

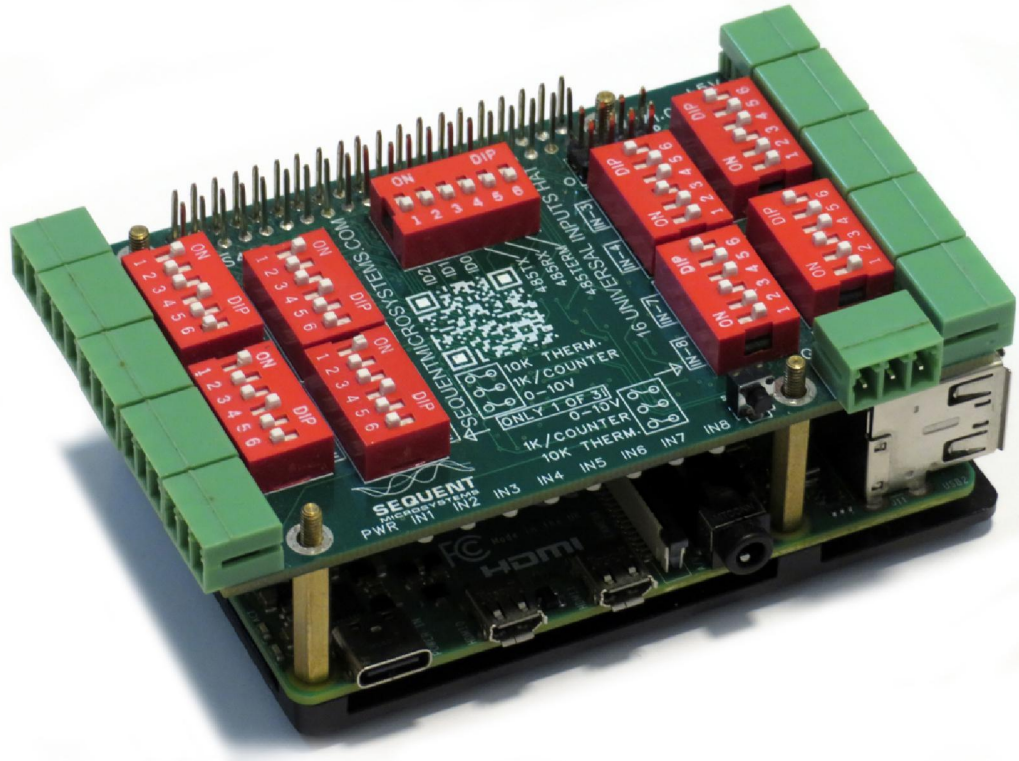
# SIXTEEN UNIVERSAL INPUTS HAT for RASPBERRY Pi

## USER'S GUIDE VERSION 1.0

[SequentMicrosystems.com](http://SequentMicrosystems.com)

GENERAL DESCRIPTION .....	2
FEATURES.....	3
WHAT IS IN YOUR KIT .....	4
QUICK START-UP GUIDE .....	5
BOARD LAYOUT .....	5
CONFIGURATION DIP-SWITCH .....	6
INPUT SELECTION JUMPERS.....	7
RS-485/MODBUS COMMUNICATION .....	7
RASPBERRY PI HEADER .....	8
POWER REQUIREMENTS.....	9
HARDWARE WATCHDOG .....	9
ANALOG INPUTS CALIBRATION .....	10
HARDWARE SPECIFICATIONS.....	10
LINEARITY OVER FULL SCALE.....	11
MECHANICAL SPECIFICATIONS .....	11
SOFTWARE SETUP .....	12

## GENERAL DESCRIPTION



The Sixteen Universal Inputs HAT brings to the Raspberry Pi platform 16 universal inputs. Configurable from convenient DIP switches, each input can be selected to process analog or digital signals:

- 1K thermistors
- 10K thermistors
- 0-10V analog inputs
- Dry contact inputs
- Counter inputs

Stackable to 8 levels, the card works with all Raspberry Pi versions, from Zero to 5.

