

## MODBUS INTERFACE

Part Number: PIM-MB-01

### 1. Introduction

IMS2 and CSX Series soft starters can be controlled and monitored across an RS485 serial communication network using the Modbus RTU and AP ASCII protocols.

For users requiring simple control of a CSX soft starter using Modbus RTU or AP ASCII, the instructions below describe the installation and operation of the Modbus Interface.

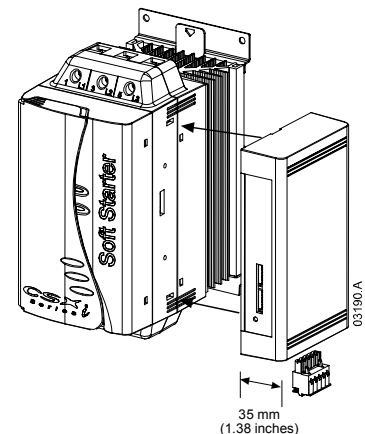
IMS2 soft starters have Modbus RTU and AP ASCII protocol support built in - refer to the IMS2 Users Manual for details of message formats.

IMS2 and CSX Series soft starters can also connect to the network via a correctly configured Remote Operator - refer to *Appendix A* for details.

### 2. Installation

Install the Modbus Interface using the following procedure:

1. Remove control power and mains supply from the soft starter.
2. Attach the Modbus Interface to the starter as shown.
3. Apply control power to the soft starter.

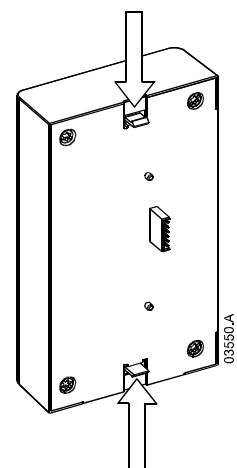


#### CAUTION

Remove control power and mains supply from the soft starter before attaching or removing accessories. Failure to do so may damage the equipment.

Remove the Modbus Interface using the following procedure:

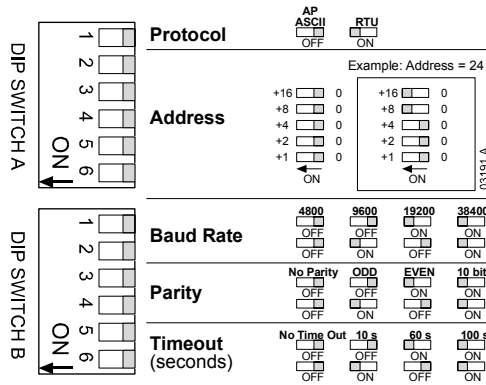
1. Take the interface off-line.
2. Remove control power and mains supply from the soft starter.
3. Push a small flat-bladed screwdriver into the slots at the top and bottom of the interface and depress the retaining clips.
4. Pull the interface away from the soft starter.



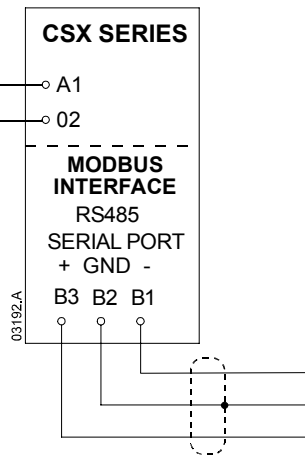
# INSTRUCTIONS

## 3. Adjustment

Network communication parameters must be set on the Modbus Interface.



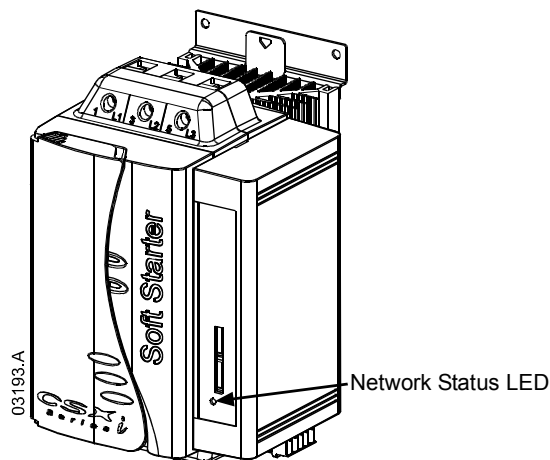
## 4. Connection



### NOTE

For the Modbus Interface to operate correctly, a link must be fitted across terminals A1-02 on the starter.

