Simple EVSE Wallbox

datasheet

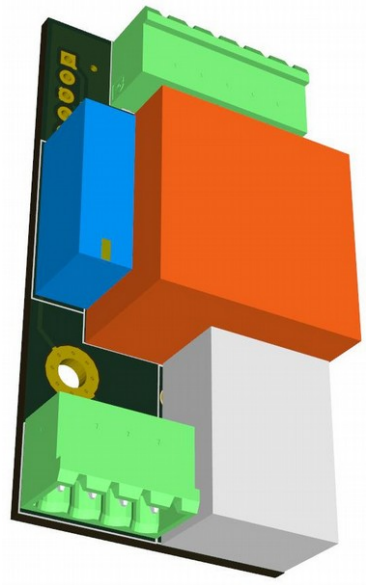


Table of Contents

Last document update: February 9, 2016

Introduction

EVSE stands for *electric vehicle supply equipment.* It is an element that supplies electric energy for the recharging of electric or plug-in vehicles.

Read me first

The EVSE board is supplied with default 32A settings. Please check the chapter "*Features"* for more information about further possibilities of changing maximum charging current. 220 Ohm R_{PP} is included with the kit (pre-hard-wired).

Theory of operation

Pilot signal duty cycle provided by EVSE defines maximum charging capacity. The car can define several states by pulling the pilot signal down to certain voltage levels (3V, 6V, 9V). Based on this feedback EVSE will trigger the relay for the vehicle to charge or evaluate the state as an error (electricity will not be provided to the output socket/connector).

For more information please check:

- http://en.wikipedia.org/wiki/IEC_62196
- http://en.wikipedia.org/wiki/SAE_J1772
- https://github.com/kortas87/simple-evse/wiki (https://code.google.com/p/simple-evse/)

Resistance PP-PE (max cable throughput)		
Resistance [ohm]	**Current limit [A]**	**Wire cross-section [mm²]**
> 1500 *	6	--
1500	13	1.5
680	20	2.5
220	32	6
100	63	16
<100 **	80	–

* no resistor connected

** recommended ~50 Ohm

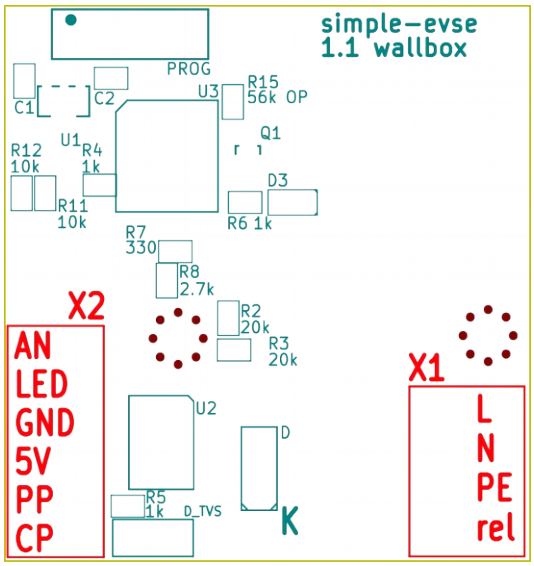
Compatibility

- Tesla Model S
- Nissan Leaf
- Mitsubishi iMiev (Peugeon iOn, Citroen cZero)
- Opel Ampera
- eGolf
- Mitsubishi Outlander PHEV
- Citroen Berlingo Electrique
- and others

Board description

Parameters	
Dimension (without connectors)	60 x 30 x 25 mm
Mounting holes spacing	30 x 20 mm
Supply voltage	90 – 265 VAC
Power consumption	<1W
Relay	5A 250V
Operating temperature	-20 to 80 °C
Protection index	IP00
Weight	80 g
Onboard resistor for external LED	1k

Simple EVSE board has 2 connectors. 4-pin X1 for high voltage side and 6-pin for signaling wires and control purposes. There is also PROG connector used for flashing new firmware (ICSP) and can be further used for adjusting EVSE current (see Features chapter).



