

# Brainboxes Ethernet to Digital IO



## Contents

|       |   |    |
|-------|---|----|
| 1     | Introduction .....                                | 7  |
| 2     | Before you start.....                             | 8  |
| 2.1   | Box Contents .....                                | 8  |
| 2.2   | Requirements .....                                | 8  |
|       | Supported Operating Systems.....                  | 8  |
|       | System Requirements.....                          | 9  |
|       | Network Requirements .....                        | 9  |
| 3     | Hardware Features .....                           | 10 |
| 3.1   | ED Range – Industrial.....                        | 10 |
| 3.2   | ED Range – Light Industrial.....                  | 11 |
| 3.3   | Usability Features .....                          | 11 |
| 3.4   | Technical Specifications .....                    | 11 |
|       | Network connection .....                          | 11 |
|       | Power .....                                       | 12 |
| 3.5   | Storage and Operating Environment Guidelines..... | 12 |
|       | LED indicators.....                               | 12 |
| 3.6   | Circuit Diagrams .....                            | 13 |
| 3.6.1 | ED-588 .....                                      | 13 |
| 3.6.2 | ED-516 .....                                      | 13 |
| 3.6.3 | ED-527 .....                                      | 14 |
| 3.6.4 | ED-538 .....                                      | 14 |
| 3.6.5 | ED-038 .....                                      | 15 |
| 3.6.6 | ED-008 .....                                      | 15 |
| 3.6.7 | ED-004 .....                                      | 16 |
| 3.6.8 | ED-204 .....                                      | 16 |
| 3.6.9 | ED-504 .....                                      | 17 |
|       | Pin Outs: .....                                   | 18 |
|       | • ED-588.....                                     | 18 |
|       | • ED-527.....                                     | 18 |
|       | • ED-516.....                                     | 18 |

|   |    |
|---|----|
| • ED-538.....   | 18 |
| • ED-504.....   | 18 |
| 3.7 Pinouts:.....   | 20 |
| 3.7.1 Terminal Block Connector Pinout (Light Industrial)..... | 21 |
| 3.8 RS-232 Standard.....                                      | 21 |
| 3.9 Device Dimensions.....                                    | 21 |
| 3.9.1 Industrial Devices.....                                 | 21 |
| 3.9.2 Light Industrial Devices.....                           | 22 |
| 3.10 Reset Button.....  | 22 |
| 3.10.1 Manual Reboot.....                                     | 22 |
| 3.10.2 Manual Hard Restore.....                               | 22 |
| 3.11 Input/Output Specification.....                          | 23 |
| 3.11.1 ED Range – Industrial.....                             | 23 |
| 3.11.2 ED Range – Light Industrial.....                       | 23 |
| 3.11.3 Relay Information (ED-538, ED-038 Only).....           | 24 |
| 3.11.4 ED-538.....  | 25 |
| 3.11.5 ED-038.....  | 25 |
| 3.11.6 Firewall Exceptions and Port Numbers.....              | 25 |
| 4 Getting Started.....  | 26 |
| 4.1 Connecting your ED device.....                            | 26 |
| 4.2 Configuring your device settings.....                     | 27 |
| 5 Boost.IO Manager.....                                       | 28 |
| 5.1 Introduction.....   | 28 |
| 5.2 Installing Boost.IO Manager.....                          | 28 |
| 5.3 Finding and Installing an ED device.....                  | 29 |
| 5.4 COM Port Settings.....                                    | 31 |
| 5.4.1 Advanced COM Port Settings.....                         | 33 |
| 5.4.2 TCP/IP Settings.....                                    | 33 |
| 5.5 Alternate Access Methods.....                             | 34 |
| 5.6 Adding a Device by IP Address.....                        | 34 |
| 5.7 Adding a Device by MAC Address.....                       | 36 |

|       |  |    |
|-------|--|----|
| 5.8   | IP Addressing .....                              | 38 |
| 5.9   | Rebooting Device .....                           | 38 |
| 5.10  | Restoring Factory Settings.....                  | 39 |
| 5.11  | Firmware Upgrade .....                           | 40 |
| 5.12  | Proxy Server Settings .....                      | 42 |
| 5.13  | Device Swapping .....                            | 42 |
| 5.14  | Adding a Remote Device Using Boost.IO.....       | 42 |
| 5.15  | Remote Access.....                               | 42 |
| 6     | Web Configuration Pages.....                     | 44 |
| 6.1   | Introduction .....                               | 44 |
| 6.2   | Home page .....                                  | 44 |
| 6.3   | Network page.....                                | 47 |
| 6.4   | Serial Port page.....                            | 48 |
| 6.5   | Protocol page.....                               | 49 |
| 6.5.1 | ASCII Protocol Settings.....                     | 50 |
| 6.5.2 | Serial Gateway .....                             | 51 |
| 6.6   | Console page.....                                | 52 |
| 6.7   | IO Lines page .....                              | 53 |
| 6.8   | Device Management page.....                      | 56 |
| 6.9   | Factory Default Settings .....                   | 56 |
| 7     | Configuration Tables.....                        | 57 |
| 7.1   | Baud Rate Settings (BB) .....                    | 57 |
| 7.2   | Data Format Settings (FF).....                   | 57 |
| 7.3   | Digital Input/Output Data Characters Format..... | 57 |
| 8     | ASCII Protocol .....                             | 58 |
| 8.1   | Introduction .....                               | 58 |
|       | Command Format.....                              | 58 |
|       | Response Format .....                            | 59 |
| 8.2   | Command List.....                                | 60 |
| 8.3   | %AANNTTCCFF.....                                 | 61 |
| 8.4   | #** .....  | 62 |
| 8.5   | #AA00DD.....                                     | 63 |
| 8.6   | #AA0ADD .....                                    | 64 |

|      |                                       |    |
|------|---------------------------------------|----|
| 8.7  | #AA0BDD .....                         | 65 |
| 8.8  | #AA1cDD .....                         | 66 |
| 8.9  | #AAAcDD.....                          | 67 |
| 8.10 | #AABcDD .....                         | 68 |
| 8.11 | #AAN.....                             | 69 |
| 8.12 | \$AA2 .....                           | 70 |
| 8.13 | \$AA4 .....                           | 71 |
| 8.14 | \$AA5 .....                           | 72 |
| 8.15 | \$AA6 .....                           | 73 |
| 8.16 | \$AAC.....                            | 74 |
| 8.17 | \$AACN .....                          | 75 |
| 8.18 | \$AAF .....                           | 76 |
| 8.19 | \$AALS .....                          | 77 |
| 8.20 | \$AAM .....                           | 78 |
| 8.21 | \$AAS1 .....                          | 79 |
| 8.22 | \$AARS .....                          | 80 |
| 8.23 | @AA.....                              | 81 |
| 8.24 | @AA(Data).....                        | 82 |
| 8.25 | ~AAO(Name) .....                      | 83 |
| 8.26 | ~** .....                             | 84 |
| 8.27 | ~AA0 .....                            | 85 |
| 8.28 | ~AA1 .....                            | 86 |
| 8.29 | ~AA2 .....                            | 87 |
| 8.30 | ~AA3EVV .....                         | 88 |
| 8.31 | ~AA4V .....                           | 89 |
| 8.32 | ~AA5V.....                            | 90 |
| 9    | Worked Hardware Examples.....         | 91 |
| 9.1  | Output to a Light Circuit.....        | 91 |
| 9.2  | Switch Input Circuit.....             | 91 |
| 9.3  | Digital Outputs: SINK.....            | 91 |
| 9.4  | Digital Outputs: TTL Compatible ..... | 92 |
| 9.5  | Turning ON Indicator Light .....      | 92 |
| 9.6  | Turning ON Indicator Sounds.....      | 93 |

|      |   |    |
|------|---|----|
| 9.7  | Digital Inputs: NPN/PNP Compatible.....                         | 93 |
| 9.8  | Digital Input/Output.....                                       | 94 |
| 9.9  | Dry Contacts.....   | 95 |
| 9.10 | Detecting Dry Contacts.....                                     | 96 |
| 9.11 | Dry Contact: PIR Detector.....                                  | 96 |
| 9.12 | Sensors & Actuators.....  | 96 |
| 10   | Lifetime Warranty and Support.....                              | 97 |
| 11   | Regulatory Approvals / Compliance.....                          | 98 |
| 11.1 | Company Accreditation.....                                      | 98 |
| 11.2 | Europe – EU Declaration of Conformity.....                      | 98 |
| 11.3 | WEEE Directive (Waste Electrical and Electronic Equipment)..... | 98 |
| 11.4 | RoHS Compliance.....  | 99 |
| 12   | Copyright.....  | 99 |

# 1 Introduction

The Brainboxes Ethernet to DIO products are a range of Ethernet devices controlled by a host computer which provide digital communication functions using Input, Output and Relay lines. The ED product range has Input and Output lines which are capable of driving high current and high voltage loads and are specifically designed for use in industrial process control and automation environments.

The ED range is designed and built on-site with close collaboration between the hardware and software teams to deliver a product that is both user-friendly and has plenty of functionality.



Fig 1.1. ED Range - Industrial



Fig 1.2 ED Range – Light Industrial

## 2 Before you start

### 2.1 Box Contents

The following items are included with your product:

- Ethernet Digital Input and/or Output/Relay Device (ED)
- Boost.IO Installation CD including manual, Microsoft signed drivers and utilities
- Quick Start Guide

Optional Items:

- Global mains power supply unit - PW-600
- USB port power supply – PW-650 – Ideal for powering your ED device from anywhere using your laptop

If any of the items are missing from your box or damaged in anyway please contact [support@Brainboxes.com](mailto:support@Brainboxes.com)

### 2.2 Requirements

#### Supported Operating Systems

The Ethernet to DIO product range can be used in the following Microsoft Operating Systems with the supplied Boost.IO drivers:

- Windows Server 2012
- Windows 8.1 32-bit
- Windows 8.1 64-bit
- Windows 8 32-bit
- Windows 8 64-bit
- Windows 2008 R2
- Windows 7 32-bit
- Windows 7 64-bit
- Windows Server 2008 32-bit
- Windows Server 2008 64-bit
- Windows Vista 32-bit
- Windows Vista 64-bit
- Windows Server 2003 32-bit
- Windows Server 2003 64-bit
- Windows XP 32-bit
- Windows XP 64-bit
- Windows 2000



Brainboxes Boost.IO drivers have undergone extensive Microsoft testing with the ED range. Upon passing these tests, the drivers were signed by Microsoft, as an indication of their quality and stability.

## System Requirements

Components: Microsoft .NET Framework 2.0 (installed automatically with Boost.IO package)

Windows Installer: Windows Installer 3.1 or later (Recommended)

Internet Explorer: If you are running Internet Explorer, then Internet Explorer 7.0 or later is required

Processor: 400 MHz Pentium processor or equivalent (Minimum); 1GHz Pentium processor or equivalent (Recommended)

RAM: 96 MB (Minimum); 256 MB (Recommended)

Hard Disk: Up to 500 MB of available space may be required

CD or DVD Drive: Not required

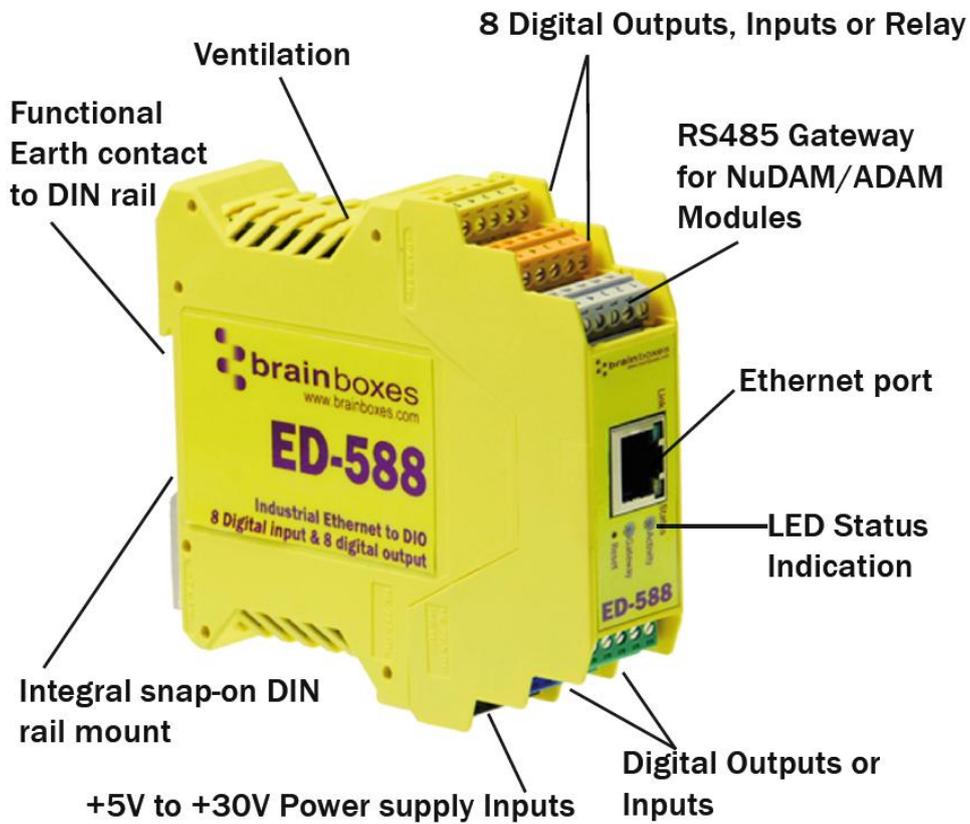
Display: 800 x 600, 256 colours (Minimum); 1024 x 768 high colour, 32-bit (Recommended)

## Network Requirements

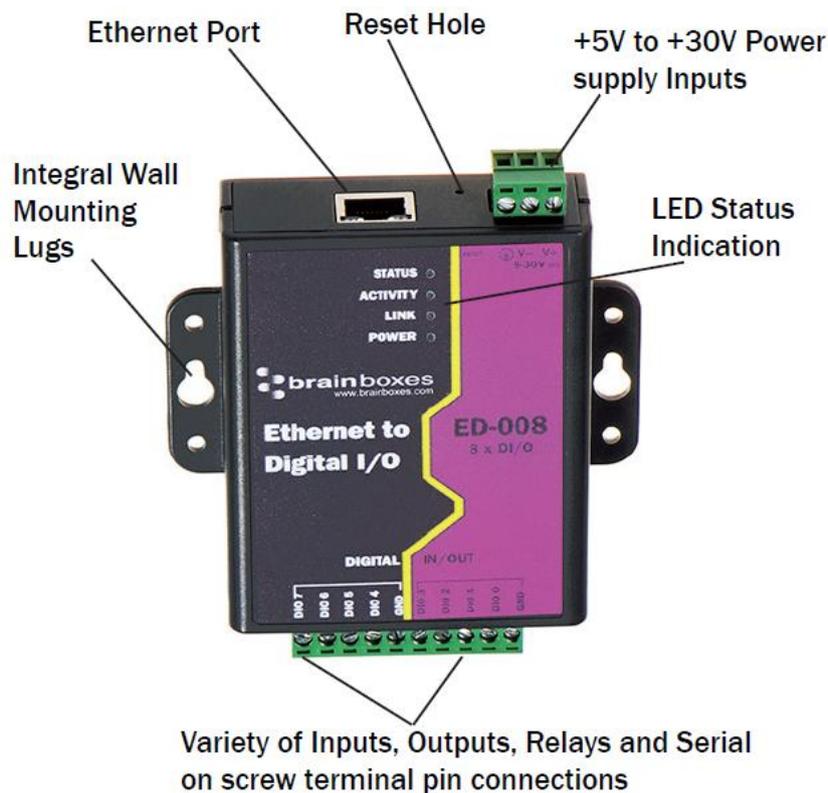
Ethernet network connection (wired).

## 3 Hardware Features

### 3.1 ED Range – Industrial



### 3.2 ED Range – Light Industrial



### 3.3 Usability Features

The ED range has been designed so that it is as easy as possible to wire up the input, output, relay, serial gateway and power wires to the terminal blocks and to connect the network cable:

- Removable screw terminal blocks make installation easier and quicker.
- Colour coded terminal blocks and ports prevent an incorrect connection. (Industrial only)
- Individually numbered pins simplify the wiring and removes confusion. (Industrial only)
- Smart Ethernet which automatically detects the polarity of the Ethernet connection so either a straight through or crossover Ethernet cable can be used.
- Functional ground to DIN rail.
- Can use the 5 Volt power from any computer USB port via the optional cable accessory PW-650.

### 3.4 Technical Specifications

#### Network connection

- 10Base-T or 100Base-TX Ethernet connection
- Standard 8P8C ("RJ45") socket connector
- TCP/IP protocol stack
- DHCP or static IP address
- Automatic transmit/receive crossover detection
- 1500V magnetic isolation

## Power

- Wide-range +5 to +30V DC 60mA@24V 1.4W Typical 120mA@24V 2.9W Max
- Reverse voltage protected
- ESD and surge protected
- Earthing connection point



**Caution** - Do not attempt to operate this product with any other power supply/rating than that specified.

### 3.5 Storage and Operating Environment Guidelines

|                               | ED Range – Industrial  | ED Range – Light Industrial |
|-------------------------------|--|-----------------------------|
| <b>Operating Temperature:</b> | -30°C to +80°C   | 0°C to +60°C                |
| <b>Storage Temperature:</b>   | -40°C to +85°C   | -40°C to +85°C              |
| <b>Humidity:</b>              | 5% to 95% non-condensing   |                             |
| <b>Housing:</b>               | IP 30 rated non-conducting polyamide case with integrated DIN rail mount |                             |

### LED indicators

There are 4 LED's on the ED devices representing the status, gateway, Ethernet link and the activity. The table below lists these LEDs, the colours of them and the meaning of the colours.

|            |                                     |                            |
|------------|-------------------------------------|----------------------------|
| Status LED | Green                               | Device Ready               |
|            | Flashing Yellow                     | Changing Settings          |
|            | Flashing between Red & Green        | Querying IP                |
|            | Flashing Green and Red              | User performing Hard Reset |
|            | Flashing between Green & Red/Yellow | IP address diagnostic      |
|            | Flashing between Green & Yellow     | Initialization diagnostic  |
| Gateway    | Flashing Red                        | RS-485 Comms error         |
|            | Flashing Green                      | RS-485                     |
| Link LED   | Green light on                      | Network Link Established   |
|            | Flashing Green                      | Network Data RX/TX         |
| Activity   | Flashing Green                      | Output set / Input Read    |
|            | Flashing Red                        | Output overload            |

### 3.6 Circuit Diagrams

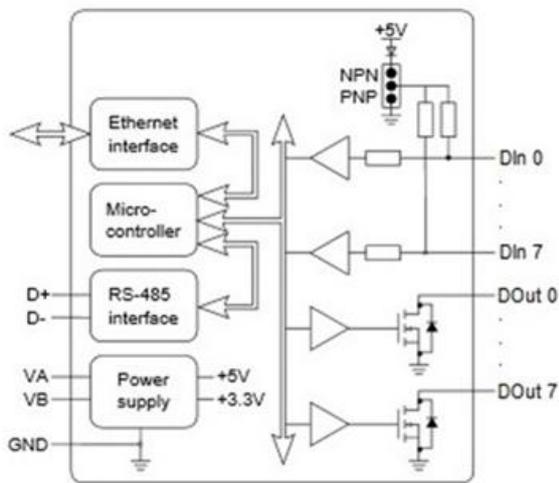
As long as you have the relevant jumper set to the 'NPN' position, you do not need an external power supply to use a contact switch. The NPN jumper position puts a pull-up resistor on the input so that you can just wire the switch between the input and ground.

**For ED-588, ED-527, ED-516, ED-504, ED-538:**

When in RS422/485 mode. There are biasing jumpers for Biasing (B+ and B-) and for Terminating Resistor (T).

These a factory default set in the ON position. (Biasing and Terminating Resistor Enabled). If these jumpers are in the PARK position, then Biasing and Terminating Resistor is Disabled.

#### 3.6.1 ED-588

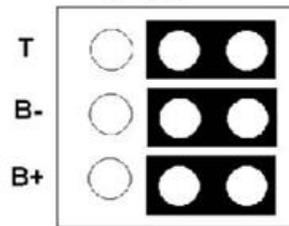


Default position

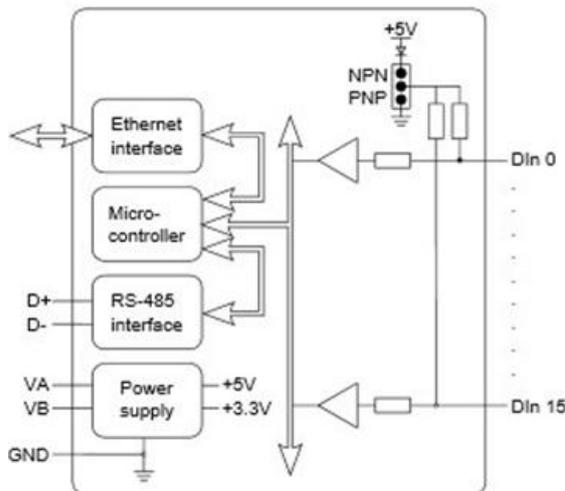


NPN or PNP jumper position should be changed depending on the type of sensor/input.

PARK



#### 3.6.2 ED-516

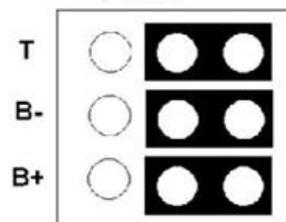


Default position

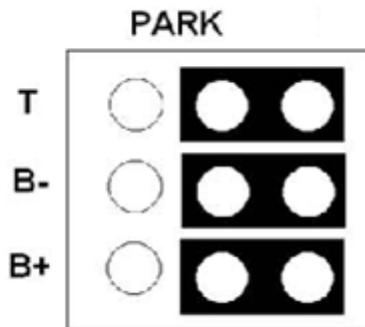
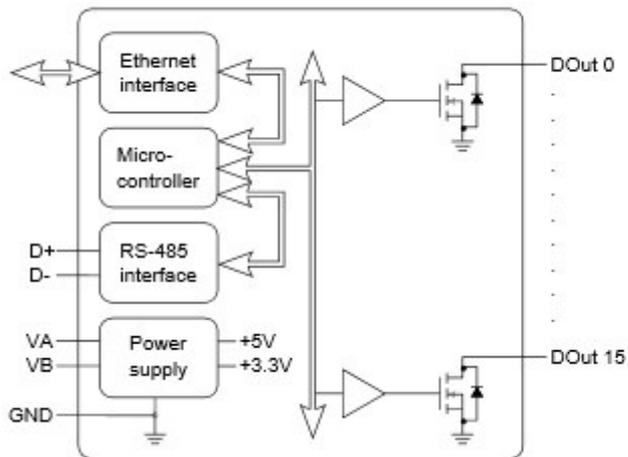


NPN or PNP jumper position should be changed depending on the type of sensor/input.

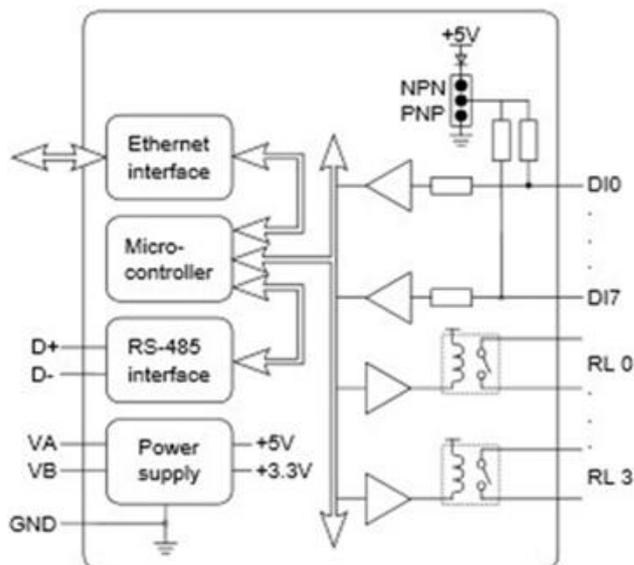
PARK



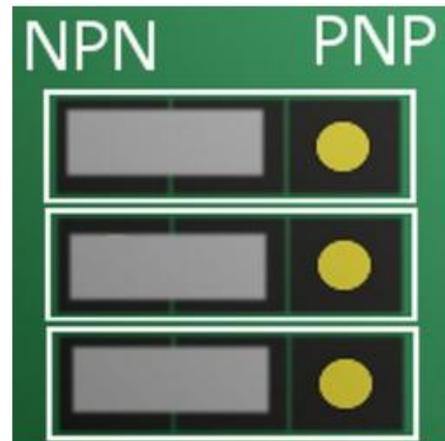
### 3.6.3 ED-527



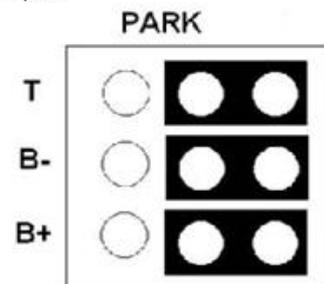
### 3.6.4 ED-538



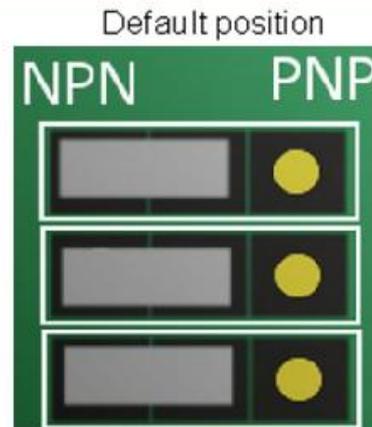
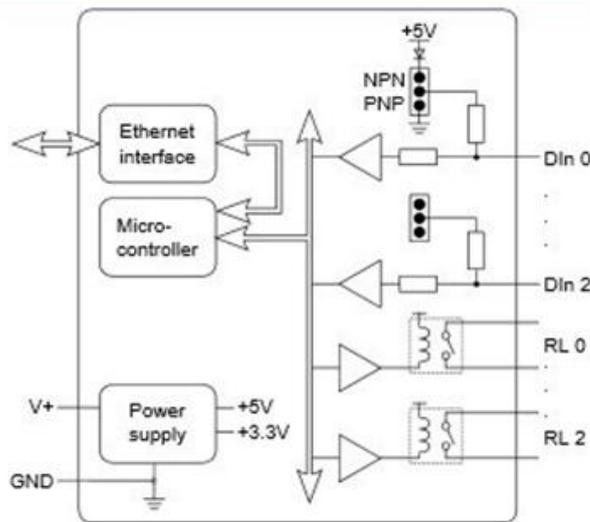
Default position



NPN or PNP jumper position should be changed depending on the type of sensor/input.



