Lab 2 - BCD Detector

Objectives:

- To understand the Sum-of-Product (SOP) form
- To apply the Karnaugh map method to simplify a Boolean equation
- To apply "don't care" states to a Karnaugh Map (K-map)
- To create a Multisim simulation of a digital circuit

Text References: Chapters 1-4, 6

Components Needed: (quantities unknown)

- 7400 IC – quad two-input NAND gate
- 7404 IC – hex inverter (NOT gate)
- 7408 IC – quad two input AND gate
- 7420 IC – 4-input NAND gate
- 7432 IC – quad two input OR gate

Discussion:

The Karnaugh map (K-map) method of simplification is a powerful graphical technique. The advantages of using K-maps for reduction is that it is easier to see when the expression has been fully simplified and the K-map method provides a structured approach to Boolean simplification.

In this experiment, you will design a BCD detector, implement it in Multisim, simplify the design using a K-map, implement the new designs and do a cost analysis. Start with a truth table for your first design. Think about which 4-input digital numbers make a valid BCD code. Pay special attention to the specifications that follow. We will look at cost of this BCD detector as changes are made. The specification for the BCD detector is as follows:

- Components:
  - 7400 IC – quad two-input NAND gate
  - 7404 IC – hex inverter (NOT gate)
  - 7408 IC – quad two input AND gate
  - 7420 IC – four-input NAND gate
  - 7432 IC – quad two input OR gate
- Need to minimize the overall cost of the BCD detector
- Cost breakdown of the components used
  - Each part number is $4 (i.e 7400, 7404, 7408,...)
  - Each other Component (LED, switch, etc) is $2
  - Each individual connector is charged by $1 per wire connection

---

- Inputs: Four from level switches (A, B, C, D) representing the binary number
- Outputs: One LED turns ON to indicate a valid BCD number as an input, turns OFF for an invalid BCD number

Truth Table

A	B	C	D	F
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

KMAP

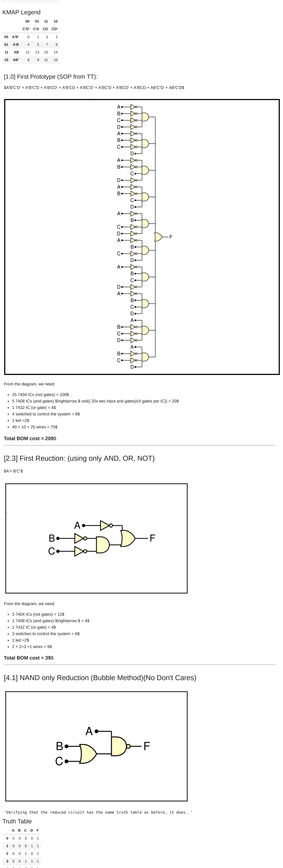
	00	01	11	10
	CD'	CD	CC	CD'
00 A'B'	1	1	1	1
01 A'B	1	1	1	1
11 AB	0	0	0	0
10 AB'	1	1	0	0

KMAP Legend

	00	01	11	10
	C'D'	C'D	CD	CD'
00 A'B'	0	1	3	2
01 A'B	4	5	7	6
11 AB	12	13	15	14
10 AB'	8	9	11	10

[1.0] First Prototype (SOP from TT):

$A'B'C'D' + A'B'C'D + A'B'CD + A'B'CD' + A'BC'D' + A'BC'D + A'BCD + A'BCD' + AB'C'D'$



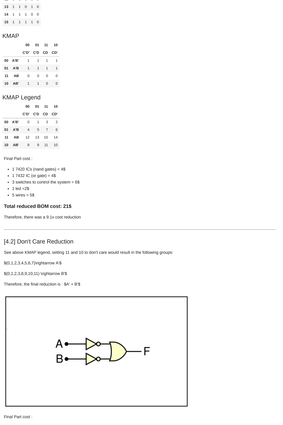
From the diagram, we need:

- 25-7404 ICs (not gates) = 1008
- 5-7408 ICs (and gates) Brightsense 8 cells) 20x two input and gates(still gates per IC)) = 206
- 1 7432 IC (or gate) = 4$
- 4 switched to control the system = 8$
- 1 led =2$
- 40 + 10 + 25 wires = 75$

Total BOM cost = 2095

[2.3] First Reuction: (using only AND, OR, NOT)