X1, X2 and PROG connector

X1 connector		
pin	**name**	**description**
L	phase	230V power supply for EVSE board and external contactor
N	neutral	
PE	protection-earth	Ground reference
rel	relay output	This output drives coil of an external contactor. Maximum allowed current is 3A.

X2 connector		
pin	**name**	**description**
AN	analog input	Used for button or current sensor input
LED	external LED	Includes 1k resistor onboard, connects to LED anode against ground
GND	ground	Ground reference
5V	5V power output	Used as a power supply for external current sensor (max 40mA)
PP	proximity pilot	To vehicle connector
CP	control pilot	To vehicle connector

PROG connector		
pin	**name**	**description**
1	V_{PP}	ICSP** / RFU* [marked by rectangle]
2	V_{DD}	5 V
3	GND	Ground
4	DAT	ICSP** / current boost function
5	CLK	ICSP** / current limit function

* reserved for future use

** firmware upgrade interface

Features

Current limitation – using PROG pin 5

Check the table for details of how you can set the maximum current. This settings will override PIN 4 settings.

PROG connector	connection	current limit
pin5 > 4.5 V	open (internal pull-up only)	32 A [default]
2.5 V < pin5 < 4.5 V	100-200k* resistor to GND	25 A
0.2 V < pin5 < 2.5 V	~3-20k* resistor to GND	16 A
pin5 < 0.2 V	tied to GND	10 A

* resistor value may differ since internal pull-up has no defined value from production (~50k supposed), in most cases 100k for 25A and 5k for 16A is recommended

Current boost – using PROG pin 4

Check the table for details of how you can set the maximum current.

PROG connector	connection	current limit
pin4 > 4.5 V	open (internal pull-up only)	32 A [default]
2.5 V < pin4 < 4.5 V	100-200k* resistor to GND	48 A
0.2 V < pin4 < 2.5 V	~3-20k* resistor to GND	63 A
pin4 < 0.2 V	tied to GND	80 A

* resistor value may differ since internal pull-up has no defined value from production (~50k supposed), in most cases 5k for 63A and 100k for 48A is recommended

Precise current setting with Analog Input

Press and hold button connected to Analog input AN of X2 for a few seconds until LED starts to blink rapidly. Then count LED blinks which correspond to number of ampers. Please note that this limit will be set until you reboot the EVSE (make a power cycle). Button is connected the way that it pulls the signal down to ground (level <1V).

Current limitation based on external hall sensor (experimental)

!! requires testing firmware !!

Simple EVSE Wallbox can tell the vehicle to follow the actual PV power plant production. Only hall sensor (Amploc 25) output must be connected to the analog input of the board (AN). Sensor is powered from 5V. Current will be gradually increased when overflow to a public network is detected. When PV production decreases then duty cycle will be reduced down to a minimum of 6A charging current. Please also check possibilities of control using MODBUS protocol at the end of this document.

External LED

Using the pin LED you can directly connect LED to indicate EVSE status. The output includes 1k resistor. External LED has the same indication function as LED onboard.

