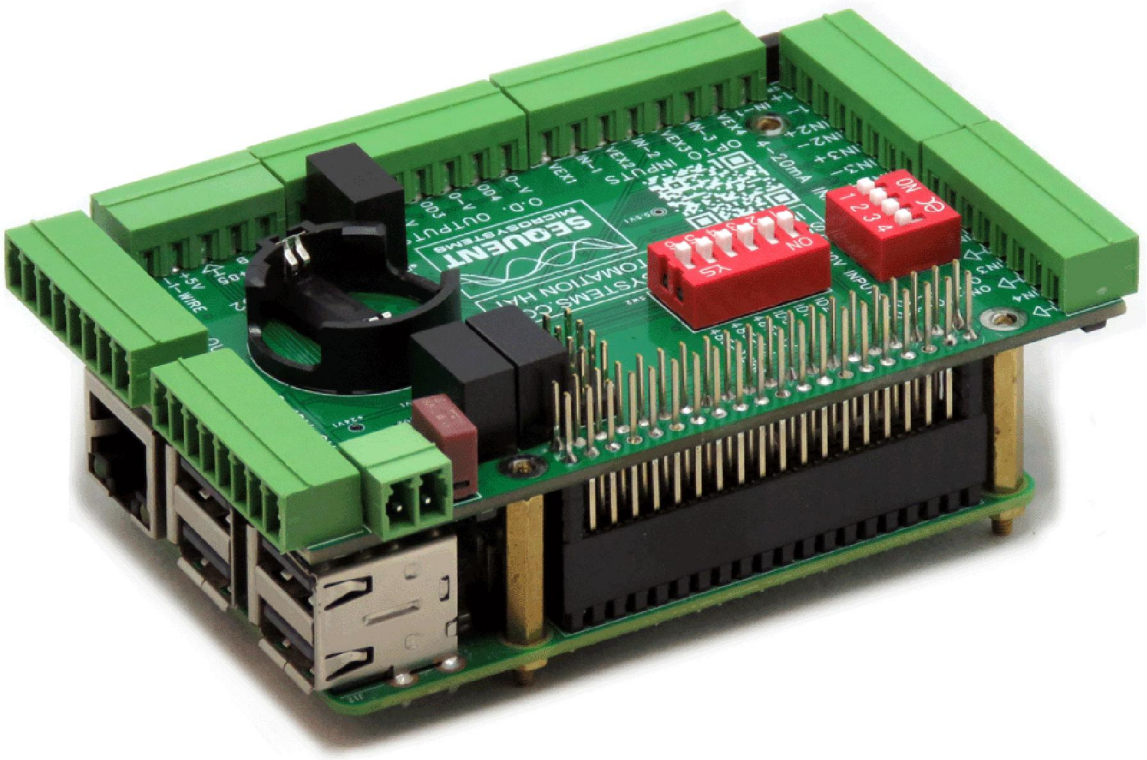
INDUSTRIAL AUTOMATION Card for RASPBERRY Pi

USER'S GUIDE VERSION 3.2

SequentMicrosystems.com

GENERAL DESCRIPTION



The second generation of our Industrial Automation card brings to the Raspberry Pi platform all the building blocks required for Industrial Automation projects. Stackable to 8 levels, the card works with all Raspberry Pi versions, from Zero to 4.

Two of the Raspberry Pi's GPIO pins are used for I2C communication. Another pin is allocated for the interrupt handler, leaving 23 GPIO pins available for the user.

Read four sensors with optically isolated, 12 bit 4-20mA inputs, and control four actuators using 4-20mA outputs. Read four sensors with 12-bit 0-10V or -10V to +10V inputs, and control four actuators with 13 bit 0-10V outputs. Drive four heavy-duty 24VDC/4A loads with MOSFET outputs. Read four optically isolated digital inputs. Program four LEDs to display the state of any digital or analog input or output. Keep time indefinitely even during a power failure using the Real-Time Clock with battery backup. Activate the hardware watchdog to monitor and power cycle the industrial Raspberry Pi in case of software lockup.

TVS diodes on all inputs protect the card for external ESD. Onboard resettable fuse protects it from accidental shorts.

