Littelfuse®
STARTCO

FPS/MPS PROFIBUS-DP INTERFACE

June 21, 2011

Revision 0

Copyright © 2011 Startco Engineering Ltd.

TABLE OF CONTENTS

LIST OF TABLES

DISCLAIMER

Specifications are subject to change without notice. Littelfuse Startco is not liable for contingent or consequential damages, or for expenses sustained as a result of incorrect application, incorrect adjustment, or a malfunction.

This product has a variety of applications. Those responsible for its application must take the necessary steps to assure that each installation meets all performance and safety requirements including any applicable laws, regulations, codes, and standards.

Information provided by Littelfuse Startco is for purposes of example only. Littelfuse Startco does not assume responsibility for liability for use based upon the examples shown.

1. GENERAL

The Profibus-DP slave interface provides access to meter data, set points, starter control functions and provides reset control. Data from the FPS/MPS (input to the network) is in the form of a fixed block of 64 bytes defined as an assembly. This assembly consists of the values of the parameters defined by the User-Defined Registers. The format of this assembly is a function of the data selected and can represent 8-bit, 16-bit and 32-bit values. The end user can configure the User Registers to select any meter, status or set-point value up to a total of 64 bytes.

Data to the FPS/MPS (output from the network) is in the form of a fixed block of 8 bytes. This output assembly is used to send control commands and set points.

The Profibus Extended User Parameters are not implemented on the Profibus interface.

See Appendix E and F of the FPS or MPS Motor Protection System Manual for the Communication Register table and data formats.

2. INTERFACE CONNECTOR

A D-SUB connector is used for the Profibus network.

TABLE 2.1 CONNECTOR TERMINATION

PIN	FUNCTION
1	NC
2	NC
3	B-Line, RS-485
4	RTS
5	GND
6	+5
7	NC
8	A-Line, RS-485
9	NC

A network termination is required on both ends of the network as per the RS-484 specification. A 120- or 150-ohm resistor is used. Additional signals are provided on the D-SUB, however, in normal applications, only the A-line, B-line and cable shield are used.

3. NETWORK SETTINGS

To enable Profibus communications, select *Anybus* from the *Setup | Hardware | Network Comms | Network Type* menu.

The slave address is selected using the OPI *Setup | Hardware | Network Comms | Network ID* menu. The address range is 1 to 125 and the default address is 125.

The network comms selections *Baud Rate, Error Check, Ethernet IP and Ethernet Mask* are not used.

NOTE: RS-485 and Profibus communications interfaces are mutually exclusive. Selecting Profibus (Anybus) will disable the RS-485 interface.

4. COMMUNICATION STATUS AND TIMEOUT

The status of the Profibus communication module is viewed using the *Metering | Comm State* menu. The status of the module is indicated as "Profibus: ONLINE" when the module is operating properly, and as "Profibus: OFFLINE" when the communication module is not operating properly. If the FPS/MPS indicates ONLINE but the module LED is RED, verify that the slave address is correct. If the slave address is changed, the module must be re-initialised. The module is initialised on power up or can be initialised using the OPI. To initialise the module using the OPI, first disable the module by selecting *None* in the *Setup | Hardware | Network Type* menu and then select *Anybus* to enable the module.

In applications where the MPS start/stop or FPS open/close functions are controlled by the network, the MPS or FPS can be configured to trip or alarm on loss of communication-writes to the module. This feature is enabled using the *Setup | Hardware | Network Comms | Network Error* menu.

Communication status for writes to the module is displayed in the *Metering | Comm* State menu. Writes to the OUTPUT memory buffer are indicated by "Output: NO" or "Output: YES". If the module is receiving output from the network, then "Output: YES" will be displayed.

4.1 MODULE ERRORS

Module errors are typically the result of interface errors. When a module error occurs, "*Anybus Error!*" is displayed along with an error code. Error code "1" is displayed if the network type is set to *Anybus* with no module installed. *Anybus* should not be selected if there is no Anybus module installed. If a module is installed and errors persist, contact the factory.

5. LED INDICATION

Module LED's can be viewed through access holes on the side of the control unit (CTU). Red indicates that the module is disabled, has an incorrect address, or is not connected. Green indicates that the module is ON-LINE with a valid slave address and data exchange is possible.

6. GSD FILE

A configuration tool uses a gsd file to configure the network. The input and output area sizes required must be setup within the configuration phase. The configuration tool must be set up with module byte sizes such that the INPUT size is 64 bytes (data from the MPS into the network) and the total OUTPUT size is 8 bytes (data from the network to the MPS).

The communication module uses the standard gsd file for the Anybus Profibus module. This file is available from www.anybus.com or www.littelfuse.com.

