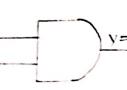
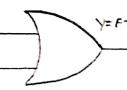
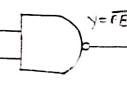
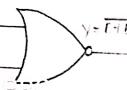


Sr No	Gate	Symbol	Boolean algebraic expression	Truth Table		
				Input A	Output P	
1	NOT		$y = \bar{F}$	0	1	
				1	0	
2	AND		$y = F \cdot E$	A	B	$y = AB$
				0	0	0
				0	1	0
				1	0	0
				1	1	1
3	OR		$y = F + E$	A	B	$y = A + B$
				0	0	0
				0	1	1
				1	0	1
				1	1	1
4	NAND		$y = \bar{F} \cdot \bar{E}$	A	B	$y = \bar{AB}$
				0	0	1
				0	1	1
				1	0	1
				1	1	0
5	NOR		$y = \bar{F} + \bar{E}$	A	B	$y = \bar{A} + \bar{B}$
				0	0	1
				0	1	0
				1	0	0
				1	1	0

REDMI NOTE 8 PRO
QDBS

Date 09/01/2015
Expt. No. 1
Page No. 1

Experiment No. 1

Aim: Simplification and realization of Boolean expressions using logic gates.

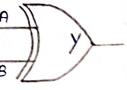
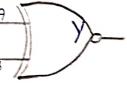
Components : IC, digital IC trainer, (Multimeter)

Theory :

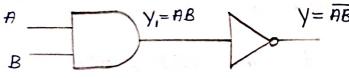
In electronics a logic gate is an idealized or physical device implementing a Boolean function, that is, it performs a logical operation on one or more logical inputs, and produces a single logical output. Logic gates are primarily implemented using diodes or transistors.

The important logic gates are NOT, AND, OR, NAND, NOR, XOR, XNOR. In electronics a NOT gate is more commonly called an inverter. NOT gate gives output true (1) for false (0) input and false (0) output for true (1) input. If it is used to construct various other gates to get output as complement of the input. AND gate gives output true (1) only when both the inputs are true while it gives zero for other cases. OR gate gives output true (1) if either one of the inputs or

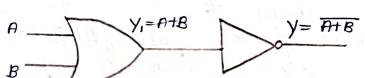
Teacher's Signature : _____

Sr No	Gate	Symbol	Boolean Algebraic expression	Truth Table															
6	XOR		$A \oplus B$	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	A	B	y	0	0	0	0	1	1	1	0	1	1	1	0
A	B	y																	
0	0	0																	
0	1	1																	
1	0	1																	
1	1	0																	
7	XNOR		$\overline{A} \oplus \overline{B}$	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	A	B	y	0	0	1	0	1	0	1	0	0	1	1	1
A	B	y																	
0	0	1																	
0	1	0																	
1	0	0																	
1	1	1																	

NAND Gate using AND and NOT gates

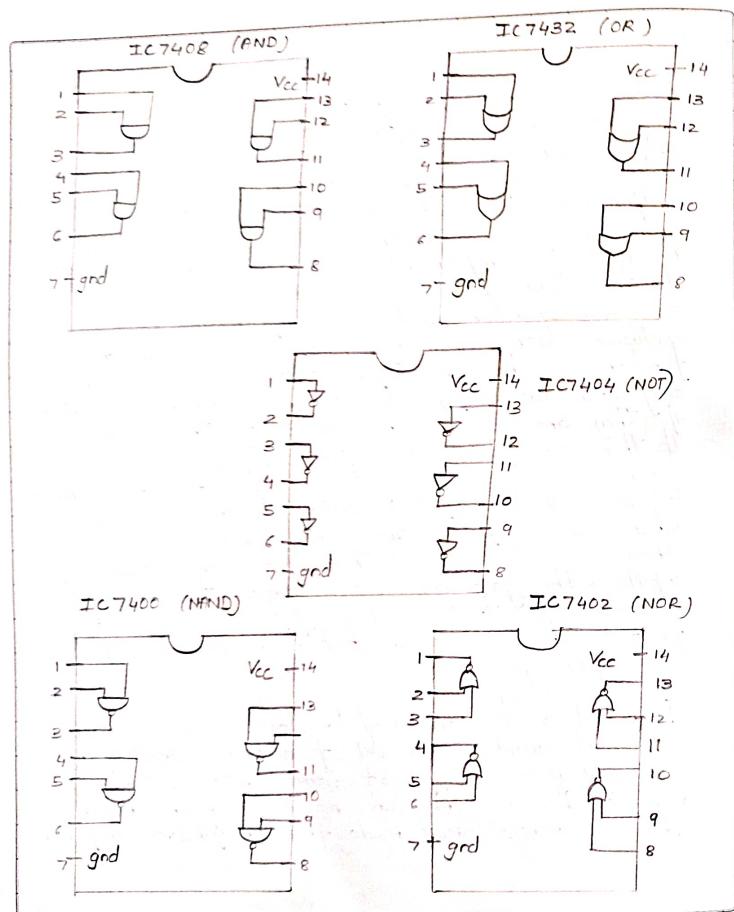


NOR Gate using OR and NOT gates

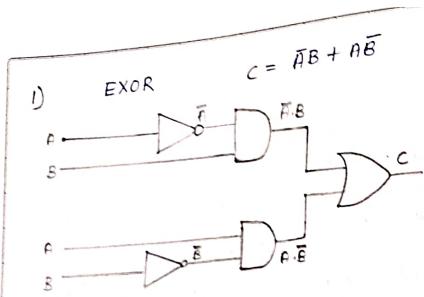


Expt. No.	Date _____	Page No. 2
<p>both are true (1). If both inputs are false (0) it gives false (0) output. NAND gate gives outputs complementary to that of AND for same (peripheral) inputs while NOR gate gives complementary outputs to that of OR. NAND gate and NOR gate are called universal gates as they can be used to implement Boolean expressions directly. Two more gates are the exclusive-OR or XOR function and its complement, the exclusive-NOR or XNOR or EGV (equivalent) function. The two input exclusive-OR is true only when two input values are different, false if they are equal, regardless of the value. If there are more than two inputs, the gate generates a true at its output if the number of true at its input is odd. X-NOR is true when two input values are same, false if they are different, if there are more than two inputs, the gate generates a true at its output if number of true at its input is even. In practice, these gates are built from combination of simpler logic gates.</p> <p>Integrated circuits are used to design gates. An integrated circuit is a set of electronic circuits on one small plate ("chip") of semiconductor material normally silicon. This can be made much smaller than a discrete circuit made from independent components.</p>		

Teacher's Signature :

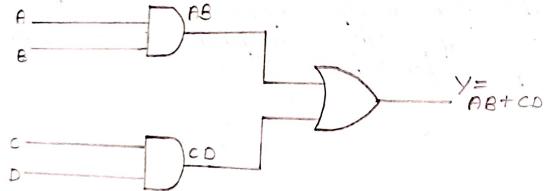


Expt. No.	Date Page No.
Procedure <ol style="list-style-type: none"> Identify the IC for different logic gates (NOT, AND, OR, NAND, NOR). Evaluate the given Boolean algebraic expression and thus design the circuit. Insert the IC in digital IC trainer. Connect the cords (wires) to the digital IC trainer at the input and output as per the designed circuit. Use multiple IC for more than one logic gate (combination of logic gates). Connect it to $V_{CC} = 5V$ and another cord to ground. Try for different circuits and logic gates by switching ON the inputs to the corresponding light for multiple inputs. Check the light corresponding to the output. Repeat for other circuits. 	
Teacher's Signature : _____	



