



USER MANUAL

Xpress CM100 Technical Reference Manual

CE FC

**80099506-001-A
09/27/11**

IDTECH Xpress CM100 Technical Reference Manual

FCC WARNING STATEMENT

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his expense.

FCC COMPLIANCE STATEMENT

This device complies with Part 15 of the FCC Rules. Operation of this device is subject to the following conditions: this device may not cause harmful interference and this device must accept any interference received, including interference that may cause undesired operation.

CANADIAN DOC STATEMENT

This digital apparatus does not exceed the Class B limits for radio noise for digital apparatus set out in the Radio Interference Regulations of the Canadian Department of Communications.

Le présent appareil numérique n'émet pas de bruits radioélectriques dépassant les limites applicables aux appareils numériques de la classe A prescrites dans le Règlement sur le brouillage radioélectrique édicté par le ministère des Communications du Canada.

CE STANDARDS

An independent laboratory performed testing for compliance to CE requirements. The unit under test was found compliant to Class B.

LIMITED WARRANTY

ID TECH warrants to the original purchaser for a period of 24 months from the date of invoice that this product is in good working order and free from defects in material and workmanship under normal use and service. ID TECH's obligation under this warranty is limited to, at its option, replacing, repairing, or giving credit for any product which has, within the warranty period, been returned to the factory of origin, transportation charges and insurance prepaid, and which is, after examination, disclosed to ID TECH's satisfaction to be thus defective. The expense of removal and reinstallation of any item or items of equipment is not included in this warranty. No person, firm, or corporation is authorized to assume for ID TECH any other liabilities in connection with the sales of any product. In no event shall ID TECH be liable for any special, incidental or consequential damages to purchaser or any third party caused by any defective item of equipment, whether that defect is warranted against or not. Purchaser's sole and exclusive remedy for defective equipment, which does not conform to the requirements of sales, is to have such equipment replaced or repaired by ID TECH. For limited warranty service during the warranty period, please contact ID TECH to obtain a Return Material Authorization (RMA) number & instructions for returning the product.

THIS WARRANTY IS IN LIEU OF ALL OTHER WARRANTIES OF MERCHANTABILITY OR FITNESS FOR PARTICULAR PURPOSE. THERE ARE NO OTHER WARRANTIES OR GUARANTEES, EXPRESS OR IMPLIED, OTHER THAN THOSE HEREIN STATED. THIS PRODUCT IS SOLD AS IS. IN NO EVENT SHALL ID TECH BE LIABLE FOR CLAIMS BASED UPON BREACH OF EXPRESS OR IMPLIED WARRANTY OF NEGLIGENCE OF ANY OTHER DAMAGES WHETHER DIRECT, IMMEDIATE, FORESEEABLE, CONSEQUENTIAL OR SPECIAL OR FOR ANY EXPENSE INCURRED BY REASON OF THE USE OR MISUSE, SALE OR FABRICATIONS OF PRODUCTS WHICH DO NOT CONFORM TO THE TERMS AND CONDITIONS OF THE CONTRACT.

The information contained herein is provided to the user as a convenience. While every effort has been made to ensure accuracy, ID TECH is not responsible for damages that might occur because of errors or omissions, including any loss of profit or other commercial damage, nor for any infringements or patents or other rights of third parties that may result from its use. The specifications described herein were current at the time of publication, but are subject to change at any time without prior notice.

ID TECH is a registered trademark of International Technologies & Systems Corporation. Xpress and Value through Innovation are trademarks of International Technologies & Systems Corporation.

IDTECH Xpress CM100 Technical Reference Manual

Revision History

Revision	Date	Description	By
50	9/27/2011	Initial Draft	JW

Table of Contents

1.0	Command Protocol	7
1.1	Interface Note	7
1.2	Host to Device Command	7
1.3	Device to Host Command	7
2.0	Suggested Commands for Transactions	8
3.0	Serial Port Control Command Set	9
3.1	SET_BAUDRATE	9
3.2	SET_STOPBITS	9
3.3	SET_PARITY	10
4.0	System Command Set	11
4.1	GET_VERSION	11
4.2	BEEP	11
4.3	LED_CONTROL	12
4.4	CANCEL_CURRENT_TRANSACTION / APPLICATION_ON	12
4.5	APPLICATION_OFF	12
5.0	Application Related Commands	14
5.1	APP_GET_DATA	14
5.2	APP_GET_TRAN_RESULT	14
5.3	APP_GET_TRANCVM	15
5.4	APP_GET_DATETIME	15
5.5	APP_SET_DATETIME	16
5.6	TERMINAL_START_APPLICATION	16
6.0	Data Setting Commands	18
6.1	RETRIEVE_TERMINAL_DATA	18
6.2	REMOVE_TERMINAL_DATA	18
6.3	SET_TERMINAL_DATA	19
6.4	SELECT_TERMINAL_APPLICATION	19
6.5	RETRIEVE_PUBLIC_KEY	20
6.6	REVOKE_PUBLIC_KEY	21
6.7	INVOKE_PUBLIC_KEY	21

IDTECH Xpress CM100 Technical Reference Manual

6.8	RETRIVE_PUBLIC_KEY_COUNT.....	22
7.0	Device Direct Control Commands.....	23
7.1	SEARCH CARDS.....	23
7.2	SELECT ONE CARD.....	24
7.3	DELETE ONE CARD.....	24
7.4	DELETE ALL CARDS.....	24
7.5	GET USING CARDS COUNT.....	24
7.6	GET FREE SLOTS.....	25
7.7	GET CARD CID.....	25
7.8	DISABLE CARD CID.....	25
7.9	DISABLE CARD NAD.....	26
7.10	SEND COMMAND TO CARD.....	26
	Appendix 1: Terminal Data List.....	27
	Appendix 2: Data Record Detail for Amex- Mag _Stripe and Discover- Mag _Stripe.....	34
	Appendix 3: Data Record Detail for Amex- EMV.....	34
	Appendix 4: Data Record Detail for Paypass- M/Chip.....	34
	Appendix 5: Data Record Detail for Paypass- Mag _Stripe.....	35
	Appendix 6_A: Data Record Detail for Visa- MSD-Legacy-Only.....	35
	Appendix 6_B: Data Record Detail for Visa- MSD- CVN17.....	36
	Appendix 7: Data Record Detail for Visa- qVSDC.....	36
	Appendix 8: Data Element Format Conventions.....	37
	Appendix 9: Error code for response to host.....	38

1.0 Command Protocol

1.1 Interface Note

- **For USB HID :** The device's VID =0x0ACD, PID =0x1550
- **For RS232 :** Default serial port parameters are: 38400 baud rate, 8 data bits, 1 stop bit, no parity.

