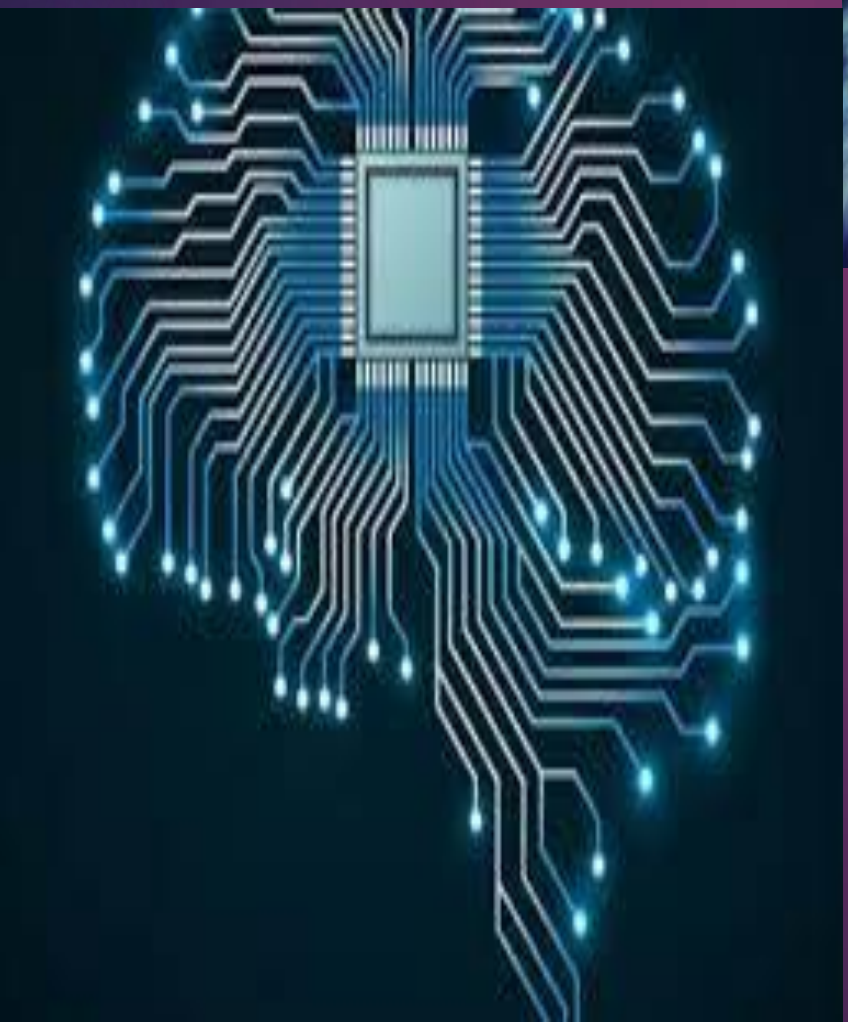
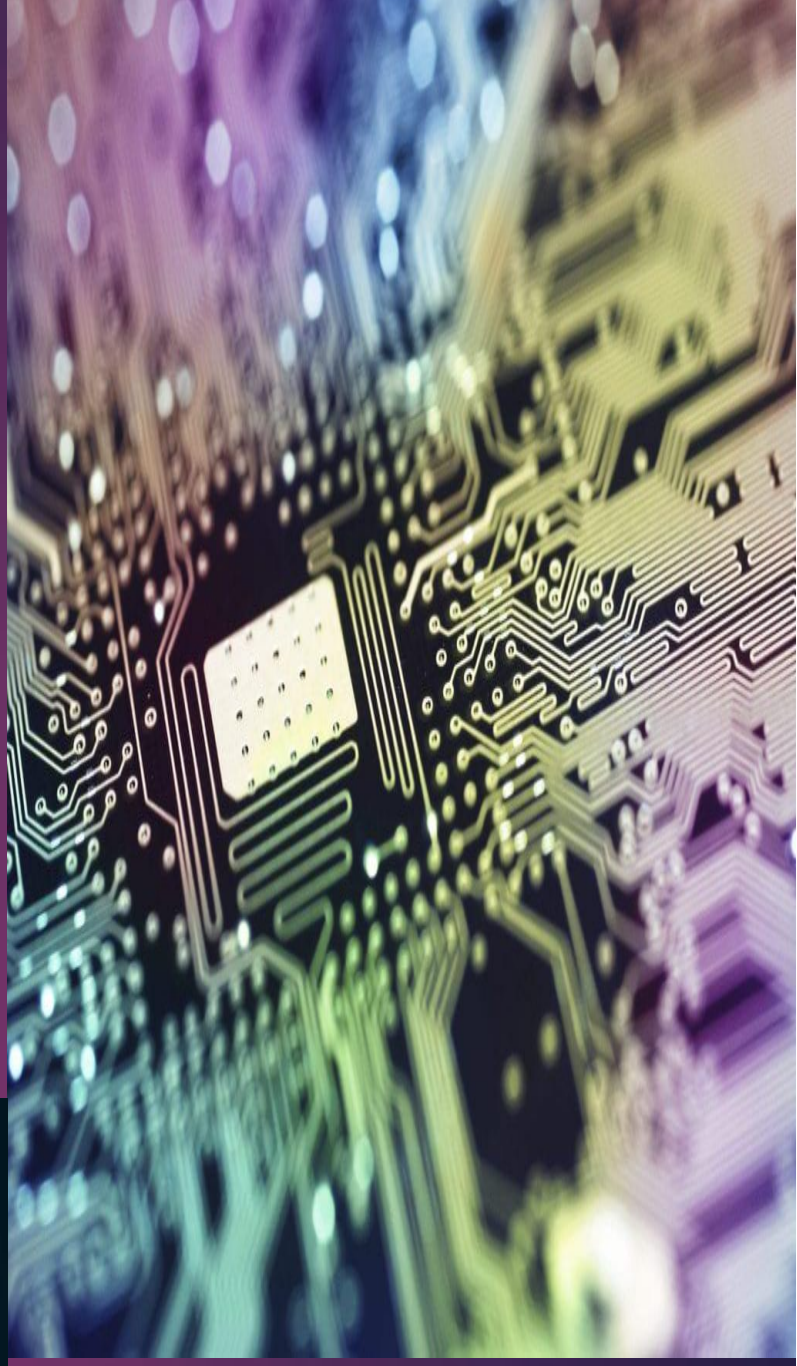
DIGITAL SYSTEMS DESIGN LAB MANUAL

REGULATION 2021





ISBN 978-93-6128-607-0

SATHYA V

DIGITAL SYSTEMS LAB MANUAL

ELECTRONICS AND COMMUNICATION ENGINEERING

REGULATION – 2021

AUTHOR:
Mrs. V. SATHYA.,M.E.,

DEPARTMENTOFELECTRONICSANDCOMMUNICATION ENGINEERING

LABORATORYMANUAL

DIGITALSYSTEMDESIGNLABORATORY

DSDLABMANUAL

III SEMESTERELECTRONICSANDCOMMUNICATION

SUBJECT CODE:EC3352

REGULATION-2021

Mrs. V. SATHYA.,M.E.,
Assistant Professor/ Electronics and Communication Engineering
Annai Mira College of Engineering and Technology
Ranipet – 632 517

PREFACE

This book on "**DIGITAL SYSTEMS LABORATORY MANUAL (Electronics and communication Engineering)**" covers the complete syllabus prescribed by the Anna University, Chennai for the fourth semester**B.E/B.Tech**.Degree course under **Outcome Based Education Credit System with the new regulation 2021**.

This book covers DeMorgan's Theorem, Full/Parallel Adders, Subtractors andMagnitude Comparator, Multiplexer using logic gates, Demultiplexers and Decoders, Flip-Flops, Shift registers and Counters We hope that this book will be useful to both teachers and students. Finally, we would request the readers to kindly send their valuable comments and suggestions towards the improvement of the manual and the same will be gratefully acknowledge.

Any suggestion from the reader for the betterment of this book can be dropped into sathyajayajothi92@gmail.com.

Mrs.V. SATHYA., M.E.,

ACKNOWLEDGEMENT

We are thankful to and fortunate enough to get constant encouragement, support and guideline from Chairman**Thiru.S.Ramadoss**, Secretary & Treasurer **Mr. G. Thamotharan** for his blessings to complete the book successfully.

We would not forget to remember our Principal **Dr.T.K.Gopinathan**and Vice Principal **Dr. D. Saravanan**for their constant assistance in preparing this book.

ANNAIMIRA

COLLEGEOFENGINEERINGANDTECHNOLOGY

ARAPAKKAM,RANIPET–632517.

DEPARTMENTOF

ELECTRONICS ANDCOMMUNICATIONENGINEERING

SUBJECTCODE/NAME:EC3352–DIGITALSYSTEMDESIGN

ASPERANNAUNIVERSITY,CHENNAIREGULATION2021

LISTOFEXPERIMENTS

1.VerificationofBooleanTheoremsusing basicgates.

2.Designandimplementationofcombinationalcircuitsusingbasicgatesforarbitraryfunctions,code converters.

3.DesignandimplementHalf/FullAdderandSubtractor.

4.Design and implementcombinationalcircuitsusingMSIdevices:4–bitbinaryadder/subtractor
Parity generator / checkerMagnitude ComparatorApplicationusing multiplexers

5.Designandimplementshift-registers.

6.Designandimplementsynchronouscounters.

7.Designandimplementasynchronouscounters.

INDEX

Ex.No.	Date	Title	Marks	Staff Sign.
1a		STUDY OFLOGICGATES		
1b		VERIFICATIONOFBOOLEAN THEOREMS USING DIGITALLOGIC GATES		
2		CODECONVERTOR		
3a		ADDERANDSUBTRACTOR		
4a		4-BITADDERANDSUBTRACTOR		
4b		PARITYGENERATOR& CHECKER		
4c		MAGNITUDECOMPARATOR		
4d		MULTIPLEXER ANDDEMULTIPLEX ER		
5		SHIFTREGISTER		
6		SYNCHRONOUSANDASYNC HRONOUSCOUNTER		

Ex.No.-1a STUDYOF LOGICGATES

AIM:

To study about logic gates and verify their truth tables

APPARATUSREQUIRED:

SL.NO.	COMPONENT	SPECIFICATION	QTY
1.	ANDGATE	IC7408	1
2.	ORGATE	IC7432	1
3.	NOT GATE	IC7404	1
4.	NANDGATE2I/P	IC7400	1
5.	NORGATE	IC7402	1
6.	X-ORGATE	IC7486	1
7.	NANDGATE3I/P	IC7410	1
8.	ICTRAINERKIT	-	1
9.	PATCHCORD	-	14

THEORY:

Circuitthattakesthelogicaldecisionandtheprocessarecalledlogicgates. Eachgatehasone ormoreinputand onlyone output.

OR, AND and NOT are basic gates. NAND, NOR and X-OR are known asuniversalgates.Basicgatesformthesegates.

ANDGATE:

The AND gate performs a logical multiplication commonly known as ANDfunction. The output is high when both the inputs are high. The output is low levelwhenanyoneoftheinputsislow.

ORGATE:

The OR gate performs a logical addition commonly known as OR function.The output is high when any one of the inputs is high. The output is low level whenboththeinputs arelow.

NOTGATE:

TheNOTgateiscalledaninverter.Theoutputishigh whentheinputislow. Theoutputislowwhentheinputishigh.

ANDGATE:

The NAND gate is a contraction of AND-NOT. The output is high when bothinputsarelowandanyoneoftheinputislow.Theoutputislowlevelwhenbothinputsare high.

NORGATE:

The NOR gate is a contraction of OR-NOT. The output is high when bothinputsarelow.Theoutputislowwhenoneorbothinputsarehigh.

X-ORGATE:

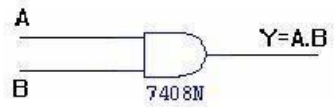
The outputishigh when any oneof theinputsishigh.The outputislowwhenboththeinputsarelow andboththeinputs arehigh.

PROCEDURE:

(i) Connectionsaregivenaspercircuit diagram.
(ii) Logicalinputsaregivenaspercircuitdiagram.
(iii) Observetheoutput and verifythetruthtable.

ANDGATE

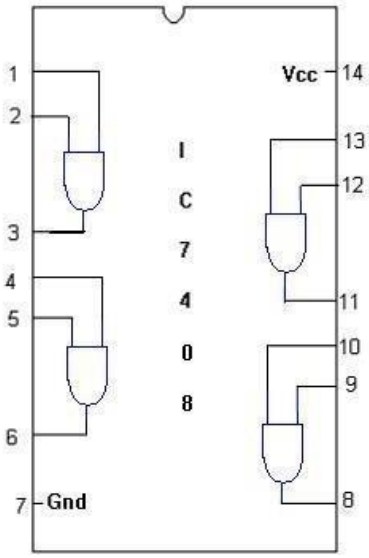
SYMBOL



TRUTH TABLE

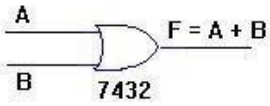
A	B	A.B
0	0	0
0	1	0
1	0	0
1	1	1

PINDIAGRAM



ORGATE

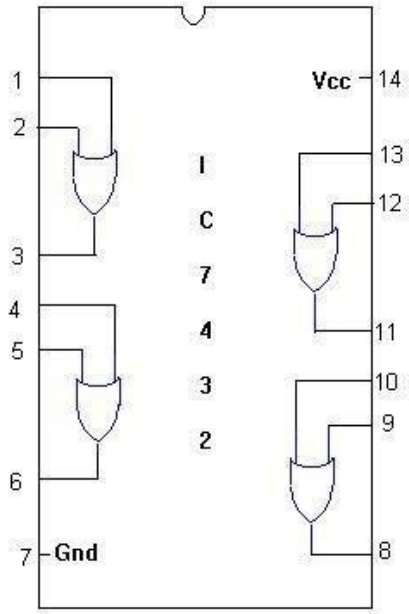
SYMBOL :



TRUTH TABLE

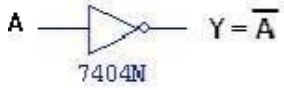
A	B	A+B
0	0	0
0	1	1
1	0	1
1	1	1

PIN DIAGRAM :

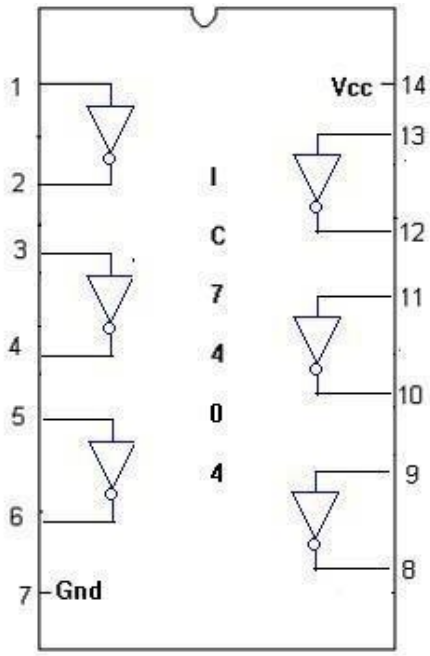


NOTGATE

SYMBOL



PINDIAGRAM

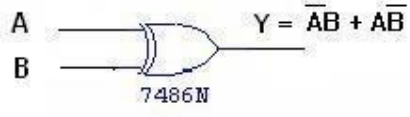


TRUTH TABLE :

