STM32 Oscilloscope

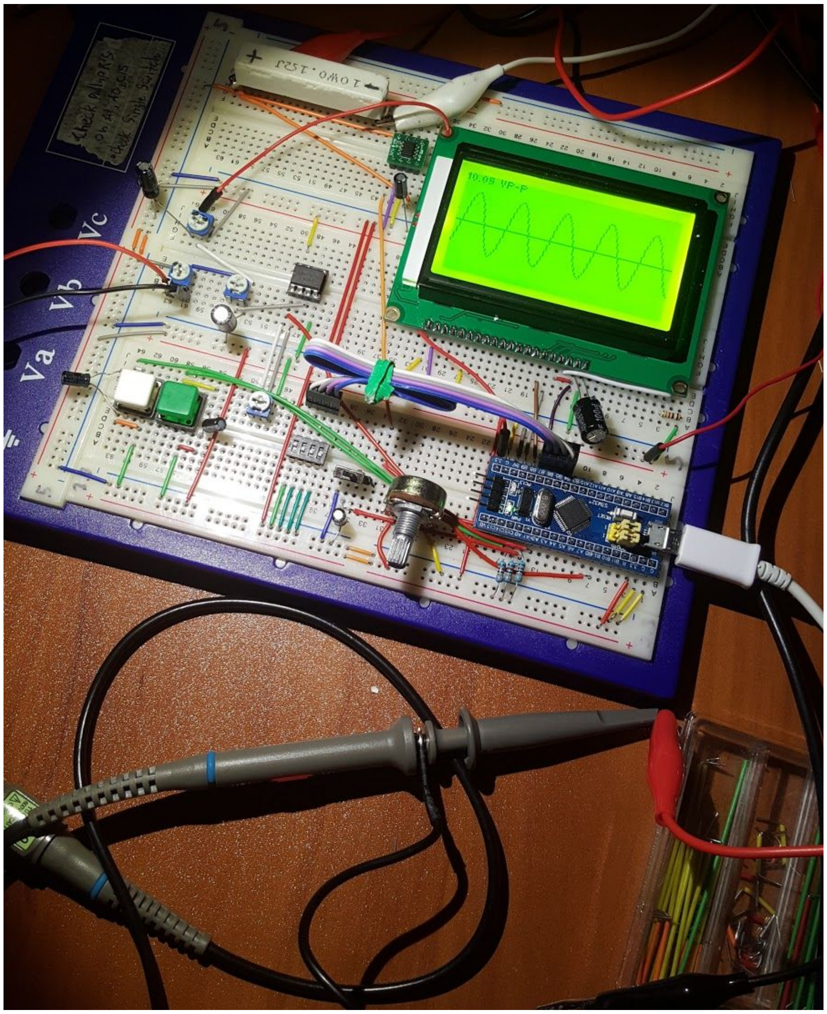
CS122A: Fall 2017

Hector Jimenez

Table of Contents

Introduction

This project is a basic function oscilloscope. Functionality is similar to an analog oscilloscope with added data processing features from a more modern digital storage oscilloscope. It features standard BNC connector oscilloscope probes and has a maximum input range of ±16.5V. Maximum input frequency before significant attenuation occurs is approximately 15kHZ. The device can also measure current and plot the input as a function of time. The device allows the user to change the voltage/current and time scales of the captured waveforms, and it allows the user to pause a waveform and scroll through it, displaying the value of voltage/current at the currently selected point.

