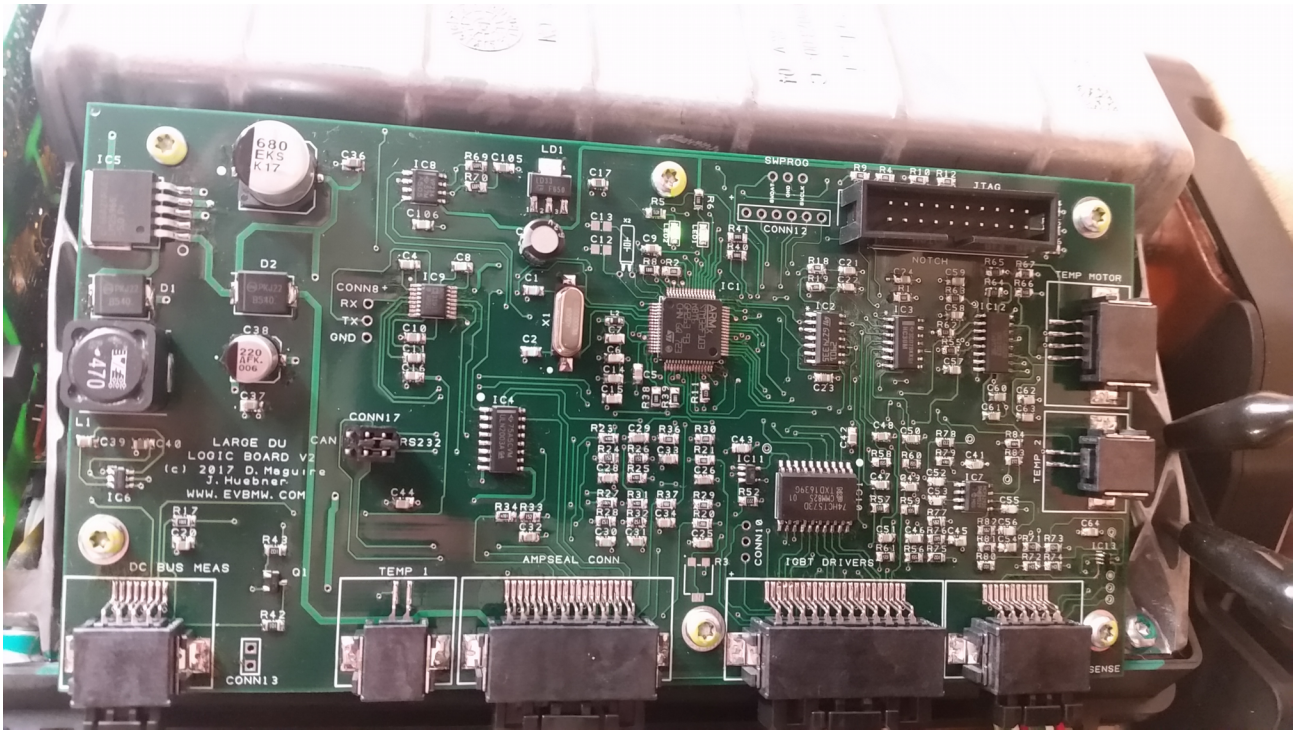
OPEN SOURCE TESLA LARGE DRIVE UNIT LOGIC BOARD

