Lab 07

Part B – Gate-Based 8:3 Priority Encoder

Description

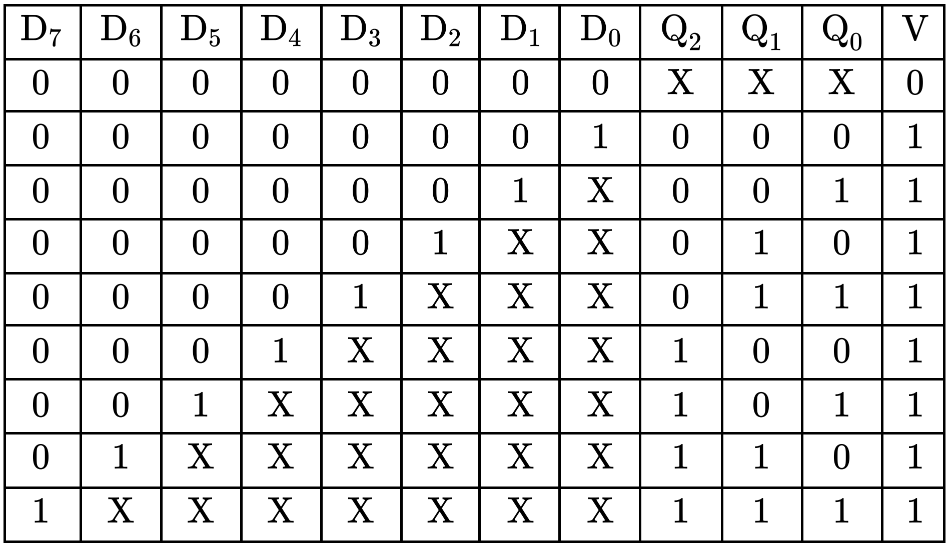
In this lab module, you will learn about the gate-level implementation of an 8:3 priority encoder, building upon the concepts covered in the lecture section. The gate-level implementation is essentially a direct conversion of the 8:3 priority encoder's boolean equations to VHDL.

Note: for this lab module, you will need to connect a USB-A to Micro-USB cable from your computer to the FPGA development board. If you are using a laptop that does not have a USB-A port, ensure that you have a USB-C to USB-A adapter at this time. Otherwise, have a partner proceed from here, or use a lab PC.

Procedure

Background

- Recall the truth table for the 8:3 priority encoder from the lecture:



D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Q_2	Q_1	Q_0	V
0	0	0	0	0	0	0	0	X	X	X	0
0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	0	1	X	0	0	1	1
0	0	0	0	0	1	X	X	0	1	0	1
0	0	0	0	1	X	X	X	0	1	1	1
0	0	0	1	X	X	X	X	1	0	0	1
0	0	1	X	X	X	X	X	1	0	1	1
0	1	X	X	X	X	X	X	1	1	0	1
1	X	X	X	X	X	X	X	1	1	1	1

- Also note the boolean expressions for this table:

$$
\begin{aligned}
Q_2 &= D_7 + D_6 + D_5 + D_4 \\
Q_1 &= D_7 + D_6 + \overline{D_5}\,\overline{D_4}\,D_3 + \overline{D_5}\,\overline{D_4}\,D_2 \\
Q_0 &= D_7 + \overline{D_6}\,D_5 + \overline{D_6}\,\overline{D_4}\,D_3 + \overline{D_6}\,\overline{D_4}\,\overline{D_2}\,D_1
\end{aligned}
$$

$$V = D_7 + D_6 + D_5 + D_4 + D_3 + D_2 + D_1 + D_0$$

- If you want an extra challenge, try writing out these equations in VHDL before looking at `lab07b.vhd`.

Project Creation

1. Download `lab07b.vhd` and `lab07b.xdc` from Canvas and place them in a new folder titled `lab07b`.
2. 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

3. 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 `lab07b.vhd` file.
4. `Open Elaborated Design`, review the schematic alongside the instructor, and take screenshots for your report.
5. 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.
6. Before proceeding, plug the FPGA development board into your computer using the provided Micro-USB to USB-A cable.

7. At the top, in the green banner (or under the `Open Hardware Manager` dropdown) click `Open Target`, then `Auto Connect`.
8. 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.
9. Click `Program` to upload the bitstream to the FPGA.

Hardware Testing and Analysis

10. Using the eight rightmost switches as the input to the priority encoder, test several different cases and observe the output.
    - The rightmost 3 LEDs are the primary output of the 8:3 priority encoder ($\mathrm{Q}_2\mathrm{Q}_1\mathrm{Q}_0$), while the leftmost LED is the valid bit (V).
    - Take at least three pictures showing different input cases. One of these should be the all zeroes input.
11. Verify the functionality. Does the behavior of the hardware match the truth table shown above?
12. Why is the valid bit (V) necessary?

Deliverables

- Include as part of your **formal report**:
  - A screenshot of the Elaborated Design schematic (Step 4)
  - A screenshot of the FPGA resource utilization (LUTs, FFs) (Step 5)
  - 3 pictures of the FPGA development board (Step 10)
  - Answer to (12)

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 what a priority encoder does and how it functions.
- Understand the gate-level implementation of an 8:3 priority encoder.