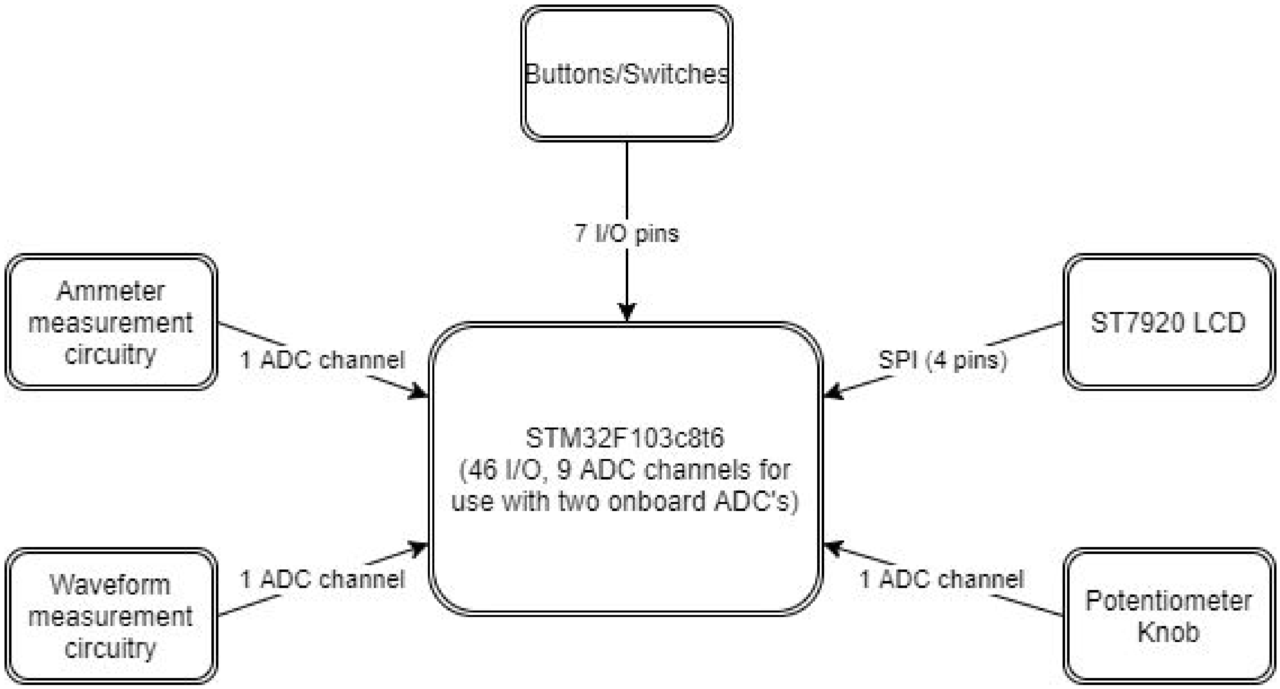


Hardware

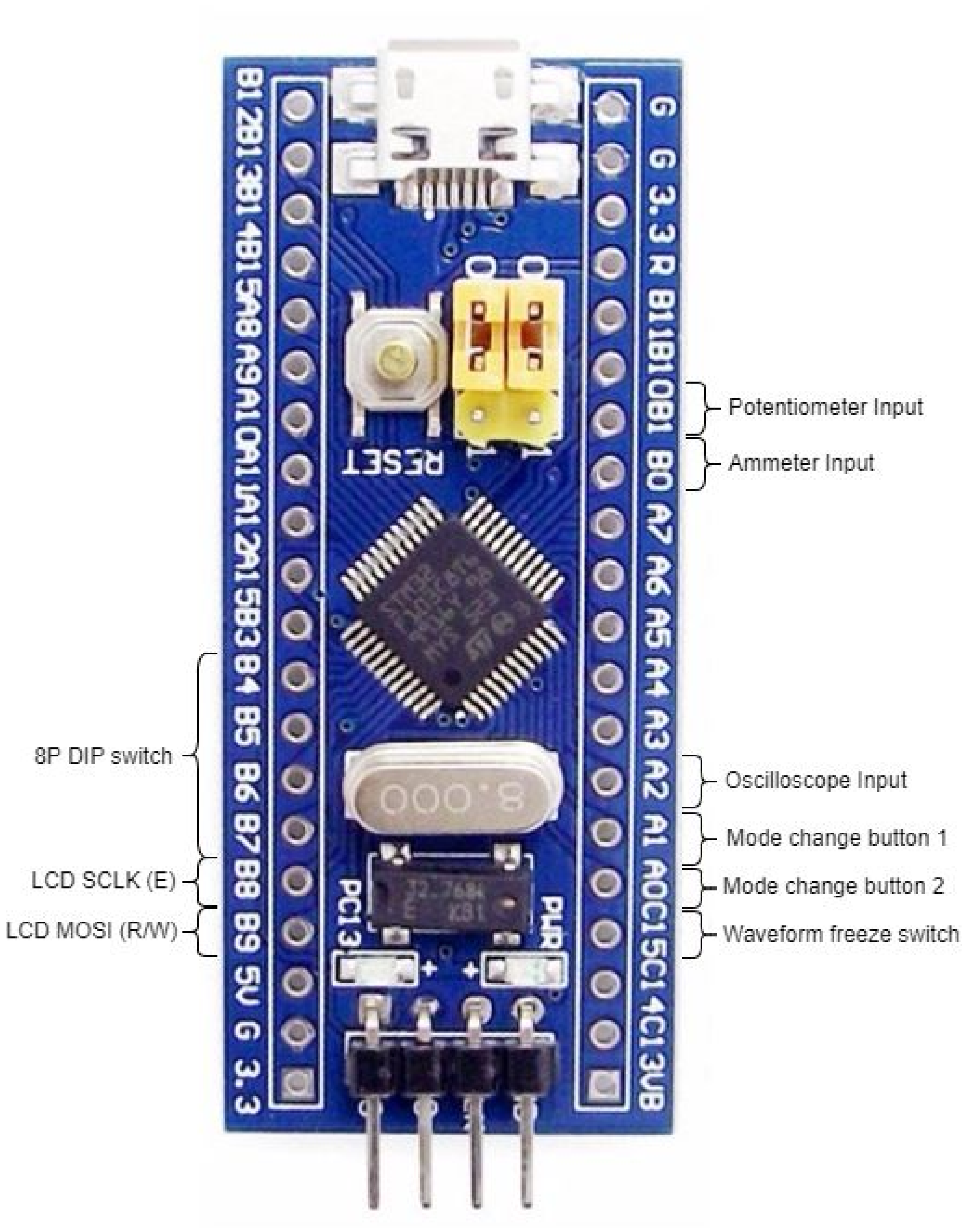
Parts List

- **STM32f103c8t6 development board**
- **ST7920 128*64 LCD**
- Potentiometers
- Button switches
- DIP switches
- Switch
- **Texas Instruments LM358 Operational amplifier**
- **Capacitors**
- Resistors
