**V3.3 PCB description and documentation**  
Kenan Henriksen  
Gridlinx

07/11-2024

# Overview of components

SHEET NUMBER 1

## [1.1] Input connectors

* [1.1.1] Power input connectors
  + Screw terminal **J107** for AC-mains input
  + SM02B connector **J103** for AC connection from EVSE
  + SM03B connector **J114** for BatteryBridge connection
* [1.1.2] UI inputs from buttons
  + **J105** for button 1
  + **J106** for button 2
  + **J108** for button 3
* [1.1.3] Type-2 plug inputs
  + **J101** terminal for CP- and PP input from EVSE
  + **J113** terminal for temperature sensor input
  + **J115** terminal for motor state measurement

### [1.1] Detailed Description

#### [1.1.1]

The connectors in **[1.1.1]** are all power input connectors. The **J103** is the main AC connector, when using power from the EVSE. This must be connected to the input side of the relay, which will power the PCB from the EVSE. Alternatively, the **J107** connector can be used, if the mains will be directly connected to the PCB instead of through the EVSE.   
**NOTE:** Both must never be attached at the same time.

The **J114** and **XXX** is used to connecting the BatteryBridge, ensuring that 12V, 3.3V and 5V will always be available, even in the case of a power outage.

#### [1.1.2]

The buttons can be used to change the state of the add-on. If button 1/2/3 (**J105, J106, J108**) is pressed, the following GPIO inputs will be pulled LOW: **GPIO5, GPIO17, GPIO4**.

The current configuration only uses 2 buttons (1 and 2), which respectively change to state **charging** and **discharging**.

#### [1.1.3]

The **J101** connector is used for connecting the CP and PP lines from the EVSE. The CP line is connected by the **WHITE** cable from the plug, and the **BLACK** cable for the PP line.

The **J113** line is used for connecting the temperature sensor. The **GREY** cable is used for the measurement and the **BROWN** is used for GROUND. The output is connected to the temperature ADC [**4.1.2**].

**Note:** The following resistances is given for different conditions:  
IF (TEMP < 110) THEN R == 790 to 1279

IF (TEMP > 110) THEN R == 1280 to 1420

IF (HYSTERIES == TRUE) THEN R < 1200 BEFORE RESUMING

IF (R < 790 OR R > 1420) THEN STOP

SEE TYPE 2 PLUG DATASHEET FOR FURTHER INFORMATION

The **J115** is used for the connection of the motor. The **BLUE/RED** and **BLUE/BROWN** cable is used for connecting the motor. The **BLUE/RED** is used for locking the motor and **BLUE/BROWN** for unlocking. The motor is controlled by **GPIO24** (FIN) and **GPIO18** (RIN). The **GPIO** lines are connected to the full H-Bridge, allowing for 12V controlling of the motor, as required by the datasheet.

**Note:** The following conditions are given for the motor:

WHEN LOCK APPLY FIN FOR MAX 600ms

WHEN UNLUCK APPLY RIN FOR MAX 600ms

SEE DATASHEET FOR TYPE 2 CONNECTOR FOR FURTHER INFORMATION

IF FIN == HIGH THEN OUT\_M1 == 12V (LOCK)

IF RIN == HIGH THEN OUT\_M2 == 12V (UNLOCK)

## [1.2] Raspberry PI connector

* [1.2.1] Raspberry PI female pin header
  + **J110** Raspberry Pi connector

### [1.2] Detailed description

#### [1.2.1]

The Raspberry Pi uses the following interfaces: **SPI, I2C** and **GPIO**.

The SPI line is only used for controlling the RED-BEET [**7.1.1**], and is using the following pins: **GPIO10, GPIO9, GPIO11, GPIO23, GPIO8**.

The I2C line is used for controlling all 3 ADCs [**4.1.1**]/[**4.1.2**]/[**4.1.3**] and the screen [**1.4.3**]. The line is using the pins: **GPIO2, GPIO3, GPIO7**.

The GPIO lines are divided into different parts for controlling:

**Buttons (1/2/3)** are using **GPIO5, GPIO17** and **GPIO4** respectively.

For controlling the EVSE state, the following pins are used: **GPIO27** and **GPIO22**. When pulling **GPIO27** HIGH, the voltage level on the CP input line drops from 12V -> 9V, i.e. Changing CP state from A -> B. When pulling **GPIO22** HIGH, the voltage level on the CP input line drops further. This results in a state change from B -> C, letting current flow from the EVSE. **Note:** GPIO27 must be pulled HIGH and maintained HIGH before pulling this pin HIGH.

The following GPIO lines are switching between sending the CP and PP line from the EVSE through to the EV and sending the generated CP and PP line from the add-on through: **GPIO6, GPIO13, GPIO19, GPIO26**.  
If **GPIO13** is pulled HIGH, the PP line from the EVSE is sent through, and if **GPIO6** is pulled HIGH, the generated PP line is sent through. If **GPIO19** is pulled HIGH, the CP line from the EVSE is sent through, and if **GPIO26** is pulled HIGH, the generated CP signal is sent through.