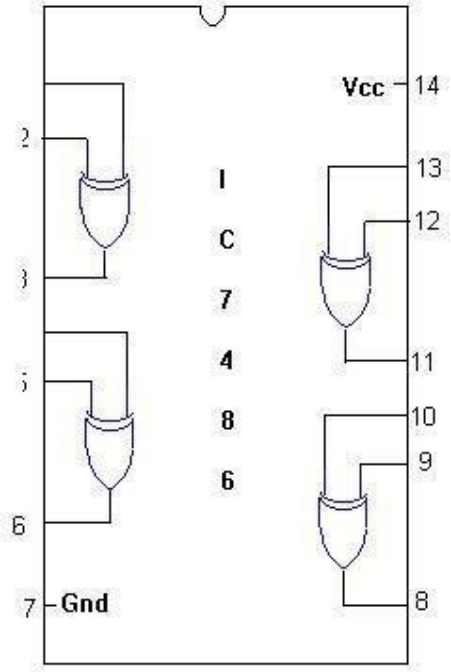
A	$\overline{A}$
0	1
1	0

EX-ORGATE

SYMBOL



PINDIAGRAM

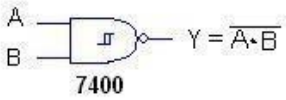


TRUTH TABLE :

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

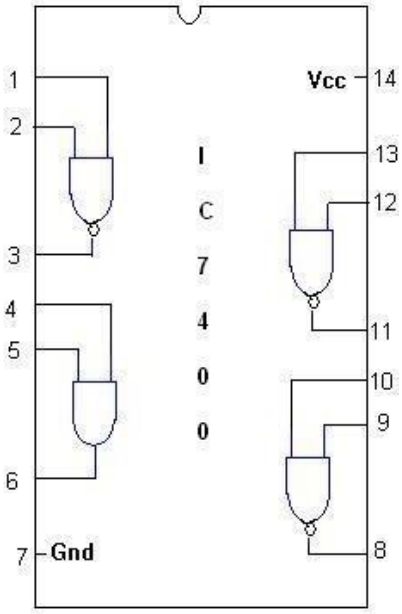
2-INPUTNANDGATE

SYMBOL



7400

PINDIAGRAM

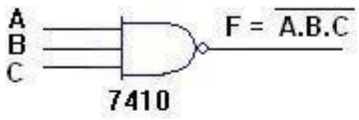


TRUTH TABLE

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

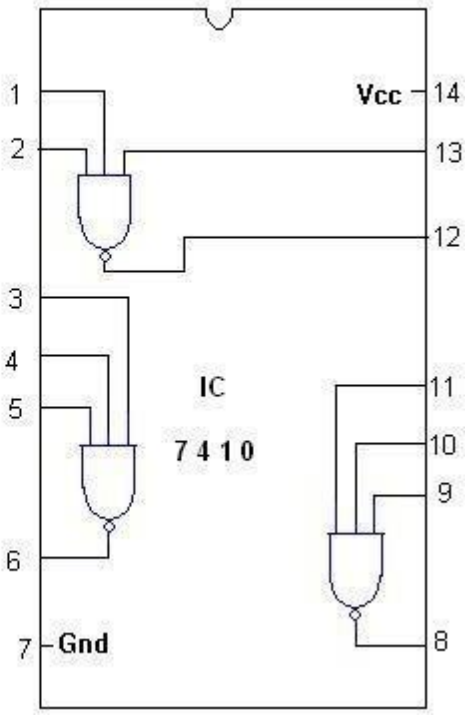
3-INPUTNANDGATE

SYMBOL :



7410

PIN DIAGRAM :

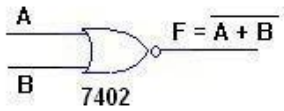


TRUTH TABLE

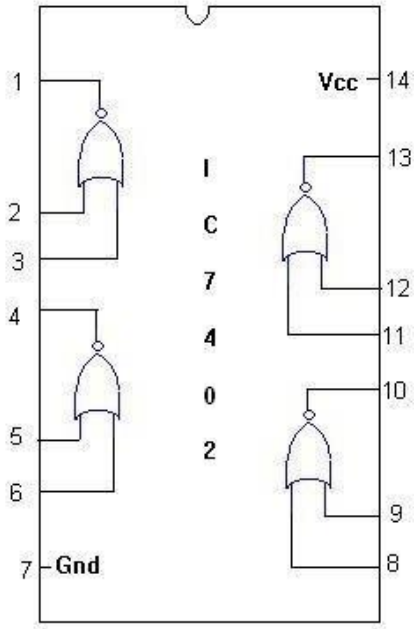
A	B	C	$\overline{A.B.C}$
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

NORGATE

SYMBOL :



PIN DIAGRAM :



TRUTH TABLE

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

RESULT:

The logic gates are studied and its truth tables are verified.

Ex.No.-1b VERIFICATION OF BOOLEANTHEOREMSUSINGDIGITALLOGICGATES

AIM:

To verify the Boolean Theorems using logic gates.

APPARATUSREQUIRED:

SL.NO.	COMPONENT	SPECIFICATION	QTY.
1.	ANDGATE	IC 7408	1
2.	ORGATE	IC 7432	1
3.	NOT GATE	IC 7404	1
4.	ICTRAINERKIT	-	1
5.	CONNECTINGWIRES	-	As perrequired

THEORY:

BASICBOOLEANLAWS

1. CommutativeLaw

ThebinaryoperatorOR,ANDissaidtobecommutativeif,

1. A+B=B+A
2. A.B=B.A

2. AssociativeLaw

The binary operator OR, AND is said to be associative if,1.A+(B+C)= (A+B)+C

2.A.(B.C)=(A.B).C

3. DistributiveLaw

ThebinaryoperatorOR,ANDissaidtobedistributiveif,1.A+(B.C)=(A+B).(A+C)

2.A.(B+C) =(A.B)+(A.C)

4. AbsorptionLaw

1. A+AB= A
2. A+AB=A+B

5. Involution(or)DoublecomplementLaw

1. A=A

6. IdempotentLaw

1. A+A= A
2. A.A=A

7. ComplementaryLaw

1.A+A'= 1

2.A.A'=0

8. DeMorgan'sTheorem

1. Thecomplementofthesumisequaltothesumoftheproductoftheindividualcomplements.

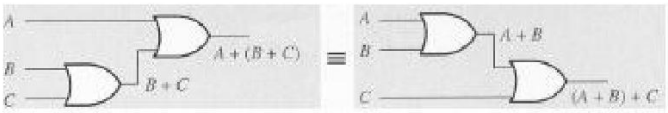
A+B=A.B

2. Thecomplementofthe productisequaltothe sumoftheindividualcomplements.

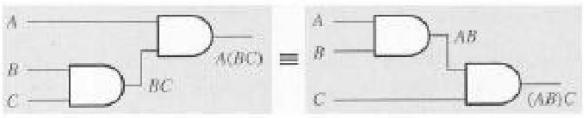
A.B=A+B

Associative Laws of Boolean Algebra

$$A + (B + C) = (A + B) + C$$



$$A \bullet (B \bullet C) = (A \bullet B) \bullet C$$



Proof of the Associative Property for the OR operation: (A+B)+C = A+(B+C)

A	B	C	(A+B)	(B+C)	A+(B+C)	(A+B)+C
0	0	0	0	0	0	0
0	0	1	0	1	1	1
0	1	0	1	1	1	1
0	1	1	1	1	1	1
1	0	0	1	0	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	1	1	1

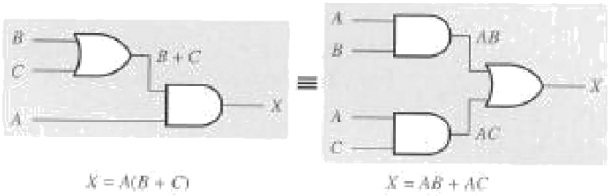
Proof of the Associative Property for the AND operation: (A·B)·C = A·(B·C)

A	B	C	(A·B)	(B·C)	A· (B·C)	(A·B)·C
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	0	0
1	0	0	0	0	0	0
1	0	1	0	0	0	0
1	1	0	1	0	0	0
1	1	1	1	1	1	1

Distributive Laws of Boolean Algebra

$$A \bullet (B + C) = A \bullet B + A \bullet C$$

$$A\,(B + C) = A\,B + A\,C$$



Proof of Distributive Rule

A	B	C	A·B	A·C	(A·B)+ (A·C)	(B+C)	A·(B+C)
0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0
0	1	0	0	0	0	1	0
0	1	1	0	0	0	1	0
1	0	0	0	0	0	0	0
1	0	1	0	1	1	1	1
1	1	0	1	0	1	1	1
1	1	1	1	1	1	1	1

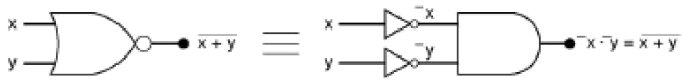
Proof of Distributive Rule

A	B	C	A+B	A+C	(A+B)· (A+C)	(B·C)	A+(B·C)
0	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0
0	1	0	1	0	0	0	0
0	1	1	1	1	1	1	1
1	0	0	1	1	1	0	1
1	0	1	1	1	1	0	1
1	1	0	1	0	1	0	1
1	1	1	1	1	1	1	1

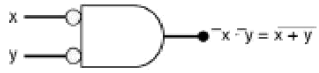
Demorgan'sTheorem

a) Proofofequation(1):

Construct the two circuits corresponding to the functions A'. B'and (A+B)'respectively. Show that for all combinations of A and B, the two circuits give identicalresults.Connectthesecircuitsandverifytheiroperations.



(a)



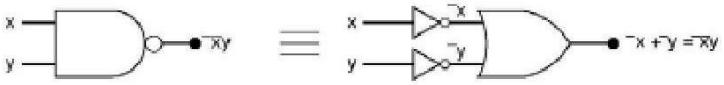
(b)

Proof (via Truth Table) of DeMorgan's Theorem $\overline{A \cdot B} = \overline{A} + \overline{B}$

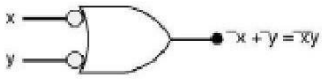
A	B	A·B	$\overline{A \cdot B}$	$\overline{A}$	$\overline{B}$	$\overline{A} + \overline{B}$
0	0	0	1	1	1	1
0	1	0	1	1	0	1
1	0	0	1	0	1	1
1	1	1	0	0	0	0

b) Proofofequation(2)

Construct two circuits corresponding to the functions A'+B'and (A.B)' A.B,respectively.Showthat,forallcombinationsofAandB,thetwocircuitsgiveidenticalresults .Connectthesecircuitsandverifytheiroperations.



(a)



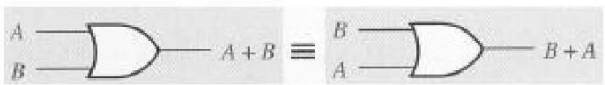
(b)

Proof (via Truth Table) of DeMorgan's Theorem $\overline{A + B} = \overline{A} \cdot \overline{B}$

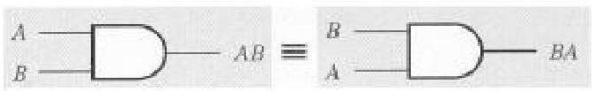
A	B	A+B	$\overline{A + B}$	$\overline{A}$	$\overline{B}$	$\overline{A} \cdot \overline{B}$
0	0	0	1	1	1	1
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	0

Commutative Laws of Boolean Algebra

A + B = B + A



A • B = B • A



Wewillalso usethefollowingsetofpostulates:

P1: BooleanalgebraisclosedundertheAND,OR, andNOToperations.

P2: Theidentityelementwithrespectto•isoneand+iszero.Thereis noidentityelementwithrespecttologicalNOT.

P3: The•and+operatorsarecommutative.

P4:•and+aredistributive withrespecttooneanother.That is,

A•(B+C)=(A•B)+(A•C)andA+ (B•C)=(A+ B)•(A+ C).

P5: ForeveryvalueAthereexistsavalueA'such that A•A'=0andA+A'=1. ThisvalueisthelogicalcomplementorNOT)ofA.

P6:•and+arebothassociative.Thatis,(A•B)•C=A•(B•C) and(A+B)+C=A+(B+C). Youcanprove allothertheoremsinbooleanalgebrausingthese postulates.

PROCEDURE:

1. ObtaintherequiredICalongwiththeDigitaltrainerkit.
2. ConnectzerovoltstoGND pinand +5voltstoVcc .
3. Applytheinputstotherespectiveinputpins.
4. Verifythe outputwiththetruthtable.

RESULT:

Thus, the above Boolean laws are verified.

Ex.No.-2 CODECONVERTOR

AIM:

To design and implement 4-bit

(i) Binary to gray code converter

(ii) Gray to Binary code converter

(iii) BCD to Excess-3 code converter

(iv) Excess-3 to BCD code converter

APPARATUSREQUIRED:

SL.NO.	COMPONENT	SPECIFICATION	QTY.
1.	X-ORGATE	IC 7486	1
2.	ANDGATE	IC 7408	1
3.	ORGATE	IC 7432	1
4.	NOT GATE	IC 7404	1
5.	ICTRAINERKIT	-	1
6.	PATCHCORDS	-	35

THEORY:

Theavailabilityoflargevarietyofcodesforthesamediscreteelementsofinformation results in the use of different codes by different systems. A conversion circuitmustbeinsertedbetweenthetwosystemsifeachusesdifferentcodesforsameinformation.Thus, code converter is a circuit that makes the two systems compatible even though eachusesdifferentbinarycode.

The bitcombination assigned to binary code to gray code. Since each code usesfour bits to represent a decimal digit. There are four inputs and four outputs. Gray code is anon-weightedcode.

The input variable are designated as B3, B2, B1, B0 and the output variables aredesignated as C3, C2, C1, Co. from the truth table, combinational circuit is designed. TheBooleanfunctionsareobtainedfromK-Mapforeachoutputvariable.

A code converter is a circuit that makes the two systems compatible even thougheach uses a different binary code. To convert from binary code to Excess-3 code, the inputlines must supply the bit combination of elements as specified by code and the output linesgenerate the corresponding bit combination of code. Each one of the four maps representsoneofthefouroutputsofthe circuitasafunctionofthe fourinputvariables.

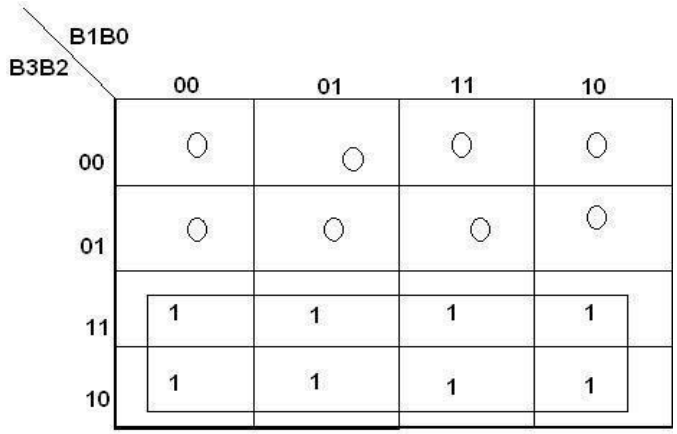
A two-level logic diagram may be obtained directly from the Boolean expressionsderivedbythemaps.Thesearevariousotherpossibilitiesforalogicdiagramthatimplementsthiscircuit.NowtheORgatewhoseoutputisC+Dhasbeenusedtoimplementpartiallyeachofthreeoutputs.

