# Hardware Laboratory
# Assignment-1 Report
# Seven-Segment Decoder

| | |
|---|---|
| Course: | COL215 |
| Term: | Semester-1 2023-24 |
| Group No.: | 3 |
| Topic: | Seven Segment Decoder Circuit |

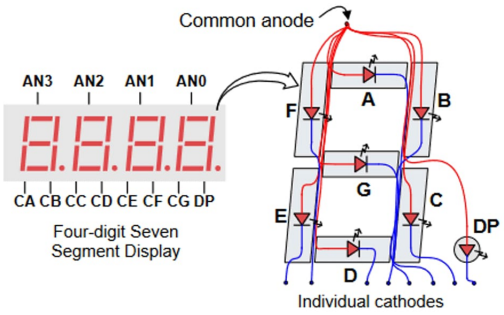

| | |
|---|---|
| Prepared by: | Abhiram Sanjay Dharme,<br>Entry No. 2022CS51139 |
| | Ujjawal Sharma,<br>Entry No. 2022CS51137 |
| | Purushottam Sharma,<br>Entry No. 2022CS51635 |


| | |
|---|---|
| Date: | August 3, 2023 |

# Seven Segment Decoder Circuit

## 1 Problem Statement

Design a combinational circuit which takes a single 4-bit binary input from the switches and produces a output on the seven-segment display of Basys3 FPGA board




## 2 Design I/O

### 2.1 Input

We use the switches on the basys3 board as the inputs. Their being ON represents 1 while OFF means 0 in our case.

### 2.2 Output

We obtain the hexadecimal representation, of the binary input given via the switches, in the 7-Segment Display

## 3 Discussion: K-maps[1]

This assignment included the usage of K-Maps or Karnaugh Maps. Karnaugh maps are used to simplify real-world logic requirements so that they can be implemented using a minimum number of logic gates. A sum-of-products expression (SOP) can always be implemented using AND gates feeding into an OR gate, and a product-of-sums expression (POS) leads to OR gates feeding an AND gate. The POS expression gives a complement of the function (if F is the function so its complement will be F'). Karnaugh maps can also be used to simplify logic expressions in software design. Boolean conditions, as

used for example in conditional statements, can get very complicated, which makes the code difficult to read and to maintain. Once minimised, canonical sum-of-products and product-of-sums expressions can be implemented directly using AND and OR logic operators

# 4 Working

Initially, the circuit takes a 4 bit binary number as input through it's 4 switches which represents 0=ACTIVE and 1=INACTIVE, but modifications were made such that ACTIVE $\Rightarrow$ 1 and INACTIVE $\Rightarrow$ 0 (If one wants to give input 11 then it the switches will be 'on' 'off' 'on' 'on' which will produce 1011 as input in binary and the 7-segment basys3 board will show 'b')

## 4.1 Equations

$$\overline{A} \Leftarrow (\overline{b_0 b_1 b_3} + \overline{b_0} b_2 + b_2 \overline{b_3} + b_0 b_1 b_2 + \overline{b_0} b_1 b_3 + b_0 \overline{b_2 b_3} + b_0 \overline{b_1 b_2})$$

$$\overline{B} \Leftarrow (\overline{b_0 b_1} + \overline{b_1 b_3} + \overline{b_0} b_2 b_3 + \overline{b_0 b_2 b_3} + b_0 \overline{b_2} b_3)$$

$$\overline{C} \Leftarrow (\overline{b_2} + \overline{b_0 b_1} b_3 + \overline{b_0} b_1 + b_0 \overline{b_1})$$

$$\overline{D} \Leftarrow (\overline{b_0 b_1 b_3} + \overline{b_1}\ b_2 b_3 + b_0 \overline{b_2} + b_1 \overline{b_2} b_3 + b_1 b_2 \overline{b_3})$$

$$\overline{E} \Leftarrow (\overline{b_1 b_2 b_3} + b_2\ \overline{b_3} + b_0 b_1 b_0\ \overline{b_1} b_2)$$

$$\overline{F} \Leftarrow (\overline{b_2 b_3} + b_0 \overline{b_1} + b_0 b_2 + \overline{b_0} b_1\ \overline{b_2} + b_1 b_2 \overline{b_3})$$

$$\overline{G} \Leftarrow (\ b_0 \overline{b_1} + b_2 \overline{b_3} + b_0 b_1 b_3 + \overline{b_0} b_1 \overline{b_2} + \overline{b_0 b_1} b_2)$$

Where A,B,C,D,E,F,G represent the cathodes of the seven-segment display

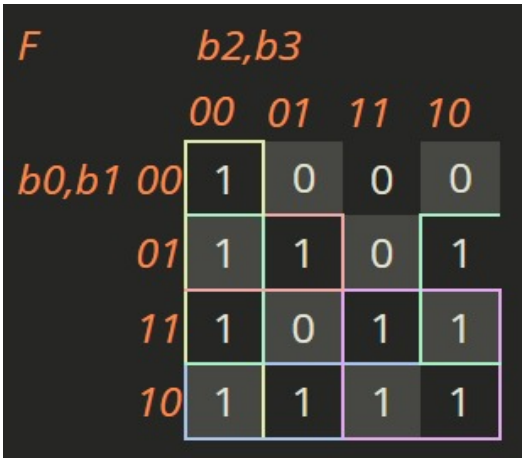
## 4.2 K-Maps[2]

