

## Development Guide

# Pump Control Center Protocol

*(Protocol for Pump Interface and Card Reader Interface)*

General Protocol.....	1
Introduction to PCC.....	1
Table of PCC and Card Reader Interface.....	2
Protocol Command Format.....	3
Pump Commands.....	4
Authorize.....	4
A — <i>Initiates a fueling operation</i> .....	4
Sale Information.....	9
B — <i>Reads the sale data or clear the Sale Ready flag</i> .....	9
Stop.....	13
C — <i>Terminates action at one or all fueling positions</i> .....	13
Resume.....	14
D — <i>Restarts action at one or all fueling positions</i> .....	14
Error.....	15
E — <i>Reads queued pump, reader or system errors</i> .....	15
Status Request.....	19
F — <i>Reads the status of the system and each fueling position</i> .....	19
Reset.....	27
G — <i>Resets the data and configuration of the controller</i> .....	27
Blend.....	28
H — <i>Sets blend ratio in the dispensers</i> .....	28
Polled Totals.....	30
I — <i>Reads totals from requested dispenser and hose number</i> .....	30
PPU.....	32
J — <i>Reads or sets the price per unit on the fueling position</i> .....	32
System Version.....	34
K — <i>Returns software versions of sections of controller</i> .....	34
Card Reader Commands.....	36
Keypad Configuration.....	36
Z — <i>Used to configure keypad layout of a specific card reader</i> .....	36
Card Reader Key Position Layouts.....	39
Layout: Gilbarco CRIND™.....	39

---

Layout: Gilbarco InfoScreen.....	39
Layout: Bennett SPM.....	40
Layout: Tokheim Premier C Series In-Site.....	41
Layout: Tokheim DPT™.....	42
Layout: Tokheim Premier DPT™.....	42
Layout: Wayne.....	42
Reader Status.....	43
Y — <i>Retrieves the status indicators pertaining to the card reader section of controller.</i> .....	43
Key Queue Control.....	46
X — <i>Reads and clears entries in the key queue.</i> .....	46
Card Queue Control.....	49
W — <i>Reads and clears entries in the card queue.</i> .....	49
Cash Queue Control.....	51
V — <i>Manages the outgoing and incoming cash queues.</i> .....	51
Print Queue Control.....	56
U — <i>Sends a print job to the printer through a queue.</i> .....	56
Display Queue Control.....	59
T — <i>Sends data to the display unit.</i> .....	59
Key Entry Control.....	59
S — <i>Activates keyboard and specifies type of keyboard input allowed.</i> .....	60
Preloadable Messages.....	62
R — <i>Accesses coded display messages.</i> .....	62
Beeper Control.....	64
Q — <i>Activates the beeper for specified number of beeps.</i> .....	64
Packet Transfer.....	65
P — <i>Passes packets of information to and from the card reader.</i> .....	65
Tank Monitor Commands.....	67
Start Report.....	67
*A — <i>Used to start a tank monitor report on a specific dispenser.</i> .....	67
Get Report Status.....	68
*B — <i>Used to get the report status on a specific tank.</i> .....	68
Get Report.....	69
*C — <i>Used to get the report on a specific tank.</i> .....	69
Car Wash Commands.....	71
Status Request.....	71
*D — <i>Used to get the status on a specific Car Wash.</i> .....	71

---

---

Get Operator Status.....	72
*E — Used to get the operator status on a specific Car Wash.....	72
Info Request.....	73
*F — Used to request information from a specific Car Wash.....	73
Extended Pump Commands.....	73
Extended Pump – Set Decimal Point Modifier.....	74
!O — Used to set a Decimal Point Modifier for data transfer.....	74
Extended Authorize.....	75
!A — <i>Initiates a fueling operation</i> .....	75
Extended Polled Totals.....	80
!! — <i>Receives totals from the dispenser</i> .....	80
EMV Reader Commands.....	82
EMV Set Preset Amount.....	82
*Q — <i>Set Preset Amount for EMV Transactions</i> .....	82
Extended Reader Status.....	83
*Y — <i>Retrieves additional status data needed for EMV transactions</i> .....	83
PIN Entry Data.....	85
*P — <i>Retrieves or Clears PIN Entry data needed for EMV transactions</i> .....	85
EMV Application Data.....	87
*R — <i>EMV Application Data</i> .....	87
Sound Payments Transaction Que.....	96
*S — <i>Sound Payments Transaction Que</i> .....	96
Generic FDSsharp Commands.....	101
Get License Info.....	101
*T — <i>Get License Info</i> .....	101
Dispenser Specific Information.....	102
Reader Specific Information.....	103

## Development Guide

# Pump Control Center Protocol

### HISTORY OF DOCUMENTATION CHANGES & REVISIONS

#### Version 2.3 — July 29, 1996

Pump Interface Section, Command Descriptions, *Key Queue Control*:  
DES encryption updated.

Card Reader Interface Section, *PCC Card Reader Keyboard Layouts*:  
Added "Wayne Keyboard" Layout.

Card Reader Interface Section, *Reader Specific Information, Gilbarco*:  
Added "Scroll" flag.

#### Version 2.4 — March 26, 1997

Pump Interface, Command Descriptions, *Sale Information Command*:  
Corrected decimal point in value of Volume of Sale (to 0000000.000).

Pump Interface, Command Descriptions, *Error Command*:

- Under Reader Codes System, added Dynamic Command Error and Queue Overflow Error.
- Under Pump Operation Codes, added Bank Error

Card Reader Interface, Command Descriptions, *Card Queue Control*  
Under Read Operation, added data for implementing Track 3.

Card Reader Interface, Command Descriptions, *Preloadable Messages Command*  
Reader numbers defined for both ports, port 1, and port 2.

PCC Card Reader Keyboard Layouts  
Added Schlumberger SAM keyboard layout.

Reader Specific Information  
Added information for Wayne/Dresser and Schlumberger readers.

**Version 2.5 — May 9, 1998**Pump Commands, *Error Command*

Under Reader Specific Codes, added Packet Queue Flush Error.

Pump Commands, *Authorize Command*

Under Flag Operation, added new limit (9).

Pump Commands, *Sale Information Command*

Under Extended Sale Field Tags, added Preset Limit (L)

Pump Commands, *Error Command*

1. Under Pump Codes, changed Reserved Bits to 06-07, added new Preset Limit (08).
2. Under Reader Specific Codes, added Dynamic Queue Flush (23) and Requires Reader Configuration (09).

Pump Commands, *System Version Command*

Added Extended System Version Command.

Pump Commands, *Code Download Command*

Incorporated PCC Download documentation.

## Card Reader Commands

Moved Card Reader Interface Comments to appropriate command sections.

Card Reader Commands, *Keyboard Configuration Command*

1. Added notes to Purpose of Keyboard Configuration Command.
2. Moved Card Reader Key Position Layouts to this section.

Card Reader Commands, *Reader Status Command*

Under Printer status Bits, changed some bits and further defined bits.

Special Commands, *Block Storage Command*

Moved from Card Reader Commands

**Version 3.0 — February 1999**

Revised to unify all Dispenser Development Guide documents.

**Version 3.1 — June 1999**

Pump Commands, *Status Request*, added Select Status Request.

**Version 4.0 — February 2000**

Pump Commands, *Authorization Command*, added New Limit feature and Multi-grade Lock Authorization.

*Status Request Command*, added Select Status.

*Cash Queue Control*, full implementation of features.

**Version 4.1 — March 2000**

Removed *Block Storage Command*. Changed command structure sequence for *Cash Queue Command*.

**Version 5.0 — July 2002**

Pump Commands, *PPU Command*, added Flag Operations E and D.

Card Reader Commands, *Key Entry Control Command*, added cursor display control to string.

Reader Specific Information, Gilbarco, added Gilbarco InforScreen Specific Control.

**Version 6.0 — February 2005**

Card Reader Commands, Keypad Configuration, added Tokheim Premier C Series In-Site keypad configuration.

**Version 6.1 — April 2005**

Minor corrections.

**Version 7.0 — April 2011**

Authorize Command and Authorize Command Examples

**Version 7.1 — July 2021**

Tank Monitor Commands

Car Wash Commands

**Version 7.2 — January 2022**

Extended Pump Commands: *Set Decimal Point Modifier* and *Extended Authorize*

**Version 7.3 — April 2022**

Extended Reader Status, PIN Entry Data and EMV Set Amounts commands

**Version 7.4 — July 2022**

EMV Application Data and Extended Totals commands

**Version 7.5 — July 2022**

State of Charge added to Sale Information command

**Version 7.6 — August 2022**

Sound Payments Transaction Que command

**Version 7.7 — December 2022**

EMV configuration command  
Get License Info command

**Version 7.8 — September 2023**

Add Bennett Reader keypad layout information  
Remove Schlumberger information

**Version 7.9 — October 2023**

Update “Diagram: Response to Select Status Request Command” to show correct status

## **PURPOSE OF THIS DOCUMENT**

This manual describes in detail the protocol used to with the Pump Control Center systems.

Instructions for installing or servicing PCC controllers, electronic fuel pumps or dispensers and POS terminals are not included. Additional documentation is available from PIE for hardware installation. For more detail on any product not manufactured by PIE, always refer to that product's accompanying documentation.

## **NOTICE**

Progressive International Electronics reserves the right to revise and improve this document as required. This publication details our Pump Control Centers at this time, and may not accurately describe these products at all times in the future. Specifications are subject to change without notice.

## **COPYRIGHT**

Copyright © 2002 Progressive International Electronics, Inc. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of Progressive International Electronics, Inc.

## Development Guide

# Pump Control Center Protocol

### EXPLANATION OF DOCUMENT STANDARDS

The following documentation standards are applied throughout this document.

**/** Comments are noted in *italics*.

**X** Variable data formats are represented by X(s).  
*Dollar fields are described by \$. Data fields are described by v. The required fixed length and format of these fields is indicated in each command format. Decimals are implied only.*

**Spaces** In formats for commands, spaces between fields are used for clarification only.  
*Spaces should not be included in the actual string sent to the controller.*

**[ ]** Numbers shown in square brackets represent hex values, such as [0x20].

These abbreviations are used.

**MOP** Method of payment.

**MPD** Multiple product dispenser.

**PCC** Pump Control Center. Term used collectively to refer to the PCC systems manufactured by Progressive International Electronics.

**PIE** Progressive International Electronics, Inc.

**POS** Point of sale terminal.

**SPD** Single product dispenser.

General Protocol  
**Introduction to PCC**  
Pump and Card Reader

This document contains the protocol used to communicate with PCC systems.

## Pump Commands

The entire process of controlling fueling pumps is accomplished with a few commands to the PCC. This protocol may be used to control up to 99 fueling positions with eight grades on each position. Currently, the PCC systems support only 32 fueling positions with eight hoses per position.

## Card Reader Commands

The Card Reader command set is used with the island card readers located in the fuel dispensers. Sending a keyboard layout activates the PCC Card Reader Interface. Each reader must also be sent a printer header and footer message. Commands are passed to and from the reader in "queues." Each queue entry contains enough information to complete the command. Entries are processed in chronological order.

The Pump Control Center Protocol uses a simple "2's" complement check byte. Command and response data are transferred in a formatted frame, beginning with a **Start of Text** (ASCII STX [0x02]), followed by the command and data or response, followed by **End of Text** (ASCII ETX [0x03]) and the check byte. All data (except the check byte) are ASCII characters. All commands are one character. The pump number is comprised of two characters; the hose number is one character. All commands are "ACKed" (0x06) "NAKed" (0x15); however, the responses are not. If the controlling device finds an error in the response, simply request the response again.

General Protocol  
**Table of PCC and Card Reader Interface**

Pump Commands		Card Reader Commands	
A	Authorize	Z	Keyboard Configuration
B	Sale Information	Y	Reader Status
C	Stop	X	Key Queue Control
D	Resume	W	Card Queue Control
E	Error	V	Cash Queue Control
F	Status Request	U	Print Queue Control
G	Reset	T	Display Queue Control
H	Blend	S	Key Entry Control
I	Polled Totals	R	Preloadable Messages
J	PPU	Q	Beeper Control
K	System Version	P	Packet Transfer
L	Code Download		
M	Event Mode Control		
!A	Extended Authorize		
!B	Extended Sale Information		
!!	Extended Polled Totals		
!O	Decimal Point Modifiers		

## Protocol Command Format

STX CMD [Pump#] [Hose#] [ . . . Data . . . ] ETX CD

STX	0x02
CMD	Command Code (one character)
Pump #	Fueling Position (two characters)
Hose #	Grade Number (one character)
Reader #	Reader Position (two characters)
Flag	Flag Digit (one character)
Data	Programming Data or Action
ETX	0x03
CD	Check Digit

*The check digit is constructed by adding all of the characters of the string, starting with the STX and ending with the ETX, then subtracting the value from "00" and sending the result. The check digit is masked off to seven (7) bits. Pump and reader numbering are from 01 - 32. A reader number of "00" is used to indicate global addressing for those commands that support this feature.*

## Pump Commands

**Authorize***A — Initiates a fueling operation***Authorize Command Character A****Purpose of Authorize Command**

Initiates a fueling operation. A limit may be set to dollar, volume amount or no limit (Fillup). If a Fillup operation is requested, the dollar and volume fields are ignored. All decimal points are implied rather than actually being sent. A hose number of zero allows any hose to be authorized.

New Limit and Multi-Grade Lock Authorization features of the Authorize command, as well as the Authorize command process for implementing them, are discussed on the following pages.

