

ASTRA controller (BR4A4AL & others)

4 Relays 16A, 4 AC Input, AC power, LED

Features an impressive array of functions for individual solutions. And cost-effective, space-saving implementation, as well as comprehensive WiFi and expansion options, pay off when automating a home.

Opens up a wide range of control options for building functions, making it the perfect choice for fast, easy, and space-saving implementation of individual home automation projects. Provides the right solution for the convenient control of your heating, air-conditioning, lighting, irrigation, access control, garage door, presence detection, and for saunas and pools.

Especially handy is the controller's ability to connect to a smart home server as well as convenient control via smartphone, tablet, and PC through WiFi.



Highlights of the ASTRA controller (Basic Module):

- Interface: WiFi and Bluetooth
- Outputs: electro-mechanical or solid state relays 10-16 A
- Inputs: isolated 30-260 V AC / dry contacts or non-isolated GPIOs
- Power in: AC 90~240 V or DC 7-36 V
- Firmware: TASMOTA, Integrated web server with web interface and MQTT protocol
- Extended temperature range -20 to 55°C, without condensation.
- Antenna internal / external

Technical Specifications – Basic and Expansion Modules:

Basic modules	BR4A4AL	BR4D4DL	BR4G8AL	BT4A4AL		
	4 Relays, 4 AC In, PWR AC	4 Relays, 4 DC In, PWR DC	4 Relays, 8 GPIO, PWR DC	4 SSR, 4 AC In, PWR AC		
Supply voltage						
Rated value (AC)	90~240V (VAC 50/110Hz)			90~240V (VAC 50/110Hz)		
Rated value (DC)		7~36V (VDC)	7~36V (VDC)			
Digital outputs	4 Relays 16A, external fusing necessary	4 Relays 16A, external fusing necessary	4 Relays 16A, external fusing necessary. 8 Input / Output 3.3V (GPIO) logic level	TRIAC (SSR) 16A, external fusing necessary		
Relays switching capacity of contacts	VAC 240V, VDC 30V	VAC 240V, VDC 30V	VAC 240V, VDC 30V			
with inductive load, max. (normal use=70%max.)	10 A	10 A	10 A			
with resistive load, max. (normal use=70%max.)	16 A	16 A	16 A			
Digital inputs	4 AC Isolated	4 DC Isolated, dry contact	8 Input/Output (GPIO) 3.3V logic level	4 AC Isolated		

Basic modules	BR4A4AL	BR4D4DL	BR4G8AL	BT4A4AL		
Analog Inputs	no	no	no	no		
Indicate						
Output	4 LED (blue)	4 LED (blue)	4 LED (blue)	4 LED (blue)		
Input	4 LED (green)	4 LED (green)	–	4 LED (green)		
WiFi status	LED (blue)	LED (blue)	LED (blue)	LED (blue)		
Power IN	LED (red)	LED (red)	LED (red)	LED (red)		
Sound	Buzzer	Buzzer	Buzzer	Buzzer		
Mounting	DIN 35mm	DIN 35mm	DIN 35mm	DIN 35mm		

Hardware and firmware:

Basic modules	BR4A4AL	BR4D4DL	BR4G8AL	BT4A4AL		
	4 Relays, 4 AC In, PWR AC	4 Relays, 4 DC In, PWR DC	4 Relays, 8 GPIO, PWR DC	4 SSR, 4 AC In, PWR AC		
Firmware	TASMOTA (version link), configuration (link) (TASMOTA weblink)					
Webserver	integrated web server with web interface					
MQTT	MQTTv2					
Interface	Serial UART, SPI, I2C (3.3V logic level)					
WiFi	802.11 b/g/n , 802.11n (2.4Ghz), up to 150Mbps					
Bluetooth	Bluetooth V4.2 BR/EDR and BLE specification					
MCU	Microcontroller Espressif ESP32-D0WDQ6, core Xtensa dual core 32-bit LX6 microprocessor, up to 600MIPS, operation voltage 3.3V					
Flash memory	Capacity 32Mbit (4Mbyte), NOR FLASH, SPI, operation voltage 3.3V					

Embedded modules:

Basic modules	BR4A4AL	BR4D4DL	BR4G8AL	BT4A4AL		
	4 Relays, 4 AC In, PWR AC	4 Relays, 4 DC In, PWR DC	4 Relays, 8 GPIO, PWR DC	4 SSR, 4 AC In, PWR AC		
Optional modules						
RF433/315	optional	optional	optional	optional		
KNX	optional	optional	optional	optional		

RS-485 / Modbus RTE	optional	optional	optional	optional		
CAN bus	optional	optional	optional	optional		
Z-Wave	optional	optional	optional	optional		
NRF24	optional	optional	optional	optional		

Firmware configuration:

Firmware preconfigured by manufacturer. Link for download latest configuration [\[link\]](#). This is detail information for manual configuration or for 3-d part firmware. MCU ESP32.

Basic modules	BR4A4AL	BR4D4DL	BR4G8AL	BT4A4AL		
	4 Relays, 4 AC In, PWR AC	4 Relays, 4 DC In, PWR DC	4 Relays, 8 GPIO, PWR DC	4 SSR, 4 AC In, PWR AC		
GPIO34	Input-1	Input-1	GPIO34	Input-1		
GPIO35	Input-2	Input-2	GPIO35	Input-2		
GPIO36	Input-3	Input-3	GPIO36	Input-3		
GPIO39	Input-4	Input-4	GPIO39	Input-4		
GPIO25	Relay-1	Relay-1	Relay-1	Relay-1		
GPIO26	Relay-2	Relay-2	Relay-2	Relay-2		
GPIO33	Relay-3	Relay-3	Relay-3	Relay-3		
GPIO32	Relay-4	Relay-4	Relay-4	Relay-4		
GPIO4	LED WiFi	LED WiFi	LED WiFi	LED WiFi		
GPIO12	Sound Buzzer	Sound Buzzer	Sound Buzzer	Sound Buzzer		
GPIO27	RF433/315 socket	RF433/315 socket	GPIO27 / RF433	RF433/315 socket		
GPIO14	–	–	IO14	–		
GPIO15	–	–	IO15	–		
RXD0	Serial1 RXD	Serial1 RXD	Serial1 RXD	Serial1 RXD		
TXD0	Serial1 TXD	Serial1 TXD	Serial1 TXD	Serial1 TXD		
GPIO16	Serial2 RXD	Serial2 RXD	Serial2 RXD	Serial2 RXD		
GPIO17	Serial2 TXD	Serial2 TXD	Serial2 TXD	Serial2 TXD		
GPIO13	Serial2 CS	Serial2 CS	GPIO13/Serial2 CS	Serial2 CS		
GPIO5	SPI CS0	SPI CS0	SPI CS0	SPI CS0		

Basic modules	BR4A4AL	BR4D4DL	BR4G8AL	BT4A4AL		
GPIO18	SPI CLK	SPI CLK	SPI CLK	SPI CLK		
GPIO19	SPI MISO	SPI MISO	SPI MISO	SPI MISO		
GPIO23	SPI MOSI	SPI MOSI	SPI MOSI	SPI MOSI		
GPIO21	I2C SDA	I2C SDA	I2C SDA	I2C SDA		
GPIO22	I2C SCL	I2C SCL	I2C SCL	I2C SCL		
GPIO2	I2C INT	I2C INT	I2C INT	I2C INT		
GPIO0	PROG	PROG	PROG	PROG		
EN	RESET	RESET	RESET	RESET		