2)
$$Y = ABCD + A\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}CD$$

$$= AB + CD$$



Date
 Expt. No. Page No. 41

Observations and Discussions:

Observation:

1) XOR gate using NOT, AND and OR

$$C = \bar{A}B + A\bar{B}$$

Theoretical:

Input A	Input B	\bar{A}	\bar{B}	$\bar{A}B$	$A\bar{B}$	$\bar{A}B + A\bar{B} = C$
0	0	1	1	0	0	0
0	1	1	0	1	0	1
1	0	0	1	0	1	1
1	1	0	0	0	0	0

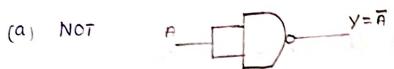
Practical:

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

Teacher's Signature :

Realization using NAND/NOR (universal) gates

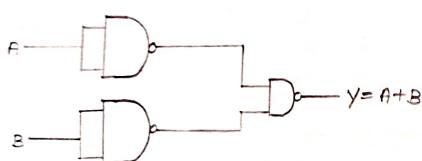
(i) Using NAND Gate



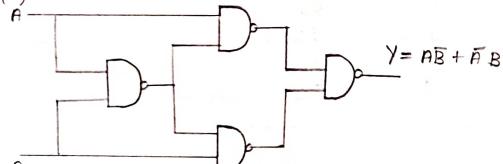
(b) AND



(c) OR



(d) XOR



Date
Expt. No. Page No. ... 5

2) $y = ABCD + ABC\bar{D} + A\bar{B}CD$
 $= AB + CD$

Theoretical / practical

A	B	C	D	AB	CD	AB + CD
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	0	1	0	0	0	0
0	0	1	1	0	1	1
0	1	0	0	0	0	0
0	1	0	1	0	0	0
0	1	1	0	0	0	0
0	1	1	1	0	1	1
1	0	0	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	0	0	0
1	0	1	1	0	1	1
1	1	0	0	1	0	1
1	1	0	1	1	0	1
1	1	1	0	1	0	1
1	1	1	1	1	1	1

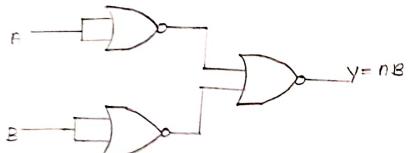
Teacher's Signature : _____

(ii) Using NOR gate

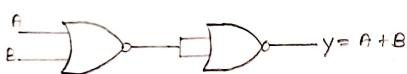
(a) NOT



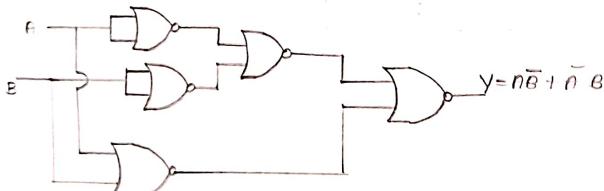
(b) AND



(c) OR



(d) XOR



Date
Expt. No.

Page No. 6.....

Discussions:

In this experiment we have studied the realization of logic gates. AND, OR, NOR gates are used to get different outputs along with NOR, NAND, which are also called universal gates. Universal gates can be used to construct (design) a circuit so as to obtain any required output. NOR and NAND gate can be constructed from OR and AND gates respectively by inverting with NOT gate. XOR and XNOR are exclusively-OR and exclusively-NOR gates and can be designed from universal gates. These gates can be used to design any digital circuits with universal gates able to design NOT, AND, OR, XNOR, XOR gates directly.

Result: Boolean functions were practically implemented by using logic gates. Logic inputs are driven by voltages having two nominal values representing logic 0 and logic 1. The output of a gate provides two nominal values of voltage only representing logic 0 and 1. In general, there is only one output to a logic gate. AND, OR, NOT, NAND, NOR, XOR and XNOR gates were designed and used to construct a circuit.

Conclusion: Boolean functions were practically implemented by using logic gates and thus circuits were designed.

Teacher's Signature : _____

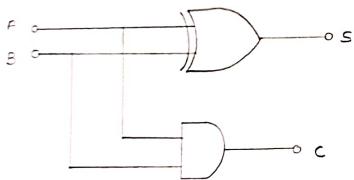
Observation and Results:

(a) half adder

Input		Output	
A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$$S = \bar{A}B + A\bar{B}$$

$$C = AB$$



Expt. No. 4

Page No. ... 15

Experiment No - 4

Aim : (a) Realization of half adder and full adder using logic gates.
(b) Realization of half subtractor and full subtractor using logic gates.

Components : Digital IC Trainer, cards, IC 7408, IC 7432, IC 7404, IC 7486

Theory:

Half adder:

The half adder adds two single binary digits A and B. It has two outputs, sum (S) and carry (C). The carry signal represents an overflow into the next digit of a multi-digit addition. The value of the sum is $2C + S$. The basic half adder design consists of an OR gate for S and an AND gate for C. With the addition of an OR gate to combine their carry outputs, two half adders can be combined to make a full adder.

full adder:

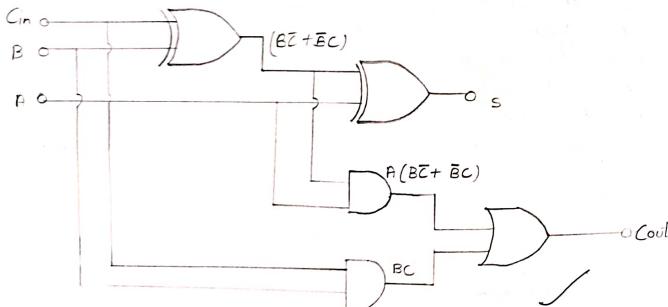
A one bit full adder adds three one-bit numbers A, B and C_{in} . A and B are the operands and C_{in}

Teacher's Signature : _____

(b) full adder

Input	Output			
A	B	C _i	S	C _{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$\begin{aligned}
 S &= \bar{A}\bar{B}C_i + \bar{A}B\bar{C}_i + A\bar{B}\bar{C}_i + ABC_i \\
 &= \bar{A}(\bar{B}C_i + BC_i) + A(B\bar{C}_i + \bar{B}C_i) \\
 &= \bar{A}x + A\bar{x} \\
 C_o &= \bar{A}BC_i + A\bar{B}C_i + ABC_i + ABC_i \\
 &= A(\bar{B}C_i + BC_i) + BC_i
 \end{aligned}$$



Expt. No.

Date

Page No. 16

is a bit carried from previous less significant stage. The circuit produces a 2-bit output, carry and sum typically represented by signals (and s, where sum = 2 × cout + s)

Half Subtractor:

The half subtractor is a combinational circuit which is used to perform subtraction of two bits. It has two inputs, X (minuend) and Y (subtrahend) and two outputs, D (difference) and B (borrow). $D = A \oplus B$ or $(B \cdot \bar{A})$ ($B = \bar{A}B$ (borrow))

full subtractor:

The full subtractor is a combinational circuit which is used to perform subtraction of three bits. It has three inputs, X (minuend) and Y (subtrahend) and Z (subtrahend) and two outputs D (difference) and B (borrow). $D = x - y - z$ (without sign) $B = 1$ if $x \leq (y+z)$

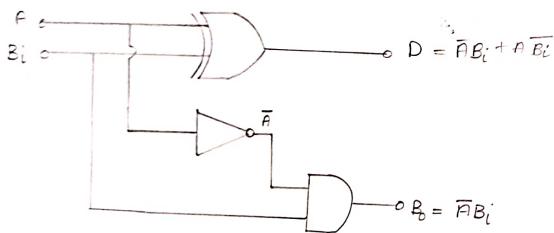
Teacher's Signature : _____

(c) half subtractor

input		output	
A	B	D	C _o
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

$$D = \bar{A}B_i + A\bar{B}_i$$

$$C_o = \bar{A}B_i$$



✓

Expt. No.

