

## **LAB#09**

**NAME : ABDULLAH ZUNORAIN**

**REG NO : 19JZELE0338**

**SUBJECT : DATA COMMUNICATION(LAB)**

**SUBMITTED TO : DR. UZAIR GILLANI**

**SECTION : A**

**DEPT : ELECTRICAL COMM**

**CAMPUS : JALOZAI**

# **To Study the behavior of Different Waveforms in Simulink**

## **OBJECTIVES OF THE LAB**

---

In this lab, we will cover the following topics:

- To understand the basics of Simulink
  - To generate and analyze various waveforms
-

### Purpose

To observe and understand how the analogue and digital waveforms changes as some of its characteristics are changed with the use of MATLAB®/Simulink® package.

### Overview

MATLAB is a high-level technical computing language and interactive environment for algorithm development, data visualization, data analysis, and numerical computation (quoted from MathWorks website). Simulink, an extension of MATLAB, is a software package for modelling, simulating and analysing dynamic systems. It provides an interactive graphical environment and a customisable set of block libraries that let you accurately design, simulate, implement, and test control, signal processing, communications, and other time-varying systems. In Simulink, the representation of a system or a process to be studied is called a 'model'. For modelling, Simulink provides a GUI like an IDE, for building models as block diagrams using click-and-drag mouse operation.

In this lab, you'll construct some basic models based on sine wave as the main component, and observe how the simulation output changes as you vary your model parameters as well as when other related components are added to your model.

### Part 1: Experiment with Analogue Waveforms

#### Step 1:

At the MATLAB command window, type up `simulink` (and Enter) at the "`>>`" prompt to bring up the Simulink Library Browser window. You will see various building blocks appearing on that window as shown in Fig. 1.

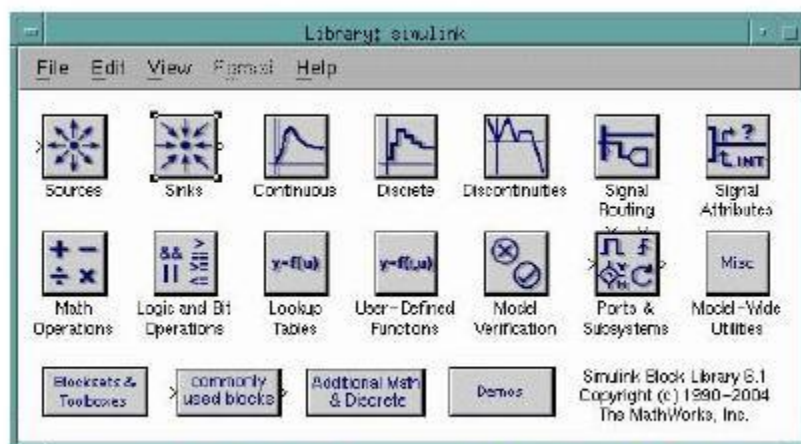


Fig. 1. Simulink main interface with its library browser

#### Step 2:

To build a new model, click on the File menu on the Simulink Library Browser window. This will bring up an empty model window that allows you to construct your new model.

From the Simulink Library Browser window (Fig. 1), double-click on the Sources block library to bring up a list of blocks under that library. Find the Sine Wave block, as shown of Figure 2. Click and drag the Sine Wave block to the empty model window (Fig. 3).