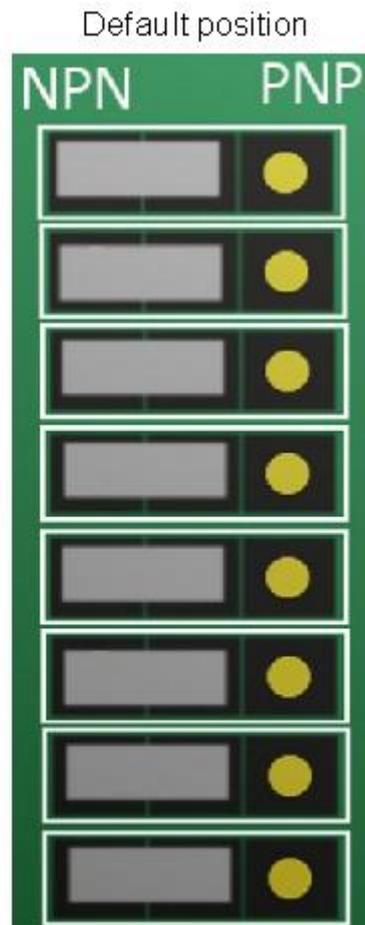
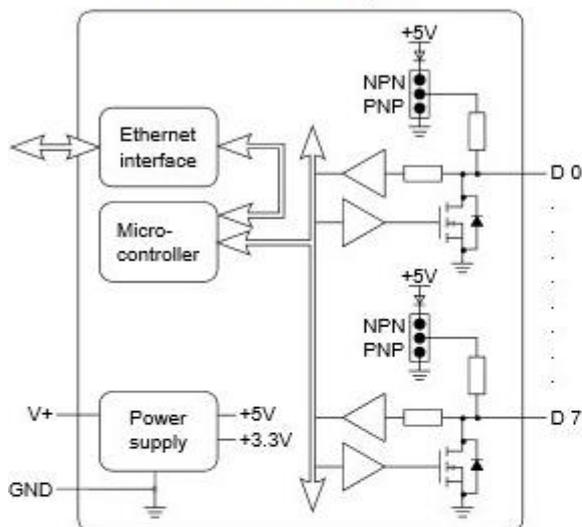
### 3.6.5 ED-038



NPN or PNP jumper position should be changed depending on the type of sensor/input.

### 3.6.6 ED-008

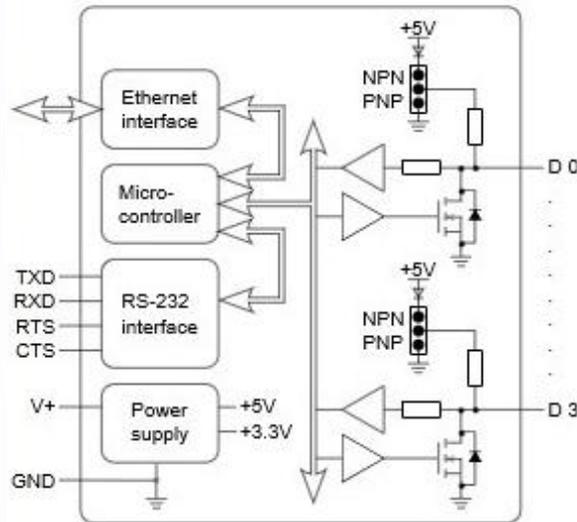
- NPN for NPN Input
- PNP for PNP Input
- Off for Output



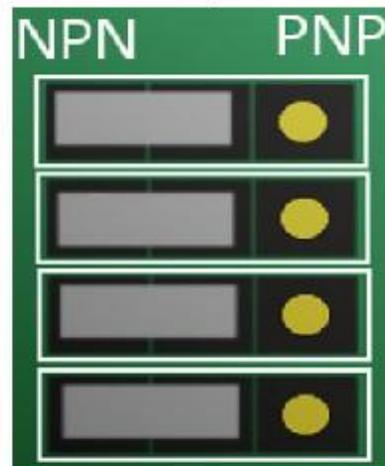
NPN or PNP jumper position should be changed depending on the type of sensor/input.

### 3.6.7 ED-004

- NPN for NPN Input
- PNP for PNP Input
- Off for Output



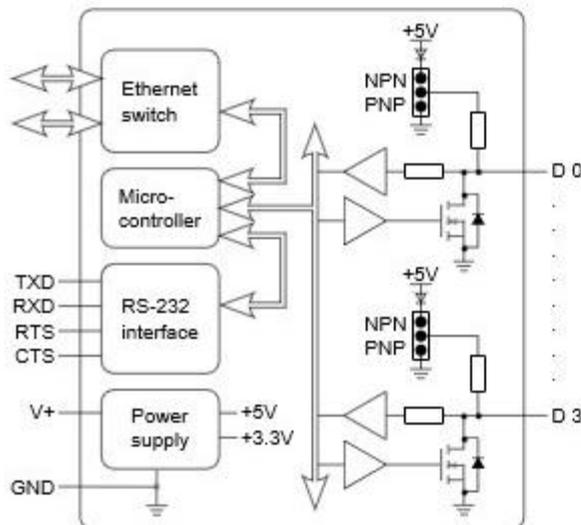
Default position



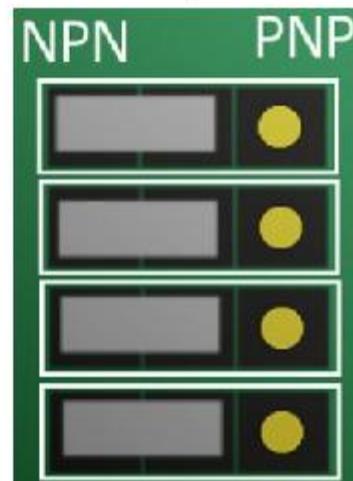
NPN or PNP jumper position should be changed depending on the type of sensor/input.

### 3.6.8 ED-204

- NPN for NPN Input
- PNP for PNP Input
- Off for Output



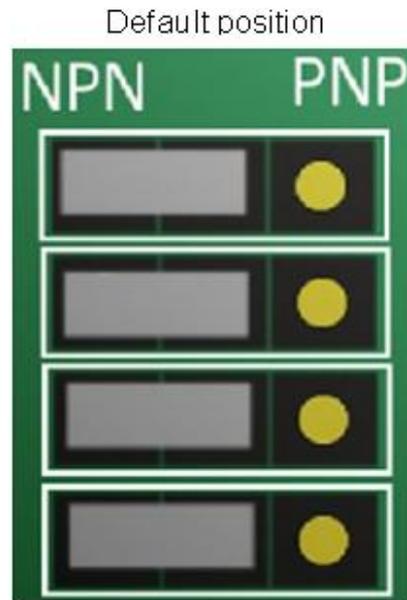
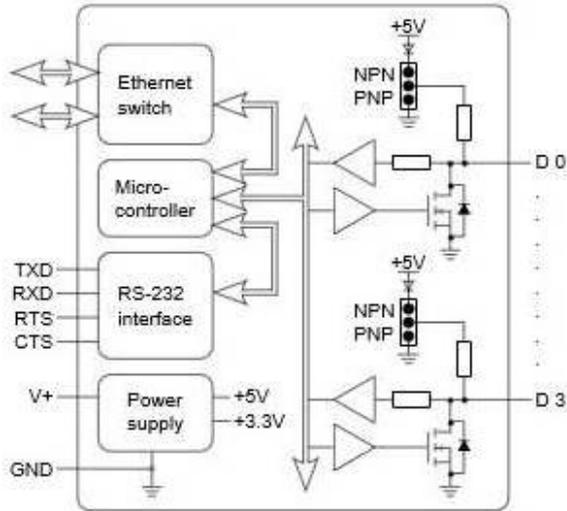
Default position



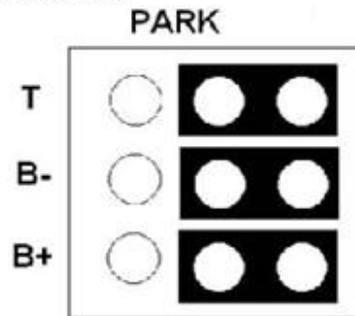
NPN or PNP jumper position should be changed depending on the type of sensor/input.

### 3.6.9 ED-504

- NPN for NPN Input
- PNP for PNP Input
- Off for Output



NPN or PNP jumper position should be changed depending on the type of sensor/input.



### Pin Outs:

- ED-588
- ED-527
- ED-516
- ED-538
- ED-504

**Functional Ground** is the EARTH connection through the DIN rail that the ED-5xx and ES-5xx is attached to. This Functional Ground (EARTH) on the ED-5xx is on PIN 5 of the BLACK power connector. It is directly connected to the FUNCTIONAL EARTH metal tab on the DIN rail clip. Thus the FUNCTIONAL EARTH electrically connects the metal DIN rail to PIN 5 of the BACK power Connector. The customer can choose to wire or not wire to PIN 5 on the Black power input.

**Power Ground** is the low side (0V) of the power input. The GND on the schematic is connected directly to PIN 1 and PIN 4 of the BLACK power input connector and is thus connected to the low side of the customers power supply.

**Signal Ground** is applicable to those ED/ES products which have a serial gateway or serial port. (ES-020/357/246/257/701/279/313/320/368/842 and ES-5xx range) This is the ground associated with the data transmission signals- it is typically connected to the cable sheath. It is import to connect this even in twisted pair situations.

### ED-588 : 8 digital outputs and 8 digital inputs

| Type of Port | Terminal Block | Pin 1     | Pin 2     | Pin 3     | Pin 4     | Pin 5     |
|--------------|----------------|-----------|-----------|-----------|-----------|-----------|
| Digital Out  | Yellow         | GND       | DOut 0    | DOut 1    | DOut 2    | DOut 3    |
| Digital Out  | Orange         | GND       | DOut 4    | DOut 5    | DOut 6    | DOut 7    |
| Gateway      | Grey           | GND       | RS-485 D- | RS-485 D+ | RS-485 D+ | RS-485 D- |
| Digital In   | Green          | GND       | DIn 0     | DIn 1     | DIn 2     | DIn 3     |
| Digital In   | Blue           | GND       | DIn 4     | DIn 5     | DIn 6     | DIn 7     |
| Power        | Black          | Power GND | +Vin A    | +Vin B    | Power GND | Func. GND |

### ED-527 : 16 digital outputs

| Type of Port | Terminal Block | Pin 1     | Pin 2     | Pin 3     | Pin 4     | Pin 5     |
|--------------|----------------|-----------|-----------|-----------|-----------|-----------|
| Digital Out  | Yellow         | GND       | DOut 0    | DOut 1    | DOut 2    | DOut 3    |
| Digital Out  | Orange         | GND       | DOut 4    | DOut 5    | DOut 6    | DOut 7    |
| Gateway      | Grey           | GND       | RS-485 D- | RS-485 D+ | RS-485 D+ | RS-485 D- |
| Digital In   | Green          | GND       | DIn 8     | DIn 9     | DIn 10    | DIn 11    |
| Digital In   | Blue           | GND       | DIn 12    | DIn 13    | DIn 14    | DIn 15    |
| Power        | Black          | Power GND | +Vin A    | +Vin B    | Power GND | Func. GND |

## ED-516

| Type of Port | Terminal Block | Pin 1     | Pin 2     | Pin 3     | Pin 4     | Pin 5     |
|--------------|----------------|-----------|-----------|-----------|-----------|-----------|
| Digital Out  | Yellow         | GND       | DIn 0     | DIn 1     | DIn 2     | DIn 3     |
| Digital Out  | Orange         | GND       | DIn 4     | DIn 5     | DIn 6     | DIn 7     |
| Gateway      | Grey           | GND       | RS-485 D- | RS-485 D+ | RS-485 D+ | RS-485 D- |
| Digital In   | Green          | GND       | DIn 8     | DIn 9     | DIn 10    | DIn 11    |
| Digital In   | Blue           | GND       | DIn 12    | DIn 13    | DIn 14    | DIn 15    |
| Power        | Black          | Power GND | +Vin A    | +Vin B    | Power GND | Func. GND |

## ED-538

| Type of Port | Terminal Block | Pin 1     | Pin 2     | Pin 3     | Pin 4     | Pin 5     |
|--------------|----------------|-----------|-----------|-----------|-----------|-----------|
| Relay Out    | Yellow         | RL 0 COM  | RL 0 N/O  | RL 1 COM  | RL 1 N/O  | -         |
| Relay Out    | Orange         | RL 2 COM  | RL 2 N/O  | RL 3 COM  | RL 3 N/O  | -         |
| Gateway      | Grey           | GND       | RS-485 D- | RS-485 D+ | RS-485 D+ | RS-485 D- |
| Digital In   | Green          | GND       | DIn 0     | DIn 1     | DIn 2     | DIn 3     |
| Digital In   | Blue           | GND       | DIn 4     | DIn 5     | DIn 6     | DIn 7     |
| Power        | Black          | Power GND | +Vin A    | +Vin B    | Power GND | Func. GND |

## ED-504

| PORT       | COLOUR | PIN 1 | PIN 2  | PIN 3  | PIN 4 | PIN 5    |
|------------|--------|-------|--------|--------|-------|----------|
| RS232      | YELLOW | GND   | CTS    | RXD    | RTS   | TXD      |
| RS422      | YELLOW | GND   | RXD-   | RXD+   | TXD+  | TXD-     |
| RS485      | YELLOW | GND   |        |        | DATA+ | DATA-    |
| DIGITAL IO | GREEN  | GND   | DIO 0  | DIO 1  | DIO 2 | DIO 3    |
| POWER      | BLACK  | GND   | +VIN A | +VIN B | GND   | FUNC GND |

### 3.7 Pinouts:

- ED-008,
- ED-038
- ED-204
- ED-004

#### ED-008

| Pin  | 1           | 2     | 3     | 4     | 5   | 6     | 7     | 8     | 9     | 10  |
|------|-------------|-------|-------|-------|-----|-------|-------|-------|-------|-----|
| Port | Digital I/O |       |       |       |     |       |       |       |       |     |
| Out  | DIO 7       | DIO 6 | DIO 5 | DIO 4 | GND | DIO 3 | DIO 2 | DIO 1 | DIO 0 | GND |

#### ED-038

| Pin  | 1            | 2    | 3     | 4    | 5     | 6    | 7          | 8     | 9     | 10  |
|------|--------------|------|-------|------|-------|------|------------|-------|-------|-----|
| Port | FORM A RELAY |      |       |      |       |      | Digital In |       |       |     |
| Out  | COM 3        | NO 3 | COM 2 | NO 2 | COM 1 | NO 1 | DIO 2      | DIO 1 | DIO 0 | GND |

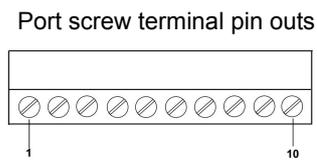
#### ED-204

| Pin  | 1     | 2   | 3   | 4   | 5   | 6           | 7     | 8     | 9     | 10  |
|------|-------|-----|-----|-----|-----|-------------|-------|-------|-------|-----|
| Port | RS232 |     |     |     |     | Digital I/O |       |       |       |     |
| Out  | TxD   | RxD | RTS | CTS | GND | DIO 3       | DIO 2 | DIO 1 | DIO 0 | GND |

#### ED-004

| Pin  | 1     | 2   | 3   | 4   | 5   | 6           | 7     | 8     | 9     | 10  |
|------|-------|-----|-----|-----|-----|-------------|-------|-------|-------|-----|
| Port | RS232 |     |     |     |     | Digital I/O |       |       |       |     |
| Out  | TxD   | RxD | RTS | CTS | GND | DIO 3       | DIO 2 | DIO 1 | DIO 0 | GND |

### 3.7.1 Terminal Block Connector Pinout (Light Industrial)

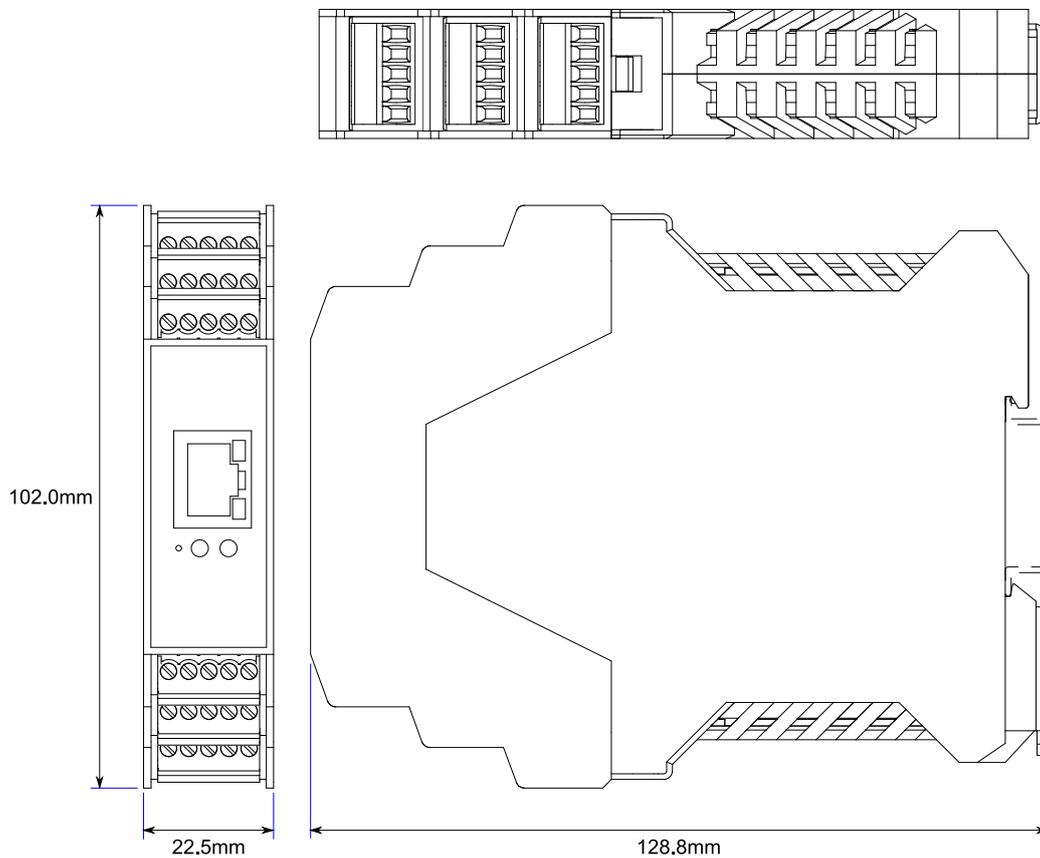


### 3.8 RS-232 Standard

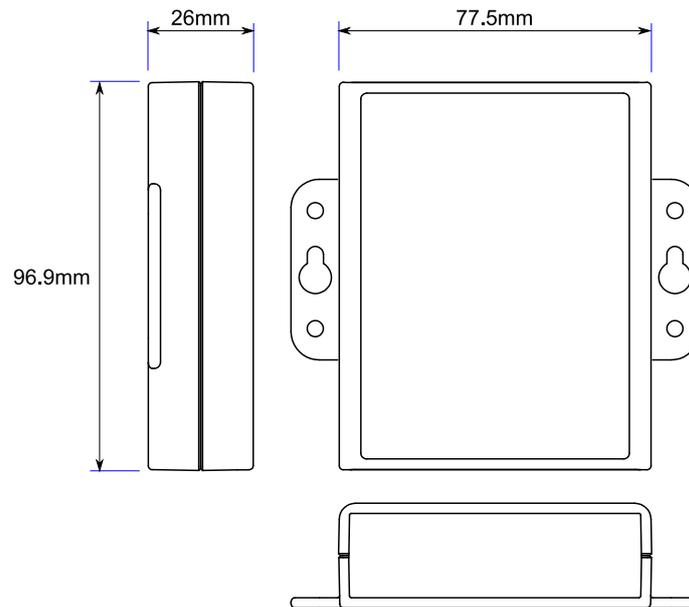
The RS-232 standard was introduced in 1962, it is now widely established. RS-232 is a slow speed, short distance, single ended transmission system (i.e. only one wire per signal). Typical RS-232 maximum cable length is 50 feet.

### 3.9 Device Dimensions

#### 3.9.1 Industrial Devices



### 3.9.2 Light Industrial Devices



### 3.10 Reset Button

#### 3.10.1 Manual Reboot

- 1) Press the reset button once.
- 2) The status LED will flash and after 5 seconds the device will reboot.
- 3) When the device is restarted, any connections you have had to the COM ports will need to be re-established.

#### 3.10.2 Manual Hard Restore

- 1) Press and hold the reset button on the device for 5 seconds.
- 2) The status LED will flash red/green and the device will be restored to factory default settings. For the factory settings, see **Factory Default Settings** section.

### 3.11 Input/Output Specification

#### 3.11.1 ED Range – Industrial

| Inputs: ED-588, ED-516, ED-538 |  |
|--------------------------------|--|
| NPN/PNP                        | One jumper configures all inputs as either pull up for NPN, active low, type sensors or pull down for PNP, active high, type sensors |
| Logic Level 0                  | 0V to +1V maximum  |
| Logic Level 1                  | +3.5V to 30V maximum   |
| Latched Inputs                 | Triggered by user programmable positive or negative edges, stays true until acknowledged   |
| Counter Inputs                 | User programmable – counts positive or negative transitions 0-65335  |

| Outputs: ED-588, ED-527     |   |
|-----------------------------|---|
| Maximum output current      | Sinks up to 1Amp per pin 40V max load<br>Max combined load 4.0 Amps per ED device   |
| Characteristic              | Open drain output, protected MosFET with intelligent short circuit protection up to 36V<br>Over temperature shutdown: 175°C typical 150°C min |
| Maximum output load Voltage | 40V   |
| ESD Protection              | 16kV  |

#### 3.11.2 ED Range – Light Industrial

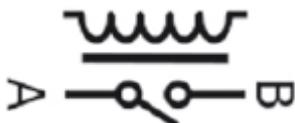
| Inputs: ED-008, ED-038 |  |
|------------------------|--|
| NPN/PNP                | One jumper configures each input as either pull up for NPN, active low, type sensors or pull down for PNP, active high, type sensors |
| Logic Level 0          | 0V to +1V maximum  |
| Logic Level 1          | +3.5V to 30V maximum   |
| Latched Inputs         | Triggered by user programmable positive or negative edges, stays true until acknowledged   |
| Counter Inputs         | User programmable – counts positive or negative transitions 0-65335  |

|                 |  |
|-----------------|--|
| Outputs: ED-008 |  |
| Maximum         | Sinks up to 0.85Amp per pin 30V max load |

|                             |   |
|-----------------------------|---|
| output current              | Max combined load 4.0 Amps per ED device                                      |
| Characteristic              | Open drain output, protected MosFET with intelligent short circuit protection |
| Maximum output load Voltage | 30V   |
| ESD Protection              | 16kV  |

### 3.11.3 Relay Information (ED-538, ED-038 Only)

A relay is an electrically operated switch used to control a circuit by a low power signal giving complete electrical isolation between the control and the controlled circuits. Relays are often used where several circuits must be controlled by one signal.

|                      |   |   |
|----------------------|---|---|
| <b>Relay Outputs</b> |   |  |
| Relay Type           | Form A (SPST-NO: Single Pole Single Throw, Normally Open)<br>When power is removed from the ED-538/ED-038, the relay is open    |   |
| Contact Rating       | 5A @ 30VDC<br>5A @ 250VAC   |   |
| Contact Resistance   | 100mΩ max. (at 1A 24VDC)  |   |
| Relay Endurance      | 100,000 operations at 3A, 30 operations/min, resistive AC load<br>50,000 operations at 5A, 20 operations/min, resistive AC load |   |

Each relay requires 2 terminal pins:

Pin 1 is COMMON and Pin 2 is NORMALLY OPEN.

The configuration and status of the relays can be viewed in the devices webpage.