1.2 Host to Device Command

- **FORMAT:**< STX, LenL, LenH, CommandData, Lrc1 , Lrc2 , ETX>
 - ✧ <STX>: 0x02. 1 byte.
 - ✧ <LenL+LenH>: sizeof(CommandData), 2 bytes.
 - ✧ <CommandData>: main command string. Several bytes needed.
 - ✧ <Lrc1>: Exclusive or of CommandData. 1 byte.
 - ✧ <Lrc2>: Sum of CommandData. 1 byte.
 - ✧ <ETX>: 0x03. 1 byte.
- **EXAMPLE:**
 - ✧ Turn application off command is: "02 03 00 78 16 00 6E 8E 03".
 - "02": STX;
 - "03 00": LenL LenH;
 - "78 16 00": CommandData;
 - "6E 8E": Lrc1 Lrc2;
 - "03": ETX

1.3 Device to Host Command

- **FORMAT:**<STX, LenL, LenH, ResponseData, Lrc1, Lrc2, ETX>
 - ✧ STX: 0x02. 1 byte.
 - ✧ LenL+LenH: size of (ResponseData), 2 bytes.
 - ✧ ResponseData: main response string. Several bytes needed.
 - ✧ Lrc1: Exclusive or of ResponseData. 1 byte.
 - ✧ Lrc2: Sum of ResponseData. 1 byte.
 - ✧ ETX: 0x03. 1 byte.
- **EXAMPLE :**
 - ✧ **Success** Response to RETRIVE_PUBLIC_KEY_COUNT as <ACK><Count(1 byte)>: "02 02 00 06 05 03 0b 03"
 - "02": STX;
 - "02 00": LenL LenH;
 - "06 05": ResponseData , <ACK><Count(1 byte)>;

- “03 0b”: Lrc1 Lrc2;
- “03”: ETX
- ✧ **Failure** Response as <NAK><Error_Code(1 byte)>: “02 02 00 15 e6 f3 fb 03”.
 - “02”: STX;
 - “02 00”: LenL LenH;
 - ”15 e6”: ResponseData ,<NAK><Error_Code(1 byte)>, for detail of Error_Code see Appendix 9;
 - “f3 fb”: Lrc1 Lrc2;
 - “03”: ETX

2.0 Suggested Commands for Transactions

- Step1:
 - ◆ 4.4 CANCEL_CURRENT_TRANSACTION / APPLICATION_ON
- Step2:
 - ◆ 5.5 APP_SET_DATETIME
 - ◆ 6.3 SET_TERMINAL_DATA
 - ◆ 6.7 INVOKE_PUBLIC_KEY
 - ◆ 6.4 SELECT_TERMINAL_APPLICATION
- Step3:
 - ◆ 5.6 TERMINAL_START_APPLICATION
- Step4:
 - ◆ 5.2 APP_GET_TRAN_RESULT
 - ◆ 5.1 APP_GET_DATA

Note1: if all the terminal data has been set correctly (such as the default setting when it leaves factory), the least step3 will be just Step3 & Step4.

Note2: when communicating with host, the reader is always working in passive mode, only replying to host's command one by one, and never sending data to host initiatively. So, the host must expect the reply to the current command from the reader before sending the next one, and if there is no response, some errors might have occurred. And after the transaction is finished, host has to send 5.2 APP_GET_TRAN_RESULT and 5.1 APP_GET_DATA initiatively to get the result data from reader. For MasterCard Paypass, the host needs to send additional command 5.3 APP_GET_TRANCVM to get the outcome of the CVM Selection function.

3.0 Serial Port Control Command Set

The default parameters are: 38400 baud rate , 8 data bits, 1 stop bit, no parity, no handshaking.

3.1 SET_BAUDRATE

- **FUNCTION:** Set the device's baud rate.
- **COMMAND:** <0x70><0x01><New Baud rate>.
- **PARAMETERS:**
 - ✧ <0x70><0x01> is the command head. 2 bytes.
 - ✧ <New Baud rate> is the required serial port communication baud rate. 1 byte:
 - 0x01 for 1200,
 - 0x02 for 2400,
 - 0x03 for 4800,
 - 0x04 for 9600,
 - 0x05 for 14400,
 - 0x06 for 19200,
 - 0x07 for 38400,
 - 0x08 for 57600,
 - 0x09 for 115200.
- **RETURN:**
 - ✧ **Success:** <ACK><Old Baud rate (1 byte)>. Returns the old Baud rate.
 - ✧ **Fail:** <NAK><Error code>. For detail of Error_Code see Appendix 9.

3.2 SET_STOPBITS

- **FUNCTION:** This command is used to set the device's stop bits. The total length is 3 bytes.
- **COMMAND:** <0x70><0x02><NewStopBits>.
- **PARAMETERS:**
 - ✧ <0x70><0x02> is the command head.
 - ✧ <NewStopBits> is the required serial port communication stop bits. 1 byte:
 - 0x01 for 1 stop bit,
 - 0x02 for 2 stop bits,
- **RETURN:**
 - ✧ **Success:** <ACK>< OldStopBits (1 byte)>. Returns the old stop bits.
 - ✧ **Fail:** <NAK><Error code>. For detail of Error_Code see Appendix 9.

3.3 SET_PARITY

- **FUNCTION:** Set the device's parity style.
- **COMMAND:** <0x70><0x03><NewParity>.
- **PARAMETERS:**
 - ✧ <0x70><0x03> is the command head.
 - ✧ <NewParity> is the required serial port communication parity. 1 byte:
 - 0x00 for no parity,
 - 0x01 for odd parity,
 - 0x02 for even parity,
 - 0x03 for mark parity,
 - 0x04 for space parity.
- **RETURN:**
 - ✧ **Success:** <ACK>< OldParity. (1 byte)>. Returns the old parity.
 - ✧ **Fail:** <NAK><Error code>. For detail of Error_Code see Appendix 9.

4.0 System Command Set

4.1 GET_VERSION

- **FUNCTION:** Get the system's version.
- **COMMAND:** <0x78><Version_Sel>.
- **PARAMETERS:**
 - ✧ <0x78> is the command.
 - ✧ <Version_Sel > is defined as following:
 - #define COMMAND_GET_VERSION_READER 0x01
 - #define COMMAND_GET_LEVEL1_VERSION 0x91
 - #define COMMAND_GET_VERSION_PAYPASS 0xA1
 - #define COMMAND_GET_VERSION_VISA 0xB1
 - #define COMMAND_GET_VERSION_EXPRESS 0xC1
 - #define COMMAND_GET_VERSION_DISCOVER 0xD1
- **RETURN:**
 - ✧ Success: <ACK><version string>.
 - ✧ Fail: This operation will succeed always.
- **EXAMPLE:**
 - ✧ send: “78 01” means: Get the reader’s version
 - ✧ respond: “06 49 44 54 45 43 48 20 58 70 72 65 73 73 20 43 4D 31 30 30 20 52 65 61 64 65 72 20 56 31 2E 30 30 00” means:
 <ACK><"IDTECH Xpress CM100 Reader V1.00">

4.2 BEEP

- **FUNCTION:** Make the reader's beeper to beep for a designated time.
- **COMMAND:** <0x78><0x04><Duration>
- **PARAMETERS:**
 - ✧ <0x78><0x04> is the command header.
 - ✧ <Duration> is the beeper's beep last time in 0.1 second. Note that the beep time will never last over 10s even if it’s set as over 100 in this byte.
- **RETURN:**
 - ✧ Success: <ACK>.
 - ✧ Fail: This operation will succeed always.
- **EXAMPLE:**
 - ✧ send: “78 04 10” means: Beep 1.6s
 - ✧ respond: “06” means: <ACK>, and with a beep lasting 1.6s

4.3 LED_CONTROL

- FUNCTION: Control the system's led.
- COMMAND: <0x78><0x05><LEDMAP>.
- PARAMETERS:
 - ✧ <0x78><0x05> is the command header.
 - ✧ <LEDMAP> is the led lighting_map.
 - 0x01: led1 on, others off;
 - 0x03: led1&led2 on, others off;
 - 0x0f: all 4 leds on;
 - 0x00: all leds off.
- RETURN:
 - ✧ Success: <ACK>.
 - ✧ Fail: This operation will succeed always.
- EXAMPLE :
 - ✧ send “78 05 03” means: light up led1 and led2, others off
 - ✧ respond: “06” means: <ACK>, and the led1&2 light up.

