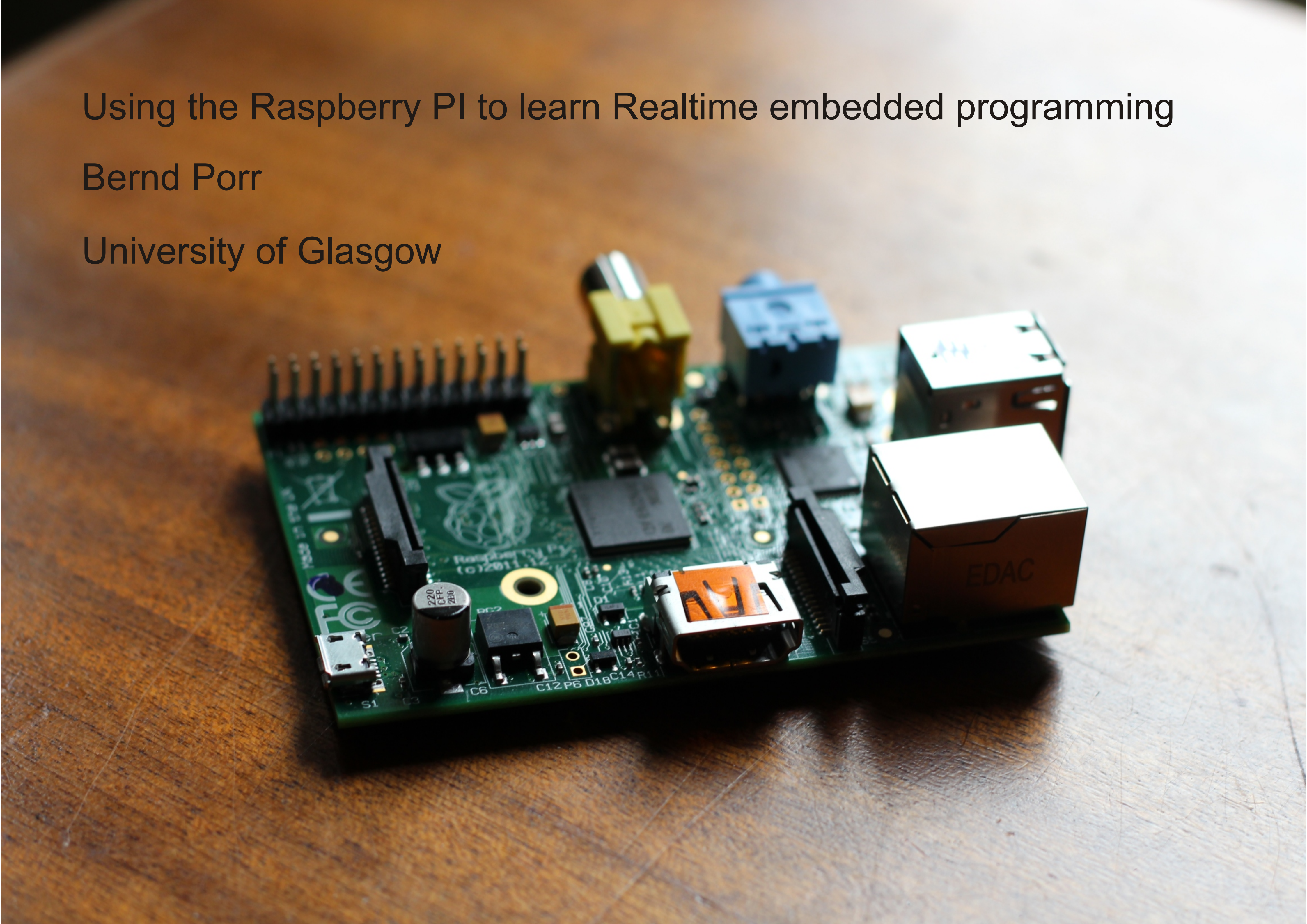
Using the Raspberry PI to learn Realtime embedded programming

Bernd Porr

University of Glasgow



Teaching realtime processing

The standard academic approach

The standard academic approach

The internet cafe approach

The internet cafe approach

The students get:

simulated data

run a simulation

and write a report

...for the bin!

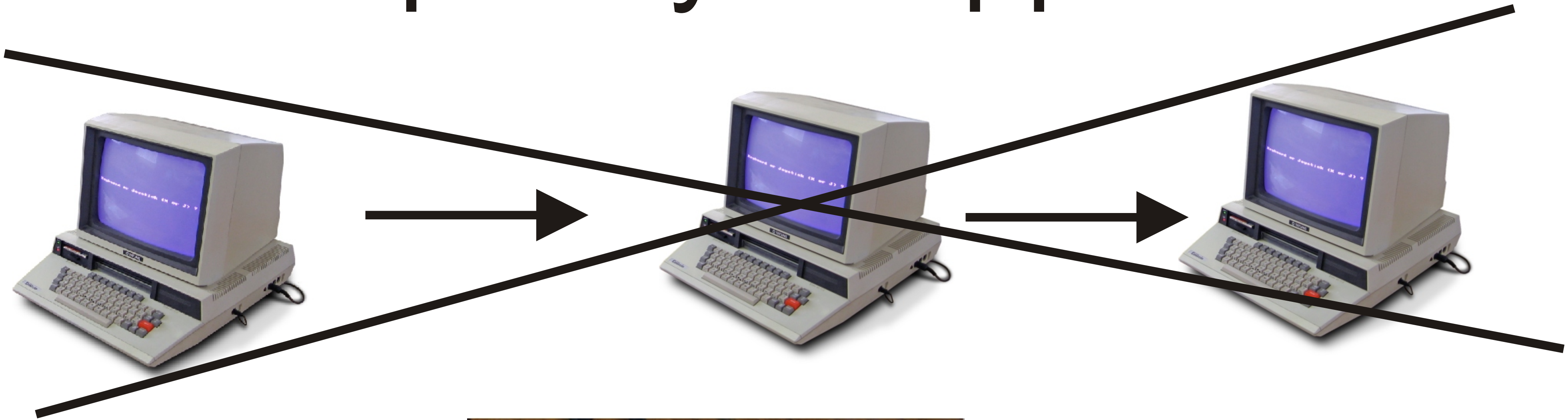
wikipedia

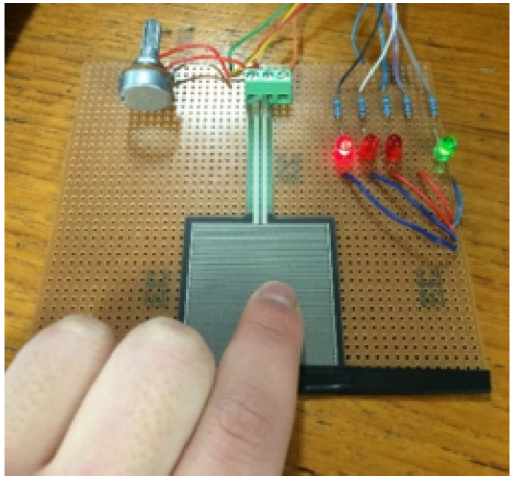
ZZZZZZZZZZ

What's the alternative?

