KEMENTERIAN PENGAJIAN TINGGI

DIGITAL SYSTEM

REVISION

PREPARED BY:

SHUKRI BIN ZAKARIA
WAN NOR SHELA EZWANE BINTI WAN JUSOH
Mechanical Engineering Department
Polytechnic Tuanku Sultanah Bahiyah

AUTHORS

SHUKRI BIN ZAKARIA
Lecturer
shukri_zakaria@ptsb.edu.my

WAN NOR SHELA EZWANE BT WAN JUSOH
Lecturer
shela@ptsb.edu.my

e ISBN 978-967-2740-22-3

Mechanical Engineering Department
Politeknik Tuanku Sultanah Bahiyah, Kulim Hi Tech Park,
09090 Kulim, Kedah.

PREFACE

Alhamdulillah to Allah SWT With His grace and mercy, the first e-book module of DJM30073 DIGITAL SYSTEM has finally completed. Thank you to my family and friends especially PTSB e-book team for being support me to complete this e-book. I also would like to thank you to my Head of Mechanical Engineering Department, Mr. Azhar Bin Fikri for giving this opportunity. As educator, I feel the need to prepare a e-book module that consist of notes ,step by step practical works and some exercises that can assist the student in the learning process.

The content of this e-book has been constructed to meet the polytechnic's syllabus requirement. Some chapter contains illustrative screenshots to match the main practical work, enhancing student comprehending ability. This e-book contains 7 chapter such as Number and Code System, Boolean Operation, Sequential Logic, Counters, Registers, Converters and also variety of exercises and examples. Any positive feedback from lecturers and students are mostly welcome and appreciated. Hopefully this e-book module is one of the step that start the long journey of road to excellent.

SYNOPSIS

DIGITAL SYSTEM provides the knowledge on the concepts and basic principles of digital circuits used in computer systems. This course focuses on sequential logic circuits, counters and registers. This course also covers the topics on the methods of signal conversion in electronic circuits.

Upon completion of this course, students should be able to:

CLO1: Distinguish the characteristics and operations of various digital circuits (C4, PLO1)

CLO2: Construct digital circuits based on schematic diagrams (P4, PLO5)

CLO3: Demonstrate the role of digital circuits in real world applications. (A3, PLO7)

CONTENTS

CHAPTER 1

NUMBER & CODE SYSTEMS

SHORT NOTE CHAPTER 1 : NUMBER SYSTEM

1.1 Introduction

The binary number system and digital codes are represented by the binary digit ‘0’ and ‘1’ which are known as bits. Bits can be divided by two group which is nibbles and byte. Nibbles have four bits and byte have eight bits in group. The binary number system has its relationship to other number systems such as decimal, hexadecimal, and octal. Binary Code Decimal (BCD) and ASCII code is also known as digital codes. Computers and many types of digital systems functioned used these arithmetic operations with binary numbers.

1.2 Binary (Base 2)

Binary to Decimal

Convert 11011_2 to N_{10}

$$\begin{aligned}\mathbf{11011_2} \quad &= (1x2^4)+(1x2^3)+(0x2^2)+(1x2^1)+(1x2^0)\\ &= 16+8+0+2+1=\mathbf{27_{10}}\end{aligned}$$

$$11011_2 = \mathbf{27_{10}}$$

Binary to Octal

Convert 111111101111000_2 to N_8

111	**111**	**101**	**111**	**000_2**
421	421	421	421	421
7	**7**	**5**	**7**	**0_8**

$$111111101111000_2 = \mathbf{77570_8}$$

Binary to Hexadecimal

Convert 001111110000_2 to N_{16}

0011	**1111**	**0000_2**
8421	8421	8421
3	**F**	**0_{16}**

$$001111110000_2 = \mathbf{3F0_{16}}$$

1.3 Decimal (Base 10)

Decimal to Binary

Convert 73_{10} to N_2

2	**73**	-1
2	36	-0
2	18	-0
2	9	-1
2	4	-0
2	2	-0
	1	

$73_{10} = \mathbf{1001001_2}$

Decimal to Octal

Convert 435_{10} to N_8

8	**435**	-3
8	54	-6
	6	

$435_{10} = \boldsymbol{663_8}$

Decimal to Hexadecimal

Convert 2890_{10} to N_{16}

16	**2890**	-10
16	180	-4
	11	

$2890_{10} = \boldsymbol{B4A_{16}}$

1.4 Octal (Base 8)

Octal to Binary

Convert 377_8 to N_2

3	**7**	**7_8**
421	421	421
011	**111**	**111_2**

$377_8 = \mathbf{011111111_2}$

Octal to Decimal

Convert 372_8 to N_{10}

$$\mathbf{372_8} = (3x8^2) + (7x8^1) + (2x8^0) = 192 + 56 + 2 = \mathbf{250_{10}}$$

Octal to Hexadecimal (OCTAL > BINARY > HEXADECIMAL)

Convert 372_8 to N_{16}

3			**7**			**2**			
4	2	1	4	2	1	4	2	1	
0	1	1	1	1	1	0	1	0	← N_2
1	8	4	2	1	8	4	2	1	
	8 + 4 + 2 + 1 = **15 = F**				8 + 2 = **10 = A**				← N_{16}

← N_8

$$372_8 = \boldsymbol{FA_{16}}$$

1.5 Hexadecimal (Base 16: 0-9 and A-F)

Hexadecimal to Binary

Convert $A14_{16}$ to N_2

A	**1**	**4_{16}**
8421	8421	8421
1010	0001	0100_2

$$A14_{16} = \mathbf{1010\ 0001\ 0100_2}$$

Hexadecimal to Decimal

Convert $5C2_{16}$ to N_{10}

$$\mathbf{5C2_{16}} = (5x16^2) + (12x16^1) + (2x16^0) = \mathbf{1474_{10}}$$

Hexadecimal to Octal (HEXADECIMAL > BINARY > OCTAL)

Convert $9F2_{16}$ to N_8

9				**F**				**2**				← N_{16}
8	4	2	1	8	4	2	1	8	4	2	1	
1	0	0	1	1	1	1	1	0	0	1	0	← N_2
4	2	1	4	2	1	4	2	1	4	2	1	
4 + 0 + 0 = **4**			4 + 3 + 1 = **7**			4 + 2 + 0 = **6**			0 + 2 + 0 **= 2**			← N_8

$$9F2_{16} = \mathbf{4762_8}$$

****REMEMBER**

10	**11**	**12**	**13**	**14**	**15**
A	**B**	**C**	**D**	**E**	**F**

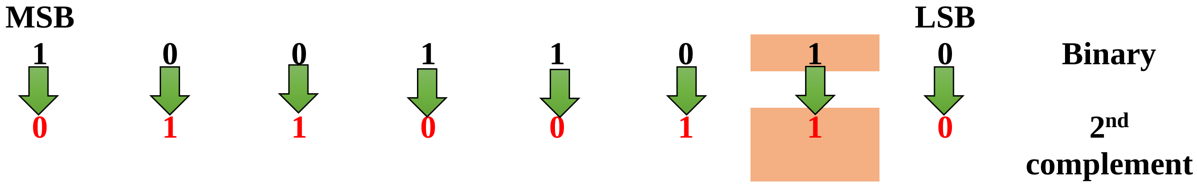
1.6 First and Second Complement

1st Complement:

First complement of a binary number is obtained by changing each 0 to 1 and each 1 to 0.

The 1st complements of 10011010 is 01100101

2nd Complement:



The 2nd complements of 10011010 is 01100110 which is start from LSB, remain the first ‘1’ and changing each 0 to 1 or vice versa after the first ‘1’

1.6.1 Signed Number of 2nd Complement:

- A signed number consists of both sign bit and magnitude bits.
- The sign bit indicates the number **Positive (0)** or **Negative (1)** at MSB.

