CSE 4342- Embedded II

Spring 2016

Final Project

*Real Time Data Acquisition With Network Communication*



Professor: Dr. Roger Walker

Student: Scott Nidell

ID#:1000921465

May 5th, 2016

**Description:**

The final project for Embedded II uses a data acquisition system (DAQ) along with network protocols via Windows Sockets to 1.) Acquire an analog signal 2.) Convert that signal to a digital format 3.) Process that information through a filter 4.) Calculate the minimum, maximum, variance, and average of the filtered signal 5.) Send all processed information back to the client to store as a file. The system will use a various amount of commands and prompts to tell the system the sample rate, which filter to use, and where to save this information.

Multiple pieces of hardware are used to accomplish this task. First the Client and the Server PC must be connected via a network cable. Next, the server must connect to the DT9816 through a USB cable. Lastly, the DT9816 is then connected to the IDL-800 to use the function generator, discrete switches, 5volts and ground. The analog inputs are connected to the function generator and the discrete switch. Both the analog and digital ground are connected to DC ground of the IDL-800. This provides a sold reference for the DT-9816 as it may give false readings otherwise. Next the digital output is connected to the LED to verify that data is being acquired. The function generator is then verified by an Oscilloscope to ensure a 100Hz signal is being produced.

The processed data is then read from a file and plotted. This information is then verified by the lab instructors for correctness.

**Solution:**

The solution itself is nearly all software. The first step is to modularize processes into smaller chunks of operation. Unfortunately, the software provided by Data Translation is so highly coupled and so tightly scoped that it is nearly impossible to full modularize the software properly. The second step splitting any work that could possible done independently of the DT9816. These included developing the maximum, minimum, average, variance, and convolution. With these modules complete the next module is the server/client communication. Sending multiple commands and keys back a forth in order to execute the various modules previously developed. The third step is to integrate these commands with the DT9816 drivers to get and set data for sampling rate and signal processing. The fourth step is the use the minimum, maximum, variance, and average modules to process this data along with filtering it. The fifth step is to send this processed data back to the client and finally save it to a file.

To accomplish this, threads were used to process the incoming and outgoing messages of the Windows sockets. Multiple streams (ostreams, fstream, sstream) and vectors were used to store and convert the data to the correct data type to send over the socket. A windows process is used to encapsulate the data being read from the buffer as it is being stored. The data is saved as a CSV file and is plotted using MATLab.

**Circuit/Block Diagram:**

LED 1 Digital Out 0

Function Generator Analog Ch. 1

SW 0. “Start Switch” Analog Ch. 2

DT-9816

IDL-800

USB communication

Network Connection via Windows Socket

PC(Server)

PC(Client)

**Flowchart of Operation**

Client/Server Listening

CCC

Open Connection

CCC

Create Socket

CCC

Server/Client Begin

Server Side

Client Side

Prompt: Enter Sample Rate

KeyWord: SampleRate

Data: Sample Rate

Message: Sample Rate Received

Prompt: Read filter coeffcients

Data: Vector of filter Coeffcients

KeyWord: FilterWeights

****

Prompt: Save filename

Message: Sample Rate Received

Store filter weights

KeyWord: FIleSave

Data: Name of File

Save file name

Prompt: Start Signal

Message: File Name Received

KeyWord: Start

Switch asserted LED on

Waiting for Switch to assert

Initialize board settings

Convolve buffer

Buffer Full

Populate pBuffer

Initialize DAQ

Apply Transient overflow to next buffer

Keyword: Complete Data

Client Side

Server Side

Data: filtered data

Save filtered Data

Calculate

MIN/MAX/AVG/Var

Keyword: MINMAX

Data: Min Max AVG VAR

Save MIN/Max etc.