- **Analog Devices AD8210 Current shunt monitor**
- **BNC oscilloscope probes**

Block Diagram

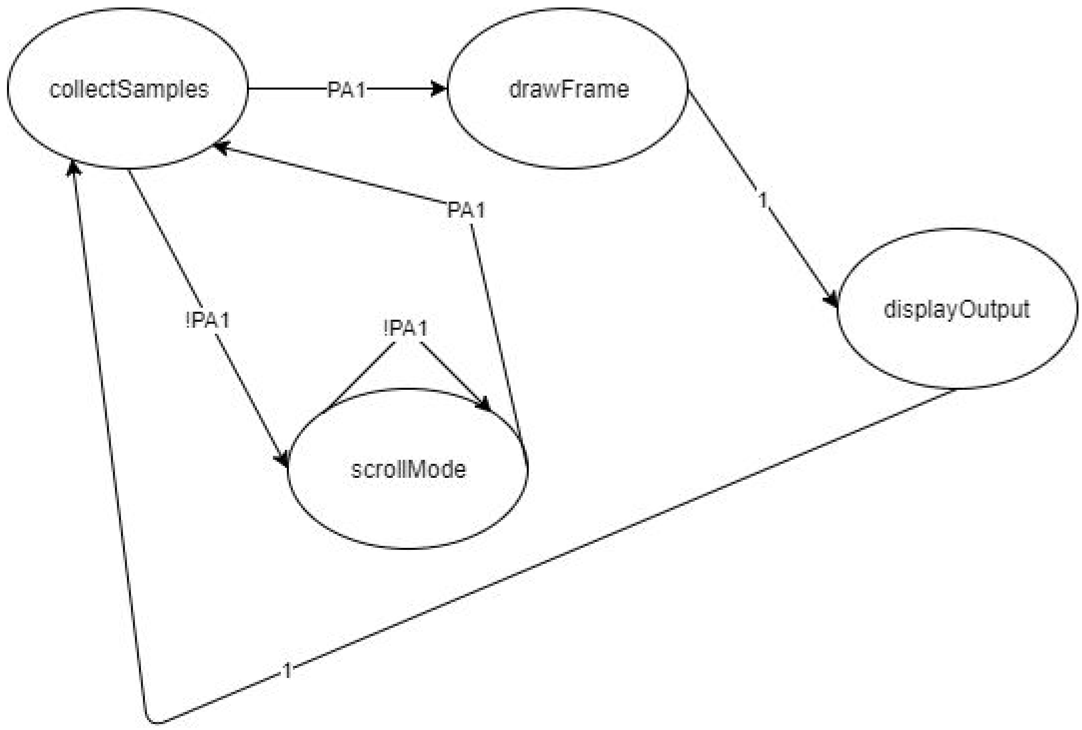


Pinout (For each microcontroller/processor)



Software

The software designed for this project was implemented using the PES standard. The overall design as a task diagram is included below.



