



**SATEC POWERMETERS**

**SATEC COMMUNICATION DRIVER FOR  
ALLEN-BRADLEY SLC 500<sup>®</sup> 1746-BAS BASIC  
MODULE**

**Installation and Operation Manual**

**SATEC Ltd.**

The driver is provided "as is" without any warranty of any kind. The entire risk to the results is assumed by the user. Those responsible for applying and using this product must satisfy themselves as to the acceptability of each application and the use of the product.

No patent liability is assumed by the manufacturer with respect to the use of information, circuits, equipment or software described in this text.

Every effort has been made to ensure complete and accurate information concerning the material presented in this book. However, the manufacturer can neither guarantee nor be held legally responsible for any mistakes in printing or faulty instructions contained in this book. The authors always appreciate receiving notice of any errors or misprints.

SLC 500™ is a trademark of Allen-Bradley Company, Inc.

COPYRIGHT © 1995 SATEC Ltd.

BG0092 Rev.A

# Table of Contents

<b>1 ABOUT THE COMMUNICATION DRIVER .....</b>	<b>4</b>
<b>2 INSTALLATION .....</b>	<b>5</b>
2.1 EPROM Installation.....	5
2.2 BASIC Module Port PRT2 Configuration .....	5
2.3 Configuring M0-M1 Files and I/O Words Using APS Software.....	5
2.4 Powermeters Communications Settings .....	6
2.5 Cable Connections .....	6
<b>3 GENERAL OPERATION.....</b>	<b>7</b>
3.1 Interfacing to the BASIC Module.....	7
3.2 Principles of Operation.....	7
<b>4 APPLICATION INFORMATION .....</b>	<b>8</b>
4.1 Acquiring Data From the BASIC Module.....	8
4.1.1 General.....	8
4.1.2 Data Sets #1 and #2.....	8
4.1.3 Data Set #3.....	9
4.1.4 Error Reporting.....	9
4.2 Writing Data to the BASIC Module.....	10
4.3 Handshaking Logic.....	10
4.3.1 General.....	10
4.3.2 Data transfer from the BASIC module to the CPU.....	11
4.3.3 Data transfer from the CPU to the BASIC module.....	11
4.4 Sample Ladder Program .....	12
<b>5 TROUBLESHOOTING .....</b>	<b>13</b>
<b>6 ORDERING INFORMATION .....</b>	<b>15</b>
<b>APPENDIX A DATA SETS.....</b>	<b>16</b>
<b>APPENDIX B SAMPLE SLC 500 DATA TRANSFER PROGRAM .....</b>	<b>19</b>
<b>APPENDIX C CABLE DRAWINGS .....</b>	<b>20</b>
C.1 RS-232 Cable Configuration .....	20
C.2 RS-422 Cable Configuration .....	21
C.2.1 RS-422 Connection.....	21
C.2.2 RS-422 Connection with Surge Voltage Protection.....	22
C.3 RS-485 Cable Configuration .....	23
C.3.1 RS-485 Connection.....	23
C.3.2 RS-485 Connection with Surge Voltage Protection.....	24

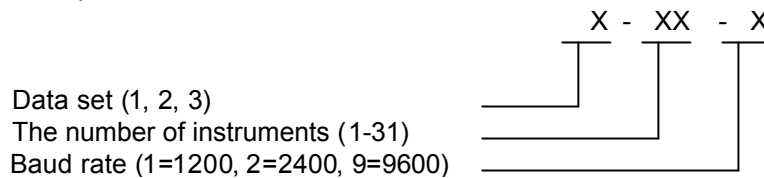
## 1 ABOUT THE COMMUNICATION DRIVER

- The driver provides data exchange between SLC 5/02, 5/03 and 5/04 modular controllers, and SATEC Powermeters via the 1746-BAS BASIC module.
- The EPROM firmware versions can be supplied for 1 to 31 instruments. Three data sets and three baud rates (1200, 2400 and 9600 baud) are available upon order. For available data sets, refer to Appendix A.
- The driver supports all models of SATEC Powermeters, although certain parameters might not be relevant to some models, and in this case will be transmitted as zeroes.
- The time required for polling one instrument and processing the acquired data until the data transfer to the SLC CPU occurs is 5 to 7 seconds, depending on the model of the instrument being polled.
- The instruments are connected to the BASIC module's PRT2 port that is configured as ASCII interface port. The PRT2 port can be configured for RS-422, RS-485 or RS-232 communication. When using RS-232 interface, only one Powermeter may be connected to the BASIC module. With RS-422/RS-485 communications, up to 31 instruments can be connected to the BASIC module in multidrop mode.

## 2 INSTALLATION

### 2.1 EPROM Installation

The SATEC communication driver is supplied in 32 Kbyte EPROM memory modules (the memory module designation is 1747-M4). The version of the driver is marked on the EPROM label by a combination of:



Before installing the EPROM memory module in the memory module socket, configure the socket for the 1747-M4 (32 Kbyte UVPROM) memory module by setting jumper JW3.

Refer to the SLC 500 BASIC Module Design and Integration Manual for additional information on EPROM location and the JW3 jumper pin assignments.

### 2.2 BASIC Module Port PRT2 Configuration

The BASIC module's PRT2 port must be configured as ASCII interface port by setting jumper JW4 as follows:

PRT1 Port = Program port with programmed communication  
PRT2 Port = ASCII Interface Port  
DH485 Port = Run time DH485 only

Refer to the SLC 500 BASIC Module Design and Integration Manual for additional information on jumper JW4 location and pin assignments.

The PRT2 port interface should be configured for the appropriate RS-422, RS-485 or RS-232 communication by setting jumper JW2. Refer to the SLC 500 BASIC Module Design and Integration Manual for additional information on the JW2 jumper location and pin assignments.