Date

Page No. 17

Procedure :

- Draw the truth table for each problem by solving the inputs accordingly to give the outputs
- Using truth table write the boolean expression for sum and (carry / adder), Difference and borrow / Subtractor
- Solve the boolean algebraic expression or by k-map
- Design the circuit according to the final expressions

Calculations (expressions)

(i) half adder

$$S = \bar{A}B + A\bar{B}$$

$$C = AB$$

(ii) full adder

$$S = \bar{A}\bar{B}C + \bar{A}BC + A\bar{B}C + ABC$$

$$= \bar{A}(\bar{B}C + BC) + A(BC + \bar{B}C)$$

$$= \bar{A}X + AX$$

$$C = \bar{A}BC + A\bar{B}C + A\bar{B}C + ABC$$

$$= A(\bar{B}C + BC) + BC$$

$$= BC + AC + AB$$

(iii) half subtractor

$$D = \bar{A}B + n\bar{B}$$

$$C = \bar{A}B$$

(iv) full subtractor

$$D = \bar{A}\bar{B}C + \bar{A}BC + A\bar{B}C + ABC$$

$$B = \bar{A}\bar{B}C + \bar{A}BC + ABC$$

$$= \bar{A}(\bar{B}C + BC) + A(BC + \bar{B}C)$$

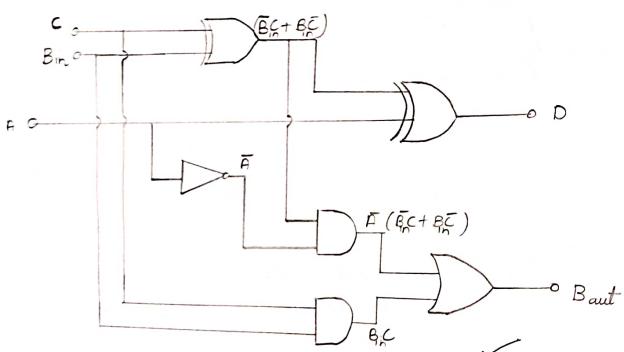
$$= \bar{A}(BC + \bar{B}C) + BC$$

Teacher's Signature : _____

(d) full subtractor

	input		output	
#	B_{in}	C	D	B_{out}
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

$$\begin{aligned}
 D &= \bar{A}\bar{B}_{in}C + \bar{A}\bar{B}_{in}\bar{C} + A\bar{B}_{in} \\
 &\quad + ABC \\
 &= A(\bar{B}_{in}C + \bar{B}_{in}\bar{C}) + A(B_{in} + \bar{B}_{in}) \\
 &= A\bar{X} + A\bar{X} \\
 B_{out} &= \bar{A}\bar{B}_{in}C + \bar{A}\bar{B}_{in}\bar{C} + \bar{A}B_{in}(A\bar{B}_{in}) \\
 &= \bar{A}(\bar{B}_{in}C + \bar{B}_{in}\bar{C}) + B_{in}C
 \end{aligned}$$



Expt. No.

Page No. 18

Discussions:

In electronics, an adder or summer is a digital circuit that performs addition of numbers. Although adders can be constructed for many numerical representations, such as binary-coded decimal or excess-3, the most common adders operate on binary numbers. The half adder adds two input bits and generates a carry and sum. A one-bit full adder adds three-one bit numbers, A, B and C_{in} to give outputs sum and S. A full adder can be implemented in different ways, one example is taken into consideration to perform experiment. As with an adder, in the general case of calculations on multi-bit numbers, three bits are involved in performing subtraction for each bit of difference. The minuend (x), subtrahend (y), and a borrow in from the previous (less significant) bit order position (B_i). The outputs are the difference bit (D_i) and borrow bit.

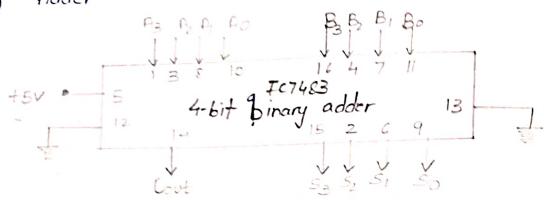
Conclusion: Half adder, full adder, half subtractor, full subtractor circuits were analysed and designed and the output was verified.

checked
by
S. G. Dinesh

Teacher's Signature : _____

Observations and Results :

(1) Adder $\rightarrow A + B$



Binary

Input		Output	
A	B	C	S
A ₃ A ₂ A ₁ A ₀	B ₃ B ₂ B ₁ B ₀	C	S ₃ S ₂ S ₁ S ₀
0 0 0 0	0 1 1 0	0	0 0 1 1 0
0 0 0 1	0 1 1 1	0	1 0 0 0 0
0 0 1 0	1 0 0 0	0	1 0 1 0
0 0 1 1	1 0 0 1	0	1 1 0 0
0 1 0 0	1 0 1 0	0	1 1 1 0
0 1 0 1	1 0 1 1	1	0 0 0 0
0 1 1 0	1 1 0 0	1	0 0 1 0
1 0 0 0	1 1 1 0	1	0 1 1 0
1 0 0 1	1 1 1 1	1	1 0 0 0
0 1 1 1	1 1 0 1	1	0 1 0 0

$$\begin{array}{r}
 0010 \\
 +1000 \\
 \hline
 1010
 \end{array}
 \quad
 \begin{array}{r}
 0101 \\
 +1011 \\
 \hline
 10000
 \end{array}
 \quad
 \begin{array}{r}
 1001 \\
 +1111 \\
 \hline
 11000
 \end{array}$$

Decimal

Input		Output	
A	B	S	
0	6	6	
1	7	8	
2	8	10	
3	9	12	
4	10	14	
5	11	16	
6	12	18	
8	14	22	
9	15	24	
7	13	20	

Date 13/02/2015

Page No. 19

Expt. No. 5

Experiment No 5

Aim : Realization of 4-bit parallel adder and subtractor.

Components : Digital IC trainer, IC 7483, IC 7482, cards.

Theory :

The addition of two binary numbers is performed in exactly the same way as the addition of decimal numbers. In adding two binary digits only following four cases can occur.

$$0 + 0 = 0$$

$$1 + 0 = 1$$

$$1 + 1 = 10 = 0 + \text{Carry of 1 into next position}$$

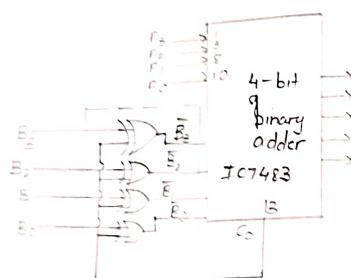
$$1 + 1 + 1 = 11 = 1 + \text{Carry of 1 into next position}$$

The last case occurs when the two bits in certain position are 1 and there is a carry from the previous position.

$$\begin{array}{r}
 011 \quad (3) \quad 1001 \quad (9) \\
 +110 \quad (6) \quad +1111 \quad (15) \\
 \hline
 1001 \quad (9) \quad 11000 \quad (24)
 \end{array}$$

Teacher's Signature :

P) Subtractor $\rightarrow A - B$



Input		Output				
F ₃	F ₂	F ₁	F ₀	B	C	S
0	0	0	0	1	1	1
0	0	0	1	1	1	0
0	0	1	0	1	0	1
0	0	1	1	0	0	1
0	1	0	0	1	0	1
0	1	0	0	1	0	0
0	1	0	1	0	1	0
0	1	1	0	0	0	1
0	1	1	0	0	1	1
1	0	0	0	1	1	1
1	0	0	1	0	0	1

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

Expt. No.