The first task in my machine captures data points to be plotted, and analyzes the data as it is received.
The second task generates the frame of data to be output to the LCD.
The third task pauses the waveform on the screen and enters a mode that allows the user to view data from the waveform using an on screen cursor.
The fourth task outputs the frame the was rendered by the above tasks by sending the data to the display.

Implementation Reflection

I am proud of the final state of my project. I created a device with real world applications that works as intended for an affordable BOM cost.

Milestone

- I was checked off for my milestone at 70-80 milestone. I believe that I was on track as I intended, as I had planned out stages of development to ensure I met the deadline.

Completed components

I completed the majority of components from all parts of my project. I was unable to implement a few features from the 80-90 and 90-100 parts due to components not functioning as intended.

Incomplete components

I was unable to use a digital potentiometer for input attenuation due to the relatively low resolution of adjustment the unit allowed. The device also distorted high frequency AC input signals, making it unsuitable for this project. I was unable to implement resistance measurement functionality due to the imprecision of the circuit I intended to use,

Youtube Links

- Short video: https://youtu.be/MNLEjgxComk
- Longer video: https://youtu.be/S1zfvUG_41o

Testing

Analog input conditioning circuitry:
Circuitry was designed using knowledge of concepts learned in electrical engineering classes. Circuitry was simulated in LTspice using components with similar characteristics of those used in this project. Upon successful verification of the designs in this simulation software, further verification was done by implementing the actual circuits and verifying its performance using a waveform generator for input and an oscilloscope to test the output. See appendix for images of results from testing.

Input debouncing:
Buttons and switches used for input caused issues with multiple inputs due to switch bounce. As inputs are handled with interrupts, and software debouncing would have created timing issues, additional hardware was used to address debouncing. This consisted of using a capacitors and resistors of known values to create RC discharging circuits for all inputs that triggered interrupts. The RC circuits ensured no bounce occurred by their performance was verified using an oscilloscope.

LCD:
SPI SCLK and MOSI from the microcontroller was observed using a logic analyzer. Basic commands were observed to ensure data was being correctly sent.

Finished project:
Project was tested by several friends in various majors who have experience using oscilloscopes. They were able to use it to view signals made with signal generators in the electrical engineering labs and with signals made with my PC-based signal generator. No bugs that affected functionality were encountered by the testers.

Known Bugs

- On occasion while resuming a paused waveform, a few random characters will appear on the screen. They last for a split second. I believe there may be noise on the LCD MOSI line as the the SCLK pin resumes high frequency operation

Resume/Curriculum Vitae (CV)

This project taught me much about processing analog signals with an ADC. I learned to take into account various factors that cause noise to signals and affect overall precision. I learned to deal with the innate imprecision of various analog components such as resistors, by instead using tunable components like potentiometers.

Future work

For a more professional presentation of internals, a custom PCB could be modeled in Altium or EAGLE PCB that would place all components neatly across the board. A case could be modeled using CAD software and 3D printed to allow for the screen and button to remain visible and usable outside the case, while covering and protecting the PCB that has the internal components mounted on it.

References

I used a set of libraries available to download on the website listed below that enabled the use of the Arduino IDE with the STM32 microcontroller I used. The library used for the LCD display was also found on this site.
http://www.stm32duino.com/

