CALIFORNIA STATE UNIVERSITY

1949

LONG BEACH

CECS 447 Fall 2019 Project 1

Tone Generator

Steven Phan ID #: 014358810

Rosswell Tiongco ID #: 016091762

9/30/19

Table of Contents

Table of Figures

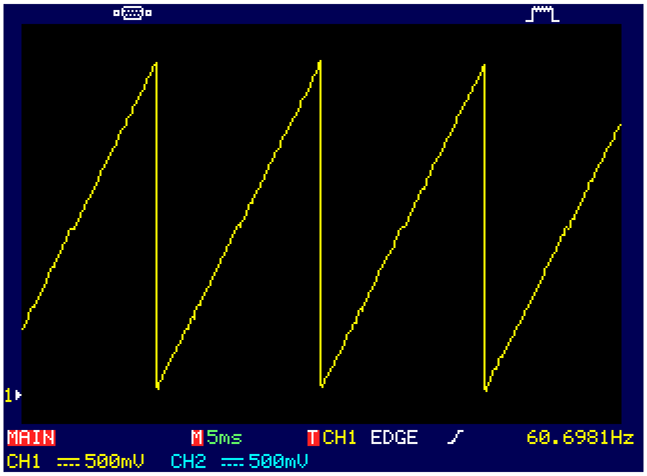
Introduction

The objective of this lab is to demonstrate knowledge of using a Digital to Analog Converter (DAC), GPIO, timers, and interrupts by building a tone generator. This was visually proven by displaying the output of the DAC into an oscilloscope. Additionally, the tone generator was connected to an (LM386) audio amplifier to power a 5 watt speaker.

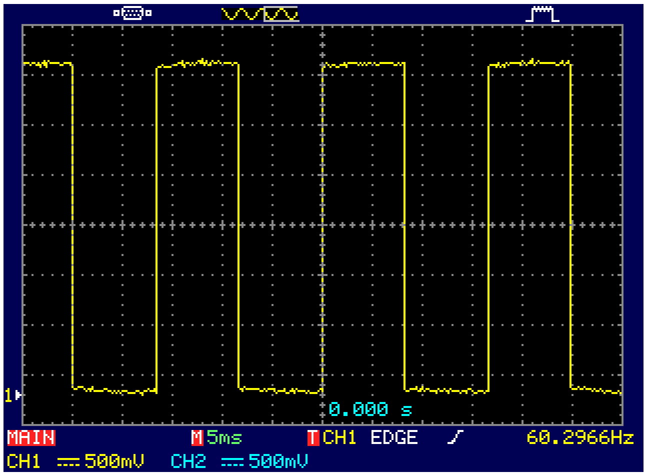
Operation

Upon reset of the microcontroller, actuating the built in push button will allow the microcontroller's built in led to cycle through 5 different modes. The different modes in order are sawtooth, triangle, sine, and square waves oscillating at 60 hz. The final mode was another sine wave cycling oscillating at a frequency based on the potentiometer. The potentiometer can then change the frequency by increasing or decreasing the delay between each change in amplitude. Attached below are waveforms from an oscilloscope generated by the different waveforms.