### 3.11.4 ED-538

| Terminal Block | Pin 1    | Pin 2    | Pin 3    | Pin 4    | Pin 5 |
|----------------|----------|----------|----------|----------|-------|
| Yellow         | RL 0 COM | RL 0 N/O | RL 1 COM | RL 1 N/O | -     |
| Orange         | RL 2 COM | RL 2 N/O | RL 3 COM | RL 3 N/O | -     |

### 3.11.5 ED-038

| Pin 1   | Pin 2 | Pin 3   | Pin 4 | Pin 5   | Pin6 |
|---------|-------|---------|-------|---------|------|
| COM 3   | NO 3  | COM 2   | NO 2  | COM 1   | NO 1 |
| Relay 3 |       | Relay 2 |       | Relay 1 |      |

### 3.11.6 Firewall Exceptions and Port Numbers

When using the ED Devices with a firewall you may need to manually add the exception entries and port numbers to the firewall list.

Below are the default port numbers and the firewall exceptions.

| Program Name      | Default Port Number                                      |
|-------------------|--|
| Device Web Server | 80   |
| TCP Port          | 9500   |
| Firmware Upgrade  | 67 (BOOTP Server)<br>68 (BOOTP Client)<br>69 (TFTP Port) |

Default Windows Firewall Exception entries:

- Brainboxes Boost.IO Suite
- Brainboxes Boost.IO Suite (Device Discovery)  
(Except Windows XP32 & 64 bits)
- UPnP Framework (Windows XP32 & 64 bits)
- Network Discovery (Windows 7 or later)

## 4 Getting Started

### 4.1 Connecting your ED device

1. Connect the ED device to your local network using an Ethernet cable and plug into the Ethernet port connection on the device. The ED Ethernet port will automatically detect the polarity of the Ethernet connection so either a straight through or crossover Ethernet cables can be used.
2. Connect a suitable power supply to the ED removable power terminal block.

The optional PW-600 power supply allows the device to be plugged into a standard power socket. Alternatively, the device can be plugged into any USB using the optional PW-650 power cable. ED devices require a power supply providing a minimum of 1.1 watts with an output voltage between + 5VDC and +30VDC.

#### ED Industrial Range

| POWER TERMINAL PIN OUTS |     |                 |                 |     |       |
|-------------------------|-----|-----------------|-----------------|-----|-------|
| PIN                     | 1   | 2               | 3               | 4   | 5     |
| FUNCTION                | GND | +V <sub>A</sub> | +V <sub>B</sub> | GND | EARTH |

The ED device is fully operational when only one power supply is attached. When only one power supply is being used connect the positive terminal of the power supply to either +VA or +VB pin and the negative terminal to one of the GND pins.

Two separate sets of power supply inputs terminals are available; one to +VA and GND and the other to +VB and GND providing redundancy when both are connected. The higher voltage of the two power supplies is selected by the device and in the event of a failure the other supply automatically takes over to keep the device running. The status of the two power supply inputs can be monitored visually via a browser from the ED devices home page and also programmatically.

#### ED – Light Industrial

| POWER TERMINAL PIN OUTS |     |     |                   |
|-------------------------|-----|-----|-------------------|
| PIN                     | 1   | 2   | 3                 |
| FUNCTION                | V + | V - | Earth/Chassis GND |



3. On connecting to the network, the device automatically checks if it is connected to a DHCP Server. If this is the case, the DHCP server will allocate an IP address automatically to the ED device.
4. If no DHCP Server is detected (e.g. you have the ED device plugged directly into the PC), the ED device will default to an IP address of 192.168.127.254 after 60 seconds. If connecting directly to a PC, make sure the PC is on the same 192.168.127.xxx subnet in order to find your device.
5. The LED's will flash when the power is first applied. When the status LED turns a steady green (after 5-60 seconds) the device is ready to use.

Make a note of device MAC address (on the side of the ED device, 00-0A-4F-XX-XX) as you will need it to identify the device on your network later.

## 4.2 Configuring your device settings

There are three methods to view and configure your Ethernet to DIO device. Which method to use depends on personal preference and convenience.

- **Boost.IO Manager:** This is the Windows application that is installed initially to find the device and install the COM port drivers if required. Configuring from here is the recommended option for ease and convenience, as it centralises all Ethernet to DIO devices in one window. However, not all of the settings of the ED device can be configured from the Boost.IO Manager. See **Section 5** for information about Boost.IO Manager.
- **Web Page Interface:** This allows the Ethernet to DIO device to be accessed from any PC within your network as it does not require Boost.IO Manager. Unlike Boost.IO Manager, every setting can be configured using the web configuration pages. See **Section 6** for more information about the web page interface.
- **ASCII Protocol:** The ASCII protocol is a query-response communication protocol which can be used to set all of the protocol settings of the device, send and read data, and get status information from the devices. To send the commands and receive responses a connection to the device needs to be set up. This can be done using either a TCP connection or through the COM port.

## 5 Boost.IO Manager

### 5.1 Introduction

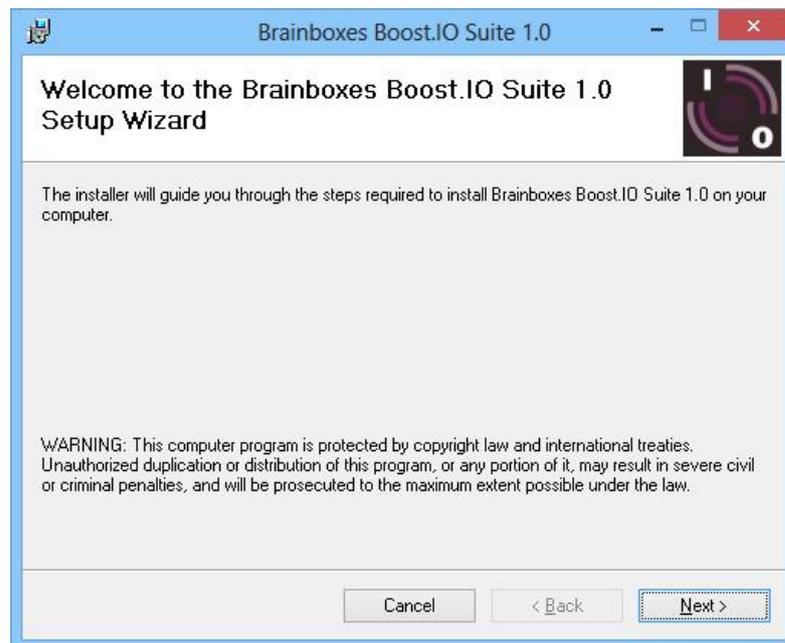
Boost.IO Manager is a graphical user interface application which will allow you to find your Brainboxes ED device on a network and then configure some of the settings in Windows. As part of the Boost.IO application a COM port can be installed which allows communication with the device using the ASCII protocol.

Boost.IO Manager is not required if you want to communicate with the device using just the IP address or if you want to configure the device only using the web configuration pages.

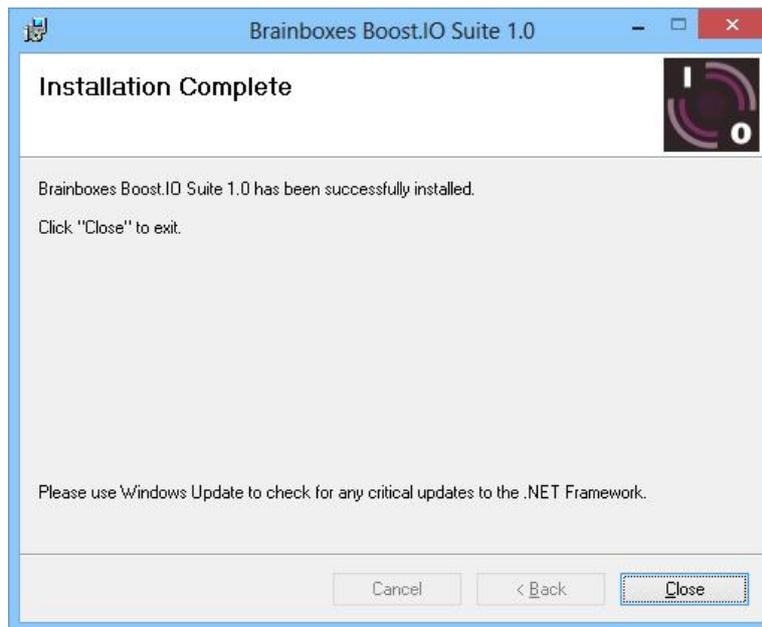
Boost.IO Manager can be found on the CD that was packaged with your device.

### 5.2 Installing Boost.IO Manager

1. Insert the CD into your PC
2. Browse the contents of the CD and locate the “Setup” program on the CD and double click to launch. Proceed to Step 3.
3. Click ‘Install’ to launch the Boost.IO setup program.
4. Follow the on screen instructions to install the Boost.IO application.



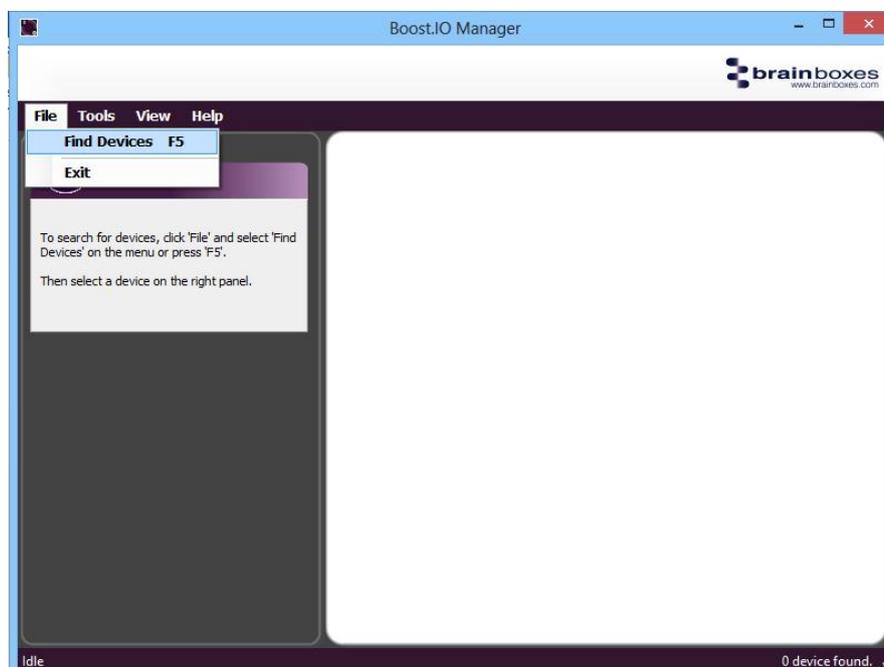
5. Once the installation process is complete you will see the setup wizard now says “Installation Complete” and there will be a Boost.IO Manager icon on your desktop.



### 5.3 Finding and Installing an ED device

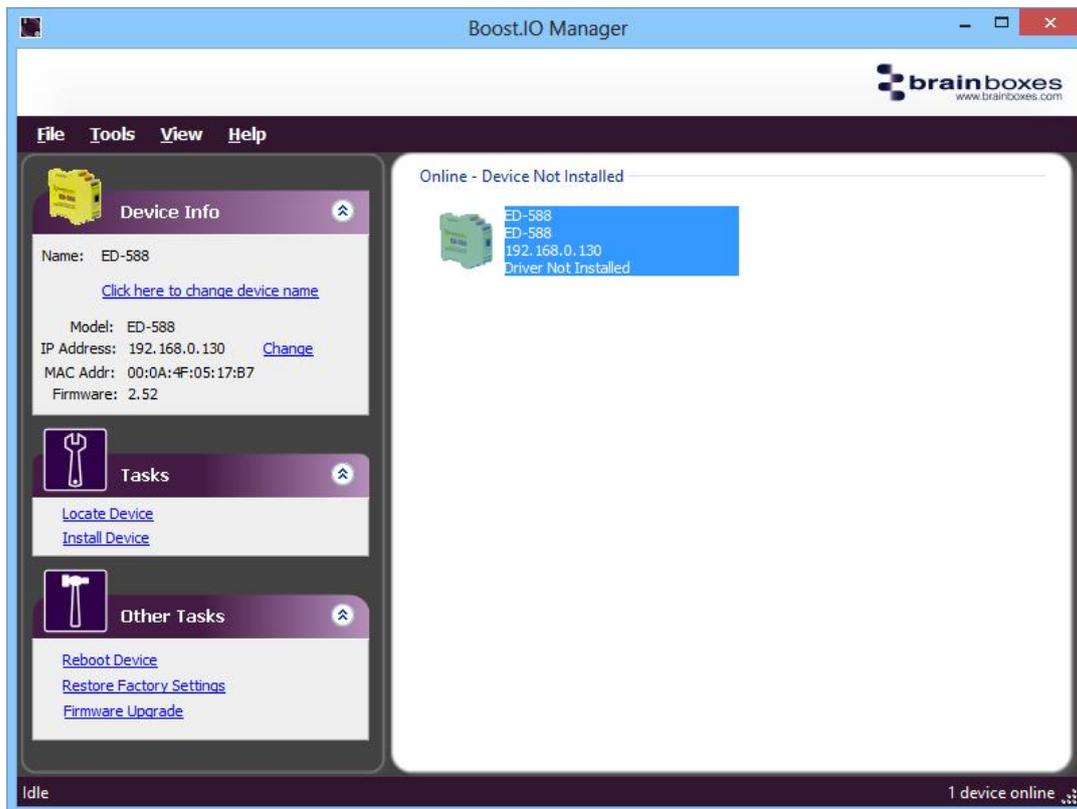
When using the ED device with Boost.IO Manager a COM port can be installed which allows the user to send ASCII commands to the device using a COM port connection.

1. Open Boost.IO Manager by double clicking the Boost.IO Manager icon on your desktop.
2. Click 'File' > 'Find Devices' to search for devices on the network (or press F5). This will find any ED devices on the same subnet as your PC. If your device is on a different subnet, please see the section entitled "Adding a Device by IP Address".

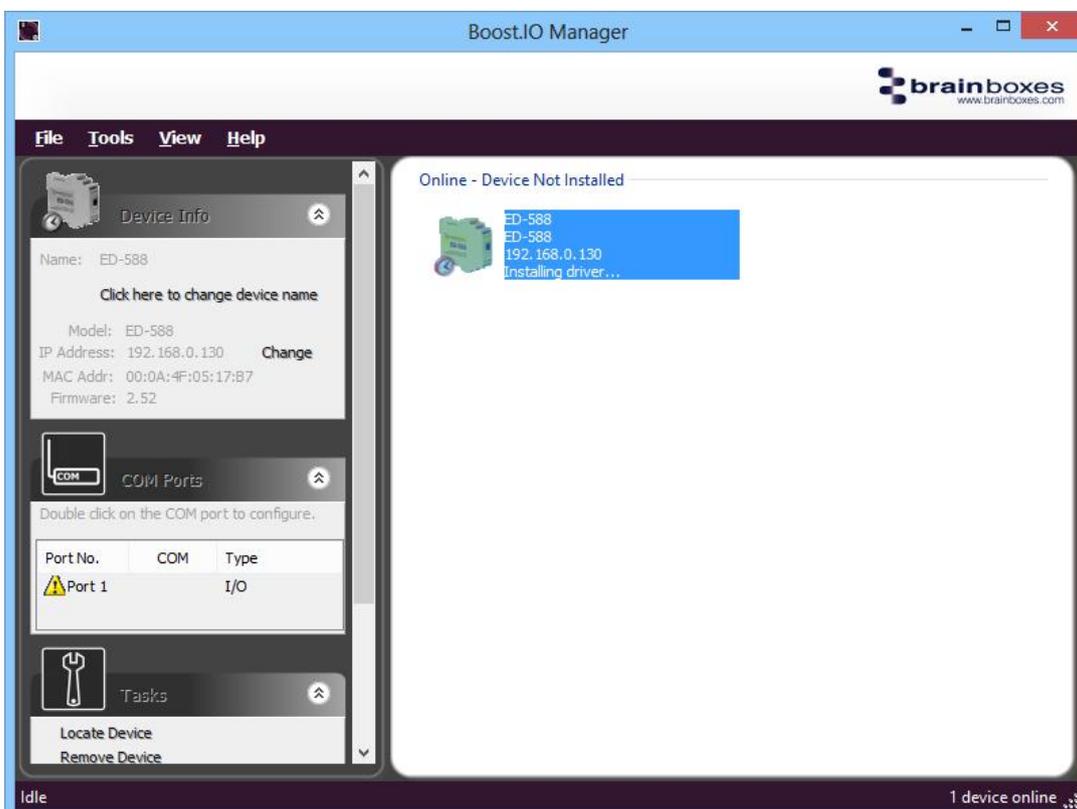


3. Once the search is complete, all the devices that have been found on the network will be displayed in the main window of Boost.IO Manager. When you select a device, the information and settings options for that device will appear in the left column.

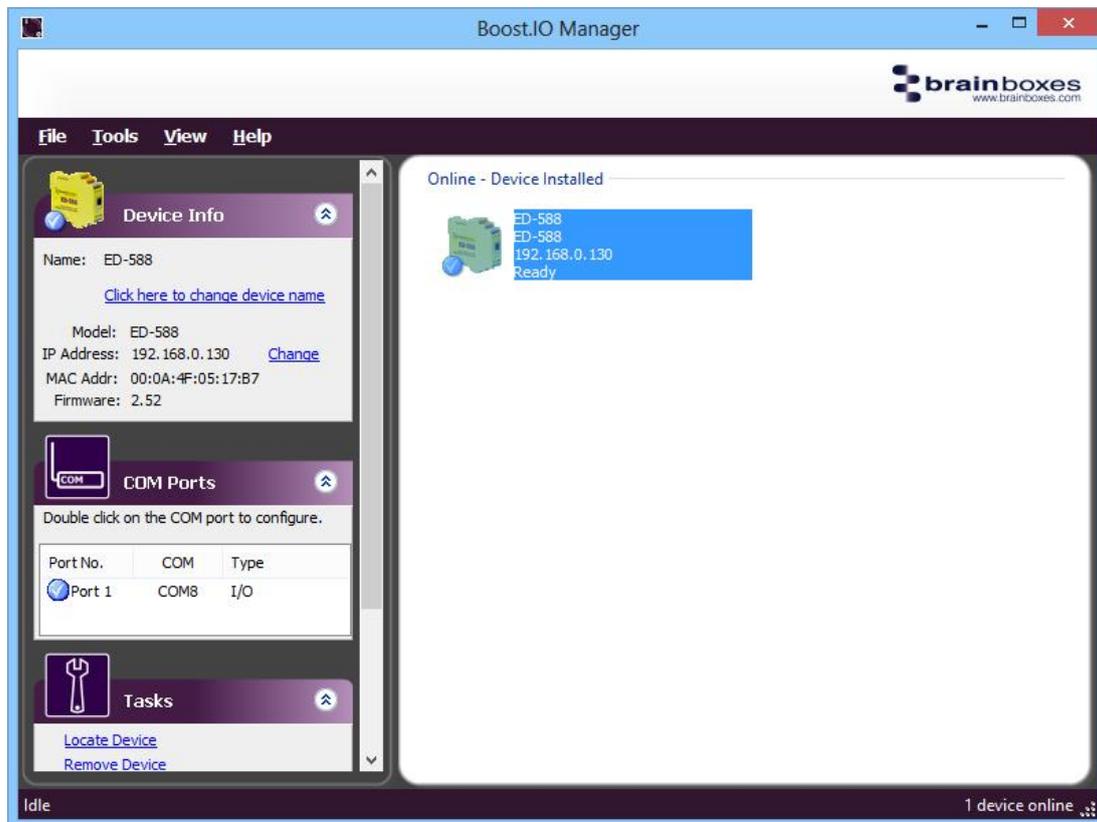
- You can identify your ED device by matching the MAC address in the left panel of Boost.IO Manager with the MAC address on the sticker of your ED device's case.



- When you have found your device in Boost.IO Manager, in the left panel under 'Tasks' click 'Install Device'.



- The driver for the device will be installed and, when complete, the icon in Boost.IO Manager will have a blue tick next to it and the status will be 'Ready'.

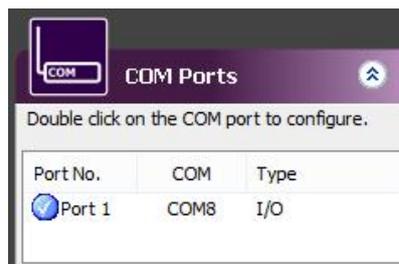


- There will now be a COM Ports section in the left panel which allows you to change the settings of the installed COM Port such as baud rate and COM label. For details on how to do this please see the COM Port Settings section.

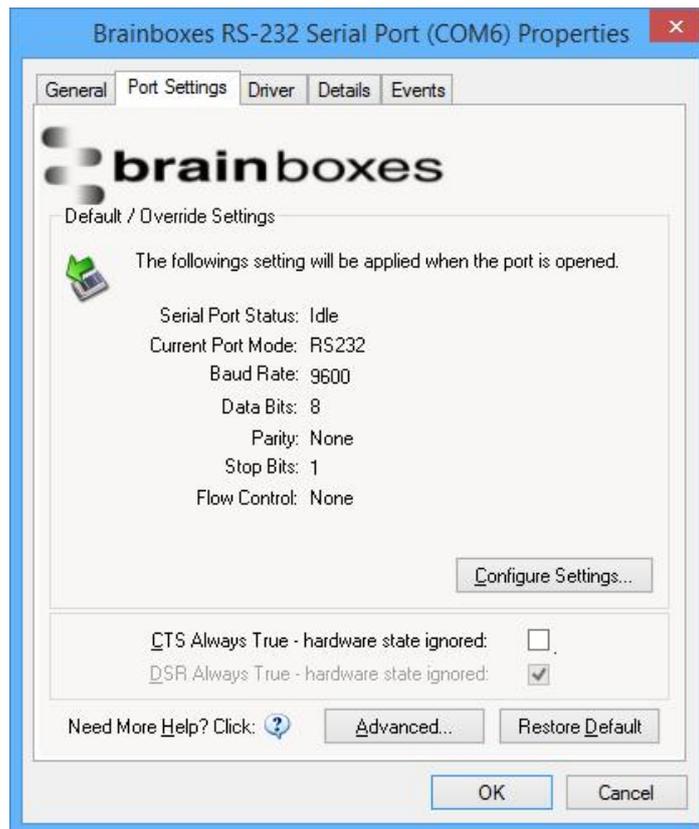
## 5.4 COM Port Settings

To change the COM Port settings of an installed ED device, follow the steps below.

- Open Boost.IO Manager, if it is not already open, by clicking the Boost.IO icon on your desktop.
- Select the installed device which you want to configure by clicking on it.
- In the left panel find the 'COM Ports' section and double click on the port which you want to configure.



- You will then see the port properties dialog box. Click on the 'Port Settings' tab, then click the 'Configure Settings.' button.



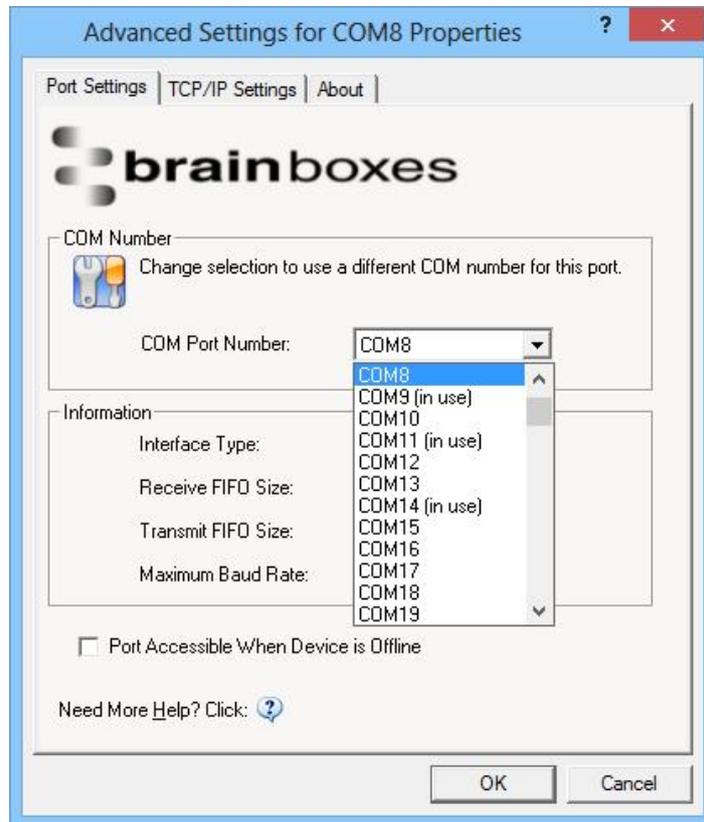
5. A webpage will be opened which allows you to configure all of the devices settings. For more information on configuring the device using the web configuration pages please see section [7. Web Configuration Pages](#)

### CTS Always True

- CTS is an incoming hardware handshaking line. This means it receives a signal from the connected device which tells the ED device when it is and isn't OK to send data out the RS232 port.
  - Sometimes these signals may want to be ignored. By forcing CTS True, the ED device will ignore those signals and always send data.
  - These settings are especially helpful when CTS is not physically connected (such as in a 3 wire setup) and it is not acceptable for the data flow to stop and start due to arbitrary variances on the unconnected signal lines.

### 5.4.1 Advanced COM Port Settings

From the Port Settings tab, you can get to the advanced settings for the COM port by clicking the 'Advanced...' button. This will open another dialog window allowing you to change the advanced settings for the COM port.



