ASSIGNMENT: (Use Journal pages to solve assignment)

EXPT.No.01

1. What are the differences between Full adder and Full subtractor circuit?
2. Justify, The parallel adder is most suitable adder circuit to speed up the operation process?
3. Design of half adder and half subtractor circuit using universal Logic gates?
4. Design of Full adder circuit using half adder circuit?
5. Design of Full subtractor circuit using half subtractor circuit?

EXPT.No.02

1. What is a need of code conversion?
2. Justify code converter circuit is a combinational logic circuit?
3. What do you mean by Gray code, BCD Code, Excess-3 code?
4. Why Excess-3 code is called as a self - complementary code?
5. What are the applications of Gray code?
6. Why gray codes are used in K-Map simplification?
7. What are the properties of gray code?

EXPT.No.03

1. What do you understand by Look Ahead Carry Generation circuit?
2. What are the features of IC-74LS83?
3. Why is six added to make valid BCD?
4. Why subtractors IC are not available in market?
5. What is meant by BCD subtraction? What are the methods?
6. What is the maximum input that can be applied to BCD adder IC?
7. Design of BCD adder circuit using IC-74LS83 and NAND gates?
8. Why EX-OR gates are used in BCD subtractor circuit?

EXPT.No.04

1. What are the features of IC-74LS153 and IC-74151?
2. List of different MUX ICs ?
3. What is a Role of Select input in multiplexer circuit?
4. Realization of 2:1 MUX. using logic gates?
5. Design of 4-bit binary to 4-bit gray code converter circuit using IC-74LS153?
6. Design of BCD to Exces-3 code converter circuit using IC-74153?
7. Design of 16:1 Multiplexer using 4:1 Multiplexer?
8. Realize 2-input EX-OR logic gates using 2:1 MUX?
9. What are the feature of IC-74138 and IC-74154?
10. List of different DEMUX ICs?
11. Realization of 1:2 DE-MUX /Decoder circuit using logic gates with chip enable input?
12. What is a difference between DEMUXand Decoder circuits?
13. Justify, decoder as a ROM?
14. Design of any four variable function using IC-74138 and IC-74154?

CLASS: S.E.COMP **SUBJECT :DEL**

EXPT. NO.:1 **DATE:**

TITLE : BINARY ADDER AND SUBTRACTOR CIRCUITS

OBJECTIVE :

1. Design and Implement Full adder circuit using basic gates and universal logic gates
2. Design and Implement Full Subtractor circuit using basic gates and universal logic gates

APPARATUS :

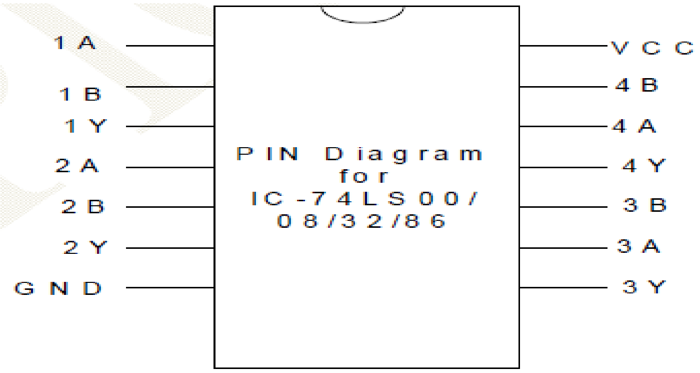
Digital-Board, GP-4Patch-Cords, IC-74LS86, IC-74LS32, IC-74LS08 / IC-74LS04 and IC-74LS00 and Required Logic gates if any.

THEORY :

Binary Adder and subtractor are a combinational logic circuit which is used to perform binary addition and subtraction .Full adder is a little more difficult to implement than a half-adder. The full-adder has three inputs and two outputs. The first two inputs are A and B and the third input is an input carry designated as CIN. When full adder logic is designed we will be able to string eight of them together to create a byte-wide adder and cascade the carry bit from one adder to the next

The full subtractor is a combinational circuit with three inputs A,B,C and two output D and C'. A is the 'minuend', B is 'subtrahend', C is the 'borrow' produced by the previous stage, D is the difference output and C' is the borrow output.

PIN DIAGRAM:



PROCEDURE :

1. Make the connections as per the Logic circuit of Full adder circuit and Verify its Truth Table.
2. Make the connections as per the Logic circuit of Full subtractor circuit and Verify its Truth Table

Design of Full adder circuit

Dec.Equ.	INPUT			OUTPUT	
	A	B	Cin	Sum	Carry

K-Map Simplification for Sum and Carry

Logic Diagram:

Design of Full subtractor circuit

Dec. Equ.	INPUT			OUTPUT	
	A	B	Cin	Difference	Borrow

K-Map Simplification for Difference and Borrow

Logic Diagram:

Logic Gates / MSI Device required for Implementation:

Sr.No.	Title	Name of the IC	Number of Gates required	IC Required
01	Full adder circuit using Basic logic gates			
	Full adder circuit using Universal logic gates			

02	Full subtractor circuit using Basic logic gates			
	Full subtractor circuit using Universal logic gates			

CONCLUSION:

____

____

____

____

____

REFFRENCE:

1. **R.P.Jain “Modern Digital Electronics” TMH 4th Edition**
2. **D.Leach,Malvino,Saha,”Digital Principles and Applications”,TMH**

Subject teacher Sign with Date Remark

CLASS: S.E.COMP **SUBJECT :DEL**

EXPT. NO.:2 **DATE:**

TITLE : CODE CONVERTER

OBJECTIVE :

1. Design and Implement 4-bit Binary to Gray code converter using minimum number of logic gates and Vice-versa
2. Design and Implement Excess-3 to BCD code converter using minimum number of logic gates and Vice-versa

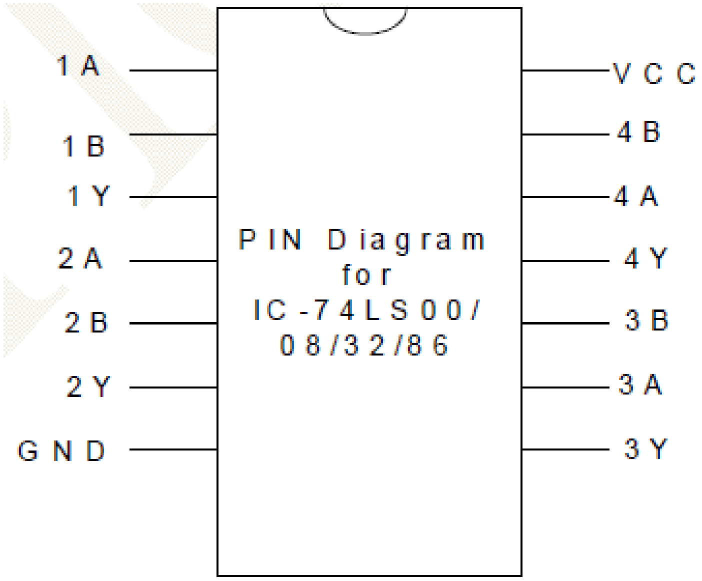
APPARATUS :

Digital-Board, GP-4Patch-Cords, IC-74LS86, IC-74LS32, IC-74LS08 / IC-74LS04 and Required Logic gates if any.