RS485 and 1-Wire ports for communication

FEATURES

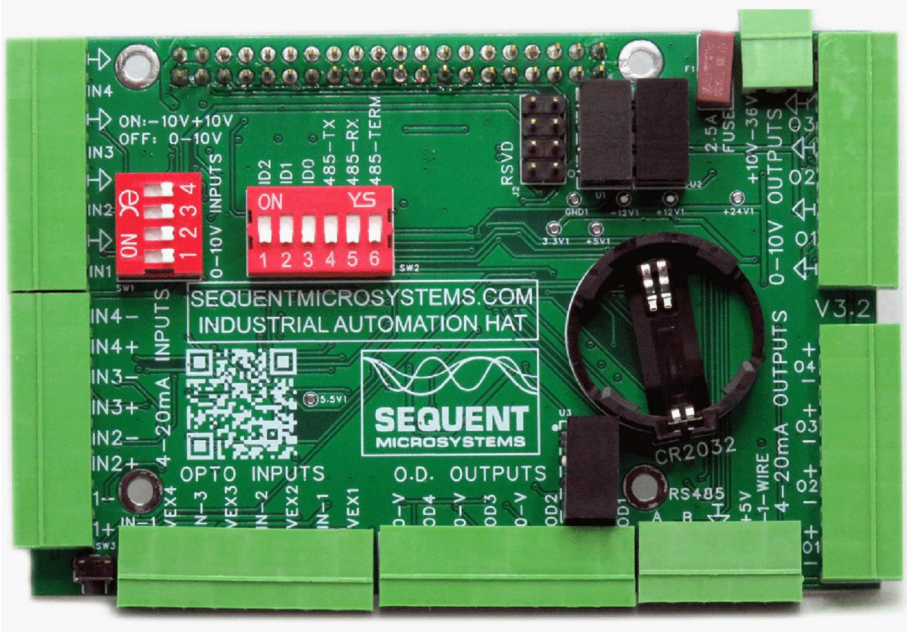
- Four Optically Isolated Digital Inputs with status LEDs
- Four 0 - 10V or ±10V Analog Inputs
- Four Optically Isolated 4-20mA Inputs
- Four Optically Isolated Open Drain Outputs, 24V/4A peak
- Four 0-10V Analog Outputs
- Four 4-20mA Analog Outputs
- Four General Purpose LEDs
- General Purpose push-button
- TVS Protection on all Inputs
- RS485 and 1-Wire Ports, MODBUS accessible without Raspberry Pi
- Wide Input 10-30VDC Power Supply
- On Board Hardware Watchdog
- On Board Resettable Fuse
- Real Time Clock With Battery Backup
- Pluggable Connectors 26-16 AWG wires
- Eight Level Stackable
- All mounting hardware included: brass stand-offs, screws and nuts
- Open source hardware and schematics
- Command Line
- Python Library
- Node-Red nodes
- MODBUS interface

All inputs and outputs use pluggable connectors which permit easy wiring access when multiple cards are stacked. Up to eight Industrial Automation 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 four general purpose LED's can be associated with the analog inputs or other controlled processes.

WHAT IS IN YOUR KIT

1. Industrial Automation Card for Raspberry Pi



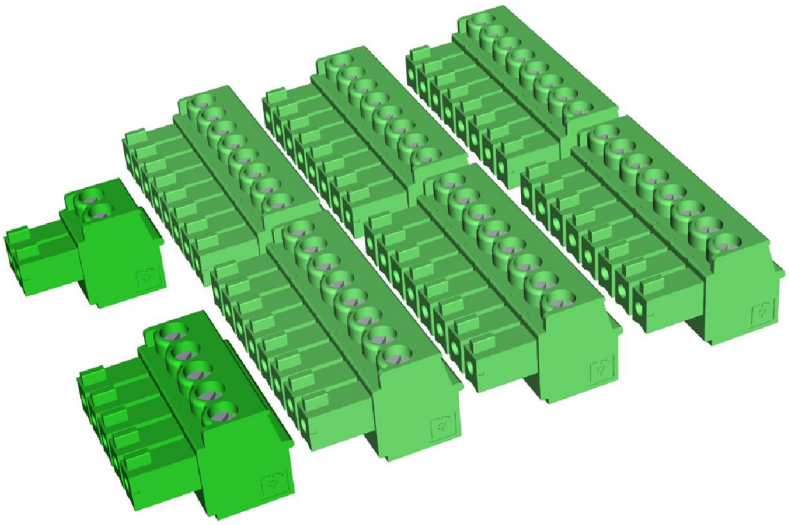
2. Mounting hardware

   a. Four M2.5x19m male-female brass standoffs
   b. Four M2.5x5mm brass screws
   c. Four M2.5 brass nuts

3. Two jumpers.

   You do not need the jumpers when using only one Industrial Automation Card. See STACK LEVEL JUMPERS section if you plan to use multiple Industrial Automation Card.