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.

Eight bi-color LED indicators show status of all inputs. Two parallel RS485/MODBUS ports permit almost unlimited expandability. On board Real Time Clock with battery backup keeps the time almost indefinitely in case of power failure. And last but not least, a hardware watchdog can power cycle the Raspberry Pi in case of software lockup.

TVS diodes on all inputs protect the card for external ESD. On-board resettable fuse protects it from accidental shorts. A single 5VDC power source supplies also power to the Raspberry Pi.

## FEATURES

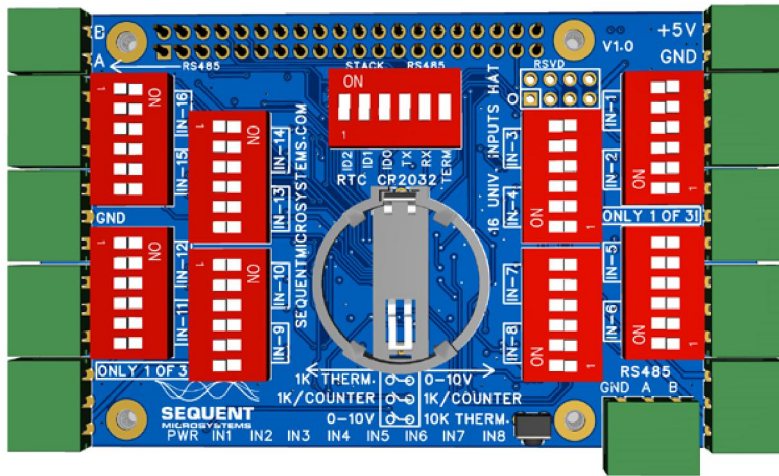
- Sixteen Universal Inputs, 8-Layer Stackable HAT for Raspberry Pi
- Each input can be configured to process analog or digital signals:
  - 1K thermistors
  - 10K thermistors
  - 0-10V analog inputs, 500Hz sample rate
  - Dry contact inputs
  - Counter inputs
- Eight Level Stackable
- Pluggable Connectors 26-16 AWG wires
- Eight software programmable bi-color LEDs
- RS485 In and Out ports
- 5VDC Power Supply powers also the Raspberry Pi
- Real Time Clock with Battery Backup
- On board hardware watchdog
- Resettable fuse
- TVS protection on all inputs
- 32-bit Processor running at 64MHz
- Uses only I2C port (address 0x58 to 0x5f), leaves all GPIO pins available
- No jumpers: input mode selection from DIP switches
- Command line
- Python Library
- Node-Red nodes
- All mounting hardware included: stand-offs, screws and nuts
- Open source hardware and schematics

All inputs and outputs use pluggable connectors which permit easy wiring access when multiple cards are stacked. Up to eight Sixteen Universal Inputs 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.

The 8 bicolor LED's can be associated with any analog or digital inputs. An on-board push button can be programmed to shut down the Raspberry Pi or to run a custom script.

