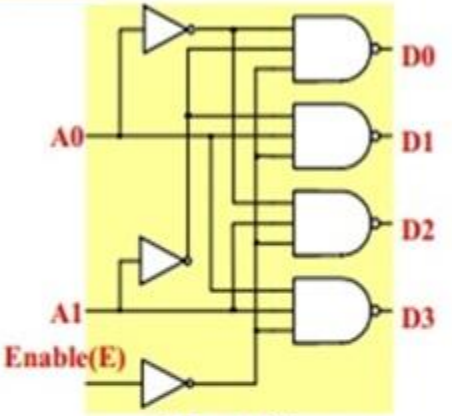
2-2 Decoder/Encoder

- NAND Gate Decoder
  - Constructed with NAND instead of AND gates
  - Logic Diagram/Truth Table : *Fig. 2-2*

* Active Low Output
* Fig. 2-1 3-to-8 Decoder ≒ Active High Output

Enable	Input		Output			
E	A1	A0	D0	D1	D2	D3
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0
1	x	x	1	1	1	1

(a) Truth Table



(b) Logic Diagram

Fig. 2-2 2-to-4 Decoder with NAND gates

- Decoder Expansion
  - Constructed decoder : *Fig. 2-3*
  - 3 X 8 Decoder constructed with two 2 X 4 Decoder
- Encoder
  - Inverse Operation of a decoder
  - 2^n input, n output
  - Truth Table : *Tab. 2-2*
    - 3 OR Gates Implementation
      - » A0 = D1 + D3 + D5 + D7
      - » A1 = D2 + D3 + D6 + D7
      - » A2 = D4 + D5 + D6 + D7

Tab. 2-2 Truth Table for Encoder

Inputs								Outputs		
D7	D6	D5	D4	D3	D2	D1	D0	A2	A1	A0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	1
0	1	0	0	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	1	1

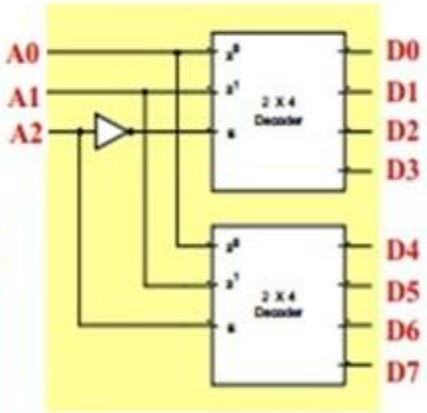


Fig. 2-3 A 3-to-8 Decoder constructed with two with 2-to-4 Decoder

2-2 Decoder/Encoder

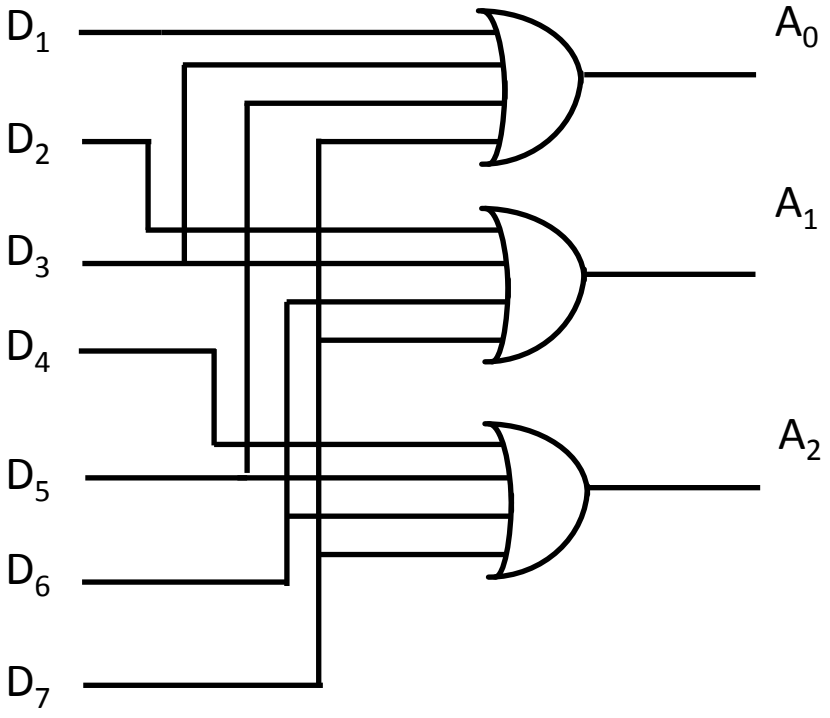
Octal to Binary Encoder

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	A_2	A_1	A_0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	1
0	1	0	0	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	1	1

