Development of a Connected Electric Vehicle Charger

Andy Begey, Brandon Metcalf, Shermaine Dayot, Luis Contreras, Paolo Caraos
Project Mentors: Dr. Michael Klopfer, Prof. G.P. Li

California Plug Load Research Center (CalPlug)
Department of Electrical Engineering and Computer Science

Funded by
UROP UCI

In collaboration with Smartenit

Introduction:

As more and more electric vehicles enter the nation's roadways, the problem of efficiently scheduling their charging times and monitoring power consumption will become increasingly important. Electric vehicle (EV) owners will require an intuitive and unobtrusive way to schedule vehicle charging during the most cost-effective times of the day and to manage multiple vehicle charging. Charging is provided by a device referred to as an Electric Vehical Supply Equipment (EVSE). Most chargers on the market provide limited energy control or connectivity. With the proliferation of Smart-Home devices and the Internet of Things (IoT), we have developed a connected, future ready set of EVSE devices that can allow integrated and corrdinated vehiclecharging.

Goal:

The aim of this project is to design two separate IoT enabled EVSE controllers with the ability to schedule charging times and monitor power consumption of electric vehicles wirelessly. **Design 1** is the world's first Microsemi FPGA based controller / Arduino Due compatible controller. **Design 2** is an Espressif ESP32 based WiFi enabled device which uses MQTT based control and is compatable with the Smartenit smart device control API. This system will enable electric vehicle owners to manually regulate the charging and power monitoring of their vehicles via a mobile application connected to a smart home IoT network. This system also provides the capability of automated charge rate control considering factors such as time of day, energy cost, net zero onsite generation, user behavior, renewable balance in generation based on time of the day, etc.

Design Approach

Using EAGLE PCB design software, the team extended elements of the OpenEVSE project to enable compatibility with the Microsemi SmartFusion2 FPGA as well as Arduino Due. The new team's design included an Analog Devices ADE7953 wattmeter for power measurement and Espressif ESP8266 for WiFi communication capability. Additionally, a second PCB was developed in collaboration with Orange County tech company, Smartenit and is designed to connect to their smart device management network.

Our controller is designed to work with the SAE J1772 EV charging protocol, the standard charging method for electric vehicles in North America. Wireless communication with the device is facilitated through Message Queuing Telemetry Transport (MQTT), a lightweight messaging protocol that will allow 2-way communication between our controller and the internet. Using MQTT we can wirelessly send the device instructions on when to charge the vehicle and with how much current. We can also transmit the power consumption data collected by the Analog Devices ADE7953 wattmeter on the board.