#### COM Number

The COM Port Number is changed using the drop down box on this page. Click the drop down box and select a COM Port Number from the list. If the COM Port number is labelled "*in use*", it is either currently used by a COM Port present on the system, or is reserved for a device which is not currently present. It is possible to select this COM number and force the change if you are sure it is not required by any other device. Click the 'OK' button to apply the changes.

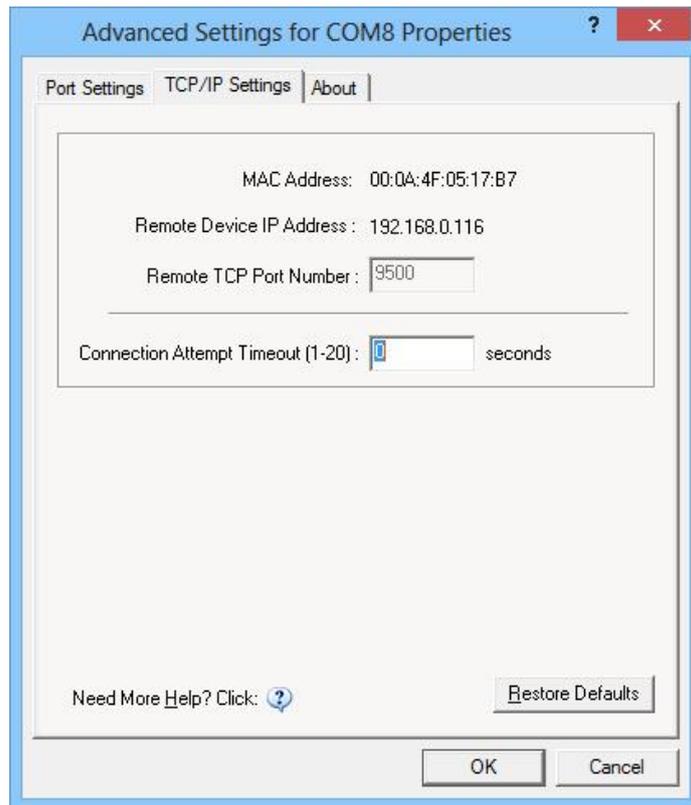
#### Port Accessible When Device is Offline

With ED devices being connected through a network, whenever a device is unplugged the device will go offline in Boost.IO Manager and any COM ports that are installed will become unavailable. By ticking this checkbox, it will keep the COM port available to use the port, even if the ED device is offline. This is useful if the device is being used in an application that requires a constant connection to a COM port.

### 5.4.2 TCP/IP Settings

Under the 'TCP/IP Settings' tab is information about the ED device. There is also a connection attempt timeout field. This value is the time that the driver will attempt to connect to the device COM port before giving up. For example, if this value is set at 5 seconds, the driver will attempt to connect to the COM port for 5 seconds. If no connection could be made after 5 seconds, then the driver will stop attempting to connect.

If this value is set to 0, the driver will attempt to connect without ever giving up.



## 5.5 Alternate Access Methods

There are three methods to view and configure your Ethernet Device once it's installed on the target PC. Which method to use depends on personal preference and convenience.

**NB:** Only certain settings can be configured using the Web Page Interface

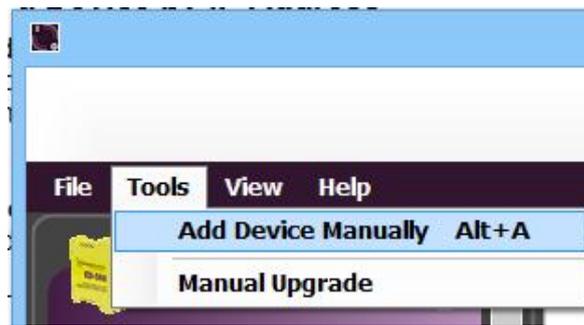
- **Boost.IO Manager:** This is the application that is installed initially to find the device and install the COM port drivers. Configuring from here is the recommended option for ease and convenience, as it centralises all ED devices.
- **Windows Device Manager:** This is the standard Windows Control Panel that allows users to view and control all hardware attached to a computer. The ED devices ports can be configured from here also.
- **Web Page Interface:** This allows the ED device to be accessed from any PC within your network as it does not require Boost.IO Manager. Configuration from here is recommended for socket based applications only.

## 5.6 Adding a Device by IP Address

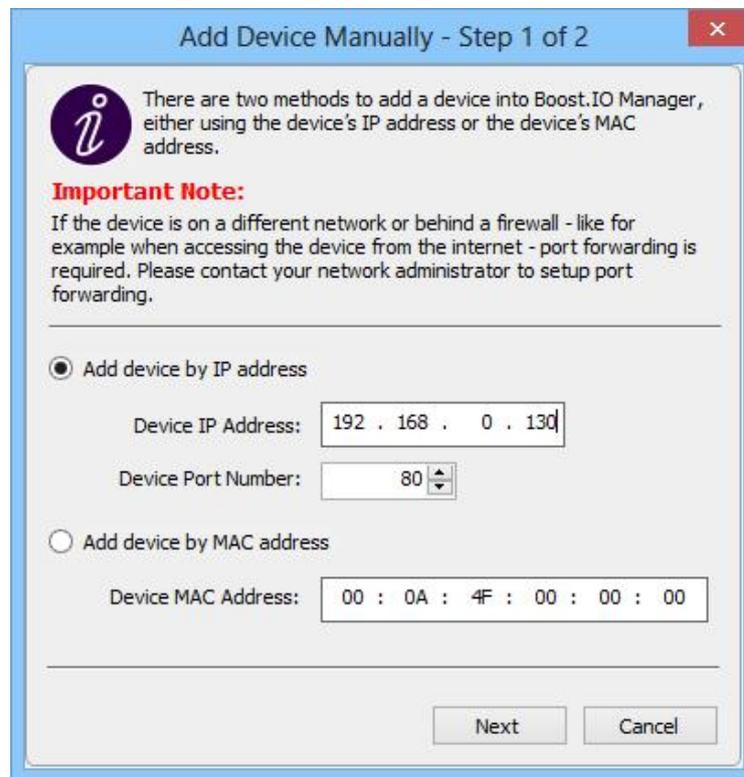
You may already know the IP address of the device that you are using. This might be because the device has had an IP address reserved for it by a system administrator or you are remotely connecting to the device. Using the IP address of the device, you can add it manually into Boost.IO Manager.

1. Open Boost.IO Manager by clicking the icon created on the desktop when the application was installed.

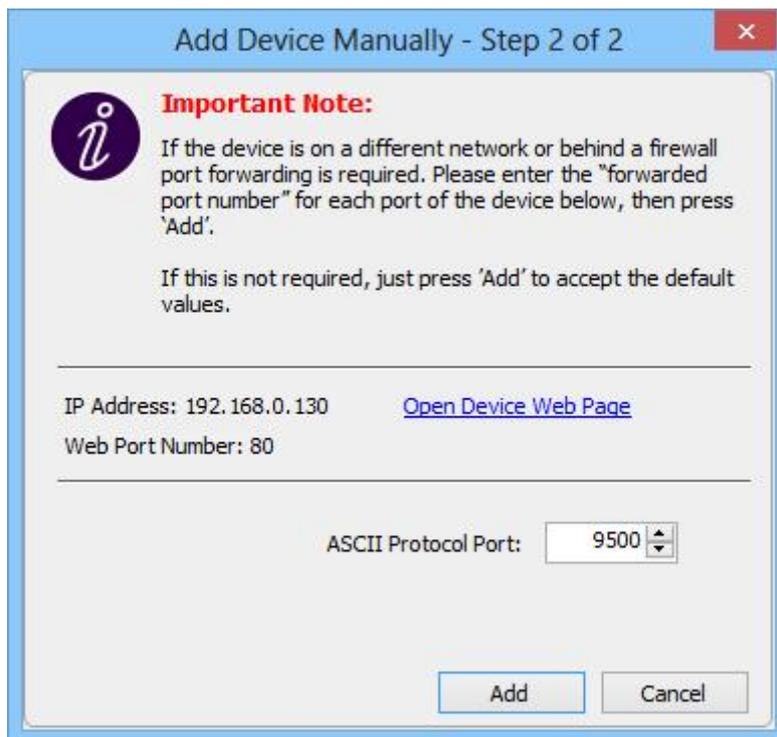
2. Select 'Tools' and then click 'Add Device Manually'.



3. Enter the IP address and port number of the device into the Device IP Address box then click the 'Next' button.

A screenshot of the 'Add Device Manually - Step 1 of 2' dialog box. The dialog has a blue title bar with a close button (X) in the top right corner. Inside, there is an information icon (i) and text explaining that there are two methods to add a device: by IP address or by MAC address. An 'Important Note' section states that port forwarding is required if the device is on a different network or behind a firewall. Below this, there are two radio button options: 'Add device by IP address' (which is selected) and 'Add device by MAC address'. Under the selected option, there are input fields for 'Device IP Address' (containing '192 . 168 . 0 . 130') and 'Device Port Number' (a spinner box set to '80'). Under the unselected option, there is an input field for 'Device MAC Address' (containing '00 : 0A : 4F : 00 : 00 : 00'). At the bottom right, there are 'Next' and 'Cancel' buttons.

4. Enter the ASCII Protocol Port number into box. The default Port Number is 9500.

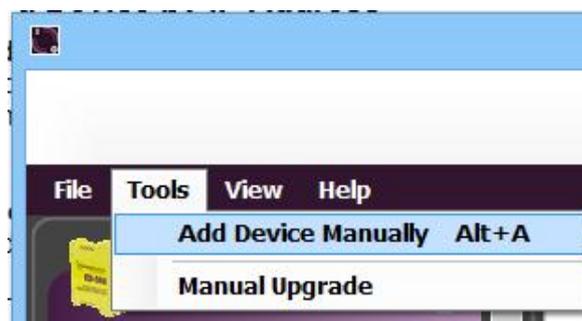


5. Click 'Add' and the device will be added to the Boost.IO Manager window.

## 5.7 Adding a Device by MAC Address

A device can be added into Boost.IO Manager by entering the MAC address. The MAC address of a device can be found on the sticker of the device. Follow the instructions below to add the device using the MAC address.

1. Open Boost.IO Manager by clicking the icon created on the desktop when the application was installed.
2. Select 'Tools' and then click 'Add Device Manually'.



3. Select the 'Add device by MAC address' radio button then enter the MAC address of your device.



**Add Device Manually - Step 1 of 2**

There are two methods to add a device into Boost.IO Manager, either using the device's IP address or the device's MAC address.

**Important Note:**  
If the device is on a different network or behind a firewall - like for example when accessing the device from the internet - port forwarding is required. Please contact your network administrator to setup port forwarding.

Add device by IP address

Device IP Address: 0 . 0 . 0 . 0

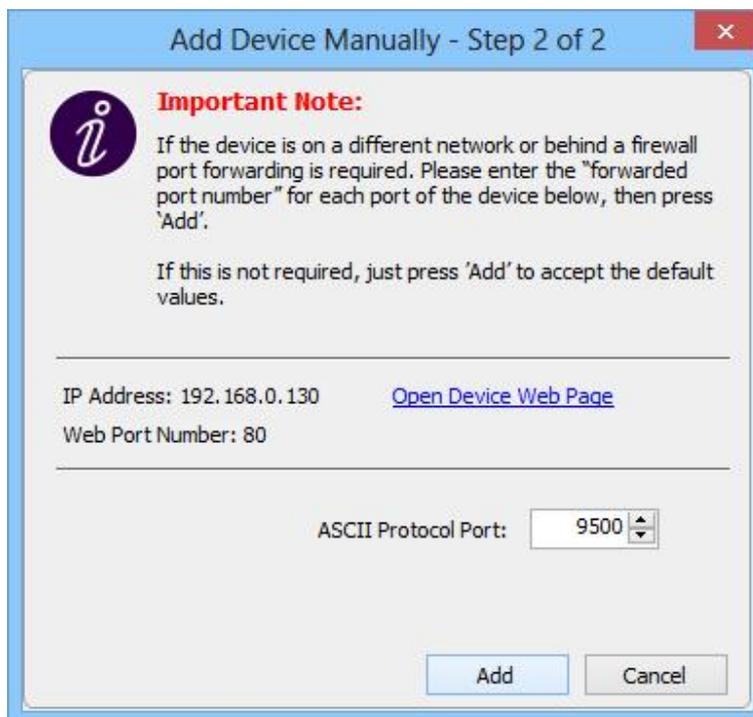
Device Port Number: 80

Add device by MAC address

Device MAC Address: 00 : 0A : 4F : 05 : 17 : B7

Next Cancel

4. Enter the MAC address of your device into the text box and click the 'Next' button.



**Add Device Manually - Step 2 of 2**

**Important Note:**  
If the device is on a different network or behind a firewall port forwarding is required. Please enter the "forwarded port number" for each port of the device below, then press 'Add'.  
If this is not required, just press 'Add' to accept the default values.

IP Address: 192.168.0.130 [Open Device Web Page](#)

Web Port Number: 80

ASCII Protocol Port: 9500

Add Cancel

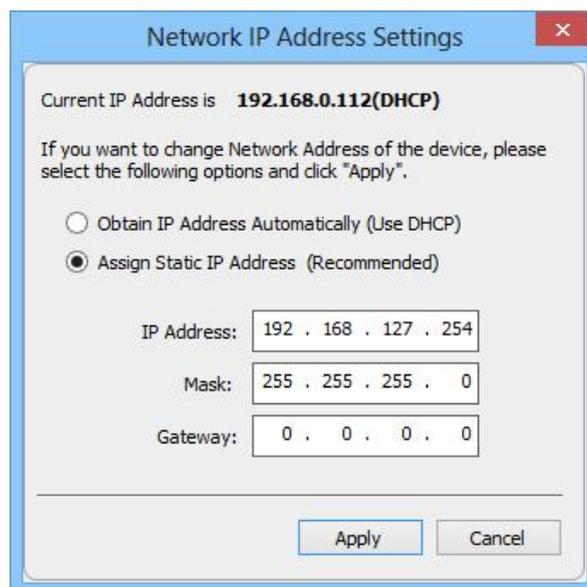
## 5.8 IP Addressing

The IP address settings can be easily changed using Boost.IO Manager.

1. Click on the device of which you want to change the settings.
2. In the 'Device Info' section on the left hand side, click the 'Change' link next to the current IP address.
3. Click 'OK' to confirm that you do want to change the IP address of your device



4. Select the 'Assign Static IP Address' radio button and enter an IP Address, Subnet Mask and Gateway Address then click 'Apply'.

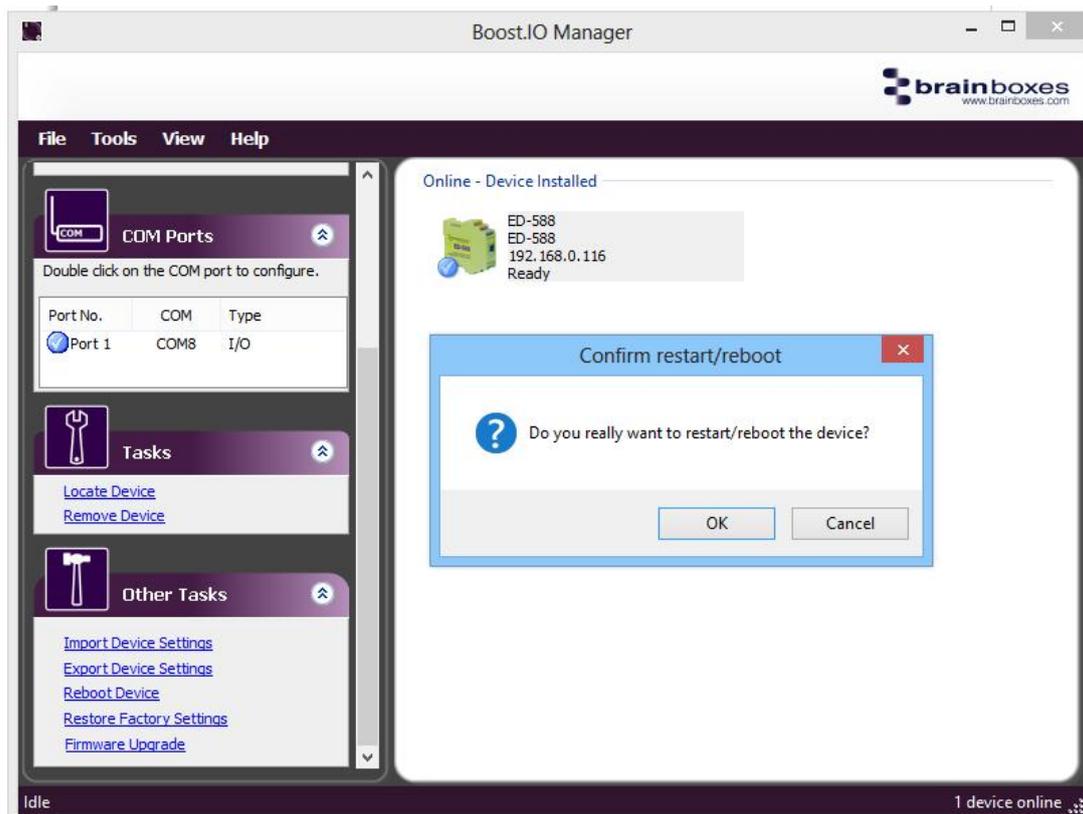


5. The ED device will reboot itself and have the new IP address once it has booted back up.

## 5.9 Rebooting Device

**Note: Please ensure your device is not in operation before rebooting it to prevent data loss.**

1. To reboot your device, firstly open Boost.IO Manager and select the device you wish to reboot.
2. On the left side of Boost.IO Manager under the 'Other Tasks' section, click 'Reboot Device'.

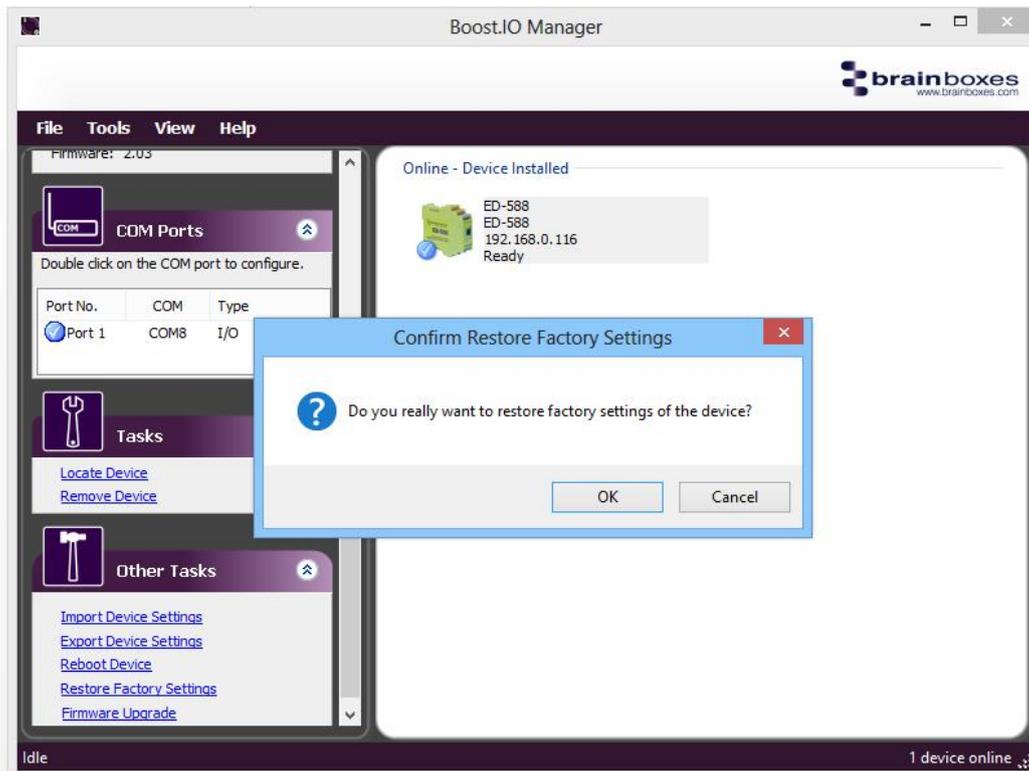


3. Confirm that you want to reboot your device.
4. The device will then reboot. Once the LED's on the device are green it is ready to use again.

## 5.10 Restoring Factory Settings

**Note: Restoring the Factory Settings will erase any current settings that you have on your ED device. Please ensure your device is not in operation before rebooting it to prevent data loss.**

1. To restore the factory settings of your device, open Boost.IO Manager and select the device which you want to restore.
2. On the left side of Boost.IO Manager, under the 'Other Tasks' section, click the 'Restore Factory Settings' link.

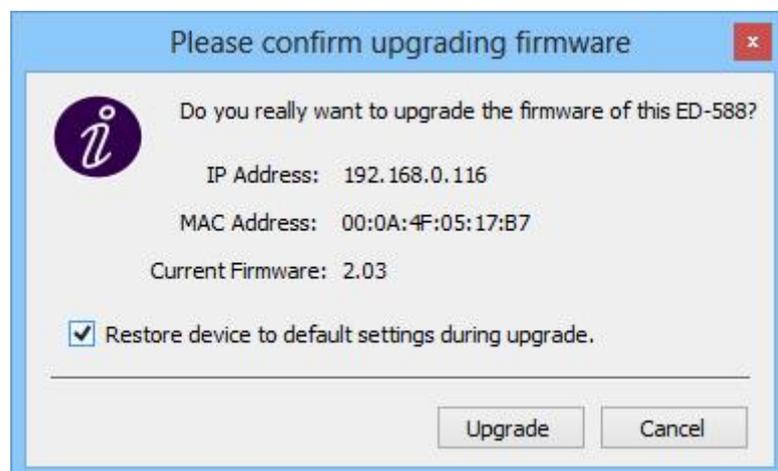


3. Confirm that you want to restore factory settings of the ED device.
4. The device will be rebooted and the factory settings will be restored.

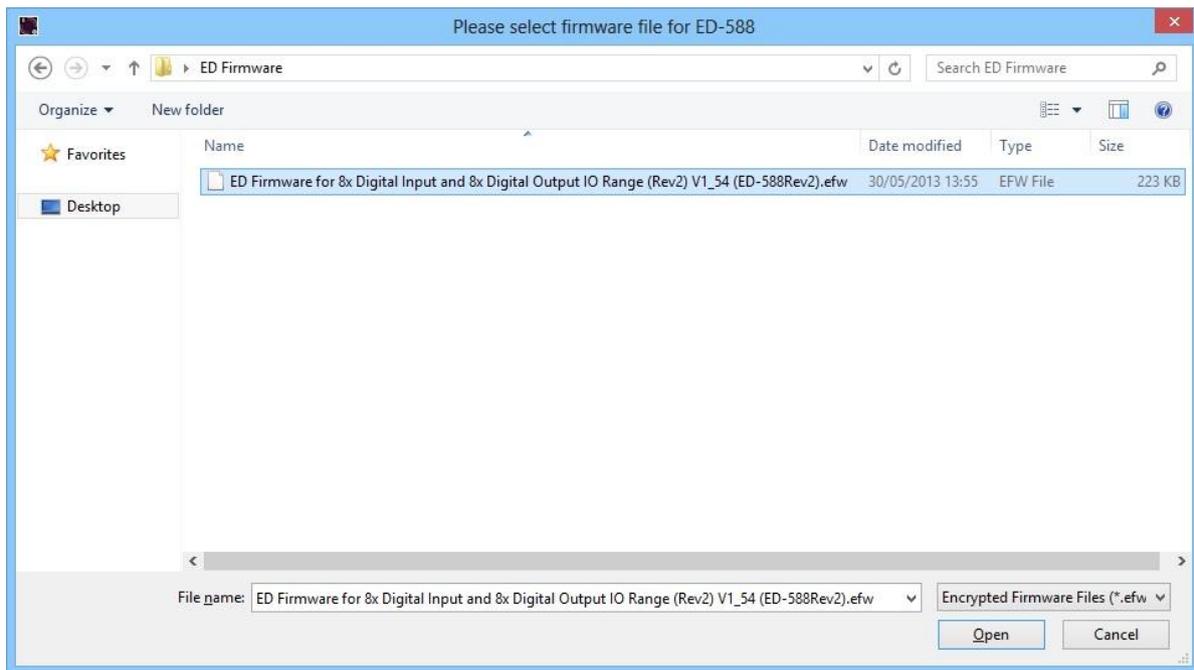
## 5.11 Firmware Upgrade

**Note: Before upgrading your firmware, please ensure your device is not in operation.**

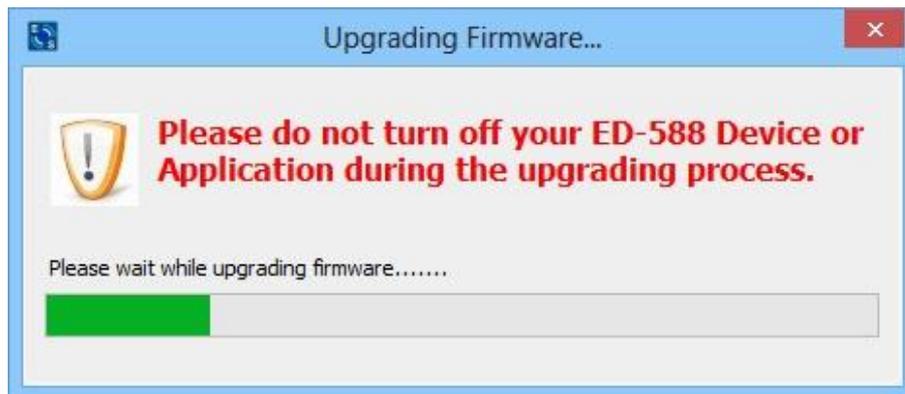
1. Open Boost.IO Manager and select your device in the window.
2. On the left side of Boost.IO Manager, under the 'Other Tasks' section, click the 'Firmware Upgrade' link.
3. When the message box appears confirm that you have selected the correct device. If you wish to restore the device to default settings click the tick option and then click 'Upgrade'. Otherwise just click 'Upgrade'



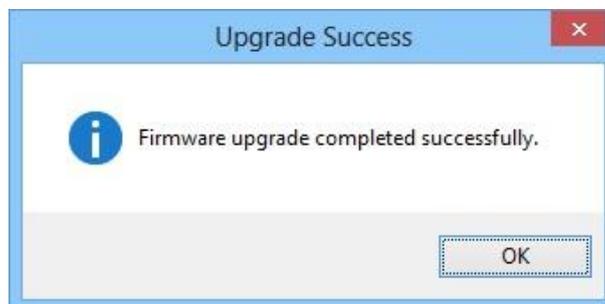
4. Navigate to the location that saved the new firmware efw file and open it.



