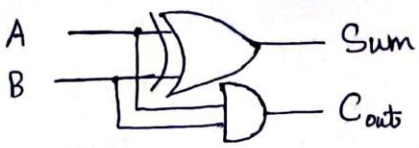
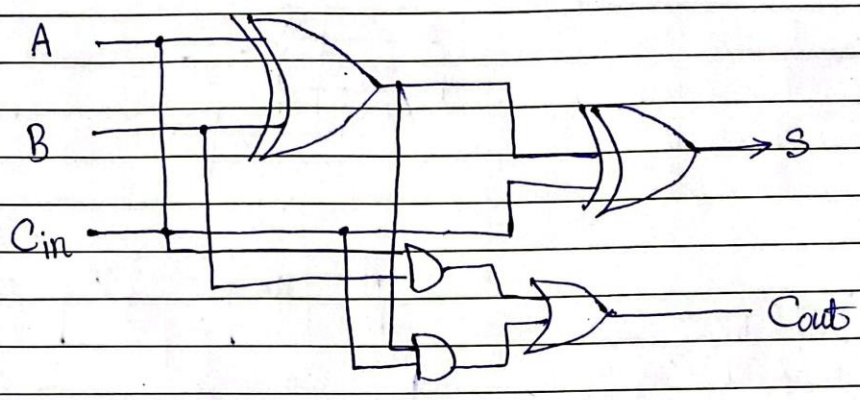
Half Adder:



Date: _____

Q: Full Adder



$(A \oplus B) \oplus C_{in}$ | $(A \cdot B) + (A \oplus B) C_{in}$

A	B	C_{in}	S	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$C_{out} = \bar{A}BC_{in} + A\bar{B}C_{in} + AB\bar{C}_{in} + ABC_{in}$

$= \bar{A}B(\bar{C}_{in} + C_{in}) + AB(\bar{A}B + A\bar{B})$

$+ AB(\bar{C}_{in} + C_{in})$

$= C_{in}(A \oplus B) + AB$

Min terms (Sum) $= \bar{A}\bar{B}C_{in} + \bar{A}B\bar{C}_{in} + A\bar{B}\bar{C}_{in} + ABC_{in}$

$= C_{in}(\bar{A}\bar{B} + AB) + \bar{C}_{in}(\bar{A}B + A\bar{B})$

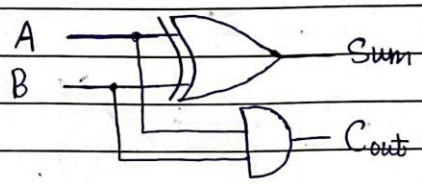
$= C_{in}(\overline{A \oplus B}) + \bar{C}_{in}(A \oplus B)$

$= C_{in} \oplus (A \oplus B)$

Date: _____

Half Adder : $\bar{A}B + A\bar{B} = A \oplus B$

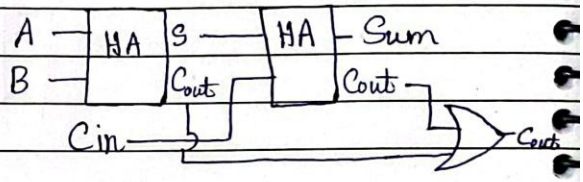
A	B	Σ	Cout
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

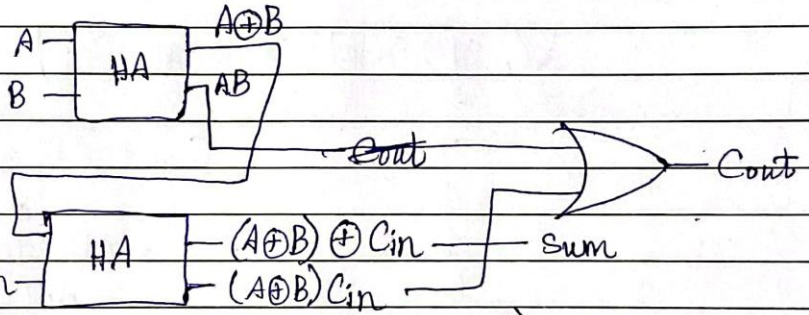


Full Adder :

