HOCHSCHULE
RHEIN-WAAL
Rhine-Waal University
of Applied Sciences

LABORATORY 4

Model-based Hardware Design

Stopwatch

Lab notes prepared by Mr. Gligor Duchev

Lab notes revised by: Prof. Andy Stamm

Contents

1. Safety instructions

ESD Warning

Caution Although this product has been designed to be as robust as possible, ESD (Electrostatic Discharge) can damage or upset this product. This product must be protected at all times from ESD. Static charges may easily produce potentials of several kilovolts on the human body or equipment, which can discharge without detection. Industry-standard ESD precautions must be employed at all times.

The NI Digital Electronics FPGA Board is designed and intended for use as a development platform for hardware or software in an educational/professional laboratory environment. To facilitate usage, the board is manufactured with its components and connecting traces openly exposed to the operator and the environment. As a result, ESD sensitive (ESDS) components on the board, such as the semiconductor integrated circuits, can be damaged when exposed to an ESD event. To indicate the ESD sensitivity of the NI Digital Electronics FPGA Board, it carries the symbol shown below.

Handling the NI Digital Electronics FPGA Board can damage the board components if ESD prevention measures are not applied. **Before handling or setup, equalize your potential with the board by touching one of the integrated ESD discharge pads.** During all handling and setup, ESD prevention measures must be applied. In addition, the NI Digital Electronics FPGA Board should be handled by the edges. Touching exposed circuits, components or connectors could result in an ESD event.

2. Introduction to Seven-Segment Display

The NI Digital Electronics FPGA Board has a two digit seven-segment display, DISP1, in **a common cathode configuration**. Try locating the seven-segment display on your FPGA board, otherwise refer to the first laboratory. The two digit seven-segment display circuitry is shown in Figure 1.

