Extra

Exercise 6.1: Determine the sum ($\sum$) and the output carry (C_{out}) of a half adder for each set of input bits:

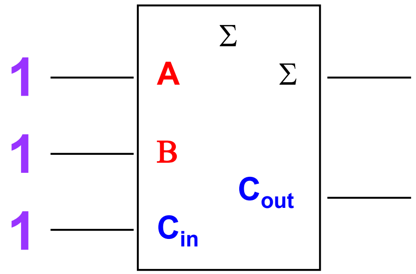
Solution:

$$\sum = A \oplus B \qquad C_{out} = AB$$

	Input, A	Input, B	Sum ($\sum$)	Output carry (C_{out})
i)	0	1	1	0
ii)	0	0	0	0
iii)	1	0	1	0
iv)	1	1	0	1

Exercise 6.2: A full adder has $C_{in}=1$. What are the sum ($\sum$) and the output carry (C_{out}) when $A=1$ and $B=1$?

Solution 6.2:



$$\sum = (A \oplus B) \oplus C_{in} \quad = 1$$

$$C_{out} = AB + (A + B)C_{in} \quad = 1$$

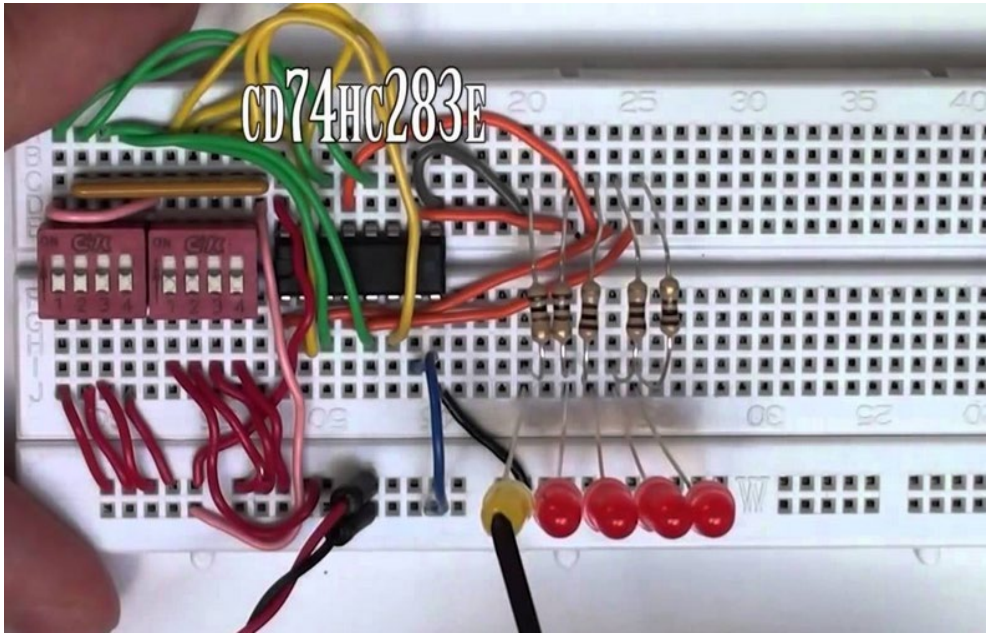
A = 1, B = 1, and C_{in} = 1

1 + 1 + 1 = 1 with carry 1

$\sum$ = 1, C_{out} = 1

Parallel Adder

Extra



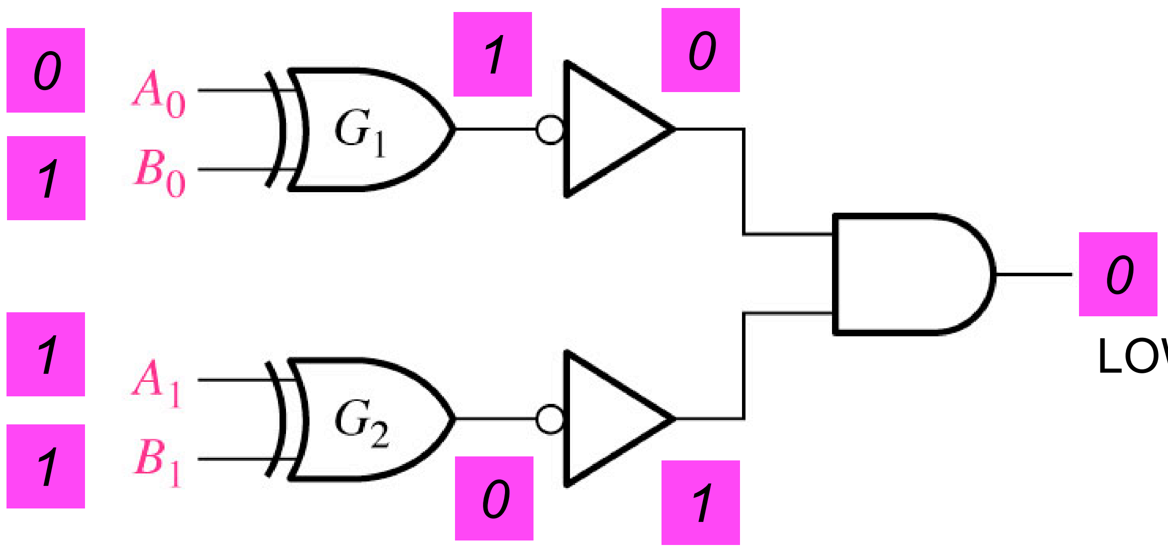
MSI chip: (74LS283)
4-bit parallel adder

Comparator

Extra

Example: Compare the equality for these 2-bits binary A=10_2 and B=11_2

General format: Binary number $A \rightarrow A_1A_0$
Binary number $B \rightarrow B_1B_0$



