Multi-IO HAT for RASPBERRY PI

USER'S GUIDE VERSION 1.3

SequentMicrosystems.com

GENERAL DESCRIPTION

The Multi-IO Card is a complex HAT which brings to Raspberry Pi a rich set of analog and digital inputs and outputs, two serial ports, two RTD inputs, an H-Bridge motor driver, two relays and two servo ports for robotic applications. Six programmable LED's can be associated with any input or output and triggered on a programmable signal threshold. The card is compatible with all Raspberry Pi versions from Zero to 5.

Pluggable connectors make the card easy to use. Loads of up to 8A and 24V can be switched by the relays. Status LEDs show when relays are on or off. A LED also shows when power is applied to the board. A pushbutton permits the user to issue any general command to Raspberry Pi or to shut it down.

Two 3-Wire RTD ports using 24-bit A/D converters can be used for precise temperature measurements.

The RS485 port can communicate with other industrial equipment using the MODBUS protocol. It can be driven either by the local processor or directly from Raspberry Pi. An RS232 transceiver is also connected to one of Raspberry Pi's USARTS.

Mechanically, the Multi-IO Card adheres to the Sequent Microsystems Modular Industrial format. It can be installed in the free 3D printable stackable enclosure. All the cards in this format have the same mechanical specifications.

FEATURES

- Multi-IO 8-Layer Stackable HAT for Raspberry Pi
- Uses only I2C port, works with all Raspberry Pi versions from Zero to 5
- Wide range 10-30V power supply provides also 5V/3A to Raspberry Pi
- Two opto-isolated or four common ground digital inputs
- Two 3-wire RTD ports with 24 bit delta-sigma A/D converters
- Two 0-10V analog inputs 200 sps
- Two 4-20mA analog inputs 200 sps
- Two 0-10V analog outputs
- Two 4-20mA analog outputs
- Two 5A/24V relays with normal-open contacts and status LEDs
- Two communication ports: RS485/Modbus and RS232
- H-Bridge PWM Motor Driver
- Two servo control ports
- Nine Status LEDs (six general purpose)
- On-board push-button
- Hardware watchdog
- Real Time Clock with battery backup
- Pluggable Connectors
- Stand-alone operation (No Raspberry Pi) using RS485/MODBUS
- On-board 5V/2.5A step-down power supply for Raspberry Pi
- Eight layer stackable for IO expansion
- Command Line Driver, Python Library, Node-Red nodes

ERRATA: The serial ports marking on the silkscreen is reversed. The lower port is RS485. The upper port is RS232.

All inputs and output use pluggable connectors which permit easy wiring access when multiple cards are stacked. Up to eight Multi-IO Cards can be stacked on top of one Raspberry Pi. The cards share a serial I2C bus using only two of the Raspberry Pi's GPIO pins to manage all eight cards. This feature leaves the remaining 24 GPIOs available for the user.

The six general purpose LED's can be associated with the analog inputs or other controlled processes. An on-board push button can be programmed to cut inputs, override outputs or shut down the Raspberry Pi