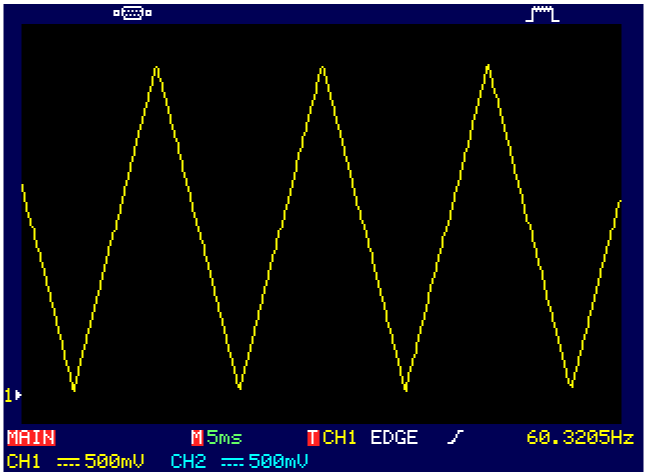
60hz Sawtooth Waveform



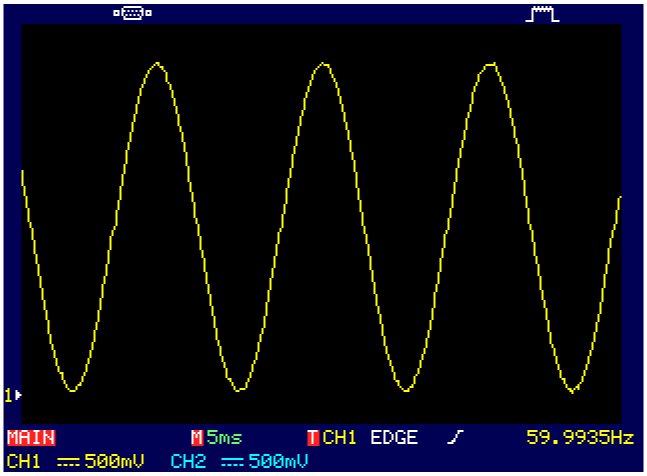
60hz Square Waveform



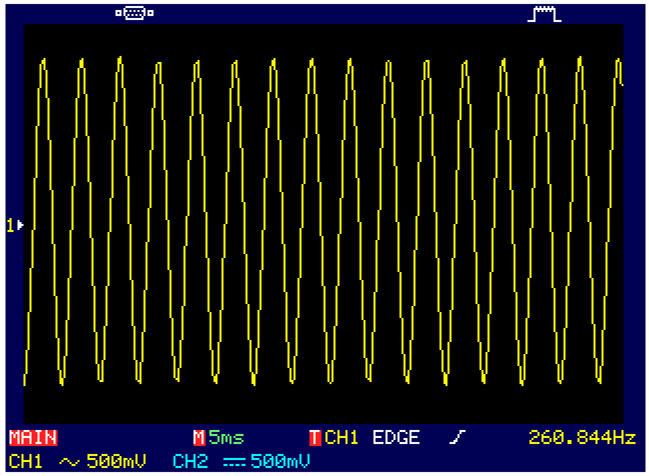
60hz Triangle Waveform



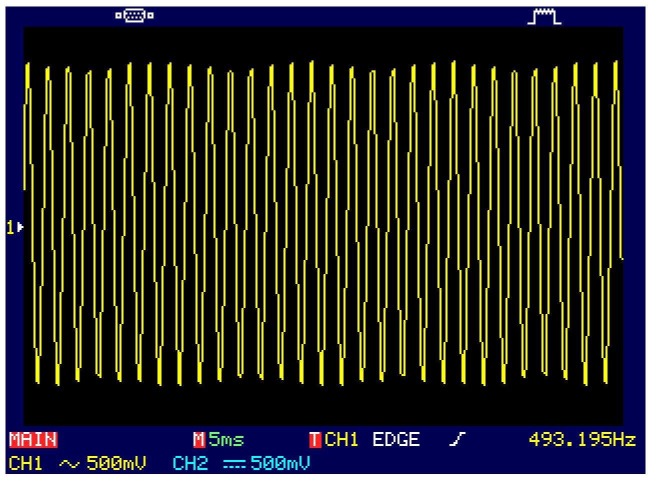
60hz Sine Waveform



262 hz Sine Waveform

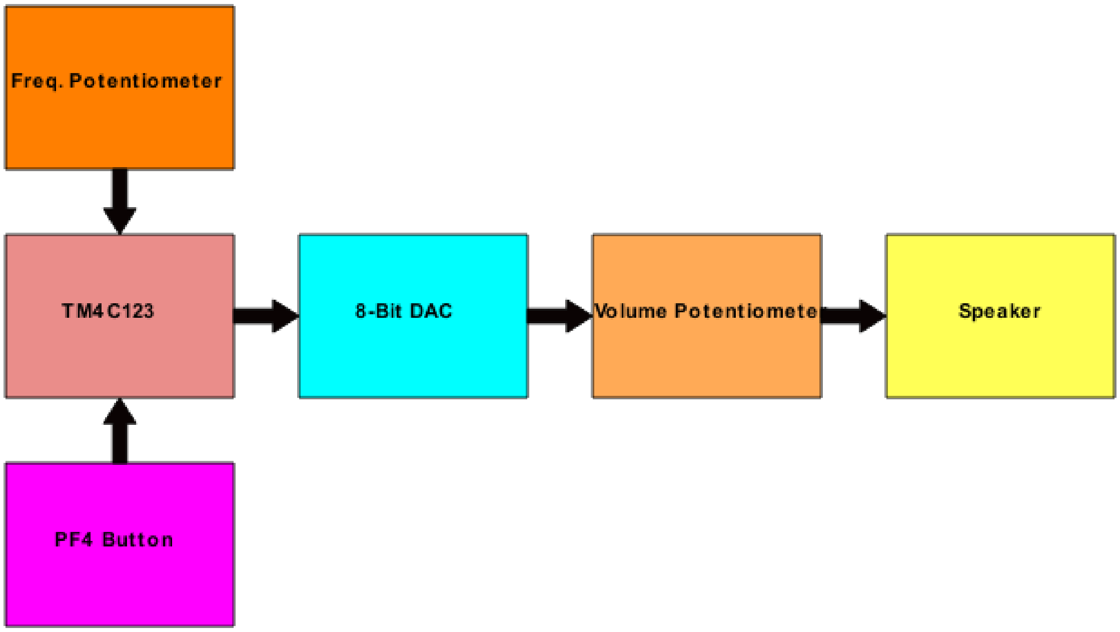


494 hz Sine Waveform

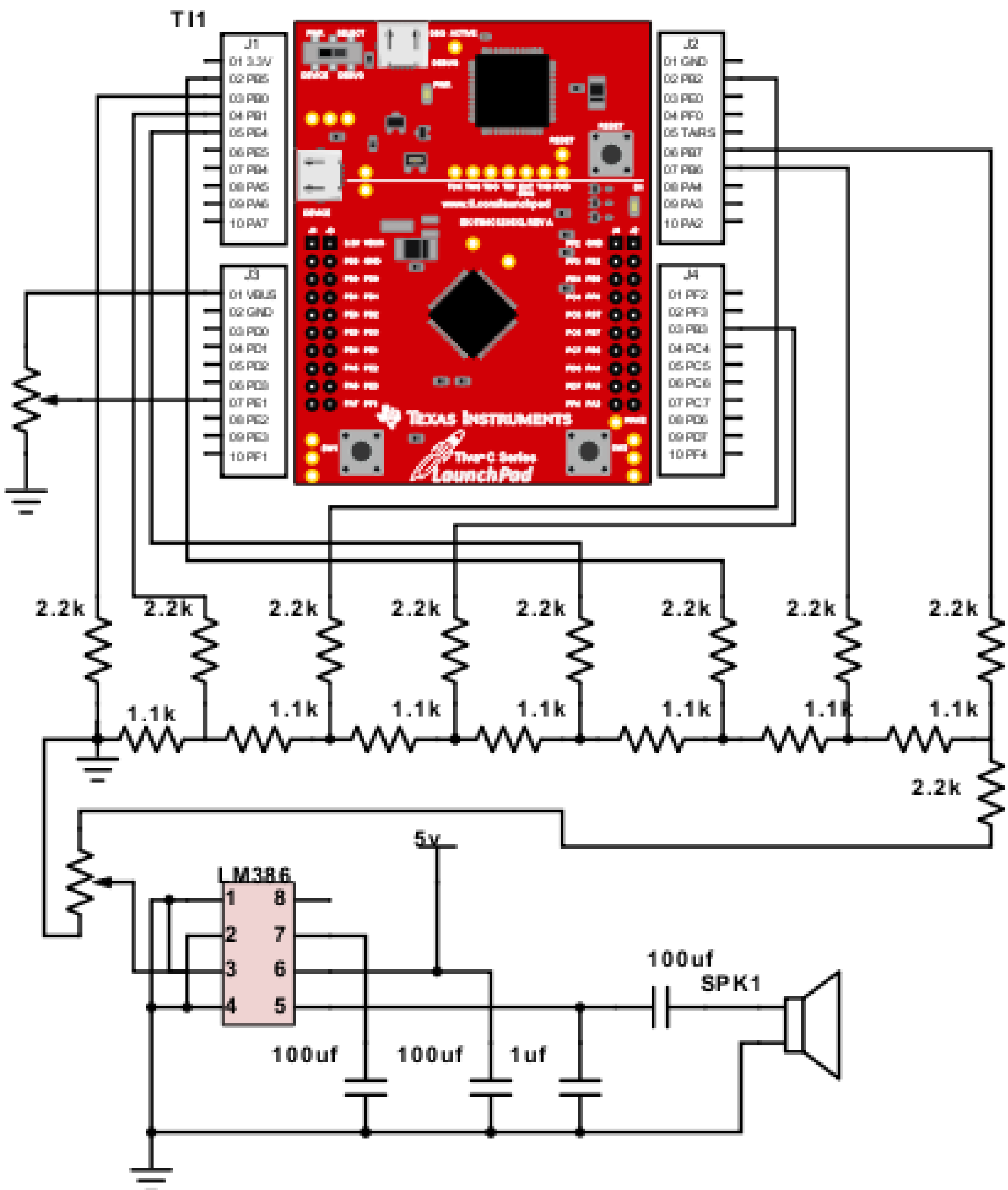


Hardware

Hardware block diagram



Schematic


TI1
J1
01 3.3V
02 PB5
03 PB0
04 PB1
05 PE4
06 PE5
07 PB4
08 PA5
09 PA6
10 PA7
J2
01 GND
02 PB2
03 PE0
04 PF0
05 TARS
06 PB7
07 PB6
08 PA4
09 PA3
10 PA2
J3
01 VBUS
02 GND
03 PD0
04 PD1
05 PD2
06 PD3
07 PE1
08 PE2
09 PE3
10 PF1
J4
01 PF2
02 PF3
03 PB3
04 PC4
05 PC5
06 PC6
07 PC7
08 PD6
09 PD7
10 PF4
TEXAS INSTRUMENTS
LaunchPad
2.2k
2.2k
2.2k
2.2k
2.2k
2.2k
2.2k
2.2k
1.1k
1.1k
1.1k
1.1k
1.1k
1.1k
1.1k
2.2k
5v
LM386
1
2
3
4
8
7
6
5
100uf
SPK1
100uf
100uf
1uf