The template has been read from the device as it comes pre-flashed with Tasmota and is listed here for reference or recovery:

```
{ "NAME": "ASTRA
R4A4", "GPIO": [1, 1, 1, 1, 576, 1, 1, 1, 480, 1, 1, 1, 3232, 3200, 1, 1, 0, 640, 608, 1, 0, 224,
225, 1152, 0, 0, 0, 0, 227, 226, 160, 161, 162, 0, 0, 163], "FLAG": 0, "BASE": 1 }
```

Falls die ESP32-Temperatur ausgeschaltet werden soll, kann man diesen Befehl eingeben:

```
SetSensor127 0
```

GPIO usage:

GPIO #	Component
GPIO00	User
GPIO01	User
GPIO02	User
GPIO03	User
GPIO04	LedLinki
GPIO05	User
GPIO09	User
GPIO10	User
GPIO12	Buzzer
GPIO13	User
GPIO14	User
GPIO15	User
GPIO16	Serial Rx
GPIO17	Serial Tx
GPIO18	User
GPIO19	User
GPIO20	None
GPIO21	I2C SDA 1
GPIO22	I2C SCL 1
GPIO23	User
GPIO24	None
GPIO25	Relay 1
GPIO26	Relay 2
GPIO27	RF Receiver
GPIO6	None
GPIO7	None
GPIO8	None
GPIO11	None
GPIO32	Relay 4
GPIO33	Relay 3
GPIO34	Switch 1
GPIO35	Switch 2
GPIO36	Switch 3
GPIO37	None
GPIO38	None
GPIO39	Switch 4

ASTRA R4A4

H.EltakoXG.154

Template parameters

Name:

Based on:

GPIO00	User	▼
GPIO01	User	▼
GPIO02	User	▼
GPIO03	User	▼
GPIO04	LedLink_i	▼
GPIO05	User	▼
GPIO06	None	▼
GPIO07	None	▼
GPIO08	None	▼
GPIO09	User	▼
GPIO10	User	▼
GPIO11	None	▼
GPIO12	Buzzer	▼
GPIO13	User	▼
GPIO14	User	▼
GPIO15	User	▼
GPIO16	Serial Rx	▼
GPIO17	Serial Tx	▼
GPIO18	User	▼
GPIO19	User	▼
GPIO20	None	▼
GPIO21	I2C SDA	▼ 1 ▼
GPIO22	I2C SCL	▼ 1 ▼
GPIO23	User	▼
GPIO24	None	▼
GPIO25	Relay	▼ 1 ▼
GPIO26	Relay	▼ 2 ▼
GPIO27	RFrecv	▼
GPIO32	Relay	▼ 4 ▼
GPIO33	Relay	▼ 3 ▼
GPIO34	Switch	▼ 1 ▼
GPIO35	Switch	▼ 2 ▼
GPIO36	Switch	▼ 3 ▼
GPIO37	None	▼
GPIO38	None	▼
GPIO39	Switch	▼ 4 ▼

Tasmota 11.1.0.2 by rockectcontroller.com

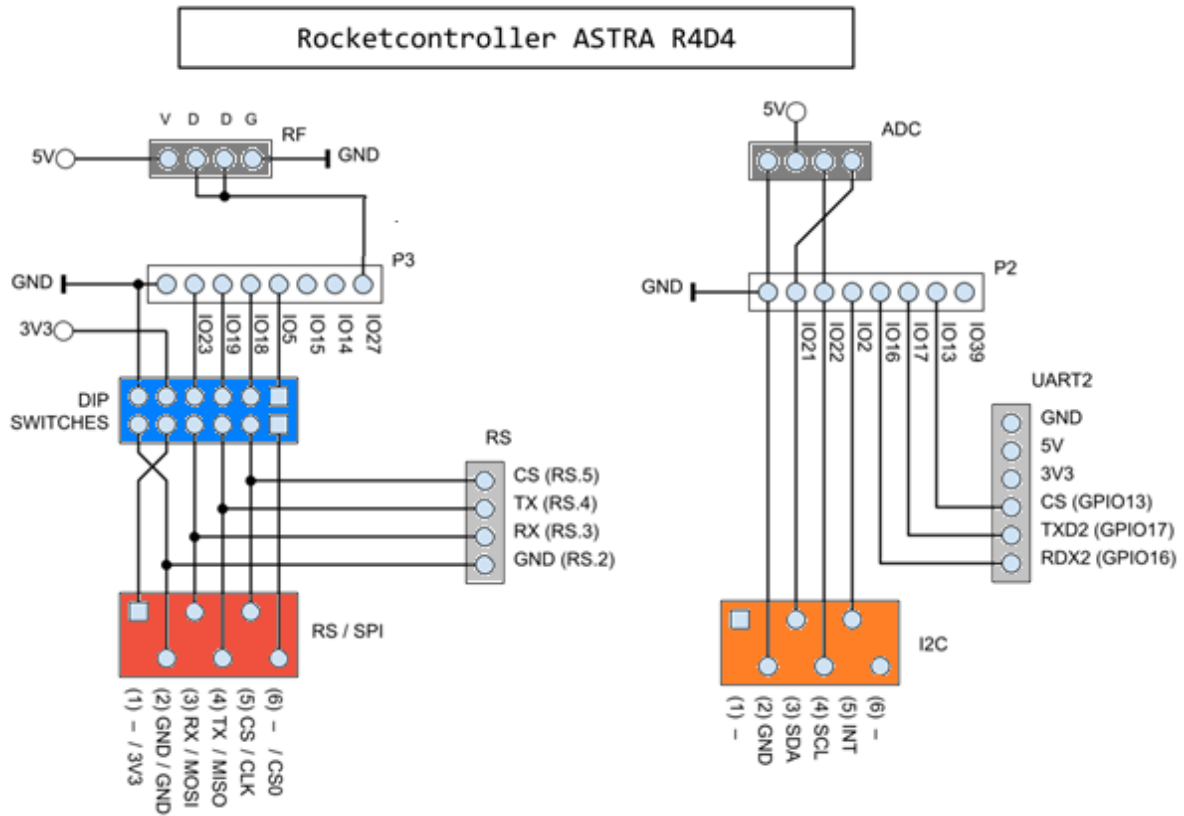
DIP-Switches usages

- **RS mode:** All switches to OFF (close to the numbers)
- **SPI mode:** All switches to ON When in RS mode, the RJ is not connected anymore to the SPI GPIOs. It is intended to receive a RS Adaptation Module plugged between the internal UART2 and RS connectors to host an adaptation module such as RS232 or RS485 / Modbus. At time (2023-1) of writing, Rocketcontroller doesn't yet sale such module as they are still in preparation. When is SPI mode, the RJ is connected to the SPI GPIO and there **MUST NOT BE** any RS module installed in the UART2/RS connectors.