The Authorize Command also includes an Extended capability. The purpose of the Extended option is to increase the preset amount. The Extended option 'E' increases amounts from 6 to 10 digits of both Dollar and Volume. This option applies to all flags of the Authorization Command. The implied decimal points remain the same (2 for Dollar and 3 for Volume) when the option 'E' is implemented. Extended Authorization commands are documented in the following Formats for Authorize Command

To deauthorize a fueling position, the Stop and Resume commands are used. First, the Stop command is sent. The Resume command is sent only after a Stop status is indicated for the fueling position.

**Format for Authorize Command**

STX A Pump# Hose# Flag \$\$\$\$.\$\$ vv.vv ETX CD

*In the actual command, decimal points are implied only.*

Extended Format for Authorize Command:

STX A Pump# Hose# E Flag \$\$\$\$.\$\$ vv.vv ETX CD

*In the actual command, decimal points are implied only.*

Authorize Command Character=A

Pump #=XX *(01 to 32)*

Hose #=X *(0 to 8, with 0 authorizing any hose — also see Multi-grade Lock Authorization which follows)*

Flag=X *(Type of authorization — see Description)*

Dollar Limit Amount=\$\$\$\$.\$\$ *(0000.01 to 9999.99)*

Volume Limit Amount=vv.vv *(000.001 to 999.999)*

Description of Flag Operation		
0	Dollar Limit	<i>(Credit price)</i>
1	Dollar Limit	<i>(Cash price)</i>
2	Volume Limit	<i>(Credit price)</i>
3	Volume Limit	<i>(Cash price)</i>
4	Fillup	<i>(No limit)</i>
9	New Limit	<i>(Only dollar preset amount may be changed, not MOP — see New Limit section which follows)</i>

Response to Authorize Command

ACK/NAK only

Authorize Command Example

- Pump 1
- Any Hose
- \$25.00

Command:
STX A 0101002500000000 ETX CD
Response:
ACK

---

## Authorize — Implementing New Limit Feature

### Authorize Command Character A

### Purpose of Authorize Command Implementing New Limit

Progressive International has implemented a feature of the Authorize command which enables the user to change the preset value for a fueling transaction after an Authorize command has been sent to the controller. This New Limit feature is commonly known as Preset-On-The-Fly.

### Format for Authorize Command

STX A Pump# Hose# Flag \$\$\$\$.\$\$ vv.vv ETX CD

#### Flag Operation — New Limit

9 New Limit *(Only preset amount may be changed, not MOP)*

#### Special Considerations:

- Due to the nature of this operation, MOP changes are not permitted.
- New Limit field must be the limit field for which the original preset was sent.
- When using New Limit, all fields must exist.
- Some dispensers can have their presets changed after flow has begun. Caution should be exercised using this option with the amount of fuel presently being dispensed is marginally close to the original preset limit. With this scenario, a possibility for an overrun condition exists.

#### Error Notes:

- A system error 06 will occur if this flag is used without a previous authorization being sent for that pump #.
- A pump error 08 will occur if the controller cannot change the preset due to dispenser-specific situations.

---

## Authorize — Implementing Multi-Grade Lock Feature

### Authorize Command Character A

#### Purpose of Authorize Command Implementing Multi-Grade Lock

To initiate a fueling transaction with specific grade combinations.

*To implement this feature, Hose# must be set to 0x3f. This alerts the controller to expect a bit pattern of grades to restrict. This bit pattern is appended to the normal Authorize command as two bytes representing the hoses to restrict: 0x3X<sub>1</sub> 0x3X<sub>2</sub>*

##### First byte 0x3X<sub>1</sub>

X= bit 0 = grade 5

bit 1 = grade 6

bit 2 = grade 7

bit 3 = grade 8

##### Second byte 0x3X<sub>2</sub>

X= bit 0 = grade 1

bit 1 = grade 2

bit 2 = grade 3

bit 3 = grade 4

#### Format for Authorize Command Implementing Multi-Grade Lock

STX A Pump# Hose# Flag \$\$\$\$.\$\$ vv.vv 0x3X 0x3X ETX CD

#### Format for Extended Authorize Command Implementing Multi-Grade Lock

STX A Pump# Hose# E Flag \$\$\$\$.\$\$ vv.vv ETX CD

### Example 1 — Authorize Command Implementing Multi-Grade Lock

grades restricted: 1, 2 & 5	Command:
	STX A Pump# 0x3f Flag \$\$\$\$\$\$ vvvvvv 0x31 0x33 ETX CD
	Response:
	ACK

### Example 2 — Authorize Command Implementing Multi-Grade Lock

grade restricted: 1 only	Command:
	STX A Pump# 0x3f Flag \$\$\$\$\$\$ vvvvvv 0x30 0x31 ETX CD
	Response:
	ACK

### Example 3 — Authorize Command Implementing Multi-Grade Lock

grades restricted: 1 through 8	Command:
	STX A Pump# 0x3f Flag \$\$\$\$\$\$ vvvvvv 0x3f 0x3f ETX CD
	Response:
	ACK

Pump Commands  
**Sale Information**

*B — Reads the sale data or clear the Sale Ready flag*

## Sale Information Command Character B

### Purpose of Sale Information Command

Used to read the sale information or clear the Sale Ready flag. This command may be used at any time during a sale in progress. All known fields will be filled in. Once the sale is complete, the information in this response indicates what actually took place at the fueling position. The R flag indicates a Read Sale operation and the C flag indicates a Clear Sale Ready operation.

*Due to the limitations of some dispensers, the information displayed will not necessarily be what was authorized via the Authorization command.*

### Format for Sale Information Command

STX B Pump# Flag ETX CD

Sale Information Command Character=B

Pump #=XX *(01 to 32)*

Flag=X *(Type of sale information — see below)*

#### Description of Flag Operation

R Read Sale Information

C Clear Sale Ready Status

E Extended Sale Information

*(See Format for Extended Sale Information Command and Response to Extended Sale Information Command on following page(s) )*

*The Clear Sale Ready status may only be accomplished when the Sale Ready status is set.*

---

 Response to Sale Information Command

For Read Operation:

STX Pump# Hose# Flag \$\$\$\$.\$\$ vvv.vvv ETX CD

Pump # =XX	<i>(Fueling position — 01 to 32)</i>
Hose # =X	<i>(Hose number — 0 to 8)</i>
Flag =X	<i>(Type of sale, cash or credit — 0 or 1)</i>
Dollar Amount = \$\$\$\$.\$\$	<i>(Amount, using 2 decimal points)</i>
Volume Amount = vvv.vvv	<i>(Amount, using 3 decimal places)</i>

**Description of Flag Indicators**

0	Credit Sale
1	Cash Sale

For Clear Operation:

ACK/NAK only

For Extended Sale Operation:

*(See Command Format and Response which follow.)*

## Format for Extended Sale Information Command

STX B Pump# Tag(s) ETX CD

*In the actual command, decimal points are implied only.*

Description of Extended Sale Field Tags		
Tag	Tag Description	Tag Response Data Format
H	Hose Number	X (0 to 8)
M	MOP of Sale	X
\$	Dollar Amount of Sale	\$\$\$\$\$\$\$\$\$\$
V	Volume of Sale	vvvvvvvvv
P	Unit Price of Sale	\$\$\$\$\$\$
S	Current Pump Status	XX <i>Refer to Status Request Command</i>
T	Pump Polled Totals	Fvvvvvvvvv\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$ <i>(Above data returned in following order: F= Totals type, 10 digits of volume totals, 10 digits of credit dollar totals, and 10 digits of cash dollar totals)</i>
L	Preset Limit	\$\$\$\$\$\$
C	State of Charge	XXXXXXXXX (formatted as XXX.XXXXXXX) <i>This value is only used for Hydrogen dispensers.</i>

## Response to Extended Sale Information Command

When a Sale Information Command is issued with an E tag, it is treated as an extended sale command and will return data in the following format:

STX Pump#Tag1Data1 . . . Tag<sub>n</sub>Data<sub>n</sub> ETX CD

*The Extended Sale Information Command will return the requested information in the order in which the fields were requested. Each field is a fixed length as shown in the Tag Response Data Format column of the table above.*

### Sale Information Command Example

- Pump 1
- Read
- Hose 2
- Cash
- \$25.75 sale
- 24.315 gal

Command:
STX B 01R ETX CD
Response:
ACK STX 0121002575024315 ETX CD

### Extended Sale Information Command Example

- Pump 1
- Hose 1
- Credit
- \$15.00 sale
- 14.164 gal
- Pump idle
- 3 Cash & Credit Totals
- Vol. Totals  
1234567.890
- Credit Dollar Totals  
\$54321543.21
- Cash Dollar Totals  
\$12312312.35
- State of Charge  
99.1234567%

Command:
STX B 01EHM\$VPSTLC ETX CD
Response:
ACK STX 01H1M0\$0000001500V0000014164P001059S02 T3123456789054321543211231231235L005000 C0991234567 ETX CD

Pump Commands

**Stop**

*C—Terminates action at one or all fueling positions*

Stop Command Character C

Purpose of Stop Command

Used to terminate the action at one or all fueling positions. Once this command is issued, a Resume command must be issued to clear the Stop status. A fueling position of '00' indicates an All Stop operation.

*When Stop command is initiated, the fueling position may or may not go to an End of Sale status, depending upon the brand and type of dispenser.*

Format for Stop Command

STX C Pump# ETX CD

Stop Command Character=C

Pump # =XX

*(01 to 32)*

Response to Stop Command

ACK/NAK only

Stop Command Example

- Stop
- Pump 5

Command:
STX C05 ETX CD
Response:
ACK

## Pump Commands

**Resume**

D — *Restarts action at one or all fueling positions*

## Resume Command Character D

## Purpose of Resume Command

Restarts action at one or all fueling positions. A fueling position of '00' indicates an All Resume operation. This command is used in response to a Stop status.

*The Resume command will allow the fueling position to resume dispensing after a Stop command has been issued only if the pump handle has not been hung up.*

## Format for Authorize Command

STX D Pump# ETX CD

Resume Command Character=D

Pump # =XX (01 to 32)

## Response to Resume Command

ACK/NAK only

## Resume Command Example

- Resume
- Pump 5

Command:
STX D05 ETX CD
Response:
ACK

## Pump Commands

**Error**

*E — Reads queued pump, reader or system errors*

**Error Command Character E****Purpose of Error Command**

Reads queued errors related to the dispensers or system and is useful in diagnosing problems with dispenser operations. Each error response includes the fueling position number (00=System Error) and an error code. Errors should be read and then cleared one at a time. If the Error Queue status bit is still set, another error is queued.

Always read and then save errors to a log file which is date and time stamped. When seeking technical assistance from Progressive International Electronics, this log file enables PIE to identify problems more quickly and accurately.

*If the Error Command is issued with a flush flag (C), all error information is cleared.*

**Format for Error Command**

STX E Flag ETX CD

Error Command Character=E

Flag=X

*(Operation type — see below)*

**Description of Flag Operation**

R	Read the Error
E	Extended Error Read
S	Extended String Read <i>(Returns a data string 25 bytes long)</i>
W	Clear the Top Error
C	Flush the Error Queue

## Response to Error Commands

Clear/Flush Error Response:

ACK/NAK only

Read Error Response:

STX Pump# EC ETX CD

Pump # = XX

*(01 to 32)*

EC = XX

*(Error Type — see descriptions)*

Extended Read Error Response:

STX Pump# EC EX ETX CD

Pump # = XX

*(01 to 32)*

EC = XX

*(Error Type — see descriptions)*

EX = XX

*(Extended Error Code)*

Extended String Read Error Response:

STX Pump# String ETX CD

Pump # = XX

*(01 to 32)*

String

*(Data string 25 bytes long)*

## Description of Error Codes

## System Codes

- 05 Invalid Command Received
- 06 Authorization Failed

## Dispenser Codes

- 01 Unit price on fueling position incorrect; price indicated on the fueling position is not the price sent to it
- 02 Dispenser did not stop at preset amount; dispenser "overran" the limit sent to it; should not use this dispenser for preset operations
- 03 Invalid data received from dispenser; cannot get valid data from dispenser; sales or totals data may not be valid
- 04 Communication down for this dispenser; dispenser is not responding to commands and had been working
- 05 Invalid dispenser status; dispenser is returning invalid status
- 06-07 Reserved
- 08 New preset error

**Reader Codes System (40)**

- 50 Reader command error
- 51 Card queue overflow error
- 52 Cash queue overflow error
- 53 Key queue overflow error
- 54 Display queue overflow error
- 60 Print queue overflow error
- 61 Print command error
- 70 Numeric mode error
- 80 Dynamic command error
- 81 Dynamic queue overflow error

**Reader Specific Codes (41 - 61)**

- 04 Reader communication down
- 05 Card reader command error
- 09 Requires reader configuration
- 50 Reader card error

*If the following entries occur, the queue entry is aborted:*

- 14 Key configure queue flush error
- 15 Print queue flush error
- 16 Display queue flush error
- 17 Dynamic queue flush error

*These values will be returned as ASCII values representing their HEX equivalents.*

**Error Command Example**

- Reader 10
- Print queue error

Command:
STX EE ETX CD
Response:
STX 4A 15 [43] ETX CD

## Pump Commands

# Status Request

F — *Reads the status of the system and each fueling position*

## Status Request Command Character F

### Purpose of Status Request Command

Reads the status of the system and each fueling position. The status indications are "bit" oriented. The first status in the response is the System Status. The remainder of the statuses are for all 32 fueling positions and are returned in order — 1 to 32. The Standard Extended Status (E flag) consists of two ASCII characters. The lower nibble (4 bits) of the character contains the status bits. The upper nibble is always a '3' (0011 binary). *See Select Status (1 flag) description for return value information.*

### Format for Status Request Command

STX FE ETX CD

Status Request Command Character F

Status Flag Character Definitions:

E = Standard Extended Status

1 = Select Status

### Response to Status Request Command

STX Ss Pp Pp . . . . Pp ETX CD

*69 characters are transferred (including STX, ETX & CD) for the E'xtended status. See System Status Definitions, Description of System Status, Pump Status Definitions, and Description of Pump Status on following pages.*

---

### System Status Definitions

Event Mode	Indicates event mode has been selected for this port. <i>See Event Mode Command M.</i>
All Comm Down	Indicates the communications link to all dispensers is down. In half duplex systems, this bit indicates that the transmit character is not even being echoed.
Error Queued	Indicates an error is stored in the error queue and needs to be retrieved.
Controller Reset	Indicates that the controller is reset and all information has been cleared. The controller requires dispenser initialization.
Emergency Stop	Indicates an emergency stop has been issued and has been sent to the dispensers.

