I have put together this document to make sure everyone understands how I have connected the datalogger together and some of the problems that were encountered. A more detailed documentation will be written by the end of August. There are also some minor tasks remaining as well:

* Need to order some cable that can fit into the DC Jacks from the storage batteries.
* The SD card inside the datalogger needs to be formatted, I have ordered new ones as well
* Order cable for installation (white/grey/black). I was thinking we use different colours so that it is easy to identify and debug. And since we can buy the cable per metre, we shouldn’t be worried about buying a whole reel and then not using it
* Gather tools that we need for the expedition
* **@David**: The code needs to be edited and could you send me the zipped file of the project so that I could edit it as well in case you are busy:
  + sd card filesystem and how we store the files and filenames (I think David has already done this)
  + Status indication of the LEDs. We need to decide what we want to indicate (I have written down my ideas below). These are the pin locations for each LED:
    - Pin23 = red led
    - Pin22 = green led middle
    - Pin21 = green led

For status indication, this is what I feel. Feel free to add/ammend anything if something comes to mind or if you have a better idea:

* Red led to indicate that the power is on
* Middle green led to indicate when modem is connected and uploading data
* Other green led can turn on when we sample data from the sensors
* We can also make the leds to flash repeatedly (e.g. every 1 second) when there are errors in uploading/writing to the sd card or infinite loop…etc.



USB

DC Jacks

Green LED (Middle)

Green LED

Red LED

For the USB, the modem will connect on the outside whereas there is a USB male connector on the inside which has a long cable which will be connected to the mbed Development board (the red PCB containing the MBED). This cable needs to trimmed.

In terms of tools, this is what I feel we need. Please feel free to add anything:

* Drill
* Stanley knife
* Soldering iron, solder wick, solder, solder sucker
* DMM
* Insulating tape
* Screwdrivers
* Wire, Wirestripper
* Crimp Tool, crimps and all the leftover connectors
* Extra components (ACS712, AD7993, IL300, MBED, MODEM)
* We may need to take some extra resistors for the DC voltage board as currently the ADC is maxing out at around 20V which is what we planned for but in case we need to measure something larger which is unlikely, we may need to scale the voltage down using a bigger divider ratio.
* Duct Tape

We may need to order an extra MBED just incase but if we don’t then we can get the one currently in Rwanda as well. We have most of the other components/chips.

Below is the code for the AC voltages and currents to get the peak-peak value. I have tested the fundamental code and it works. But I have added some code to average the data. This is just a precaution in case we get some noise spikes which may give us a wrong peak-peak value. **Also note that the ADCread function now has a shorter wait time of 1ms (0.001s).**

int ACread(int pinNO, int ADCAddress){

int val=0;

int min=512;

int max=512;

int counter=0;

int numSamples = 600; // increase this by increments of 50

int pk2pk=0;

int avg=0;

int avgcount=0;

int numCycles= 10; // number of cycles to average over

while (counter <= numSamples)

{

val = ADCread(pinNo, ADCAddress); // get sample of AC wave

if (val<min)

min = val;

if (val>max)

max = val;

if (counter%50 == 0){ // calculate average every 50 samples – this can be changed

pk2pk = max - min;

avg = (pk2pk + (avg\*avgcount)/(1+avgcount)) // calculates a cumulative average

avgcount++;

// reset min and max values after calculating average

min=512;

max=512;

}

counter++;