### 2.3 Configuring M0-M1 Files and I/O Words Using APS Software

M0 and M1 files and scanned I/O words should be configured in the SLC 500 processor as part of the I/O configuration procedure for the processor file. Refer to the SLC 500 BASIC Module Design and Integration Manual for information on how to define the length of M0 and M1 files.

The number of M0 file words required is 2 words. The number of M1 file words required is 64 words.

The minimum number of scanned input and output words required is 2.

## 2.4 Powermeters Communications Settings

The Powermeters should be configured properly to operate with the BASIC module. The Powermeters communications settings must be compatible with the BASIC module port PRT2 interface settings (RS-422/485/232) and the driver EPROM version used.

The Powermeter communications settings are as follows:

- ASCII 422, ASCII 485 or ASCII 232 mode, according to the interface used.
- Software handshake.
- Baud rate 1200, 2400 or 9600, according to the driver EPROM version used.
- 8 bits, no parity.

The Powermeters communication addresses can be selected from 1 up to the maximum number that the applied version of the driver supports.

Refer to the Powermeter User's Guide for information on how to set communications parameters in your instrument.

## 2.5 Cable Connections

For the cable drawings, refer to Appendix C.

For RS-422/RS-485 balanced data transmission, a twisted pair cable should be used for each communication link. To minimize reflections and reduce cross talk, it is recommended to terminate the ends of lines with the termination resistor of  $120\ \Omega$ , as shown in Appendixes C.2.1 and C.3.1.

When lines are routed through an electrically noisy environment, input protection against switching or lightning induced surge voltages is required in addition to line termination. Surge voltages can be generated by switching operations in power substations, or can occur as a result of voltage deep fade, phase gating controls, contactor relay controls, etc. Transient surge voltage effects may put the BASIC module or CPU out of action and in the worst case lead to destruction of unprotected electronic components. The diagrams of the input protective circuits are shown in Appendixes C.2.2 and C.3.2. You can also use complete surge voltage protection units for RS-422/RS-485 communication available from Phoenix or from other manufacturers of similar protective equipment.

## 3 GENERAL OPERATION

### 3.1 Interfacing to the BASIC Module

The BASIC module interfaces with the SLC processor via the SLC backplane interface. Data is transferred through SLC CPU Module M0 and M1 files as 16-bit signed integers up to 32767.

The data to be sent to the BASIC module must be placed in the M0 file. The size, configuration, and contents of the M0 file are discussed in Section 4.2.

The data to be sent to the CPU is placed into the M1 file. There are three user-selectable formats of data exchange. The structure and contents of these data sets are provided in Appendix A and discussed in Section 4.1.

To guarantee data integrity during data transfer between the BASIC module and the CPU, a special handshaking technique is applied. For handshaking implementation, two bits within one word in the input image table and within one word in the output image table are used. These are discussed in Section 4.3.

### 3.2 Principles of Operation

On power-up, the driver automatically enters a polling loop and continuously polls all the Powermeters listed in the polling list. The driver checks the response time, message integrity and data value ranges.

If the Powermeter response is OK, then data is placed to the CPU M1 file and the request to the CPU for data reading is issued. After the CPU has read data from the M1 file, the driver goes to the next Powermeter.

When the driver fails to receive an acceptable response from a certain Powermeter, it retries twice. If the driver has failed to get any response from the Powermeter, then it considers the Powermeter "dead" and disables it. The CPU is informed of the error by a non-zero error code in word 64 of the M1 file. For the possible error codes, refer to Table 4.1.

Note that disabled Powermeters are resuscitated after 3-6 minutes depending on the number of instruments being polled. On reconnecting a Powermeter, it goes back to the polling mode.

Before polling any Powermeter, the driver always checks whether there are any writes pending in the CPU M0 file. If so, then the appropriate command is sent to the addressed Powermeter or Powermeters, and driver goes back to the polling loop.

## 4 APPLICATION INFORMATION

### 4.1 Acquiring Data From the BASIC Module

#### 4.1.1 General

The driver reads measured values from the Powermeters and transfers them to the SLC CPU via M1 file in suitable format. There are three versions of the driver available that have different data sets being transmitted to the CPU. The CPU M1 file structures are shown in Appendix A.

To transfer numbers greater than 32767 or between 0 and 1, a special technique is used.

When a value range is above 32767, the driver splits the value into two words with the range 0-9999. The number is divided by 10,000 and represented in the next manner: the high word contains the integer part of the value in units of 10000, and the low word contains a fractional part of the value up to 9999. When the value is written to the M1 file, the low order word is written first, followed by the high order word. To process these types of numbers, the number in the high word must be multiplied by 10,000 and added to the number in the low word.

◦ Example:

The phase B voltage value of 25,100 V is represented by a combination of:

word 4 = 5100

word 5 = 2

The actual value is  $2 * 10000 + 5100 = 25100$

When a value is a fractional number, it is multiplied by scale factor of 10 or 100 depending on the number of digits in the fractional part. To process numbers received in these format, they must be divided by a scale factor.

◦ Example:

The frequency value of 50.1 scaled up by 10 is represented as:

word 33 = 501

The actual value is  $501/10 = 50.1$

Note that currents and ampere demands are transferred in one word. Therefore, the driver will not accept the current and ampere demand values above 32,767 A, although the Powermeters might measure currents up to 60,000 A. If the value is above 32767, it will be truncated to 32767.

#### 4.1.2 Data Sets #1 and #2

In data sets #1 and #2, all values are transferred as positive integers in the range of 0 to 32767. In the case of a negative value, the sign is transferred in the additional word following the value, as an integer of 0 for the positive sign and of 1 for the negative sign.