List of Components:

- (1x) Tm4c123gh Development board
- (2x) 10k Potentiometers
- (1x) 5 watt speaker
- (1x) breadboard
- (9x) 2.2k resistor
- (7x) 1.1k resistor

Hardware justification?

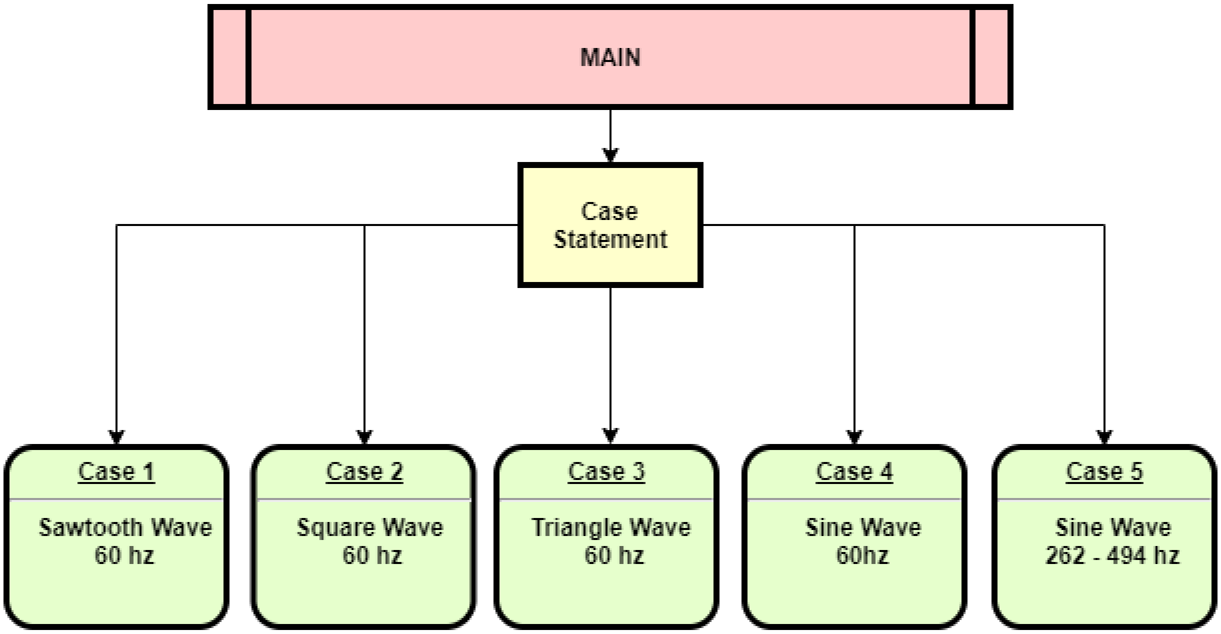
There were a variety of components that we had to use for our first project. Since the goal of the assignment was to understand how to use a digital to analog converters, we built an R2R DAC. This setup allowed us to have a higher precision of our analog signal. Using the onboard push button we're able to change the frequency of a sine wave from 262~494 Hz. We also used two separate potentiometers as well. One potentiometer would be connected from the output of the DAC to the input of our LM386 audio amplifier. This gave us a capability to control the amplitude of the sine wave, which in turn controlled the volume of the 5 watts speaker. The other potentiometer is connected as an ADC read to PE1. We would use this potentiometer to change the delay of the sine wave. This allowed us to have variable frequency control.

Software

Software Approach

The general concept of the software is fairly straightforward, especially in the while loop. It is pretty much just a case statement. The main difficulty and pieces of code where one is likely to mess up is in the initialization functions, generating each waveform, and determining how much delay should be in between each step of the waveform.

Software Flow Diagram



Conclusion

This project expanded our knowledge of being able to convert digital signals to analog and vice versa. While working on the assignment together, we experience a number of problems. One of these problems was not having enough 1% tolerance resistors. As a result we had to put a 1.1k and 100 ohm resistor in series. Another problem we encountered was with how sampling of our potentiometer would not be as consistent as we would have liked. This would affect our variable frequency output. Finally, our largest problem was not being able to finish the extra credit. We were on the right track, however our fault lies in that we were trying to read it and compute the frequency in real time instead of storing the values.