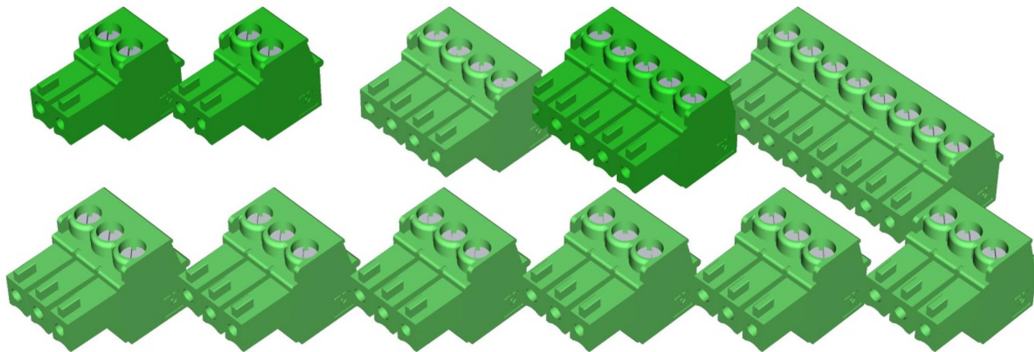# WHAT IS IN YOUR KIT

1. Multi-IO HAT for Raspberry Pi
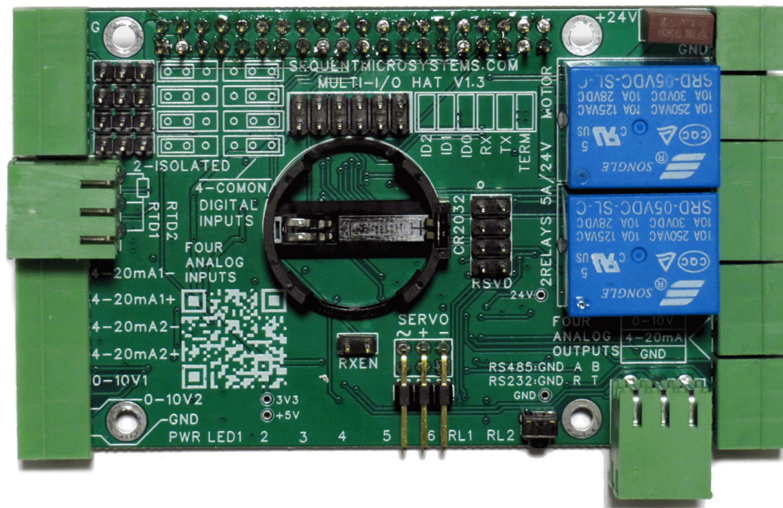
2. Mounting hardware

    a. Four M2.5x18mm male-female brass standoffs
    b. Four M2.5x5mm brass screws
    c. Four M2.5 brass nuts
    d.

3. All the required female mating connectors.

QUICK START-UP GUIDE

1. Plug your Building Automation Card on top of your Raspberry Pi and power up the system.
2. Enable I2C communication on Raspberry Pi using raspi-config.
3. Install the software from github.com:
   a. ~$ git clone https://github.com/SequentMicrosystems/multiio-rpi.git
   b. ~$ cd /home/pi/multiio -rpi
   c. ~/multiio-rpi$ sudo make install
4. ~/multiio-rpi$ multiio

   The program will respond with a list of available commands.

