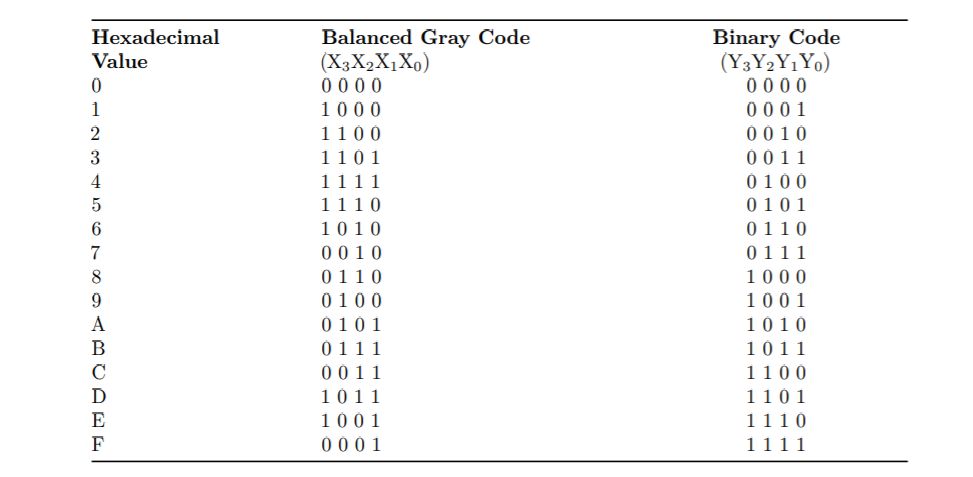
Gasser Ahmed

[gasser18@vt.edu](mailto:gasser18@vt.edu)

ECE 5484, Project 1

Section 1 – Objectives:

Design a combinational logic circuit that displays the hexadecimal value of a gray code input according to the specifications below: 

By following the steps below:

1. Create truth table by reordering the table given above.
2. Create SOP Boolean logic expressions for all four outputs Y0, Y1, Y2, andY3 from the truth table.
3. Simplify those expressions using K-Maps.
4. Design the circuit layout for the simplified expressions using Digital.
5. Debug and test the full system by observing specified versus actual hexadecimal outputs for different input combinations.

Section 2 – Truth Table:

|  |  |  |
| --- | --- | --- |
| Hexadecimal  Value | Balanced Gray Code  (X3 X2  X1  X0) | Binary Code  (Y3 Y2 Y1 Y0) |
| 0 | 0 0 0 0 | 0 0 0 0 |
| F | 0 0 0 1 | 1 1 1 1 |
| 7 | 0 0 1 0 | 0 1 1 1 |
| C | 0 0 1 1 | 1 1 0 0 |
| 9 | 0 1 0 0 | 1 0 0 1 |
| A | 0 1 0 1 | 1 0 1 0 |
| 8 | 0 1 1 0 | 1 0 0 0 |
| B | 0 1 1 1 | 1 0 1 1 |
| 1 | 1 0 0 0 | 0 0 0 1 |
| E | 1 0 0 1 | 1 1 1 0 |
| 6 | 1 0 1 0 | 0 1 1 0 |
| D | 1 0 1 1 | 1 1 0 1 |
| 2 | 1 1 0 0 | 0 0 1 0 |
| 3 | 1 1 0 1 | 0 0 1 1 |
| 5 | 1 1 1 0 | 0 1 0 1 |
| 4 | 1 1 1 1 | 0 1 0 0 |

Section 3 – Logic Expressions:

Y0 = X0X’1X’2X’3 + X’0X1 X’2X’3 + X’0X’1X2 X’3 + X0X1X2X’3 + X’0X’1X’2X3 + X0X1X’2X3 + X0X’1X2X3 + X’0X1X2X3

K-Map:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X3X2\X1X0 | X’1X’0 | X’1X0 | X1X0 | X1X’0 |
| X’3X’2 | 0 | 1 | 0 | 1 |
| X’3X2 | 1 | 0 | 1 | 0 |
| X3X2 | 0 | 1 | 0 | 1 |
| X3X’2 | 1 | 0 | 1 | 0 |

No Reduction

Y1 = X0X’1X’2X’3 + X’0X1X’2X’3 + X0X’1X2X’3 + X0X1X2X’3 + X0X’1X’2X3 + X’0X1X’2X3 + X’0X’1X2X3 + X0X’1X2X3

K-Map:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X3X2\X1X0 | X’1X’0 | X’1X0 | X1X0 | X1X’0 |
| X’3X’2 | 0 | 1 | 0 | 1 |
| X’3X2 | 0 | 1 | 1 | 0 |
| X3X2 | 1 | 1 | 0 | 0 |
| X3X’2 | 0 | 1 | 0 | 1 |

After reduction Y1 = X’1X0 + X’2X1X’0 + X’3X2X0 + X3X2X’1

Y2 = X0X’1X’2X’3 + X’0X1 X’2X’3 + X0X1X’2X’3 + X0X’1X’2X3 + X’0X1X’2X3 + X0X1X’2X3 + X’0X1X2X3 + X0X1X2X3

K-Map:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X3X2\X1X0 | X’1X’0 | X’1X0 | X1X0 | X1X’0 |
| X’3X’2 | 0 | 1 | 1 | 1 |
| X’3X2 | 0 | 0 | 0 | 0 |
| X3X2 | 0 | 0 | 1 | 1 |
| X3X’2 | 0 | 1 | 1 | 1 |