4.4 CANCEL_CURRENT_TRANSACTION / APPLICATION_ON

- FUNCTION: to cancel the current transaction and to turn on application, use this command before setting terminal data or when turning from L1_Interface or Msr_Interface to L2_Interface for transaction.
- COMMAND: <0x78><0x16><0x00>.
- PARAMETERS:
 - ✧ <0x78><0x16><0x00> is the command header.
- RETURN:
 - ✧ Success: <ACK>.
 - ✧ Fail: This operation will succeed always
- EXAMPLE:
 - ✧ send: “78 16 00” means: cancel the current transaction, and turn on application.
 - ✧ respond: “06” means: <ACK>

4.5 APPLICATION_OFF

- FUNCTION: to turn off application, use this command before sending command to L1.
- COMMAND: <0x78><0x16><0x01>.
- PARAMETERS:
 - ✧ <0x78><0x16><0x01> is the command header.
- RETURN:
 - ✧ Success: <ACK>.
 - ✧ Fail: This operation will succeed always

IDTECH Xpress CM100 Technical Reference Manual

- EXAMPLE:
 - ✧ send: “78 16 01” means: turn off application.
 - ✧ respond: “06” means: <ACK>

5.0 Application Related Commands

5.1 APP_GET_DATA

- FUNCTION: Get the transaction log data
- COMMAND: <0x7E><0x02><0x01>
- PARAMETERS:
 - ✧ <0x7E><0x02><01> is the command header.
- RETURN:
 - ✧ Success: <ACK>< Card_type ><DataLength(2bytes) >< Data>.
 - < Card_type>: 1 byte, see the following definition:
 - #define MASTER_APPLICATION 0x01
 - #define VISA_APPLICATION 0x02
 - #define EXPRESS_APPLICATION 0x03
 - #define DISCOVER_APPLICATION 0x04
 - <DataLength(2bytes)>: is the length of all bytes from <Card_type> to the end of < Data>, and the low_byte is prior.
 - < Data>: is the Data Record, all in EMV_TLV format, note here all tags remain it's original length as 1 or 2 bytes; for the data objects which has no Tag In EMV use the following proprietary Tags:
 - "Track1" : "1f11"
 - "Track2" : "1f12"
 - "POS Entry Mode": "1f13"
 - "Terminal Entry Capability": "1f14"
 - ✧ See Appendix 2~7 for the details.
 - ✧ Fail: <NAK><ERROR_NO_DATA_AVAILABLE>
 <ERROR_NO_DATA_AVAILABLE> is Error code, see Appendix 9.
- EXAMPLE :
 - ✧ send: "7e 02 01" means: Get the transaction log data
 - ✧ respond: "15 e8" means: Fail, no data available
 - ✧ respond: "06 03 a2 00 9f 36 02 08 33 82 02 59 80 95 05 00 20 00 00 00..." means:
 - "06 03": ACK, EXPRESS_APPLICATION
 - "a2 00": DataLength = 0x00a2
 - "9f 36 02 08 33 82 02 59 80 95 05 00 20 00 00 00...": ATC(9f36)_TLV, AIP(82)_TLV, TVR(95)_TLV,...

5.2 APP_GET_TRAN_RESULT

- FUNCTION: Get the last transaction result information. The reader will indicate to the terminal the outcome of its transaction processing by this command.
- COMMAND: <0x7E><0x26>
- PARAMETERS:

- ◇ <0x7E><0x26> is the command head.
- RETURN:
 - ◇ Success: <ACK><Result>
 - <Result> is 1 byte, see below definition.
 - #define TRANSACTION_TWOCARDS 0x7A
 - # TRANSACTION_TERMINATE 0x7E
 - # TRANSACTION_TERMINATE_TRY_ANOTHER 0x7D
 - # TRANSACTION_DECLINE 0x81
 - # TRANSACTION_ACCEPT_OFFLINE 0x82
 - # TRANSACTION_GO_ONLINE 0x84
 - ◇ Fail: <NAK><ERROR_NO_DATA_AVAILABLE>, for detail of Error_Code see Appendix 9.
- EXAMPLE:
 - ◇ send: “7e 26” means: Get the last transaction result information
 - ◇ respond: “15 e8” means: Fail, <NAK><ERROR_NO_DATA_AVAILABLE>
 - ◇ respond: “06 84” means: Success, TRANSACTION_GO_ONLINE.

5.3 APP_GET_TRANCVM

- FUNCTION: Get the last terminal verification CVM , this command is only for MasterCard Paypass cards.
- COMMAND: <0x7E><0x28>.
- PARAMETERS:
 - ◇ <0x7E><0x28> is the command head.
- RETURN:
 - ◇ Success: <ACK>< TRANCVM >
 - < TRANCVM > is 1 byte, see below definition.
 - # NoCVM 0x01
 - # OnlinePin 0x02
 - # Signaure 0x03
 - # OfflinePin 0x04
 - # OfflPinSign 0x05
 - ◇ Fail: <NAK><ERROR_NO_DATA_AVAILABLE>, for detail of Error_Code see Appendix 9.

5.4 APP_GET_DATETIME

- FUNCTION: Get the current date and time.
- COMMAND: <0x7E><0x18>
- PARAMETERS:
 - ◇ <0x7E><0x18> is the command head.
- RETURN:

- ◇ Success: <ACK>< DATE(YYYYMMDDHHMMSS)>
- ◇ Fail: This operation will succeed always.
- EXAMPLE
 - ◇ send: “7e 18” means: Get the current date&time
 - ◇ respond: “06 32 30 31 31 30 33 31 30 31 30 33 38 31 39” means: <ACK>< DATE(20110310103839)>

5.5 APP_SET_DATETIME

- **FUNCTION:** Set the date and time in the system.
- **COMMAND:** <0x7E><0x08><DATE(YYYYMMDDHHMMSS)>.
- **PARAMETERS:**
 - ◇ <0x7E><0x08> is the command header.
 - ◇ <DATE(YYYYMMDDHHMMSS)> is the date and time in ASCII format.
- **RETURN:**
 - ◇ **Success:** <ACK>.
 - ◇ **Fail:** This operation will succeed always.
- **EXAMPLE**
 - ◇ send: “7e 08 32 30 31 31 30 33 31 30 31 32 35 34 33 30” means: set datetime as “2011-03-10-12-54-30”
 - ◇ respond: “06” means: ACK

5.6 TERMINAL_START_APPLICATION

- **FUNCTION:** Instruct the reader to start to discover card.
- **COMMAND:** <0x7E><0x60>< Status(1 byte) >< PollingTimeOut(2 bytes in 2 seconds interval) >< TLV data >
- **PARAMETERS:**
 - ◇ <0x7E><0x60> is the command header.
 - ◇ < Status(1 byte) > indicates if the transaction is continuous or just for one time .
 - 0x00: just for one time;
 - 0x04: continuous transaction, and it will use the < TLV data > of this continuous command as transaction data.
 - ◇ < PollingTimeOut(2 bytes in 2 seconds interval) >:Polling Card Time Out , if the time is out , the interface will power off and this command need to be sent again to start another transaction.
 - < 0x00 0x00>:keep polling until reach card
 - Others: PollingTimeOut=(byte0+256*byte1)*2S, byte0 is the low_byte and is prior
 - ◇ < TLV data >: used to set some temporary Dos such as Amount, etc. Note that here Tags are all in 2bytes, and any time note that for Paypass ‘0x8100’and ‘0x9f02’ should be send along with each other, see Appendix 1.
- **RETURN:**

IDTECH Xpress CM100 Technical Reference Manual

- ✧ **Success:** <ACK>.
- ✧ **Fail:** This operation will succeed always.
- **EXAMPLE**
 - ✧ send: “7e 60 00 05 00 9f 02 06 00 00 00 00 15 00 81 00 04 00 00 05 dc” means:
 - “7e 60” is the command header, start transaction now if card is present;
 - “00”: indicates that this current transaction is just one shot, not continuous;
 - “05 00” : means that PollingTimeOut is 10s(5*2s=10s);
 - “9f 02 06 00 00 00 00 15 00 81 00 04 00 00 05 dc”: set Amount Authorized (Numeric) as “1500”
 - ✧ send: “7e 60 04 05 00 9f 02 06 00 00 00 00 15 00 81 00 04 00 00 05 dc” means :
 - “7e 60” is the command header, start transaction now if card is present;
 - “04”: indicates that this current transaction is continuous, and during the following repeated transactions Amount Authorized (Numeric) will always be “1500”.
 - “05 00” : means that PollingTimeOut is 10s(5*2s=10s);
 - “9f 02 06 00 00 00 00 15 00 81 00 04 00 00 05 dc”: set Amount Authorized (Numeric) as “1500”
 - ✧ respond: “06” means : ACK

6.0 Data Setting Commands

To use these commands, the application must be off by sending “<0x78><0x16><0x00>”.