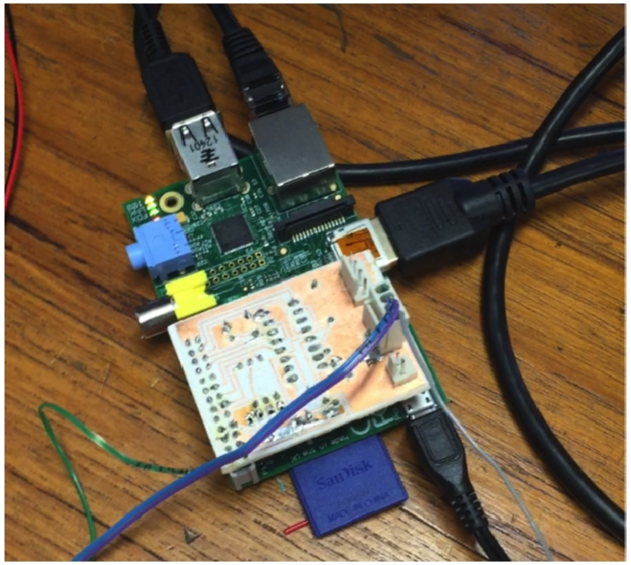
galleryhip (marked for reuse)

The Raspberry PI approach

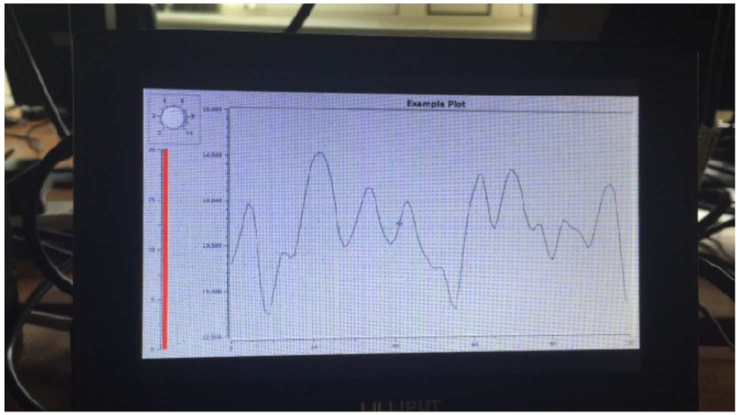




Build a sensor...



..process the data with the Raspberry PI...



..and make a YouTube video about the result.

Problem based learning with the Raspberry PI

The Task:

Measure a physical quantity and plot it in realtime on the screen.

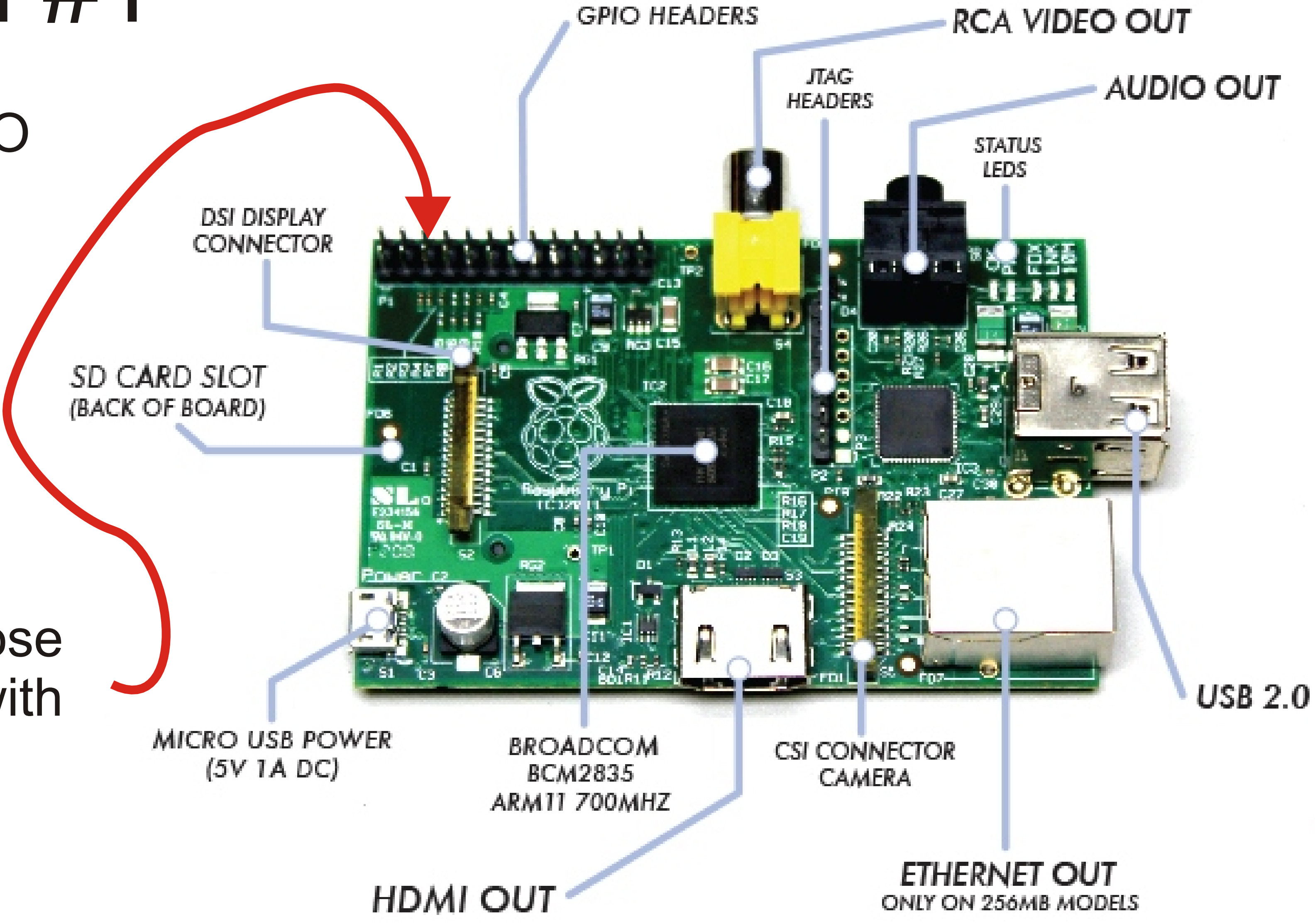
Problem based learning with the Raspberry PI

Problem #1

The RPI has NO analogue input.

