EXAMPLES

- Arithmetic operations in 2’s complement form
- Arithmetic operations with BCD numbers
- Use of Booelan algebra rules
- Deriving logical expressions from logic circuits
- Implementing logic circuits from logical expressions
- Timing diagrams of logic circuits
- Generating truth tables from verbal definition of problems.

Example #1

Let’s perform the operations below and check if there’s an overflow.

-117	117
+ 89	- (-89)

- 117 + 89 -28	10001011 + 01011001 11100100 **No Overflow**
117 - (- 89) 206	01110101 + 01011001 11001110 **Overflow**

Example #2

Let’s perform the operation below in BCD.

$(387)_{10} + (439)_{10}$

$$\begin{array}{lrrr}
 & 1 & 1 & \\
 & 0011 & 1000 & 0111 \\ \hline
+ & 0100 & 0011 & 1001 \\
 & 1000 & 1100 & 10000 \\ \hline
+ & & 0110 & 0110 \\
 & 1000 & 10010 & 0110
\end{array}$$

$(1000\ 0010\ 0110)_{BCD} = 826$

Example #3

Let’s find out the binary number represented in 32-bit floating point form as below.

01000000101101100000000000000000

Sign	Exponent+Bias	Fractional Part
0	10000001	01101100000000000000000

- Since the sign bit is 0, we can say that the number is positive.
- Bias = $2^{n-1}-1 = 2^{8-1}-1 = 2^7-1 = 127$
- Exponent + Bias = $(10000001)_2 = 129$
- Exponent = 129-127 = 2.
- Then, the number is, $+ 1.011011\times2^2 = 101.1011$

Example #4

Let’s prove that $ab'+bc'+a'c = a'b+b'c+ac'$ using Boolean algebra.

We will first extend the terms on the left hand side of the equation, then simplify it again.

$$\begin{aligned} ab'(c+c')+(a+a')bc'+a'(b+b')c &= ab'c+\underline{ab'c'+abc'}+\underline{a'bc'+a'bc}+a'b'c \\ &= b'c+ac'+a'b \end{aligned}$$

Example #5

Let’s prove that $ab+a'c+bcd = ab+a'c$ using Boolean algebra.

$ab+a'c+(a+a')bcd = \underline{ab}+a'c+\underline{abcd}+a'bcd = ab(1+cd) + a'c\,(1+bd) = ab+a'c$

Remember that, we have proven that $ab + a'c + bc = ab+a'c$ before.

We can say that, if we add additional variables to the 3rd term (bc) in the equation $ab + a'c + bc = ab+a'c$, we can still eliminate the 3rd term.

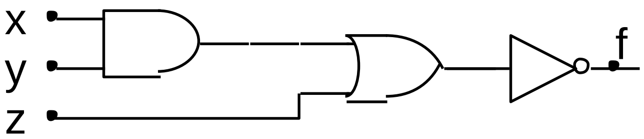
$ab+a'c+bcd = ab+a'c$

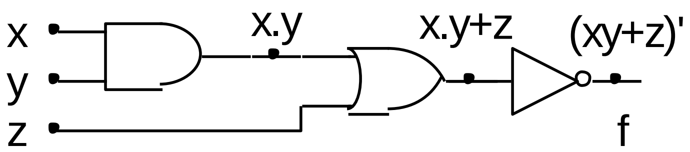
$ab+a'c+bcde = ab+a'c$

$ab+a'c+bcdef = ab+a'c$

Example #6

Let’s generate the truth table of the following logic circuit.

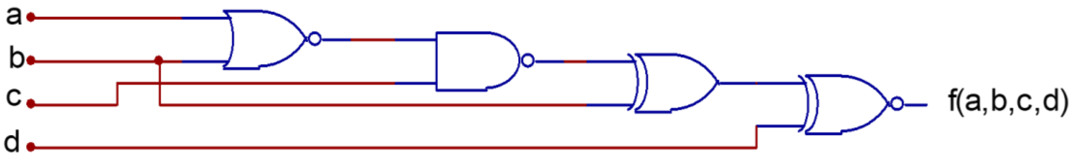


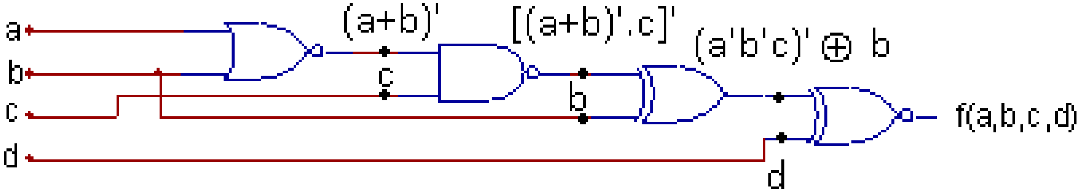


x	y	z	x.y	x.y+z	(x.y+z)’
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	0	1	0
1	0	0	0	0	1
1	0	1	0	1	0
1	1	0	1	1	0
1	1	1	1	1	0