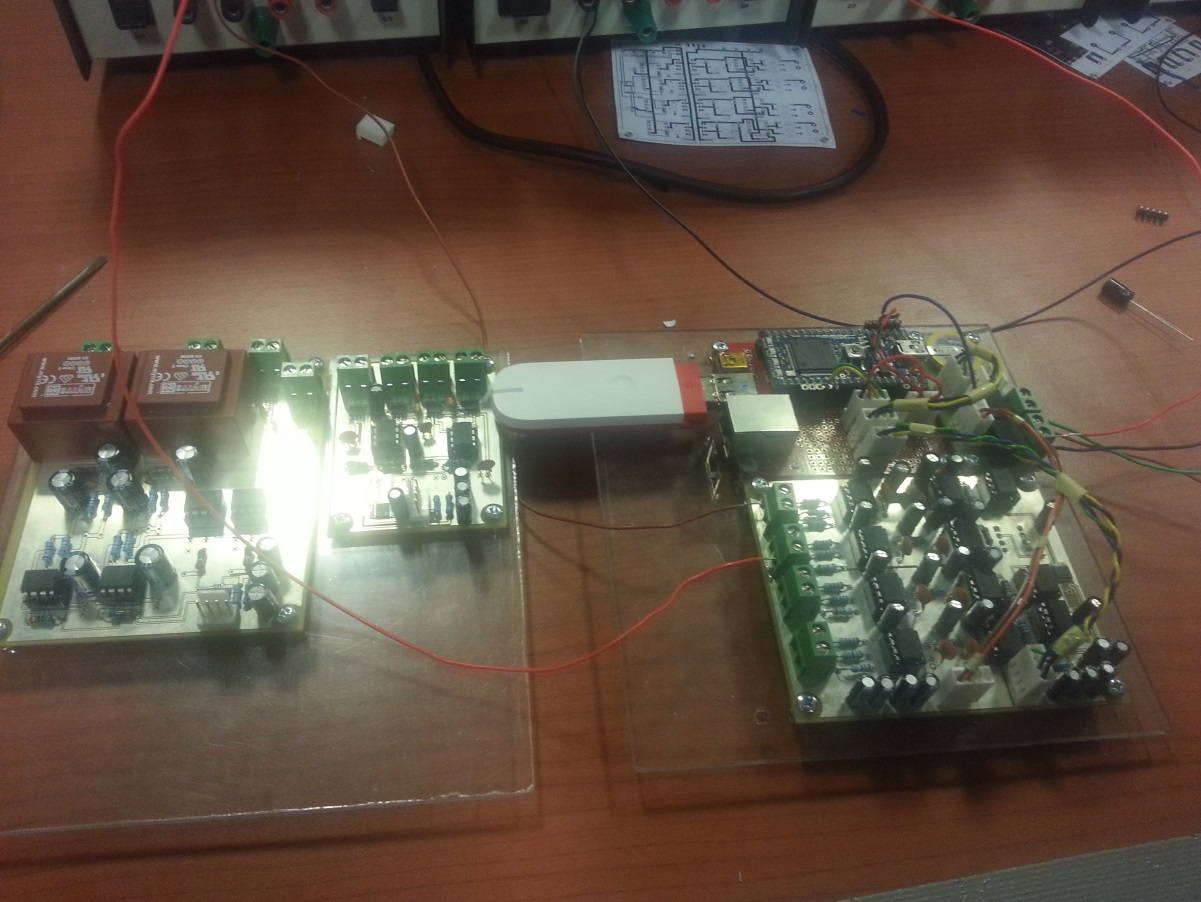
wait(0.002); // wait for a time for which will sample a whole number of 50Hz cycles

}

return avg;

}

Below is a picture of the sensors and how everything is connected.



DC Current Board

Power Board (Slightly concealed in the picture)

Main Board

DC Voltage Board

AC Board

2.54mm Male PCB header

All the boards are mounted on 3mm Polycarbonate. The mounting holes are also 3mm and the screws and nuts used are M3. However, the screws were too long to fit into the box and therefore I trimmed them using the Dremmel.

Most of the wire used to jump from board to board are stranded wires where possible because it is easier to crimp onto. The wires are then clipped onto some female sockets which then connect onto the 2.54mm PCB headers. There is a colour code I have used as well:

* Yellow: SCL
* Green: SDA
* Black: GND-1
* Blue: 5V-1
* Orange: 5V-2
* Brown: GND-2
* Red: 5V-1 (From main board into the mbed)

The 5V-1 and 5V-2 is for the 2 different supplies that we are using for the voltage board isolation. This is similar for the GND as well.

In terms of the cables for SCL and SDA, I followed some guidelines from the Phillips documentation about I2C when the cables are longer than 100mm.

