SecSi-Energy-Monitor

Version 1.1

Hardware document

HATRIX-ELECTRONICS

TABLE OF CONTENTS

1. Introduction

SecSi-Energy-Monitor is a innovative device developed for use in smart home system. Motivation comes from need to integrate a lot of different modules on a single board and thus create more centralized system.

SecSi-Energy-Monitor provides next functionalities:

- WiFi;
- Bluetooth;
- POE (Power over Ethernet);
- Ethernet communication;
- 2x P1 Ports (Power and communication);
- I2C - for example display;
- Programming MCU over WiFi or USB-C;
- 1x Analog sensor;
- 2x Pulse sensors;
- 6x GPIOs;

2. Hardware considerations

P1 Shield board consists next modules (Figure 1):

1. MCU;
2. PSU;
3. POE;
4. USB to Serial converter;
5. Ethernet;
6. P1 Port;

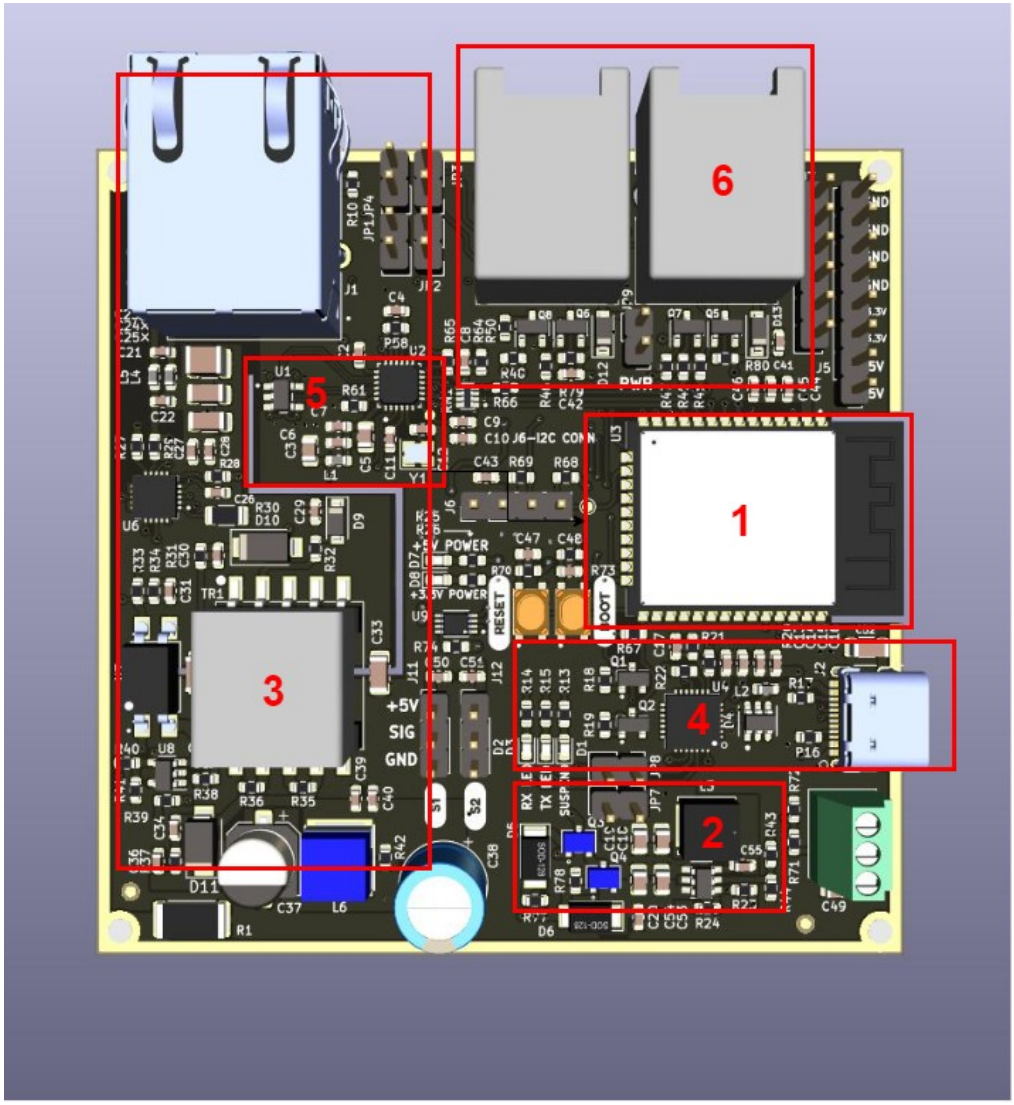

Figure 1: 3D View - Modules

PSU:

The power supply of the entire board is realized with DC-DC converter (AP62200WU-7) which works in BUCK configuration. The output voltage is +3.3V and maximum output current is 2A.

