University of Windsor

School of Computer Science
Faculty of Science

COMP-2650: Computer Architecture I: Digital Design
Fall 2020

Lab#	Date	Title	Due Date	Grade Release Date
Lab 09	Nov 23-25, 2020	**L09: Binary Codes (Gray Code)**	Dec. 02, 2020 + Extension Wednesday Midnight AoE	Dec. 09, 2020

The 9th lab's objectives will be to master the topics in logic circuit design by implementing the algorithms with a programming language, herein, C/C++.

Deadline: According to the school's policy, no assignment can be due on the last week of class or after. Also, we are not able to remove the assignments for the last two weeks since it changes the course outline for the marking schema which is not possible at this time of semester. This conflict of policies happens because we extend the deadlines for the labs for two weeks in order to reduce the workload. So, here is the solution:

- The official due date is what mentioned above: Dec. 02, 2020 Wednesday Midnight AoE.
- There is an extension of 1 week to the official deadline for all students. Therefore, we accept submission till Dec. 09, 2020 Wednesday Midnight AoE.
- The difficulty of this week and next week's assignments are in easy level that students are able to do the assignments within the official due date.

Step 1. Environment Setup

Our programming environment is the same as the first lab (Lab 01). In this lab, we want to implement Gray code, named after Frank Gray[1]. This code is an ordering of the binary numeral system such that two successive values differ in only one bit. For example, the Gray codes for the decimal numbers 4 and 5 are 0110 and 0111 where the only change is the first bit. To implement Gray code, we have to first convert a given decimal number to binary number. Then, we follow the below steps:

Moving from the highest significant bit to the lowest significant bit (last bit to the first bit)

1) The last bit of the Gray code is the same as the last bit of the binary number
2) The i-th bit of the Gray code is the XOR of the i-th and (i+1)-th bits of the binary number

For instance, the Gray code for 20 is:

[1] https://en.wikipedia.org/wiki/Frank_Gray_(researcher)

$(20)_{10}$	Binary Number	1	0	1	0	0
	Gray Code					
$(20)_{10}$	Binary Number	1	0	1	0	0
	Gray Code	1				
$(20)_{10}$	Binary Number	1	0	1	0	0
	Gray Code	1	1⊕0=1			
$(20)_{10}$	Binary Number	1	0	1	0	0
	Gray Code	1	1	0⊕1=1		
$(20)_{10}$	Binary Number	1	0	1	0	0
	Gray Code	1	1	1	1⊕0=1	
$(20)_{10}$	Binary Number	1	0	1	0	0
	Gray Code	1	1	1	1	0⊕0=0

Lab Assignment

You should implement the above algorithm under the name of a project COMP2650_Lab09_{UWinID} that asks for a decimal number:

```
Enter a decimal number:20
```

When the user enters a decimal number, the program should output a menu of commands as follows:

```
    Enter the encoding command number:
    0) Exit
    1) Gray code
```

If a user selects (1), the program should print out the Gray code for the given decimal number as shown below:

```
Gray code for 20 -> 11110
```

If the user selects (0), the program ends. Please restrict the user to enter inputs within the range [0,2^7-1=127] for the decimal number. For instance, if the user enters -1, 999, print out an error message and come back to ask for correct inputs.

It is required to write a *modular* program. Please put the part of the code that outputs the Gray code in a new function called `to_Gray()` inside the main.c file.

Deliverables

You will prepare and submit the program in one single zip file `COMP2650_Lab09_{UWinID}.zip` containing the following two items:

1. The entire project folder `COMP2650_Lab9_{UWinID},` including the code (source) files and executable file.
2. The result of the commands in the file `COMP2650_Lab09_Results_{UWinID}.jpg/pdf`. Simply make a screenshot of the results and save it. If multiple images, please print them all into a single pdf file.
3. *[Optional and if necessary]* A lab report document in the PDF file `COMP2650_Lab08_Report_{UWinID}.pdf`.

It should include:

a. Your name, UWinID, and student number
b. One paragraph describes the program that you attached, along with any prerequisites needed to build and run the program. *Please note that if your program cannot be built and run on our computer systems, you will lose marks.*

In sum, your final zip file for the submission includes 1 folder (entire project folder), 1 image/pdf (results snapshot), and 1 pdf (report). *Please follow the naming convention as you lose marks otherwise.* Instead of {UWinID}, use your own UWindsor account name, e.g., mine is hfani@uwindsor.ca, so,

```
COMP2650_Lab08_hfani.zip
    –   COMP2650_Lab09_hfani
            o   src
                    ▪   ... (whatever source/header files required to build the program)
                    ▪   main.cpp
            o   COMP2650_Lab09_hfani [.exe in MS-Windows]
    –   COMP2650_Lab09_Report_hfani.pdf
    –   COMP2650_Lab09_Results_hfani.jpg or COMP2650_Lab09_Results_hfani.pdf
```