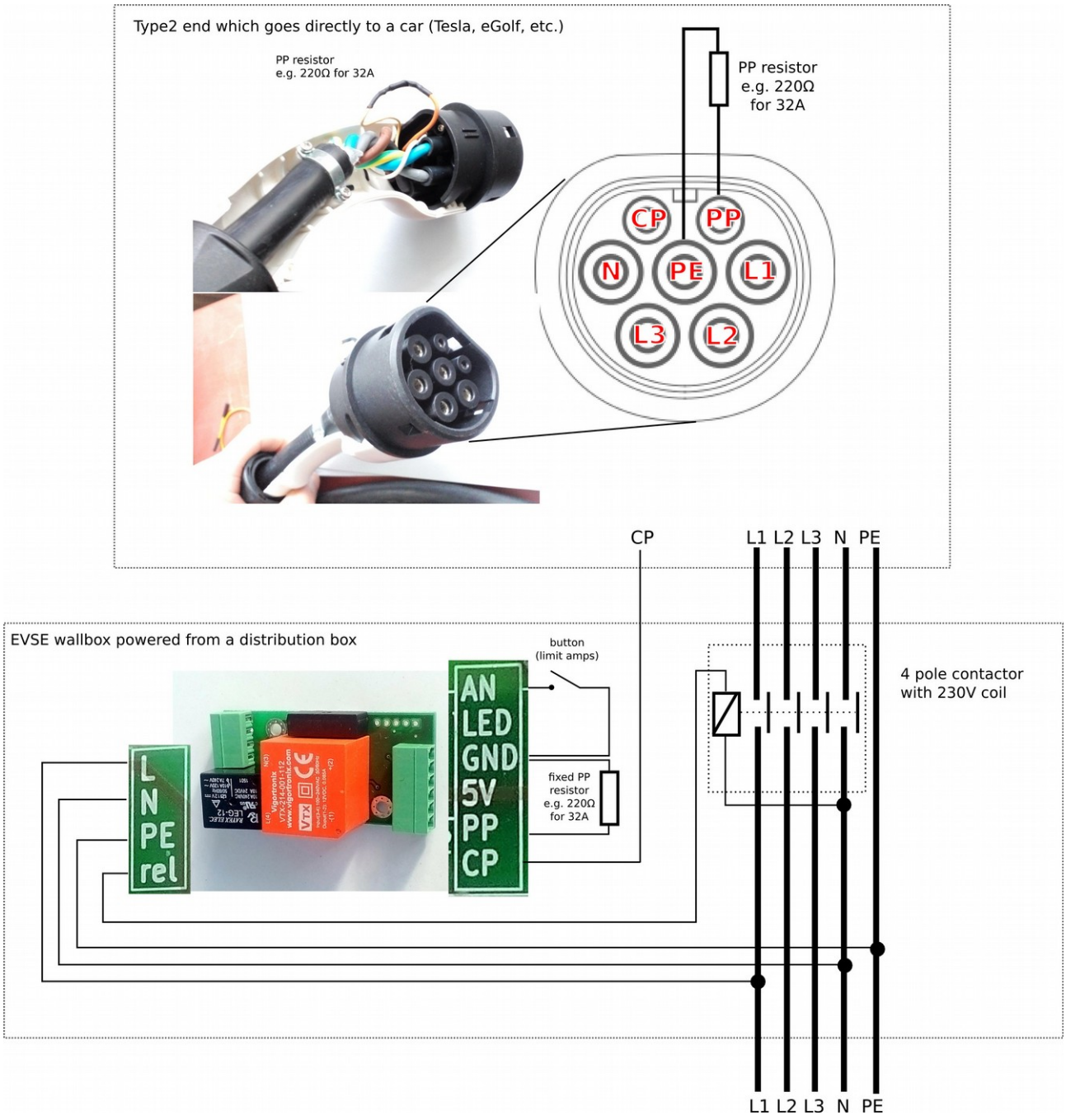
LED	connection
1x fast every 2s	pilot signal is steady +12V, no vehicle connected
2x fast every 2s	PWM signal is generated, vehicle is present
1x long every 2s	vehicle requested power, contactor is ON
20x fast in 0.5s	enter or leave current setting mode
1x each 0.3s	current setting mode – one more amper set

Application examples

In these application examples we suppose that customer uses appropriate contactor with 230V coil. Please check section “Recommended contactors”.