BINARYTOGRAYCODECONVERTOR

TRUTHTABLE:

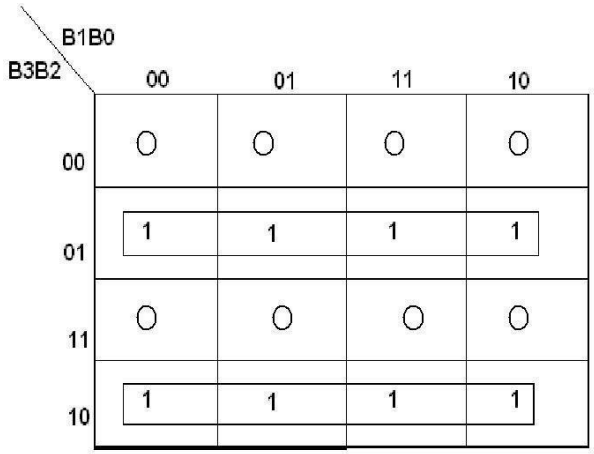
BinaryInput				GrayCodeOutput			
B3	**B2**	**B1**	**B0**	**G3**	**G2**	**G1**	**G0**
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	0

K-MapforG3



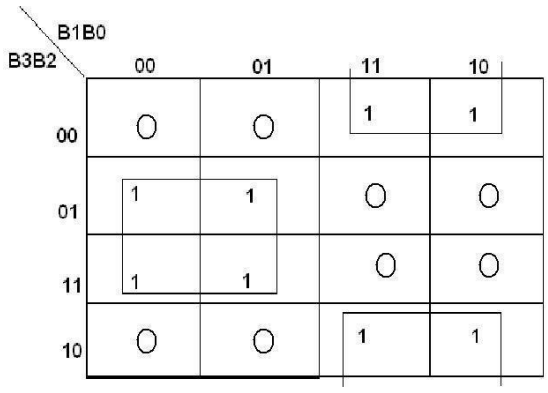
$G_3=B_3$

K-MapforG2_



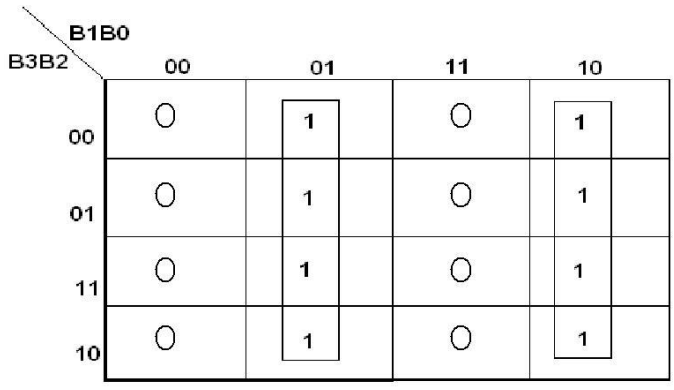
$G2 = B3 \oplus B2$

K-MapforG1_



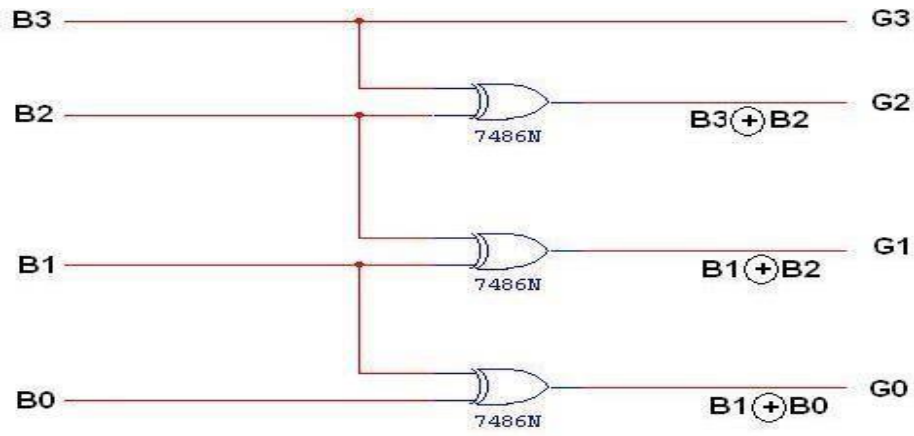
$G1 = B1 \oplus B2$

K-MapforG0_



$G0 = B1 \oplus B0$

LOGICDIAGRAM:

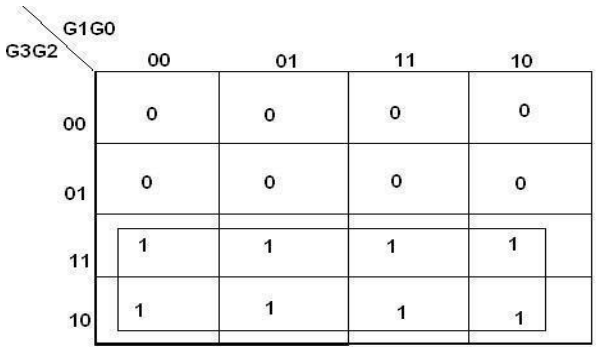


GRAYCODETOBINARY CONVERTOR

TRUTHTABLE:

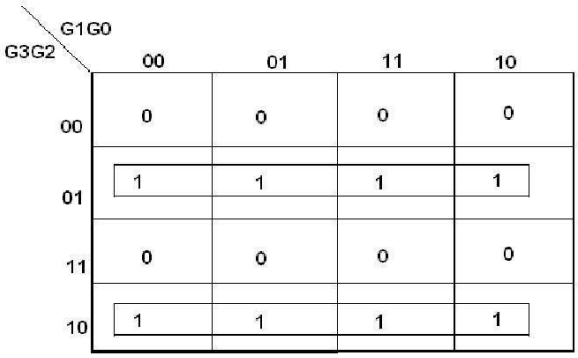
GRAYCODE				BINARYCODE			
G3	**G2**	**G1**	**G0**	**B3**	**B2**	**B1**	**B0**
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	1	0	0	1	0
0	0	1	0	0	0	1	1
0	1	1	0	0	1	0	0
0	1	1	1	0	1	0	1
0	1	0	1	0	1	1	0
0	1	0	0	0	1	1	1
1	1	0	0	1	0	0	0
1	1	0	1	1	0	0	1
1	1	1	1	1	0	1	0
1	1	1	0	1	0	1	1
1	0	1	0	1	1	0	0
1	0	1	1	1	1	0	1
1	0	0	1	1	1	1	0
1	0	0	0	1	1	1	1

K-MapforB3:



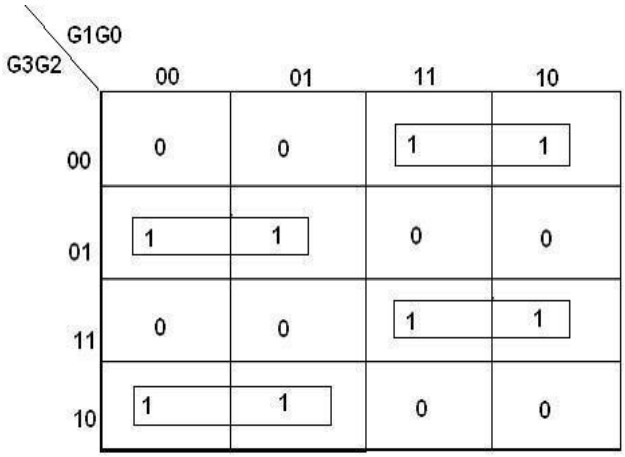
B3=G3

K-MapforB2:



B2 = G3⊕G2

K-MapforB1:



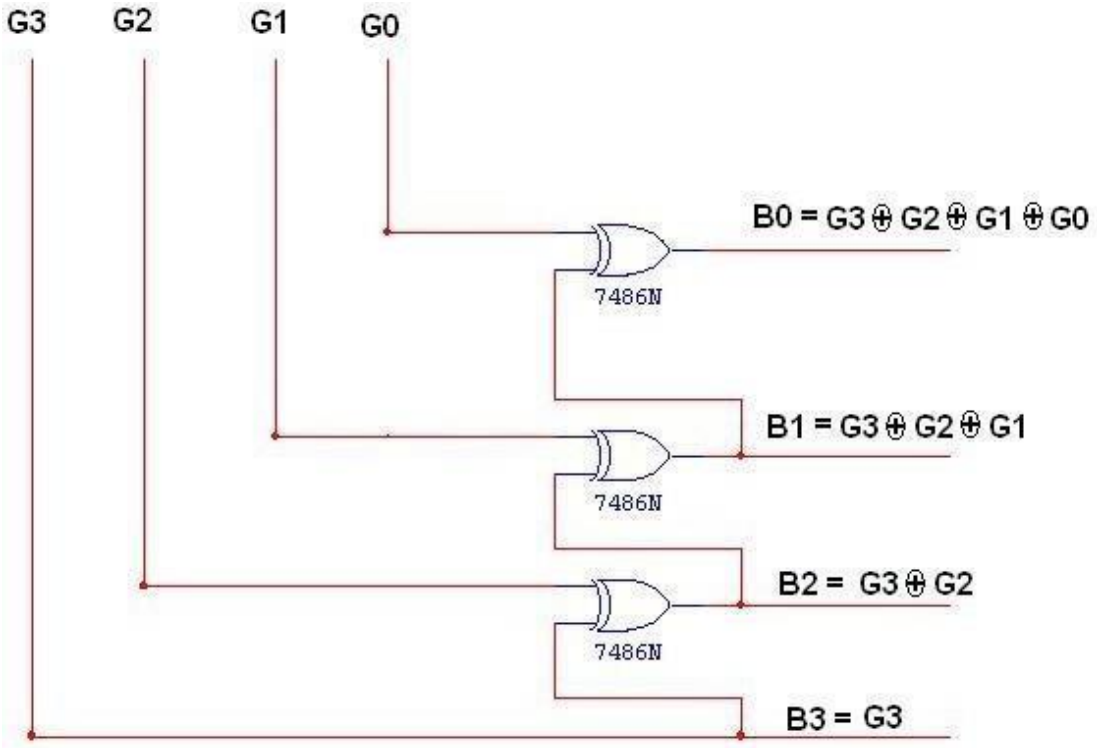
B1 = G3⊕G2⊕G1

K-MapforB0:

G3G2 \ G1G0	00	01	11	10
00	0	①	0	①
01	①	0	①	0
11	0	①	0	①
10	①	0	①	0

$$B0 = G3 \oplus G2 \oplus G1 \oplus G0$$

LOGICDIAGRAM:

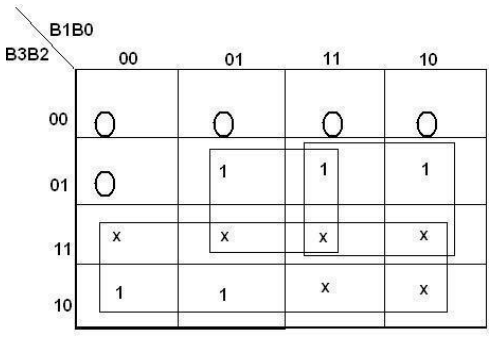


TRUTHTABLE: **BCDTOEXCESS-3CONVERTOR**

| **BCDinput** | **Excess – 3output** |

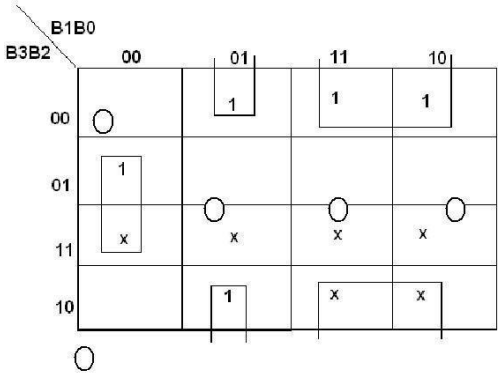
B3	B2	B1	B0	G3	G2	G1	G0
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0
1	0	1	0	x	x	x	x
1	0	1	1	x	x	x	x
1	1	0	0	x	x	x	x
1	1	0	1	x	x	x	x
1	1	1	0	x	x	x	x
1	1	1	1	x	x	x	X

K-MapforE3:



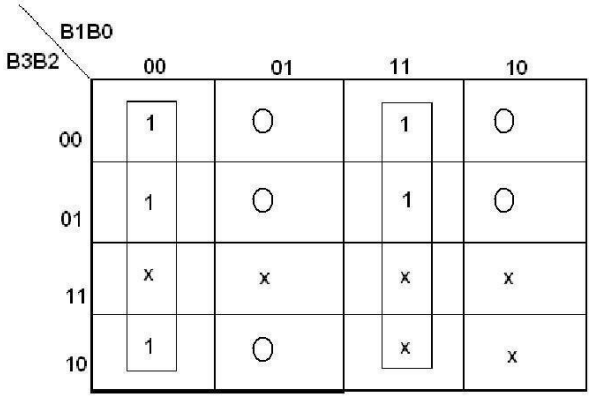
E3=B3+B2(B0+B1)

K-MapforE2:



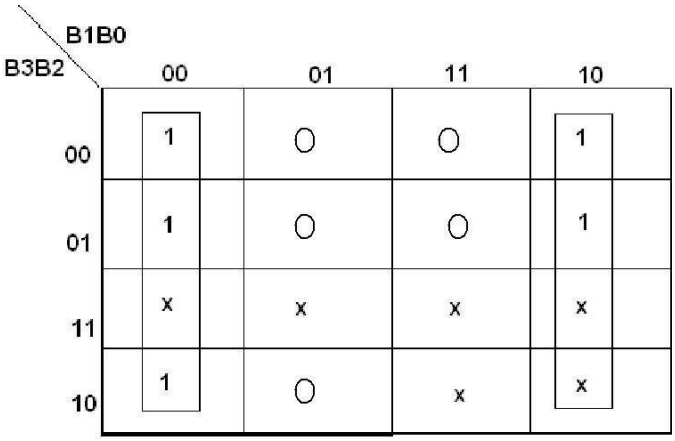
$E2 = B2 \oplus (B1 + B0)$

K-MapforE1:



$E1 = B1 \bar{\oplus} B0$

K-MapforE0:



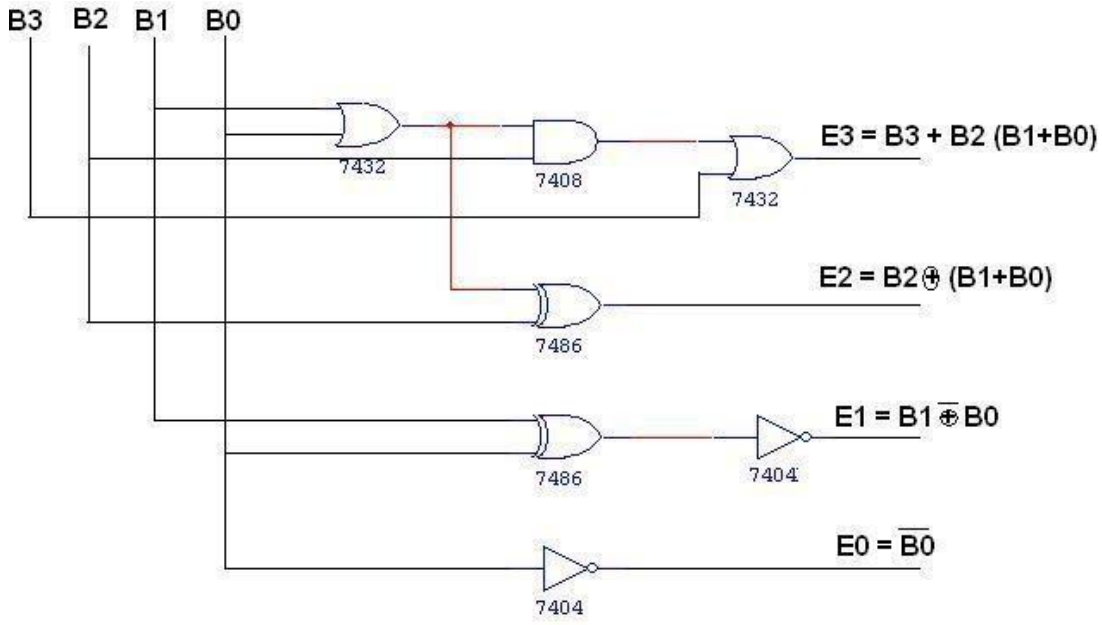
$E0 = \overline{B0}$

EXCESS-3 TO BCD

CONVERTORTRUTHTABLE:

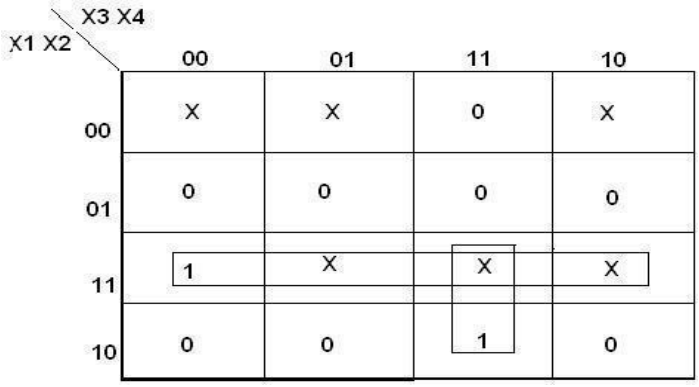
Excess– 3 Input				BCDOutput			
B3	**B2**	**B1**	**B0**	**G3**	**G2**	**G1**	**G0**
0	0	1	1	0	0	0	0
0	1	0	0	0	0	0	1
0	1	0	1	0	0	1	0
0	1	1	0	0	0	1	1
0	1	1	1	0	1	0	0
1	0	0	0	0	1	0	1
1	0	0	1	0	1	1	0
1	0	1	0	0	1	1	1
1	0	1	1	1	0	0	0
1	1	0	0	1	0	0	1

LOGICDIAGRAM:



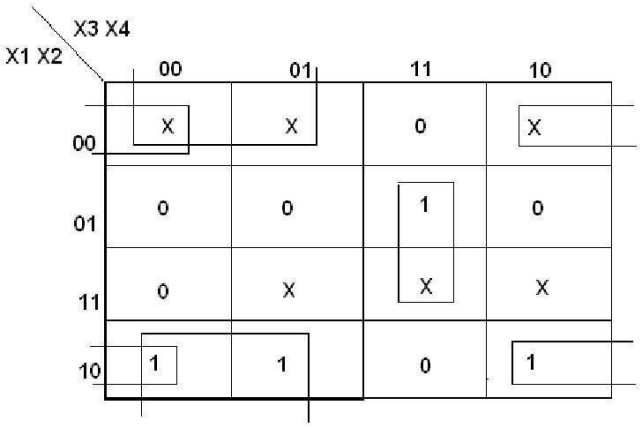
EXCESS-3TOBCDCONVERTOR

K-MapforA:



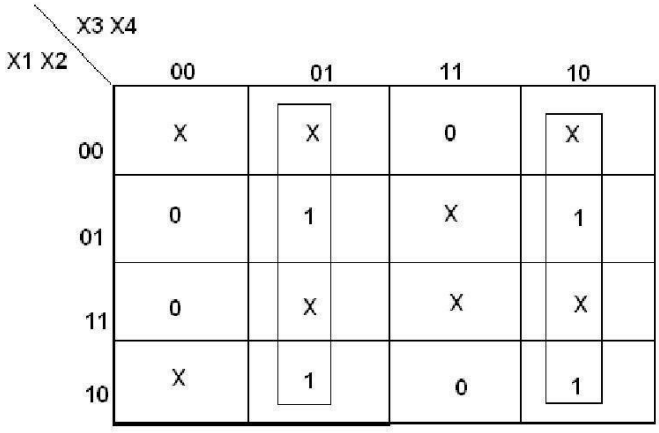
A =X1X2+X3X4X1

K-MapforB:



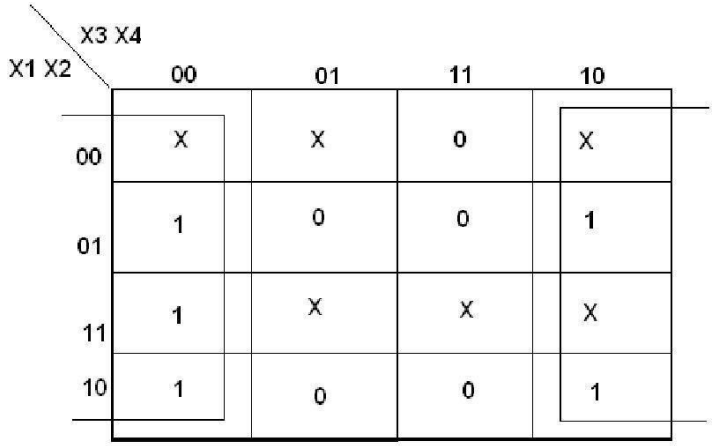
$$B = X2 \oplus (\overline{X3} + \overline{X4})$$

K-MapforC:



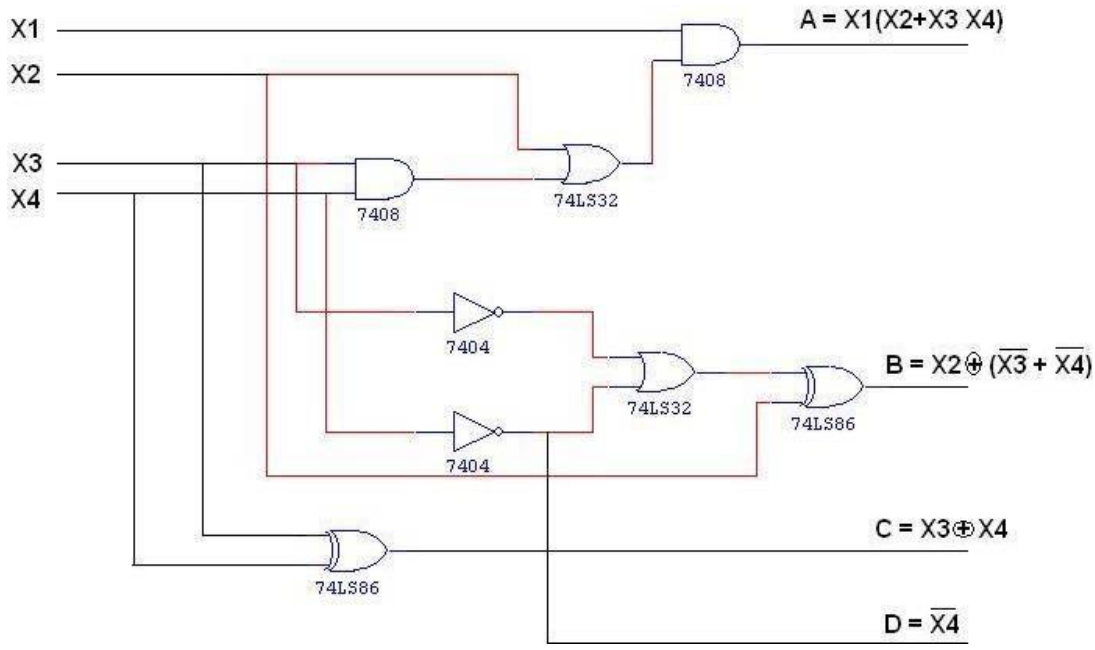
$$C = X3 \oplus X4_{23}$$

K-MapforD:



$D = \overline{X4}$

EXCESS-3TOBCDCONVERTOR



PROCEDURE:

(i) Connectionsweregivenaspercircuitdiagram.
(ii) Logicalinputswere givenaspertruthtable
(iii) Observethelogicaloutputandverifywiththe truthtables.

RESULT:

Thus, the following 4-bit converters aredesigned and constructed

(i)Binary to gray code converter
(ii)Gray to Binary code converter
(iii)BCD to Excess-3 code converter
(iv)Excess-3 to BCD code converter

Ex.No.-3 ADDERANDSUBTRACTOR

AIM:

To design and construct half adder, full adder, half subtractor and full subtractor circuits and verify the truth table using logic gates.

APPARATUSREQUIRED:

SL.NO.	COMPONENT	SPECIFICATION	QTY.
1.	ANDGATE	IC7408	1
2.	X-ORGATE	IC7486	1
3.	NOTGATE	IC7404	1
4.	ORGATE	IC7432	1
5.	ICTRAINERKIT	-	1
6.	PATCHCORDS	-	23

THEORY:

HALFADDER:

A half adder has two inputs for the two bits to be added and two outputs one fromthe sum ‘ S’ and other from the carry ‘ c’ into the higher adder position. Above circuit iscalledasacarrysignalfromtheadditionoftheless significantbitssumfromtheX-ORGatethe carryoutfromtheANDgate.

FULLADDER:

A full adder is a combinational circuit that forms the arithmetic sum of input; itconsists of three inputs and two outputs. A full adder is useful to add three bits at a time buta half adder cannot do so. In full adder sum output will be taken from X-OR Gate, carryoutputwillbetakenfromOR Gate.

HALFSUBTRACTOR:

The half subtractor is constructed using X-OR and AND Gate. The half subtractorhastwoinputandtwooutputs.Theoutputsaredifferenceand borrow.Thedifferencecanbeapplied using X-OR Gate, borrow output can be implemented using an AND Gate and aninverter.

FULLSUBTRACTOR:

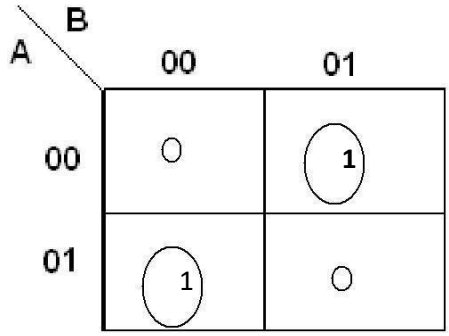
The full subtractor is a combination of X-OR, AND, OR, NOT Gates. In a fullsubtractorthelogiccircuitshouldhavethreeinputsandtwooutputs.Thetwohalfsubtractorput together gives a full subtractor .The first half subtractor will be C and A B. The outputwillbedifferenceoutputoffullsubtractor.TheexpressionABassemblestheborrowoutputo fthehalfsubtractorandthesecondtermistheinverteddifferenceoutputoffirstX-OR.

HALFADDER

TRUTHTABLE:

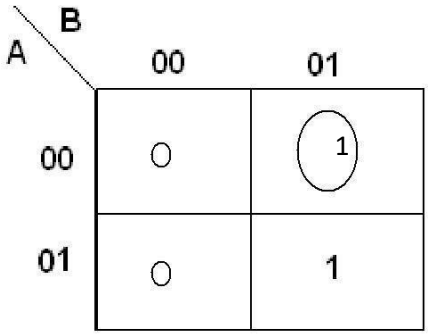
A	B	CARRY	SUM
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

K-MapforSUM:



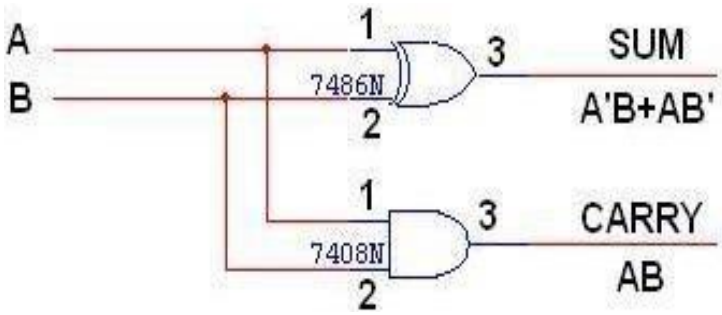
SUM=A'B+AB'

K-MapforCARRY:



CARRY=AB

LOGICDIAGRAM:

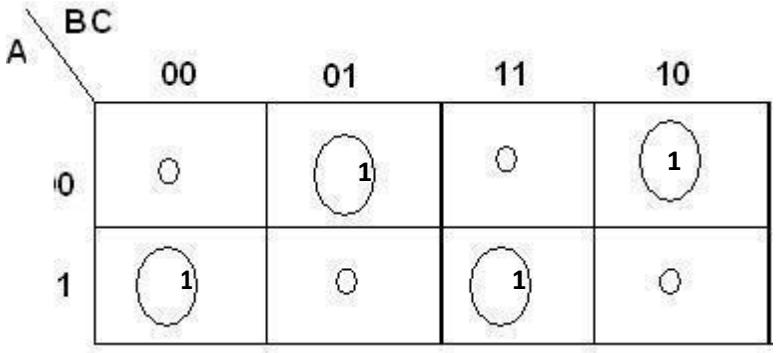


FULLADDER

TRUTHTABLE:

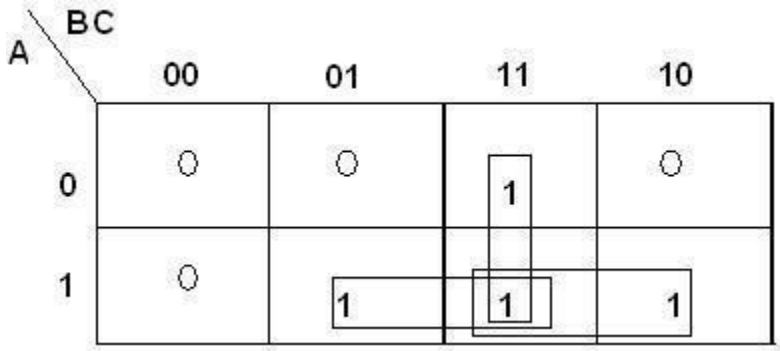
A	B	C	CARRY	SUM
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

K-MapforSUM



SUM=A'B'C+A'BC'+ABC'+ABC

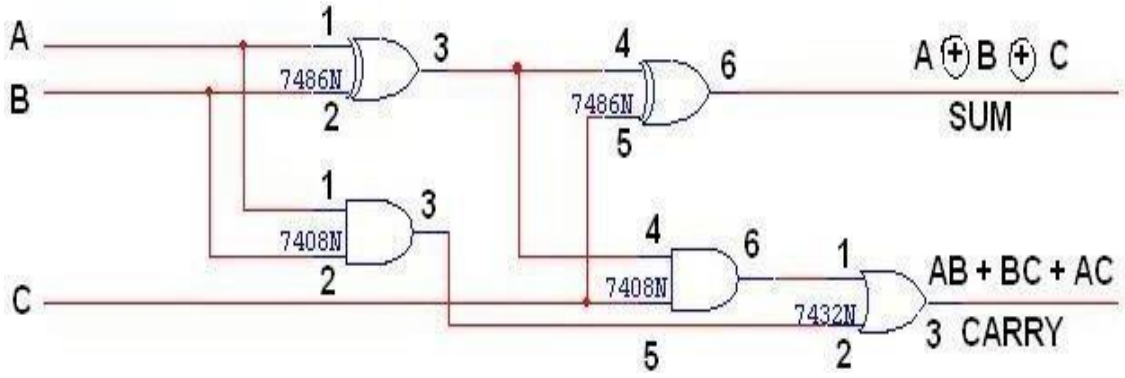
K-MapforCARRY



CARRY= AB +BC+ AC

LOGICDIAGRAM:

FULLADDERUSINGTWOHALF ADDER

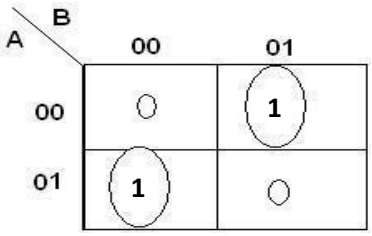


HALFSUBTRACTOR

TRUTHTABLE:

A	B	BORROW	DIFFERENCE
0	0	0	0
0	1	1	1
1	0	0	1
1	1	0	0

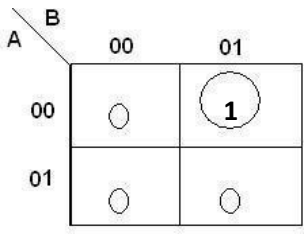
K-MapforDIFFERENCE



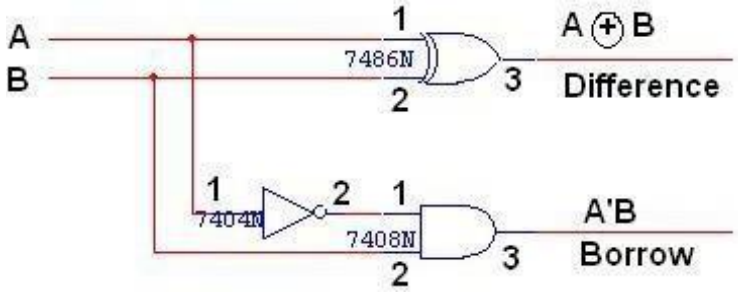
DIFFERENCE=A'B +AB'

K-MapforBORROW

BORROW=A'B



LOGICDIAGRAM

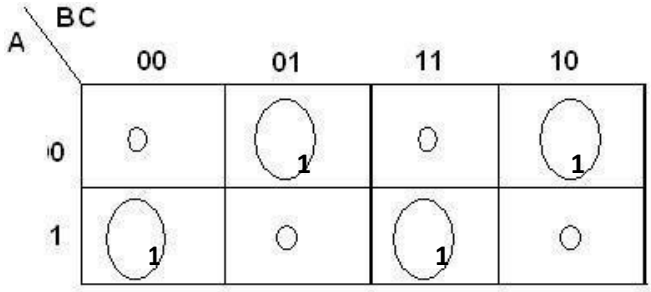


FULLSUBTRACTOR

TRUTH TABLE:

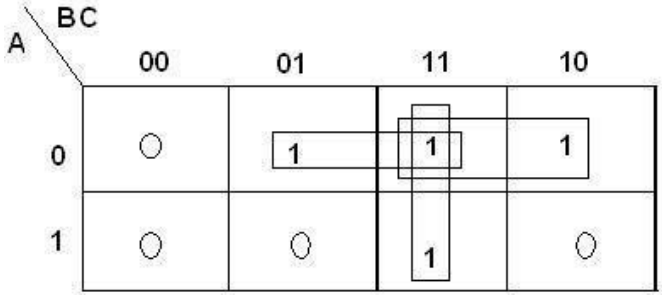
A	B	C	BORROW	DIFFERENCE
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

K-MapforDifference



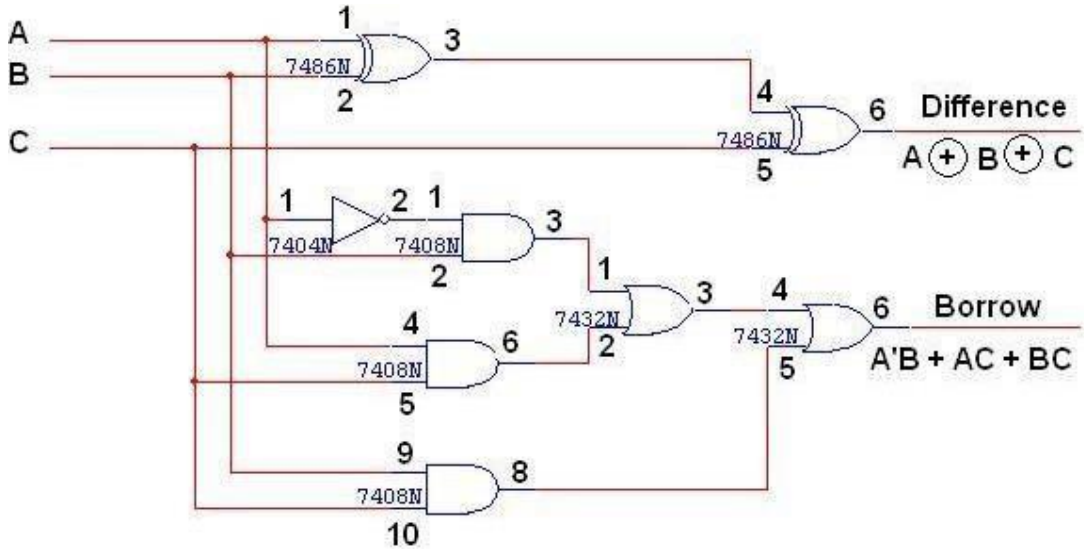
Difference =A'B'C+A'BC'+AB'C'+ABC

K-MapforBorrow

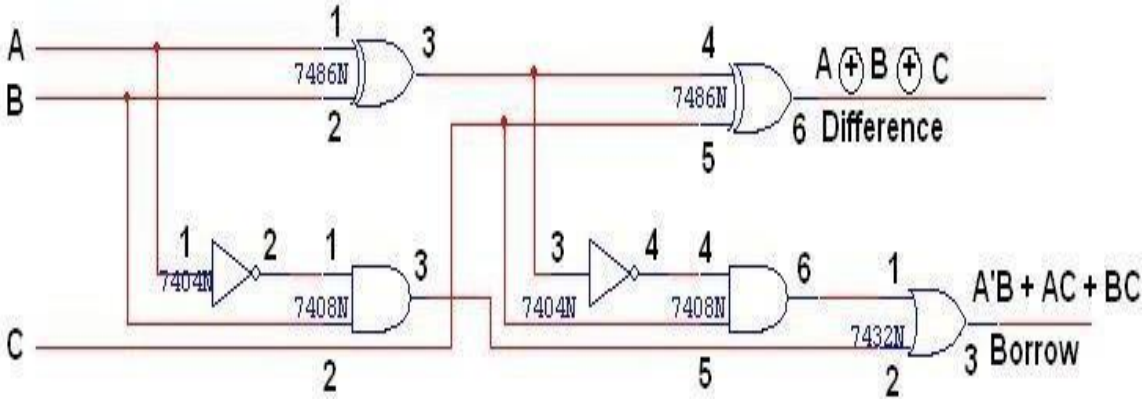


Borrow=A'B+BC+A'CLO

GICDIAGRAM:



FULLSUBTRACTORUSINGTWOHALFSUBTRACTOR



PROCEEDURE:

(i) Connectionsaregivenaspercircuitdiagram.

(ii) Logicalinputsaregivenaspercircuitdiagram.

(iii) Observetheoutput and verifythetruthtable.

RESULT:

Thus the half adder, full adder, half subtractor and full subtractor circuits are designed , constructed and verified the truth table using logic gates.

Ex.No.-4a 4-BITADDERANDSUBTRACTOR

AIM:

To design and implement 4-bit adder and subtractor using basic gates and MSI device IC7483

APPARATUSREQUIRED:

SL.NO.	COMPONENT	SPECIFICATION	QTY.
1.	IC	IC7483	1
2.	EX-ORGATE	IC7486	1
3.	NOT GATE	IC7404	1
3.	ICTRAINERKIT	-	1
4.	PATCHCORDS	-	40

THEORY:

4BITBINARYADDER:

A binary adder is a digital circuit that produces the arithmetic sum of two binarynumbers. It can be constructed with full adders connected in cascade, with the output carryfrom each full adder connected to the input carry of next full adder in chain. The augendsbits of 'A' and the addend bits of 'B' are designated by subscriptnumbersfrom right toleft,withsubscript0denotingtheleastsignificantbits.Thecarriesareconnectedinchain throughthefulladder.Theinput carrytotheadderis$C_0$and it ripplesthroughthefulladdertotheoutputcarryC_4.

4BITBINARYSUBTRACTOR:

ThecircuitforsubtractingA-Bconsistsof anadderwithinverters,placedbetween each data input 'B' and the corresponding input of full adder. The input carry C_0 must beequalto1whenperforming subtraction.

4BITBINARYADDER/SUBTRACTOR:

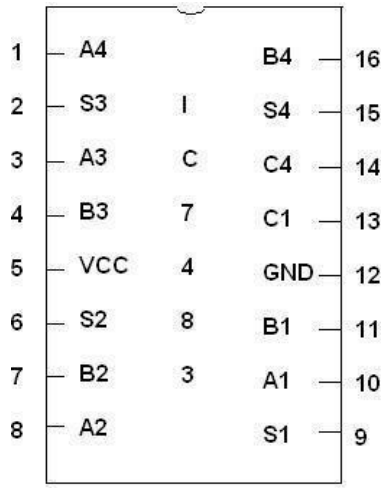
The addition and subtraction operation can be combined into one circuit with onecommon binary adder. The mode input M controls the operation. When M=0, the circuit isaddercircuit.WhenM=1,itbecomes subtractor.

4 BITBCDADDER:

Considerthearithmeticadditionoftwodecimaldigitsin BCD,togetherwithaninputcarryf romapreviousstage.Sinceeachinputdigitdoesnotexceed9, theoutputsumcannotbe greater than 19, the 1 in the sum being an input carry. The output of two decimaldigitsmustbe represented inBCD and should appearintheformlistedinthecolumns.

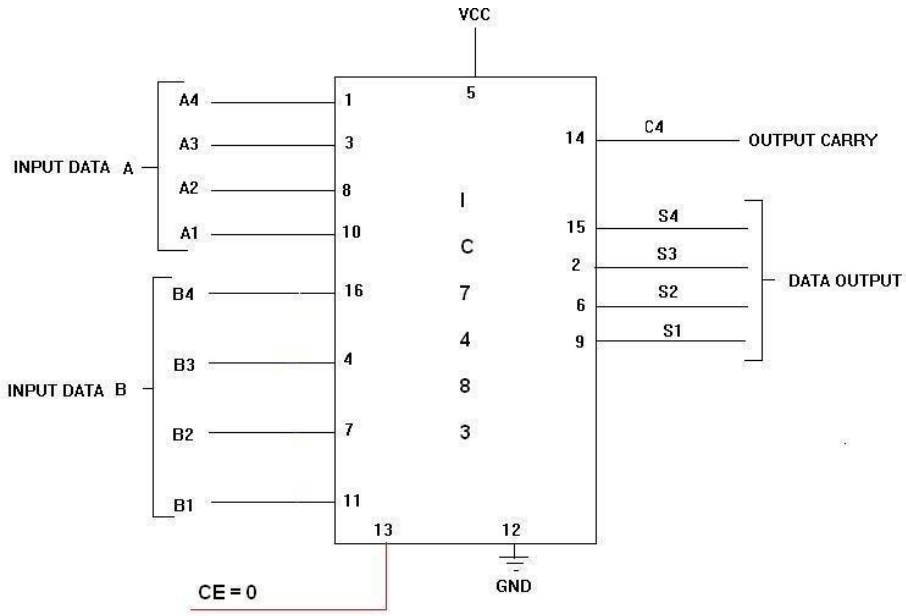
ABCDadderthatadds2BCDdigitsandproduceasumdigitinBCD.The2 decimaldigits, together with the input carry, are first addedin the top 4 bit adder to produce thebinarysum.

PINDIAGRAM FORIC7483:



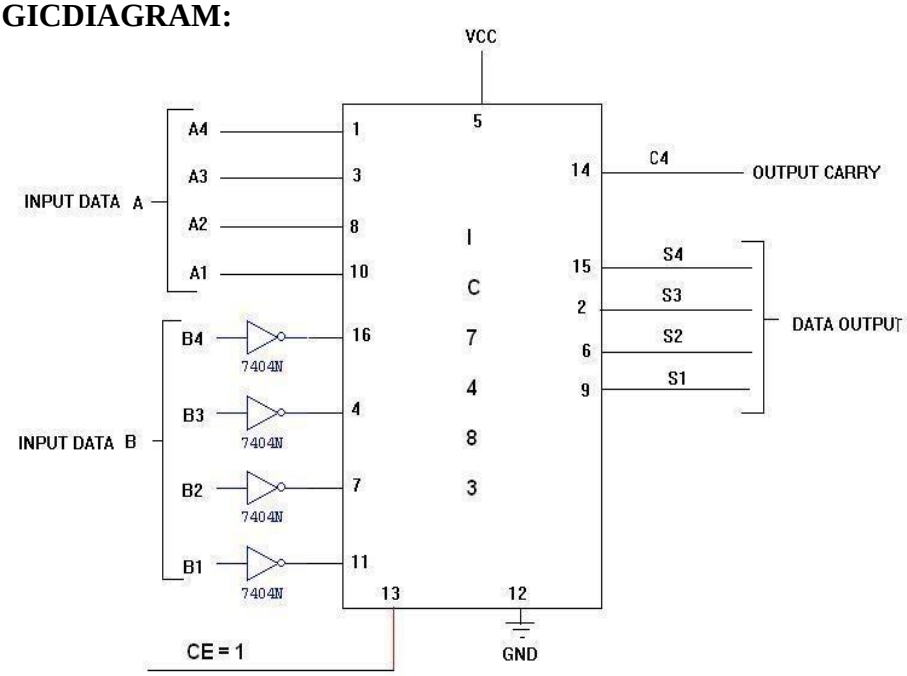
4-BITBINARY ADDER

LOGICDIAGRAM:



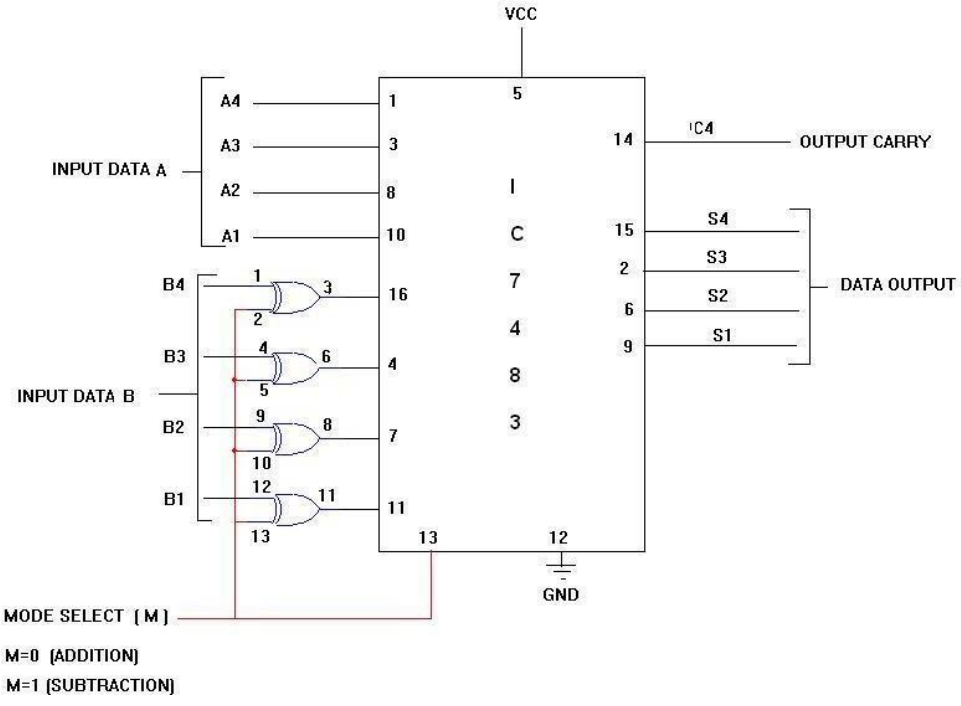
4-BITBINARYSUBTRACTOR

LOGICDIAGRAM:



4- BITBINARYADDER/SUBTRACTOR

LOGICDIAGRAM:



TRUTHTABLE:

InputDataA				InputDataB					Addition					Subtraction			
A4	A3	A2	A1	B4	B3	B2	B1	C	S4	S3	S2	S1	B	D4	D3	D2	D1
1	0	0	0	0	0	1	0	0	1	0	1	0	1	0	1	1	0
1	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0
0	0	1	0	1	0	0	0	0	1	0	1	0	0	1	0	1	0
0	0	0	1	0	1	1	1	0	1	0	0	0	0	1	0	1	0
1	0	1	0	1	0	1	1	1	0	0	1	0	0	1	1	1	1
1	1	1	0	1	1	1	1	1	1	0	1	0	0	1	1	1	1
1	0	1	0	1	1	0	1	1	0	1	1	1	0	1	1	0	1

PROCEDURE:

(i) Connectionsweregivenaspercircuitdiagram.
(ii) Logicalinputswere givenaspertruthtable
(iii) Observethelogicaloutputandverifywiththe truthtables.