MSB **LSB**

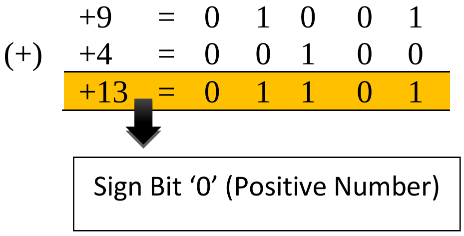
<table>
<tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>Negative Sign Bit</td><td colspan="8">Magnitude = 154_{10}</td></tr>
<tr><td colspan="9">Answer = **- 154_{10}**</td></tr>
</table>

MSB **LSB**

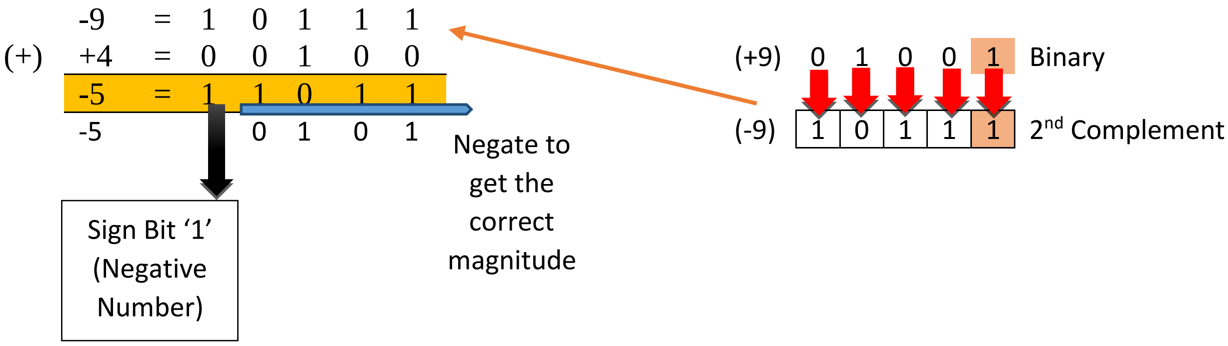
<table>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>Positive Sign Bit</td><td colspan="8">Magnitude = 154_{10}</td></tr>
<tr><td colspan="9">Answer = **+ 154_{10}**</td></tr>
</table>

1.6.2 Addition in 2nd Complement system

1. **(+9) + (+4)**



2. **(-9) + (+4)**



1.7 BCD Codes

Binary Coded Decimal (BCD) code is a different way than a straight binary coding to represent decimal numbers in binary form, but it is not a number system. BCD is widely used and combines features of both decimal and binary systems because digital systems uses binary numbers, but world is decimal in nature. Each BCD digit is converted/encoded to a 4 bits binary equivalent.

1.2 Decimal to BCD code

DECIMAL	0	1	2	3	4	5	6	7	8	9
BCD	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001

The BCD value can never be greater than 9 which can called as 8421 code mean $2^3, 2^2, 2^1, 2^0$. the primary advantage of BCD is the relative ease of converting to and from decimal. The advantages are wasting bits because it is only used 10 over 16. The example is:

DECIMAL	8	7	4
	↓	↓	↓
BCD	1000	0111	0100

So, the answer of 874_{10} after converting to BCD code is$1000\ 0111\ 0100_{BCD}$.

1.3 Relationship between number codes and BCD

Decimal	Binary	Octal	Hexadecimal	BCD
0	0	0	0	0
1	1	1	1	0001
2	10	2	2	0010
3	11	3	3	0011
4	100	4	4	0100
5	101	5	5	0101
6	110	6	6	0110
7	111	7	7	0111
8	1000	10	8	1000
9	1001	11	9	1001
10	1010	12	A	0001 0000
11	1011	13	B	0001 0001
12	1100	14	C	0001 0010
13	1101	15	D	0001 0011
14	1110	16	E	0001 0100
15	1111	17	F	0001 0101

*****For BCD: It also used 8 4 2 1**

Example: Convert 25_{10} to BCD number.

2	$\mathbf{5_{10}}$
8421	8421
0010	0101_{BCD}

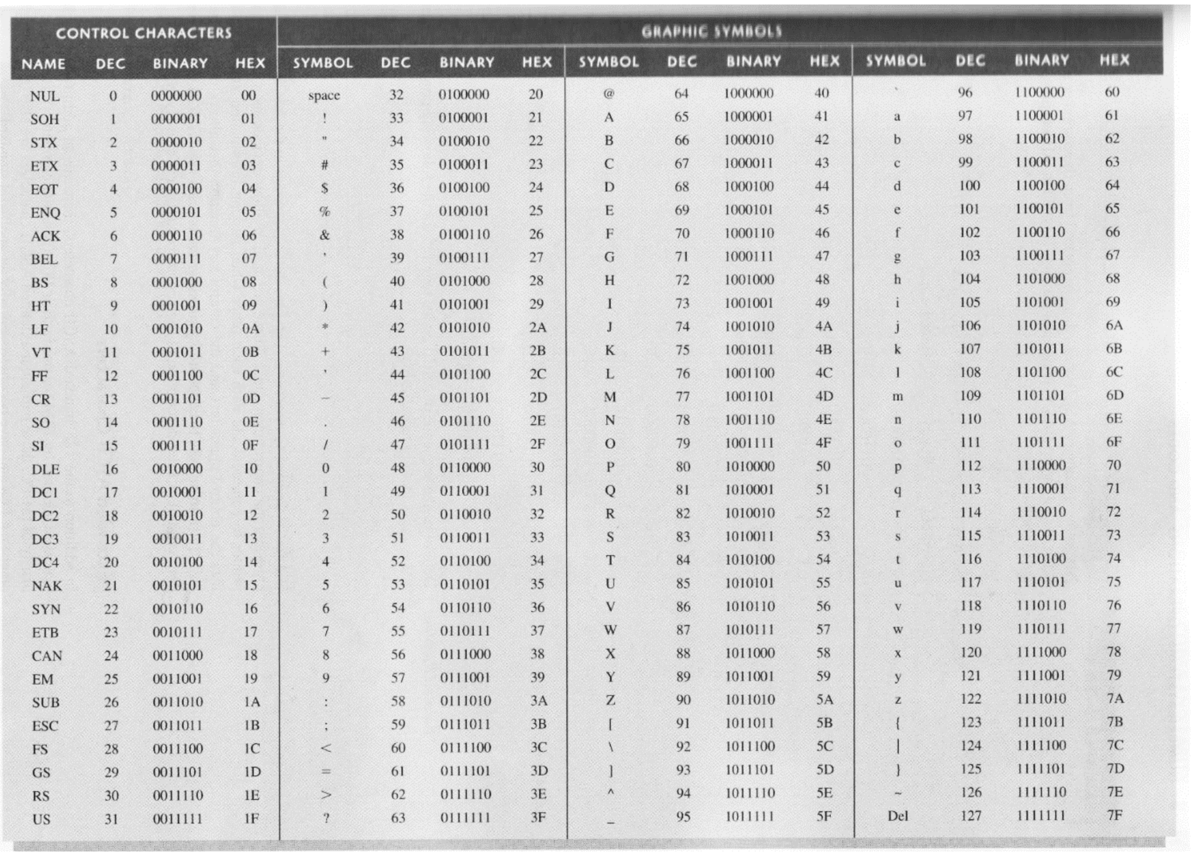
$$25_{10} = \mathbf{0010\ 0101}_{BCD}$$

1.8 ASCII Codes

The alphanumeric codes are representing characters and functions found on a computer keyboard. A complete code will include 26 lowercase letters, 26 uppercase letters, 10 numeric digits, 7 punctuation marks and anywhere from 20-40 other characters such as +, /, #, %, * and so on so forth. The most widely used code is ASCII- American Standard Code for

Information Interchange. ASCII code has 7-bit code 2^7 or 128 possible code groups. The table 1.5 list some of the ASCII code. The example will used to transfer information between computers, printers and for the internal storage.