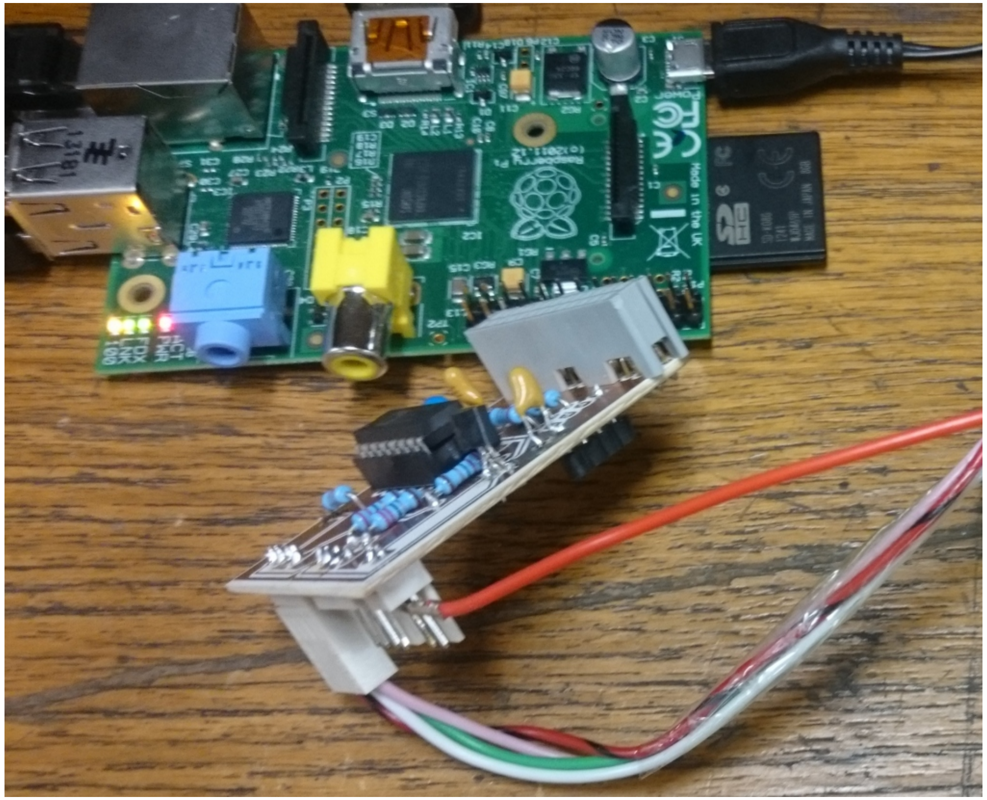
but

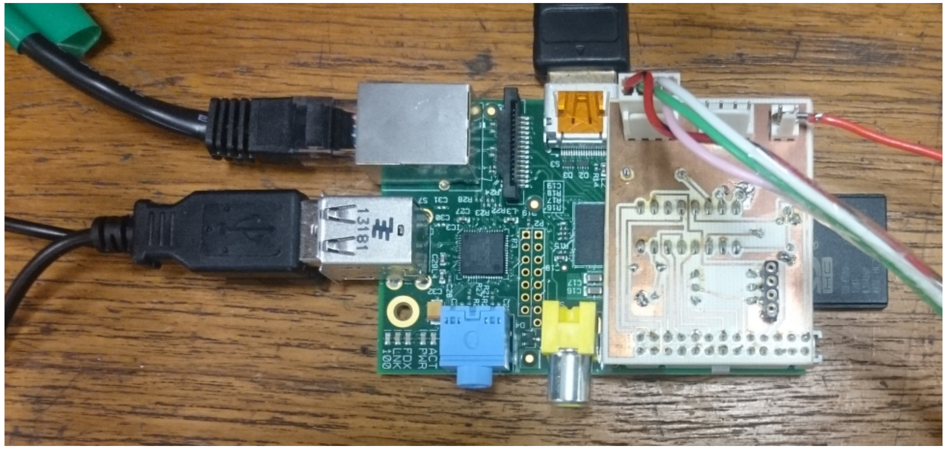
a general purpose I/O connector with SPI and I2C!



Problem based learning with the Raspberry PI

Solution





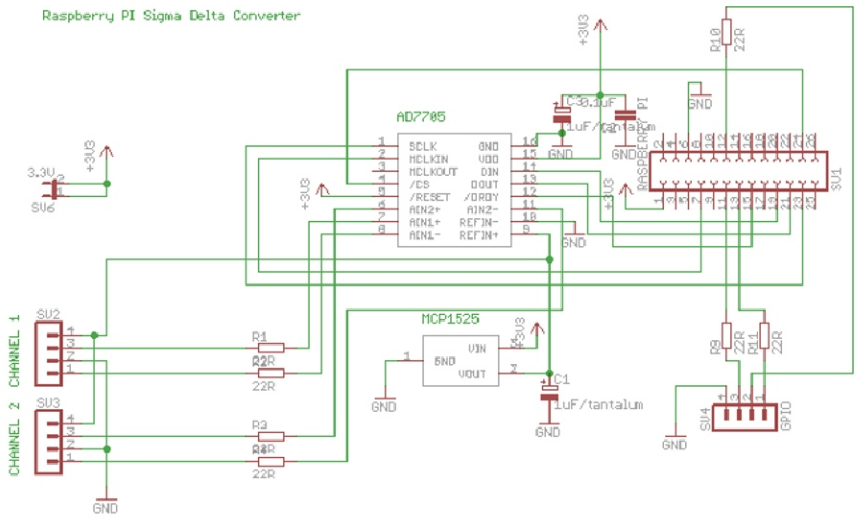


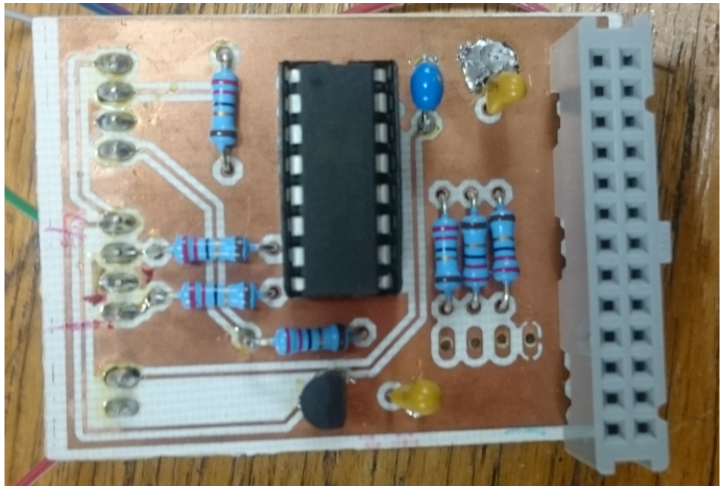
Figure 1: Raspberry Pi Sigma Delta Converter (Bernd Porr,2015)

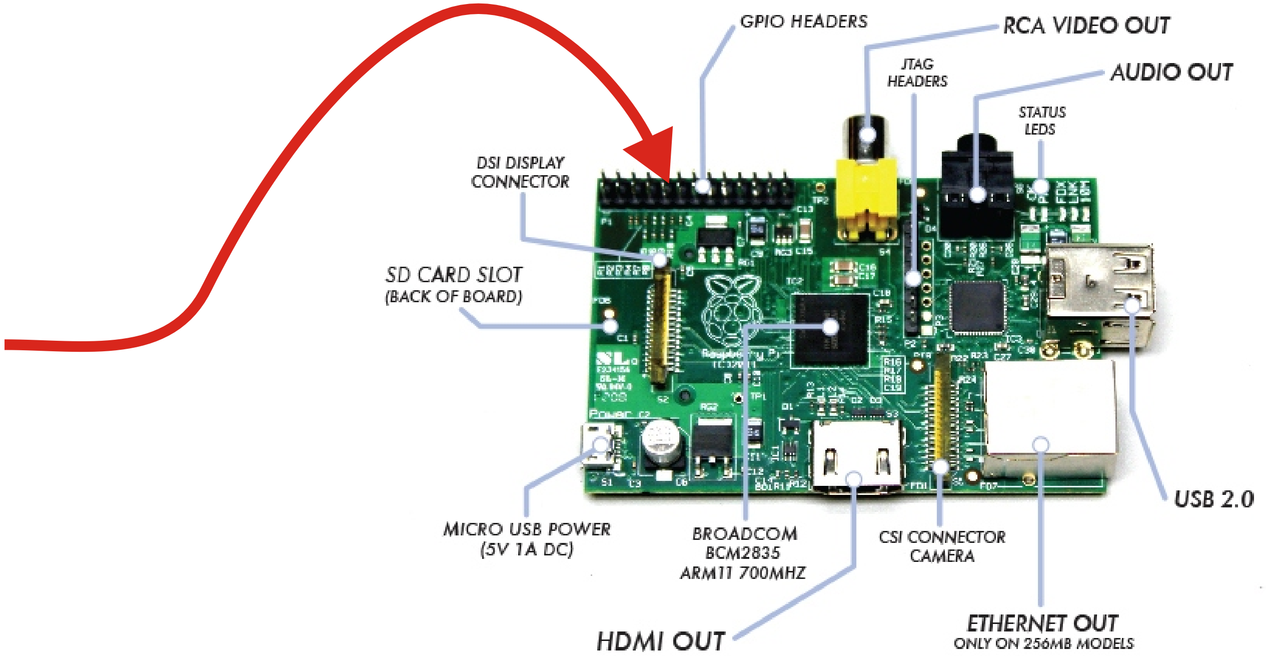
The students have to build/design/hack an analogue to digital PCB to be able to measure analogue signals.