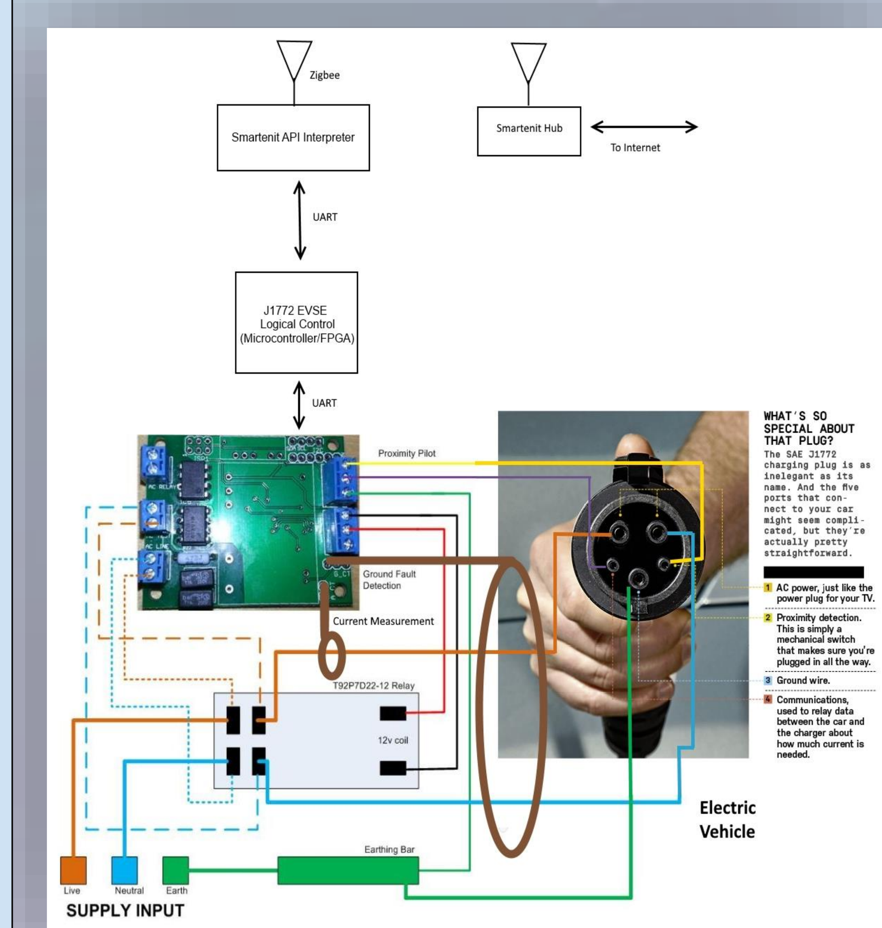
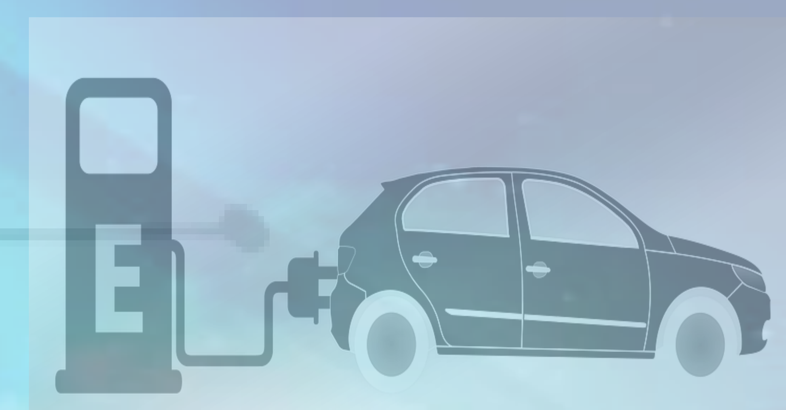