1.9.1 ASCII table code



CONTROL CHARACTERS				GRAPHIC SYMBOLS											
NAME	DEC	BINARY	HEX	SYMBOL	DEC	BINARY	HEX	SYMBOL	DEC	BINARY	HEX	SYMBOL	DEC	BINARY	HEX
NUL	0	0000000	00	space	32	0100000	20	@	64	1000000	40	`	96	1100000	60
SOH	1	0000001	01	!	33	0100001	21	A	65	1000001	41	a	97	1100001	61
STX	2	0000010	02	"	34	0100010	22	B	66	1000010	42	b	98	1100010	62
ETX	3	0000011	03	#	35	0100011	23	C	67	1000011	43	c	99	1100011	63
EOT	4	0000100	04	$	36	0100100	24	D	68	1000100	44	d	100	1100100	64
ENQ	5	0000101	05	%	37	0100101	25	E	69	1000101	45	e	101	1100101	65
ACK	6	0000110	06	&	38	0100110	26	F	70	1000110	46	f	102	1100110	66
BEL	7	0000111	07	'	39	0100111	27	G	71	1000111	47	g	103	1100111	67
BS	8	0001000	08	(	40	0101000	28	H	72	1001000	48	h	104	1101000	68
HT	9	0001001	09	)	41	0101001	29	I	73	1001001	49	i	105	1101001	69
LF	10	0001010	0A	*	42	0101010	2A	J	74	1001010	4A	j	106	1101010	6A
VT	11	0001011	0B	+	43	0101011	2B	K	75	1001011	4B	k	107	1101011	6B
FF	12	0001100	0C	,	44	0101100	2C	L	76	1001100	4C	l	108	1101100	6C
CR	13	0001101	0D	–	45	0101101	2D	M	77	1001101	4D	m	109	1101101	6D
SO	14	0001110	0E	.	46	0101110	2E	N	78	1001110	4E	n	110	1101110	6E
SI	15	0001111	0F	/	47	0101111	2F	O	79	1001111	4F	o	111	1101111	6F
DLE	16	0010000	10	0	48	0110000	30	P	80	1010000	50	p	112	1110000	70
DC1	17	0010001	11	1	49	0110001	31	Q	81	1010001	51	q	113	1110001	71
DC2	18	0010010	12	2	50	0110010	32	R	82	1010010	52	r	114	1110010	72
DC3	19	0010011	13	3	51	0110011	33	S	83	1010011	53	s	115	1110011	73
DC4	20	0010100	14	4	52	0110100	34	T	84	1010100	54	t	116	1110100	74
NAK	21	0010101	15	5	53	0110101	35	U	85	1010101	55	u	117	1110101	75
SYN	22	0010110	16	6	54	0110110	36	V	86	1010110	56	v	118	1110110	76
ETB	23	0010111	17	7	55	0110111	37	W	87	1010111	57	w	119	1110111	77
CAN	24	0011000	18	8	56	0111000	38	X	88	1011000	58	x	120	1111000	78
EM	25	0011001	19	9	57	0111001	39	Y	89	1011001	59	y	121	1111001	79
SUB	26	0011010	1A	:	58	0111010	3A	Z	90	1011010	5A	z	122	1111010	7A
ESC	27	0011011	1B	;	59	0111011	3B	[	91	1011011	5B	{	123	1111011	7B
FS	28	0011100	1C	<	60	0111100	3C	\	92	1011100	5C	|	124	1111100	7C
GS	29	0011101	1D	=	61	0111101	3D	]	93	1011101	5D	}	125	1111101	7D
RS	30	0011110	1E	>	62	0111110	3E	^	94	1011110	5E	~	126	1111110	7E
US	31	0011111	1F	?	63	0111111	3F	_	95	1011111	5F	Del	127	1111111	7F

Refer to the table, the examples are:

a) Convert decimal to ASCII code:
   a) 18 = DC2
   b) 107 = K

b) Convert ASCII to hexadecimal:
   a) HELLO = 48 45 4C 4C 4F
   b) Hi! = 48 69 21

c) Meaning of the ASCII code:
   a) 49 20 4C 4F 56 45 20 55 = I_LOVE_U
   b) 48 45 4C 5D 2D 4D 45 = HELP_ME

SOLUTION MANUAL

1. Convert Decimal to Binary using Repeat-Division Method

   $22 = 10110_2$

2	**22**	-0
2	11	-1
2	5	-1
2	2	-0
	1	

2. Convert Decimal fraction to binary using Repeat-Multiplication.

   $0.98 = 0.1111101_2$

   0.98x2=1.96 - 1
   0.96x2=1.92 - 1
   0.92x2=1.84 - 1
   0.84x2=1.68 - 1
   0.68x2=1.36 - 1
   0.36x2=1.72 - 0
   0.72x2=1.44 - 1

3. Convert from Hexadecimal to Binary

   $59 = 01011001_2$

5	**9_{16}**
8421	8421
0101	1001_2

4. Convert from Octal to Decimal
   a) $103_8 = 67_{10}$

$$103_8 = (1\text{x}8^2) + (0\text{x}8^1) + (3\text{x}8^0) = 64 + 0 + 3 = 67_{10}$$

5. Convert 110100011_2 to the following numbers system:
   a) Decimal = 419_{10}

$$\begin{aligned}110100011_2 &= (1x2^8) + (1x2^7) + (0x2^6) + (1x2^5) + (0x2^4) + (0x2^3)\\ &+ (0x2^2) + (1x2^1) + (1x2^0)\\ &= 256 + 128 + 32 + 2 + 1 = 419_{10}\end{aligned}$$

6. Get the value for the numbers:

$$11010110_2 + 11101101_2 = 111000011_2$$

	1	1	1	1	1			
	1	1	0	1	0	1	1	0
+	1	1	1	0	1	1	0	1
1	1	1	0	0	0	0	1	1

7. Perform in 2nd Complement using 8 bits including sign bit.

a) **(+9) + (+6)**

	+9	=	0	0	0	0	1	0	0	1
(+)	+6	=	0	0	0	0	0	1	1	0
	+15	=	0	0	0	0	1	1	1	1

b) **(+14) + (-17)**

	+14	=	0	0	0	0	1	1	1	0
(+)	-17	=	1	1	1	0	1	1	1	1
	-3	=	1	1	1	1	1	1	0	1
	-3			0	0	0	0	0	1	1

Sign Bit ‘1’ (Negative Number)

(+17)	1	1	1	0	1	1	1	1	Binary
(-17)	0	0	0	1	0	0	0	1	2nd Complement

c) **23 - 48 = (+23) + (-48)**

	+23	=	0	0	0	1	0	1	1	1
(+)	-48	=	1	1	0	1	0	0	0	0
	-25	=	1	1	1	0	0	1	1	1
	-25			0	0	1	1	0	0	1

Sign Bit ‘1’ (Negative Number)

(+48)	0	0	1	1	0	0	0	0	Binary
(-48)	1	1	0	1	0	0	0	0	2nd Complement

CHAPTER 2

BOOLEAN OPERATIONS

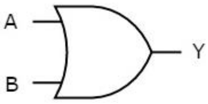
SHORT NOTE CHAPTER 2: BOOLEAN EXPRESSION

BASIC LOGIC GATE

OR GATE

OR works such that the output is true, if either of the two inputs is true. Input can be more than two.

Symbol:



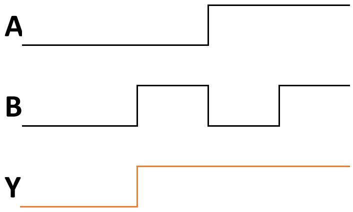
Boolean Expression:

$$Y = A + B$$

Truth table:

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

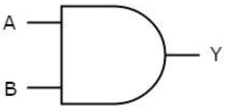
Timing diagram:



AND GATE

AND means that both conditions must be true in order for the conclusion to be true. Input can be more than two.

Symbol:



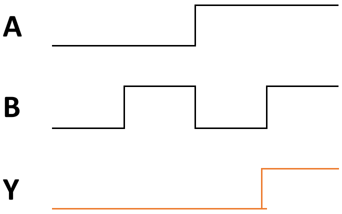
Boolean Expression:

$$Y = AB$$

Truth table:

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

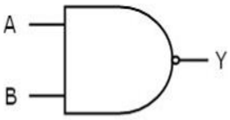
Timing diagram:



NAND GATE

Output is 1 if any or all inputs are 0. It is a combination of an AND gate followed by an inverter.

Symbol:



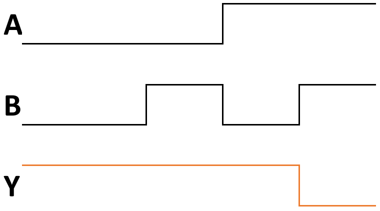
Boolean Expression:

$$Y = \overline{AB}$$

Truth table:

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

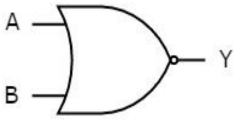
Timing diagram:



NOR GATE

The Output is 1 only if all of the inputs are 0. It is a combination of an OR gate followed by an inverter.

Symbol:



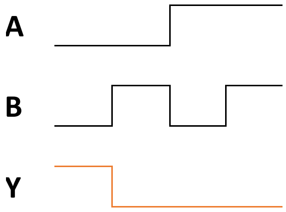
Boolean Expression:

$$Y = \overline{A + B}$$

Truth table:

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

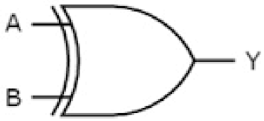
Timing diagram:



XOR (Exclusive OR) GATE

XOR works such that the Output is true, if the two logic input are opposite (different value).

Symbol:



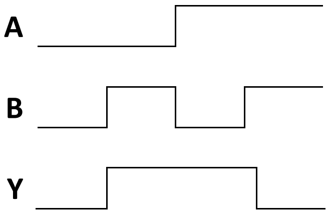
Boolean Expression:

$$Y = A \oplus B$$

Truth table:

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

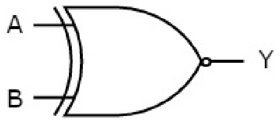
Timing diagram:



XNOR (Exclusive NOR) GATE

XNOR works such that the Output is false, if the two logic input are opposite (different value).

Symbol:



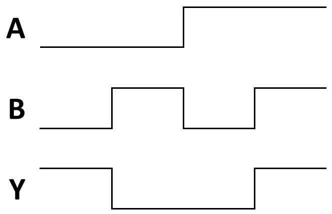
Boolean Expression:

$$Y = \overline{A \oplus B}$$

Truth table:

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	1

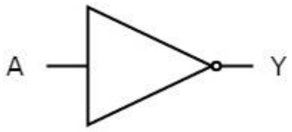
Timing diagram:



NOT/INVERTER GATE

The NOT gate reverses the input. There is only single input and single output.