Problem based learning with the Raspberry PI

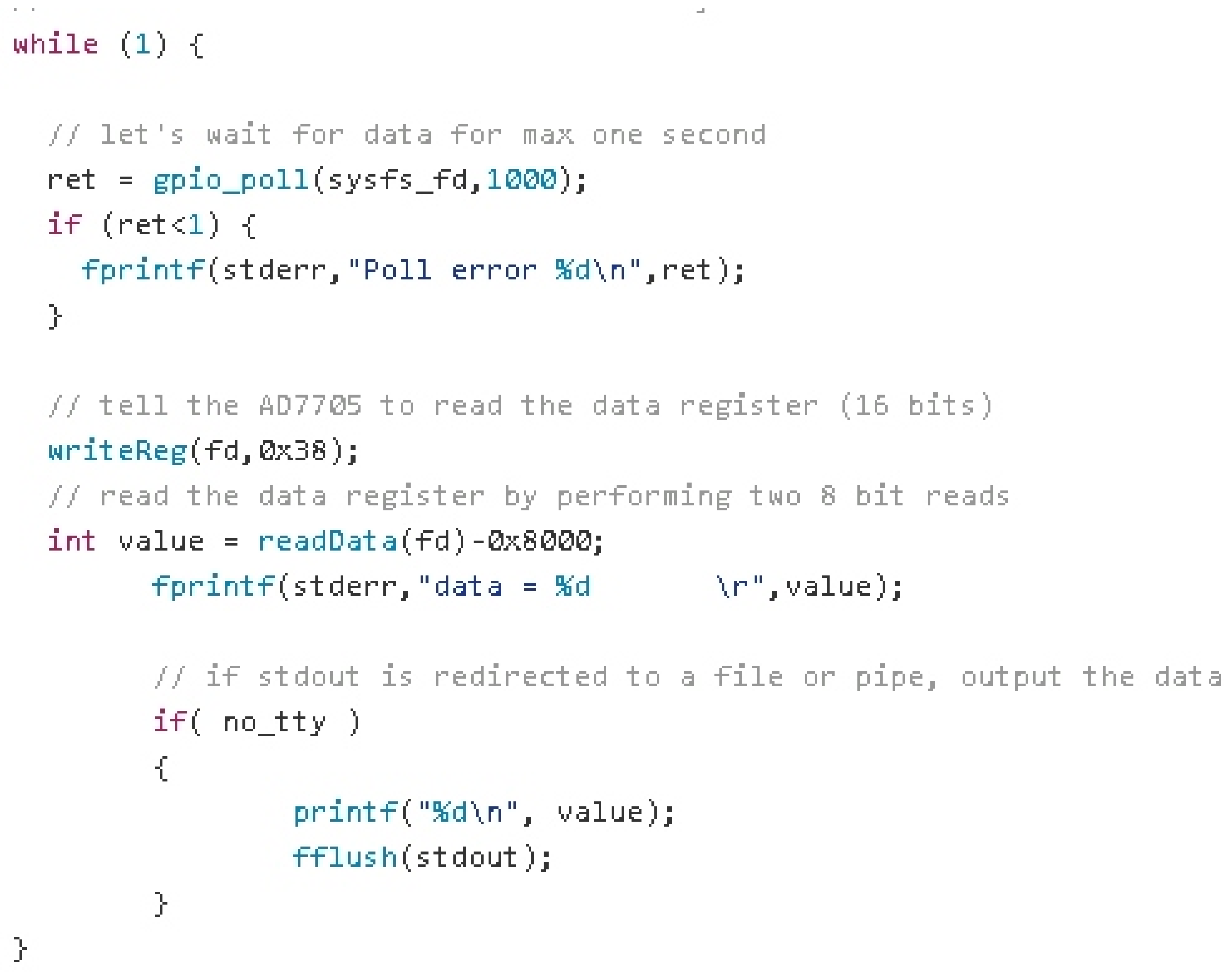
Problem #2

How to get the data into the raspberry PI?





Solution:
Learning how to use /dev/spi under Linux and C

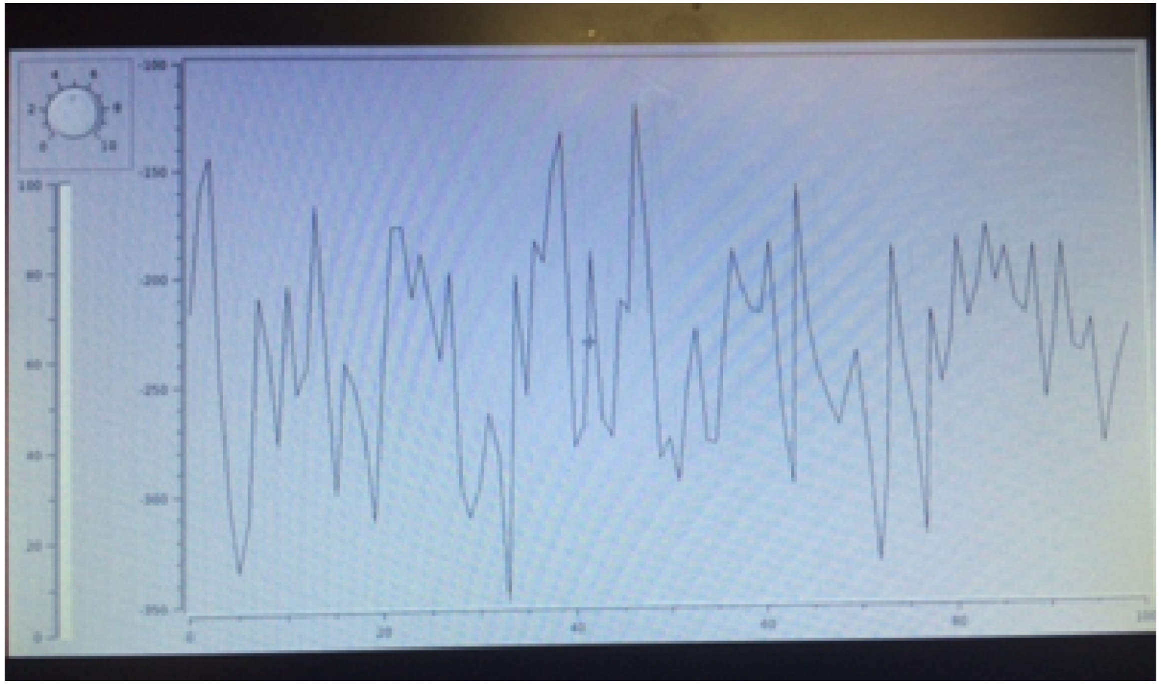


```
while (1) {

  // let's wait for data for max one second
  ret = gpio_poll(sysfs_fd,1000);
  if (ret<1) {
    fprintf(stderr,"Poll error %d\n",ret);
  }

  // tell the AD7705 to read the data register (16 bits)
  writeReg(fd,0x38);
  // read the data register by performing two 8 bit reads
  int value = readData(fd)-0x8000;
        fprintf(stderr,"data = %d       \r",value);

        // if stdout is redirected to a file or pipe, output the data
        if( no_tty )
        {
                printf("%d\n", value);
                fflush(stdout);
        }
}
```

