Manual No: **577013-790** ● Revision: **F**

Software Version F09

EMR^3

OBC Serial Commands Protocol

VEEDER-ROOT™

Notice

Veeder-Root makes no warranty of any kind with regard to this publication, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.

Veeder-Root shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this publication.

Veeder-Root reserves the right to change system options or features, or the information contained in this publication.

This publication contains proprietary information which is protected by copyright. All rights reserved. No part of this publication may be photocopied, reproduced, or translated to another language without the prior written consent of Veeder-Root.

Contact TLS Systems Technical Support for additional troubleshooting information at 800-323-1799.

DAMAGE CLAIMS / LOST EQUIPMENT

Thoroughly examine all components and units as soon as they are received. If any cartons are damaged or missing, write a complete and detailed description of the damage or shortage on the face of the freight bill. The carrier's agent must verify the inspection and sign the description. Refuse only the damaged product, not the entire shipment.

Veeder-Root must be notified of any damages and/or shortages within 30 days of receipt of the shipment, as stated in our Terms and Conditions.

VEEDER-ROOT'S PREFERRED CARRIER

1. Contact Veeder-Root Customer Service at 800-873-3313 with the specific part numbers and quantities that were missing or received damaged.
2. Fax signed Bill of Lading (BOL) to Veeder-Root Customer Service at 800-234-5350.
3. Veeder-Root will file the claim with the carrier and replace the damaged/missing product at no charge to the customer. Customer Service will work with production facility to have the replacement product shipped as soon as possible.

CUSTOMER'S PREFERRED CARRIER

1. It is the customer's responsibility to file a claim with their carrier.
2. Customer may submit a replacement purchase order. Customer is responsible for all charges and freight associated with replacement order. Customer Service will work with production facility to have the replacement product shipped as soon as possible.
3. If "lost" equipment is delivered at a later date and is not needed, Veeder-Root will allow a Return to Stock without a restocking fee.
4. Veeder-Root will NOT be responsible for any compensation when a customer chooses their own carrier.

RETURN SHIPPING

For the parts return procedure, please follow the appropriate instructions in the "General Returned Goods Policy" pages in the "Policies and Literature" section of the Veeder-Root **North American Environmental Products** price list. Veeder-Root will not accept any return product without a Return Goods Authorization (RGA) number clearly printed on the outside of the package.

Introduction

Interface Definition

Appendix A: OBC Serial Pass-Through Printing Commands

Appendix B: Serial Data Flow Diagram

Figures

Tables

Introduction

General

The mobile On Board Computer (OBC) for Fuel Delivery Systems communicates with and operates the EMR3 System. The scope of OBC to EMR3 interface is limited to all activities required for a regular fuel delivery (no initial meter setup or meter calibration).

The physical connection between the OBC and EMR3 uses the asynchronous serial communication protocol RS-232 (with no h/w flow control; 9600 baud, none parity, 8 bit data, 1 bit stop).

Packet Definition

All information between the OBC and the EMR3 is formatted into packets containing the following fields:

```
[DelimiterFlag][DestinationAddress][SourceAddress][Body of Message][CS][DelimiterFlag]
```

Table 1 describes the OBC data packets.

Table 1. OBC Packet Formats

Field	Description
`[DelimiterFlag]`	1 Byte defined as 0x7E

`[DestinationAddress]`
`[SourceAddress]`
	1 Byte representing a destination / source address: 0x01 - 0x20 - Single Meter address; 0x41 - 0x60 - Printer Device address; 0x81 - 0x90 - Interface Box address; x00 - broadcast address (both meters attached to an IB); 0xFF - OBC address

`[Body of Message]`	1- n Bytes formatted according to interface protocol
`[CS]`	1 Byte check sum* defined as: 0x00 - {[DestinationAddress] + [SourceAddress] + [Body of Message]}

*Checksum calculated according to CCITT CRC-16.

NOTE:

To eliminate forbidden characters (0x7E, 0x7D) inside the packet (defined as data between delimiter flags), the following "escape" needs to be performed:

1. XOR operation between CONTROL_CHAR (0x20) and a forbidden char.
2. Inserting an ESCAPE_CHAR (0x7D) in front of the result of step (1).

Whenever the packet contains the ESCAPE_CHAR (0x7D) the reverse operation should be performed:

1. Remove ESCAPE_CHAR from the packet.
2. XOR the following character with the CONTROL_CHAR (0x20).

Interface Definition

General Rules

The communication between OBC and third party meter will be guided by following rules:

1. All information (i.e. command type, exchanged data) is located in <Body of the Message> portion of interface packet;
2. The <Body of the Message> always starts with Command Code (1 BYTE) sometimes followed by Field Code (1 BYTE) and proper parameters;
3. There is only a single command (represented by Command Code) inside of each interface packet;
4. Check sum ([CS] in Table 1) is calculated before the "escape" operation and validated on a destination side after the "escape" operation is performed;
5. All message exchanges are solicited by the OBC - the EMR3 does not send any message without a request from the OBC;
6. Incorrectly formatted packets (missing delimiter flags, incorrect CS) will be discarded;
7. Point to point communication with an EMR System is half-duplex. A half-duplex system provides communication in both directions but not at the same time. The EMR IB can only receive and process one message at a time. After receiving a message, the EMR transmits a response message to the OBC, prior to the OBC transmitting a new message. It is the responsibility of the OBC software to ensure that a proper response was received from the EMR before transmitting the next message.
8. The following time delays, related to half-duplex communication, must be followed to avoid significant processing delays. After transmitting a message, the OBC must wait a minimum of 1 second prior to retrying to send the same message if the proper response is not received by the OBC.
9. If the EMR IB repeatedly fails to communicate a response, the OBC must wait several seconds prior to sending a new command message.

Message Description

Table 4 identifies and defines the interface messages between the OBC and EMR3 required for fuel delivery processing.

NOTES TO TABLES

Entries in the Binary Code column of tables 6, 7, 8, 9, 10, and 13 are in the form of a numeric (1 Byte) value.

Definitions:

CHAR - Signed byte (1 byte)

UCHAR - Unsigned byte (1 byte)

SHORT - Signed Short Integer (2 bytes)

USHORT - Unsigned Short Integer (2 bytes)

LONG - Signed Long Integer (4 bytes)

ULONG - Unsigned Long Integer (4 bytes)

FLOAT - signed 4-byte IEEE floating-point number that always contains a positive number

SFLOAT - signed 4-byte IEEE floating-point number

DOUBLE - signed 8-byte IEEE floating-point number

Data transmission structure is type little-endian in which the last significant byte is transmitted first.

IEEE Floating Point Format

The 32-bit float is arranged as follows:

Table 2. IEEE 32-Bit Float

Byte	1		2		3		4	
	S EEE	EEEE	E MMM	MMMM	MMMM	MMMM	MMMM	MMMM
Nibble	1	2	3	4	5	6	7	8

- S is the sign bit (0 if positive, 1 if negative).
- EEE EEEE E represents the 2's exponent. It is a 2's complement value biased by 127 (7F Hex). The exponent can be determined by subtracting 127 from the value of the E field and raising 2 to the resulting power.
- MMM MMMM MMMM MMMM MMMM MMMM represents the 23-bit mantissa. Since the mantissa describes a value which is greater than or equal to 1.0 and less than 2.0, the 24th bit is always assumed to be equal to 1 and is not transmitted or stored. The value of the mantissa can be determined by dividing the value of the M field by 8,388,608 (2^{23}) and adding 1.0.

The complete value of the floating point number can then be determined by multiplying the exponent by the mantissa and attaching the appropriate positive or negative sign.

By convention, 00 00 00 00 represents the value 0.0 even though it actually converts to 5.8775 x 10^{-39}.

The eight "nibbles" are transmitted in sequence from 1 through 8 as shown in Table 2.

IEEE Floating Point Format Examples

1. 3F800000 hex = 0011 1111 1000 0000 0000 0000 0000 0000 bin
   S = 0 = +(positive)
   E = 011 1111 1 bin = 7F hex = 127 dec
   M = 000 0000 0000 0000 0000 0000 bin = 0 hex = 0 dec

   Exponent = $2^{(127-127)}$ = 1.0
   Mantissa = 1.0 + (0/8,388,608) = 1.0
   Decimal Value = +1.0 x 1.0 = 1.0

2. B8D1B717 hex = 1011 1000 1101 0001 1011 0111 0001 0111 bin
   S = 1 = -(negative)
   E = 011 1000 1 bin = 71 hex = 113 dec
   M = 101 0001 1011 0111 0001 0111 bin = 51 B7 17 hex = 5,355,287 dec

   Exponent = $2^{(113-127)}$ = 0.0000610352
   Mantissa = 1.0 + (5,355,287/8,388,608) = 1.63840
   Decimal Value = -0.0000610352 x 1.63840 = -0.0001

3. C2C7FAE1 hex = 1100 0010 1100 0111 1111 1010 1110 0001 bin
   S = 1 = -(negative)
   E = 100 0010 1 bin = 85 hex = 133 dec
   M = 100 0111 1111 1010 1110 0001 bin = 47 FA E1 hex = 4,717,281 dec

   Exponent = $2^{(133-127)}$ = 64
   Mantissa = 1.0 + (4,717,281/8,388,608) = 1.56234
   Decimal Value = -64 x 1.56234 = -99.99
4. 461C4000 hex = 0100 0110 0001 1100 0100 0000 0000 0000 bin
   S = 0 = +(positive)
   E = 100 0110 0 bin = 8C hex = 140 dec
   M = 001 1100 0100 0000 0000 0000 bin = 1C 40 00 hex = 1,851,392 dec

   Exponent = $2^{(140-127)}$ = 8,192
   Mantissa = 1.0 + (1,851,392/8,388,608) = 1.22070
   Decimal Value = +8,192 x 1.22070 = 10,000

Table 3. IEEE 64-Bit Float*

Byte	1		2		3		4		5		6		7		8	
	S EEE	EEEE	EEEE	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF
Nibble	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

*1st bit is sign bit (S), next 11 bits are exponent bits (E), and final 52 bits are the fraction (F).