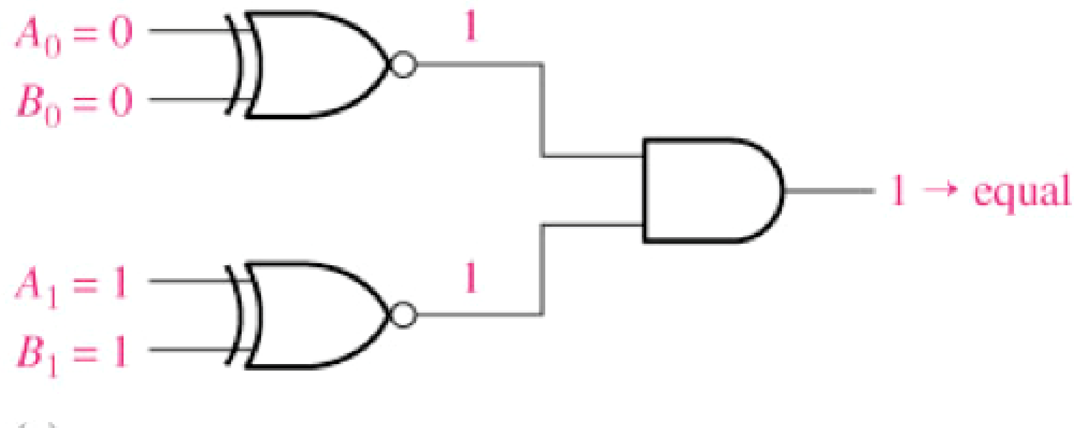
LOW: Indicate Not Equal

Comparator

Extra

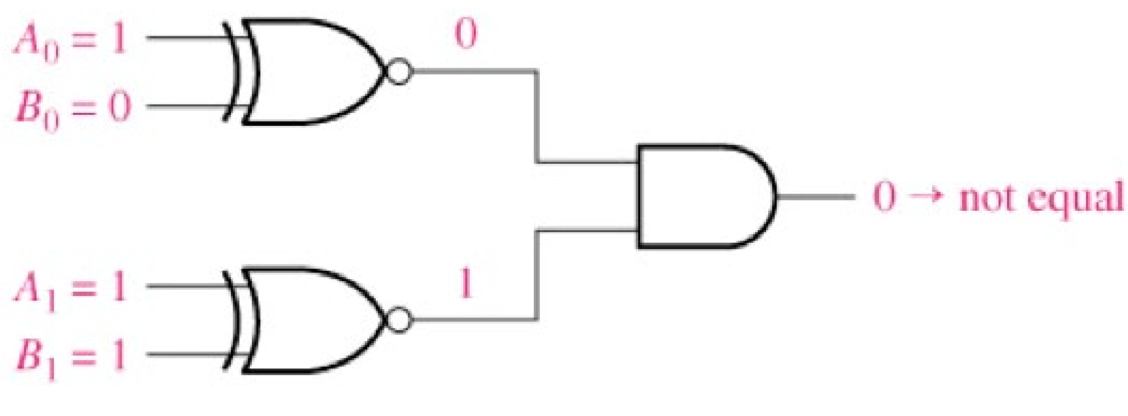
Solution:

(a) 10 and 10



(a)

(b) 11 and 10



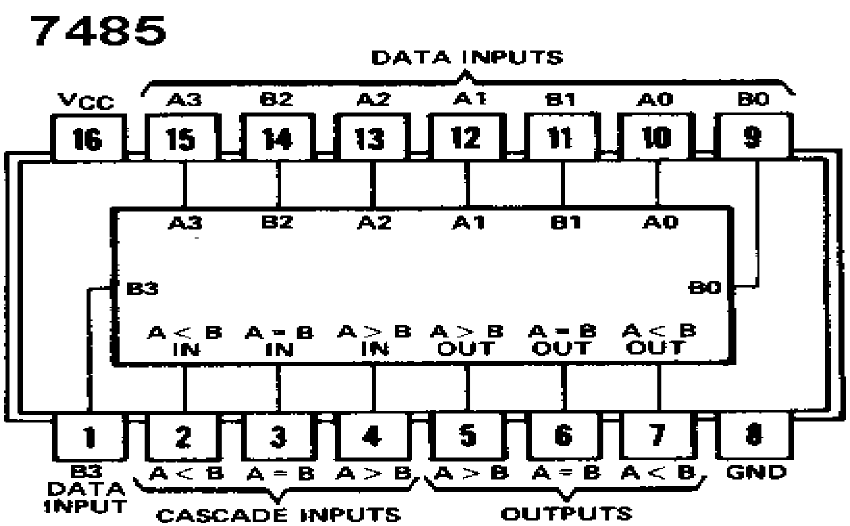
(b)

Comparator

Extra

MSI chip: (74LS85)