Symbol:



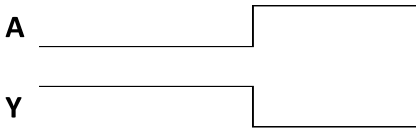
Boolean Expression:

$$Y = \overline{A}$$

Truth table:

A	Y
0	1
1	0

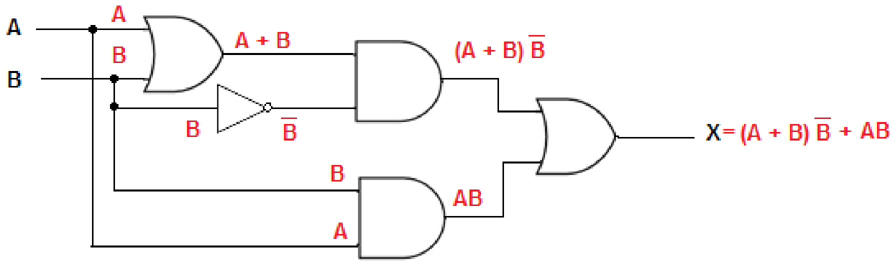
Timing diagram:



APPLICATION OF BOOLEAN EXPRESSION

a) Logic Circuit to Boolean Expression

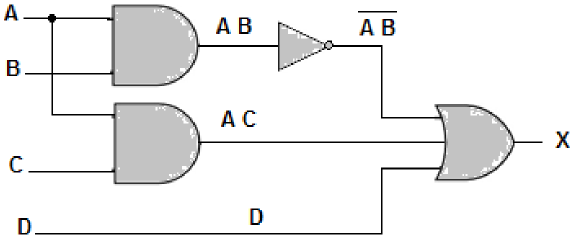
Example 1: Given logic circuit, find the Boolean Expression.



b) Boolean Expression to Logic Circuit

Example 1: Construct logic circuit from logic expression

$$Y = \overline{AB} + AC + D$$



THEOREMS OF BOOLEAN EXPRESSION

1	$A + 0 = A$	Identity Law
	$A \,.\, 1 = A$	
2	$A + 1 = 1$	Null Law
	$A \,.\, 0 = 0$	
3	$A + A = A$	Idempotent Law
	$A \,.\, A = A$	
4	$A + \overline{A} = 1$	Inverse Law
	$A \,.\, \overline{A} = 0$	
5	$\overline{\overline{A}} = A$	Double Complement Law
6	$A + \overline{A}B = A + B$	Absorption Law
	$A + AB = A$	
	$(A + B)(A + C) = A + BC$	
7	$\overline{X + Y} = \overline{X} \,.\, \overline{Y}$	De Morgan's Theorem
	$\overline{X.Y} = \overline{X} + \overline{Y}$	

Simplification Using Boolean Theorems

a. $Y = A\overline{B}D + A\overline{B}\,\overline{D}$

$$\boldsymbol{Y = A\overline{B}D + A\overline{B}\,\overline{D}}$$

$$= A\overline{B}(D + \overline{D}) \longleftarrow \textbf{Inverse Law}$$

$$= A\overline{B}(1)$$

$$= A\overline{B}$$

b. $Z = (\overline{A} + B)(A + B)$

$$\boldsymbol{Z = (\overline{A} + B)(A + B)}$$

$$= \overline{A}A + \overline{A}B + AB + BB$$

$$= 0 + B(\overline{A} + A) + B$$

$$= B + B \longleftarrow \textbf{Idempotent Law}$$

$$= B$$

Simplification Using Karnaugh Map (K-Maps)

K-Maps has a square for each ‘1’ or ‘0’ of a Boolean Function.

One variable K-Maps has $2^1 = 2$ squares

$\overline{A}$	A
0	1

Two variable K-Maps has $2^2 = 4$ squares

		0	1
		$\overline{B}$	B
0	$\overline{A}$		
1	A		

Three variable K-Maps has $2^3 = 8$ squares

		0	1
		$\overline{C}$	C
00	$\overline{A}\,\overline{B}$		
01	$\overline{A}B$		
11	AB		
10	$A\overline{B}$		

Four variable K-Maps has $2^4 = 16$ squares

		00 $\bar C\bar D$	01 $\bar C D$	11 CD	10 $C\bar D$
00	$\bar A\bar B$				
01	$\bar A B$				
11	AB				
10	$A\bar B$				

Simplification Using K-Maps

1. Simplify $X = ABC + AB\bar C$ using K-Maps

A	B	C	+	A	B	$\bar C$	=	A	B
1	1	1		1	1	0			

		0 $\bar C$	1 C
00	$\bar A\bar B$	0	0
01	$\bar A B$	0	0
11	AB	1	1
10	$A\bar B$	0	0

AB

THE SUM-OF-PRODUCT FORM (SOP) FORM

- A standard SOP expression is one in which all the variables in the domain appear in product term in the expression. Example is $A\bar BC + \bar AB\bar C$.
- **Example:**

Convert the Boolean expression $A\bar BC + \bar A\bar B + AB\bar CD$ into standard SOP form.

Solution:

The 1st term: $A\bar BC$	Missing variable D	$A\bar BC = A\bar BC(\mathbf{D} + \bar{\mathbf{D}}) = A\bar BCD + A\bar BC\bar D$
The 2nd term: $\bar A\bar B$	Missing variable C and D	$\bar A\bar B = \bar A\bar B(\mathbf{C} + \bar{\mathbf{C}}) = \bar A\bar BC + \bar A\bar B\bar C$ $\bar A\bar BC(\mathbf{D} + \bar{\mathbf{D}}) + \bar A\bar B\bar C(\mathbf{D} + \bar{\mathbf{D}})$ $= \bar A\bar BCD + \bar A\bar BC\bar D + \bar A\bar B\bar CD + \bar A\bar B\bar C\bar D$
The 3rd term: $AB\bar CD$	Standard SOP	$= AB\bar CD$
The complete standard SOP form of the original expression is a follow:		$= A\bar BCD + A\bar BC\bar D + \bar A\bar BCD + \bar A\bar BC\bar D + \bar A\bar B\bar CD +$ $\bar A\bar B\bar C\bar D + AB\bar CD$ ****if have same equation (more than one), just takes one.**

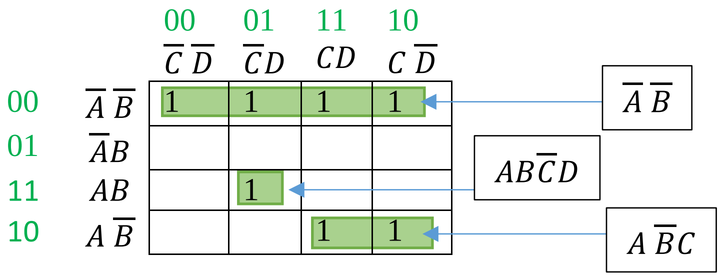
- **Develop truth table for the standard SOP expression:**

$$A\overline{B}CD + A\overline{B}C\overline{D} + \overline{A}\,\overline{B}CD + \overline{A}\,\overline{B}C\overline{D} + \overline{A}\,\overline{B}\,\overline{C}D + \overline{A}\,\overline{B}\,\overline{C}\,\overline{D} + AB\overline{C}D$$

$$\mathbf{1011} + \mathbf{1010} + \mathbf{0011} + \mathbf{0010} + \mathbf{0001} + \mathbf{0000} + \mathbf{1101}$$

INPUT				**OUTPUT**	**PRODUCT TERM**
A	**B**	**C**	**D**	**X**	
0	0	0	0	1	$\overline{A}\,\overline{B}\,\overline{C}\,\overline{D}$
0	0	0	1	1	$\overline{A}\,\overline{B}\,\overline{C}D$
0	0	1	0	1	$\overline{A}\,\overline{B}C\overline{D}$
0	0	1	1	1	$\overline{A}\,\overline{B}CD$
0	1	0	0		
0	1	0	1		
0	1	1	0		
0	1	1	1		
1	0	0	0		
1	0	0	1		
1	0	1	0	1	$A\overline{B}C\overline{D}$
1	0	1	1	1	$A\overline{B}CD$
1	1	0	0		
1	1	0	1	1	$AB\overline{C}D$
1	1	1	0		
1	1	1	1		

- **Use a K-Maps to minimize the standard SOP expression:**



		00	01	11	10
		$\overline{C}\,\overline{D}$	$\overline{C}D$	CD	$C\overline{D}$
00	$\overline{A}\,\overline{B}$	1	1	1	1
01	$\overline{A}B$				
11	AB		1		
10	$A\overline{B}$			1	1

The answer of $A\overline{B}CD + A\overline{B}C\overline{D} + \overline{A}\,\overline{B}CD + \overline{A}\,\overline{B}C\overline{D} + \overline{A}\,\overline{B}\,\overline{C}D + \overline{A}\,\overline{B}\,\overline{C}\,\overline{D} + AB\overline{C}D$ after simplify using K-maps is $\boldsymbol{\overline{A}\,\overline{B} + AB\overline{C}D + A\overline{B}C}$.