BOARD LAYOUT
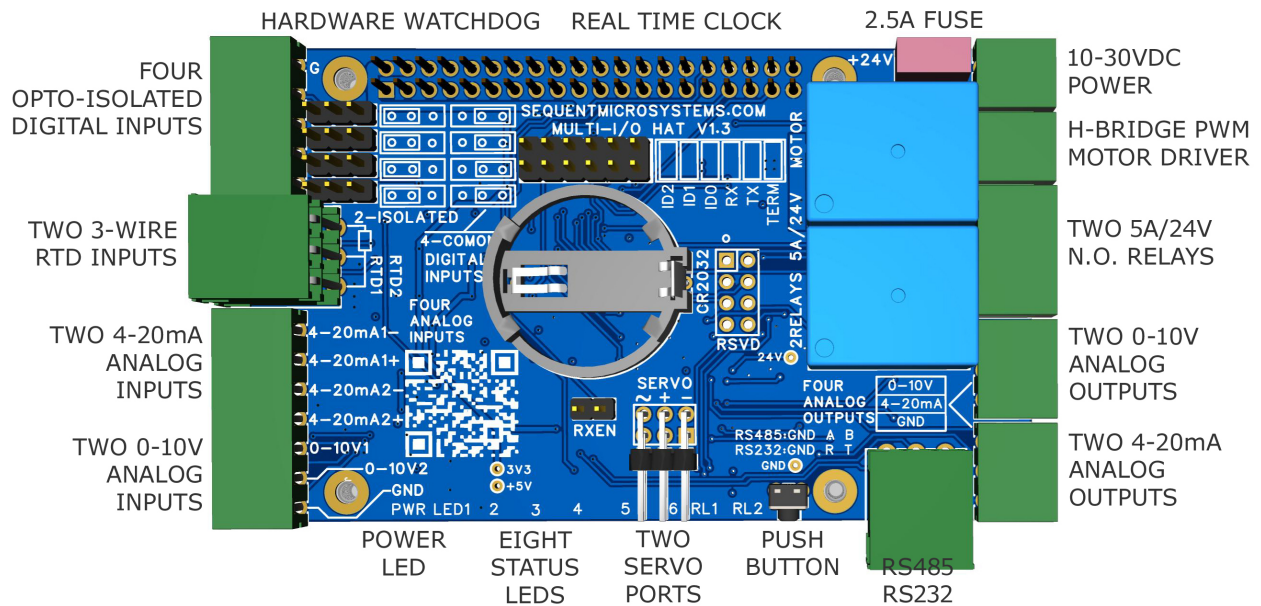
HARDWARE WATCHDOG

The Multi-IO Card contains a built-in hardware watchdog which will guarantee that your mission-critical project will continue running even if Raspberry Pi software hangs up. After power up the watchdog is disabled, and becomes active after it receives the first reset.

The default timeout is 120 seconds. Once activated, if it does not receive a reset from Raspberry Pi within 2 minutes, the watchdog cuts the power and restores it after 10 seconds.

Raspberry Pi needs to issue a reset command on the I2C port before the timer on the watchdog expires. The timer period after power up and the active timer period can be set from the command line. The number of resets is stored in flash and can be accessed or cleared from the command line. All the watchdog commands are described by the online help function

DISPLAY LEDs

Nine LEDs on the bottom edge of the board show you the status of the inputs and outputs. The leftmost LED shows when the power is connected to the board.

The six middle LEDs are general purpose. They can be connected in software to show the status of any input or output. They can also be programmed to light at a certain input or output level. For example, you can learn how to turn on a LED when a temperature reaches a threshold. You can also blink a led slower when a temperature is relatively low, and faster as it approaches a preset limit.

The two rightmost LEDs are routed to the corresponding relays and show when the relay coil is activated. A LED is lit when the corresponding relay is energized. The General Purpose Pushbutton PB1 is connected to pin 37 of the Raspberry Pi GPIO connector (GPIO26).

RS-485 COMMUNICATION

The Multi-IO Card contains a standard RS485 transceiver which can be accessed both by the local processor and by Raspberry Pi. The desired configuration is set on configuration connector J2.
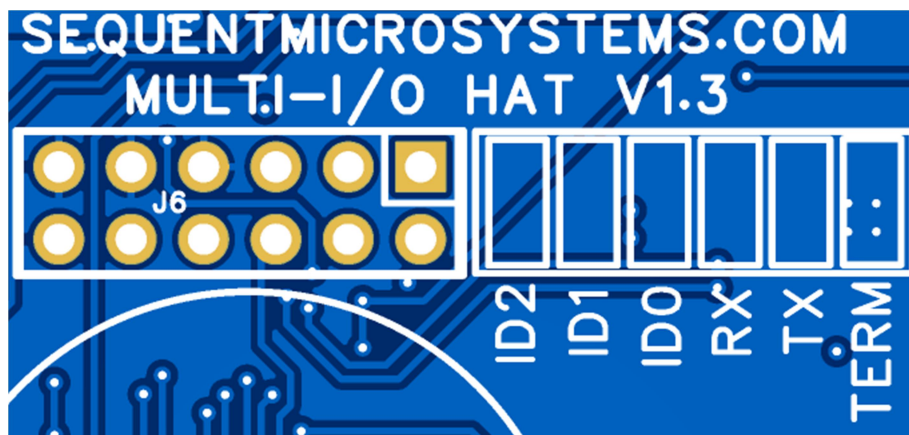
The right-most jumper labeled TERM will shunt the A and B RS485 lines with a 120 ohms resistor. Install this jumper if the Multi-IO Card is the last one on your RS485 chain.