4. All the required female mating connectors.

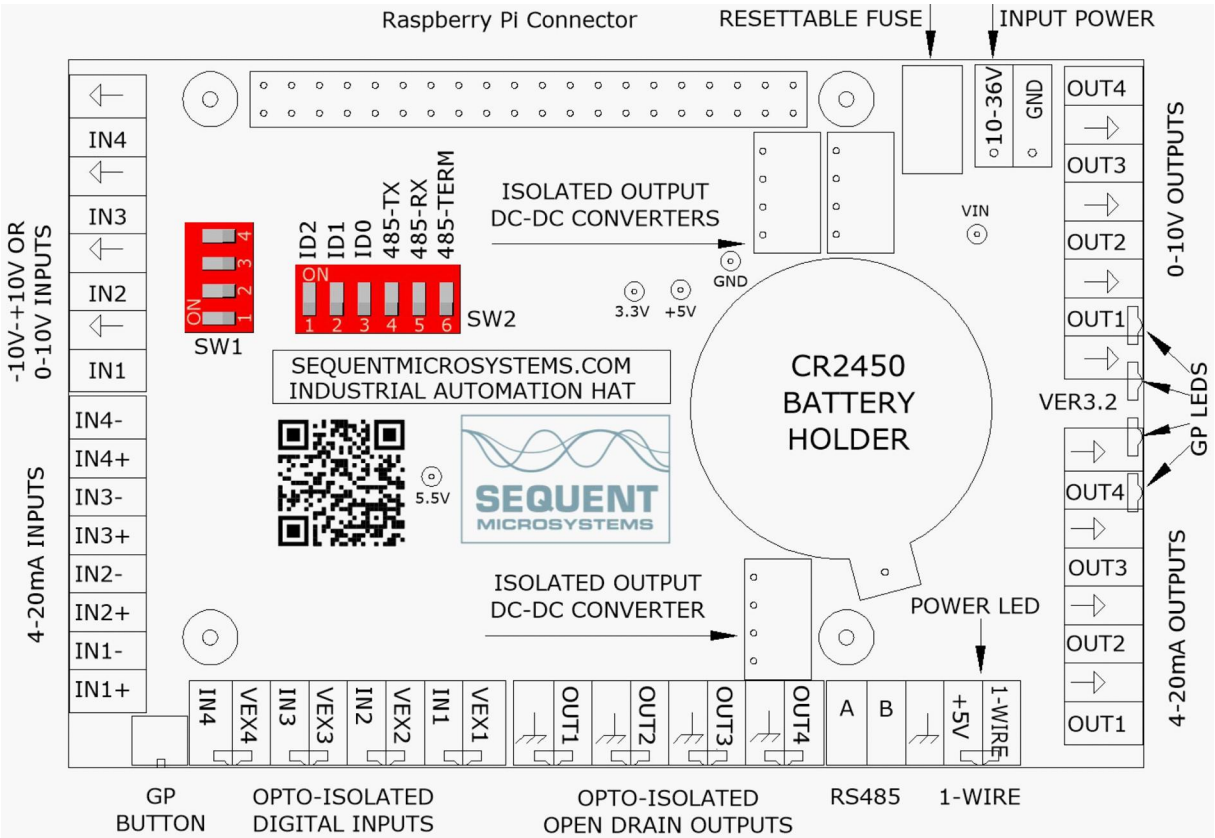


QUICK START-UP GUIDE

1. Plug your Industrial 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/megaind-rpi.git
    b. ~$ cd /home/pi/megaind-rpi
    c. ~/megaind-rpi$ sudo make install
4. ~/megaind-rpi$ megaind

    The program will respond with a list of available commands.