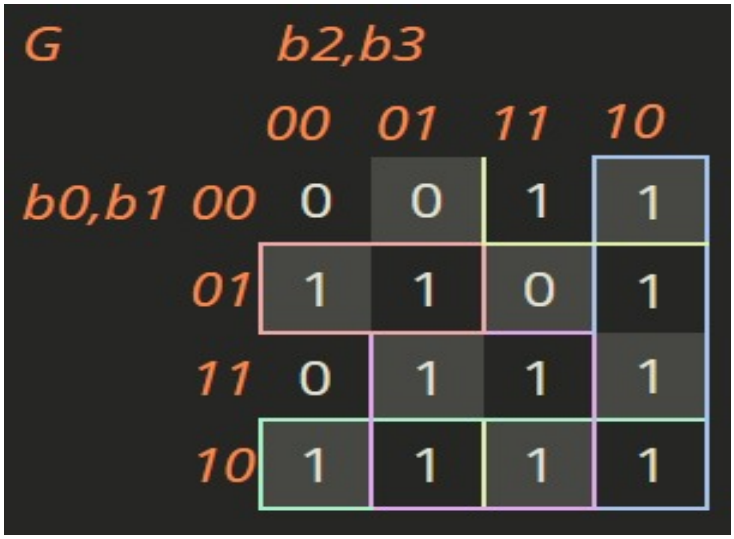


C
b2,b3
00 01 11 10
b0,b1 00 1 1 1 0
01 1 1 1 1
11 1 1 0 0
10 1 1 1 1

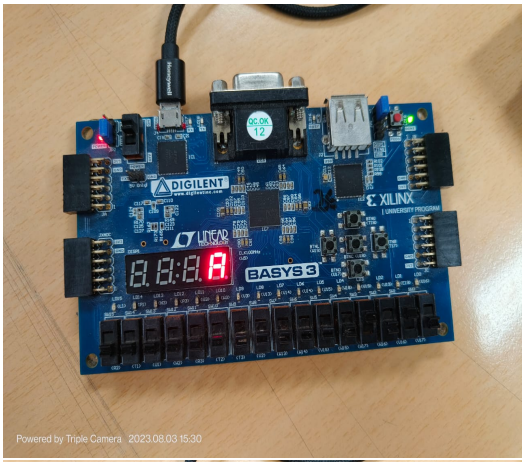

D
b2,b3
00 01 11 10
b0,b1 00 1 0 1 1
01 0 1 0 1
11 1 1 0 1
10 1 1 1 0

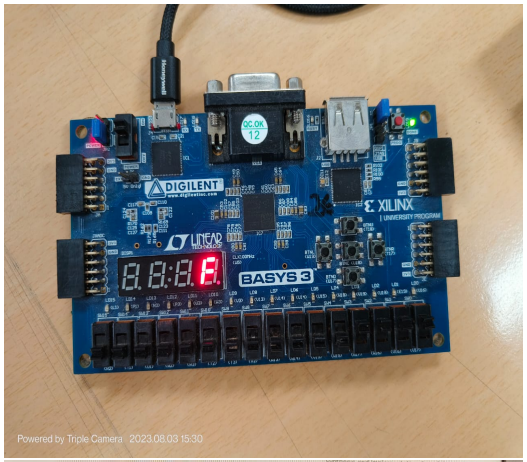

E
b2,b3
00 01 11 10
b0,b1 00 1 0 0 1
01 0 0 0 1
11 1 1 1 1
10 1 0 1 1

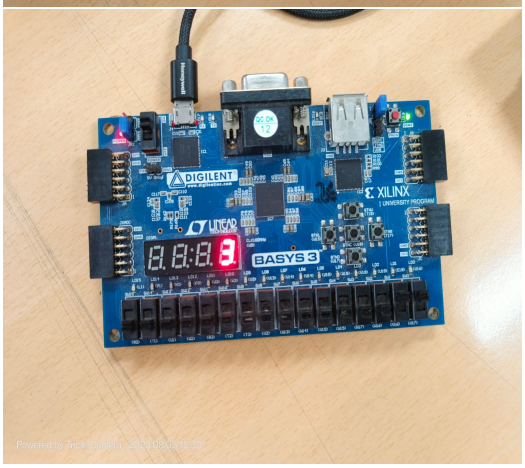

F
b2,b3
00 01 11 10
b0,b1 00 1 0 0 0
01 1 1 0 1
11 1 0 1 1
10 1 1 1 1

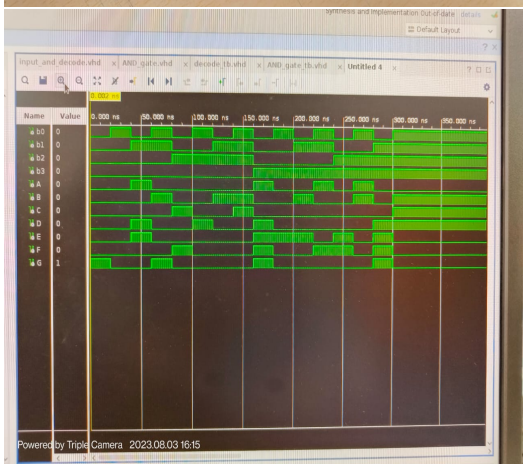

G
b2,b3
00 01 11 10
b0,b1 00 0 0 1 1
01 1 1 0 1
11 0 1 1 1
10 1 1 1 1

## 4.3 Images




## 4.4 Resource Counts

| Site Type | Use |
|---|---|
| Slice LUTs | 4 |
| as Logic | 4 |
| as Memory | 0 |
| Slice Registors | 0 |
| as Flip Flop | 0 |
| as Latch | 0 |
| Block RAM Tile | 0 |
| as RAMB36/FIFO | 0 |
| as RAMB18 | 0 |
| DSPs | 0 |

# References

[1] *Karnaugh maps*. Karnaugh maps as on Wikipedia.

[2] *Karnaugh maps generator*. Online Karnaugh maps generator.