6.1 RETRIEVE_TERMINAL_DATA

- FUNCTION: Read the terminal data from the reader
- COMMAND: <0x7E><0xF0><0x01>< AID(6 bytes) >< DataID(1 byte) >.
- PARAMETERS:
 - ✧ <0x7E><0xF0><0x01> is the command head.
 - ✧ < AID(6 bytes) >is the AID the desired data belongs to; If it doesn't belong any AID, use 6Bytes 0x00 for this segment , see Appendix 1's Note2.
 - ✧ < DataID > is the desired data element's Serial Number for Host to Identify, and independent of its Tag for some Dos haven't Tag, see Appendix 1.
- RETURN:
 - ✧ Success: <ACK><Requested data element (TLV)>
 - ✧ Fail: <NAK><Error code (1 byte)>, for detail of Error_Code see Appendix 9.
- EXAMPLE
 - ✧ send “7e f0 01 00 00 00 00 00 08” means: Retrieve Transaction Currency Code(5f2a) Data.
 - ✧ respond: “06 5f 2a 02 09 78” means: Success: ACK, Transaction Currency Code returned at TLV mode.
 - ✧ respond: “15 E6” means: Fail: NAK ,The Data does not exist

6.2 REMOVE_TERMINAL_DATA

- FUNCTION: Clear the terminal data.
- COMMAND: <0x7E><0xF0><0x02>< AID(6 bytes) >< DataID(1 byte) >.
- PARAMETERS:
 - ✧ <0x7E><0xF0><0x02> is the command head.
 - ✧ < AID(6 bytes) >is the AID the desired data belongs to ; If it doesn't belong any AID, use 6Bytes 0x00 for this segment, see Appendix 1's Note2.
 - ✧ < DataID > is the desired data element's Serial Number for Host to identify, and independent of its Tag for some Dos haven't Tag, see Appendix 1.
- RETURN:
 - ✧ Success: <ACK>
 - ✧ Fail: <NAK><Error code (1 byte)>, for detail of Error_Code see Appendix 9.
- EXAMPLE
 - ✧ send “7e f0 02 00 00 00 00 00 08” means: Remove Transaction Currency Code(5f2a) Data,
“00 00 00 00 00 00” is < AID(6 bytes) >, “08” is the < DataID > for Transaction Currency Code.
 - ✧ respond: “06” means: Success: <ACK>.

✧ respond: “15 E6” means: Fail :<NAK>,The Data does not exist

6.3 SET_TERMINAL_DATA

- FUNCTION: Set terminal data.
- COMMAND:<0x7E><0xF0><0x04><AID(6 bytes)><DataID(1byte)><DataLength(2bytes) >< Data (Tag+Len+Value) >
- PARAMETERS:
 - ✧ <0x7E><0xF0><0x04> is the command head.
 - ✧ < AID(6 bytes) >is the AID the desired data belongs to ; If it doesn't belong any AID, use 6Bytes 0x00 for this segment, see Appendix 1’s Note2.
 - ✧ < DataID > is the desired data element's Serial Number for Host to identify, and independent of its Tag for some Dos haven't Tag, see Appendix 1.
 - For Paypass, DataID:1~28(excluding 10&11) need to be present.
 - For Visa, DataID:1,2,3,4,8,13,25,26,27,29~34 need to be present
 - For Express, DataID:1~2,4,6~9,12~14,17,19~22,25~27,29,35,37~39,41~43 need to be present.
 - ✧ <DataLength(2bytes) > is the length of < Data (Tag+Len+Value) >, the low_byte is prior.
 - ✧ < Data (Tag+Len+Value) >: the Data Objects, here Tags are all in 2bytes , see Appendix 1
- RETURN:
 - ✧ Success: <ACK><Error code (1 byte)>.
 - ✧ Fail: This operation will succeed always.
- EXAMPLE
 - ✧ send: “7e f0 04 a0 00 00 00 25 01 0d 08 00 9f 40 05 60 00 00 00 00” means:
Set **Additional Terminal Capabilities** (9f40) as “60 00 00 00 00” for Aid= “a0 00 00 00 25 01”
 - “a0 00 00 00 25 01” is < AID(6 bytes) >,
 - “0d” is <DataID(1byte)> of **Additional Terminal Capabilities**(see Appendix 1)
 - “08 00”is <DataLength(2bytes) >,length is 8;
 - “9f 40 05 60 00 00 00 00” is < Data (Tag+Len+Value) >
 - ✧ respond: “06” means: Success: <ACK>

6.4 SELECT_TERMINAL_APPLICATION

- FUNCTION: Select the Reader’s mode
- COMMAND: <0x7E><0xF0><0x05><Mode>.
- PARAMETERS:
 - ✧ <0x7E><0xF0><0x05> is the command head.
 - ✧ <Mode>: 1 byte, the reader’s mode need to be , see the following definition:
 - #define USER_MODE 0x00

- #define MASTER_APPLICATION 0x01
- #define VISA_APPLICATION 0x02
- #define EXPRESS_APPLICATION 0x03
- #define DISCOVER_APPLICATION 0x04
- RETURN:
 - ✧ Success: <ACK><Old application type (1 byte)>.
 - ✧ Fail: This operation will succeed always.
- EXAMPLE:
 - ✧ Send: “7e f0 05 03”: means Select EXPRESS_APPLICATION as current application type
 - ✧ Respond “06 01” means :Success: <ACK>, Old application type is MASTER_APPLICATION.
- NOTE: This command used for the testing, in the normal transaction process, on need to send this command.

6.5 RETRIEVE_PUBLIC_KEY

- FUNCTION: Retrieve the terminal's saved public key of a special RID and its index.
- COMMAND: <0x7E><0xF1><0x01><RID(5 B YTES)><Index(1 byte)>.
- PARAMETERS:
 - ✧ <0x7E><0xF1><0x01> is the command head.
 - ✧ <RID(5 BYTES)> is the application's RID.
 - ✧ <Index(1 byte)> is the key index.
- RETURN:
 - ✧ Success: <ACK><RID(5)><Index(1)><Hash Indicator(1)><Algorithm Indicator(1)><Exponent(4)>
 <Key Module Length(1)><Key Hash(20)><Module Data(var)>.
 - ✧ Fail: <NAK><Error code (1 byte)>, for detail of Error_Code see Appendix 9.
- EXAMPLE
 - ✧ Send: “7e f1 01 a0 00 00 00 25 97”
 - ✧ Respond: “06 A0 00 00 00 25 97 01 01 00 00 00 03 90 EB DA 52 2B 63 1B 3E B4 F4 CB FC 06 79 C4 50 13 9D 2B 69 CD E1 78 FF E8 34 B4 B7 67 AF 3C 9A 51 1F 97 3D 8E 85 05 C5 FC B2 D3 76 80 75 AB 7C C9 46 A9 55 78 99 55 87 9A AF 73 74 07 15 15 21 99 6D FA 43 C5 8E 6B 13 0E B1 D8 63 B8 5D C9 FF B4 05 09 47 A2 67 6A A6 A0 61 A4 A7 AE 1E DB 0E 36 A6 97 E8 7E 03 75 17 EB 89 23 13 68 75 BA 2C A1 08 7C BA 7E C7 65 3E 5E 28 A0 C2 61 A0 33 AF 27 E3 A6 7B 64 BB A2 69 56 30 7E C4 7E 67 4E 3F 8B 72 2B 3A E0 49 8D B1 6C 79 85 31 0D 9F 3D 11 73 00 D3 2B 09” means:
 - “06” is <ACK>
 - “a0 00 00 00 25 97” is < RID(5 bytes)> <Index(1 byte)>
 - “01 01 00 00 00 03 90” is < Hash_Indicator(1 byte)>< Algorithm_Indicator(1 byte)>< Exponent(4 bytes)> < Key Module Length(1 byte)>