◦ Examples:

1. The phase A reactive power of value 130,750 kvar is transferred as:

word 34 = 750

word 35 = 13

word 36 = 0

The actual value is  $13 \times 10000 + 750 = 130,750$

2. The phase A reactive power of value -130,750 kvar is transferred as:

word 34 = 750

word 35 = 13

word 36 = 1

The actual value is  $-(13 \times 10000 + 750) = -130,750$

Total active power is transferred only for positive values. In the case of the negative total power, the value will be zeroed.

### 4.1.3 Data Set #3

In data set #3, values are transferred as signed integers in the range of -32768 to +32767. The sign is transferred as a high bit of an integer value. In the case of the negative value being scaled down and transferred in two words, both the high and low words will be negative integers.

◦ Examples:

1. The reactive power A of value 130,750 kvar is transferred as:

word 27 = 750

word 28 = 13

The actual value is  $13 \times 10000 + 750 = 130,750$

2. The reactive power A of value -130,750 kvar is transferred as:

word 27 = -750

word 28 = -13

The actual value is  $(-13) \times 10000 + (-750) = -130,750$

### 4.1.4 Error Reporting

The status of the instrument being polled is always reported in word 64 of the M1 file. When reading the M1 file, the CPU should check this word before applying data. The value of 0 shows the data in the M1 file is OK. When the BASIC module fails to receive data from the instrument, a non-zero error code is written to word 64 accompanied by the instrument address in word 1. All other words will be zeroes. The possible error code values are shown in Table 4-1.

**Table 4-1 Error Codes**

Error code (word 64)	Description
0	No errors found
1	The instrument doesn't respond
2	Check error (framing or check field error is detected)

## 4.2 Writing Data to the BASIC Module

The driver allows the CPU to reset accumulated energies and demands in the selected Powermeter, or synchronously in all Powermeters on line, if this function is supported by the instruments. The CPU request is transmitted to the BASIC module via the CPU M0 file. The CPU M0 file structure is shown in Table 4-2.

**Table 4-2 CPU M0 file structure**

Field No.	Field Name	CPU M0 file word	Range	Description
1	Instrument address	1	0-31	0 = all the instruments
2	Command	2	1-2	1 = Reset accumulated energies 2 = Reset maximum demands

The instrument address value of 0 will apply to all the instruments connected to the BASIC module.

The command value of 1 will result in resetting the active (+kWh), returned (-kWh) and reactive (kvarh) energies. The command value of 2 will reset the active power maximum demand, apparent power maximum demand, and all of the ampere maximum demand values.

The command value of 0 has no action.

## 4.3 Handshaking Logic

### 4.3.1 General

Because file integrity is not guaranteed during data transfer via Module files M0 and M1, a special handshaking logic is applied to avoid the use of data while updating the file.

For handshaking implementation, two least significant bits in word 1 in the CPU output and input image tables are used. Bit 0 in either cases controls access to the CPU M0 file, and bit 1 controls access to the M1 file. Table 4-3 shows the handshaking bits' assignments and letters that are used as their designations.

**Table 4-3 Handshaking Bits**

CPU input word 1	CPU output word 1	Bit designation	CPU File	Function
Bit 1		A	M1	Request to send
	Bit 1	B	M1	Clear to send
Bit 0		C	M0	Clear to send
	Bit 0	D	M0	Request to send

Here is how handshaking operates.

### 4.3.2 Data transfer from the BASIC module to the CPU

Table 4-4 explains the transfer algorithm in a form of the state machine.

**Table 4-4 Data transfer through the CPU M1 file**

CPU input status	CPU output status	SLC action	BASIC module action	Comment
A=0	B=0	No action	A:=1	Request to send
A=1	B=0	B:=1	No action	Clear to send
A=1	B=1	No action	1. Transfers data to M1 file 2. A:=0	M1 file has been updated
A=0	B=1	1. Copies M1 file to data file 2. B:=0	No action	M1 file has been processed
A=0	B=0	No action	No action	

The algorithm of the data transfer from the BASIC module to the CPU is as follows:

1. The BASIC module sets bit A to 1 requesting the access to the M1 file.
2. The CPU enables the access to the M1 file by setting bit B to 1.
3. The BASIC module transfers data to the M1 file.
4. The BASIC module resets bit A to 0 allowing the CPU to access the M1 file.
5. The CPU copies the M1 file to data file.
6. The CPU resets bit B to 0 enabling new requests.

### 4.3.3 Data transfer from the CPU to the BASIC module

Table 4-5 explains the transfer algorithm in a form of the state machine.

**Table 4-5 Data transfer through the CPU M0 file**

CPU input status	CPU output status	SLC action	BASIC module action	Comment
C=0	D=0	D:=1	No action	Request to send
C=0	D=1	No action	C:=1	Clear to send
C=1	D=1	1. Transfers data to M0 file 2. D:=0	No action	M0 file has been updated
C=1	D=0	No action	1. Copies M0 file to data buffer 2. C:=0	M0 file has been processed
C=0	D=0	No action	No action	

The algorithm of the data transfer from the CPU to the BASIC module is as follows:

1. The CPU sets bit D to 1 requesting the access to the M0 file.
2. The BASIC module enables the access to the M0 file by setting bit C to 1.
3. The CPU transfers data to the M0 file.
4. The CPU resets bit D to 0 allowing the BASIC module to access the M0 file.
5. The BASIC module copies the M0 file to data buffer.
6. The BASIC module resets bit C to 0 enabling new requests.

