VX520 Hybrid Solution Specification

Revised 7/19/2018

{ forte } docs

Revision History

Version	Date	Changes
v1	7/19/2018	Initial

© 2018 Forte Payment Systems

Table of Contents

Overview

Introduction

The VX520 point-of-sale (POS) terminal enables partners using Forte's Advanced Gateway Interface API to process EMV transactions for credit cards and non-pin-based debit cards. The following specification provides details on how to integrate your point-of-sale applications with the VX520 terminal.

NOTE: Another way to integrate with the VX520 is by using the Forte Payment's Device Handler which includes a Web version for browser-based POS applications and a Windows DLL version for native, Windows-based POS applications. Each package has it's own documentation.

Requirements

Before starting the integration, ensure you have the following:
- VX520 Terminal with the Forte application configured in Listening Mode
- Forte's Advanced Gateway Interface (AGI) API Specification

Connection Setup

The VX520 terminal is connected to the client machine through a mini-USB cable via the mini-USB port for Windows machines and a USB-to-serial cable via the RS232 port for Mac machines with the following port settings:
- **Baud Rate:** 19200bps
- **Stop bits:** 1
- **Parity:** None
- **Databits:** 8
- **Handshake:** None

Formatting Requests and Responses

Introduction

All requests and response packets to and from the VX520 terminal must use VISA-1 framing. This includes using *Start of Text* (STX) and *End of Text* (ETX) character pairs before and after the message data and appending a longitudinal redundancy check (LRC) that includes the ETX character to the end of the message.

Formatting a Message

All messages between the POS application and the VX520 terminal must be pipe-delimited and VISA-1 framed. All messages begin with the control character, STX (hex 0x02), which tells the external device (i.e., the VX520 terminal or the POS application) that a data transmission is underway. After message transmission, the control character, ETX (hex 0x03), indicates the end of the message. To verify data integrity, the packet should include a longitudinal redundancy check, LRC, which can be used to detect errors that may have been introduced during message transmission and storage. The calculated LRC should include the ETX character, but not the STX character.

Simply put, message packets between the POS application and the VX520 terminal should conform to the following formatting standards:

<STX>message<ETX><LRC><LF>

where,
- STX = hex 0x02 or 2 decimal
- ETX = hex 0x03 or 3 decimal
- LF = hex 0x0A or 10 decimal

NOTE: The LF control character indicates a carriage return or line break.

Calculating the LRC

The LRC included in the message packet acts like a checksum for detecting errors in the transmission of the message. Specifically, the LRC detects errors in parallel groups of bit streams, or transmission blocks. For more information on LRC and its data integrity capabilities, see Wikipedia's *Longitudinal Redundancy Check*.

Formatting Requests and Responses (continued)

Calculating the LRC (continued)

Use the following C# code snippet to calculate the `LRC` value for your messages:

```csharp
private byte calculateLRC(byte[] b)
{
    byte lrc = 0;
    for (int i = 0; i < b.Length; i++)
    {
        lrc ^= b[i];
    }
    return lrc;
}
```

Transaction Flow

Introduction

The following section provides a high-level overview of how transaction data flows from the POS application to the VX520 terminal and how the transaction response returns to the POS application. Users initiate transactions using Forte's Advanced Gateway Interface (AGI) API.

POS Application to Terminal Request Message Flow

Transactions with the VX520 terminal originate from POS applications that are integrated to Forte's AGI API. The message sent to the terminal is pipe delimited with VISA-1 framing (see the *Formatting Requests and Responses* section above) and must contain the following required fields:

- `pg_merchant_id` – The merchant's unique ID, or MID, that identifies the merchant's account

- `pg_total_amount` – The total authorization amount of the transaction to be charged/credited to the customer. This includes the subtotal amount and any taxes, tips, or convenience fees. **NOTE:** Convenience fees, if any, can be calculated and added to the transaction amount by the VX520 terminal or passed in from the POS application. The terminal's configuration depends upon the behavior of the POS calling applications. If you want to use Forte managed convenience fees please discuss it with your sales representative or account manager.

The POS application can send any field listed in the AGI API specification except for the `pg_password` field, which the VX520 transmits. Additionally, the VX520 passes the optional fields included in the request through to the host so that their values can be used for reporting. To ensure a positive customer experience, Forte recommends setting the timeout timer to 2 minutes. This gives the POS application ample time to receive a response from the terminal and gives the customer enough time to insert (dip) or swipe his or her card. Barring any data errors, the actual transaction authorization time will only require a few seconds.

Transaction Flow (continued)

The VX520 terminal always begins a transaction in Listening Mode. When the message data is written to the RS232 or miniUSB Port, the VX520 Forte app reads the data to the end of the packet (i.e., `<ETX>`). Once the POS application begins transmitting a transaction request message, the VX520 terminal completes the following actions:

VX520 Data Flow

1. Validates the incoming `pg_merchant_id` (Invalid MIDs generate `ERR` responses).
2. Checks for the presence of the `pg_total_amount` field.
3. Prompts the customer to dip or swipe his/her credit card.
4. Sends the transaction data to Forte for authorization.
5. Receives the transaction response.
6. Prints a receipt, if configured for receipts.
7. Sends the complete response to the POS application.
8. Reverts back to Listening Mode.

After receiving a transaction response from Forte, the VX520 sends a full response that echoes the submitted request fields in a pipe-delimited format. Merchants can choose whether to print a transaction receipt at the POS application or the terminal by configuring the `#NUMRCPTS` setting when the terminal is in Listener Mode (i.e., `#LISTENERMODE`). **NOTE:** Both the `#NUMRCPTS` and `#LISTENERMODE` settings are internal to the VX520 terminal, which Forte can set on behalf of the partner or merchant.

