Rasdaq Measurement System
by
ETF LTH
v1.0

Hardware

Kit includes:

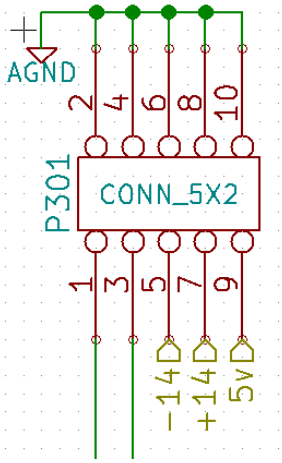
1. 1x Rasdaq
2. 1x Power supply 5V 2A
3. 1x Power Measurement Adapter
4. 2x Dummy plugs

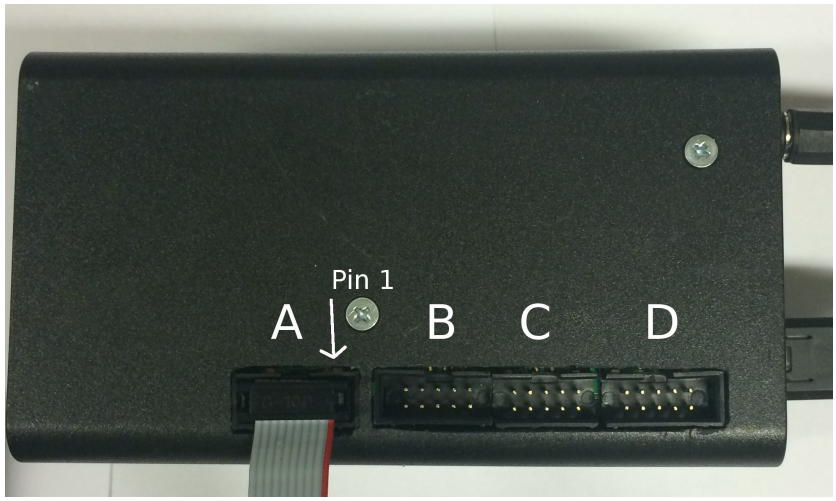
Rasdaq

The Rasdaq system is based on a Raspberry pi module, with a extension board containing one TI ADS8568 and one Atmega88 for accurate sample rate.
The ADS8568 contains eight 16-bit channels that can be simultaneously sampled at 500Khz with an input swing from -12V to 12V.

Because of restrictions in the Raspberrys non deterministic execution it's not possible to sample at full speed, the maximum sample rate that is stable is 200Khz for 1-8 channels. Higher rates might work but tends to stall the ADC from operating after first run. This might be fixable in software but there are no timetable for this.

The Rasdaq has four connection slots, named A-D each connector has two ad channels connected to it. There is also power supplied in this connector to drive external converters.





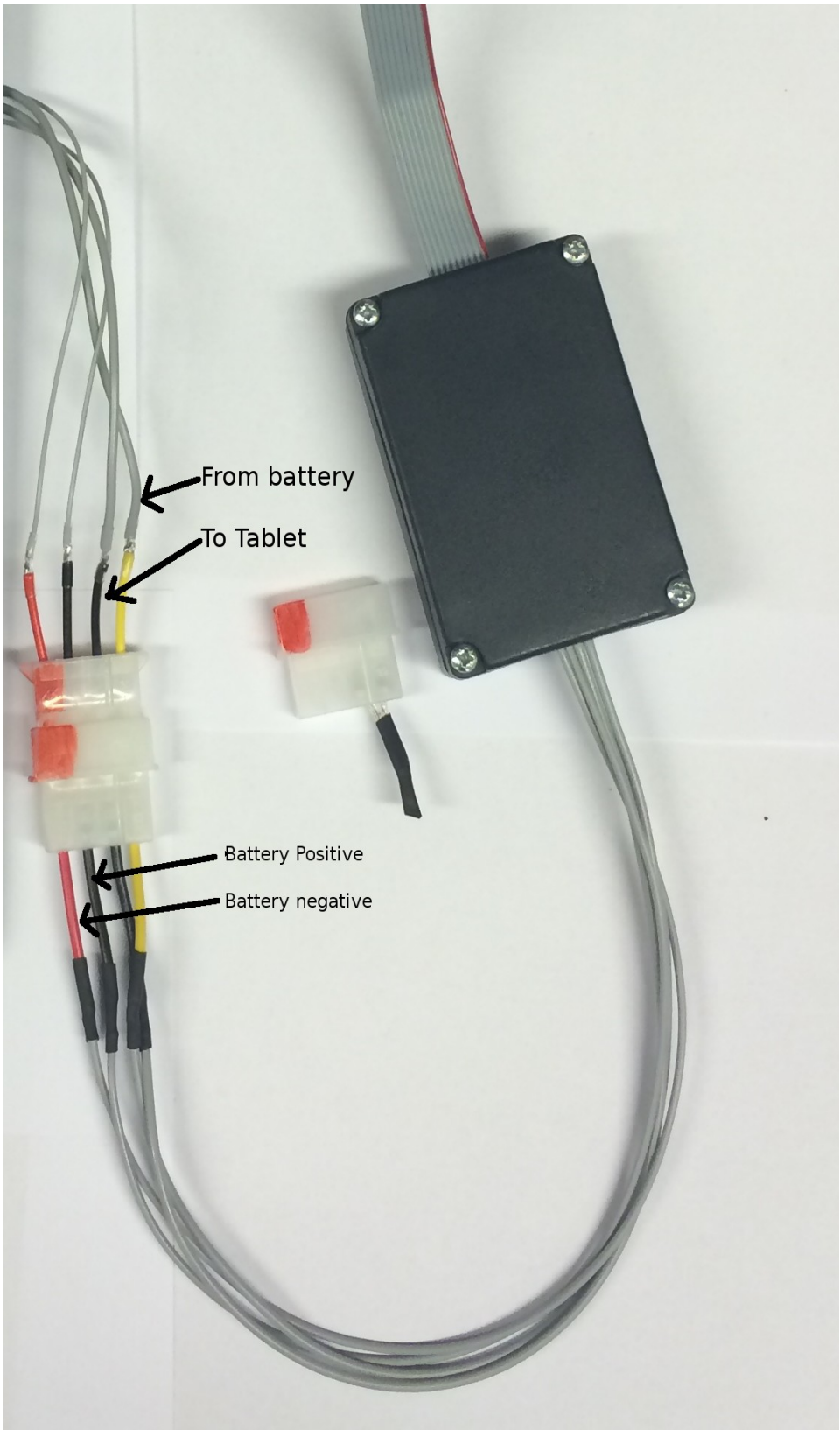
Power Supply
If this needs to be replaced make sure the polarity of the connector is correct, and that is is of good quality, to large swings in the input voltage might affect the accuracy of the measurements

