Sri Siddhartha Institute of Technology

Maralur, Tumkur-572105

(A constituent college of Sri Siddhartha University)

2012-13

Department of Electronics & Communication

LOGIC DESIGN LAB MANUAL

III SEM BE

Name : ……

Sem : …… Sec: ……

Contents

GIVEN EXPRESSION:

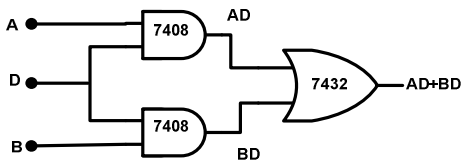
Y=(A,B,C,D)= Σ(5,7,9,11,13,15)

Truth Table:-

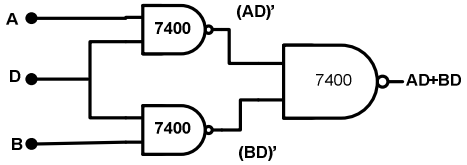
A	B	C	D	Y= AD+BD
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1

Implementation:

Using Basic Gates:



Using Universal Gates:



EXPERIMENT NO. 1: **DATE: / /**

SIMPLIFICATION, REALIZATION OF BOOLEAN EXPRESSION USING LOGIC GATES/UNIVERSAL GATES

AIM: To Simplify and Realize Boolean Expressions Using Logic Gates/Universal Gates.

APPARATUS REQUIRED:- IC Trainer Kit, patch chords, IC7408, IC7432 , IC7400, IC7402, IC 7404,IC 7486.

Procedure:

1. Place the IC in the socket of the trainer kit.

2. Make the connections as shown in the circuit diagram.

3. Apply the different combinations of input according to truth table and verify the output.

Simplification of expression:-

Y=(A,B,C,D)= Σ(5,7,9,11,13,15)

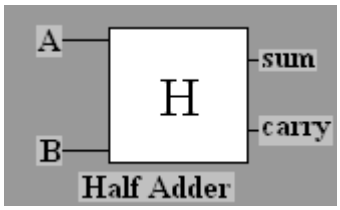
Write the expression using K-map

Y=A'BC'D+A'BCD+AB'C'D+AB'CD+ABC'D+ABCD
Y =A'BD (C+C')+AB'D(C+C')+ABD(C+C')
Y =A'BD+AB'D+ABD
Y =BD(A+A')+AB'D
Y=BD+AB'D
Y=D(B+AB')
Y=D(A+B)
Y =AD+BD

Exercise: Simplify and realize the following POS expn. and implement using nand gates only: Y(A,B,C,D)=π(5,7,9,11,13,15)

Half Adder

Logic Diagram

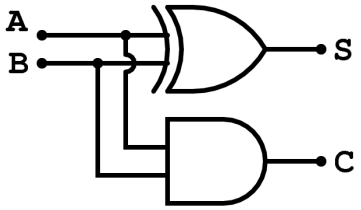


Truth Table

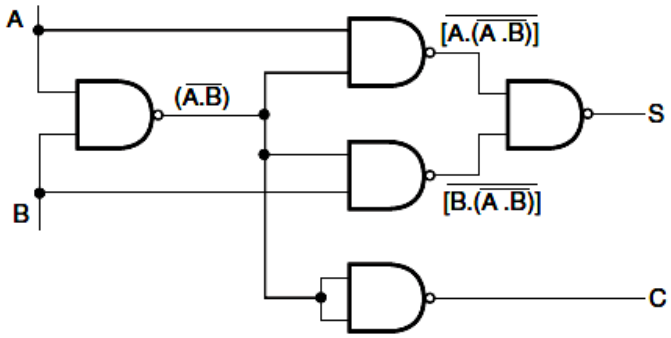
A	B	Sum (S)	Carry (C)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Circuit Diagram

USING BASIC AND XOR GATES



USING NAND GATES ONLY



EXPERIMENT NO. 2: **DATE: / /**

HALF/FULL ADDER AND HALF/FULL SUBTRACTORS.

Aim: - To realize half/full adder and half/full subtractor.

i. Using X-OR and basic gates

ii. Using only nand gates.

Apparatus Required: -

IC Trainer Kit, patch chords , IC 7486, IC 7432, IC 7408, IC 7400, etc.

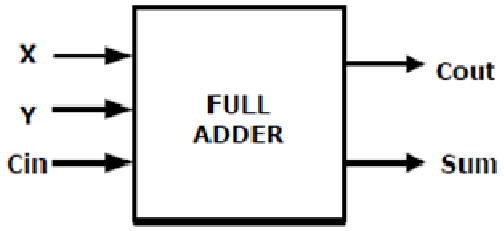
Procedure: -

1. Verify the gates.
2. Make the connections as per the circuit diagram.
3. Switch on V_{CC} and apply various combinations of input according to the truth table.
4. Note down the output readings for half/full adder and half/full subtractor sum/difference and the carry/borrow bit for different combinations of inputs.

Exercise: Implement half/full adder and half/full adder circuits using NOR gates only. Which is better, NAND or NOR? Why?

Full Adder

Logic Diagram

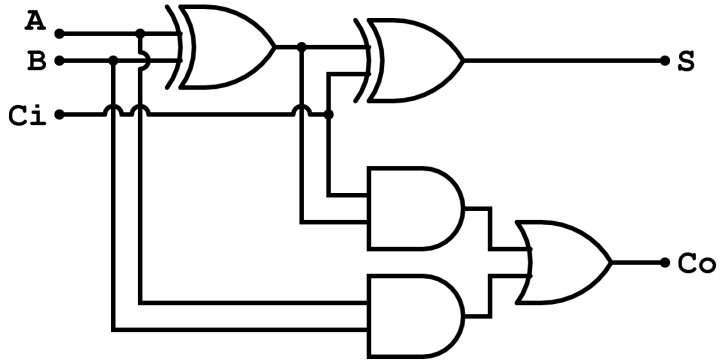


Truth Table

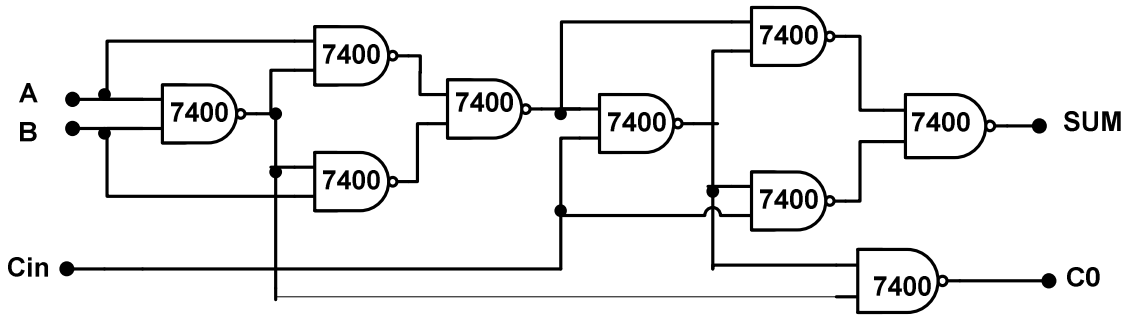
C_i	A	B	Sum (S)	Carry (C_o)
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Circuit Diagram

USING BASIC AND XOR GATES

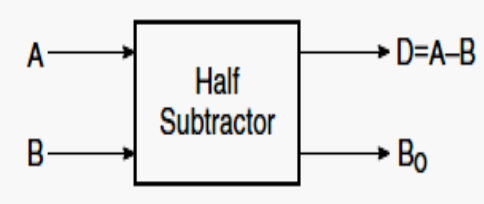


USING NAND GATES ONLY



Half subtractor

Logic Diagram

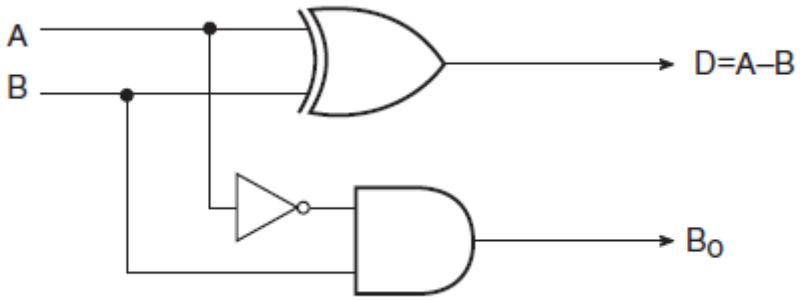


Truth Table

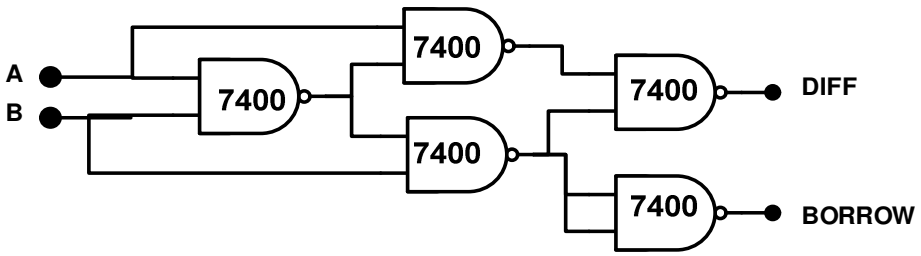
A	B	Diff (D)	Borrow (B_0)
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Circuit Diagram

USING BASIC AND XOR GATES

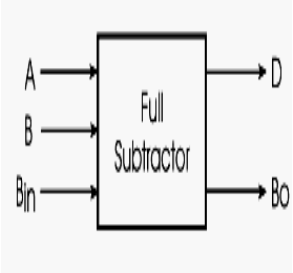


USING NAND GATES ONLY



Full subtractor

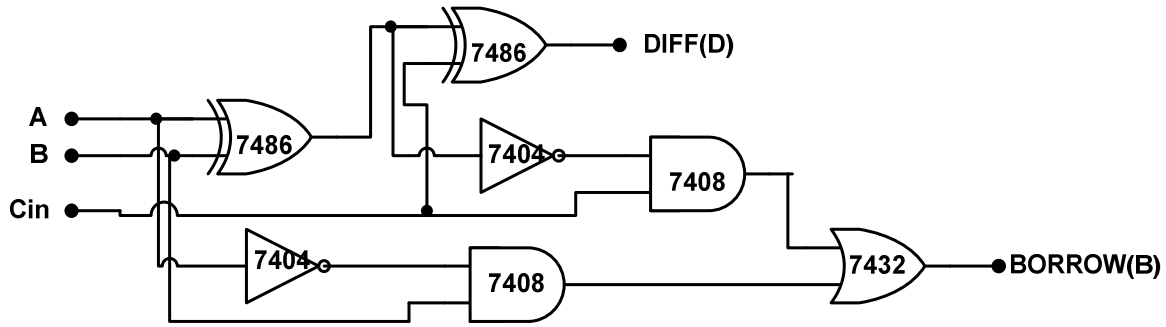
Logic Diagram:



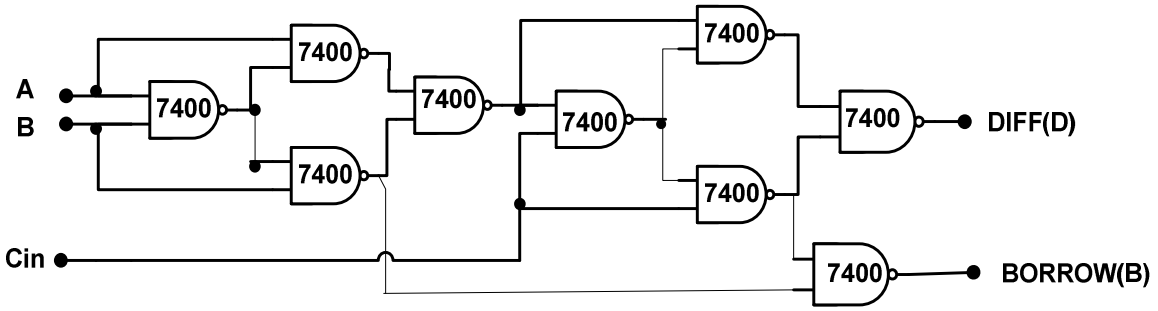
Truth Table

Minuend (A)	Subtrahend (B)	Borrow In (B_{in})	Difference (D)	Borrow Out (B_O)
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

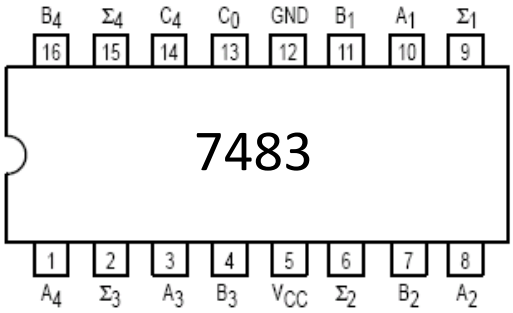
Circuit Diagram
(USING BASIC AND XOR GATES)



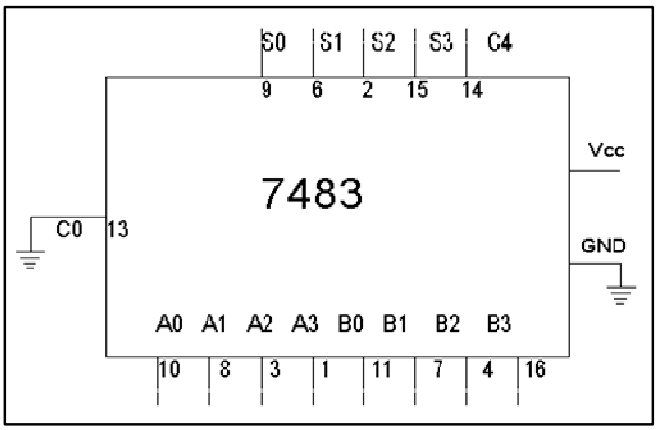
(USING NAND GATES ONLY)



Pin Detail: -



Adder: -



Truth Table: -

A3	A2	A1	A0	B3	B2	B1	B0	C4 (V)	S3(V)	S2(V)	S1(V)	S0(V)
0	0	0	1	0	0	1	0	0	0	0	1	1
0	1	0	1	1	0	1	1	1	1	0	0	0
1	0	1	0	1	0	1	0	1	0	1	0	0
1	1	1	1	1	1	1	1	1	1	1	1	0
0	1	1	1	0	0	1	1	0	1	0	1	0

EXPERIMENT NO. 3: **DATE: / /**

PARALLEL ADDER/SUBTRACTOR

AIM: - To realize IC 7483 as parallel adder / Subtractor**.**

Apparatus Required: -

IC Trainer Kit, patch chords, IC 7483, IC 7404, etc.