Date

Page No. 20

when more than two numbers are added, the first two are added together and then their sum is added to the third number and so on.

Subtractions are performed in digital machines that use the 2's complement representation for negative numbers. When subtracting one binary number (the subtrahend) from another binary number (the minuend) following procedure is used,

Negate the subtrahend. This will change the subtrahend to its equivalent value of opposite sign. Add this to the minuend. The result of this addition will represent the difference between the subtrahend and the minuend.

$$\begin{array}{r}
 01001 \quad (+9) \\
 + 11100 \quad (-4) \\
 \hline
 100101 \quad (15)
 \end{array}
 \qquad
 \begin{array}{r}
 9 \rightarrow 01001 (9) \\
 -4 \rightarrow 00100 (4) \\
 \hline
 5
 \end{array}$$

Disregard, so the result is 00101 = +5

Procedure :

Design the circuit for 4-bit binary addition and subtraction using IC7483 and IC7486. For the given inputs (two 4-bit binary numbers) perform the operation and note down the output values in the table.

Teacher's Signature : _____

Expt. No.

Discussions :

The operations of addition and subtraction of signed numbers can be performed using only the addition operation if we use the 2's-complement form to represent negative numbers. Positive and negative numbers, including the sign bits, can be added together in the basic parallel-adder circuit when the negative numbers are in 2's complement form. When the 2's complement system is used, the number to be subtracted is changed to its 2's complement and then added to the minuend (the number the subtrahend is being subtracted from). Minuend is stored in the accumulator (A register), the subtrahend is then placed in the B-register and is changed to its 2's-complement form before it is added to the number in the A register. The sum outputs of the adder circuit now represent the difference between the minuend and the subtrahend. In addition the augend is stored in register A, to which, the addend is added which is stored in B-registered. At each step addition of three bits is performed, augend bit, addend bit, carry bit and gives sum bit and carry bit.

(Conclusions) Conclusion: Addition and subtraction of binary numbers was performed using parallel binary adder

Teacher's Signature : _____

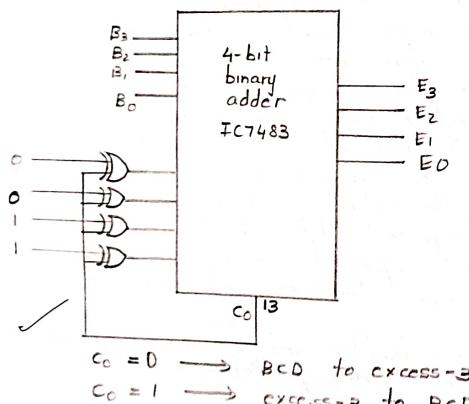
Observation and Results.

(a) BCD to excess-3

B_3	B_2	B_1	B_0	E_3	E_2	E_1	E_0
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	0	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	0	1
1	0	0	0	1	0	1	1
1	0	0	1	1	0	0	0

(b) excess 3 to BCD

E_3	E_2	E_1	E_0	B_3	B_2	B_1	B_0
0	0	1	1	0	0	0	0
0	1	0	0	0	0	0	1
0	1	0	1	0	0	1	0
0	1	1	0	0	0	1	1
0	1	1	1	0	1	1	1
1	0	0	0	1	0	0	0
1	0	0	1	1	0	0	1
1	0	1	0	1	0	1	0
1	0	1	1	1	0	1	1
1	1	0	0	1	1	0	0



Expt. No. G.....

Date 20/03/15

Page No. 22.....

Experiment No - 6

Aim: To design a circuit which converts BCD code to excess-3 code using

a) IC7483 chip

b) logic gates

Components: Digital IC trainer, IC7423, IC7404, IC7408, IC7432, IC7486

Theory:

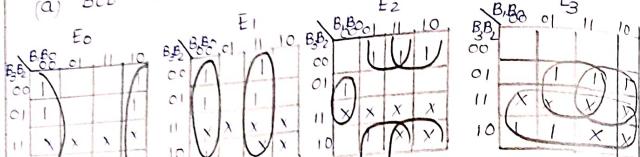
Excess 3 code is basically a binary code which is made by adding 3 with the equivalent decimal of a binary number and again converting it into binary number. So if a binary number is considered first we have to convert it into decimal number then add 3 with it and then convert it into binary and we will get the excess 3 equivalent of that number. The operation is done by following steps:

1. We have to convert the numbers into excess-3 forms by adding 0011 until each of the four bit groups.
2. Now two numbers are added using basic laws of binary addition.
3. Now which of the four groups have produced a carry we have to add 0011 with them and subtract 0011 from groups which have not produced carry during the addition.

Teacher's Signature : _____

(ii) Using logic gates

(a) BCD to excess-3



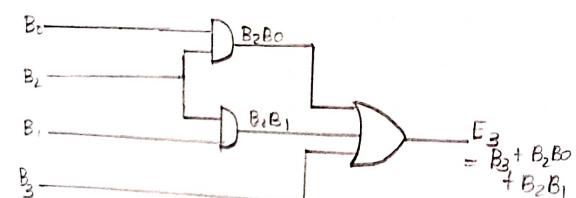
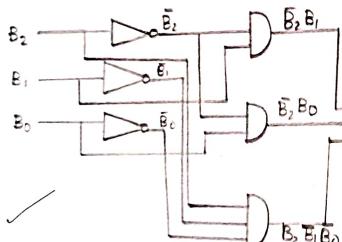
$$E_0 = \bar{B}_0$$

$$E_1 = \bar{B}_1 B_0 + \bar{B}_1 \bar{B}_0$$

$$E_2 = \bar{B}_2 B_0 + \bar{B}_2 B_1 + B_2 \bar{B}_0$$

$$E_3 = B_3 + B_2 B_0 + B_2 B_1$$

$$E_0 = B_0 \oplus \bar{B}_0$$



Expt. No.

Page No.

Excess-3 is converted to BCD by subtracting 3 (0011) from the Excess-3 code. This digital system can be designed with the help of 4-bit binary adder with the use of one toggle switch and some logic gates. If the toggle bit is 0, the code converter converts BCD to excess-3 and which 1 it converts excess-3 to BCD.

