# Lab 08

## Part A – Multiplexed 7-Segment Display

### Description

In this lab module, you learn about multiplexing applied to a 7-segment display to show multiple different values.

As usual, for this lab module, you will require a PC with a USB-A port to program the FPGA development board.

## Procedure

### Background

1. Listen to the description of the 7-segment display and its decoder's truth table. Some details are below.
    - The Basys 3 FPGA development board has a four-digit LED display, where each digit consists of seven segments (plus a dot, or decimal point).
        - You can find out more details about the seven-segment display specific to the Basys 3 board here.
    - You may be familiar with this type of display from alarm clocks, microwave ovens, or other appliances with digital clocks.
    - With this display, each of the 7 primary segments can be individually controlled, so there are $2^7 = 128$ possible patterns.
    - Thus, it is possible to display any digit from 0 through 9, as shown below, and also certain letters.

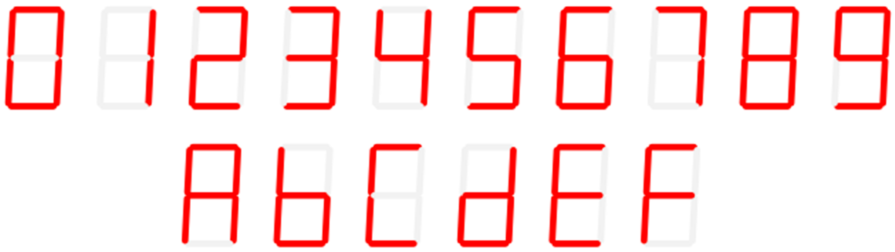

Figure 1: 7-Segment Display, Hexadecimal

## Project Creation

2. Download `lab08a.vhd` and `lab08a.xdc` from Canvas and place them in a new folder titled `lab08a`.
3. Open Vivado and create a new project titled appropriately.
   - If you have not memorized how to do this by now, consult a previous lab handout (ex. 06B).
   - For your convenience, the board identifier is `xc7a35tcpg236-1`.
     - You can copy and paste this directly into the search bar in the device select menu.
     - You can also find the board with the same options as before: `General Purpose`, `Artix-7`, `cpg236`, `-1`.

## Hardware Review and Upload

4. Open the file in the editor, and follow along with the instructor to review the hardware description.
   - To do this: inside the `Project Manager` pane, under the `Sources` subpane, and under the `Design Sources` folder, double-click the `lab08a.vhd` file.
5. `Open Elaborated Design`, review the schematic alongside the instructor, and take screenshots for your report.
6. Run synthesis and implementation and then generate the bitstream to program the device. Then, open the Hardware Manager.
   - Take a screenshot of the resource utilization (LUTs and FFs) to add to your report.
7. Before proceeding, plug the FPGA development board into your computer using the provided Micro-USB to USB-A cable.
8. At the top, in the green banner (or under the `Open Hardware Manager` dropdown) click `Open Target`, then `Auto Connect`.
9. Once the device is connected, select `Program Device` (in either of the locations where `Open Target` was previously).
   - See 06B-handout for troubleshooting steps, or ask the instructor or TA.
10. Click `Program` to upload the bitstream to the FPGA.
    - When it is complete, you should see a pattern on the 7-segment display module on the FPGA development board.

## Hardware Testing and Analysis

11. At this time, take a picture of the board and the patterns on the 7-segment display module.
12. Does the `segment` output of the hardware description match the patterns you see on the four 7-segment displays on the board? Consult `lab08a.xdc` and the figures below for assistance.