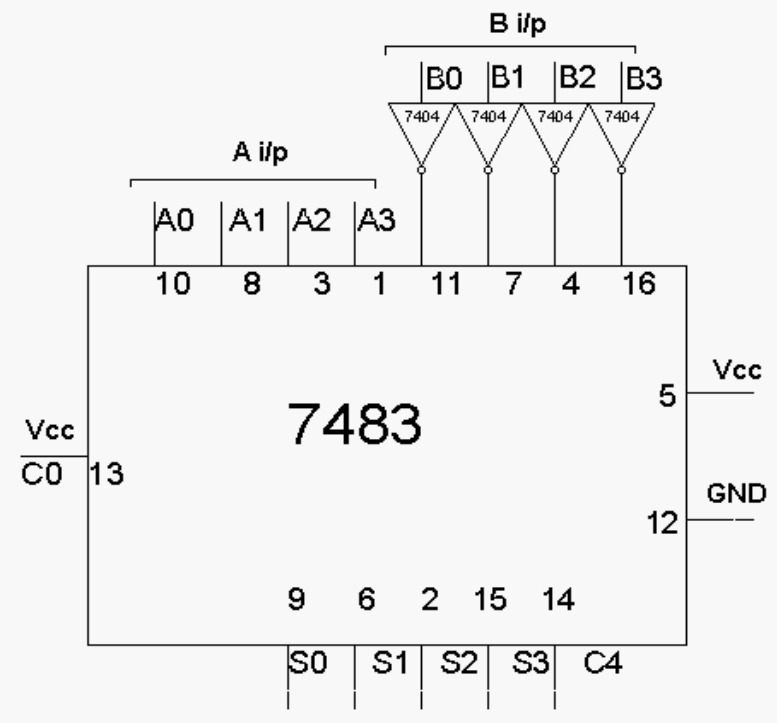
Procedure: -

1. Apply the inputs to A0 to A3 and B0 to B3.
2. Connect C0 to the Ground.
3. Check the output sum on the S0 to S3 and also C4.
4. For subtraction connect C0 to Vcc, Apply the B input through NOT gate, which gives the complement of B.
5. The truth table of adder and Subtractor are noted down.

Exercise:

Implement parallel adder/subtractor using IC 7483 and xor gates.

Subtractor:-



Truth Table for Subtractor

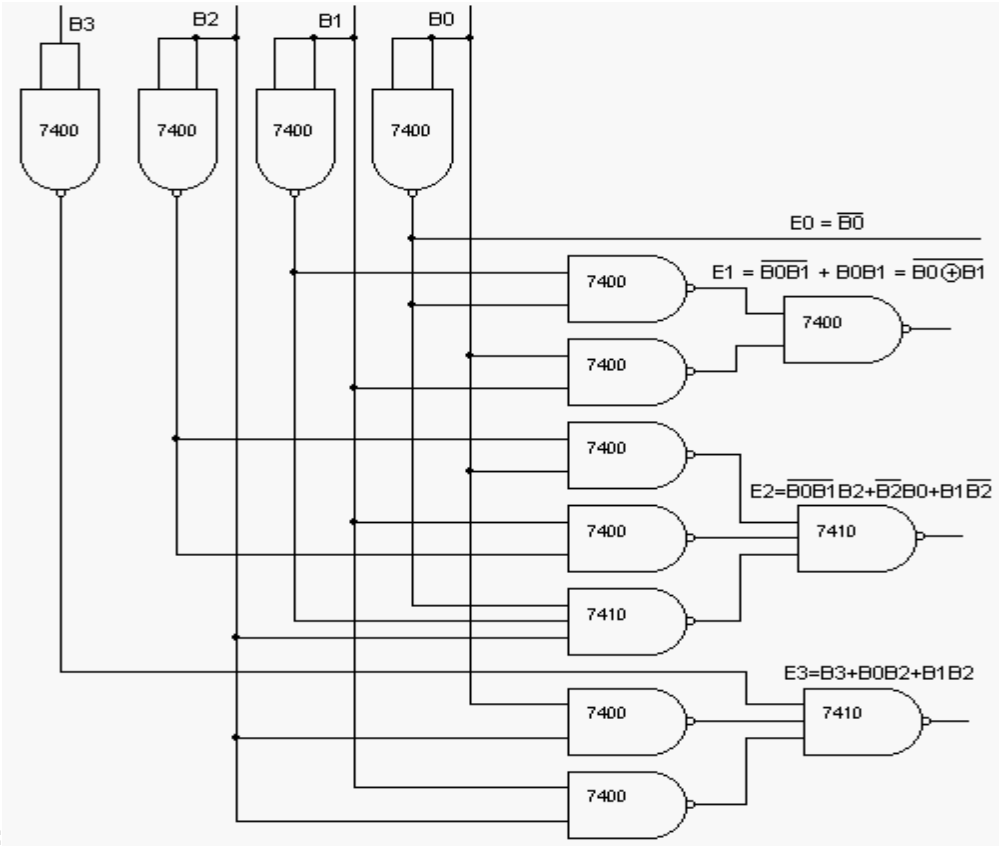
A3	A2	A1	A0	B3	B2	B1	B0	C4(V)	S3(V)	S2(V)	S1(V)	S0(V)
0	0	1	0	0	0	0	1	1	0	0	0	1
0	1	0	1	0	0	1	1	1	0	0	1	0
0	0	1	1	0	1	0	1	0	1	1	1	0
1	0	1	0	0	1	1	0	1	0	1	0	0
1	0	0	0	1	1	1	1	0	1	0	0	1

BCD To Excess-3:

Truth Table For Code Conversion: -

Inputs				Outputs			
B3	B2	B1	B0	E3 (v)	E2 (v)	E1 (v)	E0 (v)
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

Circuit diagram using NAND gates only:



EXPERIMENT NO. 4: **DATE: / /**

BCD TO EXCESS-3 AND EXCESS-3 TO BCD CODE CONVERTER

AIM: - To study and verify BCD to excess –3 code and excess-3 to BCD code conversion using NAND gates.

APPARATUS REQUIRED: -

IC Trainer Kit, patch chords, IC 7400, IC 7404, etc.

Procedure: - (BCD Excess 3 and Vice Versa)

1. Make the connections as shown in the fig.
2. Pin [14] of all IC'S are connected to +5V and pin [7] to the ground.
3. The inputs are applied at E3, E2, E1, and E0 and the corresponding outputs at B3, B2, B1, and B0 are taken for excess – 3 to BCD.
4. The inputs are applied at B3, B2, B1, and B0 and the corresponding outputs are E3, E2, E1 and E0 for BCD to excess – 3.
5. Truth tables are verified.

Exercise:

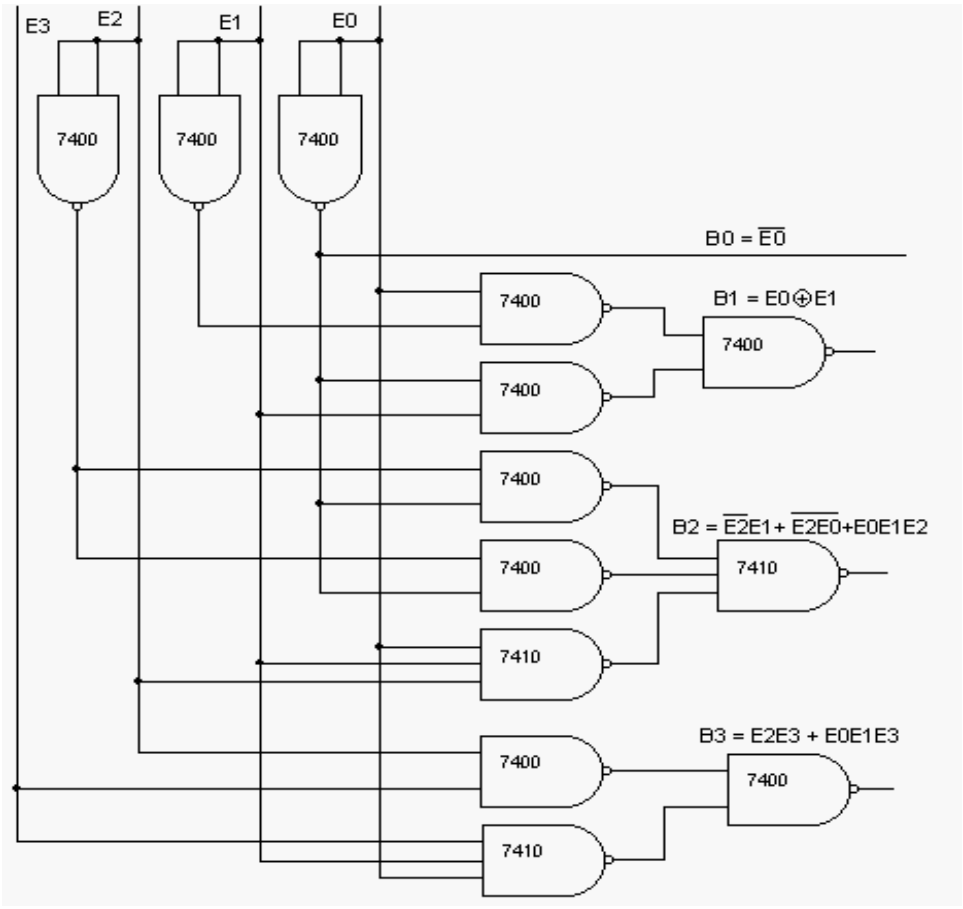
Implement BCD to excess-3 and excess-3 to BCD code converter using parallel adder IC 7483.

Excess-3 To BCD :-

Truth Table For Code Conversion: -

Inputs				Outputs			
E3	E2	E1	E0	B3 (v)	B2 (v)	B1 (v)	B0(v)
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

Circuit diagram using NAND gates only:

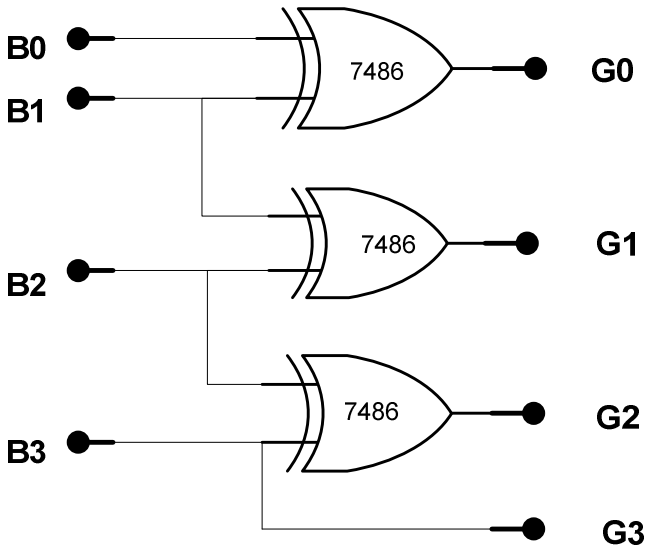


Binary to Gray:

Truth Table:

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

Circuit diagram using EX-OR Gates:



EXPERIMENT NO. 5: **DATE: / /**

BINARY-TO-GRAY AND GRAY-TO-BINARY CODE CONVERTER

AIM: - To convert given binary numbers to gray codes**.**

APPARATUS REQUIRED: -

IC Trainer Kit, patch chords, IC 7486, etc

PROCEDURE: -

1. The circuit connections are made as shown in fig.
2. Pin (14) is connected to +Vcc and Pin (7) to ground.
3. In the case of binary to gray conversion, the inputs B0, B1, B2 and B3 are given at respective pins and outputs G0, G1, G2, G3 are taken for all the 16 combinations of the input.
4. In the case of gray to binary conversion, the inputs G0, G1, G2 and G3 are given at respective pins and outputs B0, B1, B2, and B3 are taken for all the 16 combinations of inputs.
5. The values of the outputs are tabulated.

Exercise:

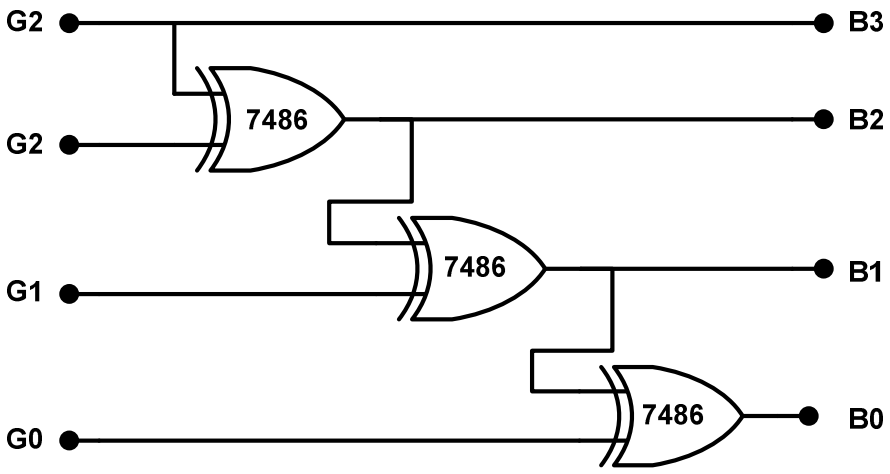
Implement binary to gray and gray to binary code converter using nand gates only.

