Signal Processing Lab Report

Subject: Microcontroller Laboratory Exercise

Lab number and name: 05 (Signal Processing)

Name: Das Anjan Kumar

Partner Name : Mohammed Shamin Yeasher Yousha

Neptun ID: D42DQA

Lecturer(s): Tamás István Krébesz

MŰEGYETEM 1782

Task 1:

The task is to convert Celsius temperature to Fahrenheit unit. For this conversion base equation has been used.

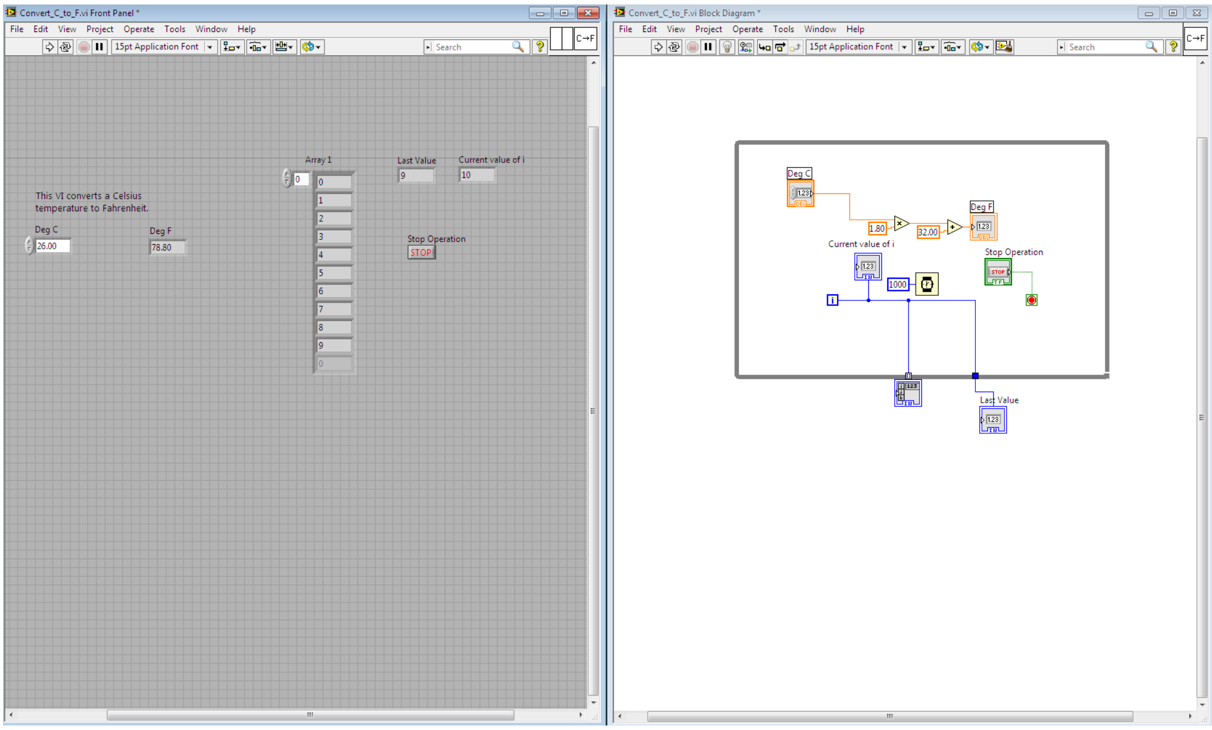
- F = 1.8 * C + 32

To solve this problem using LabView, we need to start the VI and open both block and Front panel together and start adding on the block diagram to solve the problem.

Functions used to solve this problem :

- Inputs and outputs
- Addition and Multiplication operators
- Wait timer
- While Loop
- Array

The combination is given in the following screenshot as well:



As we can see, in the above picture , task 1.1 and 1.2 is done here. For task 1.1, we are converting Celcius to Fahrenheit. To do that, we take the input value and then multiply it with1.80, and then finally add 32 to the last value.

Also, we can see the demonstration of taks 1.2 to use Highlight Execution function through last value and array 1.

As following task 1.1, we can also do task 1.3 to build a Fahrenheit-Celsius converter, for that we can subtract 32 from the initial value in Fahrenheit and then divide by 1.8 to get the value in Celsius.