DC-DC accepts 5V at it's input from 3 different sources using control power switch implemented with two P-type mosfets. Source with the highest priority is POE, followed by USB, and the lowest priority is P1 Port. Source selection is done automatically. There is no conflicts if only one source is connected or if combinations of sources are connected.

POE:

POE can be used to power the board. Segment is realized using IC (SI3404) whose role is to convert input voltage from RJ45 and adjust it for use. The POE standard applied to the board is IEEE 802.3 Type 1 and the board is galvanic isolated (1.5kV) using a transformer.

USB to Serial:

Module is used for communication and programming of the microcontroller that is placed on the board. Its role is to convert communication between USB and UART. An important item added to this module is the auto-boot function. Auto-boot function represents programming without pressing the boot button, you only need to connect the USB and start programming, the board will do the rest and inform you when programming is finished.

Ethernet:

Ethernet is designed for speed's up to 100Mb/s in configuration "Auto-negotiation enabled".

P1 Port:

Module is used for providing power to board and communication using P1 port specification with communication baud rate up to 115 200.

3. Software consideration

MCU-ESP32:

MCU is the most important part of the board, it controls and communicates with the modules. ESP32 has integrated Wi-Fi and Bluetooth communication. Programming of the controller is done using USB through a USB to Serial converter. There are two push-buttons on the board, BOOT and RESET. The reset button resets the controller, while the boot button is used during programming. Next figure represents device block diagram and pinout (Figure 2):