THEORY :

Code converter is combinational logic circuits, which can be used to convert one number system to another. Binary code is a weighted code having base 2.Gray code is a code in which one bit change is obtained; Gray code is also called *unit distance code or reflected code*. BCD code is basically a 4-bit binary code but it is valid from 0 to 9. Excess-3 code is basically 4-bit binary code which can be obtained by adding 3 to each binary, that is Excess-3 code are valid from 3 to 15.Excess-3 code is Non-Weighted code.Excess-3 code is also called as *sequential code or self-complementary code.*

PIN DIAGRAM:



PROCEDURE :

1. Make the connections as per the Logic circuit of 4-bit Binary to 4-bit Gary Code converter and Vice-versa and Verify its Truth Table.
2. Make the connections as per the Logic circuit of 4-bit BCD to 4-bit Excess-3 Code converter and Vice-versa and Verify its Truth Table

Design of 4-bit Binary to Gray code converter

Dec. Equ.	*Binary code Input*				**Gray code Output**			
	B_3	B_2	B_1	B_0	G_3	G_2	G_1	G_0

K-Map Simplification for G_3, G_2, G_1, G_o

Logic Diagram:

Design of 4-bit Gray to Binary code converter

Dec.Equ.	***Gray code Input***				**Binary code Output**			
	G_3	**G_2**	**G_1**	**G_0**	**B_3**	**B_2**	**B_1**	**B_0**

Dec.Equ.	***Gray code Input***				**Binary code Output**			
	G_3	**G_2**	**G_1**	**G_0**	**B_3**	**B_2**	**B_1**	**B_0**

K-Map Simplification for B_3, B_2, B_1, B_0

Logic Diagram:

Design of BCD code to Excess-3Code converter

Dec.Equ.	***BCD code Input***				**Excess-3 code Output**			
	B_3	**B_2**	**B_1**	**B_0**	**E_3**	**E_2**	**E_1**	**E_0**

Dec.Equ.	*BCD code Input*				Excess-3 code Output			
	B_3	B_2	B_1	B_0	E_3	E_2	E_1	E_0

K-Map Simplification for E_3, E_2, E_1, E_o

Logic Diagram:

Design of Excess-3 code to BCD Code converter

Dec.Equ.	**Excess-3 *code Input***				***BCD* code Output**			
	E_3	**E_2**	**E_1**	**E_0**	**B_3**	**B_2**	**B_1**	**B_0**

	Excess-3 *code Input*				***BCD* code Output**			
Dec.Equ.	**E_3**	**E_2**	**E_1**	**E_0**	**B_3**	**B_2**	**B_1**	**B_0**

K-Map Simplification for B3, B2, B1, Bo

Logic Diagram:

Logic Gates / MSI Device required for Implementation:

Sr.No.	Title	Name of the IC	Number of Gates	IC Required
01	Binary to Gray code			
	Gray to Binary code			
02	BCD to Excess-3code			
	Excess-3to BCD			

CONCLUSION:

____

____

____

____

____

REFFRENCE:

1. **R.P.Jain "Modern Digital Electronics" TMH 4th Edition**
2. **D.Leach,Malvino,Saha,"Digital Principles and Applications",TMH**

Subject teacher Sign with Date Remark

CLASS : S.E. COMP **SUBJECT : DEL**

EXPT. NO. : 3 **DATE :**

TITLE : BCD ADDER/ 9’S COMPLEMENT CIRCUIT

OBJECTIVE :

1. Design and Implement BCD Adder circuit using IC-74LS83
2. Design and Implement 9’s Complement circuit using IC-74LS83

APPARATUS :

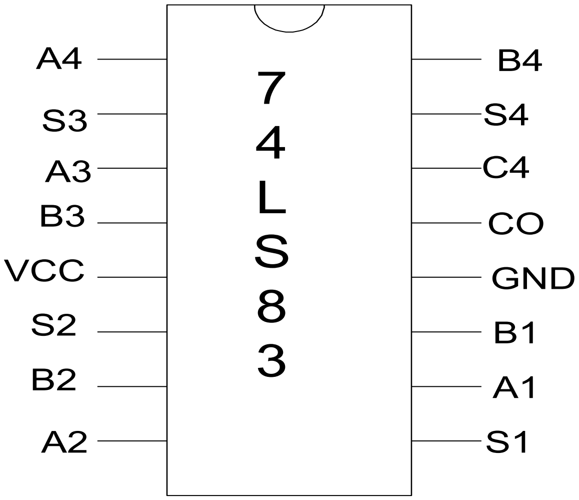
Digital-Board, GP-4 Patch-Cords, IC-74LS83, IC-74LS32, IC-74LS04/IC-74LS08 and Required Logic gates if any

THEORY :

IC 74LS83 is a 4-bit binary parallel adder. By using 74LS83 we can implement BCD adder. BCD means Binary coded Decimal. BCD numbers are valid from 0 to 9. For BCD adder when addition is below 9, carry is 0 result is valid BCD. When addition is more than 9 and carry is 0 as well as when addition is more than 15 and carry is 1 result of binary adder IC is Invalid BCD. We can convert invalid BCD to valid BCD by adding six. Max addition (9+9) result is 18 if carry input is 0 and 19 if carry input is 1. Thus for binary result greater than 9 six should be added to the result as a correction factor using combinational circuit.

74LS83 can also be used to implement the BCD subtractor. For BCD subtraction first we have to find 9’s complements. To find 9’s complements using IC-74LS83, first find 1’s complement of a given number then add to 1010.

PIN Diagram :



PROCEDURE :

1. Make the connections as per the Logic circuit of 1 digit BCD adder using IC74LS83 and Verify its Truth Table.
2. Make the connections as per the Logic circuit of 9's complement circuits using IC74LS83 and Verify its Truth Table.

Design of BCD Adder using IC-74LS83:

SUM					
Dec.Equ.	**S4**	**S3**	**S2**	**S1**	**Tens Place O/P**
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	1	0	1	0	
11	1	1	0	1	
12	1	1	0	0	
13	1	1	0	1	
14	1	1	1	0	
15	1	1	1	1	

K-map Simplification:

Logical Expression:

Logic Diagram:

Design of 9’s complement circuit using IC-74LS83:

Logic Diagram:

Logic Gates / MSI Device required for Implementation:

Sr.No.	Title	Name of the IC	Number of Gates required	IC Required
01	BCD Adder Circuit			
02	9’s complement Circuit			

CONCLUSION:

____

____

____

____

____

____

____

REFFRENCE:

1. **R.P.Jain "Modern Digital Electronics" TMH 4th Edition**
2. **D.Leach,Malvino,Saha,"Digital Principles and Applications",TMH**

Subject teacher Sign with Date   Remark

CLASS: S.E.COMP **SUBJECT :DEL**

EXPT. NO.:4 **DATE:**

TITLE : MULTIPLEXER / DEMULTIPLEXER

OBJECTIVE :

1. Verification of Function table of IC-74LS153
2. Design and Implement 8:1 MUX using IC-74LS153 and verify its truth Table
3. Realization of the following Boolean expression using IC-74153 and verify its truth-Table
   F(A,B,C,D)=∑m( )
4. Verification of Function table of IC-74LS138
5. Realization of the following Boolean expression using IC-74138 and verify its truth-Table
   F(A,B,C)=∑m( )

APPARATUS :

Digital-Board, GP-4Patch-Cords, IC-74LS32, IC-74LS08 / IC-74LS04 / IC-74LS153/IC-741LS38 and Required Logic gates if any.