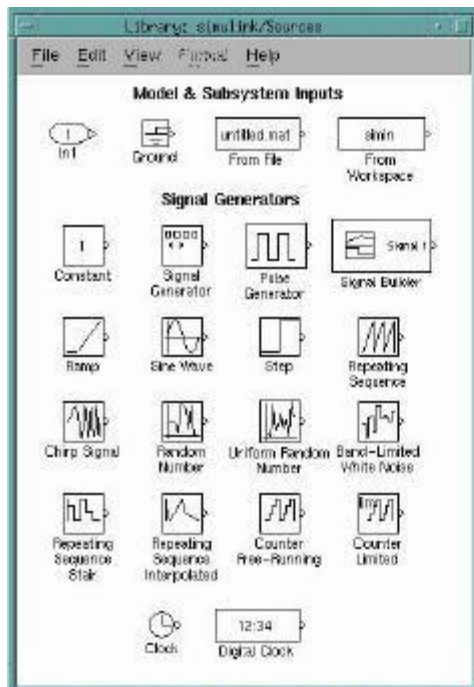


Fig. 2. Sources library window

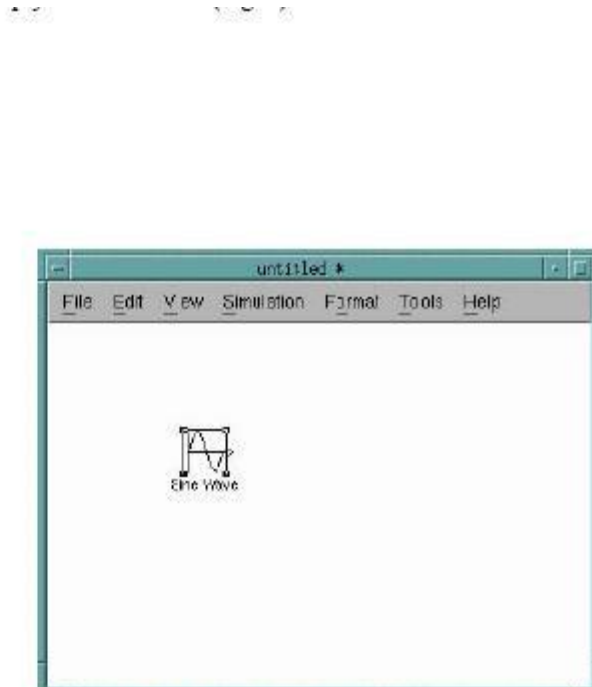


Fig. 3. New model window

### Step 3:

Similarly, open the Sinks block library from the library browser and, click and drag the Scope icon to the model window. If you examine these blocks, you see an angle bracket on them. The > symbol pointing out of a block is an *output port*; if the symbol points to a block, it is an *input port*. A signal travels out of an output port and into an input port of another block through a connecting line. When the blocks are connected, the port symbols disappear.

## Step 4:

Now it's time to connect the blocks. Position the pointer over the output port on the Sine Wave block. Notice that the cursor shape changes to crosshairs. Hold down the left mouse button and move the cursor to the input port of the Scope block. Now release the mouse button. Your model should look like this.

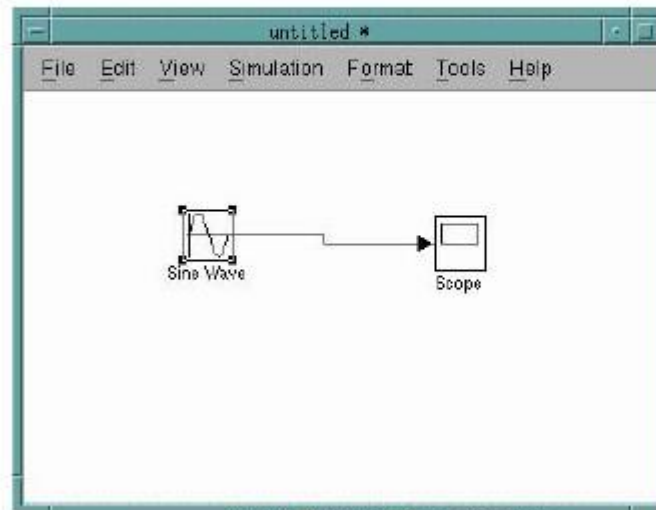


Fig. 4. Completed model view

## Step 5:

You can observe the properties of the periodic signal to be generated by double clicking the Sine Wave block. The Source Block Parameters window allows you to change the amplitude, frequency and phase of the resultant signal. The standard unit of frequency in Simulink is radians/sec (rad/sec). To convert frequency from Hertz to rad/sec, simply multiply by  $2 \times \pi$  ( $\pi$  is MATLAB constant). For example, if the frequency is 40 Hertz, it will be  $40 \times 2 \times \pi$  rad/sec. (Note: To display signal of higher frequency correctly, the sample time should be set appropriately.)

**! Save this model now; use a meaningful name such as "sinewave01.mdl", and remember the directory where you saved the file.**

## Step 6:

Before we execute the model, open Configuration Parameters window from the Simulation menu. Change the Stop Time from 10.0 to 1. This controls the duration of the simulation.