4-bit Comparator



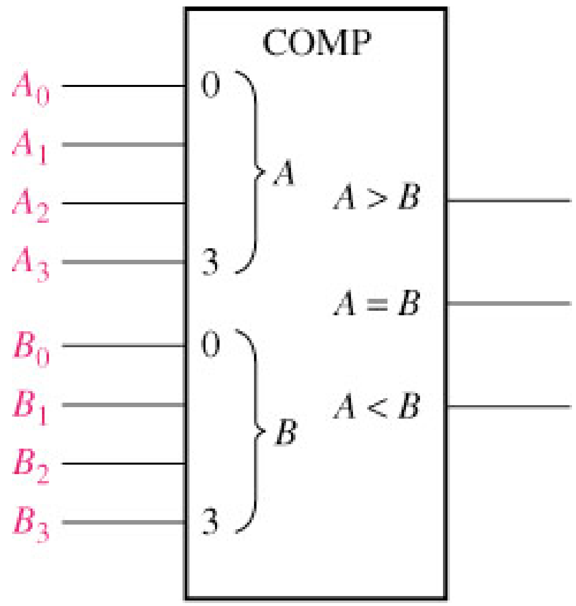


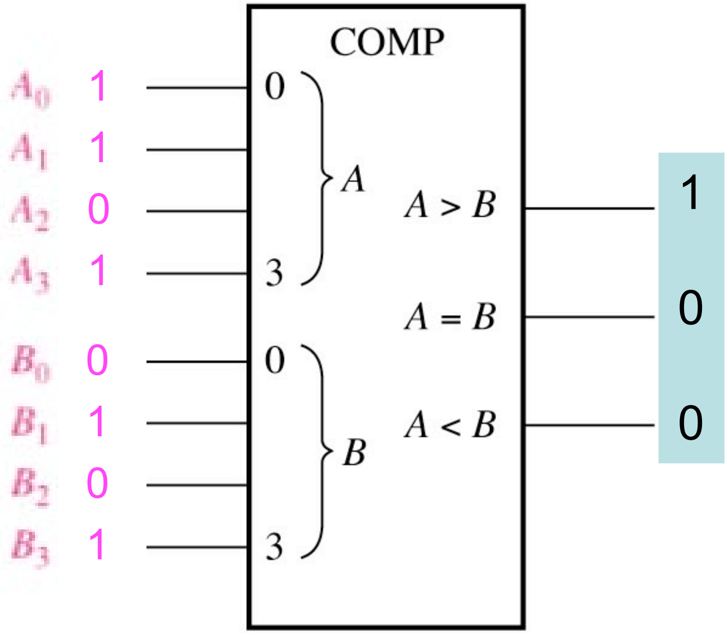
Figure: Logic symbol for a 4-bit comparator with inequality indication.

Comparator

Extra

Exercise 6.3: What are the comparator outputs when binary number of A = 1011 and B = 1010 applied as the inputs ?

Solution 6.3:



Step 1: Compare A_3 and B_3 → Number A = B; compare next bits

Step 2: Compare A_2 and B_2 → Number A = B; compare next bits.

Step 3: Compare A_1 and B_1 → Number A = B; compare next bits.

Step 4: Compare A_0 and B_0 → Number A > B

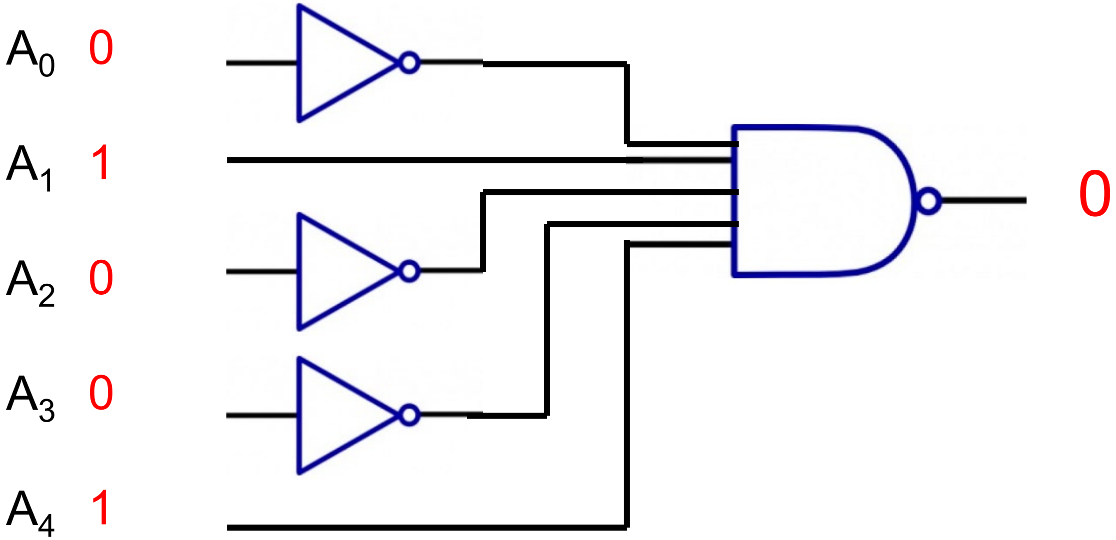
A > B = 1, A < B = 0, A = B = 0
when A = 1011 and B = 1010

Decoder

Extra

Exercise 6.4: Develop the logic required to detect the binary code 10010 and produce an active-LOW output.