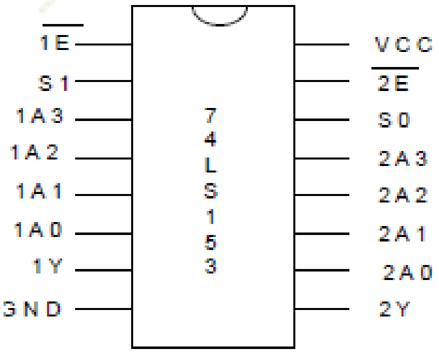
THEORY :

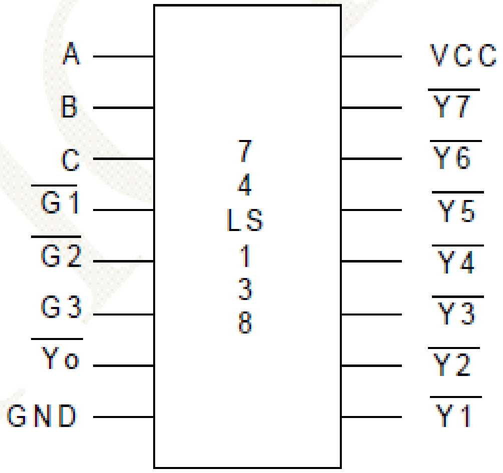
Multiplexer is a combinational logic device, which has many input & one output, this output can be selected according to select input. IC-74LS153 is Dual 4:1 MUX. It is a 16-pin dual in-line packaged IC, which has two enable pins (STROBE Active Low). We can design 8:1 MUX using cascading of Two 4:1 MUX. This is achieved with the help of enable / strobe inputs and multiplexer tree is designed. To implement 8:1 MUX we need

3 select lines and one output. Use dual 4:1 MUX for select lines 2 are common, use enable pin for 3rd select line. Connect the first strobe of first 4:1 MUX and the second strobe of second 4:1 MUX through one Inverter. This is 3rd select line only one 4:1 MUX will become active at one time. Connect outputs of 2 -MUX to OR get so that we get one output .

De-Multiplexer/Decoder is a combinational logic device, which has one input & many output, one output can be selected according to select input. IC-74LS138 is 3 to 8 Line Decoder/Demultiplexer. It is a 16-pin dual packaged IC, which has three enable pins (2-STROBE Active Low and one Active High). IC-74LS138 produces complementary output. i.e. the output of 74LS138 is Active Low. We can design any combinational circuits using IC-74LS138. DEMUX/Decoder performs reverse operation to that of Multiplexer

PIN DIAGRAM:





PROCEDURE :

1. Make the connections as per the Pin diagram of IC-74LS153 and Verify its Truth Table.
2. Make the connections as per the Logic circuit of 8:1 MUX and Verify its Truth Table
3. Make the connections as per Logic circuit of the given function and Verify its Truth Table
4. Make the connections as per the pin diagram of IC-74LS138 and Verify its Truth Table
5. Make the connections as per the Logic circuit of given function and Verify its Truth Table

Design of Multiplexer

Function Table: IC-74LS153

Chip Enable I/P		Select Input		OUTPUT	
1E	2E	S_1(MSB)	S_0(LSB)	MUX-I (1Y)	MUX-2 (2Y)
0	0	0	0		
0	0	0	1		
0	0	1	0		
0	0	1	1		

Logical Expression:

Logic Diagram:

Design of 8:1 MUX.using IC-74LS153

Design Table:

S2(MSB)	S1	S0(LSB)	OUTPUT (Y)
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Logical Expression:

Logic Diagram:

Design of Realization of Boolean expression using IC-74LS153

F(A,B,C,D)=Σ(   )

(LSB Variable Reduction Method)

Decimal Equ.	A (MSB) (s2)	B (s1)	C (so)	D(LSB)	OUTPUT Y	Derived Logic
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	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		

Logic Diagram:

Design of Realization of Boolean expression using IC-74LS153

F(A,B,C,D)=Σ(   )

(MSB Variable Reduction Method)

Decimal Equ.	A (MSB) (s2)	B (s1)	C (so)	D(LSB)	OUTPUT Y	Derived Logic
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	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		

Implementation Table:

Logic Diagram:

Design of Decoder

Function Table : IC-74LS138

Chip Enable Input			Active Low Output							
G_3	G_2	G_1	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
1	0	0								
1	0	0								
1	0	0								
1	0	0								
1	0	0								
1	0	0								
1	0	0								
1	0	0								

Logical Expression:

Logic Diagram:

Design of Realization of Boolean expression using IC-74LS138

F(A,B,C,D)=Σ( )

Decimal Equ.	A	B	C	OUTPUT F(A,B,C)
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	

Logical Expression:

Logic Diagram:

Logic Gates / MSI Device required for Implementation:

Sr.No.	Title	Name of the IC	Number of Gates required	IC Required
01	Design of 8:1 MUX.			
02	Realization of Boolean expression using LSB reduction method			

03	Realization of Boolean expression using MSB reduction method			
04	Realization of Boolean expression using Decoder IC			

CONCLUSION:

____

____

____

____

____

REFFRENCE:

1. **R.P.Jain “Modern Digital Electronics” TMH 4th Edition**
2. **D.Leach,Malvino,Saha,”Digital Principles and Applications”,TMH**

Subject teacher Sign with Date　　　　　　　　　　Remark

CLASS: S.E.COMP **SUBJECT :DEL**

EXPT. NO.:5 **DATE:**

TITLE : DIGITAL MAGNITUDE COMPARATOR CIRCUIT

OBJECTIVE :

1. Design and Implement 1-bit Digital magnitude comparator circuit using logic gates
2. Design and Implement 2-bit Digital magnitude comparator circuit using logic gates
3. Verification of function table of IC-74LS85
4. Design and Implement 5-bit Digital magnitude comparator circuit using IC-74LS85
5. Design and Implement 8-bit Digital magnitude comparator circuit using IC-74LS85

APPARATUS :

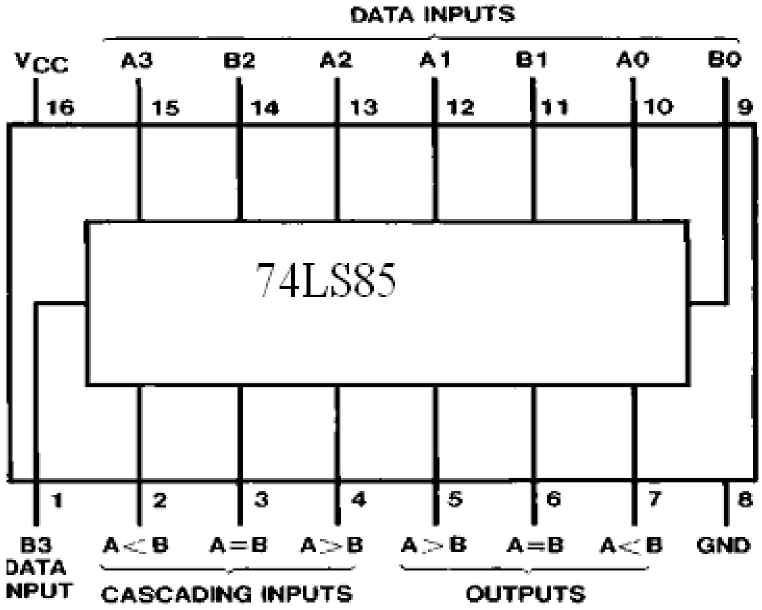
Digital-Board, GP-4Patch-Cords, IC-74LS86, IC-74LS32, IC-74LS08 / IC-74LS04/IC-74LS85 and Required Logic gates if any.