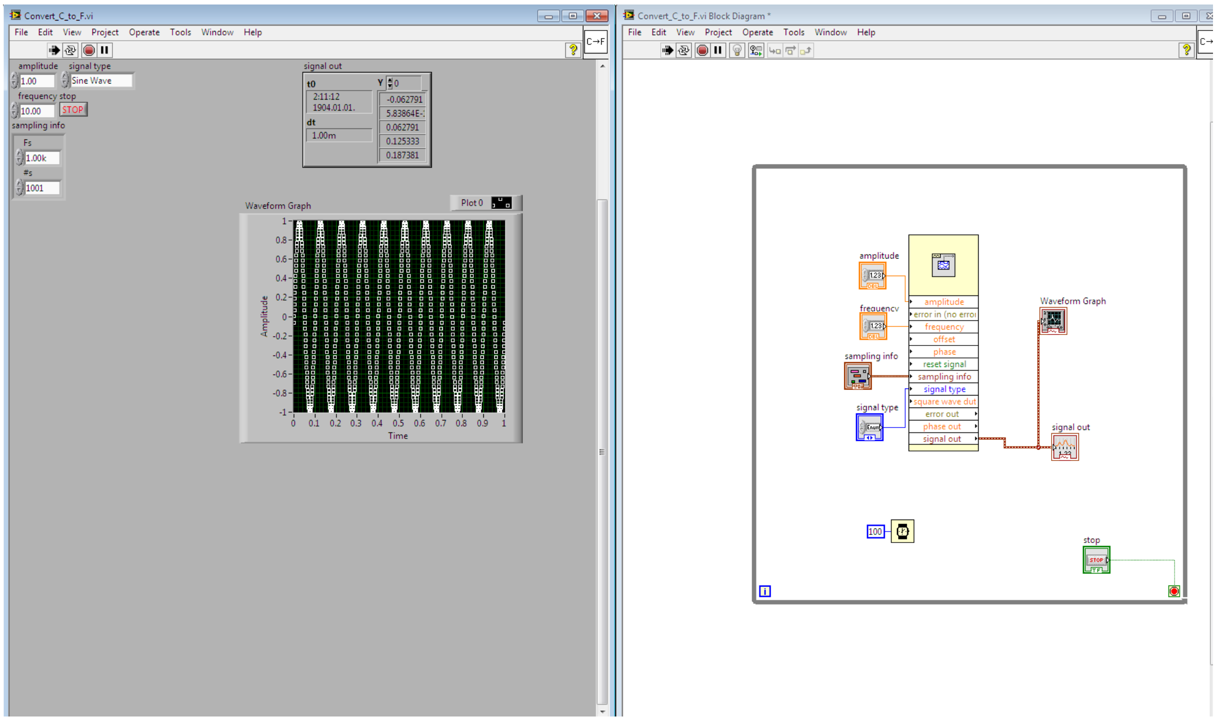
Task 2:

In this task we will be creating a virtual function generator. The one that is mostly similar to a real digital function generator (except for easier access and fewer options). To accomplish this task we will be adding a Wfm Generator from the Function pallette located in the block diagram. After adding the wfm Generator, we can right click and press „View as icons” to show all the available inputs and outputs in front of the screen for easier access.

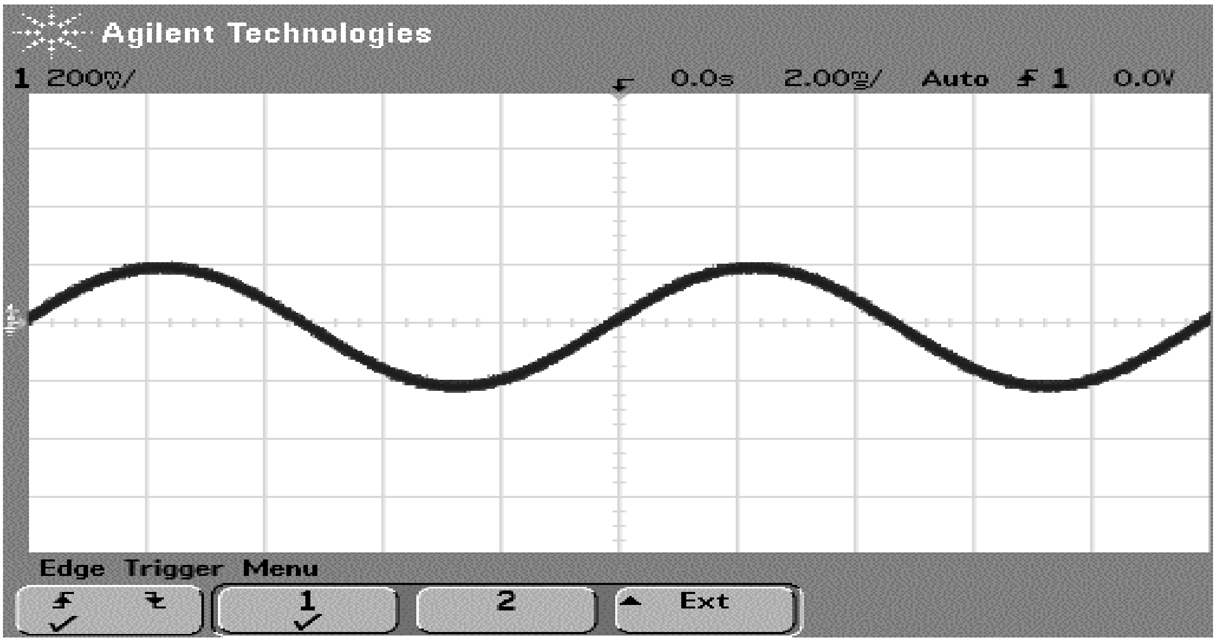
To make the generator we will not be using all the outputs and inputs but the followings:

- Amplitude [Input]
- Sampling Info [Input]
- Signal Type [Input]
- Frequency [Input]
- Signal Out [Output]

Inputs are easier to understand as a whole because most of them will be control. But for the Signal out, we need to put it into two output palletes, one will be a „WaveForm Graph” type output and the other one will be a general „indicator” output. After this setup we can actually change the waveform types and amplitude etc in the Front Panel window and see the changes/waveform in the waveform generator. We will put it in a while loop, to ensure that the waveform generation is done continuously. We can see the setup in the screenshot below:

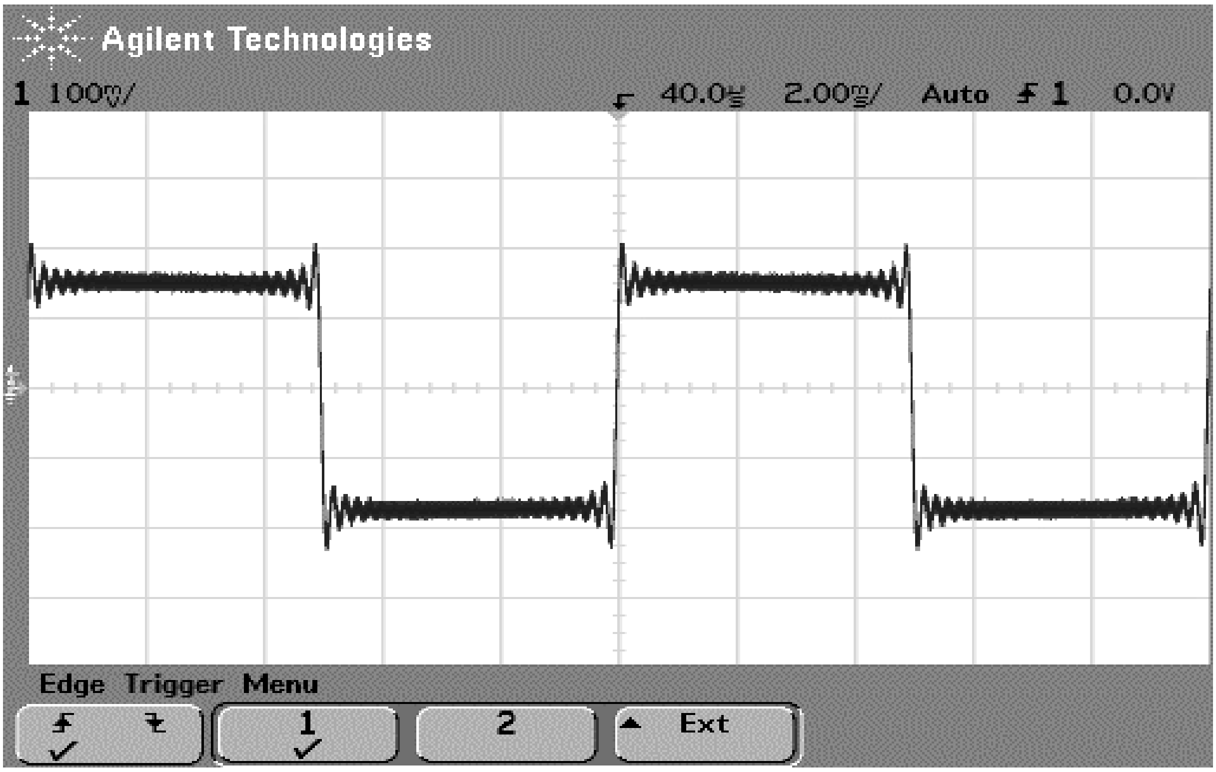


First we will try to impliment a normal sinusoidal signal with amplitude 1 and frequency 1 KHz. Since we are working with waveform, it is a very good idea to get our LabView setup to link with the Oscilloscope. To do that we can use the „Sound Output" of the block Diagram Function Pallette and Get started From there. After the set up with the oscilloscope, we can see our sinusoidal signal in the Oscilloscope screen. Here is the Following screenshot from the Oscilloscope screen.

