Design Team 1 Progress Report

Hardware

Initial Pin Connection

The Arduino pins were routed to PSoC Pins using the hardware components. Soldering was done in the Arduino shield to allow for the two components to communicate. Figure 1 shows the hardware configuration of the initial testing.

Figure 1

Custom PCB

The PCB was designed. The layout is set to be a side by side arrangement with the PSoC and the Arduino shield. Pins were routed to connect the two components using analog, digital and ICSP pins. The design is 15cm by 9.5 cm and there is a 3cm in between the components to account for larger shields. Holes were placed in the corners of the PCB to allow for the board to be packaged in the near future. Figure 2 shows the PCB layout.

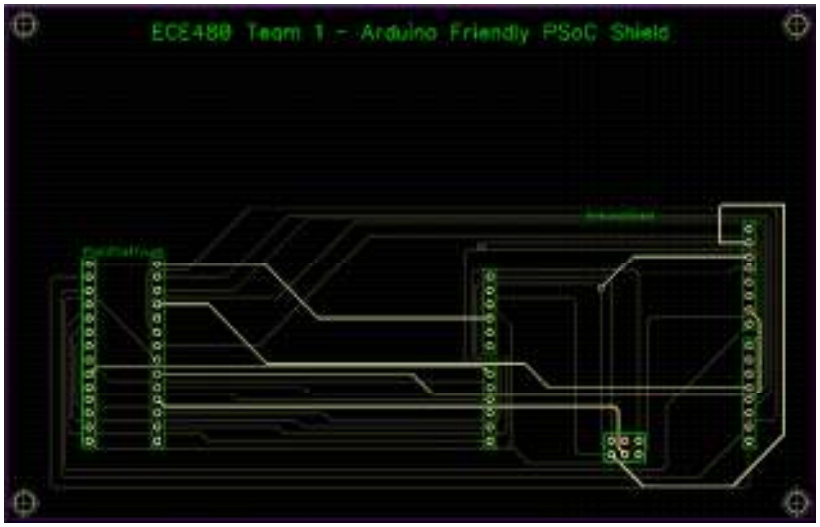

Figure 2

Software

PSoC Creator

PSoC Creator was used to program the PSoC 5. Arduino was emulated by programming input and output pins as well as built in component such as SPIs and Clocks. Components for this design were added using Creator icons. These components thus far have included an SPI module for communication with the Ethernet Shield, and a counter module to emulate Arduino functionality.

Programming

The code used for this project is the open source Arduino library along with generated code from PSoC Creator. At this point, the team has ported over the base Arduino library along with the Ethernet library from Arduino. This code has been modified at the lowest level so that it will compile. This included generating API files for the SPI module and for the various pins used. These source headers were included as C files in C++ files so it was necessary to use 'extern "C"' in order to make this code compile together. The implementations of writing to or reading from any pin and of communicating with SPI have been replaced with Cypress PSoC code. A few other library specific functions had to be changed to work with the gcc compiler and standard libraries which are included in PSoC Creator.

Testing

The next step was to develop a short main function which would demonstrate that the Arduino library had been successfully ported to PSoC. To test this, the team used a short test program which set the IP and MAC address of the Ethernet shield then the program should read these values back so they can be viewed in the debugger. This should test that the PSoC is communicating with the Ethernet Shield over SPI. Unfortunately, the data coming back to the PSoC has been all zeros so the communication is not working.

Debugging

The team is working to capture the SPI outputs from the PSoC and inputs from the Ethernet Shield using a National Instruments Data Acquisition (DAQ) device. This will help diagnose some of the errors to determine what the problem is.

If the DAQ can be used to measure the signal of the PSoC and the Arduino, the waveforms can be compared to evaluate the key differences in outputs between these two platforms. The current Labview Virtual Instrument is shown on the next page in figure 3. The current Virtual Instrument takes a sample from the DAQ every few milliseconds and sends these values to be plotted by MATLAB. Modifications to this program will be made such that measurements will be triggered by clock pulses.

The team is also working with an Arduino board to write similar code and test that. This will provide a benchmark for the demonstration code that will be later run on the PSoC board.

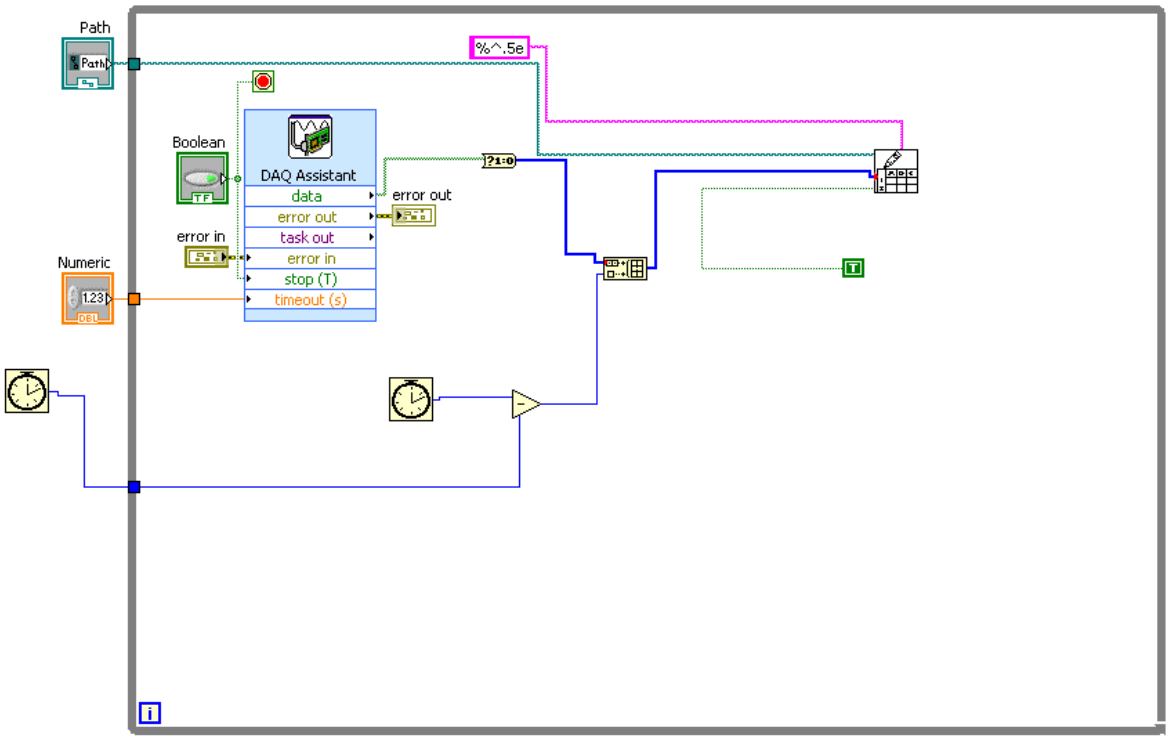

Figure 3 Labview Virtual Instrument for reading from the DAQ