THEORY :

Magnitude Comparator is a logical circuit, which compares two Signals A and B and generates three logical outputs, whether. A > B, A = B, or A < B IC 7485 is a high speed 4-bit Magnitude, Comparator which compares two 4-bit words. The A = B Input must be held high for proper compare operation. These 4-bit magnitude comparators perform comparison of Straight binary or BCD codes. Three fully-decoded decisions about two, 4-bit words (A, B) are made and are externally available at three outputs. These devices are fully expandable to any number of bits without external gates. Words of greater Words of greater length may be compared by connecting Comparators in cascade.

The A > B, A < B, and A = B outputs of a stage handling less-significant bits are connected to the Corresponding inputs of the next stage handling more-significant bits. The stage handling the least significant bits must have a High-leveL voltage applied to the A = B input The cascading path is Implemented with only a two-gate-level delay to reduce overall comparison times for long word

PIN DIAGRAM:



PROCEDURE :

1. Make the connections as per the Logic circuit of 1-bit Digital comparator circuit and Verify its Truth Table.
2. Make the connections as per the Logic circuit of 1-bit Digital comparator circuit and

Verify its Truth Table.

3. Make the connections as per the pin diagram of IC-74LS85 and Verify its Function Table
4. Make the connections as per the Logic circuit of 5-bit Digital comparator circuit and Verify its Function Table
5. Make the connections as per the Logic circuit of 8-bit Digital comparator circuit and Verify its Function Table

Design of 1-bit Digital comparator

Design-Table:

Input		OUTPUT		
A	**B**	**Y1=A<B**	**Y2=A=B**	**Y3=A>B**
0	**0**			
0	**1**			
1	**0**			
1	**1**			

K-Map Simplification for Y1=A<B, Y2=A=B, Y3=A>B

Logic Diagram:

Design of 2-bit Digital comparator

Design-Table:

Dec. Equ.	Input				OUTPUT		
	A1	A0	B1	B0	Y1=A<B	Y2=A=B	Y3=A>B
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	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			

K-Map Simplification for Y1=A<B, Y2=A=B, Y3=A>B

Logic Diagram:

Design using IC-74LS85

Verification of Function Table:IC-74LS85

Design of 5-bit Digital comparator using IC-74LS86

Design of 8-bit Digital comparator using IC-74LS85

Logic Gates / MSI Device required for Implementation:

Sr.No.	Title	Name of the IC	Number of Gates required	IC Required
01	1-bit compartor using logic gates			
02	2-bit compartor using logic gates			
03	5-bit comparator			
04	8-bit comparator			

CONCLUSION:

____

____

____

____

____

REFFRENCE:

1. **R.P.Jain "Modern Digital Electronics" TMH 4th Edition**
2. **D.Leach,Malvino,Saha,"Digital Principles and Applications",TMH**

Subject teacher Sign with Date Remark

CLASS : S.E. COMP **SUBJECT : DEL**

EXPT. NO. : 6 **DATE :**

TITLE : RIPPLE COUNTER CIRCUIT

OBJECTIVE :

1. Design and Implement 3-bit Asynchronous (Ripple) UP Counter circuit using IC-74LS76

   Draw Timing Diagram.
2. Design and Implement 3-bit Asynchronous (Ripple) DOWN Counter circuit using IC-74LS76

   Draw Timing Diagram.
3. Design and Implement MOD-N Asynchronous (Ripple) UP Counter circuit using IC-74LS76

   Draw Timing Diagram.

APPARATUS :

Digital-Board, GP-4 Patch-Cords, IC-74LS76, IC-74LS32, IC-74LS04/IC-74LS08 and Required Logic gates if any

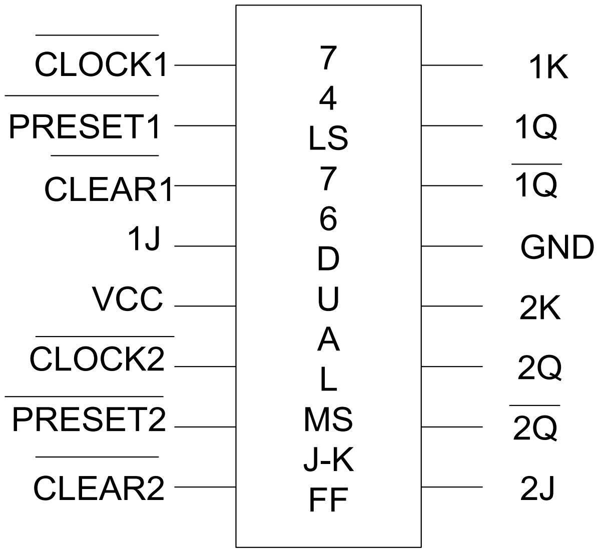
THEORY :

Counter is a Sequential Logic device which can be use to count the number of pulses given to the circuit. Counter can be classified into two category one is Synchronous and other is Asynchronous (Ripple) In case of Asynchronous counter output of first flip-flop goes to the clock of next and so on, and input of all flip-flop is connected to VCC for IC-

74LS76.

All set and reset pin is connected to VCC.Asynchronous counter is easy to design as compared to Synchronous Counter. Synchronous Counter is faster than Asynchronous Counter.IC-74LS76 is Dual M/S-JK filp-flop, which means in one IC there are two M/S-JK filp-flop are contained.

PIN Diagram :


$\overline{\text{CLOCK1}}$
$\overline{\text{PRESET1}}$
$\overline{\text{CLEAR1}}$
1J
VCC
$\overline{\text{CLOCK2}}$
$\overline{\text{PRESET2}}$
$\overline{\text{CLEAR2}}$
7 4 LS 7 6 D U A L MS J-K FF
1K
1Q
$\overline{\text{1Q}}$
GND
2K
2Q
$\overline{\text{2Q}}$
2J

PROCEDURE :

1. Make the connections as per the Logic circuit of 3-bit Ripple UP Counter circuit using IC-74LS76 and Verify its Truth Table.
2. Make the connections as per the Logic circuit of 3-bit Ripple DOWN Counter circuit using IC-74LS76 and Verify its Truth Table.
3. Make the connections as per the Logic circuit of MOD-N Ripple UP Counter circuit using IC-74LS76 and Verify its Truth Table.

Design of 3- bit Asynchronous UP- Counter

Logic Diagram:

Observation Table:

CLOCK PULSE	Output		
	Q_A	Q_B	Q_C
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

Design of 3- bit Asynchronous DOWN- Counter

Logic Diagram:

Observation Table:

CLOCK PULSE	Output		
	Q_A	Q_B	Q_C
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

Design of (MOD - N =)Asynchronous UP-Counter

Dec.Equ.	OUTPUT			RESET LOGIC
	Q_A	Q_B	Q_C	Y
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	

K-MAP for RESET LOGIC

Logic Diagram:

Logic Gates / MSI Device required for Implementation:

Sr.No.	Title	Name of the IC	Number of Gates required	IC Required
01	3-bit Ripple Up Counter			
02	3-bit Ripple DOWN Counter			
03	(MOD-N=) UP -Counter			

CONCLUSION:

____

____

____

____

____

____

____

REFFRENCE:

1. **R.P.Jain “Modern Digital Electronics” TMH 4th Edition**
2. **D.Leach,Malvino,Saha,”Digital Principles and Applications”,TMH**

Subject teacher Sign with Date Remark

CLASS : S.E. COMP **SUBJECT : DEL**

EXPT. NO. : 7 **DATE :**

TITLE : UP/DOWN COUNTER CIRCUIT

OBJECTIVE :

1. Design and Implement 3-bit Asynchronous (Ripple) UP/DOWN –Counter using Mode control switch (Use-IC-74LS76) and verify its truth-table.

