Laboratory 3: Combinational MSI Circuits

Learning Objectives

- Learn about encoders and decoders
- Synthesize and analyze encoders and decoders

In this lab, you design and implement circuits using a few medium scale ICs (MSIs). Your circuit will take a 3-bit binary code as input, transform it into Gray code, and decode it back to binary code.

The theoretical background required for this lab are

- Decoders
- Encoders
- Multiplexers
- Exclusive ORs

Gray Code

Gray code has the characteristics that only one bit position changes between two consecutive decimal numbers represented by the code. The 3-bit Gray code is defined as

Table 1. Decimal, Binary and Gray code conversions

Decimal	Binary			Gray		
	B2	B1	B0	G2	G1	G0
0	0	0	0	0	0	0
1	0	0	1	0	0	1
2	0	1	0	0	1	1
3	0	1	1	0	1	0
4	1	0	0	1	1	0
5	1	0	1	1	1	1
6	1	1	0	1	0	1
7	1	1	1	1	0	0

The advantage of Gray code is that it reduces the probability of false outputs caused by multiple bit changes.

For this lab, you will design your binary/ Gray encoder/ decoder first, and then implement it in hardware.

The theoretical background required for this lab are

- Boolean algebra
- Karnaugh maps
- Multiplexers
- Decoders
- Exclusive ORs

1. Pre-Lab

Note that you need to submit individualized pre-lab report for this lab.

For the prelab, you are asked to synthesize, simulate and analyze the encoder/decoder functions. Additionally you are asked to use Vivado, and present the Register Transfer Logic (RTL) diagram and the timing diagram with all the possible combinations for the inputs.

1.1 Synthesis and Simulation of a 3-bit Gray Encoder

The encoder part of your circuit will be realized using an 8-to-1 multiplexer (MUX) for each of the three Gray code bits. The eight input lines of a MUX have to be connected to logic '0' or '1' based on the code conversion in **Table 1**. The three input binary bits (B2, B1, B0) control the selection process.

1. Draw the logic diagram and include in the prelab report.
2. Create a Verilog file (*.v) and write the code for implementing a 8-to-1 multiplexer. Include this code in your prelab report.
3. Use the 8-to-1 multiplexer to implement each of three Gray code bits. Write the code for synthesizing the 3-bit encoder. Include this code in your prelab report.
4. Open the elaborated design under the RTL analysis tab. Submit the Elaborated design schematic diagram with your prelab report. This diagram will contain the schematic for the encoder. Does this match your initial design in step 1?
5. Create a simulation file in Vivado and write the code for running the simulation for each possible combination of the binary inputs.
6. Simulate the testbench and obtain the timing diagram which demonstrates each of the possible input combination along with the corresponding output. Include this in your prelab report.

1.2 Synthesis and Simulation of a 3-bit Gray Decoder

To design the decoder, you must first find the expression for the binary code bits (B2,Bl,B0) as

functions of the three Gray code bits (G2,Gl,G0).

1. Derive the Karnaugh map for the 3-bit Gray Decoder using Table 1. Include this in your prelab report.

2. Find the minimal sum-of-product for the decoder circuit. Include this in your prelab report. How many gates are used here? Discuss advantages of reducing the number of ICs/logic gates when implementing digital functions. Include these discussions in your prelab report.

3. Using Boolean algebra, prove that the decoder circuit can be implemented using the XOR gates as specified below. Include these in your prelab report.

   $B2 = G2$

   $B1 = G2 \oplus G1$

   $B2 = G2 \oplus G1 \oplus G0$

Bonus:

1. Write the code for synthesizing the 3-bit decoder in Vivado. Include this code in your prelab report.

2. Open the elaborated design under the RTL analysis tab. Submit the Elaborated design schematic diagram with your prelab report. This diagram will contain the schematic for the decoder.

3. Create a simulation file in Vivado and write the code for running the simulation for each possible combinations of the binary input.

4. Simulate the testbench and obtain the timing diagram which demonstrates each of the possible input combinations along with the corresponding output. Include this in your report.

2. Experiments

2.1 Synthesizing the 3-bit Gray Encoder

1. Build the ICOD unit that you used in Lab1. Check that the ICOD on your breadboard is working properly, by toggling each of the input pins from logic ‘0’ (GND) to logic ‘1’ (5V), and from logic ‘1’ (5V) to logic ‘0’ (GND).

2. Implement the encoder part of the circuit using the ICs specified below:

   74LS86A: quad 2-input XOR

   74LS151A: 8-input, 1-bit (8-to-1) Multiplexer (MUX)

3. Verify that the circuit follows the conversion table using the 3 LED's on the ICOD unit. Show your TA the circuit and verify some truth table values in his/her presence.

2.2 Synthesizing the 3-bit Gray Decoder

1. Implement the decoder part of the circuit using the Exclusive OR gates. Connect the binary outputs (B2, B1, B0) to the 3 LEDs of the ICOD unit.

2. Verify that the circuit follows the conversion table using the 3 LED's on the ICOD unit. Show your TA the circuit and verify some truth table values in his/her presence.

3. Lab 3 Report

Your final report for Lab 3 should include the following:

- your individual prelab reports, with all the materials requested

- a section describing what Gray codes are, and examples of their applications (e.g., in digital communications)

- images of the circuits that you implemented to realize the encoder and decoder

- include a table, describing the role of each individual in performing of the lab experiments and writing of the final lab report

Resources for Verliog and Vivado

The following resources have been posted on Canvas :

- Vivado Tutorial.pdf
- Vivado-How to Run.pdf
- FPGA Compiler.pdf
- Intro to Verilog.pdf
- Verilog_Reference_Guide.pdf