5. The ED device will be upgraded with the firmware file you selected and then will reboot.



6. Click 'OK' when the upgrade success dialog box appears.



## 5.12 Proxy Server Settings

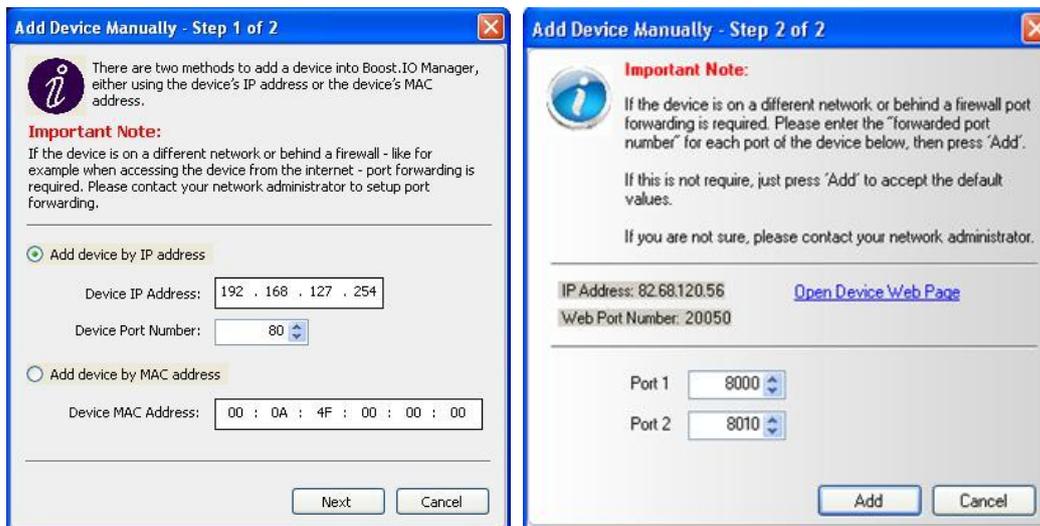
If you have a proxy server enabled on your PC this will restrict access to the web page interface. You may need to add the ED device to the Proxy Server's exceptions list. If you need help doing this contact your network administrator.

## 5.13 Device Swapping

In the unlikely event of a device failing, it can be easily replaced by swapping it with a device which has the same IP address. This is particularly useful when using a large number of ED devices together which have already been installed and setup and are already communicating with peripherals. The faulty device can be replaced without having to set up and install a new device. To set the device to the same IP address use the web page interface or Boost.IO to set the IP address to the static address of the device you are replacing.

## 5.14 Adding a Remote Device Using Boost.IO

1. To access the device through Boost.IO go to Tools and then click 'Add Device Manually'.
2. Enter the public IP address of your router into the Device IP Address field, and the port forwarding number into the Device Port Number field.
3. Click the 'Next' button.



**Add Device Manually - Step 1 of 2**

There are two methods to add a device into Boost.IO Manager, either using the device's IP address or the device's MAC address.

**Important Note:**  
If the device is on a different network or behind a firewall - like for example when accessing the device from the internet - port forwarding is required. Please contact your network administrator to setup port forwarding.

Add device by IP address

Device IP Address: 192 . 168 . 127 . 254

Device Port Number: 80

Add device by MAC address

Device MAC Address: 00 : 0A : 4F : 00 : 00 : 00

Next Cancel

---

**Add Device Manually - Step 2 of 2**

**Important Note:**  
If the device is on a different network or behind a firewall port forwarding is required. Please enter the "forwarded port number" for each port of the device below, then press 'Add'.  
If this is not require, just press 'Add' to accept the default values.  
If you are not sure, please contact your network administrator.

IP Address: 82.68.120.56 [Open Device Web Page](#)

Web Port Number: 20050

Port 1: 8000

Port 2: 8010

Add Cancel

4. Enter the port forwarding number for Port 1 and then click the 'Add' button. The device will then appear in the Boost.IO manager window. It can then be installed and used as if it were on a local network.

## 5.15 Remote Access

The Remote access feature of the ED devices allows access to the device over the internet or from a different network. The device can be accessed either through the webpage interface, or through Boost.IO Manager. To access the ED device remotely, you will need the IP address of the router and have port forwarding set up on your router for the device and the ports on your device. If you need assistance setting up port forwarding on your network, contact your network administrator.

Once you have the IP address and port forwarding numbers of the device and the ports, you can either access the device through the webpage or add the device manually using Boost.IO Manager.

The ED devices store their settings in one place, inside the device. When accessing the device either from the web configuration pages or using Boost.IO Manager, it will get and set the settings inside the device itself. Therefore, if Boost.IO Manager is closed and a setting is updated using the web configuration pages, Boost.IO won't know that any settings have changed until it is opened. When the application opens, it will automatically get the settings of any installed devices.

## 6 Web Configuration Pages

### 6.1 Introduction

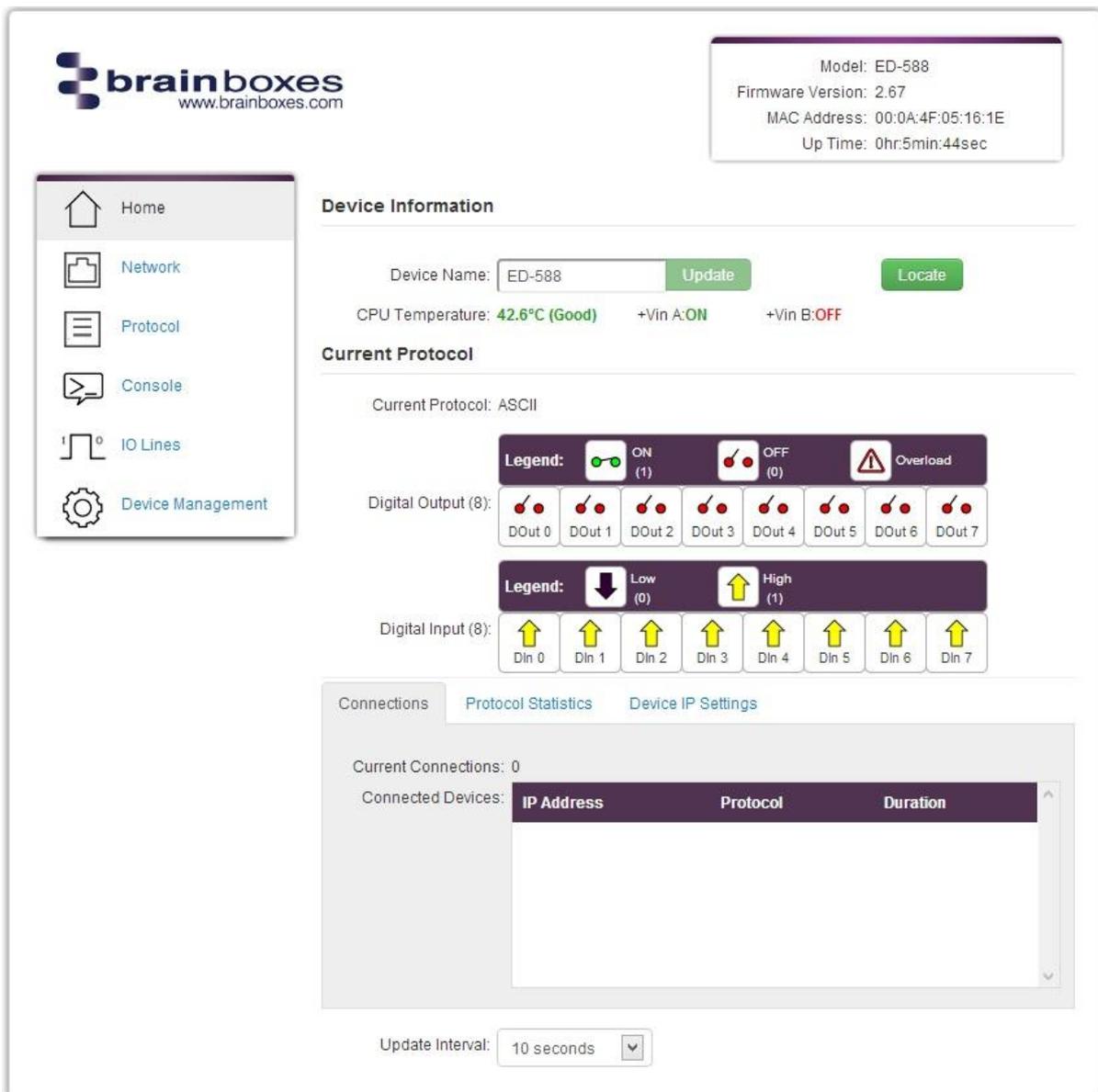
The web configuration pages for the ED device allow you to configure all the settings on the device and view the status of the device. To access the web configuration pages, type the IP address of the device into a web browser. To find the IP address of your device, see section [6.3 Finding and Installing an ED Device](#).

In the top right corner of every page there will be a box that displays information about your device. It will show you the model, firmware version, MAC address and how long the device has been powered on for.

Note: We recommend that you use Internet Explorer Version 7 and above

### 6.2 Home page

The Home page is the default page that loads when you enter the IP address of the device into a web browser. This page mainly displays information about the device and a visual display of the states of the input and output/relay lines.



**brainboxes**  
www.brainboxes.com

Model: ED-588  
Firmware Version: 2.67  
MAC Address: 00:0A:4F:05:16:1E  
Up Time: 0hr:5min:44sec

**Device Information**

Device Name:

CPU Temperature: **42.6°C (Good)**    +Vin A: **ON**    +Vin B: **OFF**

**Current Protocol**

Current Protocol: ASCII

**Digital Output (8):**

Legend: ON (1)    OFF (0)    Overload

DOut 0   DOut 1   DOut 2   DOut 3   DOut 4   DOut 5   DOut 6   DOut 7

**Digital Input (8):**

Legend: Low (0)    High (1)

DIn 0   DIn 1   DIn 2   DIn 3   DIn 4   DIn 5   DIn 6   DIn 7

**Connections**    Protocol Statistics    Device IP Settings

Current Connections: 0

Connected Devices:

| IP Address | Protocol | Duration |
|------------|----------|----------|
|            |          |          |

Update Interval: 10 seconds

## Device Information

Under the Device Information section is a text box which shows the current device name. This is the name that you can use to identify the device on your network. By default the device name is set to the product name but this can be changed by typing the name you would like into the text box and clicking 'Update'.

When the Locate button is pressed, all the LED's on the front of the device will start flashing for 10 seconds. If you have a number of devices next to each other on a DIN rail, you may not be able to see the sticker on the side to identify the devices. Using the Locate button will allow you to clearly see which device you are configuring on the DIN rail.

Below this is the CPU temperature and voltages of the power supplies. The temperature will be displayed in green unless the CPU is getting hotter than it should in which case the temperature will be displayed in red.

Also, displayed is the state of both power inputs, Vin A and Vin B. The state will either be on or off, depending on whether they have power or not.

## Current Protocol

The Current Protocol section displays the state of the digital inputs and outputs. The digital input lines have three states that they can be in: on, off or overload. The digital output lines have 2 states, either high or low. These states are represented by the diagrams in the legends.

**Current Protocol**

---

Current Protocol: ASCII

**Legend:**  ON (1)  OFF (0)  Overload

Digital Output (8):

 DOut 0

 DOut 1

 DOut 2

 DOut 3

 DOut 4

 DOut 5

 DOut 6

 DOut 7

**Legend:**  High (0)  Low (1)

Digital Input (8):

 Din 0

 Din 1

 Din 2

 Din 3

 Din 4

 Din 5

 Din 6

 Din 7

## Connections

The Connections tab displays information about any active connections to the device. The number of connections currently being made to the device will be displayed. The table will display more information about any devices connecting to the ED device.

Connections Protocol Statistics Device IP Settings

---

Current Connections: 1

Connected Devices:

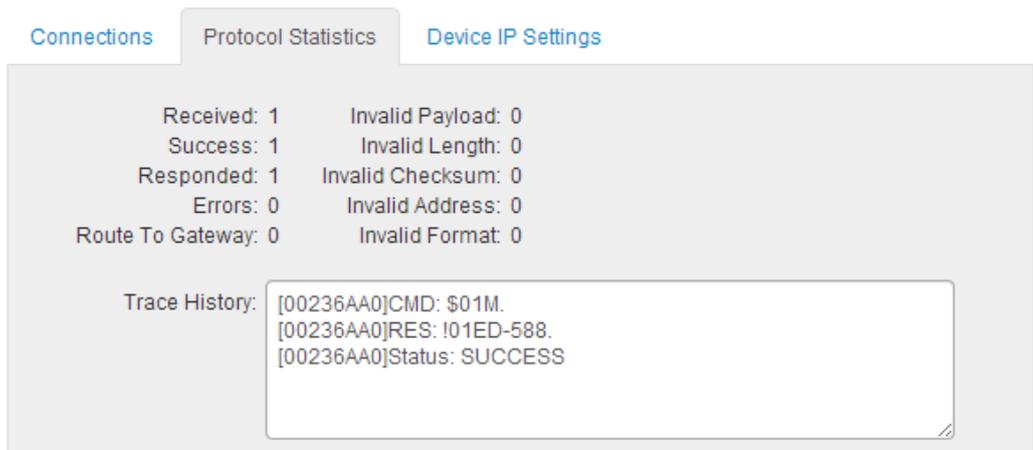
| IP Address         | Protocol | Duration |
|--------------------|----------|----------|
| 192.168.0.69:60157 | ASCII    | 00:00:07 |

Also, when a connection is made there will be a warning at the top of every page. When changing some of the settings, the device is required to be restarted for the settings to be applied and the ED device cannot be restarted while a connection is being made to it.

**Warning!** There is an active connection to the device. Close all connections before changing any settings.

### Protocol Statistics

The Protocol Statistics tab will display information about the protocol commands that have been sent to the device since it was powered on, including a history of all the commands sent and the response of each command.



The screenshot shows the 'Protocol Statistics' tab selected. It displays a table of statistics and a trace history window.

|                     |                     |
|---------------------|---------------------|
| Received: 1         | Invalid Payload: 0  |
| Success: 1          | Invalid Length: 0   |
| Responded: 1        | Invalid Checksum: 0 |
| Errors: 0           | Invalid Address: 0  |
| Route To Gateway: 0 | Invalid Format: 0   |

Trace History:

```
[00236AA0]CMD: $01M.  
[00236AA0]RES: !01ED-588.  
[00236AA0]Status: SUCCESS
```

### Device IP Settings

Under the Device IP Settings tab there is information about the device's network settings. These settings can be changed on the Network page. See section [6.3 Network page](#).

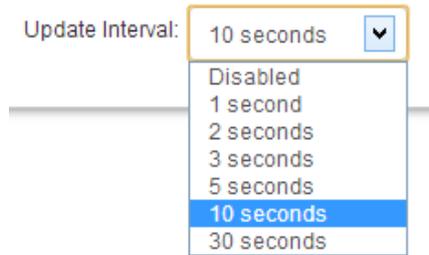


The screenshot shows the 'Device IP Settings' tab selected. It displays the following network settings:

- IP Address: 192.168.0.130
- Subnet Mask: 255.255.255.0
- Gateway Address: 192.168.0.254
- Web Server Port: 80

### Update Interval

The Update Interval dropdown box controls how often the device information on the Home page will be refreshed. For example, if set to 5 seconds, the webpage will update all the temperature and voltage information, IO lines states, current connections, protocol statistics and device IP settings automatically, every 5 seconds. If set to 'Disabled', the webpage will not update these settings and you will have to refresh the page manually.



### 6.3 Network page

The Network page allows you to configure all of the network settings of the ED device.

#### Network IP Address

Please consult the Network Administrator before changing any settings.



In the Network IP Address tab you can select whether the device is using DHCP or a static IP address which you specify. By default the ED device is set to use DHCP to obtain an IP address automatically. If there is no DHCP server present, after 60 seconds the device will switch to a static IP address of 192.168.127.254.

#### Web Server/UPnP

Please consult the Network Administrator before changing any settings.



Under the Web Server/UPnP tab you can change the Web Server Port number and also turn UPnP on or off. The default Web Server Port number is 80.

## 6.4 Serial Port page

The serial port page (available on some models) allows you to configure all the serial settings on the serial port on the device.

### Serial Port 1

Port Status: Connected

Port Type: RS232

#### Default Serial Port Settings

Always use these settings (Ignore settings from application)

Baud Rate:  bps

Data Bits:  ▼

Stop Bits:  ▼

Parity:  ▼

Flow Control:  ▼

#### Protocol Settings

Protocol Type:  ▼

Local TCP Port:

Connection Timeout:  seconds

**Note:** Connection timeout will not be used if set to '0'.

Save

Cancel Changes

### Always Use These Settings

If the tick box “Always use these settings” is ticked then the serial port will communicate using the baud rate, data bits, parity, stop bits and flow control set in this section regardless of what settings the application is using. For example, this will allow you to force baud rates that your application does not allow you to select. This can be useful for interfacing to equipment which uses higher baud rates or unusual baud rates which your application does not support.

### Protocol Type

If you are using the Brainboxes Windows driver to create a COM port then the driver will use Telnet regardless of the protocol setting set on the webpage.

- RAW TCP: Data sent over the network is the same as the data sent over the serial link. Any configuration of the serial port settings will need to be set via the web page.
- Telnet: This is based on the RFC2217 standard which lets the serial ports settings be configured dynamically. Data is then encapsulated over serial link

## Local TCP Port

This configures which TCP port the serial port of the ED device will use.

## Connection Timeout

The Connection Timeout is set to 0 by default. If it is set to anything other than 0, the connection to the port will be disconnected automatically after being idle (i.e. no data is being received or sent to and from the port) for the specified time. The LED for the port will go off at this point. If the Connection Timeout setting is 0, the port will need to be disconnected manually by the user.

## 6.5 Protocol page

The protocol page has all of the ASCII settings and serial gateway settings.

ASCII Protocol

### ASCII Protocol Settings

|   |   |                          |  |
|---|---|--------------------------|--|
| Device Address (1-255)                    | <input type="text" value="1"/> (Hex:01) | Counter Update Direction | <input type="text" value="Falling Edge"/>  |
| TCP Port (1-65535)                        | <input type="text" value="9500"/>       | Counter Mode             | <input type="text" value="16bit Counter"/> |
| Idle Timeout (0: no timeout)<br>(0-65535) | <input type="text" value="0"/>          | Checksum                 | <input type="text" value="Disabled"/>      |
|   |   | Input Polarity           | <input type="text" value="High = 1"/>      |
|   |   | Output Polarity          | <input type="text" value="1 = ON"/>        |

---

### Serial Gateway Settings

|                 |                                      |
|-----------------|--------------------------------------|
| Baud Rate       | <input type="text" value="9600"/>    |
| Command Timeout | <input type="text" value="1000"/> ms |

## 6.5.1 ASCII Protocol Settings

### Device Address

The device address is a hexadecimal value which specifies the target device. If an ASCII command is being sent to a specific device, this address is used to specify which device the command is sent to. Also, any responses will contain a device address to indicate which device sent the response.

The number entered into the text box should be a decimal number between 1 and 255. Any decimal numbers entered will have their Hex value displayed to the right of the text box.

### TCP Port

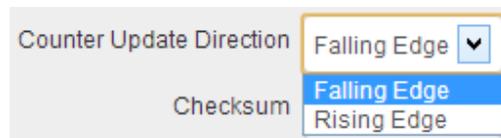
The TCP port is the network port that is used to connect to the device at the IP address that it's using. The default port number is 9500.

### Idle Timeout

When the idle timeout is set, if there is no communication to the device for the specified period of time, the connection will be closed. The default idle connection is 0, meaning the connection will never be dropped automatically.

### Counter Update Direction

The counter update direction can either be set to falling edge or rising edge using the drop down box. If the Update Direction is set to 'Falling Edge', the counter will be incremented every time the signal goes from high to low. If the Update Direction is set to 'Rising Edge' the counter will be incremented every time the signal goes from low to high. By default the Counter Update Direction is set to Falling Edge.



### Counter Mode

The Counter mode box allows you set the counter mode to either a 16-bit Counter or 32-bit Counter. A 16-bit counter will count from 0 to 65535 and a 32-bit counter will count from 0 to 4294967295.



### Checksum

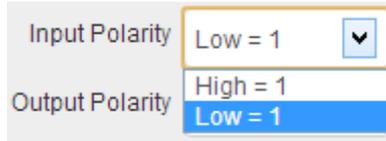
The purpose of the checksum is to help the PC and devices detect the communication errors that have corrupted the command strings. When the checksum is enabled all commands from the PC to the devices and all responses from the devices must contain a valid checksum otherwise the data is discarded. When the checksum is enabled, commands sent without a valid checksum will be ignored by the devices and the device will not respond to the host PC. By default, the checksum is disabled.



### Input Polarity

The input polarity drop down box has 2 options. The first and default is 'High = 1'. When this option is selected, if a High voltage is applied to the input line then this will be displayed a 1. If there is no voltage applied to the input line, this will be Low and the input will be displayed as 0.

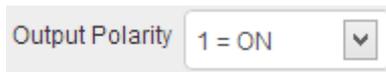
The second option is 'Low = 1'. If this is selected, when a High voltage is applied to the input line, then this will be displayed as 0. When no voltage is applied to the input line then 1 will be displayed.



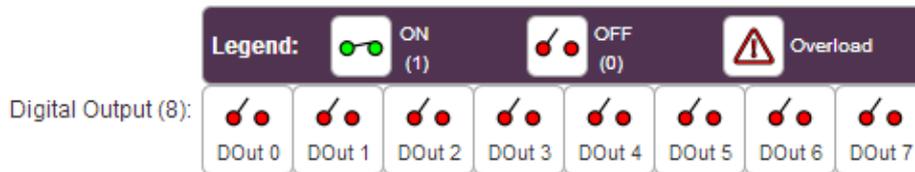
### Output Polarity

In the Output Polarity drop down box there are 2 options, '1 = ON' or '0 = ON'. When '1 = ON' is set, if there is a voltage on the output line, then the digital output will be 1. If there is no voltage being sent out then the digital output will be 0. When '0 = ON' is selected, when there is a voltage on the line the digital output will be 0. When there is no voltage on the output line then the digital output will be 1.

For example, when the output polarity is set to '1 = ON' in the combo box...



...the Digital Output diagrams on the home page will display the 'OFF (0)' symbol when there is no voltage on the input lines.



## 6.5.2 Serial Gateway

The Serial Gateway feature (available on ED-5XX models) turns the ED device into a serial gateway which allows it to send the commands out the gateway port to another ASCII protocol compatible device such as NuDAM, eDAM and ADAM modules. The gateway port uses half duplex RS485 with 8 data bits, no parity and 1 stop bit.

### Baud Rate

The baud rate of the Serial Gateway can be changed using the drop down box. The default baud rate is 9600.



## Command Timeout

The Command Timeout option determines how long the ED device will wait for a response from the Serial Gateway once a command has been sent. If, after the specified time no response has been received through the gateway port then the ED device will stop waiting for a response. If a response does come in after the timeout value, the response will be discarded. The default timeout is 200 milliseconds.

Command Timeout  ms

## 6.6 Console page

The Console Page contains the console window which allows ASCII commands to be sent straight to the device and be executed immediately. The response of the command is displayed in the console window in green. This is the simplest way to send ASCII commands to the ED device to either set or read settings.

### Terminal Console

To communicate with the device, please enter any valid ASCII command.  
Please [click here](#) to view a list of ASCII Protocol commands.

```
ASCII Command Console
> @01
>0000

> $01F
!012.69

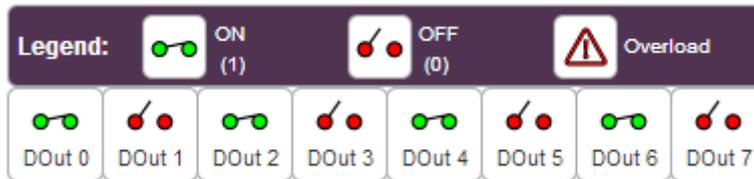
> $01M
!01ED-588

> █
```

Note: Please type "clear" to clear the console screen.

Also on the Console page is a link to a web view of all the ASCII protocol commands that can be sent to the device and a description of what each command does.





