OpenEVSE

Electric Vehicle Charging

J1772 / OpenEVSE

Presented by:
Chris Howell

OpenEVSE

Topics:

- Electric Vehicle Supply Equipment
- J1772 Recommended Practice
- Challenges
- OpenEVSE
- Technology Development

OpenEVSE

Electric Vehicle Supply Equipment (EVSE)

The EVSE provides a safe connection from the Electrical source to the Plug in Vehicle.

The EVSE provides several safety features:

- Power pins not hot until EVSE-EV negotiation
- Ground Fault Circuit Interrupt (GFCI)
- Graceful start-up/shut-down
- Ground verification
- Pilot signal detection and verification
- Stuck Relay detection
- Plug rated for many plug-in /disconnect cycles

*Not All EVSE implement every feature

OpenEVSE

J1772 Overview

J1772 is a SAE ***Recommended Practice*** for a electric vehicle conductive charge system which covers:

- General physical
- Electrical
- Performance requirements

The intent is to define a common electric vehicle charging system architecture including operational requirements and the functional and dimensional requirements for the vehicle inlet and mating connector.

OpenEVSE

POWER INVERTER
AND TRANSFORMER
BATTERY CHARGER
CONNECTOR BOX
LEAF
LITHIUM-ION
BATTERY
ELECTRIC MOTOR
AND REDUCER

OpenEVSE

J1772 Properties

Charge Level	Voltage	Max Current
Level 1 (L1)	120VAC	16A - 1.9kw
Level 2 (L2)	208 - 240VAC	80A - 20kw
DC Level 1 (L3)	200 – 500V DC	80A – 40kW
DC Level 2 (L3)	200 – 500V DC	200A - 100kW

- Pilot Signal – 1khz pilot to communicate EVSE – EV state
- Duty Cycle – EVSE defines the maximum current available to the EV
- Proximity – Allows for graceful start-up and shutdown of current flow

OpenEVSE

Level 1 Charging

Charge Level	Voltage	Max Current
Level 1 (L1)	120VAC	16A - 1.9kw

- Adds < 5 Miles per every hour charging
- Best suited for Plug-in-Hybrid with low EV range
- Painfully slow for most BEVs
- Great in location where EVs park for several days at time and high density is desired such as Airport

OpenEVSE

Level 2 Charging

Charge Level	Voltage	Max Current
Level 2 (L2)	208 – 240VAC	80A - 20kw

- Adds up to 62 Miles range per hour of charge
- Rate Limited by on-board charger of vehicle
- Slightly more costly than L1
- Great in location where Plug-ins park. Home – Work – Malls - Attractions

Level 3 Charging

Charge Level	Voltage	Max Current
Level 3 (DC-FC)	300 – 460VDC	250A+

- Adds up to 300 Miles range per hour of charge
- Much more costly than L1/L2
- Several competing standards (CHAdeMO, J1772, Tesla)
- Requires 3 Phase AC infrastructure
- Great in location between cities, near the highway and where recharge speed is important

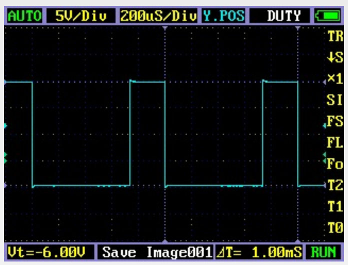
OpenEVSE

Charging Placement

- Charging Stations in prime locations tend to be “ICE”d, locate close to power but in less desirable parking locations
- Charging speed should match time at location, less time spent = quicker chargers. Fast food – DCQC… Airport long term L1
- Place EVSE between spaces so 1 EVSE can service 2 – 4 spaces each.
- Good Signage - Reserved for plug-in

OpenEVSE

J1772 Pilot Signal

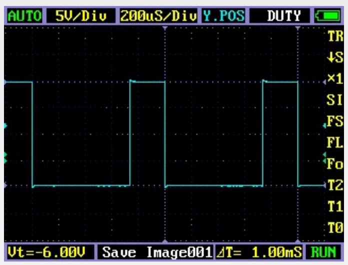


The J1772 Pilot is a 1khz +12V to -12V square wave, the voltage defines the state. The EV adds resistance pilot to Ground to vary the voltage. The EVSE reads the voltage and changes state accordingly.

State	Pilot High	Pilot Low	Frequency	EV Resistance	Description
State A	+12V	N/A	DC	N/A	Not Connected
State B	+9V	-12V	1000hz	2.74k	EV Connected (Ready)
State C	+6V	-12V	1000hz	882	EV Charge
State D	+3V	-12V	1000hz	246	EV Charge Vent. Required
State E	0V	0V	N/A		Error
State F	N/A	-12V	N/A		Unknown/Error

OpenEVSE

J1772 Duty Cycle



The J1772 Pilot is a 1khz +12V to -12V square wave, the Duty cycle (ratio high state to low state) determined the maximum available current. The EVSE sets the duty cycle the EV must comply to original setting or changes to the duty cycle.