Table 4. List of Command/Response Codes[1]

COMMAND				RESPONSE		
From/To	Command Code (ASCII)	Description	Field Code (R = read only)	From/To	Response Code (ASCII)	Field Code (R = read only)
OBC/Meter	D	Set register configuration	See Table 13	Meter/OBC	A	See Table 7
	O	Set delivery status	See Table 6			
	P	Print the following buffer in “print through” mode. Print status bit 1 must be true.	STRING [100] - null terminated			
	R	Reset meter request, bring it to default state (i.e., “on power up”, with ACK code 2 if meter in a maintenance/ C&C mode)	None			
	S	Set Meter Fields	See Table 5			
OBC/Meter	E	Get Register Configuration	See Table 13	Meter/OBC	C	See Table 13
OBC/Meter	G	Get Meter FIelds	See Table 5	Meter/OBC	F	See Table 5
OBC/Meter	H	Transaction Record Request w/o Custom Fields	See Table 9	Meter/OBC	I	See Table 10
OBC/Meter	J	Transaction Record Request with Custom Fields	See Table 9	Meter/OBC	K	See Table 10
OBC/Meter	T	Get Meter Status	See Table 8 (**R**)	Meter/OBC	M	See Table 8
OBC/Meter	V	Request version numbers of device addressed. Field Code = 0	None (**R**)	Meter/OBC	U	CHAR Main Number[15] BootNumber[2] (**R**)
OBC/Printer	p	Print device control	See Table 16	Printer/OBC	p	See Table 17

[1] ASCII codes W, X, Y, and Z are reserved.

Table 5. Get Register Configuration = ‘G’ and Set Register Configuration = ‘S’

ASCII Code	Description	Parameters (R = read only)
a	Net total quantity of current shift (for current product, counted from last reset)	DOUBLE - volume (**R**)
b	Gross total quantity of current shift (for current product, counted from last reset)	DOUBLE - volume (**R**)
c	Preset volume (compensated) for current product	FLOAT - volume
d	Current date	BYTE - century [20-99] BYTE - year [1-99] BYTE - month [1-12] BYTE - day [1-31]
e	Meter net totalizer reading for current product	DOUBLE - volume (**R**)
f	Meter gross totalizer reading for current product	DOUBLE - volume (**R**)
g	Gross volume (quantity) of current delivery	DOUBLE - volume (**R**)
h	Number of decimal digits for all volume values	BYTE - [0,1,2] (**R**)
i	Current time	BYTE - hour [00-23] BYTE - minutes [00-59] BYTE - seconds [00-59]
j	Gross totalizer	DOUBLE - volume (**R**)
k	Get Register display field current value	First Byte Response: Current Display Mode 0: VOLUME MODE 1: CURRENCY MODE 2: RATE MODE Second Byte to the end of the response: Null terminated string of value currently displayed in the Register display field. In VOLUME MODE value is the current volume displayed In CURRENCY MODE value is the current sale In RATE MODE value is the rate of fuel flow
l	Get Totalizer display field current value	Null terminated string of the value currently displayed in the Totalizer display field.
m	Timeout for no flow delivery - it will terminate delivery when paused	USHORT - time in seconds. 5 sec< timeout<20 min
n	Preset volume (gross) for current product	FLOAT - volume

Table 5. Get Register Configuration = ‘G’ and Set Register Configuration = ‘S’

ASCII Code	Description	Parameters (R = read only)
o	Get Preset display field current value	Null terminated string of the value currently displayed in the Preset display field.
p	Current (active) product	1 byte - product index [0,1,2]
q	Print pause. When On, the meter will not attempt to print at the printer. New transactions can not be started until print pause is set Off and queued transaction is printed.	1 Byte [0=Off, 1=On]
r	Meter serial number	STRING[20] - Null terminated (**R**)
s	Current sale number	ULONG - incremental number (**R**)
t	Current product temperature	SFLOAT - temperature (**R**)
u	Enters key stroke into EMR3 head. Write only.	1 byte, key to enter 0 = Start, 1=Finish, 2=Mode, 3=Preset, 4=Next, 5=Plus, 6=Minus, 7=Clr, 8=Enter, 9=keypad 0, 10=K1, 11=K2, 12=K3, 13=K4, 14=K5, 15=K6, 16=K7, 17=K8, 18=K9
v	Compensated volume (quantity) of current delivery	DOUBLE - volume (**R**)
w	Current tank ID	ASCII string. UCHAR TankID[N] where N <= 10.
K	Real-time volume displayed during delivery. Unformatted, no rounding applied	DOUBLE - volume (**R**)
L	Real-time totalizer displayed. Unformatted, no rounding applied.	DOUBLE - volume (**R**)
O	Preset countdown value displayed. Only valid when preset is active. Unformatted, no rounding applied	SFLOAT - preset (**R**)
R	Delivery rate as seen on the display. Only valid during delivery. Unformatted, no rounding applied	DOUBLE - volume (**R**)

Table 6. Set Delivery Status = ‘O’

Binary Code	Description	Parameters
1	Start, resume a delivery, use optional byte to select product code when starting a delivery - no effect when resuming a delivery	1 byte (optional) 0, 1, 2, product index
2	Pause a delivery	None
3	End a delivery	
4	Force a mandatory ticket printing for current delivery	

Table 6. Set Delivery Status = 'O'

Binary Code	Description	Parameters
5	Start multiple delivery, use optional byte to select product code when starting a delivery - no effect when resuming a delivery	1 byte (optional) 0, 1, 2, product index
6	Authorize delivery state	1 byte 0 = delivery not authorized 1 = delivery authorized
7		(Reserved – Internal Use)
8	Set Current Price Value	4 Bytes – IEEE price value to set Response: ACK or NACK **Note: Only works in PRE_DELIVERY mode.**

Table 7. All Set commands use these results

Binary Code	Description
0	No error - OBC message acknowledged
1	Error - requested code/action not understood
2	Error - requested action can not be performed

Table 8. Get Meter Status = 'T'

Binary Code	Description	Parameters
1	Meter status	1 BYTE bit coded: Bit 0 - delivery mode = No, product flowing=No Bit 1 - delivery mode = Yes, product flowing=Yes Bit 2 - delivery mode = Yes, product flowing=No Bit 3 - delivery mode = No, product flowing=Yes Bit 4 - the printer is busy Bit 5 - incorrect meter UI position. Set when meter cannot perform requested command due to state of user switch/button e.g., can't reset if C&C switch enabled) Bit 6 - meter error Bit 7 - Set if C&C mode enabled
2	Printer status	1 BYTE bit coded: Bit 0 - delivery ticket requested (insert invoice) Bit 1 - set after ticket printing is complete and waiting for slip/paper to be removed. This is the only time "pass through" printing is allowed. (For use with slip printer only!) Bit 2 - printer is busy Bit 3 - printer error

Table 8. Get Meter Status = ‘T’

Binary Code	Description	Parameters
3	Delivery status	2 BYTES bit coded: Bit 0 - ATC error Bit 1 - pulser/encoder error Bit 2 - preset error Bit 3 - preset stop. Set when delivery stopped after reaching preset volume. Bit 4 - no flow stop (timeout) Bit 5 - pause delivery request Bit 6 - delivery end request Bit 7 - waiting for authorization Bit 8 - delivery ticket is pending Bit 9 - flow is active Bit 10 - delivery is active Bit 11 - net preset is active Bit 12 - gross preset is active Bit 13 - ATC is active Bit 14 - delivery completed Bit 15 - delivery error
4	Request Current Setup Mode	1 BYTE - 0 = display head not in Setup Mode, 1 = display head is in Setup Mode
5	Authorization Required State	1 BYTE - 0 = authorization not required for all deliveries, 1 = authorization required for all deliveries
6	Get Current Price Value	4 BYTE - Float IEEE with the current delivery unit price
7	Get Current Product Price Code Value	4 BYTE - Float IEEE with the unit price for the current product price code
8	Get Current EMR State	1 BYTE - 0 : PRE_DELIVERY_STATE 1 : KEY_TIMEOUT_STATE 2 : DELIVERY_STATE 3 : FINISH_STATE 4 : POPUP_STATE 5 : DISPLAY_TEST_STATE

Table 9. Transaction Request : ‘H’ w/o custom fields – ‘J’ with custom fields

Binary Code	Description	Parameters (R = read only)
0	Total number of transaction records in device addressed	None (**R**)
1	Transaction with the matching index for device addressed	2 byte index number [0...199] (**R**)
2	Transaction with the matching ticket number in device addressed	4 byte ticket number (**R**)

Table 9. Transaction Request : ‘H’ w/o custom fields – ‘J’ with custom fields

Binary Code	Description	Parameters (R = read only)
3	Number of transaction records in device address that match parameter address. When polling the head, the head address parameter is ignored if it is included. It is required when polling the IB.	1 byte head address of meter [1...32] (**R**)
4	Transaction in device addressed that matches head & index parameters. When polling the head, the head address parameter is ignored if it is included. It is required when polling the IB.	2 byte index number [0...199] 1 byte head address of meter [1...32] (**R**)

Table 10. Transaction response field code – ID ‘I’ w/o custom fields – ID ‘K’ with custom fields

Binary Code	Description	Parameter
0	Number of transaction records in device	2 byte number of records stored
3	Transaction record	See Table 11 or Table 12, as req’d., for record description.