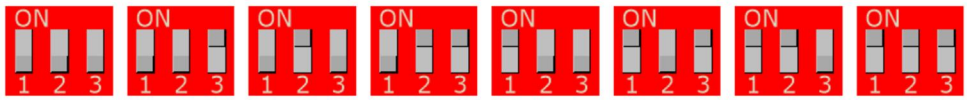
BOARD LAYOUT



Four General Purpose LEDs can be controlled in software. The LEDs can be activated to show the status of any input, output or external process.

STACK LEVEL JUMPERS

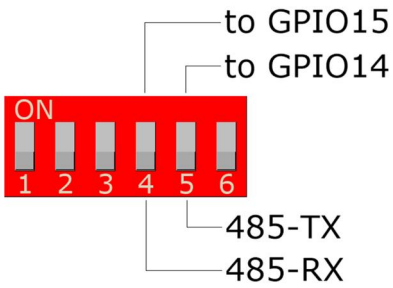
The left three positions of the switch SW2 are used to select the stack level of the card:



STACK LEVEL:	0	1	2	3	4	5	6	7
I2C ADDRESS:	0x50	0x51	0x52	0x53	0x54	0x55	0x56	0x57

RS-485/MODBUS COMMUNICATION

The Industrial Automation Card contains a standard RS485 transceiver which can be accessed both by the local processor and by Raspberry Pi. The desired configuration is set from two bypass switches on the configuration switch SW2.



If switches are ON, Raspberry Pi can communicate with any device with an RS485 interface. In this configuration the 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:

~$ megaind [0] rs4845wr 0 0 0 0 0

If switches are OFF, 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/megaind-rpi/blob/master/Modbus.md

