**Department of Computer Engineering**

BLG 242E  
Digital Circuits Laboratory Experiment Report

Experiment : 2 Combinational Logic Circuits

Experiment Date : 11.03.2016

Group Number : 11

Group Members :

|  |  |  |
| --- | --- | --- |
| **ID** | **Name** | **Surname** |
| 150140031 | Pelin | Hakverir |
| 150140051 | Mertcan | Yasakçı |
| 150140044 | Doğay | Kamar |
|  |  |  |

Laboratory Assistant : Atakan Aral

# Introduction

The aim in this experiment is to find the expression with the lowest cost for combinational logic circuits and implement them.

# Requirements

**2.1** First, we solved the Preliminary Question 1 and then we built the circuit we found using the ICs (integrated circuits). Preliminary Question was finding prime implicants of a function F using Karnaugh diagram and Quine-McCluskey method and then creating the prime implicant chart. Finally, we draw the circuit with AND, OR and NOT gates, then implemented it on the CADET.

F(a, b, c, d) = U1(0, 3, 5, 7, 11, 12, 13) + UØ (1, 8, 15)

The Function’s Karnaugh diagram:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| F  ab\cd | 00 | 01 | 11 | 10 |
| 00 | 1 | X | 1 | 0 |
| 01 | 0 | 1 | 1 | 0 |
| 11 | 1 | 1 | X | 0 |
| 10 | X | 0 | 1 | 0 |

Set of all prime implicants: bd+cd+a’d+a’b’c’+abc’+ac’d’+b’c’d’

Letter: A B C D E F G

P.I.: bd cd a’d a’b’c’ abc’ ac’d’ b’c’d’

Cost: 4 4 5 9 7 8 9

Prime implicant chart:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 3 | 5 | 7 | 11 | 12 | 13 | Cost |
| A |  |  | X | X |  |  | X | 4 |
| B |  | X |  | X | X |  |  | 4 |
| C |  | X | X | X |  |  |  | 5 |
| D | X |  |  |  |  |  |  | 9 |
| E |  |  |  |  |  | X | X | 7 |
| F |  |  |  |  |  | X |  | 8 |
| G | X |  |  |  |  |  |  | 9 |

11 is a distinguished point and B is removed with its minterms&added to the final set of prime implicants.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0 | 5 | 12 | 13 | Cost |
| A |  | X |  | X | 4 |
| C |  | X |  |  | 5 |
| D | X |  |  |  | 9 |
| E |  |  | X | X | 7 |
| F |  |  | X |  | 8 |
| G | X |  |  |  | 9 |

A covers C and E covers F, they are added to the final set of prime implicants.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0 | 5 | 12 | 13 | Cost |
| A |  | X |  | X | 4 |
| D | X |  |  |  | 9 |
| E |  |  | X | X | 7 |
| G | X |  |  |  | 9 |

5 and 12 are distinguished points so A and E are removed and added to the final set of prime implicants.

Final set of prime implicants: cd+bd+abc’+a’b’c’

Total cost: 24

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a | b | c | d | F |
| 0 | 0 | 0 | 0 | **1** |
| 0 | 0 | 0 | 1 | **X** |
| 0 | 0 | 1 | 0 | **0** |
| 0 | 0 | 1 | 1 | **1** |
| 0 | 1 | 0 | 0 | **0** |
| 0 | 1 | 0 | 1 | **1** |
| 0 | 1 | 1 | 0 | **0** |
| 0 | 1 | 1 | 1 | **1** |
| 1 | 0 | 0 | 0 | **X** |
| 1 | 0 | 0 | 1 | **0** |
| 1 | 0 | 1 | 0 | **0** |
| 1 | 0 | 1 | 1 | **1** |
| 1 | 1 | 0 | 0 | **1** |
| 1 | 1 | 0 | 1 | **1** |
| 1 | 1 | 1 | 0 | **0** |
| 1 | 1 | 1 | 1 | **X** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | a | b | c | d |
| 0 | **0** | 0 | 0 | 0 |
| 1 | **0** | 0 | 0 | 1 |
| 8 | **1** | 0 | 0 | 0 |
| 3 | **0** | 0 | 1 | 1 |
| 5 | **0** | 1 | 0 | 1 |
| 12 | **1** | 1 | 0 | 0 |
| 7 | **0** | 1 | 1 | 1 |
| 11 | **1** | 0 | 1 | 1 |
| 13 | **1** | 1 | 0 | 1 |
| 15 | **1** | 1 | 1 | 1 |