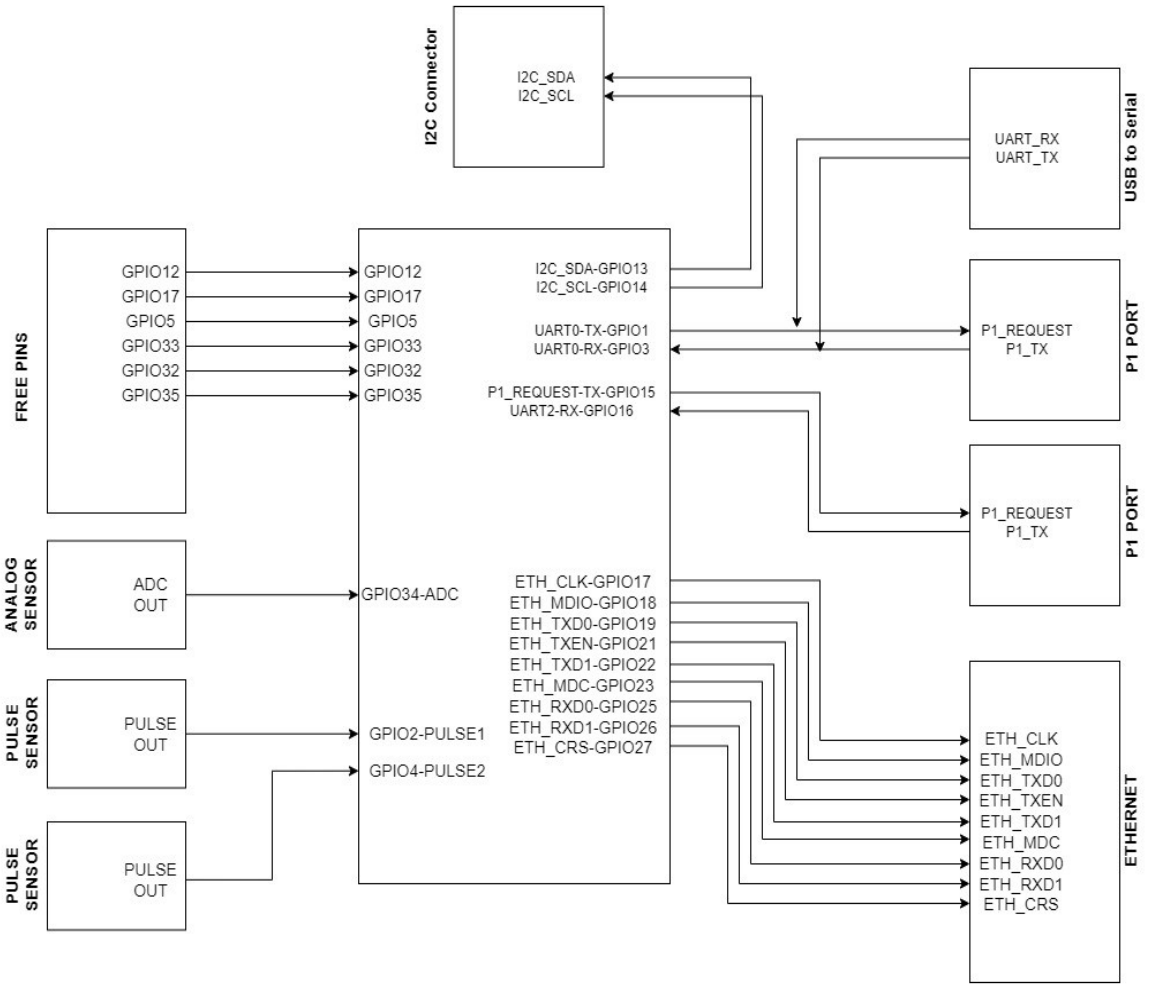

Figure 2: Block diagram and pinout

ESP32 uses UART, I2C and RMII interface (Ethernet) for communication with modules.

When programming or communicating with the board, jumpers JP7 and JP8 must be placed. UART0 is used for programming.

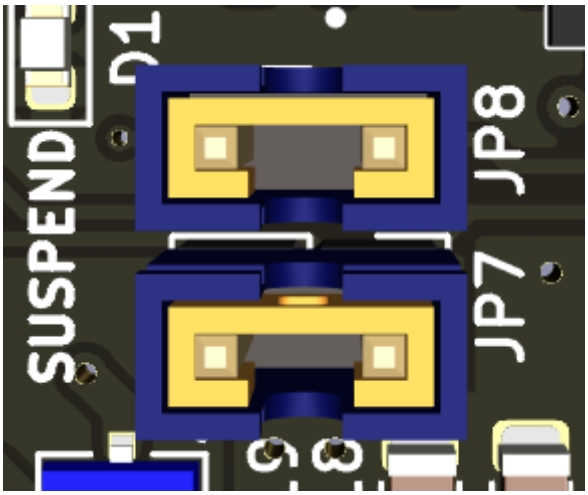

Figure 3: USB Programming of ESP32

4-pin header on board contains I2C communication pinout with power supply pins (connector J6). It can be used for displays or sensors. There is a +3.3V power supply on the connector and **communication voltage level must be +3.3V**.

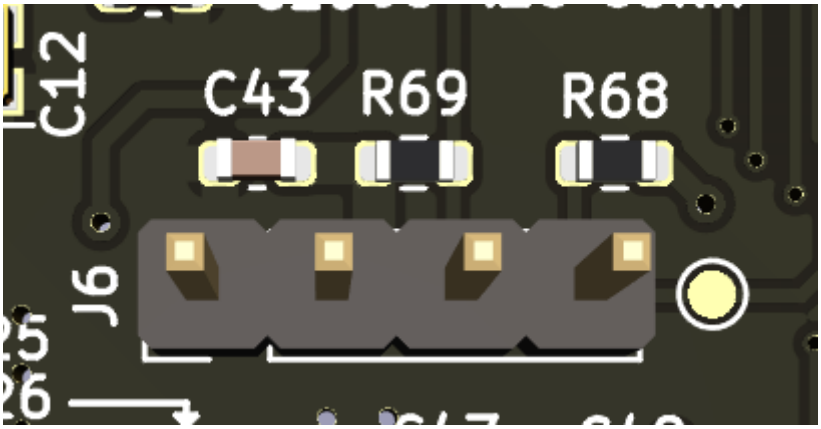

Figure 4: I2C header

Board has the possibility to connect one analog sensor. Sensor is connected to the terminal block connector (J10) and the connector has a +5V power supply. **Maximum input voltage from sensor is 5V.**