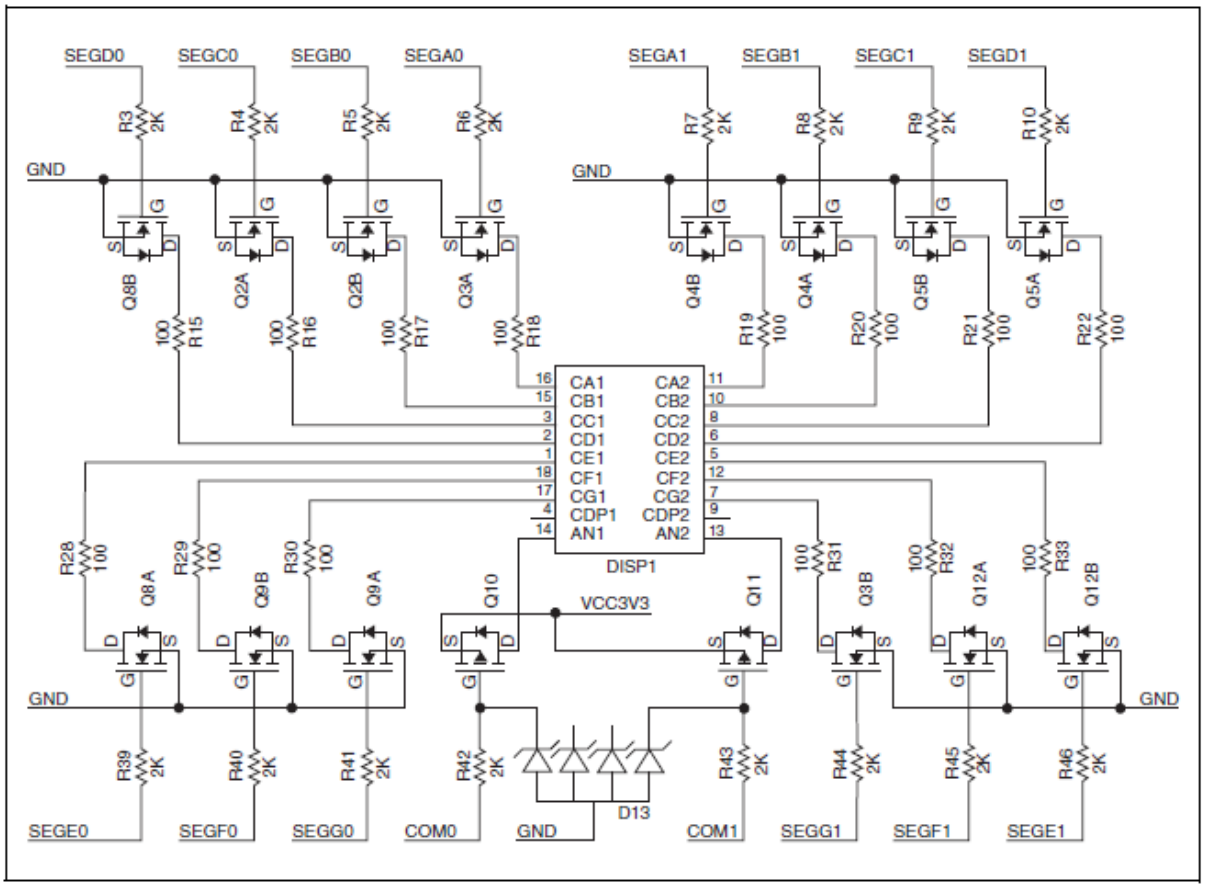

Figure 1: Two Digit Seven-Segment Display Circuit Diagram

Each digit is composed of seven segments arranged in a figure 8 pattern, with an LED embedded in each segment. Segment LEDs are SEG*x*0 for digit 0, and SEG*x*1 for digit 1, as shown in Figure 2.