- “eb da 52 2b 63 1b 3e b4 f4 cb fc 06 79 c4 50 13 9d 2b 69 cd” is < Key Hash(20 bytes)>
- “e1 78 ff e8 34 b4 b7 67 af 3c 9a 51 1f 97 3d 8e 85 05 c5 fc b2 d3 76...” is < Module Data(var)>

6.6 REVOKE_PUBLIC_KEY

- FUNCTION: Revoke the terminal's saved public key of a special RID and its index.
- COMMAND: <0x7E><0xF1><0x02><RID(5 BYTES)><Index(1 byte)>.
- PARAMETERS:
 - ✧ <0x7E><0xF1><0x02> is the command head.
 - ✧ <RID(5 BYTES)> is the application's RID.
 - ✧ <Index(1 byte)> is the key index.
- RETURN:
 - ✧ Success: <ACK>
 - ✧ Fail: This operation will succeed always.
- EXAMPLE
 - ✧ Send: “7e f1 02 a0 00 00 00 25 97” means:
 - “a0 00 00 00 25” is <RID(5 BYTES)>.
 - “97” is <Index(1 byte)>
 - ✧ Respond: “06” means: <ACK>

6.7 INVOKE_PUBLIC_KEY

- FUNCTION: Invoke a public key of a certain RID and its index in the terminal.
- COMMAND: <0x7E><0xF1><0x04>< RID(5 bytes)>< Index(1 byte)>< Hash_Indicator(1 byte)> < Algorithm_Indicator(1 byte)> < Exponent(4 bytes)>< Key Module Length(1 byte)> < Key Hash(20 bytes)> < Module Data(var)>.
- PARAMETERS:
 - ✧ <0x7E><0xF1><0x04> is the command head.
- RETURN:
 - ✧ Success: <ACK>
 - ✧ Fail: This operation will succeed always.
- EXAMPLE :
 - ✧ Send: “7e f1 04 a0 00 00 00 25 97 01 01 00 00 00 03 90 eb da 52 2b 63 1b 3e b4 f4 cb fc 06 79 c4 50 13 9d 2b 69 cd E1 78 FF E8 34 B4 B7 67 AF 3C 9A 51 1F 97 3D 8E 85 05 C5 FC B2 D3 76 80 75 AB 7C C9 46 A9 55 78 99 55 87 9A AF 73 74 07 15 15 21 99 6D FA 43 C5 8E 6B 13 0E B1 D8 63 B8 5D C9 FF B4 05 09 47 A2 67 6A A6 A0 61 A4 A7 AE 1E DB 0E 36 A6 97 E8 7E 03 75 17 EB 89 23 13 68 75 BA 2C A1 08 7C BA 7E C7 65 3E 5E 28 A0 C2 61 A0 33 AF 27 E3 A6 7B 64 BB A2

69 56 30 7E C4 7E 67 4E 3F 8B 72 2B 3A E0 49 8D B1 6C 79 85 31 0D 9F 3D 11
73 00 D3 2B 09”means:

- “a0 00 00 00 25 97” is < RID(5 bytes)> <Index(1 byte)>
 - “01 01 00 00 00 03 90” is < Hash_Indicator(1 byte)>< Algorithm_Indicator(1 byte)>< Exponent(4 bytes)> < Key Module Length(1 byte)>
 - “eb da 52 2b 63 1b 3e b4 f4 cb fc 06 79 c4 50 13 9d 2b 69 cd” is < Key Hash(20 bytes)>
 - “e1 78 ff e8 34 b4 b7 67 af 3c 9a 51 1f 97 3d 8e 85 05 c5 fc b2 d3 76...” is < Module Data(var)>
- ✧ Respond: “06” means: <ACK>

6.8 RETRIVE_PUBLIC_KEY_COUNT

- FUNCTION: Retrieve the count of the terminal's saved public key of a special RID.
- COMMAND: <0x7E><0xF1><0x08><RID (5 BYTES)>
- PARAMETERS:
 - ✧ <0x7E><0xF1><0x08> is the command head.
- RETURN:
 - ✧ Success: <ACK><Count(1 byte)>.
 - ✧ Fail: This operation will succeed always.
- EXAMPLE
 - ✧ Send: “7e f1 08 a0 00 00 00 25”means:
 - “a0 00 00 00 25” is <RID (5 BYTES)>
 - ✧ respond: “06 05”means:
 - <ACK>< 5>, there are 5 public_keys for Express RID.

7.0 Device Direct Control Commands

These commands are sent to L1 to control card directly. We call it L1_Interface.

This L1_Interface is available only in application_off state by sending command “4.5 APPLICATION_OFF”.

Flow may like the following:

- Step1:
 - ◆ 4.5 APPLICATION_OFF

- Step2:
 - ◆ 7.1 SEARCH CARDS

- Step3:
 - ◆ 7.2 SELECT ONE CARD

- Step4:
 - ◆ 7.5 ~ 7.10

- Step5:
 - ◆ 7.3 or 7.4

7.1 SEARCH CARDS

- **FUNCTION:** Detect cards in the field.
- **COMMAND:** <0xFE><0x01><A/B><AFI>
- **PARAMETERS:**
 - ✧ <0xFE><0x01> is the command head.
 - ✧ <A/B> is card type: 0 for type A and type B, 1 for type A and 2 for type B. 1 byte.
 - ✧ <AFI> is application family identifier for type B. 0 for type A. 1 byte.
- **RETURN:**
 - ✧ **Success:**

<ACK><CardCount>[<CardType><UIDLen><UID>][<CardType><UIDLen><UID>] means:

 - <CardCount> is the count of found cards in the field.
 - [<CardType><UIDLen><UID>] will happen <CardCount> times.
 - <CardType> is the card's type: 1 for type A and 2 for type B.
 - <UIDLen> is the following UID's length.
 - <UID> is the found card's UID (for type A card) or PUID (for type B card).
 - ✧ **Fail:** <ACK><0x00>.

7.2 SELECT ONE CARD

- **FUNCTION:** Select one card.
- **COMMAND:** <0xFE><0x02><CardType><LenOfUID/PUID><UID/PUPI>
- **PARAMETERS:**
 - ✧ <0xFE><0x02> is the command head.
 - ✧ <CardType> is the card's type: 1 for type A and 2 for type B. 1 byte.
 - ✧ <LenOfUID/PUID> is the length of the selected UID for type A or PUPI for type B 1byte.
 - ✧ <UID/PUPI>: UID for type A. PUPI for type B.
- **RETURN:**
 - ✧ **Success:** <ACK><CardHandle> means:
 - <CardHandle> is the selected card's handle, it will be used by other commands to communicate with a specified card.
 - ✧ **Fail:** <ACK><0x00>

7.3 DELETE ONE CARD

- **FUNCTION:** Delete one card from the system.
- **COMMAND:** <0xFE><0x03><CardHandle>.
- **PARAMETERS:**
 - ✧ <0xFE><0x03> is the command head.
 - ✧ <CardHandle> is the Card's handle returned by **SELECT ONE CARD** command.
- **RETURN:**
 - ✧ **Success:** <ACK><Free slots available>
 - ✧ **Fail:** This operation will succeed always.