## 4.4 Sample Ladder Program

This section offers a sample SLC 500 ladder program that communicates with the BASIC module driver and implements handshaking described above. The ladder program diagram is shown in Appendix B.

The program assumes the following:

1. The BASIC module slot number is 6.
2. The data file N7 is used for output. The CPU command words is written into words N7:0 and N7:1.
3. Bit B3/0 is used for triggering the CPU command output to implement one-shot write to the BASIC module.
4. The data file N9 is used for input. Data being copied from the M1 file is located in words N9:0 through N9:63. Note that word N9:63 will contain an error code.

For more information concerning the programming technique when applying M0 and M1 files, refer to the SLC 500 BASIC Module Design and Integration Manual.

## 5 TROUBLESHOOTING

This chapter describes common problems you might encounter during installation of the communication driver, and offers suggestions for corrective action.

It is recommended that you apply a sample ladder program discussed in Section 4.4 for testing the CPU-BASIC module interface operation. With this program you can monitor the data exchange between the CPU and the BASIC module through the CPU data files.

### The CPU doesn't interface to the BASIC module

The symptom indicating the problem is that the address of the instrument being polled that is located in word 1 in the CPU input data file is still unchanged while you are monitoring the file with APS software.

To ensure that the BASIC module operates properly, carry out the following procedure:

1. Switch the power supply off.
2. Unplug the network connector from port PRT2.
3. Turn power on.

If on power up, LED PRT2 located on the front panel of the BASIC module doesn't flash, it could be for one of the following reasons:

1. Incorrect setting of the JW3 jumper.
2. Improper EPROM installation.

For information on the jumper setting for jumper JW3, see section 2.1. For additional information on EPROM location and jumper JW3 pin assignments, refer to the SLC 500 BASIC Module Design and Integration Manual.

If on power up, LED PRT2 flashes three times with short intervals, for about 1/4 of a second, between flashes, which means that the driver requests the instrument, then the BASIC module is operating properly and the problem is located in the CPU. Determine whether one of the following is the cause:

1. Check whether the CPU is in RUN mode.
2. Make sure the I/O configuration settings for the BASIC module slot are correct. Refer to Section 2.3 for information about configuring I/O for the BASIC module.
3. Check whether your ladder program is correct. If you are using a sample ladder program provided in Appendix B, make sure that you have accommodated it properly to the slot and to your data files.

If the handshaking mechanism operates properly, then, while the network cable is disconnected, LED PRT2 will flash continuously with short intervals between flashes signifying the driver polls instruments and transmits data to the CPU via the M1 file.

## The BASIC module doesn't interface to the instruments

The symptom that indicates a problem is that a non-zero error code is received by the CPU in word 64 in the M1 file. It might be reported for all or some of the Powermeters connected to the BASIC module. Note that for the disconnected instruments, that would be a normal response.

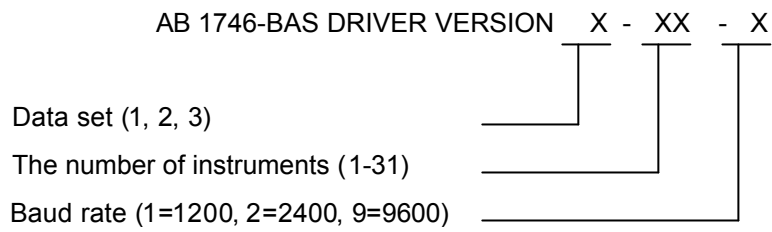
If you encounter that the problem exists for all the instruments, determine whether one of the following is the cause:

1. Make sure that port PRT2 is configured properly and the settings for jumpers JW2 and JW4 are correct. For information on the jumper settings for port PRT2, refer to Section 2.2.
2. Check whether the Powermeters turned on, you have the correct cable, and it is properly connected to the BASIC module port PRT2 and to the Powermeters. For cable drawings, refer to Appendix C.
3. Make sure all communications settings (baud rate, data bits, parity, protocol, instrument address) are set correctly in the instruments, and the baud rate setting conforms to the driver version. For information about communications settings for the Powermeters, refer to Section 2.4.

When the problem exists for some of the Powermeters, the cause is often related to improper cable connection or incompatible communications settings made in the instruments.

## 6 ORDERING INFORMATION

The ordering number is formed by a combination of:



Example: order for 16 instruments, data set #2, 9600 baud:

AB 1746-BAS DRIVER VERSION 2-16-9

## APPENDIX A DATA SETS

**Table A-1 Data set #1**

Field No.	Field Name	CPU M1 file word			Units	Value evaluation formula	Range
		LO	HI	SIGN			
1	Instrument address	1				LO	1 - 31
2	Voltage A	2	3		V	HI*10000+LO	0 - 999,000
3	Voltage B	4	5		V		
4	Voltage C	6	7		V		
5	Current A	8			A	LO	0 - 32767
6	Current B	9			A		
7	Current C	10			A		
8	Active Power A	11	12	13	kW	HI*10000+LO	± 0 - 5,184,000
9	Active Power B	14	15	16	kW	SIGN = 0 means '+'	
10	Active Power C	17	18	19	kW	SIGN = 1 means '-'	
11	Power Factor A	20		21		LO/100	± 0 - 1.00
12	Power Factor B	22		23		SIGN = 0 means '+'	
13	Power Factor C	24		25		SIGN = 1 means '-'	
14	Total Active Power	26	27		kW	HI*10000+LO	0 - 5,184,000
15	System Power Factor	28		29		LO/100 SIGN = 0 means '+' SIGN = 1 means '-'	± 0 - 1.00
16	Active Energy	30	31		kWh	HI*10000+LO	0 - 99,999,000
17	Unbalanced Current or Ground Leakage	32			A or mA	LO	0 - 32767
18	Frequency	33			Hz	LO/10	45.0 - 65.0
19	Reactive Power A	34	35	36	kvar	HI*10000+LO	± 0 - 5,184,000
20	Reactive Power B	37	38	39	kvar	SIGN = 0 means '+'	
21	Reactive Power C	40	41	42	kvar	SIGN = 1 means '-'	
22	Apparent Power A	43	44		kVA	HI*10000+LO	0 - 5,184,000
23	Apparent Power B	45	46		kVA		
24	Apparent Power C	47	48		kVA		
25	Reactive Energy	49	50	51	kvarh	HI*10000+LO SIGN = 0 means '+' SIGN = 1 means '-'	-9,999,000 - 99,999,000
26	Total Reactive Power	52	53	54	kvar	HI*10000+LO SIGN = 0 means '+' SIGN = 1 means '-'	± 0 - 5,184,000
27	Total Apparent Power	55	56		kVA	HI*10000+LO	0 - 5,184,000
28	Active Power Maximum Demand	57	58		kW	HI*10000+LO	0 - 5,184,000
29	Accumulated Active Power Demand	59	60		kW	HI*10000+LO	0 - 5,184,000
30	Error code	64				LO	see Table 4-1

**Table A-2 Data set #2**

Field No.	Field Name	CPU M1 file word			Units	Value evaluation formula	Range
		LO	HI	SIGN			
1	Instrument address	1				LO	1 - 31
2	Voltage A	2	3		V	HI*10000+LO	0 - 999,000
3	Voltage B	4	5		V		
4	Voltage C	6	7		V		
5	Current A	8			A	LO	0 - 32767
6	Current B	9			A		
7	Current C	10			A		
8	Active Power A	11	12	13	kW	HI*10000+LO	± 0 - 5,184,000
9	Active Power B	14	15	16	kW	SIGN = 0 means '+'	
10	Active Power C	17	18	19	kW	SIGN = 1 means '-'	
11	Power Factor A	20		21		LO/100	± 0 - 1.00
12	Power Factor B	22		23		SIGN = 0 means '+'	
13	Power Factor C	24		25		SIGN = 1 means '-'	
14	Total Active Power	26	27		kW	HI*10000+LO	0 - 5,184,000
15	System Power Factor	28		29		LO/100 SIGN = 0 means '+' SIGN = 1 means '-'	± 0 - 1.00
16	Active Energy	30	31		kWh	HI*10000+LO	0 - 99,999,000
17	Unbalanced Current or Ground Leakage	32			A or mA	LO	0 - 32767
18	Frequency	33			Hz	LO/10	45.0 - 65.0
19	Reactive Power A	34	35	36	kvar	HI*10000+LO	± 0 - 5,184,000
20	Reactive Power B	37	38	39	kvar	SIGN = 0 means '+'	
21	Reactive Power C	40	41	42	kvar	SIGN = 1 means '-'	
22	THD of Voltage A	43			%	LO/10	0 - 99.9
23	THD of Voltage B	44			%		
24	THD of Voltage C	45			%		
25	THD of Current A	46			%		
26	THD of Current B	47			%		
27	THD of Current C	48			%		
28	Reactive Energy	49	50	51	kvarh	HI*10000+LO SIGN = 0 means '+' SIGN = 1 means '-'	-9,999,000 - 99,999,000
29	Total Reactive Power	52	53	54	kvar	HI*10000+LO SIGN = 0 means '+' SIGN = 1 means '-'	± 0 - 5,184,000
30	Total Apparent Power	55	56		kVA	HI*10000+LO	0 - 5,184,000
31	Active Power Maximum Demand	57	58		kW	HI*10000+LO	0 - 5,184,000
32	Accumulated Active Power Demand	59	60		kW	HI*10000+LO	0 - 5,184,000
33	Ampere Maximum Demand A	61			A	LO	0 - 32767
34	Ampere Maximum Demand B	62			A		
35	Ampere Maximum Demand C	63			A		
36	Error code	64				LO	see Table 4-1

**Table A-3 Data set #3**

Field No.	Field Name	CPU M1 file word		Units	Value evaluation formula	Range
		LO	HI			
1	Instrument address	1			LO	1 - 31
2	Voltage A	2	3	V	HI*10000+LO	0 - 999,000
3	Voltage B	4	5	V		
4	Voltage C	6	7	V		
5	Current A	8		A	LO	0 - 32767
6	Current B	9		A		
7	Current C	10		A		
8	Active Power A	11	12	kW	HI*10000+LO	± 0 - 5,184,000
9	Active Power B	13	14	kW		
10	Active Power C	15	16	kW		
11	Power Factor A	17			LO/100	± 0 - 1.00
12	Power Factor B	18				
13	Power Factor C	19				
14	Total Active Power	20	21	kW	HI*10000+LO	± 0 - 5,184,000
15	System Power Factor	22			LO/100	± 0 - 1.00
16	Active Energy	23	24	kWh	HI*10000+LO	0 - 99,999,000
17	Unbalanced Current or Ground Leakage	25		A or mA	LO	0 - 32767
18	Frequency	26		Hz	LO/10	45.0 - 65.0
19	Reactive Power A	27	28	kvar	HI*10000+LO	± 0 - 5,184,000
20	Reactive Power B	29	30	kvar		
21	Reactive Power C	31	32	kvar		
22	Apparent Power A	33	34	kVA	HI*10000+LO	0 - 5,184,000
23	Apparent Power B	35	36	kVA		
24	Apparent Power C	37	38	kVA		
25	Reactive Energy	39	40	kvarh	HI*10000+LO	-9,999,000 - 99,999,000
26	Total Reactive Power	41	42	kvar	HI*10000+LO	± 0 - 5,184,000
27	Total Apparent Power	43	44	kVA	HI*10000+LO	0 - 5,184,000
28	Active Power Maximum Demand	45	46	kW	HI*10000+LO	0 - 5,184,000
29	Accumulated Active Power Demand	47	48	kW	HI*10000+LO	0 - 5,184,000
30	Ampere Maximum Demand A	49		A	LO	0 - 32767
31	Ampere Maximum Demand B	50		A		
32	Ampere Maximum Demand C	51		A		
33	External contacts status (for 290/B)	52			LO	0 - 255
34	Returned Energy	53	54	kWh	HI*10000+LO	-9,999,000 - 0
35	Apparent Power Maximum Demand	55	56	kVA	HI*10000+LO	0 - 5,184,000
36	THD of Voltage A	57		%	LO/10	0 - 99.9
37	THD of Voltage B	58		%		
38	THD of Voltage C	59		%		
39	THD of Current A	60		%		
40	THD of Current B	61		%		
41	THD of Current C	62		%		
42	Error code	64			LO	see Table 4-1