Procedure :

- (i) Draw the truth tables for BCD to excess-3 and excess-3 to BCD
- (ii) Using a 4-bit binary adder design the code converter.
- (iii) Given one input as 4-bit BCD while other input as binary equivalent of 3 (0011).
- (iv) Set the toggle bit such that it performs the required operation.
- (v) Draw the K-maps for excess-3 to BCD and BCD to excess-3.
- (vi) Derive the logical expressions and design the circuit

Teacher's Signature : _____

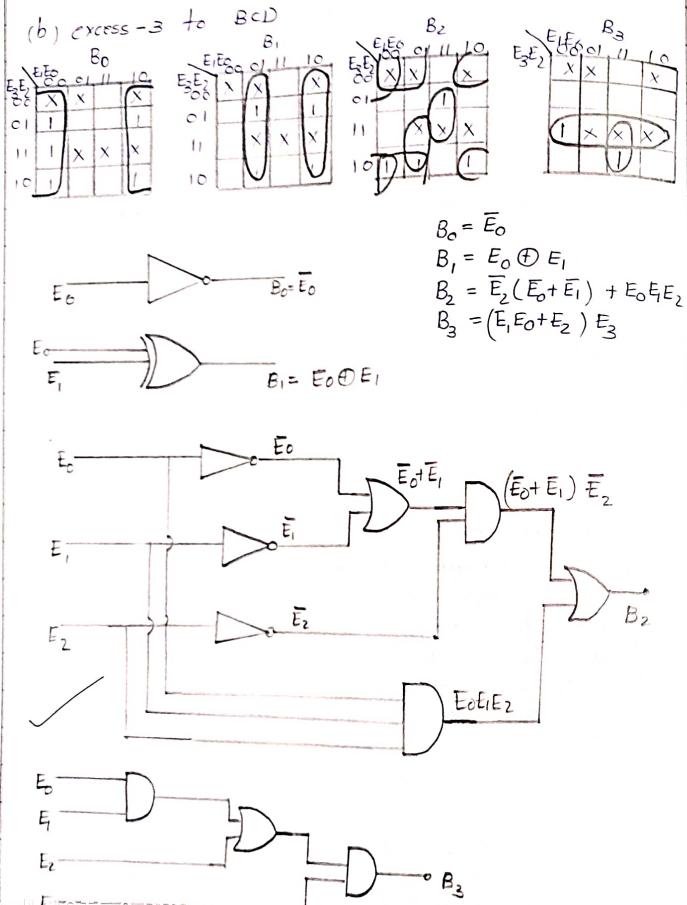
Expt. No.

Discussions:

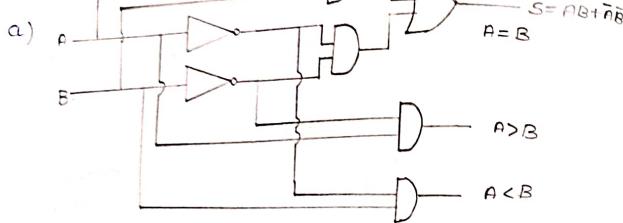
In this experiment we have designed a code converter which converts BCD to excess-3 code and also vice versa. BCD is converted to excess-3 by adding 3 to the decimal equivalent of the binary and again converting it back to binary. While converting excess-3 to BCD 3 is subtracted from its decimal equivalent. While designing the code converter with the help of 4-bit binary adder, we use toggle switch (1 bit). When the toggle bit is 0, the code converter converts BCD to excess-3 and when the bit is 1, excess-3 is converted to BCD. The code converter was also designed with the use of logical gates only. The Boolean expressions were derived from the K-maps and circuits were designed accordingly.

~~Conclusion~~
Conclusion: Circuits were designed which convert BCD to excess-3 code with the help of 4-bit binary adder and logic gates.

Teacher's Signature : _____

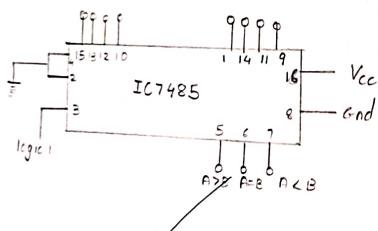


Observations and Results :



A	B	$A = B$	$A \oplus B$	$A < B$
0	0	1	0	0
0	1	0	0	1
1	0	0	1	0
1	1	1	0	0

b)



Expt. No. 7

Date 13/03/15

Page No. 25....

Experiment No - 7

Aim : To design a Circuit, which performs the following operations

- 1 - bit Comparator (using logic gates)
- 2 - bit Comparator (using IC 7485)
- A 7- segment LED to display numbers from 0-9 using IC 7447 and FND507
- 4 : 1 multiplexer using
 - IC 74153 (4 NAND gates)
- To set up 1 : 4 demux using IC 74139

Components : Digital IC trainer, logic gates, IC 7447, FND507, IC 74153, IC 74139

Theory :

(a) Magnitude Comparator takes two numbers as input in binary form and determines whether one number is greater than, less than, or equal to the number.

Consider two 4-bit binary numbers A and B

$$A = A_3 A_2 A_1 A_0 \quad B = B_3 B_2 B_1 B_0$$

$$S_1 = A_1 B_1 + \bar{A}_1 \bar{B}_1 \quad (\text{XNOR gate})$$

$$S = S_3 S_2 S_1 S_0$$

for equality, $S_3 S_2 S_1 S_0 = 1 = (A = B)$

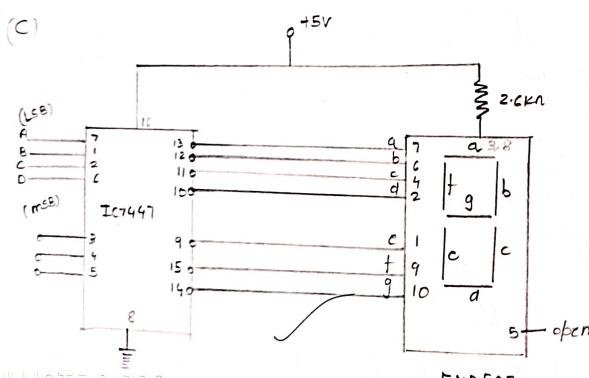
$$A_3 = B_3, A_2 = B_2, A_1 = B_1, A_0 = B_0$$

Teacher's Signature :

Date

Page No. 25

$A_1 A_0$	$B_1 B_0$	$A > B$	$A = B$	$A < B$
00	00	0	1	0
00	01	0	0	1
00	10	0	0	1
00	11	0	0	1
01	00	1	0	0
01	01	0	1	0
01	10	0	0	1
01	11	0	0	1
10	00	1	0	0
10	01	1	0	0
10	10	0	1	0
10	11	0	0	1
11	00	1	0	0
11	01	1	0	0
11	10	1	0	0
11	11	0	1	0



Expt. No.

$$A \neq B \rightarrow S = 0$$

$$(A > B) = A_3 \bar{B}_3 + S_3 A_2 \bar{B}_2 + S_3 S_2 A_1 \bar{B}_1 + S_3 S_2 S_1 A_0 \bar{B}_0$$

$$(A < B) = \bar{A}_3 B_3 + S_3 \bar{A}_2 B_2 + S_3 S_2 \bar{A}_1 B_1 + S_3 S_2 S_1 \bar{A}_0 B_0$$

for 1-bit A, B

$$S = AB + \bar{A} \bar{B}$$

$$S = 1, A = B$$

$$(A > B) = A \bar{B}$$

$$(A < B) = \bar{A} B$$

for 2-bit $A_1 A_0, B_1 B_0$

$$S_1 S_0 = 1 \quad A = B$$

$$(A > B) = A_1 \bar{B}_1 + S_1 A_0 \bar{B}_0$$

$$(A < B) = \bar{A}_1 B_1 + S_1 \bar{A}_0 B_0$$

(C) 7-segment LED display

A 7-segment display is a form of electronic display device for displaying decimal numerals.

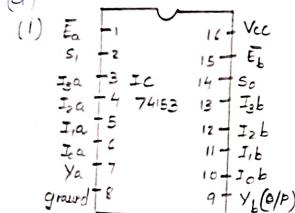
The seven segments are arranged as a rectangle of two vertical segments on each side with one horizontal segment on the top, middle and bottom. The seventh segment bisects the rectangle horizontally. The segments of a 7-segment display are referred to by letters A to G.

Teacher's Signature : _____

Expt. No.