“The SCL line can be twisted with a Vss return and the SDA line twisted with a Vdd return. With the second option capacitors are needed to decouple the Vdd line to the Vss line at BOTH ends of the twisted pairs. Interference can be minimised by using shielded cable with the shield connected to Vss, but there must be low capacitive coupling between the SDA and SCL lines to minimise crosstalk.”

I twisted the SCL with GND and SDA with 5V and then taped the cables to keep things neat and everything seemed to work.

For connections into the mbed pins, I have used the breakaway PCB headers. These fit quite perfectly and require a plier to pry them out. I have then soldered the wires from the other boards to the PCB header pins and used some heat shrink for insulation. This may be difficult to describe and since I don’t have a picture, you may have to see it to understand.

**Note: The MBED gives a 5V supply from the VU pin when it is connected via a PC only. This 5V is then connected to the USB 5V for where the modem is supposed to be connected. Therefore, when not connected to the PC the modem will not be powered even if you power the mbed through the Vin pin. To ensure that the modem is powered, we need to make sure that 5V is connected into the VU pin.**

**Notes on the PCBs:**

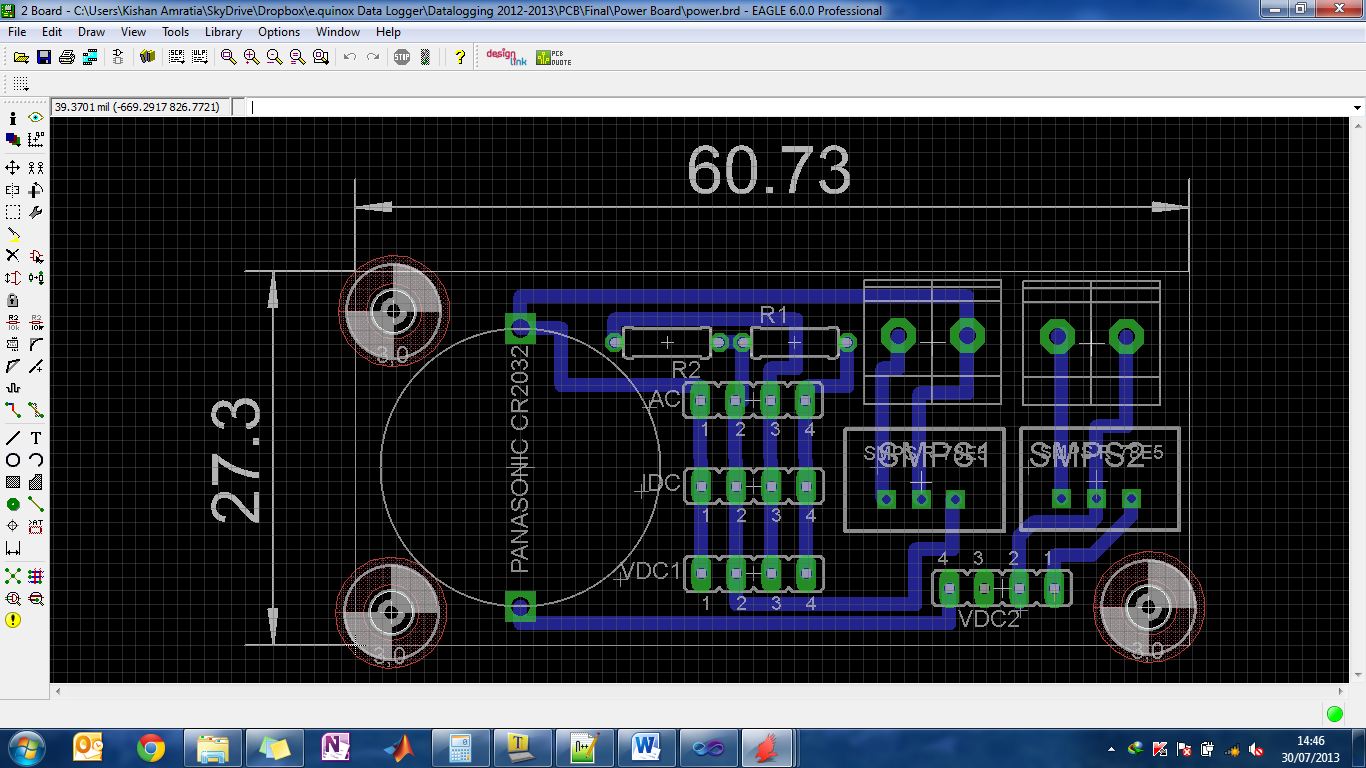
* The DC Voltage board and DC Current board have some minor problems. The DC voltage board had 2 separate GND planes to ensure that we do not connect both sides of the kiosk together but this was problematic as the IL300s and the opamps need to be referenced from the same GND. Therefore I asked Vic for his advice and he said since this is a DC board, we can use one ground plane and there wouldn’t be a problem. So to fix this, I just created a solder joint between the 2 GND planes. Again, this may be difficult to explain in words without seeing it.
* The DC current board had a problem with 2 of its sensors. There was track connected at the wrong place and needed to be cut. This was partly my fault as I had checked the final PCBs before ordering them however, it must have slipped by me. It isn’t a major drawback as the PCB works fine after cutting the track.