Terminal to POS Application Response Message Flow

When the `#NUMRCPTS=0,` the receipt must be printed by the POS application. The necessary information to print an EMV-compliant receipt comes from the following fields in the transaction response:

- `pg_emv_cardentrymode`
- `pg_emv_applicationlabel`
- `pg_emv_cvm`
- `pg_emv_aid`
- `pg_emv_tvr`
- `pg_emv_iad`
- `pg_emv_tsi`
- `pg_emv_arc`

Transaction Flow (continued)

Terminal to POS Application Response Message Flow (continued)

Optional information that echoes back in the response can be included on the receipt in the `pg_receiptfooter1`, `pg_receiptfooter2`, etc. and `pg_cfgreceiptfooter1`, `pg_cfgreceiptfooter2`, etc.

When the `#NUMRCPTS=1 or #NUMRCPTS=2`, the terminal prints either one or two copies of an EMV-compliant receipt.

NOTE: `#NUMRCPTS` and `pg_receiptfooterX` are configurable by Forte on behalf of the merchant.

Sample Transactions

Introduction	The following section provides sample request and response messages to and from the POS application and the VX520. These code samples illustrate common scenarios you may encounter with the VX520.

Minimum Data Transaction to VX520 from POS Application

The following message contains the minimum data fields required by the VX520 to process a transaction:

```
<STX>pg_merchant_id=123456|pg_total_amount=1.00|endofdata<
ETX><LRC><LF>
```

Transaction Data to VX520 from POS Application

The following message contains both optional and required transaction data fields passed in from the AGI API specification:

```
<STX>pg_merchant_id=123456|pg_total_amount=1.00|ecom_consu
merorderid=12345|pg_consumer_id=ABCDEFG|ecom_billto_postal
_name_first=Jane|ecom_billto_postal_name_last=Doe|ecom_bil
lto_postal_street_line1=1234 Any
St|ecom_billto_postal_street_line2=Apt
2|ecom_billto_postal_city=Allen|ecom_billto_postal_statepr
ov=TX|ecom_billto_postal_postalcode=75013|ecom_billto_onli
ne_email=jane.doe@gmail.com|pg_line_item_header=SKU,Price,
Qty|pg_line_item_1=021000021,45.00,2|pg_line_item_2=021000
022,36.99,10|pg_line_item_3=021000023,27.50,7|endofdata<ET
X><LRC><LF>
```

Sample Transactions (continued)

The following response message displays a typical approved transaction response. Depending on how you've configured your receipt settings, this response would generate a printed receipt for the customer.

Approved Response Back to POS Application

```
<STX>pg_response_type=A|pg_response_code=A01|pg_response_d
escription=APPROVAL|pg_authorization_code=123456|pg_trace_
number=EB4558C4-4863-4B7F-81F8-
61D57DA20643|pg_avs_code=Y|pg_cvv_code=M|pg_merchant_id=14
7056|pg_emv_data=9F40~7000F0A001;9F02~000000000000;9F03~00
0000000000;9F26~6AAC15913DF4B9D0;4F~A0000000031010;9F06~A0
000000031010;82~1800;9F36~0156;9F34~5E0000;9F27~80;9F39~05
;9F33~E0F8C8;9F1A~0840;9F35~22;95~8080000000;5F2A~0840;9A~
170413;9B~6800;9F21~132005;9C~00;9F37~6117FD1A;5F2D~656E;5
F34~01;84~A0000000031010;9F10~06010A03A00000;9F09~008C;9F0
D~FC50AC8800;9F0E~0000000000;9F0F~FC70BC9800;57~XXXXXXXXXX
XX9736D181120115141671901F;5A~XXXXXXXXXXXX9736;5F20~485552
53542F5041554C20;|ecom_billto_postal_name_first=Kristie|ec
om_billto_postal_name_last=Hamm|pg_transaction_type=10|pg_
total_amount=1.50|pg_card_type=VISA|VX_LAST_4=************
9736|expdate_year=18|expdate_month=11|pg_emv_cardentrymode
=CHIP|pg_emv_applicationlabel=Capital One
Visa|pg_emv_cvm=SIGN|pg_emv_aid=A0000000031010|pg_emv_tvr=
8080000000|pg_emv_iad=06010A03600000|pg_emv_tsi=6800|pg_em
v_arc=01|pg_sequencenumber=002|pg_receiptfooter1=I agree
to pay the above total amount|pg_receiptfooter2=according
to the card issuer agreement|pg_receiptfooter3=(Merchant
agreement if credit voucher)|pg_cfgreceiptfooter1=We
appreciate your payment!|pg_cfgreceiptfooter4=Thank You
Very Much!|endofdata<ETX><LRC><LF>
```

The following response message displays a typical decline transaction response.

Declined Response Back to POS Application

```
<STX>pg_response_type=D|pg_response_code=U83|pg_response_d
escription=AUTH DECLINE|pg_trace_number=0E573D8E
-ED92-433F-BADD-CB479C6C4EC1|pg_avs_code=N|
pg_cvv_code=N|pg_merchant_id=147056|ecom_billto_postal_nam
e_first=ACCOUNT|ecom_billto_postal_name_last=HOLDER|pg_tra
nsaction_type=10|pg_total_amount=19.83|pg_card_type=VISA|V
X_LAST_4=************1111|expdate_year=18|expdate_month=05
|endofdata<ETX><LRC><LF>
```

Sample Transactions (continued)

Declined Response Back to POS Application – Cancellation at Terminal

The following response message displays a decline transaction response that originates from the customer cancelling the transaction at the terminal. Timeouts will also generate this type of decline response.

```
<STX>pg_response_type=D|pg_response_code=CAN|pg_response_d
escription=CANCELLED|endofdata<ETX><LRC><LF>
```