S_3	S_2	S_1	S_0	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	0	0	0	0
2	0	0	1	0	1	0	1	0	0	1
3	0	0	1	1	1	1	1	0	0	1
4	0	1	0	0	1	0	1	0	0	1
5	0	1	0	1	0	1	1	0	1	1
6	0	1	1	0	1	0	1	1	1	1
7	0	1	1	1	1	0	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1
9	1	0	0	1	1	1	0	0	1	1

(d)



S_1	S_0	y
0	0	I ₀
0	1	I ₁
1	0	I ₂
1	1	I ₃

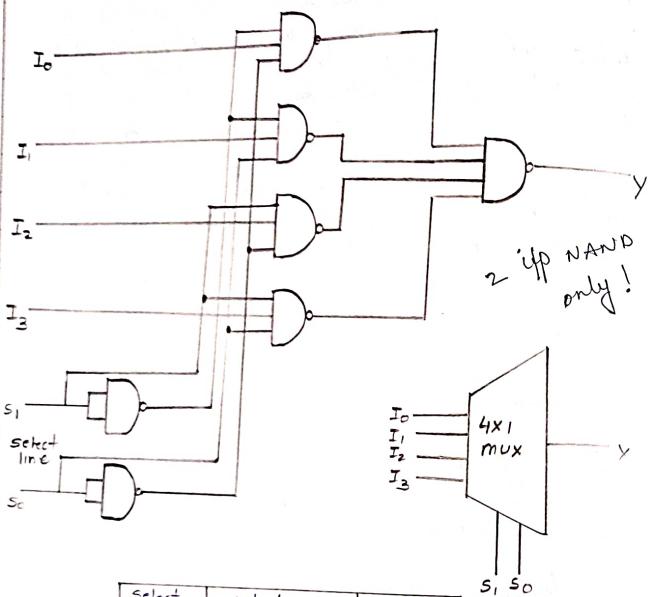
- (d) A multiplexer is a device that selects one of several input signals and forwards the selected input into a single line. A multiplexer at 2^n outputs has n select lines which are used to select which input line to send to the output.
- for 2×1 mux, there are two input lines and one selection line $y = I_0 \bar{S} + I_1 S$
- for 4×1 mux, there are four input lines and two select lines $y = I_0 \bar{S}_1 \bar{S}_0 + I_1 \bar{S}_1 S_0 + I_2 S_1 \bar{S}_0 + I_3 S_1 S_0$

- (e) A demultiplexer is a device taking a single input signal and selecting one of many data-output lines, which is connected to the single input. Demultiplexer can be considered as a single-input, multiple output switch.

A decoder converts n -inputs to 2^n outputs. Enable inputs must be on for the decoder to function, otherwise its outputs assume a single disabled output code word.

(ii) using NAND Gates

4x1 mux



Select	Inputs	Output
S ₁ S ₀	I ₃ I ₂ I ₁ I ₀	Y
0 0	X X X	
0 1	X X Y	
1 0	X X Y	
1 1	X X X	

Date

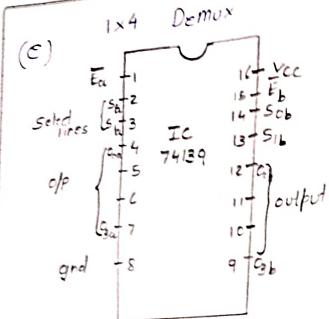
Expt. No.

Page No. 28

Procedure :

Derive the Boolean expressions for magnitude Comparator. Design the circuit with the use of logic gates for 1-bit comparator. Use IC 7495 to compare 2-bit numbers. Design LED display using 7C7447 and FND1507. Design mux and Demux using IC 74153 and IC 74139 respectively.

Teacher's Signature : _____



E	F	G	D _c	D _i	D _o	B ₃
S ₁	S ₂					
0	0	0	1	0	0	0
0	0	1	0	1	0	0
0	1	0	0	0	1	0
0	1	1	0	0	0	1

Date _____

Page No. 27

Expt. No. _____

Discussions:

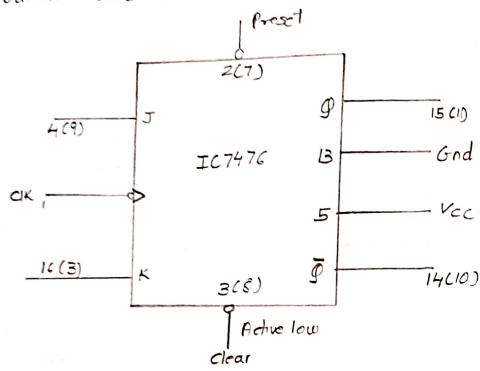
Magnitude Comparator is used to compare the magnitude of any two given numbers. It first checks the msd and thus follows checking the digits till lsd if they are found equal it falls. If the number is equal to, greater than, or less than the given number. It can be designed using logic gates or with the use of IC 7485. LED display is used to display decimal numbers from 0 to 9. LED display's are used in digital clocks, watches, and other measuring instruments. Multiplexers are used to select one of several inputs and forwards the single selected input into a single line. A multiplexer of 2^n outputs has n select lines. A demultiplexer is a device that has single input and selects one of many output lines which is connected to single input.

~~checked
May 2015~~

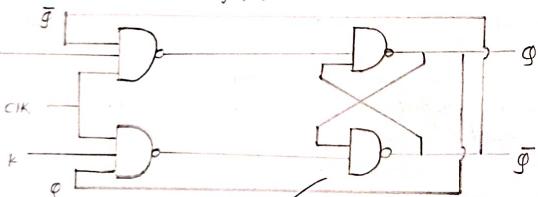
Conclusion: Circuits were designed for each of the operations and the result was analysed

Teacher's Signature: _____

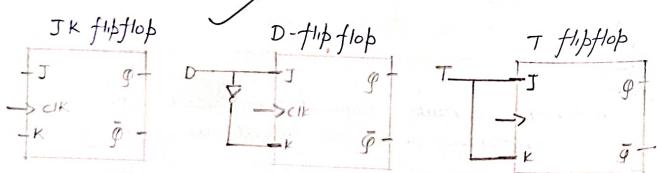
Observation and Results



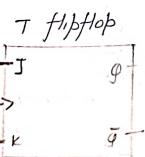
JK flip flop



JK flip flop



D-flip flop



T flip flop

Expt. No.....8.....

Date 19/03/15

Page No....30....

Experiment No - 8

Aim : To design the following flip-flops using IC 7476

- JK flip flop
- D flip flop
- T flip flop

Components : Digital IC trainer , IC 7476 , NOT gate

Theory :

A flip-flop is a circuit that has two stable states and can be used to store state information. The circuit can be made to change state by signals applied to one or more control inputs and will have one or two outputs. It is the basic storage element in the sequential logic. Flip-flops are a fundamental building block of digital electronics systems used in computers, communications and many other types of systems.

A flip-flop stores a single bit of data; one of its two states represents a "one" and the other represents a "zero". Such data storage can be used for storage of state and such

Teacher's Signature :

JK flip-flop characteristic table

CIR	J	K	Q_n	Q_{n+1}
0	X	X	0	0
0	X	X	1	1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

Excitation table

Q_n	Q_{n+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

$$Q_{n+1} = J\bar{Q}_n + \bar{K}Q_n$$

NOTE & PRO

Date

Page No....31

Expt. No.....

a circuit is described as sequential logic.

flip-flops are divided as SR ("set-reset"), JK, D ("data" or "delay"), T ("toggle")

The JK flip-flop analyzes the behaviour of the SR flip-flop ($J = \text{set}$, $K = \text{Reset}$) by interpreting the $S=R=1$ condition as a flip or toggle command. Specifically, the combination $J=1, K=0$ is a command to Set the flip-flop; the combination $J=0, K=1$ is a command to toggle, and the combination $J=K=1$ is a command to toggle the flip-flop; it changes its output to the logical complement of its current value. Setting $J=K=0$ will hold the current state.