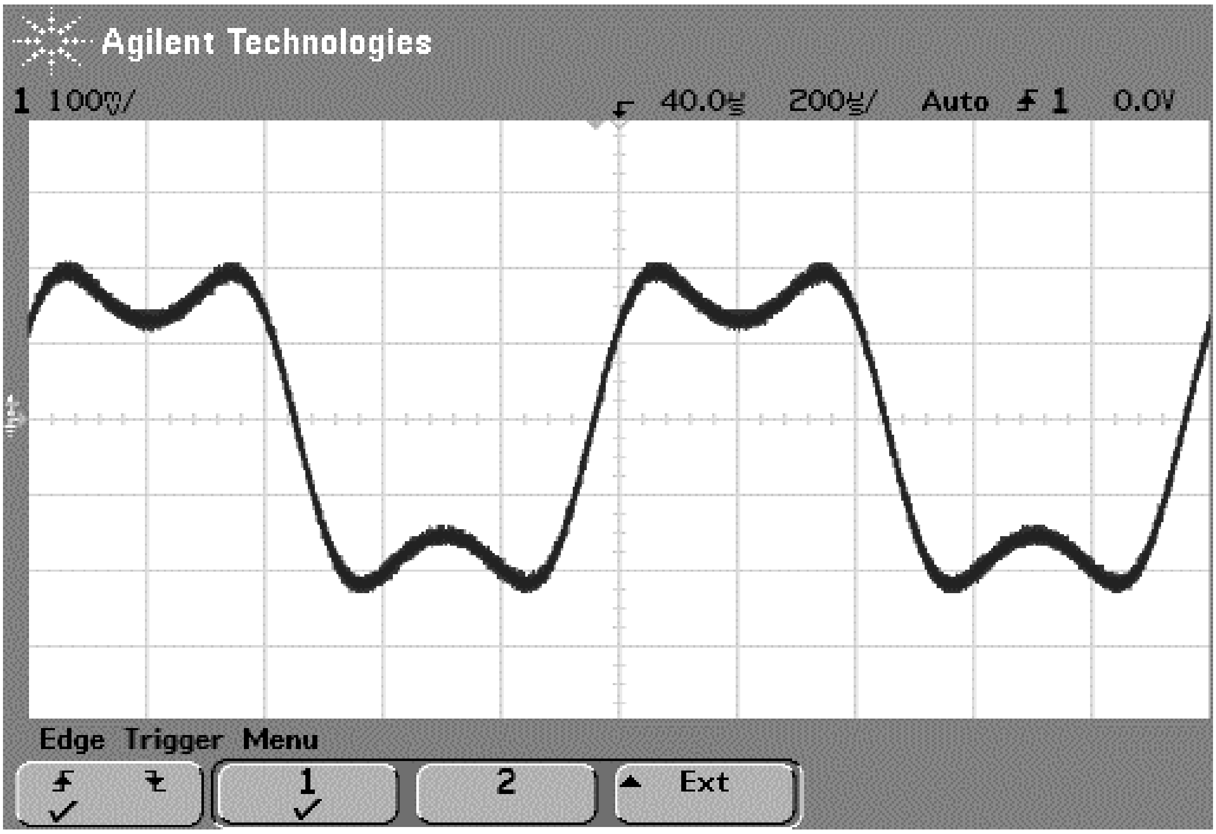


In the Next one, we will try to impliment a Square Wave from the Front panel Window. After selecting the square wave with 100 Hz, we can see that the wave is not a pure square wave, since there are ripples in the edges of the square wave.