3phase wallbox including a cable and plug

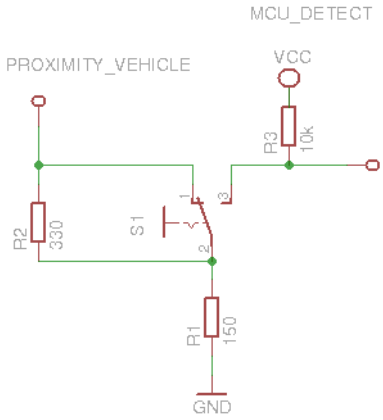
In this example we make 3 phase Wallbox using DSIEC-2E cable. PWM duty will be limited by the size of R_{PP} (refer to the *Theory of operation* chapter). If you do not connect any R_{PP} current will be limited to only 6A. If your EVSE includes a cable which is fixed then you can hard-wire R_{PP} resistor for cable's nominal value.



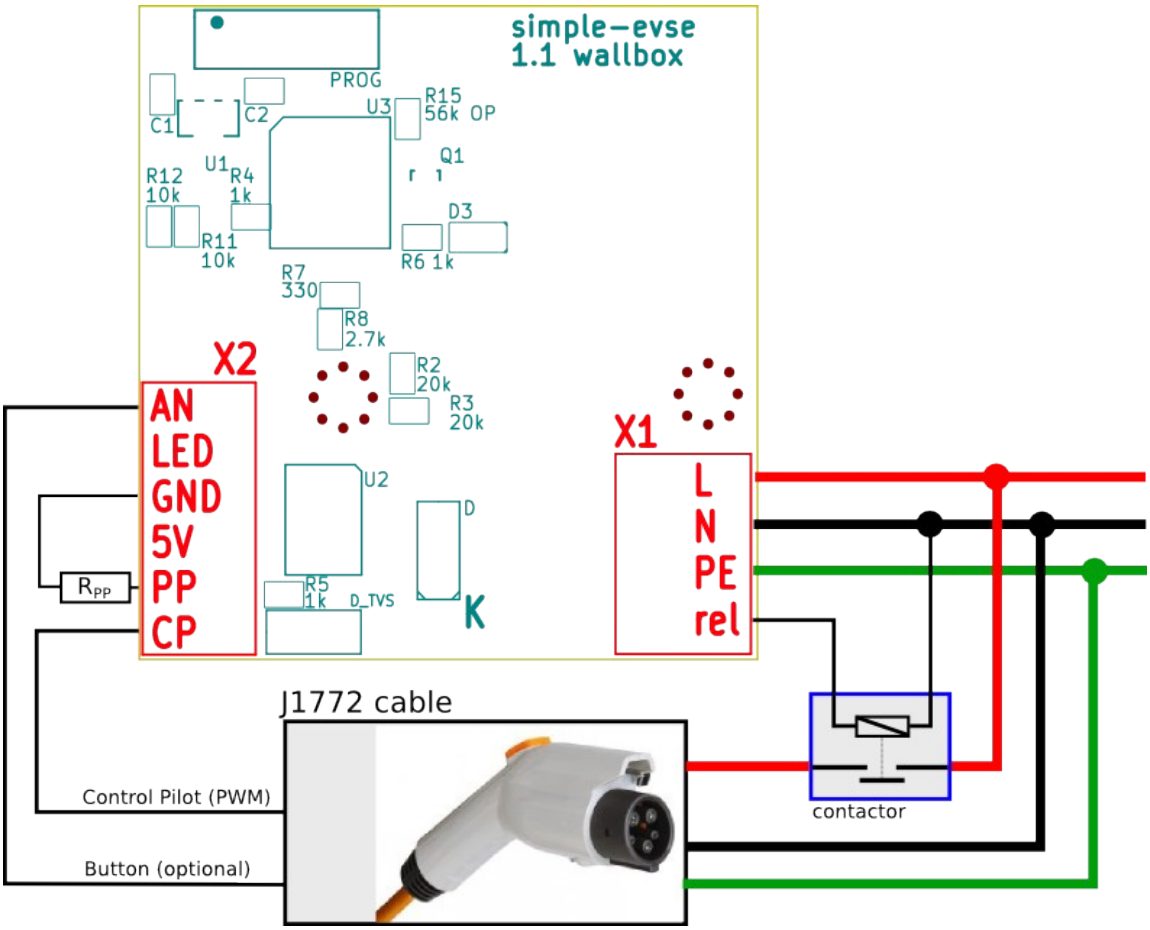
32A EVSE with connector J1772

With EVSE Wallbox board you can quickly build a charging station for your Nissan Leaf 6.6kW or any other vehicle equipped with J1772 plug.