6A - 51A

Amps = Duty cycle x 0.6

Duty cycle = Amps / 0.6

51A - 80A

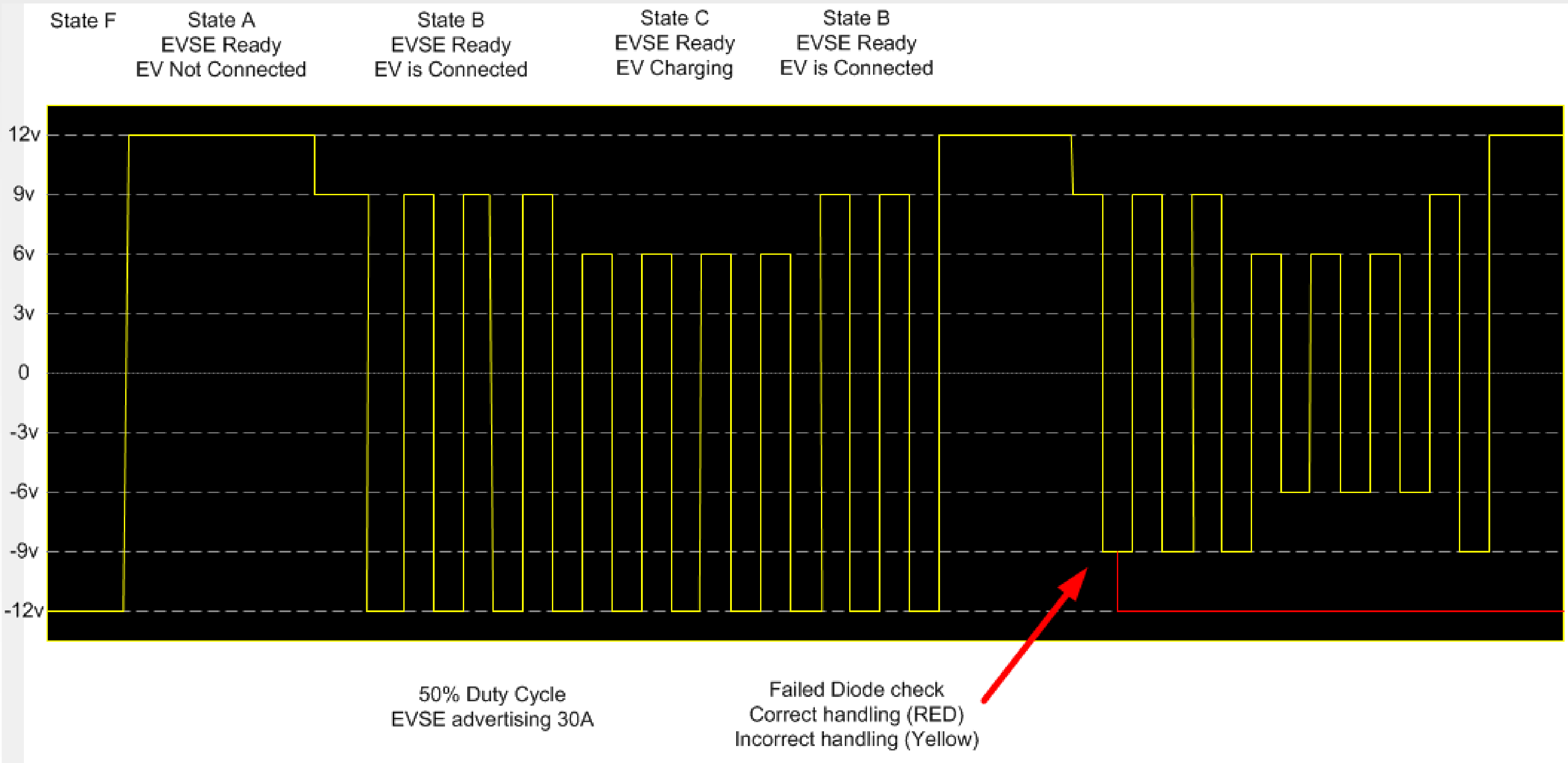
Amps = (Duty Cycle - 64) 2.5

Duty cycle = (Amps / 2.5) + 64

Amp	Duty Cycle	Amp	Duty Cycle
6A	10%	40A	66%
12A	20%	48A	80%
18A	30%	65A	90%
24A	40%	75A	94%
30A	50%	80A	96%

OpenEVSE

J1772 Negotiation



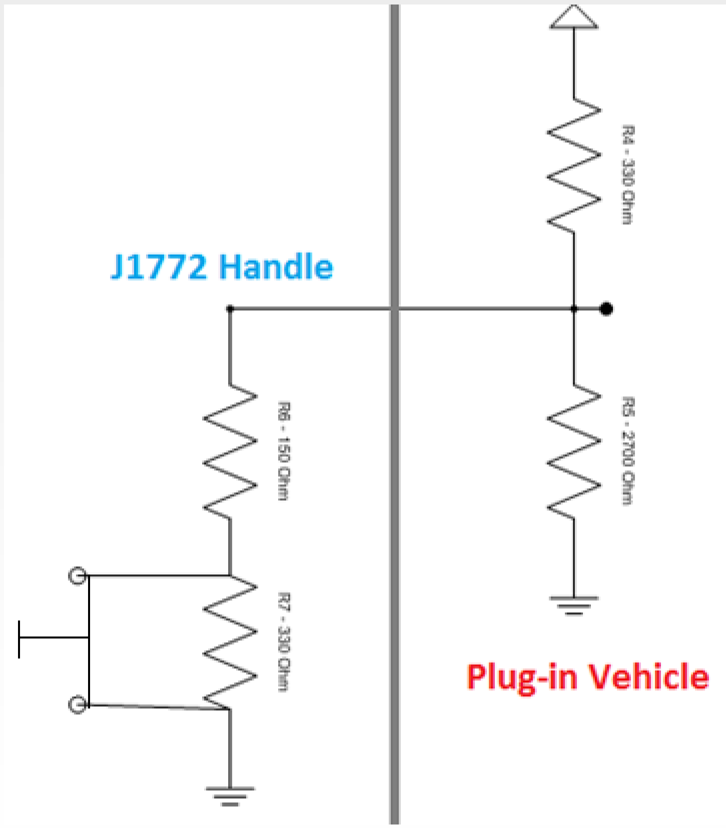
OpenEVSE

J1772 Proximity

The J1772 Proximity circuit is present in the Electric Vehicle and the J1772 plug. It uses a voltage divider circuit with resistors in Parallel and series to achieve different measured voltages for each state.

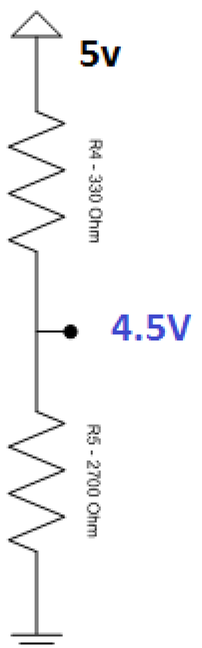
State	Voltage on Proximity pin
Not Connected	4.5v
Button Pressed	3.0v
Connected	1.5v

	Resistance
R4	330
R5	2700
R6	150
R7	330



J1772 Proximity

State	Voltage on Proximity pin
Not Connected	4.5v



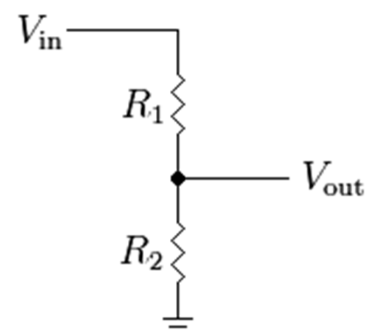
$$V_{\text{out}} = \frac{2700}{330 + 2700} \cdot 5$$

$$V_{\text{out}} = .9 \cdot 5$$

$$V_{\text{out}} = 4.5$$

Voltage Divider

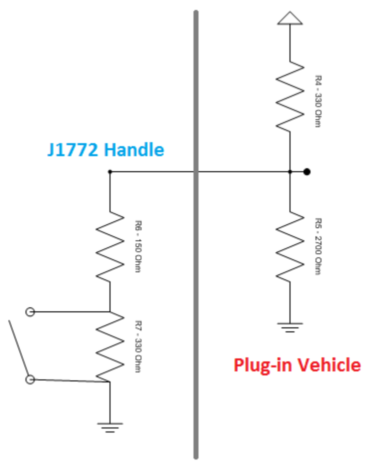
$$V_{\text{out}} = \frac{R_2}{R_1 + R_2} \cdot V_{\text{in}}$$

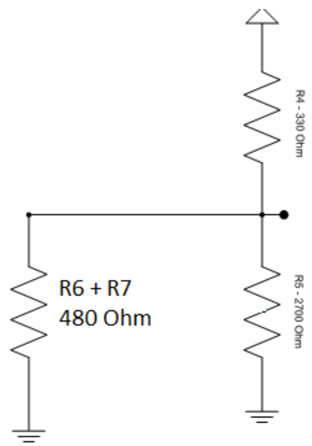


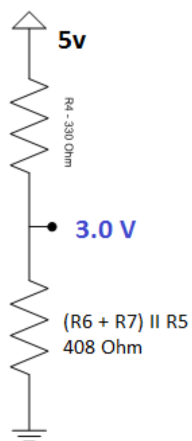
	Resistance
R4	330
R5	2700
R6	150
R7	330