### Safe Value

The Safe Value drop down boxes set the state that the output lines will go when the device goes into a Watchdog state. This ensures that even if there is a problem with the communication to the device, the output lines will always revert to a safe, known state.

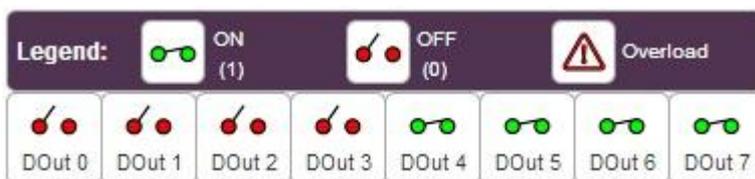
When the safe value is set like the pattern below...

| Line   | Mode           | Power-On Value | Safe Value | Quick Stop Value |
|--------|----------------|----------------|------------|------------------|
| DOut 0 | Digital Output | OFF            | OFF        | OFF              |
| DOut 1 | Digital Output | OFF            | OFF        | OFF              |
| DOut 2 | Digital Output | OFF            | OFF        | OFF              |
| DOut 3 | Digital Output | OFF            | OFF        | OFF              |
| DOut 4 | Digital Output | OFF            | ON         | OFF              |
| DOut 5 | Digital Output | OFF            | ON         | OFF              |
| DOut 6 | Digital Output | OFF            | ON         | OFF              |
| DOut 7 | Digital Output | OFF            | ON         | OFF              |

...and the device goes into a Watchdog state for whatever reason, there will be a message at the top of every configuration page warning about the state.



Also, the output lines will be set to the state which has been set on the IO Lines page.



### Quick Stop Value

When the Quick Stop Value is enabled, all the output lines will go to a user specified known state when the user programmed input transitions to a preconfigured high or low state.

To set the Quick Stop Value, the first thing that needs to be selected is the input line which is going to wait for the high or low signal.

Please Select Quick Stop IO Line :

Once the input line is selected, another drop down box will appear which allows you to select when the output line state is set. This could be on the falling edge, when the input line goes from high to low, or the rising edge, when the input line goes from low to high.

Please Select Quick Stop IO Line :   when

Once this has been selected, set the digital output line pattern needs to be set.

As an example, from the settings on the page below, when the signal of the first input line (Din 0) goes from high to low, the first four outputs, DOut 0, DOut 1, DOut, 2 and DOut 3 will be set high and the second four outputs will be set low.

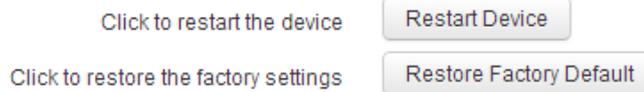
| Line   | Mode           | Power-On Value                       | Safe Value                           | Quick Stop Value                     |
|--------|----------------|--------------------------------------|--------------------------------------|--------------------------------------|
| DOut 0 | Digital Output | OFF <input type="button" value="v"/> | OFF <input type="button" value="v"/> | OFF <input type="button" value="v"/> |
| DOut 1 | Digital Output | OFF <input type="button" value="v"/> | OFF <input type="button" value="v"/> | OFF <input type="button" value="v"/> |
| DOut 2 | Digital Output | OFF <input type="button" value="v"/> | OFF <input type="button" value="v"/> | OFF <input type="button" value="v"/> |
| DOut 3 | Digital Output | OFF <input type="button" value="v"/> | OFF <input type="button" value="v"/> | OFF <input type="button" value="v"/> |
| DOut 4 | Digital Output | OFF <input type="button" value="v"/> | ON <input type="button" value="v"/>  | ON <input type="button" value="v"/>  |
| DOut 5 | Digital Output | OFF <input type="button" value="v"/> | ON <input type="button" value="v"/>  | ON <input type="button" value="v"/>  |
| DOut 6 | Digital Output | OFF <input type="button" value="v"/> | ON <input type="button" value="v"/>  | ON <input type="button" value="v"/>  |
| DOut 7 | Digital Output | OFF <input type="button" value="v"/> | ON <input type="button" value="v"/>  | ON <input type="button" value="v"/>  |

Please Select Quick Stop IO Line :   when

## 6.8 Device Management page

The Device Management page has 2 options. The first is to restart your device and the second to restore the factory default settings of the device.

### Device Management



#### Restart Device

Clicking the 'Restart Device' button will power cycle the device. In order to restart the device, there must be no connections being made to it. If there is a connection being made to the device, a warning message will be displayed asking you to close the connection.

#### Restore Factory Default

Clicking the 'Restore Factory Default' button will revert all the settings on the device back to their factory default. Just like restarting the device, in order to restore the factory settings, there must be no connection to the device active. For a list of the factory settings of the device please see section [7.8 Factory Default Settings](#).

## 6.9 Factory Default Settings

| Network Settings         |              |
|--------------------------|--------------|
| Network IP Address       | DHCP Mode    |
| Web Server Port          | 9500         |
| ASCII Protocol Settings  |              |
| Device Address           | 01           |
| TCP Port                 | 9500         |
| Idle Timeout             | 0            |
| Counter Update Direction | Falling Edge |
| Checksum                 | Disabled     |
| Input Polarity           | High = 1     |
| Output Polarity          | 1 = ON       |
| Serial Gateway Settings  |              |
| Baud Rate                | 9600         |
| Command Timeout          | 200 ms       |

## 7 Configuration Tables

### 7.1 Baud Rate Settings (BB)

|            |      |      |      |      |       |       |       |        |
|------------|------|------|------|------|-------|-------|-------|--------|
| Code:      | 03   | 04   | 05   | 06   | 07    | 08    | 09    | 0A     |
| Baud Rate: | 1200 | 2400 | 4800 | 9600 | 19200 | 38400 | 57600 | 115200 |

### 7.2 Data Format Settings (FF)

|    |    |    |          |   |   |   |   |
|----|----|----|----------|---|---|---|---|
| 7  | 6  | 5  | 4        | 3 | 2 | 1 | 0 |
| CU | CS | CM | Reserved |   |   |   |   |

CS: Checksum Setting

0: Disabled  
1: Enabled

CU: Counter Update

0: Falling edge in the input signal  
1: Rising edge in the input signal

CM: Counter Mode

0: 16 bit Counter  
1: 32 bit Counter

**Note:** The counter value will be preserved until it is either cleared or restored to factory settings. The counter will be preserved when the device is rebooted or powered off and on.

### 7.3 Digital Input/Output Data Characters Format

Some of the ASCII commands require data to be sent to set the output lines. Other ASCII commands will return the status of the input and output lines. The status of the output and input lines in the commands and responses will be in a 2 digit hexadecimal format.

For example, the response from the @AA command for the ED-588 is: >(first 2 hex digits)(second 2 hex digits)

The table below details which input/output lines these hexadecimal digits correspond to for your device.

| Device | First 2 hexadecimal digits |       | Second 2 hexadecimal digits |       |
|--------|----------------------------|-------|-----------------------------|-------|
| ED-588 | DOut 7 - DOut 0            | 00-FF | DIn 7 – DIn 0               | 00-FF |
| ED-516 | DIn 15 – DIn 8             | 00-FF | DIn 7 – DIn 0               | 00-FF |
| ED-527 | DOut 15 – DOut 8           | 00-FF | DOut 7 – DIn 0              | 00-FF |
| ED-538 | RL 0 – RL 3                | 00-0F | DIn 7 – DIn 0               | 00-FF |
| ED-038 | RL 0 – RL 2                | 00-07 | Din 2 – Din 0               | 00-07 |
| ED-008 | DOut 7 - DOut 0            | 00-FF | DIn 7 – DIn 0               | 00-FF |
| ED-204 | DOut 3 – DOut 0            | 00-0F | DIn 3 – DIn 0               | 00-0F |
| ED-004 | DOut 3 – DOut 0            | 00-0F | DIn 3 – DIn 0               | 00-0F |

## 8 ASCII Protocol

### 8.1 Introduction

The ASCII protocol is a query-response or a question and answer communication protocol in which a host PC uses ASCII characters to send commands to a device and then receives responses back from that device. The ASCII command set is used to configure devices, send data to devices and to read data and status information back from devices

#### Command Format

The command string is made up of several different parts. For example, the command #AA00DD can be broken down into as many as 6 separate parts.

| #         | <u>AA</u> | 00      | <u>(Data)</u> | <u>[CS]</u> | (CR)            |
|-----------|-----------|---------|---------------|-------------|-----------------|
| Delimiter | Address   | Command | Data*         | Checksum**  | Carriage Return |

\* Only applicable for commands setting digital outputs.

\*\* Optional parameter for the command.

Every ASCII command sequence is a series of ASCII characters starting with a prefix delimiter and terminating with a carriage return character. All of the ASCII characters used are easily entered from a PC keyboard and every ASCII command is terminated with a Carriage Return character; hex 0D, denoted by (CR) All commands being sent to the device must be in uppercase characters.

**Prefix or Delimiter:** Each ASCII command starts with a single character command prefix or delimiter. The prefix will be one of the following five characters:-

|    |   |
|----|---|
| #  | the hash or pound sign, ASCII value hex 0x23    |
| %  | the percentage sign, ASCII value hex 0x25       |
| \$ | the dollar sign, ASCII value hex 0x24           |
| @  | the at sign, ASCII value 0x40                   |
| ~  | the tilde or approx. sign, ASCII value hex 0x7E |

These prefix signs cannot be used interchangeably but are particular to the command string which follows.

**Address:** The ASCII protocol was first introduced in the 1980s by Analog Devices with its RS485 Half Duplex connected 6B Series modules and digital I/O boards and has been adopted and adapted by many other companies since. It is the de-facto communication protocol on the widely used RS485 Half Duplex connected ADAM/NuDam/eDAM modules. A very wide range of PC based data acquisition packages have support for this command protocol communicating over PC COM port. The Brainboxes ED-xxx range of devices are completely backwards compatible with these devices.

Since it was initially developed for an RS485 bus system containing many devices, each ASCII command must include the address of the particular device the command is directed to. The address is a two character field giving the hexadecimal address of the device (00-FF). The default address is 01. The address field is written as AA in the examples that follow.

A few commands do not have an address as these are broadcast commands that go to all the devices. Here the address field AA is replaced by the wildcard two star signs \*\*. Two such examples are #\*\* and ~\*\*.

**Command:** The command field contains the command that you want to send to the device. The next section ([Command List](#)) gives a detailed description of how to use each command and what the commands do.

**Data:** The data being sent to the device which is required in the command. Depending on the command this data is in different formats.

**Checksum:** An optional two character checksum denoted **[CS]** can be included immediately before the terminating **(CR)**. The purpose of the checksum is to help the PC and devices detect the communication errors that have corrupted the command strings. When the checksum is enabled all commands from the PC to the devices and all responses from the devices must contain a valid checksum otherwise the data is discarded. When the checksum is enabled, commands sent without a valid checksum will be ignored by the devices and the device will not respond to the host PC. By default the checksum is turned off. To turn checksum on, you must configure the device via its webpage using a browser.

In the following sections of this chapter, those parts of the command string in bold and underlined are parameters that the user must enter. Those parts of the command string in normal text are to be used literally in the command without replacement by the user.

Additionally, any parts of the command string that are in square brackets are optional. The checksum is an example of this as it is only required when the checksum is enabled in the firmware of the device.

**Example Commands:** Assuming that the checksum has not been turned on by the user, the simplest command is:

**@AA(CR)**

This command reads the status of the digital input port. Assuming the ED devices address is 01, that is **AA=01**, then the command the user would issue would be:

**@01(CR)**

A typical response from the device would be:

**>01FA(CR)**

### Response Format

The response received from the device will depend on the ASCII command that has been sent. The response for each command is detailed in the [Command List](#) Section.

| !         | <u>AA</u> | <u>(Data)</u> | <u>[CS]</u> | <u>(CR)</u>     |
|-----------|-----------|---------------|-------------|-----------------|
| Delimiter | Address   | Data*         | Checksum**  | Carriage Return |

\* Only applicable for certain commands that return data.

\*\* Optional parameter for the command.

**Prefix or Delimiter:** Each ASCII response starts with a single character prefix or delimiter. The prefix will be one of the following three characters:-

|   |   |
|---|---|
| > | Greater than sign, ASCII value hex 0x3E |
| ! | Exclamation sign, ASCII value hex 0x21  |
| ? | Question mark sign, ASCII 0x3F          |

**Address:** The address is a two character field giving the hexadecimal address of the device (00-FF). The default address is 01. The address field is written as AA in the examples that follow.

**Data:** The data being sent from the device in response to the command sent.

**Checksum:** Optional 2 character checksum.

## 8.2 Command List

| Command     | Response    | Description                                     | Supported Devices                |
|-------------|-------------|---|----------------------------------|
| %AANNTTCCFF | !AA         | Set Device Configuration                        | All Devices                      |
| #**         | No Response | Synchronized Sampling                           | All Devices                      |
| #AA00DD     | >           | Set Digital Output of Lower 8 Channels          | Devices with Outputs             |
| #AA0ADD     | >           | Set Digital Output of Lower 8 Channels          | Devices with Outputs             |
| #AA0BDD     | >           | Set Digital Output of Upper 8 Channels          | Devices with Outputs             |
| #AA1cDD     | >           | Set a Single Digital Output of Lower 8 Channels | Devices with Outputs             |
| #AAAcDD     | >           | Set a Single Digital Output of Lower 8 channels | Devices with Outputs             |
| #AABcDD     | >           | Set a Single Digital Output of Upper 8 Channels | Devices with more than 8 Outputs |
| #AAN        | !AA(Data)   | Read Digital Input Counter                      | Devices with Inputs              |
| \$AA2       | !AANNTTCCFF | Read Device Configuration                       | All Devices                      |
| \$AA4       | !S(Data)    | Read Synchronized Data                          | All Devices                      |
| \$AA5       | !AAS        | Read Reset Status                               | All Devices                      |
| \$AA6       | !(Data)     | Read Digital I/O Status                         | All Devices                      |
| \$AAC       | !AA         | Clear Latched Digital Input                     | Devices with Inputs              |
| \$AACN      | !AA         | Clear Digital Input Counter                     | Devices with Inputs              |
| \$AAF       | !AA(Data)   | Read Firmware Version                           | All Devices                      |
| \$AALS      | !(Data)     | Read Latched Digital Input                      | Devices with Inputs              |
| \$AAM       | !AA(Data)   | Read Device Name                                | All Devices                      |
| \$AAM0      | !AA(Data)   | Read Device Model                               | All Devices                      |
| \$AAM1      | !AA(Data)   | Read Device Location                            | All Devices                      |
| \$AAS1      | !AA         | Restore Factory Defaults                        | All Devices                      |
| \$AARS      | No Response | Reset the module to power-on state              | All Devices                      |
| @AA         | >(Data)     | Read Digital I/O Status                         | All Devices                      |
| @AA(Data)   | >           | Set Digital Output                              | Devices with Outputs             |
| ~AA0(Name)  | !AA         | Set Device Name                                 | All Devices                      |
| ~**         | No Response | Host OK   | All Devices                      |
| ~AA0        | !AASS       | Read Watchdog Timeout Status                    | All Devices                      |
| ~AA1        | !AA         | Reset Watchdog Timeout Status                   | All Devices                      |
| ~AA2        | !AAVV       | Read Watchdog Timeout Settings                  | All Devices                      |
| ~AA3EVV     | !AA         | Set Watchdog Timeout Value                      | All Devices                      |
| ~AA4V       | !AA(Data)   | Read Power On/Safe Value                        | Devices with Outputs             |
| ~AA5V       | !AA         | Set Power On/Safe Value                         | Devices with Outputs             |
| ~AAL        | !AASS       | Set Device Location                             | All Devices                      |

## 8.3 %AANNTCCFF

Description:

Set the device configuration.

Command Syntax:

| % <u>AANNTCCFF</u> [ <u>CS</u> ](CR) |   |
|--------------------------------------|---|
| <u>%</u>                             | Delimiter character   |
| <u>AA</u>                            | Address of the device to be configured in hexadecimal format (00 to FF)                           |
| <u>NN</u>                            | New device address in hexadecimal format (00 to FF)   |
| <u>TT</u>                            | Type code, should be 40 for DIO module  |
| <u>CC</u>                            | New Baud Rate code. (See <a href="#">Baud Rate Settings</a> )                                     |
| <u>FF</u>                            | Used to set the counter update direction and checksum (See <a href="#">Data Format Settings</a> ) |
| [ <u>CS</u> ]                        | Checksum  |
| (CR)                                 | Carriage Return   |

Response:

Valid Command: !AA[CS](CR)

Invalid Command: ?AA[CS](CR)

|               |  |
|---------------|--|
| !             | Delimiter for a valid command                          |
| ?             | Delimiter for an invalid command                       |
| <u>AA</u>     | Address of the device in hexadecimal format (00 to FF) |
| [ <u>CS</u> ] | Checksum   |
| (CR)          | Carriage Return  |

Examples:

Change the device address from 01 to 02. The device returns a valid response.

Command: %0102400600(CR)

Response: !02(CR)

Change the Baud Rate of device 01 to 115200. Device returns a valid response.

Command: %0101400A00(CR)

Response: !01(CR)

Change the Baud Rate of device 01 to 115200. Device returns invalid command because the baud rate code is incorrect.

Command: %010140FF00(CR)

Response: ?01(CR)

**Note:** Any change to the address and counter update direction take effect once the command has been received. Changes to the Baud Rate and checksum settings take effect the next time the device is restarted.

## 8.4 #\*\*

### Description:

Allows the device to read and store data for later retrieval.

### Command Syntax:

| #** <u>[CS]</u> (CR) |                               |
|----------------------|-------------------------------|
| #                    | Delimiter character           |
| **                   | Synchronized Sampling Command |
| <u>[CS]</u>          | Checksum                      |
| (CR)                 | Carriage Return               |

### Response:

There is no response for this command. To access the data, use the command \$AA4.

### Examples:

Send synchronized sampling command.

Command: #\*\* (CR)

No Response

## 8.5 #AA00DD

### Description:

Sets the digital output value of the lower eight channels.

### Command Syntax:

| # <u>AA</u> 00 <u>DD</u> [ <u>CS</u> ](CR) |  |
|--|--|
| #  | Delimiter character  |
| <u>AA</u>                                  | Address of the device to be configured in hexadecimal format (00 to FF)  |
| 00   | Command to set the digital output value of the lower 8 channels  |
| <u>DD</u>                                  | Two digit hexadecimal value. Bit 0 corresponds to DOut 0, bit 1 corresponds to DOut 1, etc. If the bit is 1, the digital output channel will be on, if the bit is 0, the digital output channel will be off. |
| [ <u>CS</u> ]                              | Checksum   |
| (CR)                                       | Carriage Return  |

### Response:

Valid Command: >[CS](CR)

Invalid Command: ?[CS](CR)

Ignored Command: ![CS](CR)

|               |                                  |
|---------------|----------------------------------|
| >             | Delimiter for a valid command    |
| ?             | Delimiter for an invalid command |
| !             | Delimiter for an ignored command |
| [ <u>CS</u> ] | Checksum                         |
| (CR)          | Carriage Return                  |

### Examples:

Set DOut 0, DOut 1, DOut 2, DOut 3 to on, and DOut 4, DOut 5, DOut 6, DOut 7 to off and device returns a valid response.

Command: #01000F(CR)

Response: >(CR)

**Note:** This command is only applicable on devices with digital output channels.

## 8.6 #AA0ADD

### Description:

Sets the digital output value of the lower eight channels.

### Command Syntax:

| # <u>AA</u> 0 <u>ADD</u> [ <u>CS</u> ](CR) |  |
|--|--|
| #  | Delimiter character  |
| <u>AA</u>                                  | Address of the device to be configured in hexadecimal format (00 to FF)  |
| 0A   | Command to set the digital output value of the lower 8 channels  |
| <u>DD</u>                                  | Two digit hexadecimal value. Bit 0 corresponds to DOut 0, bit 1 corresponds to DOut 1, etc. If the bit is 1, the digital output channel will be on, if the bit is 0, the digital output channel will be off. |
| [ <u>CS</u> ]                              | Checksum   |
| (CR)                                       | Carriage Return  |

### Response:

Valid Command: >[CS](CR)

Invalid Command: ?[CS](CR)

Ignored Command: ![CS](CR)

|               |                                  |
|---------------|----------------------------------|
| >             | Delimiter for a valid command    |
| ?             | Delimiter for an invalid command |
| !             | Delimiter for an ignored command |
| [ <u>CS</u> ] | Checksum                         |
| (CR)          | Carriage Return                  |

### Examples:

Set DOut 0, DOut 1, DOut 2, DOut 3 to on, and DOut 4, DOut 5, DOut 6, DOut 7 to off and device returns a valid response.

Command: #010A0F(CR)

Response: >(CR)

**Note:** This command is only applicable on devices with digital output channels.

## 8.7 #AA0BDD

Description:

Sets the digital output value of the upper eight channels.

Command Syntax:

| # <u>AA</u> 0 <u>BDD</u> [ <u>CS</u> ](CR) |  |
|--|--|
| #  | Delimiter character  |
| <u>AA</u>                                  | Address of the device to be configured in hexadecimal format (00 to FF)  |
| 0 <u>B</u>                                 | Command to set the digital output value of the upper 8 channels  |
| <u>DD</u>                                  | Two digit hexadecimal value. Bit 0 corresponds to DOut 0, bit 1 corresponds to DOut 1, etc. If the bit is 1, the digital output channel will be on, if the bit is 0, the digital output channel will be off. |
| [ <u>CS</u> ]                              | Checksum   |
| (CR)                                       | Carriage Return  |

Response:

Valid Command: >[CS](CR)

Invalid Command: ?[CS](CR)

Ignored Command: ![CS](CR)

|               |                                  |
|---------------|----------------------------------|
| >             | Delimiter for a valid command    |
| ?             | Delimiter for an invalid command |
| !             | Delimiter for an ignored command |
| [ <u>CS</u> ] | Checksum                         |
| (CR)          | Carriage Return                  |

Examples:

Set DOut 8, DOut 9, DOut 10, DOut 11 to on, and DOut 12, DOut 13, DOut 14, DOut 15 to off and device returns a valid response.

Command: #010B0F(CR)

Response: >(CR)

**Note:** This command is only applicable on devices with digital output channels.

## 8.8 #AA1cDD

### Description:

Sets the specified digital output channel of the lower eight channels.

### Command Syntax:

| # <u>AA</u> 1 <u>c</u> <u>DD</u> [ <u>CS</u> ](CR) |   |
|--|---|
| #  | Delimiter character   |
| <u>AA</u>  | Address of the device to be configured in hexadecimal format (00 to FF)               |
| 1  | Command to set a single digital output of the lower 8 channels                        |
| <u>c</u>   | Specifies the digital output channel to be set (0-7)                                  |
| <u>DD</u>  | 00: set the digital output channel to off<br>01: set the digital output channel to on |
| [ <u>CS</u> ]                                      | Checksum  |
| (CR)   | Carriage Return   |

### Response:

Valid Command: >[CS](CR)

Invalid Command: ?[CS](CR)

Ignored Command: ![CS](CR)

|               |                                  |
|---------------|----------------------------------|
| >             | Delimiter for a valid command    |
| ?             | Delimiter for an invalid command |
| !             | Delimiter for an ignored command |
| [ <u>CS</u> ] | Checksum                         |
| (CR)          | Carriage Return                  |

### Examples:

Set DOut 1, to on and device returns a valid response.

Command: #011101(CR)

Response: >(CR)

**Note:** This command is only applicable for the devices with digital output channels.

## 8.9 #AAAcDD

### Description:

Sets the specified digital output channel from the lower eight channels.

### Command Syntax:

| # <u>AA</u> <u>c</u> <u>DD</u> [ <u>CS</u> ](CR) |   |
|--|---|
| #  | Delimiter character   |
| <u>AA</u>  | Address of the device to be configured in hexadecimal format (00 to FF)               |
| A  | Command to set a single digital output of the lower 8 channels                        |
| <u>c</u>   | Specifies the digital output channel to be set (0-7)                                  |
| <u>DD</u>  | 00: set the digital output channel to off<br>01: set the digital output channel to on |
| [ <u>CS</u> ]                                    | Checksum  |
| (CR)   | Carriage Return   |

### Response:

Valid Command: >[CS](CR)

Invalid Command: ?[CS](CR)

Ignored Command: ![CS](CR)

|               |                                  |
|---------------|----------------------------------|
| >             | Delimiter for a valid command    |
| ?             | Delimiter for an invalid command |
| !             | Delimiter for an ignored command |
| [ <u>CS</u> ] | Checksum                         |
| (CR)          | Carriage Return                  |

### Examples:

Set DOut 1, to on and device returns a valid response.

Command: #01A101(CR)

Response: >(CR)

**Note:** This command is only applicable for the devices with digital output channels.

## 8.10 #AABcDD

Description:

Sets the specified digital output channel from the upper eight channels.

Command Syntax:

| # <u>A</u> <u>A</u> <u>B</u> <u>c</u> <u>D</u> <u>D</u> [ <u>C</u> <u>S</u> ](CR) |   |
|---|---|
| #   | Delimiter character   |
| <u>AA</u>   | Address of the device to be configured in hexadecimal format (00 to FF)                                       |
| <u>B</u>  | Command to set a single digital output of the upper 8 channels  |
| <u>c</u>  | Specifies the digital output channel to be set (0-7) where 0 stands for channel 8, 1 stands for channel 9 etc |
| <u>DD</u>   | 00: set the digital output channel to off<br>01: set the digital output channel to on                         |
| [ <u>CS</u> ]   | Checksum  |
| (CR)  | Carriage Return   |

Response:

Valid Command: >[CS](CR)

Invalid Command: ?[CS](CR)

Ignored Command: ![CS](CR)

|               |                                  |
|---------------|----------------------------------|
| >             | Delimiter for a valid command    |
| ?             | Delimiter for an invalid command |
| !             | Delimiter for an ignored command |
| [ <u>CS</u> ] | Checksum                         |
| (CR)          | Carriage Return                  |

Examples:

Set DOut 11, to on and device returns a valid response.