Problem based learning with the Raspberry PI

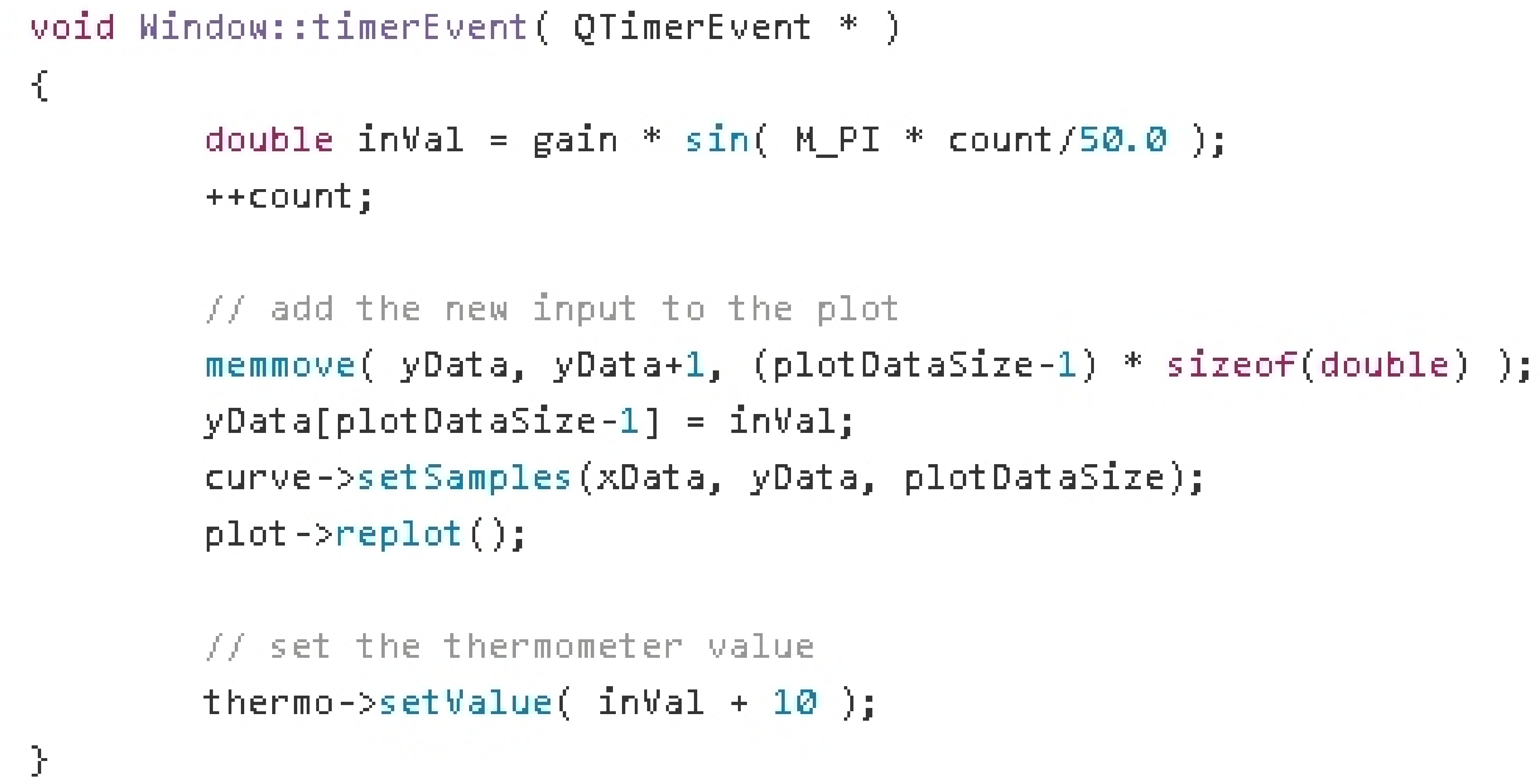
Problem #3

How to plot the data?



Solution:

Learn how to use the QT library (in C++) and the scientific plotting library QWT.

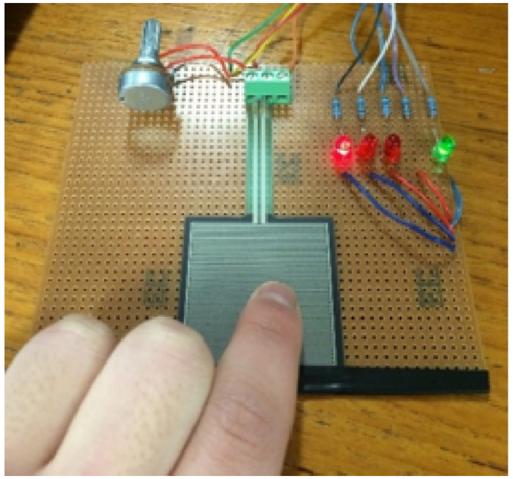


```
void Window::timerEvent( QTimerEvent * )
{
        double inVal = gain * sin( M_PI * count/50.0 );
        ++count;

        // add the new input to the plot
        memmove( yData, yData+1, (plotDataSize-1) * sizeof(double) );
        yData[plotDataSize-1] = inVal;
        curve->setSamples(xData, yData, plotDataSize);
        plot->replot();

        // set the thermometer value
        thermo->setValue( inVal + 10 );
}
```

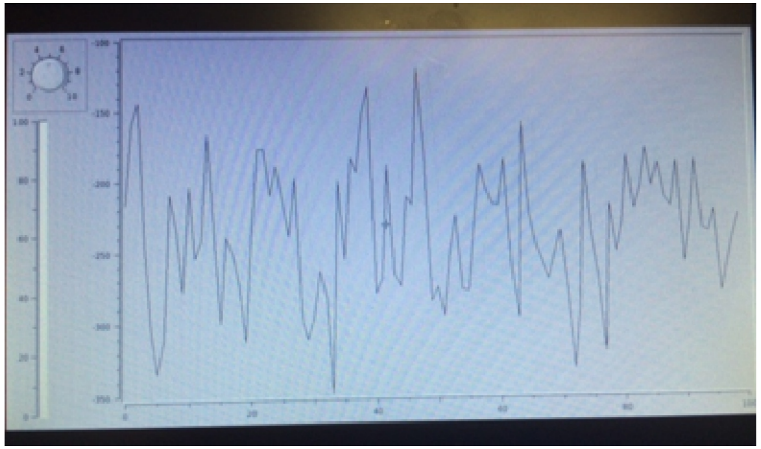
Problem based learning with the Raspberry PI

Problem #4

How to record data at a high speed
and
plot it at a lower speed?