## WHAT IS IN YOUR KIT

### 1. Sixteen Universal Inputs Card for Raspberry Pi

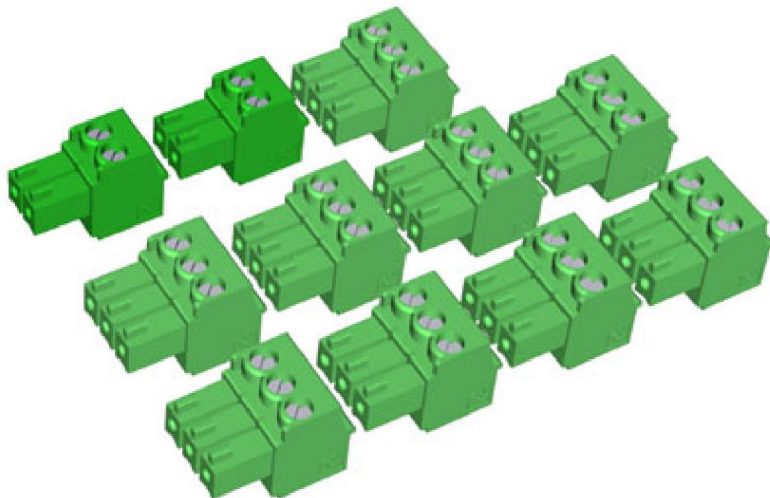


### 2. Mounting hardware



- Four M2.5x19mm male-female brass standoffs
- Four M2.5x5mm brass screws
- Four M2.5 brass nuts

### 3. All the required female mating connectors.

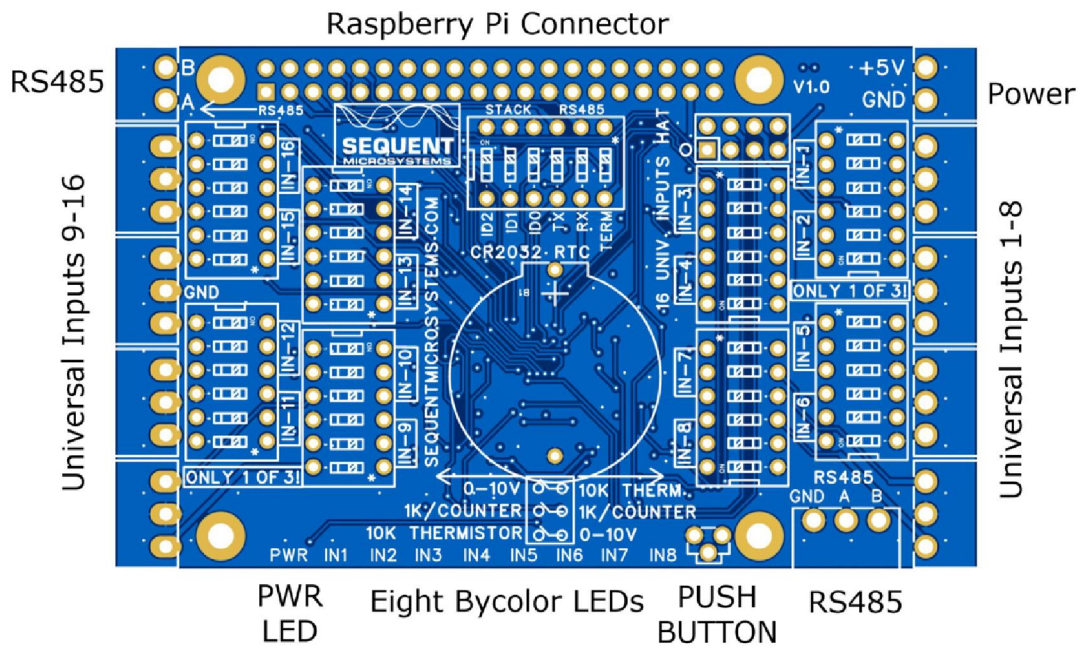


## QUICK START-UP GUIDE

1. Plug your Sixteen Universal Inputs 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/16univin-rpi.git`
  - b. `~$ cd /home/pi/16univin-rpi`
  - c. `~/16univin-rpi$ sudo make install`
4. `~/16univin-rpi$ 16univin`

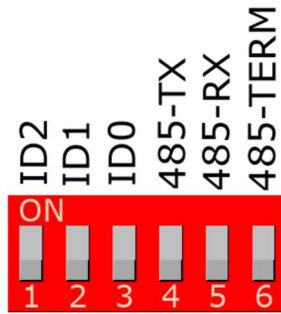
The program will respond with a list of available commands.