Switch asserted LED off

Keyword: Stop

Stop received

Stop DAQ

Close Connection Client/Server

CCC

System (exit)

**Screenshots:**

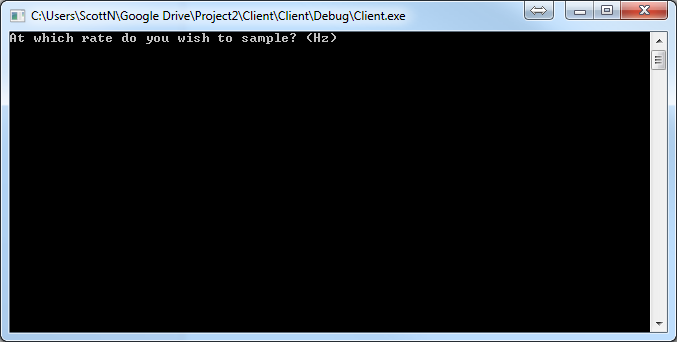
*Initial Hardware Setup*

*LED Off when the switch is not asserted (Collection Stopped)*

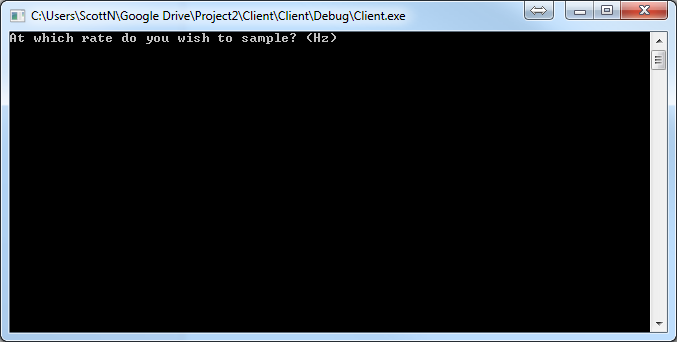
*LED on when the switch is asserted (Data Collecting)*

***Client Screenshots***

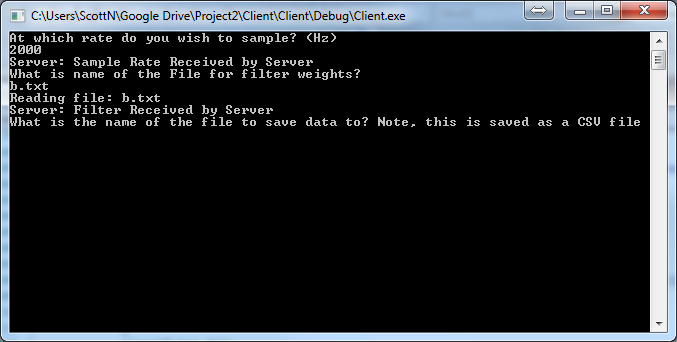
*Initial Screen prompting for Sample Rate*