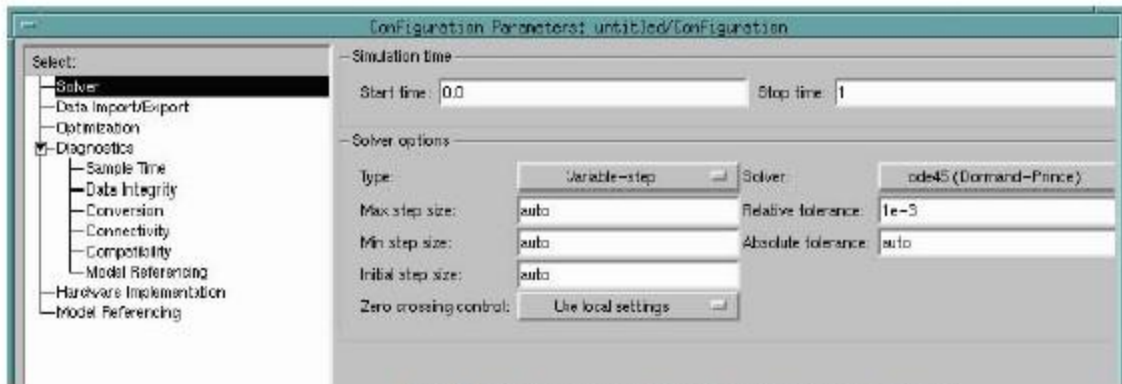


Fig. 5. Simulation Configuration Parameters window

## Step 7:

Now, double-click the Scope block to open a display window. Choose Start from the Simulation menu to run the simulation and watch the simulation output on the scope (Fig. 6). Figure 6 shows the sine wave with  $A = 1$ ,  $F = 10 \text{ Hz}$  ( $2 \times \pi \times 10$ ) and phase = 0 (Sample time = 0.001). Note that there are ten cycles per second for this configuration. Experiment with different values of the sine wave parameters. Each time change one parameter, rerun the model and observe the behaviour.

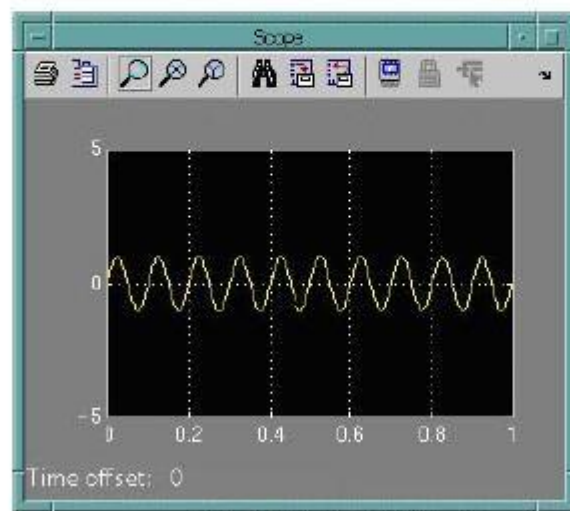


Fig. 6. Output on the Scope.

## Step 8:

To be able to compare one signal to another, it will be better to display them on the same scope display. To achieve this, we need to use a multiplexer that could combine several input signals into a vector or bus output signal. Open Signal Routing block from the Simulink Library window. Click and drag Mux block to the model window. To generate another sine wave, copy another Sine Wave block from the Sources library. Reconnect the block as shown in Fig. 7.

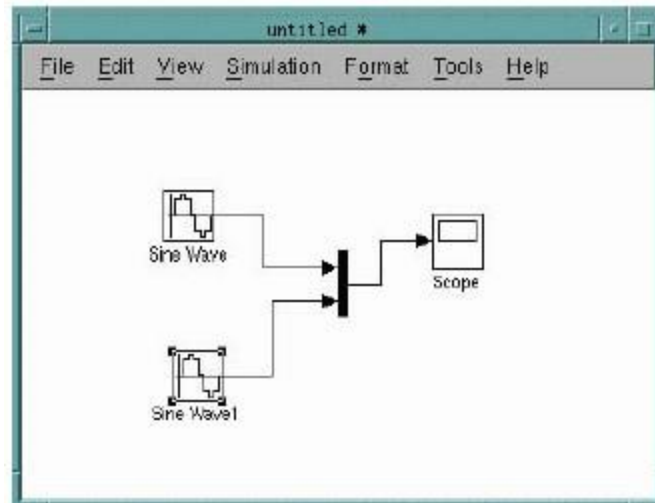


Fig. 7. Two signal generators connected to a single scope

Step 9:

Change amplitude of the signal from for the Sine Wave1 block to 3 by double-clicking on the block to open the Source Block Parameters window. Run the simulation as before. The display should be similar to this.