### Description of System Status

S	= Bit 3 =	Event mode active
	Bit 2 =	Reserved
	Bit 1 =	Card reader status changed
	Bit 0 =	Communication down for all dispensers
s	= Bit 3 =	Error in the error queue
	Bit 2 =	Controller has completed a reset and requires configuration
	Bit 1 =	Reserved
	Bit 0 =	Emergency stop sent to dispensers

### Dispenser Status Definitions

Stop	Indicates that a Pump Stop command was sent by the POS or that the dispenser is in a stopped state. This bit can only be cleared by issuing a Resume command from the POS.
Drive Away	Asserted when the dispenser dollar value has not increased for a period of seven (7) seconds. This bit may go on and off many times during a sale. Its purpose is to alert the store clerk to check for a possible drive-away situation.
Flowing	Indicates the sale dollar amount is not zero. Some dispensers are unable to indicate the sale flowing, in which case this bit will not be asserted.
Sale Complete	Indicates a completed sale and that all information is collected and available, including dollar and volume amounts as well as updated polled pump totals.
Auth Allowed	POS has sent an Authorization command to a fueling position. The authorization will be held until a handle signal is received from the dispenser.
Auth Sent	Pump has been sent an Authorization command and dispenser is armed.
Logged On	Indicates an active communication for this dispenser. This bit may be disabled if a communication error is detected. At this point, this bit will remain off until communication is reestablished.
Handle	Indicates the current handle position for this dispenser.
Customer Made Selection	Indicates customer has selected one of many different options at the dispenser, such as cash/credit, push-to-start, grade-select, etc. This bit is dispenser/application dependent.

## Description of Extended Flag (E) Dispenser Status

- P = Bit 3 = Dispenser has been sent a Stop command  
Bit 2 = Drive away — *dispenser flowing and amount has not increased for 7 seconds*  
Bit 1 = Dispenser is dispensing fuel  
Bit 0 = Sale is complete and ready to be read
- p = Bit 3 = Controller allowed to authorize dispenser  
Bit 2 = Authorization sent to the dispenser  
Bit 1 = Dispenser logged on — *dispenser configured and communication established; bit will reset if communication for this dispenser is lost and then set when communication is restored*  
Bit 0 = A dispenser handle is lifted and a request for service is active

### Description of Select Flag (1) Dispenser Status

This status has the hose number embedded in the upper nibble to the second dispenser status byte. There are 10 handle conditions — No Handle, Unknown Handle, and Handle 1-8. No Handle occurs when no handle is lifted. Unknown Handle occurs when a handle is lifted, but the number is unknown. Handle 1-8 reflects the handle selected.

P = Bit 7 =	1 (always returned)
Bit 6 =	Reserved
Bit 5 =	Reserved
Bit 4 =	Customer made selection
Bit 3 =	Pump stopped
Bit 2 =	Drive away
Bit 1 =	Flowing
Bit 0 =	Sale complete
p = Bit 7 =	Handle Bit 8 1
Bit 6 =	Handle Bit 4 1 binary handle number bits
Bit 5 =	Handle Bit 2 1 <i>Refer to Handle Bit Definitions which follow</i>
Bit 4 =	Handle Bit 1 1
Bit 3 =	Authorized allowed
Bit 2 =	Authorized sent
Bit 1 =	Dispenser logged on — <i>dispenser configured and communication established; this bit will reset if the communication for this dispenser is lost and set when communication is restored</i>
Bit 0 =	A dispenser handle is lifted and a request for service is active

### Handle Bit Definitions

#### 8 4 2 1 ← binary coding

1 0 0 1	No Handle — <i>handle bit off</i>
1 0 0 1	Unknown Handle — <i>handle bit on</i>
0 0 0 1	Handle 1
0 0 1 0	Handle 2
0 0 1 1	Handle 3
0 1 0 0	Handle 4
0 1 0 1	Handle 5
0 1 1 0	Handle 6
0 1 1 1	Handle 7
1 0 0 0	Handle 8

### Status Request Command Example

Command:
STX FE ETX CD
Response:
<i>See following diagrams</i>

Diagram: Response to Standard Extended Status Request Command

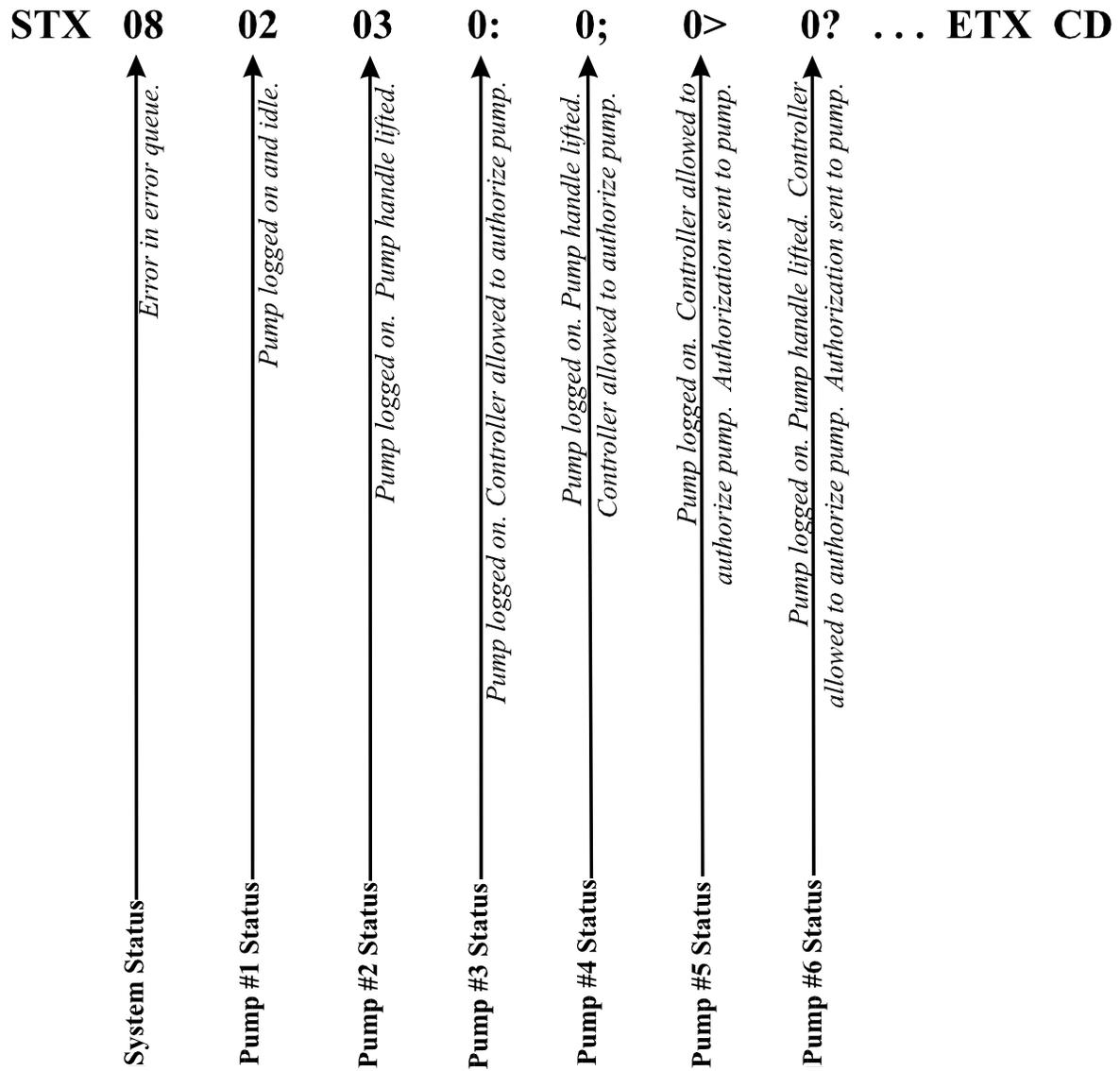
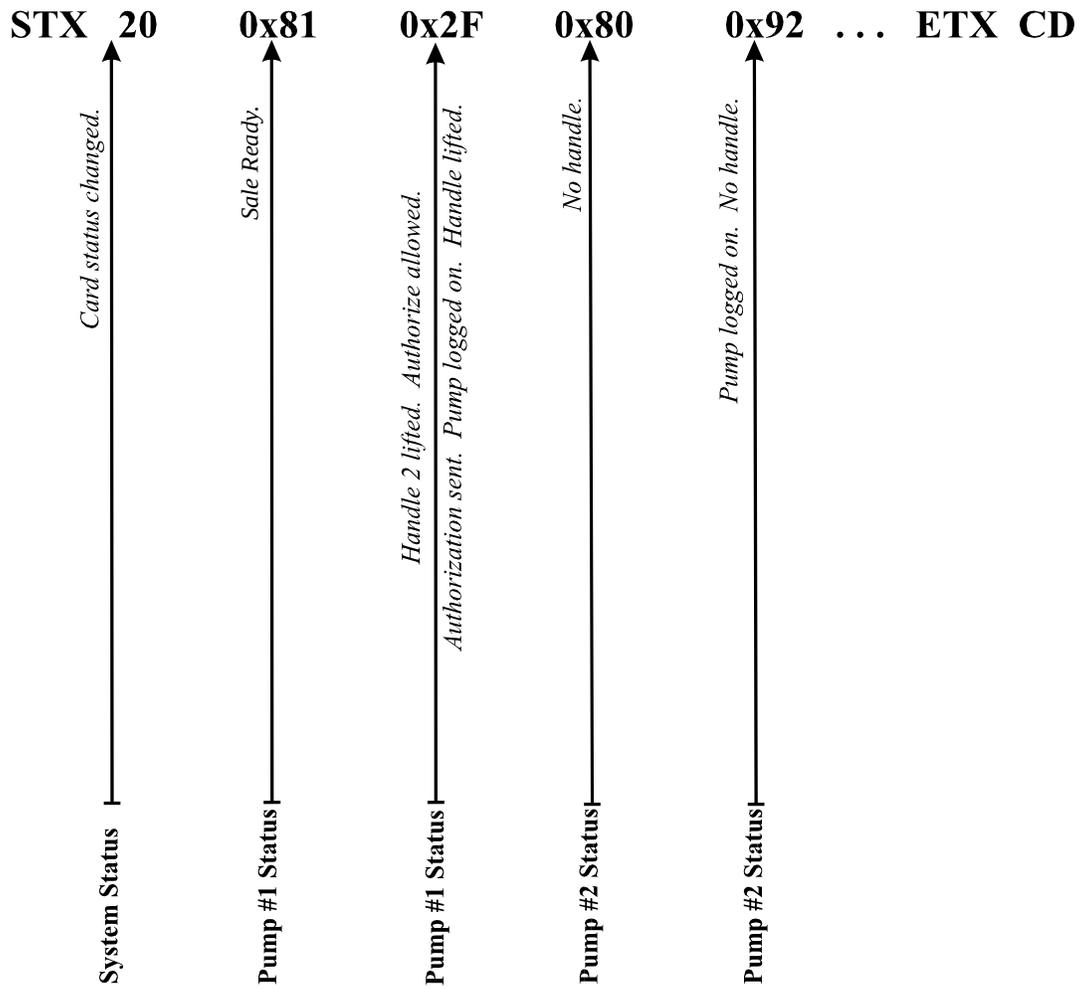


Diagram: Response to Select Status Request Command



## Pump Commands

**Reset**

*G — Resets the data and configuration of the controller*

**Reset Command Character G****Purpose of Reset Command**

Resets the controller. Extreme caution must be exercised when issuing this command. Use of the Reset Command within the POS application is not recommended, since the result is a total reset of all data and configuration information. Also, the Controller Reset status bit (bit 2 of system status two) is set by this command.

*After issuing a Reset Command, the controlling program must configure the controller before any operations can be performed (i.e., send PPU's and/or reader keypad configurations).*

**Format for Reset Command**

STX G ETX CD

**Response to Reset Command**

ACK/NAK only

**Reset Command Example**

Command:
STX G ETX CD
Response:
ACK

## Pump Commands

**Blend***H — Sets blend ratio in the dispensers***Blend Command Character H****Purpose of Blend Command**

Sets the blend ratio in dispensers that allow the controller to adjust the gasoline grade blend ratio. The blend is set by dispenser and hose. The value specified is the percent of Hose 1 used in each hose (i.e., hose 1 = 100% and the last hose = 0%). Blend ratios can be set in 1% increments.

