



# XS SERIES™ PMBUS™ OPTION CARD



# Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
1.1	Revision History . . . . .	4
<b>2</b>	<b>Option Card Connectors</b>	<b>4</b>
2.1	PMBus Slave Address . . . . .	4
2.2	Programming Connector - J1 . . . . .	6
2.3	PMBus Connector - PL1 . . . . .	6
2.3.1	PMBus Alert . . . . .	7
2.3.2	SDA - Serial Data Line . . . . .	7
2.3.3	SCL - Serial Clock Line . . . . .	7
2.3.4	PMBus Control . . . . .	7
2.3.5	GND - Signal GND . . . . .	7
2.4	Parallel Operation . . . . .	8
<b>3</b>	<b>PMBus Communication</b>	<b>9</b>
3.1	PMBus Linear Format . . . . .	9
3.1.1	Linear Format Decoding . . . . .	9
<b>4</b>	<b>Supported Commands</b>	<b>11</b>
4.1	Monitoring Commands . . . . .	11
4.1.1	VOUT_MODE (0x20) - [READ BYTE] . . . . .	11
4.1.2	READ_VOUT (0x8B) - [READ WORD] . . . . .	11
4.1.3	READ_IOUT (0x8C) - [READ WORD] . . . . .	12
4.1.4	READ_TEMPERATURE_1 (0x8D) - [READ WORD] . . . . .	12
4.2	Calibration Commands . . . . .	12
4.2.1	VOUT_SCALE_MONITOR (0x2A) - [READ/WRITE WORD] . . . . .	12
4.2.2	IOUT_CAL_GAIN (0x38) - [READ/WRITE WORD] . . . . .	12
4.2.3	IOUT_CAL_OFFSET (0x39) - [READ/WRITE WORD] . . . . .	13
4.2.4	STATUS_MANUFACTURER_SPECIFIC (0x80) - [READ BYTE] . . . . .	13
4.2.5	STORE_USER_ALL (0x15) - [SEND BYTE] . . . . .	13
4.3	Identification Commands . . . . .	14
4.3.1	MFR_REVISION (0x9B) - [READ BLOCK (8 Bytes)] . . . . .	14
4.3.2	PMBUS_REVISION (0x9D) - [READ BYTE] . . . . .	14
4.3.3	MFR_ID (0x99) - [READ BLOCK (8 Bytes)] . . . . .	14
4.3.4	MFR_MODEL (0x9A) - [READ BLOCK (5 Bytes)] . . . . .	14
<b>5</b>	<b>Calibration Procedure</b>	<b>15</b>
5.0.5	Current Measurement Calibration . . . . .	15
5.0.6	Current Calibration Example . . . . .	16
5.0.7	Voltage Measurement Calibration . . . . .	17
5.0.8	Voltage Calibration Example . . . . .	18
<b>6</b>	<b>Measurement Specifications</b>	<b>19</b>
6.1	Output Voltage Measurement Specifications . . . . .	19

6.1.1	Output Voltage Measurement Resolution . . . . .	19
6.1.2	Output Voltage Measurement Accuracy . . . . .	19
6.2	Output Current Measurement Specifications . . . . .	19
6.2.1	Output Current Measurement Resolution . . . . .	19
6.2.2	Output Current Measurement Accuracy . . . . .	20
6.3	Temperature Measurement Specifications . . . . .	20
6.3.1	Temperature Measurement Resolution . . . . .	20
6.3.2	Temperature Measurement Accuracy . . . . .	20
<b>7</b>	<b>Troubleshooting . . . . .</b>	<b>21</b>
7.1	Packet Error Checking . . . . .	21
7.2	Communication Issues . . . . .	21

## List of Figures

1	PMBus Option Card Connections . . . . .	5
2	PMBus Connector Pinout . . . . .	6
3	Option Card Grounding Arrangement . . . . .	8
4	Linear Format Data Structure . . . . .	9
5	Two-point calibration procedure . . . . .	16
6	Temperature Sensor Location . . . . .	20

## List of Tables

1	Slave Address Structure . . . . .	4
2	Available Address Space . . . . .	5
3	Command Table . . . . .	11
4	VOUT_MODE Command Data Structure . . . . .	11
5	PMBUS_REVISION Command Data Structure . . . . .	14
6	MFR_ID Command Data Structure . . . . .	14
7	MFR_MODEL Command Data Structure . . . . .	14

## List of Equations

1	Linear mode conversion equation . . . . .	10
2	Current Gain Calculation Equation . . . . .	15
3	Current Offset Calculation Equation (1) . . . . .	15
4	Current Offset Calculation Equation (2) . . . . .	15
5	ADC Voltage Calculation Equation . . . . .	17
6	Voltage Scaling Factor Calculation Equation . . . . .	17
7	Voltage Measurement Resolution Equation . . . . .	19
8	Current Measurement Resolution Equation . . . . .	19
9	Sensor Temperature Calculation Equation . . . . .	20