Gray to binary:

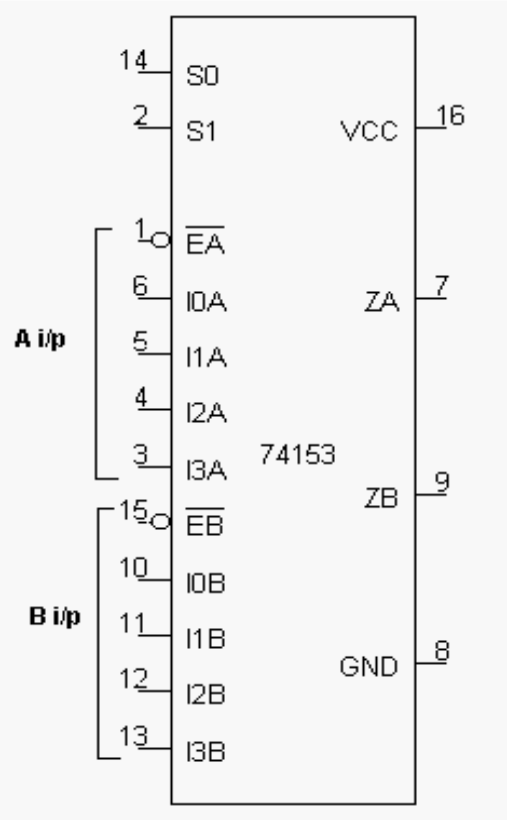
Truth-table:

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

Circuit Diagram using EX-OR Gates



Pin Details: -



Truth Table: -

CHANNEL – A							
INPUTS					SELECT LINES		O/P
Ēa	Ioa	I1a	I2a	I3a	S1	S2	Za(v)
1	X	X	X	X	X	X	0
0	0	X	X	X	0	0	0
0	1	X	X	X	0	0	1
0	X	0	X	X	0	1	0
0	X	1	X	X	0	1	1
0	X	X	0	X	1	0	0
0	X	X	1	X	1	0	1
0	X	X	X	0	1	1	0
0	X	X	X	1	1	1	1

CHANNEL – B							
INPUTS					SELECT LINES		O/P
Ēa	Iob	I1b	I2b	I3b	S1	S2	Za(v)
1	X	X	X	X	X	X	0
0	0	X	X	X	0	0	0
0	1	X	X	X	0	0	1
0	X	0	X	X	0	1	0
0	X	1	X	X	0	1	1
0	X	X	0	X	1	0	0
0	X	X	1	X	1	0	1
0	X	X	X	0	1	1	0
0	X	X	X	1	1	1	1

EXPERIMENT NO. 6: **DATE: / /**

MULTIPLEXER USING IC 74153

Aim: - To verify the truth table of multiplexer using 74153

To study the arithmetic circuits half-adder and half Subtractor, full adder and full Subtractor using multiplexer.

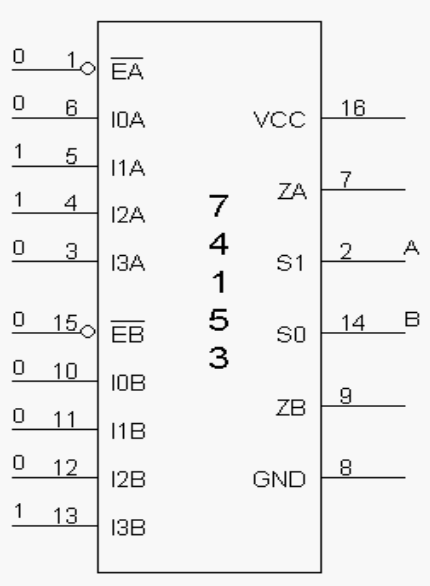
Apparatus Required: -

IC Trainer Kit, patch chords**,** IC 74153, IC 74139, IC 7404, etc.

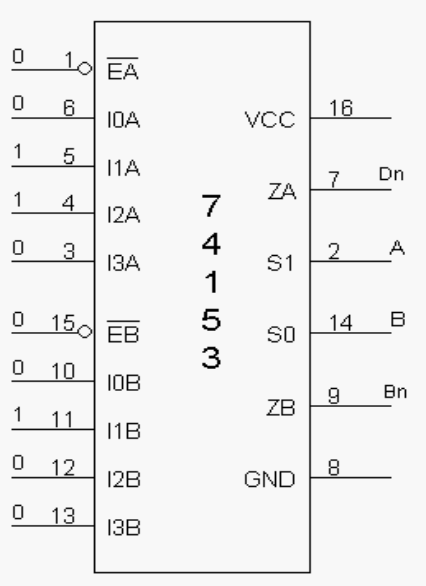
Procedure: -

1. The Pin [16] is connected to + Vcc.
2. Pin [8] is connected to ground.
3. The inputs are applied either to 'A' input or 'B' input.
4. If MUX 'A' has to be initialized, Ea is made low and if MUX 'B' has to be initialized, E_b is made low.
5. Based on the selection lines one of the inputs will be selected at the output and thus the truth table is verified.
6. In case of half adder using MUX, sum and carry is obtained by applying a constant inputs at I_{0a}, $I_{1a,}$ $I_{2a,}$ I_{3a} and $I_{0b,}$ $I_{1b,}$ I_{2b} and I_{3b} and the corresponding values of select lines are changed as per table and the output is taken at Z0a as sum and Z0b as carry.
7. In this case, the channels A and B are kept at constant inputs according to the table and the inputs A and B are varied. Making Ea and Eb zero and the output is taken at Za, and Zb.
8. In full adder using MUX, the input is applied at Cn-1, An and Bn. According to the table corresponding outputs are taken at Cn and Dn.

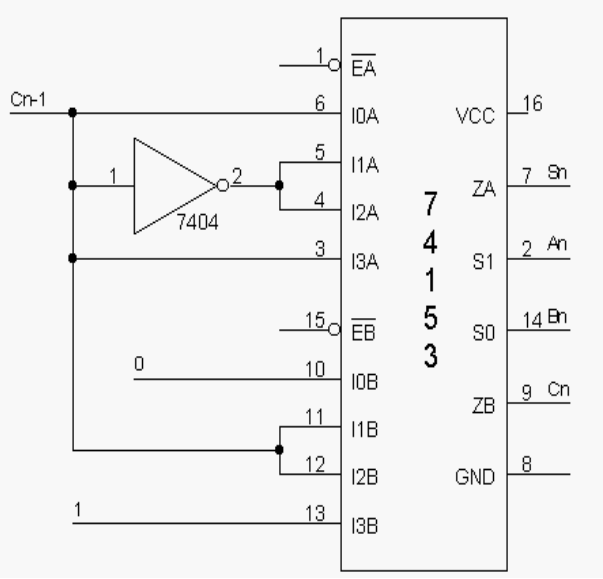
Half Adder Using 74153 –



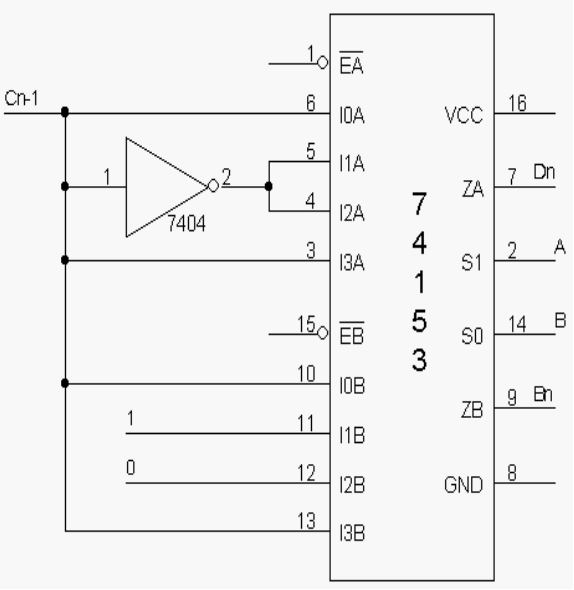
Half Subtractor Using 74153:



Full Adder Using 74153: -



Full Subtractor Using 74153: -



Truth Tables: -

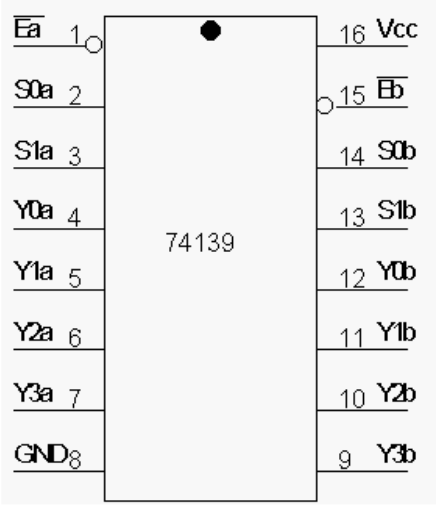
Half adder			
A	B	Sn (V)	Cn (V)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Full Adder				
An	Bn	Cn-1	Sn (V)	Cn (V)
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

Half subtractor			
A	B	Dn (V)	Bn (V)
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Full subtractror				
An	Bn	Cn-1	Dn (V)	Bn (V)
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

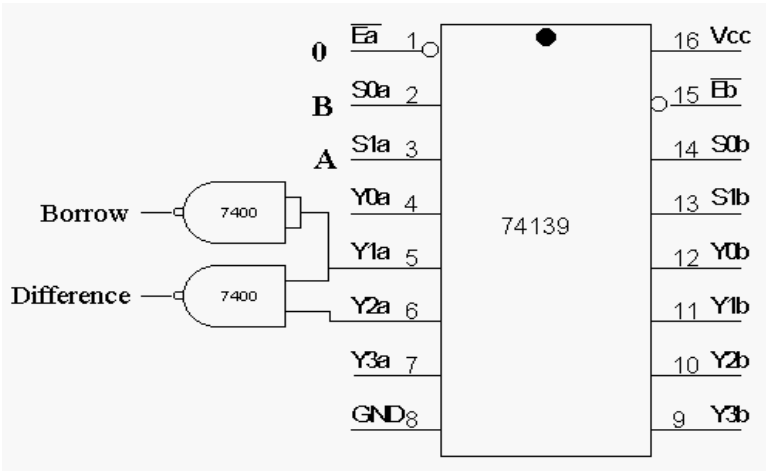
Pin details:



Truth Table For Demux: -

CHANNEL – A								CHANNEL – B						
Inputs			Outputs					Inputs			Outputs			
Ēa	S1a	S0a	Y0a	Y1a	Y2a	Y3a		Ēb	S1b	S0b	Y0b	Y1b	Y2b	Y3b
1	X	X	1	1	1	1		1	X	X	1	1	1	1
0	0	0	0	1	1	1		0	0	0	0	1	1	1
0	0	1	1	0	1	1		0	0	1	1	0	1	1
0	1	0	1	1	0	1		0	1	0	1	1	0	1
0	1	1	1	1	1	0		0	1	1	1	1	1	0

Half subtractor:-



EXPERIMENT NO. 7: **DATE: / /**

DE-MULTIPLEXER USING IC 74139

Aim: - To verify the truth table of de-multiplexer using 74139.
To study the arithmetic circuits half-adder and half Subtractor, full adder and full Subtractor using de- multiplexer.

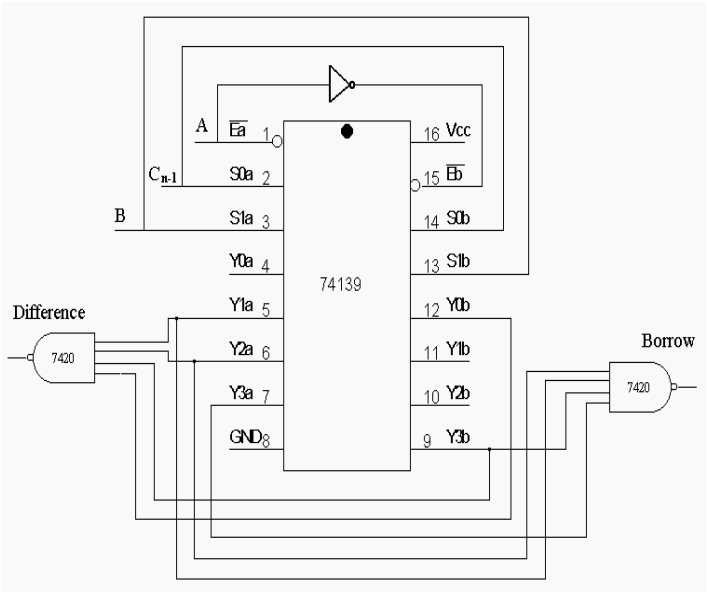
Apparatus Required: -

IC Trainer Kit, patch chords**,** IC 74153, IC 74139, IC 7404, etc

Procedure: -

1. The inputs are applied to either 'a' input or 'b' input
2. The demux is activated by making Ea low and Eb low.
3. The truth table is verified.

Full subtractor using IC 74139:-

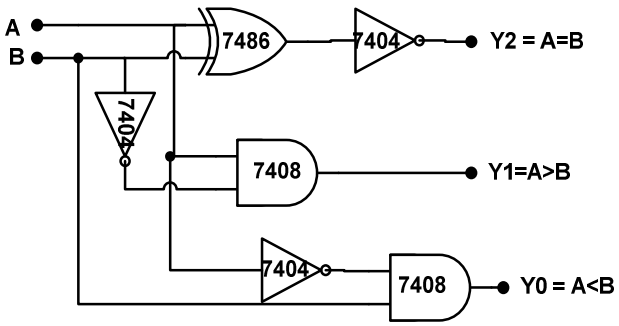


Truth Table

Half Subtractor			
A	**B**	**Dn (V)**	**Bn (V)**
0	**0**	**0**	**0**
0	**1**	**1**	**1**
1	**0**	**1**	**0**
1	**1**	**0**	**0**

Full Subtractor				
An	**Bn**	**Cn-1**	**Dn (V)**	**Bn (V)**
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**

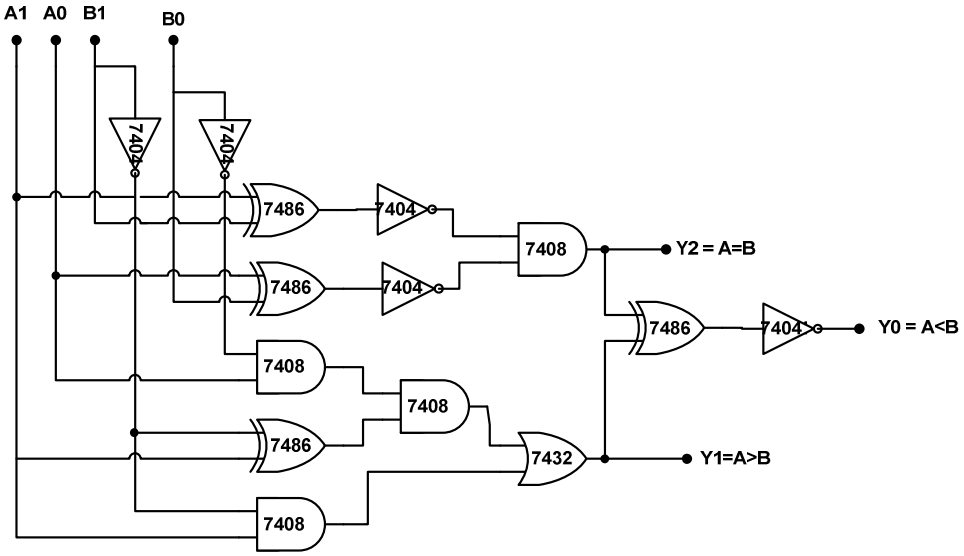
One Bit Comparator: -



Truth-table

A	B	Y0 (A<B)	Y1 (A>B)	Y2 (A=B)
0	0	0	0	1
0	1	1	0	0
1	0	0	1	0
1	1	0	0	1

Two Bit Comparator: -



A1	A0	B1	B0	Y2(A=B)	Y1(A>B)	Y0(A<B)
0	0	0	0	1	0	0
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	0	1
0	1	0	0	0	1	0
0	1	0	1	1	0	0
0	1	1	0	0	0	1
0	1	1	1	0	0	1
1	0	0	0	0	1	0
1	0	0	1	0	1	0
1	0	1	0	1	0	0
1	0	1	1	0	0	1
1	1	0	0	0	1	0
1	1	0	1	0	1	0
1	1	1	0	0	1	0
1	1	1	1	1	0	1

Experiment No:8

Date: __/__/____

COMPARATORS

Aim: - To verify the truth table of one bit and two bit comparators using logic gates and four bit and eight bit comparators using IC 7485.