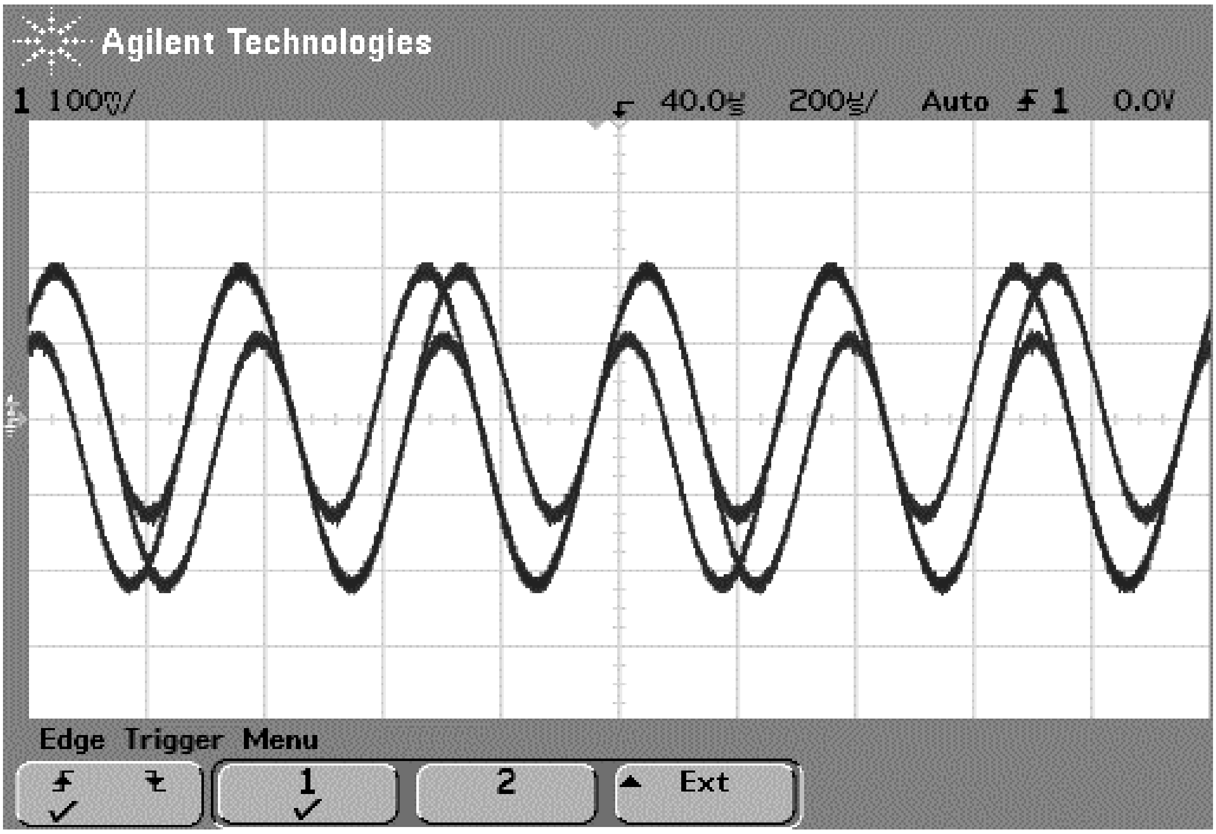
<u>Square Waveform with 100 hz Frequency</u>:



Square Waveform with 1k hz Frequency:

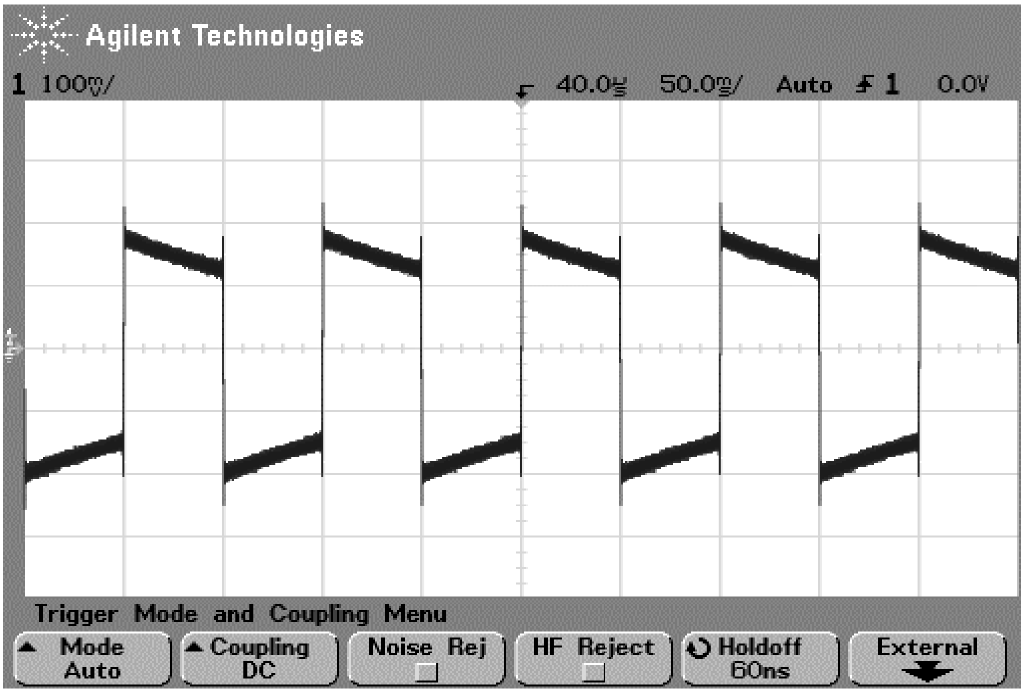


Square Waveform with 1k hz Frequency: With sampling frequency of 10k, sampling nubmer of 10k