**Format for Blend Command****STX H Pump# Hose# XXX ETX CD**

Command Code Character=H

Pump # =XX (01 to 32)

Hose # =X (0 to 8)

XXX (Percent of Hose #1 ratio -- 001 to 100)

**Response to Blend Command**

ACK/NAK only

### Blend Command Example

- Hose 2
- 50%

Command:
STX 012050 ETX CD
Response:
ACK

Pump Commands  
**Polled Totals**

I — *Reads totals from requested dispenser and hose number*

## Polled Totals Command Character I

### Purpose of Polled Totals Command

Used to read the totals from the requested dispenser and hose number.

*Some dispensers may not have totals. Also, totals cannot be read if the fueling position is Not Idle. Some dispensers can return volume, along with separate cash and credit totals. Other dispensers combine cash and credit totals into one money total. This combined cash/credit money total will be returned by the controller in the credit position and the cash position will be 0.*

*Volume totals returned by dispensers are always combined cash and credit volume totals.*

### Format of Polled Totals Command

STX I Pump# Hose# ETX CD

Polled Totals Command Character=I  
Pump # =XX (01 to 32)  
Hose # =X (0 to 8)

## Response to Polled Totals Command

STX Pump# Hose# Flag vvvvvv.vv XXXXXXXX.XX YYYYYYYY.YY ETX CD

Pump #=XX	(01 to 32)
Hose #=X	(0 to 8)
Flag=X	(Totals type — see following description)
Volume Total=vvvvvv.vv	(0000000.001 to 9999999.999)
Credit Total=XXXXXXX.XX	(00000000.01 to 99999999.99)
Cash Total=YYYYYYY.YY	(00000000.01 to 99999999.99)

### Description of Totals Flag

- |   |                                                           |
|---|-----------------------------------------------------------|
| 0 | Totals not available for this dispenser                   |
| 1 | Pump busy; try later                                      |
| 2 | Cash & Credit combined money totals with combined volume  |
| 3 | Cash & Credit separated money totals with Combined volume |

*If the flag is a 0 or 1, the totals fields will be zero. Decimal points are implied.*

## Polled Totals Command Example

<ul style="list-style-type: none"> <li>• Pump 1</li> <li>• Hose 3</li> <li>• Totals Type 3</li> <li>• Volume 1234512.345</li> <li>• Credit Total \$9875432.10</li> <li>• Cash Total \$5555123.45</li> </ul>	Command:
	STX I 013 ETX CD
	Response:
	ACK STX 0133123451234598765432105555512345 ETX CD

## Pump Commands

### PPU

*J — Reads or sets the price per unit on the fueling position*

## PPU Command Character J

### Purpose of PPU Command

Reads or sets the price per unit on the fueling position. With this command, the controller may be configured. A price must be sent to hose number 1 if the dispenser exists. If multiple hoses are sent prices, the dispenser is considered to be a multi-product dispenser (MPD). Only those fueling positions which exist should be sent prices. Otherwise, the system operation will be slowed drastically. Also, it is good programming practice to send the prices starting with the highest hose on the dispenser and ending with hose number 1. Both price fields must be filled in. If only one price is required, fill both fields with the same price. If a dispenser is to be taken off-line, send a price of "00.00" to hose number one. (Other hoses are optional.)

*For a Read operation, the price fields are ignored and decimal places are implied only.*

### Format for PPU Command

STX J Pump# Hose# Flag XXX.XXX YYY.YYY ETX CD

PPU Command Character=J

Pump # =XX (01 to 32)

Hose # =X (0 to 8)

Flag=X (Type of PPU information — see below)

Credit Pricel=XXX.XXX (000.001 to 999.999)

Cash Total=YYY.YYY (000.001 to 999.999)

#### Description of Flag Operation

R	Read Operation
W	Write Operation
E	Enable PPU with Handle Off Hook
D	Disable PPU with Handle Off Hook

## Response to PPU Command

For Write Operation:

ACK/NAK only

For Read Operation:

STX Pump# Hose# XXX.XXX YYY.YYY ETX CD

Pump # =XX	(Dispenser — 01 to 32)
Hose # =X	(Hose number — 0 to 8)
Credit Price =XXX.XXX	(Credit )
Cash Price =YYY.YYY	(Cash)

### PPU Command Example 1

- Pump 5
- Hose 1
- Credit Price \$1.119
- Cash Price \$1.019

Command:
STX J051W001119001019 ETX CD
Response:
ACK

### PPU Command Example 2 — Enable PPU with Handle Off Hook

- Pump 3
- Hose 2
- Credit Price \$1.129
- Cash Price \$1.029

Command:
STX J032E001129001029 ETX CD
Response:
ACK

Pump Commands  
**System Version**

*K — Returns software versions of sections of controller*

## System Version Command Character K

### Purpose of System Version Command

Returns the software versions of the sections of the controller.

*Each section version is terminated with a NULL character. Actual spaces [20] will be embedded in information returned.*

### Format for System Version Command

STX K ETX CD

### Response to System Version Command

STX Sys Pmp1 Rdr1 Pmp2 Rdr2 ETX CD

System Version Command Character=K

Sys=system version *(format — sys x.xx MM/DD/YY)*

Pmpx=pump version *(format — pmp x.xx MM/DD/YY)*

Rdrx=reader version *(format — rdr x.xx MM/DD/YY)*

### Format for Extended System Version Command

STX KE ETX CD

### Response to Extended System Version Command

Same as above response — PEx.xx (indicating Progressive's core code version).

Format for Extended System Version Command:

STX KE ETX CD

Response to Extended System Version Command:

STX Sys Pmp1 Rdr1 Pmp2 Rdr2 Core Code ETX CD

Core Code PROM Version *(Format — PIE 1.00 MM/DD/YY)*

### System Version Command Example

Command:					
STX KE ETX CD					
Response:					
ACK STX	PCC	4.40	04/17/98	[00]	
	DW	P5.90	04/16/98	[00]	
	DW	C3.80	01/06/98	[00]	
	DW	P5.90	04/16/98	[00]	
	DW	C5.80	01/06/98	[00]	
	PIE	1.00	04/17/98	[00]	ETX CD

## Card Reader Commands Keypad Configuration

*Z — Used to configure keypad layout of a specific card reader*

### Keypad Configuration Command Character Z

Used to configure the keypad layout of the appropriate pump's card reader. It also communicates to the controller that a reader exists at this address. The Keypad Configuration Command clears Key Enable settings.

Various graphic overlays may be used on any given brand dispenser. All card reader keypads are defined by assigning a Keycode to a key position.

*Refer to the following pages for reader key position layouts for specific pump brands. The keypad assignment must represent what the controlling application expects the PCC system to return for a given key entry. All key positions must be assigned. If a key position is unused on the keypad, then a null character [00] must be assigned to this position. To take a card reader off-line, assign a null [00] to all key positions. To put the reader back on-line, send a valid keypad configuration with actual keycodes.*

### Format for Keypad Configuration Command

STX Z Reader# Key<sub>1</sub>Key<sub>2</sub> . . . Key<sub>n</sub> e ETX CD

Keypad Configuration Command Character=Z	
Reader number=XX	(01 to 32)
e	(End of keycode data — hex [65])
Key <sub>n</sub>	(Keycode for position n)

## Predefined Keys

S	Start code	D	Debit
E	Enter code	N	No
L	Clear code	R	Credit
B	Backspace code	A	Cash
C	Cancel code	Y	Yes
H	Help		

*Keys are predefined as upper case. These predefined keys must be used where applicable.*

## Response to Keypad Configuration Command

ACK/NAK only

### Keypad Configuration Command Example

- Reader 1  
Gilbarco CRIND

Command:
STX Z01123[00]R456AZ789NYL0ECe ETX CD
Response:
ACK/NAK

### Keypad Configuration Sample

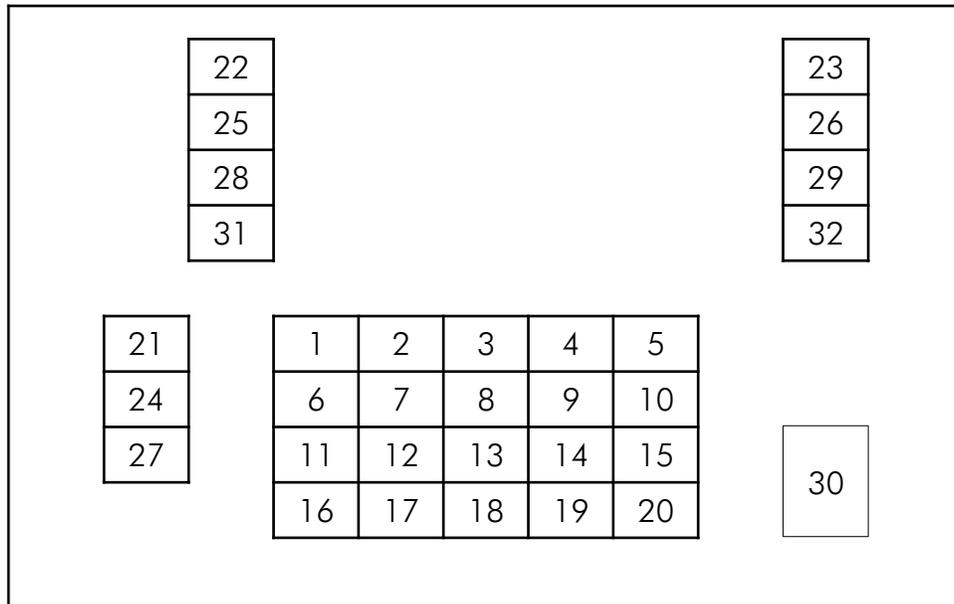
<sup>1</sup> 1	<sup>2</sup> 2	<sup>3</sup> 3	<sup>4</sup> 	<sup>5</sup> R
<sup>6</sup> 4	<sup>7</sup> 5	<sup>8</sup> 6	<sup>9</sup> A	<sup>10</sup> Z
<sup>11</sup> 7	<sup>12</sup> 8	<sup>13</sup> 9	<sup>14</sup> N	<sup>15</sup> Y
<sup>16</sup> L	<sup>17</sup> 0	<sup>18</sup> E	<sup>19</sup> H	<sup>20</sup> C

*Small numbers in the upper left-hand corner of each key indicate the key position in this sample configuration of a Gilbarco CRIND™. (See Card Reader Key Position Layout 1.)*

## Card Reader Key Position Layouts

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20

Layout: Gilbarco CRIND™



Layout: Gilbarco InfoScreen



Layout: Bennett SPM

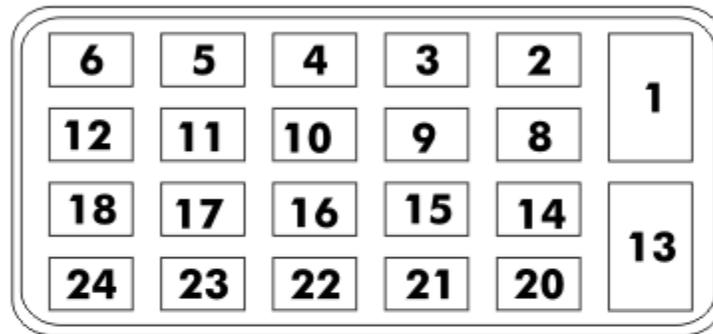
The keypad is fixed and only the 7 soft keys around the screen need to be configured as shown above. The bottom-right key is fixed as Cancel.

Sample Keypad configuration: STUVWXYe

Product Selection Keypad Key Numbers



Operator Interface Keypad Numbers  
(Keypad numbers 7 & 9 are not used)