# 1 Introduction

The Power Management Bus (PMBus™) is an open standard which defines a means of communication for power conversion devices. It defines a full set of commands and data structures required by power control and management components. The XS Series™ PMBus interface card extends the capabilities of any XS Series power converter to allow communication of operating parameters such as output voltage, output current and internal temperature with other PMBus™ enabled devices. For more information about PMBus™, please see the System Management Interface Forum website [www.PowerSIG.org](http://www.PowerSIG.org)

## 1.1 Revision History

Revision	Date	ECN #	Details
00	20.06.14	N/A	First Release
01	19.08.14	E0560	Update for new PCB layout and functions.
02	19.08.14	E0581	Added calibration functions.
03	19.08.15	E0608	Added 36V model.
04	18.09.15	E0614	Added accuracy specs. Added grounding diagram. Added temp. sensor diagram.
05	11.05.17	E0681	Change to calibration procedure.

# 2 Option Card Connectors

## 2.1 PMBus Slave Address

The PMBus standard utilizes 7 bits for addressing. These 7 bits form what is referred to as a slave address. This is the hardware address of the PMBus device and is independent of the direction of data transfer. During any read or write operation, a direction bit is appended by the master to the 7 bit slave address to indicate the direction of data transfer: 1 = READ operation; 0 = WRITE operation. Sometimes PMBus/I<sup>2</sup>C vendors will include the direction bit in the address leading to a separate READ ADDRESS and WRITE ADDRESS. In this convention, the WRITE ADDRESS can be calculated as (SLAVE ADDRESS \* 2) and the READ ADDRESS can be calculated as ((SLAVE ADDRESS \* 2)+1).

The XS Series PMBus option card allows the user to modify the lower 3 bits of the slave address, leading to a fixed part of the address and a variable part of the address. The fixed part of the slave address consists of the 4 most significant bits A6, A5, A4 and A3 and *always* equals 0010. The variable part of the address consists of the 3 least significant bits A2, A1 and A0 and these bits can be modified by the placement of jumpers on the corresponding 3 pin headers.

A6	A5	A4	A3	A2	A1	A0
0	0	1	0	A2*	A1*	A0*

Table 1: Slave Address Structure

\* Determined by the position of the jumper link. Jumper in = logic 0; Jumper out = logic 1.

The address lines for A2, A1 and A0 are internally pulled up by resistors and therefore default to logic 1 (the default address  $\Rightarrow = 0010111 = 0x17$ ). The placement of a jumper on a header pulls the corresponding address line to a logic 0.

If multiple option cards are to be used in parallel (see section 2.4 on page number 8) on the same bus then each option card will need to be assigned a unique address through the fitting of one or more jumpers according to Table 2 below. In total, 8 unique addresses are available which limits the maximum amount of devices on a single bus to 8.

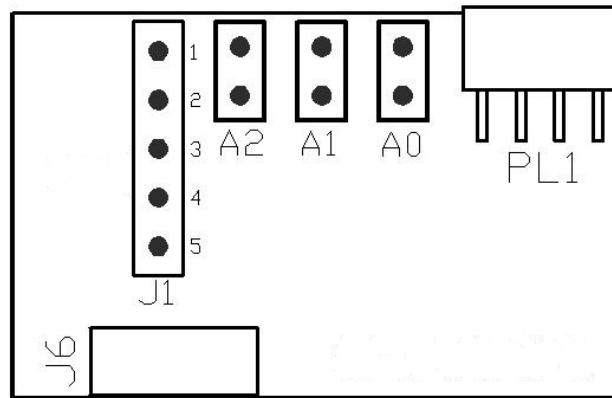


Figure 1: PMBus Option Card Connections

A6	A5	A4	A3	A2	A1	A0	SLAVE Address	WRITE Address	READ Address
0	0	1	0	1 / OUT	1 / OUT	1 / OUT	0x17	0x2E	0x2F
0	0	1	0	1 / OUT	1 / OUT	0 / IN	0x16	0x2C	0x2D
0	0	1	0	1 / OUT	0 / IN	1 / OUT	0x15	0x2A	0x2B
0	0	1	0	1 / OUT	0 / IN	0 / IN	0x14	0x28	0x29
0	0	1	0	0 / IN	1 / OUT	1 / OUT	0x13	0x26	0x27
0	0	1	0	0 / IN	1 / OUT	0 / IN	0x12	0x24	0x25
0	0	1	0	0 / IN	0 / IN	1 / OUT	0x11	0x22	0x23
0	0	1	0	0 / IN	0 / IN	0 / IN	0x10	0x20	0x21

Table 2: Available Address Space

## 2.2 Programming Connector - J1

This connector is solely for the purpose of downloading of firmware to the option card. This will already have been completed during factory assembly so this connector has no further use for the end user. In the event that firmware modifications are required, details of the firmware downloading process shall be provided by Excelsys Technologies Ltd.