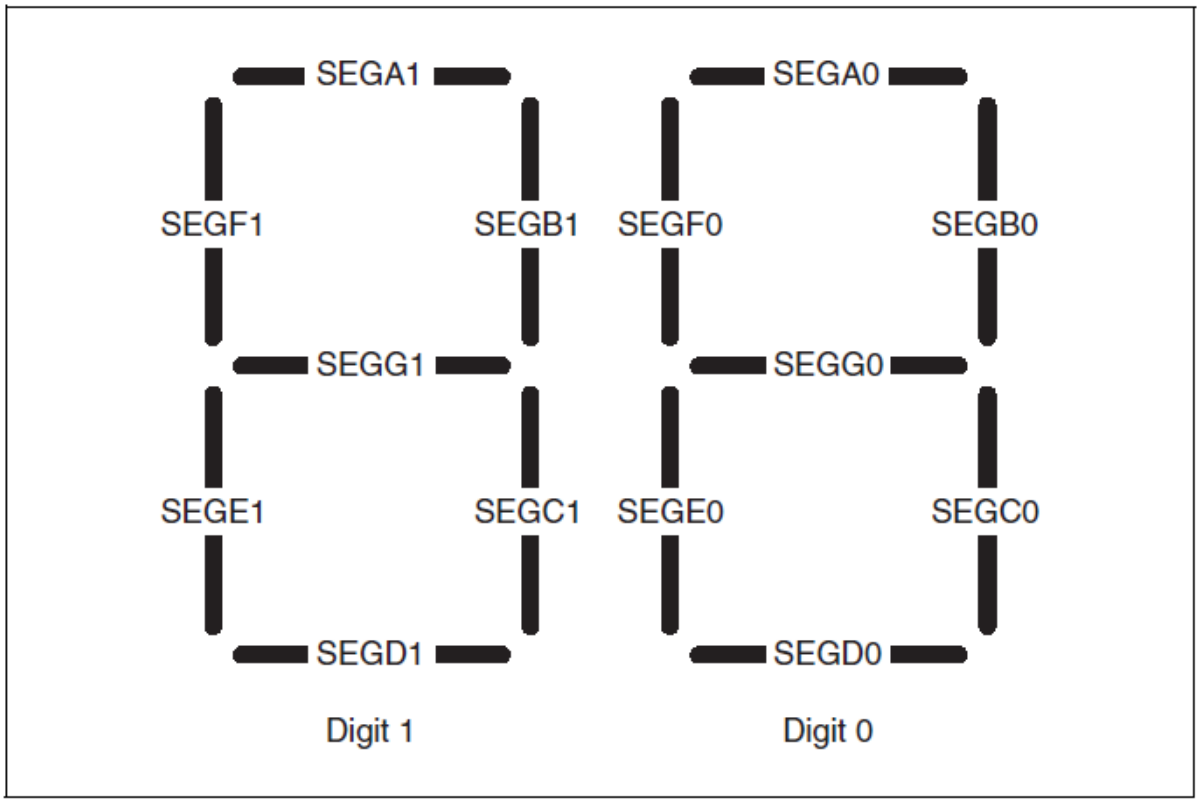

Figure 2. Seven-Segment Display

Segment LEDs can be individually illuminated, so different patterns can be displayed on a digit by lighting certain LED segments. To light an individual LED segment, drive the associated line High (using 3.3 V or 5 V).

Lines COM0 and COM1 can be used to enable/disable each digit of the display to allow using the display in multiplexed mode. (**low = enable**). Both lines are connected to the FPGA.

3. UCF File Constraints

Slide Switches

The UCF file constraints for the eight slide switches, SW0 to SW7, are listed as follows. SWx refers to the slide switch line, LOC indicates the FPGA line location, and IOSTANDARD is the I/O standard used.

```
Net "SW0" LOC="J11" | IOSTANDARD = LVCMOS33;
Net "SW1" LOC="J12" | IOSTANDARD = LVCMOS33;
Net "SW2" LOC="H16" | IOSTANDARD = LVCMOS33;
Net "SW3" LOC="H13" | IOSTANDARD = LVCMOS33;
Net "SW4" LOC="G12" | IOSTANDARD = LVCMOS33;
Net "SW5" LOC="E14" | IOSTANDARD = LVCMOS33;
Net "SW6" LOC="D16" | IOSTANDARD = LVCMOS33;
```

Seven-Segment Displays

The UCF file constraints for the seven-segment displays are listed as follows: SEGxx refers to the display segment line, COMx refers to the display anode line, LOC indicates the FPGA line location, IOSTANDARD is the I/O standard used, SLEW refers to the slew rate, the maximum rate of change of a signal, and DRIVE indicates the current drive strength on the FPGA in milliamps.

```
Net "SEGA0" LOC="E3" | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 8;
Net "SEGB0" LOC="E1" | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 8;
Net "SEGC0" LOC="G5" | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 8;
Net "SEGD0" LOC="D1" | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 8;
Net "SEGE0" LOC="E4" | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 8;
Net "SEGF0" LOC="C1" | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 8;
Net "SEGG0" LOC="C2" | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 8;
Net "COM0" LOC="B2" | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 8;
Net "SEGA1" LOC="H6" | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 8;
Net "SEGB1" LOC="K2" | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 8;
Net "SEGC1" LOC="H3" | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 8;
Net "SEGD1" LOC="K1" | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 8;
Net "SEGE1" LOC="G4" | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 8;
Net "SEGF1" LOC="J2" | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 8;
Net "SEGG1" LOC="G3" | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 8;
Net "COM1" LOC="G2" | IOSTANDARD = LVCMOS33 | SLEW = SLOW | DRIVE = 8;
```

4. Task 1: Stopwatch

Design a stopwatch that counts seconds and displays them on the 2 digit SSD. It should have a start button which starts the counting and a restart button which returns the count to 0. When the stopwatch reaches 99, the display should start blinking. Verify the programs functionality on the prototyping board.

5. Bonus Task 2: Fireworks

Design a state machine which repeats a simple 3 state pattern on the SSD at a rate of 60Hz. Verify the programs functionality on the prototyping board.

6. Bonus Task 3: Traffic lights

Simulate a simple traffic light system on a four-way intersection. This should be done by sequencing states for a certain amount of time. One of the streets is going north-south and the other east-west. To simulate the traffic lights use led a,g and d on the seven segment displays.

a) Create a program which uses counter variables to alternate between the states provided in table 1.
b) Add two buttons (for pedestrians) which when pressed cuts the delay of the respective green light down to 5 seconds.

Table 1: LED states

State	North-South	East-West	Delay (sec.)
0	Green(d)	Red(a)	10
1	Yellow(g)	Red(a)	2
2	Red(a)	Red(a)	2
3	Red(a)	Green(d)	10
4	Red(a)	Yellow(g)	2
5	Red(a)	Red(a)	2