Apparatus Required: -

IC 7485, IC 7404, IC 7408, etc.

Procedure: -

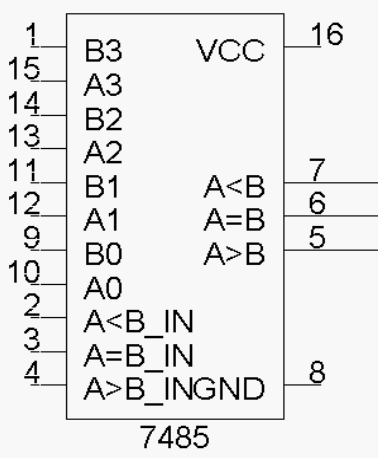
1. **Verify the gates.**
2. **Make the connections as per the circuit diagram.**
3. **Switch on Vcc.**
4. **Applying i/p and Check for the outputs.**
5. **The voltameter readings of outputs are taken and tabulated in tabular column.**
6. **The o/p are verified.**

Exercise:

- **Implement one bit, two bit comparator circuit using nand gates only.**
- **Write the function table for 8 bit comparator .**

4-bit Comparator

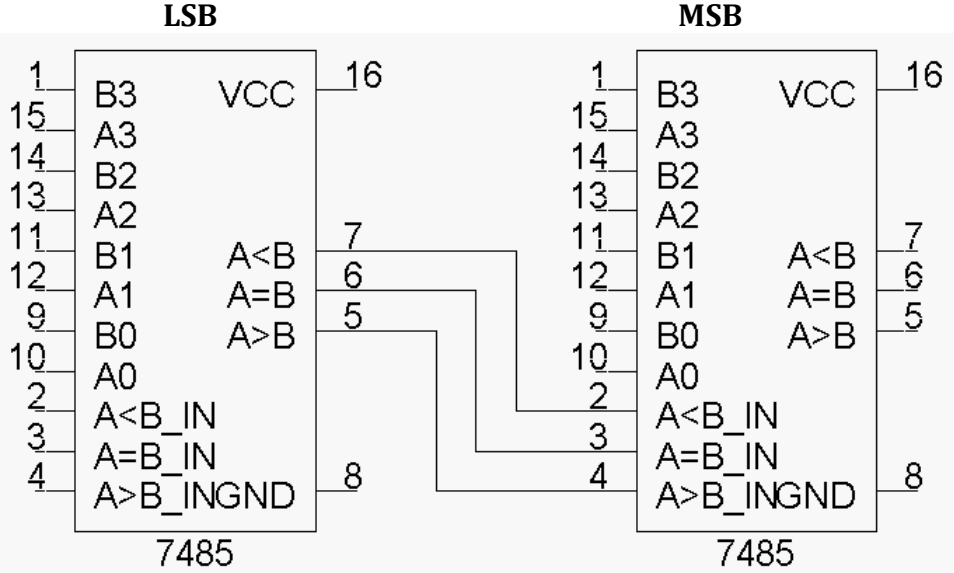
LOGIC DIAGRAM:



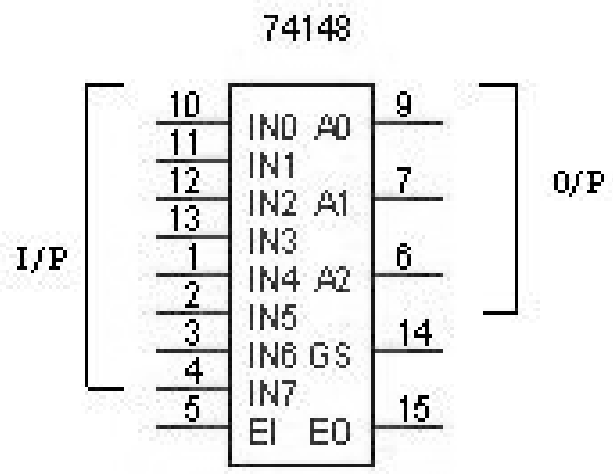
FUNCTION TABLE :

Comparing I/Ps	Cascading I/Ps			Outputs		
	A>B	A=B	A<B	A>B	A=B	A<B
A>B	X	X	X	1	0	0
A=B	1	0	0	1	0	0
	X	1	X	0	1	0
	0	0	1	0	0	1
	0	0	0	1	0	0
	1	0	1	0	0	0
A<B	X	X	X	0	0	1

8-Bit Comparator: -



PIN DETAILS:-



FUNCTION TABLE:

INPUTS									OUTPUTS				
EI	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**A2**	**A1**	**A0**	**GS**	**E0**
H	X	X	X	X	X	X	X	X	H	H	H	H	H
L	H	H	H	H	H	H	H	H	H	H	H	H	L
L	X	X	X	X	X	X	X	L	L	L	L	L	H
L	X	X	X	X	X	X	L	H	L	L	H	L	H
L	X	X	X	X	X	L	H	H	L	H	L	L	H
L	X	X	X	X	L	H	H	H	L	H	H	L	H
L	X	X	X	L	H	H	H	H	H	L	L	L	H
L	X	X	L	H	H	H	H	H	H	L	H	L	H
L	X	L	H	H	H	H	H	H	H	H	L	L	H
L	L	H	H	H	H	H	H	H	H	H	H	L	H

Experiment No: 9 **DATE: __/__/____**

ENCODER & DECODER

AIM:-To convert a given octal input to the binary output and to study the LED display using 7447 7-segment decoder/ driver.

APPARATUS REQUIRED: -

IC 74148, IC 7447, 7-segment display, etc.

PROCEDURE: - (Priority Encoder)

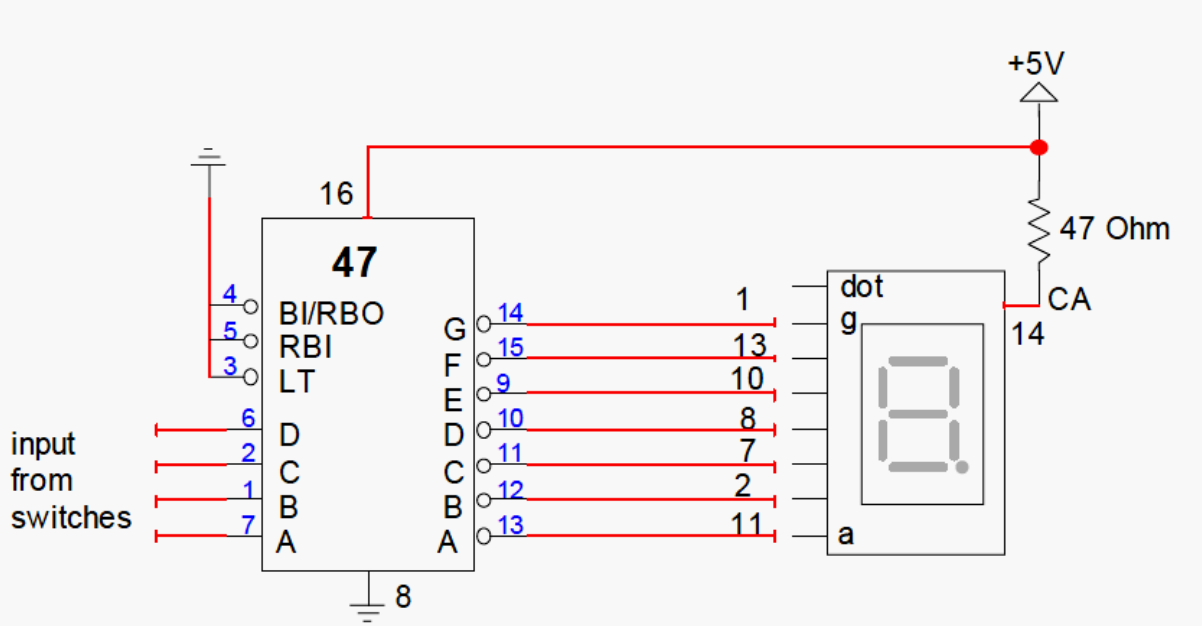
1. **Connections are made as per circuit diagram.**
2. **The octal inputs are given at the corresponding pins.**
3. **The outputs are verified at the corresponding output pins.**

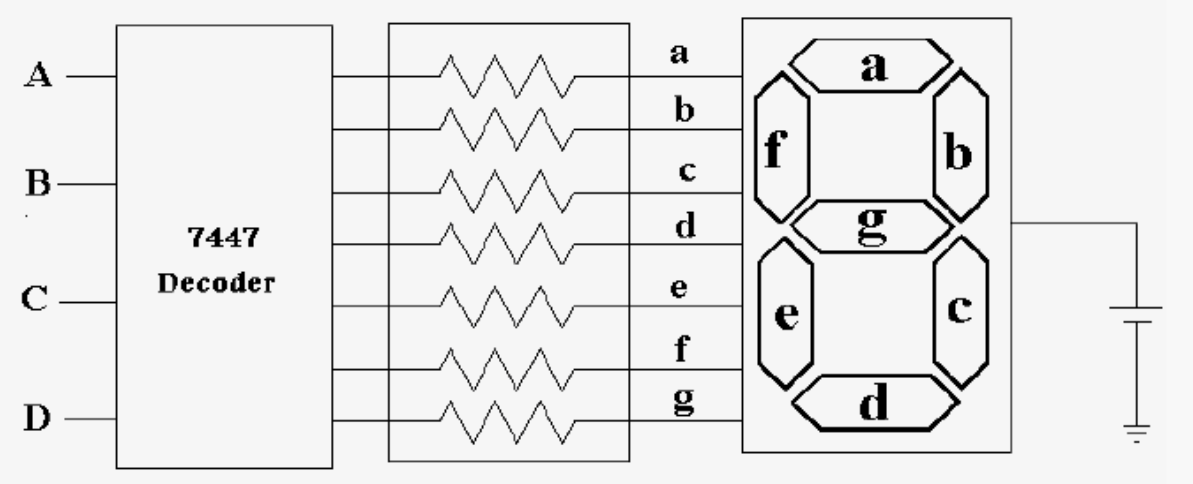
PROCEDURE: - (Decoder)

1. **Connections are made as per the circuit diagram.**
2. **Connect the pins of IC 7447 to the respective pins of the LED display board.**
3. **Give different combinations of the inputs and observe the decimal numbers displayed on the board.**

Exercise: Implement the following expression using decoder and logic gates.

BCD TO SEVEN SEGMENT DECODER CIRCUIT DIAGRAM WITH LED DISPLAY

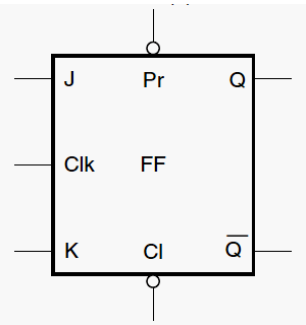


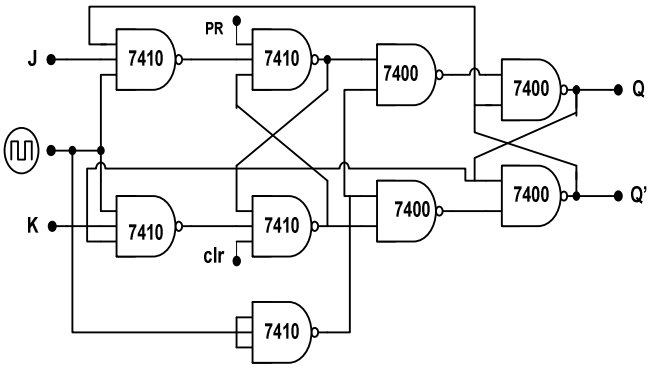


TRUTH-TABLE:

DECIMAL DIGIT	INPUTS				LED DISPLAY VALUE	SEVEN SEGMENTS LEVEL						
	D	C	B	A		a'	b'	c'	d'	e'	f'	g'
0	0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	1	0	1	1	0	0	0	0
2	0	0	1	0	2	1	1	0	1	1	0	1
3	0	0	1	1	3	1	1	1	1	0	0	1
4	0	1	0	0	4	0	1	1	0	0	1	1
5	0	1	0	1	5	1	0	1	1	0	1	1
6	0	1	1	0	6	0	0	1	1	1	1	1
7	0	1	1	1	7	1	1	1	0	0	0	0
8	1	0	0	0	8	1	1	1	1	1	1	1
9	1	0	0	1	9	1	1	1	0	0	1	1
10	1	0	1	0								
11	1	0	1	1								
12	1	1	0	0								
13	1	1	0	1								
14	1	1	1	0								
15	1	1	1	1	No seg. glows							

Circuit Diagram: - (Master Slave JK Flip-Flop)

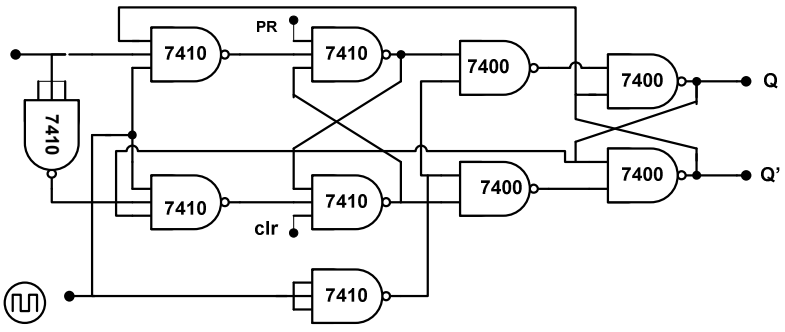




Truth Table:- (Master Slave JK Flip-Flop)

Preset	Clear	J	K	Clock	Qn+1	$\overline{Qn+1}$	
0	1	X	X	X	1	0	Set
1	0	X	X	X	0	1	Reset
1	1	0	0	⎍	Qn	$\overline{Qn}$	No Change
1	1	0	1	⎍	0	1	Reset
1	1	1	0	⎍	1	0	Set
1	1	1	1	⎍	$\overline{Qn}$	Qn	Toggle

D Flip-Flop:-



Truth Table:-

Preset	Clear	D	Clock	Qn+1	$\overline{Qn+1}$
1	1	0	⎍	0	1
1	1	1	⎍	1	0

Date: __/__/____

Experiment No: 10

FLIP-FLOPS

Aim: Verification of truth-tables of the following types of flip-flops using NAND gates and using IC 7476:

i. **JK Master Slave FF**
ii. **D-FF**
iii. **T-FF**
iv. **SR-FF**

Apparatus Required: -

IC 7410, IC 7400, IC 7476, etc.