Figure 1:
Conceptual EVSE charging system



Design 1

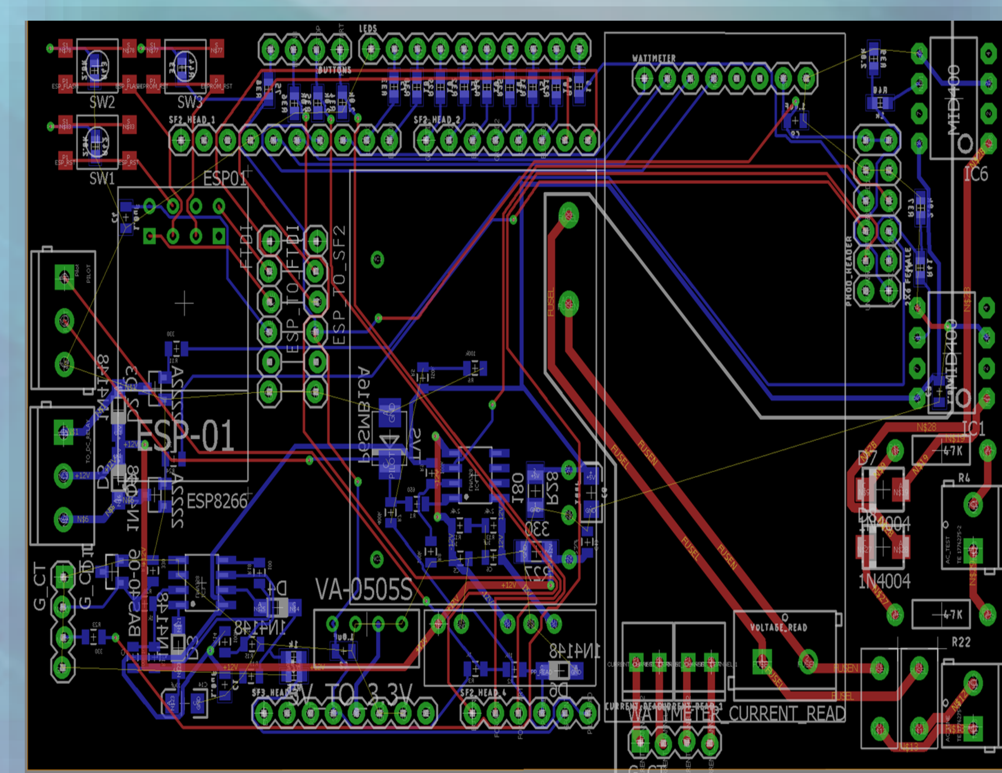


Figure 2:
PCB layout of the design 1 FPGA/Arduino interface in EAGLE

Design 2

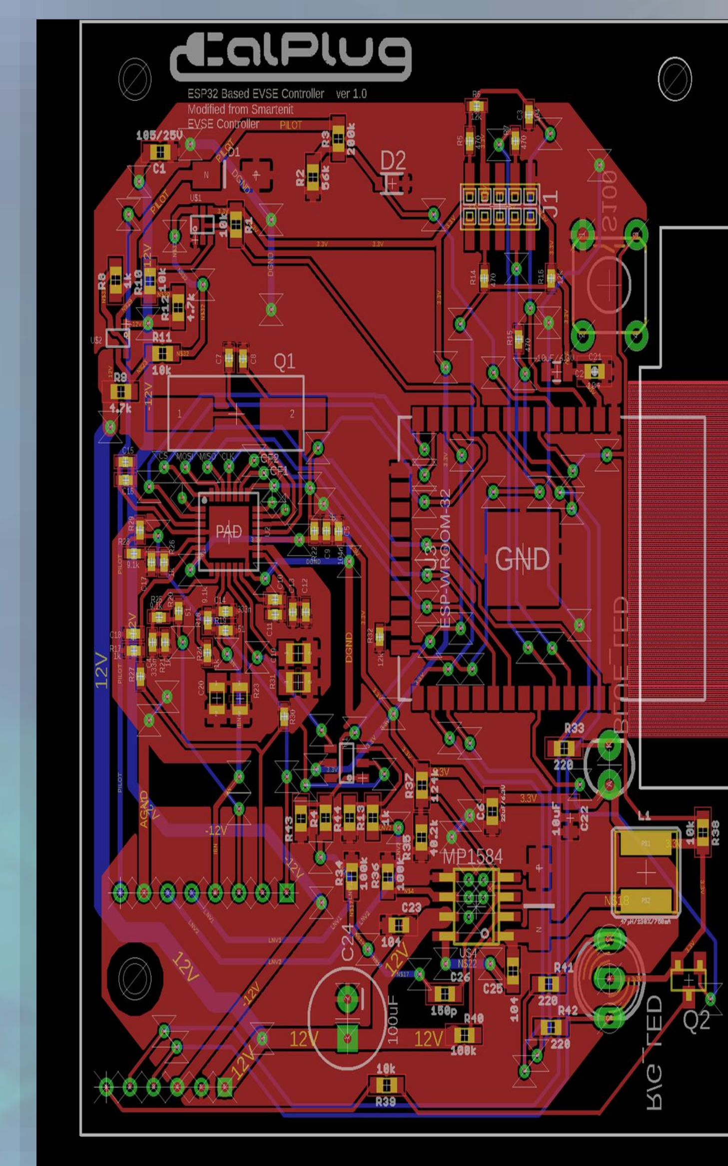


Figure 3:
PCB layout of the design 2 Smartenit WiFi-enabled EVSE controller in Autodesk EAGLE PCB design software

Design 1:

The team built design 1 to interface with an Arduino Due as well as a Microsemi SmartFusion2 FPGA demo board made by Future Electronics. This PCB features analog control circuitry for:

1. **Ground fault detection**
2. **Pilot signal generation**
3. **EV Level 1 / Level 2 charging detection**

The board also features terminals for connection to Calplug's isolated ADE7953 wattmeter board for voltage and current measurement.

The board uses an Espressif ESP8266 communication module For WiFi connectivity. Figure 4 shows a fully populated, tested, and working board from the initial Eagle design in Figure 2.

Figure 4:
Fully assembled and tested design 1 board

Design 2:

The team built design 2 to meet Smartenit's specification in order to use their general device casing and power board for their Zigbee enabled controller board. The team's design features an ESP32 and an ADE7953 chip integrated solution.