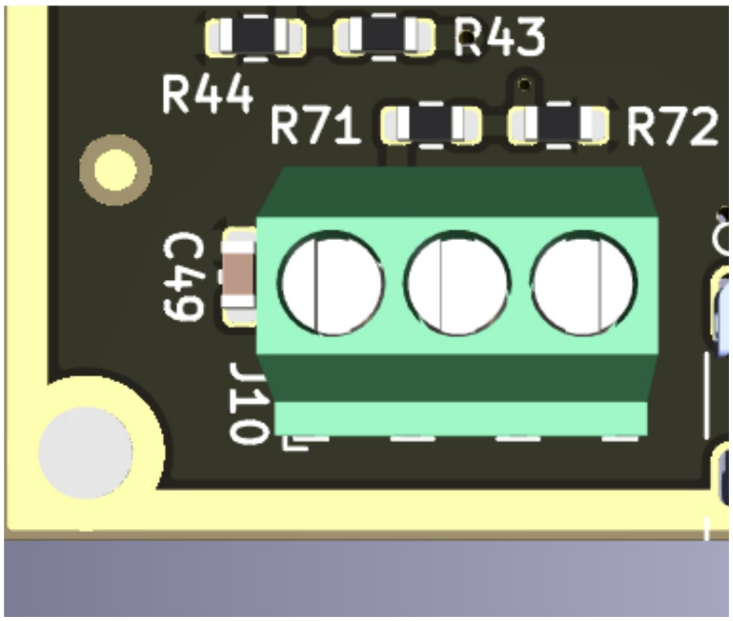

Figure 5: Terminal block for analog sensor

Pulse sensors can be connected to header connectors (J11 and J12). The power supply at connectors is +5V. Pulse sensors are designed for maximum 5V input voltage.

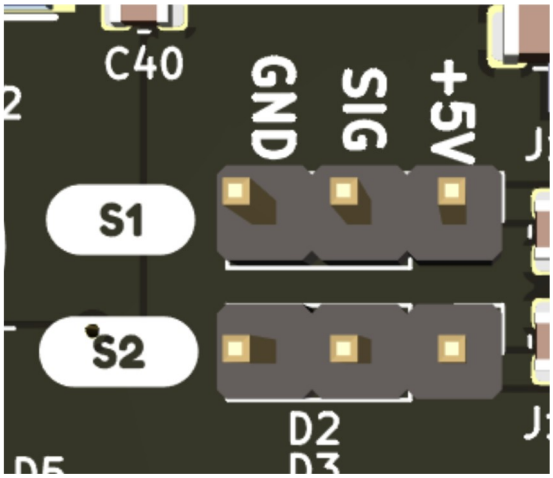

Figure 6: Headers for pulse sensors

Free pins are connected to header connector (J7), power supply pinout is placed on the header connector (J5) and contains 2x +3.3V and 2x +5V power supplies.

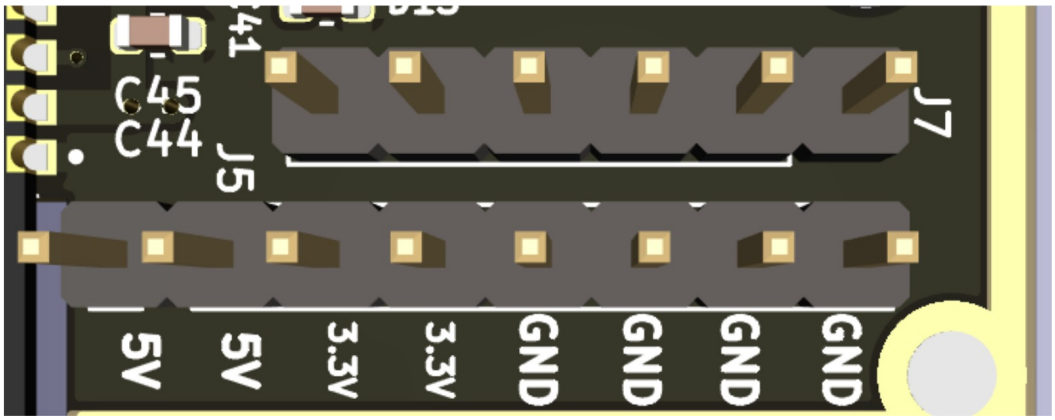

Figure 7: Free pins and power supply pins for free use

P1 Port:

P1 Port is used for communication with external modules that send data of consumption and the status of devices connected to them. There are two P1 ports on the board.

One P1 port (J4) is the main one and it has direct communication with ESP32 via **UART2**. The second P1 port (J3) can be separate, sharing **UART0** with the programming part or it can be a continuation of the main P1 Port.