## 5. Network Status LED



The Network Status LED indicates the state of the communications link between the interface and the network. LED operation is as follows:

| Off           | On             | Flashing         |
|---------------|----------------|------------------|
| No connection | Network active | Network inactive |



**NOTE**

If the network is inactive, the soft starter may trip if the Communications Timeout function has been set on the interface. When communication is restored, the CSX will require an individual Reset.

**6. Modbus Register**

| Address | Function          | Type  | Description   |   |
|---------|-------------------|-------|---|---|
| 40002   | Command           | Write | 1 = Start<br>2 = Stop<br>3 = Reset<br>4 = Quick stop<br>5 = Forced communication trip   |   |
| 40003   | Starter status    | Read  | <b>Bit</b> <b>Description</b>   |   |
|         |                   |       | 0 to 3  | 1 = Ready<br>2 = Starting<br>3 = Running<br>4 = Stopping<br>6 = Tripped |
|         |                   |       | 4   | 1 = Positive phase sequence   |
|         |                   |       | 5   | <i>Unallocated</i>  |
|         |                   |       | 6   | <i>Unallocated</i>  |
| 40004   | Trip code         | Read  | 7   | <i>Unallocated</i>  |
|         |                   |       | 255 = No trip<br>1 = Excess start time <sup>1</sup><br>2 = Motor overload <sup>1</sup><br>3 = Motor thermistor <sup>1</sup><br>4 = Phase imbalance <sup>1</sup><br>5 = Supply frequency<br>6 = Phase sequence <sup>1</sup><br>8 = Power circuit fault<br>15 = Communication failure between starter and interface<br>16 = Communication failure between interface and network |   |
| 40005   | Motor current     | Read  | <sup>1</sup>  |   |
| 40006   | Motor temperature | Read  | <sup>1</sup>  |   |

<sup>1</sup> = CSXi models only.

## 7. Modbus Hex Functions

CSX soft starters support two functions:

- 03 (multiple read)
- 06 (single write)

The CSX does not support broadcast functions.



### NOTE

Command, Starter status, Trip code, Motor current and Motor temperature must be sent individually (ie one data word request at a time).

Least significant bit is transmitted first.

Examples

Command: Start

| Message | Starter Address | Function Code | Register Address | Data | CRC        |
|---------|-----------------|---------------|------------------|------|------------|
| In      | 20              | 06            | 40002            | 1    | CRC1, CRC2 |
| Out     | 20              | 06            | 40002            | 1    | CRC1, CRC2 |

Starter Status: Running

| Message | Starter Address | Function Code | Register Address | Data     | CRC        |
|---------|-----------------|---------------|------------------|----------|------------|
| In      | 20              | 03            | 40003            | 1        | CRC1, CRC2 |
| Out     | 20              | 03            | 2                | xxxx0011 | CRC1, CRC2 |

Trip Code: Motor overload

| Message | Starter Address | Function Code | Register Address | Data     | CRC        |
|---------|-----------------|---------------|------------------|----------|------------|
| In      | 20              | 03            | 40004            | 1        | CRC1, CRC2 |
| Out     | 20              | 03            | 2                | 00000010 | CRC1, CRC2 |

## 8. Modbus Error Codes

| Code | Description               | Example   |
|------|---------------------------|---|
| 01   | Illegal function code     | Function other than 03 or 06  |
| 02   | Illegal data address      | Not in range 40002 to 40255   |
| 03   | Not readable data         | Register not allowed for data reading                                   |
| 04   | Not writable data         | Register not allowed for data writing                                   |
| 05   | Data boundary fault       | Multiple data transfer across data boundary, or data size more than 127 |
| 06   | Invalid command code      | Writing "6" into 40002  |
| 07   | Illegal parameter read    | Invalid parameter number  |
| 08   | Illegal parameter write   | Invalid parameter number, read only, or hidden parameter                |
| 09   | Unsupported command       | Writing parameter with starter running (not applicable for CSX Series)  |
| 10   | Local communication error | Communication error between Modbus slave and starter                    |



### NOTE

Some of the above codes are different from those defined in the Modbus Application Protocol Specification available on [www.modbus.org](http://www.modbus.org).

AP ASCII Protocol

This section describes AP ASCII communications with CSX soft starters.

The message fragments used in communicating with the CSX are shown below. The message fragments may be assembled into complete messages as described in the sections that follow.