RESULT:

Thus the 4-bit adder and subtractor using basic gates and MSI device IC7483 is designed and implemented

Ex.No.-4b

PARITYGENERATORANDCHECKER

AIM:

To design and verify the truth table of a three-bit Odd Parity generator and checker

APPARATUSREQUIRED:

SL.NO.	NAMEOFTHEAPPARATUS	RANGE	QUANTITY
1.	DigitalICtrainerkit		1
2.	EX-ORgate	IC 7486	
3.	NOTgate	IC 7404	
4.	Connectingwires		Asrequired

THEORY:

A parity bit is used for the purpose of detecting errors during transmission of binaryinformation. A parity bit is an extra bit included with a binary message to make the numberof 1's either odd or even. The message including the parity bit is transmitted and thenchecked at the receiving end for errors. An error is detected if the checked parity does notcorrespondwiththeonetransmitted.Thecircuitthatgeneratestheparitybitinthetransmitteris called a parity generator and the circuit that checks the parity in the receiveris called aparitychecker.

In even parity the addedparity bitwill make the total numberof 1's an evenamount and in odd parity the added parity bit will make the total number of 1's an oddamount.

In a three bit odd parity generator the three bits in the message together with theparity bit are transmitted to their destination, where they are applied to the parity checkercircuit.Theparitycheckercircuitchecksforpossibleerrorsinthe transmission.

Sincetheinformationwastransmittedwithoddparitythefourbitsreceivedmusthaveanod dnumberofl's. Anerroroccursduringthetransmissionifthefourbitsreceivedhavean evennumberof 1's,indicatingthat onebithaschangedduringtransmission.Theoutputoftheparitycheckerisdenotedby PEC(parityer rorcheck)andit will be equal to 1 if an error occurs, i.e., if the four bits received has an even number ofl's.

ODDPARITYGENERATOR

TRUTHTABLE:

SL.NO.	INPUT (Threebitmessage)			OUTPUT (OddParity bit)
	A	B	C	P
1.	0	0	0	1
2.	0	0	1	0
3.	0	1	0	0
4.	0	1	1	1
5.	1	0	0	0
6.	1	0	1	1
7.	1	1	0	1
8.	1	1	1	0

Fromthetruthtable

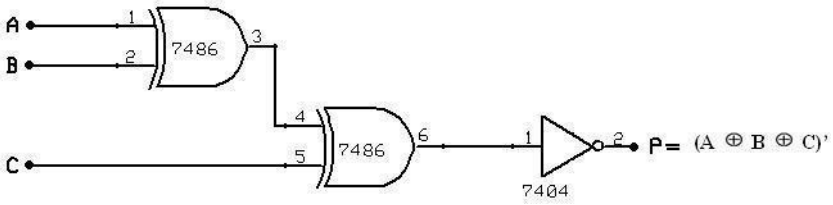
theexpressionfortheoutputparitybitis,P(A,B,C)=Σ (0,3,5,6)

Alsowrittenas,

P =A'B'C'+A'BC+AB'C+ABC'=(A ⊕B ⊕C)'

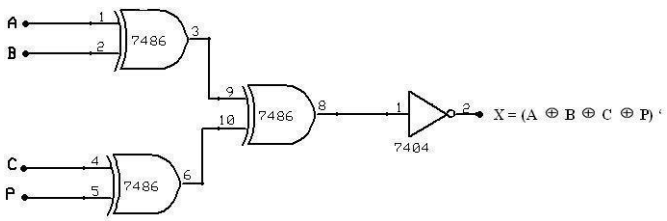
ODDPARITYGENERATOR

CIRCUITDIAGRAM:



ODDPARITYCHECKER

CIRCUITDIAGRAM:



ODDPARITYCHECKER

TRUTHTABLE:

SL.NO.	INPUT (4-BitMessage Received)				OUTPUT (ParityErrorCheck)
	A	B	C	P	X
1.	0	0	0	0	1
2.	0	0	0	1	0
3.	0	0	1	0	0
4.	0	0	1	1	1
5.	0	1	0	0	0
6.	0	1	0	1	1
7.	0	1	1	0	1
8.	0	1	1	1	0
9.	1	0	0	0	0
10.	1	0	0	1	1
11.	1	0	1	0	1
12.	1	0	1	1	0
13.	1	1	0	0	1
14.	1	1	0	1	0
15.	1	1	1	0	0
16.	1	1	1	1	1

Fromthetruthtabletheexpression fortheoutput paritycheckerbitis,X

$(A,B,C,P) = \Sigma(0,3,5, 6,9,10,12,15)$

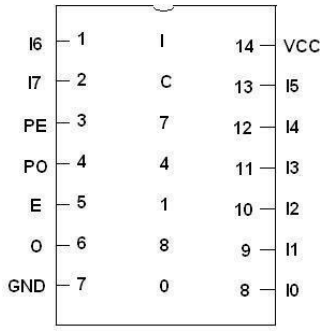
Theaboveexpression isreducedas,

$X=(A \oplus B \oplus C \oplus P)$

PROCEDURE:

1. Connectionsaregivenasper thecircuitdiagrams.
2. ForalltheICs7thpin isgroundedand14thpin isgiven+5 Vsupply.
3. Applytheinputsand verifythetruthtableforthe Paritygeneratorandchecker.

PINDIAGRAM FORIC74180:

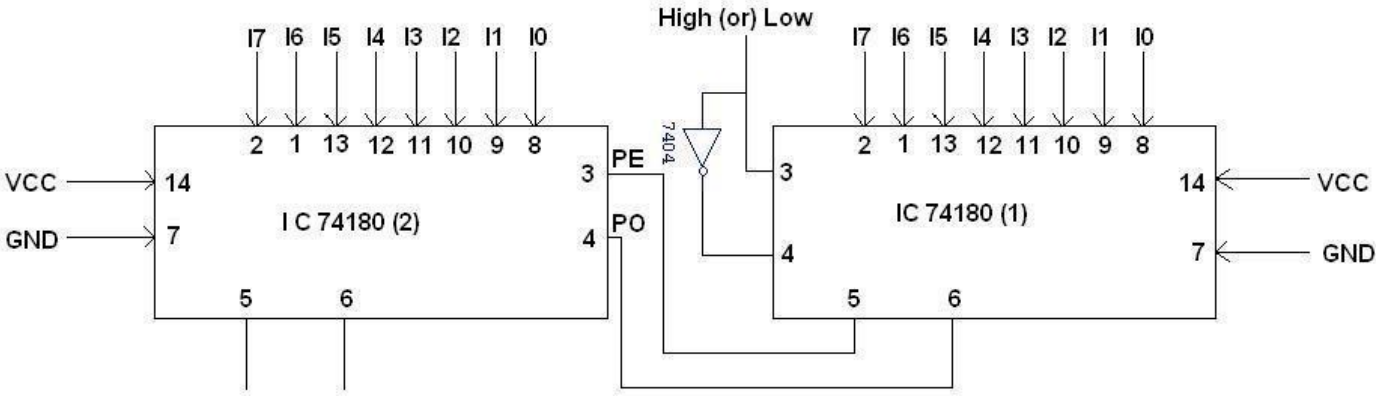


FUNCTIONTABLE:

INPUTS			OUTPUTS	
NumberofHighData Inputs(I0–I7)	PE	PO	∑E	∑O
EVEN	1	0	1	0
ODD	1	0	0	1
EVEN	0	1	0	1
ODD	0	1	1	0
X	1	1	0	0
X	0	0	1	1

16BITODD/EVENPARITYGENERATOR

LOGICDIAGRAM:

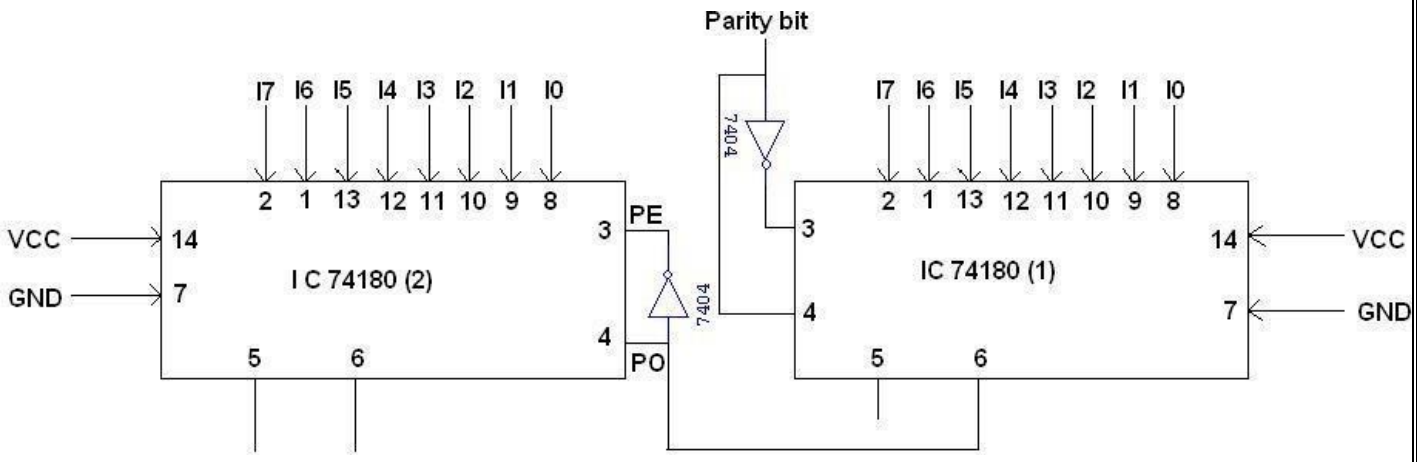


TRUTHTABLE:

I7I6I5I4I3I2I1I0	I7I6I5I4I3I2I1I0	Active	∑E	∑O
1 1 0 0 0 0 0 0	11 0 0 0 0 00	1	1	0
1 1 0 0 0 0 0 0	11 0 0 0 0 00	0	0	1
1 1 0 0 0 0 0 0	01 0 0 0 0 00	0	1	0

16BITODD/EVENPARITYCHECKER

LOGICDIAGRAM



TRUTHTABLE:

I7I6I5I4I3I2I1I0	I7'I6'I5'I4'I3'I2'11'I0'	Active	∑E	∑O
00 0 0 0 0 0 1	00 0 0 0 0 00	1	1	0
0 0 0 0 0 1 1 0	00 0 0 0 1 10	0	1	0
0 0 0 0 0 1 1 0	00 0 0 0 1 10	1	0	1

RESULT:

Thus the 3-bit and 16-bit odd parity generator and checker circuits were designed, implemented and their truth tables were verified.

.

Ex.No.-4c MAGNITUDE

COMPARATORAIM:

To design and implement the magnitude comparator using MSI device

APPARATUSREQUIRED:

SL.NO.	COMPONENT	SPECIFICATION	QTY.
1.	ANDGATE	IC7408	2
2.	X-ORGATE	IC7486	1
3.	ORGATE	IC7432	1
4.	NOT GATE	IC7404	1
5.	4-BITMAGNITUDECOMPARATOR	IC7485	2
6.	ICTRAINERKIT	-	1
7.	PATCHCORDS	-	30

THEORY:

The comparison of two numbers is an operator that determine one number is greaterthan, less than (or) equal to the other number. A magnitude comparator is a combinationalcircuit that compares two numbers A and B and determine their relative magnitude. Theoutcome of the comparator is specified by three binary variables thatindicate whetherA>B,A=B(or)A<B.

$A=A_3A_2A_1A_0B$

$= B_3B_2B_1B_0$

The equality of the two numbers and B is displayed in a combinational circuitdesignatedbythesymbol(A=B).

This indicates A greater than B, then inspect the relative magnitude of pairs ofsignificantdigitsstartingfrommostsignificantposition.Ais0andthat ofBis0.

WehaveA<B,the sequentialcomparisoncanbeexpandedas

$A>B=A3B^1+XAB^1+X\ XAB\ ^1+XXXAB^1$

3 32 2 3 2 11 3 2 1 0 0

$A<B=A_3{}^1B_3+X_3A_2{}^1B_2+X_3X2A_1{}^1B_1+X_3X_2X_1A_0{}^1B_0$

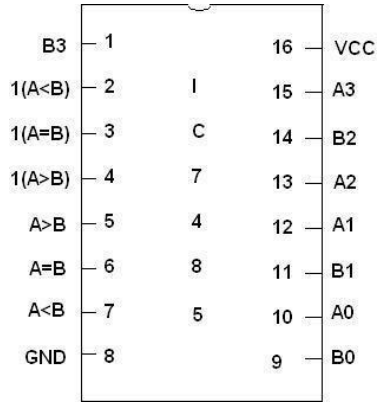
Thesamecircuitcanbeusedtocomparetherelativemagnitudeoftwo BCDdigits.Where,

A= Bisexpandedas,

$A=B=(A_3+B_3)(A_2+B_2)(A_1+B_1)(A_0+B_0)$

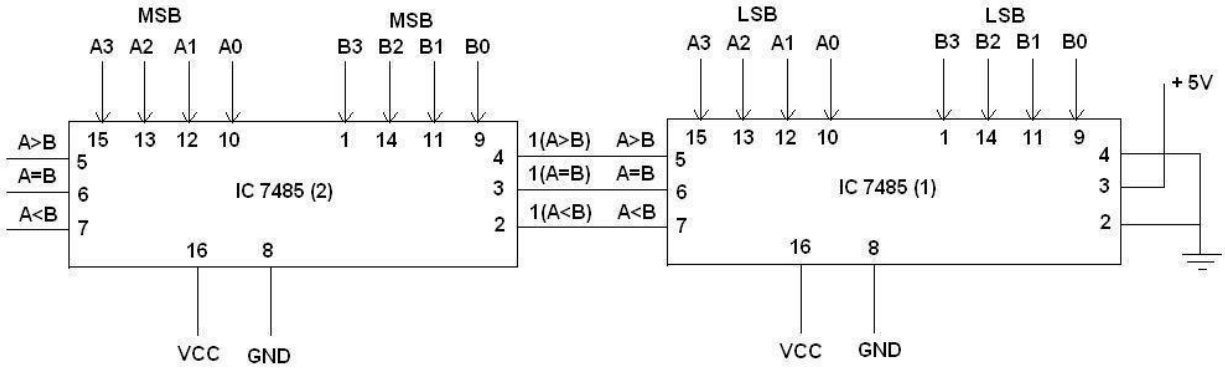
X_3 X_2 X_1 X_0

PINDIAGRAM FORIC7485:



8-BITMAGNITUDECOMPARATOR

LOGICDIAGRAM:



TRUTHTABLE:

A		B		A>B	A=B	A<B
0000	**0000**	**0000**	**0000**	**0**	**1**	**0**
0001	**0001**	**0000**	**0000**	**1**	**0**	**0**
0000	**0000**	**0001**	**0001**	**0**	**0**	**1**

PROCEDURE:

(i) Connectionsaregivenaspercircuitdiagram.
(ii) Logicalinputsaregivenaspercircuitdiagram.
(iii) Observetheoutput and verifythetruthtable.