7.4 DELETE ALL CARDS

- **FUNCTION:** Delete all cards from the system.
- **COMMAND:** <0xFE><0x04>
- **PARAMETERS:**
 - ✧ <0xFE><0x04> is the command head.
- **RETURN:**
 - ✧ **Success:** <ACK><Free slots available>. 2 bytes
 - ✧ **Fail:** This operation will succeed always.

7.5 GET USING CARDS COUNT

- **FUNCTION:** Get the cards count using by the device system.
- **COMMAND:** <0xFE><0x05>

- **PARAMETERS:**
 - ✧ <0xFE><0x05> is the command head.
- **RETURN:**
 - ✧ **Success:** <ACK><Using cards count by the device>. 2 bytes
 - ✧ **Fail:** This operation will succeed always.

7.6 GET FREE SLOTS

- **FUNCTION:** Get the free slots available in the device system.
- **COMMAND:** <0xFE><0x06>.
- **PARAMETERS:** <0xFE><0x06> is the command head.
- **RETURN:**
 - ✧ **Success:** <ACK><Free slots available>. 2 bytes
 - ✧ **Fail:** This operation will succeed always.

7.7 GET CARD CID

- **FUNCTION:** Retrieve one card's current CID used in the device system.
- **COMMAND:** <0xFE><0x07><CardHandle>.
- **PARAMETERS:**
 - ✧ <0xFE><0x07> is the command head.
 - ✧ <CardHandle> is the Card's handle returned by **SELECT ONE CARD** command.
- **RETURN:**
 - ✧ **Success:** <ACK><Card's CID> if available. CID is between 0 and 14.
 - ✧ **Fail:** This operation will return <NAK><ERROR_INVALID_CID_RETURNED> if failed, for detail of Error_Code see Appendix 9.

7.8 DISABLE CARD CID

- **FUNCTION:** Inform the device do not use card's CID in the system.
- **COMMAND:** <0xFE><0x08><CardHandle>
- **PARAMETERS:**
 - ✧ <0xFE><0x08> is the command head.
 - ✧ <CardHandle> is the Card's handle returned by **SELECT ONE CARD** command.
- **RETURN:**
 - ✧ **Success:** <ACK>.
 - ✧ **Fail:** This operation will succeed always.

7.9 DISABLE CARD NAD

- **FUNCTION:** Inform the device do not use card's NAD in the system.
- **COMMAND:** <0xFE><0x09><CardHandle>
- **PARAMETERS:**
 - ✧ <0xFE><0x09> is the command head.
 - ✧ <CardHandle> is the Card's handle returned by **SELECT ONE CARD** command.
- **RETURN:**
 - ✧ **Success:** <ACK>.
 - ✧ **Fail:** This operation will succeed always.

7.10 SEND COMMAND TO CARD

- **FUNCTION:** Make the device to communicate with the card in the system.
- **COMMAND:** <0xFE><0x0A><CardHandle><CommandBlock>
- **PARAMETERS:**
 - ✧ <0xFE><0x0A> is the command head.
 - ✧ <CardHandle> is the Card's handle returned by **SELECT ONE CARD** command.
 - ✧ <CommandBlock> is the input command.
- **RETURN:**
 - ✧ **Success:** <ACK><Card returned data>
 - ✧ **Fail:** <NAK><Error code>, for detail of Error_Code see Appendix 9.

IDTECH Xpress CM100 Technical Reference Manual

Appendix 1: Terminal Data List

Data ID	Name	Tag (Note1)	Format (Note4 & Appendix 8)	Length (bytes)	AID -Flag (Note 2)	PayPass Default	VISA Default	American Express Default
1	IFD Serial Number	0x9F1E	an8	8	0	Only written once during production testing, read-only for the customers		
2	Terminal Country Code	0x9F1A	n3	2	0	0x00 0x56		
3	Amount Authorized (Binary)	0x8100	b4	4	0	Can be none, set by “ 5.6 TERMINAL_START_APPLICATION ”		
4	Amount Authorized (Numeric) Equal to Amount Authorized (Binary) Range(0~0x4294967295)	0x9F02	n12	6	0			
5	Amount Other (Binary)	0x9F04	b4	4	0	0x00 0x00 0x00 0x00		
6	Amount Other (Numeric) Equal to Amount Other (Binary) Range(0~0x4294967295)	0x9F03	n12	6	0			
7	Transaction Category Code	0x9F53	an1	1	0	0x00		
8	Transaction Currency Code	0x5F2A	n3	2	0	0x09 0x78		
9	Transaction Currency Exponent	0x5F36	n1	1	0	0x02		
10								
11								
12	Transaction Type	0x9C00	n2	1	0	0x00		
13	Additional Terminal Capabilities	0x9F40	b5	5	1	0x60 0x00 0x00 0x00 0x00	0x02 0x00 0x00 0x00	0x60 0x00 0x00 0x00

IDTECH Xpress CM100 Technical Reference Manual

							0x01	0x00
14	Application Version Number	0x9F09	b2	2	1	0x00 0x02	None	0x00 0x01
15	Default UDOL	0x0000	b3	3	1	0x9F 0x6A 0x04	None	None
16	MagStripe Application Version Number	0x9f6d	b2	2	1	0x00 0x01	None	None
17	Merchant Category Code	0x9F15	n4	2	1	0x07 0x42	None	0x07 0x42
18	PayPass - Mag Stripe Indicator	0x0000	b1	1	1	0x01	None	None
19	Terminal Action Codes-default	0x0000	b5	5	1	0x00 0x00 0x00 0x00 0x00	None	0xc4,0x00,0x00,0x00,0x00,0x00
20	Terminal Action Codes-denial	0x0000	b5	5	1	0x00 0x00 0x00 0x00 0x00	None	0xc4,0x00,0x00,0x00,0x00,0x00
21	Terminal Action Codes-online	0x0000	b5	5	1	0x00 0x00 0x00 0x00 0x00	None	0xc4,0x00,0x00,0x00,0x00,0x00
22	Terminal Type	0x9F35	n2	1	1	0x25	None	0x25
23	Terminal Capabilities - No CVM Required	0x0000	b3	3	1	AID(A0000000041010): 0x00 0x08 0x88 AID(B012345678): 0x00 0x08 0x88 AID(A0000000043010): 0x00 0x08 0x08	None	None
24	Terminal Capabilities - CVM Required	0x0000	b3	3	1	AID(A0000000041010): 0x00 0x28 0x88 AID(B012345678): 0x00 0x28 0x88 AID(A0000000043010): 0x00 0x28 0x08	None	None
25	Terminal Contactless Transaction Limit Range(0-0x4294967295)	0x0000	n12	6	1	0x00 0x00 0x00 0x03 0x00 0x00	0x00 0x00 0x00 0x03 0x00 0x00	0x00 0x00 0x00 0x03 0x00 0x00
26	Terminal Contactless Floor Limit	0x0000	n12	6	1	0x00 0x00 0x00 0x01 0x00	0x00 0x00 0x00 0x01	0x00 0x00 0x00 0x01