OpenEVSE

J1772 Proximity







$$R_{\text{total}} = 150 + 330$$

$$R_{\text{total}} = 480$$

$$R_{\text{total}} = \frac{(480)2700}{480 + 2700}$$

$$R_{\text{total}} = \frac{1296000}{3180}$$

$$R_{\text{total}} = 408$$

$$V_{\text{out}} = \frac{408}{330 + 408} \cdot 5$$

$$V_{\text{out}} = .6 \cdot 5$$

$$V_{\text{out}} = 3.0$$

State	Voltage on Proximity pin
Button Pressed	3.0v

Resistance Series

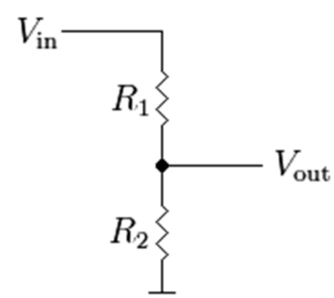
$$R_{\text{total}} = R_1 + R_2$$

Resistance Parallel

$$R_{\text{total}} = \frac{R_1 R_2}{R_1 + R_2}.$$

Voltage Divider

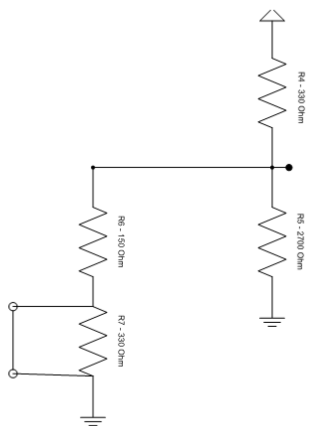
$$V_{\text{out}} = \frac{R_2}{R_1 + R_2} \cdot V_{\text{in}}$$

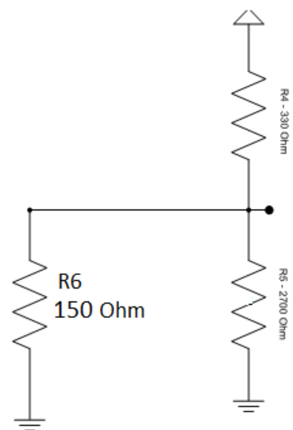


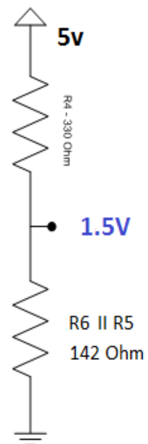
	Resistance
R4	330
R5	2700
R6	150
R7	330

OpenEVSE

J1772 Proximity







State	Voltage on Proximity pin
Connected	1.5v

Resistance Series

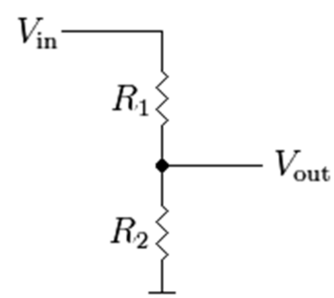
$$R_{\text{total}} = R_1 + R_2$$

Resistance Parallel

$$R_{\text{total}} = \frac{R_1 R_2}{R_1 + R_2}.$$

Voltage Divider

$$V_{\text{out}} = \frac{R_2}{R_1 + R_2} \cdot V_{\text{in}}$$



$$R_{\text{total}} = \frac{(150)2700}{150+2700}$$

$$R_{\text{total}} = \frac{405000}{2850}$$

$$R_{\text{total}} = 142$$

$$V_{\text{out}} = \frac{142}{330+142} \cdot 5$$

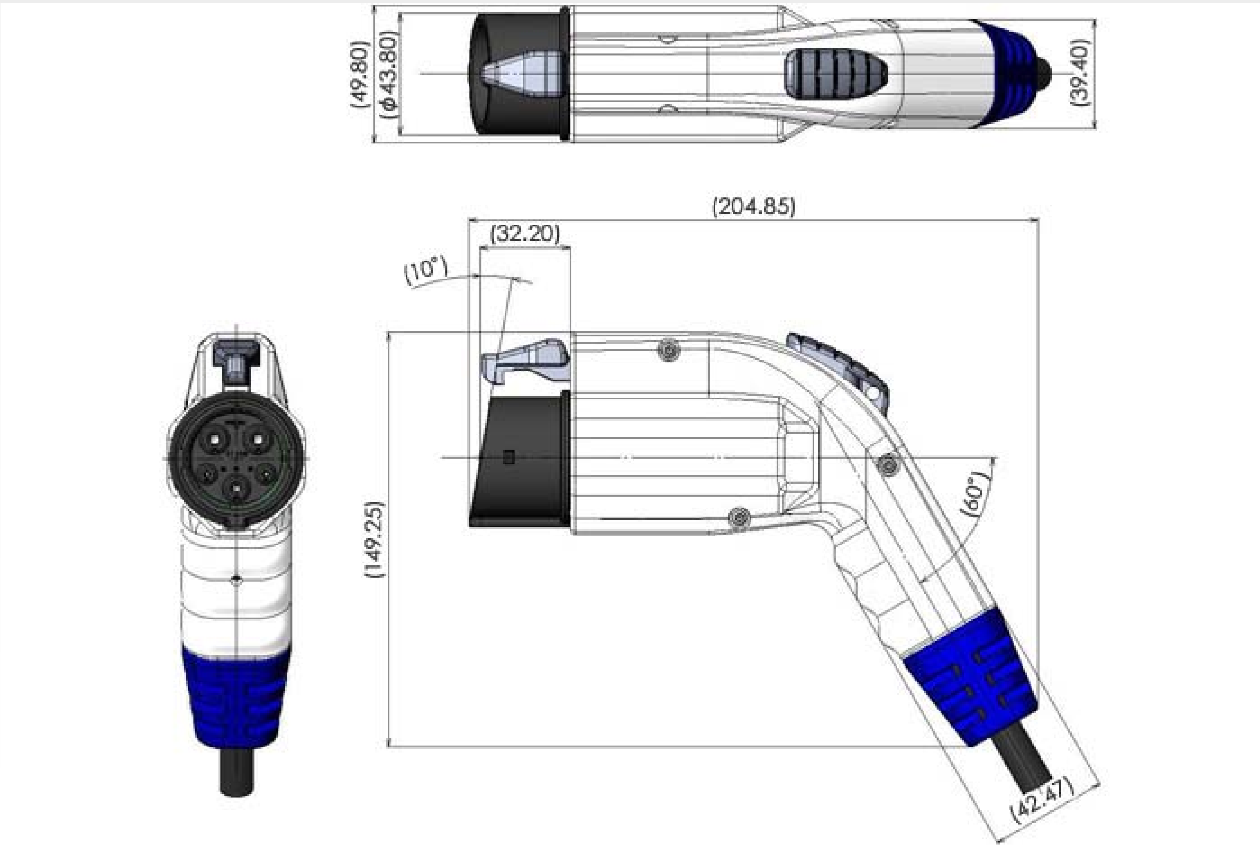
$$V_{\text{out}} = .3 \cdot 5$$

$$V_{\text{out}} = 1.5$$

	Resistance
R4	330
R5	2700
R6	150
R7	330

OpenEVSE

J1772 Plug



SAE Charging Configurations and Ratings Terminology

SAE International

AC level 1
(SAE J1772™)