Layout: Tokheim Premier C Series In-Site

1	7	13	19
2	8	14	20
3	9	15	21
4	10	16	22
5	11	17	23
6	12	18	24

Layout: Tokheim DPT™

6	5	4	3	2	1
12	11	10	9	8	7
18	17	16	15	14	13
24	23	22	21	20	19

Layout: Tokheim Premier DPT™

16	17	18	19	20
11	12	13	14	15
6	7	8	9	10
1	2	3	4	5

Layout: Wayne



---

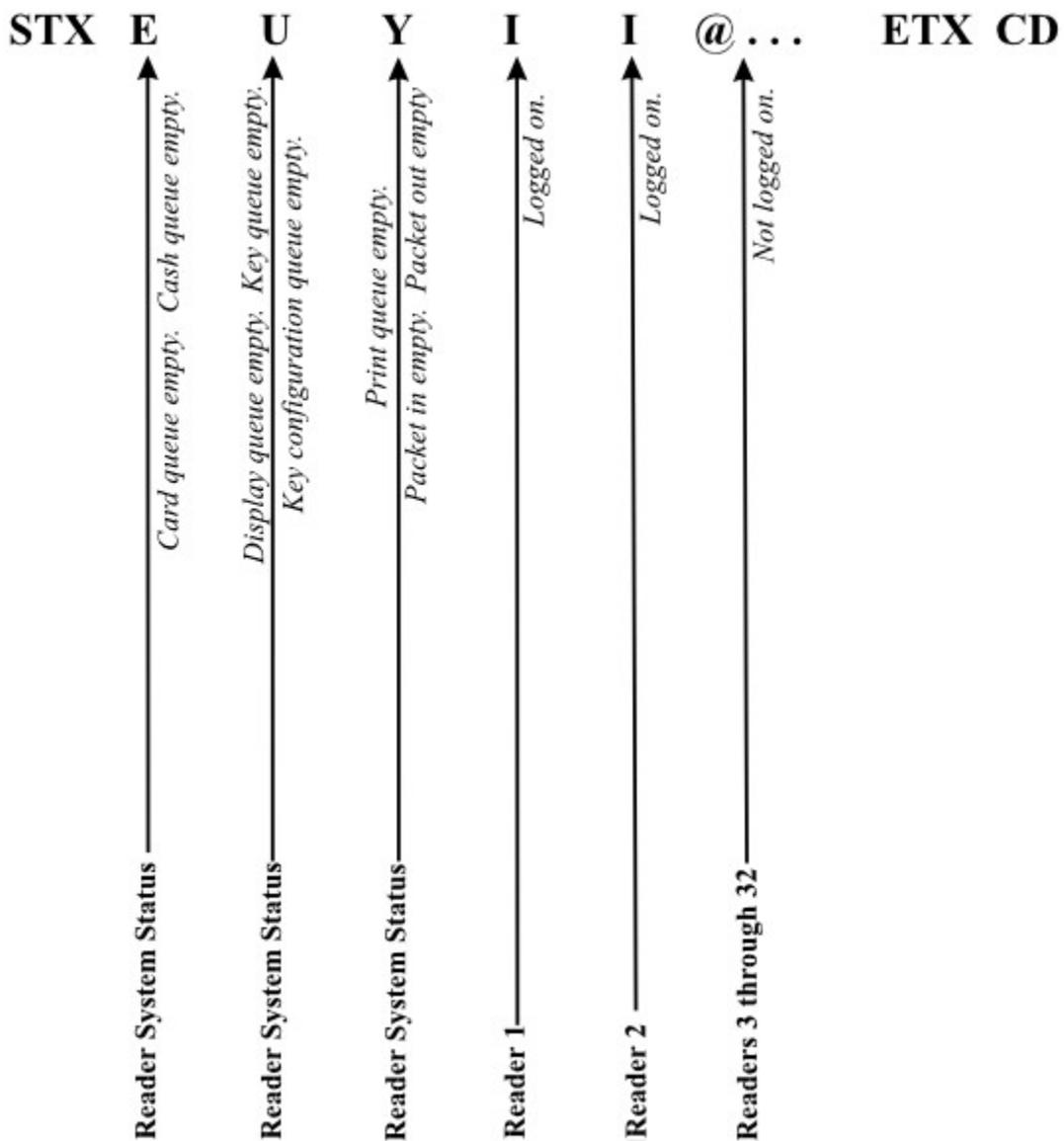
S2	= bit 7	<i>(Don't care)</i>
	bit 6	<i>(1)</i>
	bit 5	<i>(Key Config Queue full)</i>
	bit 4	<i>(Key Config Queue empty)</i>
	bit 3	<i>(Key Queue full)</i>
	bit 2	<i>(Key Queue empty)</i>
	bit 1	<i>(Display Queue full)</i>
	bit 0	<i>(Display Queue empty)</i>
S3	= bit 7	<i>(Don't care)</i>
	bit 6	<i>(1)</i>
	bit 5	<i>(Packet Out full)</i>
	bit 4	<i>(Packet Out empty)</i>
	bit 3	<i>(Packet In empty)</i>
	bit 2	<i>(Print Entry active)</i>
	bit 1	<i>(Print Queue full)</i>
	bit 0	<i>(Print Queue empty)</i>
R	=	<i>(Reader Dependent Status)</i>
	bit 7	<i>(Don't care)</i>
	bit 6	<i>(1)</i>
	bits 5 & 4	<i>(Printer Status — see below)</i>
	0 0	<i>(No Error)</i>
	0 1	<i>(Error)</i>
	1 0	<i>(Paper Out)</i>
	1 1	<i>(Paper Low)</i>
	bit 3	<i>(Printer Idle — 1=Idle)</i>
	bits 2 & 1	
	0 0	<i>(Not Active)</i>
	0 1	<i>(Numeric — No Echo)</i>
	1 0	<i>(Function only)</i>
	1 1	<i>(Numeric Echo)</i>
	bit 0	<i>(Reader Logged)</i>

### Reader Status Command Example

- Readers 1 & 2 logged on
- Readers 3-32 not logged on

Command:
STX Y ETX CD
Response:
<i>See description below</i>

#### Description of Reader Status Command Response



Card Reader Commands  
**Key Queue Control**

*X — Reads and clears entries in the key queue*

## Key Queue Control Command Character X

### Purpose of Key Queue Control Command

Reads and clears entries in the key queue. The reader status will indicate “Key Queue Not Empty”, signaling that there is information in the queue to be read.

*The top entry is not cleared until a “clear” command is issued.*

### Format for Key Queue Control Command

STX X Flag ETX CD

Key Queue Control Command Character=X  
 Flag=X *(Type of operation — see below)*

#### Description of Flag Operation

R	Read Entry
C	Clear Entry

### Response to Key Queue Control Command

Clear Operation:

ACK/NAK only

Read Operation:

STX HH kk... [NULL] ETX CD

HH

*(Reader number — 2 characters)*

k

*(Returned key code)*

## Key Queue Control Command Example

- Read Key Queue
- Keys returned are 54321E

Command:
STX X R ETX CD
Response:
ACK STX 0154321E[00] ETX CD

Card Reader Commands  
**Card Queue Control**

*W — Reads and clears entries in the card queue*

## Card Queue Control Command Character W

### Purpose of Card Queue Control Command

Reads and clears entries in the card queue. The reader status will indicate "Card Queue Not Empty," signaling that there is information in the queue to be read.

*The top entry is not cleared until a "clear" command is issued.*

### Format for Card Queue Control Command

STX W Flag ETX CD

Flag=X

*(Type of operation — see below)*

#### Description of Flag Operation

R	Read Entry
C	Clear Entry

### Response to Card Queue Control Command

Clear Operation:

ACK/NAK only

## Read Operation:

STX HH track 1 [NULL] track 2 [NULL] track 3 [NULL] ETX CD

HH	<i>(Reader number — 01 to 32)</i>
track 1	<i>(Track 1 data terminated by a Null character)</i>
track 2	<i>(Track 2 data terminated by a Null character)</i>
track 3	<i>(Track 3 data terminated by a Null character)</i>

*Return information can be up to 250 characters. If both track 1 and track 2 are Null data, then the card was read with an error.*

## Card Queue Control Command Example

- Reader 1
- Card data  
123451234512345TESTCARDDATA

Command:
STXWR ETX CD
Response:
STX01 123451234512345TEST CARDATA[00] ETX CD

Card Reader Commands  
**Cash Queue Control**

*V — Manages the outgoing and incoming cash queues*

## Cash Queue Control Command Character V

*The Cash Queue Control Command may only be implemented if PCCZ/PCXZ code is Signature 4 or later version.*

## Purpose of Cash Queue Control Command

This command is used to send commands to and retrieve data from the cash acceptor on a dispenser reader. The commands and data are managed via incoming and outgoing queues. *See the reader status command (Y) for information on the status bits associated with this command.*

## Format for Cash Queue Command

There are three fundamental types of Cash Queue command:

- Write — Issues a command to a cash acceptor via the Cash Out queue
- Read — Read the data in the cash acceptor queue via the Cash In queue
- Clear — Clear an entry in the cash acceptor Cash In queue

The command structure for each of these operations is shown in this section.

## WRITE COMMAND STRUCTURE

STX V W HH Flag [optional data] ETX CD

## Description of Flag Operation

*[optional data] field is used to send additional numeric data with certain flags shown below.*

EE	Enable cash acceptor (2 byte flag)
ED	Disable cash acceptor (2 byte flag)
Z	Cash acceptor status request*
S	Stack bill in acceptor**
R	Return bill to customer***
C	Clear accumulated bills
T	Set maximum bill type — use with the following optional data fields:
	001 \$1 bill maximum
	005 \$5 bill maximum
	010 \$10 bill maximum
	020 \$20 bill maximum
	050 \$50 bill maximum
	100 \$100 bill maximum
O	Set bill orientation — use with the following optional data fields:
	B Black side up
	G Green side up
	2 Both directions

\* Forces a cash acceptor status request queue entry. A solicited cash acceptor status request always returns a CB=000.

\*\* Generates two cash acceptor status request queue entries (stacked and idle).

\*\*\* Generates two cash acceptor status request queue entries (returned and idle).

## Response to Write

ACK/NAK only

---

 READ COMMAND STRUCTURE

STX V R ETX CD

Format for Cash In Queue

HH S<sub>1</sub>S<sub>2</sub>S<sub>3</sub>S<sub>4</sub>S<sub>5</sub>S<sub>6</sub>[sp]CB=XXX[sp]AB=YYY[00]

HH	Head number with which data is associated (01 to 32)
S <sub>1</sub>	Status 1 (controller status) — <i>see following Status Return chart</i>
S <sub>2</sub>	Status 2 (controller status)
S <sub>3</sub>	Status 3 (controller status)
S <sub>4</sub>	Status 4 (controller status)
S <sub>5</sub>	Status 5 (acceptor status)
S <sub>6</sub>	Status 6 (acceptor status)
XXX	Current bill in acceptor (3 digit \$ value)
YYY	Accumulated bills in acceptor since last clear (3 digit \$ value)
[sp]	Space character [0x20]

*A solicited cash acceptor status request always returns a CB=000.*

Status Return Values

All values returned as hex values (30h — 46h) representing 0 — 9 and A — F

Controller Status Codes (bit oriented values)

S1	S2	S3	S4	
0	0	0	0	Cash acceptor disabled
x	x	x	1	Cash acceptor enabled
x	x	1	x	Solicited status
•	•	•	•	Reserved values
•	•	•	•	
[h]	[h]	[h]	[h]	

Acceptor Status Codes (numeric values)

S5	S6	
0	0	Note acceptor idle
0	1	Note escrowed
0	2	Note stacked
0	3	Note returned
0	4	Bill rejected
0	5	Acceptor jammed
0	6	Acceptor cassette full
0	7	Acceptor cassette removed
0	8	Acceptor cassette replaced
0	9	Acceptor power up
•	•	
•	•	Reserved codes
•	•	
F	E	Note status unknown**
F	F	Note acceptor error

\*\* *If this status is returned, an additional hex value is returned [hh], showing the value the controller was given*

*x represents a variable bit value*

*[h] represents a hexadecimal digit*

*It is highly recommended that an audit trail file for all cash queue related data (cash queue commands and cash queue responses) be maintained*

### Response to Read

ACK only with data/NAK

*If the command is ACKed, data will follow from the current Cash In Queue entry.*

### CLEAR COMMAND STRUCTURE

STX V C ETX CD

Response to Clear

ACK/NAK

Card Reader Commands  
**Print Queue Control**

U — *Sends a print job to the printer through a queue*

## Print Queue Control Command Character U

### Purpose of Print Queue Control Command

Sends a print job to the printer through a "queue." Multiple strings may be sent to the queue so as to allow large messages to be sent. Each type of job sent must be complete and for only one printer at a time. Each print job is tagged with the reader number and message type.

- *Each string must follow sequential numbering. Maximum string size is 240 characters.*
- *The maximum header or footer job size is 240 characters, including carriage returns, line feeds and spaces. Header and footer messages must be sent for each printer immediately after logging on.*
- *The maximum receipt job size is 700 characters.*
- *Linefeed only (New Line) = [0A]*
- *Do not send papercut character. A papercut will be automatically appended to a footer print job.*
- *Add enough linefeeds after footer text to ensure that complete message is ejected from printer.*
- *00 is used for Header #.*

---

## Format for Print Queue Control Command — Data Strings

STX U nn 'sss....sss'[NULL] dd ETX CD

Print Queue Control Command Character=U

nn *(String number — 2 ASCII decimal digits)*

ss *(Print data)*

dd *(Next string number — 2 ASCII digits)*

## Response to Print Queue Control Command — Data Strings

ACK/NAK only

## Format for Print Queue Control Command — Ending String

STX U FF hh t ETX CD

Print Queue Control Command Character=U

FF *(Ending flag — 2 ASCII 'F' characters)*

hh *(Reader number — 2 ASCII digits)*

t *(Print job type)*

## Response to Print Queue Control Command — End String

ACK/NAK only

### Description of Print Job Types

H	Header
F	Footer
R	Receipt (causes a receipt to be printed)

Description of String Flags

- OO First data string
- nn Subsequent data strings
- FF Ending string

Print Queue Control Command Example

- GLOBAL print header
- Text is "Header Test"

Command:
STX U00 Header Test [00] 01 ETX CD
Response:
ACK
Command:
STX UFF 00H ETX CD
Response:
ACK

Card Reader Commands  
**Display Queue Control**  
*T — Sends data to the display unit*

Display Queue Control Command Character T

Purpose of Display Queue Control Command

Sends data to the display unit.

*The maximum number of display characters is 80. Each display message is complete and assumed to start at the first display position.*

*Line feed only (new line) = 0A<sub>16</sub>*

Format for Display Queue Control Command

STX T HH 'sss....sss' [NULL] ETX CD

HH = Reader Number *(01 to 32 for specific reader or 00 for global — all — readers)*  
 ss = Display Data

Response to Display Queue Control Command

ACK/NAK only

Print Queue Control Command Example

- Display message is test display

Command:
STX T01 TestDisplay[00] ETX CD
Response:
ACK/NAK

Card Reader Commands  
**Key Entry Control**

*S — Activates keyboard and specifies type of keyboard input allowed*

## Key Entry Control Command Character S

### Purpose of Key Entry Control Command

Activates the keyboard and specifies the type of keyboard input allowed.