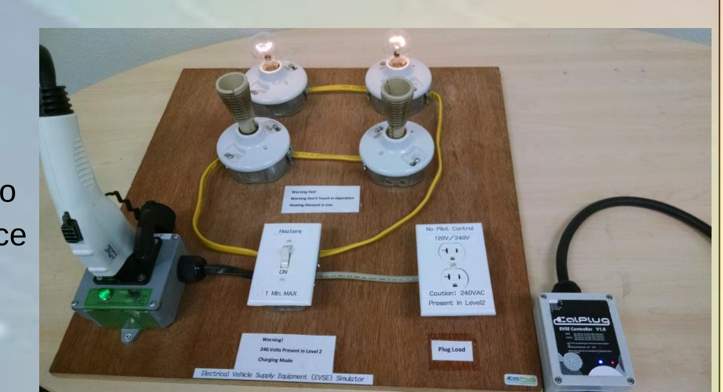


Figure 5:
Simulated Vehicle and EVSE

The team's design features an ESP32 and an ADE7953 chip integrated into the board for level detection, ground fault detection, and power readings.

Pilot control is also featured through analog circuitry in this board; however, relay signals are amplified off-board on the power daughter board. Additionally, this power board provides the AC circuitry that is input to our control PCB to determine power usage by the electric vehicle charger. Initial testing was performed using a simulated vehicle that used a set of loads (heaters) and a TRIAC which modulated the drawn load relative to the J1772 pilot indicated supply potential.

Accomplishments:

Both Design 1 and Design 2 have been fully built, tested, and verified as functional for the purposes of this project. This means that both devices are capable of controlling when an EV will charge and the rate it will charge at whilst checking for ground faults, stuck relays, publishing all data to an MQTT server, and being able to take data from said server as well.. Since Design 2 featured a fully operation device, it was tested much more thoroughly than design 1. Design 2 was tested for charging at level 1 on a 2012 Chevrolet Volt EV (Figure 6). The test successfully charged the vehicle, whilst displaying the correct power Readings for delivered current.

For MQTT connectivity, the device continuously publishes and subscribes to data cached on an MQTT broker. The protocol for interfacing and receiving signals is compliant with Smartenit's device control API.

A controller was developed for testing design 2. The Processing 3 based user interface for this controller is shown in Figure 7 below. The controller effectively operates as the typical interface a smart-home owner may use for controlling this device in the context of a larger device network at home. It can request the charging rate and power supplied to the vehicle, as well as set the charge rate of the car and show active feedback of the controller's charging status. All code in this controller has also maintained Smartenit API compatibility.

Figure 6:
Successful test charging a 2012 Chevrolet Volt EV using the design 2 device

This project has concluded with the successful construction of the world's first FPGA based EVSE and Arduino shield . The project also created a fully communicating WiFi based EVSE compatable with the Smartenit device control API.

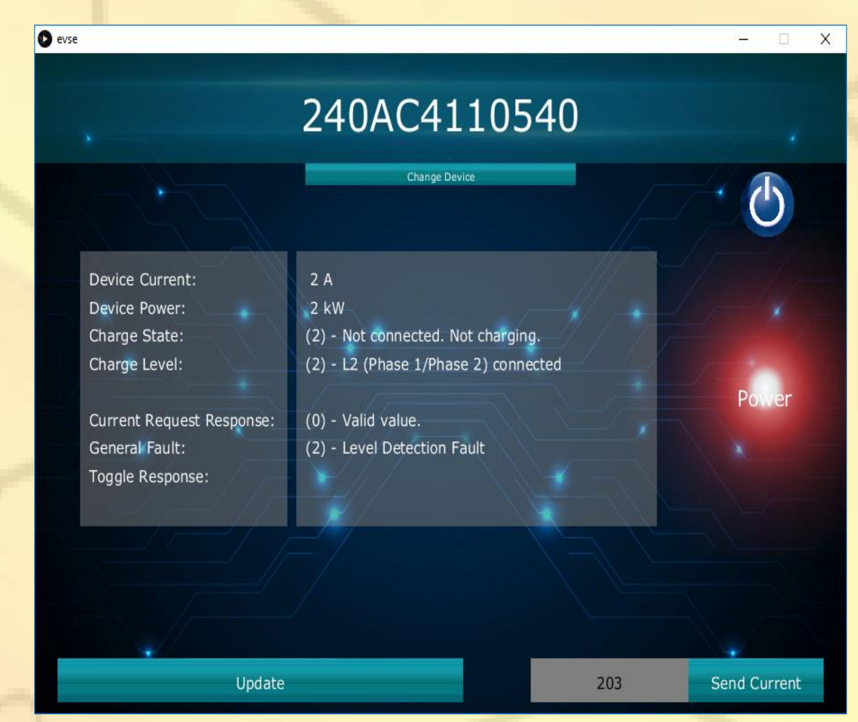


Figure 7: EVSE MQTT control app for design 2