IDTECH Xpress CM100 Technical Reference Manual

	Range(0~0x4294967295)					0x00	0x00 0x00	0x00 0x00
27	Terminal CVM Required Limit Range(0~0x4294967295)	0x0000	n12	6	1	0x00 0x00 0x00 0x03 0x00 0x00	0x00 0x00 0x00 0x00 0x60 0x00	0x00 0x00 0x00 0x03 0x00 0x00
28	Merchant Custom Data	0x9f7c	b20	20	1	0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08 0x09 0x0A 0x0B 0x0C 0x0D 0x0E 0x0F 0x10 0x12 0x13 0x14 0x15	None	None
29	Terminal Floor Limit	0x9f1b	b4	4	1	None	0x00,0x00,0x27,0x10	0x00,0x00,0x27,0x10
30	Merchant Name and Location	0x9f4e	b	var.	0	None	0x49 0x44 0x20 0x54 0x45 0x43 0x48 0x20 0x20 0x20 0x20 0x20 0x20 0x20 0x43 0x41 0x20 0x55 0x53 0x41	None
31	Merchant Identifier	0x9f16	ans15	15	0	None	0x49 0x44 0x30 0x54 0x45 0x43 0x48 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00	None
32	Visa Terminal Transaction Qualifiers(TTQ)	0x9f66	b4	4	1	None	0xA0 0x00 0x00 0x00	None
33	Visa Terminal SetFlags (See Note5)	0x0000	b4	4	1	None	0x5E 0x00 0x01 0x01	None
34	Visa Terminal UpFlags (See Note6)	0x0000	b5	5	1	None	0xFF 0xFF 0xFF 0xFF 0xFF	None
35	Terminal Capabilities	0x9f33	b3	3	1	None	None	0x00,0x28,0x88
37	Target Percentage for Random Selection	0x0000	b1	1	1	None	None	0x3C Range(0x00-0x63)
38	Maximum Target Percentage for Biased Random Selection	0x0000	b1	1	1	None	None	0x3C Range(Value in DataId=37) ~0x63)
39	Threshold Value for Biased Random Selection	0x0000	b4	4	1	None	None	0x00,0x00,0x00,0x00 Range: >=0 &&< Terminal Contactless

IDTECH Xpress CM100 Technical Reference Manual

								Floor Limits
41	Expresspay Terminal Capabilities	0x9f6d	b1	1	1	None	None	0x80
42	Express Terminal SetFlags(Note7)	0x0000	b4	4	1	None	None	0x0A,0x00,0x00,0x00
43	Unpredictable Number Range(Note8)	0x0000	b4	4	1	None	None	0x00,0x00,0x00,0x3C Range(0 to 1200)

Note1:

In all commands which relating to Terminal Data Management and involving tag, including :

- ✧ “5.6 TERMINAL_START_APPLICATION”
- ✧ “6.1 RETRIEVE_TERMINAL_DATA”
- ✧ “6.3 SET_TERMINAL_DATA”

all Dos' tags are all in 2bytes, e.g. 0x8100 is for "Amount Authorized (Binary)" which emv_tag is 0x81.

Note2:

AID-Flag indicates if the Data Object belongs to some certain AIDs, 1 is yes, and 0 is no;

- ✧ for AID-Flag=0:
 - AID=(0x00, 0x00, 0x00, 0x00, 0x00, 0x00);
- ✧ for AID-Flag=1:
 - Visa AID=(0xA0, 0x00, 0x00, 0x00, 0x03, 0x00),
 - Paypass AID=(0xA0, 0x00, 0x00, 0x00, 0x04, 0x10) or (0xA0, 0x00, 0x00, 0x00, 0x04, 0x30) or (0xB0, 0x12, 0x34, 0x56, 0x78, 0x00),
 - Express AID=(0xA0, 0x00, 0x00, 0x00, 0x25, 0x01)

Note3:

For Visa and Paypass, any time Amount Authorized (Binary) '0x8100' and Amount Authorized (Numeric) '0x9f02' should be set along with each other, but not must for Amex.

Note4:

In the command line, all the Value Parts are in the EMV format, see Appendix 8 ; and for those the Value segment is in multi-bytes , the value bytes come in display_turn , and conforming to EMV.

Note5:

“Visa Terminal SetFlags”value:Byte0~3, definition is as following, note that B1 viz.Byte0, is the first one comes in the command line.

- ✧ B1b1 (b1 viz.bit0) : "Status Check" ---see: " Visa Contactless Payment Specification: Req 5.52"
- ✧ B1b2: "Amount, Authorized of Zero Check" -----see: "Req 5.52"
- ✧ B1b3: "Reader Contactless Transaction Limit Check" ---see: "Req 5.52"
- ✧ B1b4: "Reader CVM Required Limit Check" -----see: "Req 5.52"
- ✧ B1b5: "Reader Contactless Floor Limit Check": Configurable to indicate whether this check is enabled or disabled. ----see: "Req 5.52"
- ✧ B1b6: "Terminal Floor Limit present", indicates if “Terminal Floor Limit '9f1b” is present, ----see: “Req 5.36”
- ✧ B1b7 (b7 viz.bit6) : "Reader Contactless Floor Limit present", indicates if “Reader Contactless Floor Limit” is present. If it's present, FloorLimit_Check must use "Reader Contactless Floor Limit" and ignore "terminal_floor_limit": ----see: “Req 5.36”
- ✧ B1b8: "Disable Online Currently" ---for "CLQ.K.001.02" in “Test Plan for Visa Contactless Payment Specification Version 2.1.1 qVSDC Contactless Readers”
- ✧ B2: RFU
- ✧ B3:(viz. Byte2) "Amount, Authorized of Zero Check: option(1) or (2)" ----see "Req 5.32 option 1 or 2"
- ✧ B4:(viz. Byte3) "MSD CVN17 enable(1) or disable(0)"

Note6:

“Visa Terminal UpFlags” indicates the items need to transport to host after transaction finished. It’s value has 4bytes:byte0~byte4, defination is as following, note that B1 viz.Byte0, is the first one comes in the command line.

- ✧ B1: RFU
- ✧ B2b1(b1 viz.bit0): 0X9F6E, Form Factor Indicator (FFI)
- ✧ B2b2: 0X9F7C, Customer Exclusive Data (CED)
- ✧ B2b3: 0X9F27, Cryptogram Information Data

IDTECH Xpress CM100 Technical Reference Manual

- ✧ B2b4: 0X9F02, Amount, Authorized (Numeric)
- ✧ B2b5: 0X9F03, Amount, Other (Numeric)
- ✧ B2b6: 0X9F33, Terminal Capabilities
- ✧ B2b7~8: RFU
- ✧ B3b1: 0X9A , Transaction Date
- ✧ B3b2: 0X9C , Transaction Type
- ✧ B3b3: 0X9F10, Issuer Application Data
- ✧ B3b4: 0X9F1A, Terminal Country Code
- ✧ B3b5: 0X9F26, Application Cryptogram
- ✧ B3b6: 0X9F36, Application Transaction Counter (ATC)
- ✧ B3b7: 0X9F37, Unpredictable Number
- ✧ B3b8: 0X9F5D, Available Offline Spending Amount (AOSA)
- ✧ B4b1: 0X57 , Track 2 Equivalent Data
- ✧ B4b2: 0X5A , Application Primary Account Number (PAN)
- ✧ B4b3: 0X5F20, Cardholder Name
- ✧ B4b4: 0X5F24, Application Expiration Date
- ✧ B4b5: 0X5F2A, Transaction Currency Code
- ✧ B4b6: 0X5F34, Application Primary Account Number Sequence Number (PSN)
- ✧ B4b7: 0X82 , Application Interchange Profile (AIP)
- ✧ B4b8: 0X95 , Terminal Verification Results (TVR)
- ✧ B5b1: TRACK1_0x1f11
- ✧ B5b2~4: RFU
- ✧ B5b5: TRACK2_0x1f12
- ✧ B5b6~8: RFU

Note7:

“Express Terminal SetFlags” value: Byte0~3, definition is as following, note that B1 viz. Byte0, is the first one comes in the command line:

- ✧ B1b1 (viz. bit0): support another interface flag

IDTECH Xpress CM100 Technical Reference Manual

- ✧ B1b2:Terminal sends GENAC command requesting ARQC with CDA signature
- ✧ B1b3:Full online supported
- ✧ B1b4:Optimised Processing Supported
- ✧ B1b5:Online Capable , but now cannot go online

Note8:

"Unpredictable Number Range" value:Byte0~3,

EXAMPLE:

- ✧ {0x00,0x00,0x00,0x3C}:means the range is 0~60;
- ✧ {0x00,0x00,0x01,0x60}:means the range is 0~316;
- ✧ {0x00,0x00,0x04,0xb0}:means the range is 0~1200;

The default range is 0 to 0x3C, which is also the minimum range.