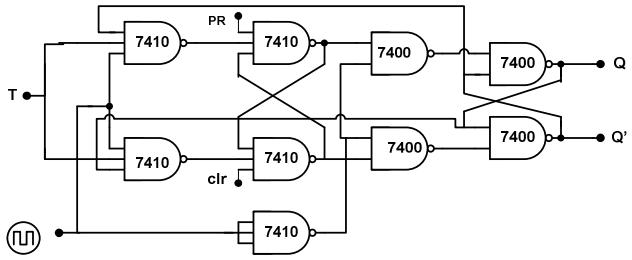
Procedure: -

1. **Connections are made as per circuit diagram.**
2. **The truth table is verified for various combinations of inputs.**

Exercise:-

- **Write the timing diagrams for all the above Flip-Flops**

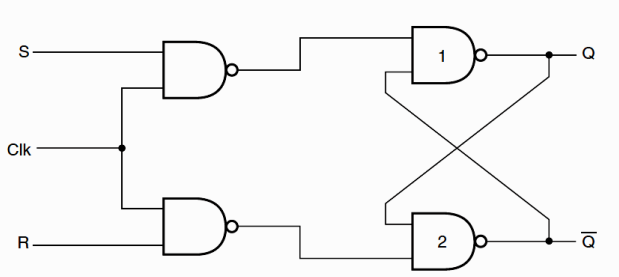
T Flip-Flop:-



Truth Table:-

Preset	Clear	T	Clock	Qn+1	$\overline{Qn+1}$
1	1	0	⎍	Qn	$\overline{Qn}$
1	1	1	⎍	$\overline{Qn}$	Qn

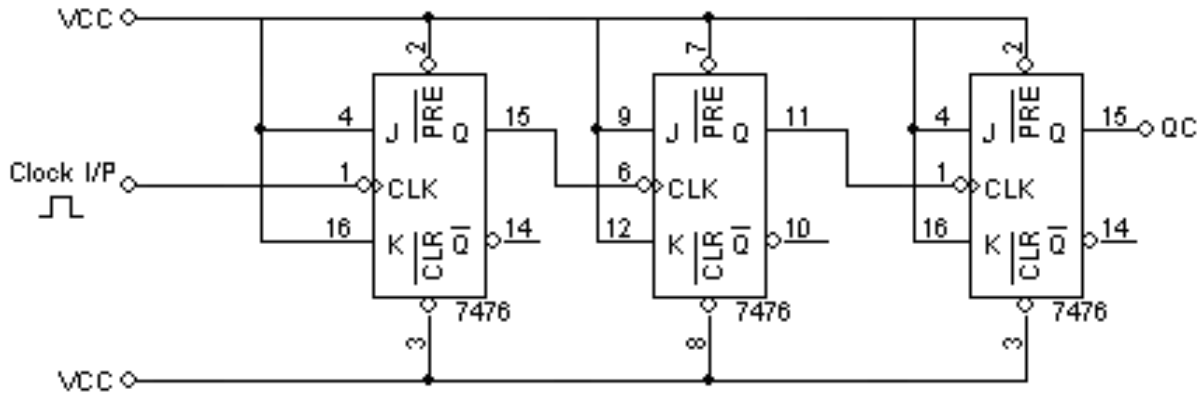
SR Flip-flop:



Truth-table:

S	R	Clk	Q_{n+1}
0	0	0	Q_n
0	0	1	Q_n
0	1	0	Q_n
0	1	1	0
1	0	0	Q_n
1	0	1	1
1	1	0	Q_n
1	1	1	Invalid

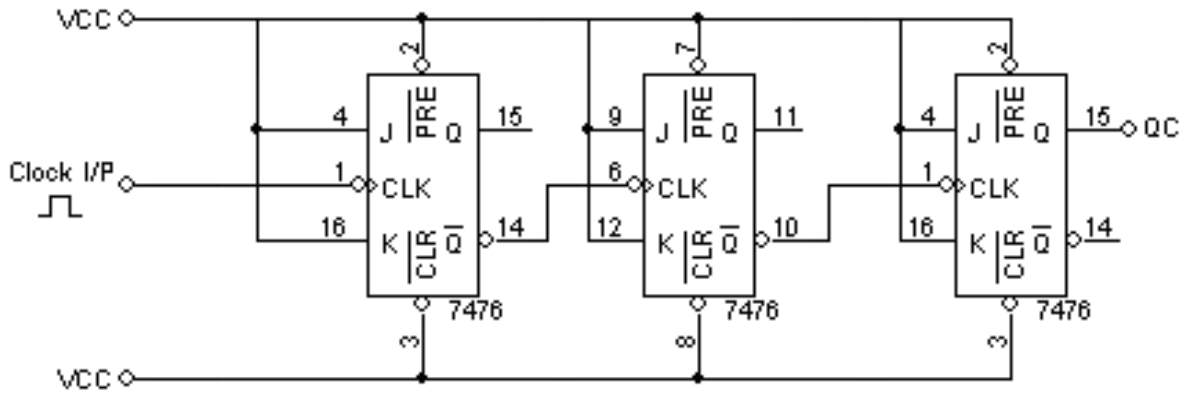
Circuit Diagram: - 3-Bit Asynchronous Up Counter



Truth-table:

3-bit Asynchronous up counter			
Clock	**QC**	**QB**	**QA**
0	**0**	**0**	**0**
1	**0**	**0**	**1**
2	**0**	**1**	**0**
3	**0**	**1**	**1**
4	**1**	**0**	**0**
5	**1**	**0**	**1**
6	**1**	**1**	**0**
7	**1**	**1**	**1**
8	**0**	**0**	**0**
9	**0**	**0**	**1**

Circuit Diagram: - 3-Bit Asynchronous Down Counter



Experiment No: 11 **Date: __/__/____**

ASYNCHRONOUS COUNTERS

Aim: Realization of 3-bit asynchronous up/down counters and design of Mod-N counter design.

Apparatus Required: -

IC 7408, IC 7476, IC 7400, IC 7432 etc.

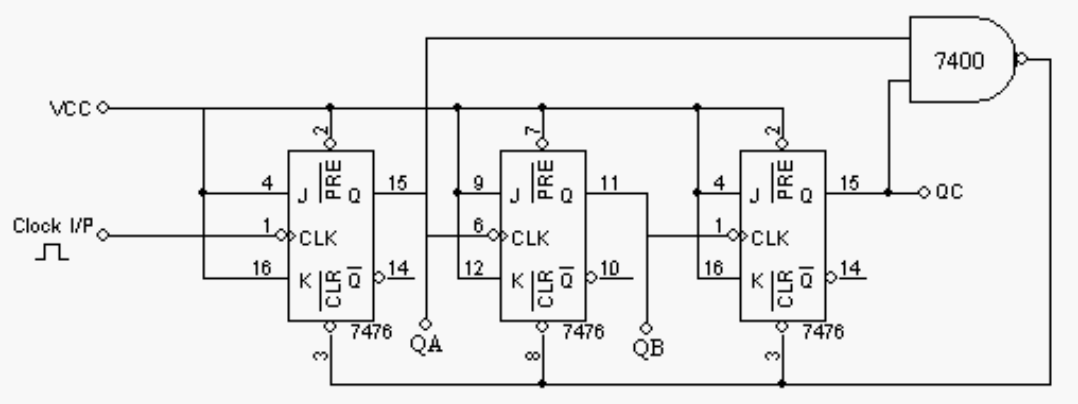
Procedure: -

1. **Connections are made as per circuit diagram.**
2. **Clock pulses are applied one by one at the clock I/P and the O/P is observed at QA, QB & QC for IC 7476.**
3. **Truth table is verified.**

Truth-table:

3-bit Asynchronous down counter			
Clock	**QC**	**QB**	**QA**
0	**1**	**1**	**1**
1	**1**	**1**	**0**
2	**1**	**0**	**1**
3	**1**	**0**	**0**
4	**0**	**1**	**1**
5	**0**	**1**	**0**
6	**0**	**0**	**1**
7	**0**	**0**	**0**
8	**1**	**1**	**1**
9	**1**	**1**	**0**

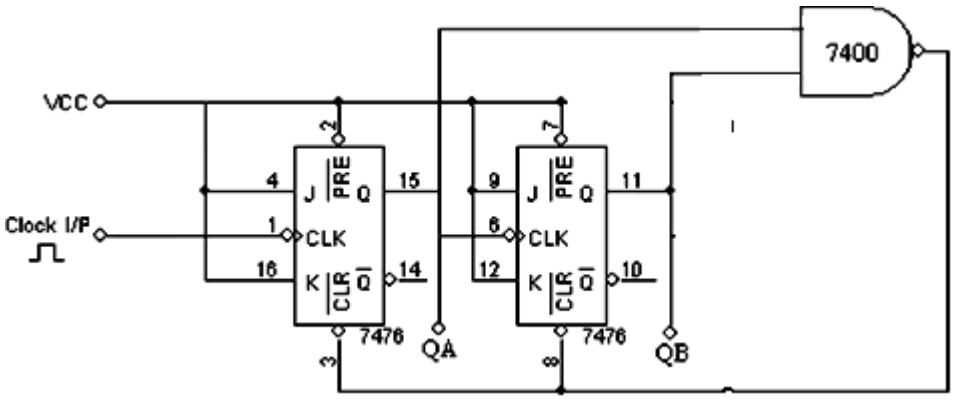
Mod 5 Asynchronous up Counter:-



Truth-table:

Mod 5 Asynchronous counter			
Clock	**QC**	**QB**	**QA**
0	**0**	**0**	**0**
1	**0**	**0**	**1**
2	**0**	**1**	**0**
3	**0**	**1**	**1**
4	**1**	**0**	**0**
5	**0**	**0**	**0**

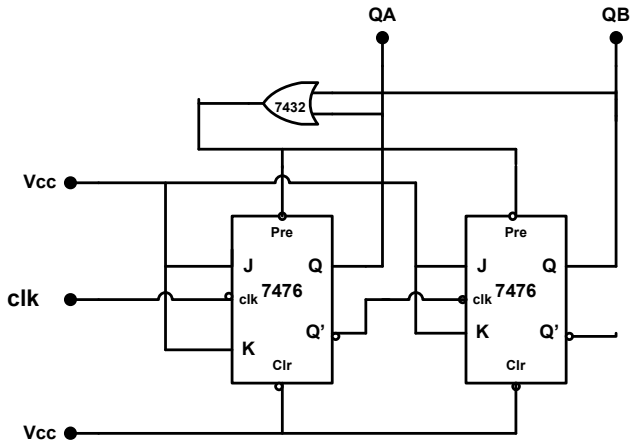
Mod 3 Asynchronous up Counter:-



Truth-table:

Clock	QB	QA
0	0	0
1	0	1
2	1	0
3	0	0

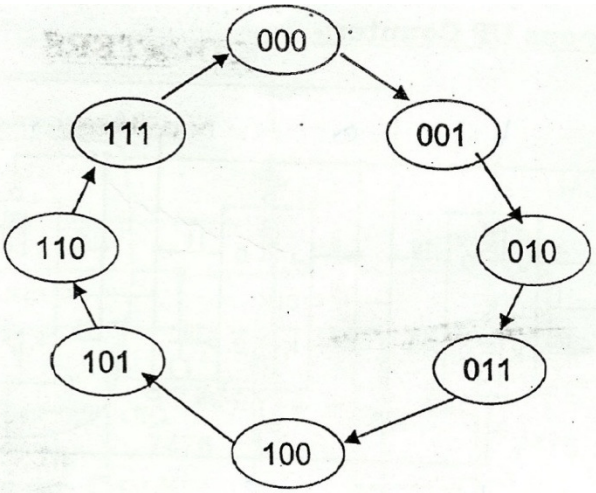
Mod 4 Asynchronous down Counter:-



Truth-table:

Clock	QB	QA
0	1	1
1	1	0
2	0	1
3	0	0

State Diagram

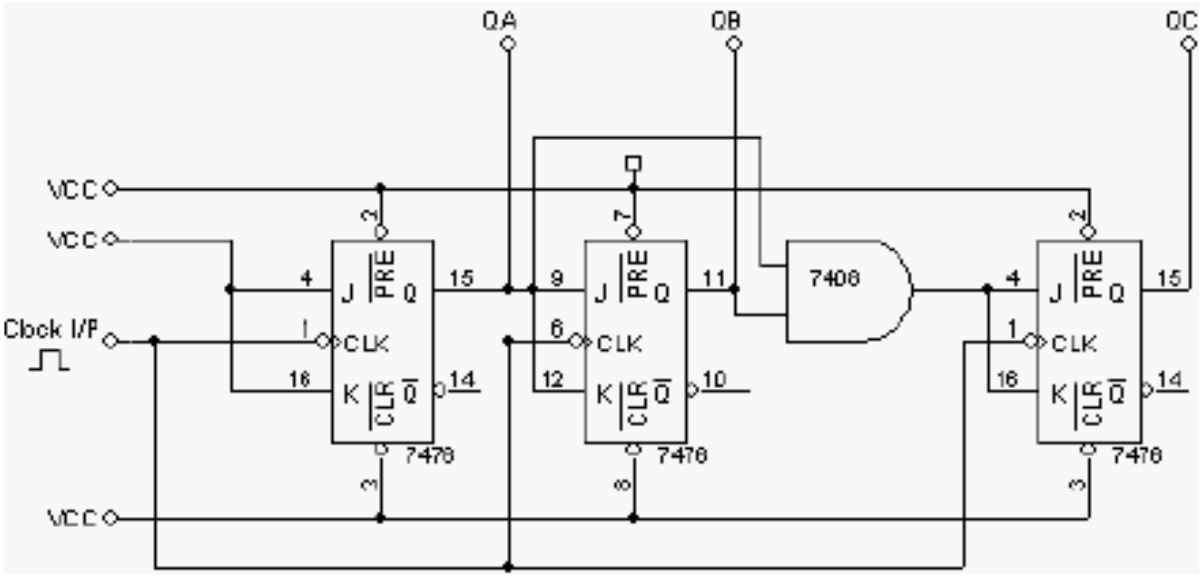


Excitation Table:

State Change		J-K Input	
From	To	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

Transition Table:

Present State			Next State		
QC	QB	QA	QC	QB	QA
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



Experiment No: 12 **Date: __/__/____**

SYNCHRONOUS COUNTERS

Aim: **Design and Realization of 3-bit synchronous counters.**

Apparatus Required: -

IC 7408, IC 7476, IC 7400, IC 7432, etc.