To synthesize a T flip-flop set K equal to J and for D flip-flop set K equal to the complement of J. If the T input is high, the T flip-flop changes state ("toggles") when the clock input is strobed. If the T input is low, the flip-flop holds the previous value. The D flip-flop captures the value of the D input at a definite portion of the clock cycle (such as rising edge of the clock). That captured value becomes the Q output. At other times, the output Q does not change. The D flip-flop can be viewed as a memory cell a zero order hold or a delay time.

Teacher's Signature : _____

D flip flop

characteristic table

D	q_n	q_{n+1}
0	0	0
0	1	0
1	0	1
1	1	1

Excitation table

q_n	q_{n+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

T flip flop

characteristic table

T	q_n	q_{n+1}
0	0	0
0	1	1
1	0	1
1	1	0

Excitation table

q_n	q_{n+1}	T
0	0	0
0	1	1
1	0	1
1	1	0

MINI NOTE & PRO
ES

Expt. No.

Date

Page No. 32

Procedure:

Write the truth table for the given flip flop. From the truth table derive the characteristic table. Using K-map derive the Boolean expression for q_{n+1} as a function of inputs and the present state q_n . Thus write the simplest form and find the flip flop value as a function of q_n and T by drawing the excitation table. Using the given IC 7476 design the circuit for each flip flop.

Teacher's Signature : _____

Expt. No.

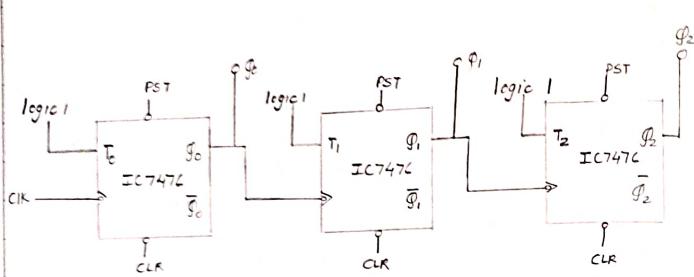
Discussions:

flip flops are a fundamental building block of digital electronics systems used in many electronic devices. A flip flop stores a single bit (binary digit) of data and the two states are '1' and '0'. Such data storage can be used for storage of state and such a circuit is described as sequential logic. Flip flops can be either simple or clocked (Synchronous or edge-triggered). A flip-flop's output only changes on a single type (positive going or negative going) of clock edge. Flip flops can be divided into common types: the SR ("set-reset"), D ("data" or "delay"), T ("toggle") and JK types. Clocked devices are specially designed for synchronous systems, such devices ignores their inputs except at the transition of a dedicated clock signal. Clocking causes the flip flop to change or retain its output signal based upon the values of the input signals at the transition.

Conclusion: Circuits were designed for JK flip flop, T and D flip flop and the result was analyzed.

Teacher's Signature : _____

Observations & Results :



Present state	Next state
$Q_2 \ Q_1 \ Q_0$	$Q'_2 \ Q'_1 \ Q'_0$
0 0 0	0 0 1
0 0 1	0 1 0
0 1 0	0 1 1
0 1 1	1 0 0
1 0 0	1 0 1
1 0 1	1 1 0
1 1 0	1 1 1
1 1 1	0 0 0

Excitation table
for T flip-flop

\emptyset	Q_{H1}	T
0	0	0
0	1	1
1	0	1
1	1	0



Expt. No. 9

Date 27/03/15

Page No. ... 34

Experiment No - 9

Aim : (i) To design a 3-bit ripple up counter
(ii) To design Counter to count following sequence (01, 10, 11)

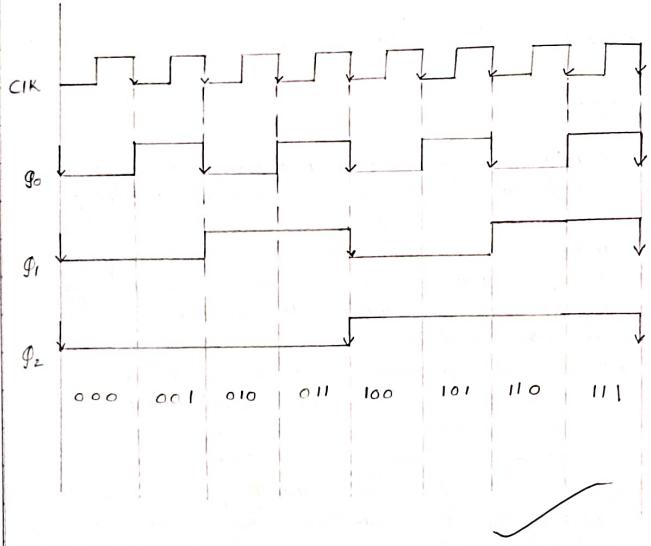
Components : Digital IC trainer, IC 7476

Theory:

A counter is essentially a register that goes through a predetermined sequence of binary states. The gates in the counter are connected in such a way as to produce the prescribed sequence of states. Although counters are a special type of register, it is common to differentiate them. Counters are available in two categories : ripple counters and synchronous counters. In a ripple counter, a flip flop output transition serves as a source for triggering other flip flops. The C inputs of some or all flip-flops are triggered, not by the common clock pulses, but rather by the transition that occurs in other flip flop outputs. In a synchronous counter, the C inputs of all flip-flops receive the common clock. Synchronous counters are faster compared to the asynchronous counters and can count any sequence.

Teacher's Signature : _____

Negative edge Triggering



Expt. No.

Date

Page No. 35

A binary ripple counter consists of a series connection of complementing flip-flops, with the output of each flip-flop connected to the *c*-input of the next higher order flip-flop. The flip-flop holding the least significant bit receives the incoming count pulses. A complementing flip-flop can be obtained from a *J-K* flip-flop with the *J* and *K* inputs tied together or from a *T*-flip-flop. A decimal counter follows a sequence of 10 states and returns to 0 after the count of 9. Such a counter must at least have four flip-flops to represent each decimal digit, since a decimal digit is represented by a binary code with at least 4 bits.

Procedure :

Draw the circuit for each of the counters and analyse the theoretical output. Design the circuit as a combination of flip-flops with the use of IC 7476 and clock pulse. Set the inputs as required. Note the output for each change in clock pulse. Analyse the practical result and compare it to the theoretical input.

Teacher's Signature : _____

Date

Expt. No.

Page No. 36

Discussions :

A register that goes through a prescribed sequence of states upon the application of input pulses is called a counter. The input pulses may be clock pulses or they may originate from some external source and may occur at fixed interval of time or random. The sequence of states may follow the binary number sequence or any other sequence of states. A counter that follows the binary number sequence is called a binary counter. An n -bit binary counter consists of n flip flops and can count in binary from 0 through $2^n - 1$. Counters are divided as synchronous and asynchronous counters. Synchronous counters are faster and have same clock to all the flip flops unlike asynchronous counters which have different clock pulses to different flip flops. Ripple counter is an asynchronous counter where all flip flops operate in toggle mode. Ring counter is a synchronous counter. Counters are used in timers, frequency dividers and Radars.