The next two jumpers labeled RX and TX are routing the RS485 port directly to Raspberry Pi. In this configuration the Multi-IO Card is a passive bridge which implements only the hardware levels required

by the RS485 protocol. To use this configuration, you need to tell the local processor to release control of the RS485 bus:

~$ multi-io [0] wcfgmb 0 0 0 0

If jumpers are removed, the card operates as MODBUS slave and implements the MODBUS RTU protocol. Any MODBUS master can access all the card's inputs, and set all the outputs using standard MODBUS commands. A detailed list of commands implemented can be found on GitHub:

https://github.com/SequentMicrosystems/multiio-rpi/blob/main/MODBUS.md

In both configurations the local processor needs to be programmed to release (jumpers installed) or control (jumpers removed) the RS485 signals. See the command line online help for further information.

STACK LEVEL SETTINGS

Up to eight Multi-IO Cards can be installed on top of one Raspberry Pi. The card is controlled by Raspberry Pi using only the I2C interface. The card occupies the address space 0x20 - 0x27. The local address can be configured using the three left positions of the J2 jumper block. A maximum stack of eight cards requires a total of 12 jumpers. Cards can be installed on Raspberry Pi in any order.

JMP/SW	OFF	OFF	OFF		ON	OFF	OFF		OFF	ON	OFF		ON	ON	OFF
STACK LEVEL	0				1				2				3		
I2C ADDR.	0x27				0x26				0x25				0x24		

JMP/SW	OFF	OFF	ON		ON	OFF	ON		OFF	ON	ON		ON	ON	ON
STACK LEVEL	4				5				6				7		
I2C ADDR.	0x23				0x22				Ox21				Ox20		

To find out the stack level of all the cards installed in your system run the command line with the "-list" option.

POWER REQUIREMENTS

The Multi-IO Card requires an external 24VDC regulated power supply. Power is supplied to the board through a 2 pin pluggable connector located in the upper right corner of the board. When power is applied to the board, the power led is flashing to indicate normal operation. The top pin is the positive. An on-board 2.5A fuse protects your power supply and wires from accidental shorts. The fuse is not resettable. The board is not protected if reverse polarity power is applied.


A local regulator supplies up to 3A/5V power to Raspberry Pi, and a 3.3V regulator powers the digital circuits. Another local regulator supplied 4V power to the RTD inputs.

If multiple Multi-IO Cards are stacked on top of each other, we recommend using a single 24VDC power supply to power all the cards. The user must split the cable and run the wires to each card.

DIGITAL INPUTS CONFIGURATION

Optically isolated digital inputs can be configured as 4 inputs with common ground, or 2 isolated inputs. Four 3-pin jumpers in the upper left corner of the board are provided to change the digital inputs configuration. All four jumpers have to be installed in the same position (all left or all right)