## 2.3 PMBus Connector - PL1

This 8 pin connector is used to connect the PMBus option card to the bus. The pinout of this connector is as follows:



Figure 2: PMBus Connector Pinout

1. NC.
2. [GND](#).
3. NC.
4. NC.
5. [SCL](#).
6. [PMBus Control](#).
7. [PMBus Alert](#).
8. [SDA](#).

### 2.3.1 PMBus Alert

This is an optional output signal which allows a PMBus slave device to interrupt the host in response to status changes etc. As this behaviour is not supported by the PMBus option card, it is unnecessary to make a connection to this pin.

### 2.3.2 SDA - Serial Data Line

This is the data line over which all serial communication takes place. It is essential that this pin is connected to the PMBus SDA line.

### 2.3.3 SCL - Serial Clock Line

This is the clock line which synchronizes all serial communication over the PMBus. It is essential that this pin is connected to the PMBus SCL line. The XS Series PMBus option card is designed to operate with a PMBus clock frequency of 100KHz.

### 2.3.4 PMBus Control

This is an optional input signal on a PMBus device which allows to unit to be enabled/disabled in response to commands received over the serial bus. As the PMBus option card does *not* support any such commands, it is unnecessary to make a connection to this signal.

### 2.3.5 GND - Signal GND

This should be connected to GND or Signal Return of the PMBus Host device. Note that this is at the same potential as the Vout- terminal as illustrated by the diagram below.

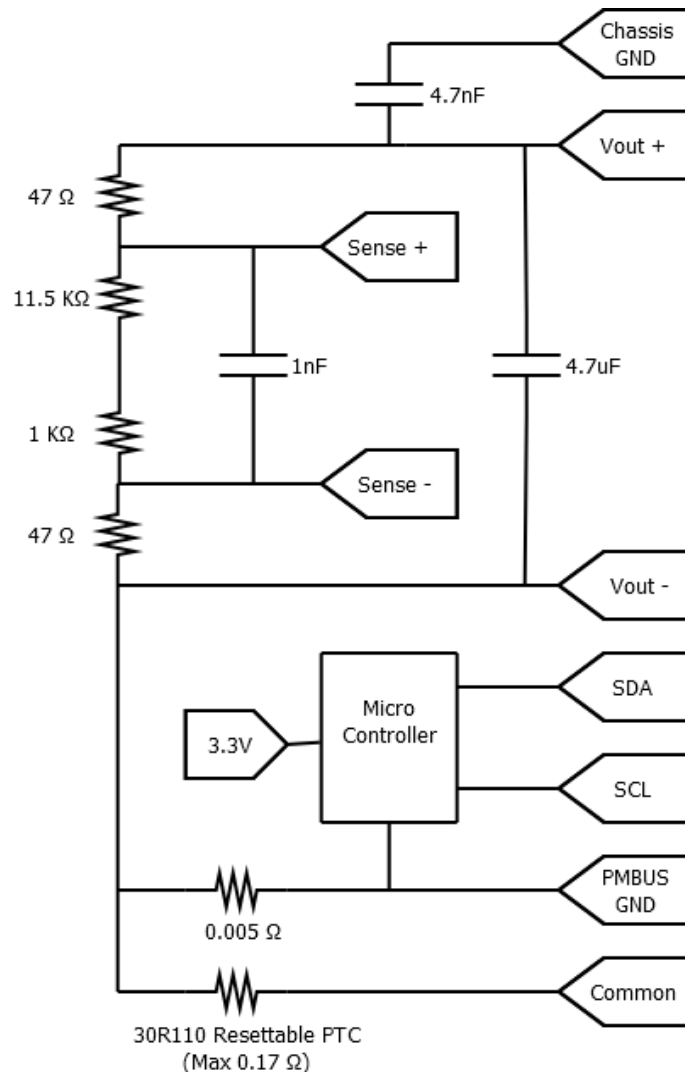


Figure 3: Option Card Grounding Arrangement

## 2.4 Parallel Operation

It is possible to connect up to 8 PMBus option cards on the same bus for simultaneous communication with multiple XS Series power converters from a single host. To do this, simply connect all required PMBus signals (i.e. SDA, SCL & GND) in parallel. Each PMBus option card must then be assigned a unique address following the procedure shown in section 2.1 on page number 4.



### 3 PMBus Communication

#### 3.1 PMBus Linear Format

The XS Series PMBus interface utilizes the linear data format defined in the PMBus Specification to represent voltage, current and temperature readings. This format presents real world units (Amps, Volts, Degrees) to the host system in a manner which is less computationally difficult for the host system than the alternative direct system.

The data returned consists of the following:

- An 11 bit, two's complement mantissa.
- A 5 bit, two's complement exponent (scaling factor).