Controlling the relay is done through **GPIO14** and **GPIO15**. Only one of these pins is needed for controlling the relay. In the current setup **GPIO15** is used.

**Note:** Circuit description:

The circuit is an open-colletor circuit ie. the coil at the relay is always at 12V.

When closing the relay pull RelayCont1 (**GPIO15**) or RelayCont2 (**GPIO14**) (whichever is connected to the relay) HIGH.

**GPIO18** and **GPIO24** are used for controlling the motor. When locking the motor **GPIO24** is pulled HIGH, and when unlocking the motor **GPIO18** is pulled HIGH. See section [**1.1.3**]for further details.

**GPIO20** is used for generating the PWM signal for the CP line. This pin must use a frequency of 1kHz and must be turned on when SLAC must be established.  
**Note:** Use 5% OR 100% duty-cycle for SLAC.

## [1.3] Reference LEDs

* [1.3.1] LEDs
  + UI LEDs for 3.3V, 5V and 12V

### [1.3] Detailed description

#### [1.3.1]

The LEDs are used to verify correct voltage levels on the board. If all 3 LEDs are illuminated, the correct voltage levels are on the board.

## [1.4] Output connectors

* [1.4.1] Plug and motor connectors
  + **J112** for relay coil connection
  + **J111** for Type-2 plug motor controlling
* [1.4.2] Output connectors for ISO15118/IEC61851 communication
  + **J104** for CP output and Protected Earth
  + **J102** for PP output
* [1.4.3] Display connector
  + **J109** for I2C communication interface for Joy-IT display

### [1.4] Detailed description

#### [1.4.1]

The **J112** connector is used for connecting the relay. Only cables for pin 1 and pin 2 are connected, as only 1 pin is used for controlling the relay.  
**Note**: See section [**1.2.1**] for further information regarding GPIO configurations.

The **J111** is used for connecting the motor in the plug to the PCB.  
**Note**: See section [**1.2.1**] for further information regarding GPIO configurations.

#### [1.4.2]

The **J104** connector is used for the connection of the CP output line and the PE for the EV. The **J102** connector is used for the connection of the PP output line.  
**Note**: See section [**1.1.3**]for cable configuration.

#### [1.4.3]

The **J109** connector is used for connecting the Joy-IT screen. The connector is a JST type connector, and the cables are wired directly to the pins, therefore proper connection is always ensured.

**Note**: See section [**1.2.1**] for interface description.

## [1.5] Mounting holes

* [1.5.1] Mounting holes for box mount and BatteryBridge mounting
  + **H101-H102** for box mounting
  + **H103** for box mounting and BatteryBridge mounting
  + **H104** for BatteryBridge mounting

SHEET NUMBER 2

## [2.1] CP switches and EV simulator circuit

* [2.1.1] Switches for CP line and PP line
  + **U203A** switch for ISO communication
  + **U206A** switch for IEC communication
  + **U204A** switch for PP line for IEC communication
  + **U207A** switch for PP line for ISO communication
* [2.1.2] Power inputs for switches
* [2.1.3] EV simulator circuit
  + **U205A** changes CP state A -> B
  + **U208A** changes CP state B -> C

### [2.1] Detailed description

#### [2.1.1]

To switch between the CP/PP lines and EV simulation ADG417 Analog Switches are used. These switches can handle negative voltages up to ±22V, therefore making them a good fit to control the CP line. The switches are active LOW and need 5V logic levels for controlling. The power inputs must be tied to 5V for controlling, and ±12V for controlling CP line voltage levels.

**Note**: See Section [**2.2**]for further description of the high-side switches.

**Note**: See section [**2.3**]for further description of the -12V power input circuitry.

#### [2.1.2]

Power inputs for the ADG417 switches. Section [**2.4**] describes decoupling capacitors.

#### [2.1.3]

This circuit is the EV simulator circuit, as described in IEC 61851-1. The circuit is used for simulating an EV connection to the EVSE, therefore leading to power transfer from the EVSE. The analog switches are used to control the circuit.

## [2.2] Switch enable circuits

* [2.2.1] Control for CP state B -> C
* [2.2.2] Control for ISO communication
* [2.2.3] Control for PP line for ISO communication
* [2.2.4] Control for PP line for IEC communication
* [2.2.5] Control for IEC communication
* [2.2.6] Control for CP state A -> B

### [2.2] Detailed description

The switch-enable circuits that allow 5V controlling of the analog switches, as required by the datasheet. The circuit works as a high-side switch, therefore when pulling a transistor base HIGH, the output of the switch will be pulled LOW, as required by the analog switches.

## [2.3] Charge pump circuit

* [2.3.1] Charge pump 12V -> -12V inverter circuit

### [2.3] Detailed description

#### [2.3.1]

This circuit is used for inverting the 12V into -12V. To drive OP-AMPs, a transistor with its base must be connected to ground, and its collector is connected to the output i.e. -12V. The circuit can deliver a maximum of 100mA, as also documented in the schematic of the PCB.