Table 11. Transaction Record Table w/o Custom Fields (ID: ‘H’ – Answer ID: ‘I’)

Field Description	Size (Bytes)	Type
Ticket #	4	LONG
Transaction Type 0 = Single delivery Record 1 = Multiple Delivery Record 2 = Summary Record 3 = Calibration Record	2	CHAR
Index: 0 for single, 1-N for multiple, -1 for summary	1	CHAR
# of summary records	1	CHAR
# of records summarized	1	CHAR
Product ID 0-2 (MAX_PRODUCTS - 1)	1	UCHAR
Product Info	15	CHAR
Product Info Null Termination	1	CHAR

Table 11. Transaction Record Table w/o Custom Fields (ID: 'H' – Answer ID: 'I')

Field Description	Size (Bytes)	Type
Start time date		
Minute 00-59	1	UCHAR
Hour 00-23	1	UCHAR
Day of month 01-31	1	UCHAR
Second 00-59	1	UCHAR
Month 01-12 (01=January)	1	UCHAR
Year 00-255 (2000-2255)	1	UCHAR
Finish time date - Same format as Start time date	6	
Tank Load	4	FLOAT
Subtotal	4	FLOAT
TotalizerStart	8	DOUBLE
TotalizerEnd	8	DOUBLE
GrossVolume. This is always the raw, uncompensated volume	8	DOUBLE
Volume. This is either gross or TC, depending on fuel	8	DOUBLE
Average temp	4	FLOAT
Unit price (if Transaction type is 3 = New calibration factor)	4	FLOAT
Six individual tax/discount fields, each 6 bytes (if Transaction type is 3 = unused)	36	
Tax discount type (percent, money/volume, surcharge) - 1 byte		CHAR
Assigned lines mask - 1 byte		CHAR
Tax/discount value - 4 bytes		FLOAT
Number non-zero flow - counts the number of 0.1 second periods with flow >0 for use in estimating average flow rate	2	USHORT
Bit field: #number of bits isVolumeOnly:1 (if Transaction is type 3 = 1) We have 16 bits available (2 bytes) Is TCProduct:1 IsOdometerUsed:1 IsPresetUsed:1 IsStarted:1 (used to control multiple delivery) IsStopped:1 (used to control multiple delivery) IsFirstPrint:1 (used to denote duplicate ticket print) IsBackedUp:1 (used to denote that it was saved to IB NVRAM) IsEncoderError:1 (used to indicate encoder sequence error) IsOverspeed:1 (used to indicate encoder too fast) UnusedBits:6 (unused bits)	2	USHORT
Tank ID[10]	10	CHAR
Tank ID Null Termination	2	CHAR
Total cost	8	DOUBLE
CRC of entire transaction data record	2	USHORT

Table 11. Transaction Record Table w/o Custom Fields (ID: ‘H’ – Answer ID: ‘I’)

Field Description	Size (Bytes)	Type
Total Bytes	146	

Table 12. Transaction Record Table w/o Custom Fields (ID: ‘J’ – Answer ID: ‘K’)

Field Description	Size (Bytes)	Type
Ticket #	4	LONG
Transaction Type 0 = Single delivery Record 1 = Multiple Delivery Record 2 = Summary Record 3 = Calibration Record	2	CHAR
Index: 0 for single, 1-N for multiple, -1 for summary	1	CHAR
# of summary records	1	CHAR
# of records summarized	1	CHAR
Product ID 0-2 (MAX_PRODUCTS - 1)	1	UCHAR
Product Info	15	CHAR
Product Info Null Termination	1	CHAR
Field Description	Size (Bytes)	Type
Start time date Minute 00-59 Hour 00-23 Day of month 01-31 Second 00-59 Month 01-12 (01=January) Year 00-255 (2000-2255)	 1 1 1 1 1 1	 UCHAR UCHAR UCHAR UCHAR UCHAR UCHAR
Finish time date - Same format as Start time date	6	
Tank Load	4	FLOAT
Subtotal	4	FLOAT
TotalizerStart	8	DOUBLE
TotalizerEnd	8	DOUBLE
GrossVolume. This is always the raw, uncompensated volume	8	DOUBLE
Volume. This is either gross or TC, depending on fuel	8	DOUBLE
Average temp	4	FLOAT

Table 12. Transaction Record Table w/o Custom Fields (ID: ‘J’ – Answer ID: ‘K’)

Field Description	Size (Bytes)	Type
Unit price (if Transaction type is 3 = New calibration factor)	4	FLOAT
Six individual tax/discount fields, each 6 bytes (if Transaction type is 3 = unused) Tax discount type (percent, money/volume, surcharge) - 1 byte Assigned lines mask - 1 byte Tax/discount value - 4 bytes	36	 CHAR CHAR FLOAT
Number non-zero flow - counts the number of 0.1 second periods with flow >0 for use in estimating average flow rate	2	USHORT
Bit field: #number of bits isVolumeOnly:1 (if Transaction is type 3 = 1) We have 16 bits available (2 bytes) Is TCProduct:1 IsOdometerUsed:1 IsPresetUsed:1 IsStarted:1 (used to control multiple delivery) IsStopped:1 (used to control multiple delivery) IsFirstPrint:1 (used to denote duplicate ticket print) IsBackedUp:1 (used to denote that it was saved to IB NVRAM) IsEncoderError:1 (used to indicate encoder sequence error) IsOverspeed:1 (used to indicate encoder too fast) UnusedBits:6 (unused bits)	2	USHORT
Tank ID[10]	10	CHAR
Tank ID Null Termination	2	CHAR
Total Cost	8	DOUBLE
Custom Field 1: Customers use these Custom Fields (1 to 7) for different purposes, the names of these data fields should be customizable (Plus Null).	13+1	CHAR
Custom Field 2	13+1	CHAR
Custom Field 3	8+1	CHAR
Custom Field 4	6+1	CHAR
Custom Field 5	6+1	CHAR
Custom Field 6	6+1	CHAR
Custom Field 7	6+1	CHAR
Null Termination	1	CHAR
CRC of entire transaction data record	2	USHORT
Total Bytes	214	

Table 13. Get Register Configuration: ‘E’ – Set Register Configuration: ‘D’

Binary Code	Description	Parameters	C&C Mode Enabled
0	Price configuration	1 byte CHAR Price code index (1-15) of price code of interest 1 byte CHAR Tax code index (1-15) assigned to this price code 4 bytes FLOAT raw price value	No
1	Tax code configuration	1 byte CHAR tax code index (1-15) of tax code of interest 1 byte CHAR tax type (See Table 14) 1 byte CHAR tax line mask (see Table 15) 4 bytes FLOAT amount of tax for line 1 1 byte CHAR tax type (See Table 14) 1 byte CHAR tax line mask (see Table 15) 4 bytes FLOAT amount of tax for line 2 1 byte CHAR tax type (See Table 14) 1 byte CHAR tax line mask (see Table 15) 4 bytes FLOAT amount of tax for line 3 1 byte CHAR tax type (See Table 14) 1 byte CHAR tax line mask (see Table 15) 4 bytes FLOAT amount of tax for line 4 1 byte CHAR tax type (See Table 14) 1 byte CHAR tax line mask (see Table 15) 4 bytes FLOAT amount of tax for line 5 1 byte CHAR tax type (See Table 14) 1 byte CHAR tax line mask (see Table 15) 4 bytes FLOAT amount of tax for line 6	No
2	Delivery report format	Null terminated string of report codes (see Table 18 for report code definition) UCHAR formatCode[N] where n<50.	Yes
3	Shift Report format	Null terminated string of report codes (see Table 19 for report code definition) UCHAR formatCode[N] where N<50.	Yes
4	Reserved	N/A	
5	Tank ID default	Null terminated ASCII string. UCHAR TankID[N] where N<=10.	Yes

Table 13. Get Register Configuration: ‘E’ – Set Register Configuration: ‘D’

Binary Code	Description	Parameters	C&C Mode Enabled
6	Default Volumes	1 byte Defaults Mode: $00 – Volume Presets Disabled Currency Presets Disabled Batch Presets Disabled $10/$50 – Volume Presets Enabled Currency Presets Disabled Batch Presets Disabled $20 – Volume Presets Disabled Currency Presets Enabled Batch Presets Disabled $30/$70 – Volume Presets Enabled Currency Presets Enabled Batch Presets Enabled 4 bytes FLOAT Product 1 Volume Preset 4 bytes FLOAT Product 2 Volume Preset 4 bytes FLOAT Product 3 Volume Preset 4 bytes FLOAT Product 1 Currency Preset 4 bytes FLOAT Product 2 Currency Preset 4 bytes FLOAT Product 3 Currency Preset 4 bytes FLOAT Tank Load Preset 4 bytes FLOAT Batch Volume Preset 1 4 bytes FLOAT Batch Volume Preset 2 4 bytes FLOAT Batch Volume Preset 3 4 bytes FLOAT Batch Volume Preset 4 4 bytes FLOAT Batch Volume Preset 5 4 bytes FLOAT Batch Volume Preset 6 4 bytes FLOAT Batch Volume Preset 7 4 bytes FLOAT Batch Volume Preset 8	No
7	Relay Control Timing	4 bytes FLOAT Slow Relay Open Volume 4 bytes FLOAT Stop Relay Open Volume	No
8	Security Enable	1 byte UCHAR enable (0=no, 1=yes)	No
9	Printer Option	1 byte UCHAR Printer Status Bit Mask 0x01 = Printer Enable 0x02 = Slip Printer (0=roll printer, 1=slip printer) 0x04 = Font Size (0=5x5, 1=7x7 only valid for slip printer) 1 byte UCHAR Printer Advance 1 byte UCHAR Print Test Page; (Always 0 on get)	No
10	Shift Start Information	1 byte UCHAR hour (100 if unset) 1 byte UCHAR minute (100 if unset) 1 byte UCHAR Product Used Bit Map (0x01=product 1, etc.) 4 bytes FLOAT load 8 bytes DOUBLE Begin Volume 8 bytes DOUBLE End Volume	No
11	Shift End Information	(No Get functionality – Set only)	No