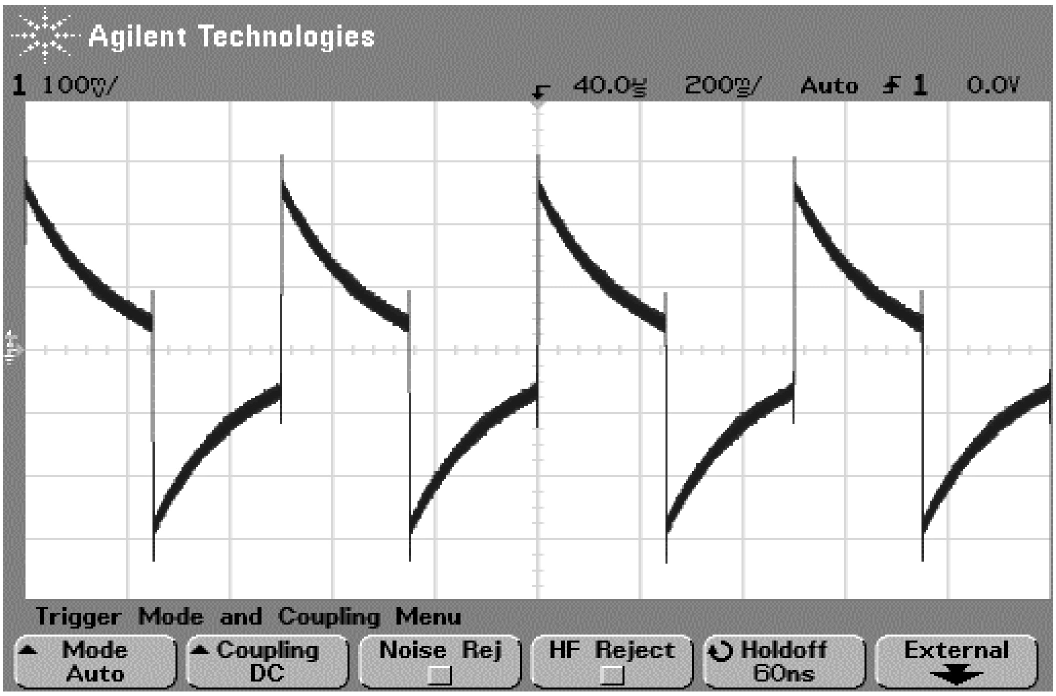


There is a weird phenomenon that can be observed when the frequency is being lowered constantly. There is a capacitor at the output and when we keep lowering the frequency, the capacitor value as a whole will keep increasing since we already know that Value of C = 1/jwc and w is proportional to frequency. And when the frequency is lowered, the value of C will increase which as a result will form a high pass filter. That is the reason when frequency is lowered, we can see shapes similar to what we could actually see in case of a high pass filter wave form.

10 hz: the lower the frequeny, the higher the cahratericstic of the high pass filter



2 hz: the lower the frequeny, the higher the cahratericstic of the high pass filter

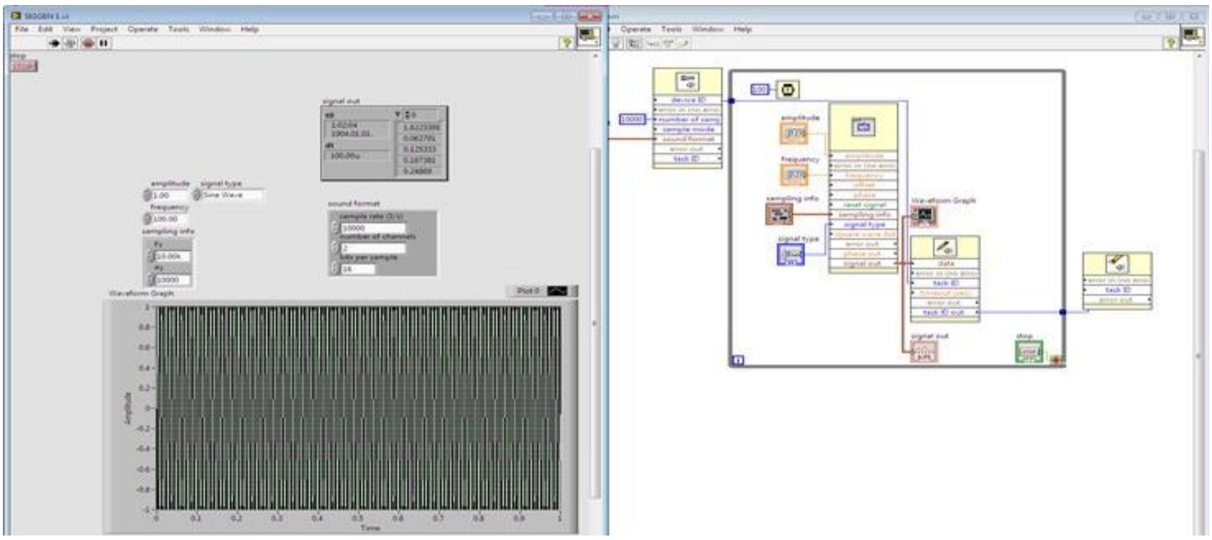


Task 3:

In this task, we will be graphing the waveform of sound from our computer system, previously we have portraited the waveform from the function generator from the VI and as well as the function generator. But now we will be connecting the input to the computer and from the sound card build in inside the computer itself. First we need to connect a Jack – RCA cable to the output of the sound card, and using a RCA-BNC converter, connect the cable to an oscilloscope To do that we will be using the „Functions” → „Graphics & Sound” → „Sound” → „Output”. In this pallette we will use three types of functions. They are as following:

- Configure
- Write
- Clear

After this, we will sort the connections as follwing to the screen shot and we will be able to take output from the sound card as well.