*Entry can be any one of the following:*

- *Any Key*
- *Numeric with Echo*
- *Numeric Without Echo*

### Format for Key Entry Control Command

STX S HH n e d cETX CD

Key Entry Control Command Character=S	
HH	<i>(Reader Number — 2 characters)</i>
n	<i>(N=normal or #=numeric)</i>
e	<i>(e=echo on or n=echo off)</i>
d	<i>(d=DES encryption on or n=DES encryption off)</i>
	<i>(c=display control s=set cursor for numeric n=no cursor control)</i>

*DES encryption can only be set on numeric entry.*

### Response to Key Entry Control Command

ACK/NAK only

### Key Entry Control Command Example 1

- Reader 1
- Numeric mode with echo
- No cursor control

Command:
STX S01 # e n nETX
Response:
ACK

### Key Entry Control Command Example 2

- Reader 2
- Numeric mode with echo
- With cursor control

Command:
STX S01 # e n sETX
Response:
ACK

Card Reader Commands  
**Preloadable Messages**

R — *Accesses coded display messages*

## Preloadable Messages Command Character R

### Purpose of Preloadable Messages Command

Accesses the reader's series of display messages used for "Print Receipt" and "Card Swipe" functions.

*Programming must be implemented before sending the keyboard configuration command (command character Z).*

#### Print Default Messages

- 1 Printing Receipt
- 2 Receipt Complete
- 3 Please Take Receipt
- 4 Printer Error

#### Card Default Messages

- 1 Remove Card Quickly
- 2 Insert Card Again
- 3 Invalid Card
- 4 One Moment Please
- 5 Card Inserted Wrong/Please Try Again

#### Miscellaneous Display

- 1 Cancel Key Message
- 2 Invalid Key

## Format for Preloadable Messages Command

STX R HH d n 'sssss...ss' [NULL] ETX CD

HH	<i>(Reader Number — 2 characters; see below)</i>
d	<i>(P=Printer Display or C=Card Display or M=Miscellaneous)</i>
n	<i>(Display Message Number — 1 character)</i>
ss	<i>(Display Data)</i>

### Reader Number Port Specifications

00	both ports
01	port 1 only
17	port 2 only

## Response to Preloadable Messages Command

ACK/NAK only

## Preloadable Messages Command Example

- Preload card message 1 to "Remove Card Now"

Command:
STX R01C1 Remove Card Now [00] ETX CD
Response:
ACK

## Card Reader Commands

**Beeper Control**

**Q** — *Activates the beeper for specified number of beeps*

**Beeper Control Command Character Q****Purpose of Beeper Control Command**

Activates the beeper for a specified number of beeps.

*This command utilizes the display buffer queue. Refer to the display queue flags during the processing of this command.*

*Some reader brands do not support selectable beep counts. Best effort will be made to comply with the commands request.*

**Format for Beeper Control Command**

STX Q HH cc ETX CD

Beeper Control Command Character=Q

Reader Number=HH (01 to 32)

Number of Beeps to Sound=cc (01 to 10)

**Response to Beeper Control Command**

ACK/NAK only

**Beeper Control Command Example**

- Reader 1
- Beep 5 times

Command:
STX Q0105 ETX CD
Response:
ACK

## Card Reader Commands

**Packet Transfer**

*P — Passes packets of information to and from the card reader*

**Packet Transfer Command Character P****Purpose of Packet Transfer Command**

Passes packets of information to and from the card reader.

*Contact Progressive International Electronics for further instructions on the implementation of this command.*

**Format for Packet Transfer Command**

Write:

STX P Flag Reader# CCC [ . . . DATA . . . ] ETX CD

Packet Transfer Command Character=P

Reader #=XX (2 characters)

Data Character Count=CCC (3 characters)

Data Transfer data characters —  
maximum 240)

Read/Clear:

STX P Flag ETX CD

**Description of Flag Operation**

W Write

R Read

C Clear

**Response to Packet Transfer Command**

Write:

ACK/NAK only

Read:

STX HH CCC [ . . . DATA . . . ] ETX CD

Clear:

ACK/NAK only

*If a data character is STX, ETX or 10 HEX, it must be preceded by a 10 HEX. This character is not part of the byte count. It will be stripped from the command, but the byte following the 10 HEX is included in the CD and the byte count.*

Tank Monitor Commands  
**Start Report**

*\*A — Used to start a tank monitor report on a specific dispenser*

**Start Tank Monitor Report Command Characters \*A**

**Purpose of Start Tank Monitor Report Command**

Starts a Tank Monitor Report on a specific tank number.

**Format for Start Tank Monitor Report Command**

STX \*A nn ##### ETX CD

Tank Monitor Number=nn (01 to 32)  
 ID Token=##### (00000000 to #####)  
*\*ID Token is a 32 bit unique ID token used to identify each independent outside client.*

**Response to Start Tank Monitor Report Command**

ACK/NAK only

**Start Tank Monitor Report Command Example**

<ul style="list-style-type: none"> <li>• Tank Number 1</li> <li>• ID 0x00253681</li> </ul>	Command:
	STX *A0100253681 ETX CD
	Response:
	ACK

Tank Monitor Commands

## Get Report Status

*\*B — Used to get the report status on a specific tank*

### Get Tank Monitor Report Status Command Characters \*B

### Purpose of Get Tank Monitor Report Status Command

Gets the Tank Monitor Report status using a unique ID.

### Format for Get Tank Monitor Report Status Command

STX \*B ##### ETX CD

ID Token=##### (00000000 to #####)

*\*ID Token is a 32 bit unique ID token used to identify each independent outside client.*

### Response to Get Tank Monitor Report Status Command

STX s ETX CD

Report Status=s (1 to 3)

1 = Report Busy

2 = Report Ready

3 = Report Error

### Get Tank Monitor Report Status Command Example

- ID 0x00253681
- Report Ready

Command:
STX *B00253681 ETX CD
Response:
STX 2 ETX CD

## Tank Monitor Commands

**Get Report***\*C — Used to get the report on a specific tank***Get Tank Monitor Report Command Characters \*C****Purpose of Get Tank Monitor Report Command**

Gets a Tank Monitor Report using a unique ID.

**Format for Get Tank Monitor Report Command**

STX \*C ##### ETX CD

ID Token=##### (00000000 to #####)

*\*ID Token is a 32 bit unique ID token used to identify each independent outside client.***Response to Get Tank Monitor Report Command**

STX nn s m l bbbb iii... A bbbb aaa... L bbbb lll... D bbbb ddd... ETX CD

Tank Number=nn (01 to 32)

Report Status=s (1 to 3)

1 = Report Busy

2 = Report Ready

3 = Report Error

Tank Type=m (0 to 2)

0 = Error

1 = TLS 250

2 = TLS 350+

b – Length of upcoming report (0 to 5000, always 4 characters)

l – Inventory report next (always l)

i – Inventory report (up to 1000 characters)

A – Alarm report next (always A)

a – Alarm report (up to 1000 characters)

- L – Leak report next (always L)
- l – Leak report (up to 200 characters)

*\*Only for TLS 350+ (not returned for TLS 250)*

- D – Delivery report next (always D)
- d – Delivery report (up to 5000 characters)

### Get Tank Monitor Report Command Example

- ID 0x00253681
- Tank Number 1
- Report Ready
- Tank Type TLS 250

Command:
STX *C00253681 ETX CD
Response:
STX 012110005abcdeA0010abcdefghijL0003abc ETX CD

## Car Wash Commands Status Request

*\*D — Used to get the status on a specific Car Wash*

### Request Car Wash Status Command Characters \*D

### Purpose of Request Car Wash Status Command

Gets the Car Wash status of a specific Car Wash.

### Format for Request Car Wash Status Command

STX \*D ii ETX CD

Car Wash ID=ii (00 to 32)

### Response to Request Car Wash Status Command

STX ii uu oo ww ETX CD

Car Wash ID=ii (00 to 32)

Customer Status=uu (always 00)

Controller Status=oo (always 00)

Wash Status=ww (00-02)

00 – Idle

02 – Out of Service

### Request Car Wash Status Command Example

- Car Wash ID 01
- Wash Status: Idle

Command:
STX *D01 ETX CD
Response:
STX 01 000000 ETX CD

Car Wash Commands  
**Get Operator Status**

*\*E — Used to get the operator status on a specific Car Wash*

### Get Car Wash Operator Status Command Characters \*E

### Purpose of Get Car Wash Operator Status Command

Gets the Car Wash operator status of a specific Car Wash.

### Format for Get Car Wash Operator Status Command

STX \*E ii ETX CD

Car Wash ID=ii (00 to 32)

### Response to Get Car Wash Operator Status Command

STX ii oo ETX CD

Car Wash ID=ii (00 to 32)

Operator Status=oo (0-3)

0 – Idle

1 – Busy

2 – Data Ready

3 – Error

### Get Car Wash Operator Status Command Example

- Car Wash ID 01
- Operator Status: Data Ready

Command:
STX *E01 ETX CD
Response:
STX 012 ETX CD

## Car Wash Commands Info Request

*\*F — Used to request information from a specific Car Wash*

### Car Wash Info Request Command Characters \*F

### Purpose of Car Wash Info Request Command

Requests specific information from a specific Car Wash.

### Format for Car Wash Info Request Command

STX \*F ii ETX CD

Car Wash ID=ii (00 to 32)

### Response to Car Wash Info Request Command

STX ii oo ETX CD

Car Wash ID=ii (00 to 32)

Operator Status=oo (0-3)

0 – Idle

1 – Busy

2 – Data Ready

3 – Error

### Car Wash Info Request Command Example

- Car Wash ID 01
- Operator Status: Data Ready

Command:
STX *E01 ETX CD
Response:
STX 012 ETX CD

### Extended Pump Commands

## Extended Pump – Set Decimal Point Modifier

*!O — Used to set a Decimal Point Modifier for data transfer*

### Set Decimal Point Modifier Command Characters *!O (letter O, not zero)*

### Purpose of Decimal Point Modifier Command

Stores a specific decimal point format for Dollar, Volume, and PPU that the POS wants to send/receive data in. For example, if the POS sends a Dollar decimal point of 0 in this command, then all data sent from this pump will come across in the format \$000000, rather than \$0000.00.

### Format for Decimal Point Modifier Command

STX \*O iii ETX CD

*Dispenser ID=ii (0001 to 9999)*

*Dollar DP=D (0-3)*

*Volume DP=V (0-3)*

*PPU DP-P (0-3)*

### Set Decimal Point Modifier Command Example

- Dispenser ID 01
- Dollar DP of 0
- Volume DP of 2
- PPU DP of 1

Command:
STX *O0001021 ETX CD
Response:
ACK

Extended Pump Commands  
**Extended Authorize**

!A — *Initiates a fueling operation*

## Extended Authorize Command Characters !A

### Purpose of Extended Authorize Command

Authorizes a specified dispenser. This functions the same as the non-extended authorize command listed above in this document, but allows for higher dispenser Ids (up to 9999) and formats the decimal points of the data depending on what is set in the optional Decimal Point Modifier command.

### Format for Authorize Command

STX !A Pump# Hose# Flag \$\$\$\$\$\$\$\$\$\$ vvvvvvvv ETX CD

*In the actual command, decimal points are implied only.*

Extended Format for Authorize Command:

STX !A Pump# Hose# E Flag \$\$\$\$\$\$\$\$\$\$ vvvvvvvv ETX CD

*Decimal point format will vary depending on what is sent in the optional Decimal Point Modifier command.*

Authorize Command Character = !A

Pump # = XXXX (0001 to 9999)

Hose # = X (0 to 8, with 0 authorizing any hose — also see Multi-grade Lock Authorization which follows)

Flag = X (Type of authorization — see Description)

Dollar Limit Amount = \$\$\$\$\$\$\$\$\$\$ (0000000.001 to 9999999999)

Volume Limit Amount = vvvvvvvv (0000000.001 to 9999999999)

Description of Flag Operation

0	Dollar Limit	<i>(Credit price)</i>
1	Dollar Limit	<i>(Cash price)</i>
2	Volume Limit	<i>(Credit price)</i>
3	Volume Limit	<i>(Cash price)</i>
4	Fillup	<i>(No limit)</i>
9	New Limit	<i>(Only dollar preset amount may be changed, not MOP — see New Limit section which follows)</i>

Response to Authorize Command

ACK/NAK only

Authorize Command Example

- Pump 1
- Any Hose
- \$25.00
- 0 Dollar DP

Command:
STX !A 00010100000000250000000000 ETX CD
Response:
ACK

---

## Extended Authorize — Implementing New Limit Feature

### Extended Authorize Command Characters !A

### Purpose of Extended Authorize Command Implementing New Limit

Progressive International has implemented a feature of the Authorize command which enables the user to change the preset value for a fueling transaction after an Authorize command has been sent to the controller. This New Limit feature is commonly known as Preset-On-The-Fly.

### Format for Extended Authorize Command

STX !A Pump# Hose# Flag \$\$\$\$\$\$\$\$\$\$ vvvvvvvv ETX CD

#### Flag Operation — New Limit

9    New Limit    *(Only preset amount may be changed, not MOP)*

#### Special Considerations:

- Due to the nature of this operation, MOP changes are not permitted.
- New Limit field must be the limit field for which the original preset was sent.
- When using New Limit, all fields must exist.
- Some dispensers can have their presets changed after flow has begun. Caution should be exercised using this option with the amount of fuel presently being dispensed is marginally close to the original preset limit. With this scenario, a possibility for an overrun condition exists.

#### Error Notes:

- A system error 06 will occur if this flag is used without a previous authorization being sent for that pump #.
- A pump error 08 will occur if the controller cannot change the preset due to dispenser-specific situations.