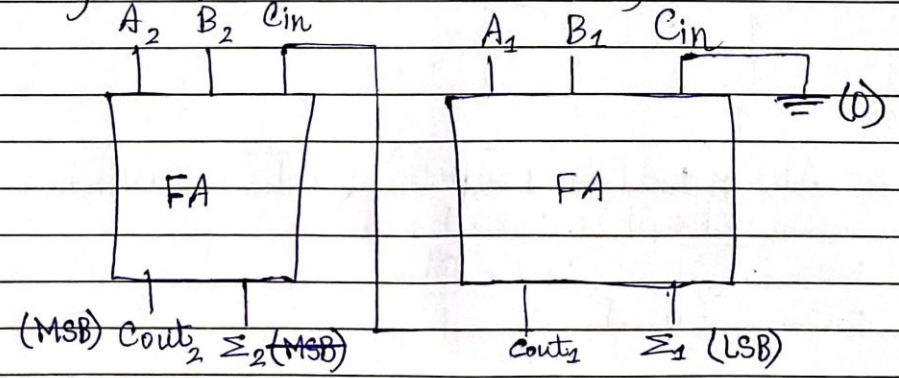
$\Sigma = (A \oplus B) \oplus C_{in}$

$C_{out} = AB + (A \oplus B) C_{in}$





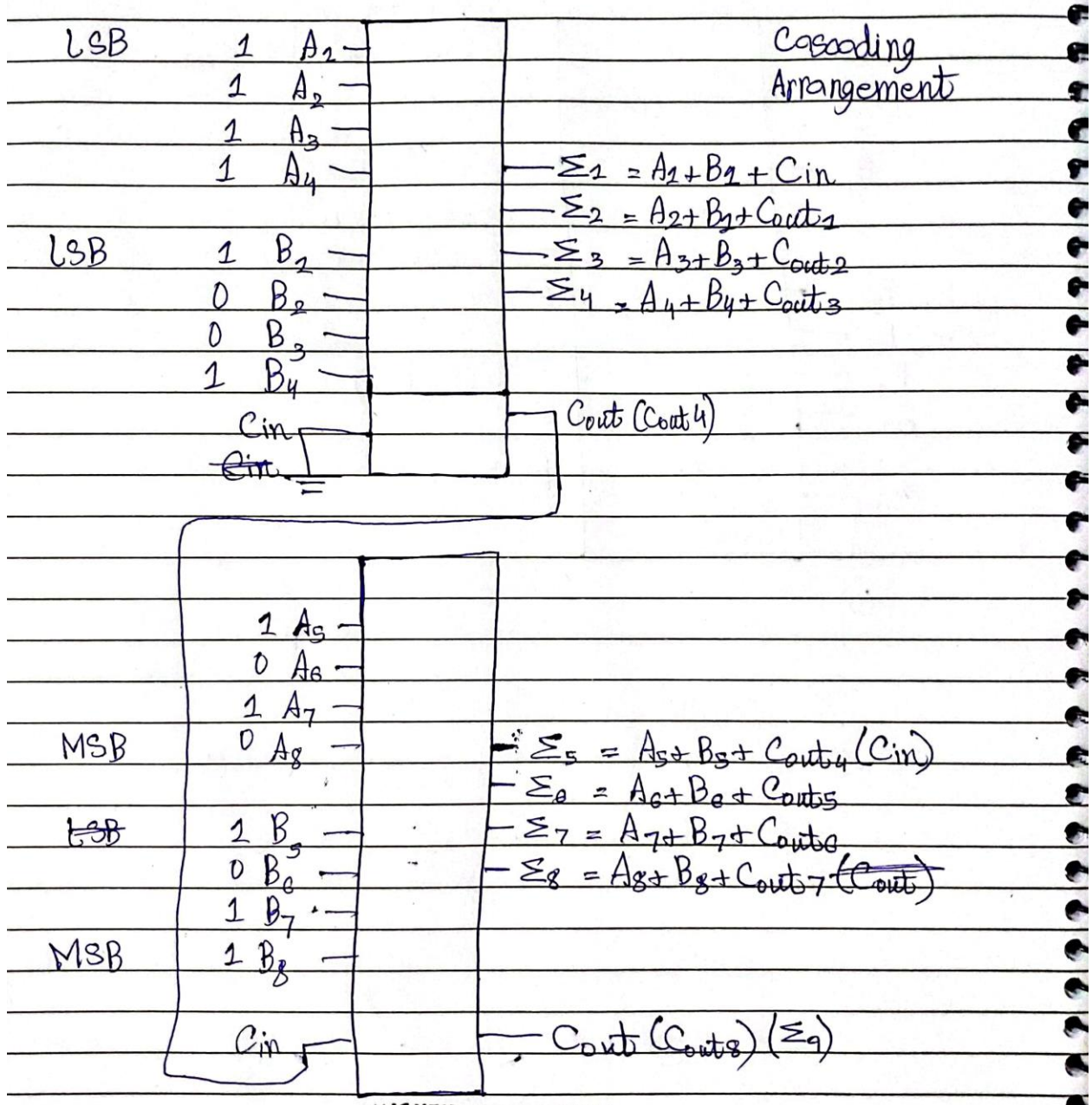
Cascading (When more than 1 IC used) :



Adding binary numbers with more than one bits

Date: _____

A = 01011111 + B = 11011001

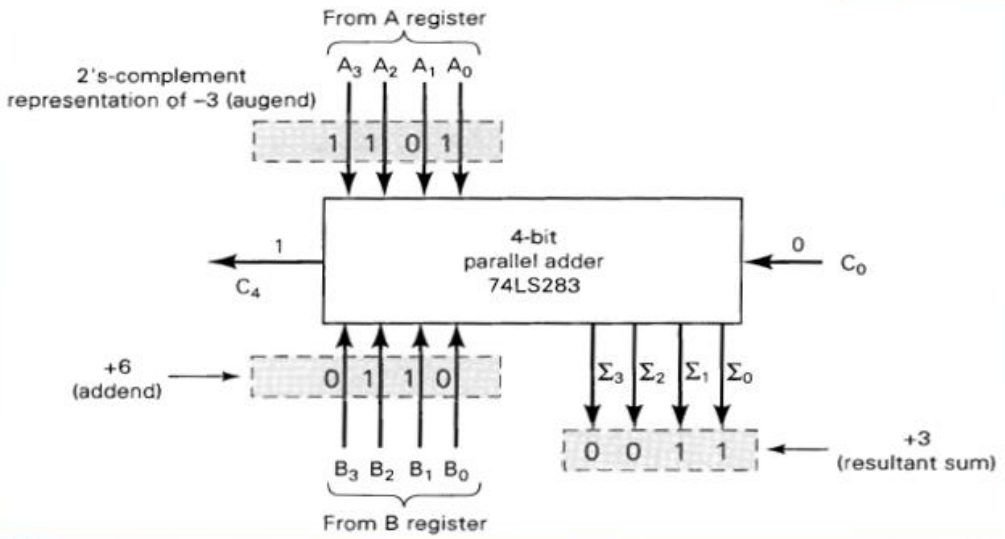


2's-COMPLEMENT SYSTEM

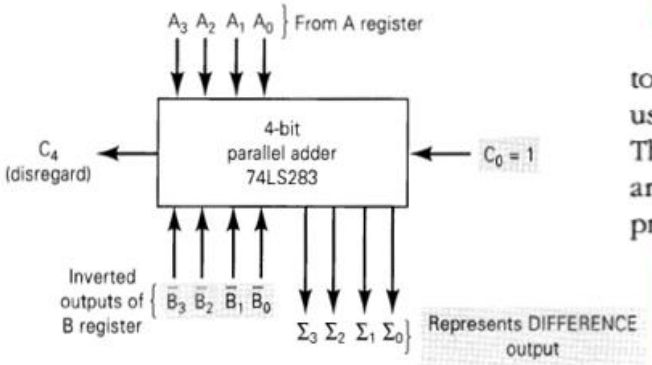
Addition

Positive and negative numbers, including the sign bits, can be added together in the basic parallel-adder circuit when the negative numbers are in 2's-complement form. for the addition of −3 and +6. The −3 is represented in its 2's-complement form as 1101

Parallel adder used to add + and − numbers in 2's-complement system.



Subtraction



Parallel adder used to perform subtraction ($A - B$) using the 2's-complement system. The bits of the subtrahend (B) are inverted, and $C_0 = 1$ to produce the 2's complement.

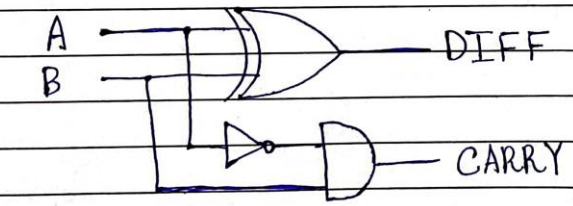
1. +4 is stored in the A register as 0100.
2. +6 is stored in the B register as 0110.
3. The inverted outputs of the B-register FFs (1001) are fed to the adder.
4. The parallel-adder circuitry adds $[A] = 0100$ to $[\overline{B}] = 1001$ along with a carry, $C_0 = 1$, into the LSB. The operation is shown below.

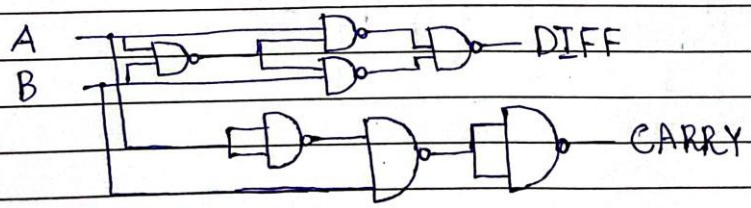
$$\begin{array}{rl} 1 & \leftarrow C_0 \\ 0100 & \leftarrow [A] \\ + \underline{1001} & \leftarrow [\overline{B}] \\ 1110 & \leftarrow [\Sigma] = [A] - [B] \end{array}$$

Write the function table for a half subtractor (input A and B, output DIFF and CARRY). From the function table, design two logic circuits that will act as half subtractor.