Table 13. Get Register Configuration: ‘E’ – Set Register Configuration: ‘D’

Binary Code	Description	Parameters	C&C Mode Enabled
12	Shift Report (for Printing)	1 byte UCHAR Shift (1-4, 0 if shift not used) 1 byte UCHAR Product (1-8, 0 if no delivery on product) 1 byte UCHAR Begin Time Century 1 byte UCHAR Begin Time Second 1 byte UCHAR Begin Time Minute 1 byte UCHAR Begin Time Hour 1 byte UCHAR Begin Time DayOfWeek 1 byte UCHAR Begin Time DayOfMonth 1 byte UCHAR Begin Time Month 1 byte UCHAR Begin Time Year 1 byte UCHAR End Time Century 1 byte UCHAR End Time Second 1 byte UCHAR End Time Minute 1 byte UCHAR End Time Hour 1 byte UCHAR End Time DayOfWeek 1 byte UCHAR End Time DayOfMonth 1 byte UCHAR End Time Month 1 byte UCHAR End Time Year 8 bytes DOUBLE Begin Volume 4 bytes FLOAT load 8 bytes DOUBLE End Volume 8 bytes DOUBLE Total Gross Volume 2 bytes SHORT Number of Single Deliveries 2 bytes USHORT Number of Multiple Deliveries 2 bytes USHORT Number of Itemized Deliveries 8 bytes DOUBLE Total Volume 8 bytes DOUBLE Total Volume Priced 8 bytes DOUBLE Total No Tax Discount 8 bytes DOUBLE Total Tax Discount Line 1 8 bytes DOUBLE Total Tax Discount Line 2 8 bytes DOUBLE Total Tax Discount Line 3 8 bytes DOUBLE Total Tax Discount Line 4 8 bytes DOUBLE Total Tax Discount Line 5 8 bytes DOUBLE Total Tax Discount Line 6 8 bytes DOUBLE Total Sales 2 bytes USHORT Number of Transfer Deliveries	No
13	Date Format	1 byte UCHAR format bits 0x01: year format (0=2 digit year, 1=4 digit year) 0x02: month format (0=2 digit month, 1=3 char month) 0x0C: date order (0=MDY, 1=YMD, 2=DMY) 0x30 separator char (0=’/’, 1=’-‘, 2=’.’)	No

Table 13. Get Register Configuration: ‘E’ – Set Register Configuration: ‘D’

Binary Code	Description	Parameters	C&C Mode Enabled
14	Time Delays	1 byte UCHAR which fields being returned are valid: 0x01: backlight delay 0x02: display delay 0x04: volume return delay 2 bytes USHORT backlight turn off delay in seconds 2 bytes USHORT display delay in seconds 2 bytes USHORT volume return delay in seconds	No
15	Language	2 bytes USHORT string selection bit mask 0x0001: Language Choice: 0=English, 1=French… ?!? 0x0002: PRICE UNITS 0x0004: CHOOSE PRICE nn 0x0008: PRESET CURRENCY UNITS 0x0010: CURRENCY REMAIN 0x0020: CURR PRESET n 0x0040: VOLUME UNITS 0x0080: PRESET VOLUME UNITS 0x0100: VOL PRESET n 0x0200: RATE MODE 0x0400: ENTER PRICE 0x0800: CLEAR PRICE 0x1000: CHANGE PRICE 0x2000: PRICE CODE nn 0x4000: NO PRICE 0x8000: PRICE nn 24 CHAR formatCode string (may or may not be 0 terminated)	Yes
16	Display Syntax Setup	1 byte UCHAR Decimal Symbol (0=’.’ 1=’,’) 1 byte UCHAR Thousand Symbol (0=’.’ 1=’,’ 2=’ ‘) 1 byte UCHAR Preset Volume Precision (0-3) 1 byte UCHAR Preset Currency Precision (0-3) 1 byte UCHAR Preset Price Precision (0-3) 1 byte UCHAR Register Volume Precision (0-3) 1 byte UCHAR Register Rate Precision (0-3) 1 byte UCHAR Register Currency Precision (0-3) 1 byte UCHAR Register Price Precision (0-3) 1 byte UCHAR Totalizer Volume Precision (0-3)	Yes
17	Fuel Source Description	1 byte UCHAR Product Descriptor (1-6) 1 byte UCHAR Product Type 15 byte CHAR name (0 terminated) 1 byte UCHAR is Temperature Compensated (0=no, 1=yes) 12 pairs of 4 bytes FLOAT Temperature(n) 4 bytes FLOAT Thermal Coefficient(n) 1 byte UCHAR Product Calibration Number 1 byte UCHAR Printer Format	Yes

Table 13. Get Register Configuration: 'E' – Set Register Configuration: 'D'

Binary Code	Description	Parameters	C&C Mode Enabled
18	Manual Meter Calibration	1 byte UCHAR Product Calibration Number 8 pairs of 4 bytes FLOAT Flow Rate(n) 4 bytes FLOAT Encoder Counts Per Unit Volume(n)	Yes
19	Multi-Auto Meter Calibration	1 byte UCHAR Product Calibration Number 1 byte UCHAR Product Calibration Rate 4 bytes FLOAT Flow Rate 4 bytes FLOAT Encoder Counts Per Unit Volume	Yes
20	Reserved	N/A	

21	I/O Configuration	1 byte UCHAR I/O Flags Bitmap
		0x01 = Emergency Stop Supported?
		0x02 = Remote Start/Stop Supported?
		0x04 = Printer Interlock Supported?
		0x08 = Preset Interlock Supported?
		0x10 = Pulse Output Supported?
		0x20 = Is A Remote Display?
		0x40 = Is Master Address Set? // Only if Remote Display
		0x80 = Transfer Interlock Supported?
		4 bytes FLOAT Pulses Per Unit Volume
		1 byte UCHAR Master Address
			Yes

22	OEM Message	1 byte UCHAR Is OEM Messsage Enabled? (0=no, 1=yes) 24 bytes CHAR OEM Message // null at end not guaranteed	Yes

Table 13. Get Register Configuration: ‘E’ – Set Register Configuration: ‘D’

Binary Code	Description	Parameters	C&C Mode Enabled
23	Delivery Options	1 byte UCHAR I/O Flags Bitmap 0x01 = Delivery Pricing Supported? (0=no, 1=yes) 0x02 = Delivery Timeout Supported? (0=no, 1=yes) 0x04 = Tank ID Enabled? (0=no, 1=yes) 4 bytes LONG Pre-Flow Delivery Abort Timeout (seconds) 4 bytes LONG No-Flow Delivery Stop Timeout (seconds) 25 bytes CHAR Tank ID Descriptor (always has 0 termination) 11 bytes CHAR Default Tank ID (always has 0 termination) 1 byte UCHAR Custom Field Enable (0x01=field 1, 0x08=field 2) 1 byte UCHAR Custom Field Enable (0x01=field 3, 0x08=field 4) 1 byte UCHAR Custom Field Enable (0x01=field 4, 0x08=field 5) 1 byte UCHAR Custom Field Enable plus flags: 0x01 = field 6 0x08 = Tank Load Enable 0x10 = Gross Preset Enable 0x20 = Gross View Enable 0x40 = Multi-Calibrate Diff Enable 0x80 = Multiple Delivery Enable 14 bytes CHAR Custom Field 1 Name 14 bytes CHAR Custom Field 1 Default 14 bytes CHAR Custom Field 2 Name 14 bytes CHAR Custom Field 2 Default 14 bytes CHAR Custom Field 3 Name 14 bytes CHAR Custom Field 3 Default 14 bytes CHAR Custom Field 4 Name 8 bytes CHAR Custom Field 4 Default 14 bytes CHAR Custom Field 5 Name 7 bytes CHAR Custom Field 5 Default 14 bytes CHAR Custom Field 6 Name 7 bytes CHAR Custom Field 6 Default 14 bytes CHAR Custom Field 7 Name 7 bytes CHAR Custom Field 7 Default	Yes
24	Printed Report Format	Null terminated string of report codes (see Table 20 for report code definition) UCHAR formatCode[N] where N<75.	Yes
25	Printer Label	1 byte UCHAR Printer Label ID (see Table LID) 24 bytes CHAR Label (may not be 0 terminated)	Yes
26	Temperature Setup	1 byte UCHAR Is Fahrenheit (0=Centigrade, 1=Fahrenheit) 1 byte UCHAR Is Density Temperature Compensated (0=no,1=yes) 4 bytes FLOAT Reference Temperature 4 bytes FLOAT Thermistor Calibration Temperature	Yes
27	Restart Command	N/A (No Get functionality – Set only)	Yes
28	Security Codes	6 byte CHAR security code (not 0 terminated) 2 bytes USHORT Security Mask ()	Yes

Table 13. Get Register Configuration: ‘E’ – Set Register Configuration: ‘D’

Binary Code	Description	Parameters	C&C Mode Enabled
29	Configure Port	1 byte UCHAR Port Assignment	v
30	Restore Records	N/A (No Get functionality – Set only)	No
31	Print Record	N/A (No Get functionality – Set only)	No
32	Meter ID and Input type	15 bytes CHAR Meter ID (0 terminated) 1 byte UCHAR Input Type ()	Yes
33	View Record	{ packed transaction record }	No
34	PIN Enable	1 byte UCHAR PIN Enable (0=disable, 1=enable)	Yes
35	PIN Code Setup	7 bytes CHAR PIN Code (6 byte length string, 0 terminated)	No
36	Set Delivery Options	N/A (No Get functionality – Set only)	No
37	Request Authorization For All Deliveries state	0 = authorization not required for all deliveries 1 = authorization required for all deliveries	No

Table 14. Tax Discount Types

Type	Value
Tax discount unused	-1
Tax percent	0
Tax per unit volume	1
Tax surcharge	2
Discount percent	3
Discount per unit volume	4
Discount surcharge	5

Table 15 shows the meaning of each bit in the mask (left of the thick line). It also shows the set of bits each tax line can use (right of thick line). Tax line 2 can only be applied to Line 1 and Unit price (bit 0 and 1) resulting in a YES in the first two rows. Tax line 2 can not be applied to Tax lines 3 - 6 that follow. Note: none of the Tax lines can be applied to Tax line 6, which is why the mask does not include a bit for line 6. However, line 6 can be applied to the Unit price and all lines that precede it.