The function’s truth table and Quine-McCluskey method solution

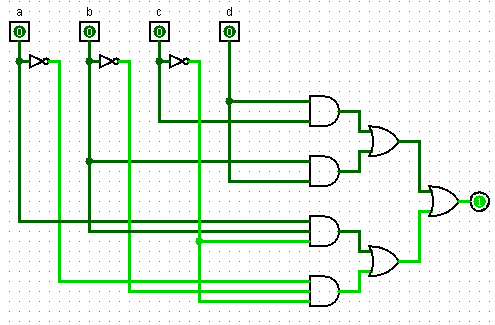
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | a | b | c | d |
| 0,1 | 0 | 0 | 0 | - |
| 0,8 | - | 0 | 0 | 0 |
| 1,3 | 0 | 0 | - | 1 |
| 1,5 | 0 | - | 0 | 1 |
| 8,12 | 1 | - | 0 | 0 |
| 3,7 | 0 | - | 1 | 1 |
| 3,11 | - | 0 | 1 | 1 |
| 5,7 | 0 | 1 | - | 1 |
| 5,13 | - | 1 | 0 | 1 |
| 12,13 | 1 | 1 | 0 | - |
| 7,15 | - | 1 | 1 | 1 |
| 11,15 | 1 | - | 1 | 1 |
| 13,15 | 1 | 1 | - | 1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | a | b | c | d |
| 1,3,5,7 | 0 | - | - | 1 |
| 3,7,11,15 | - | - | 1 | 1 |
| 5,7,13,15 | - | 1 | - | 1 |

Set of all prime implicants: a’b’c’+b’c’d’+ac’d’+abc’+a’d+cd+bd

Same steps are applied to the prime implicants with the Karnaugh diagram’s prime implicant chart.

The circuit:



We implemented the circuit with:

-74xx08 - Quadruple 2-input Positive AND Gates

- 74xx11 - Triple 3-input Positive AND Gates

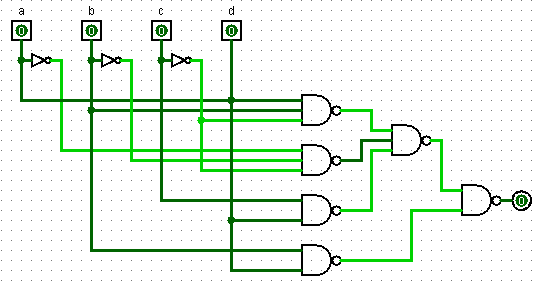
- 74xx32 - Quadruple 2-input Positive OR Gates

- 74xx04 - Hex Inverters

**2.2.** The same function with only NAND and NOT gates:

[(a↓b↓c’) ↓ (a’↓b’↓c’)] ↓ [(c↓d) ↓ (b↓d)]

The circuit:



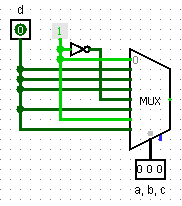
We implemented the circuit with:

- 74xx00 - Quadruple 2-input Positive NAND Gates

- 74xx10 - Triple 3-input Positive NAND Gates

- 74xx04 - Hex Inverters

**2.3.** The same function with only 8:1 MUX and NOT gates:



We implemented the circuit with:

- 74xx151 - 8:1 Multiplexer

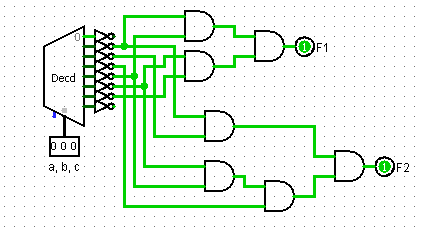
- 74xx04 - Hex Inverters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a | b | c | F1 | F2 |
| 0 | 0 | 0 | **1** | **1** |
| 0 | 0 | 1 | **0** | **0** |
| 0 | 1 | 0 | **1** | **0** |
| 0 | 1 | 1 | **1** | **0** |
| 1 | 0 | 0 | **0** | **0** |
| 1 | 0 | 1 | **0** | **0** |
| 1 | 1 | 0 | **0** | **1** |
| 1 | 1 | 1 | **1** | **1** |

2.4. Truth table of the functions and the circuit:

F1(a,b,c)= a’c’+bc

F2(a,b,c)= a’b’c’+ab



We used AND gates instead of OR gates because the decoder that we used was giving the complement of the output. We could take the minterms and use NAND gates but because that we took maxterms, we used AND gates to reach the answer.

We implemented the circuit with:

- 74xx138 - 3:8 Decoder

- 74xx08 - Quadruple 2-input Positive AND Gates

# Conclusion

We were comfortable with the solutions of the questions and the implementations till the last question. The outputs of the decoder was the complement of the normal result and the question was saying to combine them with OR gates but because of that the result was always the same. Then we asked the assistant about it and learnt that it should’ve been NAND. We were still a little bit confused. But we decided to take the maxterms and combine them with the NAND gates. When we tried it, we saw that the result was still always the same. Later we realised that we were taking the complement two times because we were taking maxterms. So we didn’t want to lose time with changing the cables so we just changed NAND gate with AND gate and the result was what we expected. We didn’t have any other trouble during the experiment.