9.

A	B	DIFF	CARRY (BORROW)
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0





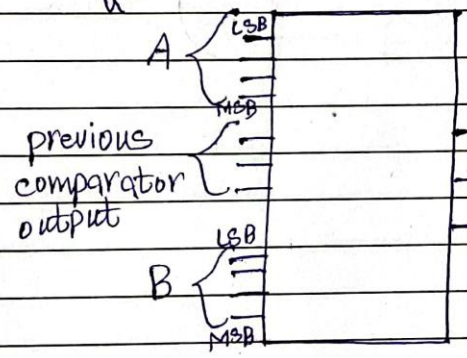
Date: _____

Comparator:

A	B	A<B	A=B	A>B
0	0	0	1	0
0	1	1	0	0
1	0	0	0	1
1	1	0	1	0

$\bar{A}B$ $\quad$ $\bar{A}\bar{B}+AB$ $\quad$ $A\bar{B}$

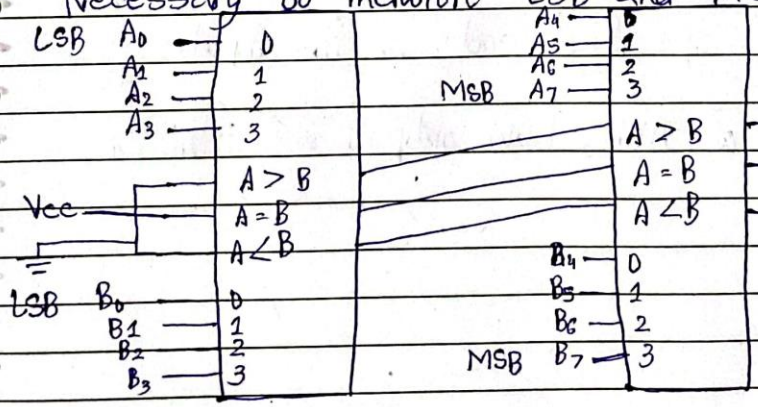
4 bit numbers



Cascading for larger than 4 bits

(MSB compared of both numbers (A_3 and B_3)) If both equal then next lower bit compared

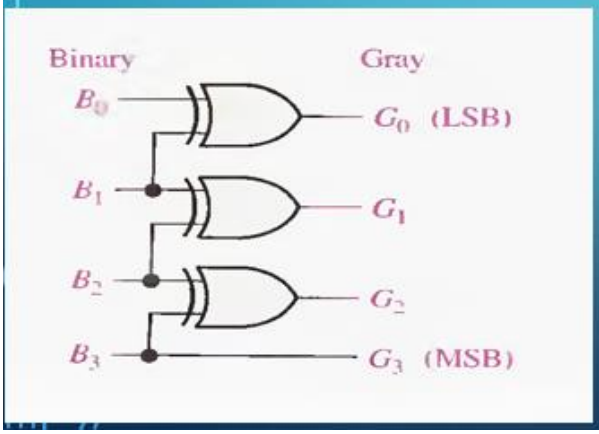
Necessary to mention LSB and MSB



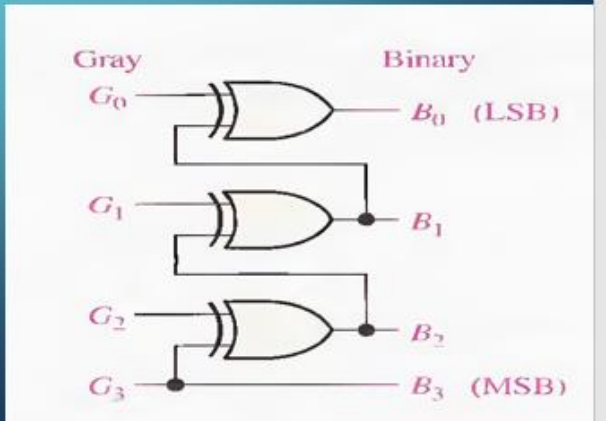
Binary-to-Gray and Gray-to-Binary Conversion

There are various code converters that change one code to another. Two examples are the four bit binary-to-Gray converter and the Gray-to-binary converter.

Four-bit binary-to-Gray conversion logic.



Four-bit Gray-to-binary conversion logic.



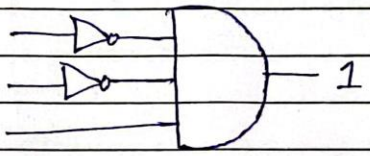
Date: _____

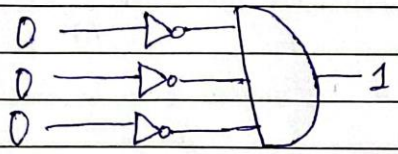
Decoder

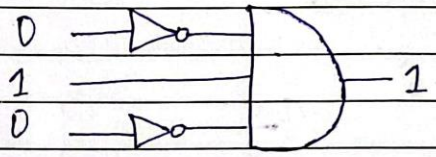
Binary to decimal

Active High

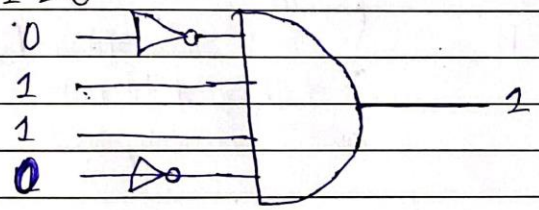
A	B	C	X	0	1	2	0
0	0	0	0	1			0
0	0	1	1	1			1
0	1	0	2	1			







0120



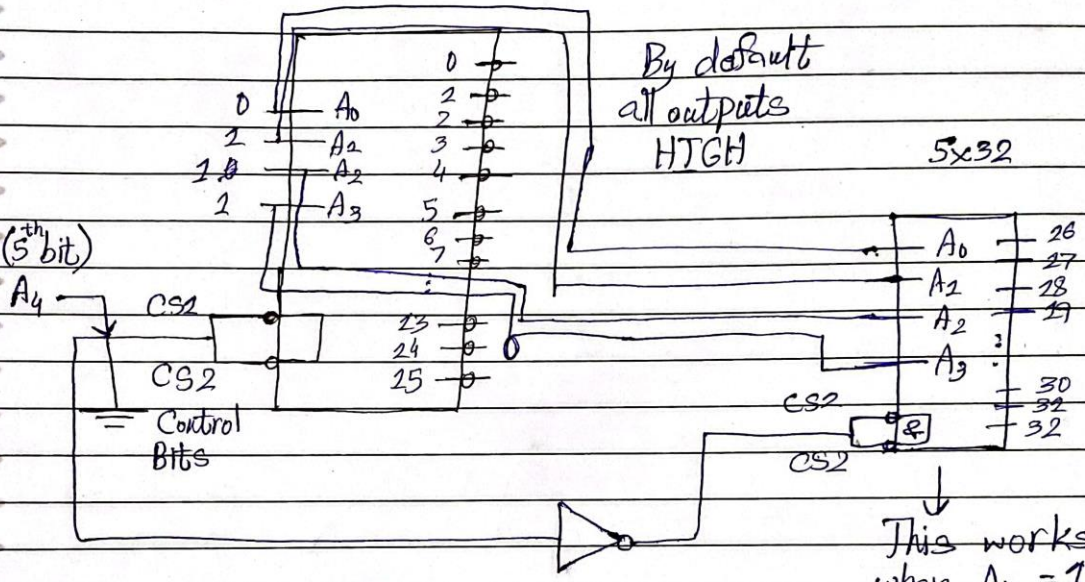
Indicates presence of specified code on its inputs