TWO/FOUR
OPTO-ISOLATED
DIGITAL INPUTS
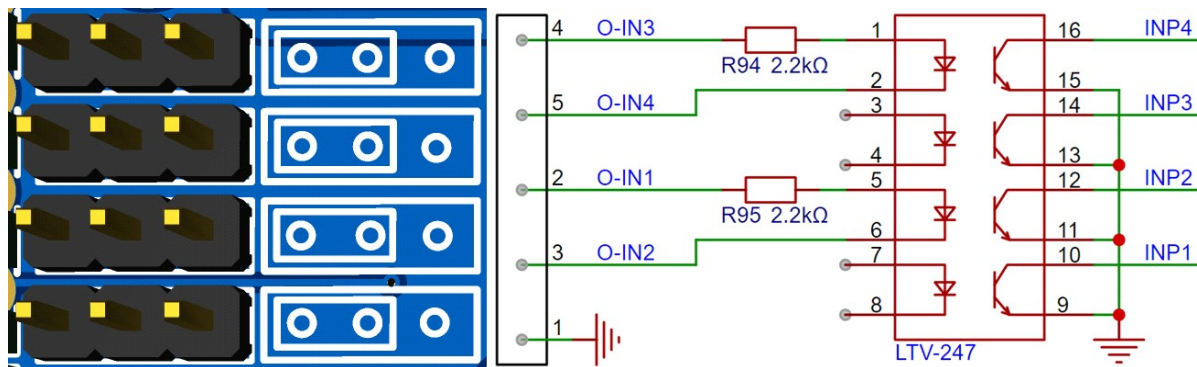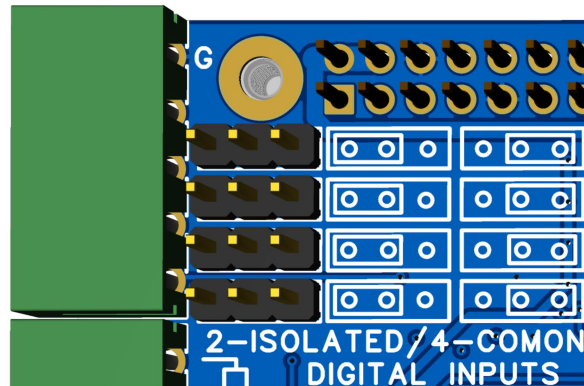
Jumpers installed to the left side, two isolated inputs
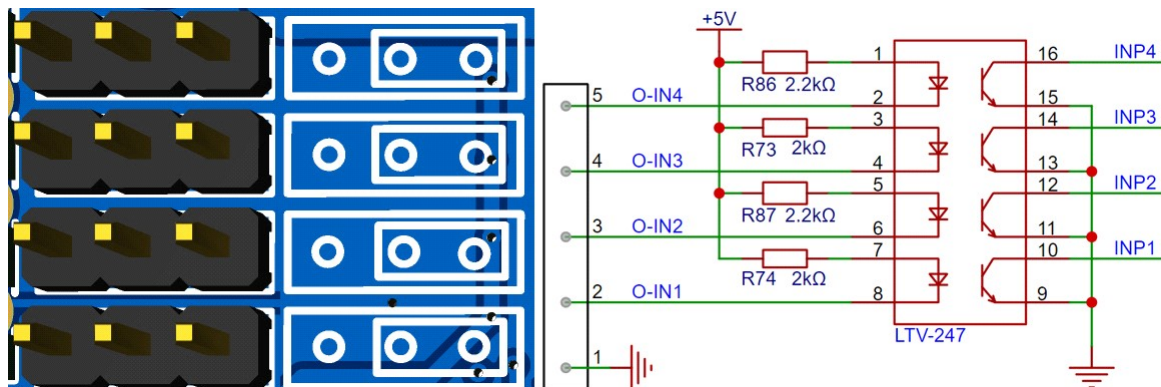
Jumpers installed to the right side, four inputs with common ground

RTD INPUTS

Two RTD Inputs using 24 bit delta sigma operational amplifiers can be used for precision temperature reading with 3-wire PT100 probes. The inputs share a 2-level 3 pin pluggable connector. Wire the PT100 probes as shown on the card's silkscreen:
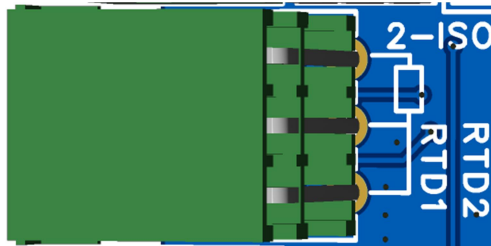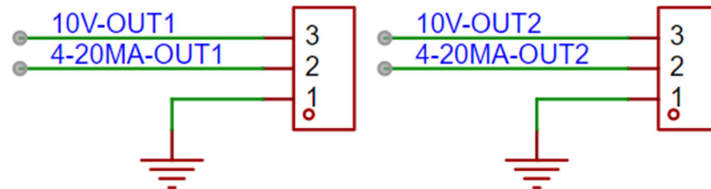