## BOARD LAYOUT




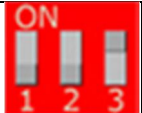


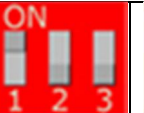

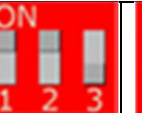
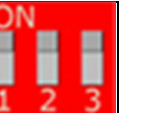
Eight bycolor LEDs can be controlled in the software. The LEDs can be activated to show the status or threshold of any input.s

## CONFIGURATION DIP-SWITCH



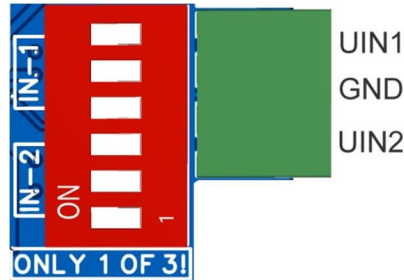
The left three positions of the DIP-Switch J3 are used to select the stack level of the card. The next three are used for the RS485/MODBUS communication.

### STACK LEVEL SELECTION

DIP SWITC H								
STACK LEVEL	0	1	2	3	4	5	6	7
I2C ADDR.	0x18	0x19	0x1A	0x1B	0x1C	0x1D	0x1E	0x1F

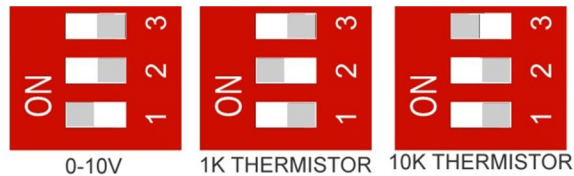
## INPUT SELECTION JUMPERS

Pairs of two universal inputs share a 3-pin connector with a common ground in the middle. Six position DIP switches are used to configure the two inputs. Three DIP switches are allocated to each input, as shown in the following picture:



Digital inputs and event counters can be processed in any switch configuration. Analog inputs can be selected as shown below:

### SELECTION FOR CHANNELS 1-8 (RIGHT SIDE OF THE BOARD)



### SELECTION FOR CHANNELS 9-16 (LEFT SIDE OF THE BOARD)



## RS-485/MODBUS COMMUNICATION

The Sixteen Universal Inputs 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 three bypass jumpers on configuration DIP switch J3.

If switches are ON, the RS485 is driven by the default serial port of Raspberry Pi. In this configuration the Sixteen Universal Inputs 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:

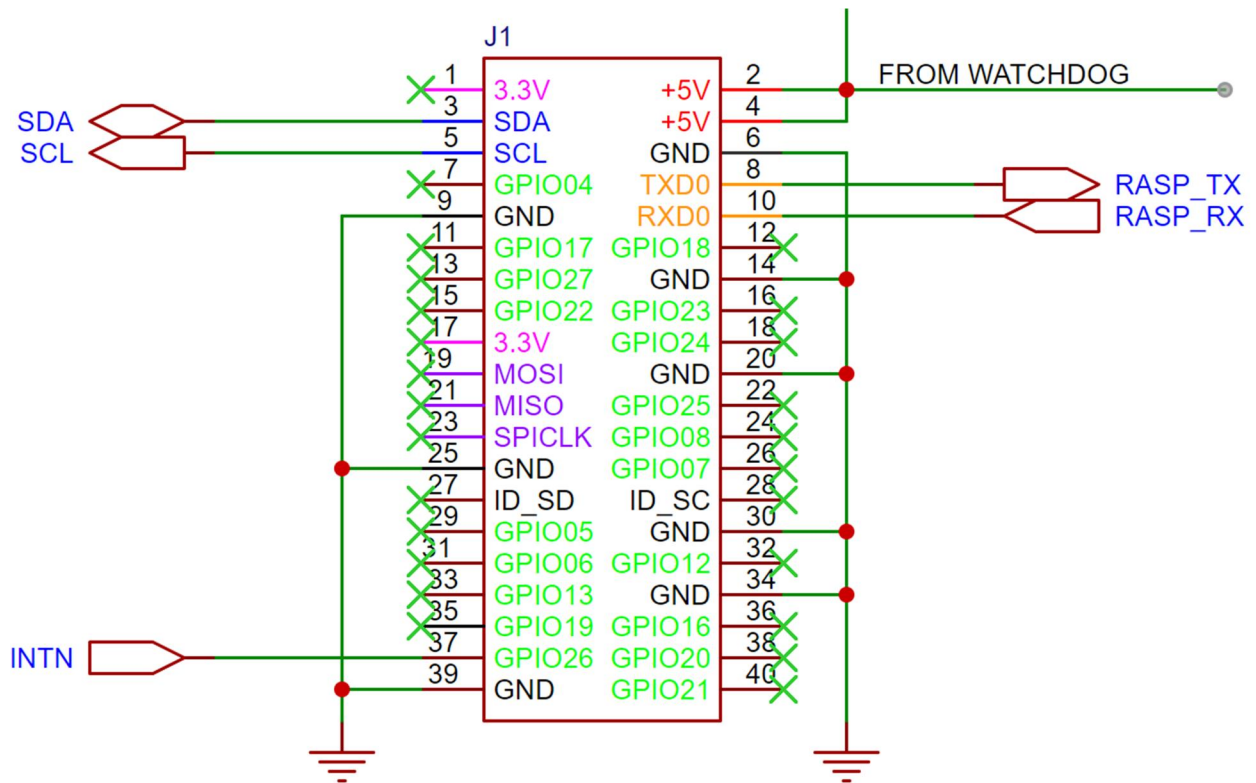
```
~$ 16univin [0] cfgmbwr 0 0 0 0 0
```

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/16univin-rpi/blob/main/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.

## RASPBERRY PI HEADER



## POWER REQUIREMENTS

The Sixteen Universal Inputs Card requires an external 5VDC regulated power supply. Power is supplied to the board through the dedicated connector in the upper right corner (see BOARD LAYOUT).

A card supplies up to 3A power to the Raspberry Pi, and a 3.3V regulator powers the digital circuits.

### POWER CONSUMPTION:

- 50 mA @ +5V

## HARDWARE WATCHDOG

The Sixteen Universal Inputs Card contains a built-in hardware watchdog which will guarantee that your mission-critical project will continue running even if the Raspberry Pi software hangs up. After power up the watchdog is disabled, and it 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.

The 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 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 as possible to the two ends of scale. To calibrate the analog inputs, the user must provide analog signals. (Example: to calibrate 0-10V inputs, the user must provide a 10V adjustable power supply).

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 following commands are available for calibration:

CALIBRATE 0-10V INPUTS:	16univin <id> cuin <channel> <value>
RESET CALIBRATION OF 0-10V INPUTS:	16univin <id> rcuin
CALIBRATE 10K INPUTS:	16univin <id> cres10k <channel> <value>
RESET 10K INPUTS:	16univin <id> rcres10k
CALIBRATE 1K INPUTS:	16univin <id> cres1k <channel> <value>
RESET 1K INPUTS:	16univin <id> rcres1k

## HARDWARE SPECIFICATIONS

**ON BOARD RESETTABLE FUSE:** 4A

### 0-10V INPUTS:

- Maximum Input Voltage: 12V
- Input Impedance: 20K $\Omega$
- Resolution: 12 bits
- Sample rate: 250 sps