PEV includes on-board charger
120V, 1.4 kW @ 12 amp
120V, 1.9 kW @ 16 amp
Est. charge time:
PHEV: 7hrs (SOC– 0% to full)
BEV: 17hrs (SOC – 20% to full)

AC level 2
(SAE J1772™)

PEV includes on-board charger (see below for different types)
240 V, up to 19.2 kW (80 A)
Est. charge time for 3.3 kW on-board charger
PEV: 3 hrs (SOC - 0% to full)
BEV: 7 hrs (SOC – 20% to full)
Est. charge time for 7 kW on-board charger
PEV: 1.5 hrs (SOC - 0% to full)
BEV: 3.5 hrs (SOC – 20% to full)
Est. charge time for 20 kW on-board charger
PEV: 22 min. (SOC - 0% to full)
BEV: 1.2 hrs (SOC – 20% to full)

DC Level 1
(SAE J1772™)

EVSE includes an off-board charger
200-500 V DC, up to 40 kW (80 A)
Est. charge time (20 kW off-board charger):
PHEV: 22 min. (SOC - 0% to 80%)
BEV: 1.2 hrs. (SOC – 20% to 100%)

DC Level 2
(SAE J1772™)

EVSE includes an off-board charger
200-500 V DC, up to 100 kW (200A)
Est. charge time (45 kW off-board charger):
PHEV: 10 min. (SOC - 0% to 80%)
BEV: 20 min. (SOC – 20% to 80%)

Voltages are nominal configuration voltages, not coupler ratings.
Rated power is at nominal configuration operating voltage and coupler rated current.
Ideal charge times assume 90% efficient chargers, 150W to 12V loads and no balancing of Traction Battery Pack.

Notes:
1) BEV (25 kWh usable pack size) charging always starts at 20% SOC, faster than a 1C rate (total capacity charged in one hour) will also stop at 80% SOC instead of 100%
2) PHEV can start from 0% SOC since the hybrid mode is available.

ver. 100312

Challenges

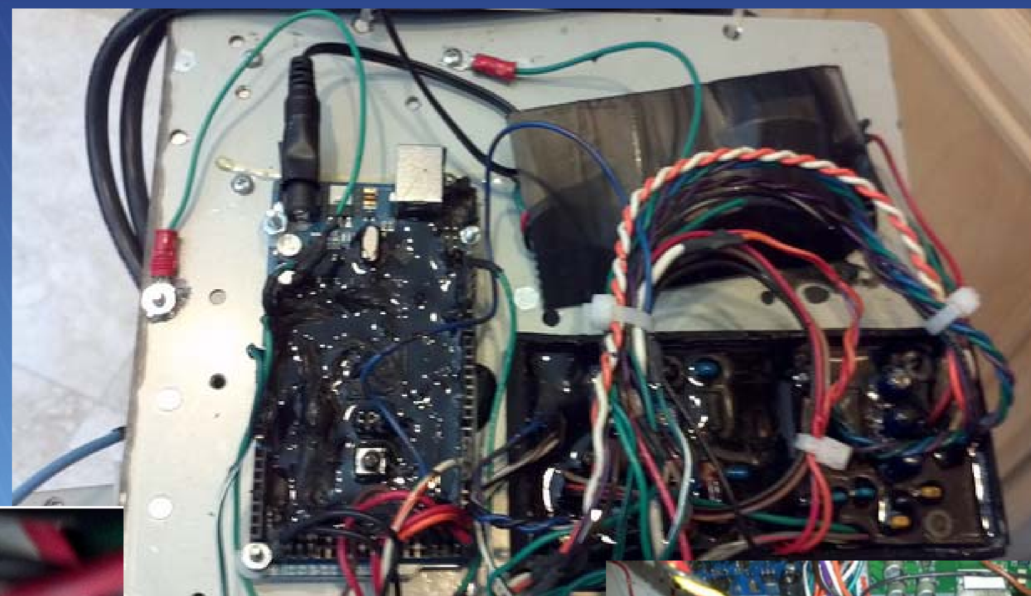
- Incompatibility of devices
- Missing safety features (Diode check, Vent required state)
- Poor quality of devices
- Overheating at or below rated current
- Cost of deployment
- Devices to bypass/circumvent/ignore J1772 NEC requirements

OpenEVSE

Home Built or Commercial Product???

Hints:

- Device has cord but no fuses
- Device uses wrong type of relays (SSR not Mechanical)
- Poor construction
- Metal Shavings



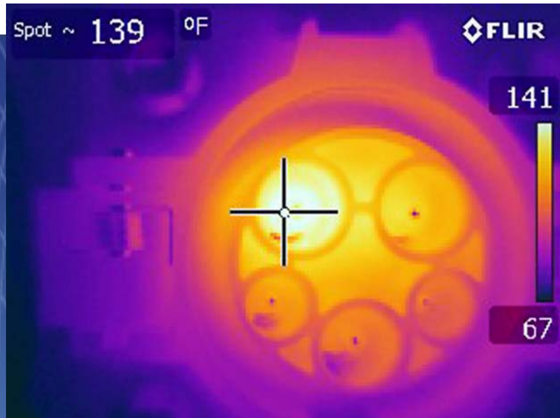
Almost touching….

OpenEVSE

Home Built or Commercial Product???

Hints:
- Open Source LINUX board
- Thermal issues at/below rated power
- Off the shelf power meter inside
- High percentage out of order, on the blink.
- Improper crimp on Power Connector

OpenEVSE



Newer EVs capable of drawing higher current are causing problems for even UL listed commercial EVSEs running at or below their rated limit.

- Honda first to implement cutoff in the inlet

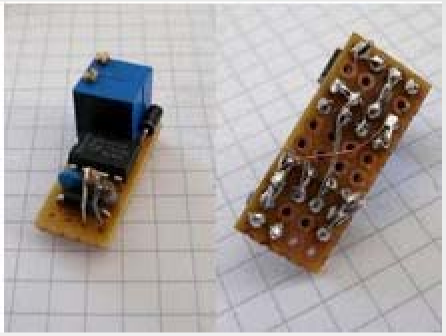
OpenEVSE

Don't try this at home…

Ignoring J1772, NEC, Local code etc. can be hazardous to people and property

The pictured solutions are work around to bypass/trick J1772 protections

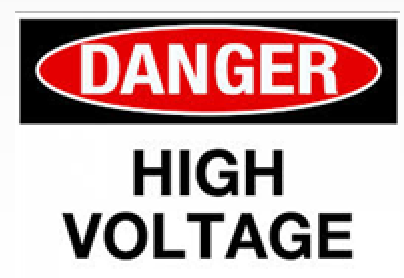
- No relay to remove power from connecter
- No GFCI protection
- Must be connected in certain order with quick timing
- Causes vehicle error codes
- Could cause damage to charging system



OpenEVSE

Warnings

OpenEVSE hardware/firmware is intended for use for ENGINEERING DEVELOPMENT, DEMONSTRATION, OR EVALUATION PURPOSES ONLY and is NOT considered to be a finished end-product fit for general consumer use.