ANALOG INPUTS

Two 4-20mA inputs and two 0-10V inputs share a 7-pin 3.5mm pitch pluggable connector.
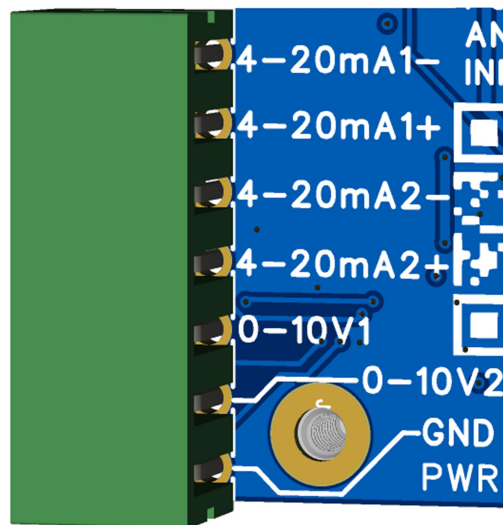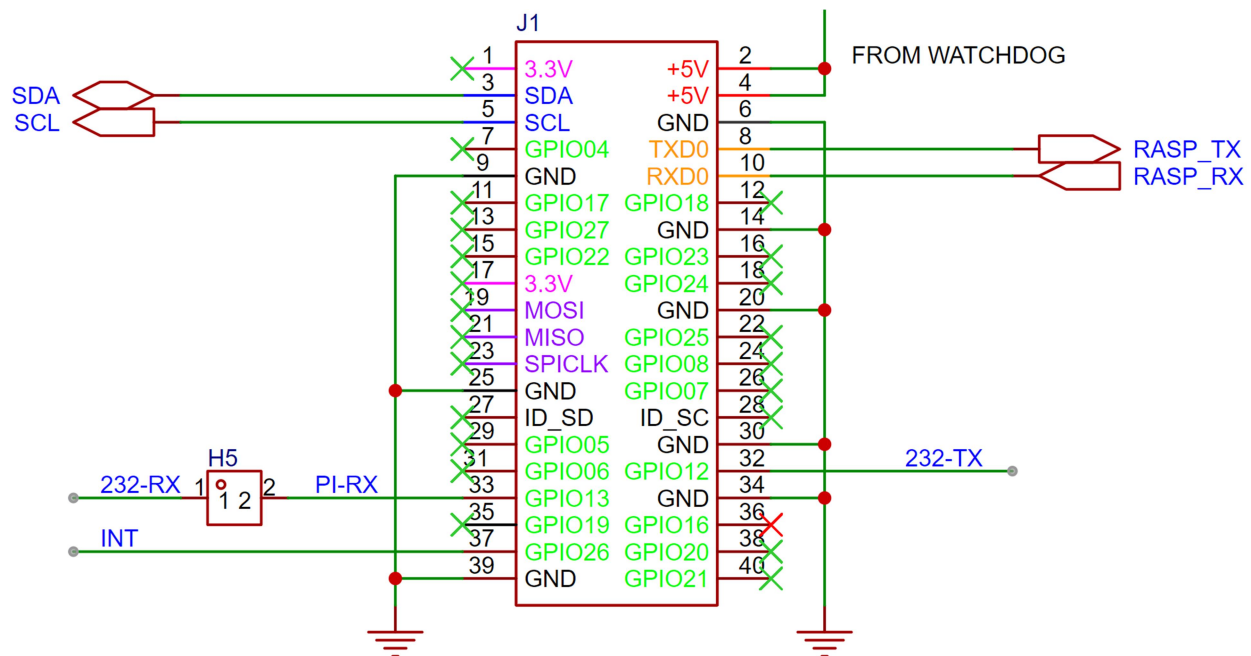
ANALOG OUTPUTS