IF NAND used instead of AND, LOW outputs will indicate presence of code.

Date: _____

4-Bit Decoder: inputs combinations

4x16

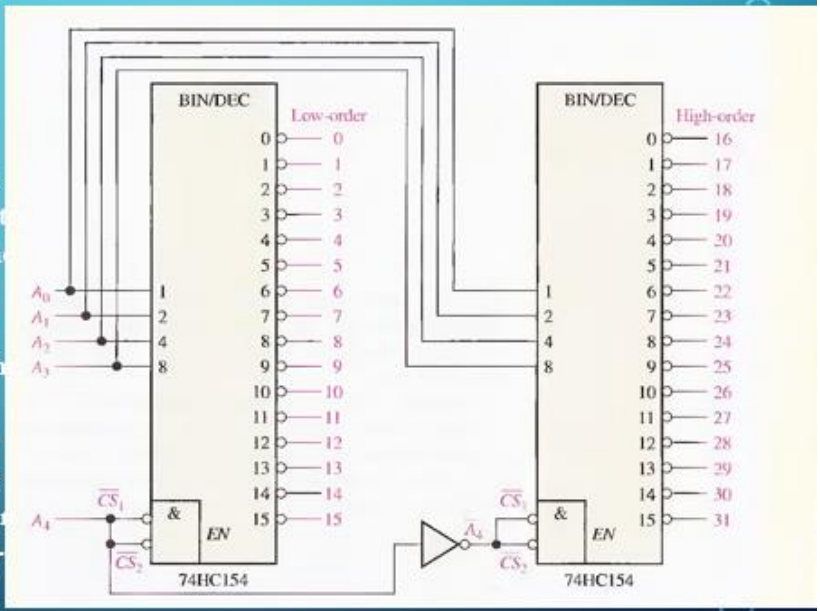


This works when $A_4 = 1$, otherwise first IC works

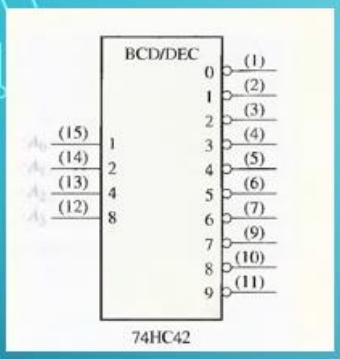
EXAMPLE

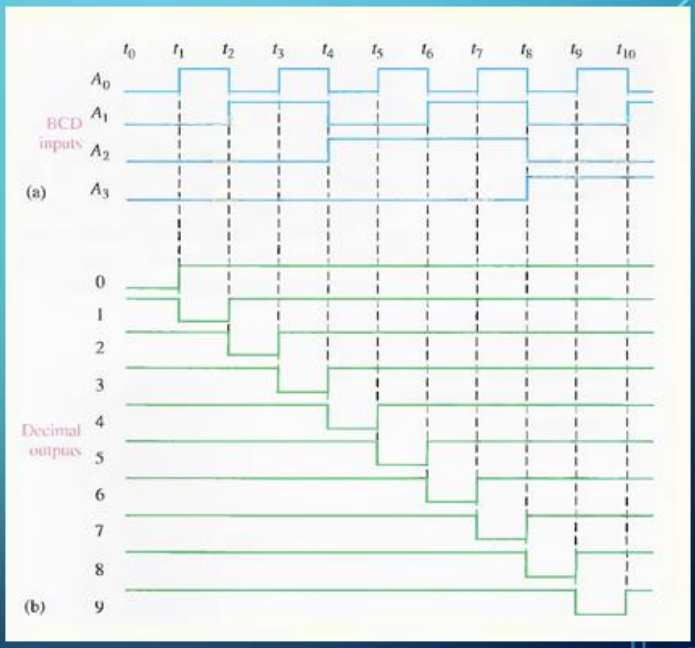
A certain application requires that a 5-bit number be decoded. Use 74HC154 decoders to implement the logic. The binary number is represented by the format $A_4A_3A_2A_1A_0$.

Since the 74HC154 can handle only four bits, two decoders must be use d to decode five bits. The fifth bit, A 4 , is connected to the chip select inputs, CSl' and CS 2' , of one decoder, and A4 is connected t the CS l'and CS 2' inputs of th other decoder, as shown in Figure . When the decimal number is 15 or less, A4 = 0, th low-order decoder is enabled, and the high-order decoder is disabled. When the decimal number is greater tha 15, A4 = I so A4' = 0, the high-order decoder is enabled. and the low -order decoder is disabled.



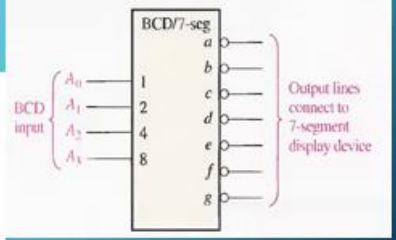
The 74HC42 is an integrated circuit BCD-to-decimal decoder.





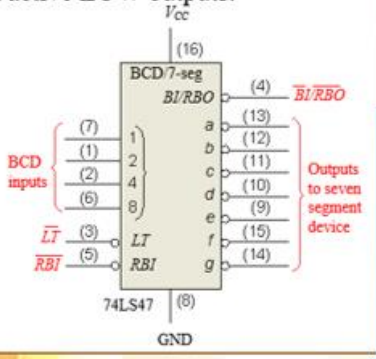
The BCD-to-7-Segment Decoder

The BCD-to-7-segment decoder accepts the BCD code on its inputs and provides outputs to drive 7-segment display devices to produce a decimal readout. The logic diagram for a basic 7-segment decoder is shown in Figure 6–34.