GND-1

12V IN-1

**Power Board**

12V IN-2



Primary

GND-2

Secondary

GND-1

5V-1

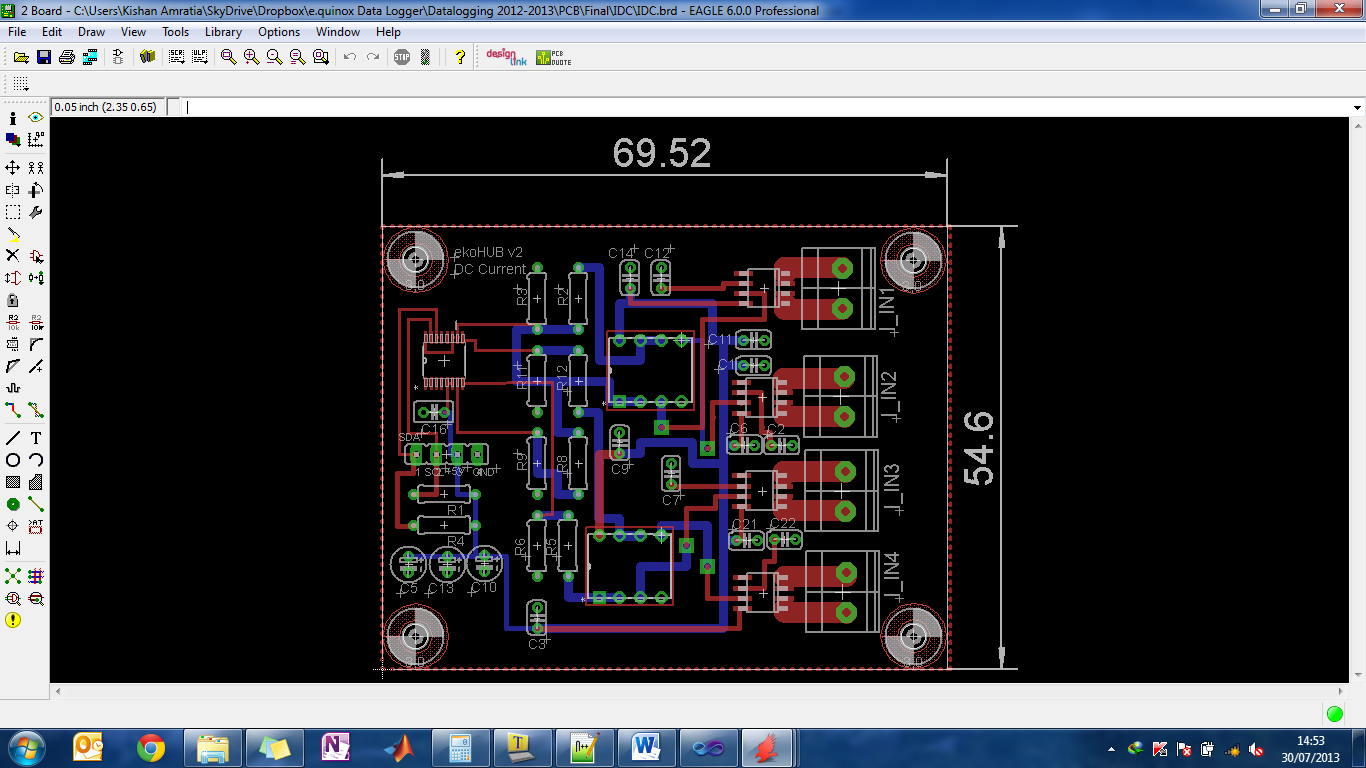
SCL

SDA

VB

GND-2

5V-2

**DC Current Board**

GND-1

5V-1

SDA

SCL

Need to cut this track

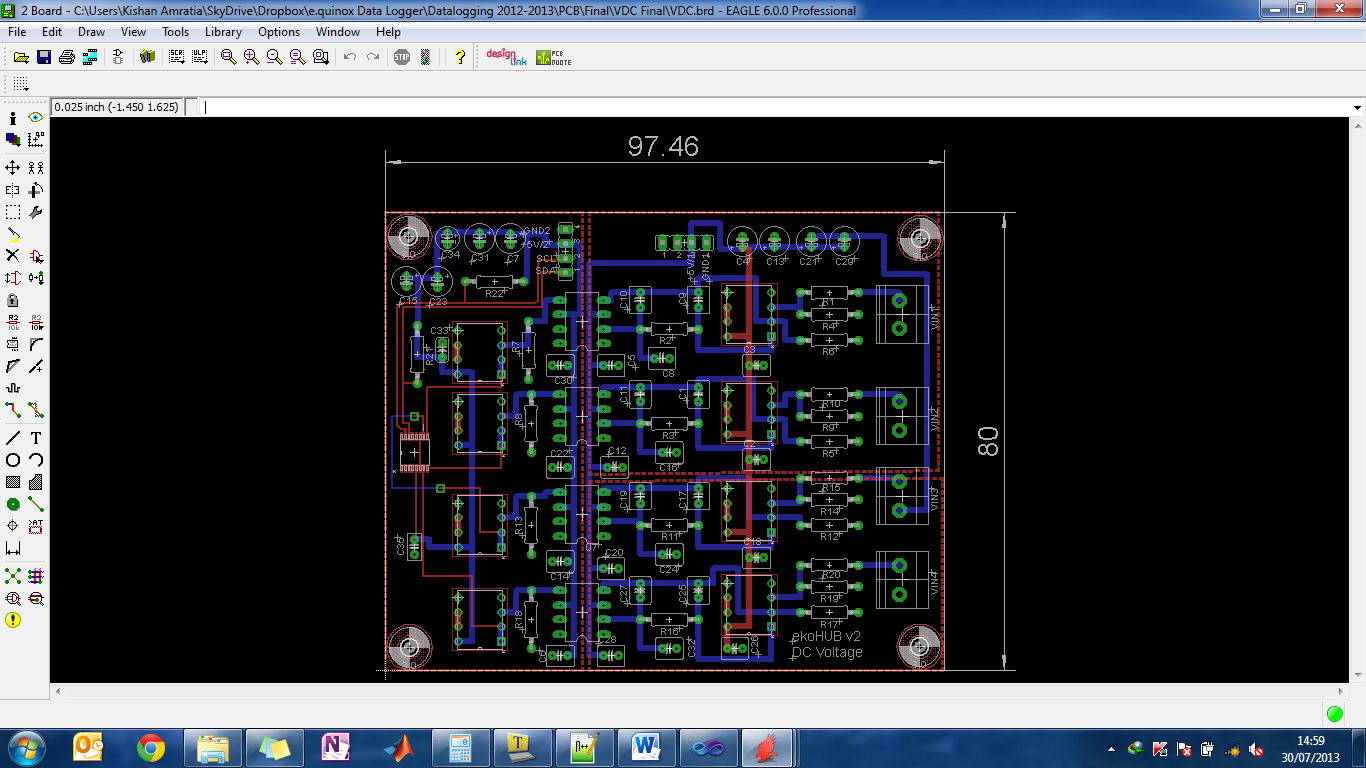
Another thing about the DC Current board is that we decided not to use the potential divider at the end since it is very insensitive and we would rather have the centre of the chip at half the reference voltage than something below. So instead we used a 1ohm resistor and connected it to the input of the ADC.