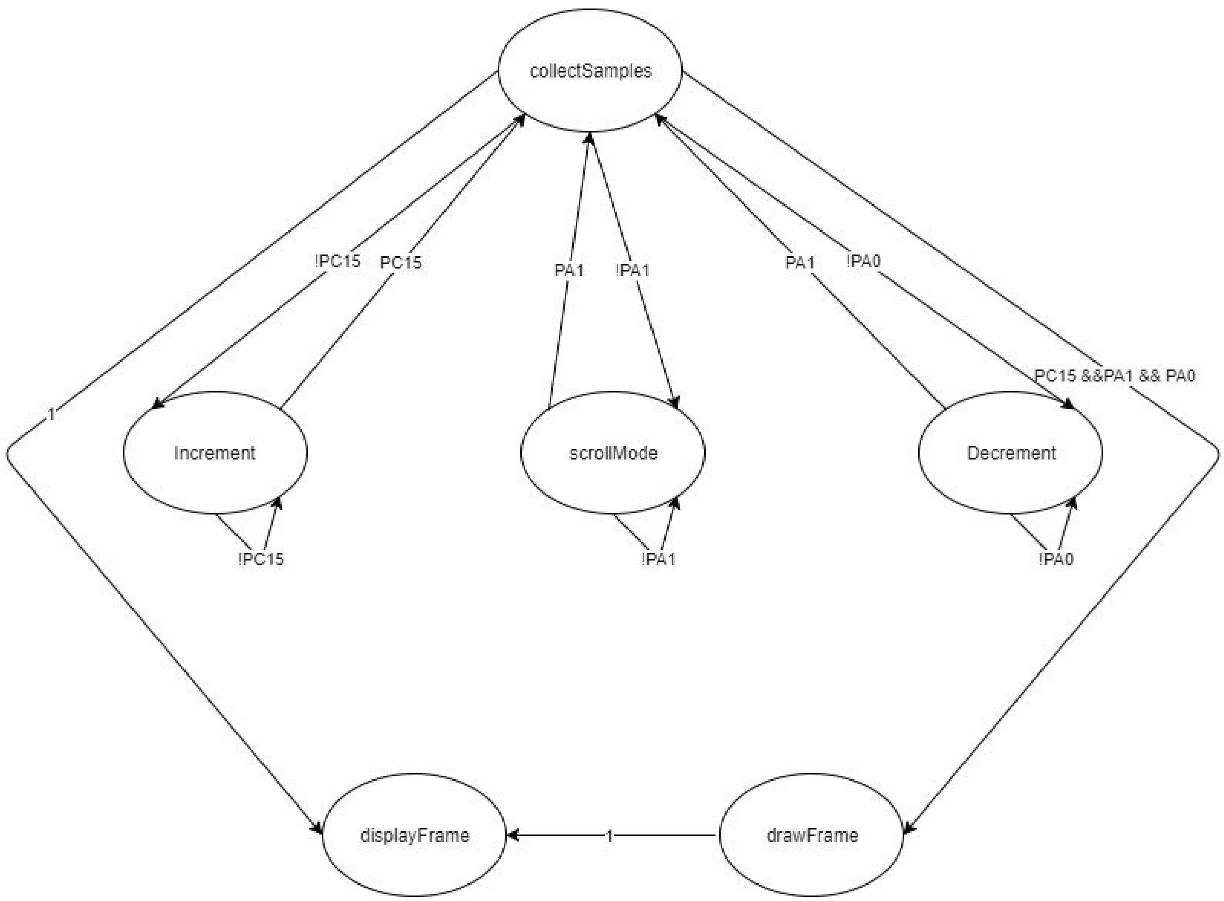
Datasheets used:

http://www.ti.com/lit/ds/symlink/lm158-n.pdf - For LM358 operational amplifier
http://www.analog.com/media/en/technical-documentation/data-sheets/AD8210.pdf - For AD8210 current shunt monitor
http://www.lcd-module.de/eng/pdf/zubehoer/st7920_chinese.pdf - For ST7920 LCD controller