OpenEVSE

“Open Source”

- Source materials (source code, schematics, recipes, documents) are published and made available to the public
- Enables anyone to copy, modify and redistribute without paying royalties or fees.
- Open-source code can evolves through community cooperation

open source

open hardware

OpenEVSE

GNU GPL v3

OpenEVSE Source code is licensed open source under GNU GPLv3

Nobody should be restricted by the software they use. There are four freedoms that every user should have:

- the freedom to use the software for any purpose,
- the freedom to change the software to suit your needs,
- the freedom to share the software with your friends and neighbors, and
- the freedom to share the changes you make.

http://www.gnu.org/licenses/gpl.html

OpenEVSE

Creative Commons Attribution-ShareAlike 3.0

All other OpenEVSE content licensed under CC BY SA 3.0

You are free:

to Share — to copy, distribute and transmit the work

to Remix — to adapt the work

to make commercial use of the work

Under the following conditions:

Attribution — You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).

Share Alike — If you alter, transform, or build upon this work, you may distribute the resulting work only under the same or similar license to this one.

OpenEVSE

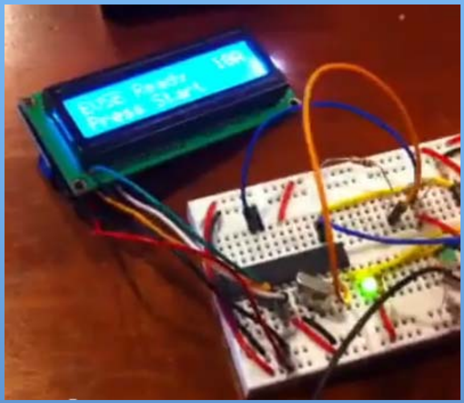
OpenEVSE

OpenEVSE is a Open Source Electric Vehicle J1772 Charging Station Controller

- Both Hardware and Firmware Open Source
- Fully supports SAE J1772 Recommended Practice
- Software adjustable pilot (6A – 80A)
- Built in GFCI with 20ma trip point
- Supports all J1772 states including “ventilation required”
- Supports Diode check
- AC L1 – L2 auto detect Current setting for each
- Ground verification and Stuck Relay detection

OpenEVSE

February 13, 2011 - Experiments with pilot began
June 15, 2011 – Nissan LEAF Delivered
July 1, 2011 – Successfully Charged LEAF
July 2011 – Joined forces with Lincomatic
October 2011 – Started OpenEVSE open sourced hardware and firmware
December 2011 – First prototype OpenEVSE boards available



OpenEVSE

OpenEVSE Plus

OpenEVSE Plus

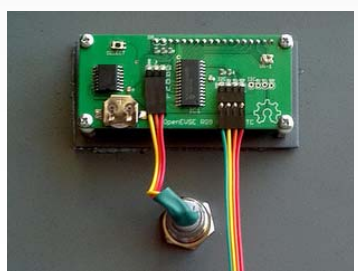
- Board and Schematic Files Available
- Available in limited quantities as Kit or Built
- All surface mount component work complete
- Firmware pre-loaded
- Tiny 2.2 x 1.75
- Power Supply Integrated
- Inexpensive - $135 (kit) $155 (Built)

OpenEVSE

OpenEVSE LCD

OpenEVSE RGB LCD

- Board and Schematic Files Available
- Basic or with Real Time Clock (RTC)
- RTC adds EVSE based timer support
- Optional button adds LCD Menu Interface
- Available in limited quantities as Kit or Built
- All surface mount component work complete
- Basic $30 (kit) $40 (Built)
- RTC $40 (kit) $50 (Built)

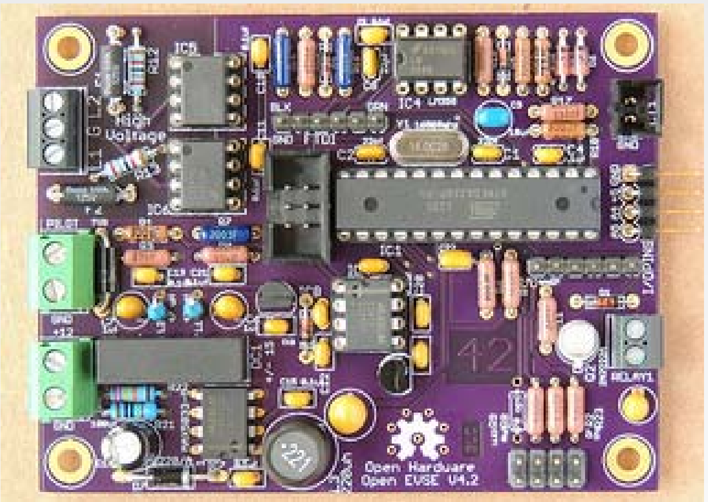


OpenEVSE

OpenEVSE DIY

OpenEVSE DIY boards

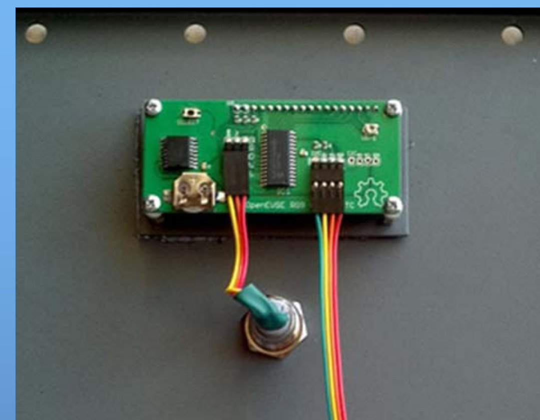
- Board and Schematic Files Available
- Board available in OSHpark Store
- Source your own components
- Build yourself
- 3.4 x 2.5
- Inexpensive - Board and PS ~ $100

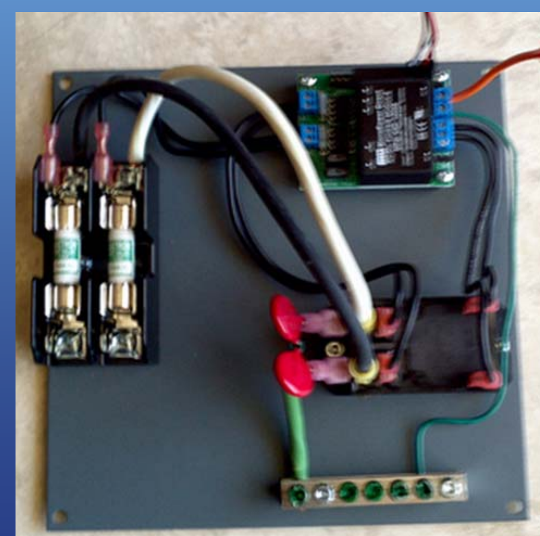


OpenEVSE

Example EVSE built with OpenEVSE

- Diversified Stage Enclosure
- 30A ITT/Leviton J1772 Cable
- OpenEVSE Plus
- OpenEVSE RGB LCD with RTC