## [2.4] Decaps

* [2.4.1] Decaps for switches

### [2.4] Detailed description

#### [2.4.1]

Decoupling capacitors for the switches. Might not be needed, but good practice to always place decoupling capacitors near power input pins. These are all 100nF capacitors, and are placed near all input pins on the analog switches.

SHEET NUMBER 3

## [3.1] AC/DC converter

* [3.1.1] Mains AC to 12V DC converter

### [3.1] Detailed description

#### [2.3.1]

This AC/DC converter converts the 230V AC into 12V DC for the board. The current configuration uses a module to lower the complexity of the board.

## [3.2] 12V -> 5V DC/DC converter

* [3.1.1] 12V -> 5V DC/DC converter from either BatteryBridge or AC/DC converter

### [3.2] Detailed description

#### [3.2.1]

For board version 3.3 and later, a buck converter is used for 12V -> 5V conversion for the board. This converter has been chosen due to higher efficiency than the old version as described in section [**3.4**].

## [3.3] 12V -> 3.3V DC/DC converter

* [3.1.1] 12V -> 3.3V DC/DC converter for RED-BEET and other 3.3V circuitry

### [3.3] Detailed description

#### [3.3.1]

This circuit is a high-performance 12V -> 3.3V buck converter with both first and second stage filtering for the RED-BEET. This specific converter is used as recommended by 8Devices for this specific module. It also supplies the rest of the board with 3.3V and can deliver 1A.

## [3.4] 12V -> 5V DC/DC converter (old design)

* [3.1.1] 12V -> 5V DC/DC converter from either BatteryBridge or AC/DC converter

### [3.4] Detailed description

#### [3.4.1]

This is the old 12V -> 5V converter. It is still on the board, if the buck converter is nonfunctional, however will not be placed as shown by the DNP symbol.

SHEET NUMBER 4

## [4.1] ADCs

* [4.1.1] ADC for reading the PP line
* [4.1.2] ADC for temperature reading in plug
* [4.1.3] ADC for reading the voltage level at the CP output line

### [4.1] Detailed description

#### [4.1.1]

This ADC converts the PP resistance into digital signal for the Raspberry Pi. The Address is **0x50**.

#### [4.1.2]

This ADC converts the resistance of the temperature sensor output into digital signal for the Raspberry Pi. The address is **0x51**. The resistance and its respective voltage is shown in notes below.

**Note**: IF (R == 790 to 1279) THEN VIN\_ADC3 == 1.457V to 1.852V

IF (R == 1280 to 1420) THEN VIN\_ADC == 1.852V to 1.936V

#### [4.1.3]

This ADC converts the voltage level on the CP line into digital signal for the Raspberry Pi. The address is 0x52.

SHEET NUMBER 5

## [5.1] PWM generation- and read-circuit

* [5.1.1] PWM generation for the CP line
* [5.1.2] Voltage divider circuit for reading the CP voltage level

### [5.1] Detailed description

#### [5.1.1]

This circuit is a complete copy of the circuit described in the DB2605 EV Charging Controller Datasheet (section 5.1). It is a basic OP-AMP circuit, which gives a 12V HIGH output when PWM signal is HIGH and -12V output when PWM signal is LOW.

#### [5.1.2]

This circuit is a copy of the circuit described in the DB2605 EV Charging Controller Datasheet (section 5.2). The circuit has been modified to be in the range of 0V-3.3V for the ADC. The output of the circuit is connected to the input of the ADC in section [**4.1.3**]

SHEET NUMBER 6

## [6.1] Relay control circuit

* [6.1.1] Control circuit for relay

### [6.1] Detailed description

#### [6.1.1]

This circuit is used for controlling the 12V coil on the relay. The diodes is required for when the coil collapses, so that the pins on the transistor will not be destroyed.

**Note**: See section [**1.2.1**] for further circuit description.

## [6.2] Motor control circuit

* [6.2.1] H-bridge motor driver

### [6.2] Detailed description

#### [6.2.1]

This motor driver is a full H-bridge. It is used to control the 12V motor in the plug.

**Note**: See section [**1.2.1**] and [**1.1.3**] for further description.

SHEET NUMBER 7

## [7.1] RED-BEET module

* [7.1.1] RED-BEET module for Homeplug Green PHY communication

### [7.1] Detailed description

#### [7.1.1]

This circuit is used for sending the Homeplug Green PHY communication through the CP line. The module is bought from Codico and is made by 8Devices. The module is a finished module, except for the 1:1:1 transformer on the output lines. It connects directly to the analog switches, so that the CP line can be switched on by these. The SPI lines have termination resistors of 0 ohms, but can be changed in case of problems with the communication for the RED-BEET.

## [7.2] Zero-crossing circuit

* [7.2.1] Zero-crossing circuit for RED-BEET module

### [7.2] Detailed description

#### [7.2.1]

This circuit is the zero-crossing circuit used to align the Homeplug Green PHY protocol with the frequency of the grid. This is needed for longer charging sessions, as small deviations in the clock frequency can lead to package losses. The circuit is further described in the documentation of the RED-BEET module.