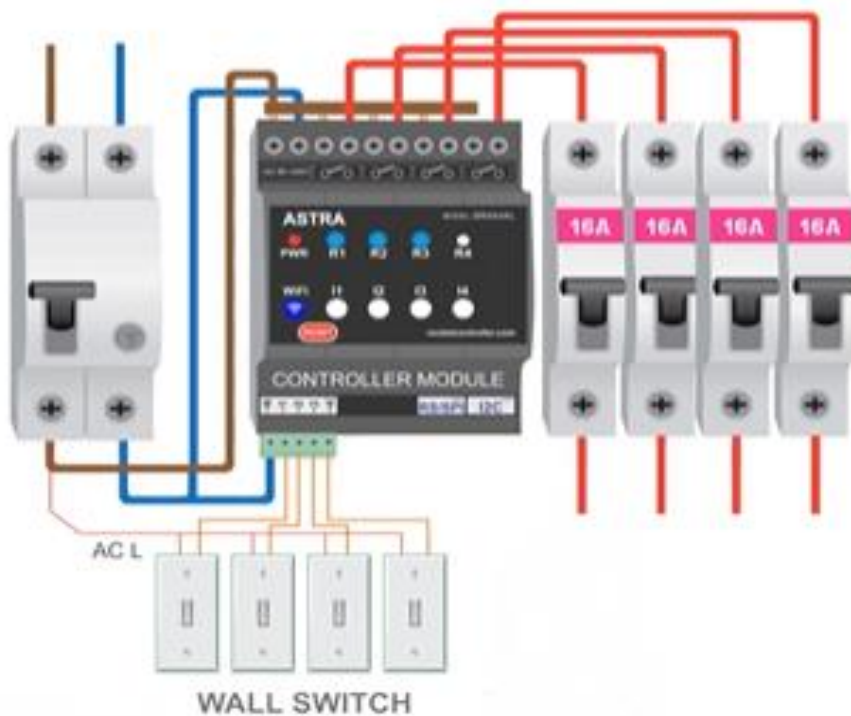


Interfaces Socket:

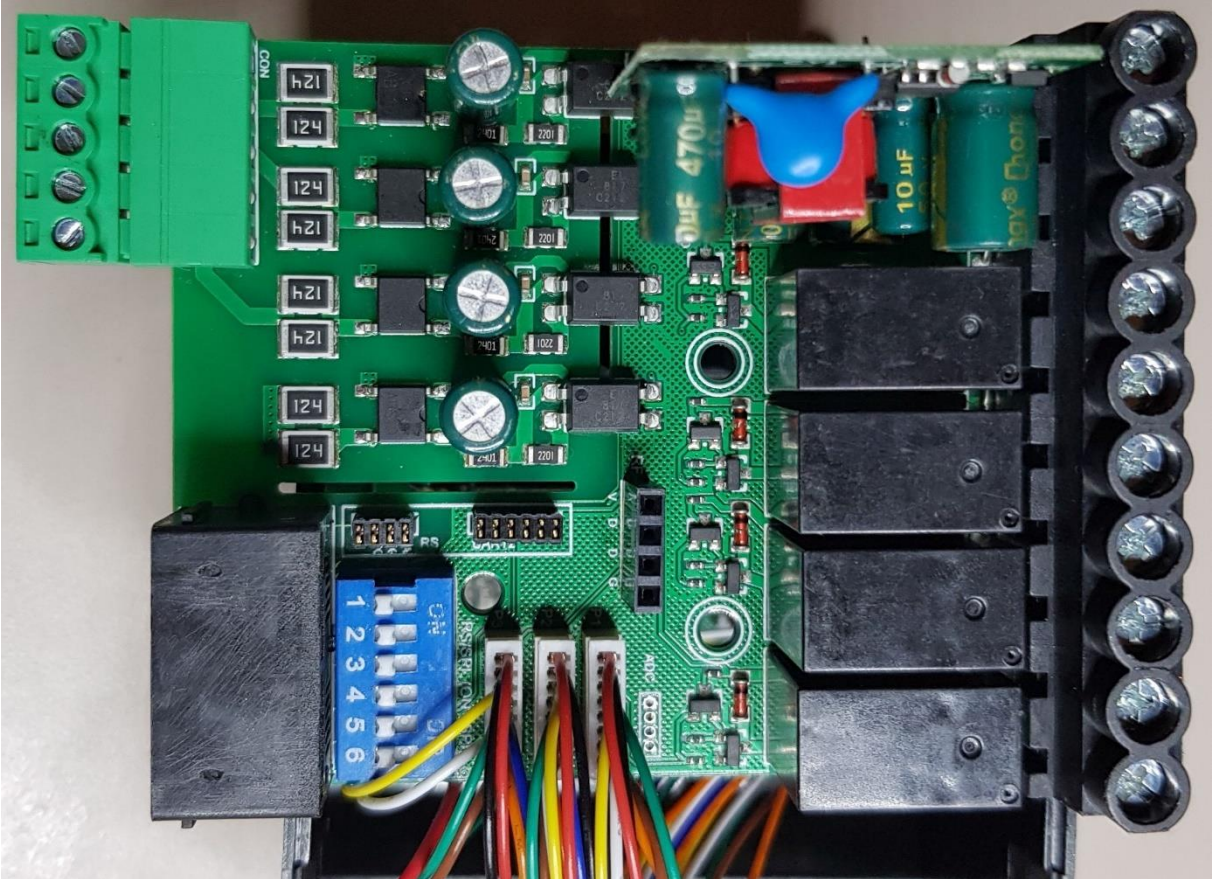
Basic modules	BR4A4AL	BR4D4DL	BR4G4DL	BT4A4AL	
	4 Relays, 4 AC In, PWR AC	4 Relays, 4 DC In, PWR DC	4 Relays, 8 GPIO, PWR DC	4 SSR, 4 AC In, PWR AC	
I2C					
1	–	–	–	–	
2	GND	GND	GND	GND	
3	I2C SDA	I2C SDA	I2C SDA	I2C SDA	
4	I2C SCL	I2C SCL	I2C SCL	I2C SCL	
5	I2C INT	I2C INT	I2C INT	I2C INT	
6	–	–	–	–	
RS mode					
1	–	–	–	–	
2	GND	GND	GND	GND	
3	Serial2 RXD	Serial2 RXD	Serial2 RXD	Serial2 RXD	
4	Serial2 TXD	Serial2 TXD	Serial2 TXD	Serial2 TXD	
5	Serial2 CS	Serial2 CS	Serial2 CS	Serial2 CS	
6	–	–	–	–	
SPI mode					
1	3.3V	3.3V	3.3V	3.3V	
2	GND	GND	GND	GND	
3	SPI MOSI	SPI MOSI	SPI MOSI	SPI MOSI	
4	SPI MISO	SPI MISO	SPI MISO	SPI MISO	
5	SPI CLK	SPI CLK	SPI CLK	SPI CLK	
6	SPI CS0	SPI CS0	SPI CS0	SPI CS0	