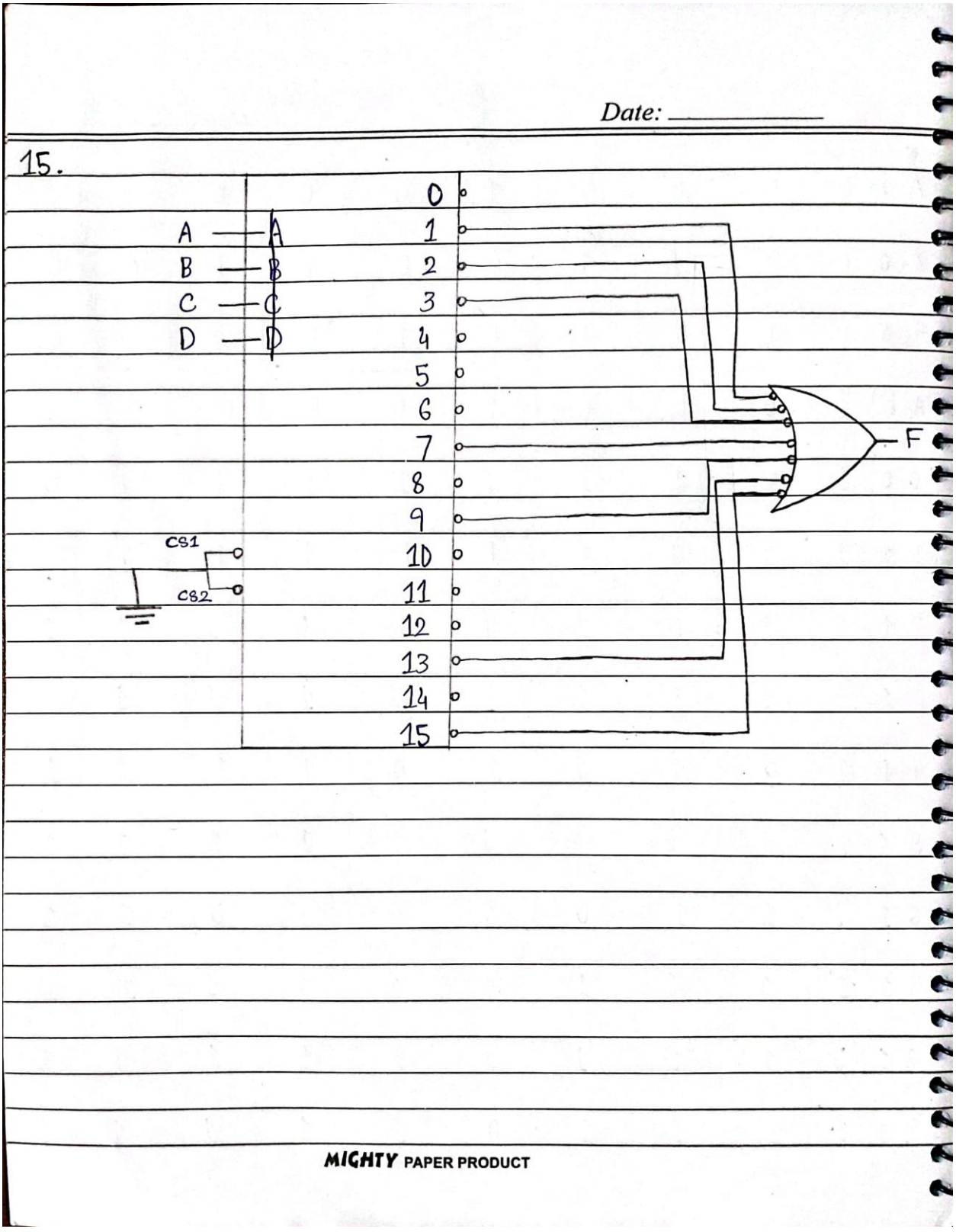
Another useful decoder is the 74LS47. This is a BCD-to-seven segment display with active LOW outputs.

The *a*-*g* outputs are designed for much higher current than most devices (hence the word driver in the name).



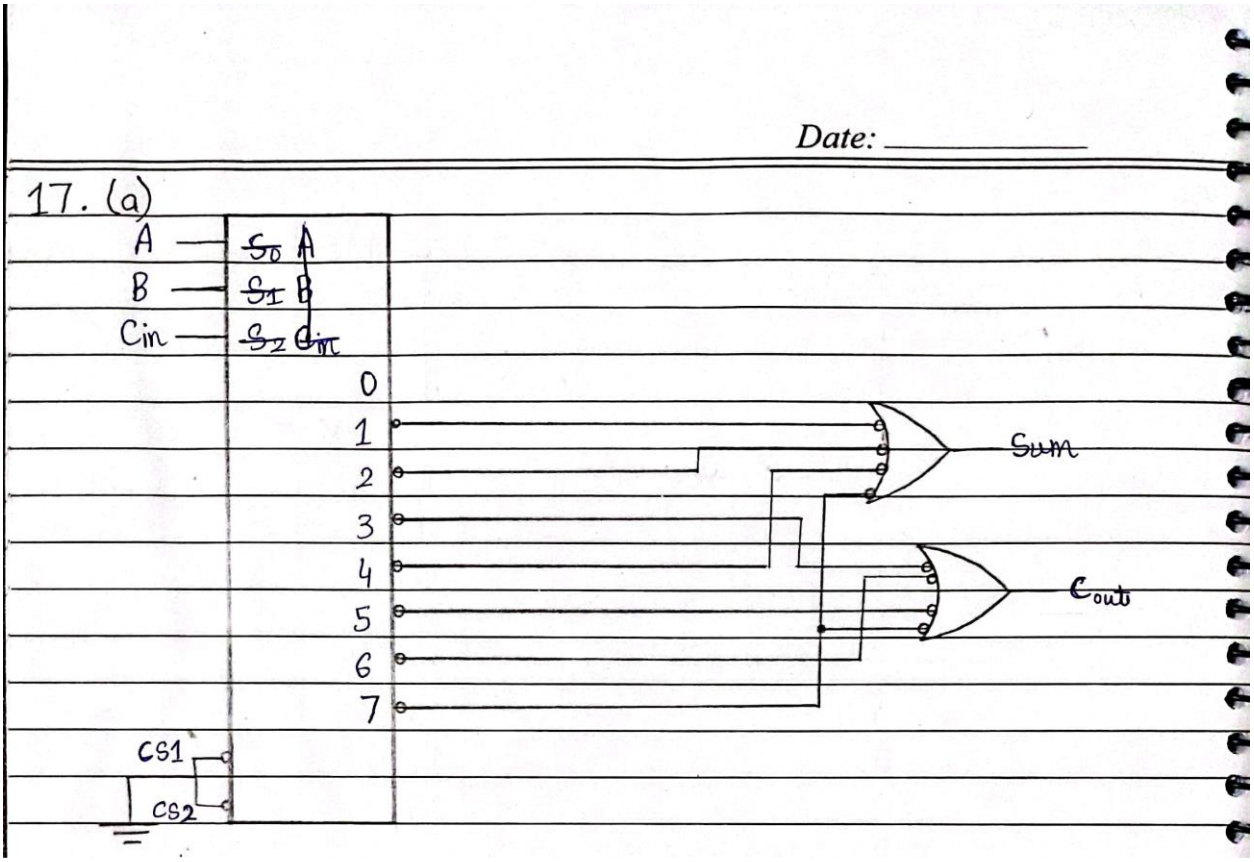
15. Implement the following Boolean function using the decoder.