**NOTE**

Data transmitted to and from the Modbus Interface must be in 8 bit ASCII, no parity, 1 stop bit.

| Message Fragment Type      | ASCII Character String or (Hexadecimal Character String) |
|----------------------------|--|
| Send address               | EOT [nn] [lrc] ENQ or (04h [nn] [lrc] 05h)               |
| Send command               | STX [ccc] [lrc] ETX or (02h [ccc] [lrc] 03h)             |
| Send request               |  |
| Receive data               | STX [dddd] [lrc] ETX or (02h [dddd] [lrc] 03h)           |
| Receive status             | STX [ssss] [lrc] ETX or (02h [ssss] [lrc] 03h)           |
| ACK (acknowledge)          | ACK or (06h)   |
| NAK (negative acknowledge) | NAK or (15h)   |
| ERR (error)                | BEL or (07h)   |

nn = two byte ASCII number representing the soft starter address where each decimal digit is represented by n.

lrc = two byte longitudinal redundancy check in hexadecimal.

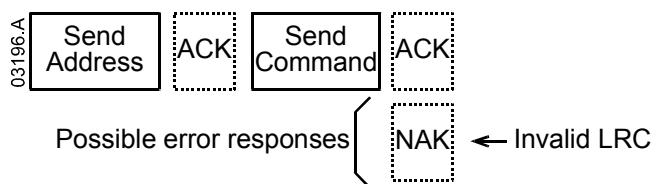
ccc = three byte ASCII command number where each character is represented by c.

dddd = four byte ASCII number representing the current or temperature data where each decimal digit is represented by d.

ssss = four byte ASCII number. The first two bytes are ASCII zero. The last two bytes represent the nibbles of a single byte of status data in hexadecimal.

**Commands**

Commands can be sent to the soft starter using the following format:



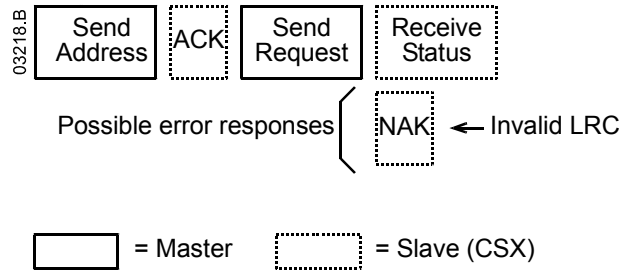
= Master     = Slave (CSX)

| Command                   | ASCII | Comment   |
|---------------------------|-------|---|
| Start                     | B10   | Initiates a start   |
| Stop                      | B12   | Initiates a stop  |
| Reset                     | B14   | Resets a trip state   |
| Coast to stop             | B16   | Initiates an immediate removal of voltage from the motor. Any soft stop settings are ignored. |
| Forced communication trip | B18   | Causes a communications trip  |

# INSTRUCTIONS

## Status Retrieval

Soft starter status can be retrieved using the following format:

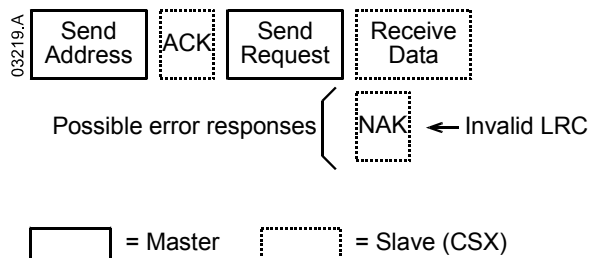


| Request        | ASCII | Receive Status (ssss)   |   |
|----------------|-------|---|---|
| Trip code      | C18   | Requests the trip status of the starter.<br>255 = No trip<br>1 = Excess start time <sup>1</sup><br>2 = Motor overload <sup>1</sup><br>3 = Motor thermistor <sup>1</sup><br>4 = Phase imbalance <sup>1</sup><br>5 = Supply frequency<br>6 = Phase sequence <sup>1</sup><br>8 = Power circuit fault<br>15 = Communication failure between starter and interface<br>16 = Communication failure between interface and network |   |
| Starter status | C22   | <b>Bit</b>  | <b>Description</b>  |
|                |       | 0 to 3  | 0 = Not used<br>1 = Ready<br>2 = Starting (including Pre-start tests)<br>3 = Running<br>4 = Stopping<br>6 = Tripped |
|                |       | 4   | 1 = Forward phase sequence detected   |
|                |       | 5   | <i>Unallocated</i>  |
|                |       | 6   | <i>Unallocated</i>  |
|                |       | 7   | <i>Unallocated</i>  |

<sup>1</sup> = CSXi models only.