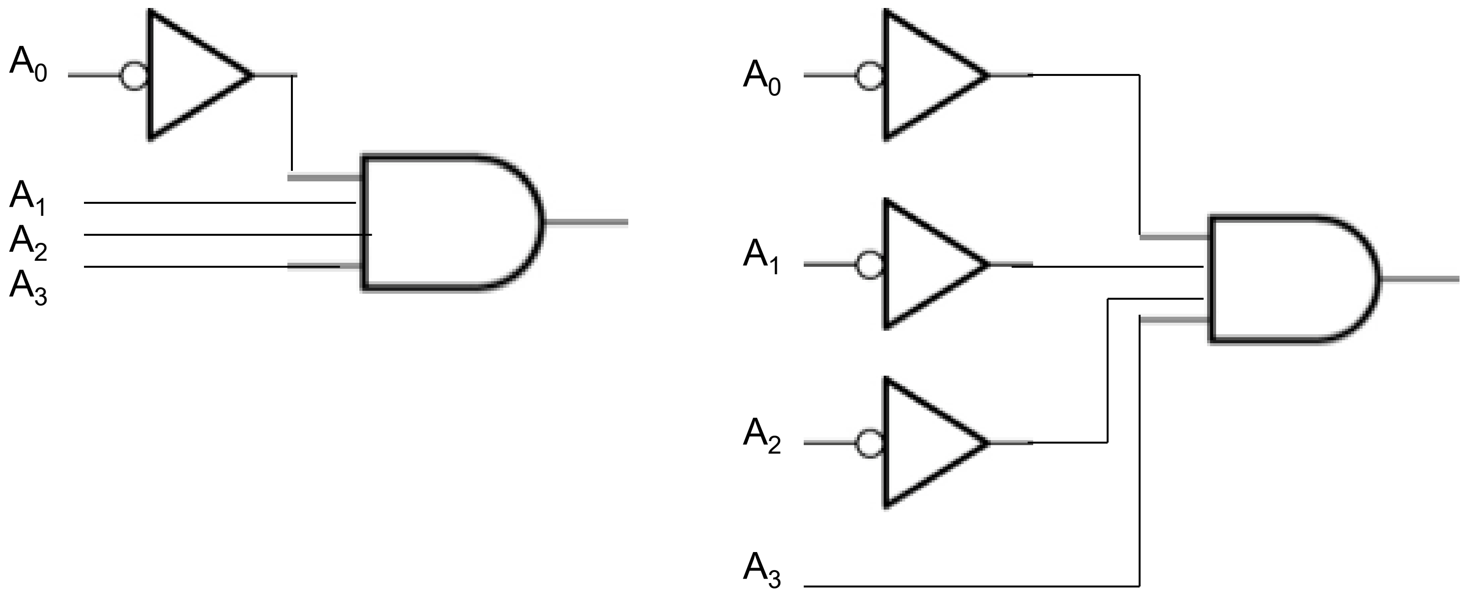
Solution 6.4:



Decoder

Extra

Exercise 6.5: When the output is active-HIGH for each of decoding gates in the Figure, what is the binary code appearing on the inputs? The MSB is A_3.