RESULT:

Thus , the magnitude comparator using MSI device was designed and implemented

Ex.No.-4d MULTIPLEXERANDDEMULTIPLEXER

AIM:

To design and implement the multiplexer and demultiplexer using logic gates and study of IC74150 and IC74154

APPARATUSREQUIRED:

SL.NO.	COMPONENT	SPECIFICATION	QTY.
1.	3I/PANDGATE	IC7411	2
2.	ORGATE	IC7432	1
3.	NOT GATE	IC7404	1
2.	ICTRAINERKIT	-	1
3.	PATCHCORDS	-	32

THEORY:

MULTIPLEXER:

Multiplexer means transmitting a large number of information units over a smallernumber of channels or lines. A digital multiplexer is a combinational circuit that selectsbinary information from one of many input lines and directs it to a single output line. Theselectionofaparticularinputlineiscontrolledbyasetofselectionlines. Normallythere

n

are2 inputlineandnselectionlineswhosebitcombinationdeterminewhichinputis selected.

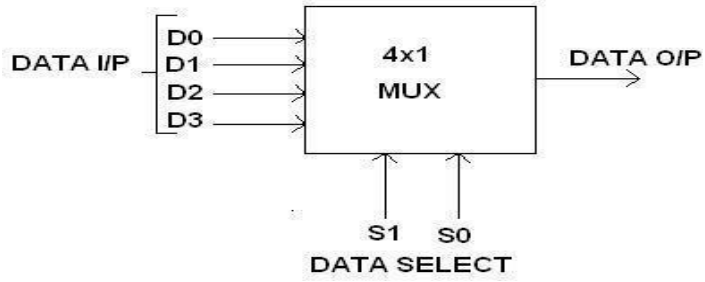
DEMULTIPLEXER:

ThefunctionofDemultiplexerisincontrasttomultiplexerfunction.Ittakesinformationfro monelineanddistributesittoagivennumberofoutputlines.Forthisreason,thedemultiplexerisalso knownasadatadistributor.Decodercanalsobeusedasdemultiplexer.

Inthe1:4demultiplexercircuit,thedatainputlinegoestoalloftheANDgates.Thedata select lines enable only one gate at a time and the data on the data input line will passthroughtheselectedgatetotheassociateddataoutputline.

4:1MULTIPLEXER

BLOCKDIAGRAMFOR4:1MULTIPLEXER:



FUNCTIONTABLE:

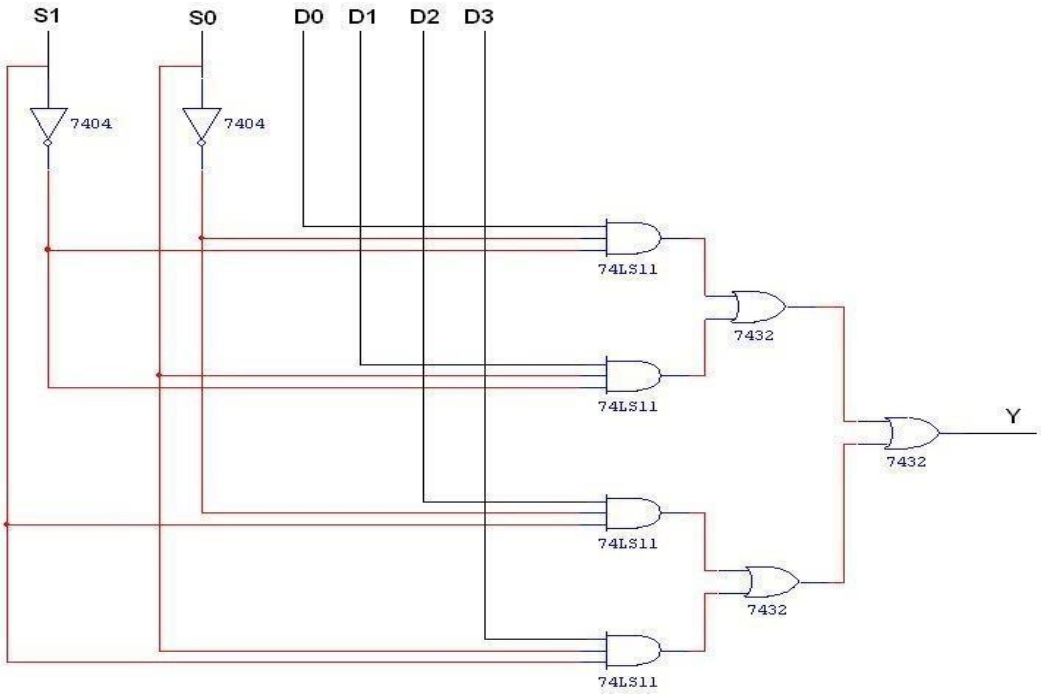
S1	S0	INPUTSY
0	0	D0→D0S1'S0'
0	1	D1→D1S1'S0
1	0	D2→D2S1S0'
1	1	D3→D3S1S0

Y

=D0S1'S0'+D1S1'S0+D2S1S0'+D3S1S0TRUTHTABLE:

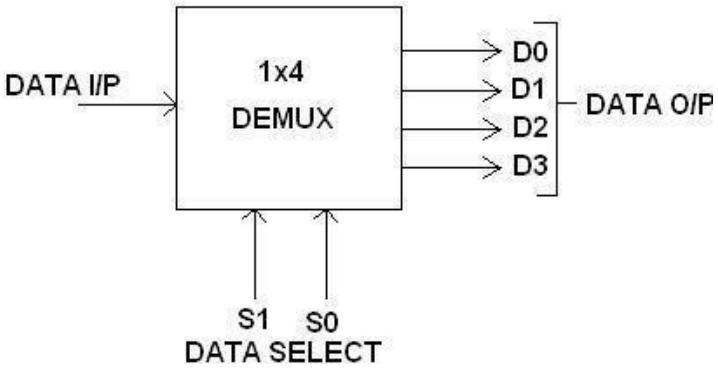
S1	S0	Y=OUTPUT
0	0	D0
0	1	D1
1	0	D2
1	1	D3

CIRCUITDIAGRAMFORMULTIPLEXER:



1:4DEMULTIPLEXER

BLOCKDIAGRAMFOR1:4DEMULTIPLEXER:



FUNCTIONTABLE:

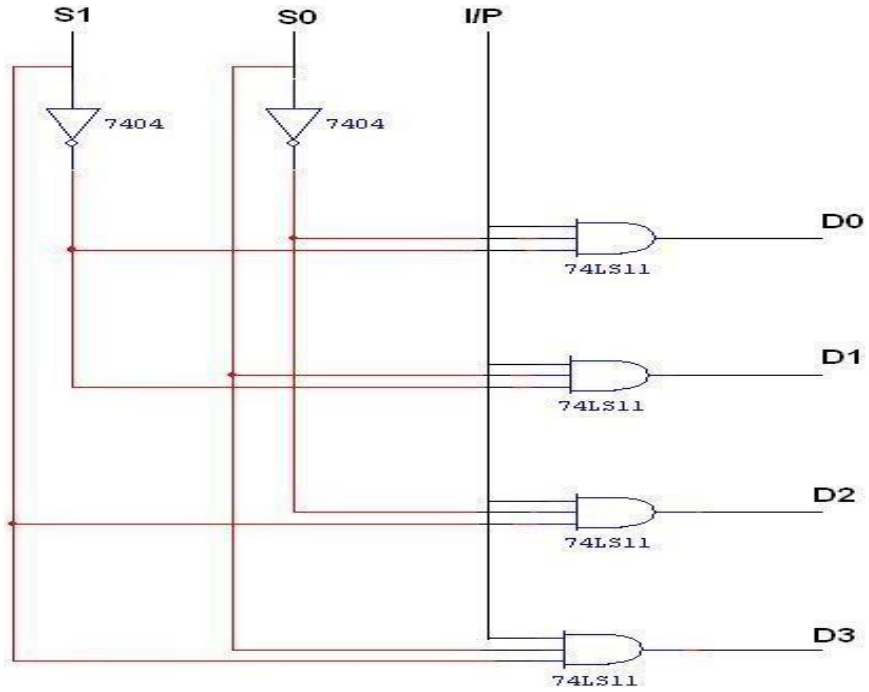
S1	S0	INPUT
0	0	X→D0=XS1'S0'
0	1	X →D1=XS1'S0
1	0	X →D2=XS1S0'
1	1	X →D3=XS1S0

Y=XS1'S0' +XS1'S0+XS1S0'+XS1S0

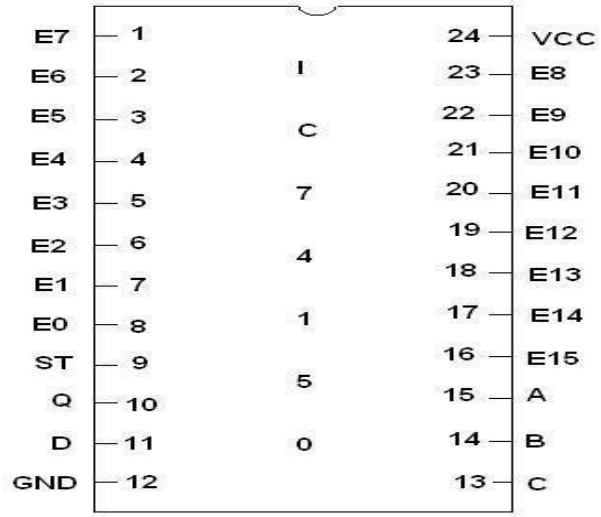
TRUTHTABLE:

INPUT			OUTPUT			
S1	**S0**	**I/P**	**D0**	**D1**	**D2**	**D3**
0	0	0	0	0	0	0
0	0	1	1	0	0	0
0	1	0	0	0	0	0
0	1	1	0	1	0	0
1	0	0	0	0	0	0
1	0	1	0	0	1	0
1	1	0	0	0	0	0
1	1	1	0	0	0	1

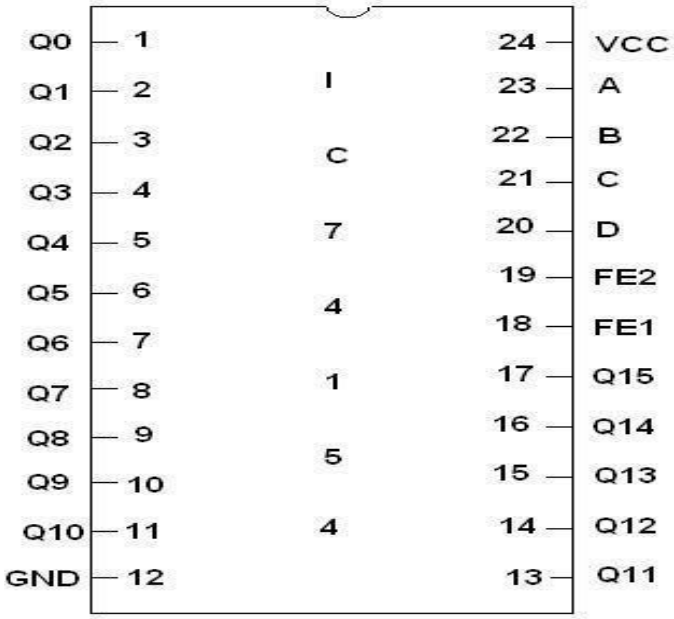
LOGICDIAGRAMFORDEMULTIPLEXER:



PINDIAGRAMFORIC74150:



PINDIAGRAMFORIC74154:



PROCEDURE:

(i) Connectionsaregivenaspercircuitdiagram.
(ii) Logicalinputsaregivenaspercircuitdiagram.
(iii) Observetheoutput and verifythetruthtable.

RESULT:

Thus, the multiplexer and demultiplexer using logic gates were designed and implemented.

Ex.No.-5 SHIFT REGISTER

AIM:

To design and implement the following shift registers

(i)Serial in serial out

(ii)Serial in Parallel out

(iii)Parallel in serial out

(iv)Parallel in Parallel out

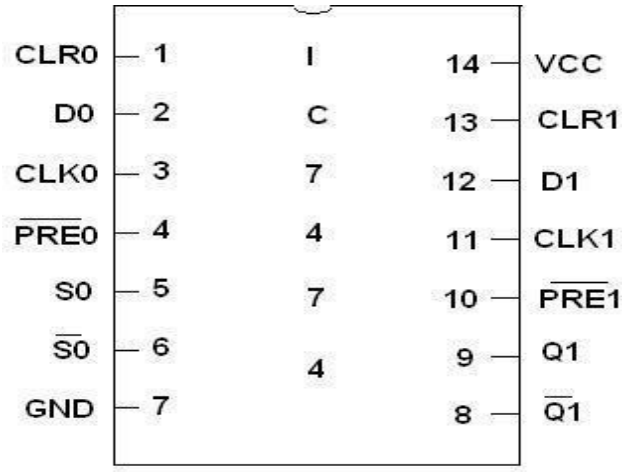
APPARATUSREQUIRED:

SL.NO.	COMPONENT	SPECIFICATION	QTY.
1.	DFLIPFLOP	IC7474	2
2.	ORGATE	IC7432	1
3.	ICTRAINERKIT	-	1
4.	PATCHCORDS	-	35

THEORY:

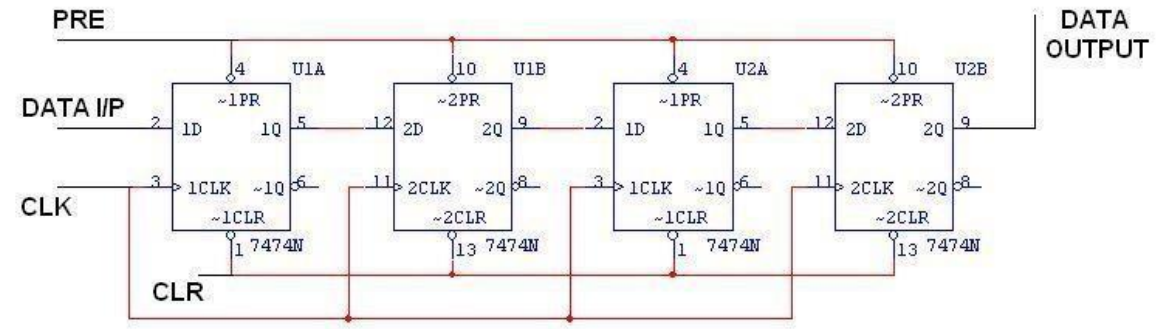
A register is capable of shifting its binary information in one or both directions isknown as shift register. The logical configuration of shift register consist of a D-Flip flopcascadedwithoutputofoneflipflopconnectedtoinputofnextflipflop.Allflipflopsreceivecom mon clock pulses which causes the shift in the output of the flip flop. The simplestpossible shift register is one that uses only flip flop. The output of a given flip flop isconnected to the input of next flip flop of the register. Each clock pulse shifts the contentofregisteronebitpositiontoright.