## Data Retrieval

Data can be retrieved from closed loop CSXi models using the following format:



| Request           | ASCII | Receive Data (dddd)  |
|-------------------|-------|--|
| Motor current     | D10   | Requests motor current. The data is four byte decimal ASCII. Minimum value 0000 A, maximum value 9999 A.   |
| Motor temperature | D12   | Requests the calculated value of the motor thermal model as a % of motor thermal capacity. The data is four byte decimal ASCII. Minimum value is 0000%. Trip point is 0105%. |

**Calculating the Checksum (LRC)**

Each command string sent to and from the starter includes a checksum. The form used is the longitudinal redundancy check (LRC) in ASCII hex. This is an 8-bit binary number represented and transmitted as two ASCII hexadecimal characters.

To calculate LRC:

1. Sum all ASCII bytes
2. Mod 256
3. 2's complement
4. ASCII convert

For example Command String (Start):

ASCII STX B 1 0  
 or 02h 42h 31h 30h

| ASCII | Hex     | Binary    |                    |
|-------|---------|-----------|--------------------|
| STX   | 02h     | 0000 0010 |                    |
| B     | 42h     | 0100 0010 |                    |
| 1     | 31h     | 0011 0001 |                    |
| 0     | 30h     | 0011 0000 |                    |
|       | A5h     | 1010 0101 | SUM (1)            |
|       | A5h     | 1010 0101 | MOD 256 (2)        |
|       | 5Ah     | 0101 1010 | 1's COMPLEMENT     |
|       | 01h     | 0000 0001 | + 1 =              |
|       | 5Bh     | 0101 1011 | 2's COMPLEMENT (3) |
| ASCII | 5 B     |           | ASCII CONVERT (4)  |
| or    | 35h 42h |           | LRC CHECKSUM       |

The complete command string becomes:

ASCII STX B 1 0 5 B ETX  
 or 02h 42h 31h 30h 35h 42h 03h

To verify a received message containing an LRC:

1. Convert last two bytes of message from ASCII to binary
2. Left shift 2<sup>nd</sup> to last byte four bits
3. Add to last byte to get binary LRC
4. Remove last two bytes from message
5. Add remaining bytes of message
6. Add binary LRC
7. Round to one byte
8. The result should be zero

Response or status bytes are sent from the starter as an ASCII string:

STX [d1]h [d2]h [d3]h [d4]h LRC1 LRC2 ETX  
 d1 = 30h  
 d2 = 30h  
 d3 = 30h plus upper nibble of status byte right shifted by four binary places  
 d4 = 30h plus lower nibble of status byte

For example status byte = 1Fh, response is:

STX 30h 30h 31h 46h LRC1 LRC2 ETX

**10. Appendix A - Modbus Control via Remote Operator**

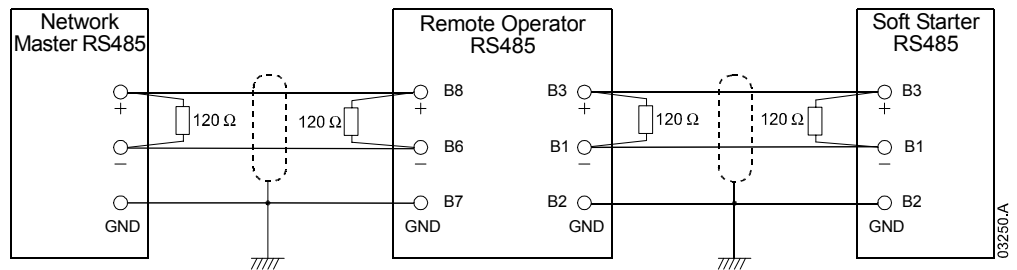
To control a soft starter via an RS485 serial communications network using the Remote Operator, connect the Remote Operator to the network as described in the following sections.

**Grounding and Shielding**

Twisted pair data cable with earth shield is recommended. The cable shield should be connected to a GND device terminal at both ends and one point of the site protective earth.

**Termination Resistors**

In long cable runs prone to excessive noise interference, termination resistors should be installed between B1 (-) and B3 (+) of the soft starter and the Remote Operator. This resistance should match the cable impedance (typically 120 Ω). Do not use wire wound resistors.



