LAB-6

Binary Subtraction

Binary Number	Subtraction Value	
	Diff	Borro
0 - 0	0	0
1 - 0	1	0
0 - 1	0	1
1 - 1	0	0

(10 = 8)

```
  1  1  1  -1  0
  1  0  0  1  1
-------------------
  0  1  0  1  0
```

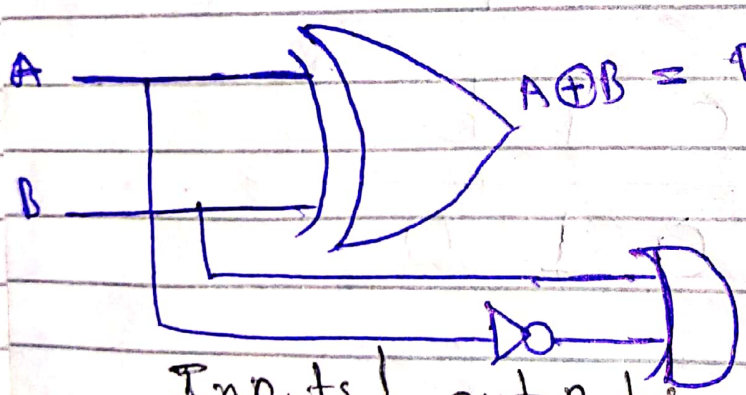
(Half Subtractor)

=> Half Subtractor is a combinational circuit with two inputs and two outputs (difference and borrow).

=> It produces the difference between the two binary bits at the input and also produces an output (Borrow) to indicate if 1 has been borrowed.

Difference (D) = $A \oplus B = A'B + AB'$

Borrow (B) = $A' \cdot B$

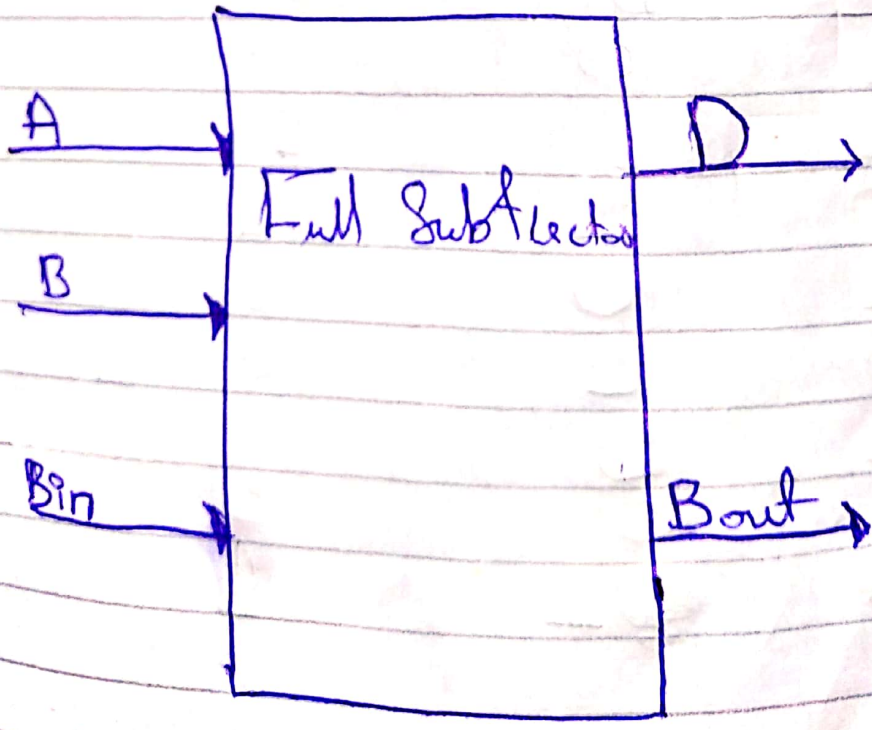


Inputs		outputs	
A	B	A-B	Borrow
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

(Full Subtractor)