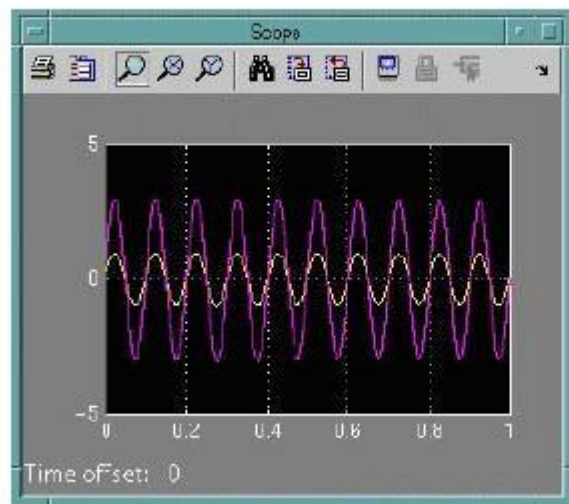


Fig. 8. Scope display for the two sources model

Step 10:

Change its frequency and then phase to see the effect. You should be able to explain the different behaviours. Normal right-click mouse action also works here to open a context menu. For example, right-click on the **Mux** block and select Mux Parameters option. Change the Number of inputs textbox to 3 to allow more signals to be multiplexed through it.

## Part 2: Experiment with Digital Waveforms

1. Create a new model as before. To generate a square wave, open Sources library from the Simulink Library browser. Click and drag **Pulse Generator** block. Copy also the Scope block to the model window. Connect these blocks as shown.

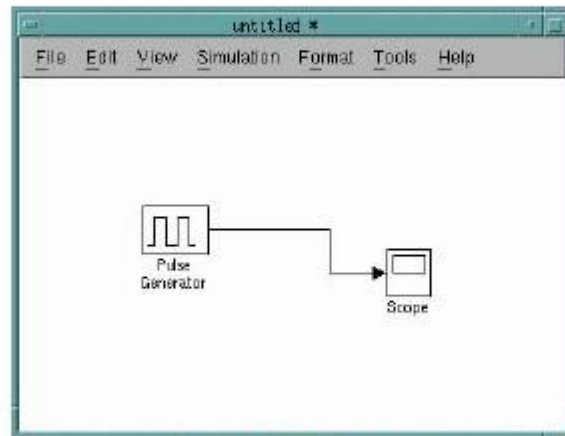


Fig. 9. Modelling a square wave

2. Double-click on the **Pulse Generator** block to open the Source Block Parameters window. Check the defaults values of the generator.

3. Open the scope display window by double-clicking on the **Scope** block. Execute the model by selecting Start from the Simulation Menu. Observe the digital waveform on this display.

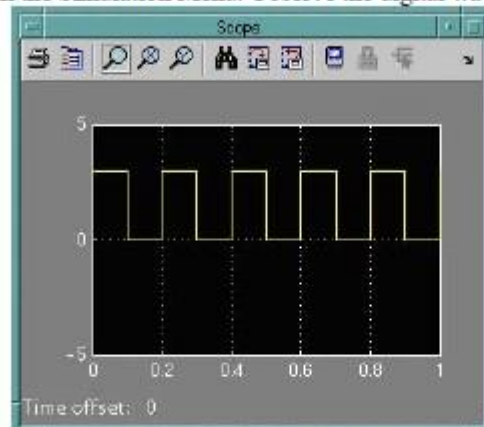


Fig. 10. Digital waveform display

4. Change the amplitude, run the model and observe the output. Now, vary pulse period and repeat the experiment. Use the Mux block if need to compare two different signals as before.