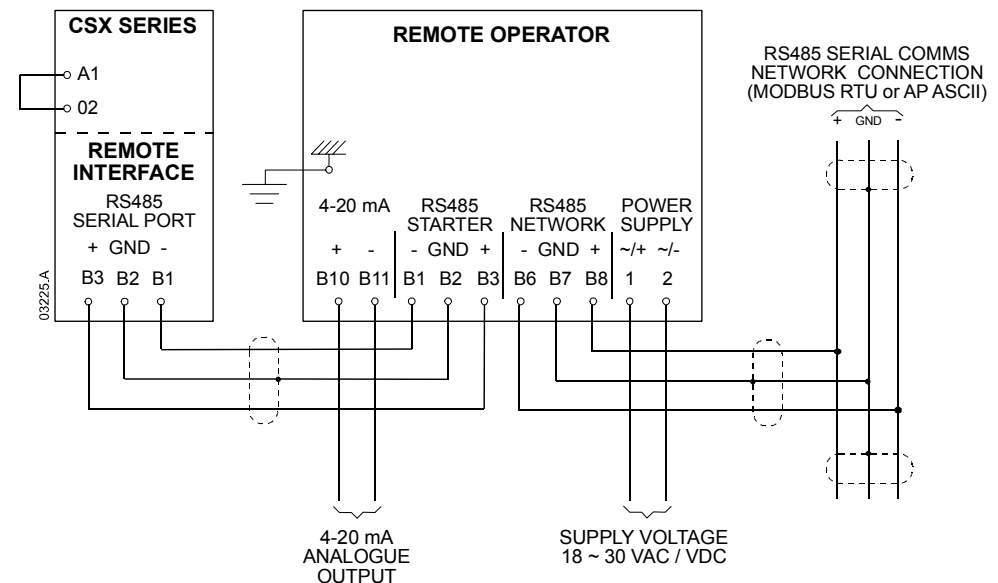
**RS485 Data Cable Connection**

Daisy chain connection is recommended. This is achieved by parallel connections of the data cable at the actual device terminals.

**Remote Operator RS485 Network Connection Specifications**

- Input impedance: 12 kΩ
- Common mode voltage range: - 7 V to + 12 V
- Input sensitivity: ± 200 mV
- Minimum differential output voltage: 1.5 V (with max loading of 54 Ω)

**Using the Remote Operator with CSX**

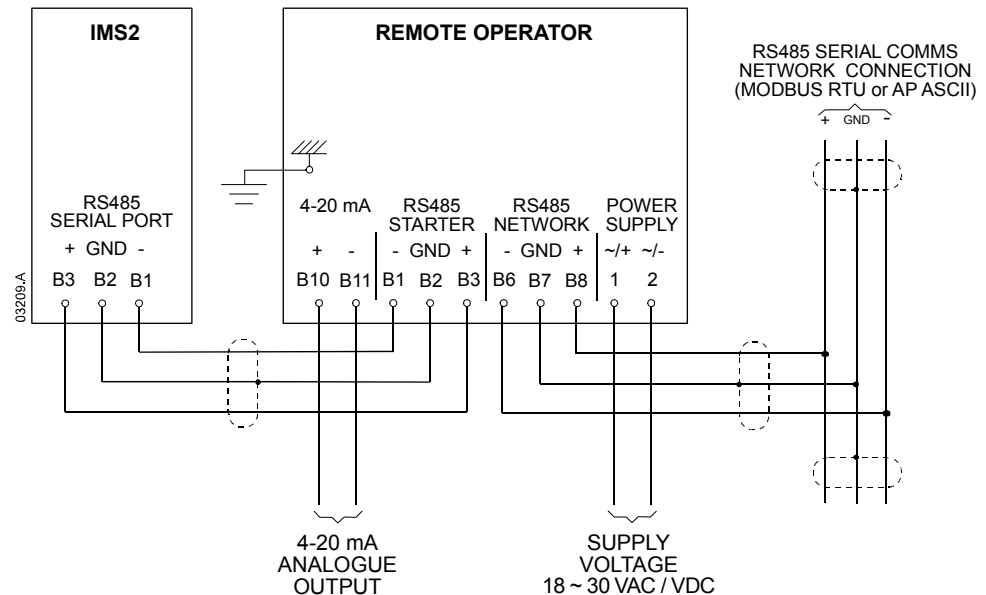




## Using the Remote Operator with IMS2

In order to operate correctly on the network, the IMS2 must be set for local operation only (ie set Parameter 20 = 2). The Remote Operator's default communications protocol setting is AP ASCII.

The RS485 Network Timeout setting on the Remote Operator applies to the link between the Remote Operator and the network. This can be set to any value between 0 and 100 seconds.



The Serial Timeout setting on the IMS2 (Parameter 60) applies to communications between the Remote Operator and the IMS2. Refer to the IMS2 Users Manual for soft starter configuration details.