These combine to form a two byte word as follows:

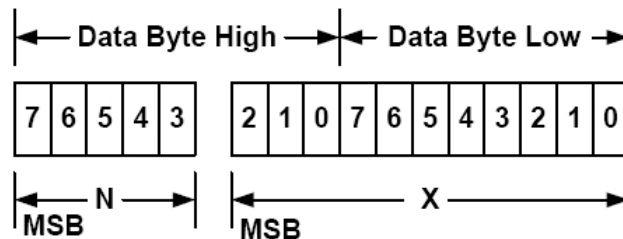


Figure 4: Linear Format Data Structure

##### 3.1.1 Linear Format Decoding

To understand the decoding of the linear format data to obtain the real-world measurement, we will work through an example of an output voltage measurement:

**Sample Data:**

- Returned Byte 1 = 0xDB
- Returned Byte 2 = 0x12

The first step is to extract the exponent data:

- 0xDB12 in binary format= **1101101100010010**.
- Exponent bits = 11011
- 11011 converted from two's complement = -5.
- The exponent in this example is therefore -5.

The second step is to extract the mantissa data:

- 0xDB12 converted to binary = 11011**101100010010**
- Mantissa bits = 01100010010
- 01100010010 converted from two's complement = 786.
- The Mantissa in this example is therefore 786.

The final step is simply to calculate the real world value using the two figures obtained above and the equation:

$$Y = X * (2^N) \quad (1)$$

Where:

Y = The real-world value to be calculated e.g. output voltage (in Volts) in this example.

X = The Mantissa obtained above e.g. 768 in this example.

N = The Exponent obtained above e.g. -5 in this example.

$$V_{out} = 768 * (2^{-5}) = 24.5625 \text{ Volts}$$

The exact same process is used to calculate output current and temperature readings. <sup>1</sup>

---

<sup>1</sup>A Microsoft Excel <sup>TM</sup> worksheet is available for download from the Excelsys website [www.excelsys.com](http://www.excelsys.com) to assist with conversion to/from linear-mode data.

## 4 Supported Commands

The full list of commands currently supported by the XS Series PMBus interface is as follows:

Monitoring Commands	Calibration Commands	Identification Commands
VOUT_MODE (0x20)	VOUT_SCALE_MONITOR (0x2A)	PMBUS_REVISION (0x98)
READ_VOUT (0x8B)	IOUT_CAL_OFFSET (0x39)	MFR_ID (0x99)
READ_IOUT (0x8C)	IOUT_CAL_GAIN (0x38)	MFR_MODEL (0x9A)
READ_TEMPERATURE_1 (0x8D)	STORE_USER_ALL (0x15)	MFR_REVISION (0x9B)
	STATUS_MFR_SPECIFIC (0x80)	

Table 3: Command Table

Commands can return either: a BYTE, a WORD or a BLOCK (multiple bytes including a byte count) as indicated below.

### 4.1 Monitoring Commands

#### 4.1.1 VOUT\_MODE (0x20) - [READ BYTE]

The VOUT\_MODE command is used to obtain information about the format of voltage measurements which are returned by a PMBus™ device if this information is not already known. The data returned from this command consists of a single byte. The first three bits determine if the device is using the Linear, VID or direct modes for data representation. The remaining five bits give additional information, depending on the data format.

Mode	Bits [7:5]	Bits [4:0]
Linear	000	5 bit mantissa which is returned as part of the output voltage data.
VID	001	VID Code Identifier
Direct	010	Always set to 00000.

Table 4: VOUT\_MODE Command Data Structure

For this device, the data format is fixed as linear mode with an exponent of -5 so the data returned by this command shall always be 00011011 = 0x1B. Attempts to write to this parameter shall be negatively acknowledged (NACKed).

#### 4.1.2 READ\_VOUT (0x8B) - [READ WORD]

This command returns an output voltage measurement (in Volts) formatted in linear format as described in section 3.1 on page 9.

#### 4.1.3 READ\_IOUT (0x8C) - [READ WORD]

This command returns an output current measurement (in Amps) formatted in linear format as described in section 3.1 on page 9.

#### 4.1.4 READ\_TEMPERATURE\_1 (0x8D) - [READ WORD]

This command returns a temperature measurement (in degrees celsius) from the on-board temperature sensor of the interface module. It is formatted in linear format as described in section 3.1 on page 9.

### 4.2 Calibration Commands

#### 4.2.1 VOUT\_SCALE\_MONITOR (0x2A) - [READ/WRITE WORD]

As the XS Series output voltage range is too high to be measured directly by the microprocessor (MPU), the output is measured through a resistive potential divider which reduces the voltage to a lower level which is capable of being read by the MPU. To calculate the real-world output voltage from this reduced voltage therefore requires that the MPU is aware of the exact (including component tolerances) scale factor to be applied. This scale factor can be viewed/edited using the VOUT\_SCALE\_MONITOR command. If issued as a read command, the option card shall return the two byte linear formatted voltage scaling factor. This can be used to calibrate the output voltage measurement if required. If issued as a write command, with one or two bytes data representing the required scaling factor, the scaling factor shall be updated immediately in *operating memory*. If accepted, the option card shall respond with an ACK.

Note: After a power cycle, this shall revert to the previous value. If it is required to make the change permanent, the [STORE\\_USER\\_ALL](#) command should be issued to update the value in non-volatile storage. Please see the [Calibration](#) section for correct usage of this command.

