Victor Yuan

September 29, 2016

Lab Report 5

Introduction:

The purpose of this lab was to create a truth table for a arithmetic function in this case adding the number value 3 to BCD numbers using only logic gates. Then implement the function into hardware with the minimal usage of gates.

Team Member Responsibilities:

This lab was done alone.

List of Materials:

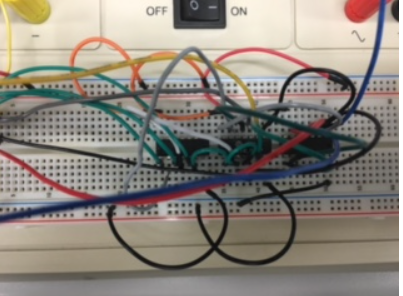
ELENCO Analog kit assorted wires and cables 74LS08 Quad 2-Input AND

74LS04 Hex Inverters 74LS32 Quad 2-Input OR

Procedure:

A truth table for all the combinations of the variables W, X, Y, Z, was derived. Each combination stood for a number between 0 and 15. The number three was then added to each of these numbers and the output was converted into BCD format these numbers ranged from 3 to 12 all numbers after 12 where labeled as x’s (don’t cares). Karnaugh maps were used to simplify these truth tables.

Then we were given 5 logic gate chips however for this design only the 74LS08 Quad 2-Input AND, 74LS04 Hex Inverters, and the 74LS32 Quad 2-Input OR chips were needed. Our truth table that we derived and simplified earlier with the Karnaugh maps were then implemented with the task of trying to limit the amount of gates used.



Questions:

I. Theoretical Design:

1. Show the truth-table for your design.

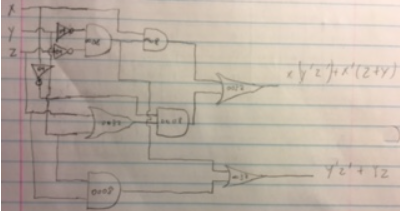
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | w | x | y | z | a | b | c | d |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 2 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 |
| 3 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| 4 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| 5 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 |
| 6 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |
| 7 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 |
| 8 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| 9 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 10 | 1 | 0 | 1 | 0 | x | x | x | x |
| 11 | 1 | 0 | 1 | 1 | x | x | x | x |
| 12 | 1 | 1 | 0 | 0 | x | x | x | x |
| 13 | 1 | 1 | 0 | 1 | x | x | x | x |
| 14 | 1 | 1 | 1 | 0 | x | x | x | x |
| 15 | 1 | 1 | 1 | 1 | x | x | x | x |

2. Show the Karnaugh maps used for the minimal SOP realizations.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| wx |  |  |  |  |  |  |
| output A | 00 | 01 | 11 | 10 |  |  |
| 00 | 0 | 0 | x | 1 |  |  |
| 01 | 0 | 1 | x | 1 | sop | w + xz+x |
| 11 | 0 | 1 | x | x |  |  |
| 10 | 0 | 1 | x | x |  |  |
| wx |  |  |  |  |  |  |
| output B | 00 | 01 | 11 | 10 |  |  |
| 00 | 0 | 1 | x | 0 | SOP | xy'z' + x'z + x'y |
| 01 | 1 | 0 | x | 1 |  |  |
| 11 | 1 | 0 | x | x |  |  |
| 10 | 1 | 0 | x | x |  |  |
| wx |  |  |  |  |  |  |
| output C | 00 | 01 | 11 | 10 | sop | y'z' + yz |
| 00 | 1 | 1 | x | 1 |  |  |
| 01 | 0 | 0 | x | 0 |  |  |
| 11 | 1 | 1 | x | x |  |  |
| 10 | 0 | 0 | x | x |  |  |
| wx |  |  |  |  |  |  |
| output D | 00 | 01 | 11 | 10 |  |  |
| 00 | 1 | 1 | x | 1 | sop | z' |
| 01 | 0 | 0 | x | 0 |  |  |
| 11 | 0 | 0 | x | x |  |  |
| 10 | 1 | 1 | x | x |  |  |

II. Constrained Hardware Realization

1. Provide a schematic diagram for your constrained hardware realization.



2. Describe briefly all alternate realization technique you used?

For three input and gates I used two two-input and gates to group three variables together. To minimize the number of gates I grouped a AND, OR, AND gate sequence to a AND, OR gate sequence. Ex[x’z + x’y = x’(z+y)]

3. Indicate all common hardware (common terms) used in your design.

I used 4 AND gates on the 74LS08 Quad 2-Input chip, and 3 OR gates on the 74LS32 Quad 2-Input chip, and 3 invertor gates on the 74LS04 Hex Inverters.

III. Hardware Implementation:

1. Demonstrate to the Lab TA that your hardware design operates correctly.

2. Describe briefly how you verified the correct functioning of your circuit.

I wired the input to three logic gates that opened and closed to symbolize the value of X, Y, Z, 1’s and 0’s. I tested each sequence with the expected output from the truth table of output B and out put C

Conclusion:

1. Discuss the ease/difficulty of realizing circuit with hardware constraints.

The simplification was easy because there were only two simple adjustments that was needed to satisfy the 5 total gate requirement for output B, and breaking down the three input and gate into two two-input AND gate.

1. Discuss any debugging techniques that you may have used.

I followed the wires, trying to see where the connection issues were.