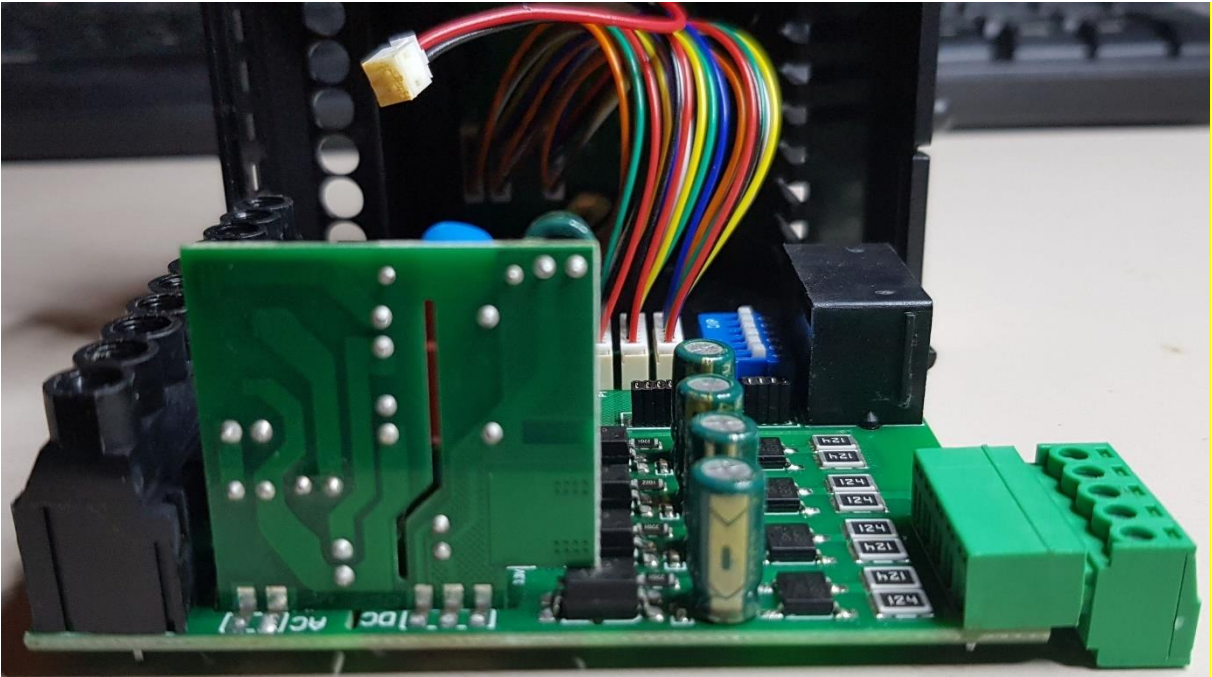
Usage of ASTRA BR4A4AL Input for 230 V AC:



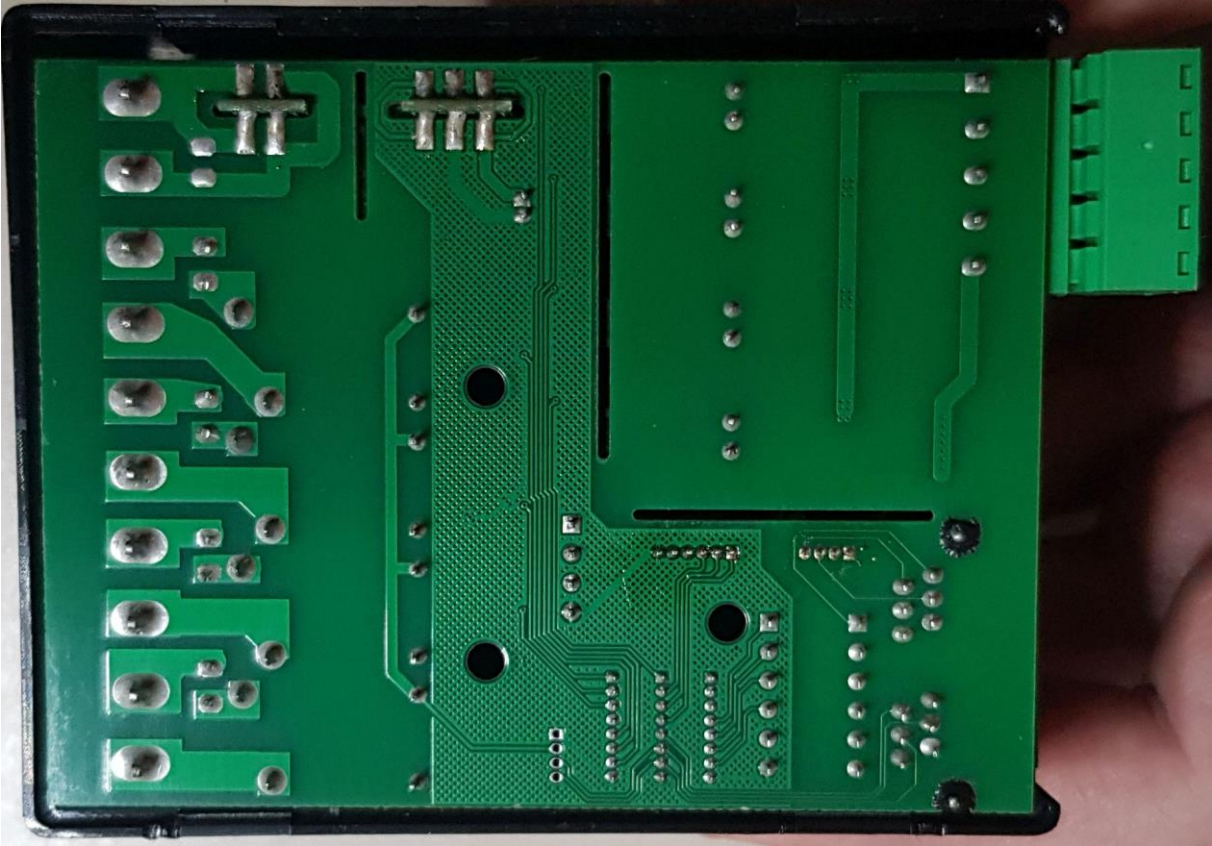
Top view of main PCB:



Main PCB with power supply in front:



Bottom view of main PCB:



Processor and LED PCB glued in top of case:



ASTRA expansion (ER4A4A & others)

4 Relays 16A, 4 AC Input, AC power, LED

Features an impressive array of functions for individual solutions. And cost-effective, space-saving implementation, as well as comprehensive WiFi and expansion options, pay off when automating a home.

Opens up a wide range of control options for building functions, making it the perfect choice for fast, easy, and space-saving implementation of individual home automation projects. Provides the right solution for the convenient control of your heating, air-conditioning, lighting, irrigation, access control, garage door, presence detection, and for saunas and pools.