Design steps :

1. **Write the truth-table or state diagram.**
2. **From truth-table/state-diagram, derive the state transition table.**
3. **Decide the no. and type of flip-flops to be used.**
4. **Using the corresponding excitation tables, derive the input and output equations and simplify using K-maps.**
5. **Using the derived simplified input and output expressions with flip-flops, draw the circuit diagram.**

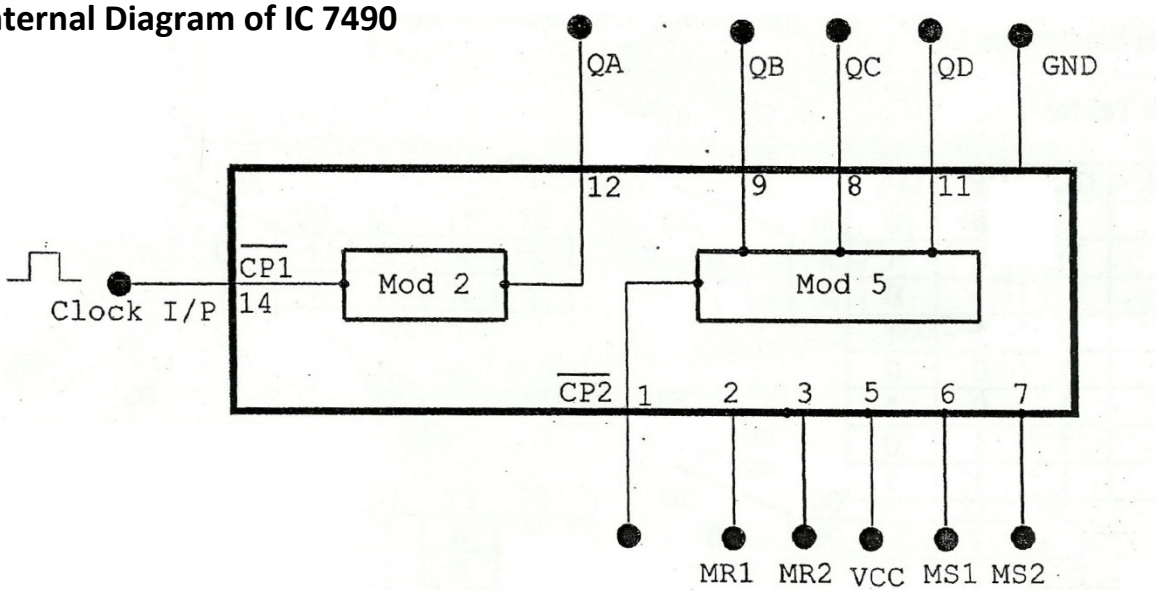
Procedure: -

1. **Connections are made as per circuit diagram.**
2. **Clock pulses are applied simultaneously at the clock I/Ps of all 7476 ICs used in the circuit and the O/P is observed at the outputs of ICs 7476.**
3. **Truth table is verified.**

Truth Table:

QC	QB	QA	JC	KC	JB	KB	JA	KA
0	0	0	0	X	0	X	1	X
0	0	1	0	X	1	X	X	1
0	1	0	0	X	X	0	1	X
0	1	1	1	X	X	1	X	1
1	0	0	X	0	0	X	1	X
1	0	1	X	0	1	X	X	1
1	1	0	X	0	X	0	1	X
1	1	1	X	1	X	1	X	1

Internal Diagram of IC 7490



MODE SELECTION — FUNCTION TABLE

RESET/SET INPUTS				OUTPUTS			
MR_1	MR_2	MS_1	MS_2	Q_0	Q_1	Q_2	Q_3
H	H	L	X	L	L	L	L
H	H	X	L	L	L	L	L
X	X	H	H	H	L	L	H
L	X	L	X	Count			
X	L	X	L	Count			
L	X	X	L	Count			
H	L	L	X	Count			

H = HIGH voltage level
L = LOW voltage level
X = Don't care

BCD COUNT SEQUENCE — FUNCTION TABLE

COUNT	OUTPUTS			
	Q_0	Q_1	Q_2	Q_3
0	L	L	L	L
1	H	L	L	L
2	L	H	L	L
3	H	H	L	L
4	L	L	H	L
5	H	L	H	L
6	L	H	H	L
7	H	H	H	L
8	L	L	L	H
9	H	L	L	H

NOTE:
Output Q_0 connected to input $\overline{CP}_1$.

Experiment No: 13 **Date: __/__/____**

DECADE AND BINARY COUNTERS

Aim: Realization of decade and binary counters.

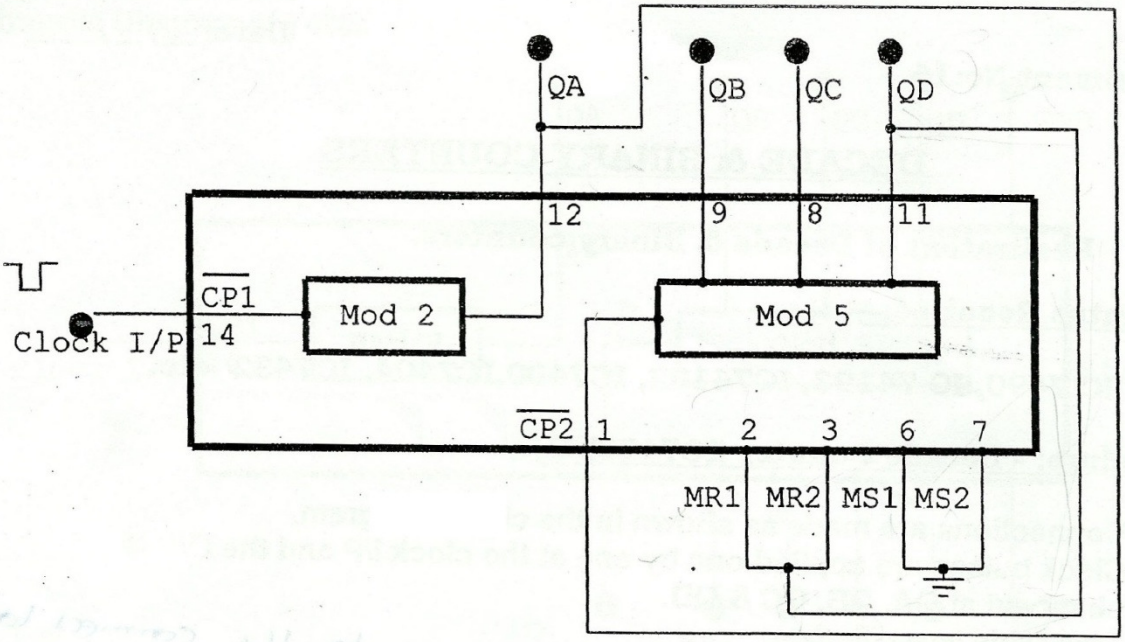
Apparatus Required: -

IC 7408, IC 7476, IC 7400, IC 7432, IC 7490, IC 74192, IC 74193 etc.

Procedure (IC 74192, IC 74193):-

1. **Connections are made as per the circuit diagram except the connection from output of NAND gate to the load input.**
2. **The data (0011) = 3 is made available at the data i/ps A, B, C & D respectively.**
3. **The load pin made low so that the data 0011 appears at QD, QC, QB & QA respectively.**
4. **Now connect the output of the NAND gate to the load input.**
5. **Clock pulses are applied to "count up" pin and the truth table is verified.**
6. **Now apply (1100) = 12 for 12 to 5 counter and remaining is same as for 3 to 8 counter.**

Mod 8 Counter using 7490:



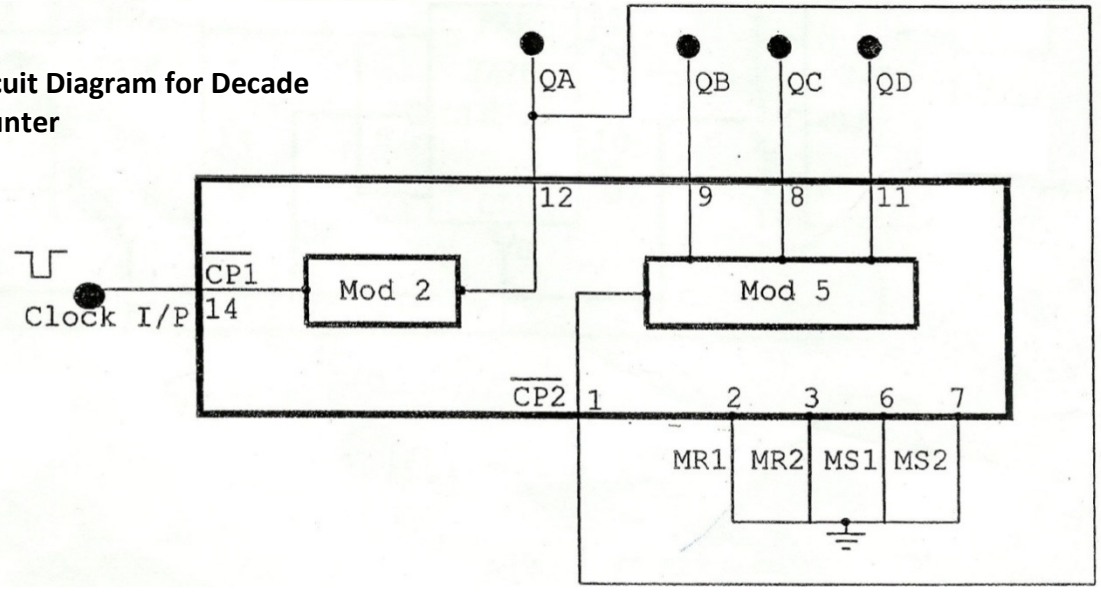
Truth Table: Decade Counter:

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

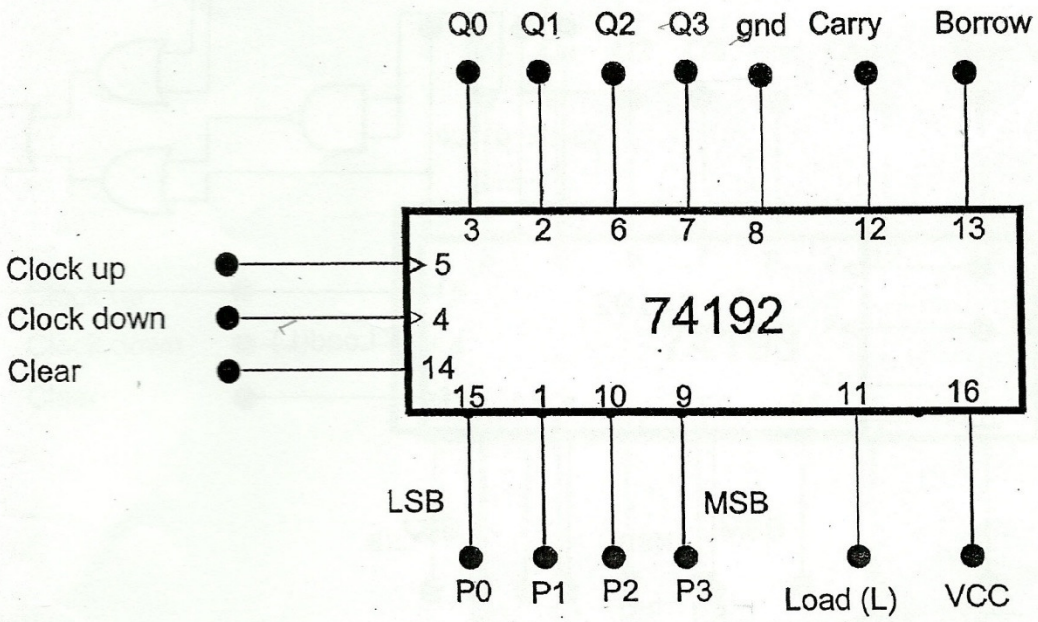
Mod 8 Counter:

Clock	QD	QC	QB	QA
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	0	0	0	0

Circuit Diagram for Decade Counter



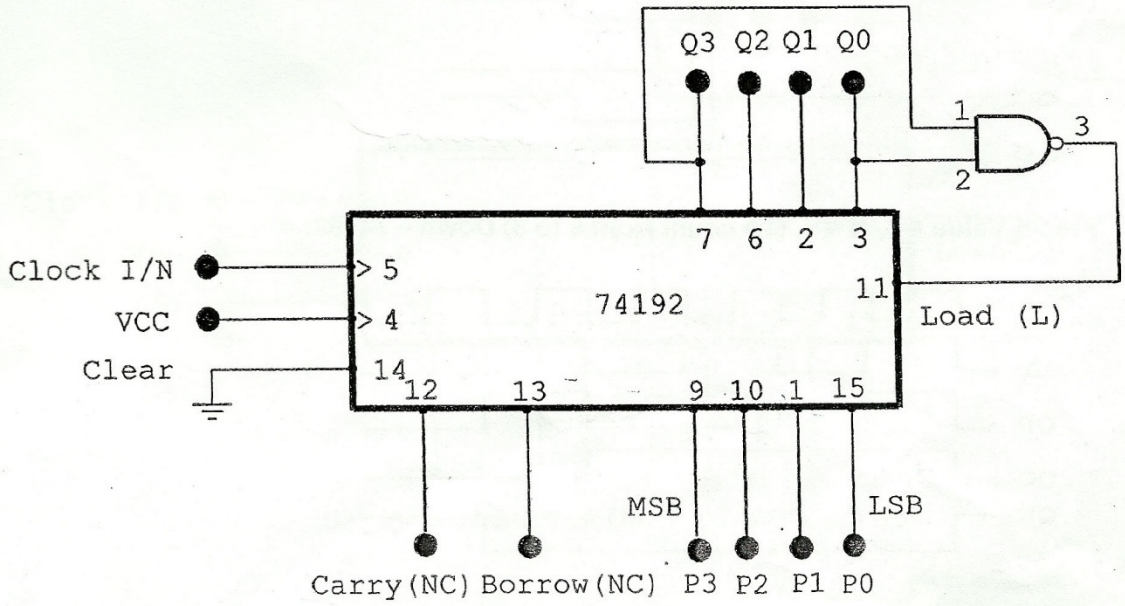
Logic Diagram of 74192 IC:



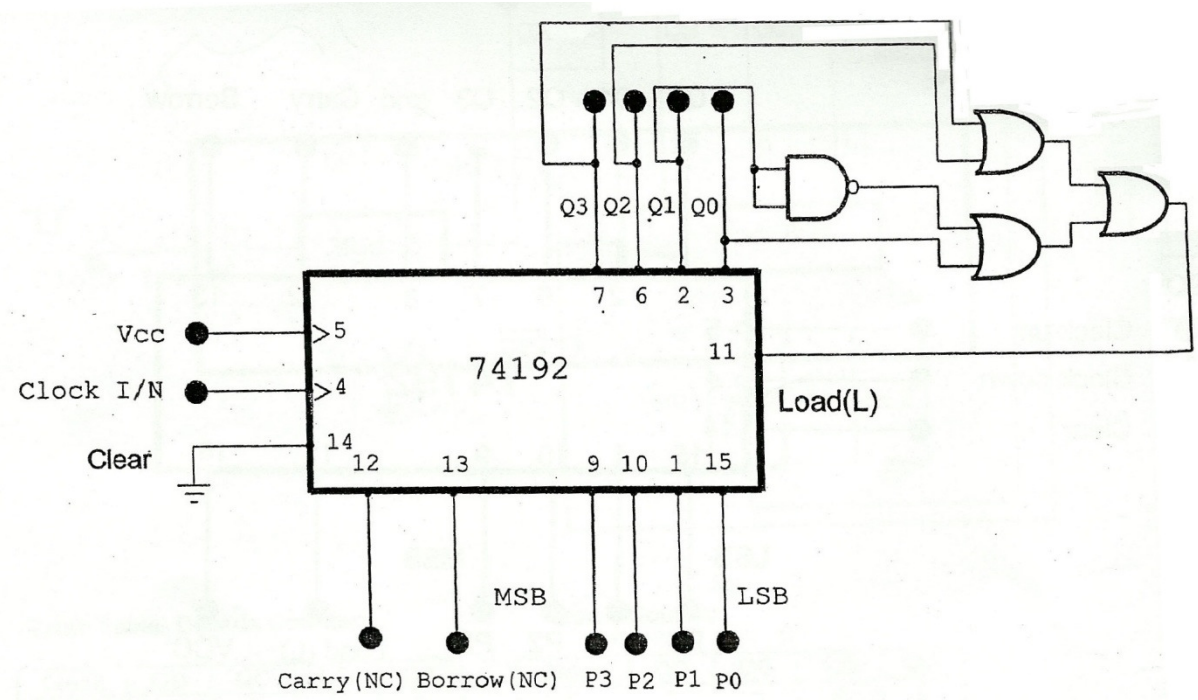
Function Table:

Load	Clear	Clk-up	Clk-down	Mode
X	1	X	X	Reset to zero
1	0	↑	1	Up-count
1	0	1	↑	Down-count
0	0	X	X	Preset
1	0	1	1	Stop count

Circuit diagram for Preset value = 5, N = 4, (To count from 5 to 8)



Circuit diagram for Preset value=8, N=6 (To count from 8 to 3)



Truth Table : From 5 to 8 :

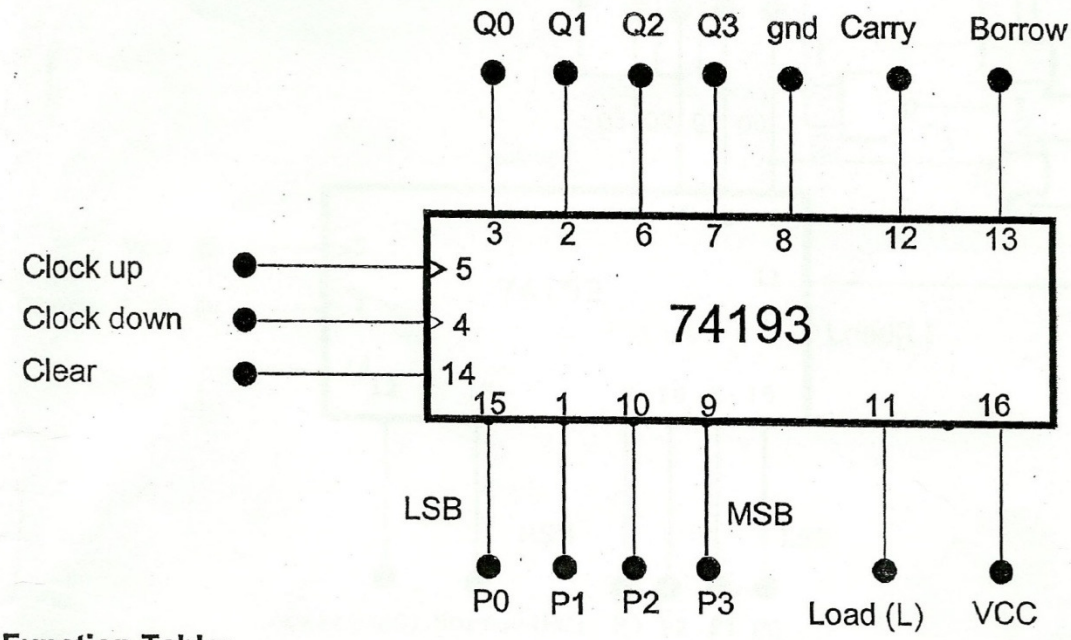
Clock	QD	QC	QB	QA
1	0	1	0	1
2	0	1	1	0
3	0	1	1	1
4	1	0	0	0
5	0	1	0	1

From 8 to 3 :

Clock	QD	QC	QB	QA
1	1	0	0	0
2	0	1	1	1
3	0	1	1	0
4	0	1	0	1
5	0	1	0	0
6	0	0	1	1
7	1	0	0	0

BINARY COUNTER:

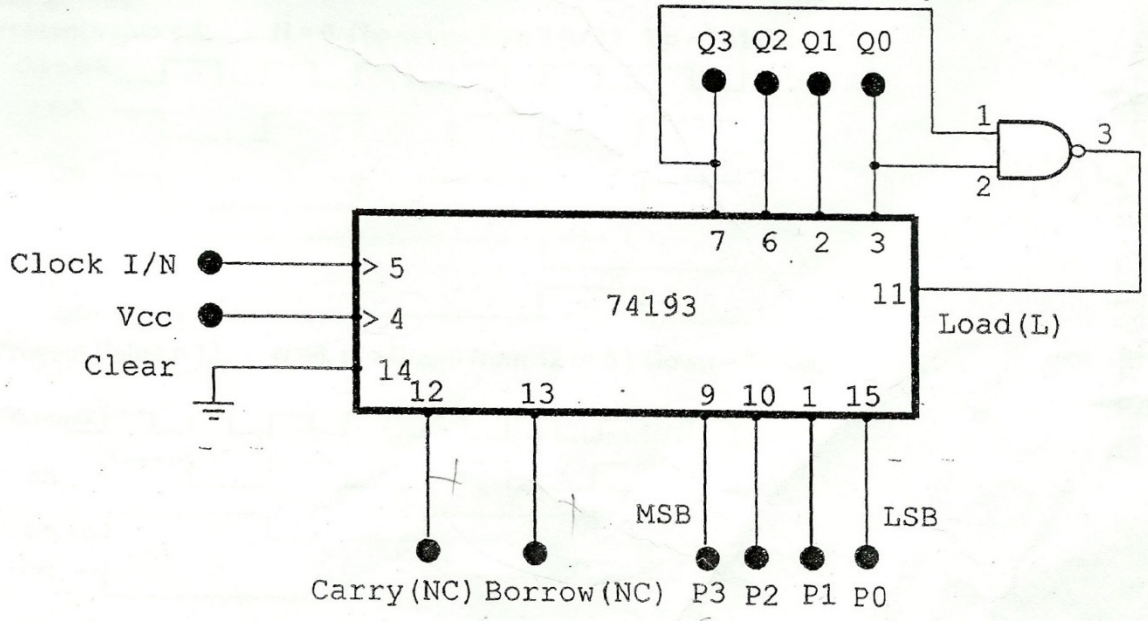
Logic Diagram of 74193 IC:



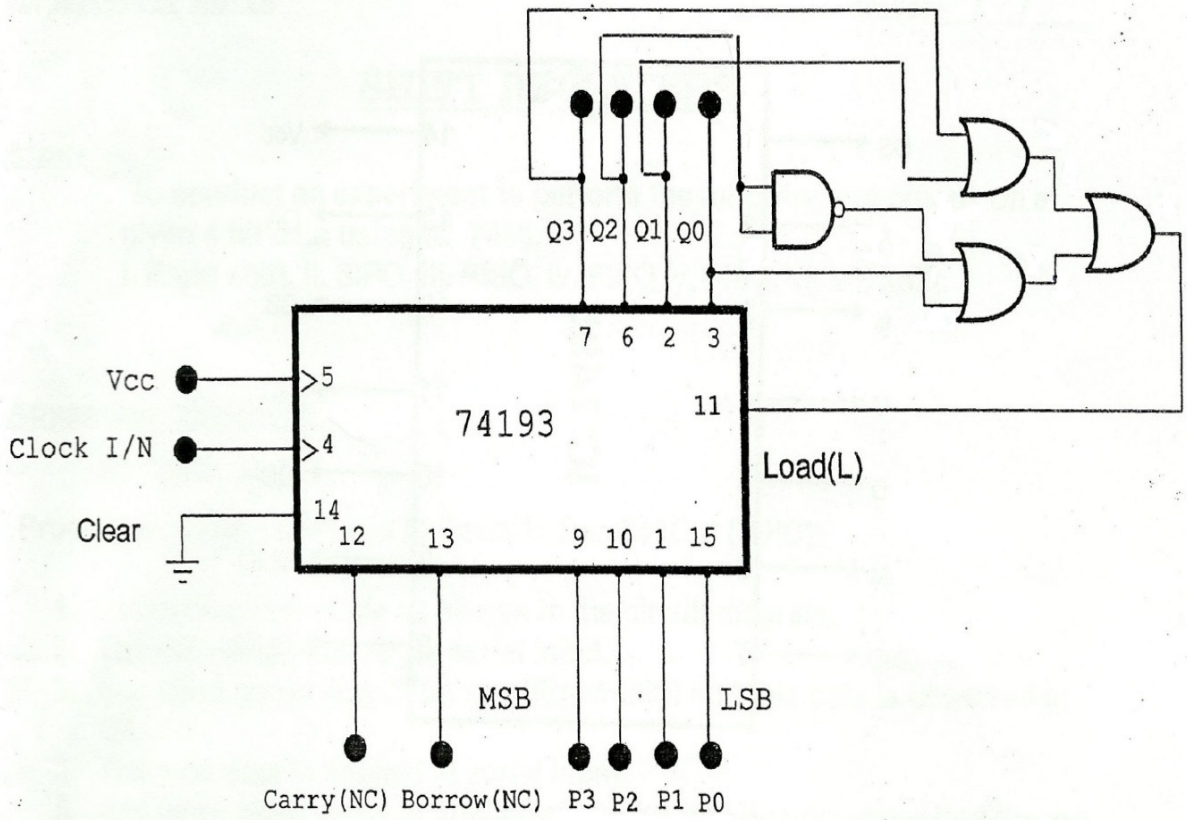
Function Table:

Load	Clear	Clk-up	Clk-down	Mode
X	1	X	X	Reset to zero
1	0	↑	1	Up-count
1	0	1	↑	Down-count
0	0	X	X	Preset
1	0	1	1	Stop count

Circuit Diagram for Preset value =3, N = 6,(To count from 3 to 8):



Circuit Diagram for Preset value =12, N = 8,(To count from 12 to 5):



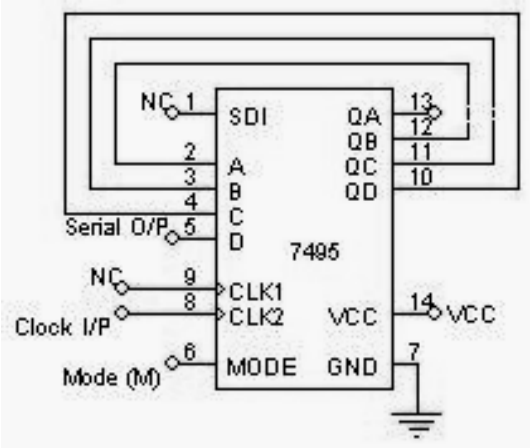
Truth Table: From 3 to 8

Clock	QD	QC	QB	QA
0	0	0	1	1
1	0	1	0	0
2	0	1	0	1
3	0	1	1	0
4	0	1	1	1
5	1	0	0	0
6	0	0	1	1

from 12 to 5

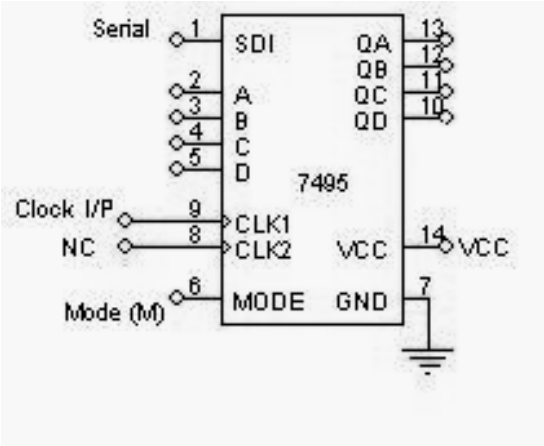
Clock	QD	QC	QB	QA
0	1	1	0	0
1	1	0	1	1
2	1	0	1	0
3	1	0	0	1
4	1	0	0	0
5	0	1	1	1
6	0	1	1	0
7	0	1	0	1
8	1	1	0	0

Circuit Diagram: - Shift Left



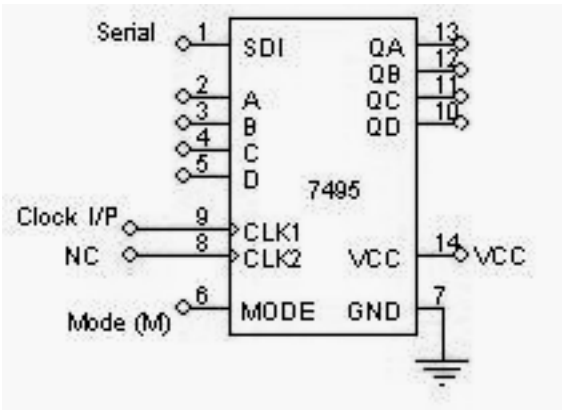
Clock	Serial i/p	QA	QB	QC	QD
1	1	X	X	X	1
2	0	X	X	1	0
3	1	X	1	0	1
4	1	1	0	1	1

SIPO (Right Shift):-



Clock	Serial i/p	QA	QB	QC	QD
1	0	0	X	X	X
2	1	1	0	X	X
3	1	1	1	0	X
4	1	1	1	1	0

SISO:-



Clock	Serial i/p	QA	QB	QC	QD
1	do=0	0	X	X	X
2	d1=1	1	0	X	X
3	d2=1	1	1	0	X
4	d3=1	1	1	1	0=do
5	X	X	1	1	1=d1
6	X	X	X	1	1=d2
7	X	X	X	X	1=d3

Experiment No: 14 **Date: __/__/____**

SHIFT REGISTERS

Aim:- **Realization of 3-bit counters as a sequential circuit and Mod-N counter design (7476, 7490, 74192, 74193).**

Apparatus Required: -

IC 7495, etc.