THE PRODUCT-OF-SUM (POS) FORM

- A standard POS expression is one in which all the variables in the domain appear in sum term in the expression. Example is $(A + \overline{B} + C)(\overline{A} + B + \overline{C})$.

- **Example:**

Convert the Boolean expression $\left(A+\overline{B}+C\right)\left(\overline{B}+C+\overline{D}\right)\left(A+\overline{B}+\overline{C}+D\right)$ into standard POS form.

Solution:

The 1st term: $\left(A+\overline{B}+C\right)$	Missing variable D	$\left(A+\overline{B}+C\right)+\boldsymbol{D\overline{D}}$ $=(A+\overline{B}+C+D)(\boldsymbol{A}+\overline{\boldsymbol{B}}+\boldsymbol{C}+\overline{\boldsymbol{D}})$
The 2nd term: $\left(\overline{B}+C+\overline{D}\right)$	Missing variable A	$\left(\overline{B}+C+\overline{D}\right)+\boldsymbol{A\overline{A}}$ $=(\boldsymbol{A}+\overline{\boldsymbol{B}}+\boldsymbol{C}+\overline{\boldsymbol{D}})(\overline{A}+\ \overline{B}+C+\overline{D})$
The 3rd term: $\left(A+\overline{B}+C+D\right)$	Standard SOP	$=\left(A+\overline{B}+\overline{C}+D\right)$
The complete standard SOP form of the original expression is a follow:		$=\left(A+\overline{B}+C+D\right)\left(\boldsymbol{A}+\overline{\boldsymbol{B}}+\boldsymbol{C}+\overline{\boldsymbol{D}}\right)$ $\left(\overline{A}+\ \overline{B}+C+\overline{D}\right)\left(A+\overline{B}+\overline{C}+D\right)$ ****if have same equation (more than one), just takes one.**

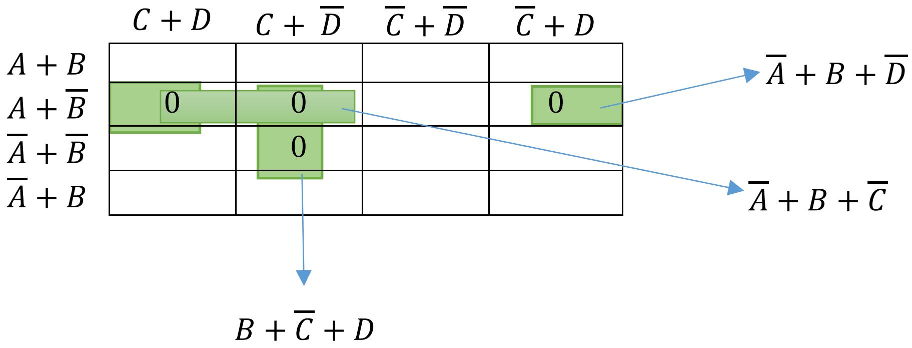
- **Develop truth table for the standard POS expression:**

$$\left(A+\overline{B}+C+D\right)\left(A+\overline{B}+C+\overline{D}\right)\left(\overline{A}+\ \overline{B}+C+\overline{D}\right)\left(A+\overline{B}+\overline{C}+D\right)$$

$$(\mathbf{0}+\mathbf{1}+\mathbf{0}+\mathbf{0})(\mathbf{0}+\mathbf{1}+\mathbf{0}+\mathbf{1})(\mathbf{1}+\mathbf{1}+\mathbf{0}+\mathbf{1})(\mathbf{0}+\mathbf{1}+\mathbf{1}+\mathbf{0})$$

INPUT				**OUTPUT**	**SUM TERM**
A	**B**	**C**	**D**	**X**	
0	0	0	0		
0	0	0	1		
0	0	1	0		
0	0	1	1		
0	1	0	0	0	$A+\overline{B}+C+D$
0	1	0	1	0	$A+\overline{B}+C+\overline{D}$
0	1	1	0	0	$A+\overline{B}+\overline{C}+D$
0	1	1	1		
1	0	0	0		
1	0	0	1		
1	0	1	0		
1	0	1	1		
1	1	0	0		
1	1	0	1	0	$\overline{A}+\ \overline{B}+C+\overline{D}$
1	1	1	0		
1	1	1	1		

Use a K-Maps to minimize the standard POS expression:

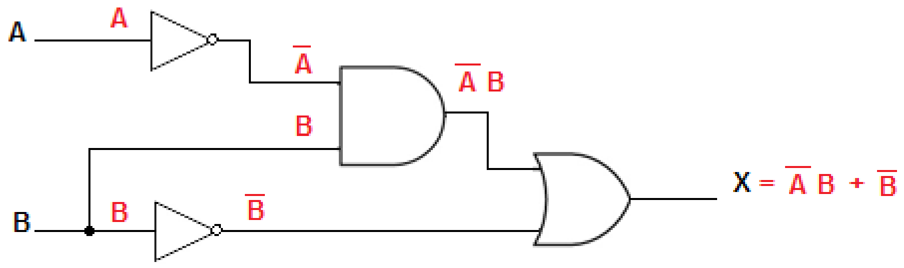


	$C+D$	$C+\overline{D}$	$\overline{C}+\overline{D}$	$\overline{C}+D$
$A+B$				
$A+\overline{B}$	0	0		0
$\overline{A}+\overline{B}$		0		
$\overline{A}+B$				

The answer of $\left(A+\overline{B}+C+D\right)\left(A+\overline{B}+C+\overline{D}\right)\left(\overline{A}+\overline{B}+C+\overline{D}\right)\left(A+\overline{B}+\overline{C}+D\right)$ after simplify using K-maps is $(\boldsymbol{B}+\overline{\boldsymbol{C}}+\boldsymbol{D})(\overline{\boldsymbol{A}}+\boldsymbol{B}+\overline{\boldsymbol{C}})(\overline{\boldsymbol{A}}+\boldsymbol{B}+\overline{\boldsymbol{D}})$.

a) Logic Circuit to Boolean Expression

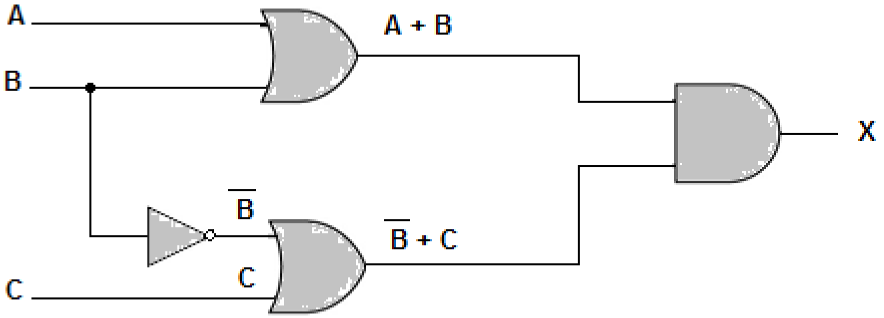
Given logic circuit, find the Boolean Expression.



b) Boolean Expression to Logic Circuit

Construct logic circuit from logic expression

$$X = (A + B)(\overline{B} + C)$$



c) Simplification Using Boolean Theorems

a. $Z = \overline{(\overline{A} + C)(B + \overline{D})}$

$$\boldsymbol{Z} = \overline{(\overline{\mathbf{A}} + \mathbf{C})(\mathbf{B} + \overline{\mathbf{D}})}$$

$$= \overline{(\overline{A} + C)} + \overline{(B + \overline{D})}$$

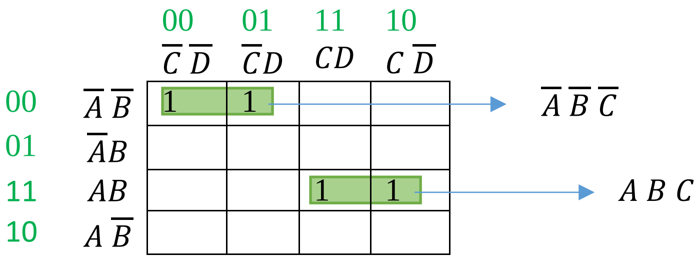
$$= \overline{\overline{A}}\,\overline{C} + \overline{B}\,\overline{\overline{D}}$$

$$= A\overline{C} + \overline{B}D$$

d) Use a Karnaugh Map to simplify each expression to a minimum SOP form.