$A_0 = D_1 + D_3 + D_5 + D_7$

$A_1 = D_2 + D_3 + D_6 + D_7$

$A_2 = D_4 + D_5 + D_6 + D_7$



2-3 Multiplexers

- Multiplexer(Mux)
  - A combinational circuit that receives binary information from one of 2^n input data lines and directs it to a single output line
  - A 2^n -to 1 multiplexer has *2^n input data lines* and *n input selection lines*(Data Selector)
  - 4-to-1 multiplexer Diagram : ***Fig. 2-4***
  - 4-to-1 multiplexer Function Table : ***Tab. 2-3***

Tab. 2-3 Function Table for 4-to-1 line Multiplexter

Select		Output
S1	S0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

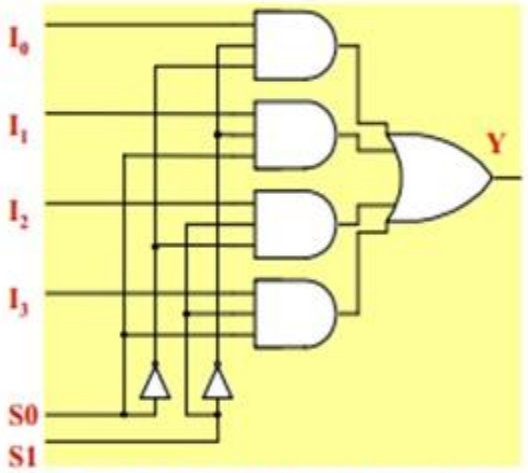


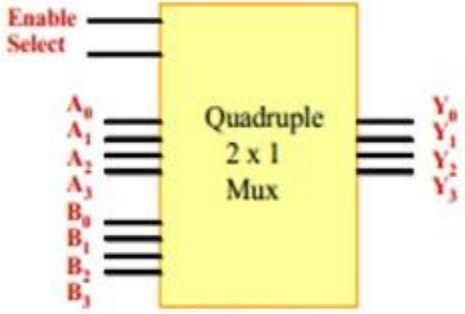
Fig. 2-4 4-to-1 Line Multiplexer

- Quadruple 2-to-1 Multiplexer
  - Quadruple 2-to-1 Multiplexer : ***Fig. 2-5***

Fig. 2-5 Quadruple 2-to-1 line Multiplexter

Select		Output
E	S	Y
0	0	All 0's
1	0	A
1	1	B

(a) Function Table



(b) Block Diagram

2-3 Multiplexers

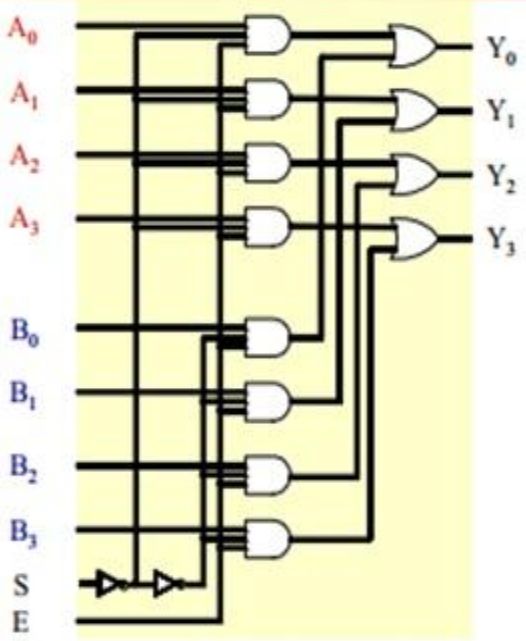
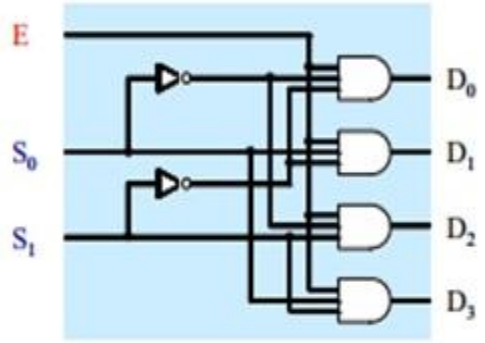


Fig A. Combinational logic diagram with four 2×1 multiplexer



E → Data input
S_0, S_1 → Select Data

Fig B. Demultiplexer

A **Demultiplexer,** sometimes abbreviated **DMUX** is a circuit that has one input and more than one output. It is used when a circuit wishes to send a signal to one of many devices

- Suppose a MUX is having 16 input lines. How many selection input lines are required?