Procedure: -

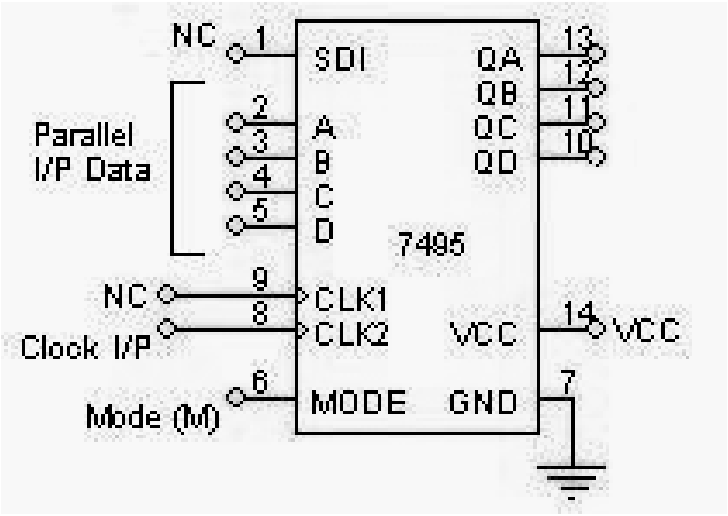
Serial In Parallel Out:-

5. **Connections are made as per circuit diagram.**
6. **Apply the data at serial i/p**
7. **Apply one clock pulse at clock 1 (Right Shift) observe this data at QA.**
8. **Apply the next data at serial i/p.**
9. **Apply one clock pulse at clock 2, observe that the data on QA will shift to QB and the new data applied will appear at QA.**
10. **Repeat steps 2 and 3 till all the 4 bits data are entered one by one into the shift register.**

Serial In Serial Out:-

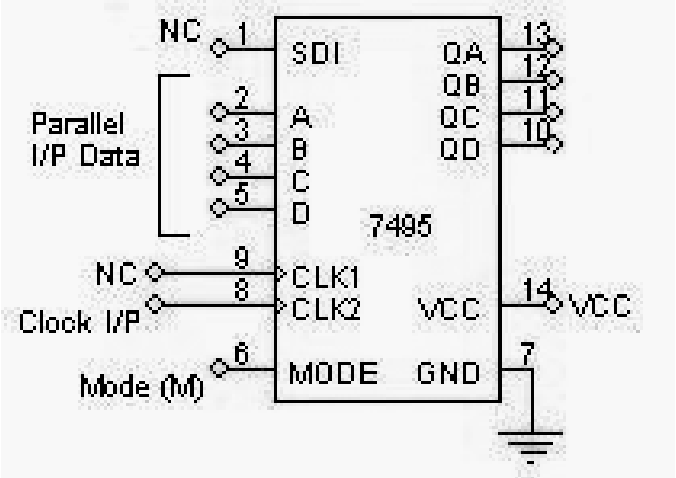
1. **Connections are made as per circuit diagram.**
2. **Load the shift register with 4 bits of data one by one serially.**
3. **At the end of 4^{th} clock pulse the first data 'd0' appears at QD.**
4. **Apply another clock pulse; the second data 'd1' appears at QD.**
5. **Apply another clock pulse; the third data appears at QD.**
6. **Application of next clock pulse will enable the 4^{th} data 'd3' to appear at QD. Thus the data applied serially at the input comes out serially at QD**

PISO:-



Mode	Clock	Parallel i/p				Parallel o/p			
		A	B	C	D	QA	QB	QC	QD
1	1	1	0	1	1	1	0	1	1
0	2	X	X	X	X	X	1	0	1
0	3	X	X	X	X	X	X	1	0
0	4	X	X	X	X	X	X	X	1

PIPO:-



Clock	Parallel i/p				Parallel o/p			
	A	B	C	D	QA	QB	QC	QD
1	1	0	1	1	1	0	1	1

Parallel In Parallel Out:-

1. **Connections are made as per circuit diagram.**
2. **Apply the 4 bit data at A, B, C and D.**
3. **Apply one clock pulse at Clock 2 (Note: Mode control M=1).**
4. **The 4 bit data at A, B, C and D appears at QA, QB, QC and QD respectively.**

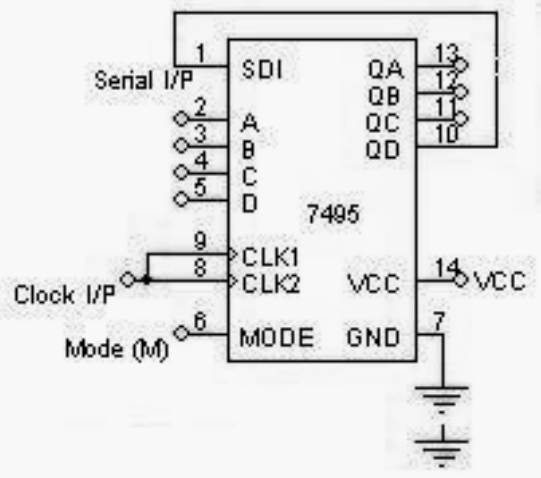
Parallel In Serial Out:-

1. **Connections are made as per circuit diagram.**
2. **Apply the desired 4 bit data at A, B, C and D.**
3. **Keeping the mode control M=1 apply one clock pulse. The data applied at A, B, C and D will appear at QA, QB, QC and QD respectively.**
4. **Now mode control M=0. Apply clock pulses one by one and observe the data coming out serially at QD.**

Left Shift:-

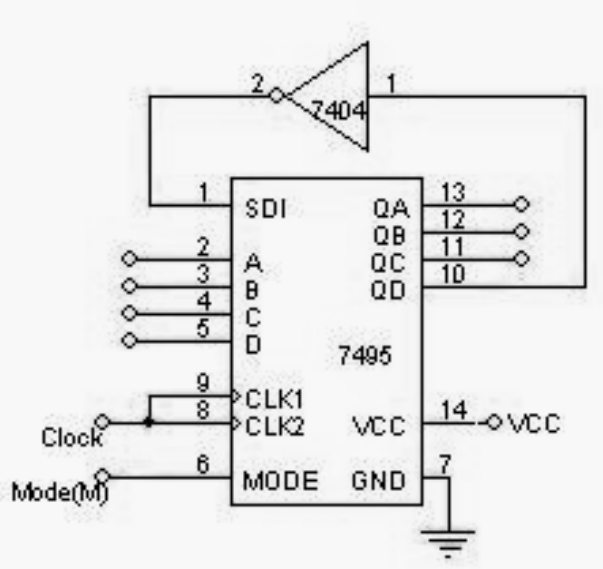
1. **Connections are made as per circuit diagram.**
2. **Apply the first data at D and apply one clock pulse. This data appears at QD.**
3. **Now the second data is made available at D and one clock pulse applied. The data appears at QD to QC and the new data appears at QD.**
4. **Step 3 is repeated until all the 4 bits are entered one by one.**
5. **At the end 4^{th} clock pulse the 4 bits are available at QA, QB, QC and QD.**

Circuit Diagram: - Ring Counter



Mode	Clock	QA	QB	QC	QD
1	1	1	0	0	0
0	2	0	1	0	0
0	3	0	0	1	0
0	4	0	0	0	1
0	5	1	0	0	0
0	6	repeats			

Johnson Counter:-



Mode	Clock	QA	QB	QC	QD
1	1	1	0	0	0
0	2	1	1	0	0
0	3	1	1	1	0
0	4	1	1	1	1
0	5	0	1	1	1
0	6	0	0	1	1
0	7	0	0	0	1
0	8	0	0	0	0
0	9	1	0	0	0
0	10	repeats			

Experiment No: 15 **Date: __/__/____**

JOHNSON COUNTERS / RING COUNTER

Aim:- **Design and testing of Ring counter/ Johnson counter.**

Apparatus Required: -

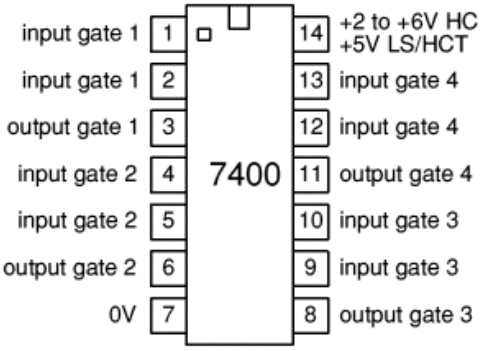
IC 7495, IC 7404, etc.

Procedure: -

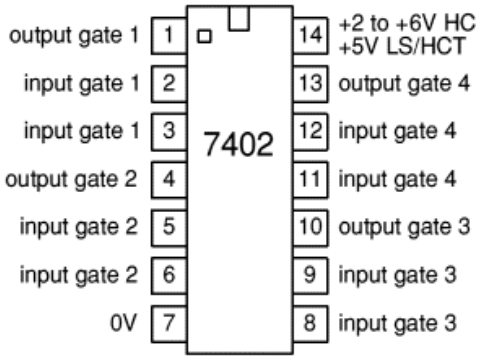
1. **Connections are made as per the circuit diagram.**
2. **Apply the data 1000 at A, B, C and D respectively.**
3. **Keeping the mode M = 1, apply one clock pulse.**
4. **Now the mode M is made 0 and clock pulses are applied one by one and the truth table is verified.**
5. **Above procedure is repeated for Johnson counter also.**

Useful IC Pin details

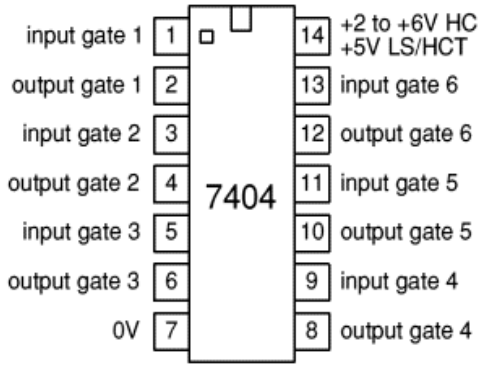
7400(NAND)



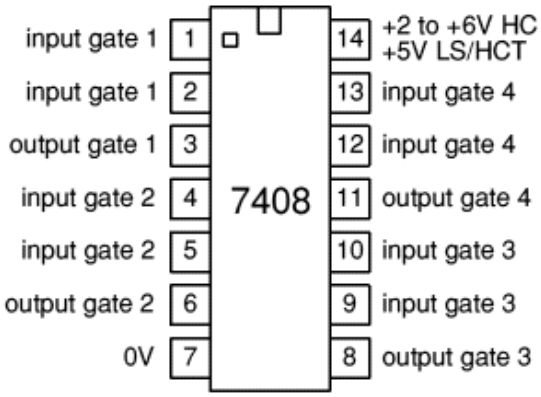
7402(NOR)



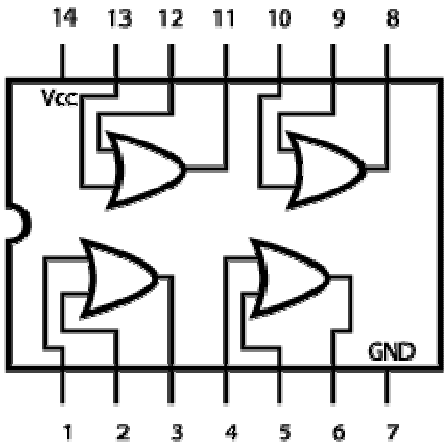
7404(NOT)



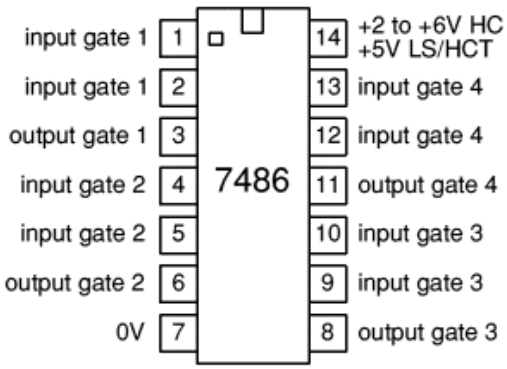
7408(AND)



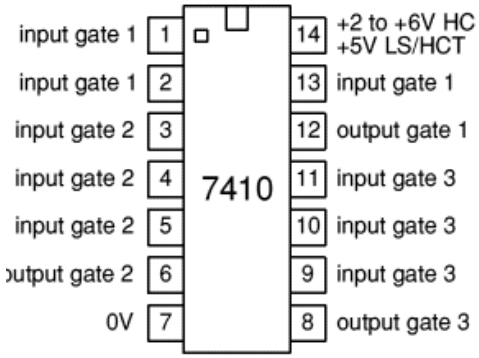
7432(OR)



7486(XOR)



7410(3-i/p NAND)



7420(4-i/p NAND)