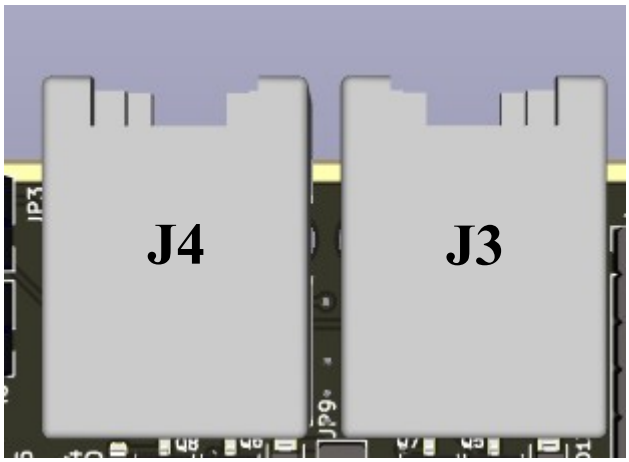

Figure 8: P1 port connectors

In the configuration where the second P1 Port (J3) is independent, it is necessary to place jumpers **JP2 and JP4.** In that configuration, it is necessary to **remove USB jumpers JP7 and JP8** so UART0 can be free to be used from second P1 Port. Communication between ESP32 and P1 Port is done via UART0.

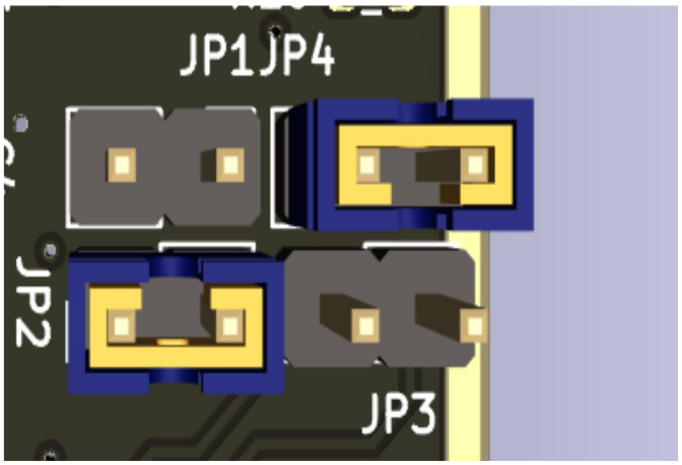

Figure 9: Second P1 port jumpers configuration – Independent mode

In the configuration where the second P1 Port is an extension of the main P1 Port, the information coming to the main P1 Port is also forwarded to the second one. Then it is necessary to place jumpers **JP1 and JP3**, as well as the jumper for the power supply **JP9** if necessary.

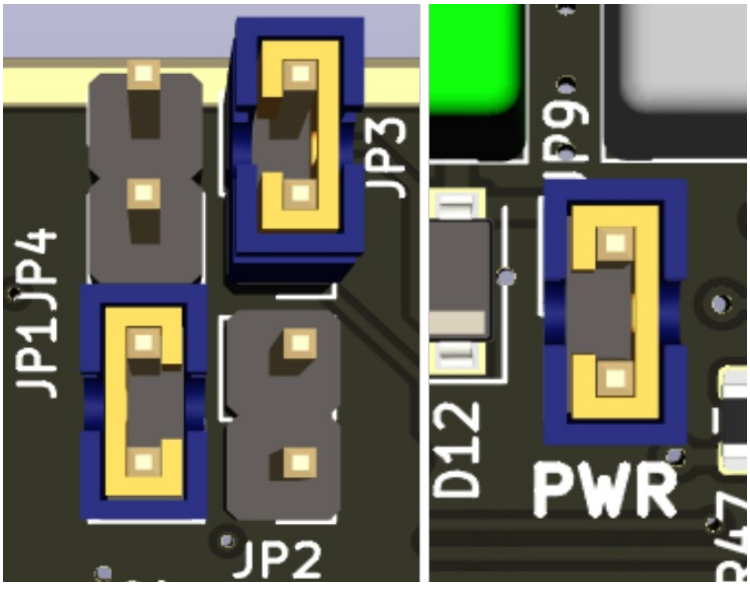

Figure 10: Second P1 port jumpers configuration – Continuation mode

4. Datasheet

MCU-ESP32:

https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32e_esp32-wroom-32ue_datasheet_en.pdf

POE-SI3404:

https://www.skyworksinc.com/-/media/Skyworks/SL/documents/public/data-sheets/si3404-datasheet.pdf

Ethernet-LAN8720:

https://ww1.microchip.com/downloads/aemDocuments/documents/OTH/ProductDocuments/DataSheets/00002165B.pdf

USB to Serial-CP2102:

https://www.silabs.com/documents/public/data-sheets/CP2102-9.pdf

PSU-AP62200WU-7:

https://www.diodes.com/assets/Datasheets/AP62200_AP62201_AP62200T.pdf