Table 15. Tax Mask Table

Mask Definition		Mask bit can be used by Tax Line					
Bit	Tax applied to	1	2	3	4	5	6
0	Unit price	Yes	Yes	Yes	Yes	Yes	Yes
1	Line 1	No	Yes	Yes	Yes	Yes	Yes
2	Line 2	No	No	Yes	Yes	Yes	Yes
3	Line 3	No	No	No	Yes	Yes	Yes
4	Line 4	No	No	No	No	Yes	Yes
5	Line 5	No	No	No	No	No	Yes

Table 16. Print Device Control = ‘p’

Binary Code	Description	Parameters	Field Code
0	Printer Request - ready to print (see Figure 1).	None	Table 17: 0,1,2
1	Print Start, Beginning of data buffer	None	Table 7
2	Print Data, format and send a data buffer to printer. Repeat for additional data buffers (number constrained by printer’s memory).	n data byes n<= 150 bytes	Table 7
3	Print End, print set of data buffers sent - no more data buffers to follow, release printer.	1 Byte. Number buffers sent.	Table 17: 03 No error, Table 17: 04 - 08 Error
4	Flush Buffer. Same as P_END, printer not released, new P_START, P_DATA sequence to follow. Print buffer limited to 4096 bytes. To exceed this limit flush the buffers sent with P_DATA and then continue. Print does not occur until this command or P_END is sent.	Number of packets sent since Print Start was sent.	Table 17: 04-08 Error
5	Removal of paper (slip printer) and cut of paper (roll printer) controlled by OBC, not automatic by printer driver. Sent after Print Request before Print Start. Power up default is Off. Only needs to be turned On once after power up. Allowable sequence to set remote operation: Print Request, Print Remote, Print End.	1 byte - 0=Off (normal operation) 1=On (remote operation)	Table 17: 09

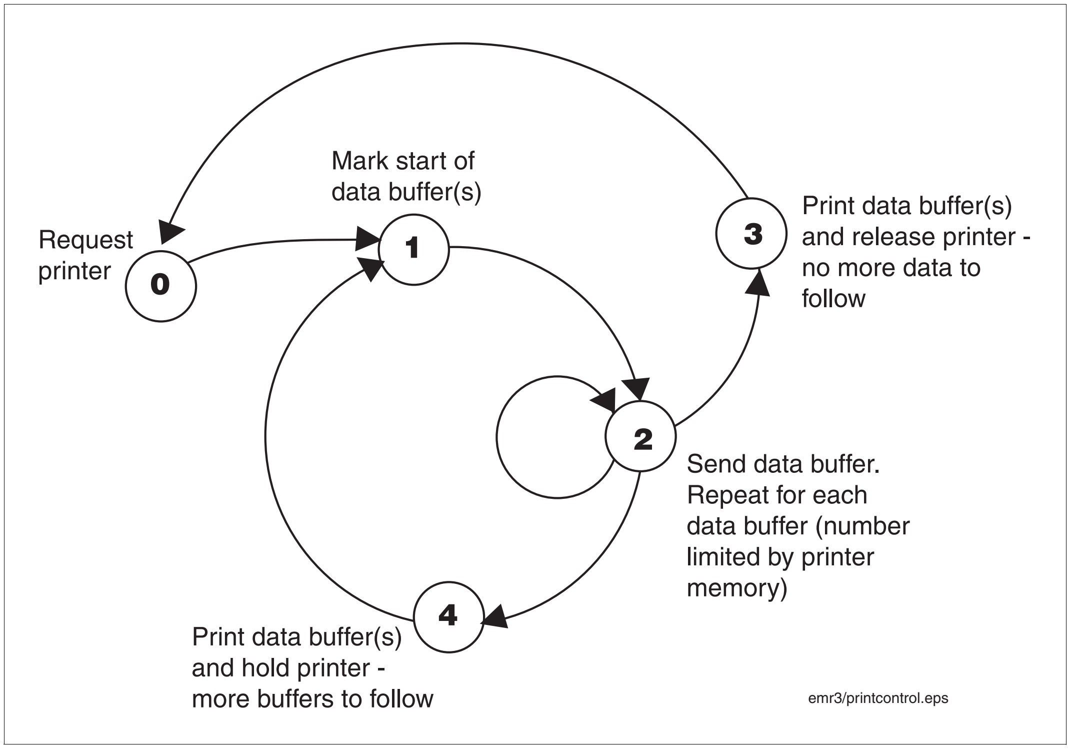

Request printer
0
Mark start of data buffer(s)
1
2
Send data buffer. Repeat for each data buffer (number limited by printer memory)
3
Print data buffer(s) and release printer - no more data to follow
4
Print data buffer(s) and hold printer - more buffers to follow
emr3/printcontrol.eps

Figure 1. Print Device Control Flow Diagram

Table 17. Print Device Control Response = ID: ‘p’ – Answer ID: ‘p’

Binary Code	Description	Parameters
00	Printer granted	None
01	Printer busy	None
02	Printer needs service	None
03	Print complete	None
04	Print data error	None
05	Print comm abort	None
06	Print error abort	None
07	Print remove slip	None
08	Printer paper out	None
09	Print remote end	None
0A	Print data flush complete	None

Table 18. Delivery Report Line Codes

Report Line Description	Binary Code
Printer advance (must be first code)	1
Full row of asterisks	2
Blank line	3
Blank line with an asterisk at each end	4
1st line of header	5
2nd line of header	6
3rd line of header	7
4th line of header	8
1st line of header with an asterisk at each end	9
2nd line of header with an asterisk at each end	10
3rd line of header with an asterisk at each end	11
4th line of header with an asterisk at each end	12
1st line of trailer	13
2nd line of trailer	14
3rd line of trailer	15
4th line of trailer	16
Start time	17
End time	18
Start volume	19
Gross end volume	20
Net end volume	21
Delivery temperature (if TC)	22
Reference temperature (if TC)	23
Temperature coefficient (if TC)	24
Ticket number	25
Ticket status (e.g., multiple delivery, summary, ...)	26
Meter identification	27
Unused	28
Product name	29

Table 18. Delivery Report Line Codes

Report Line Description	Binary Code
Totalizer start volume	30
Totalizer end volume	31
Average flow rate	32
Gross price per unit volume	33
Total price without taxes/discounts	34
Taxes/discounts	35
net price per unit volume	36
total of all taxes	37
Total of all non-taxes (e.g., discounts, surcharges, ...)	38
Total price with all taxes	39
Total price with all non-taxes	40
Grand total	41
Reserved	42
Error message (when required)	43

Table 19. Shift Report Line Codes

Report Line Description	Binary Code
Printer advance (must be first line)	1
Full row of asterisks	
1st line of header with an asterisk at each end	3
2nd line of header with an asterisk at each end	4
3rd line of header with an asterisk at each end	5
4th line of header with an asterisk at each end	6
1st line of header	7
2nd line of header	8
3rd line of header	9
4th line of header	10
Blank line	11

Table 19. Shift Report Line Codes

Report Line Description	Binary Code
Starting date-time of shift	12
Ending date-time of shift	13
Reserved (Internal Use)	14
	15
	16
Product name	17
Meter identification	18
Totalizer volume at start of shift	19
Loaded volume	20
Totalizer volume at end of shift	21
Total gross volume	22
Number of single deliveries	23
Number of multiple deliveries	24
Total number of deliveries (single + multiple)	25
Total net volume	26
Total unpriced volume	27
Total priced volume	28
Total sales without taxes/discounts	29
Total sales contribution from tax line 1	30
Total sales contribution from tax line 2	31
Total sales contribution from tax line 3	32
Total sales contribution from tax line 4	33
Total sales contribution from tax line 5	34
Total sales contribution from tax line 6	35
Grand total	36
1st line of trailer	37
2nd line of trailer	38
3rd line of trailer	39
4th line of trailer	40

Table 20. Product Type: Aux_Dev

Binary Code	Description/Name
0	GASOLINE
1	DIESEL
2	LPG
3	KEROSENE
4	AV_GAS
5	FUEL_OIL
6	CRUDE_OILS
7	LUBRICATING_OILS
8	USER_DEFINED
9	NO_PRODUCT_DEFINED

Table 21. List Of String Numbers And Corresponding Internal String Name

Code	String Description
00	STR_UNUSED
01	STR_RPT_START_TIME
02	STR_RPT_START_TIME
03	STR_RPT_VOLUME_START
04	STR_RPT_NET_VOLUME_FINISH (hidden: STR_RPT_GROSS_VOLUME_FINISH)
05	STR_RPT_GROSS_VOLUME_FINISH (hidden: STR_RPT_NET_VOLUME_FINISH)
06	STR_RPT_AVERAGE_TEMP
07	STR_RPT_REFERENCE_TEMP
08	STR_RPT_COEF_TEMP
09	STR_RPT_DENSITY
10	STR_RPT_TICKET_NUMBER
11	STR_RPT_CALIB_TICKET
12	STR_RPT_SUMMARY_TICKET
13	STR_RPT_SUMMARY_TICKET_OF
14	STR_RPT_TRANSFER_TICKET
15	STR_RPT_METER_ID