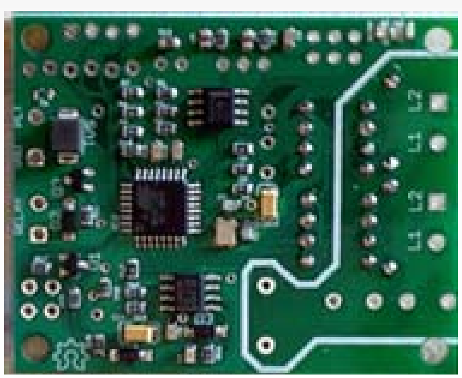


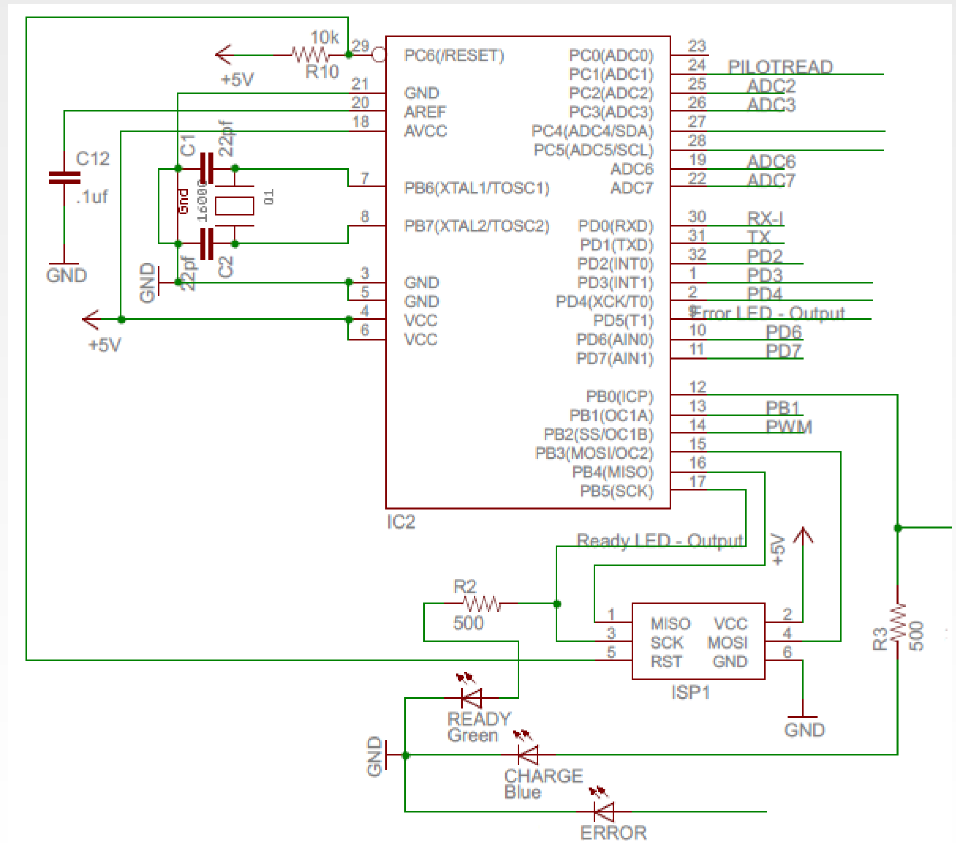
OpenEVSE

OpenEVSE CPU

OpenEVSE is based on the ATMEL AVR:

- 8-bit microprocessor
- 16mhz
- Compatible with Arduino IDE

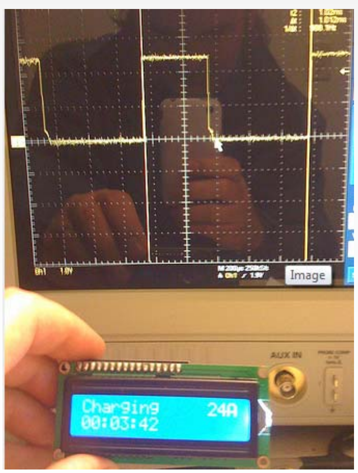


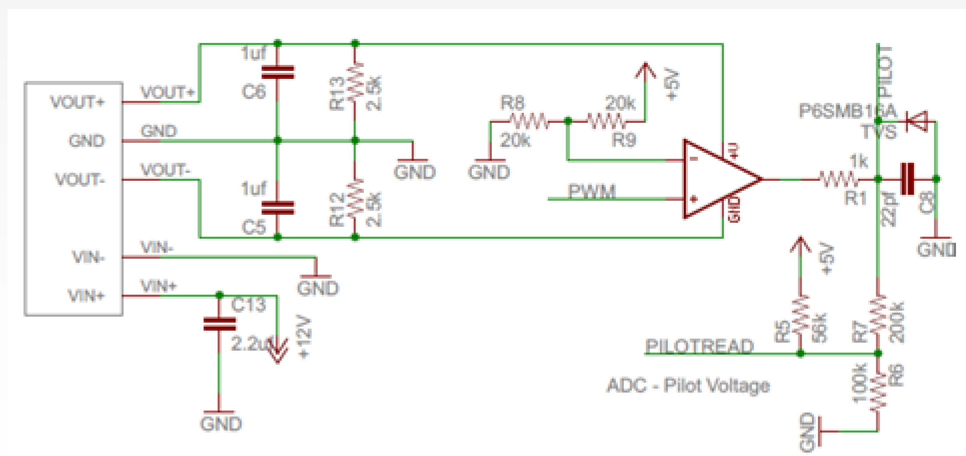


OpenEVSE

OpenEVSE Pilot

- The OpenEVSE pilot uses a 1w DC/DC converter to generate +12v and -12v.
- The Opamp takes the 1khz pilot from the microprocessor 0 – 5v and switched -12v to +12v.
- The pilot is read by the microprocessor, R5 – R6 – R7 scale the -12v - +12v signal to 0 – 5v.