Two 0-10V and two 4-20mA analog outputs share two 3-pin pluggable connectors in the lower right corner of the board:

GPIO CONNECTOR

The Multi-IO Card uses only the I2C port to communicate with Raspberry Pi. In addition to the two I2C pins, the following GPIO pins can be used if needed:

TXD0/RXD0 (pins 8 and 10) are routed the RS485 transceiver through jumpers. Set the jumpers ON to drive the RS485 port from Raspberry Pi.

GPIO12/GPIO13 (pins 32 and 33) are routed to the RS232 transceiver. Pin 33 (RX) is routed through a jumper. Set the jumper on to drive the RS485 port from Raspberry Pi.

GPIO26 is designed as an interrupt and is routed to the local processor. You can set the pin in software to trigger an interrupt when a specified condition of the Multi-IO Card inputs occurs.

MECHANICAL SPECIFICATIONS

 Mechanically, the Multi-IO Card adheres to the Sequent Microsystems Modular Industrial format. It can be installed in the free 3D printable stackable enclosure. All the cards in this format have the same mechanical specifications.
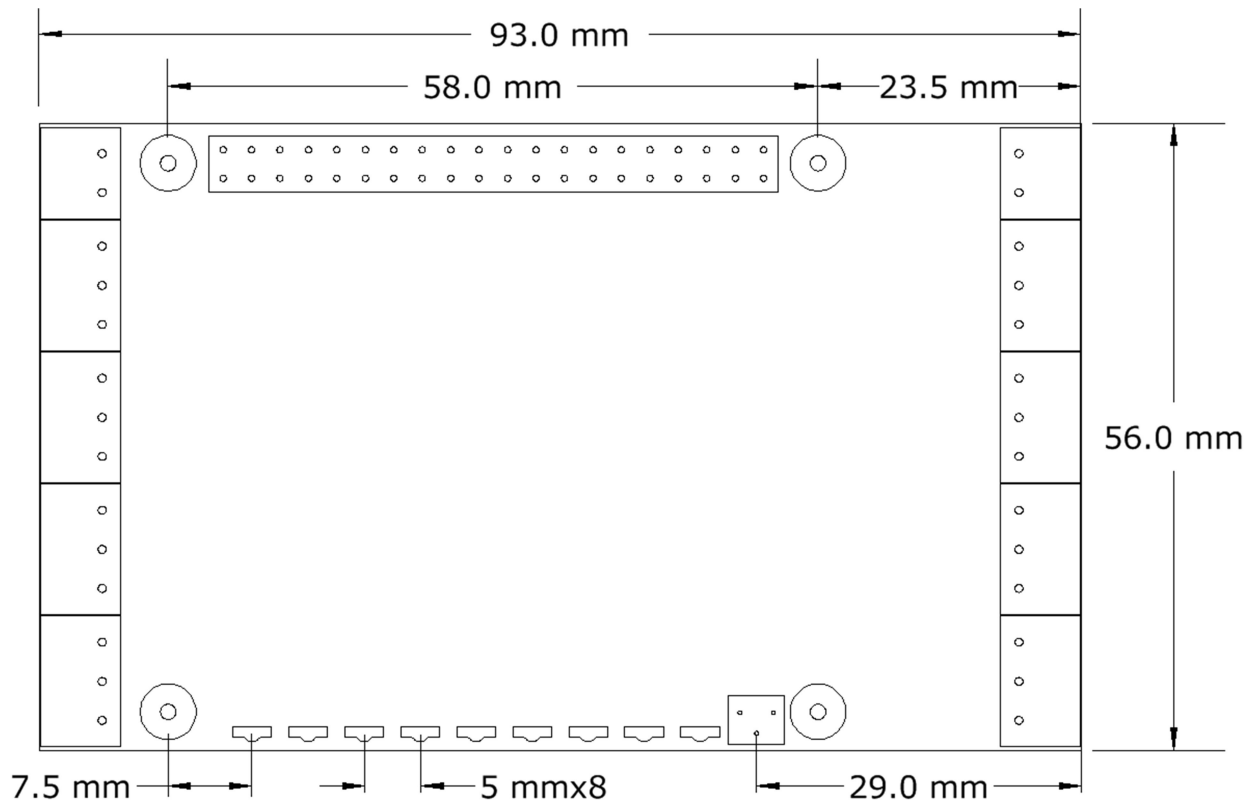
ANALOG INPUTS/OUTPUTS CALIBRATION