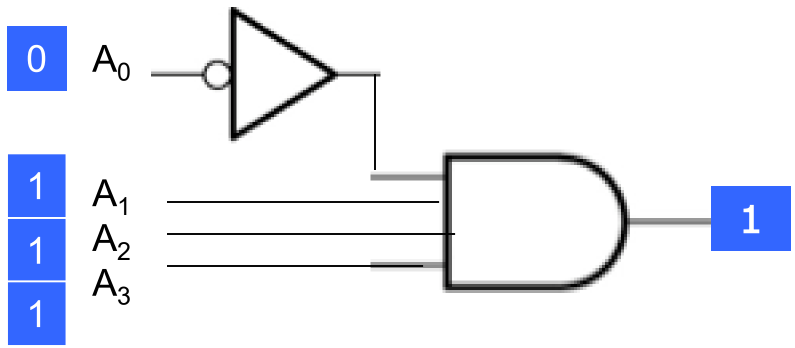


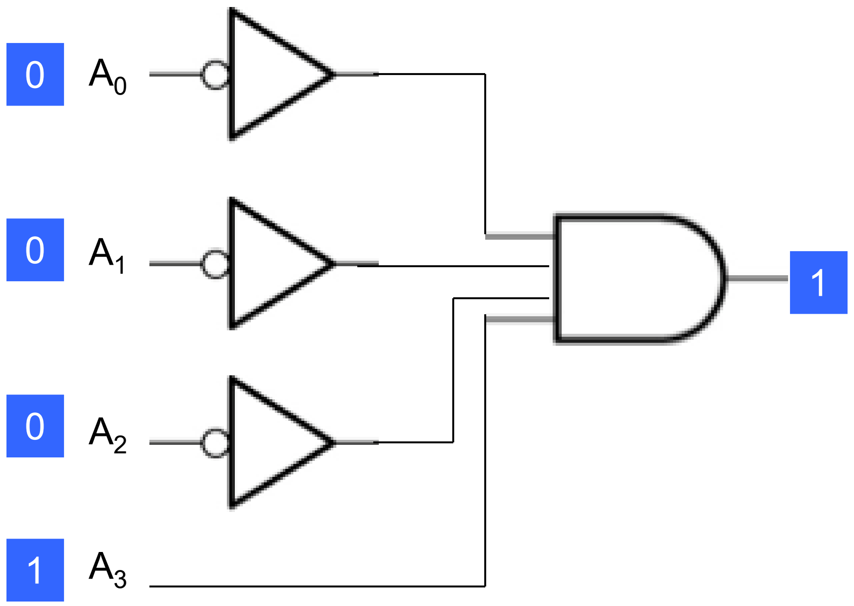
(a) (b)

Decoder

Extra

Solution 6.5:





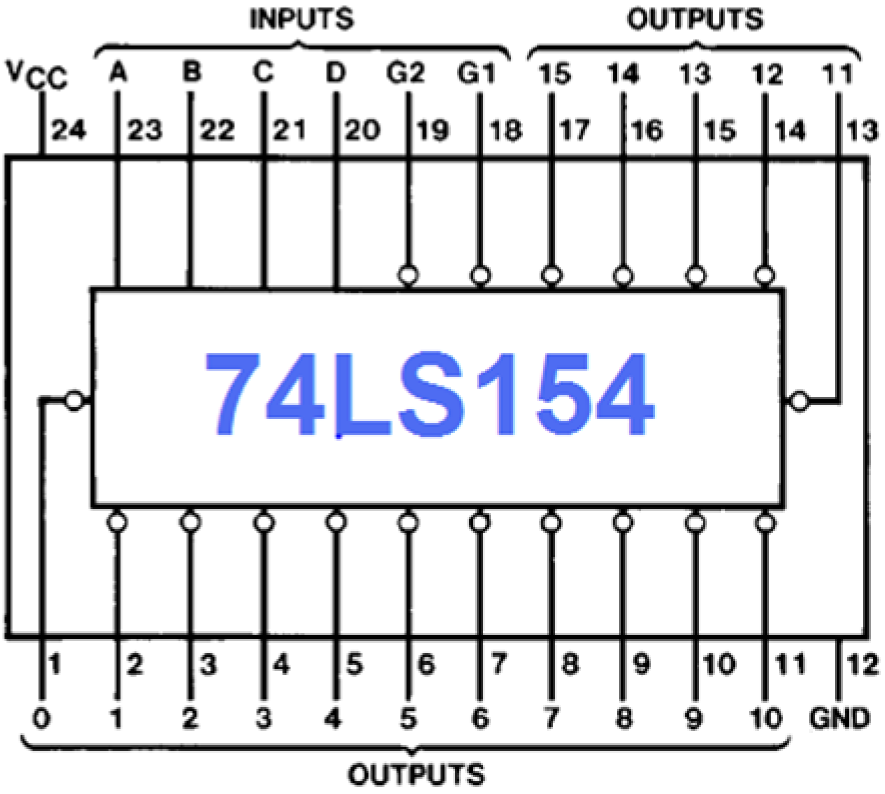
(a) $A_3A_2A_1A_0 = 1110$

(b) $A_3A_2A_1A_0 = 1000$

Decoder IC:

4-Bit Decoder

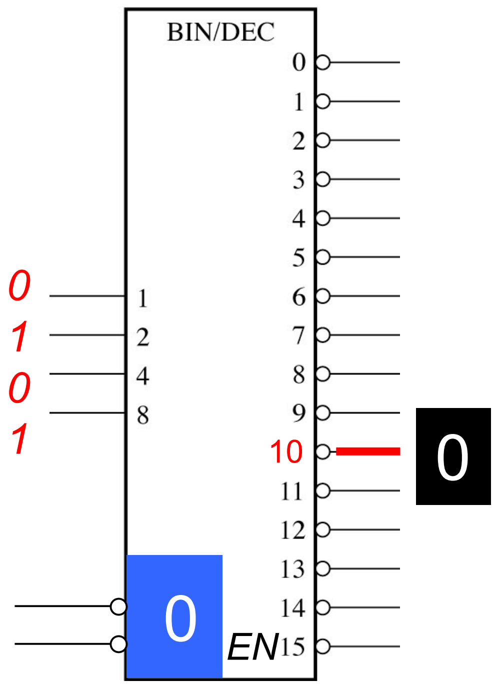
Extra



Decoder:

Extra

4-Bit Decoder



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

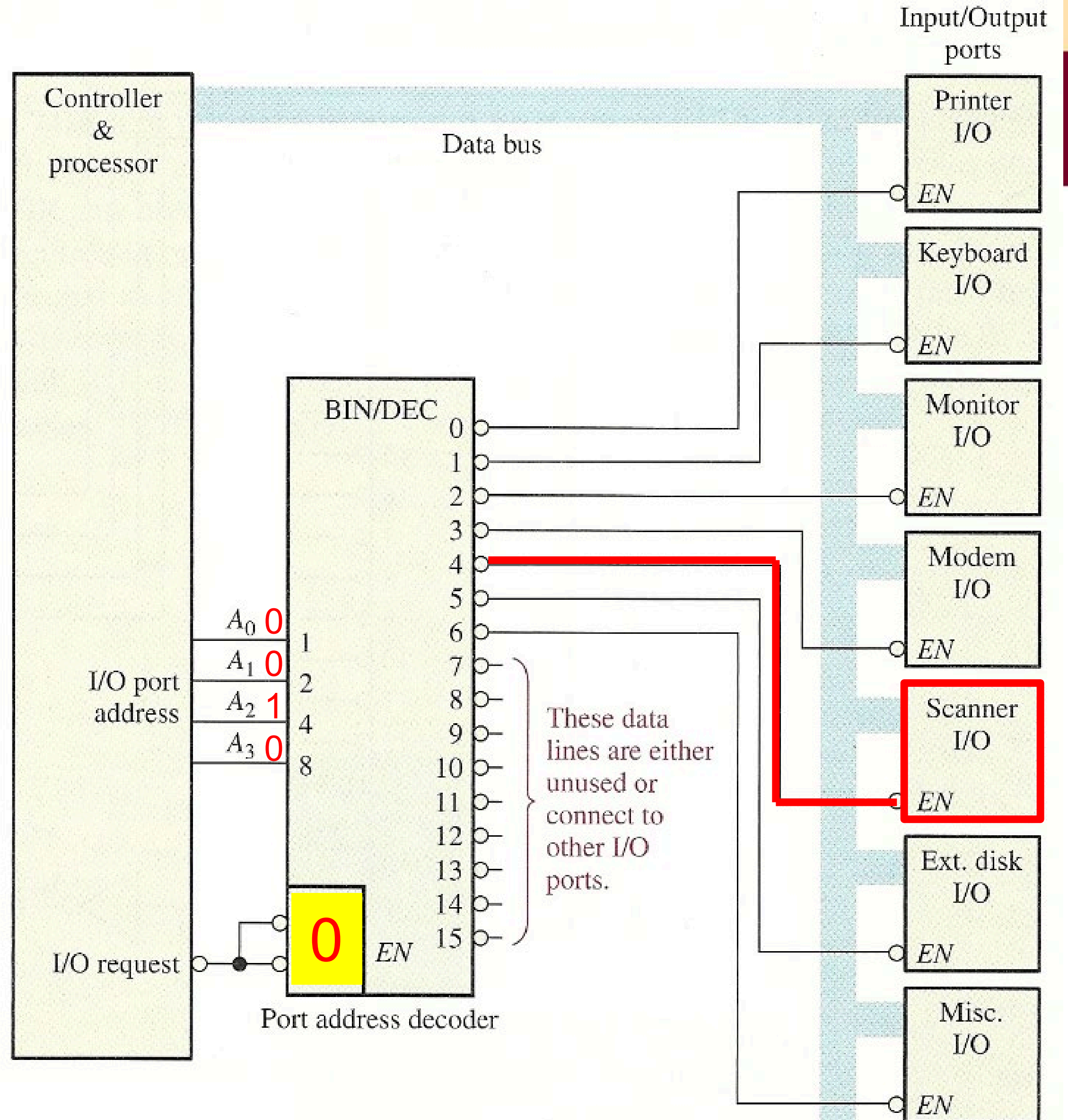
Extra

4-Bit Decoder

Example:

Port Address: 0100_2
I/O Request : LOW(0)

I/O Port: Scanner



A simplified computer I/O port system with a port address decoder with only four address lines shown.

Encoder

Extra