---

## Extended Authorize — Implementing Multi-Grade Lock Feature

### Extended Authorize Command Characters !A

### Purpose of Extended Authorize Command Implementing Multi-Grade Lock

To initiate a fueling transaction with specific grade combinations.

*To implement this feature, Hose# must be set to 0x3f. This alerts the controller to expect a bit pattern of grades to restrict. This bit pattern is appended to the normal Authorize command as two bytes representing the hoses to restrict: 0x3X<sub>1</sub> 0x3X<sub>2</sub>*

#### First byte 0x3X<sub>1</sub>

X= bit 0 = grade 5

bit 1 = grade 6

bit 2 = grade 7

bit 3 = grade 8

#### Second byte 0x3X<sub>2</sub>

X= bit 0 = grade 1

bit 1 = grade 2

bit 2 = grade 3

bit 3 = grade 4

### Format for Authorize Command Implementing Multi-Grade Lock

STX A Pump# Hose# Flag \$\$\$\$\$\$\$\$\$\$ vvvvvvvv 0x3X 0x3X ETX CD

### Format for Extended Authorize Command Implementing Multi-Grade Lock

STX A Pump# Hose# E Flag \$\$\$\$\$\$\$\$\$\$ vvvvvvvv ETX CD

### Example 1 — Authorize Command Implementing Multi-Grade Lock

Grades restricted: 1, 2 & 5	Command:
	STX !A Pump# 0x3f Flag \$\$\$\$\$\$\$\$\$\$ vvvvvvvvv 0x31 0x33 ETX CD
	Response:
	ACK

### Example 2 — Authorize Command Implementing Multi-Grade Lock

Grade restricted: 1 only	Command:
	STX !A Pump# 0x3f Flag \$\$\$\$\$\$\$\$\$\$ vvvvvvvvv 0x30 0x31 ETX CD
	Response:
	ACK

### Example 3 — Authorize Command Implementing Multi-Grade Lock

Grades restricted: 1 through 8	Command:
	STX !A Pump# 0x3f Flag \$\$\$\$\$\$\$\$\$\$ vvvvvvvvv 0x3f 0x3f ETX CD
	Response:
	ACK

Extended Pump Commands  
**Extended Polled Totals**

*!! — Receives totals from the dispenser*

## Extended Polled Totals Command Characters - !!

### Purpose of Polled Totals Command

*This command has the same purpose as the non-extended Polled Totals command, but, in addition, allows for 4 digit pump numbers, and returns state of charge information for Hydrogen dispensers.*

## Format of Extended Polled Totals Command

STX !! Pump# Hose# ETX CD

Extended Polled Totals Command Characters=!!  
Pump # =XXXX (0001 to 9999)  
Hose # =X (0 to 8)

## Response to Extended Polled Totals Command

STX Pump# Hose# Flag vvvvvv.vv XXXXXXXX.XX YYYYYYYY.YY ZZZZZZZZ.ZZ ETX  
CD

Pump # =XXXX	(01 to 9999)
Hose # =X	(0 to 8)
Flag =X	(Totals type — see following description)
Volume Total =vvvvvv.vv	(0000000.001 to 9999999.999)
Credit Total =XXXXXXXX.XX	(00000000.01 to 99999999.99)
Cash Total =YYYYYYY.YY	(00000000.01 to 99999999.99)
State of charge =ZZZZZZZ.ZZ	(00000000.01 to 99999999.99)

### Description of Totals Flag

- |   |                                                           |
|---|-----------------------------------------------------------|
| 0 | Totals not available for this dispenser                   |
| 1 | Pump busy; try later                                      |
| 2 | Cash & Credit combined money totals with combined volume  |
| 3 | Cash & Credit separated money totals with Combined volume |

*If the flag is a 0 or 1, the totals fields will be zero. Decimal points are implied.*

## Extended Polled Totals Command Example

<ul style="list-style-type: none"> <li>• Pump 1</li> <li>• Hose 3</li> <li>• Totals Type 3</li> <li>• Volume 1234512.345</li> <li>• Credit Total \$9875432.10</li> <li>• Cash Total \$5555123.45</li> <li>• State of Charge</li> </ul>	Command:
	STX !! 00013 ETX CD
	Response:
	ACK STX  013312345123459876543210555551234577777543 21 ETX CD

77777543.21



EMV Reader Commands

### EMV Set Preset Amount

*\*Q — Set Preset Amount for EMV Transactions*

### EMV Set Preset Amount Command Characters \*Q

### Purpose of EMV Set Preset Amount Command

Sets the maximum amount to be used as the preset amount for EMV transactions. 2 decimal points are assumed. If this command is sent after the readers have already been logged on, then the controller will need to be reset to reflect the changes.

### Format for EMV Set Preset Amount Command

STX \*Q aaaaaaaaaaaaa ETX CD

EMV Set Preset Amount Command Characters=\*Q  
a = preset amount (12 digits, 2 decimal points assumed)

### Response to EMV Set Preset Amount Command

NAK if command has invalid data.

ACK if was successful.

### EMV Set Preset Amount Command Example

•Preset amount of \$100.

Command:
STX !Q 000000010000 ETX CD
Response:
ACK

## EMV Reader Commands

### Extended Reader Status

*\*Y — Retrieves additional status data needed for EMV transactions*

#### Extended Reader Status Command Characters \*Y

#### Purpose of Extended Reader Status Command

Retrieves status indicators pertaining to the controller's card reader section. The first three status bytes indicate the current queue conditions, and are broken down in the original Reader Status command definition above. There are two additional status bytes that follow (S4 and S5), these contain information needed for EMV transactions and the queue data needed for the applicable commands. The next bytes (one per reader), indicate reader conditions. The amount of Reader bytes is variable, and there will be 16 bytes for each reader channel defined in the controller. If there is 1 reader channel defined, expect 16 reader bytes, 32 for 2 channels, 48 for 3 channels, and so on. It is a good practice for POS developers to read in the command until you see the 0x03 [ETX] flag that marks the end of the data, and simply use what data is needed for your purposes.

#### Format for Extended Reader Status Command

STX \*Y ETX CD

#### Response to Extended Reader Status Command

*STX S1 S2 S3 S4 S5 RRRRRRRRRRRRRRRR ... ETX CD*

S4	= bit 7	<i>(Don't care)</i>
	bit 6	<i>(1)</i>
	bit 5	<i>(EMV App. Selected Queue full)</i>
	bit 4	<i>(EMV App. Selected Queue empty)</i>
	bit 3	<i>(EMV OAR Queue full)</i>
	bit 2	<i>(EMV OAR Queue empty)</i>
	bit 1	<i>(EMV Trans. Result Queue full)</i>
	bit 0	<i>(EMV Trans. Result Queue empty)</i>
S5	= bit 7	<i>(Don't care)</i>
	bit 6	<i>(1)</i>
	bit 5	<i>(PIN Entry Queue full)</i>
	bit 4	<i>(PIN Entry Queue empty)</i>
	bit 3	<i>(APDU Queue full)</i>
	bit 2	<i>(APDU Queue empty)</i>
	bit 1	<i>(PIN Block Queue full)</i>

- Readers 1 & 2 logged on
- Readers 3-16 not logged on
- One reader Channel defined in controller.
- All queues empty

Command:
STX *Y ETX CD
Response:
STX EUYUUII@@@@@@@@@@@@@@@@ ETX CD

## EMV Reader Commands

## PIN Entry Data

*\*P — Retrieves or Clears PIN Entry data needed for EMV transactions*

## PIN Entry Data Command Characters \*P

## Purpose of PIN Entry Data Command

Retrieves the top entry in the PIN Entry Que. This will contain both the PIN Block and the KSN needed to pass along to the card server used by the POS.

## Format for PIN Entry Data Command

STX \*P f ETX CD

Extended Reader Status Command Characters=\*P

Function = f

R = Reads the top entry in the PIN Entry Que

C = Clears the top entry in the PIN Entry Que

## Response to PIN Entry Data Command - Read

NAK if empty

or

*STX rr c PP pppp... KK kkkk... ETX CD*

Reader ID=rr (00-99)

Was Cancelled=c (1 for cancelled, 0 for not cancelled)

Pin Block Length=PP (00-FF, is a hex value)

PIN Block Data=pppp ... (contains the PIN Block Data, expect the number of

bytes given in the PIN Block Length)

KSN Length=KK (00-FF, is a hex value)

KSN Data=kkkk... (contains the KSN Data, expect the number of bytes given in the KSN Length)

### Response to PIN Entry Data Command - Clear

ACK if cleared successfully

NAK if empty

### PIN Entry Data Command Example

<ul style="list-style-type: none"> <li>• Reader ID 1</li> <li>• Not cancelled</li> <li>• PIN Block Length of 0x10 (16)</li> <li>• PIN Block Data: 1234567890123456</li> <li>• KSN Length of 0x10 (16)</li> <li>• KSN Data: 0987654321098765</li> </ul>	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr style="background-color: #f2f2f2;"> <td style="padding: 5px;">Command:</td> </tr> <tr> <td style="padding: 5px; text-align: center;">STX !P R ETX CD</td> </tr> <tr style="background-color: #f2f2f2;"> <td style="padding: 5px;">Response:</td> </tr> <tr> <td style="padding: 5px; text-align: center;">STX 01 0 10 1234567890123456 10 0987654321098765 ETX CD</td> </tr> </table>	Command:	STX !P R ETX CD	Response:	STX 01 0 10 1234567890123456 10 0987654321098765 ETX CD
Command:					
STX !P R ETX CD					
Response:					
STX 01 0 10 1234567890123456 10 0987654321098765 ETX CD					

EMV Reader Commands  
**EMV Application Data**  
*\*R — EMV Application Data*

## EMV Application Data Command Characters \*R

### Purpose of EMV Application Data Command

Sends/Clears the EMV Application Data and Public Keys needed to operate readers using the Conexxus protocol. Also used to save the data to a file, and to check the status to see whether FDSharp needs to be reset. If new data is sent after FDSharp already begins sending data to a terminal, then the file will need to be saved and FDSharp will need to be restarted for the changes to take effect. This file is saved as "EMVData.xml" in the same folder as FDSharp.

### Format for EMV Application Data Command

STX \*R rr F ... ETX CD

Flag=F	<i>(Type of operation — see below)</i>
Reader ID=rr	(00-99) Reader ID 0 performs the action for ALL readers

#### Description of Flag Operation

- A Set Application Data
- B Set Public Key Data
- C Clear all Public Keys
- D Clear all Application Data
- F Save/signal finished
- S Get status

## Format for Set Application Data Command – A flag (Set TAC values and other data)

STX \*RA rr A aaa B bbb C ccc D ddd E eee F fff G ggg H hhh I iii J jjj K kkk L lll  
M mmm N nnn O ooo P ppp ETX CD

Set Application Data Command – A flag=\*RA  
ReaderID = rr (00-99)

Each of the upper case letters that follow \*RA are flags to signal which data is to follow (e.g. B). These flags are defined below. The corresponding lowercase letters (e.g. bbb) must be formatted with the size of that dataset, followed by the data. Each of the datasets (e.g. bbb) in this command expects 2 bytes (00-99) containing the size of the dataset, followed by the actual dataset. For example, the B flag (application name) would be formatted as B04VISA or B10MASTERCARD. If an RID/PIX combination that already exists is sent by the POS, then the values for that RID/PIX combination will be overwritten with the new data.

A – RID  
B – Application Name  
C – PIX  
D – Application Version  
E – TAC Default  
F – TAC Denial  
G – TAC Online  
H – DDOL  
I – TDOL  
J – Selection Indicator  
K – Floor Limit  
L – Threshold Value  
M – Target Percentage  
N – Max Target Percentage  
O – Terminal Capabilities  
P – Pin Bypass (Y for yes, N for no. e.g. P01Y)

\*The only required fields are A, B, C, D, E, F, and G, but it is recommended to send all fields where possible.

## Response to Set Application Data Command – A flag

### Set Application Data Command – A flag Example

- Reader ID 1
- A flag
- RID (A) A000000003
- Application Name (B) "VISA CREDIT"
- PIX (C) 1010
- Application Version (D) 84
- TACDefault (E) D84000A800
- TACDenial (F) 0010000000
- TACOnline (G) D84004F800
- DDOL (H) 9F3704
- TDOL (I) 9F02065F2A029A039C0195059F3704
- Selection Indicator (J) 01
- Floor Limit (K) 0
- Threshold Value (L) 0
- Target Percentage (M) 0
- Max Target Percentage (N) 0
- Terminal Capabilities (M) not present
- Pin Bypass (P) Off

Command:
STX *R A 01 A10A000000003B11VISA CREDITC041010D0284E10D84000A80 0F100010000000G10D84004F800H0 69F3704I309F02065F2A029A039C019 5059F3704J0201K010L010M0299N02 99P01N ETX CD
Response:
ACK

## Format for Set Application Data Command – B flag (Set Public Key data)

STX \*RB rr A aaa Q qq R rrr S sss T tt U uuuu V vvv ETX CD

Set Application Data Command – B flag=\*RB  
ReaderID = rr (00-99)

Each of the upper case letters that follow \*RB are flags to signal which data is to follow (e.g. A). These flags are defined below. The corresponding lowercase letters (e.g. aaa) must be formatted with the size of that dataset, followed by the data. Most of the datasets (e.g. aaa) in this command expects 2 bytes (00-99) containing the size of the dataset, followed by the actual dataset. Hash Value (U) and Modulus (V) both expect a size of 3 however (000-999), since the length of these can be greater than 99. If an RID/Number combination that already exists is sent by the POS, then the values for that RID/Number combination will be overwritten with the new data.