   Draw Timing diagram

   When M=0: Circuit perform UP –Counting operation

   When M=1: Circuit perform DOWN –Counting operation

2. Design and Implement 3-bit Synchronous UP/DOWN –Counter using Mode control switch (Use-IC-74LS76) and verify its truth-table.

   Draw Timing diagram

   When M=0: Circuit performs UP –Counting operation

   When M=1: Circuit performs DOWN –Counting operation

3. Design and Implement MOD-N Counter circuit using IC-74LS90 and IC-74LS191

   Draw Timing Diagram.

APPARATUS :

Digital-Board, GP-4 Patch-Cords, IC-74LS76, IC-74LS90, IC-74LS191, IC-74LS32, IC-74LS04/IC-74LS08 and Required Logic gates if any

THEORY :

Counter is a Sequential Logic device which can be use to count the number of pulses given to the circuit. Counter can be classified into two category one is Synchronous and other is Asynchronous (Ripple) In case of Asynchronous counter output of first flip-flop goes to the clock of next and so on, and input of all flip-flop is connected to VCC for IC-74LS76.

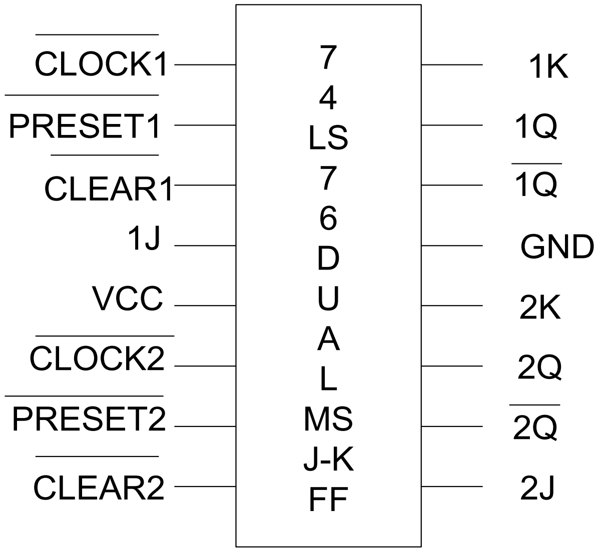
All set and reset pin is connected to VCC.Asynchronous counter is easy to design as compared to Synchronous Counter. Synchronous Counter is faster than Asynchronous Counter.IC-74LS76 is Dual M/S-JK filp-flop, which means in one IC there are two M/S-JK filp-flop are contained.

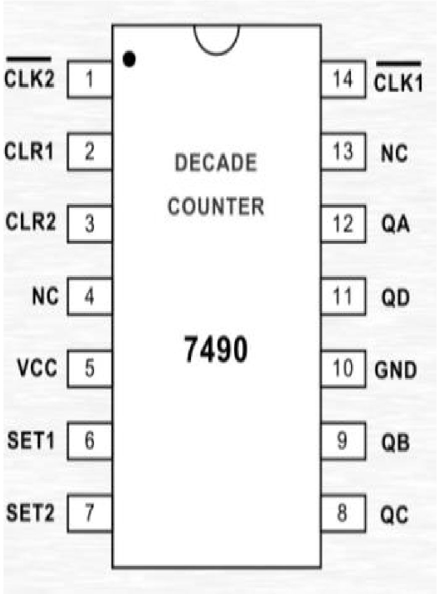
IC -7490 is called as a 4-bit MS-JK Flip-flop decade (BCD) Ripple counter. It contains 4 -master slave flip flops internally connected to provide MOD-2 i.e. divide by 2 and MOD-5 i.e. divide by 5 counter.MOD-2 and Mod-5 counters can be used independently or in cascading. Each Counter has a separate clock input to initiate state changes of the counter on the high to low clock transition. Since the o/p from the divide by 2 section is not internally connected to the succeeding stages. The device may be operated in various counting modes. In a BCD counter the CP1 input must be externally connected to Q0 output. The CP0 input receives the incoming count producing a BCD count sequence.

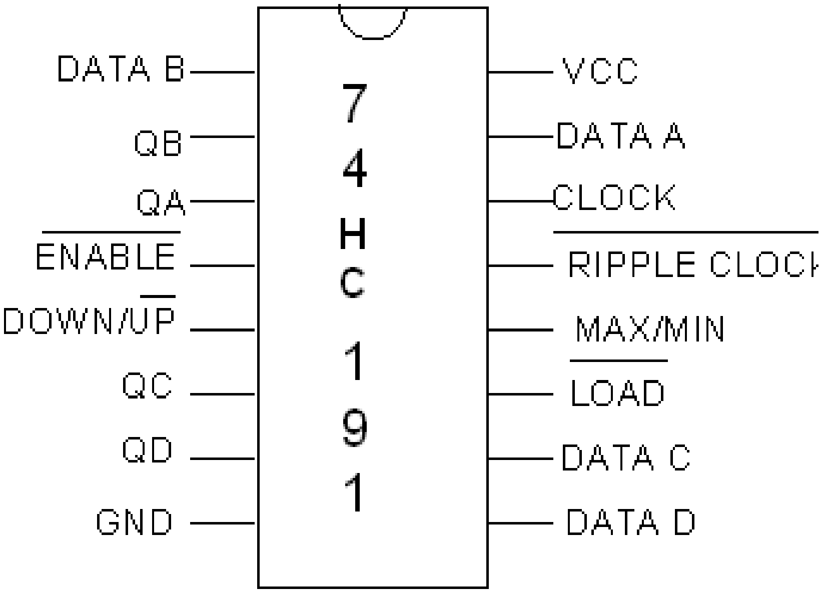
It is also provided with additional gating to provide a divide by 2 counter and binary counter for which the count cycle length is divide by 5. The device may be operated in various counting modes.

There are 2 reset inputs R0 (1) and R0 (2) both of which need to be connected to the 'logic 1' for clearing all flip flops. Two set inputs Rg(1) and Rg(2) when connected to logic are used for setting counter to 1001 (BCD 9).IC74191 is 4-bit binary parallel presettable Programmable UP/DOWN synchronous counter. It contains 4 master slave J-K flip flops with internal gating and steering logic to provide asynchronous reset and synchronous count up/down operations; its asynchronous parallel capability permits the counter to be preset to any desire number. D0 to D3 are the parallel data inputs. Information present on the parallel data inputs D0 to D3 is loaded into the counter and appears on the output when the load **PL** input is low this operation is overrides the counting function .counting is inhabited by the high level on the enable CE input, when CE input is low internal state changes are initiated synchronously by the low to high transitions of the clock inputs the up/down input signal determines the direction of input.

PIN Diagram :


$\overline{\text{CLOCK1}}$
$\overline{\text{PRESET1}}$
$\overline{\text{CLEAR1}}$
1J
VCC
$\overline{\text{CLOCK2}}$
$\overline{\text{PRESET2}}$
$\overline{\text{CLEAR2}}$
74LS76 DUAL MS J-K FF
1K
1Q
$\overline{\text{1Q}}$
GND
2K
2Q
$\overline{\text{2Q}}$
2J





PROCEDURE :

1. Make the connections as per the Logic circuit of 3-bit Asynchronous UP/ DOWN Counter circuit using IC-74LS76 and Verify its Truth Table.
2. Make the connections as per the Logic circuit of 3-bit Synchronous UP/ DOWN Counter circuit using IC-74LS76 and Verify its Truth Table.
3. Make the connections as per the Logic circuit of MOD-N Counter circuit using IC-74LS90 /IC-74LS191 and Verify its Truth Table.