Recommended starting points for this factor are as follows:

1. XS500-24N/XS1000-24N = 0xD968.
2. XS500-36N/XS1000-36N = 0xDA0A.
3. XS500-48N/XS1000-48N = 0xDA00.

#### 4.2.2 IOUT\_CAL\_GAIN (0x38) - [READ/WRITE WORD]

In order to maximize the dynamic range of the ADC while minimizing the size of the current sense resistor, the current sense signal is passed through an amplifier before it reaches the ADC. Therefore, to accurately calculate the real-world output current from this amplified signal, the MPU must be aware of the exact relationship between the output current in Amps and the corresponding digitized value returned by the ADC. This relationship takes the form of a gain factor plus a small offset factor. The gain factor part can be edited/viewed using the IOUT\_CAL\_GAIN command. If issued as a read command, the option card shall return the two byte linear formatted current scaling factor. This can be used to calibrate the output current measurement if required. If issued as a write command, with one or two bytes data representing the required scaling factor, the scaling factor shall be updated immediately in *operating memory*. If accepted, the option card shall respond with an ACK.

Note: After a power cycle, this shall revert to the previous value. If it is required to make the change permanent, the

**STORE\_USER\_ALL** command should be issued to update the value in non-volatile storage. Please see the [Calibration](#) section for correct usage of this command.

Recommended starting points for this factor are as follows:

1. XS500-24N/XS1000-24N = 0xB0C2.
2. XS500-36N/XS1000-36N = 0xB086.
3. XS500-48N/XS1000-48N = 0xB10E.

#### 4.2.3 IOUT\_CAL\_OFFSET (0x39) - [READ/WRITE WORD]

In order to maximize the dynamic range of the ADC while minimizing the size of the current sense resistor, the current sense signal is passed through an amplifier before it reaches the ADC. Therefore, to accurately calculate the real-world output current from this amplified signal, the MPU must be aware of the exact relationship between the output current in Amps and the corresponding digitized value returned by the ADC. This relationship takes the form of a gain factor plus a small offset factor. The offset factor part can be edited/viewed using the **IOUT\_CAL\_OFFSET** command. If issued as a read command, the option card shall return the two byte linear formatted current offset factor. This can be used to calibrate the output current measurement if required. If issued as a write command, with one or two bytes data representing the required offset factor, the offset factor shall be updated immediately in *operating memory*. If accepted, the option card shall respond with an ACK.

Note: After a power cycle, this shall revert to the previous value. If it is required to make the change permanent, the **STORE\_USER\_ALL** command should be issued to update the value in non-volatile storage. Please see the [Calibration](#) section for correct usage of this command.

Recommended starting points for this factor are as follows:

1. XS500-24N/XS1000-24N = 0xB10E.
2. XS500-36N/XS1000-36N = 0xB069.
3. XS500-48N/XS1000-48N = 0xB078.

#### 4.2.4 STATUS\_MANUFACTURER\_SPECIFIC (0x80) - [READ BYTE]

This is a manufacturer specific command which is used solely as a tool to aid in the calibration of the output current measurement where necessary. This command returns a single byte value which represents the digital measurement of the output current before any scaling or offsets are applied. Please see the [Calibration](#) section for more detailed instructions on how to use this command.

#### 4.2.5 STORE\_USER\_ALL (0x15) - [SEND BYTE]

This command instructs the option card to copy the entire operating memory to non-volatile storage (flash). Upon start up, the option card loads the operating memory with the contents of the flash memory. This means that if any calibration parameters have been changed in the operating memory since the device has been powered up, these changes shall become permanent and will be loaded by the option card at every start up. This command should therefore be used with caution and should only be used as part of a controlled calibration procedure (see [Calibration Procedure](#) section).

### 4.3 Identification Commands

#### 4.3.1 MFR\_REVISION (0x9B) - [READ BLOCK (8 Bytes)]

This command returns the revision of firmware which is running on the PMBus Option Card as a string of ASCII encoded characters e.g. "80002r02".

#### 4.3.2 PMBUS\_REVISION (0x9D) - [READ BYTE]

This command returns the revision of PMBus to which the device is compliant. Bits [7:5] indicate the revision of PMBus Specification Part I to which the device is compliant. Bits [3:0] indicate the revision of PMBus Specification Part II to which the device is compliant (Bit 4 = don't care). The permissible values are shown in Table 5.

Bits [7:5]	Part I Revision	Bit [4]	Bits [3:0]	Part II Revision
000	1.0	X	0000	1.0.
001	1.1	X	0001	1.2.
010	1.2	X	0010	1.2.

Table 5: PMBUS\_REVISION Command Data Structure

#### 4.3.3 MFR\_ID (0x99) - [READ BLOCK (8 Bytes)]

This command simply returns a string of ASCII encoded characters which identify the manufacturer of the PMBus device.