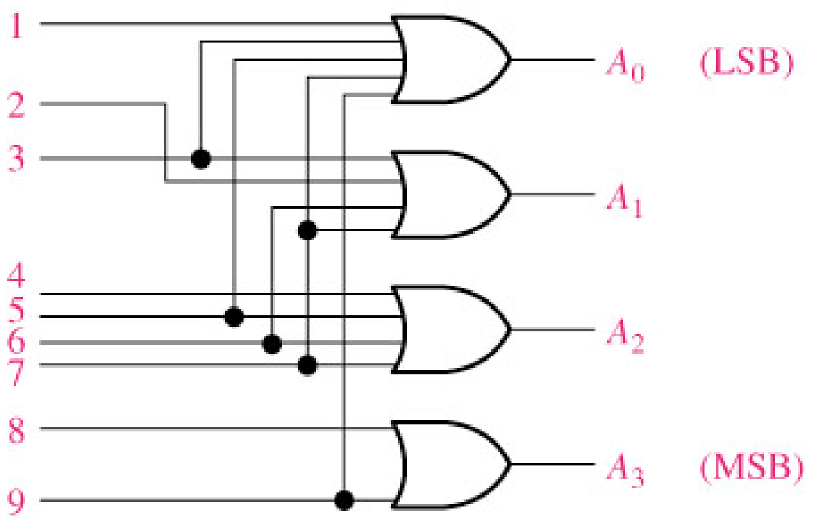
Exercise 6.6: Suppose, HIGH are applied to input 2 and 9 of the circuit

(a) What are the states of the output lines?

(b) Does this represent a valid BCD code?

(c) What is the restriction on the encoder logic?

Solution 6.6:



(a) $A_3 = 0, A_2 = 0, A_1 = 1, A_0 = 0$
$A_3 = 1, A_2 = 0, A_1 = 0, A_0 = 1$

(b) YES. The valid BCD code is between 0-9

(c) Only one input can be HIGH; the rest must be LOW