**DC Voltage Board**

The 2 GND planes on the right hand side stacked vertically of this PCB need to be connected together.

GND-2

5V-2



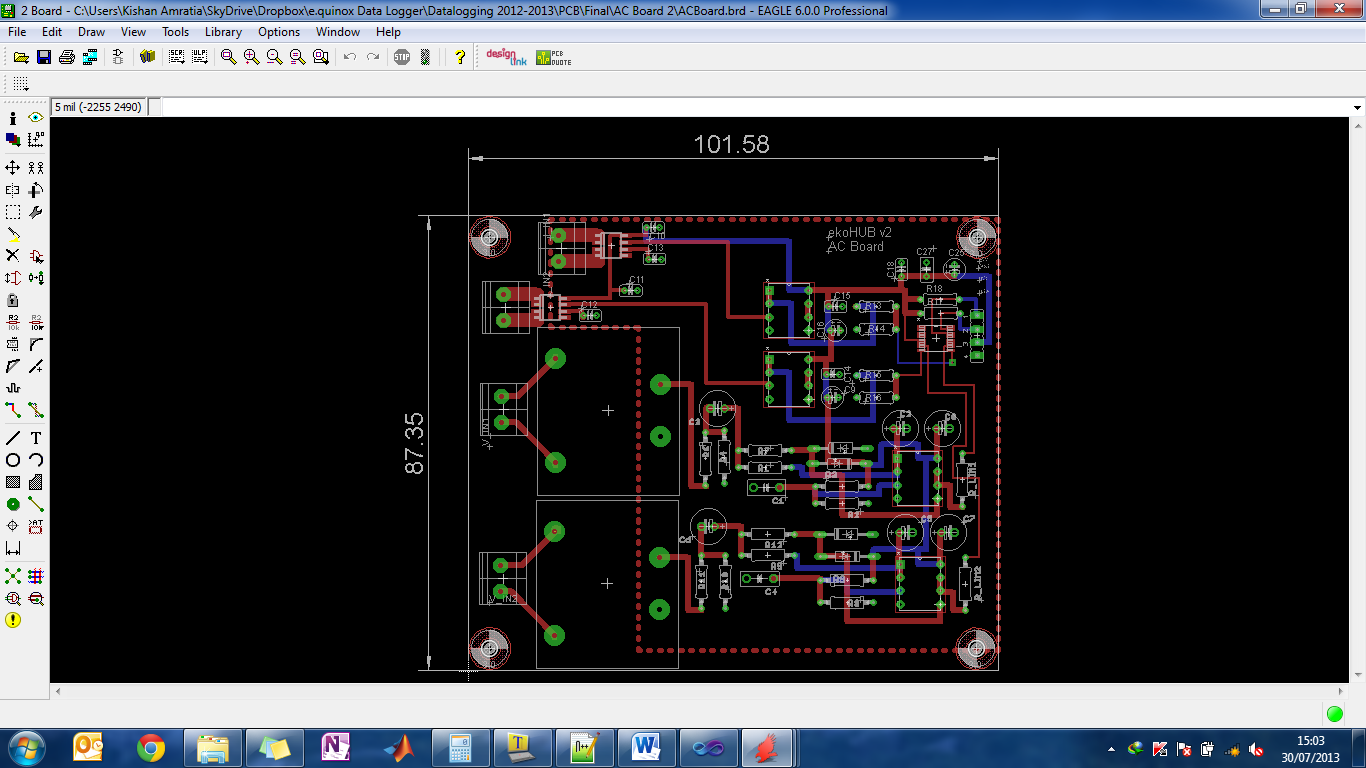
SDA

SCL

GND-1

5V-1

**AC Board**



SDA

GND-1

5V-1

SCL