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
<b>HEX DATA</b>	0x45	0x58	0x43	0x45	0x4c	0x53	0x59	0x53
<b>ASCII DATA</b>	E	X	C	E	L	S	Y	S

Table 6: MFR\_ID Command Data Structure

#### 4.3.4 MFR\_MODEL (0x9A) - [READ BLOCK (5 Bytes)]

This command simply returns a string of ASCII characters which identifies the model of the PMBus device.

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4
<b>HEX DATA</b>	0x58	0x53	0x4F	0x4C	0x4F
<b>ASCII DATA</b>	X	S	O	L	O

Table 7: MFR\_MODEL Command Data Structure

## 5 Calibration Procedure

### 5.0.5 Current Measurement Calibration

In the event that the values returned using the [READ\\_IOUT](#) command are found to be inaccurate, the following procedure can be used to calibrate the measurement for increased accuracy. The ideal procedure to be followed is a two-point calibration which involves taking a measurement at a low level of output current and a measurement at a high level of output current. The required gain and offset to be applied can then be calculated from these two data points.<sup>2</sup> The details of the procedure are as follows:

1. Apply a low load to the output through an ammeter or current-shunt so that the current can be measured externally. The recommended load level for optimum accuracy is between 5% and 10% of nominal. Record the output current reading. Let's call this value *ANALOG<sub>LOW</sub>*.
2. Read one byte using the [STATUS\\_MANUFACTURER\\_SPECIFIC](#) command [0x80]. The byte returned represents the digital value of the output current measurement. This is the value used by the MCU to calculate the actual output current in amps by applying the gain and offset associated with the amplifier circuit. Record this value as *DIGITAL<sub>LOW</sub>*.
3. Apply a high load of 90% or more of nominal to the output through an ammeter or current-shunt so that the current can be measured externally. Record the output current reading. Let's call this value *ANALOG<sub>HIGH</sub>*.
4. Read one byte using the [STATUS\\_MANUFACTURER\\_SPECIFIC](#) command [0x80]. Record this value as *DIGITAL<sub>HIGH</sub>*.
5. The gain coefficient can now be calculated from the equation:

$$GAIN = \frac{ANALOG_{HIGH} - ANALOG_{LOW}}{DIGITAL_{HIGH} - DIGITAL_{LOW}} \quad (2)$$

6. This value must then be converted to [PMBUS Linear Format](#) and programmed into the device using the command [IOUT\\_CAL\\_GAIN](#).
7. The offset coefficient can be calculated from the equation:

$$OFFSET = ANALOG_{HIGH} - (GAIN * DIGITAL_{HIGH}) \quad (3)$$

OR

$$OFFSET = ANALOG_{LOW} - (GAIN * DIGITAL_{LOW}) \quad (4)$$

8. This value must then be converted to [PMBUS Linear Format](#) and programmed into the device using the command [IOUT\\_CAL\\_OFFSET](#) [0x39].
9. The changes are made permanent by issuing the command [STORE\\_USER\\_ALL](#) [0x15].

<sup>2</sup>A Microsoft Excel™ worksheet is available for download from the Excelsys website [www.excelsys.com](http://www.excelsys.com) to automatically calculate the calibration coefficients from two given data points. It also provides functions for converting to and from linear-mode format.

### 5.0.6 Current Calibration Example

Sample output current calibration for XS1000-24N:

1. Applied load = 1A. Iout measured with ammeter = 1.022A  $\Rightarrow ANALOG_{LOW} = 1.022$ .
2. STATUS\_MANUFACTURER\_SPECIFIC returned 0x04  $\Rightarrow DIGITAL_{LOW} = 4$ .
3. Applied load = 40A. Iout measured with ammeter = 40.062A  $\Rightarrow ANALOG_{HIGH} = 40.062$ .
4. STATUS\_MANUFACTURER\_SPECIFIC returned 0xD2 (210 decimal)  $\Rightarrow DIGITAL_{HIGH} = 210$ .
5. The Gain coefficient is calculated:

$$GAIN = \frac{40.062 - 1.022}{210 - 4} = 0.1895$$

6. Converted to linear (-10 Exponent) = 0xB0C2  $\Rightarrow IOUT\_CAL\_GAIN = 0xB0C2$ .

7. The Offset coefficient is calculated:

$$OFFSET = 40.062 - (0.1895 * 210) = 0.267$$

8. Converted to linear (-10 Exponent) = 0xB111  $\Rightarrow IOUT\_CAL\_OFFSET = 0xB111$ .

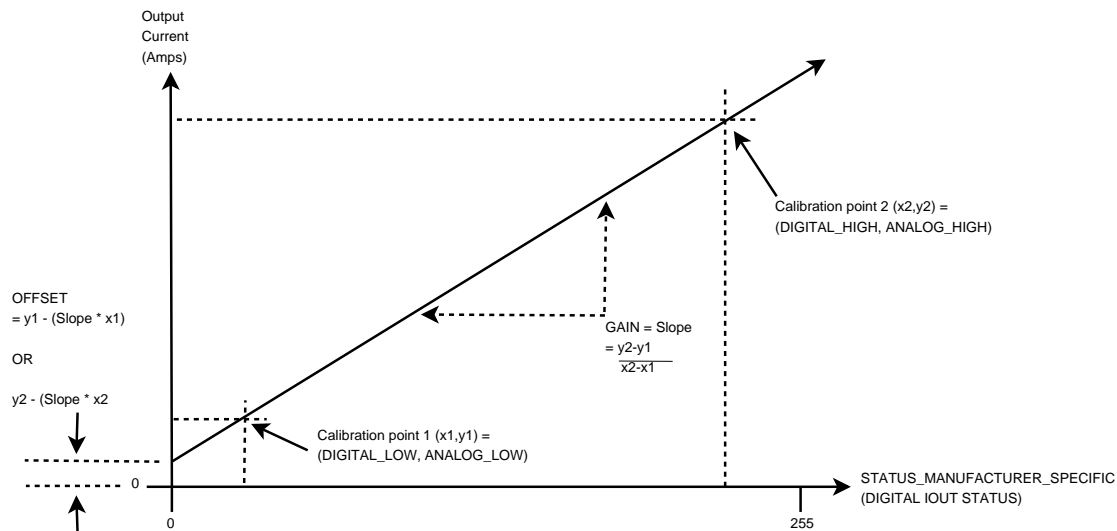


Figure 5: Two-point calibration procedure



### 5.0.7 Voltage Measurement Calibration

In the event that the values returned using the `READ_VOUT` command are found to be inaccurate, the following procedure can be used to calibrate the measurement for increased accuracy. The procedure to be followed is a simple one point calibration procedure which involves taking one external output voltage measurement and reading a couple of values from the unit in order to calculate the exact scaling factor to be applied to bring the output voltage measurement within range.<sup>3</sup> The details of the procedure are as follows:

1. Power up the unit and measure the output voltage at the output terminals using a Voltmeter. Record this value as  $V_{OUT-ACTUAL}$ .
2. Read one word from the device using the command `READ_VOUT` [0x8B]. Convert this linear-mode word to real-world voltage and record this value as  $V_{OUT-PMBUS}$ . This is the actual output voltage being reported over the PMBus™. We will compare this to the actual output voltage and use it to calculate the required scaling factor.
3. Read one word from the device using the command `VOUT_SCALE_MONITOR` [0x2A]. Convert this linear-mode word to real-world gain and record this value as  $V_{OUT\_SCALE\_MONITOR_{OLD}}$ . This tells us the current scaling factor which is being applied.
4. The new scaling factor can now be calculated :

First, calculate the actual voltage present on the ADC for the current output voltage:

$$V_{SENSE} = \frac{V_{OUT-PMBUS}}{V_{OUT\_SCALE\_MONITOR_{OLD}}} \quad (5)$$

Now calculate the scaling factor which is required by dividing the actual output voltage ( $V_{OUT-ACTUAL}$ ) by the voltage seen by the ADC ( $V_{SENSE}$ ):

$$SCALE - FACTOR = \frac{V_{OUT-ACTUAL}}{V_{SENSE}} \quad (6)$$

5. This scale factor must then converted to a linear-mode word and written to the unit using the command `VOUT_SCALE_MONITOR` [0x2A].
6. The change is made permanent by issuing the command `STORE_USER_ALL` [0x15].

---

<sup>3</sup>A Microsoft Excel™ worksheet is available for download from the Excelsys Technologies Ltd. website [www.excelsys.com](http://www.excelsys.com) to automatically calculate the scaling factor from the given data. It also provides functions for converting to/from linear-mode format.

### 5.0.8 Voltage Calibration Example

Sample output voltage calibration for XS1000-24N:

1.  $V_{OUT-ACTUAL}$  measured with voltmeter = 24.050 V
2.  $V_{OUT-PMBUS}$  returned = 0xDA68 = 19.25 V
3.  $V_{OUT\_SCALE\_MONITOR_{OLD}}$  returned = 0xD920 = 9.0
4. The required scale factor is calculated:

$$V_{SENSE} = \frac{19.25V}{9.0} = 2.139V$$

$$SCALE - FACTOR = \frac{24.05V}{2.139V} = 11.244$$

5. 11.244 converted to linear-mode = 0xE0B4  $\Rightarrow$   $V_{OUT\_SCALE\_MONITOR} = 0xE0B4$

## 6 Measurement Specifications

### 6.1 Output Voltage Measurement Specifications

#### 6.1.1 Output Voltage Measurement Resolution

The resolution of the output voltage measurement is the smallest change in output voltage which can be detected by the Analogue to Digital Converter (ADC). The output voltage is measured by an 8-bit ADC. The output voltage measurement resolution can therefore be calculated as follows:

$$Resolution(Volts) = \frac{V_{max}}{2^8} \quad (7)$$

$V_{max}$  is the highest specified output voltage for the XS Series model in which the option card is installed. This results in the following resolutions for the three different models.