QUICK START GUIDE rev0.1

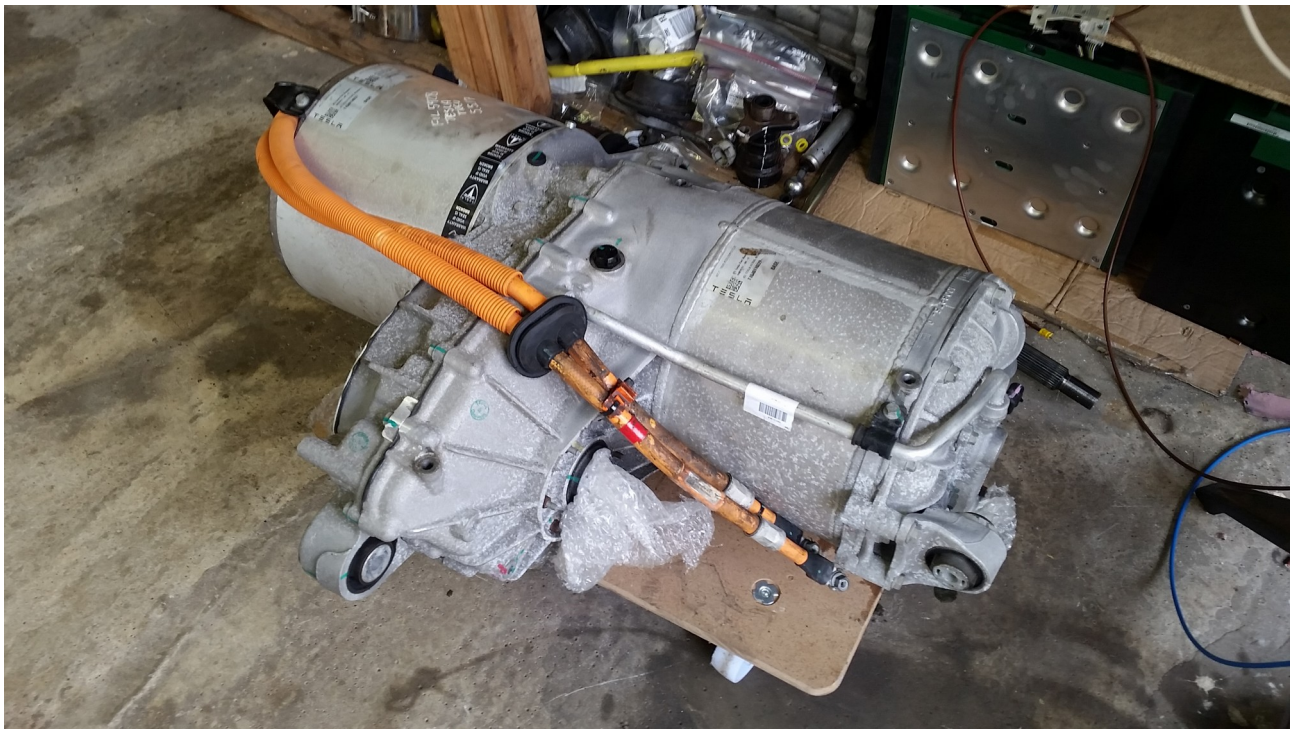


WWW.EVBMW.COM

INFO@EVBMW.COM

INTRODUCTION

The Tesla motors large drive unit has been available now for some five years through vehicle salvage years and recyclers. Prices vary depending on age, type (Base or Sport) and juridiction but typically sit somewhere within $4000-$6000USD. Probably one of the most powerful and compact combination of AC 3 phase induction motor , inverter and gear/ differential unit available today. It is the go to drive unit for those of us wishing to build or convert our own electric vehicles.



Sadly, making it run outside of a Tesla vehicle presents several challenges. Some have been sucessful in decoding the complex CAN (Controller Area Network) communication required by the drive unit to make it run in custom vehicles. These solutions have some limitiations and disadvantages. Discussing these is beyond the scope of this document and instead we will focus on replacing the logic board within the drive unit with an open source replacement allowing full control of the inverter and motor.

BUILDING AND PROGRAMMING THE LOGIC BOARD

As the name implies, the open source logic board for the Tesla Large drive unit is open source. This means that the design files, list of materials and software are freely available in the publinc domain for anyone to use under the terms of the GPL license. No charge or payment is required if you wish to build your own. You can even make changes and improvements but if you do so, I ask that you feed these modifications back into the public domain.

The files are on my Github page :

https://github.com/damienmaguire/Tesla-Drive-Unit

At the time of writing the most recent version (sucessfully bench tested) is Version 2. Vehicle testing is on the way!

The design of this board is largely based on the excellent work of Johannes Heubner and his open source AC Induction motor inverter design. Indeed the software to run the Tesla motor is of his design.

His website can be accessed here: http://johanneshuebner.com/quickcms/index.html%3Fen_features,8.html

and I strongly recommend anyone who intends to build a diy version of this board should read this website throughly.

So if you are not scared away as yet then great! Let's continue.

How to build an SMD pcb is beyond the scope of this document and at this point I am assuming you have built the board according to the design and bill of materials and checked for assembley defects.