high speed
acquisition

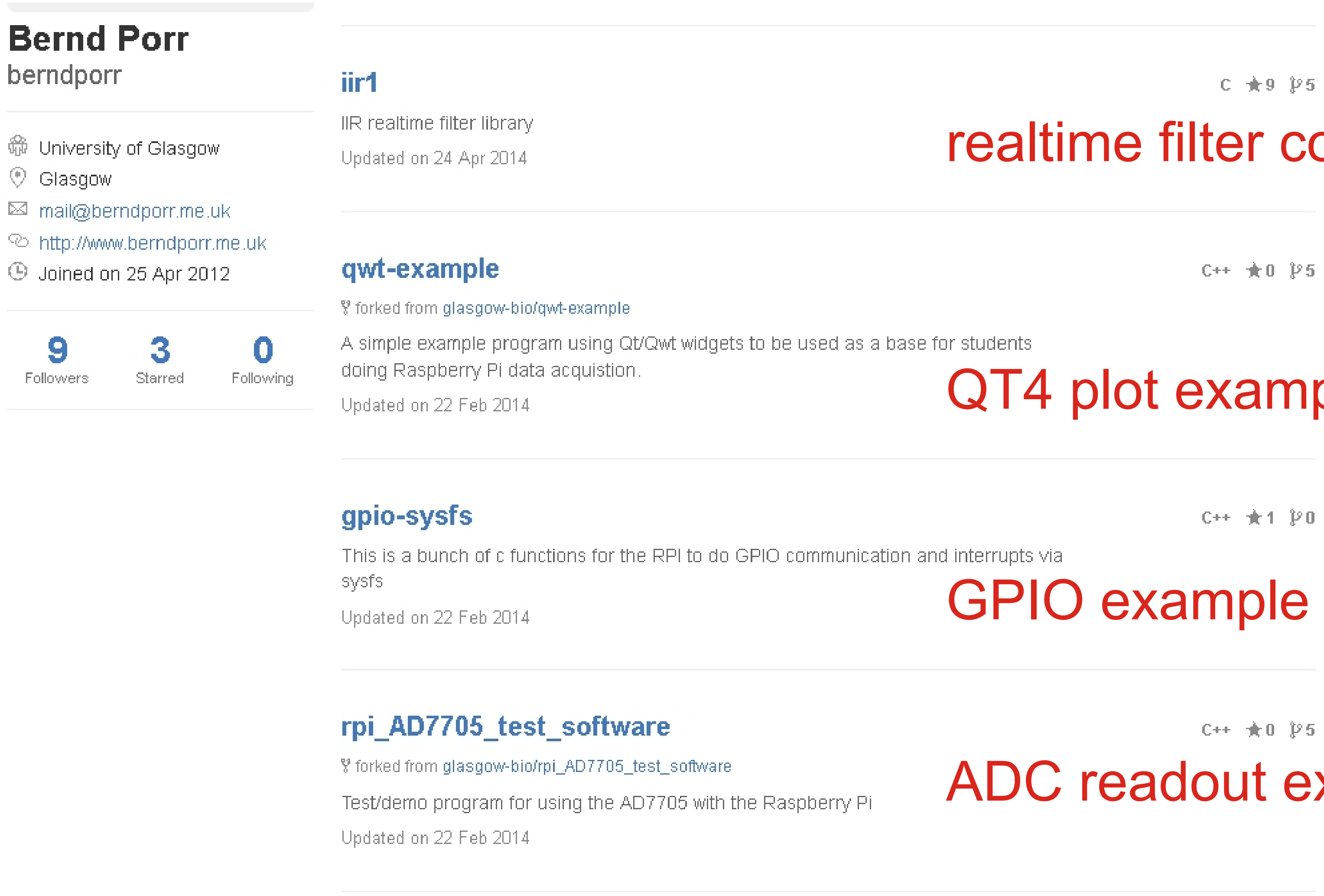


low refresh rate

Solution: multithreaded programming (different threads for data acquisition and plotting using a ringbuffer)

Problem based learning needs support

Example code on GITHUB for the project



Bernd Porr
berndporr

University of Glasgow
Glasgow
mail@berndporr.me.uk
http://www.berndporr.me.uk
Joined on 25 Apr 2012

9 Followers
3 Starred
0 Following

iir1 C ★9 ⑂5

IIR realtime filter library

Updated on 24 Apr 2014

realtime filter code

qwt-example C++ ★0 ⑂5

⑂ forked from glasgow-bio/qwt-example

A simple example program using Qt/Qwt widgets to be used as a base for students doing Raspberry Pi data acquistion.

Updated on 22 Feb 2014

QT4 plot example + QThread

gpio-sysfs C++ ★1 ⑂0

This is a bunch of c functions for the RPI to do GPIO communication and interrupts via sysfs