Example #7

Let’s find the minterms of the output.





One of the inputs of the EXOR gate is *((a+b)’.c)’* = *(a’b’c)’*, and the other is *b*.

One of the inputs of EXNOR gate is *(a’b’c).b+(a’b’c)’b’* = *(a+b+c’).b’*, and the other one is *d*.

Then, the output is;

f = (ab’+b’c’)d+(ab’+b’c’)’d’

= ab’d+b’c’d+(a’+b)(b+c)d’ = ab’d+b’c’d+$\underline{a'bd'}$+a’cd’+$\underline{bd'}$+$\underline{bcd'}$

= ab’d+b’c’d+bd’+a’cd’

Continue to next page

Example #7

f= ab’d+b’c’d+bd’+a’cd’

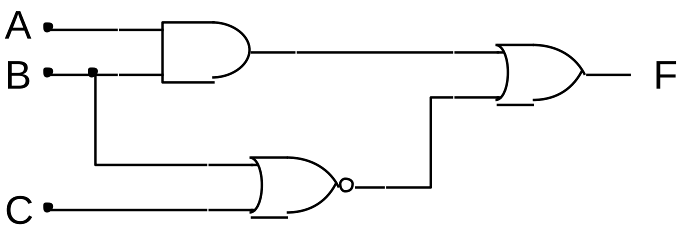
Now, let’s find the minterms.

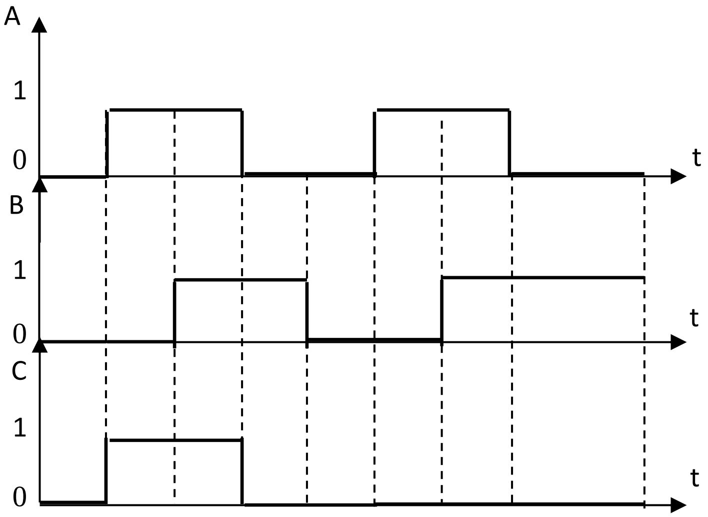
ab’d	**b’c’d**	**bd’**	**a’cd’**
abcd	abcd	abcd	abcd
1001	0001	0100	0010
1011	1001	0110	0110
		1100	
		1110	

f(a,b,c,d)=∑(1,2,4,6,9,11,12,14)