1. XS500-24N/XS1000-24N:  $V_{max} = 28V \Rightarrow Resolution \approx 110 \text{ mV}$ .
2. XS500-36N/XS1000-36N:  $V_{max} = 40V \Rightarrow Resolution \approx 160 \text{ mV}$ .
3. XS500-48N/XS1000-48N:  $V_{max} = 56V \Rightarrow Resolution \approx 230 \text{ mV}$ .

#### 6.1.2 Output Voltage Measurement Accuracy

The output voltage measurement accuracy is specified as  $\pm 1\%$ .<sup>4</sup>

### 6.2 Output Current Measurement Specifications

#### 6.2.1 Output Current Measurement Resolution

The resolution of the output current measurement is the smallest change in output current which can be detected by the Analogue to Digital Converter (ADC). The output current is measured by an 8-bit ADC. The output current measurement resolution can therefore be calculated as follows:

$$Resolution(Amps) = \frac{I_{max}}{2^8} \quad (8)$$

$I_{max}$  is the highest specified output current which can be read for the XS Series model in which the option card is installed. This results in the following resolutions for the three different models.

1. XS500-24N/XS1000-24N:  $I_{max} = 48A \Rightarrow Resolution \approx 187 \text{ mA}$ .
2. XS500-36N/XS1000-36N:  $V_{max} = 32A \Rightarrow Resolution \approx 125 \text{ mA}$ .
3. XS500-48N/XS1000-48N:  $V_{max} = 24A \Rightarrow Resolution \approx 95 \text{ mA}$ .

---

<sup>4</sup>The specified accuracy is achieved following the calibration procedure.

### 6.2.2 Output Current Measurement Accuracy

The output current measurement accuracy is specified as  $\pm 2.5\%$ .<sup>5</sup>

## 6.3 Temperature Measurement Specifications

### 6.3.1 Temperature Measurement Resolution

The resolution of the temperature measurement is the smallest change in temperature which can be detected by the temperature sensor. The temperature sensor has a fixed resolution of 1° Celsius.

### 6.3.2 Temperature Measurement Accuracy

The temperature measurements returned from the option card are calibrated to correlate to chassis temperature. These readings can be offset to reflect the actual temperature seen at the temperature sensor by application of the following equation:

$$\text{Sensor\_Temperature} = (\text{Returned\_Temperature} * 1.0361) + 8.948 \quad (9)$$

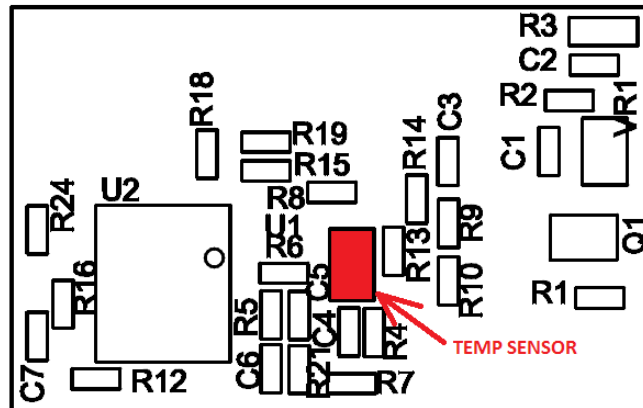


Figure 6: Temperature Sensor Location

The temperature measurement accuracy is specified as  $\pm 2^\circ\text{C}$ .<sup>6</sup>

<sup>5</sup>The current measurement accuracy is not specified below 1.0A where the resolution is comparable with the quantity being measured.

<sup>6</sup>The specified accuracy is achieved following application of the conversion equation (9)

## 7 Troubleshooting

### 7.1 Packet Error Checking

Packet Error Checking (PEC) is an optional feature of SMBus/PMBus in which an error code byte is appended to each transaction. The PMBus option card is designed to operate without PEC so this feature should be disabled if present on the PMBus host device.

### 7.2 Communication Issues

No data is returned from the PMBus option card in response to issued commands (i.e. no ACK or NACK):

1. Check that the address which the PMBus host is trying to communicate with has been correctly programmed on the PMBus option card according to the procedure in section 2.1 on page 4. The simplest configuration is obtained by ensuring all 3 links are removed which will guarantee an address of 0x17 (hexadecimal) or 23 (decimal).
2. Verify that the signals SDA, SCL and GND have been correctly connected to the PMBus host.
3. Verify that the PMBus host device is *not* configured to use Packet Error Checking (PEC) as this is unsupported by the PMBus option card. Note: PMBus compliant master devices are required to be able to auto-detect the PEC status and act accordingly however some SMBus/PMBus interface GUIs have the option to fix the PEC mode which would cause communication failure.
4. Verify that the PMBus host device is configured to use a bus speed of 100KHz.
5. Verify that the PMBus host device is configured to pull up the signals SDA and SCL.

The returned output voltage and/or current measurements are very inaccurate:

1. See the [Calibration Procedure](#) section for instructions on how to calibrate the output measurements.

For other support options, please visit [www.excelsys.com](http://www.excelsys.com).