Updated on 22 Feb 2014

GPIO example code

rpi_AD7705_test_software C++ ★0 ⑂5

⑂ forked from glasgow-bio/rpi_AD7705_test_software

Test/demo program for using the AD7705 with the Raspberry Pi

Updated on 22 Feb 2014

ADC readout example code

Problem based learning needs support

Example hardware design for data acquisition

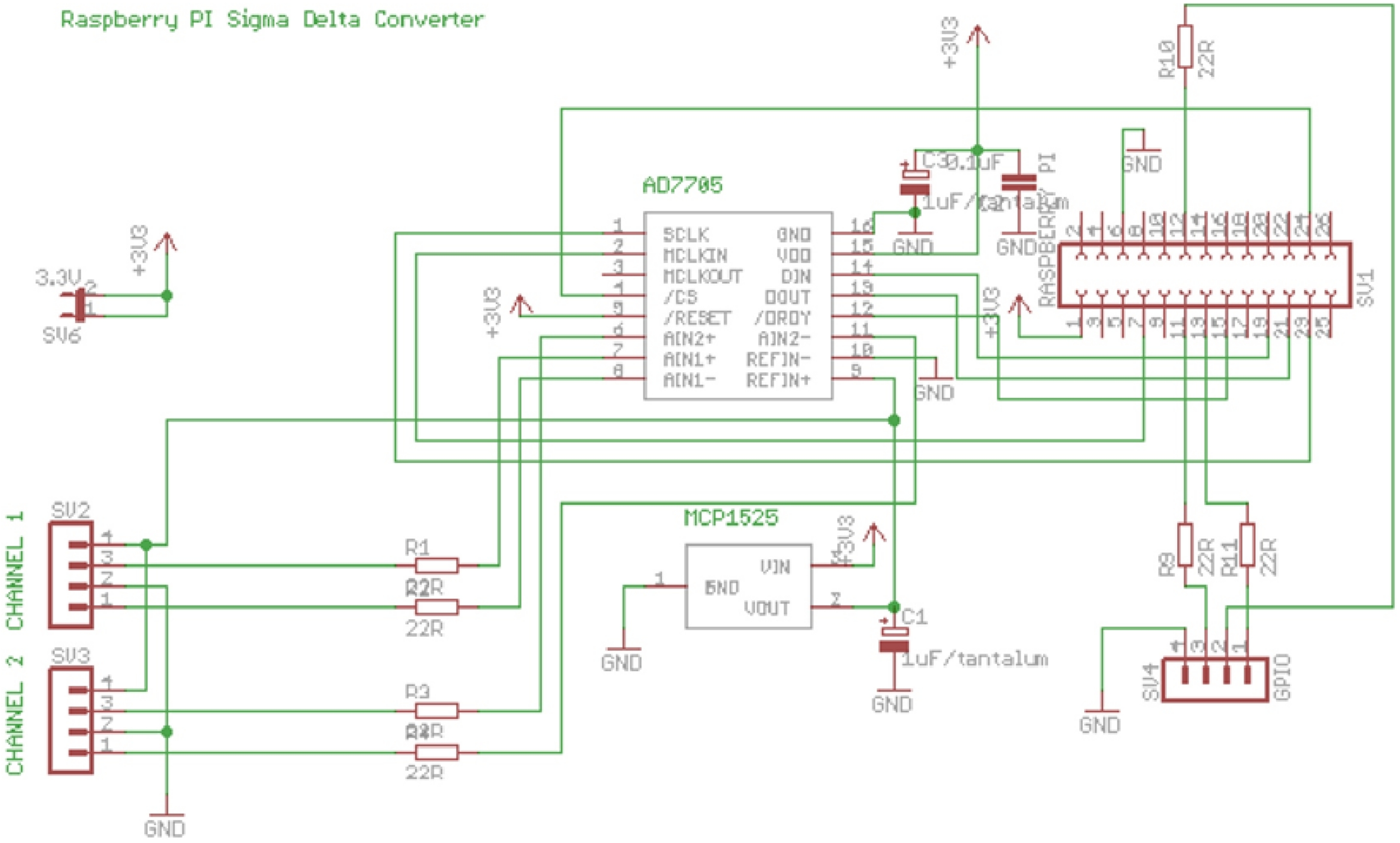
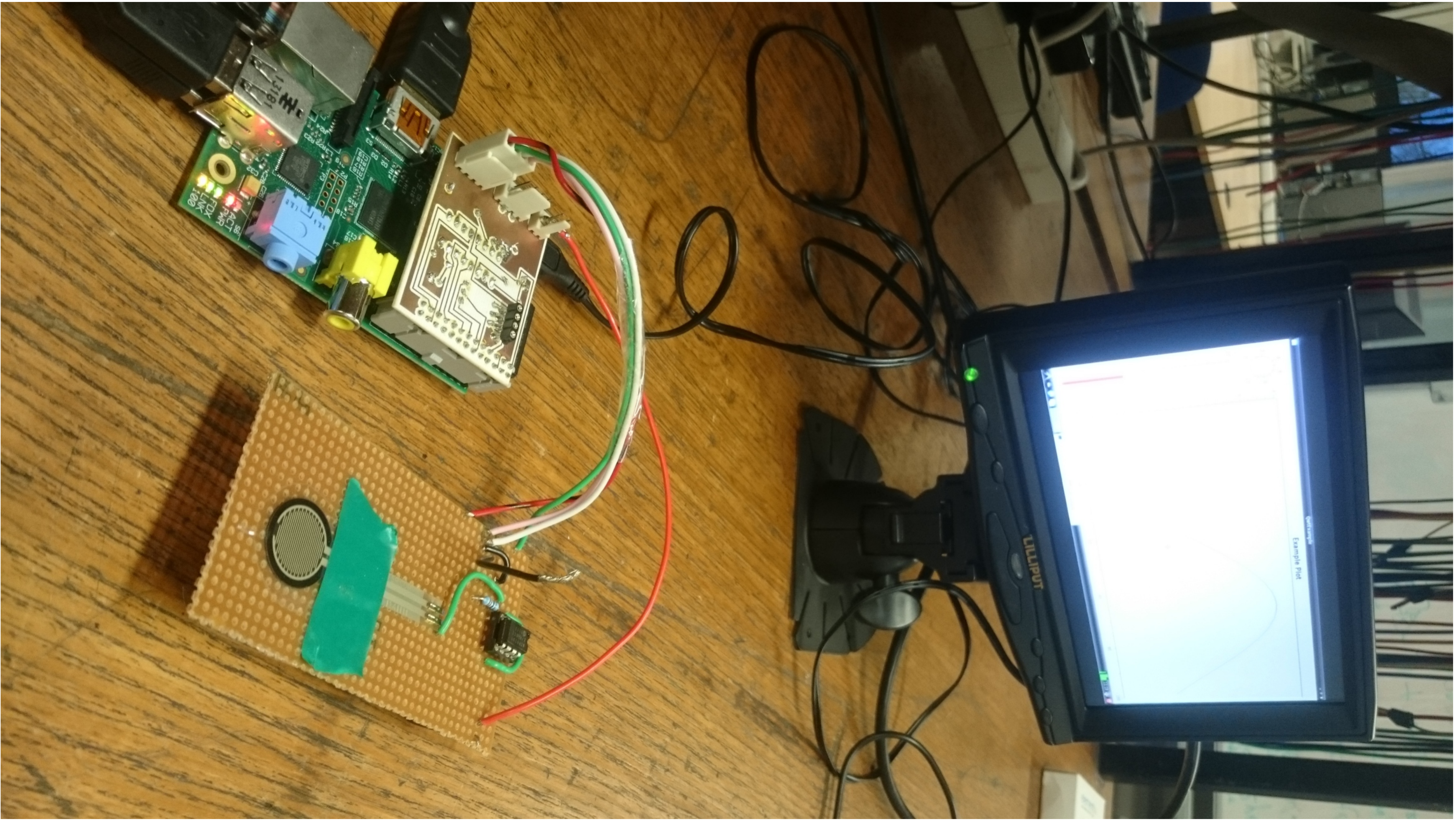


Figure 1: Raspberry Pi Sigma Delta Converter (Bernd Porr,2015)

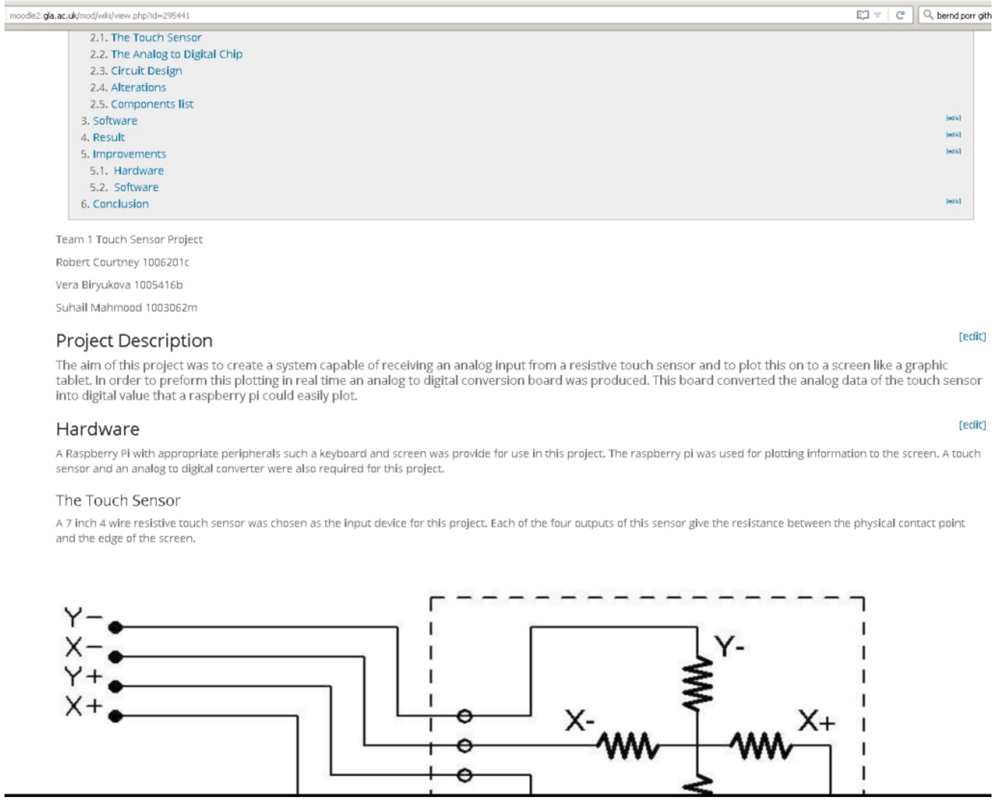
Problem based learning needs support

12 raspberry PI setups with monitor,
power supply and local network (via NAT) and electronics lab



