Cpr E 281 LAB5
ELECTRICAL AND COMPUTER ENGINEERING
IOWA STATE UNIVERSITY

Programming and Testing the Altera DE2-115 Board

PRELAB!

Read the entire lab, and **complete** the prelab questions (Q1-Q2) on the answer sheet **before** coming to the laboratory.

1.0 Objectives

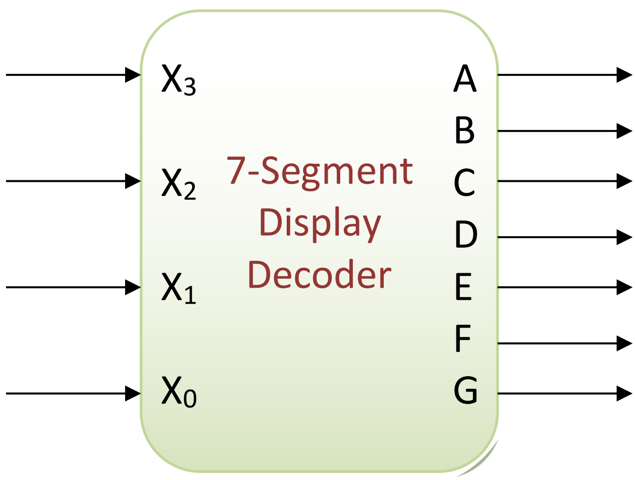
The main objective of this lab is to gain experience in programming a 7-segment display module. We will design a control circuit for a 7-segment display, and then we will show how to efficiently reuse that circuit for multiple 7-segment displays. Then we will test each possible input and observe the output in ModelSim.

2.0 Setup

Begin by creating a new folder **U:\CPRE281\Lab05**, and then two sub-folders **\Lab05\lab5step0** and **\Lab05\lab5step1**. You will be saving your work for this lab in these directories.

3.0 The 7-segment Display (lab5step0)

Start a new project, in the *lab5step0* folder, and name it **seven_seg_decoder**. Then click **Finish**.



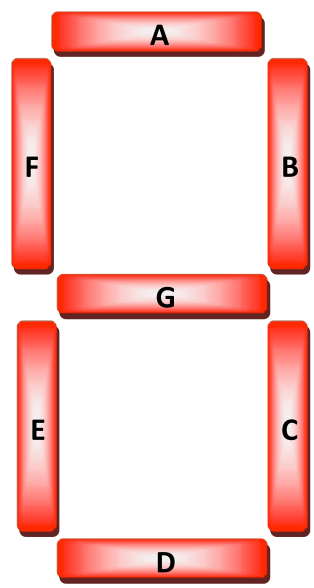


Figure 1: 7-Segment Display Decoder

A 7-segment display is shown in **Figure 1**. The decoder provides the outputs A-G needed to light the proper segments of the 7-segment display based on a 4-bit input.

IMPORTANT:

There are two different types of 7-segment displays on that turn on with logic 1 and others with 0. Note for this lab assume that the LED's of the 7-segment display boards are lit when there is a logic 0 instead of a logic 1 connected to them.

Create the decoder in Verilog. Your inputs {X_3, X_2, X_1, X_0} represent a hexadecimal value (by convention, X_0 is the LSB). Your outputs are {A, B, C, D, E, F, G}. Review previous lab instructions if needed for Verilog code. Name your Verilog module **seven_seg_decoder** (same as your project name). The choice of Verilog coding method is left to you.

Compile your code and fix any errors. Run your file on ModelSim and make sure to review output representation of numbers 0-F.

4.0 Creating Symbols - Design for Reuse (lab5step1)

You have designed and tested a circuit that can display hexadecimal values on single 7-segment display. Now we will create a circuit that can control four 7-segment displays each with its own set of inputs. It would be ideal to be able to utilize your existing design, and just duplicate it four times. This can be done by simply turning your **seven_seg_decoder** design into a symbol that can be used (and reused) in a schematic (or block diagram).

While still in your *seven_seg_decoder* project select the **seven_seg_decoder.v** tab to view your Verilog code. Then click **Processing->Analyze Current File**. Select the **seven_seg_decoder.v** tab again to view your Verilog code. Click **File->Create / Update->Create Symbol Files for Current File**. When successful, you have created a symbol that you can use again.

Using Custom Symbols:

- Navigate to your *lab5step0* folder and copy the **seven_seg_decoder.bsf** and **seven_seg_decoder.v** files to your *lab5step1* folder.
- Create a new project in the *lab5step1* folder and name it *lab5step1*.
- Create a new block diagram and save it as *lab5step1.bdf*.
- Click on the **Symbol Tool** in the upper toolbar. Notice that there is a **Project** directory under '*Libraries:*', and your new symbol is now available for use.
- Place 4 instances of the **seven_seg_decoder** symbol.

Verify your circuit on the ModelSim and show your results in the lab report.

5.0 Complete

You are done with this lab. Ensure that all lab files are closed, exit Quartus Prime, log off the computer and submit your answer sheet. **Don't forget to write down your name, student ID, and your lab section number**.