The first thing to do is to load the firmware onto the STM32 microprocessor. To do this, we require a device called a JTAG programmer. The one I use is sold by Olimex :

https://www.olimex.com/Products/ARM/JTAG/ARM-USB-OCD/

You will also require a PC or laptop with a usb port to connect the programmer and a piece of software called OPENOCD to communicate with the programmer and allow us to upload the initial program and bootloader. Note that once the bootloader is installed we will not need to use JTAG for uploading new software revisions. This can be done over serial as we will see in a later chapter.

OPENOCD is available in Windoze , MAC and Linux versions. As I use Linux this will be what we cover in this document. In order to make life easy and save having to spend hours downloading files and libraries, I have made a directory of all the files we will need :

https://github.com/damienmaguire/Tesla-Drive-Unit/blob/master/openocd_inverter.tar.gz

Download and extract this archive.

Copy the Openocd binary to /usr/bin

Copy the Openocd folder to /usr/local/share

Now, in order to allow Openocd communicate with the JTAG programmer we need to install libftdi with the following command : sudo apt-get install libftdi-dev

So now we have our programmer software setup it's time to connect the programmer to the PC and the programmer to the logic board like so :

Create a directory called "bin" on your root and download and extract this archinve to that directory :

https://github.com/damienmaguire/Tesla-Drive-Unit/blob/master/JTAG_inverter.tar.gz

With the programmer connected to the PC and logic board we now execute the script to load the bootloader and firmware on the processor :

sudo ./flash.sh

If all goes well after a few seconds the program will load onto the STM32 microprocessor and the second LED on the board will start flashing rapidly. Well done!

Of course there are other ways and other JTAG programmers you can use to load the firmware but this is the one I use. If you have success with another please let me know!

So now we have a basic firmware on the logic board. Now it’s time to load the most recent firmware which you will always find on my github :

https://github.com/damienmaguire/Tesla-Drive-Unit/blob/master/stm32_sineHWCONFIG_TESLA.bin

To do this we need to power up the logic board and connect over the serial interface.

SERIAL COMMUNICATION

Now that we have loaded the bootloader onto the STM32 processor we no longer need to use JTAG for updates. The inverter features an RS232 serial interface and a loader program that runs on Linux that allows us upload new firmware without needing to open the inverter. This is very important for the Tesla inverter as it would be under the car!

All 12v power and signals are carried by the original Tesla 20 pin ampseal connector. Some have been changed from the OEM pin assignments to accomodate the opensource board. A document is availabe with this information on Github :

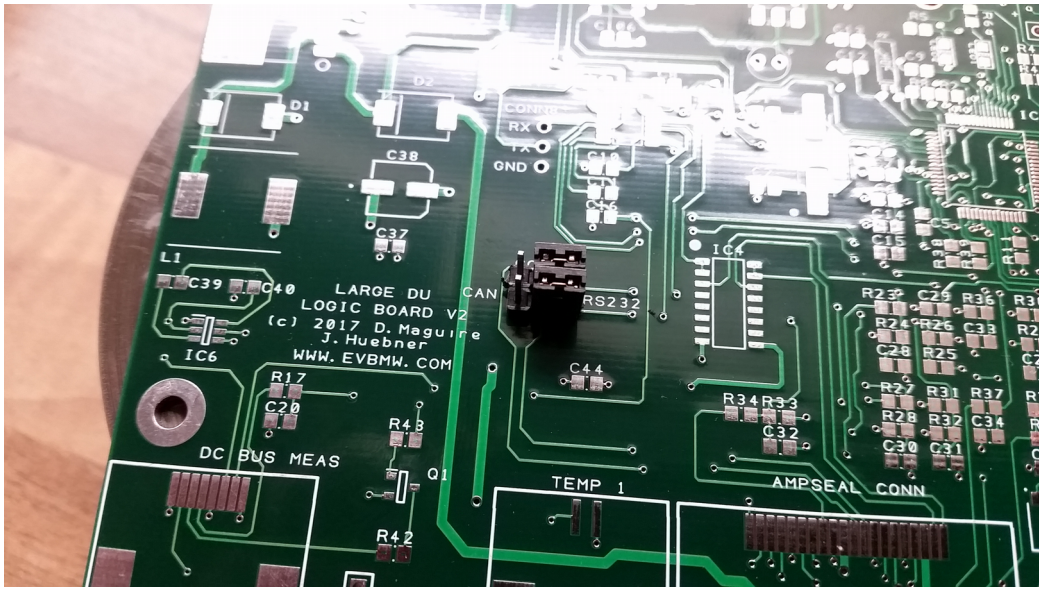
https://github.com/damienmaguire/Tesla-Drive-Unit/blob/master/AMPSEAL_PINOUT_NEW_V2.ods