A) 2

B) 3

C) 4

D) 5

2-4 Registers

- Register
  - A group of flip-flops with each flip-flop capable of storing one bit of information
  - An n-bit register has a group of n flip-flops and is capable of storing any binary information of n bits
  - The simplest register consists only of flip-flops, with no external gate : *Fig. 2-6*
  - A clock input C will load all four inputs in parallel
    - The clock must be *inhibited* if the content of the register must be left unchanged
- Register with Parallel Load
  - A 4-bit register with a load control input : *Fig. 2-7*
  - The clock inputs receive clock pulses at all times
  - The buffer gate in the clock input will increase "fan-out"
  - Load Input
    - 1 : Four input transfer
    - 0 : Input inhibited, Feedback from output to input(**no change**)

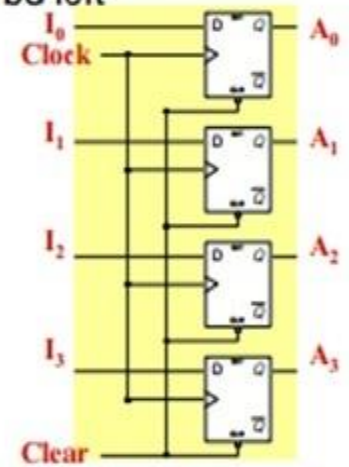


Fig. 2-6 4-bit register

2-4 Registers

- When the load input is 1 , the data in the four inputs are transferred into the register with the next positive transition of a clock pulse

- When the load input is 0, the data inputs are inhibited and the D-output of flip flop are connected to their inputs.

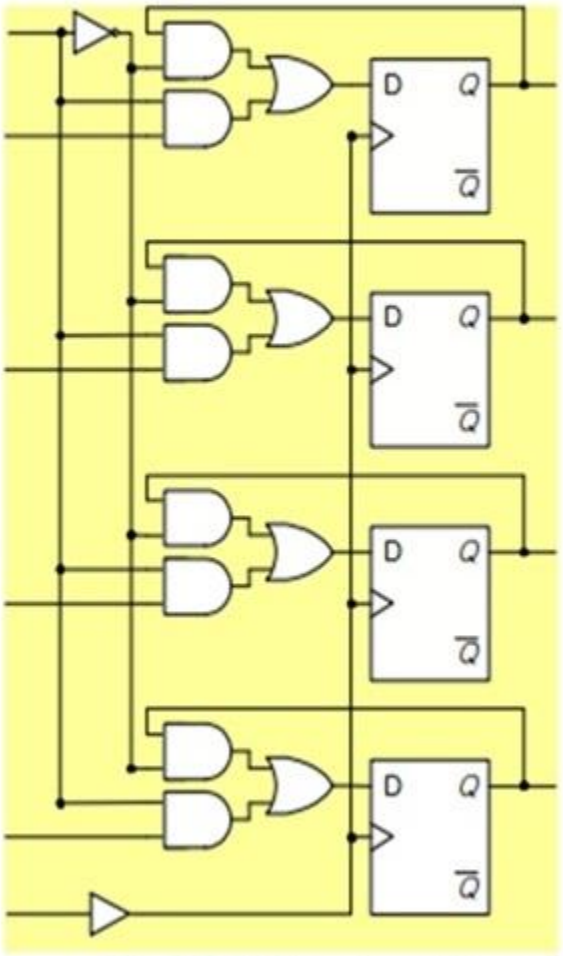


Fig. 2-7 4-bit register with parallel load

2-5 Shift Registers

- Shift Register
  - A register capable of shifting its binary information in one or both directions
  - The logical configuration of a shift register consists of a chain of flip-flops in cascade
  - The simplest possible shift register uses only flip-flops : *Fig. 2-8*
  - The ***serial input*** determines what goes into the leftmost position during the shift
  - The ***serial output*** is taken from the output of the rightmost flip-flop