### CONTACT CLOSURE INPUTS

- Maximum count frequency: 100 Hz

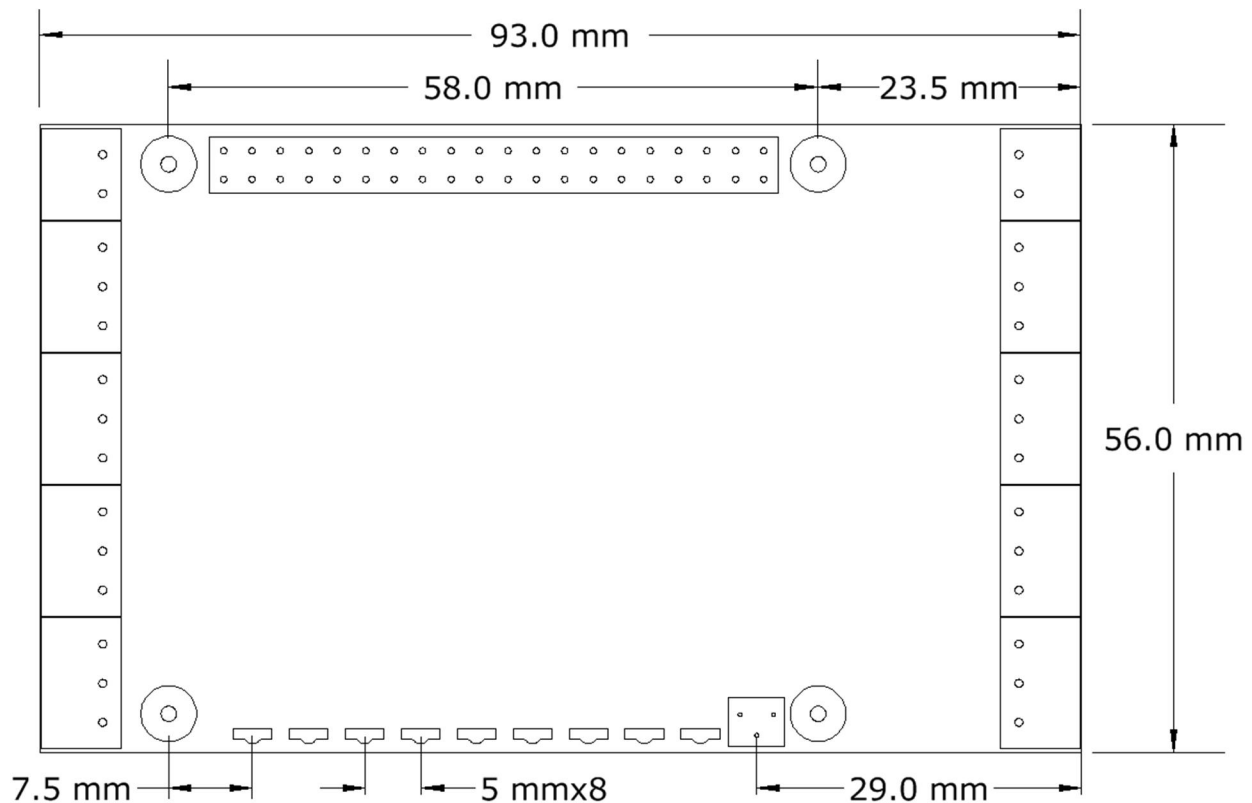
## LINEARITY OVER FULL SCALE

Analog inputs are processed using 12 bit A/D converters internal to the on-board processor. The inputs are sampled at 250 Hz.

All inputs 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 a maximum full scale error of  $15\mu\text{V}$  (0.15%)

## MECHANICAL SPECIFICATIONS



The card is compatible with any card manufactured by Sequent Microsystems and can be mounted in any order. You may also mix cards from other vendors, assuming they do not use the same I2C address.

## 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 16univin software from github.com:

```
~$ git clone https://github.com/SequentMicrosystems/16univin-rpi.git
```

4. 

```
~$ cd /home/pi/16univin-rpi
```

5. 

```
~/16univin-rpi $ sudo make install
```

6. 

```
~/16univin-rpi$ 16univin
```

The program will respond with a list of available commands.

Type "**16univin -h**" for online help.

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

```
~$ cd /home/pi/16univin-rpi
```

```
~/16univin-rpi$ git pull
```

```
~/16univin-rpi$ sudo make install
```