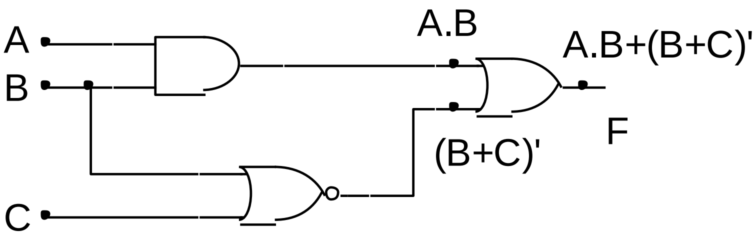
Example #8

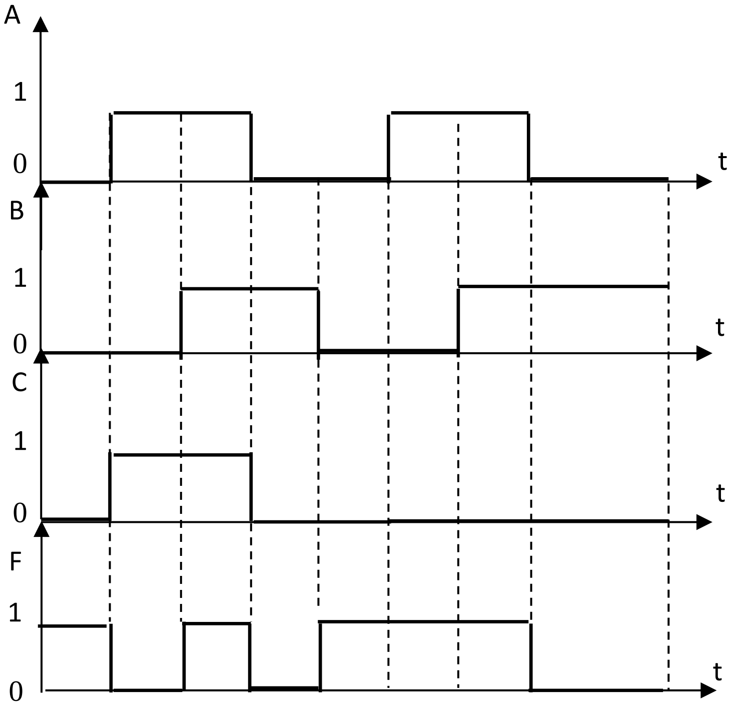
A combinational logic circuit and the input signals' timing diagram is given below. Let's find out the timing diagram of the output.





Example #8





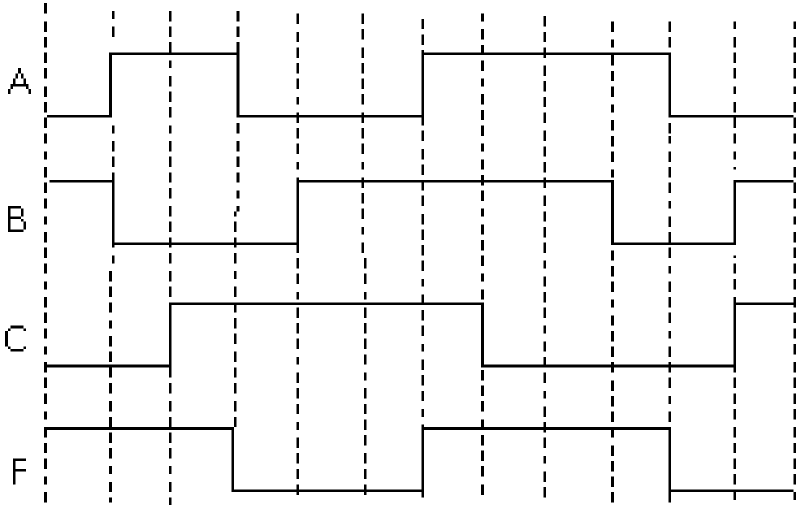
F = AB+(B+C)’ = AB+B’C’

The term AB tells us that, the output would be 1 when A and B are both 1.

The term B’C’ tells us that, the output would be 1 when B and C are both 0.

Example #9

Let’s generate the truth table and the simplified logical expression from the timing diagram below.



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

F = A+B.C’

Example #10

Let’s assume that we are asked to build a combinational logic circuit with 3 inputs (a,b,c) and a single output (z). This circuit must output 1 when the binary value of the inputs is less than 3. Let’s generate the truth table and find out the simplified expression.

a	b	c	z
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

$$z = a'b'c'+a'b'c+a'bc' = a'b'(c'+c)+a'bc'$$

$$= a'b'+a'bc' = a'(b'+bc') = a'(b'+c') = a'b'+a'c'$$

Example #11

Let’s assume that we are asked to build a combinational logic circuit with 3 inputs (a,b,c) and a single output (z). This circuit must output 1 when the number of 1’s are greater than the number of 0’s at the input **(Majority Function)**. Let’s generate the truth table and find out the simplified expression.

a	b	c	z
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$$z = a'bc+ab'c+abc'+abc = a'bc+ab'c+ab(c+c')$$

$$= a'bc+ab'c+ab = a'bc+a(b'c+b) = a'bc+a(b+c)$$

$$= a'bc+ab+ac = b(a'c+a)+ac = b(a+c)+ac = ab+bc+ac$$

Example #12

Let’s assume that we are asked to build a combinational logic circuit with 2 inputs (A,B). This circuit must output the binary value calculated by this formula:

Output Binary Value = 4 x Input Binary Value + 3

Let’s generate the truth table and find out the simplified expressions of the outputs.