Table 21. List Of String Numbers And Corresponding Internal String Name

Code	String Description
16	STR_RPT_MULT_DEL_TICKET
17	STR_TANK_ID
18	STR_TAX_PERCENT
19	STR_TAX_PER_VOLUME
20	STR_TAX_SURCHARGE
21	STR_DIS_PERCENT
22	STR_DIS_PER_VOLUME
23	STR_DIS_SURCHARGE
24	STR_ASSIGN_TO_LINE
25	STR_ENTER_NEW_TAX
26	STR_ENTER_NEW_DIS
27	STR_UNUSED
28	STR_RPT_START_TIME
29	STR_RPT_FINISH_TIME
30	STR_RPT_VOLUME_START
31	STR_RPT_GROSS_VOLUME_FINISH
32	STR_RPT_NET_VOLUME_FINISH
33	STR_RPT_AVERAGE_TEMP
34	STR_RPT_REFERENCE_TEMP
35	STR_RPT_COEF_TEMP
36	STR_RPT_DENSITY
37	STR_RPT_TICKET_NUMBER
38	STR_RPT_CALIB_TICKET
39	STR_RPT_SUMMARY_TICKET
40	STR_RPT_SUMMARY_TICKET_OF
41	STR_RPT_TRANSFER_TICKET
42	STR_RPT_METER_ID
43	STR_RPT_MULT_DEL_TICKET
44	STR_RPT_PROD_DSCRPT
45	STR_RPT_SHIFT_BEGIN

Table 21. List Of String Numbers And Corresponding Internal String Name

Code	String Description
46	STR_RPT_SHIFT_END
47	STR_RPT_ODOMETER_START
48	STR_RPT_ODOMETER_END
49	STR_RPT_MILES_DRIVEN
50	STR_RPT_BEGINNING_VOLUME
51	STR_RPT_LOADED_VOLUME
52	STR_RPT_ENDING_VOLUME
53	STR_RPT_TOTAL_GROSS_VOLUME
54	STR_RPT_SINGLE_DELIVERIES
55	STR_RPT_MULTIPLE_DELIVERIES
56	STR_RPT_ITEMIZED_DELIVERIES (hidden: STR_RPT_TRANSFER_DELIVERIES)
57	STR_RPT_TRANSFER_DELIVERIES (hidden: STR_RPT_ITEMIZED_DELIVERIES)
58	STR_RPT_TOTAL_TC_VOLUME
59	STR_RPT_TOTAL_VOLUME
60	STR_RPT_TOTAL_TC_VOL_NO_PRICE
61	STR_RPT_TOTAL_VOL_NO_PRICE
62	STR_RPT_TOTAL_TC_VOL_PRICED
63	STR_RPT_TOTAL_VOL_PRICED
64	STR_RPT_TOTAL_NO_TD_SALES
65	STR_RPT_TOTAL_TAX_DISCOUNT1
66	STR_RPT_TOTAL_TAX_DISCOUNT2
67	STR_RPT_TOTAL_TAX_DISCOUNT3
68	STR_RPT_TOTAL_TAX_DISCOUNT4
69	STR_RPT_TOTAL_TAX_DISCOUNT5
70	STR_RPT_TOTAL_TAX_DISCOUNT6
71	STR_RPT_TOTALIZER_START
72	STR_RPT_TOTALIZER_END
73	STR_RPT_AVG_FLOW_RATE
74	STR_RPT_UNIT_PRICE
75	STR_RPT_FINAL_PRICE

Table 21. List Of String Numbers And Corresponding Internal String Name

Code	String Description
76	STR_RPT_SUBTOTAL
77	STR_RPT_PRICE_WITH_TAX
78	STR_RPT_PRICE_WITH_NON_TAX
79	STR_RPT_GRAND_TOTAL
80	STR_RPT_MULT_DELIVERY_NOTE
81	STR_ON_ONE_SITE
82	STR_RPT_DUPLICATE_TICKET
83	STR_RPT_TRANSFER_TICKET
91	STR_RPT_CF1_ID
92	STR_RPT_CF2_ID
93	STR_RPT_CF3_ID
94	STR_RPT_CF4_ID
95	STR_RPT_CF5_ID
96	STR_RPT_CF6_ID
97	STR_RPT_CF7_ID
98	STR_RPT_T_D_SUBTOTAL
99	STR_RPT_TANK_LOAD

Sample Commands

1. Set a product (index 0) as a current product for meter address 01:

   7E 01 FF 53 70 00 3D 7E

   where:

   7E - delimiter flag

   01 - destination address (meter 1);

   FF - source address (OBC);

   53 - 'S' - set the meter field command (from Table 4 on page 5)

   70 - 'p' current (active product) (from Table 5 on page 6)

   00 - product index

   3D - check sum calculated as: (BYTE)(0 - (01+FF+53+70+00)

2. Read (Get) the meter (address 1) current product:

   7E 01 FF 47 70 49 7E

3. Meter answers to the "Read current product" command:

   7E FF 01 46 70 00 4A 7E

Appendix A: OBC Serial Pass-Through Printing Commands

This document explains how to use the pass-through feature of the EMR3 product. It assumes that the reader has a working knowledge of the EMR serial commands, as described in the OBC Serial Command Protocol document, Veeder-Root part number 577013-790. If you do not have this appendix, please obtain it as information it contains is not duplicated in this document.

Overview of Pass-Through Printing

The pass-through printing feature allows another party to add lines of text both before and after the normal EMR printout. This feature will work on both the Roll and Slip printers that the EMR supports.

The pass-through printing feature works as follows:

1. The OBC requests control of the printer
2. The OBC sends data to the EMR printer buffer
3. The EMR prints the data
4. The OBC closes the connection

There are a two important size limits that must be taken into account.

1. The OBC commands used to send data to the EMR print buffer can only have 150 bytes of data. Multiple commands must be used to send more data.
2. The print buffer is only 4KB in length. To print more data, the buffer must be flushed, and then more data sent.

In order to work within these size limits, the protocol has the commands defined in Table 1. Using the example above, the commands are used as follows:

1. The OBC requests control of the printer
    a. Printer Request
2. The OBC sends data to the EMR printer buffer
3. The EMR prints the data
    a. Print Start
    b. One or more Print Data
    c. **Either** Print Flush, go to 2.a,
    d. **OR** Print End
4. The OBC closes the connection

Note that the destination address of the print commands is the printer, not the EMR or IB.

The state machine that the OBC should follow is shown in Figure A-1.

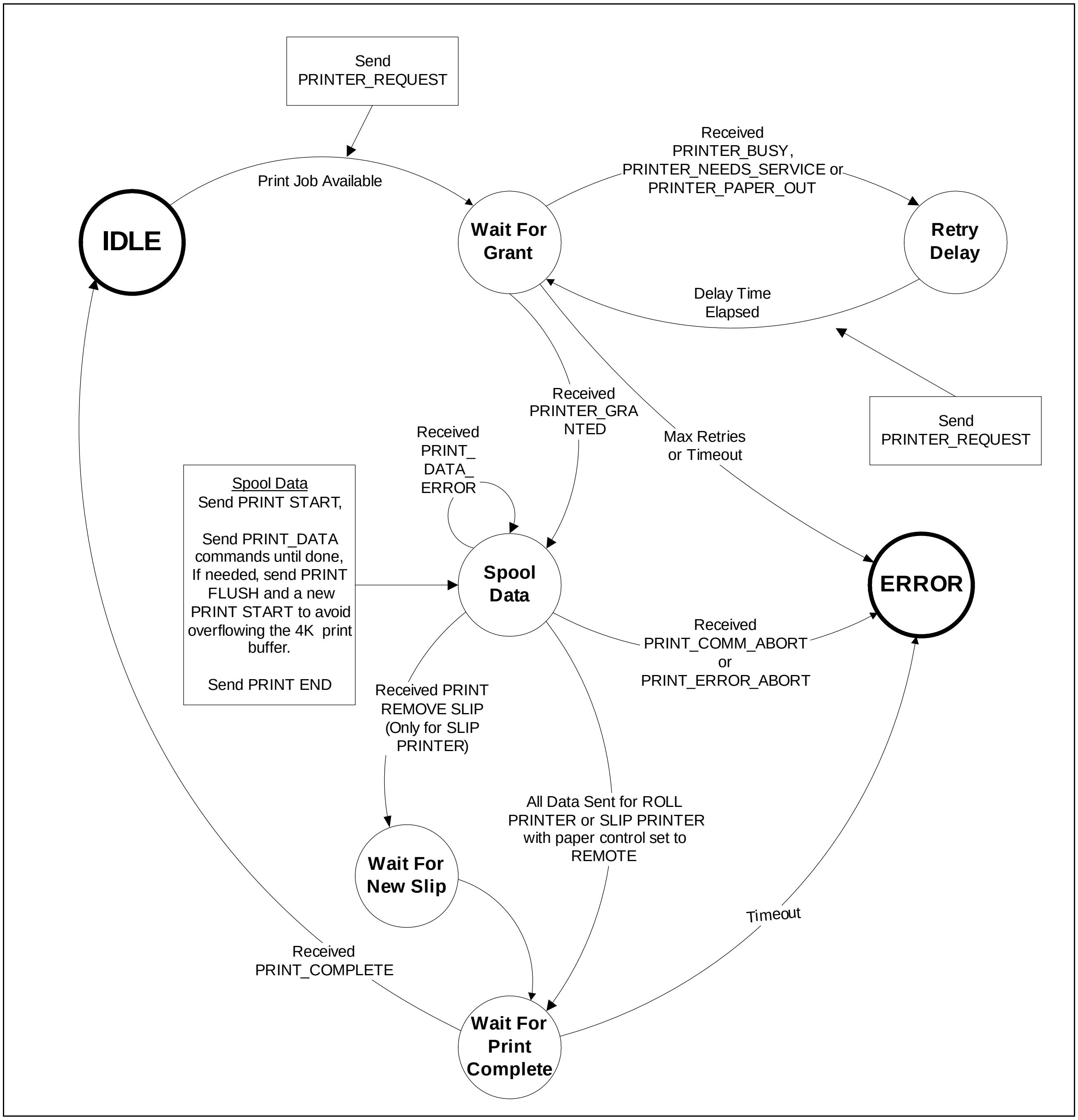

Send PRINTER_REQUEST
Print Job Available
IDLE
Wait For Grant
Received PRINTER_BUSY, PRINTER_NEEDS_SERVICE or PRINTER_PAPER_OUT
Retry Delay
Delay Time Elapsed
Send PRINTER_REQUEST
Received PRINTER_GRANTED
Max Retries or Timeout
Received PRINT_DATA_ERROR
Spool Data
Send PRINT START,
Send PRINT_DATA commands until done, If needed, send PRINT FLUSH and a new PRINT START to avoid overflowing the 4K print buffer.
Send PRINT END
Spool Data
ERROR
Received PRINT_COMM_ABORT or PRINT_ERROR_ABORT
Received PRINT REMOVE SLIP (Only for SLIP PRINTER)
All Data Sent for ROLL PRINTER or SLIP PRINTER with paper control set to REMOTE
Wait For New Slip
Timeout
Received PRINT_COMPLETE
Wait For Print Complete