Please bear in mind the maximum range as per the Amex spec is 0 to 1200

Appendix 2: Data Record Detail for Amex- Mag _Stripe and Discover- Mag _Stripe

Tag	Name
0x1f11	Magstripe Track1
0x1f12	Magstripe Track2

Appendix 3: Data Record Detail for Amex- EMV

Tag	Name
0x82	Application Interchange Profile
0x9F36	Application Transaction Counter
0x9F26	ARQC
0x9F27	CID
0x9F34	CVM Results
0x9F1E	IFD Serial Number
0x9F10	Issuer Application Data
0x9F33	Terminal Capabilities
0x9F35	Terminal Type
0x95	TVR
0x9F37	Unpredictable Number
0x9F02	Amount, Authorized
0x9F03	Amount, Other
0x5F25	Application Effective Date
0x5F24	Application Expiration Date
0x5A	Application PAN
0x5F34	Application PAN Sequence Number
0x9F15	Merchant Category Code
0x9F1A	Terminal Country Code
0x57	Track 2 Equivalent Data
0x5F2A	Transaction Currency Code
0x9A	Transaction Date
0x9F21	Transaction Time
0x9C	Transaction Type

Appendix 4: Data Record Detail for Paypass- M/Chip

Tag	Name
0x57	Track 2 Equivalent Data
0x9F6E	PayPass Third Party Data
0x84	DF Name

IDTECH Xpress CM100 Technical Reference Manual

0x50	Application Label
0x9F12	Application Preferred Name
0x9F11	Issuer Code Table Index
0x9F26	AC
0x9F27	CID
0x9F10	IAD
0x9F36	ATC
0x95	TVR 6
0x9F37	UN
0x5F2A	Transaction Currency Code
0x9C	Transaction Type
0x9f33	Terminal Capabilities
0x9A	Transaction Date
0x9F02	Transaction Amount
0x9F1A	Terminal Country Code
0x9F34	CVM Results
0x82	AIP
0x5A	Application PAN
0x5F34	Application PAN Sequence Number
0x5F24	Application Expiration Date

Appendix 5: Data Record Detail for Paypass- Mag _Stripe

Tag	Name
0x9F6B	Track 2 Data
0x56	Track 1 Data
0x1f11	DDCARD,TRACK1
0x1f12	DDCARD,TRACK2
0x9F6E	PayPass Third Party Data
0x84	DF Name
0x50	Application Label
0x9F12	Application Preferred Name
0x9F11	Issuer Code Table Index

Appendix 6 _A: Data Record Detail for Visa- MSD-Legacy-Only

Tag	Name
1f11	Track1
1f12	Track2
1f13	POS Entry Mode
1f14	Terminal Entry Capability

Appendix 6_B: Data Record Detail for Visa- MSD- CVN17

Tag	Name
1f11	Track1
1f12	Track2
1f13	POS Entry Mode
1f14	Terminal Entry Capability
0X9F6E	Form Factor Indicator (FFI)
0X9F7C	Customer Exclusive Data (CED)
0X9F02	Amount, Authorized (Numeric)
0X9F10	Issuer Application Data
0X9F26	Application Cryptogram
0X9F36	Application Transaction Counter (ATC)
0X9F37	Unpredictable Number
0X5F34	Application Primary Account Number Sequence Number (PSN)
0X9f66	Terminal Transaction Qualifiers

Appendix 7: Data Record Detail for Visa- qVSDC

Tag	Name
1f12	Track2
1f13	POS Entry Mode
1f14	Terminal Entry Capability
0X9F6E	Form Factor Indicator (FFI)
0X9F7C	Customer Exclusive Data (CED)
0X9F27	Cryptogram Information Data
0X9F02	Amount, Authorized (Numeric)
0X9F03	Amount, Other (Numeric)
0X9F33	Terminal Capabilities
0X9A	Transaction Date
0X9C	Transaction Type
0X9F10	Issuer Application Data
0X9F1A	Terminal Country Code
0X9F26	Application Cryptogram
0X9F36	Application Transaction Counter (ATC)
0X9F37	Unpredictable Number
0X9F5D	Available Offline Spending Amount (AOSA)
0X57	Track 2 Equivalent Data
0X5A	Application Primary Account Number (PAN)

IDTECH Xpress CM100 Technical Reference Manual

0X5F20	Cardholder Name
0X5F24	Application Expiration Date
0X5F2A	Transaction Currency Code
0X5F34	Application Primary Account Number Sequence Number (PSN)
0X82	Application Interchange Profile (AIP)
0X95	Terminal Verification Results (TVR)

Appendix 8: Data Element Format Conventions

a	Alphabetic data elements contain a single character per byte. The permitted characters are alphabetic only (a to z and A to Z, upper and lower case).
an	Alphanumeric data elements contain a single character per byte. The permitted characters are alphabetic (a to z and A to Z, upper and lower case) and numeric (0 to 9).
ans	Alphanumeric Special data elements contain a single character per byte. The permitted characters and their coding are shown in the Common Character Set table in Appendix B of Book 4. There is one exception: The permitted characters for Application Preferred Name are the non-control characters defined in the ISO/IEC 8859 part designated in the Issuer Code Table Index associated with the Application Preferred Name.
b	These data elements consist of either unsigned binary numbers or bit combinations that are defined elsewhere in the specification. Binary example: The Application Transaction Counter (ATC) is defined as —b with a length of two bytes. An ATC value of 19 is stored as Hex '00 13'.
cn	Compressed numeric data elements consist of two numeric digits (having values in the range Hex '0'-'9') per byte. These data elements are left justified and padded with trailing hexadecimal 'F's. Example: The Application Primary Account Number (PAN) is defined as —cn with a length of up to ten bytes. A value of 1234567890123 may be stored in the Application PAN as Hex '12 34 56 78 90 12 3F FF' with a length of 8.
n	Numeric data elements consist of two numeric digits (having values in the range Hex '0'-'9') per byte. These digits are right justified and padded with leading hexadecimal zeroes. Other specifications sometimes refer to this data format as Binary Coded Decimal (—BCD) or unsigned packed. Example: Amount, Authorized (Numeric) is defined as —n 12 with a length of six bytes. A value of 12345 is stored in Amount, Authorized (Numeric) as Hex '00 00 00 01 23 45'.
var	Variable data elements are variable length and may contain any bit combination. Additional information on the formats of specific variable data elements is available elsewhere.

Appendix 9: Error code for response to host

✧	#define ERROR_WITHOUT_ERROR	0xE0
✧	#define ERROR_PARAMETER	0xE1
✧	#define ERROR_LOWOUTBUFFER	0xE2
✧	#define ERROR_CARD_NOT_FOUND	0xE3
✧	#define ERROR_COLLISION_CARD_EXIST	0xE4
✧	#define ERROR_TOOMANY_CARDS_EXIST	0xE5
✧	#define ERROR_SAVED_DATA_NOT_EXIST	0xE6
✧	#define ERROR_NO_DATA_AVAILABLE	0xE8
✧	#define ERROR_INVALID_CID_RETURNED	0xE9
✧	#define ERROR_INVALID_CARD_EXIST	0xEA
✧	#define ERROR_COMMAND_UNSUPPORTED	0xEC
✧	#define ERROR_COMMAND_PROCESS	0xED
✧	#define ERROR_INVALID_COMMAND	0xEE