EECE 544 Embedded Systems Design

Lab 2: Oscillator

1. **Pre-Lab**
   - Download and unzip **Oscilator_Lab2** from Blackboard into your computer.
   - Open and compile the oscillator project. Make sure you are getting no errors before you start adding any new lines of code.
   - Examine the code and make sure you understand all lines, procedures, and guidelines.
2. **Required Parts:** 4-10KΩ resistors, audio jack, and 2 normally open switches.
3. **Objective**

   The objective of this lab is to create a variable frequency oscillator. The system has two digital inputs, *input0* and *input1*, and two digital outputs, *output0* and *output 1*. If *input1* is true the digital output "*output1*" oscillates at *262 Hz* (becomes high for half the period T, and low for the other half. Remember *T = 1/f*). if *input1* is false, *output1* remains low. If *input0* is true *output0* oscillates at *392 Hz*, if *input0* is false, *output0* remains low. If both inputs are true, both outputs remain low. When you connect each output to a 10KΩ resistor as shown in figure 1, you will be able to hear the tones as middle C and middle G.

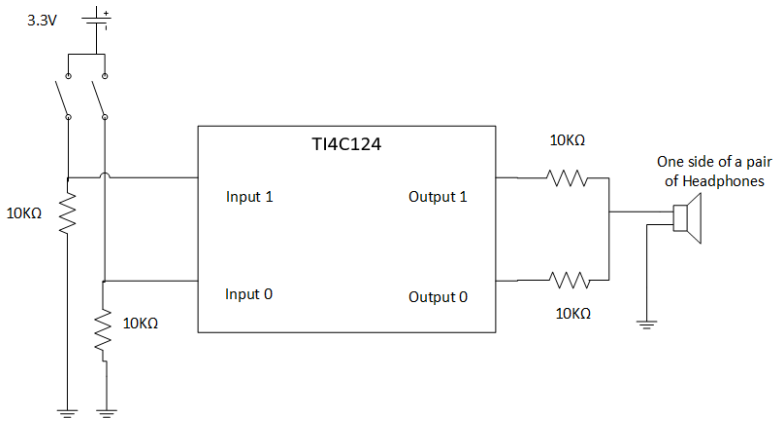

Figure 1

4. **Procedure**

   You will be conducting this lab using the simulator then building the lab using actual hardware.

   1. Read the problem description carefully and create your flow chart for the program
   2. Write pseudo-code for your algorithm before you start coding.
   3. Open your lab project by double clicking on Oscillator.uvproj
   4. Open the main.c file and insert your solution. Make sure to create separate functions to initialize PORTF and PORTE (you can choose any other port; just make sure you have access to these ports on the board). Compile your file before you start

adding any lines of code and recompile any time you add a new function. It is easier to find mistakes and problems when you debug one addition at the time.

5. Build the project and run it in the simulator. You can input values for your switches by hardcoding the input into the code to test it or entering it using the command window.
6. Download the project into your board and debug using Logic Analyzer. (Ask your TA if you need help using the logic analyzer).
7. After you run your design and make sure it's functioning as expected, connect the switches, resistors and audio jack to create your system as shown in figure 1. Refer to figure 2 for the audio jack connections. (You will be reusing this audio jack again in future labs, please don't lose it).

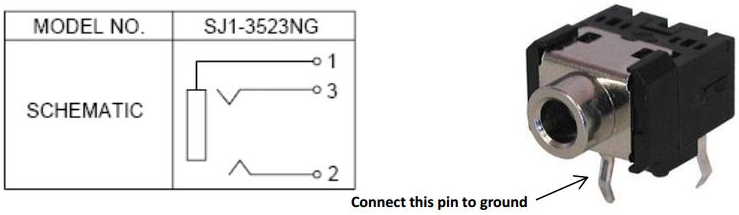

Figure 2

8. Insert your headphones into the audio jack; you should be able to hear two different notes when you press on the switches.

5. Demonstration

When demonstrating the program, you are expected to explain each line of code if asked. Each student in a team will be asked a different question to demonstrate his or her understanding of the project at hand. To demonstrate, connect input0, input1, output0 and output1 pins to four different channels of the logic analyzer and take a snapshot of your output to include in your report. Your snapshot should look similar to the signal shown in figure 3.

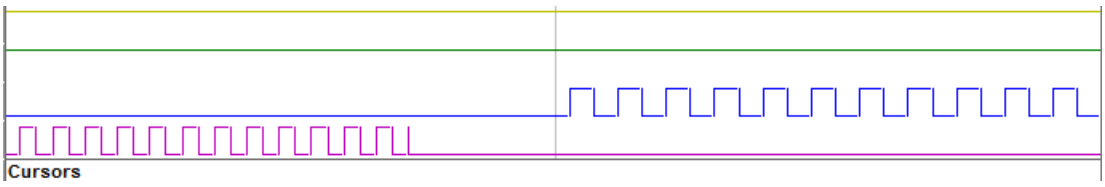

Figure 3

6. Deliverables

a. C source code of your project. Your program will be graded for good documentation.
b. A snapshot of the logic analyzer capture.