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

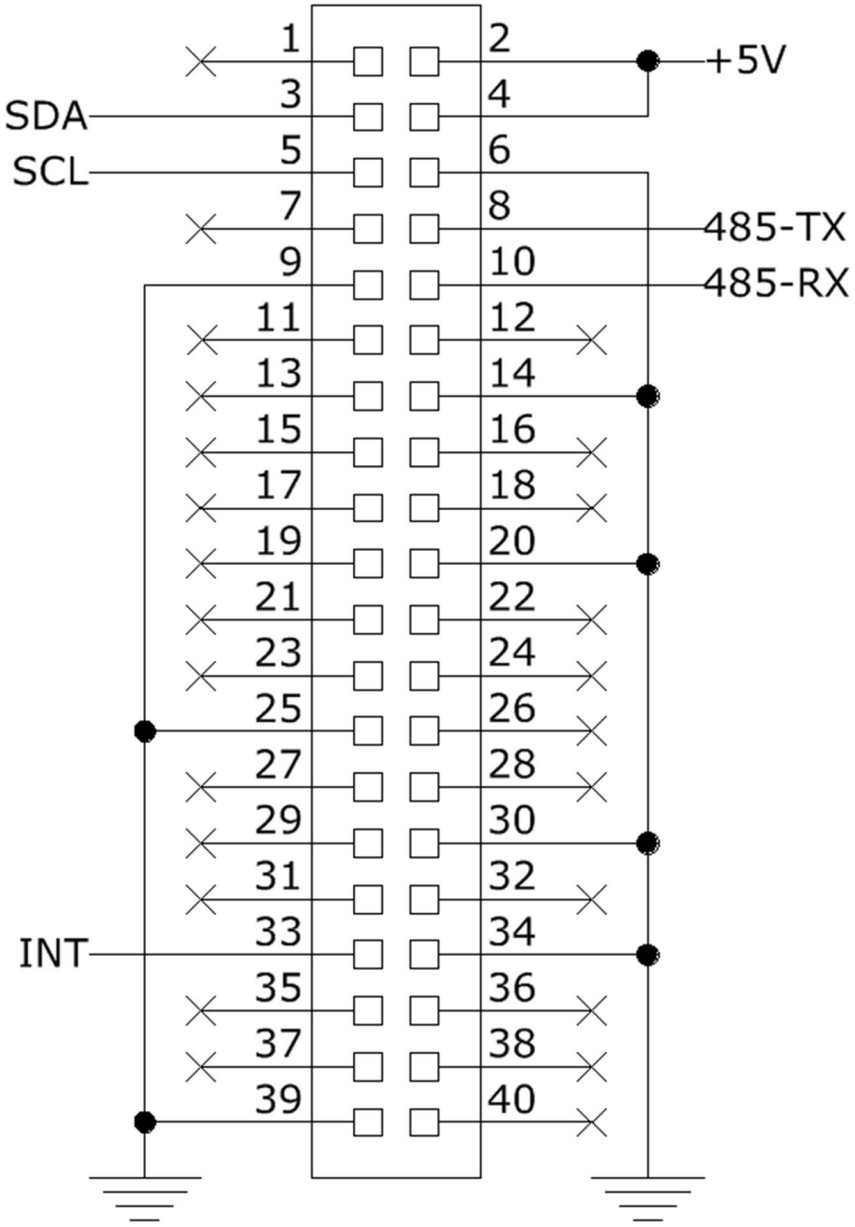
RS485 TERMINATOR

The rightmost position of the configuration switch SW2 is used to enable the RS485 terminator. Set it ON if the Industrial Automation HAT is the last one on the RS485 chain.

1-WIRE PORT

The 1-Wire port is controlled by the local processor. It can be used to read temperatures using the DS18B20 temperature sensor.

RASPBERRY PI HEADER



POWER REQUIREMENTS

The Industrial Automation 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. The power led is controlled by the processor and is flashing to indicate normal operation.

A local 5V regulator supplies up to 3A power to Raspberry Pi, and a 3.3V regulator powers the digital circuits. An isolated 5V to 5V DC-DC converter is used to power the optically isolated open drain outputs.

Two more 5V to 12V converters are used to generate the +/-12V required to read the -10V to +12V inputs.

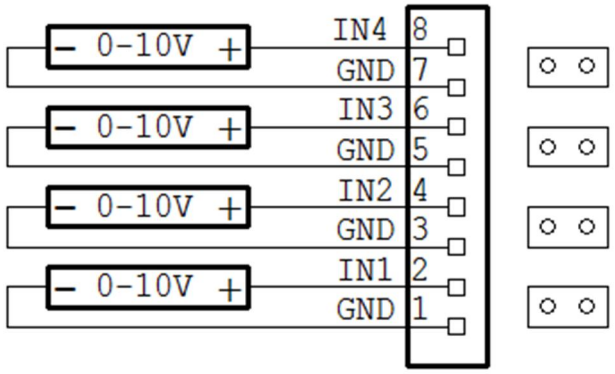
If multiple Industrial Automation Card are stacked on top of each other, we recommend using a single 24VDC power supply to power all the cards. The user must split the cable and run the wires to each card.

POWER CONSUMPTION:

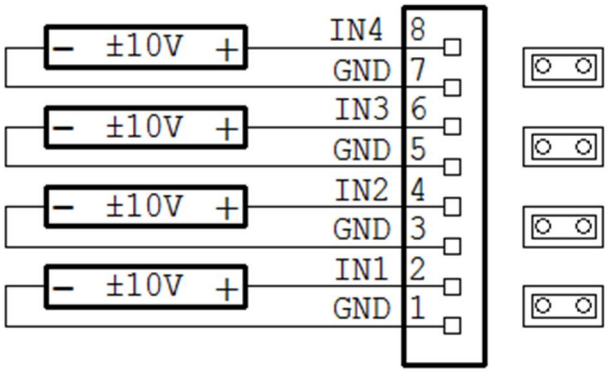
- 50 mA @ +24V

0-10V INPUTS

The Industrial Automation Card can measure four 0-10V or -10V to +10V signals using 12 bit A/D converters embedded in the STM microcontroller. Thus, the 1 bit error in the first case is 2.4mV, while in the second case is 4.9mV. One jumper per each channel permits switching the input range from 0-10V to -10V-+10V.



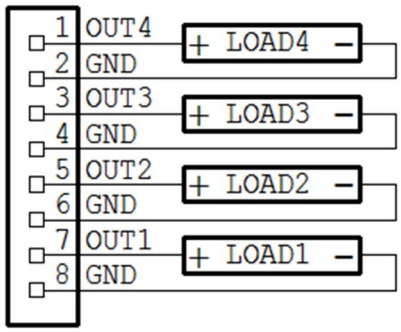
0-**10V INPUTS CONFIGURATION**: JUMPERS OFF



±10V INPUTS CONFIGURATIONL JUMPERS ON

0-10V OUTPUTS

The Industrial Automation Card has four 0-10V outputs. Each output can supply a current of up to 10mA.