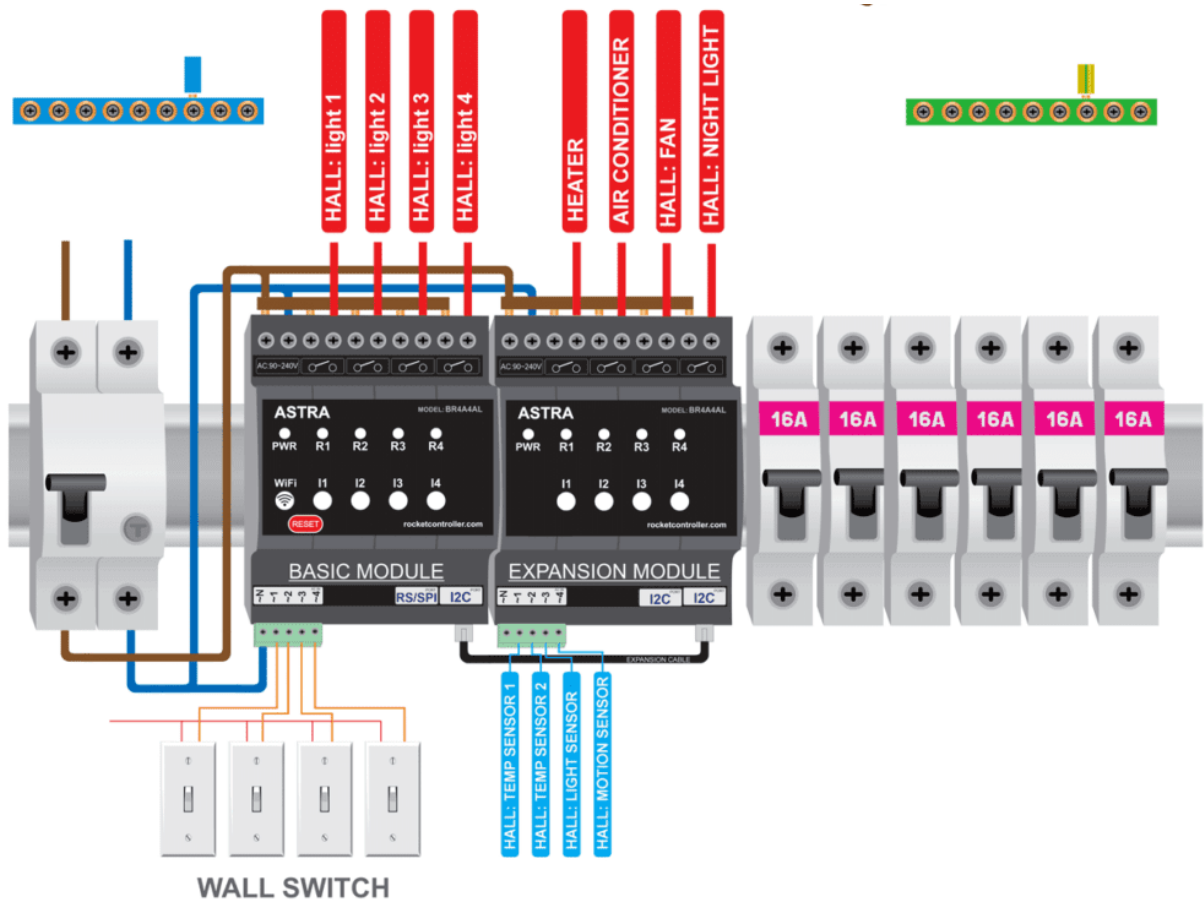
Especially handy is the controller's ability to connect to a smart home server as well as convenient control via smartphone, tablet, and PC through WiFi.



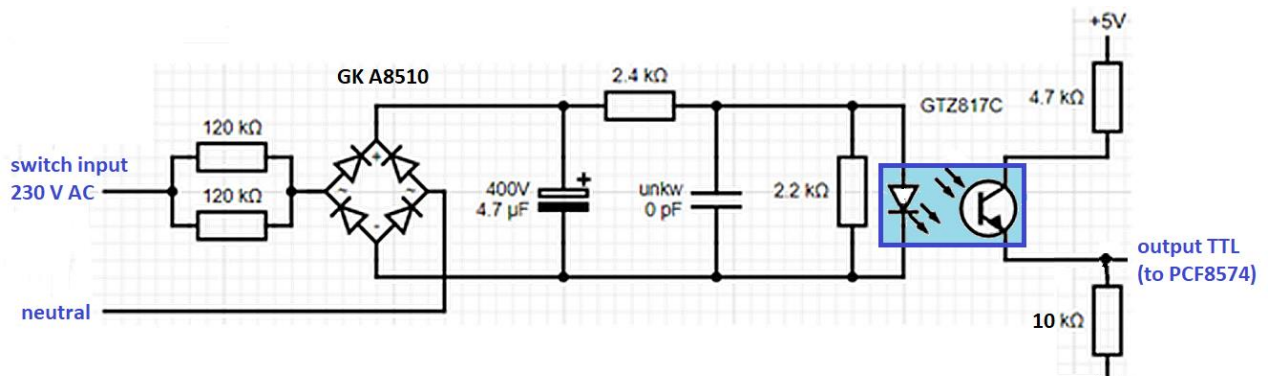
Highlights of the ASTRA expansion modules:

- Interface: I2C
- Outputs: electro-mechanical relays 16 A
- Inputs: isolated 30-260 V AC / dry contacts
- Power in: AC 90~240 V or DC 7-36 V
- Firmware: TASMOTA, Integrated web server with web interface and MQTT protocol
- Extended temperature range -20 to 55°C, without condensation.
- Antenna internal / external

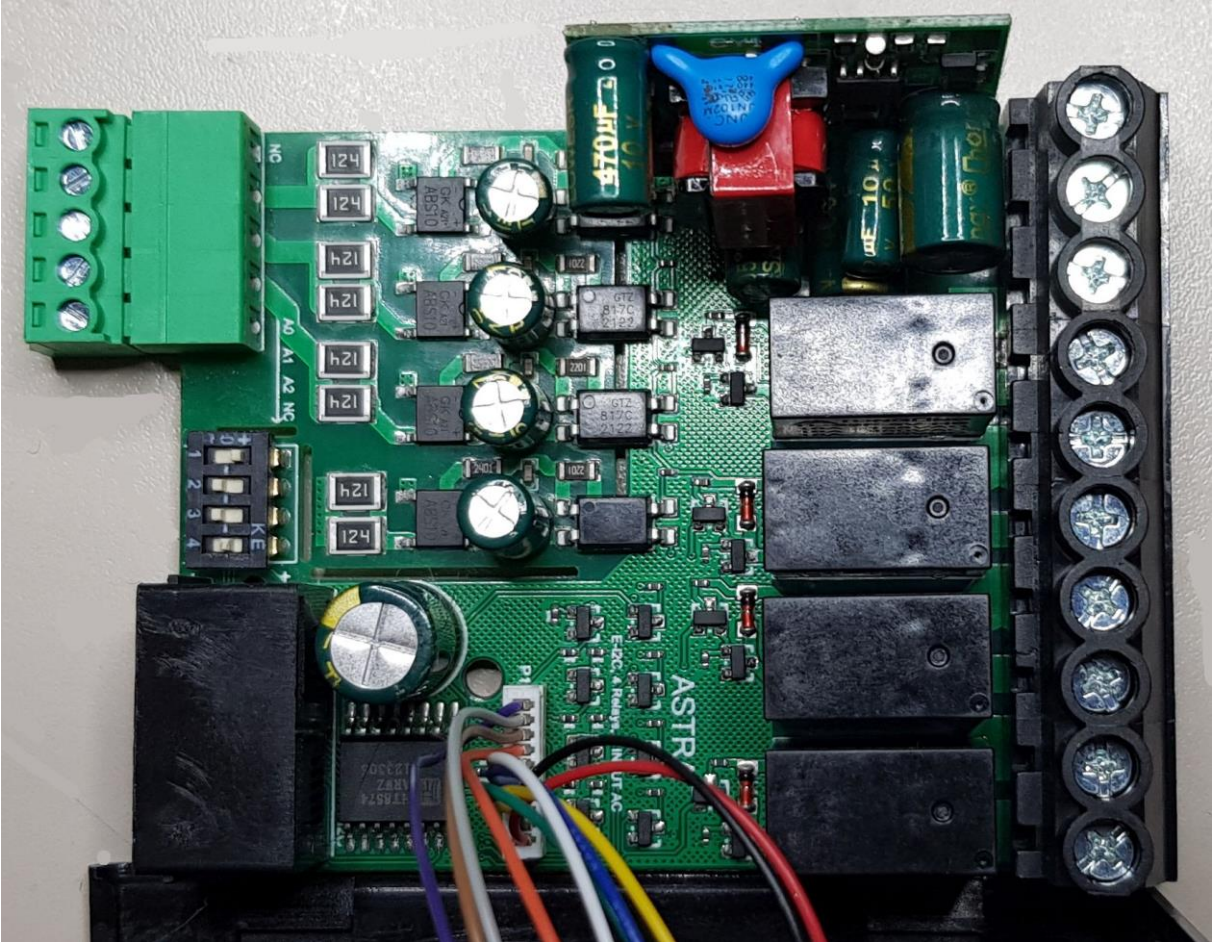
Usage of ASTRA BR4A4AL / ER4A4A Input for 230 V AC:



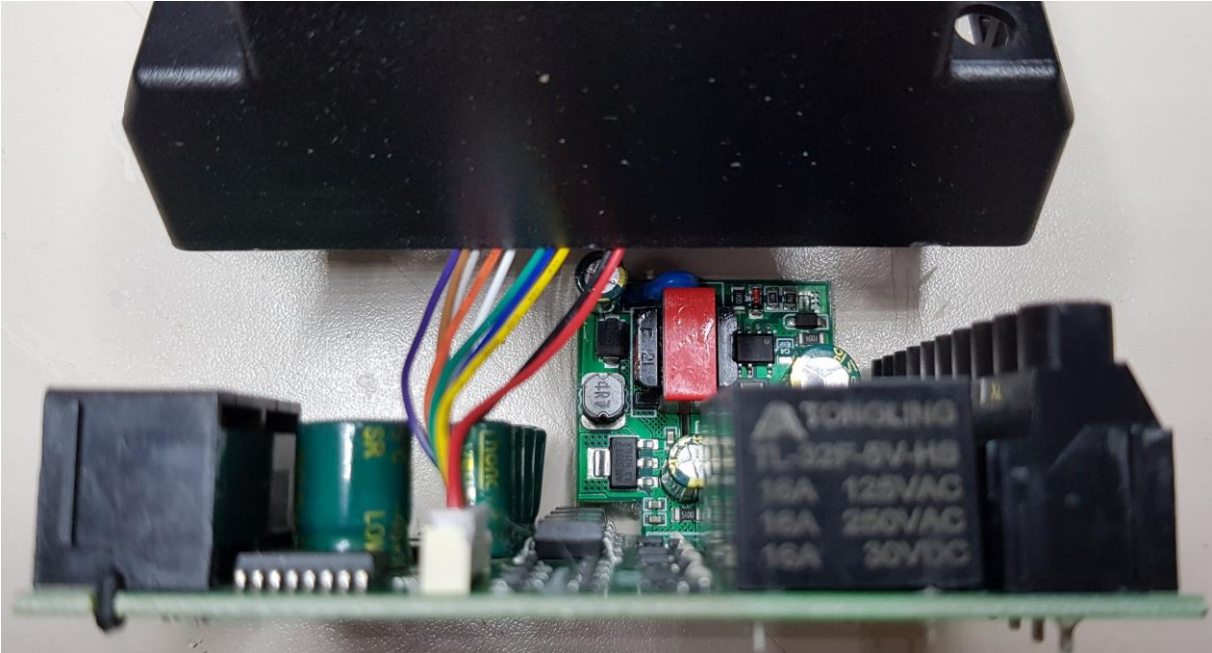
Partly schematics of ASTRA ER4D4 Input for 230 V AC:



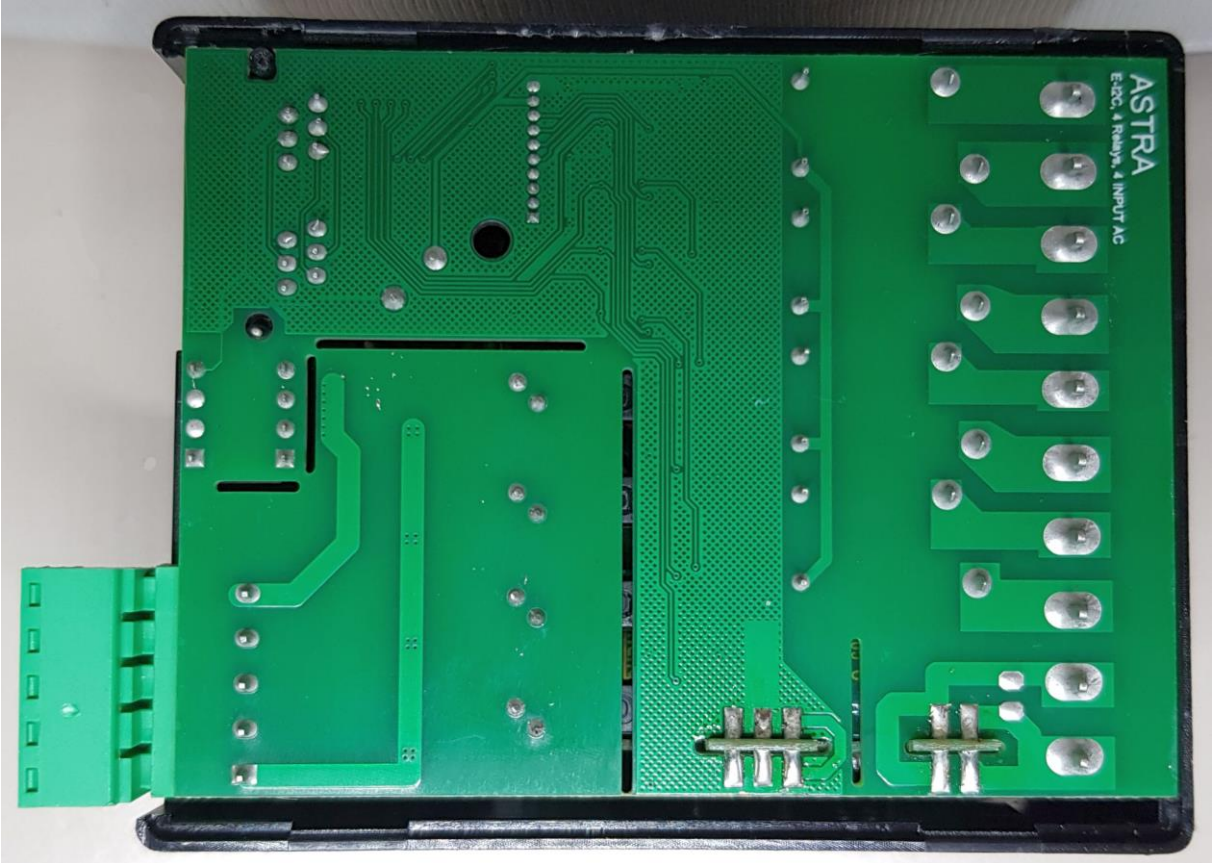
Top view of main PCB:



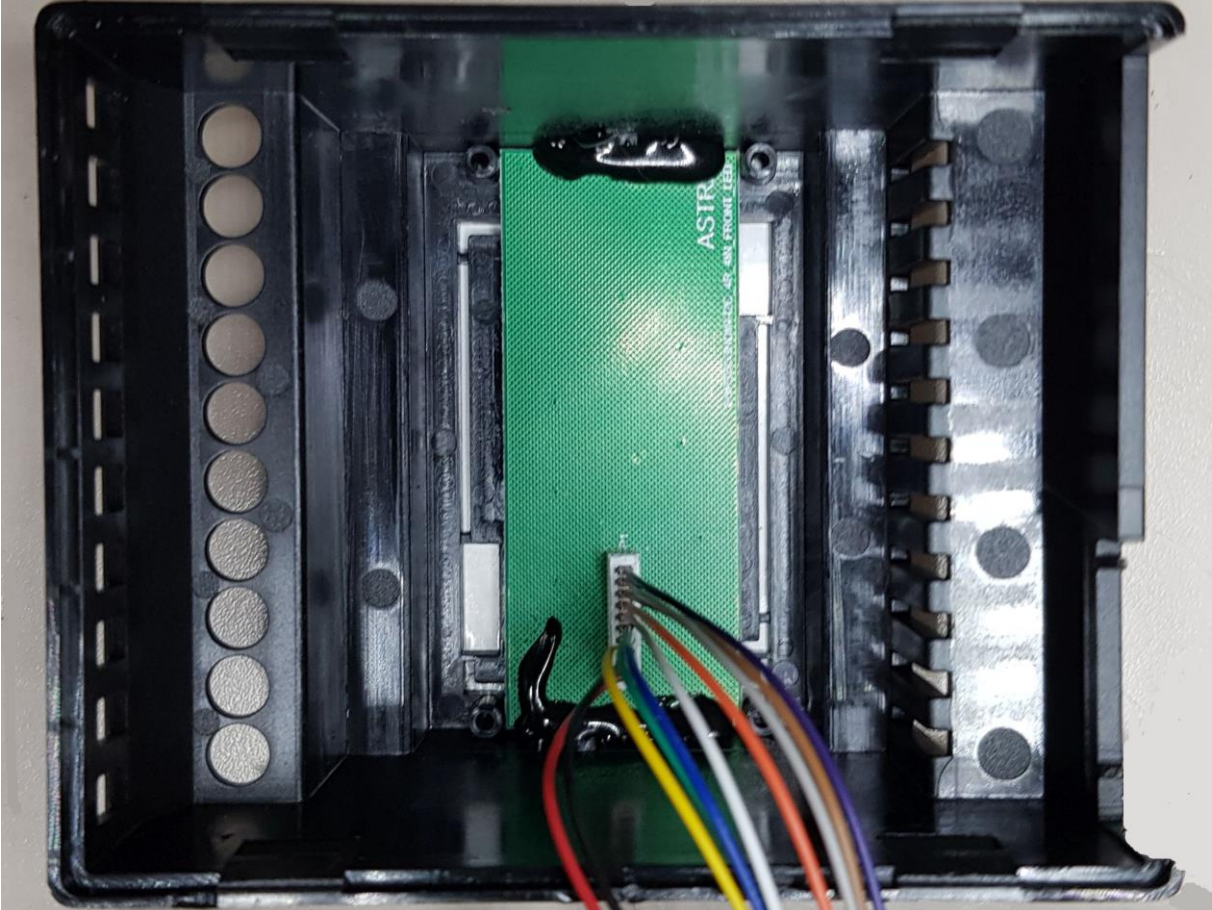
Main PCB with power supply in back:



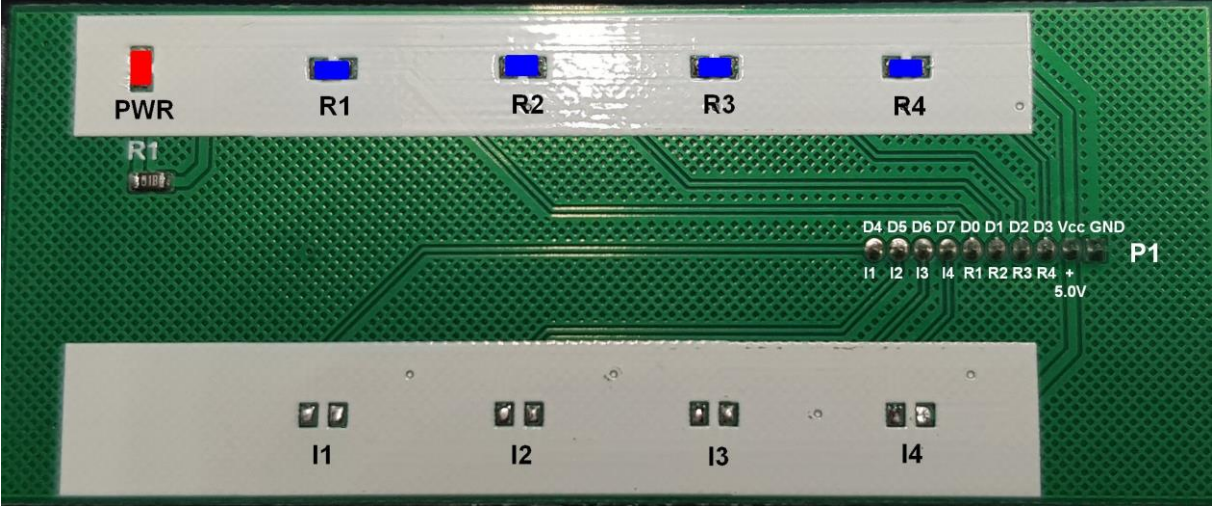
Bottom view of main PCB:



LED PCB glued in top of case:



LED PCB dismantled and view from top (input LED not mounted):



PCF8574 / PCF8574A GPIO Expander

Used by Astra expansion module

This feature is not included in precompiled binaries of Tasmota. When compiling your build add the following to `user_config_override.h`:

```
#define USE_I2C // Add support for I2C
#define USE_PCF8574 // [I2cDriver2] Enable PCF8574 I/O Expander
                    // (I2C addresses 0x20..0x26 and 0x39..0x3F)
                    // (+1k9 code)
// #define USE_PCF8574_SENSOR // enable PCF8574 inputs and outputs
//                             // in SENSOR message
// #define USE_PCF8574_DISPLAYINPUT // enable PCF8574 inputs display in Web page
// #define USE_PCF8574_MQTTINPUT // enable MQTT message & rule process
//                               // on input change detection :
//                               // stat/%topic%/PCF8574_INP = {"Time":
//                               // "2021-03-07T16:19:23+01:00",
//                               // "PCF8574-1_INP":{"D1":1}}
```

In order to use PCF8574 inputs, uncomment the last three lines.

Introduction

PCF8574 and PCF8574A are I2C 8-bit IO port extender originally designed by Philips (now NXP) but also now available from various manufacturers.

Supported I2C addresses and number of PCF8574

PCF8574 and PCF8574A are functionally identical and each can be configured to work on 1 of 8 possible I2C address. PCF8574 can use one address of 0x20 to 0x27 and PCF8574A can use one of 0x38 to 0x3F.

As of today, Tasmota driver supports:

- Up to 4 PCF8574 OR PCF8574A is supported by Tasmota allowing up to 32 additional GPIO pins.
- Addresses 0x27 and 0x38 are excluded to avoid conflict with other I2C peripheral which can't be differentiated at run-time.

If `USE_MCP230xx_ADDR` is defined, this address is reserved for MCP230XX IO expander.

The first 2 lines are mandatory to enable I2C support and including the driver in the build. The 3 other lines allow to add optional features to support inputs. By default, only the "outputs" feature is enabled.