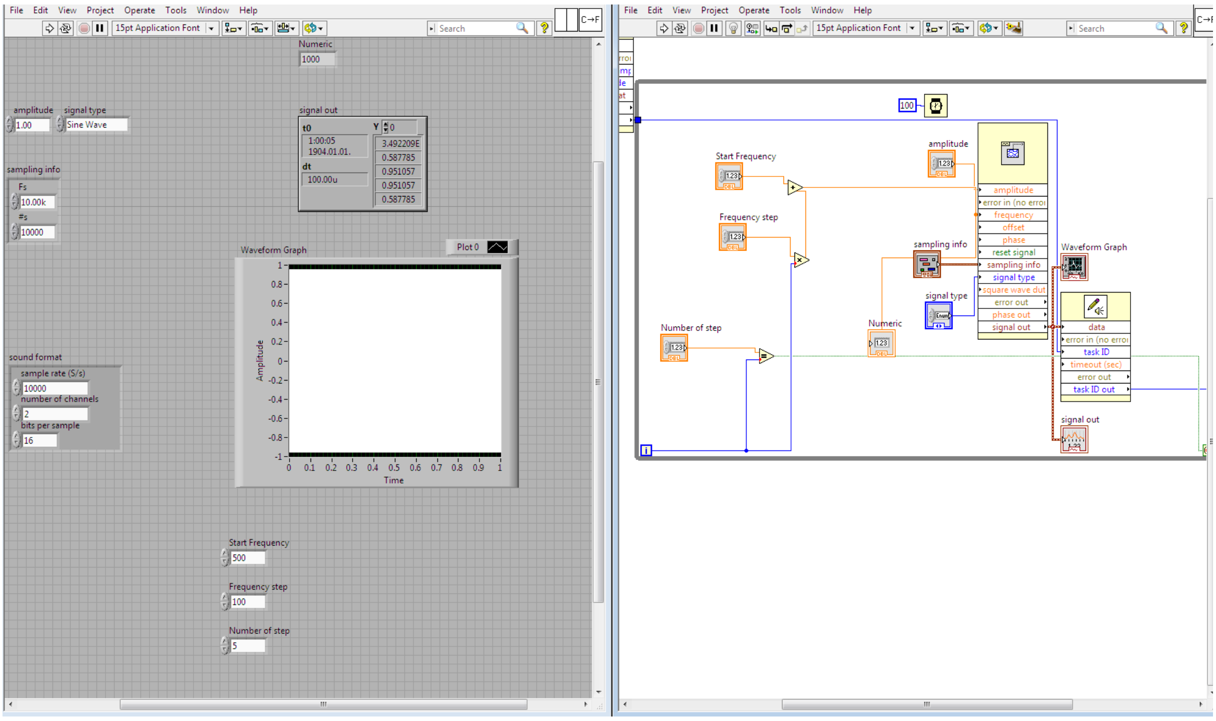


Task 4:

In this task we will be making a variable frequency generator. We will modify the function generator to make the frequency of the output signal change in predefined steps. To do that we have to implement the following specification:

- Introduce Input variables with initial values : the Start Frequency, the Frequency Step, and the Number of Steps
- Each frequency has to be active for 1 second. For that we will be using the Wait functionality.

The front panel and the block diagram will look as such:

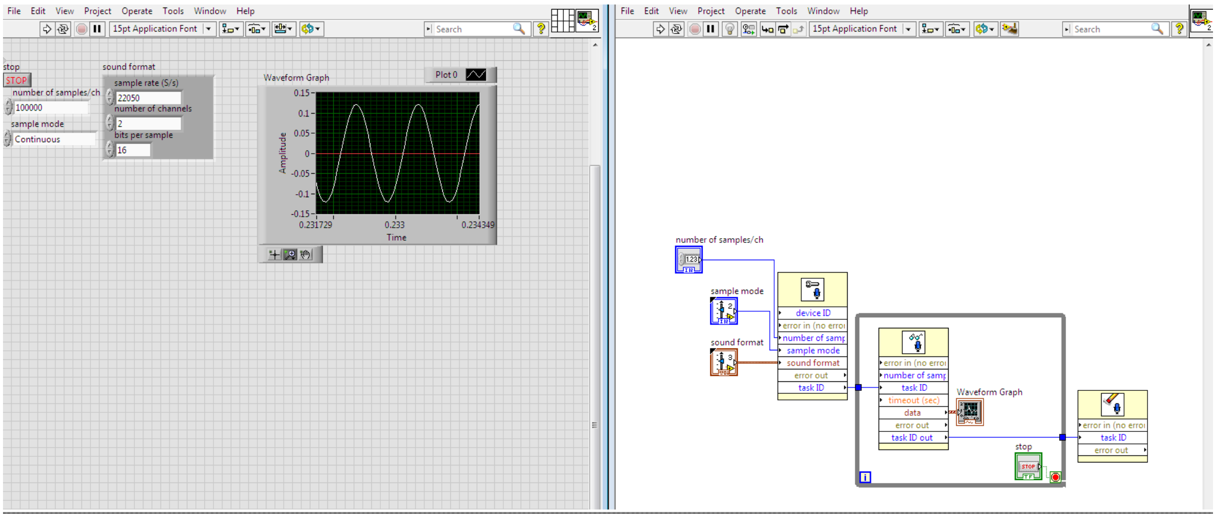


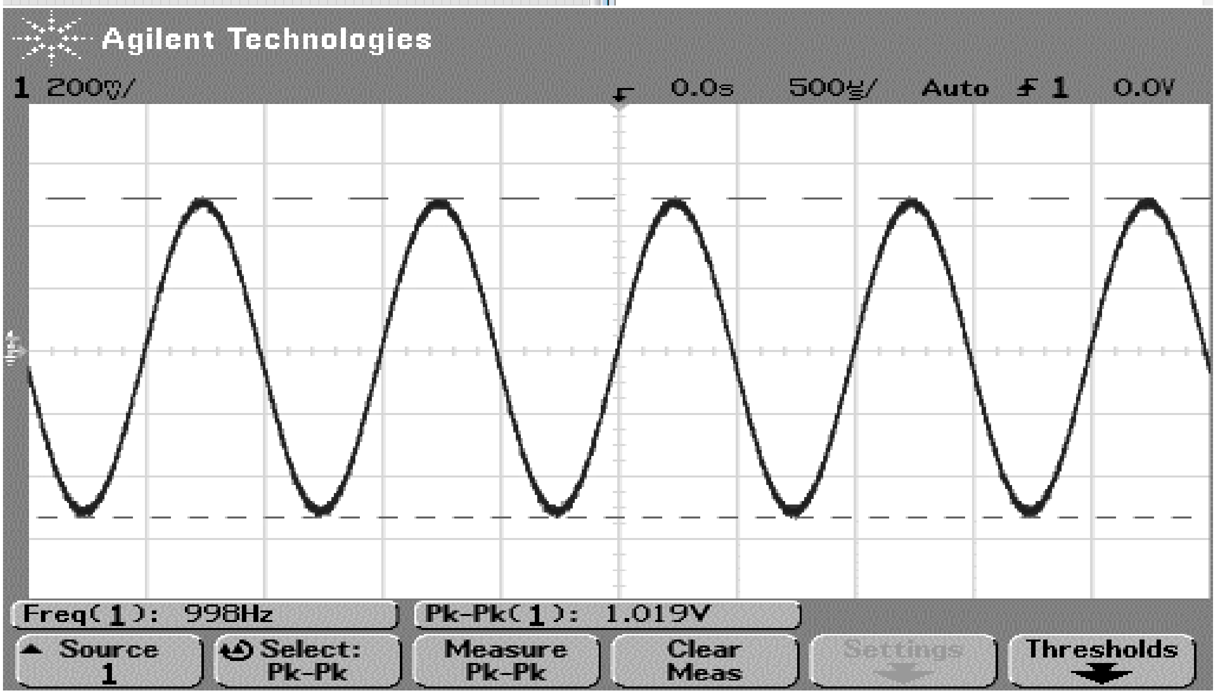
Here we will be seeing, for whatever signal we choose , the waveform graph will show the signal with the starting frequency. And then it will increase by the predefined frequency step , and it will increase defined number of times. But most importantly, it will be in the screen for 1 second. We can change this time by configuring the wait function. And we also have observed this signal and its changing through function generator and oscilloscope.

Task 5

In this task we will visualize a input signal in the Oscilloscope. Before connecting the cable to the sound card, we should double check the signal using an oscilloscope. The Line In input of the sound card can be driven by 1 V RMS signal at maximum. We also have to make sure that the cable is connected to the Line In input, and not to the Mic In input.

To do this, we have to select input sound configurations from the block diagram panel. We have to specify the number of samples per channel, sampling mode. Also we need the input sound read , which will be connected with the waveform graph. And lastly we need to connect task ID out with the sound input clear.





Finally, we have observed the signal using oscilloscope also, and we can see that we have a signal of frequency 1kHz and 1 V peak to peak amplitude as expected.