Design of 3-bit (Ripple) UP/DOWN-Counter Using Mode control S/W

Dec. Equ.	M	Q	Qbar	Derived Logic (Y)
	0	0	1	
	0	1	0	
	1	0	1	
	1	1	0	

K-MAP for Derived Logic

Logic diagram:

Observation Table:

CLOCK PULSE	Mode control (M)	Output		
		Q_A	Q_B	Q_C
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
CLOCK PULSE	Mode control (M)	Output		
		Q_A	Q_B	Q_C
0	1	1	1	1
1	1	1	1	0
2	1	1	0	1
3	1	1	0	0
4	1	0	1	1
5	1	0	1	0
6	1	0	0	1
7	1	0	0	0

Design of 3-bit Synchronous UP/DOWN-Counter Using Mode control Input (Switch)

M	PRESENTSTATE			NEXT STATE			INPUT					
	Q_A	Q_B	Q_C	Q_A^+	Q_B^+	Q_C^+	J_A	K_A	J_B	K_B	J_C	K_C
0	0	0	0									
0	0	0	1									
0	0	1	0									
0	0	1	1									
0	1	0	0									
0	1	0	1									
0	1	1	0									
0	1	1	1									
1	1	1	1									
1	1	1	0									
1	1	0	1									
1	1	0	0									
1	0	1	1									
1	0	1	0									
1	0	0	1									
1	0	0	0									

K-Map Simplification for J_A , K_A , J_B , K_B , J_C , K_C

Logic Diagram:

Design of MOD-2 / MOD-5 /MOD-10/MOD-20 Counter using IC-7490
Logic Diagram: (MOD-2 / MOD-5 Counter)

Logic Diagram: (MOD-10 / MOD-20 Counter)

Observation Table: Decade Counter

CLOCK PULSE	Output			
	Q_3	Q_2	Q_1	Q_0
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

Design of MOD-7 Counter using IC-7490

Dec.Equ.	OUTPUT				RESET LOGIC
	Q_3	Q_2	Q_1	Q_0	Y
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	

K-Map for Reset Logic

Logic Diagram: (MOD-7 Counter)

Design of 4-bit UP and DOWN /Programmable MOD-N Counter using IC-74191

Function Table: IC-74191

PL	CE	CLK	Up/DOWN	Data Input				Output				Mode of Operation
				D_3	D_2	D_1	D_0	Q_3	Q_2	Q_1	Q_0	
0	×	×	×	1	0	0	0	1	0	0	0	Parallel Load
1	0	1	0	-	-	-	-	0	0	0	0	Up Counting
								To				
								1	1	1	1	
1	0	1	1	-	-	-	-	1	1	1	1	Down Counting
								To				
								0	0	0	0	

Logic Diagram: (4-bit UP Counter)

Logic Diagram: (4-bit Down Counter)

Logic Diagram :(4-bit Presettable Programmable MOD-N UP/Down Counter)

Logic Gates / MSI Device required for Implementation:

Sr.No.	Title	Name of the IC	Number of Gates required	IC Required
01	3-bit Ripple UP/DOWN Counter			
02	3-bit Synchronous UP/DOWN Counter			

03	MOD-02/05/10 Counter			
04	MOD-20 Counter			
05	MOD-07 Counter			
06	4-Bit UP/DOWN Counter			
07	MOD-N Presettable Programmable Counter			

CONCLUSION:

____

____

____

____

____

REFFRENCE:

1. **R.P.Jain "Modern Digital Electronics" TMH 4th Edition**
2. **D.Leach,Malvino,Saha,"Digital Principles and Applications",TMH**

Subject teacher Sign with Date Remark

PUNE INSTITUTE OF COMPUTER TECHNOLOGY, PUNE - 411043

Department of Computer Engineering

CLASS : S.E. COMP **SUBJECT : DEL**

EXPT. NO. : 8 **DATE :**

TITLE : SEQUENCE GENERATOR CIRCUIT

OBJECTIVE :

1. Design and Implement the following Sequence generator circuit using IC-74LS76 and verify its truth-table.

   Sequence: ( )

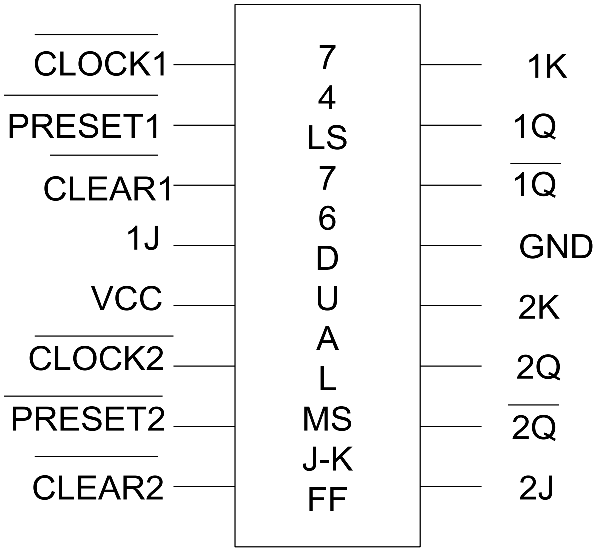
APPARATUS :

Digital-Board, GP-4 Patch-Cords, IC-74LS76, IC-74LS32, IC-74LS04/IC-74LS08 and Required Logic gates if any

THEORY :

Sequence Generator is a Sequential Logic circuits which can be use to generate the Pre-determined sequence. Sequence Generator is classified into two categories: Sequential Sequence Generator & Non-Sequential Sequence Generator. Ring Counter can be constructed using IC-74LS76.In case of Ring Counter output of last Flip-flop is connected to the J_A input of First Flip-Flop and complementary output of Last Flip-Flop is connected to K_A Input of First Flip-Flop. Output of first flip-flop (Q_A&Q_A) is connected to the inputs of second flip-flop (J_B & K_B) and so on. And connect set & reset pin to Vcc.

PIN Diagram :


$\overline{\text{CLOCK1}}$
$\overline{\text{PRESET1}}$
$\overline{\text{CLEAR1}}$
1J
VCC
$\overline{\text{CLOCK2}}$
$\overline{\text{PRESET2}}$
$\overline{\text{CLEAR2}}$
7 4 LS 7 6 DUAL MS J-K FF
1K
1Q
$\overline{\text{1Q}}$
GND
2K
2Q
$\overline{\text{2Q}}$
2J

PROCEDURE :

1. Make the connections as per the Logic circuit of Sequence generator circuit using IC-74LS76 and Verify its Truth Table.

