# Introduction

The project is to create an accurate, low distortion signal generator for sine, square and triangle waveforms, alongside a frequency meter for the same. This will be implemented using an ARM processor carrier board from ST Microelectronics (STM32F407VGT6) fitted with various peripherals.

The project will be split into two parts, software and hardware. From the combination of these two parts, the maximum possible frequency range for the waveforms needs to be achieved, and again for the frequency meter using the lowest power achievable. The input signal to the frequency meter on the board is provided by the hardware and displays the frequency on the LCD display. Further work on the product may include using the LCD screen, I2C connector, USB connector, and Direct Digital Synthesis (DDS) using a AD9850 chip to generate accurate pulse and arbitrary functions.

Environment consideration…

Input protection, output impedance, power supplies

Power consumption – minimisation of power

Component list, housing, form factor

# Specification

## Level 2 Requirements

* Generate a sine wave from 0.01Hz to ?HZ with a maximum amplitude of +/- 12V.
* Generate a square wave from 0.01Hz to ?Hz with a maximum amplitude of +/- 12V.
* Generate a triangle wave from 0.01Hz to ?Hz with a maximum amplitude of +/- 12V.
* Measure the frequency of sine, square and triangle waves from 0.01Hz to ?Hz with a minimum signal of 0.1V rms.
* Control the amplitude of sine, square and triangle waves by sending control signals to a digital potentiometer, which will control the gain of the amplifier circuit.
* Create a pulse generator from ?Hz to ?Hz with a maximum amplitude of +12V.
* Measure the duty cycle for digital waveforms.
* Vary the duty cycle for digital waveforms.
* Amplitude modulate an input signal by using a sine wave to send control signals to a digital potentiometer, which will control the gain of the amplifier circuit.
* Frequency modulate an input signal by feeding it into the ADC and using the resulting values to vary the frequency of the output signal, based on a sine wave with a fixed frequency.

## Further Requirements

* Generate square waves using DDS on the AD9850 chip.
* Generate sine waves using DDS on the AD9850 chip.
* Create a random noise generator using the internal PR sequence generator.

# Software Design/Implementation

## Tasks so Far

After following a few brief example programs provided as part of lab script 1 [REF ME], we started the wave generation section.

To start with we generated a square wave with a 50:50 duty cycle, by outputting a high for 5 milliseconds followed by a low for 5 milliseconds on GPIO pin PA0. This gave a nice clean square wave which we viewed on the oscilloscope, and measured the frequency and period. The theoretical period was 10 milliseconds giving a 100Hz frequency, however when measured on the oscilloscope the period was 32 milliseconds and the frequency at 31.25Hz. Consequently this method of setting the frequency is very poor in terms of accuracy, and therefore unsuitable given our specification.

## Tasks Still to Complete

# Hardware Design/Implementation

## Tasks so Far

The hardware team so far has been working together to investigate a variety of circuits for generation of the square, triangle and sine waves. Their main focus has been trying to generate a 1MHz signal from each circuit, with 100kHz being the absolute minimum requirement to achieve the level one design. They have managed to generate all three types of signal with varying degrees of success in terms of the highest frequency obtained.

**Square-waves**

Three circuits have been utilised for square wave generation so far: a Schmitt trigger configuration and two astable multivibrator circuits as described on the Intersil datasheet . Figure 1 shows the Schmitt trigger circuit, with a LM318 op-amp; a TL071 was also used though it has a lower slew rate than the LM318 meaning it would be less likely to achieve the 1MHz goal. Figure 2 below shows the output from this circuit set up as given in the lab script to give a 400Hz frequency output. As can been seen this configuration gives a very clear square wave output. The next task was to work out whether to change the capacitor or the resistor to vary the frequency as both components effect this. Figures 3 and 4 show the generation of a 100Hz wave, with figure 3 having changed the capacitor and figure 4 using a different resistor to that in the initial circuit. It is quite clear that at low frequencies it doesn’t matter which component is changed as both give the same result. It was then decided to try a frequency of 10kHz again varying just one of the components. Again both produced the same results however, as can be seen in Figures 5 and 6, the slew rate of the op-amp became a limiting factor and thus the rise and fall times for the square wave have increased significantly. This circuit is clearly unsuitable for use in the function generator as it is unable to produce precise squarewaves at even moderately high frequencies.

The next circuit investigated utilised the 7555 timer IC as part of a multivibrator circuit, set up following figure 2B from the datasheet. This configuration allows for the duty cycle to be varied as well as the frequency, however it was quickly found that trying to adjust one would also cause the other to change and that it would be difficult to achieve a fixed 50/50 duty cycle while also being able to change the frequency. Fortunately the datasheet provided another circuit (Figure 2A from ) that had a fixed 50/50 duty cycle, this circuit has two outputs, both of which have been considered. Figure 7 shows both of these outputs for a 1kHz squarewave, each one giving a very clean signal. The next step was to attempt to output a 100kHz signal, this is show in figure 8. Here the outputs have become much noisier and the alternate output’s rise time is affected drastically by the capacitor. Ultimately a 700kHz output was obtained, though the signal at this point had a noticeable rise time and was fairly noisy, the other output at this frequency was giving a sawtooth wave, figure 9 shows the scope display. The team found that a frequency much higher than this would be hard to obtain as both the resistor (100Ω) and the capacitor (100pF) were heading towards being dominated by the internal resistance and capacitance of the wiring used to create the circuit. It was also decided to check that this circuit operated correctly at very low frequencies. Figure 10 shows a 0.01Hz signal. The scope was unable to measure such low frequencies however a very clean signal was obtained meaning that this circuit could be used if the software team stuggled to generate lower frequency signals.

* Triangle wave from 7555 (including square wave)
  + 500Hz – amplitude decreases with increasing frequency
* Sine wave from Wien Bridge oscillator
  + Changed capacitor for each 10x frequency change

## Tasks Still to Complete

* Switching circuit
* Compare CMOS squarewave oscillator with 7555
* Better triangle wave generator
* Pulse generator
* Investigate other sine wave generators
* Create amplifier and dc offset stage
  + Digital potentiometer
* Comparator with and without hysteresis
* Environment consideration…
* Input protection, output impedance, power supplies
* Power consumption – minimisation of power
* Component list, housing, form factor

# Gannt Chart