$$ABC\overline{D} + \overline{A}\,\overline{B}\,\overline{C}D + \overline{A}\,\overline{B}\,\overline{C}\,\overline{D} + ABCD$$

1 1 1 0 + 0 0 0 1 + 0 0 0 0 + 1 1 1 1



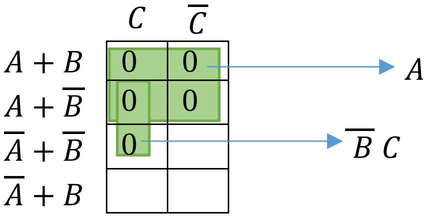
		00	01	11	10
		$\overline{C}\,\overline{D}$	$\overline{C}D$	CD	$C\,\overline{D}$
00	$\overline{A}\,\overline{B}$	1	1		
01	$\overline{A}B$				
11	AB			1	1
10	$A\,\overline{B}$				

So the answer is $\boldsymbol{\overline{A}\,\overline{B}\,\overline{C} + A\,B\,C}$

e) Use a Karnaugh Map to simplify each expression to a minimum POS form

$$(A+B+C)(A+B+\overline{C})(A+\overline{B}+C)(A+\overline{B}+\overline{C})(\overline{A}+\overline{B}+C)$$

(0 + 0 + 0) (0 + 0 + 1) (0 + 1 + 0) (0 + 1 + 1) (1 + 1 + 0)



	C	$\overline{C}$
$A+B$	0	0
$A+\overline{B}$	0	0
$\overline{A}+\overline{B}$	0	
$\overline{A}+B$		

So the answer is $\boldsymbol{A(\overline{B} + C)}$

CHAPTER 3

SEQUENTIAL LOGIC

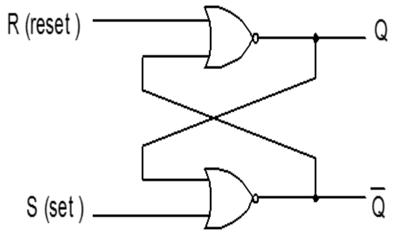
SHORT NOTE CHAPTER 3: SEQUENTIAL LOGIC

Definition:
A sequential logic circuit which has two stable states. A circuit that has two stable state and can be used to store state information. It can store 1-bit binary number (1 or 0)

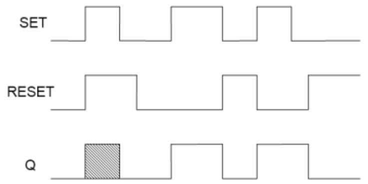
Types of Flip Flop

Symbol/ Circuit | **Truth Table** | **Timing Diagram**

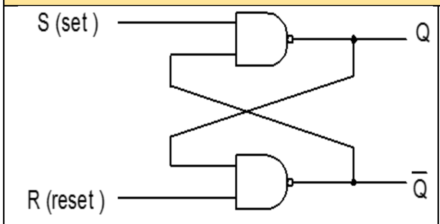
NOR GATE SR LATCH



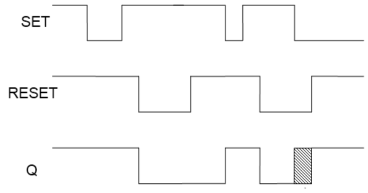
S	R	Q	$\bar{Q}$	OPERATION
0	0	0	1	No change
0	1	0	1	Reset
1	0	1	0	Set
1	1	1	1	Invalid



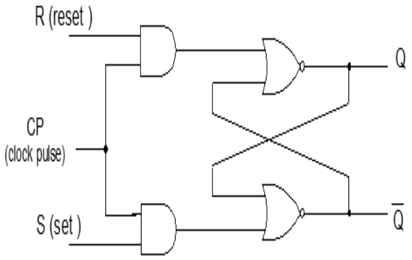
NAND GATE SR LATCH



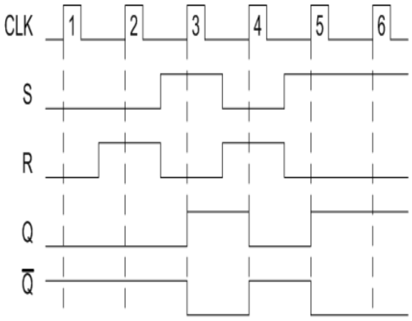
S	R	Q	$\bar{Q}$	OPERATION
0	0	1	1	Invalid
0	1	1	0	Set
1	0	0	1	Reset
1	1	0	1	No change



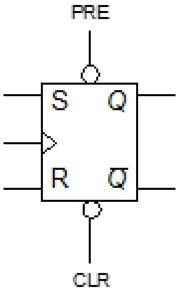
CLOCKED SR



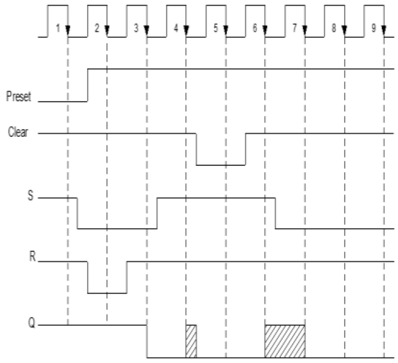
Q	S	R	$Q_{(t+1)}$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	Invalid
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	Invalid



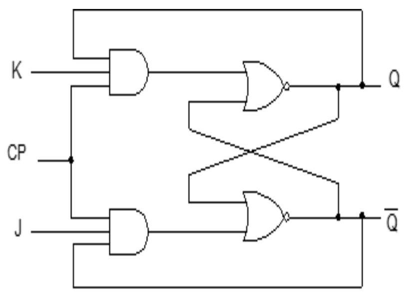
CLOCKED SR WITH PRESET CLEAR

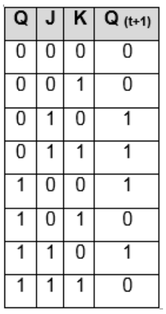


Mode	Input					Output
	Asynchronous		Synchronous			
	Preset	Clear	S	R	CLK	$Q_{(t+1)}$
Asynchronous set	0	1	x	x	x	1
Asynchronous reset	1	0	x	x	x	0
No change	1	1	0	0	1	No change
Reset	1	1	0	1	1	0
Set	1	1	1	0	1	1
Invalid	1	1	1	1	1	Invalid

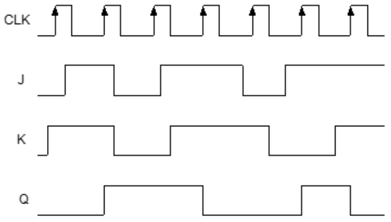


JK FLIP FLOP

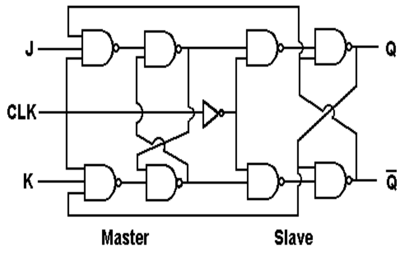


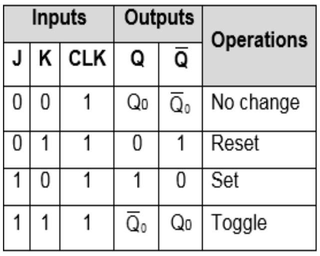


Q	J	K	$Q_{(t+1)}$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

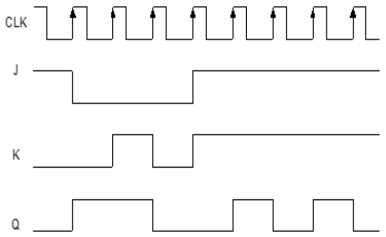


MASTER/SLAVE JK FLIP FLOP

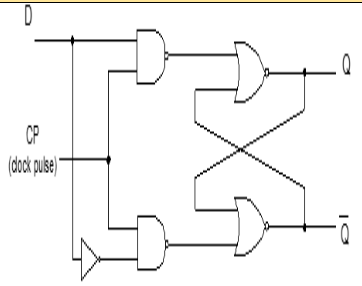


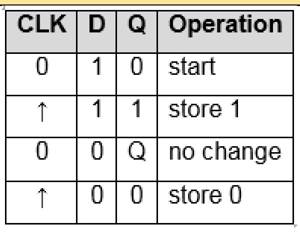


Inputs			Outputs		Operations
J	K	CLK	Q	$\bar{Q}$	
0	0	1	Q_0	$\bar{Q}_0$	No change
0	1	1	0	1	Reset
1	0	1	1	0	Set
1	1	1	$\bar{Q}_0$	Q_0	Toggle