Conclusion: 3-bit ripple up counter was designed and the result was analysed.

Checked

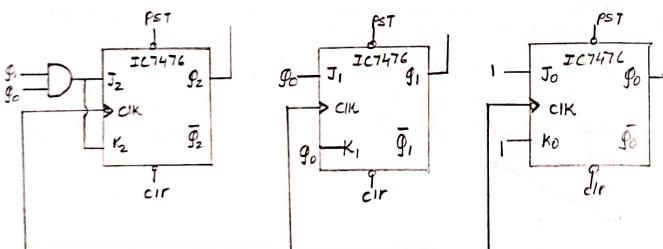
Smt. 10/01/15

Teacher's Signature : _____

Observations and Results:

(i)

Present state			Next state			Excitation table		
J_2	J_1	J_0	\bar{J}_2	\bar{J}_1	\bar{J}_0	K_2	K_1	K_0
0 0 0	0 0 1		0 X	0 X	1 X			
0 0 1	0 1 0		0 X	1 X	X 1			
0 1 0	0 1 1		0 X	X 0	1 X			
0 1 1	1 0 0		1 X	X 1	X 1			
1 0 0	1 0 1		X 0	0 X	1 X			
1 0 1	1 1 0		X 0	1 X	X 1			
1 1 0	1 1 1		X 0	X 0	1 X			
1 1 1	0 0 0		X 1	X 1	X 1			



Expt. No. 10

Date 10/04/2015

Page No. 31

Experiment No 10

Aim : (1) To realize a 3-bit synchronous counter

(ii) To Design and realize a synchronous Counter to count the given Sequence
0-2-3-C-5-1-0 .

Components: Digital IC trainer, IC 7476, IC 7402.

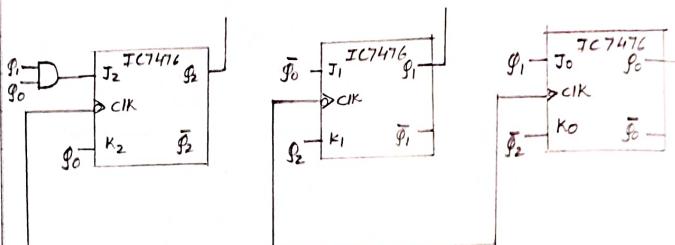
Theory:

In synchronous counters same clock pulse is applied to the the inputs of all flip flops. A common clock triggers all flip flops simultaneously rather than one at a time in succession as in a ripple counter. The decision whether a flip flop is to be complemented is determined from the values of the data inputs such as T or J and K at the time of the clock edge. If $T=0$ or $J=K=0$, the flip flops stays at the change state. If $T=1$ or $J=K=1$ the flip flops complements. In a synchronous binary counter, the flip flop in the least significant position is complemented with every pulse. A flip flop in any other position is complemented when all the bits in the lower significant positions are equal to 1.

Teacher's Signature: _____

(i)

Present state			Next state			Excitation table					
q_2	q_1	q_0	q_2^+	q_1^+	q_0^+	J_2	K_2	J_1	K_1	J_0	K_0
0	0	0	0	1	0	0	X	1	X	0	X
0	0	1	0	0	0	0	X	0	X	X	1
0	1	0	0	1	1	0	X	X	0	1	X
0	1	1	1	1	0	1	X	X	0	X	1
1	0	0	X	X	X	X	X	X	X	XX	
1	0	1	0	0	1	X	1	0	X	XO	
1	1	0	1	0	1	X	0	X	1	1	X
1	1	1	X	X	X	X	X	X	X	XX	



Expt. No. Date
Page No.

(i) K-map

q_2	q_1	q_0	J_2	K_2	J_1	K_1	J_0	K_0
0	0	0	0	1	1	1	0	1
1	X	X	1	X	X	X	1	X

$J_2 = q_1 q_0$ $J_1 = \bar{q}_0$ $J_0 = 1$

q_2	q_1	q_0	J_2	K_2	J_1	K_1	J_0	K_0
0	X	X	1	X	X	X	1	X
1	1	X	1	X	X	1	X	X

$K_2 = \bar{q}_1 q_0$ $K_1 = q_0$ $K_0 = 1$

q_2	q_1	q_0	J_2	K_2	J_1	K_1	J_0	K_0
0	1	1	1	X	X	X	1	X
1	X	X	1	X	X	X	1	X

$J_2 = q_1 q_0$ $J_1 = \bar{q}_0$ $J_0 = q_1$

q_2	q_1	q_0	J_2	K_2	J_1	K_1	J_0	K_0
0	X	X	1	X	X	X	1	X
1	X	1	X	X	X	X	1	X

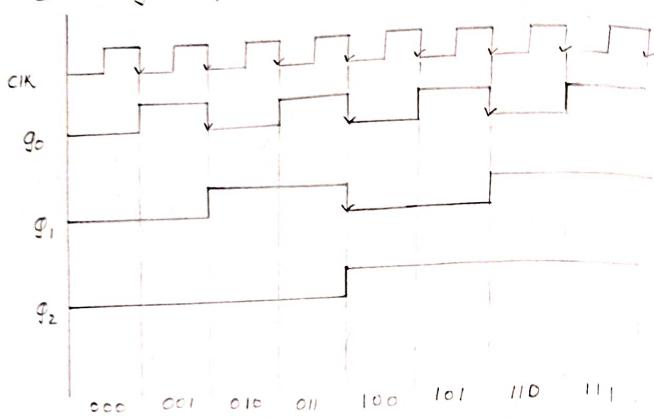
$K_2 = \bar{q}_0$ $K_1 = q_2$ $K_0 = \bar{q}_2$

Procedure:

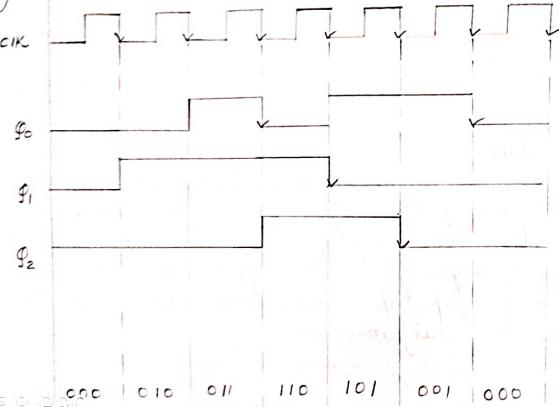
As per the given question Draw the characteristic table for present state and Next state.
Thus draw the excitation table for JK flip flop.
Using K-maps find the expressions for inputs of each flip flop. Thus design the circuit and draw the waveforms.

Teacher's Signature : _____

(i) Timing waveform



(ii)



Expt. No.

Date

Page No.

Discussions:

In synchronous counters, all flip flops are triggered simultaneously, rather than one at a time in succession. In synchronous binary Counter, the flip flop in the least significant position is complemented with every pulse. A flip flop in any other position is complemented when all the bits in the lower significant positions are equal to 1. For example, if the present state of a four bit counter $Q_3 Q_2 Q_1 Q_0 = 0011$, the next count is 0100. Q_0 is always complemented. Q_1 is complemented because the present state of $Q_0 = 1$. Q_2 is complemented because the present state of $Q_0 = 1$. However, Q_3 is not complemented because present state of $Q_3 Q_2 Q_1 Q_0 = 0011$, which does not give an all - 1's condition.

Conclusion: 3-bit synchronous counters were designed for binary count of different sequences and the result was analysed.

Teacher's Signature: A. K. S. 13/4/115