Dissemination

WIKI



Team 1 Touch Sensor Project

Robert Courtney 1006201c

Vera Biryukova 1005416b

Suhail Mahmood 1003062m

Project Description [edit]

The aim of this project was to create a system capable of receiving an analog input from a resistive touch sensor and to plot this on to a screen like a graphic tablet. In order to preform this plotting in real time an analog to digital conversion board was produced. This board converted the analog data of the touch sensor into digital value that a raspberry pi could easily plot.

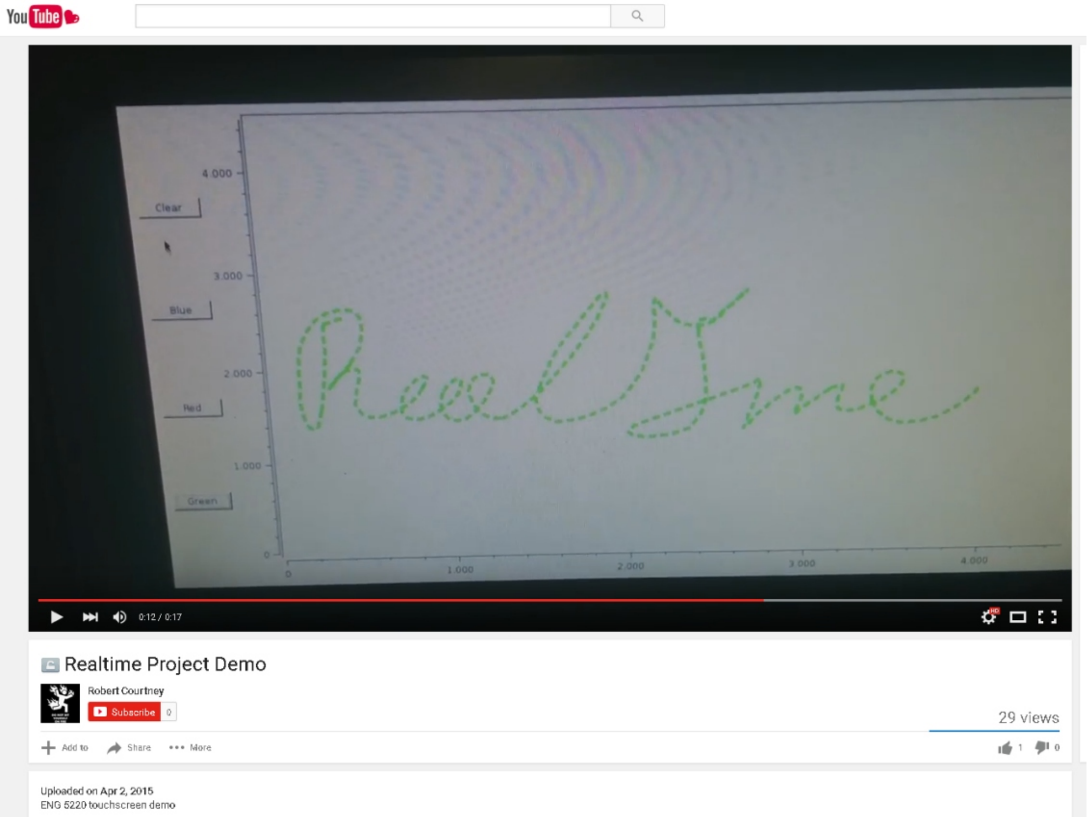
Hardware [edit]

A Raspberry Pi with appropriate peripherals such a keyboard and screen was provide for use in this project. The raspberry pi was used for plotting information to the screen. A touch sensor and an analog to digital converter were also required for this project.

The Touch Sensor

A 7 inch 4 wire resistive touch sensor was chosen as the input device for this project. Each of the four outputs of this sensor give the resistance between the physical contact point and the edge of the screen.

YOUTUBE



Realtime Project Demo

Robert Courtney

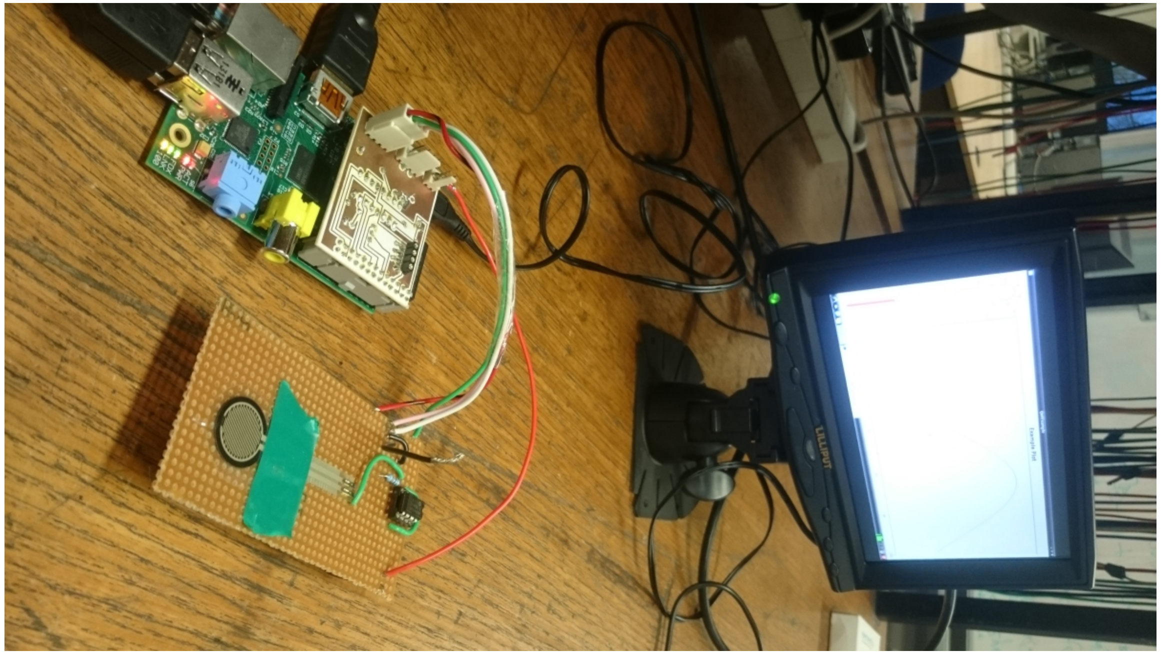
29 views

Uploaded on Apr 2, 2015
ENG 5220 touchscreen demo

Example videos

Conclusion

Real data
Realtime processing
C/C++ programming relevant for the future career
Dissemination via YouTube (to share with their friends)
Loads of fun



Bernd Porr

www.berndporr.me.uk

bernd.porr@glasgow.ac.uk

https://github.com/berndporr

http://web.eng.gla.ac.uk/rpi/