Tasmota Configuration

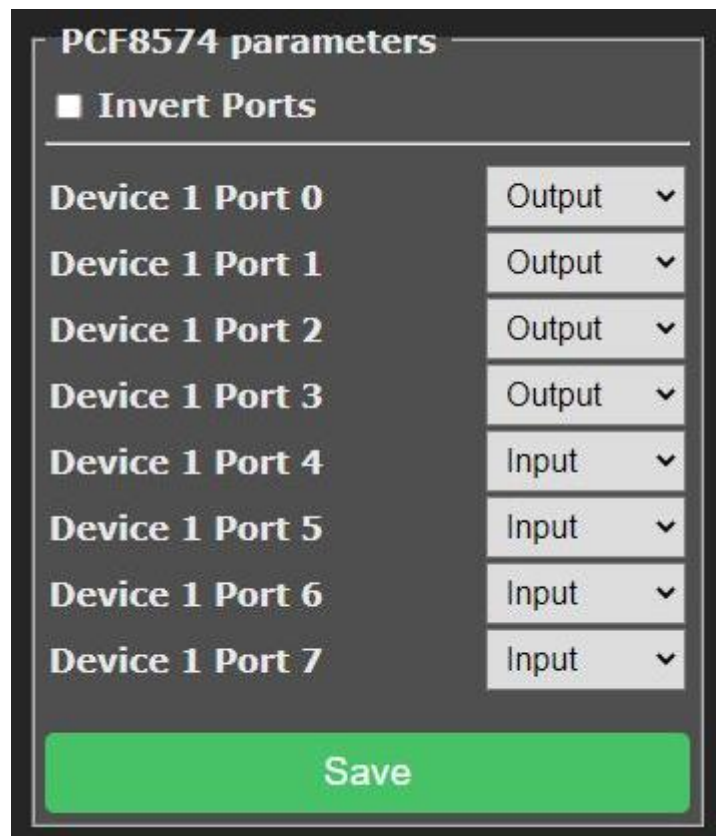
Once the firmware with the PCF8574 driver has been loaded, make sure to have it enabled with `I2Cdriver2 1`.

PCF8574 can be configured from Tasmota web GUI in "Configure" => "Configure PCF8574"

Each IO port can be configured as Input or Output in a similar way as a native GPIO of the ESP.

If you are using outputs to drive relays, it is possible to choose if the relay is activated by a HIGH signal (checkbox "Invert Ports" unchecked) or a LOW signal (checkbox checked). The selection applies to all output ports. This checkbox can also be controlled by `SetOption81 x`.

Once configuration is complete, it must be saved by clicking on the green "Save" button. Like for general ESP GPIO configuration, this will trigger a reboot of the ESP.



PCF8574 parameters	
<input checked="" type="checkbox"/> Invert Ports	
Device 1 Port 0	Output ▾
Device 1 Port 1	Output ▾
Device 1 Port 2	Output ▾
Device 1 Port 3	Output ▾
Device 1 Port 4	Input ▾
Device 1 Port 5	Input ▾
Device 1 Port 6	Input ▾
Device 1 Port 7	Input ▾
<input type="button" value="Save"/>	

It is not possible to change pin definition at run-time.

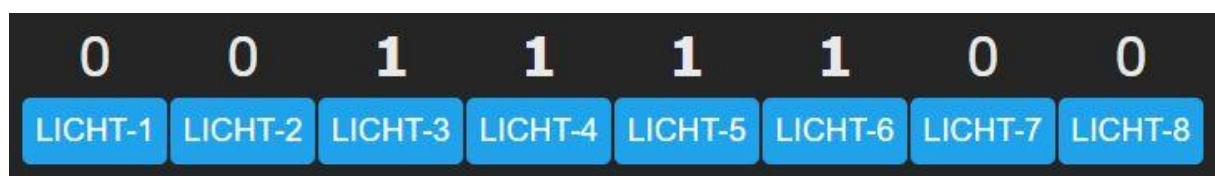
Outputs

A PCF8574 pin configured as an output support all features of a Tasmota Relay component.

It is assigned a Power index and can be controlled by `Power` command (on, off, toggle). Power indexes of PCF8574 outputs are assigned after the ESP GPIO configured as Relay. For example, if you have Relay 1 (Power1) to Relay 4 (Power4) configured on the ESP's GPIO, PCF8574 outputs will start at Power5.

A state text and an on/off button are automatically created on the Web GUI and syncs with the pin state.

All `Power` features are supported including `PowerOnState`, `PulseTime`, `Blink`, `SetOption0`, ...



Usage

Enabling `USE_PCF8574_SENSOR` adds a `PCF8574-xx` field into the JSON payload of the `tele/topic/SENSOR` message. The form of the message is:

```
{"Time": "2021-03-11T19:50:58+01:00",  
  "PCF8574-1": {"D0": 1, "D1": 1, "D2": 1, "D3": 1, "D4": 0, "D5": 0, "D6": 0, "D7": 0}}
```

As you can see, all pins are listed, including both inputs and outputs. The value reported is the digital level of the pin. If "Invert Ports" has been enabled, Power ON will be reported as 0 as the pin is at a LOW level.

As for any sensor published in the `tele/topic/SENSOR` message, it is possible to use Rules triggers such as:

```
ON tele-PCF8574-1#D0 DO something_with %value% ENDON
```

Numerical operators such as `==` can be used to compare to 0 or 1. See also [change detection](#).

PCF8574 inputs pins in the Web GUI

Enabling `USE_PCF8574_DISPLAYINPUT` will add the state of PCF8574 inputs displayed as sensors in the Web GUI. Outputs are not represented here as they are already shown as Power.



Value of pin is updated in almost "real-time".

Input Change Detection

While reporting the pin level in `SENSOR` or on the GUI is interesting, it is even better to detect pin change. This is enabled by `USE_PCF8574_MQTTINPUT`. When this feature is enabled at build time, a test will be performed every 50ms to detect if an input pin has changed. In that case, Tasmota will publish on `stat/topic/PCF8574_INP` a JSON payload with the PCF8574 index and the pin level:

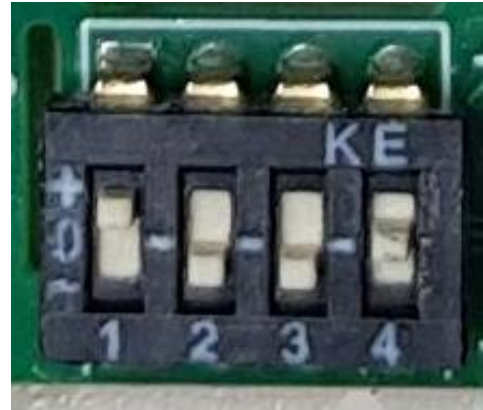
```
20:19:39.385 MQT: stat/topic/PCF8574_INP =  
  {"Time": "2021-03-11T20:19:39+01:00", "PCF8574-1_INP": {"D0": 0}}  
20:19:39.584 MQT: stat/topic/PCF8574_INP =  
  {"Time": "2021-03-11T20:19:39+01:00", "PCF8574-1_INP": {"D0": 1}}
```

This can be caught in rules such as:

Implementing a Power push "Button":

1. DIP switch needs address of module for select I2C. If you want to connect many expansion modules to controller, for each module need dedicate address. DIP switch 4 is not relevant. Follow this table:

Address	A0	A1	A2	A3
0x20	- / 0	- / 0	- / 0	n.c.
0x21	- / 0	- / 0	+ / 1	n.c.
0x22	- / 0	+ / 1	- / 0	n.c.
0x23	- / 0	+ / 1	+ / 1	n.c.
0x24	+ / 1	- / 0	- / 0	n.c.
0x25	+ / 1	- / 0	+ / 1	n.c.
0x26	+ / 1	+ / 1	- / 0	n.c.
0x27	+ / 1	+ / 1	+ / 1	n.c.



2. After selected address, pls connect expansion module to controller with cable.
3. Start controller (switch on or reset button).
4. Find in interface menu PCF8574
5. For IO0-3 select output, For IO4-7 select input. Save.
6. Power off expansion module. Connect all relays / outputs (don't connect inputs).
7. Power on expansion module. Connect all inputs.