## Programming

The Remote Operator must be configured to operate on the network. In order to access Programming Mode, the Remote Operator must be powered up when the soft starter is not running.

### Programming Procedure

1. To enter Programming Mode, hold down the Data/Prog pushbutton for four seconds. The default value of the first parameter will be displayed.
2. Use the Data/Prog pushbutton to advance to the next parameter.
3. Use the Stop/+ and Reset/- pushbuttons to adjust parameter values.

Programming Mode closes when the Data/Prog pushbutton is pressed after Parameter 8.



### NOTE

There is a 20 second timeout when the Remote Operator is in Programming Mode. Programming Mode will automatically close if no input is registered for 20 seconds. Any changes already made will be saved.

## Programmable Parameters

The Remote Operator offers the following programmable parameters:

| Parameter Number | Description                              | Default Setting   | Adjustable Range   |
|------------------|--|-------------------|--|
| 1                | RS485 network baud rate                  | 4<br>(9600 baud)  | 2 = 2400 baud<br>3 = 4800 baud<br>4 = 9600 baud<br>5 = 19200 baud<br>6 = 38400 baud  |
| 2                | RS485 network satellite address          | 20                | 1 to 99  |
| 3                | RS485 network timeout                    | 0 seconds (= off) | 0 to 100 seconds   |
| 4                | RS485 network protocol                   | 1<br>(AP ASCII)   | 1 = AP ASCII protocol<br>2 = Modbus RTU protocol   |
| 5                | Modbus protocol parity                   | 0<br>(no parity)  | 0 = no parity<br>1 = odd parity<br>2 = even parity<br>3 = 10-bit transmission  |
| 6                | Motor FLC (A)                            | 10                | 1 to 2868  |
| 7                | Analogue output 4 mA offset (%)          | 100               | 80 to 120  |
| 8                | Start, Stop, Quick stop function disable | 0                 | 0 = Remote Operator and Network start, stop, quick stop function enabled.<br>1 = Remote Operator start, stop, quick stop function enabled. Network start, stop, quick stop function disabled. <sup>2</sup><br>2 = Remote Operator start, stop, quick stop function disabled. Network start, stop, quick stop function enabled. <sup>1</sup><br>3 = Remote Operator start, stop, quick stop function disabled. Network start, stop, quick stop function disabled. <sup>1, 2</sup> |

<sup>1</sup> Remote Operator Reset/- pushbutton is always enabled.

<sup>2</sup> RS485 Network reset and forced communication trip functions are always enabled.

**Troubleshooting**

The Remote Operator display and status indication LEDs can indicate abnormal operating and system conditions. The following messages indicate an error in the Remote Operator's link to the RS485 network:

| Display Indication     | Problem   | Possible Solution   |
|------------------------|---|---|
| nEt on display         | A loss of communication has been detected on the RS485 link to the network. | The Remote Operator has an RS485 Network Timeout Protection setting (Parameter 3). This error is reported when no communication occurs for longer than the timeout setting. The system will become active as soon as communication is restored.<br>To clear nEt from the display, press the Data/Prog pushbutton momentarily or send a Reset command from the network Master. |
| SP flashing on display | Soft starter is off and being programmed from the serial network.           | Finish soft starter network programming procedure and exit Programming Mode.  |

11. Appendix B - Specifications

|  |  |
|--|--|
| <b>Enclosure</b>                           |  |
| Dimensions .....                           | 35 mm (W) x 157 mm (H) x 90 mm (D)                     |
| Weight .....                               | 250 g  |
| Protection .....                           | IP20   |
| <b>Mounting</b>                            |  |
| Spring-action plastic mounting clips (x 2) |  |
| <b>Connections</b>                         |  |
| Soft starter .....                         | 6-way pin assembly                                     |
| Network .....                              | 5-way male and unpluggable female connector (supplied) |
| Maximum cable size .....                   | 2.5 mm <sup>2</sup>                                    |
| <b>Settings</b>                            |  |
| Protocol .....                             | Modbus RTU, AP ASCII                                   |
| Address range .....                        | 0 to 31  |
| Data rate (bps) .....                      | 4800, 9600, 19200, 38400                               |
| Parity .....                               | None, Odd, Even, 10-bit                                |
| Timeout .....                              | None (off), 10 s, 60 s, 100 s                          |
| <b>Certification</b>                       |  |
| C✓ .....                                   | IEC 60947-4-2  |
| CE .....                                   | IEC 60947-4-2  |