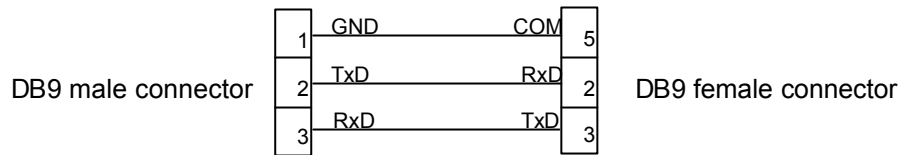


## APPENDIX C CABLE DRAWINGS

### C.1 RS-232 Cable Configuration

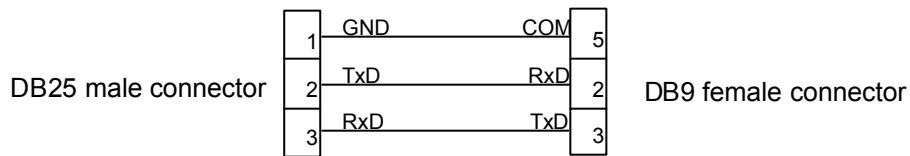
**Series PM -280/288/270/290/290H  
Powermeter**

**1746-BAS BASIC module**



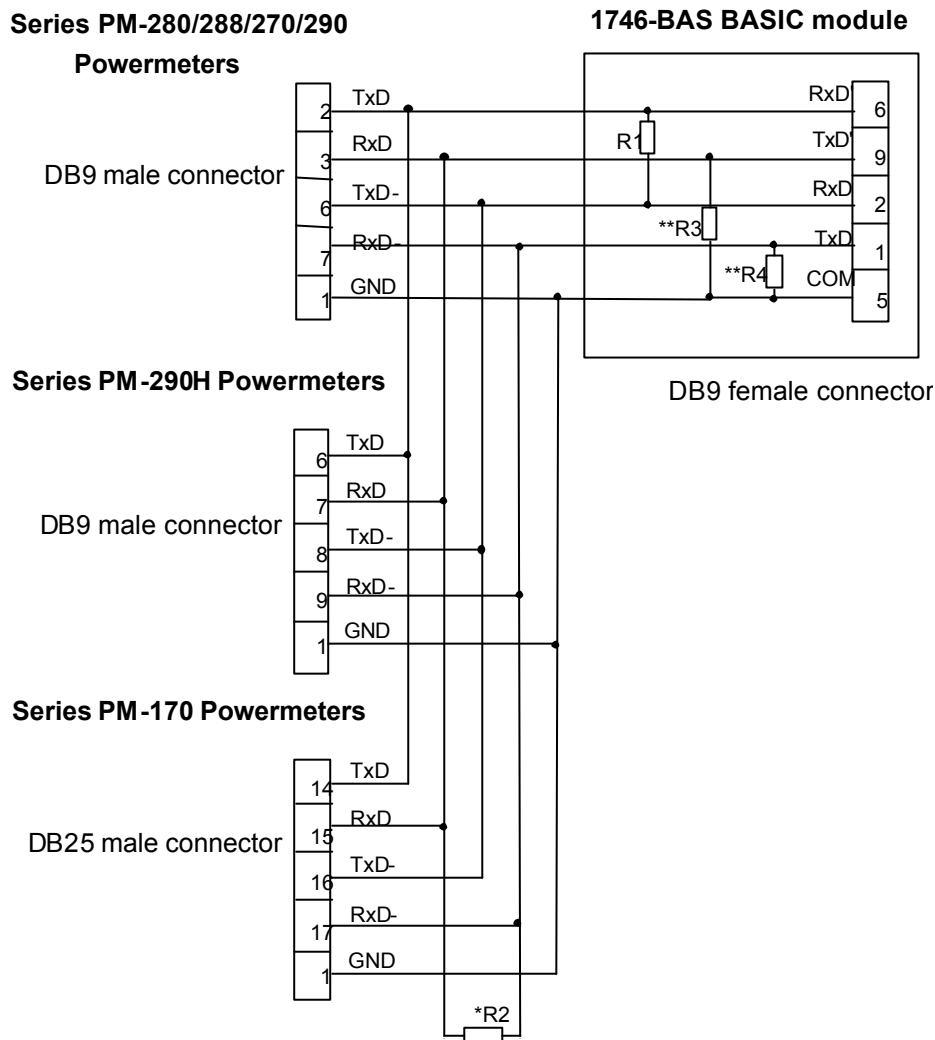
**Series PM -170 Powermeter**

**1746-BAS BASIC module**



## C.2 RS-422 Cable Configuration

### C.2.1 RS-422 Connection



R1,R2 - 120  $\Omega$

R3,R4 - 1.2 k  $\Omega$

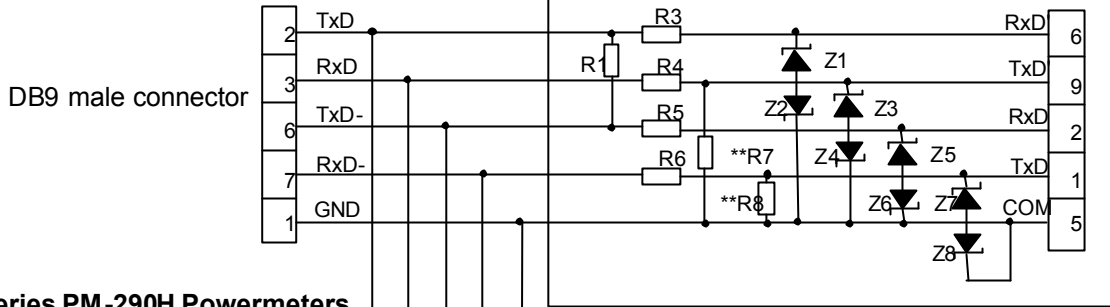
\* A termination resistor should only be connected at the furthest end of the cable

\*\* In RS-422 communications, a reference or common line should be used in addition to signal lines for stable operation. Connect pin 1 (instrument) to pin 5 on the BASIC module. If you do not use a reference line, place pull-down resistors R3 and R4 on both outputs of the driver to avoid leaving the line floating when the line is idle.

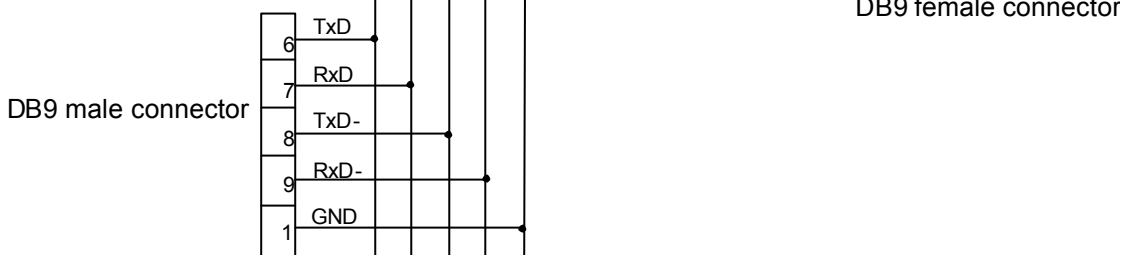
## C.2.2 RS-422 Connection with Surge Voltage Protection