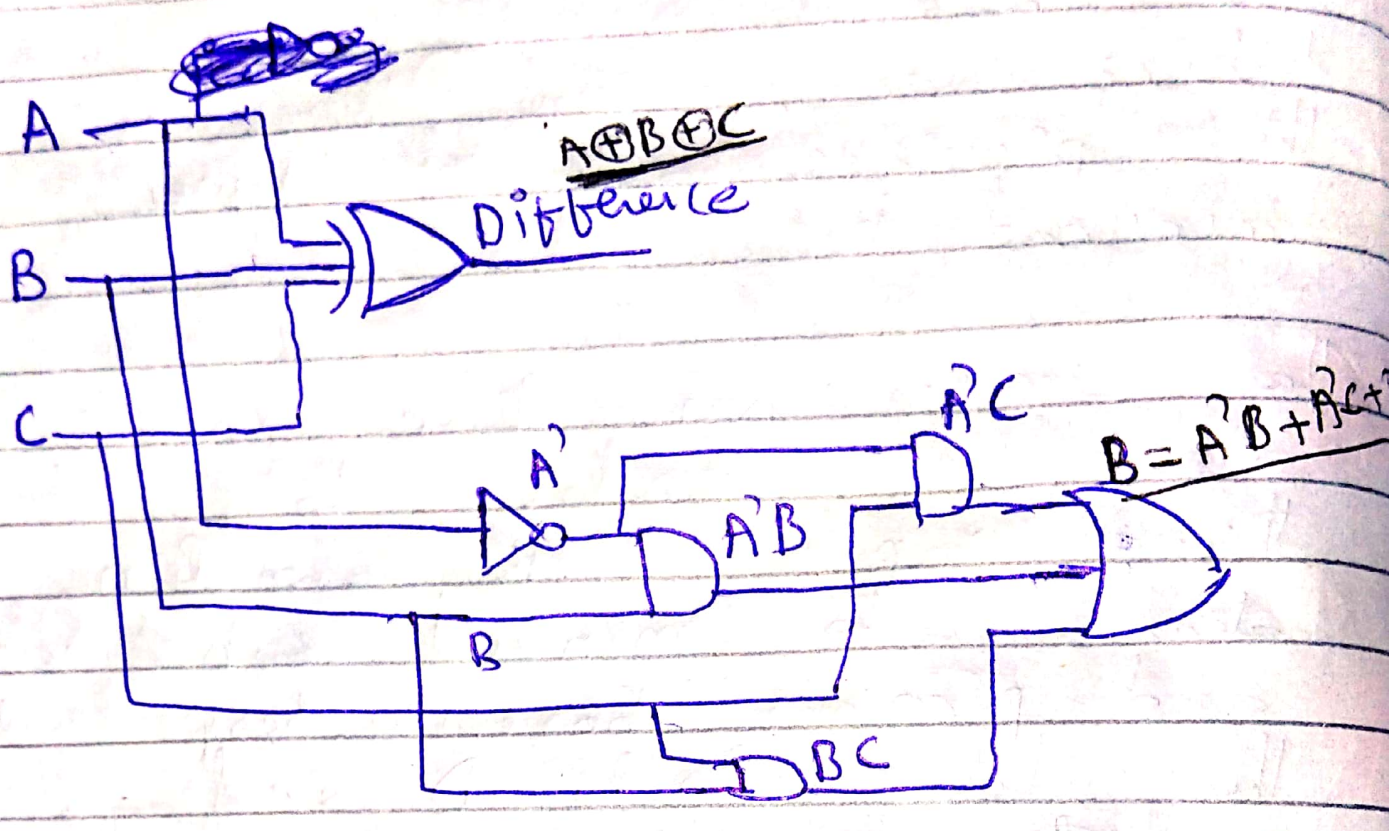
=> A full subtractor is a combinational circuit that performs subtraction of two bits, one is minued and other is subtrahend, taking into account borrow. of the previous adjacent lower minuend bit.

=> The circuit has three inputs and two outputs. The three inputs A, B and Bin denote the minued subtrahend and previous borrow, respectively the two outputs, D and Bout the difference and out put borrow



Difference (D) = $A \oplus B \oplus C$

Borrow (B) = $A'B + A'.C_{in} + B.C$



A	B	C	Diff	Borr
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Formula

(1st step)

A - B = R

2nd step

R - C = Diff , Borr

A=0, B=0, C=1

A - B = R
0 - 0 = 0

R - C = Diff , Borrow
0 - 1 = 1 , 1

A=0 , B=1, C=1

A - B = R , Diff , Borr
0 - 1 = , 1

it we borrow 1 then
0 will become 10

A B = R , Borr
10 - 1 = 1 , 1

(Code Convertors)

The code converter is used is used to convert one type of binary code to another.

There are different types of binary codes like BCD code gray code, excess code, etc different codes are used for different types of digital application.

To get the required code from any one type of code the simple code conversion process is done with the help of combinational circuits.

A code converter circuit will convert coded information in one coding form to a different

Decimal Digit	BCD code	Excess 3 code
0	0000	0011
1	0001	0100
2	0010	0101
3	0011	0110
4	0100	0111
5	0101	1000
6	0110	1001
7	0111	1010
8	1000	1011
9	1001	1100

We can make truth table like this

	(BCD-8421)				Excess-3			
	A	B	C	D	W	X	Y	Z
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

Now we can make Boolean Expression for each w, x, y, z, with k-map

$W = A + BC + BD$

$X = B'C + B'D + BC'D'$

$X = CD + C'D'$

$Z = D'$

Q: BCD to Excess 3-code

Q We will add 3 three (11) in BCD code to get Excess 3-code

Exp

$$\begin{array}{r} 1000 \longrightarrow \text{BCD-code} \\ 0011 \\ \hline 1011 \longrightarrow \text{Excess 3 code} \end{array}$$

Things to Remember

⇒ (Half Subtractor)

Diff = $A \oplus B$

Borrow = $A' \cdot B$

⇒ Full Subtractor

Diff = $A \oplus B \oplus C$

Borrow = $A' \cdot B + B \cdot C + A' \cdot C$

or

$A' \cdot B + C(\overline{A \oplus B})$