Design of Sequence Generator circuits with Lockout condition

Dec. Equ.	PRESENT STATE			NEXT STATE			INPUT					
	Q_A	Q_B	Q_C	Q_A^+	Q_B^+	Q_C^+	J_A	K_A	J_B	K_B	J_C	K_C

K-Map Simplification for J_A , K_A , J_B , K_B , J_C , K_C

Logic diagram:

Design of Sequence Generator without Lockout Condition

Dec. Equ.	PRESENT STATE			NEXT STATE			INPUT					
	Q_A	Q_B	Q_C	Q_A^+	Q_B^+	Q_C^+	J_A	K_A	J_B	K_B	J_C	K_C

K-Map Simplification for J_A , K_A , J_B , K_B , J_C , K_C

Logic Diagram:

Logic Gates / MSI Device required for Implementation:

Sr.No.	Title	Name of the IC	Number of Gates required	IC Required
01	Sequence Generator with lockout condition			
02	Sequence Generator with lockout condition			

CONCLUSION:

REFFRENCE:

1. **R.P.Jain "Modern Digital Electronics" TMH 4th Edition**
2. **D.Leach,Malvino,Saha,"Digital Principles and Applications",TMH**

Subject teacher Sign with Date Remark

CLASS : S.E. COMP **SUBJECT : DEL**

EXPT. NO. : 9 **DATE :**

TITLE : PSEUDO RANDOM NUMBER GENERATOR CIRCUIT

OBJECTIVE :

1. Design and Implement the Pseudo Random number generator circuit using IC-74LS194 and verify its truth-table.(Use Left Shift)
2. Design and implement the Pseudo Random number generator circuit using IC-74LS76 and verify its truth-table(Use Right Shift)

APPARATUS :

Digital-Board, GP-4 Patch-Cords, IC-74LS194, IC-74LS86

THEORY :

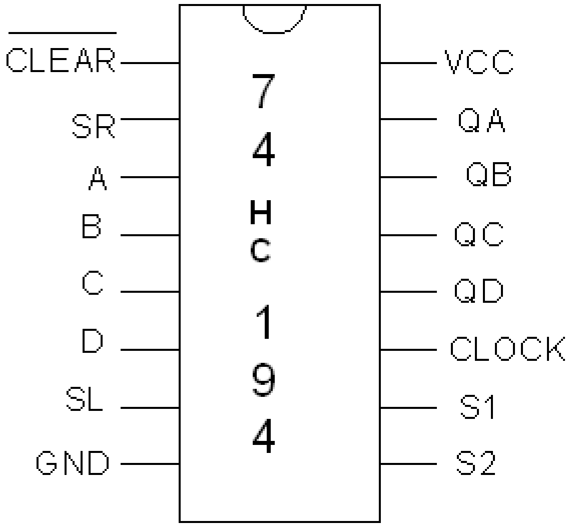
Register is a sequential logic device, which can be used to store the number of bits. Register whose internal bits can be shifted towards right and left is called as shift register. IC-74HC194 is a bi-directional universal shift register. This is called universal shift register because it performs all modes of operations of shift register. **IC-74HC194 provides four different modes of operation:**

1) Serial in Serial Out. (SISO) 2) Serial in Parallel out (SIPO)

3) Parallel In Serial out (PISO) 4) Parallel in Parallel out (PIPO)

IC-74HC194 can be used to implement pulse train generator, ring counter, twisted ring counter, Random number generator circuit etc.

PIN Diagram :



PROCEDURE :

1. Make the connections as per the Logic circuit of Pseudo Random number generator circuit using Left shift and Verify its Truth Table.
2. Make the connections as per the Logic circuit of Pseudo Random number generator circuit using Right shift and Verify its Truth Table.

Design of Pseudo Random Number Generator Circuit using Left Shift

Output				Decimal Equivalent
Q_A	Q_B	Q_C	Q_D	

Logic diagram

Design of Pseudo Random Number Generator Circuit using Right Shift

Output				Decimal Equivalent
Q_A	Q_B	Q_C	Q_D	

Logic diagram:

Logic Gates / MSI Device required for Implementation:

Sr.No.	Title	Name of the IC	Number of Gates required	IC Required
01	Pseudo Random Number generator circuit using Left and Right shift			

CONCLUSION:

REFFRENCE:

1. R.P.Jain “Modern Digital Electronics” TMH 4th Edition
2. D.Leach,Malvino,Saha,”Digital Principles and Applications”,TMH

Subject teacher Sign with Date   Remark

CLASS : S.E. COMP **SUBJECT : DEL**

EXPT. NO. : 10 **DATE :**

TITLE : ALGORITHMIC STATE MACHINE

OBJECTIVE :

1. Design and Implement waveform generator circuit using ASM based Multiplexer controller method

APPARATUS :

Digital-Board, GP-4Patch-Cords, IC-74LS74, IC74153,IC-74LS32, IC-74LS08 ,IC-74LS04 and Required Logic gates if any.

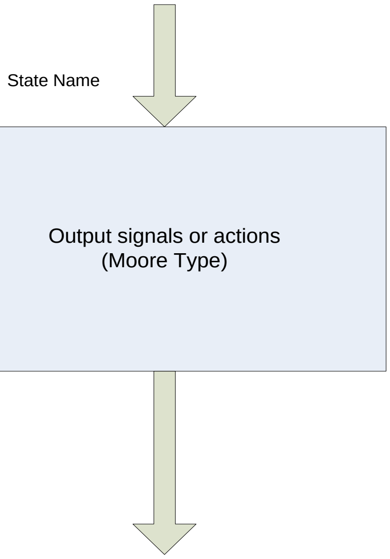
THEORY :

1. ASM means algorithmic state machine.
2. It is a type of flowchart that can be used to represent the state transitions and generated outputs for finite state machine(FSM)
3. ASM charts are similar to traditional flowcharts.
4. Unlike a traditional flowchart, this includes timing information. This chart specifies that the FSM flows from one state to another only after each active clock edge.

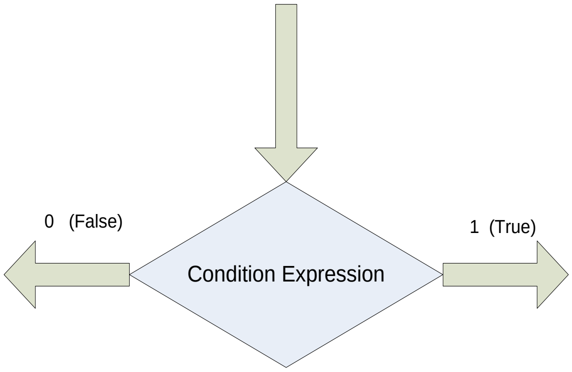
Basic elements used in ASM Chart:

1. State Box
2. Decision Box
3. Conditional output Box

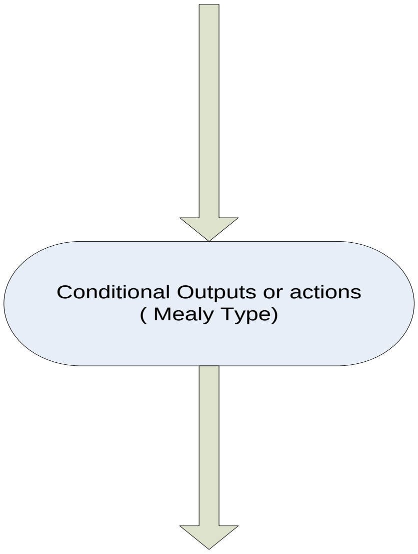
These are shown in the figure given below:



State Box



Decision Box



Conditional output box

State Box:

A rectangle represents a state of the FSM.It is equivalent to node in the state diagram or row in the state table. The name of the state should be indicated outside the box in left top corner. Moore types of outputs are listed inside the box.

Decision Box:

A diamond indicates that the stated condition expression has to be tested and an exit path has to be chosen accordingly. The condition expression Consists of one or many inputs.

Conditional output Box:

The oval denotes the output signals that are of Mealy type. These output depend on the values of state variables and the Inputs of FSM.The condition that determines whether such outputs are generated is specified in a decision box

Significance:

It is an aid to design the complex circuits.ASM charts are used to describe complex circuits that include one or more FSM's and another circuitry such as registers, counters, adders, multipliers etc.