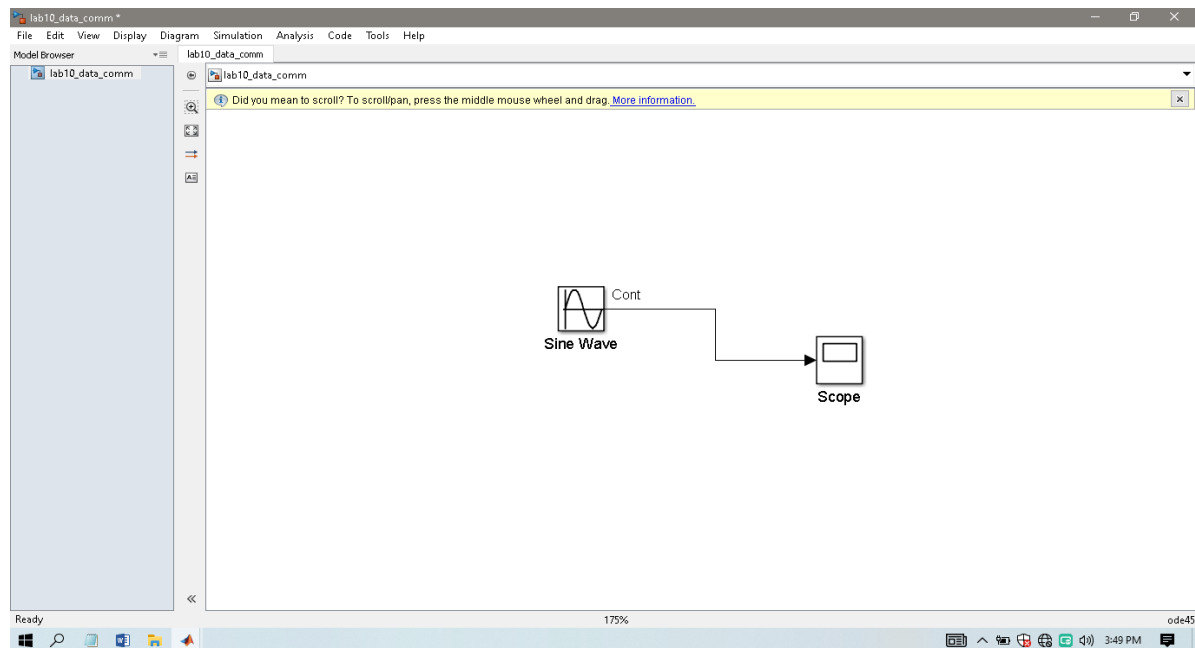


## -----TASK#01-----

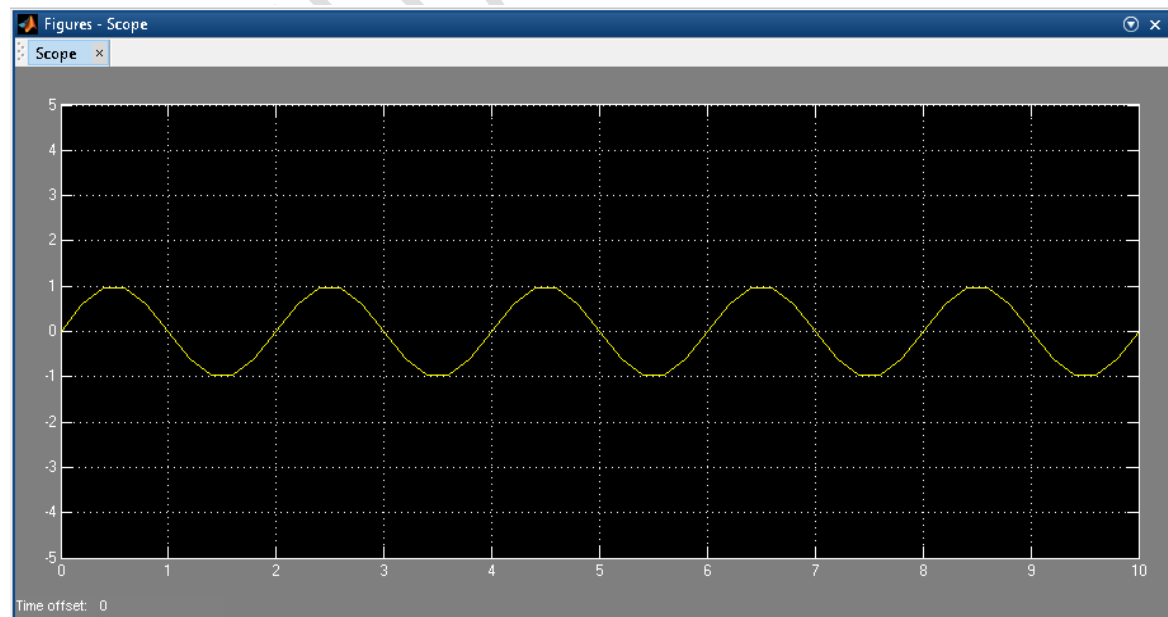
### 1. Behavior of analogue Waveforms in Simulink:

ANS:

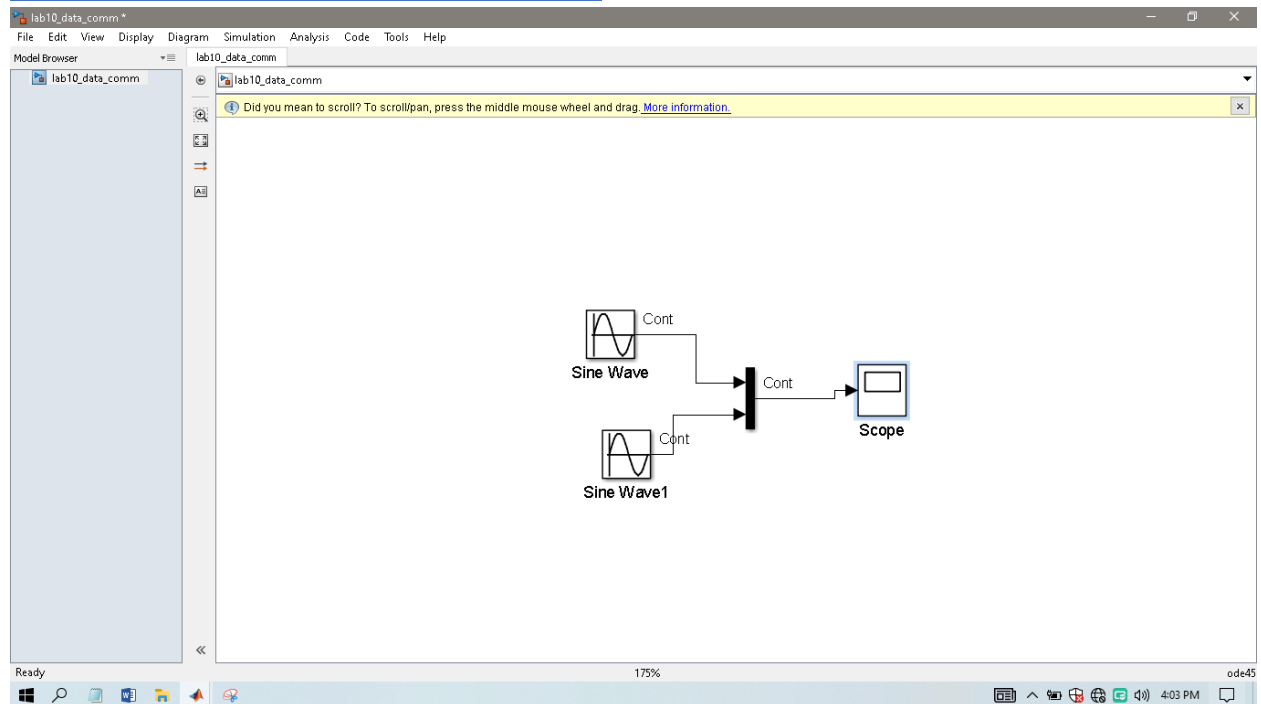
TO STUDY AND ANALYZE THE BEHAVIOR OF THE ANALOGUE SIGNAL:



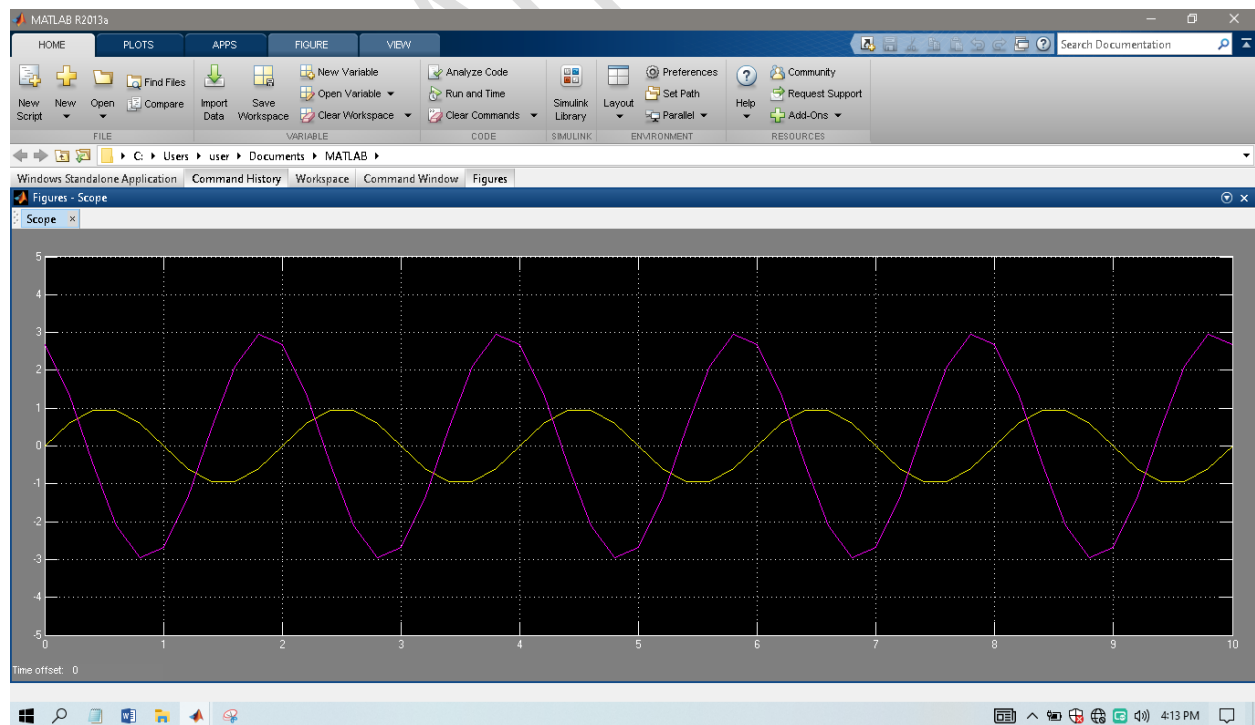
First we have to set the parameters of the sine wave by clicking on the **sine wave box** and then we have to run the simulation and click on the scope to analyze the sine wave as shown in the figure below;