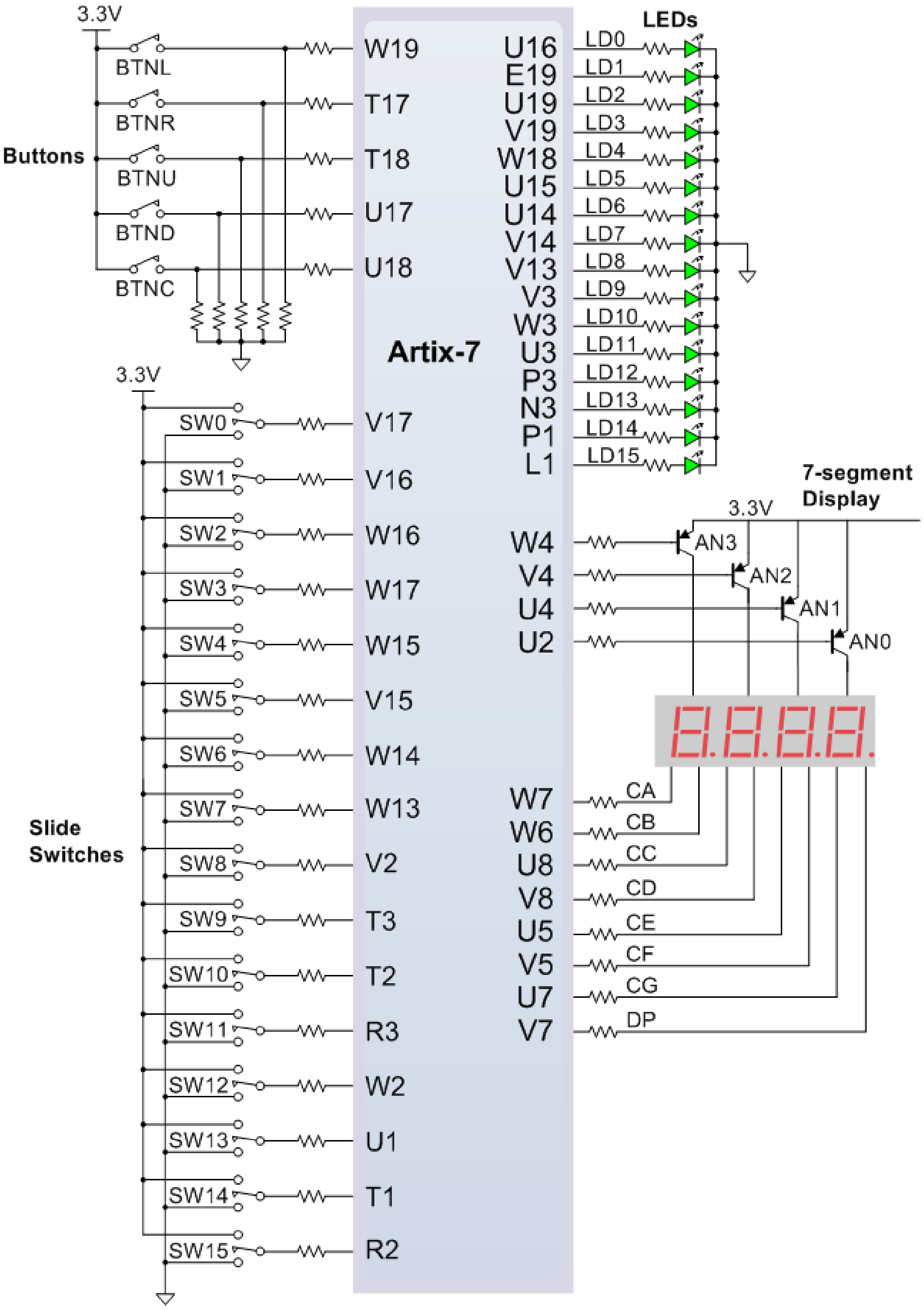

Figure 2: Basys 3 General Purpose Input-Output (GPIO) Diagram

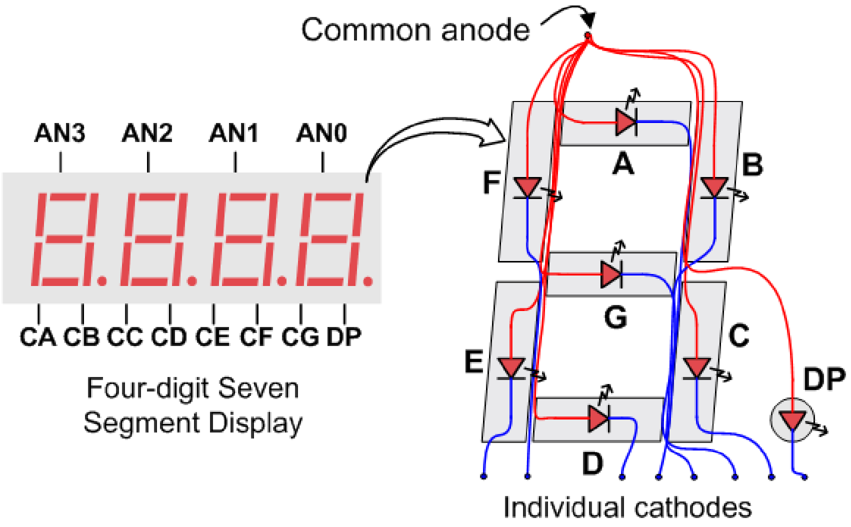

Figure 3: Basys 3 Seven-Segment Display Diagram

13. Change the limit on the counter (`counter_limit`) from `2**18` ($2^{18}$) to `2**10` ($2^{10}$). Re-run the upload process (steps 6-10) and discuss with your lab group how the display output is affected. Take a picture of the board.
14. Change the limit to `2**26` ($2^{26}$). Re-upload, take a picture, and discuss.
15. What is the optimal value of the limit, and why?

# Deliverables

- Include as part of your **informal report**:
  - A screenshot of the Elaborated Design schematic (Step 5)
  - A screenshot of the FPGA resource utilization (LUTs, FFs) (Step 6)
  - Pictures of the FPGA development board (Steps 11, 13, and 14)
  - Answers to handout questions (Steps 12, 13, 14, and 15)

# Outcomes

- Practice working with VHDL.
- Practice using Vivado for hardware synthesis and implementation.
- Practice programming and testing a hardware description on an FPGA development board.
- Understand how a seven-segment display functions.
- Understand the role of multiplexing in the seven-segment display module.