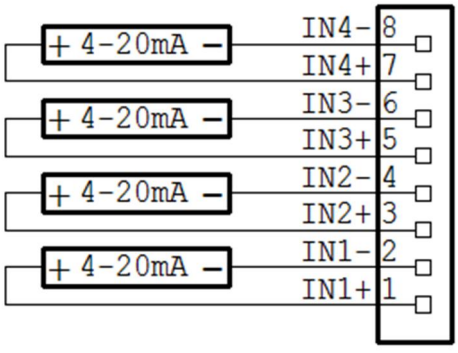


4-20mA CURRENT LOOPS

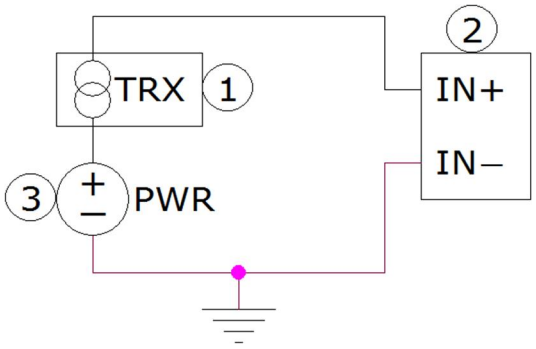
The Industrial Automation Card has four 4-20mA optically isolated inputs and four 4-20mA outputs, driven by open-drain MOSFETS.

Since 4-20mA current loops are optically isolated from the system ground, they can be used with isolated on non-isolated, 2, 3 or 4 wire transmitters. In all cases you need to provide external power supply (up to 24VDC).

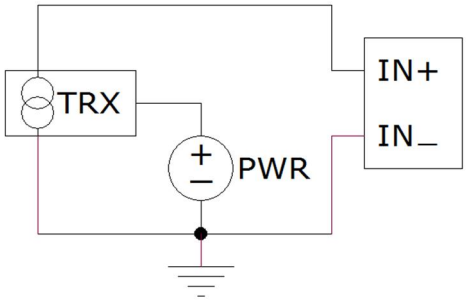
Typical configuration of the input loops:



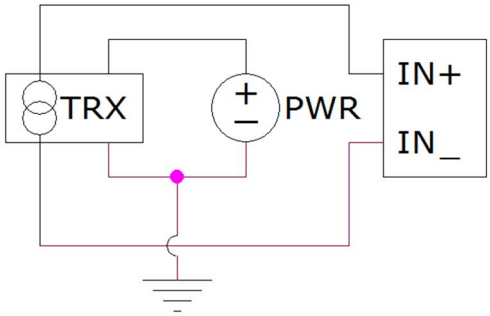
4-20mA INPUTS CONFIGURATION



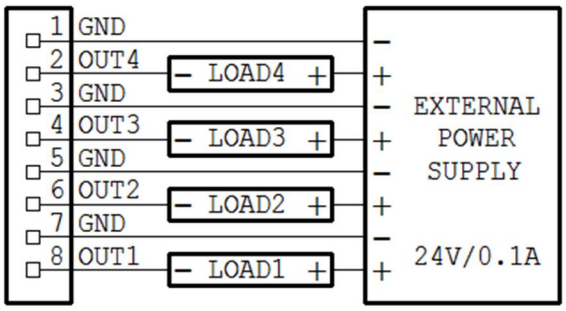
Case 1: Two wire transmitter, shared ground.



Case 2: Three wire transmitter, shared ground.

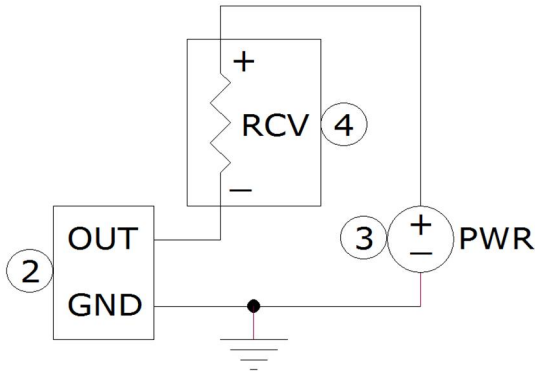


Case 3: Four wire transmitter, isolated ground.