Optional Analog Input connection:
The internal J1772 connection allows to use S1 proximity button as an auxiliary button for EVSE. With the help of this button you can easily change charging current with the smallest step of 1A (see Features - Precise current setting).



J1772 proximity button connection detail



J1772 connector - signal and power wires

Recommended contactors

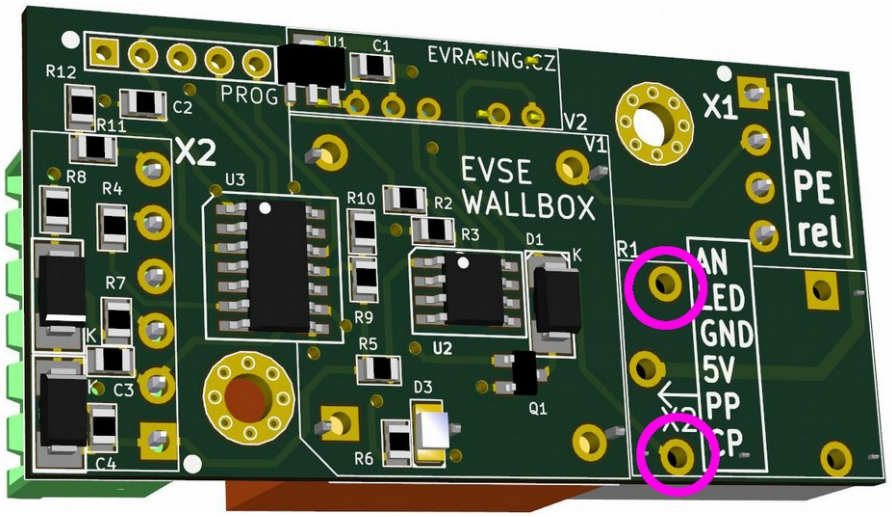
You should use relays / contactors equipped with 230V coil and connect them directly to the board connector X1. Here are some examples which relays can be used. The most common one (4pole contactor 25A) can be easily obtained in local electrical accessories shop (Conrad, K&V Elektro...)

- 3-phase, maximum current 20A VS420 (e.g. Tesla 11 kW)
  http://eshop.elkoep.com/vs420-40-230v-ac-detail-YU00000101.aspx
- 3-phase, maximum current 40A VS440 (e.g. Tesla 22 kW, ZOE 22 kW)
  http://eshop.elkoep.com/vs440-40-230v-ac-dc-detail-OV00000101.aspx
  http://www.tme.eu/cz/details/rik40-22-230/stykace-hlavni-moduly/iskra-sistemi/ik40-22230vac/
- 1-phase, maximum current 16-20A (e.g. Peugeot iOn 3 kW)
  http://eshop.elkoep.com/vs120-10-230v-ac-dc-detail-CM60000101.aspx
  http://www.tme.eu/cz/details/rik20-11-230/stykace-hlavni-moduly/iskra-sistemi/ikd20-11230vac/
  http://www.tme.eu/cz/details/rt314730/elektromagneticka-rele-miniaturni/te-connectivity/0-1393240-7/
- 1-phase, maximum current 40A (e.g. Nissan Leaf 6.6 kW)
  http://www.tme.eu/cz/details/r40n1021855220/elektromagneticka-rele-pro-velke-proudy/relpol/r40n-1021-85-5220/
  or use normal 4-pole contactor

Using external 12V relay

In some cases it would be possible to use another 12V coil relay (maximum 0.8W coil consumption), however this approach requires desoldering of the original relay and is recommended only to advanced users. Automotive relays can handle sufficient currents however their voltage rating is not high enough.