After reduction Y2 = X’2X0 + X’2X1 + X3X1

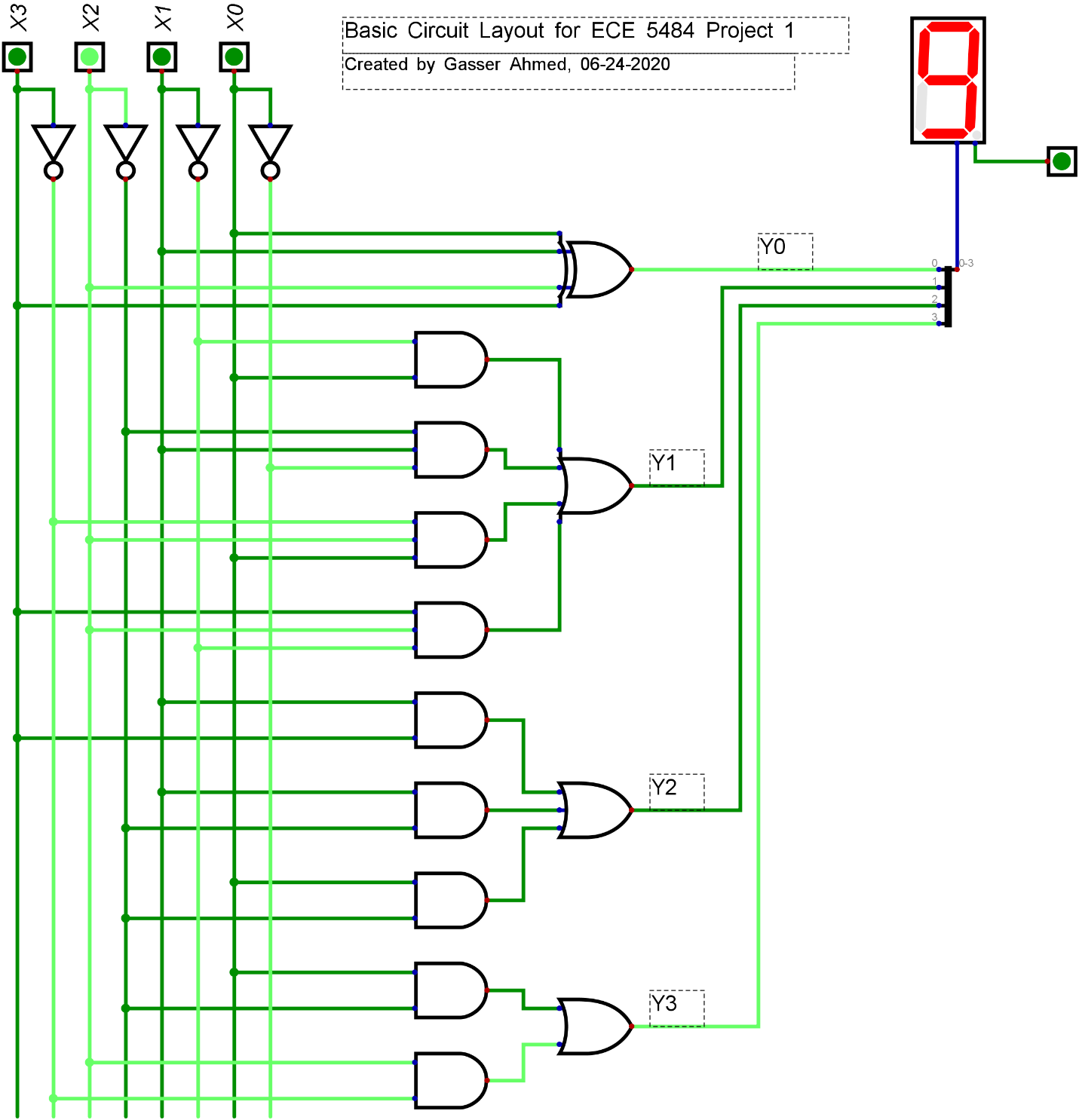
Y3 = X0X’1X’2X’3 + X0X1X’2X’3 + X’0X’1X2 X’3 + X0X’1X2X’3 + X’0X1X2X’3 + X0X1X2X’3 + X0X’1X’2X3 + X0X1X’2X3

K-Map:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X3X2\X1X0 | X’1X’0 | X’1X0 | X1X0 | X1X’0 |
| X’3X’2 | 0 | 1 | 1 | 0 |
| X’3X2 | 1 | 1 | 1 | 1 |
| X3X2 | 0 | 0 | 0 | 0 |
| X3X’2 | 0 | 1 | 1 | 0 |

After reduction Y3 = X’2X0 + X’3X2

Section 4 – Circuit Design:



Section 5 – Conclusions:

After designing/simulating the circuit in Digital and testing all different 16 input combinations, I was able to get the correct hexadecimal output for each corresponding input combination. However, in the beginning, I was testing all input combinations, but I was getting wrong outcomes for inputs 1100 and 1110. So, I kept undoing any changes made for the starter circuit for each output Y1­, Y2­, and Y3 until I noticed that those 2 input combinations started to give wrong hexadecimal outcomes when I added the circuit for Y3, so I concluded that there should had been something wrong with logic expression for that output Y3. So, when I redid the K-Map for Y3, I noticed it the current K-Mapwas wrong compared to the new one which accordingly led to a wrong simplified expression that caused the wrong outcome. Then, after updating the simplified logic expression to match the new K-Map, I tested again all input combinations and I was able to get the correct hexadecimal output for each corresponding input combinations including 1100 and 1110.

So, I learned that testing and debugging all different input combinations is important to make sure the expected outcome is always correct. Also, I learned that I should always doublecheck that I have the correct K-Maps and logical expressions before I start simulating the circuit design in Digital to avoid any conflicts or confusions during testing.

Lastly, the approximate number of hours I devoted to the project was about 8-12 hours. In general, the project was an interesting, learning, and enjoyable experience. Also, the professor’s explanations for the project was very helpful and explained a lot of unclear items/objectives.