### Series PM-280/288/270/290

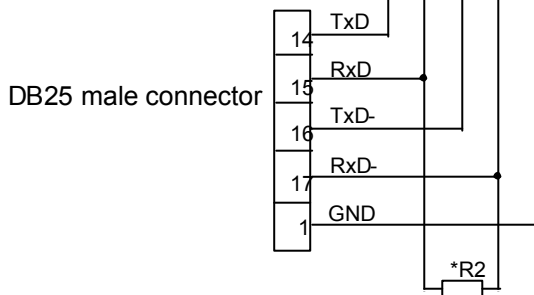
#### Powermeters



### Series PM-290H Powermeters



### Series PM-170 Powermeters



Z1,Z3,Z5,Z7 - 1N4736A

Z2,Z4,Z6,Z8 - 1N4728A

R1,R2 - 120  $\Omega$

R3,R4,R5,R6 - 33  $\Omega$

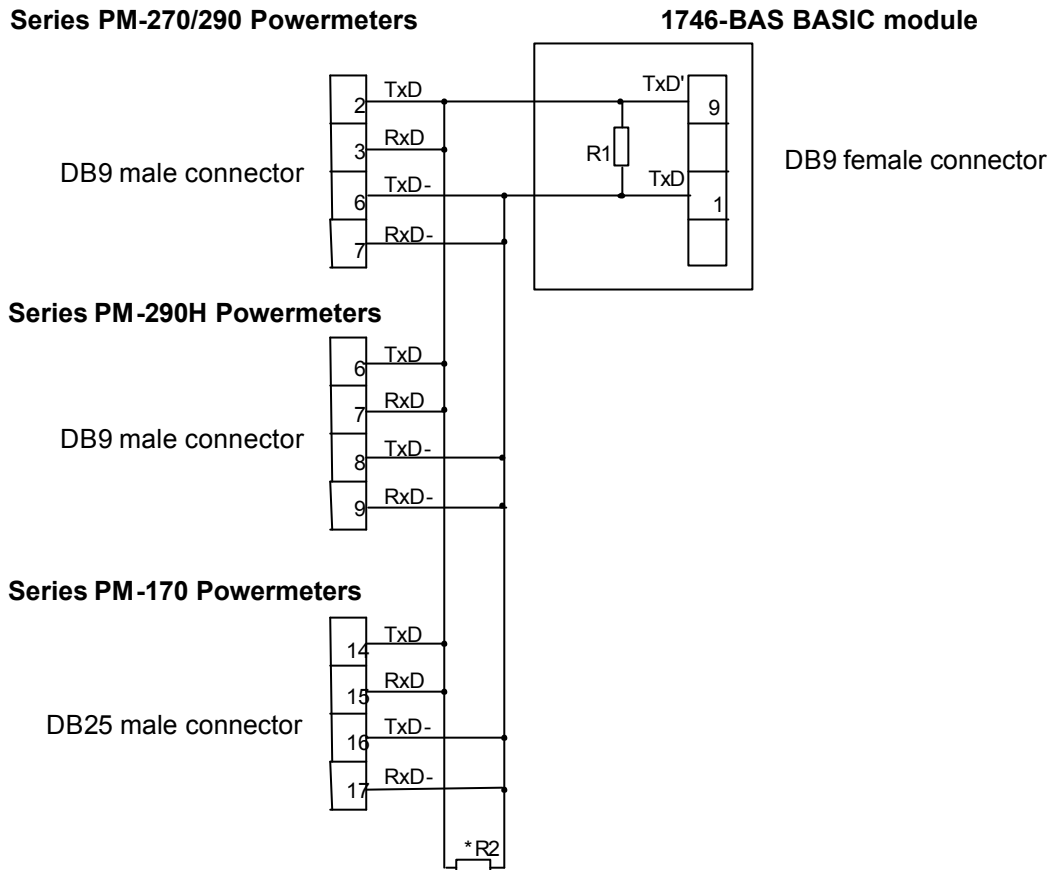
R7,R8 - 1.2 k  $\Omega$

\* A termination resistor should only be connected at the furthest end of the cable

\*\* In RS-422 communication, a reference or common line should be used in addition to signal lines for stable operation. Connect pin 1 of the instruments with pin 5 on the BASIC module. If