Figure A-1. OBC State Machine For Pass-Through Printing

Important notes on the communications shown in Figure A-1 are discussed in the steps below.

1. There are 2 possible paths to take once the Print End is sent, depending on whether the printer is a Slip Printer or a Roll Printer (or a Slip Printer with printer paper control set to “remote”). Please pay attention to this, because the print complete message may not come when expected.

2. Once the printer is granted to the OBC, all subsequent print commands must be sent with less than 2 seconds between them (less than 5 seconds in version F08 or more recent version). Taking longer than this between messages is a protocol error.
3. If 2 seconds or more (5 seconds in version F08 or more recent version) pass between print commands, the EMR3 will send an unsolicited DATA ERROR reply. It will do this twice, once every 2 seconds, then the EMR3 will rescind the grant, send a COMM ABORT, and flush the print buffer.
4. The rationale for the DATA ERROR message is that the EMR3 assumes that the OBC message it was waiting for was lost, and it wants the OBC to resend it. Please pay attention to this, because slow OBC timing can result in apparent command failures (DATA ERROR replies). [NOTE: In future versions of the protocol, an explicit “TIMEOUT” message (6) will be added to the protocol for this condition.]
5. Once the error state is entered, normal operation is not possible without external intervention. Entering the error state means that a serious problem has occurred, and this can only be resolved by external intervention. To clear the error state, it is necessary to disable printing, and then re-enable it. This will re-initialize the print engine.

Pass-Through Printing Command Tables

Print Device Control Field Codes are shown in Table A-1.

Table A-1. Print Device Control Field Codes

Code	Description	Parameters	Response Fields
0	Printer Request If printer is available it will be granted and ACK returned. If printer is unavailable (in use), NAK will be returned.	None	Table A-2 Codes 0,1,2
1	Print Start. Initializes the print buffer.	None	ACK/NAK
2	Print Data. Data will be stored in the print buffer (4096 bytes) until sent to printer.	n data byes n<= 150 bytes	ACK/NAK
3	Print End of Data. Argument is the number of Print Data commands sent since Print Start was sent. If this does not agree with the number the EMR received, a NAK will be sent. OBC should send Print Start, and resend the data. If number is the same, the buffer will be printed, and printer released.	1 Byte Number of data blocks sent.	Table A-2 Code 3 No Error Codes 4 – 8 Error
4	Print Flush Buffer. Same as Print End of Data, however, the printer will **not** be released. The EMR will print the current buffer, and will expect a new Print Start and Print Data to follow.	Number of packets sent since Print Start was sent.	Table A-2 Code A No Error Codes 4 – 8 Error

Table A-1. Print Device Control Field Codes

5	Print Paper Control Enables the OBC control of the removal of paper (slip printer) and cut of paper (roll printer). Normal operation is OBC control OFF, where the EMR will control paper removal. Only needs to be turned on once after power up. This command is only valid after a Printer Request.	1 byte 0 – OFF normal operation 1 – ON Remote op	Table A-2 Code 9

Status bytes used in the ‘p’ reply are shown in Table A-2.

Table A-2. Status Bytes Used In The ‘p’ Reply

Code	Description	Parameters
00	PRINTER GRANTED	NONE
01	PRINTER BUSY	NONE
02	PRINTER NEEDS SERVICE	NONE
03	PRINT COMPLETE	NONE
04	PRINT DATA ERROR	NONE
05	PRINT COMM ABORT	NONE
06	PRINT ERROR ABORT	NONE
07	PRINT REMOVE SLIP	NONE
08	PRINTER PAPER OUT	NONE
09	PRINT REMOTE END	NONE
0A	PRINT DATA FLUSH	NONE

Examples of Pass-Through Printing

Communication Example 1. Less than 4K of data

This is the general structure of the commands sent and replies received. This is an overview. The actual bytes transmitted are shown below.

```
Printer Request-->
                              <-- Printer Granted
Print Start-->
                              <-- ACK
Print Data-->
                              <-- ACK
```

Print Data-->

<-- ACK

Print Data-->

<-- ACK

Print Data-->

<-- ACK

Print End-->

<-- Print Complete

Example showing only the command & reply bytes.

Transmit Data	**Received Data**
<Printer Request> $70,$00	<Printer granted> $70,$00
<Print Start> $70,$01	<CMD ACK> $41,$00
<Print Data> $70,$02,$2A,$2A,$2A,$20,$44,$49,$52,$45,$43,$54,$20,$50,$52,$49,$4E,$54,$20,$54,$45,$53,$54,$20,$2A,$2A,$2A,$0D,$0A,$0D,$0A	<CMD ACK> $41,$00
<Print Data> $70,$02,$2A,$2A,$20,$50,$52,$49,$4E,$54,$20,$54,$45,$53,$54,$20,$4C,$49,$4E,$45,$20,$31,$20,$2A,$2A,$0D,$0A	<CMD ACK> $41,$00
<Print Data> $70,$02,$2A,$2A,$20,$50,$52,$49,$4E,$54,$20,$54,$45,$53,$54,$20,$4C,$49,$4E,$45,$20,$32,$20,$2A,$2A,$0D,$0A	<CMD ACK> $41,$00
<Print Data> $70,$02,$2A,$2A,$2A,$20,$44,$49,$52,$45,$43,$54,$20,$50,$52,$49,$4E,$54,$20,$54,$45,$53,$54,$20,$45,$4E,$44,$20,$2A,$2A,$2A,$0D,$0A,$0D,$0A,$0D,$0A,$0D,$0A	<CMD ACK> $41,$00
<Print End> $70,$03,$04	<Print complete> $70,$03

Example showing the full packets with packet start & end bytes, addressing bytes, checksums, and the command & reply bytes.

Full Transmit Data	**Full Received Data**
<Printer Request> $7E,$41,$FF,**$70,$00**,$50,$7E	<Printer granted> $FF,$41,**$70,$00**,$50
<Print Start> $7E,$41,$FF,**$70,$01**,$4F,$7E	<CMD ACK> $FF,$C1,**$41,$00**,$FF

<Print Data> $7E,$41,$FF,**$70,$02,$2A,$2A,$2A,$20,$44,$49,$52,$45,$43,$54,$20,$50,$52,$49,$4E,$54,$20,$54,$45,$53,$54,$20,$2A,$2A,$2A,$0D,$0A,$0D,$0A**,$1C,$7E	<CMD ACK> $FF,$C1,**$41,$00**,$FF
<Print Data> $7E,$41,$FF,**$70,$02,$2A,$2A,$20,$50,$52,$49,$4E,$54,$20,$54,$45,$53,$54,$20,$4C,$49,$4E,$45,$20,$31,$20,$2A,$2A,$0D,$0A**,$C9,$7E	<CMD ACK> $FF,$C1,**$41,$00**,$FF
<Print Data> $7E,$41,$FF,**$70,$02,$2A,$2A,$20,$50,$52,$49,$4E,$54,$20,$54,$45,$53,$54,$20,$4C,$49,$4E,$45,$20,$32,$20,$2A,$2A,$0D,$0A**,$C8,$7E	<CMD ACK> $FF,$C1,**$41,$00**,$FF
<Print Data> $7E,$41,$FF,**$70,$02,$2A,$2A,$2A,$20,$44,$49,$52,$45,$43,$54,$20,$50,$52,$49,$4E,$54,$20,$54,$45,$53,$54,$20,$45,$4E,$44,$20,$2A,$2A,$2A,$0D,$0A,$0D,$0A,$0D,$0A,$0D,$0A**,$F7,$7E	<CMD ACK> $FF,$C1,**$41,$00**,$FF
<Print End> $7E,$41,$FF,**$70,$03,$04**,$49,$7E	<Print complete> $FF,$41,**$70,$03**,$4D

Communication Example 2. More than 4K of data

Printer Request -->

<-- Printer Granted

Print Start -->

<-- ACK

Print Data -->

<-- ACK

Print Data -->

<-- ACK

Print Flush -->

<-- ACK

Print Start -->

<-- ACK

Print Data -->

<-- ACK

Print Data -->

<-- ACK

Print End -->

<-- Print Complete

Example showing only the command & reply bytes.