Positive in center and negative on outer ring.

Power Measurement Adapter

This box connects to the rasdaq and use two channel to measure both the current and the voltage.
It measures the current by using a small resistor (0.05ohm) and amplifies the voltage drop 50 times.

Warning: The plug to Tablet need to be positioned the right way, RED to RED. Failure to do this might destroy the battery.
The connector is polarized so that it should only fit in one way but the leads might still touch if it's tested.



Dummy plugs

Measuring the current requires the path from the battery to the tablet to be broken, this means the the tablet only function when connected to the Rasdaq.
By using the dummy plugs the connection is made again and the tablet can be used as normal.
This plug is also useful when charging the tablets.

Warning: This plug need to be positioned the right way, RED to RED. Failure to do this might destroy the battery.

The connector is polarized so that it should only fit in one way but the leads might still touch if it's tested.

Software

Rasdaq

The Rasdaq connects through Ethernet and expects to get an IP address from DHCP. To start and downloading measurements SSH is used.

Log on to the rasdaq with
User: root
Password: toor

For doing measurements run the program rdqtool

```
# rdqtool -h
rdqtool (c) Fredrik Ahlberg, 2013 <fredrik@etf.nu>

Options:    -q      quiet
            -c[n]   number of channels, 1..8, default: 8
            -r[fs]  sample rate in Hz, 1..500000, default: 10000
            -b[n]   maximum buffer size in tuples, 1..8192
            -o[fn]  output filename, default: stdout
            -n[ns]  Stop after n samples, default: run forever
```

This gives a number of choices on how to to the measurements.
The most important parts are number of channels, sample rate an output filename.

The channels are divided as following
Slot A CH 1,2
Slot B CH 3.4
Slot C CH 5,6
Slot D CH 7.8

Number of channels always starts from channel 1 and forward. It is for example not possible to only sample CH 8.

Example:
rdqtool -c 2 -r 100000 -o test.bin -n 10000
Samples takes 10000 samples in 100Khz from CH 1,2 and saves it to the file test.bin

It is recommended to save the measurements to the internal memory and then download the file as the Raspberry have limited Ethernet capability while it's running the ADC

Computer

The files created from the Rasdaq save the samples as signed 16-bit in the channel sequence.
So for two channels the file looks as following.

CH1_1,CH2_1, CH1_2,CH2_2 and so on.

Example octave script to plot the measured Current and Voltage

```
fid = fopen('test.bin');
                        % Open file
adc = fread(fid, [2, inf], '*int16');
        % Read file with 2 Channels
fclose(fid);

% sample rate [Hz]
fs = 100000.0;

% calculate number of samples read
samples = size(adc, 2);

% create time vector
t = linspace(0, samples/fs, samples);

% recalculate raw readings as voltage.
% the preamplifiers invert the measured voltages.
adc = double(adc);
adc = -adc * 12.0/32768.0;

% calculate current from voltage (shut resistance = 50 mohm, 0.05% tol, 50 V/V gain)
ch0 = adc(1, :) / (0.05 * 50);

% battery voltage on channel 1
ch1 = adc(2, :);

% apply a low-pass filter at 1kHz
fc = 1000.0;
[b,a] = butter(2, fc/fs);
ch0 = filter(b, a, ch0);
ch1 = filter(b, a, ch1);
% calculate power
ptot = ch0 .* ch1;

% plot
close all
clf
figure(1)
hold on
plot(t, ch1, 'r')
plot(t, ch0, 'g')
plot(t, ptot, 'b')
xlabel('Time [s]')
legend('Battery Voltage [V]', 'Battery Current [A]', 'Power [W]')
```

Taking samples

1. Connect Rasdaq to Ethernet
2. Connect power measurement box
3. Connect Tablet to power measurement box,
   Make sure the connector is the correct way, Red to red
4. Use ssh to connect to Rasdaq and start taking samples
5. Start demo on Tablet
6. When demo done stop program on Rasdaq and download data.