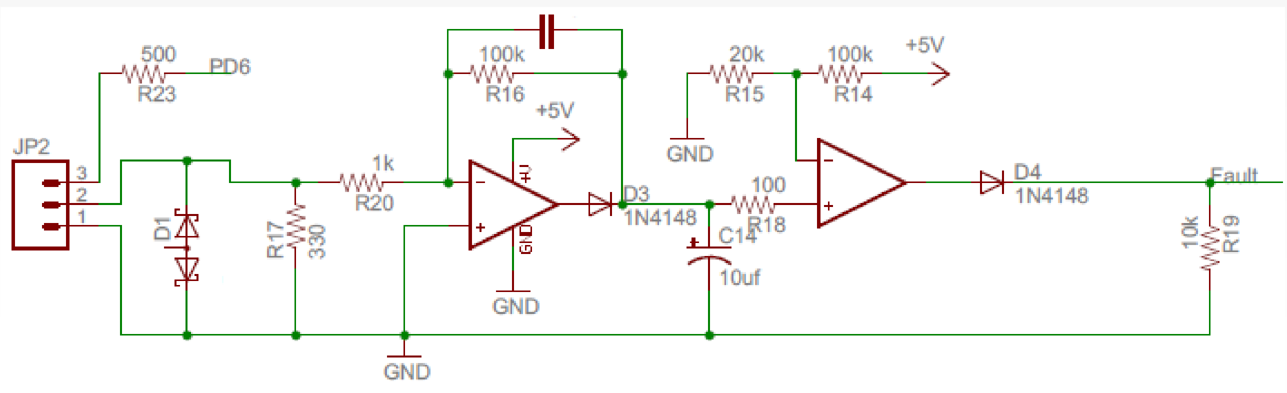




OpenEVSE

OpenEVSE GFCI

GFCI measures the difference of current going in vs. current going out. The circuit “trips” if an imbalance of > 20ma. The trip point can be adjusted by modifying the burden resistor R17 or the ratio of R14 – R15. The output of the fault line is monitored by the microprocessor as an interrupt.

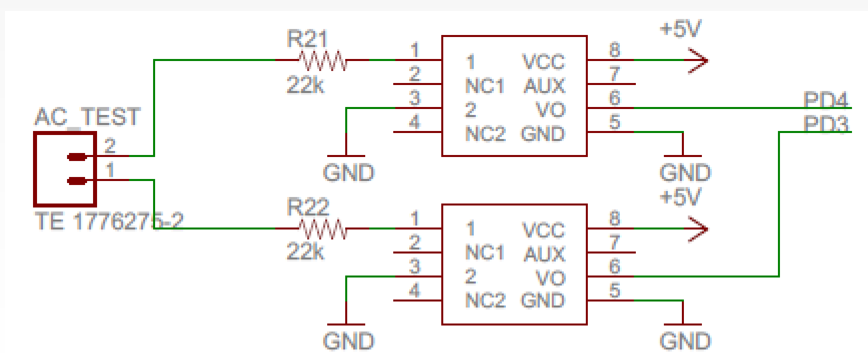


OpenEVSE

OpenEVSE Power

OpenEVSE uses 2 MID400 Optical Isolators to detect the presence of voltage on each Hot line by sending a small current to ground. The AC_Test leads are connected after the power relay to allow stuck relay detection as well as Ground Verification and L1/L2 auto-detection (1/3 Phase detection in Europe).

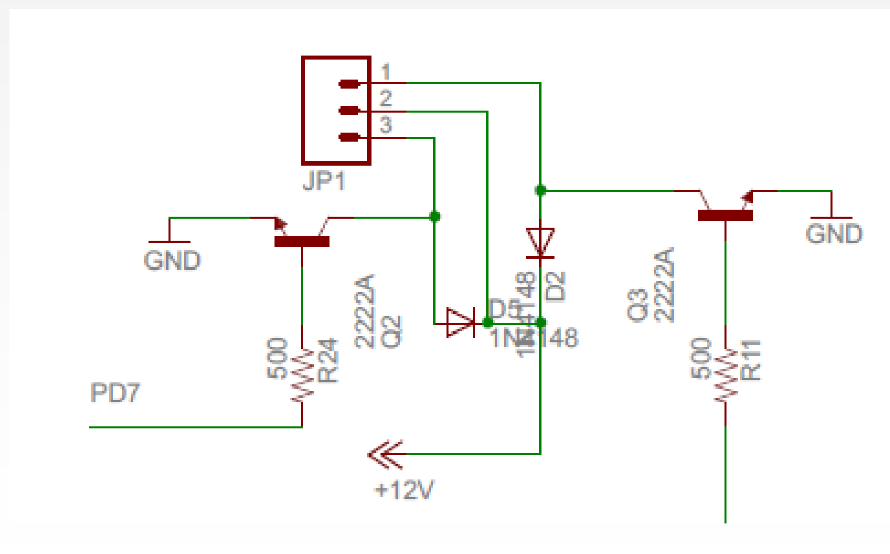
1	2	
L	L	L2
L	H	L1
H	L	L1
H	H	Bad Ground



OpenEVSE

OpenEVSE Relay

The relay circuit uses 2 2222 NPN transistors to switch 12V to the relay(s). Beginning in 2.0B2 OpenEVSE supports both 1 DPST or 2 SPST. Using 2 relays allows the self tests to be run one leg at a time avoiding powering the J1772 handle during the test. Also power can be removed from 1 leg if there is a stuck relay condition.



OpenEVSE

Coming Soon (hopefully) / Areas you can contribute

- Simple Communications Protocol
- Android App
- Raspberry Pi integration
- Wi-Fi support
- Vehicle info to EVSE - CAN / WiFi
- LCD touch screen
- Energy Monitoring
- General code clean up
- RTC / Timer code library

OpenEVSE

Simple Communications protocol

- Work in conjunction with existing Command Line Interface
- Work in progress / High Priority
  https://docs.google.com/document/d/1e00CnEpSUb6BpQho9srvDj8HuKcuVMxiXaSICXDvHeA/edit?usp=sharing

- EVSE Status and Control
- Energy Monitoring input
- Data from EV via CAN or WiFi (TESLA REST API)
- Needed for Android App, LCD touchscreen / Raspberry pi
- UART / I2C - Wifi / Bluetooth

OpenEVSE

L1 and L2 EVSEs are currently dumb devices

- No - 2 way data communication between EV and EVSE
- Data can be provided to EVSE via
  - CAN bus
  - Wifi/3G/4G
  - Bluetooth
- EV – EVSE communication allows EVSE to know and act on:
  - State of Charge
  - Battery Voltage
  - Current

OpenEVSE

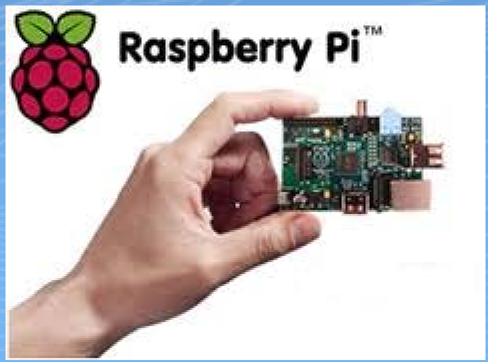
Android App

- Tablet/Phone
- Touchscreen for EVSE
- Fusion EVSE and EV info/control
- Serial SPP or WiFi

OpenEVSE

Raspberry Pi

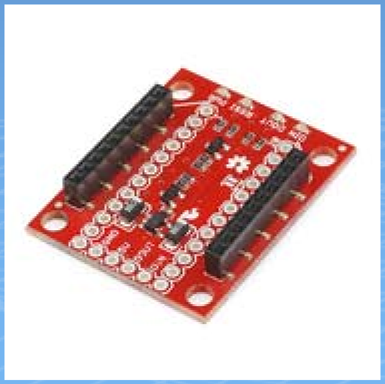
- Inexpensive method to add Ethernet / Wi-Fi etc.
- Web front end to EVSE
- Fusion EVSE and EV info/control
- TTL Serial or I2C
- Headless or LCD Touchscreen