## NOW FOR ANALOGUE MULTIPLEXER :



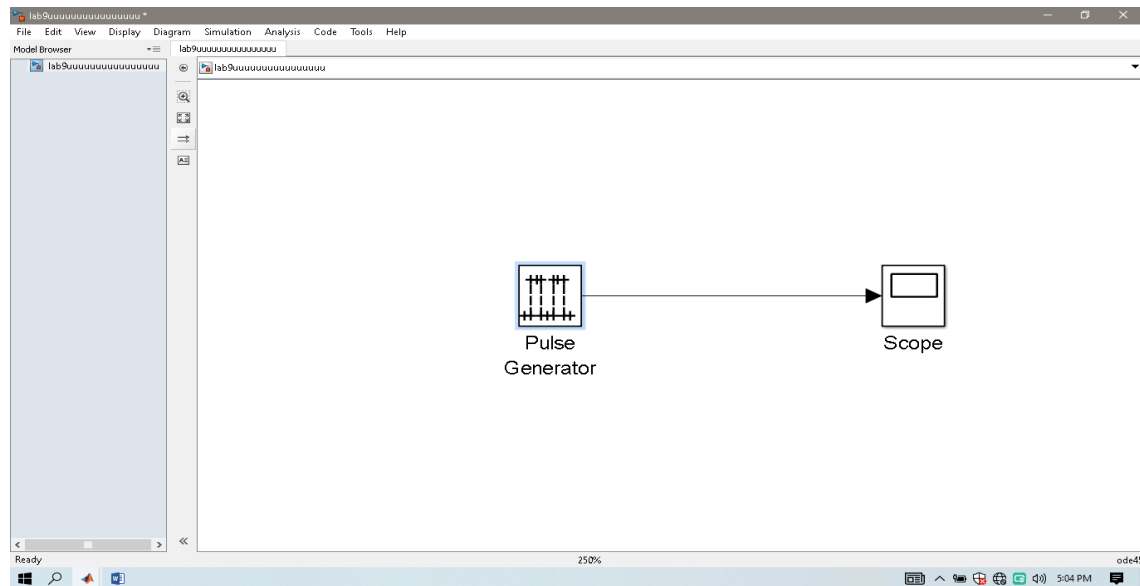
The similar way to analyze the behavior of the **multiplexer** we have to add 2 or more sine wave boxes and a **Mux** from the **simulink library** , now to set the peremeters then run the simulation and simply click on the scope to see the display as given below in the fig;