PINDIAGRAMOFIC7474:



SERIALINSERIALOUT

LOGICDIAGRAM:

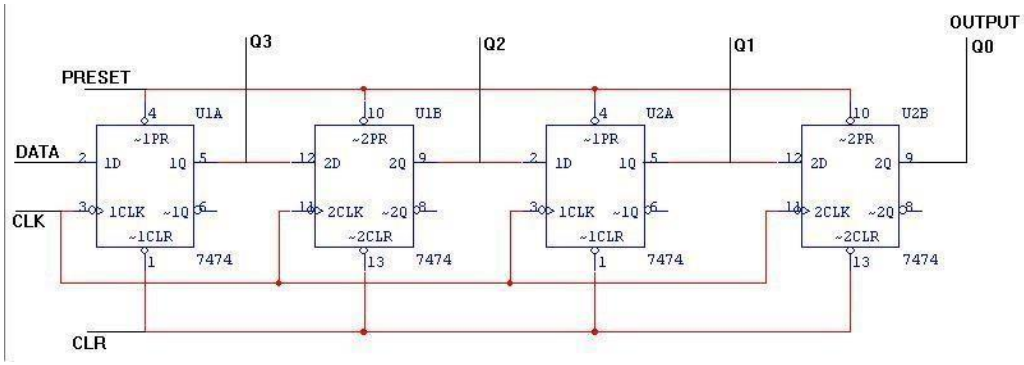


TRUTHTABLE:

CLK	SerialIn	SerialOut
1	1	0
2	0	0
3	0	0
4	1	1
5	X	0
6	X	0
7	X	1

SERIALINPARALLELOUT

LOGICDIAGRAM:

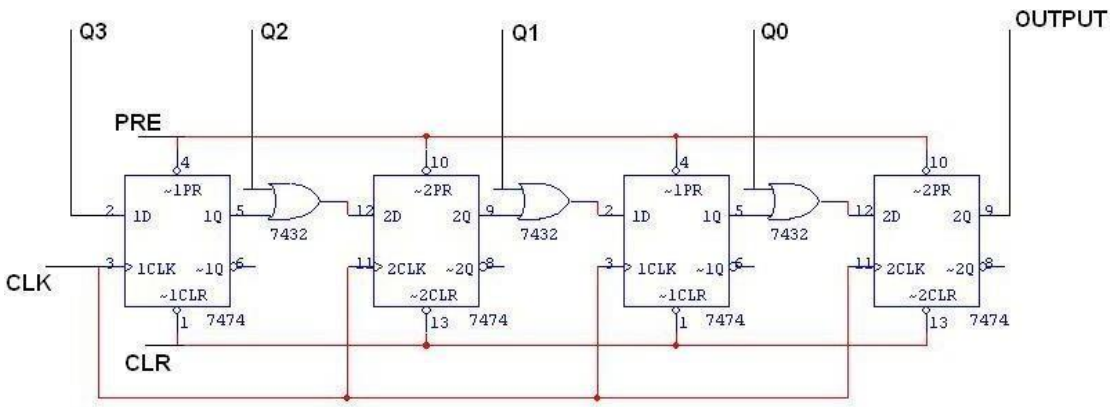


TRUTHTABLE:

CLK	DATA	OUTPUT			
		Q_A	Q_B	Q_C	Q_D
1	1	1	0	0	0
2	0	0	1	0	0
3	0	0	0	1	1
4	1	1	0	0	1

PARALLELINSERIALOUT

LOGICDIAGRAM:

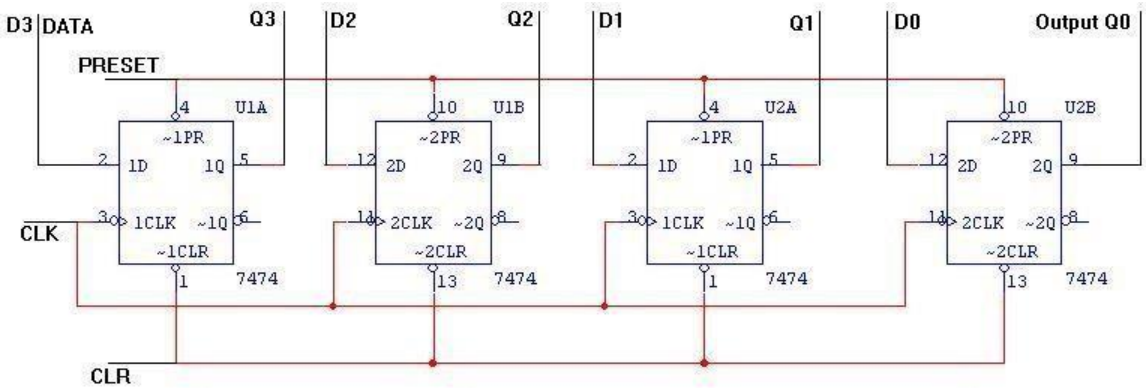


TRUTHTABLE:

CLK	Q3	Q2	Q1	Q0	O/P
0	1	0	0	1	1
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	1

PARALLELINPARALLELOUT

LOGICDIAGRAM:



TRUTHTABLE:

CLK	DATAINPUT				OUTPUT			
	D_A	D_B	D_C	D_D	Q_A	Q_B	Q_C	Q_D
1	1	0	0	1	1	0	0	1
2	1	0	1	0	1	0	1	0

PROCEDURE:

(i) Connectionsaregivenaspercircuitdiagram.
(ii) Logicalinputsaregivenaspercircuitdiagram.
(iii) Observetheoutput and verifythetruthtable.

RESULT:

Thus, the following Shift registers were designed and implemented

(i)Serial in serial out
(ii)Serial in Parallel out
(iii)Parallel in serial out
(iv)Parallel in Parallel out

Ex.No.-6

SYNCHRONOUSANDASYNCHRONOUSCOUNTERA

IM:

To design and implement synchronous and asynchronous counter

APPARATUSREQUIRED:

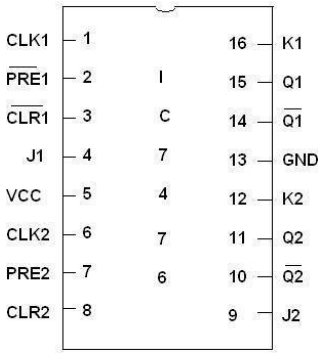
S.NO.	NAMEOFTHEAPPARATUS	RANGE	QUANTITY
1.	DigitalICtrainer kit		1
2.	JKFlipFlop	IC7473	2
3.	DFlipFlop	IC7473	1
4.	NAND gate	IC7400	1
5.	Connectingwires		Asrequired

THEORY:

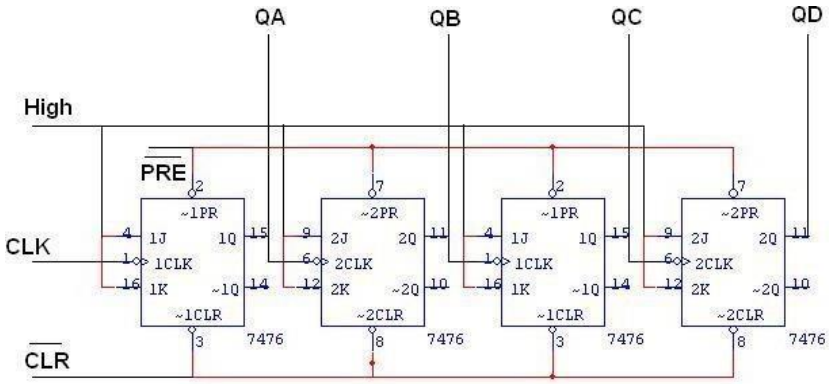
Asynchronousdecadecounterisalsocalledasripplecounter.Inaripplecountertheflipflop outputtransitionservesasasourcefortriggeringotherflipflops.Inotherwordstheclockpulseinput sofalltheflipflopsaretriggerednotbytheincomingpulsesbutrather bythe transition that occurs in other flip flops. The term asynchronous refers to the events thatdo not occur at the same time. With respect to the counter operation, asynchronous meansthat the flip flop within the counter are not made to change states at exactly the same time,they do not because the clock pulses are not connected directly to the clock input of eachflipflopinthecounter.

A counter is a register capable of counting number of clock pulse arriving at itsclock input. Counter represents the number of clock pulses arrived. A specified sequenceof states appears as counter output. This is the main difference between a register and acounter. There are two types of counter, synchronous and asynchronous. In synchronouscommon clock is given to all flip flop and in asynchronous first flip flop is clocked byexternal pulse and then each successive flip flop is clocked by Q or Q output of previousstage. A soon the clock of second stage is triggered by output of first stage. Because ofinherent propagation delay time all flip flops are not activated at same time which resultsinasynchronous operation.

PINDIAGRAMFORIC7476:



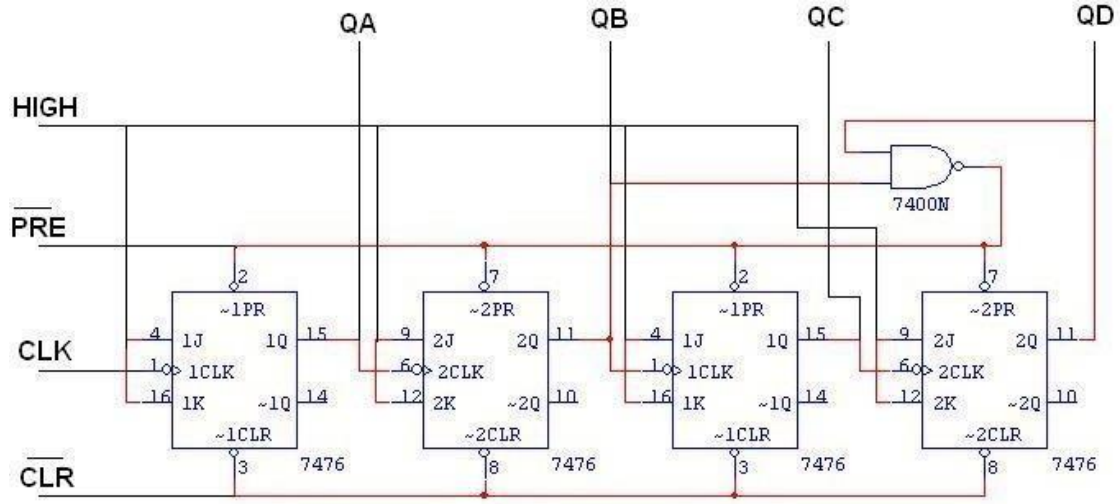
CIRCUITDIAGRAM:



TRUTHTABLE:

CLK	QA	QB	QC	QD
0	0	0	0	0
1	1	0	0	0
2	0	1	0	0
3	1	1	0	0
4	0	0	1	0
5	1	0	1	0
6	0	1	1	0
7	1	1	1	0
8	0	0	0	1
9	1	0	0	1
10	0	1	0	1
11	1	1	0	1
12	0	0	1	1
13	1	0	1	1
14	0	1	1	1
15	1	1	1	1

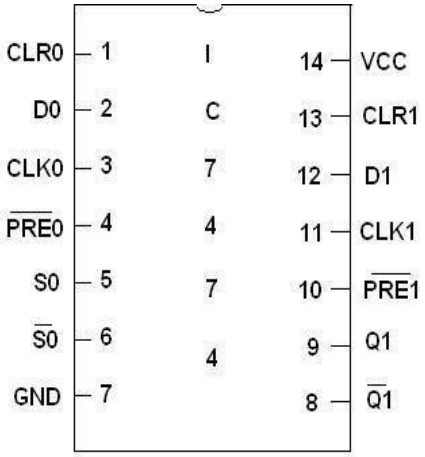
LOGICDIAGRAM FORMOD-10RIPPLECOUNTER:



TRUTH TABLE:

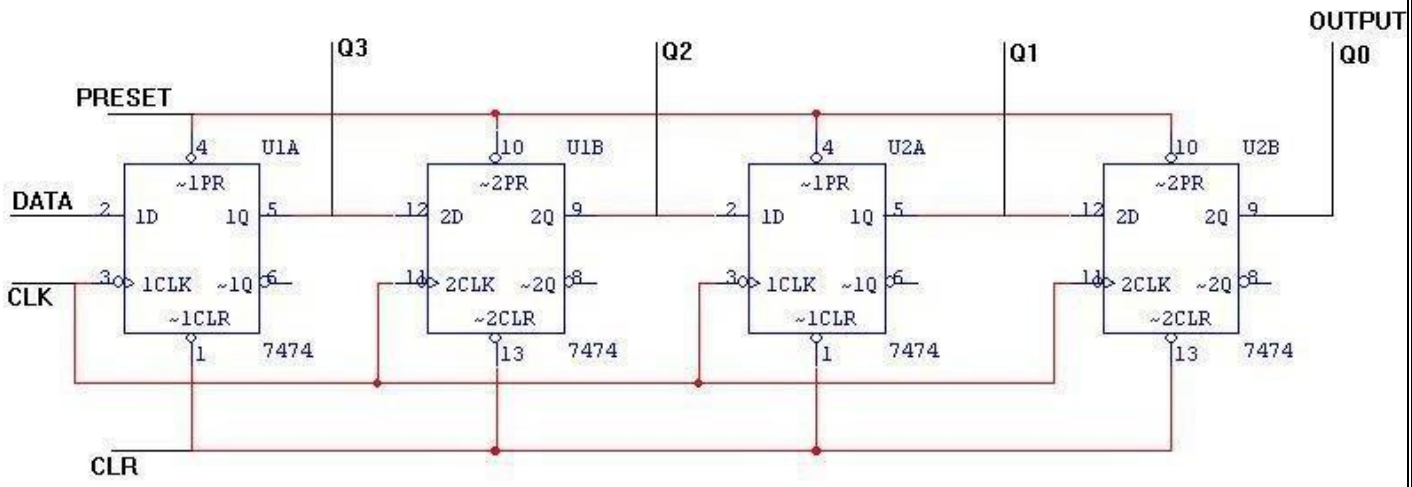
CLK	QA	QB	QC	QD
0	0	0	0	0
1	1	0	0	0
2	0	1	0	0
3	1	1	0	0
4	0	0	1	0
5	1	0	1	0
6	0	1	1	0
7	1	1	1	0
8	0	0	0	1
9	1	0	0	1
10	0	0	0	0

PINDIAGRAM:



SYNCHRONOUS COUNTER

LOGICDIAGRAM:



TRUTHTABLE:

CLK	DATA	OUTPUT			
		Q_A	Q_B	Q_C	Q_D
1	1	1	0	0	0
2	0	0	1	0	0
3	0	0	0	1	1
4	1	1	0	0	1

PROCEDURE:

(i) Connectionsaregivenaspercircuitdiagram.
(ii) Logicalinputsaregivenaspercircuitdiagram.
(iii) Observetheoutput and verifythetruthtable.

RESULT:

Thus, the synchronous and asynchronous counter were designed and implemented