Pins marked by pink are connected to a 12V relay coil:



J1772 EVSE with 32A CEE 5pin plug (sent by 1gachren)

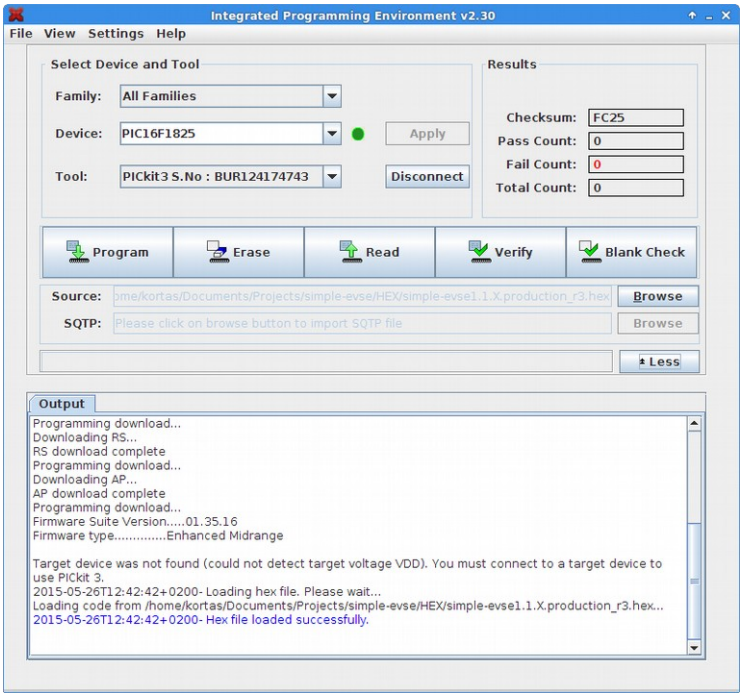
Schuko 16A Tesla charging plug (sent by sefik)

Flashing new firmware

By flashing new firmware you could possibly upgrade the EVSE to support some future improvements which are currently not known. Microchip IPE utility + PICKit 3 hardware is required for this purpose. (https://microchip.wikidot.com/ipe:what-is-ipe)

Using PICKit3 to flash new software

1. install MPLAB X software (http://www.microchip.com/mplabx/)
2. enable advanced mode in settings and check “Power Target Circuit from Tool” in “Power” tab on the left
3. put device id “PIC16F1825” and connect to your PICKit
4. select HEX file and click “Program” the device



Special “MODBUS firmware”

It is possible to download special “MODBUS firmware” (please contact us for more details).

Some values can be then read or written over MODBUS protocol (UART interface). PROG pin header is used for this purpose with following pinout:

PROG connector	**connection**
pin3	GND
pin4	TX
pin5	RX

This feature is useful for further development and testing with Simple EVSE Wallbox board and can be also a great way to interface other devices such as Raspberry PI, Ethernet UART bridges (for example WIZnet serial-to-ethernet boards), various WiFi modules etc.

Description of registers:

Register address	**R/W**	**Description**
1000	R/W	Actual amps value (6-80A)
1001	R	Vehicle state: 1 = ready 2 = EV is present 3 = charging 4 = charging with ventilation
1002	R	6 / 13 / 20 / 32 / 63 / 80 A Maximum current limitation according to a cable based on PP resistor detection
2000	R/W	Default amps value after boot (initial value 32A) - this gets saved to internal EEPROM

Register addresses are in decimal format!

NOTE: Only functions 03 (Read Holding Registers) and 16 (Preset Multiple Registers) are implemented. For more details please check: http://www.simplymodbus.ca/FAQ.htm

How to communicate with the board?

You can use any MODBUS master utility for Windows / Linux / Mac. For common users we recommend using QModMaster (http://sourceforge.net/projects/qmodmaster/).

For advanced users and further development you can use Python / pymodbus (https://github.com/bashwork/pymodbus) or any other library for your favorite programming language.