Command: #01B301(CR)

Response: >(CR)

**Note:** This command is only applicable for the devices with 8 or more digital output channels.

## 8.11 #AAN

Description:

Reads the digital input counter value of specified channel.

Command Syntax:

| # <u>AAN</u> [ <u>CS</u> ](CR) |   |
|--------------------------------|---|
| #                              | Delimiter character   |
| <u>AA</u>                      | Address of the device to be configured in hexadecimal format (00 to FF) |
| <u>N</u>                       | Digital input channel to be read (0 to F)                               |
| [ <u>CS</u> ]                  | Checksum  |
| (CR)                           | Carriage Return   |

Response:

Valid Command: !AAData[CS](CR)

Invalid Command: ?AA[CS](CR)

|               |   |
|---------------|---|
| !             | Delimiter for a valid command   |
| ?             | Delimiter for an invalid command  |
| <u>AA</u>     | Address of the device in hexadecimal format (00 to FF)  |
| <u>Data</u>   | If the counter mode of the device is set to 16-bit, then the data will be a five digit decimal representing the digital input counter value. (00000 to 65535)<br>If the counter mode is set to 32-bit, then the data will be a ten digit decimal representing the digital input counter value. (0000000000 to 4294967295) |
| [ <u>CS</u> ] | Checksum  |
| (CR)          | Carriage Return   |

Examples:

Read counter value of digital input channel 3 and the returned value is 00274 as a five digit decimal value.

Command: #013(CR)

Response: !0100274(CR)

Read data from channel 9. An error is response is returned because channel 9 is an invalid channel.

Command: #019(CR)

Response: ?01(CR)

**Note:** This command is only applicable for the devices with digital input channels.

## 8.12 \$AA2

Description:

Reads device configuration.

Command Syntax:

| <b>\$AA2[CS](CR)</b> |   |
|----------------------|---|
| <b>\$</b>            | Delimiter character   |
| <b>AA</b>            | Address of the device to be configured in hexadecimal format (00 to FF) |
| <b>2</b>             | Command to read the device configuration                                |
| <b>[CS]</b>          | Checksum  |
| <b>(CR)</b>          | Carriage Return   |

Response:

Valid Command: **!AATTBBFF[CS](CR)**

Invalid Command: **?AA[CS](CR)**

|             |   |
|-------------|---|
| <b>!</b>    | Delimiter for a valid command                                     |
| <b>?</b>    | Delimiter for an invalid command                                  |
| <b>AA</b>   | Address of the device (00 to FF)                                  |
| <b>TT</b>   | Device Type (40 for DIO Devices)                                  |
| <b>BB</b>   | Baud Rate of the device (See <a href="#">Baud Rate Settings</a> ) |
| <b>FF</b>   | Configuration value (See <a href="#">Data Format Settings</a> )   |
| <b>[CS]</b> | Checksum  |
| <b>(CR)</b> | Carriage Return   |

Examples:

Read the configuration of device 01.

Command: **\$012(CR)**

Response: **!010400600(CR)**

### 8.13 \$AA4

Description:

Reads the synchronized data retrieved from the last **###** command.

Command Syntax:

| <b>\$AA4[CS](CR)</b> |   |
|----------------------|---|
| <b>\$</b>            | Delimiter character   |
| <b>AA</b>            | Address of the device to be configured in hexadecimal format (00 to FF) |
| <b>4</b>             | Command to read the synchronized data from the device.                  |
| <b>[CS]</b>          | Checksum  |
| <b>(CR)</b>          | Carriage Return   |

Response:

Valid Command: **!SDDDD00[CS](CR)**

Invalid Command: **?AA[CS](CR)**

|             |  |
|-------------|--|
| <b>!</b>    | Delimiter for a valid command  |
| <b>?</b>    | Delimiter for an invalid command   |
| <b>S</b>    | Status of synchronized data:<br>1 = First read of data<br>0 = Data has been at least once before |
| <b>DDDD</b> | Four digit hexadecimal value representing the status of the I/O lines.                           |
| <b>00</b>   | Always 00  |
| <b>[CS]</b> | Checksum   |
| <b>(CR)</b> | Carriage Return  |

Examples:

Read synchronized data. Device returns the synchronized data and sets first byte to 1 to indicate this is the first time the synchronized data has been read. (Assumes the set synchronized data command has been sent)

Command: **\$014(CR)**

Response: **!1000F00(CR)**

Read synchronized data. Device returns synchronized data and sets status byte to 0 to indicate the data has been read before.

Command: **\$014(CR)**

Response: **!0000F00(CR)**

## 8.14 \$AA5

### Description:

Read the reset status of the device.

### Command Syntax:

| <b>\$AA5[CS](CR)</b> |   |
|----------------------|---|
| <b>\$</b>            | Delimiter character   |
| <b>AA</b>            | Address of the device to be configured in hexadecimal format (00 to FF) |
| <b>5</b>             | Command to read the reset status of the device                          |
| <b>[CS]</b>          | Checksum  |
| <b>(CR)</b>          | Carriage Return   |

### Response:

Valid Command: **!AAS[CS](CR)**

Invalid Command: **?AA[CS](CR)**

|             |   |
|-------------|---|
| <b>!</b>    | Delimiter for a valid command   |
| <b>?</b>    | Delimiter for an invalid command  |
| <b>AA</b>   | Address of the device (00 to FF)  |
| <b>S</b>    | Reset status of the device:<br>1 = First time the command has been sent since device was powered on<br>0 = Not the first time the command has been sent since the device was powered on |
| <b>[CS]</b> | Checksum  |
| <b>(CR)</b> | Carriage Return   |

### Examples:

Read reset status of the device. Response shows it is the first time the device has received the \$AA5 command since it was powered on.

Command: **\$015(CR)**

Response: **!011(CR)**

Read reset status of the device. Response shows that the device has received the \$AA5 command while it's been powered on.

Command: **\$015(CR)**

Response: **!010(CR)**

## 8.15 \$AA6

Description:

Reads the digital input and digital output channel values.

Command Syntax:

| <b>\$AA6[CS](CR)</b> |   |
|----------------------|---|
| <b>\$</b>            | Delimiter character   |
| <b>AA</b>            | Address of the device to be configured in hexadecimal format (00 to FF) |
| <b>6</b>             | Command to read the digital I/O channels                                |
| <b>[CS]</b>          | Checksum  |
| <b>(CR)</b>          | Carriage Return   |

Response:

Valid Command: **!DDDD00[CS](CR)**

Invalid Command: **?AA[CS](CR)**

|             |  |
|-------------|--|
| <b>!</b>    | Delimiter for a valid command  |
| <b>?</b>    | Delimiter for an invalid command                                       |
| <b>DDDD</b> | Four digit hexadecimal value representing the status of the I/O lines. |
| <b>00</b>   | Always 00  |
| <b>[CS]</b> | Checksum   |
| <b>(CR)</b> | Carriage Return  |

Examples:

Reads the values of the digital input and digital output channels. Returns !FF0000 meaning that the all the output lines are high and all the input lines are low.

Command: **\$016(CR)**

Response: **!FF0000(CR)**

## 8.16 \$AAC

### Description:

Clears the status of the latched digital input channels.

### Command Syntax:

| <u>\$AAC[CS](CR)</u> |   |
|----------------------|---|
| \$                   | Delimiter character   |
| <u>AA</u>            | Address of the device to be configured in hexadecimal format (00 to FF) |
| C                    | Command to clear latched digital input                                  |
| <u>[CS]</u>          | Checksum  |
| (CR)                 | Carriage Return   |

### Response:

Valid Command: !AA[CS](CR)

Invalid Command: ?AA[CS](CR)

|             |                                       |
|-------------|---------------------------------------|
| !           | Delimiter for a valid command         |
| ?           | Delimiter for an invalid command      |
| <u>AA</u>   | Address of the device which responded |
| <u>[CS]</u> | Checksum                              |
| (CR)        | Carriage Return                       |

### Examples:

Sends command to clear the status of the latched digital input channels. Returns a valid response.

Command: \$01C(CR)

Response: !01(CR)

**Note:** This command is only applicable on devices with digital input channels.

## 8.17 \$AACN

### Description:

Clears the digital input counter of the specified channel.

### Command Syntax:

| <b>\$<u>AA</u><u>CN</u>[<u>CS</u>](CR)</b> |   |
|--|---|
| <b>\$</b>                                  | Delimiter character   |
| <b><u>AA</u></b>                           | Address of the device to be configured in hexadecimal format (00 to FF) |
| <b>C</b>                                   | Command to clear digital input counter                                  |
| <b><u>N</u></b>                            | Channel to be cleared (0 to F)  |
| <b>[<u>CS</u>]</b>                         | Checksum  |
| <b>(CR)</b>                                | Carriage Return   |

### Response:

Valid Command: **!AA[CS](CR)**

Invalid Command: **?AA[CS](CR)**

|                    |                                       |
|--------------------|---------------------------------------|
| <b>!</b>           | Delimiter for a valid command         |
| <b>?</b>           | Delimiter for an invalid command      |
| <b><u>AA</u></b>   | Address of the device which responded |
| <b>[<u>CS</u>]</b> | Checksum                              |
| <b>(CR)</b>        | Carriage Return                       |

### Examples:

Sends command to clear the value of channel 3. Returns a valid response.

Command: **\$01C3(CR)**

Response: **!01(CR)**

**Note:** This command is only applicable on devices with digital input channels.

## 8.18 \$AAF

Description:

Reads the firmware version of the device.

Command Syntax:

| <b><u>\$AAF[CS](CR)</u></b> |   |
|-----------------------------|---|
| <b>\$</b>                   | Delimiter character   |
| <b><u>AA</u></b>            | Address of the device to be configured in hexadecimal format (00 to FF) |
| <b>F</b>                    | Command to read the firmware version                                    |
| <b><u>[CS]</u></b>          | Checksum  |
| <b>(CR)</b>                 | Carriage Return   |

Response:

Valid Command: **!AA(Data)[CS](CR)**

Invalid Command: **?AA[CS](CR)**

|                      |  |
|----------------------|--|
| <b>!</b>             | Delimiter for a valid command              |
| <b>?</b>             | Delimiter for an invalid command           |
| <b><u>AA</u></b>     | Address of the device which responded      |
| <b><u>(Data)</u></b> | Firmware version of the responding device. |
| <b><u>CHKSUM</u></b> | Checksum                                   |
| <b>(CR)</b>          | Carriage Return                            |

Examples:

Reads the firmware version of the device and shows it as version 1.9.

Command: **\$01F(CR)**

Response: **!011.9(CR)**

## 8.19 \$AALS

Description:

Reads the status of the latched digital input channels.

Command Syntax:

| <u>\$AALS[CS](CR)</u> |  |
|-----------------------|--|
| <u>\$</u>             | Delimiter character  |
| <u>AA</u>             | Address of the device to be configured in hexadecimal format (00 to FF)                  |
| <u>L</u>              | Command to read the latched status   |
| <u>S</u>              | Reads the latched status:<br>0 = read low latched status<br>1 = read high latched status |
| <u>[CS]</u>           | Checksum   |
| <u>(CR)</u>           | Carriage Return  |

Response:

Valid Command: !DDDD00[CS](CR)

Invalid Command: ?AA[CS](CR)

|             |   |
|-------------|---|
| <u>!</u>    | Delimiter for a valid command             |
| <u>?</u>    | Delimiter for an invalid command          |
| <u>AA</u>   | Module Address ID                         |
| <u>DDDD</u> | Status of latched digital input channels. |
| <u>00</u>   | This value is always 00                   |
| <u>[CS]</u> | Checksum                                  |
| <u>(CR)</u> | Carriage Return                           |

Examples:

Reads the status of the low latched digital input. Command is valid and device returns 00FF

Command: \$01L0(CR)

Response: !00FF00(CR)

**Note:** This command is only available on devices with digital input channels.

## 8.20 \$AAM

Description:

Read the name of the device.

Command Syntax:

| <b><u>\$AAM[CS]</u>(CR)</b> |   |
|-----------------------------|---|
| <b>\$</b>                   | Delimiter character   |
| <b><u>AA</u></b>            | Address of the device to be configured in hexadecimal format (00 to FF) |
| <b>M</b>                    | Command to read the device's name                                       |
| <b><u>[CS]</u></b>          | Checksum  |
| <b>(CR)</b>                 | Carriage Return   |

Response:

Valid Command: **!AA(Data)[CS](CR)**

Invalid Command: **?AA[CS](CR)**

|                      |                                  |
|----------------------|----------------------------------|
| <b>!</b>             | Delimiter for a valid command    |
| <b>?</b>             | Delimiter for an invalid command |
| <b><u>AA</u></b>     | Address of the responding device |
| <b><u>(Data)</u></b> | Name of the device               |
| <b><u>[CS]</u></b>   | Checksum                         |
| <b>(CR)</b>          | Carriage Return                  |

Examples:

Reads the device name. Command sent to the ED-588 and a valid response is returned with the device's name, ED-588.

Command: **\$01M(CR)**

Response: **!01ED-588(CR)**

## 8.21 \$AAS1

### Description:

Restores the device settings back to factory defaults

Any Digital Outputs on the device will be reset to OPEN (0)

Digital Input reset values are dependant on 2 properties:

- Whether the line is connected to an external device.
  - If the input is not connected then it will float
  - If the input is connected then it will be at the value set by the external device.
- NPN/PNP jumper setting
  - NPN jumper setting means the Inputs will float HIGH (1). This is the factory default.
  - PNP jumper setting means the Inputs will float LOW (0)

Therefore a factory reset ED-588 which is disconnected from input lines, and with factory default jumper setting, will have its IO lines reset to: FF00 - inputs float HIGH (1), outputs open (0)

Note: After this device has responded, it will reboot and come back online in factory default state. The device can take a number of seconds to reboot. Close all connections to the device after sending the command. Connection can be re-established once the device has rebooted.

### Command Syntax:

| <b>\$<u>AA</u>S1[<u>CS</u>](CR)</b> |   |
|-------------------------------------|---|
| <b>\$</b>                           | Delimiter character   |
| <b><u>AA</u></b>                    | Address of the device to be configured in hexadecimal format (00 to FF) |
| <b>S1</b>                           | Command to restore the factory defaults                                 |
| <b>[<u>CS</u>]</b>                  | Checksum  |
| <b>(CR)</b>                         | Carriage Return   |

### Response:

Valid Command: **!AA[CS](CR)**

Invalid Command: **?AA[CS](CR)**

|                    |                                  |
|--------------------|----------------------------------|
| <b>!</b>           | Delimiter for a valid command    |
| <b>?</b>           | Delimiter for an invalid command |
| <b><u>AA</u></b>   | Address of the responding device |
| <b>[<u>CS</u>]</b> | Checksum                         |
| <b>(CR)</b>        | Carriage Return                  |

### Examples:

Restores the device back to factory settings. Command sent to the ED-588 and a valid response returned.

Command: **\$01S1(CR)**

Response: **!01(CR)**

## 8.22 \$AARS

### Description:

Resets the device to power on state. Will reboot the device to reset.

Note: The device can take a number of seconds to reboot. Close all connections to the device after sending the command. Connection can be re-established once the device has rebooted.

### Command Syntax:

| <u>\$AARS[CS](CR)</u> |   |
|-----------------------|---|
| \$                    | Delimiter character   |
| <u>AA</u>             | Address of the device to be configured in hexadecimal format (00 to FF) |
| RS                    | Command to reset device   |
| <u>[CS]</u>           | Checksum  |
| (CR)                  | Carriage Return   |

### Response:

No Response

The device can take a number of seconds to reboot. Close all connections to the device after sending the command. Connection can be re-established once the device has rebooted.

### Examples:

Reset device to power on state. Command sent to the ED-588 and no response is expected.

Command: \$01RS (CR)

No response

## 8.23 @AA

Description:

Read the status of the digital I/O lines.

Command Syntax:

| <b>@AA[CS](CR)</b> |   |
|--------------------|---|
| <b>@</b>           | Delimiter character   |
| <b>AA</b>          | Address of the device to be configured in hexadecimal format (00 to FF) |
| <b>[CS]</b>        | Checksum  |
| <b>(CR)</b>        | Carriage Return   |

Response:

Valid Command: **>DDDD[CS](CR)**

Invalid Command: **?AA[CS](CR)**

|             |  |
|-------------|--|
| <b>&gt;</b> | Delimiter for a valid command  |
| <b>?</b>    | Delimiter for an invalid command                                       |
| <b>DDDD</b> | Four digit hexadecimal value representing the status of the I/O lines. |
| <b>[CS]</b> | Checksum   |
| <b>(CR)</b> | Carriage Return  |

Examples:

Reads the digital input/output port status of the device's lines and returns F000 representing lines DOut 0-DOut 3 being low, DOut 4-DOut 7 being high and DIn0-DIn7 as being low.

Command: **@01(CR)**

Response: **>0203(CR)**

## 8.24 @AA(Data)

Description:

Sets the digital output channel value at a specified address.

Command Syntax:

| <b>@AA(Data)[CS](CR)</b> |   |
|--------------------------|---|
| <b>@</b>                 | Delimiter character   |
| <b>AA</b>                | Address of the device to be configured in hexadecimal format (00 to FF) |
| <b>(Data)</b>            | Data to be set to the digital output channels in a hexadecimal format.  |
| <b>[CS]</b>              | Checksum  |
| <b>(CR)</b>              | Carriage Return   |

Response:

Valid Command: >**[CS]**(CR)

Invalid Command: ?**AA[CS]**(CR)

|             |                                  |
|-------------|----------------------------------|
| >           | Delimiter for a valid command    |
| ?           | Delimiter for an invalid command |
| !           | Ignore command                   |
| <b>[CS]</b> | Checksum                         |
| <b>(CR)</b> | Carriage Return                  |

Examples:

Set output of device to 55 and return a valid command. Command sets D0out 0, D0out 2, D0out 4, D0out 6 to high and D0out 1, D0out 3, D0out 5, D0out 7 to low.

Command: @0155(CR)

Response: >(CR)

**Note:** This command is only available on devices with digital output channels.

## 8.25 ~AA0(Name)

Description:

Sets a new device name.

Command Syntax:

| <u>~AA0(Name)[CS](CR)</u> |   |
|---------------------------|---|
| ~                         | Delimiter character   |
| <u>AA</u>                 | Address of the device to be configured in hexadecimal format (00 to FF) |
| 0                         | Command to set the name of the device                                   |
| <u>(Name)</u>             | New device name (10 characters max)                                     |
| <u>[CS]</u>               | Checksum  |
| (CR)                      | Carriage Return   |

Response:

Valid Command: **!AA[CS](CR)**

Invalid Command: **?AA[CS](CR)**

|             |                                  |
|-------------|----------------------------------|
| !           | Delimiter for a valid command    |
| ?           | Delimiter for an invalid command |
| <u>AA</u>   | Address of the device            |
| <u>[CS]</u> | Checksum                         |
| (CR)        | Carriage Return                  |

Examples:

Set the device name to 588 and receives a valid command response.

Command: ~010588(CR)

Response: !01(CR)

## 8.26 ~\*\*

### Description:

Command sent to all devices to say the host is OK.

### Command Syntax:

| ~** <u>[CS]</u> (CR) |                                 |
|----------------------|---------------------------------|
| ~                    | Delimiter character             |
| **                   | Command to check the host is OK |
| <u>[CS]</u>          | Checksum                        |
| (CR)                 | Carriage Return                 |

### Response:

No Response

### Examples:

Send a "Host OK" command to all the devices

Command: ~\*\* (CR)

No Response

## 8.27 ~AA0

Description:

Command to read the watchdog status of the device.

Command Syntax:

| ~ <u>AA</u> 0[ <u>CS</u> ](CR) |   |
|--------------------------------|---|
| ~                              | Delimiter character   |
| <u>AA</u>                      | Address of the device to be configured in hexadecimal format (00 to FF) |
| 0                              | Command to read the device watchdog status                              |
| [ <u>CS</u> ]                  | Checksum  |
| (CR)                           | Carriage Return   |

Response:

Valid Command: !AASS[CS](CR)

Invalid Command: ?AA[CS](CR)

|               |  |
|---------------|--|
| !             | Delimiter for a valid command  |
| ?             | Delimiter for an invalid command   |
| <u>AA</u>     | Address of the device  |
| <u>SS</u>     | Two Hexadecimal digits indicating the host watchdog status<br>SS=00 – Watchdog timeout is cleared<br>SS=04 – Watchdog timeout is set |
| [ <u>CS</u> ] | Checksum   |
| (CR)          | Carriage Return  |

Examples:

Read the host watchdog status and response is 00 meaning the host watchdog is disabled

Command: ~010(CR)

Response: !0100(CR)

Read the host watchdog status and response is 04 meaning that a host watchdog timeout has occurred.

Command: ~010(CR)

Response: !0104(CR)

## 8.28 ~AA1

Description:

Command to reset the watchdog timeout status of the device.

Command Syntax:

| <b>~<u>AA</u>1[<u>CS</u>](CR)</b> |   |
|-----------------------------------|---|
| ~                                 | Delimiter character   |
| <u>AA</u>                         | Address of the device to be configured in hexadecimal format (00 to FF) |
| 1                                 | Command to reset the watchdog timeout status                            |
| <u>[CS]</u>                       | Checksum  |
| (CR)                              | Carriage Return   |

Response:

Valid Command: **!AA[CS](CR)**

Invalid Command: **?AA[CS](CR)**

|             |                                  |
|-------------|----------------------------------|
| !           | Delimiter for a valid command    |
| ?           | Delimiter for an invalid command |
| <u>AA</u>   | Address of the device            |
| <u>[CS]</u> | Checksum                         |
| (CR)        | Carriage Return                  |

Examples:

Send command to reset the watchdog status and return valid response.

Command: ~011(CR)

Response: !01(CR)

## 8.29 ~AA2

Description:

Command to read the watchdog timeout value of the device.

Command Syntax:

| <b>~AA2[CS](CR)</b> |   |
|---------------------|---|
| ~                   | Delimiter character   |
| <u>AA</u>           | Address of the device to be configured in hexadecimal format (00 to FF) |
| 2                   | Command to read the watchdog timeout value                              |
| <u>[CS]</u>         | Checksum  |
| (CR)                | Carriage Return   |

Response:

Valid Command: **!AAEVV[CS](CR)**

Invalid Command: **?AA[CS](CR)**

|             |   |
|-------------|---|
| !           | Delimiter for a valid command   |
| ?           | Delimiter for an invalid command  |
| <u>AA</u>   | Address of the device   |
| <u>E</u>    | Watchdog enabled status<br>E=1 – Watchdog enabled<br>E=0 – Watchdog disabled  |
| <u>VV</u>   | Two hexadecimal digits representing watchdog timeout value in tenths of a second<br>01 = 0.1 seconds, FF=25.5 seconds |
| <u>[CS]</u> | Checksum  |
| (CR)        | Carriage Return   |

Examples:

Send command to read the watchdog timeout value return valid response with FF meaning the watchdog timeout value is 25.5 seconds.

Command: ~012(CR)

Response: !011FF(CR)

### 8.30 ~AA3EVV

Description:

Command to enable/disable the watchdog and set the watchdog timeout value.

Command Syntax:

| <u>~AA3EVV[CS]</u> (CR) |   |
|-------------------------|---|
| ~                       | Delimiter character   |
| <u>AA</u>               | Address of the device to be configured in hexadecimal format (00 to FF)   |
| 3                       | Command to read the watchdog timeout value  |
| <u>E</u>                | Watchdog enabled status<br>E=1 – Watchdog enabled<br>E=0 – Watchdog disabled  |
| <u>VV</u>               | Two hexadecimal digits representing watchdog timeout value in tenths of a second<br>01 = 0.1 seconds, FF=25.5 seconds |
| <u>[CS]</u>             | Checksum  |
| (CR)                    | Carriage Return   |

Response:

Valid Command: **!AA[CS]**(CR)

Invalid Command: **?AA[CS]**(CR)

|             |                                  |
|-------------|----------------------------------|
| !           | Delimiter for a valid command    |
| ?           | Delimiter for an invalid command |
| <u>AA</u>   | Address of the device            |
| <u>[CS]</u> | Checksum                         |
| (CR)        | Carriage Return                  |

Examples:

Send command to enable watchdog and set the timeout value to 25.5 seconds.

Command: ~0131FF(CR)

Response: !01(CR)

## 8.31 ~AA4V

Description:

Command to read the power on value and safe value of the device.

Command Syntax:

| <b>~AA4V[CS](CR)</b> |   |
|----------------------|---|
| ~                    | Delimiter character   |
| <b>AA</b>            | Address of the device to be configured in hexadecimal format (00 to FF) |
| 4                    | Command to read the power on and safe value                             |
| <b>V</b>             | V=P – Read power on value<br>V=S – Read safe value                      |
| <b>[CS]</b>          | Checksum  |
| (CR)                 | Carriage Return   |

Response:

Valid Command: **!AA(Data)[CS](CR)**

Invalid Command: **?AA[CS](CR)**

|               |  |
|---------------|--|
| !             | Delimiter for a valid command  |
| ?             | Delimiter for an invalid command   |
| <b>AA</b>     | Address of the device  |
| <b>(Data)</b> | For ED-588 - two digit hexadecimal value followed by 00.<br>For ED-526 – four digit hexadecimal value. |
| <b>[CS]</b>   | Checksum   |
| (CR)          | Carriage Return  |

Examples:

Send command to read the power on value. Power on value is DOut 0, DOut 2, DOut 4, DOut 6 to high and DOut 1, DOut 3, DOut 5, DOut 7 to low.