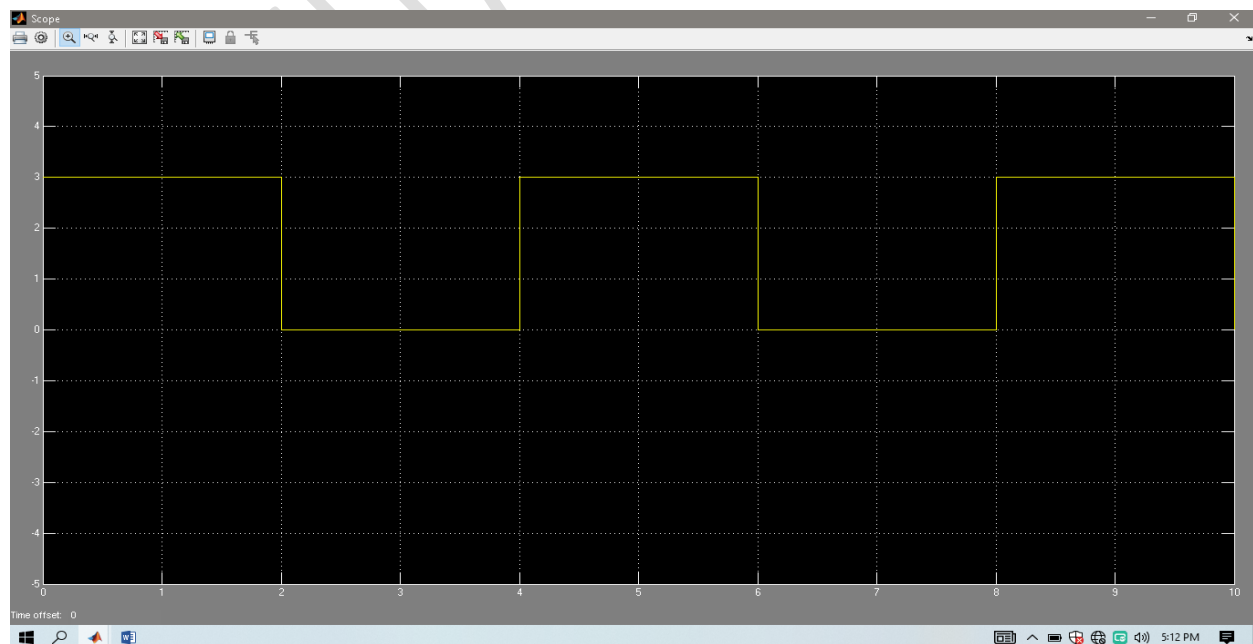
## TASK#02

### 2. STUDYING THE BEHAVIOR OF DIGITAL WAVE FORM:

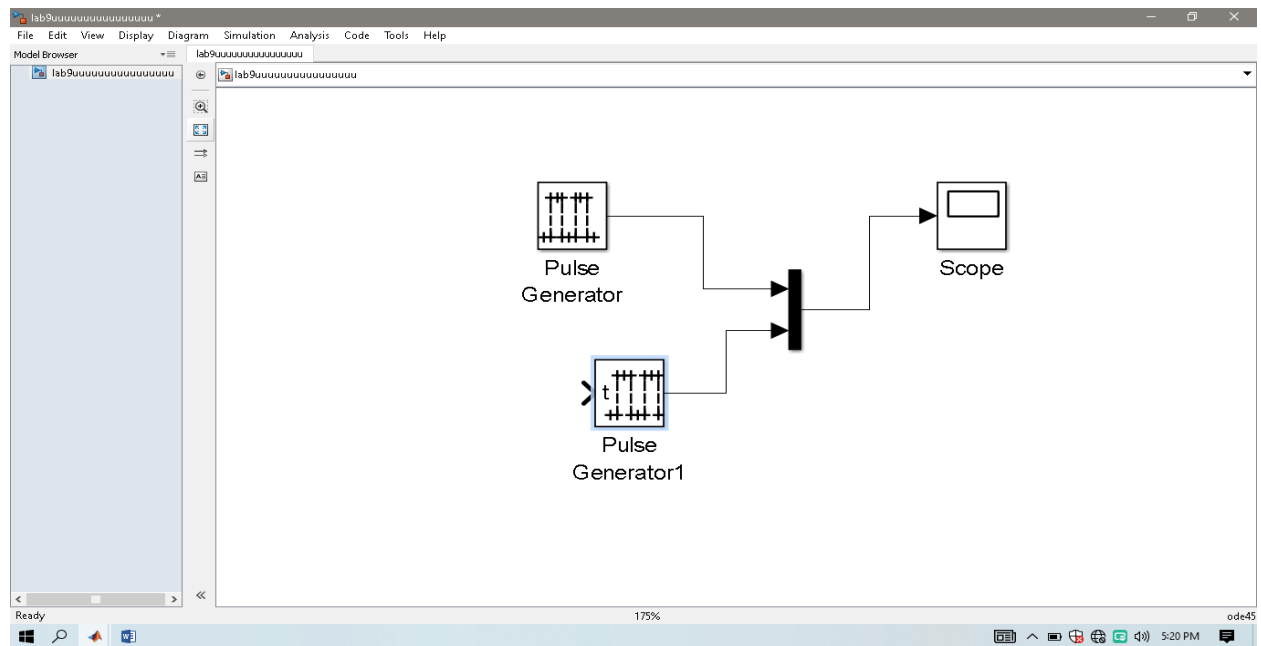
**ANS:**



First we have to set the parameters of the digital wave by clicking on the **pulse generator box** and then we have to run the **simulation** and click on the scope to analyze the digital wave forms as shown in the fig below;



## NOW FOR DIGITAL MULTIPLEXER:



Now for digital multiplexer we use the same method as discuss in case of analogue but here in this case we use pulse generator for multiplexing , as shown in fig below;