you don't use a reference line, place pull-down resistors R7 and R8 on both outputs of the driver to avoid leaving the line floating when the line is idle

## C.3 RS-485 Cable Configuration

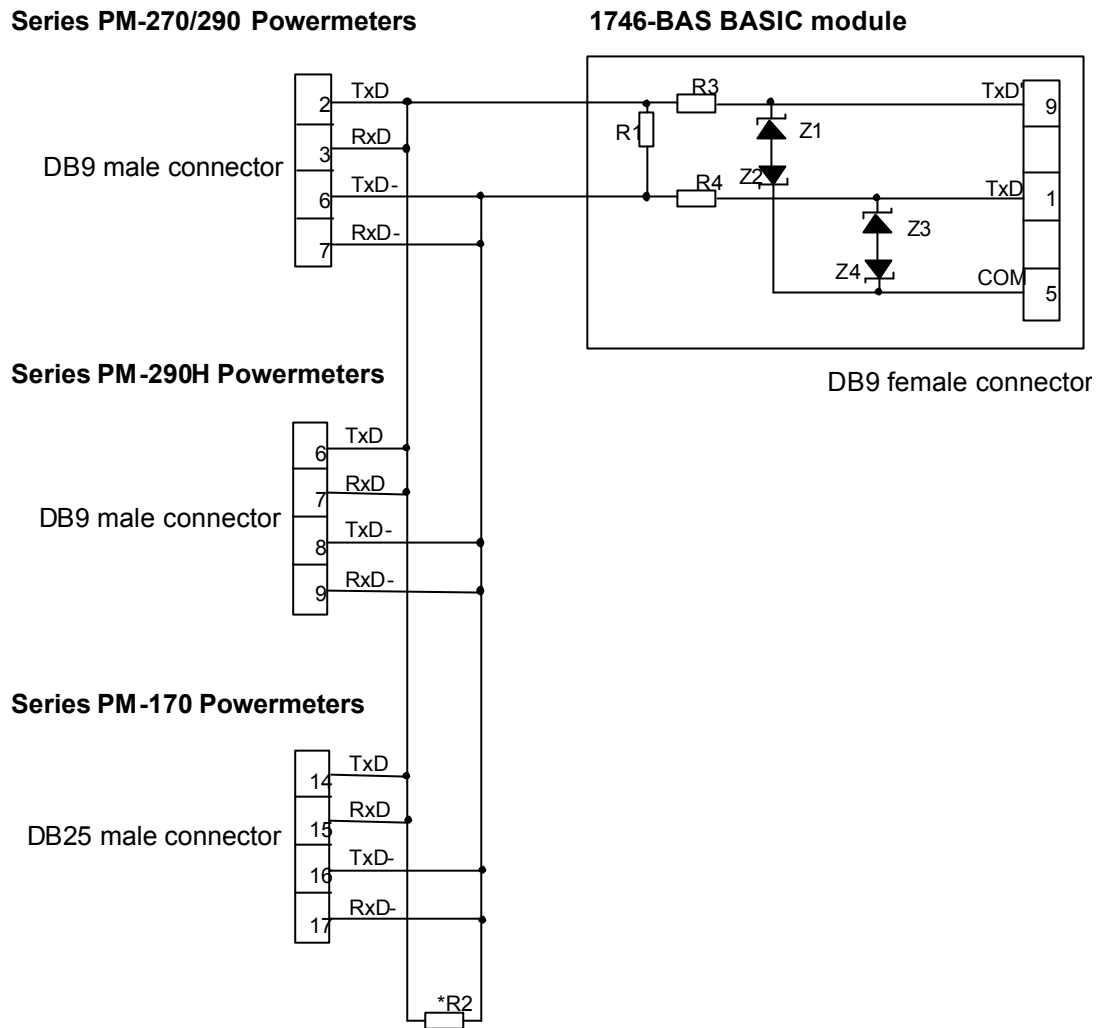
### C.3.1 RS-485 Connection



R1,R2 - 120  $\Omega$

\* A termination resistor should only be connected at the furthest end of the cable

### C.3.2 RS-485 Connection with Surge Voltage Protection



Z1,Z3 - 1N4736A

Z2,Z4 - 1N4728A

R1,R2 - 120  $\Omega$

R3,R4 - 33  $\Omega$

\* A termination resistor should only be connected at the furthest end of the cable