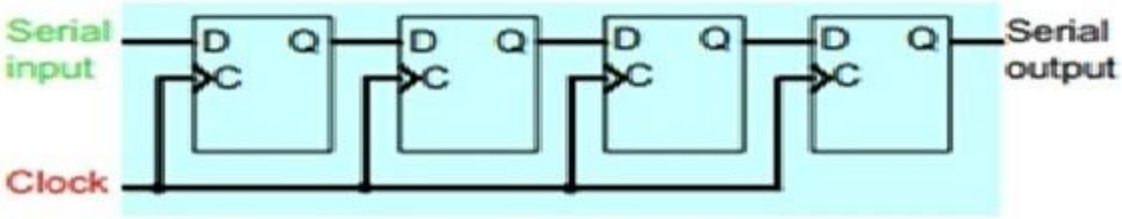


Fig. 2-8 4-bit shift register

- Data can be transferred to a register only when its load input is 1.

A) True

B) False

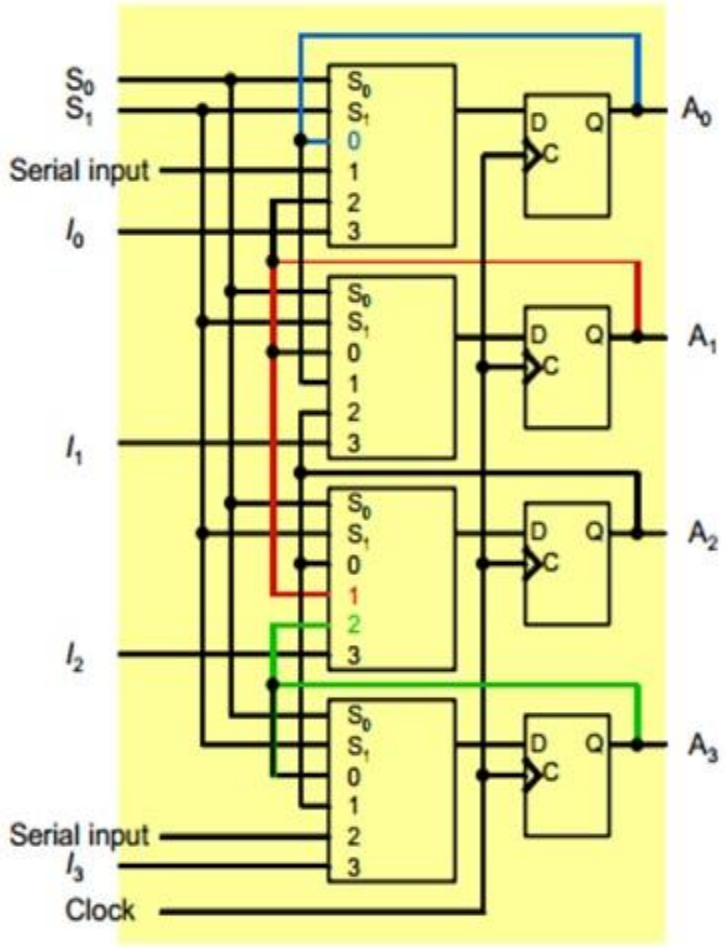
2-5 Shift Registers

- Bidirectional Shift Register with Parallel Load
  - A register capable of shifting in *one direction only* is called a ***unidirectional shift register***
  - A register that can shift in *both directions* is called a ***bidirectional shift register***
  - The most general shift register has all the capabilities listed below:
    - An input clock pulse to synchronize all operations
    - A shift-right /left (serial output/input)
    - A parallel load, n parallel output lines
    - The register unchanged even though clock pulses are applied continuously
  - 4-bit bidirectional shift register with parallel load : *Fig. 2-9*
    - 4 X 1 Mux = 4 , D F/F = 4

Tab. 2-4 Function Table for Register of Fig. 2-9

Mode		Operation
S1	S0	
0	0	No chage
0	1	Shift right(down)
1	0	shift left(up)
1	1	Parallel load

2-5 Shift Registers



$S_1S_0 = 00$: $A_i \rightarrow A_i$ (No change)
$S_1S_0 = 01$: $A_{i-1} \rightarrow A_i$ (Shift)
$S_1S_0 = 10$: $A_{i+1} \rightarrow A_i$ (Shift)
$S_1S_0 = 11$: Parallel load

- Shift Register

Interface digital systems situated remotely from each other

Fig. 2-9 Bidirectional shift register

- Which of the following is not a type of a Register?

A) SISO

B) SIPO

C) PIPO

D) None of the above