4-20mA OUTPUTS CONFIGURATION

The 4-20mA outputs are driven by Open Drain MOSFETS with a common ground. Connect your 4-20mA receiver as shown in the following diagram:



LEGEND:

(1) External 4-20mA Transmitter

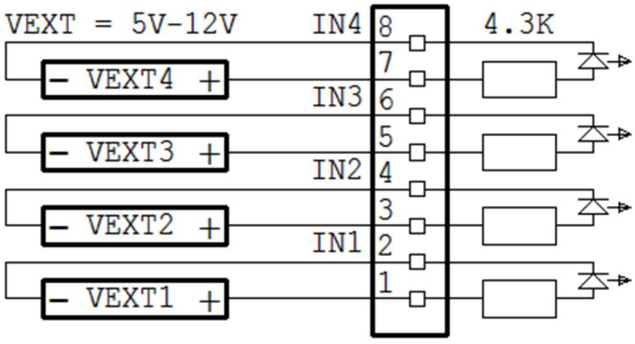
(2) Industrial Automation Card

(3) External Power Supply

(4) External 4-20mA Receiver

OPTICALLY-ISOLATED INPUTS

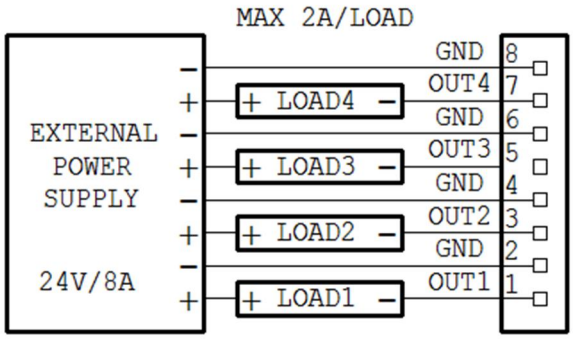
The Industrial Automation Card has four optically isolated inputs which can measure signals between 5V to 12V. Each channel has a 4.3K internal resistor in series with the input.



OPTICALLY ISOLATED INPUTS CONFIGURATION

OPEN DRAIN OUTPUTS

The Industrial Automation Card has four open drain outputs. Each output can drive maximum 24V and 2A continuous current, or 4A peak current for a limited time.



OPEN DRAIN OUTPUTS CONFIGURATION

HARDWARE WATCHDOG

The Industrial Automation 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.

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 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:	megaind <id> uincal <channel> <value>
RESET CALIBRATION OF 0-10V INPUTS:	megaind <id> uncalrst <channel>
CALIBRATE 4-20mA INPUTS:	megaind <id> iincal <channel> <value>
RESET CALIBRATION OF 4-20mA INPUTS:	megaind <id> iincalrst <channel>
CALIBRATE 0-10V OUTPUTS:	megaind <id> uoutcal <channel> <value>
RESET CALIBRATION OF 0-10V OUTPUTS:	megaind <id> uoutcalrst <channel>
CALIBRATE 4-20mA OUTPUTS:	megaind <id> ioutcal <channel> <value>
RESET CALIBRATION OF 4-20mA INPUTS:	megaind <id> ioutcalrst <channel>

HARDWARE SPECIFICATIONS

ON BOARD RESETTABLE FUSE: 1A

0-10V INPUTS:

- Input Impedance: 20KΩ
- Resolution: 12 bits
- Sample rate: 1KHz.

0-10V OUTPUTS:

- Minimum Output Load: 1KΩ
- Resolution: 13 bits

4-20mA INPUTS:

- Sample rate: 1KHz.
- Input impedance: 150Ω

4-20mA OUTPUTS:

- Resistive load: Maximum 1 KΩ @ 24V external voltage
- Maximum external voltage: 24V