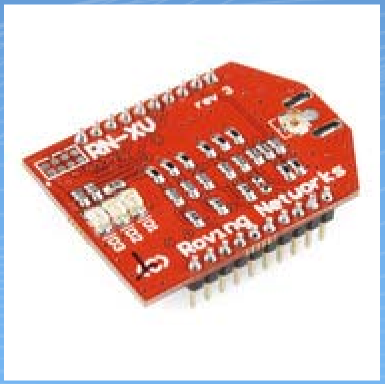


OpenEVSE

WiFi

- Telnet/SSH access to SerialCLI (Done)
- Web client – provide input to webserver
- Fusion EVSE and EV info/control
- TTL Serial
- Con – twice as expensive as adding Raspberry pi



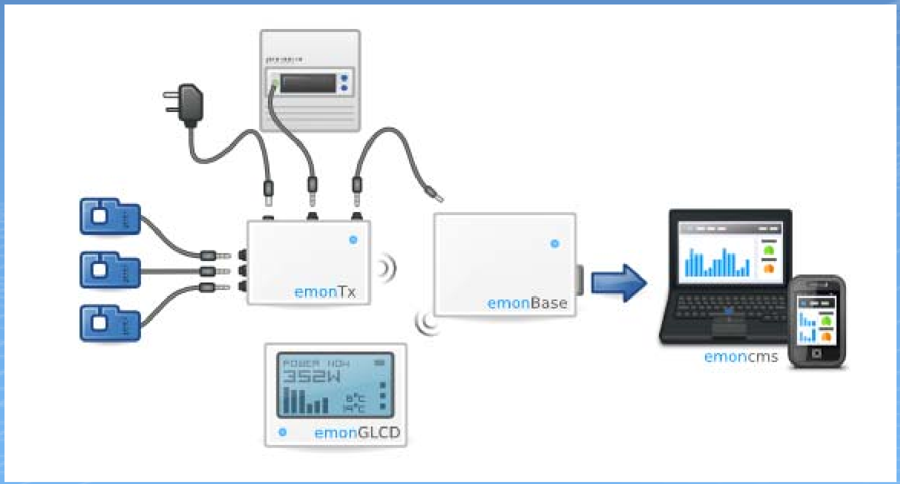


OpenEVSE

Simple LCD touchscreen
- Arduino Touch Screens
- 4D Systems
- Fusion EVSE and EV info/control
- TTL Serial or I2C

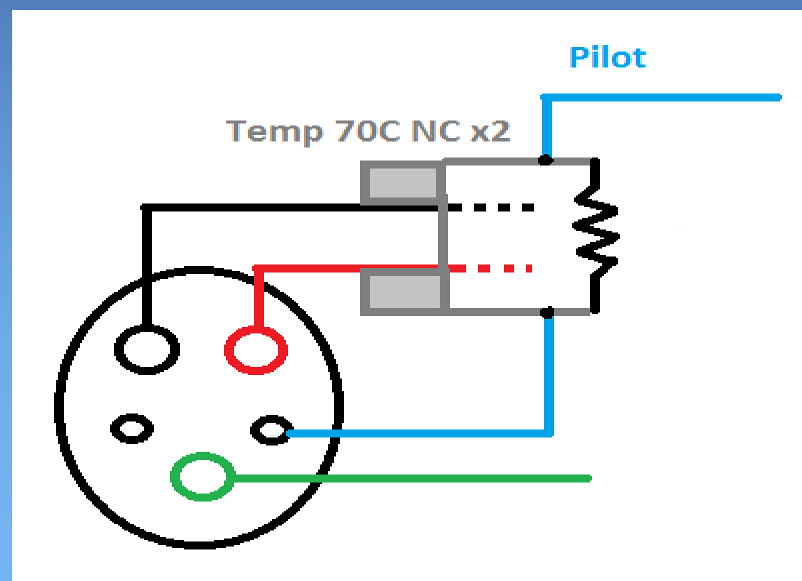
Energy Monitoring

- Based on Open Energy Monitor
- Reads both Voltage and Current
- May not be necessary if data can be obtained from EV
- TTL Serial or I2C



OpenEVSE

Modifications to J1772 inlet and/or handle can provide information to the EV/EVSE to act on over temp conditions and reduce current or terminate charge.

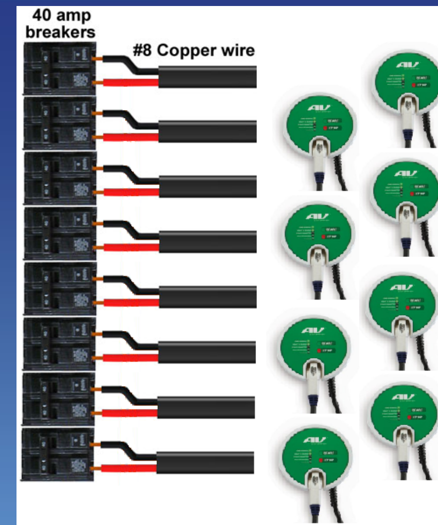


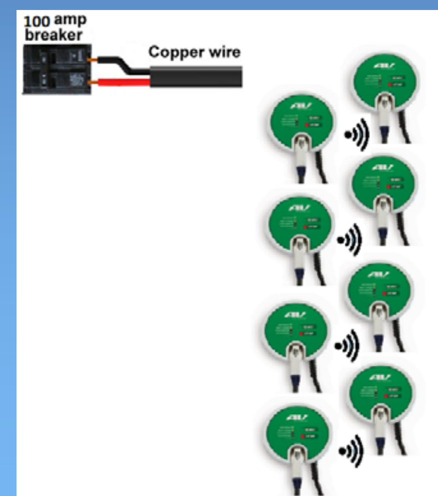
OpenEVSE

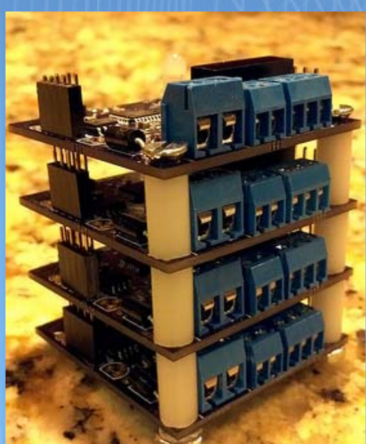
Infrastructure is expensive, Smart EVSEs could share a circuit and share load.

- J1772 allows for dynamic current settings
- Vehicles with low current requirement of those finished can give capacity to those who need it

Example: 8 - 30A EVSE







1	30A	5	16A
2	30A	6	13A
3	26A	7	11A
4	20A	8	10A

OpenEVSE

Resources:

Main Project Page:

http://www.openevse.com

http://code.google.com/p/open-evse/

Discussion

https://groups.google.com/forum/?fromgroups#!forum/OpenEVSE

http://www.mynissanleaf.com/viewtopic.php?f=26&t=6546

Development Code

https://github.com/lincomatic/open_evse

Questions??? / Demos…