ASM Blocks:

1. It is a structure which consists of single state box and any decision and conditional output boxes that the state box may be connected to.
2. It has one entry path and any number of exit paths.
3. Each block describes the state of the system during the interval of one clock pulse.

PROCEDURE :

1. Make the connections as per the Logic circuit of waveform generator circuit using and Verify its functionality.

Design of Waveform generator using multiplexer controller method

Timing Diagram:

1. Waveform generator has 4 states i.e. 00,01,10,11.
2. In the waveform generator, circuit always goes to next state irrespective of values of X1 and X2.
3. Output of each state is observed according to the input combinations X1X2.

ASM Chart:

State Transition Table:

Dec. Equ.	PRESENT STATE		Ext. I/Ps		NEXT STATE		MUX. OUTPUT		Final O/P
	X1	X2	(S_1) Q_A	(S_0) Q_B	(D_A) Q_A^+	(D_B) Q_B^+	M1	M2	Z
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	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					

K- Map Simplification for Z:

Logic Diagram:

Logic gates/MSI Device required for Implementation

Sr.No.	Title	Name of the IC	Number of Gates required	IC Required
01	Waveform Generator Circuit using Mux.controller method			

CONCLUSION:

____

____

____

____

____

____

____

REFFRENCE:

1. R.P.Jain "Modern Digital Electronics" TMH 4th Edition
2. D.Leach,Malvino,Saha,"Digital Principles and Applications",TMH

Subject teacher Sign with Date Remark

CLASS : S.E. COMP **SUBJECT : DEL**

EXPT. NO. : 11 **DATE :**

TITLE : COMBINATIONAL LOGIC DESIGN USING VHDL

OBJECTIVE :

1. Design and simulation of 4:1 Multiplexer circuit using all modeling style
2. Design and simulation of Full adder circuit using all modeling style

SOFTWARE : Modelsim version 6.5

THEORY :

VHDL is very high speed integrated circuit hardware description language. It is a widely used language that describes the behavior of a digital system. The language has constructs that enable to express the concurrent or the sequential behavior of digital systems with or without timing.

Features of VHDL:

1. VHDL can be used at different complexity levels from single logic gate to complete system in the same simulation environment.
2. VHDL supports flexible design methodologies like top-down, bottom-up or mixed.
3. It supports both synchronous and asynchronous timing models.

VHDL code is composed of at least 3 fundamental sections:

1. Library declarations
2. Entity
3. Architecture

Library Declaration: It is a collection of commonly used codes which can be shared by other designs as well. The code is usually written in the form of functions, procedures or components which are placed inside the packages and then compiled into the destination library.

```
Syntax: Library library_name;
            Use library_name. package_name.package ports;
```

Three following libraries are used:
IEEE, Standard and work.

As standard and work libraries are included by default, library IEEE should be declared in a program.
The library, package and ports used and declared in the program are:

```
Librart IEEE;
Use IEEE.std_logic_1164.all;
```

Entity Declaration: It is an interface between a design and the external environment. An entity gives the list and specifications of all the inputs and outputs in the design. A design may contain multiple entities and each entity has architecture.

```
Syntax: entity entity_name is
            Port( port_name1: mode signal_type;
                    Port_name2: mode signal_type);
              End entity_name;
```

Architecture Declaration: The architecture describes the underlying functionality or internal organization or operation of the entity. A single entity can have multiple architectures.

Every architecture has two parts- Declaration (optional) and code.

```
Syntax: architecture architecture_name of entity_name is
            Declarations if any
            Begin
            ---------
           End architecture_name;
```

Object types:

- Signal: It represents the circuit interconnections. It is used to pass the values in and out of the component or a circuit.
  Syntax: signal signal_name:signal_type;
- Variable: It is used to represent the local information.
  Syntax: variable variable_name:variable_type;
- Constant: It is used to assign default values.
  Syntax: constant constant_name:constant_type:=value;

Data types:

- Bit: It is a two level logic assigned to a constant or variable.
- Bit_vector(x downto 0): It represents 8-bit vector with leftmost bit as MSB.
- Bit_vector(0 to x): It represents 8-bit vector with rightmost bit as MSB.

Types of modeling:

- Dataflow modeling
- Structural modeling
- Behavioral modeling
- Mixed modeling

Data Flow Modeling:

It uses set of concurrent assignment statements.

The data is viewed as flowing through the design from input to output.

Each of the statements can be activated when any of its input changes its value.

4:1 multiplexer:

4 inputs: I0, I1, I2, I3
1 output: Y
2 select lines: S1, S0

The circuit can be described by the following equation:

Y= not (S1) . not (S0). I0 + not (S1). S0.I1 + S1.not(S0). I2 + S1. S0. I3

Structural Modeling:
A digital circuit can be described with the help of small modules where outputs are functions of the inputs.
In this modeling, set of interconnected components are used which can be used to create a very low level description of a circuit.
Components that build the circuit are declared and used in the architecture of the design.

Component declaration:

It is a virtual design entity which needs to be declared in the architecture. The component is the basic circuit which builds the complete design.

```
Syntax: Component component_name
            Port(port_name1:mode port_type;
                   Port_name2:mode port_type);
            End component_name;
```

Syntax for port map: Port map is used for mapping the intermediate signals.

Signal: entity_name port map(variables to port map with)

PROCEDURE :

- Create a project in modelsim.
- Create a VHDL Source file.
- Library Declaration
- Entity Declaration
- Architecture Declaration
- Code writing using different modeling style
- End

CONCLUSION:

____

____

____

____

REFFRENCE:

1. **R.P.Jain "Modern Digital Electronics" TMH 4th Edition**
2. **J.Bhaskar " VHDL Primer S"Prentice Hall 3rd Edition**
3. **Douglas Perry" VHDL Programming by Example"TMH 4th Edition**

Subject teacher Sign with Date Remark

CLASS : S.E. COMP **SUBJECT : DEL**

EXPT. NO. : 11 **DATE :**

TITLE : SEQUENTIAL LOGIC DESIGN USING VHDL

OBJECTIVE :

1. Design and simulation of 4-bit Asynchronous UP/DOWN Counter using mode control Input (M) (Use Behavioral Modeling style)
   When M=0;Circuit perform UP-Counting operation
   When M=1;Circuit perform DOWN-Counting operation

2. Design and simulation of 4-bit Asynchronous UP/DOWN Counter using mode control Input (M) using T-Flip-flop as a Component
3. Design and simulation of 4-bit synchronous UP/DOWN Counter using mode control input(M) using T-Flip-flop as a Component

SOFTWARE : Modelsim version 6.5

THEORY : Counter is a sequential logic design which can be used to count the number of clock pulses given to the circuit. Counter can be classified into two categories: Asynchronous and Synchronous. Asynchronous counter output of the first flip-flop goes to the clock input of next flip-flop and so on. Inputs of all flipflop are connected to Logic 1 (VCC) because counter is a frequency divider application and being a suitable flip-flop is a T-flip-flop.

Asynchronous counter is is easy to design as compared to Synchronous counter circuit but Synchronous counter is faster than the Asynchronous counter.

PROCEDURE :

- Create a project in modelsim.
- Create a file.
- Library Declaration
- Entity Declaration
- Architecture Declaration
- Code writing using Behavioral / Structural modeling style
- End

CONCLUSION:

____

____

____

____

REFFRENCE:

1. **R.P.Jain "Modern Digital Electronics" TMH 4th Edition**
2. **J.Bhaskar " VHDL Primer S"Prentice Hall 3rd Edition**
3. **Douglas Perry" VHDL Programming by Example"TMH 4th Edition**

Subject teacher Sign with Date Remark