$A'+ B'C'$

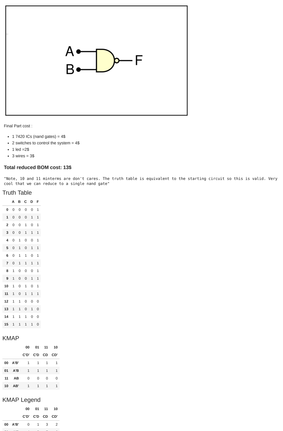


From the diagram, we need:

- 3 7404 ICs (not gates) = 12$
- 1 7408 ICs (and gates) Brightsense 8 = 4$
- 1 7432 IC (or gate) = 4$
- 3 switches to control the system = 6$
- 1 led =2$
- 2 + 2+3 + 1 wires = 8$

Total BOM cost = 39$

[4.1] NAND only Reduction (Bubble Method)(No Don't Cares)




```
'Verifying that the reduced circuit has the same truth table as before, it does.'
```

Truth Table

A	B	C	D	F
0	0	0	0	1
1	0	0	0	1
2	0	0	1	1
3	0	0	1	1
4	0	1	0	1
5	0	1	0	1
6	0	1	1	0
7	0	1	1	0
8	1	0	0	1
9	1	0	0	1
10	1	0	1	0
11	1	0	1	0
12	1	1	0	0
13	1	1	0	0
14	1	1	1	0
15	1	1	1	0

KMAP

	00	01	11	10
	C'D'	C'D	CD	CD'
00 A'B'	1	1	1	1
01 A'B	1	1	1	1
11 AB	0	0	0	0
10 AB'	1	1	0	0

KMAP Legend

	00	01	11	10
	C'D'	C'D	CD	CD'
00 A'B'	0	1	3	2
01 A'B	4	5	7	6
11 AB	12	13	15	14
10 AB'	8	9	11	10

Final Part cost :

- 1 7420 ICs (nand gates) = 4$
- 1 7432 IC (or gate) = 4$
- 3 switches to control the system = 6$
- 1 led =2$
- 5 wires = 5$

Total reduced BOM cost: 215

Therefore, there was a 9 1x cost reduction

[4.2] Don't Care Reduction

See above KMAP legend, setting 11 and 10 to don't care would result in the following groups:

$(0,1,2,3,4,5,6,7)$rightarrow#$

$(0,1,2,3,8,9,10,11)$ rightarrowA'

Therefore, the final reduction is $A' + B'C'$

Final Part cost :

- 2 7404 ICs (not gates) = 8$
- 1 7432 IC (or gate) = 4$
- 3 switches to control the system = 6$
- 1 led =2$
- 5 wires = 5$

Total reduced BOM cost: 195

[4.4] NAND only Don't Care Reduction

Convert the don't care logic diagram to use only NAND gates.

Note: I used the demorgans law here

Final Part cost :

- 1 7420 ICs (nand gates) = 4$
- 2 switches to control the system = 4$
- 1 led =2$
- 3 wires = 3$

Total reduced BOM cost: 135

```
'Note, 10 and 11 minterms are don't cares. The truth table is equivalent to the starting circuit so this is valid. Very cool that we can reduce to a single nand gate'
```

Truth Table

A	B	C	D	F
0	0	0	0	1
1	0	0	0	1
2	0	0	1	1
3	0	0	1	1
4	0	1	0	1
5	0	1	0	1
6	0	1	1	1
7	0	1	1	1
8	1	0	0	1
9	1	0	0	1
10	1	0	1	0
11	1	0	1	0
12	1	1	0	0
13	1	1	0	0
14	1	1	1	0
15	1	1	1	0

KMAP

	00	01	11	10
	CD'	CD	CC	CD'
00 A'B'	1	1	1	1
01 A'B	1	1	1	1
11 AB	0	0	0	0
10 AB'	1	1	0	0

KMAP Legend

	00	01	11	10
	C'D'	C'D	CD	CD'
00 A'B'	0	1	3	2
01 A'B	4	5	7	6
11 AB	12	13	15	14
10 AB'	8	9	11	10

[R] Results

[R.1] "Why isn't input D needed in the BCD detector design?" :

Because, D was factored out during minimum SOP reduction. In the Don't care reduction, C is also eliminated

[R.2] Cost Table

Final Part cost :

- 1 7420 ICs (nand gates) = 4$
- 2 switches to control the system = 4$
- 1 led =2$
- 3 wires = 3$

Total reduced BOM cost: 135

Part Cost	Chip Cost	Connection Cost	Total Cost
OriginalPoint	12	126	75	209
Simplified	8	20	5	33
Don't Care	4	4	3	19
NAND	4	4	3	19