A – RID  
Q – Sign Algorithm  
R – Exponent  
S – Number  
T – Hash Algorithm  
U – Hash Value  
V – Modulus

## Response to Set Application Data Command – B flag

### Set Application Data Command – B flag Example

- Reader ID 1
- B flag
- RID (A) A000000003
- Sign Algorithm (Q) 01
- Exponent (R) 03
- Number (S) 01
- Hash Algorithm (T) 01
- Hash Value (U)  
D34A6A776011C7E7CE3AEC5  
F03AD2F8CFC5503CC
- Modulus (V)  
C696034213D7D8546984579  
D1D0F0EA519CFF8DEFFC429  
354CF3A871A6F7183F1228D  
A5C7470C055387100CB935A  
712C4E2864DF5D64BA93FE7  
E63E71F25B1E5F5298575EBE  
1C63AA617706917911DC2A7  
5AC28B251C7EF40F2365912  
490B939BCA2124A30A28F54  
402C34AECA331AB67E1E79B  
285DD5771B5D9FF79EA630B  
75

Command:
STX *R B 01 A10A000000003Q0201R0203S0201T0 201U40D34A6A776011C7E7CE3AEC5 F03AD2F8CFC5503CCV256C6960342 13D7D8546984579D1D0F0EA519CFF 8DEFFC429354CF3A871A6F7183F122 8DA5C7470C055387100CB935A712C 4E2864DF5D64BA93FE7E63E71F25B1 E5F5298575EBE1C63AA61770691791 1DC2A75AC28B251C7EF40F2365912 490B939BCA2124A30A28F54402C34 AECA331AB67E1E79B285DD5771B5D 9FF79EA630B75 ETX CD
Response:
ACK

## Format for Set Application Data Command – C flag (Clear all Public Keys)

STX \*RC rr ETX CD

Set Application Data Command – C flag=\*RC  
Reader ID = rr

This command is used to clear all the data sent from the A flag for a specific Reader ID. Reader ID 00 does this for all Readers.

## Response to Set Application Data Command – C flag

### Set Application Data Command – C flag Example

- Reader ID 1
- C flag

Command:
STX *R C 01 ETX CD
Response:
ACK

## Format for Set Application Data Command – D flag (Clear all Public Keys)

STX \*RD rr ETX CD

Set Application Data Command – D flag=\*RD  
Reader ID = rr

This command is used to clear all the public keydata sent from the B flag for a specific Reader ID. Reader ID 00 does this for all Readers.

## Response to Set Application Data Command – D flag

### Set Application Data Command – D flag Example

- Reader ID 1
- D flag

Command:
STX *R D 01 ETX CD
Response:
ACK

### Format for Set Application Data Command – F flag (Save all EMV Data)

STX \*RF 00 ETX CD

Set Application Data Command – F flag = \*RF

This command is used to save all EMV data for all readers to “EMVData.xml”, which is located in the same folder as FDSsharp.exe.

### Response to Set Application Data Command – F flag

### Set Application Data Command – F flag Example

- F flag

Command:
STX *R F 00 ETX CD
Response:
ACK

## Format for Set Application Data Command – S flag (Get Status)

STX \*RS 00 ETX CD

Set Application Data Command – S flag=\*RS

This command is used to get the status and other information for which readers are sending the EMV configuration data, if the file is saved, and other data.

## Response to Set Application Data Command – S flag

STX \*RD 00 ETX CD

Set Application Data Command – S flag=\*RS

This command is used to save all EMV data for all readers to “EMVData.xml”, which is located in the same folder as FDSharp.exe.

aa = number of channels currently sending application data to a terminal (2 bytes, 00-99).

bb = number of channels currently sending public key data to a terminal (2 bytes, 00-99).

c = is loading save file. (1 byte, 0 if currently loading or 1 for not currently loading)

d = is saving save file. (1 byte, 0 if currently saving or 1 for not currently saving)

e = \*did receive new data (1 byte, 0 if no new data received, 1 if new data was received).

f = \*has started sending app data (1 byte, 0 if no terminals have started sending app data or public key data, 1 if any terminal has started sending app data or public key data)

\*note on e/f: These bytes are used to determine when to send a “save/signal finish” flag, and when FDSsharp requires a reset for changes to apply. The e byte will only be set to 1 if any “new” data is sent to FDSsharp, so if we get any duplicate data that was already loaded from the save file, then e is 0. The e byte will be reset to 0 once the Save/signal finish flag is sent from the POS. If byte e is 1 after you finish sending data, then send a save/signal finish command, and if both e and f are 1 after you finish sending data, then restart fdsharp after saving for the changes to take effect. If both bytes are 0, or only f is 1 and e is 0, after the POS finishes sending data, then there is no need to save or restart FDSsharp, and normal operations can continue.

- e=0, f=0: no save or restart required.
- e=0, f=1: no save or restart required.
- e=1, f=0: save required, no restart required.
- e=1, f=1: save required, then restart required.

*(Future EMV documentation will be implemented after certification)*

### Set Application Data Command – S flag Example

- S flag
- 1 terminal sending app data
- 1 terminal sending public key data
- Is not loading save file
- Is not saving save file
- Did receive new data from POS
- Has started sending EMV configuration data to the terminal.

Command:
STX *R S 00 ETX CD
Response:
STX 01010011 ETX CD

Card Reader Commands  
**Sound Payments Transaction Que**

*\*S — Sound Payments Transaction Que*

### Sound Payments Transaction Que Characters \*S

### Purpose of Sound Payments Transaction Que Command

Used to retrieve and clear the reader and dispenser information associated with sales authorized from the Sound Payments IFSF bridge. There are two sub-commands: Read the top queue entry and Clear the top queue entry.

### Format for Sound Payments Transaction Que Command - Read

STX \*SR ETX CD

Sound Payments Transaction Que Command -Read Characters=\*SR

### Response to Sound Payments Transaction Que Command - Read

NAK if queue is empty

or

*STXdbDV PcrpheINLE ETX CD*

Pump ID=d	(0-9999, 4 bytes)
Is From Bridge=b	(1 if sale was authorized from the sound payments bridge, 0 if not)
Sale Dollar=D	(10 bytes, defaults to 2 decimal places)
Sale Volume=V	(10 bytes, defaults to 3 decimal places)
Sale PPU=P	(10 bytes, defaults to 3 decimal places)
Has Card Info=c	(1 if card info follows, 0 if not. If this byte is 0 then the below bytes will not be sent from FDSharp and this will end the command)
Reader ID=r	(0-9999, 4 bytes)
Last 4 digits of PAN=p	(0-9999, 4 bytes)

---

PAN Hash=h	(128 bytes, this is an encrypted hash of the PAN)
Has EMV Data=e	(1 if emv data follows, 0 if not. If this bytes is 0 then the below bytes will not be sent from FDSsharp and this will end the command)
Cardholder name length=l	(2 bytes, holds the length of the cardholder name that follows)
Cardholder name=N	(variable length, expect the number of bytes given in the cardholder name length)
EMV data length=L	(4 bytes, holds the length of the EMV Data that follows)
*EMV data=E	(variable length, expect the number of bytes given in the EMV data length)

\*EMV data description:

The EMV data section will contain several EMV tags, shown below. After each EMV tag will follow 2 bytes (hexadecimal) that define the length of the data in that tag, the amount of characters to follow will be twice as many as the length. So an 8A tag with a value of 5A33 will have a length of 2, and look like this:

"8A025A33". It is recommended to:

- Find if the tag is 2 or 4 characters (if the second character is an F, then the tag will be 4 characters)
- Read the next 2 bytes, after the tag, for the length (x)
- Then read the next (x\*2) number of bytes before reading another tag

These are the tags that will be sent in the Sound Payments Transaction Que  
Command:

- 82
- 8A
- 9A
- 95
- 9B
- 5F2A
- 5F34
- 9F02
- 9F03
- 9F07
- 9F0D
- 9F0E
- 9F0F
- 9F10
- 9F12
- 9F1A
- 9F21
- 9F26
- 9F27
- 9F34
- 9F36
- 9F37
- 9F06

The descriptions for these tags can be found on <https://emvlab.org/emvtags/>

### Sound Payments Transaction Que Command - Read Example

- Dispenser ID: 0001
- Transaction was from bridge
- Sale Dollar: 137.16
- Sale Volume: 123.456
- Sale PPU: 1.111
- Has Card Info
- Reader ID: 0001
- Last 4 digits of PAN: 5678
- PAN Hash:  
Oxd7t6W5ZHz2HUJO3Rvw0OpZZmdf49QRM5lxFSgag8cYljR4bg4cP3GOSf28Xmis3ec7OfBCd+20D5r4M+NiprnZTFwD/4RZmx0gVl5jrDakMfyCFZPC81FPasDqnUGJ
- Has EMV Data
- Name Length: 9
- Name: "Test Name"
- EMV Data Length: 0336
- EMV Data:  
5F2A0208405F340103820238008A025A339A032208189F0206000000001009F0306000000000009F0702FFC09F0D05FC50A000009F0E05000000009F0F05F870A498009F1012021060000000000000000000FF9F120A4D6173746572436172649F1A0208409F21031602209F2608A2C3B5FC189048969F2701809F34034203009F360200279F37040DBAA612950500000480009B02E8009F0607

Command:
STX *SR ETX CD
Response:
STX 0001100000137160000123456000000 1111100015678OXd7t6W5ZHz2HUJO3 Rvw0OpZZmdf49QRM5lxFSgag8cYljR4bg 4cP3GOSf28Xmis3ec7OfBCd+20D5r4M +NiprnZTFwD/ 4RZmx0gVl5jrDakMfyCFZPC81FPasDqnU GJ109Test Name03365F2A0208405F3401038202 38008A025A339A032208189F0206000 0000001009F030600000000000009F07 02FFC09F0D05FC50A000009F0E05000 00000009F0F05F870A498009F101202 10600000000000000000000000000000 0FF9F120A4D6173746572436172649F 1A0208409F21031602209F2608A2C3B 5FC189048969F2701809F3403420300 9F360200279F37040DBAA6129505000 00480009B02E8009F0607A000000004 1010 ETX CD

A0000000041010

--

### Format for Sound Payments Transaction Que Command - Clear

STX \*SC ETX CD

Sound Payments Transaction Que Command -Clear Characters=\*SC

This command clears the top entry in the Sound Payments Transaction Que.

### Response to Sound Payments Transaction Que Command - Clear

ACK if successful, or NAK if queue is empty

### Sound Payments Transaction Que Command - Read Example

Command:
STX *SC ETX CD
Response:
ACK

## Generic FDSsharp Commands

**Get License Info***\*T — Get License Info***Get License Info Characters \*T****Purpose of Get License Info Command**

Used to get the status and expiration date of the software license in FDSsharp.

**Format for Get License Info Command**

STX \*T ETX cd

**Response to Get License Info Command**

STX S MMdyyyhhmmss ETX cd

Status=S	(1 for permanent license, 0 for timed license)
Expiration Month=MM	(2 bytes, 01-12)
Expiration Day=dd	(2 bytes, 01-31)
Expiration Year=yyyy	(4 bytes, 2022-9999)
Expiration Hour=hh	(2 bytes, 00-23, military time)
Expiration Minute=mm	(2 bytes, 00-59)
Expiration Second=ss	(2 bytes, 00-59)

The Expiration date bytes will only be sent if the Status flag is returned as 0, indicating that it is a timed license. If the Status flag returns as 1, the status flag will be the last character in the response. The Expiration date/time is in UTC, and should be converted by the POS to the applicable timezone.

**Get License Info Command Example**

- Timed License
- Expiration Date 12/22/2022
- Expiration Time 03:51:23 pm

Command:
STX *T ETX CD
Response:
STX 012222022155123 ETX CD

## Dispenser Specific Information

## Tokheim

Authorize Command — new limit

- Dispenser must support the AE46 command.

## Gilbarco

Authorize Command — new limit

- PCXZ must have gil\_pump version 3.70 8/28/97 or later.
- New limit must be sent prior to flow.

## Wayne-Dresser

Authorize Command — new limit

- PCXZ must have dw\_pump version 5.00 9/02/97 or later.
- New limit may be sent at any time prior to previous limit being reached.
- If increasing or decreasing original limit set, allow for some delay in communication. If insufficient time is allowed, error message may result or new limit may not be attained.

## Reader Specific Information

## Tokheim

- Display:
  - 4 x 20 characters
  - no graphics
  - scrolling not supported
- DES encryption to be supported – *Contact PIE for release date*
- Beeper options – 1 or 3
- Reader software must be Version JP020800 or later

## Gilbarco

- Display:
  - 1 x 20 characters
  - Scroll Flag  $0_x13_{\text{HEX}}$  (Must be first display character)
- DES encryption supported
- Beeper options – 1 to 10
- Only Print Default preloadable messages are supported – P1 through P4
- Receipts must be 40 lines long
- CRIND software must be version 51.1.6 or later

## Wayne-Dresser

- Display:
  - 2 x 20 characters
  - no graphics
  - scrolling not supported
- DES encryption supported
- DES key read returned in packet with K as first character  
*If DES key read is double-wide, one position must be unused*
- Packet Command Flag
  - Include as first character of data flag to describe to which type device the packet is being sent — card reader or MSM
    - C = Card Reader
    - M = MSM
  - Device #01 Bank 0 MSM
  - Device #17 Bank 1 MSM
- Beeper options — 1 to 10
- Only Print and Card Default preloadable messages are supported —  
P1 through P4 and C1 through C5