PIN NUMBER	OEM	OPEN SOURCE
1	IGN +12V	IGN +12V
2	BRAKE ON N.O.	BRAKE ON
3	BRAKE OFF N.C.	PRECHARGE RELAY
4	CAN HIGH	CAN HIGH
5	CAN LOW	CAN LOW
6	CHG PROXIMITY	MAIN CONTACTOR
7	HVIL IN	FORWARD
8	HVIL OUT	REVERSE
9	ENC +5V	ENC +5V
10	ENC A	ENC A
11	GND	GND
12	ACCEL 1 +5V	ACCEL 5V
13	ACCEL 1	ACCEL INPUT
14	ACCEL 2	BRAKE TRANSDUCER
15	ACCEL 1 GND	ACCEL GND
16	ENC B	ENC B
17	ENC GND	ENC GND
18	ENC SHIELD	ENC SHIELD
19	CAN HIGH OUT	
20	CAN LOW OUT	
21	ACCEL 2 +5V	CRUISE IN
22	ACCEL 2 GND	GND
23	12V ALWAYS T30	START

In order to power up the inverter for the first time we connect a 12vdc power supply capabale of at least 5 amps to pin 1 and pin 11. Positive to pin 1 and ground to pin 11.

The inverter should now power up and the leds on the logic board should light with one on constantly and one flashing around 5 times a second.

Jumper CONN17 on the logic board allows pins 4 and 5 to be used as either CAN (controller area network) or RS232 communicatrion. We need RS232 so fit two jumpers such that the middle and right hand pins are joined :



Pin 4 is now RS232 Transmit (Tx) and Pin 5 RS232 Receive (Rx).

If you do not have an RS232 port on your PC (few have these days) you will need an RS232 to USB adapter cable like this one :

Connect a 9 pin D socket to pins 4 and 5. Pin 4 from the inverter goes to pin 2 on the DB9 and pin 5 from the inverter to pin 3 on the DB9. Also connect pin 5 on the DB9 to ground.

To upload firmware we need one final piece of software :

https://github.com/damienmaguire/Tesla-Drive-Unit/blob/master/updater_inverter.tar.gz

Extract this archive to your /bin directory and also copy in the latest firmware from Github :

https://github.com/damienmaguire/Tesla-Drive-Unit/blob/master/stm32_sineHWCONFIG_TESLA.bin

With the inverter powered up and connected via serial type :

./updater stm32_sineHWCONFIG_TESLA.bin /dev/ttyUSB0

With a bit of luck this will now upload the latest firware over serial to the STM32 processor and reset.

WEB INTERFACE AND PARAMETERS

So we have built a board, loaded it with JTAG, connected it to the inverter, loaded the latest firmware over serial. Now what ? Well, now we need to load the configuration parameters and setup the web interface on your PC.

Firstly, we need to install a LAMP stack on our PC. There are many guides on the web. Here is one good example :

https://www.linode.com/docs/web-servers/lamp/install-lamp-stack-on-ubuntu-16-04

Basically, what we are doing is creating a local webserver on the PC that will communicate over serial with the inverter and display data in a nice graphical manner. We can also issues commands, change parameters and upload new firmware with the web interface.

Download the latest files from Johannes's website here :

http://johanneshuebner.com/quickcms/files/inverter.zip

and extract the archive.

Copy the contents of /inverter/tools/web interface to /var/www/html

and restart your computer.

Now, when the inverter is connected over serial to the PC and powered on open a web browser and type localhost/ in the address bar. Not the serach bar as Google won't know what your doing!

You should now have a webpage with all the inverter parameters listed.

PHYSICAL INSTALLATION

In order to install the open source logic board:

- Remove the inverter cover.

- Unclip the seven wiring plugs from the original logic board.
- Unscrew and remove the six screws holding the old Tesla logic board in place and remove this board.
- Place the open source logic board in the same location and secure it in place the with six screws.
- Re connect the seven wiring plugs.

- Once done your inverter should look something like this :