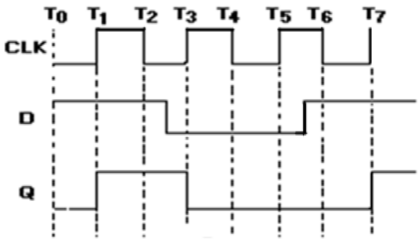


D FLIP FLOP

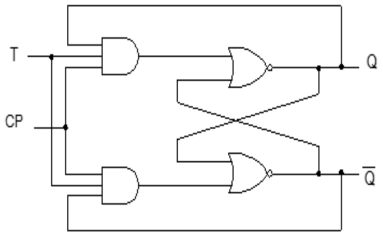


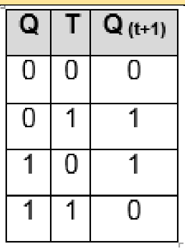


CLK	D	Q	Operation
0	1	0	start
↑	1	1	store 1
0	0	Q	no change
↑	0	0	store 0

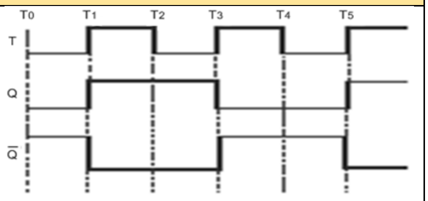


T FLIP FLOP

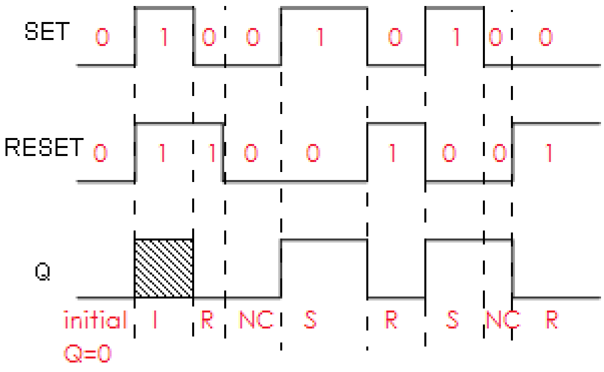




Q	T	$Q_{(t+1)}$
0	0	0
0	1	1
1	0	1
1	1	0

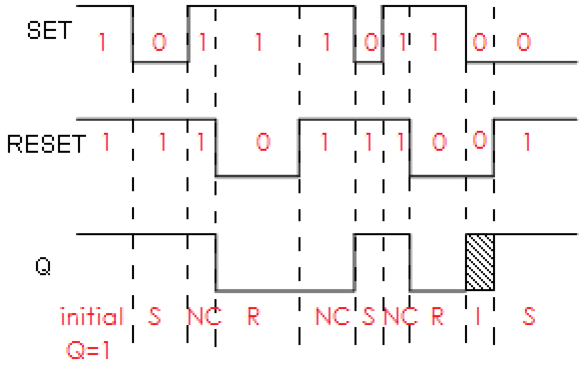


a) Sketch the output Q for SR NOR gate flip flop.



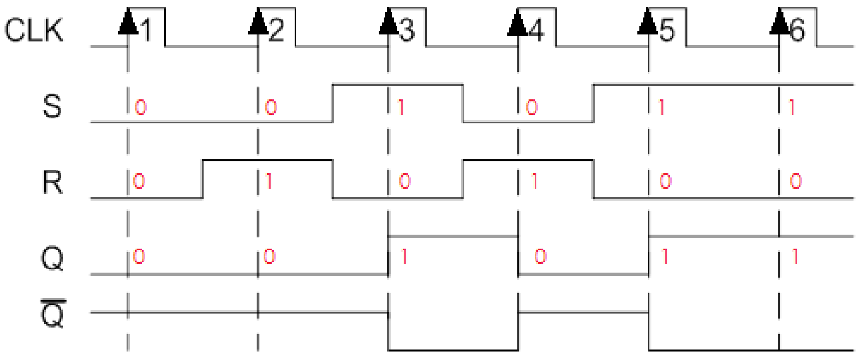
S	R	Q	$\bar{Q}$	OPERATION
0	0	0	1	No change
0	1	0	1	Reset
1	0	1	0	Set
1	1	1	1	Invalid

b) Sketch the output Q and Q' for SR NAND gate flip flop. Assuming Qinitial = 1.



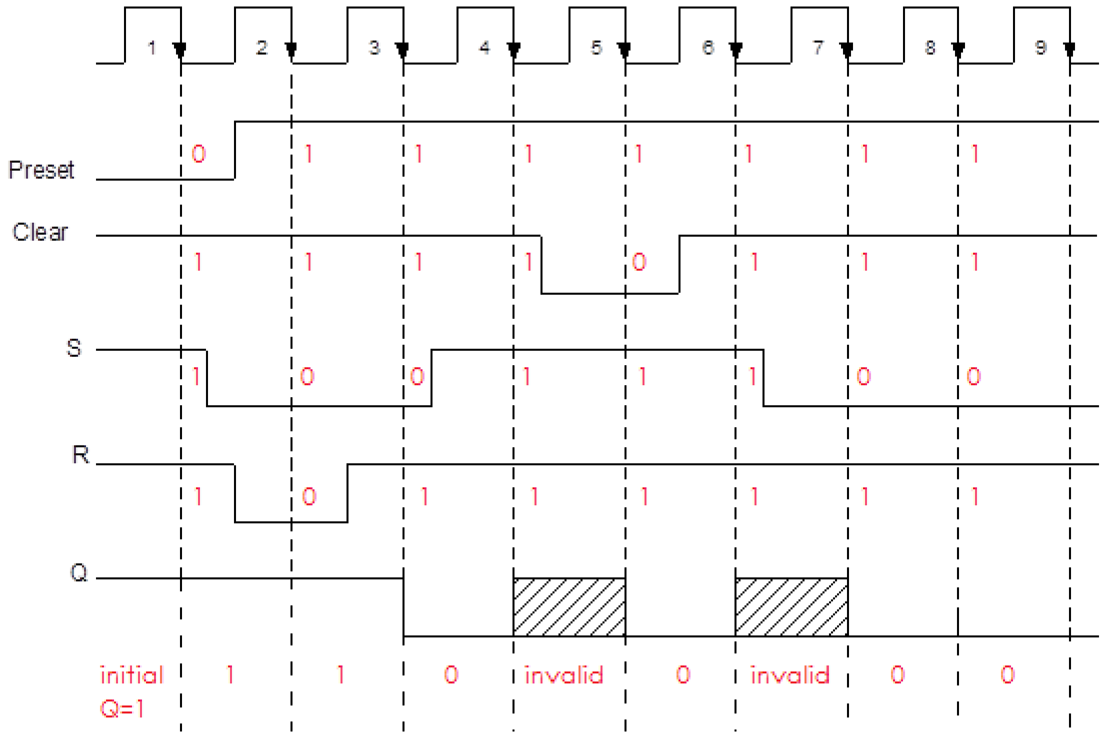
S	R	Q	$\bar{Q}$	OPERATION
0	0	1	1	Invalid
0	1	1	0	Set
1	0	0	1	Reset
1	1	0	1	No change

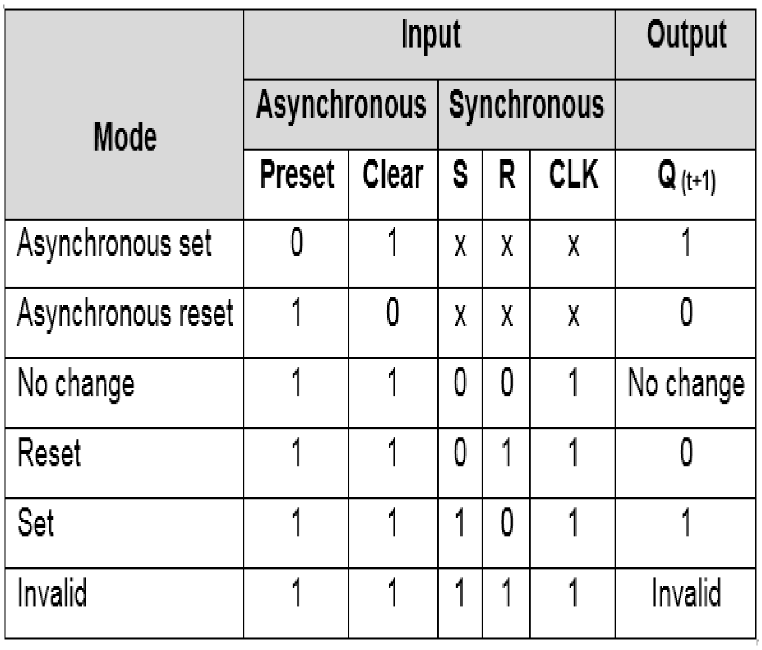
c) Determine the Q and $\bar{Q}$ output waveforms of the flip flop in Figure below for SR Clocked inputs. Assume that the Positive Edge Trigger Clock (PGT) flip flop is initially RESET.



