o Most Global Configuration Tags were removed
		  o PayPass Group Configuration TLVs are now in Master Group Configuration TLVs

Rev 87.1	08/16/2019	• Get DRS Info (C7-3A):
		  o Changed Source ID 2 from "Reserved" to "TM4 Slave Check Value Error."
		• Pass-through Data Exchange Commands:
		  o Added Poll for Token with ATS (2C-0E).
		• Added Appendix A.7: VP6800 Commands
		  o Includes commands, limitations, and a screen creation tutorial for the VP6800.

Rev 87.2	08/23/2019	• Response codes
		  o Added A0h, A1h, and A2h.
		• Appendix A.7: VP6800 Commands
		  o Get External Chip Status (83-51):
		    ▪ Renamed from Get RT1050 Status.
		  o Get RT1050 Firmware Version (29-80):
		    ▪ Renumbered command from (29-00)
		  o Updated several command and **Response Frame**s in multiple commands.
		  o Added additional description for the response.
		  o Moved 61-30, 61-31, 61-41, 61-42, 61-43, and 61-44 from general commands to Appendix A.7
		  o Added video support information to VP6800 display tables.

Rev 88.1	10/7/2019	• General
		  ○ Format overhaul for current branding style.
		  ○ Revised text for style, grammar, and clarity from the start of the document through section 8.2.
		• Commands
		  ○ 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: White List Commands for VP3300.