7. INPUT DATA FROM FPS/MPS

Input data from the FPS/MPS is a fixed block of 32 words (64 bytes) representing the values of the parameters defined by the User-Defined Register block in the FPS/MPS. The User Registers are configured using the OPI menu system or by using the SECOMM PC program. The user-register table contains 32 pointers to the Registers. For register definitions see FPS/MPS manual Appendix E. The data from these registers is used to build the 64-byte assembly. For example, to configure an assembly to read the first four RTD temperatures in RTD Module 1, enter register numbers 902, 903, 904, 905, 906, 907, 908, 909 in the User-Defined data area. In the resulting assembly, the first 8 words (16 bytes) will contain the four float values of the RTD temperatures. The remaining values are a function of the corresponding User-Register pointers. To prevent a read error, unused User-Defined data must be set to a valid Register number.

NOTE: To maintain compatibility with the Profibus standard, the word order for float values is reversed from what is indicated in FPS/MPS Appendix E, however, the User-Register values are still specified in sequence (902, 903...) as per the manual.

---

8. OUTPUT DATA TO FPS/MPS

Output data to the FPS/MPS is defined as a fixed block of 4 words (8 bytes). This assembly is used to send control commands and set points to the FPS/MPS. For control commands, only 4 of the 8 bytes are used. For set points, all 8 bytes are used. Unused bytes are ignored.

For all output formats, the first 16-bit word is defined as the Request Header. A transition from zero to a non-zero value signals the FPS/MPS to process the assembly. In applications where the assembly is not written as one packet, this value, specifically the low-order byte, must be updated last to prevent the FPS/MPS from using incomplete assembly data.

8.1 CONTROL COMMAND AND TIMEOUT FORMAT

Control commands are issued to the FPS/MPS using a 2-word command sequence, as shown in Table 8.3. The first word is the Request Header (word 1). The second word is the Command (word 2). The Request Header must be zero except when the Command action is to be sent. The Request Header must transition from 0 to 3 (0x0003) for the command sequence to be accepted. Once a valid command has been sent, the Request Header should be set to zero.

Byte order is in "big-endian" format where the high byte is followed by the low byte.

To prevent a trip or alarm on loss of communications, the Control Command is used to write new data to the OUTPUT memory buffer of the FPS/MPS at an interval less than the trip time of 3 seconds. At regular intervals

thereafter, write an incrementing value to the FPS/MPS Command(word 2) while keeping the Request Header (word 1) at 0. Keeping the Request Header at 0 prevents the FPS/MPS from interpreting the Command data as a valid control command. Incrementing the Command data ensures that a "changed data" event is posted to indicate valid communications. Reading data from the module is not sufficient to satisfy the timeout timer.

TABLE 8.1 MPS COMMAND TABLE

COMMAND CODE	ACTION
0	STOP
1	START1
2	START2
3	Reset Trips
4	Set Real-Time Clock
5	Clear Data-Logging Records
6	Clear Trip Counters
7	Clear Energy Totals
8	Clear Running Hours
9	Emergency I^2t and Trip Reset
10	Select Local Control
11	De-select Local Control
12	Re-enable Temperature Protection

TABLE 8.2 FPS COMMAND TABLE

COMMAND CODE	ACTION
0	None
1	Open
2	Close
3	Reset Trips
4	Set RTC
5	Clear Data-Logging Records
6	Clear Trip Counters
7	Clear Energy Totals
8	Clear Running Hours
9	Emergency I^2t Reset
10	Select Local Control
11	De-select Local Control
12	Re-enable Temperature Protection
13	Remote Trip Set
14	Remote Trip Clear
15	Remote Alarm Set
16	Remote Alarm Clear
17	Net Relay Set
18	Net Relay Clear

TABLE 8.3 MPS COMMAND ASSEMBLY

BYTE NUMBER	DESCRIPTION
0	Request Header (High)
1	Request Header (Low)
2	MPS Command (High)
3	MPS Command (Low)
4-7	Not used

8.2 SET-POINT COMMAND FORMAT

In process-control applications where a set point must be changed dynamically, the PLC uses the command buffer to send set points to the FPS/MPS.

The first word (byte 0,1) is the Request Header. This value must transition from 0 to 19 (0x0013 hex) to send a 1-word set point and from 0 to 35 (0x0023 hex) to send a 2-word set point. The second word (byte 2,3) is the FPS/MPS Register number of the set point to be changed. This must be a valid set point with write access (product manual Appendix E WRITE access column should indicate R/W). The remaining bytes contain the set-point data the format of which depends on the set-point data. For word set points (short, 16-bit), byte 4 is the high byte, byte 5 is the low byte, and byte 6 and 7 are not used. For float set points (IEEE 32-bit), byte 4 is the high byte and byte 7 is the low byte of the 4-byte float value. Character values are set using one or two word values consisting of a pair of characters where the high byte is the first character and the low byte is the second character.

TABLE 8.4 MPS SET-POINT ASSEMBLY

BYTE NUMBER	DESCRIPTION
0	Request Header (High)
1	Request Header (Low)
2	Destination Register (High)
3	Destination Register (Low)
4-7	Set-Point Data. 2 bytes for words, 4 bytes for floats