All the analog inputs and outputs are calibrated at the factory, but firmware commands permit the user to re-calibrate the board, or to calibrate it to better precision. All inputs and outputs are calibrated in two points. Select the two points as close to possible to the two ends of scale. To calibrate the inputs, the user must provide analog signals. (Example: to calibrate 4-20mA inputs, the user must provide a 4mA and 20mA current source). To calibrate the outputs, the user must issue a command to set the output to a desired value, measure the result and issue the calibration command to store the value.

The values are stored in flash and the input curve is assumed to be linear. If a mistake is made during calibration by typing the wrong command, a RESET command can be used to reset all the channels in the corresponding group to factory values. After RESET the calibration can be restarted.

The board can be calibrated without a source of analog signals, by calibrating first the outputs and then routing the calibrated outputs to corresponding inputs. The following commands are available for calibration:

CALIBRATE 0-10V INPUTS: multiio <id> uincal <value>

RESET CALIBRATION OF 0-10V INPUTS: multiio <id> uncalrst

CALIBRATE 4-20mA INPUTS: multiio <id> iincal <value>

RESET CALIBRATION OF 4-20mA INPUTS: multiio <id> iincalrst

CALIBRATE 0-10V OUTPUTS: multiio <id> uoutcal <value>

RESET CALIBRATION OF 0-10V OUTPUTS: multiio <id> uoutcalrst

CALIBRATE 4-20mA OUTPUTS: multiio <id> ioutcal <value>

RESET CALIBRATION OF 4-20mA INPUTS: multiio <id> ioutcalrst

SOFTWARE SETUP

1. Have your Raspberry Pi ready with the latest OS.

2. Enable I2C communication:

~$ sudo raspi-config

1. Change User Password	Change password for default user
2. Network Options	Configure network settings
3. Boot Options	Configure options for start-up
4. Localisation Options	Set up language and regional settings to match..
5. Interfacing Options	Configure connections to peripherals
6. Overclock	Configure overclocking for your Pi
7. Advanced Options	Configure advanced settings
8. Update	Update this tool to the latest version
9. About raspi-config	Information about this configuration

P1	Camera	Enable/Disable connection to the Raspberry Pi Camera
P2	SSH	Enable/Disable remote command line access to your Pi
P3	VNC	Enable/Disable graphical remote access to your Pi using...
P4	SPI	Enable/Disable automatic loading of SPI kernel module
P5	I2C	Enable/Disable automatic loading of I2C kernel module
P6	Serial	Enable/Disable shell and kernel messages to the serial port
P7	1-Wire	Enable/Disable one-wire interface
P8	Remote GPIO	Enable/Disable remote access to GPIO pins

3. Install the multi-io software from github.com:

~$ git clone https://github.com/SequentMicrosystems/multiio-rpi.git

4.　　　~$ cd /home/pi/multiio-rpi

5.　　　~/multiio-rpi$ sudo make install

6.　　　~/multiio-rpi$ multiio

The program will respond with a list of available commands.

Type "multiio -h" for online help.

After installing the software, you can update it to the latest version with the commands:

~$ cd /home/pi/multiio-rpi

~/multiio-rpi$ git pull

~/multiio-rpi$ sudo make install