Command: **~014P(CR)**

Response: **!01F000(CR)**

Send command to read the safe value. No safe value set.

Command: **~014S(CR)**

Response: **!010000(CR)**

**Note:** This command is only available on devices with digital output channels.

## 8.32 ~AA5V

Description:

Command to set the current output value as the power on value or safe value.

Command Syntax:

| ~ <u>AA</u> 5 <u>V</u> [ <u>CS</u> ](CR) |  |
|--|--|
| ~  | Delimiter character  |
| <u>AA</u>                                | Address of the device to be configured in hexadecimal format (00 to FF)              |
| 5  | Command to set the power on and safe value   |
| <u>V</u>                                 | V=P – Set current output as power on value<br>V=S – Set current output as safe value |
| [ <u>CS</u> ]                            | Checksum   |
| (CR)                                     | Carriage Return  |

Response:

Valid Command: !AA[CS](CR)

Invalid Command: ?AA[CS](CR)

|               |                                  |
|---------------|----------------------------------|
| !             | Delimiter for a valid command    |
| ?             | Delimiter for an invalid command |
| <u>AA</u>     | Address of the device            |
| [ <u>CS</u> ] | Checksum                         |
| (CR)          | Carriage Return                  |

Examples:

Send command to set device current output as safe on value

Command: ~015S(CR)

Response: !01(CR)

Send command to set device current output as power on value

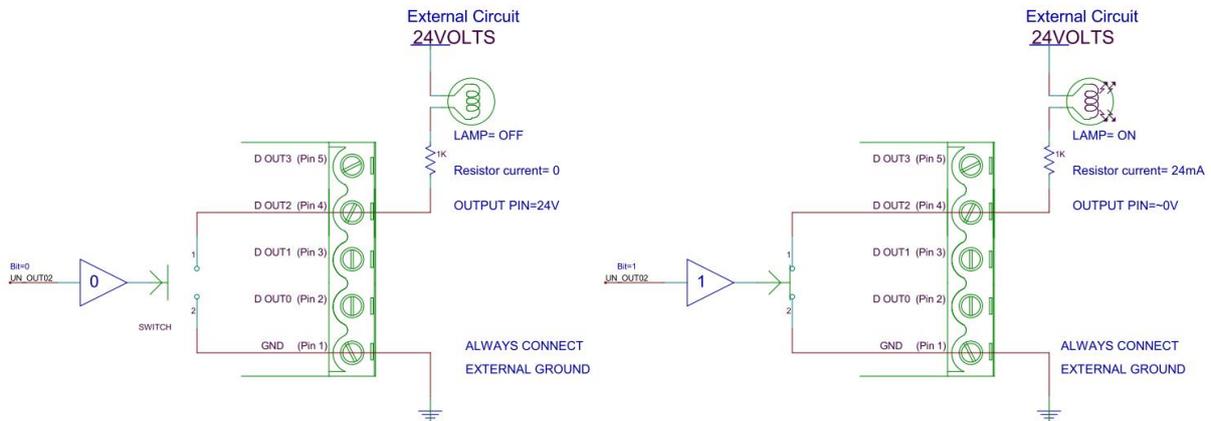
Command: ~015P(CR)

Response: !01(CR)

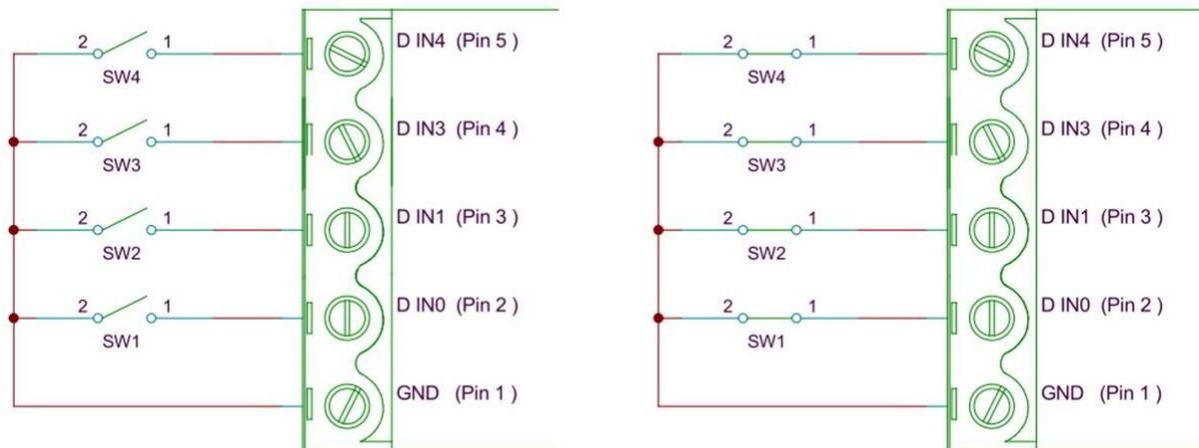
**Note:** This command is only available on devices with digital output channels.

## 9 Worked Hardware Examples

### 9.1 Output to a Light Circuit.



### 9.2 Switch Input Circuit

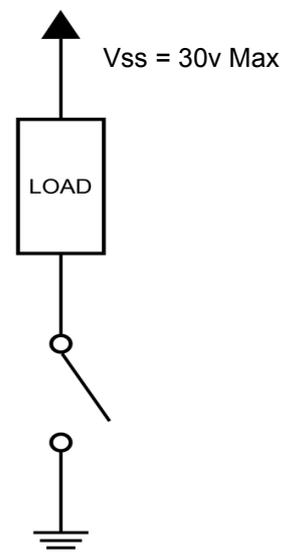


### 9.3 Digital Outputs: SINK

The ED-XXX outputs are current sinking outputs, they can easily pull the output down to ground but they cannot pull the output up.

These Sink Outputs act like a switch shorting the Digital Output to the GND pin.

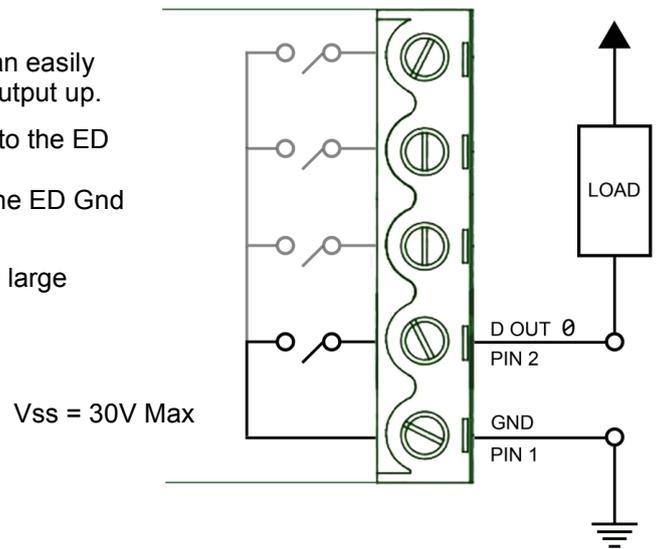
When the Output is off the external, customer supplied, circuit pulls up to its supply Voltage, V<sub>ss</sub>.



The ED-XXX outputs are current sinking outputs, they can easily pull the output down to ground but they cannot pull the output up.

The +ve supply,  $V_{ss}$ , to the load isn't related in any way to the ED power supply voltage, Ensure that the external circuit Ground is connected to the ED Gnd connection.

The Maximum Current one ED port can sink is 1 Amp so large loads can be switched.



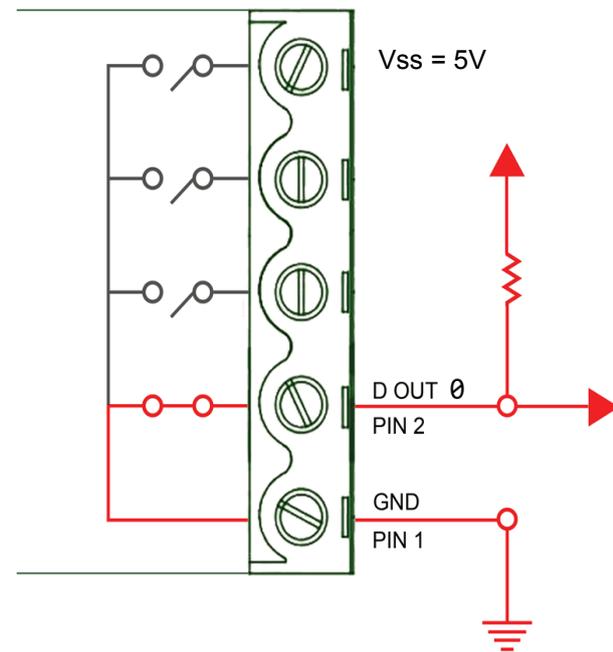
### 9.4 Digital Outputs: TTL Compatible

TTL outputs must drive below 0.35V for logic Low, and above 3.3V for logic High.

The ED outputs can drive down to below 0.35V while sinking over 1A, so the Low side of the output spec is not a problem.

ED Sink outputs do not drive high, to make the outputs TTL (or 5V CMOS) compatible, add a pull-up resistor, say 10k $\Omega$ , from the outputs to 5V.

Ensure the external circuit Ground is connected to ED GND connection.

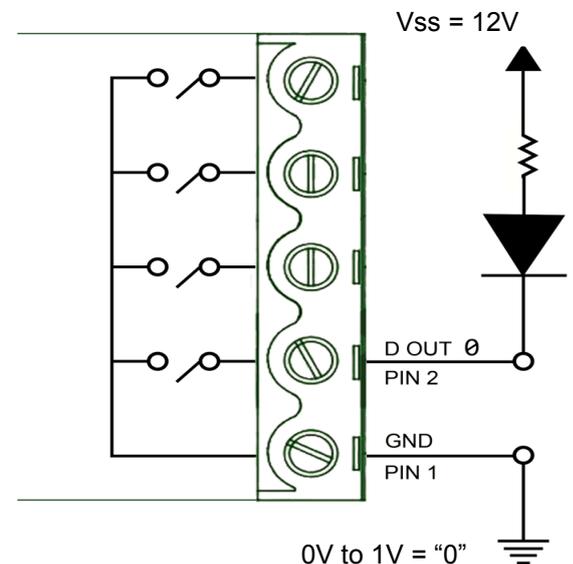


### 9.5 Turning ON Indicator Light

The ED-XXX outputs are current sinking outputs, they can easily pull the output down to ground but they cannot pull the output up.

When an LED and resistor is connected as shown the LED lights when the Output is turned ON.

12V supply, 20mA current, 500 Ohm R.



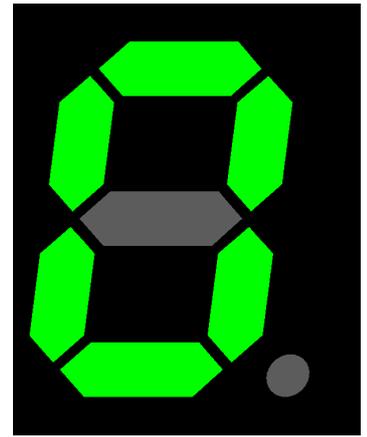
0V to 1V = "0"

The ED-XXX outputs are current sinking outputs, they can easily pull the output down to ground but they cannot pull the output up.

When RS part# 719-2522 , 7 Segment Display, is connected

12V supply,  $V_f=7.8V$  20mA current, 210 Ohm R

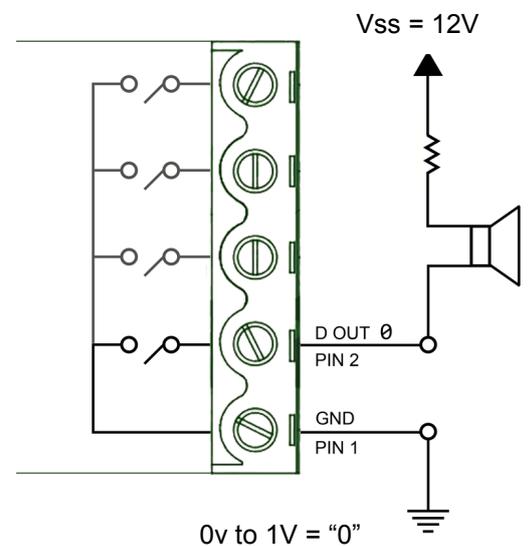
$8 \times 20mA = 160mA$  still a small load even when all 7 segments plus decimal point are lit.



### 9.6 Turning ON Indicator Sounds

The ED-XXX outputs are current sinking outputs, they can easily pull the output down to ground but they cannot pull the output up.

When a Buzzer RS Part# 754-2003, and resistor are connected as shown the Buzzer sounds when the Output is turned ON.



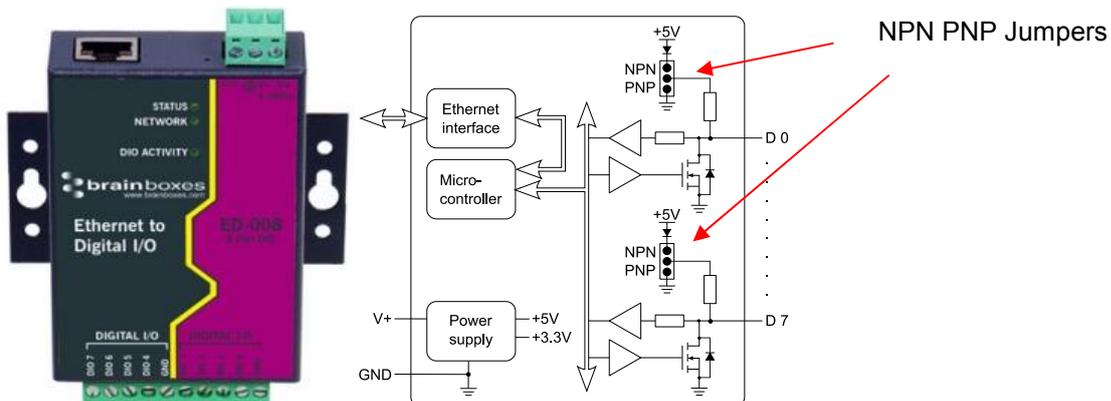
### 9.7 Digital Inputs: NPN/PNP Compatible

ED-XXX Has 8 Digital Inputs.

Each Input can be individually set to accept either NPN or PNP signals.

Demo: ED-XXX IP address is 192.168.127.253

My PC IP= 192.168.127.250

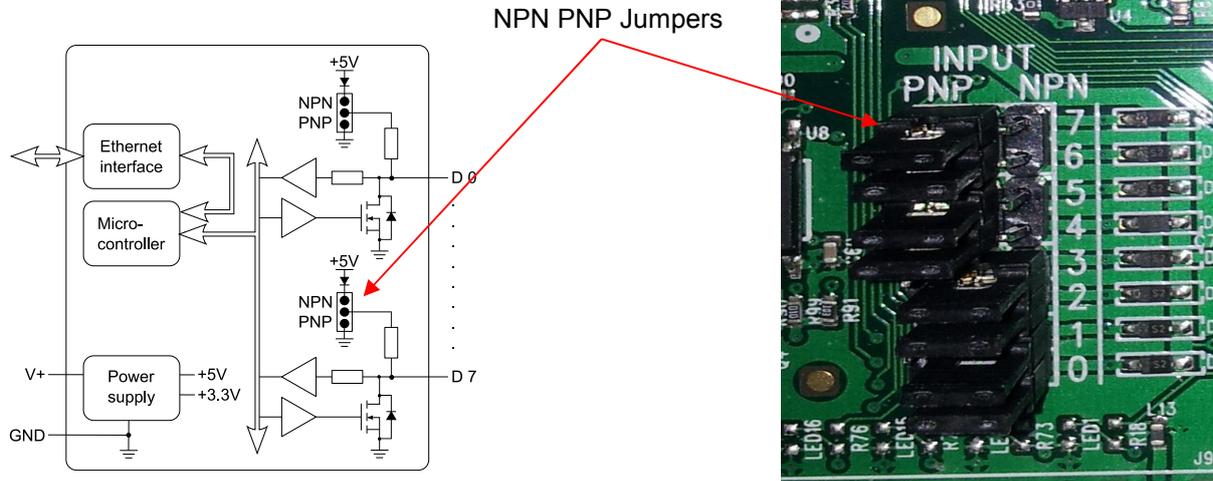


ED-XXX Has 8 Digital Inputs.

Each Input can be individually set to accept either NPN or PNP signals.

Set to LEFT For PNP.

Set to RIGHT for NPN.



### 9.8 Digital Input/Output

A NPN Sensor

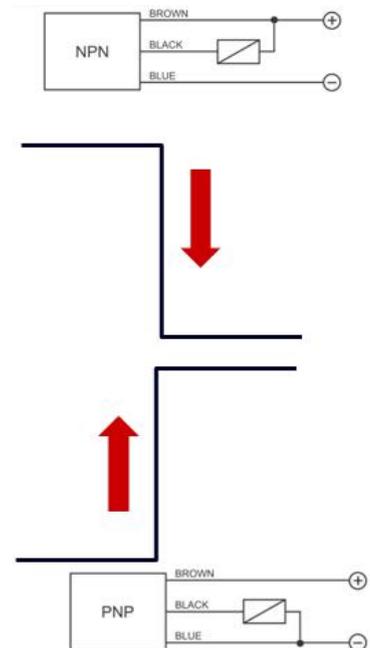
NPN Proximity Sensor Output is normally HIGH when nothing is near it

NPN Proximity Sensor Output goes LOW when metal object is near it

A PNP Sensor

PNP Proximity Sensor Output is normally LOW when nothing is near it

PNP Proximity Sensor Output goes HIGH when metal object is near it



Each sensor requires a Power Input of 12 Volts

A PNP Sensor is on DIO 5 = 020x

PNP Proximity Sensor Output is normally LOW when nothing is near it

PNP Proximity Sensor Output goes HIGH when metal object is near it

A NPN Sensor is on DIO 0 =01x

NPN Proximity Sensor Output is normally HIGH when nothing is near it

NPN Proximity Sensor Output goes LOW when metal object is near it



## 9.9 Dry Contacts

A "Dry" contact is so called because it doesn't have any voltage on it.

A switch is a typical dry contact.

Problem: If there is no voltage on it how do I tell whether its open or closed?

Switch is DI 02= Bit 4



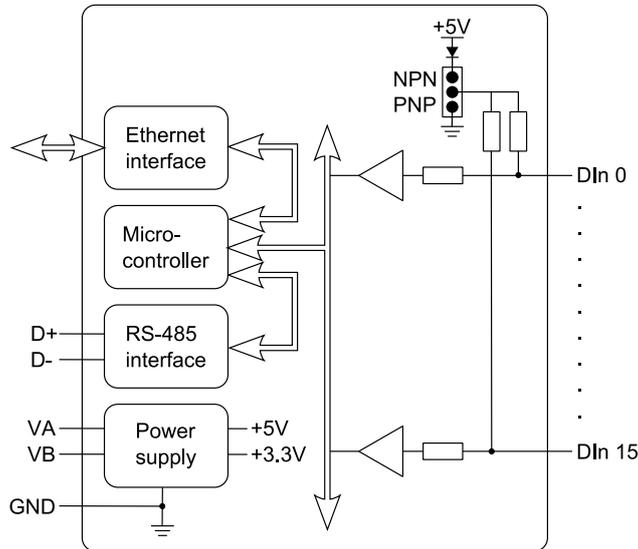
## 9.10 Detecting Dry Contacts

### Set the User Configurable PNP/NPN Jumper to NPN

Dry Contact / Volt Free:

Logic Level 0: Short to GND

Logic Level 1: Leave unconnected, inputs pull high



## 9.11 Dry Contact: PIR Detector

Inputs: A relay is a dry contact.

PIR Alarm = DIO 1 = 02x



## 9.12 Sensors & Actuators

### Sensors:

Sensors: Connect to Inputs

Sensor: a mechanical device sensitive to light, temperature, radiation level, or the like, that transmits a signal to a measuring or control instrument.

### Actuators:

Actuators: Connect to Outputs

What can I Control?

Actuator: a servomechanism that supplies and transmits a measured amount of energy for the operation of another mechanism or system.

## 10 Lifetime Warranty and Support

To receive the lifetime Warranty, you need to register your product with us using our online form.

**NB: this must be done within 28 days of Purchase.**



[Lifetime Warranty Sign up](#)

\* Terms and Conditions are available online. Standard warranty period is 3 years if a product is not registered.



Since 1983, Brainboxes have designed, tested and manufactured our products all under one Roof. One of our greatest strengths is in after sales service. Technical Support is provided by members of our Test Team, who know our products inside out and have direct access to the chip and board designers as well as the technicians who built and tested your product.

If you have any issues, questions or suggestions about our Products and Services, then please contact us.

Technical Support is free\*. As long as you have a Brainboxes Product we will be happy to help, even if it's discontinued or out of warranty. Excellent Customer Service, just as it should be.

For the quickest solution to your issue, if you email us, please include as much detail of your setup and the fault you are experiencing.

\* Standard rate call charges for phone support apply.

### Email

Technical Support: [support@Brainboxes.com](mailto:support@Brainboxes.com)

Sales Enquiries: [sales@Brainboxes.com](mailto:sales@Brainboxes.com)

### Telephone

You can speak to Brainboxes Support or Sales teams direct,

**Monday – Friday, 9am to 5pm (UK time)**

Tel: +44 (0)151 220 2500

# 11 Regulatory Approvals / Compliance

For up to date details of global certifications, please check the product datasheet on the Brainboxes website: [www.brainboxes.com](http://www.brainboxes.com)

## 11.1 Company Accreditation

Brainboxes is accredited to internationally recognised standards for our Quality and Environmental Management Systems. Our ISO9001 Quality System was first accredited in 1994, followed by our ISO14001 Environmental System in 2008. These standards help ensure we can demonstrate effective management of all our quality systems and our environmental impacts, together with a process of continuous improvement.

All our Quality systems are subject to internal and external assessment on a regular basis. Copies of Certificates are available for download from our website: [www.brainboxes.com](http://www.brainboxes.com)

Linked with our Lean and Six Sigma techniques, we believe we have the most reliable products on the market, and to back this up we are offering a Lifetime Warranty\* on all our Serial Products.

## 11.2 Europe – EU Declaration of Conformity

Brainboxes products are designed to conform to the protection requirements of European Council Directive 2004/108/EC and its subsequent revisions.

A Declaration of Conformity and supporting Technical Construction File is available by request from Brainboxes, and will identify any updated legislation that may have been introduced since the publication of this Manual.

## 11.3 WEEE Directive (Waste Electrical and Electronic Equipment)

The Waste Electric and Electronic Equipment (WEEE) Regulations 2013 became law in the UK on the 1st of January 2014 and replaced the 2006 Regulations. Brainboxes is fully compliant with this legislation.

### Customer Responsibilities

You are encouraged to dispose of WEEE in an environment friendly way.

This can be done through your local civic amenities site, an approved treatment facility or alternatively through a relevant compliance scheme.

### Brainboxes' Responsibilities

Brainboxes has a legal responsibility, as producer, to provide a free of charge collection service to our customers for our obligated WEEE.

Brainboxes is defined as a producer under the WEEE regulations because we sell own brand Electrical & Electronic Equipment (EEE) in the UK. Our WEEE Producer Registration Number is WEE/AH0004XR.

For details of our WEEE recovery service options, please see our Website, or email us at:

[weeerecovery@Brainboxes.com](mailto:weeerecovery@Brainboxes.com)

## 11.4 RoHS Compliance

All Brainboxes Serial and Bluetooth products are fully RoHS compliant.

Brainboxes identified at an early stage the importance of rapid compliance to RoHS guidelines and established a project team to actively manage the transition. The initial step in the process was to use our close relationships with suppliers to ensure early access to RoHS compliant components for all of our Bluetooth and Serial Products. In addition, the project team worked to ensure that our manufacturing processes meet all RoHS requirements well in advance of the deadline.



To verify supplier declarations on RoHS compliancy, we have also sent fully built products to an external test house for X-Ray system XDAL. This technique is capable of determining percentages of different elements and is accurate to 0.1% Wt.

RoHS Compliant Brainboxes products have been available since January 2005.

### What is the RoHS Directive?

The Restriction of the Use of Certain Hazardous Substances (RoHS) in Electrical and Electronic Equipment (EEE) Directive ([2011/65/EU](#)) was transposed into UK law on 2 January 2013.

This legislation bans the placing on the EU market of new EEE containing more than the agreed levels of:

- lead (Pb)
- cadmium (Cd)
- mercury (Hg)
- hexavalent chromium (Cr6+)
- polybrominated biphenyls (PBB)
- polybrominated diphenyl ethers (PBDE)

Any future revisions to this legislation will be complied with, and identified in a Declaration of Conformity, available on request.

## 12 Copyright

### Copyright © Brainboxes Ltd

All rights reserved. No part of this hardware, circuitry or manual may be duplicated, copied, transmitted or reproduced in any way without the prior consent of the Manufacturer.