

Introduction to the Xilinx Foundation Design Environment

You will learn how to use the Xilinx Foundation software to enter, simulate, and implement a simple design. You will also learn how to program the XESS logic board and how to control its inputs and observe its outputs.

1.0 Overview

The main goal of this first experiment is to familiarize you with the lab setup and to give you your first exposure to the Xilinx Foundation CAD software and the XESS programmable logic board. You will perform all of the design stages outlined in the [EECS 270 Laboratory Overview](#) and can expect to learn how to enter a design using the Schematic and HDL Editors, how to simulate it functionally, how to implement it on the logic board, how to control its inputs using the software and hardware switches, and how to observe its outputs on the LED indicators.

2.0 Preparation

This being your first lab, the actual design work required is quite minimal. However, you need to do quite a bit of reading in order to make this a useful exercise. Read the following before you do the lab, and refer to it as needed later:

1. The [EECS 270 Laboratory Overview](#) document. This is a good starting point and a useful reference for data that you'll frequently need.
2. Chapter 1 of Wakerly. This gives yet one more exposition of what digital design is. It also discusses the X74-157 4-bit 2-way multiplexer that is the primary component used in this design.

FIGURE 1.

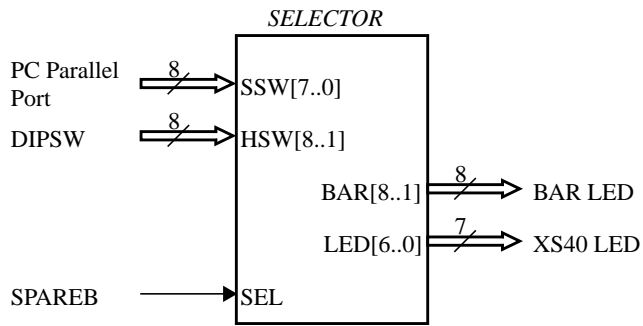
Top level schematic of
SELECTOR

TABLE 1.

Required input/output
mapping

PC Port	DIP Switches	BAR LED	XS40 LED
D0	DIPSW1	DB1	S0
D1	DIPSW2	DB2	S1
D2	DIPSW3	DB3	S2
D3	DIPSW4	DB4	S3
D4	DIPSW5	DB5	S4
D5	DIPSW6	DB6	S5
D6	DIPSW7	DB7	S6
D7	DIPSW8	DB8	None

3.0 Design Specification

The top-level schematic of the circuit you need to design is shown in Figure 1. Named *SELECTOR*, it has two 8-bit inputs connected, respectively, to the PC parallel port and to the bank of DIP switches. It also has an input connected to the SPAREB push button. It has an 8-bit output connected to the bargraph LED, and a 7-bit output connected to the XS40 LED. *SELECTOR* should behave as follows:

- When the SPAREB push button is depressed, the 8-bit word specified by the settings of the DIP switches should be displayed on both the BAR LED and the XS40 LED.
- When the SPAREB push button is not depressed, the 8-bit word on the PC port should be displayed on both the BAR LED and the XS40 LED.

The mappings between input and output bits should be as shown in Table 1. (Note that bits D7 and DIPSW8 are not mapped to the XS40 LED because it only has 7 segments!) The LED segment corresponding to a particular input should glow when that input is 1, and should dim when that input is 0.

4.0 Design Notes and Hints

- Despite the fact that the top-level schematic is drawn using busses, you don't have to use busses in your design; simply use single-bit wires. We'll have plenty of practice with busses later on.
- Deep in the XESS board documentation you'll discover that the DIP and push button switches are active low: the ON position of the DIP switches and the depressed position of the push button switches correspond to applying logic 0 input signals.

FIGURE 2. Schematic symbol of the X74_157 multiplexer

- Note that the DIP switches and BAR LEDs are numbered 1 to 8 from left to right; the PC port bits, on the other hand, are numbered 7 to 0 from left to right.
- Even though we have yet to study multiplexers in detail (we covered them somewhat in the overview lecture), it is not too hard to realize the functionality of *SELECTOR* with a pair of X74-157 4-bit 2-way multiplexers. This component is available as a library primitive that can be instantiated in the Schematic Editor (see Figure 2.) It has two sets of 4-bit inputs (labeled A1-A4 and B1-B4), a 4-bit output (Y1-Y4), a select input S, and an active-low enable input G. When enabled ($G = 0$), this component behaves according to the following equation:

$$Y_i = \begin{cases} A_i & \text{if } S = 0 \\ B_i & \text{if } S = 1 \end{cases} \quad (1)$$

5.0 Deliverables

5.1 Pre-Lab

1. Hardcopy of *SELECTOR*'s schematic and UCF files. Insert your name, student ID number, and unique name on the schematic by clicking the Table Setup option on the File menu. Note that schematics look best when printed in portrait format.
2. Hardcopy of *SELECTOR*'s simulation results showing its response on the appropriate BAR and LED outputs to all possible combinations of the SPAREB, D3, and DIPSW4 inputs. This can be conveniently done by connecting the stimulator counter bits B2, B1, and B0 to SPAREB, D3, and DIPSW4, respectively. Set the period of B0 to 10ns and print a simulation trace from 0 to 40ns (click the Page Setup option on the File menu). Write your name, student ID number, and unique name on the simulation print-out (note that, unlike the Schematic Editor, the simulator does not allow you to type your name on the printed sheet).

3. For a few extra bonus points, you may want to design *SELECTOR* from primitive gates or by writing an ABEL program. You will have to take the initiative to read ahead and figure out how you would accomplish this. This part is completely voluntary, though.

5.2 In-Lab

Generate the .bit file for *SELECTOR*, download it to the XESS board, and verify that your implementation is working properly. When you are satisfied that you have a correctly-functioning circuit, demonstrate its operation to your lab GSI and have him sign your experiment's cover sheet.

5.3 Post-Lab

Prepare your lab report as described in the [EECS270 Laboratory Overview](#) handout. Make sure you complete and include all parts of the report including the *Cover Sheet*, the *Design Narrative* section, and the *Design Documentation* section. In the *Post-Lab Questions* section, provide answers to the following questions:

1. If you were to construct a truth table for *SELECTOR*, how many rows would it have? How many (input and output) columns?
2. How does DB5 depend on D2? How does it depend on SPAREB?
3. Assuming that $D[7..0] = 0x3D$ and $DIPSW[8..1] = 0xEA$ (these are hexadecimal values), which BAR LED and XS40 LED segments remain lit regardless of whether the SPAREB button is pushed or not?
4. What should the PC port and DIPSW inputs be set to in order to display the decimal digits 0 and 1 on the XS40 LED when SPAREB is 0 and 1 respectively?