Transmit Data	Received Data
<Printer Request> $70,$00	<Printer Granted> $70,$00
<Print Start> $70,$01	<CMD ACK> $41,$00
<Print Data> $70,$02,$2A,$2A,$2A,$20,$44,$49,$52,$45,$43,$54,$20,$50,$52,$49,$4E,$54,$20,$54,$45,$53,$54,$20,$2A,$2A,$2A,$0D,$0A,$0D,$0A	<CMD ACK> $41,$00
<Print Data> $70,$02,$2A,$2A,$20,$50,$52,$49,$4E,$54,$20,$54,$45,$53,$54,$20,$4C,$49,$4E,$45,$20,$31,$20,$2A,$2A,$0D,$0A	<CMD ACK> $41,$00
<Print Flush> $70,$04,$02	<Print flush> $70,$0A
<Print Start> $70,$01	<CMD ACK> $41,$00
<Print Data> $70,$02,$2A,$2A,$20,$50,$52,$49,$4E,$54,$20,$54,$45,$53,$54,$20,$4C,$49,$4E,$45,$20,$32,$20,$2A,$2A,$0D,$0A	<CMD ACK> $41,$00
<Print Data> $70,$02,$2A,$2A,$2A,$20,$44,$49,$52,$45,$43,$54,$20,$50,$52,$49,$4E,$54,$20,$54,$45,$53,$54,$20,$45,$4E,$44,$20,$2A,$2A,$2A,$0D,$0A,$0D,$0A,$0D,$0A,$0D,$0A	<CMD ACK> $41,$00
<Print End> $70,$03,$02	<Print Complete> $70,$03

Example showing the full packets with packet start & end bytes, addressing bytes, checksums, and the command & reply bytes.

Full Transmit Data	Full Received Data
<Printer Request> $7E,$41,$FF,$70,$00,$50,$7E	<Printer Granted> $FF,$41,$70,$00,$50
<Print Start> $7E,$41,$FF,$70,$01,$4F,$7E	<CMD ACK> $FF,$C1,$41,$00,$FF
<Print Data> $7E,$41,$FF,$70,$02,$2A,$2A,$2A,$20,$44,$49,$52,$45,$43,$54,$20,$50,$52,$49,$4E,$54,$20,$54,$45,$53,$54,$20,$2A,$2A,$2A,$0D,$0A,$0D,$0A,$1C,$7E	<CMD ACK> $FF,$C1,$41,$00,$FF

<Print Data> $7E,$41,$FF,$70,$02,$2A,$2A,$20,$50,$52,$49,$4E,$54,$20, $54,$45,$53,$54,$20,$4C,$49,$4E,$45,$20,$31,$20,$2A,$2A, $0D,$0A,$C9,$7E	<CMD ACK> $FF,$C1,$41,$00,$FF
<Print Flush> $7E,$41,$FF,$70,$03,$02,$47,$7E	<Print flush done> $FF,$41,$70,$0A,$4D
<Print Start> $7E,$41,$FF,$70,$01,$4F,$7E	<CMD ACK> $FF,$C1,$41,$00,$FF
<Print Data> $7E,$41,$FF,$70,$02,$2A,$2A,$20,$50,$52,$49,$4E,$54,$20, $54,$45,$53,$54,$20,$4C,$49,$4E,$45,$20,$32,$20,$2A,$2A, $0D,$0A,$C8,$7E	<CMD ACK> $FF,$C1,$41,$00,$FF
<Print Data> $7E,$41,$FF,$70,$02,$2A,$2A,$2A,$20,$44,$49,$52,$45,$43, $54,$20,$50,$52,$49,$4E,$54,$20,$54,$45,$53,$54,$20,$45, $4E,$44,$20,$2A,$2A,$2A,$0D,$0A,$0D,$0A,$0D,$0A,$0D,$0A, $F7,$7E	<CMD ACK> $FF,$C1,$41,$00,$FF
<Print End> $7E,$41,$FF,$70,$03,$02,$47,$7E	<Print Complete> $FF,$41,$70,$03,$4D

Communication Example 3. Slip Printer

Printer Request-->

<-- Printer Granted

Print Start-->

<-- ACK

Print Data-->

<-- ACK

Print End-->

<-- Print Remove Slip

(user removes slip and inserts new slip)

<-- Print Complete

Example showing only the command & reply bytes.

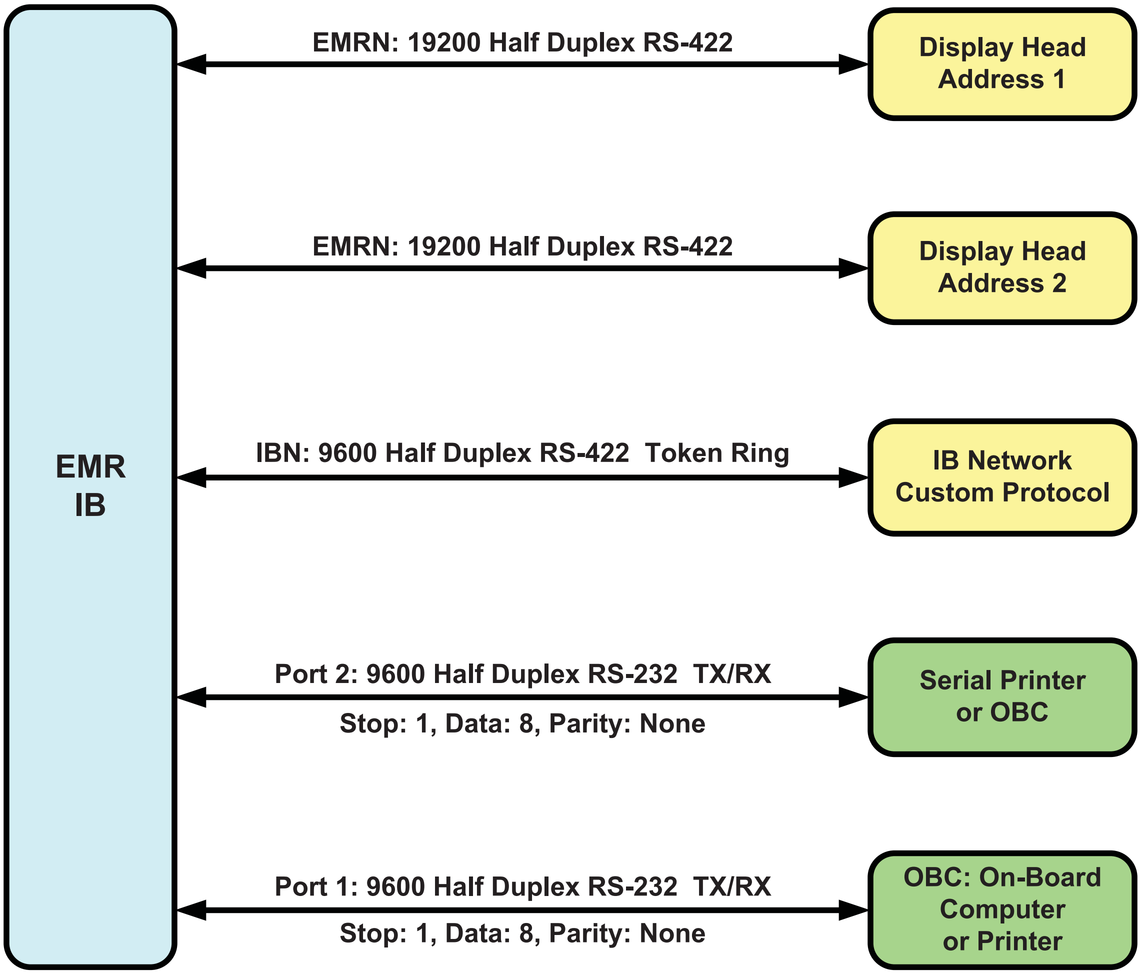
Transmit Data	**Received Data**
<Printer Request> $70,$00	<Printer granted> $70,$00
<Print Start> $70,$01	<CMD ACK> $41,$00
<Print Data> $70,$02,$2A,$2A,$2A,$20,$44,$49,$52,$45,$43,$54,$20,$50, $52,$49,$4E,$54,$20,$54,$45,$53,$54,$20,$2A,$2A,$2A,$0D, $0A,$0D,$0A	<CMD ACK> $41,$00

<Print End> $70,$03,$04	<Print Remove Slip> $70,$07
User removes slip and inserts new slip	<Print Complete> $70,$03

Example showing the full packets with packet start & end bytes, addressing bytes, checksums, and the command & reply bytes.

Full Transmit Data	**Full Received Data**
<Printer Request> $7E,$41,$FF,$70,$00,$50,$7E	<Printer Granted> $FF,$41,$70,$00,$50
<Print Start> $7E,$41,$FF,$70,$01,$4F,$7E	<CMD ACK> $FF,$C1,$41,$00,$FF
<Print Data> $7E,$41,$FF,$70,$02,$2A,$2A,$2A,$20,$44,$49,$52,$45,$43, $54,$20,$50,$52,$49,$4E,$54,$20,$54,$45,$53,$54,$20,$2A, $2A,$2A,$0D,$0A,$0D,$0A,$1C,$7E	<CMD ACK> $FF,$C1,$41,$00,$FF
<Print End> $7E,$41,$FF,$70,$03,$04,$49,$7E	<Print Remove Slip> $FF,$41,$70,$07,$49
User removes slip and inserts new slip	<Print Complete> $FF,$41,$70,$03,$4D

Appendix B: Serial Data Flow Diagram


EMR
IB
EMRN: 19200 Half Duplex RS-422
Display Head Address 1
EMRN: 19200 Half Duplex RS-422
Display Head Address 2
IBN: 9600 Half Duplex RS-422 Token Ring
IB Network Custom Protocol
Port 2: 9600 Half Duplex RS-232 TX/RX
Stop: 1, Data: 8, Parity: None
Serial Printer or OBC
Port 1: 9600 Half Duplex RS-232 TX/RX
Stop: 1, Data: 8, Parity: None
OBC: On-Board Computer or Printer

VEEDER-ROOT™

For technical support, sales or
other assistance, please visit:
www.veeder.com