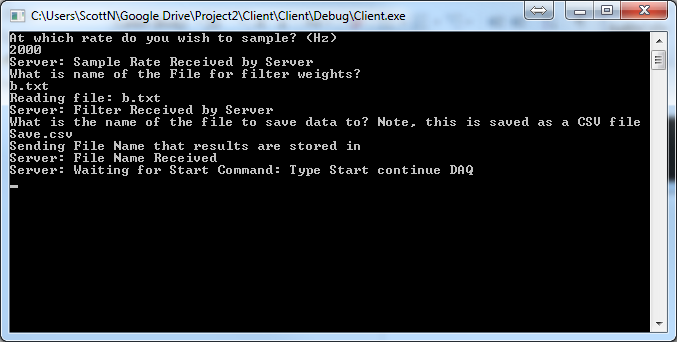
*Sample Rate Captured*



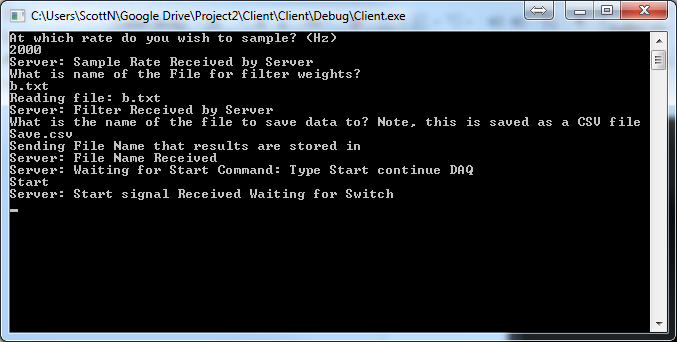
*Filter Sent*



*Save File stored*



*Start Signal Sent*

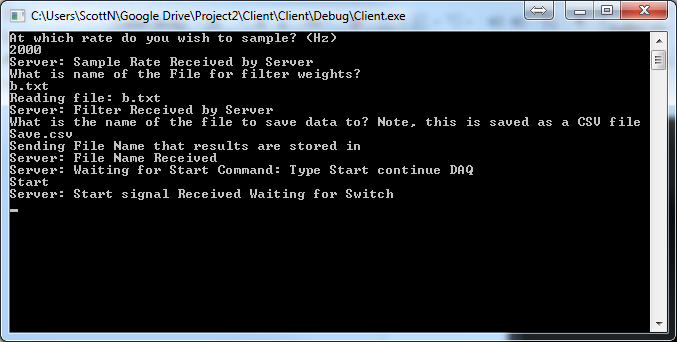


*Collecting Data*

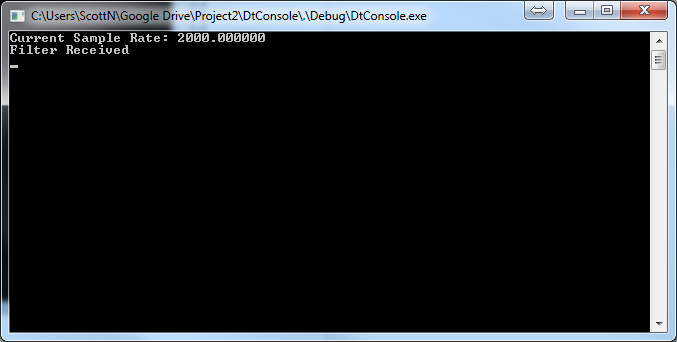
*Stop Signal Sent*

***Server Screenshots***

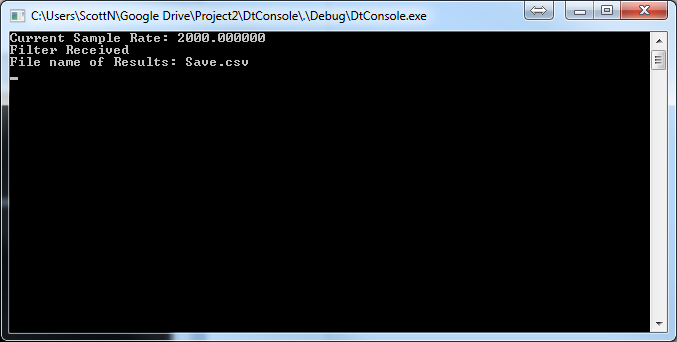
*Receiving Sample Rate*



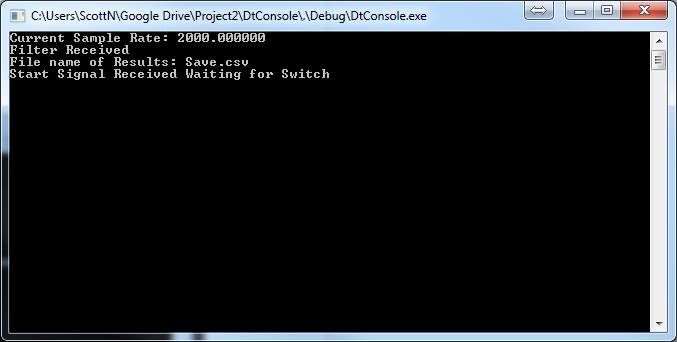
*Filter Received*



*Filename received*



*Start Signal Received*



*Collecting Data*

*Stop Signal Received*

**Comments:**