F(A, B, C,D) = Σ (1, 2, 3, 7,9,13,15)



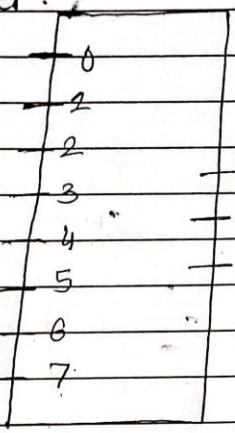
17. Implement a full adder circuit by using:

(a)3 – to - 8 line Decoder

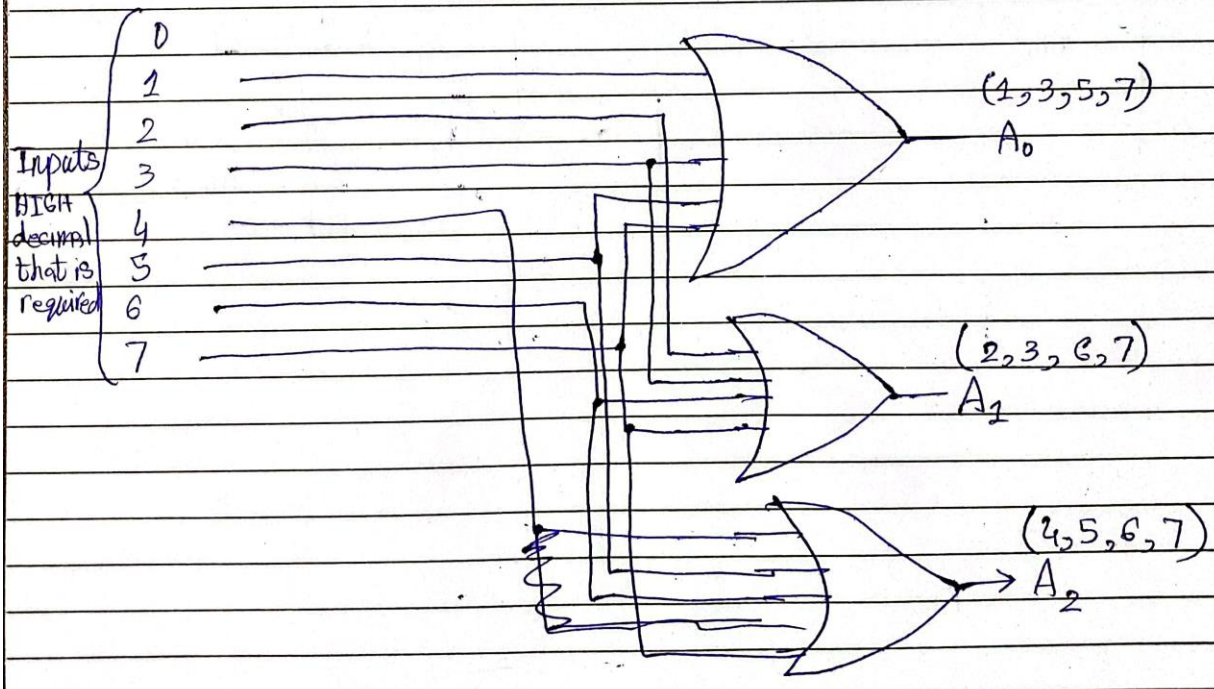


Date: _____

Encoder:



A_{2}	A_{1}	A_{0}	X
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	



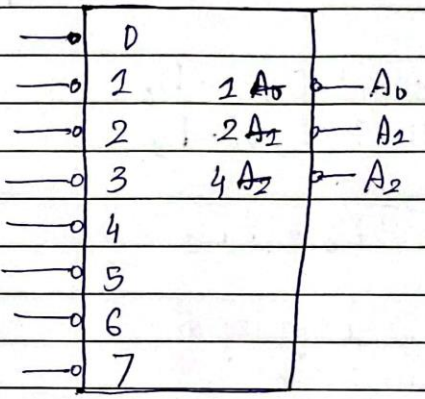
Date: _____

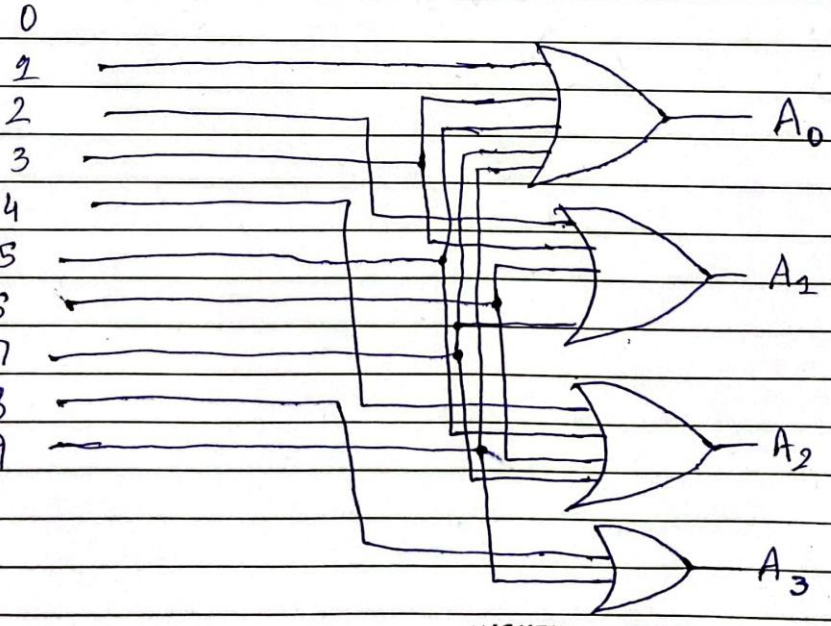
BCD To ~~Decimal~~ Decimal to BCD:

~~A_3 A_2 A_1 A_0 Decim~~

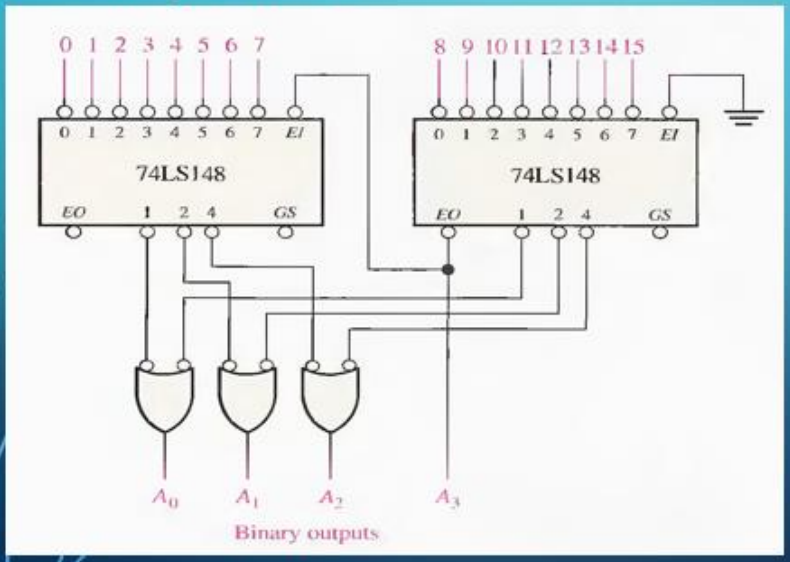
A_3	A_2	A_1	A_0	Decimal
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9