The maximum binary value that can be input is $11_2 = 3_{10}$. So, the maximum binary value of the output can be $4x3+3 = 15_{10} = 1111_2$. It means that, the circuit must have 4 outputs.

A	B	F3	F2	F1	F0
0	0	0	0	1	1
0	1	0	1	1	1
1	0	1	0	1	1
1	1	1	1	1	1

F3 = AB’+AB = A

F2 = A’B+AB = B

F1 = 1

F0 = 1

Example #13

A router connects multiple computers together and lets them send messages to each other. But the network can transmit a single message at a time. So, if more than one computers try to send messages at the same time, a collision occurs. We are asked to build a collision detector for a network with 3 computers. The circuit has 3 inputs (x,y,z) and these inputs indicate that, the computer connected to that input sends a message. So, if more that a single input is 1, it means that there's a collision and the circuit must output 1.

x	y	z	d
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$$d = \underline{x'yz} + xy'z + xyz' + \underline{xyz} = yz + x(y'z+yz') = yz+x(y\oplus z)$$

Example #14

There are 3 compulsory and 2 optional questions in an exam.

A student must answer 3 compulsory questions, or 2 compulsory + 2 optional questions in order to pass the exam.

The inputs z_1, z_2, and z_3 represent the compulsory questions.

The inputs s_1 and s_2 represent the optional questions.

We can write the expression directly without generating the truth table.

$G = z_1.z_2.z_3 + z_1.z_2.s_1.s_2 + z_1.z_3.s_1.s_2 + z_2.z_3.s_1.s_2$

Example #15

We have a car alarm with 3 sensors.

Sensor D: The value is 0 if all the doors are closed, and 1 if any door is open

Sensor G: The value is 0 if the engine is stopped, and 1 if the engine is started.

Sensor L: The value is 0 if the headlights are off, and 1 if the headlights are off.

The alarm rings (output Y is 1) under these conditions:

1. Headlights are on when the engine is stopped.
2. Any door is open when the engine is started.

Let’s generate the truth table and find the expression in SoP form.

D	**G**	**L**	**Y**
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$$Y=\sum (1,5,6,7) = DG+G'L$$

Example #16

We are asked to build a controller for an air conditioner. The controller has 3 inputs (A, B, T) and 2 outputs (I, S).

Input A is used to select automatic mode (A=1) or manual mode (A=0).

Input B is used to select heating mode (B=1) or cooling mode (B=0).

Input T is connected to a sensor which tells if the room is warmer than desired (T=1) or cooler than desired (T=0).

Output I is used to switch the heater on (I=1) or off (I=0).

Output S is used to switch the cooler on (S=1) or off (S=0).

Example #16

The desired behavior of the controller is described below.

In Automatic Mode (A=1)	In Manual Mode (A=0)
1. When the user switches to heating mode, the controller will turn on the heater if the room is cooler than desired, or turn off the heater if the room is warmer than desired.	**1.** When the user switches to heating mode, the controller will turn on the heater no matter what the temperature is.
2. When the user switches to cooling mode, the controller will turn on the cooer if the room is warmer than desired, or turn off the cooler if the room is cooler than desired.	**2.** When the user switches to cooling mode, the controller will turn on the cooler no matter what the temperature is.

The truth table can be generated from the description above.

A	B	T	S	I	Rule
0	0	0	**1**	0	Manual Mode Rule #2
0	0	1	**1**	0	Manual Mode Rule #2
0	1	0	0	**1**	Manual Mode Rule #1
0	1	1	0	**1**	Manual Mode Rule #1
1	0	0	**0**	0	Automatic Mode Rule #2
1	0	1	**1**	0	Automatic Mode Rule #2
1	1	0	0	**1**	Automatic Mode Rule #1
1	1	1	0	**0**	Automatic Mode Rule #1

$$S(A,B,T) = \sum(0,1,5)$$

$$I(A,B,T) = \sum(2,3,6)$$