Q	S	R	$Q_{(t+1)}$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	Invalid
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	Invalid

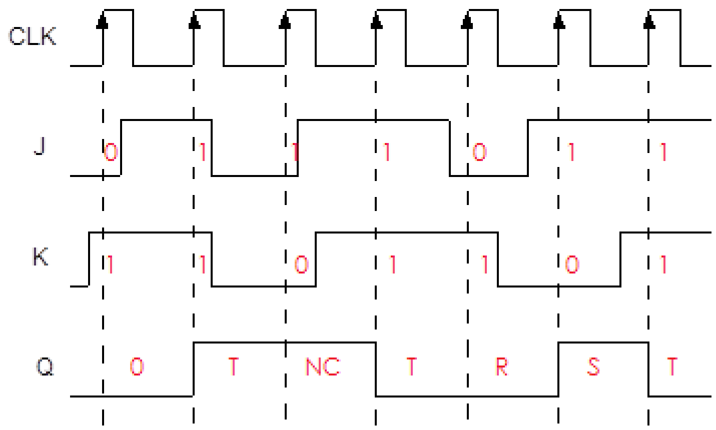
d) Determine the Q output waveforms of the flip flop in Figure below for SR Clocked with Preset and Clear input control. Assume that the Negative Edge Trigger Clock flip flop is with Qinitial = 1.





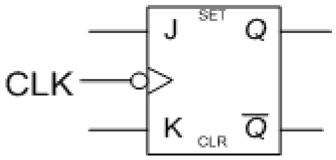
Mode	Input					Output
	Asynchronous		Synchronous			
	Preset	Clear	S	R	CLK	Q (t+1)
Asynchronous set	0	1	x	x	x	1
Asynchronous reset	1	0	x	x	x	0
No change	1	1	0	0	1	No change
Reset	1	1	0	1	1	0
Set	1	1	1	0	1	1
Invalid	1	1	1	1	1	Invalid

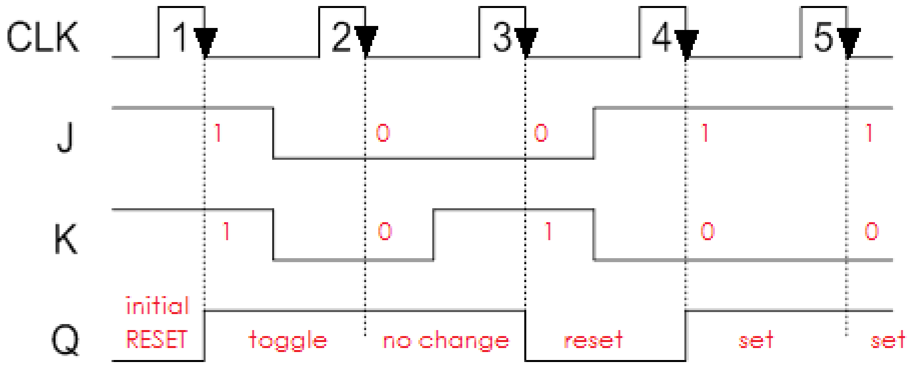
e) Sketch the output Q for JK flip flop.



Q	J	K	Q (t+1)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

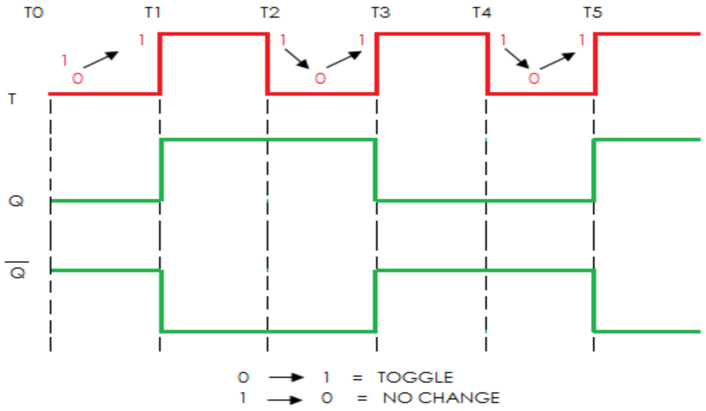
f) The waveforms in figure (a) below are applied to the J, K, and clock inputs are indicated. Determine the Q output, assuming that the flip-flop is initially RESET.





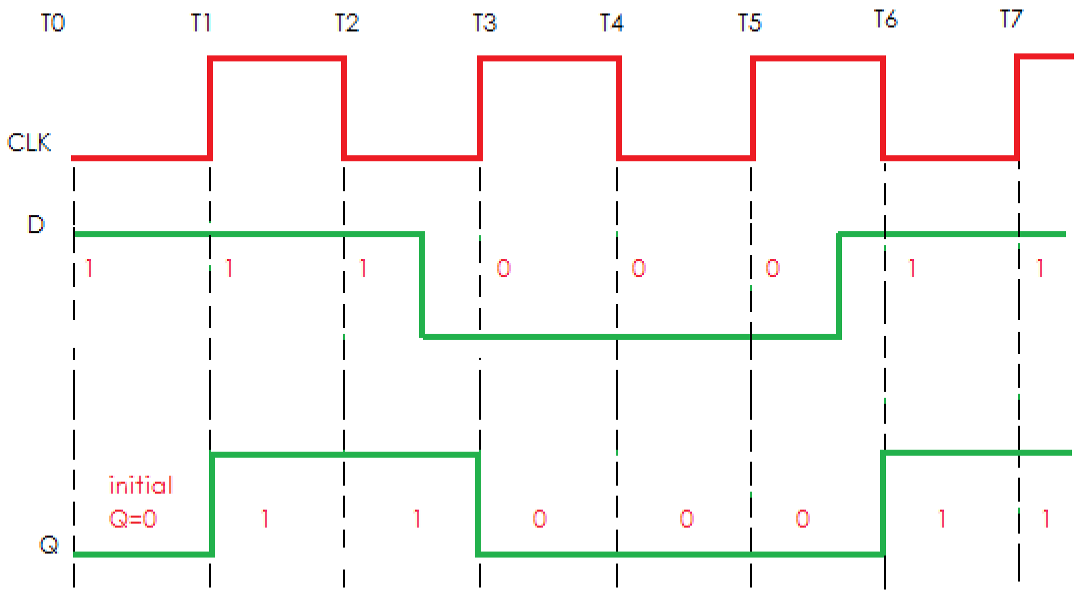
Q	J	K	Q (t+1)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

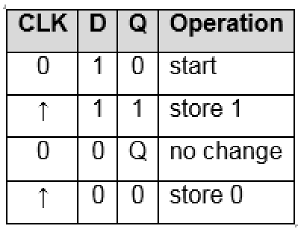
g) Determine the Q and $\bar{Q}$ output waveforms of the flip flop in Figure below for T inputs. Assume that the Q initially LOW.



Q	T	Q (t+1)
0	0	0
0	1	1
1	0	1
1	1	0

h) Determine the Q output waveforms of the flip flop in Figure below for D inputs. Assume that the Q initially LOW.





CLK	D	Q	Operation
0	1	0	start
↑	1	1	store 1
0	0	Q	no change
↑	0	0	store 0

REFERENCES

Anil K.Maini (2007), "Digital Electronics:principles, devices and applications" John Wiley & Sons, England.

Dr. Kuo-pao Yang, CMPS375 Class Notes (Chap03), Boolean Algebra and Digital Logic, page 1-28.

Dr. I. J. Wassell, Digital Electronics Part I – Combinational and Sequential Logic - Sequential Logic Flip-flops and Latches, page 128-157

Eric Coates MA BSc, Learn about electronics Digital Electronics - Numbers System (Revision 15.00 29th December 2020).

Floyd, T. L. (2003). *Digital Fundamentals with Vhdl*. Prentice Hall.

Kaushik, D. K. (2005). *Digital Electronics*. Dhanpat Rai Publishing Company, New Delhi.

Maini, A. K. (2008). Register. In *Digital Electronics: Principles, devices and applications*. essay, Wiley India.

Maini, A. K. (n.d.). *Digital Electronics: Principles, devices and applications*. Retrieved August 29, 2022, from https://www.shahucollegelatur.org.in/Department/Studymaterial/sci/it/BCA/FY/digielec.pdf

Salina, M., & Khalid, M. N. M. (2013). counter. In *Digital Electronics*. essay, Oxford Unicersity Press.

Storr, W. (2022, August 6). *Synchronous counter and Decade counter tutorial*. Basic Electronics Tutorials. Retrieved August 29, 2022, from https://www.electronics-tutorials.ws/counter/count_3.html

KEMENTERIAN PENGAJIAN TINGGI

POLITEKNIK MALAYSIA

e ISBN 978-967-2740-22-3

9 7 8 9 6 7 2 7 4 0 2 2 3

Digital System Revision