8-3 Encoder





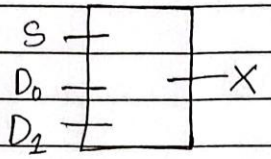
The 74LS148 can be expanded to a 16-line-to-4-line encoder by connecting the EO of the higher-order encoder to the EI of the lower-order encoder and negative-ORing the corresponding binary outputs as shown in Figure . The EO is used as the fourth and MSB. This particular configuration produces active-HIGH outputs for the 4-bit binary number.



A 16-line-to-4 line encoder using 74LS148s and external logic.

Date: _____

Multiplexer (MUX):



When $S=0$, $X=D_0$
When $S=1$, $X=D_1$

$X = \bar{S}D_0 + SD_1$

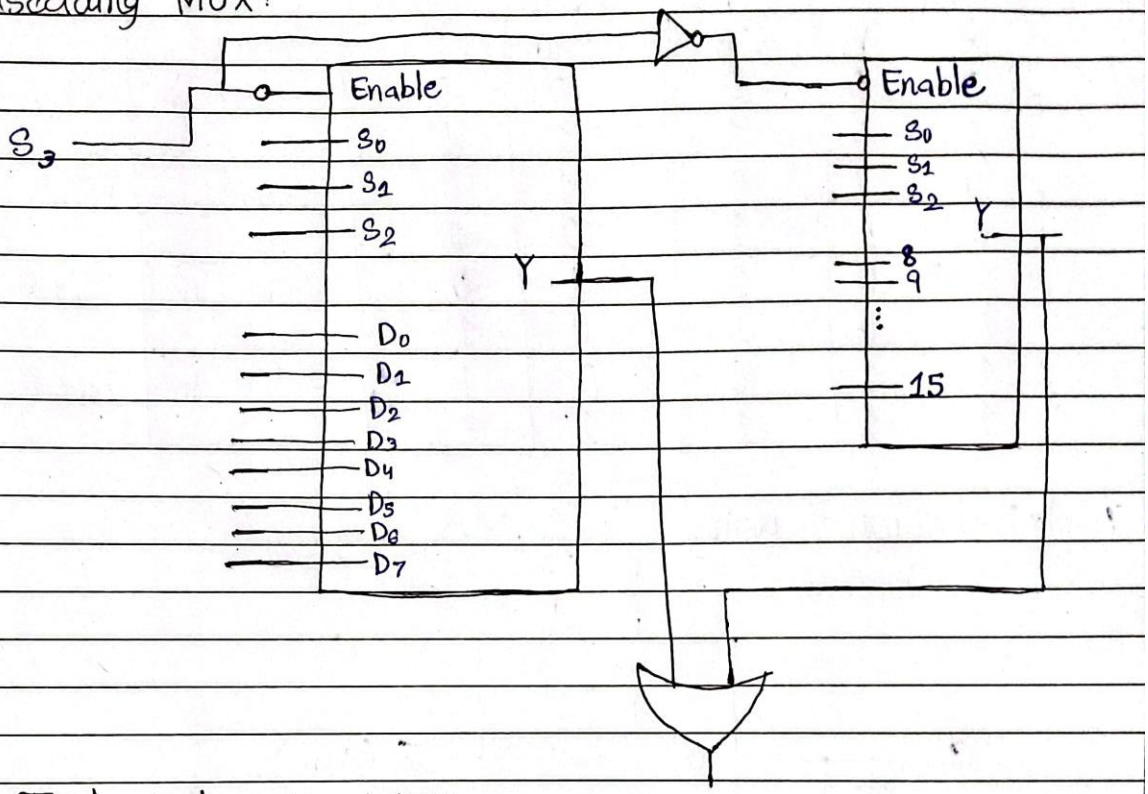
If 4-bit input

S_1	S_0	X
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3

$X = \bar{S}_1\bar{S}_0 D_0 + \bar{S}_1 S_0 D_1 + S_1\bar{S}_0 D_2 + S_1 S_0 D_3$

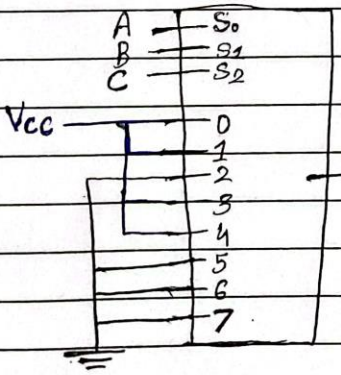
Date: _____

Cascading MUX:



Q: Implement using MUX

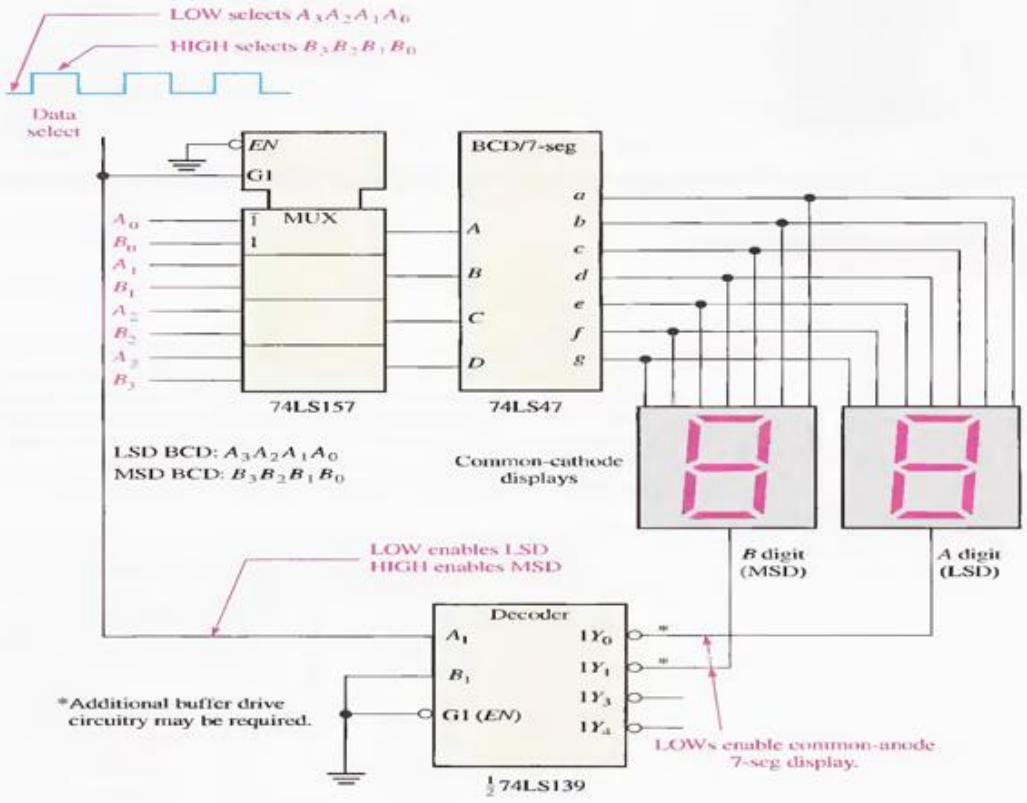
$f(A, B, C) = \Sigma(0, 1, 3, 4)$



$X = D_0 \bar{A}\bar{B}\bar{C} + D_1 \bar{A}\bar{B}C + D_3 \bar{A}BC + D_4 A\bar{B}\bar{C}$

Applications

A 7-Segment Display Multiplexer

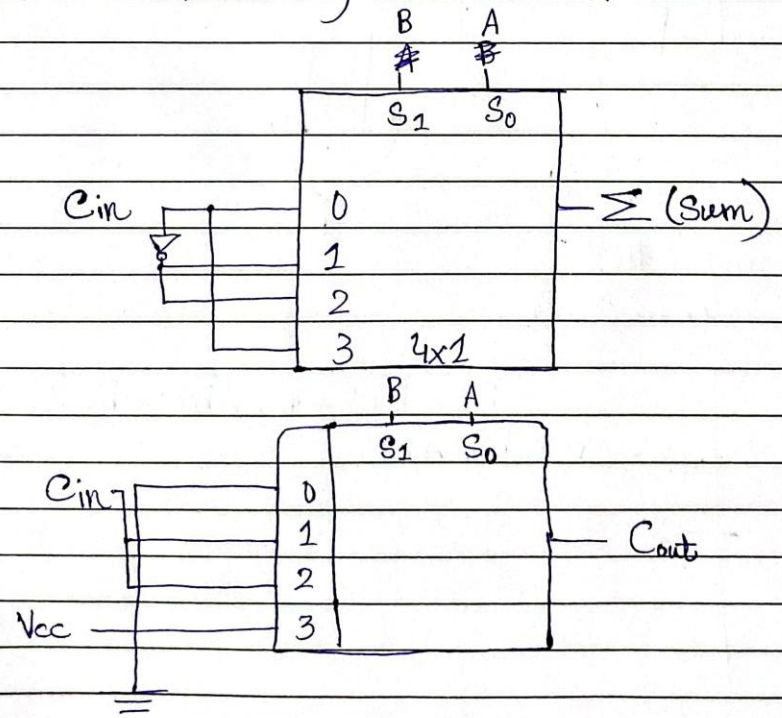


Date: _____

Full Adder:

	A	B	Cin	Sum	Cout		
0	0	0	0	0	0	Sum = Cin	Cout = 0
	0	0	1	1	0		
1	0	1	0	1	0	Sum = $\overline{Cin}$	Cout = Cin
	0	1	1	0	1		
2	1	0	0	1	0	Sum = $\overline{Cin}$	Cout = Cin
	1	0	1	0	1		
3	1	1	0	0	1	Sum = Cin	Cout = 1
	1	1	1	1	1		

Implementation using 4x1 Mux

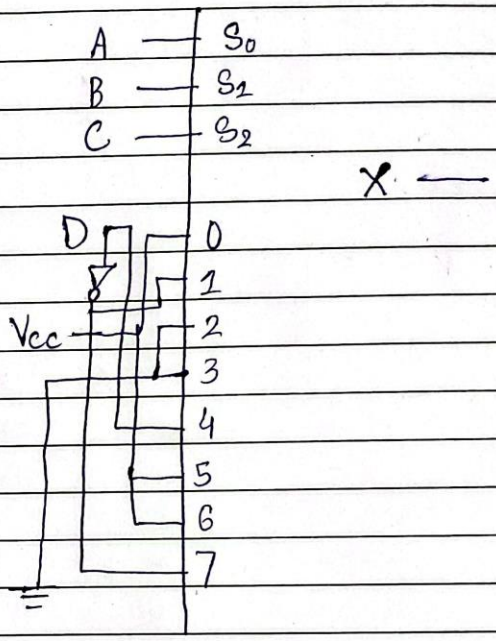


Date: _____

4-Bit Implementation Using Mux: // if restricted to use 8x1 only

Q: $f(A,B,C,D) = \Sigma(0,1,2,9,10,11,12,13,14)$

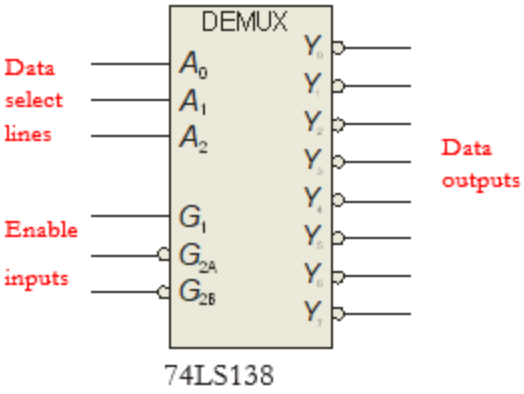
	A	B	C	D	X	
0	0	0	0	0	1	X = 1
	0	0	0	1	1	
1	0	0	1	0	1	$X = \bar{D}$
	0	0	1	1	0	
2	0	1	0	0	0	X = 0
	0	1	0	1	0	
3	0	1	1	0	0	X = 0
	0	1	1	1	0	
4	1	0	0	0	0	X = D
	1	0	0	1	1	
5	1	0	1	0	1	X = 1
	1	0	1	1	1	
6	1	1	0	0	1	X = 1
	1	1	0	1	1	
7	1	1	1	0	1	$X = \bar{D}$
	1	1	1	1	0	



Demultiplexers

A demultiplexer (DEMUX) performs the opposite function from a MUX. It switches data from one input line to two or more data lines depending on the select inputs.

The 74LS138 was introduced previously as a decoder but can also serve as a DEMUX. When connected as a DEMUX, data is applied to one of the enable inputs, and routed to the selected output line depending on the select variables. Note that the outputs are active-LOW as illustrated in the following example…



Section 6–9 Demultiplexers

31. Develop the total timing diagram (inputs and outputs) for a 74HC154 used in a demultiplexing application in which the inputs are as follows: The data-select inputs are repetitively sequenced through a straight binary count beginning with 0000, and the data input is a serial data stream carrying BCD data representing the decimal number 2468. The least significant digit (8) is first in the sequence, with its LSB first, and it should appear in the first 4-bit positions of the output.