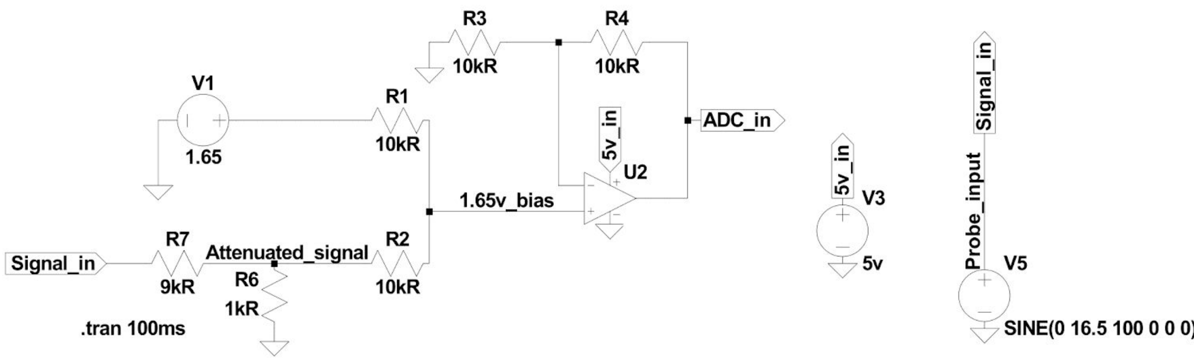
Github Repo

https://github.com/hjime008/STM32-Oscilloscope/tree/master/STM32Oscilloscope

Appendix

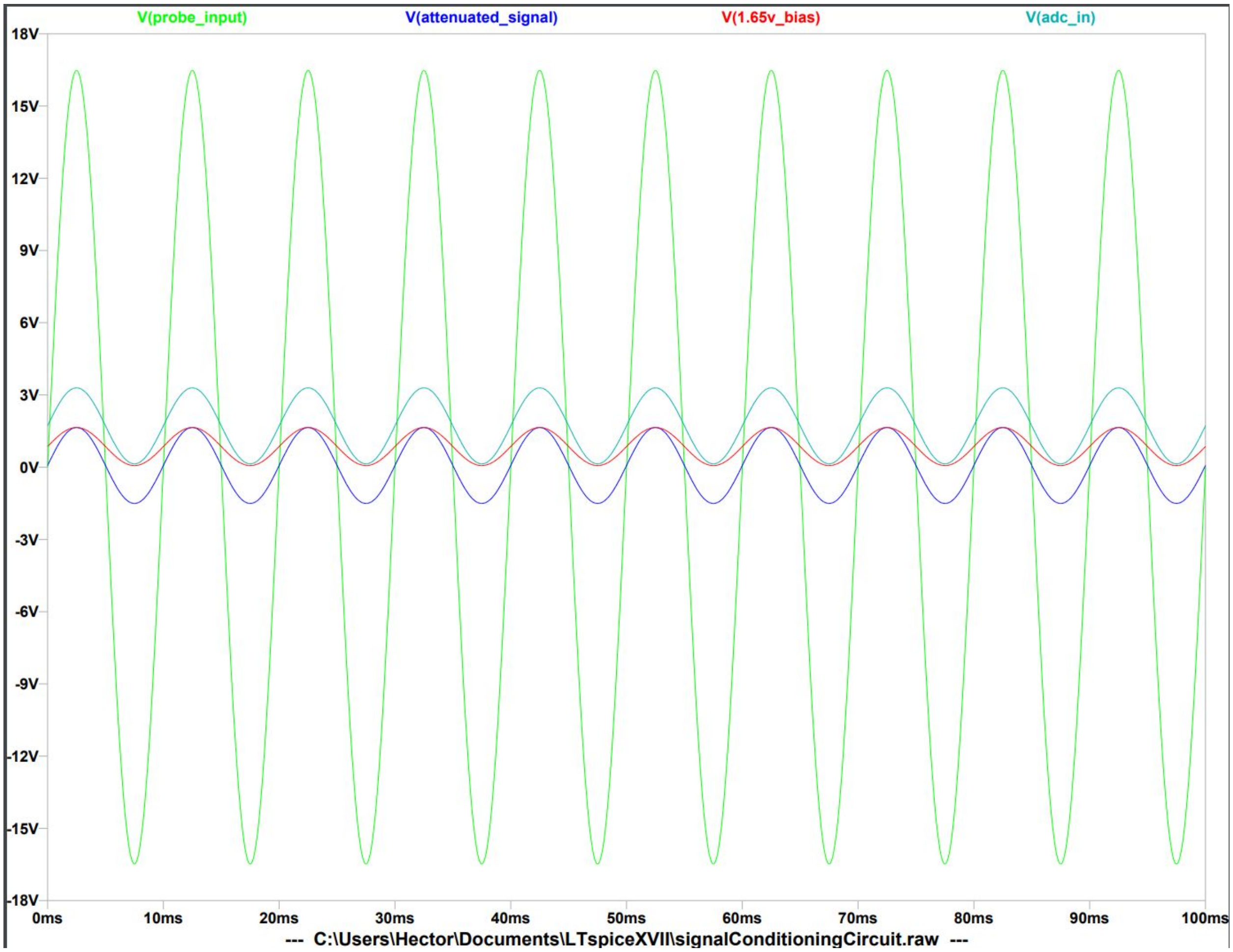


(State machine design for this project)
***Note: State Machine is asynchronous to allow fastest ADC sampling speed. States change automatically or change using interrupts.**



--- C:\Users\Hector\Documents\LTspiceXVII\signalConditioningCircuit.asc ---

(Circuit diagram of implementation of signal conditioning circuitry for voltage signals)



(Output waveforms of simulated circuit performance at various nodes of circuit)
***Note: V(probe_input) is a raw signal from -16.5v to +16.5v. V(adc_in) is the final signal output that the microcontroller ADC will read in.**