OPEN DRAIN OUTPUTS

- Drive capability: 24V/2A continuous, 4A peak
- PWM Frequency 4.8KHz

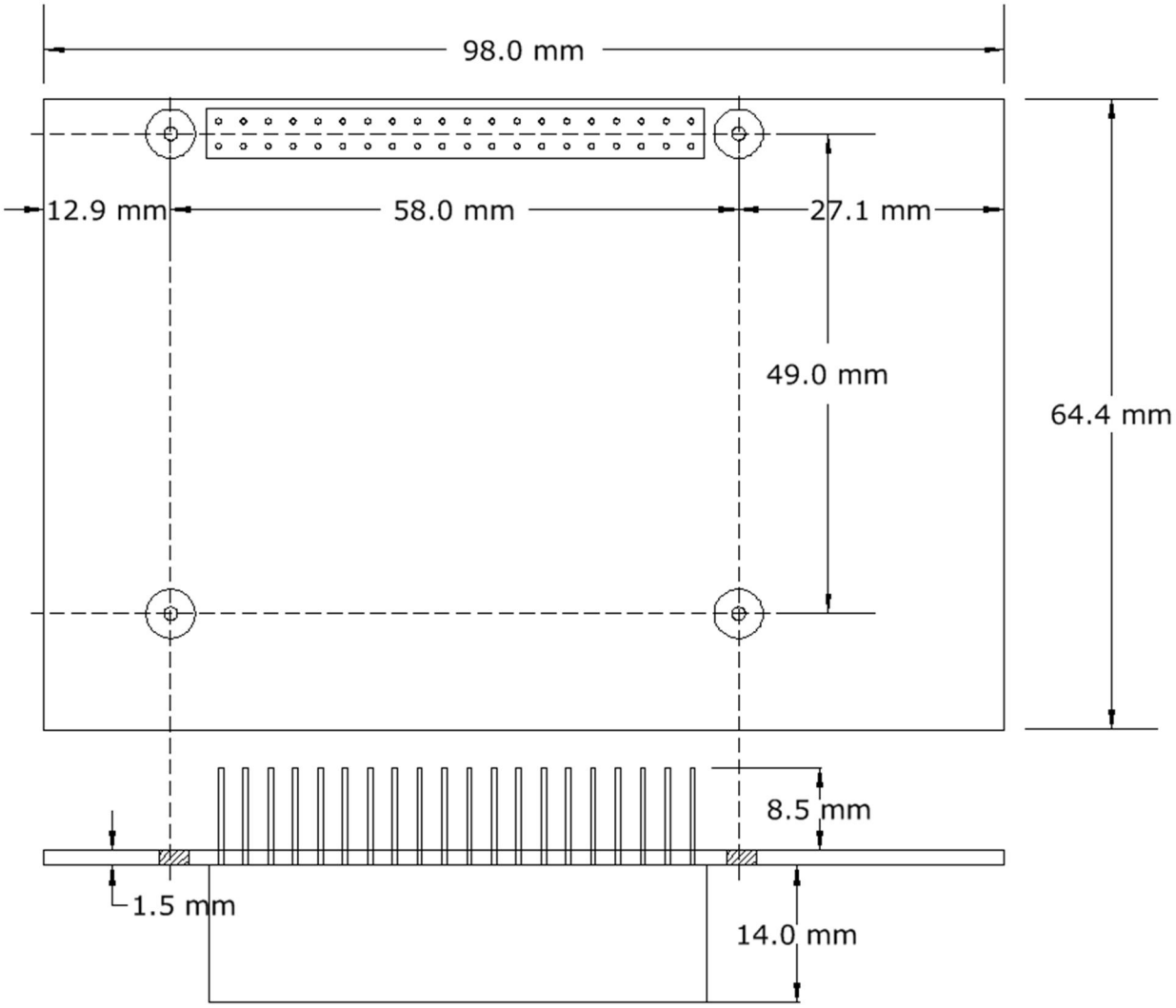
LINEARITY OVER FULL SCALE

Analog inputs are processed using 12 bit A/D converters internal to the on-board processor. The inputs are sampled at 1KHz. All inputs and outputs are calibrated at test time at the end points and values are stored in flash. After calibration we checked the linearity over full scale and obtained the following results:

Channel	Max Error	%
4-20mA IN	32µA	0.2%
4-20mA OUT	10µA	0.06%
0-10V IN	15uV	0.15%
0-10V OUT	10uV	0.1%

Better calibration is possible using the calibration commands

MECHANICAL SPECIFICATIONS



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 megaind software from github.com:

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

4. `~$ cd /home/pi/megaind-rpi`

5. `~/megaioind-rpi$ sudo make install`

6. `~/megaioind-rpi$ megaind`

The program will respond with a list of available commands.

Type "megaind -h" for online help.

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

```
~$ cd /home/pi/megaind-rpi

~/megaind-rpi$ git pull

~/megaind-rpi$ sudo make install
```