BILKENT UNIVERSITY
Department of Electrical and Electronics Engineering
EEE102 Introduction to Digital Circuit Design
MidTerm Exam II SOLUTION
29-04-2006
Duration 110 minutes

Surname:_____

Name:_____

ID-Number:_____

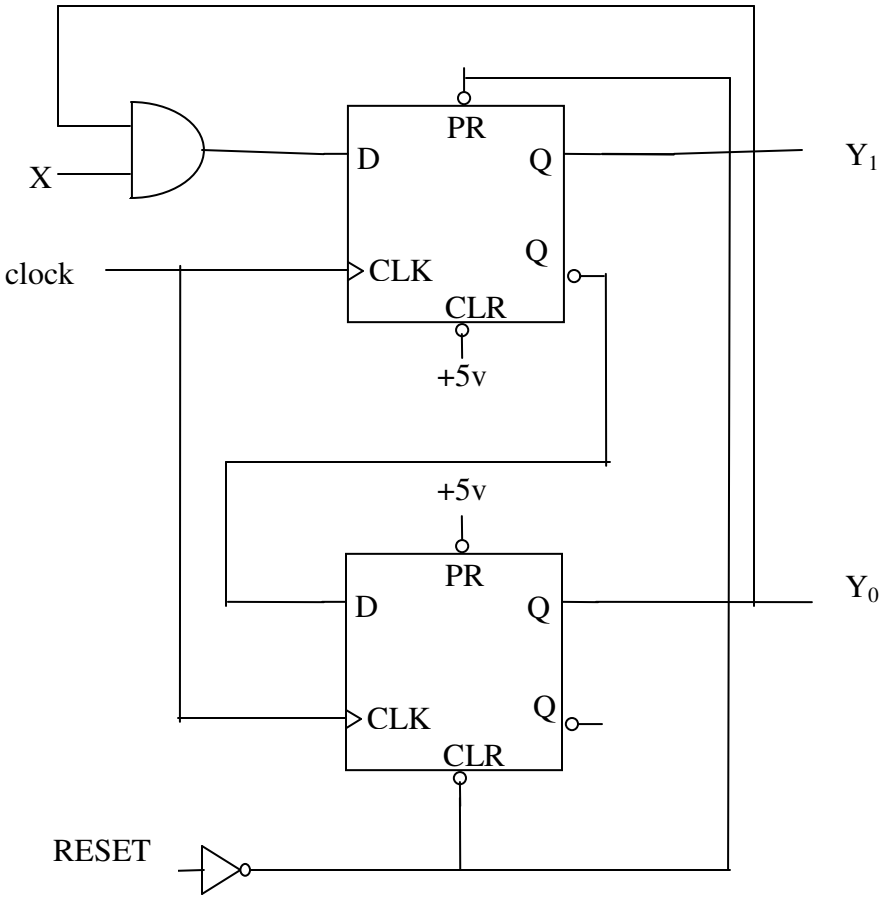
Signature:_____

There are 5 questions of equal weights. Total weight of the exam is 110 points (10 points bonus). Solve all. Do not detach pages.
Show all your work.

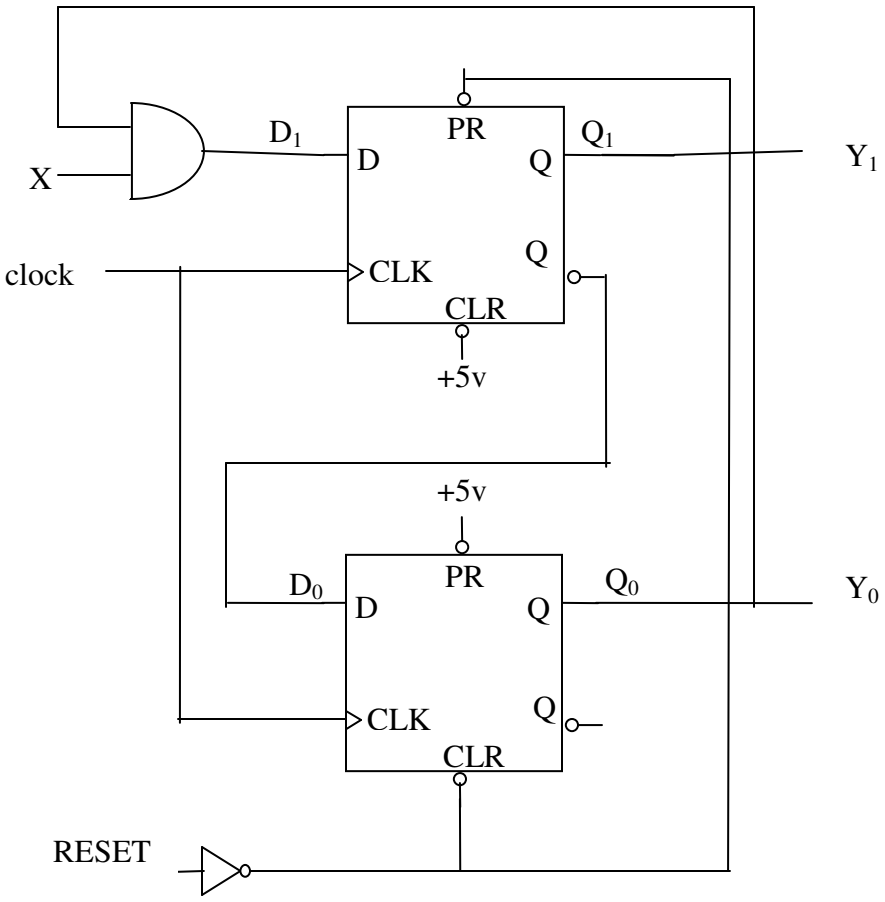
| | |
|---|---|
| Q1 | |
| Q2 | |
| Q3 | |
| Q4 | |
| Q5 | |
| TOTAL | |

**Q1. For the circuit below obtain and write excitation table, output table, next state table, and also the state definitions and diagram. Is this a Moore or Mealy FSM, why? X is the input and $Y_1Y_0$ is the output of this circuit. Explain the function of the RESET input, and include it in your state diagram**.

SOLUTION:




SOLUTION:




Excitation Equations: $D_1 = XQ_0$ and $D_0 = Q_1'$

Excitation table is

| $Q_1$ | $Q_0$ | X(at tick) | $D_1$ | $D_0$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 |

Next state table is the same as the excitation table with $D_1$ replaced by $Q_1$* and $D_0$ replaced by $Q_0$*.

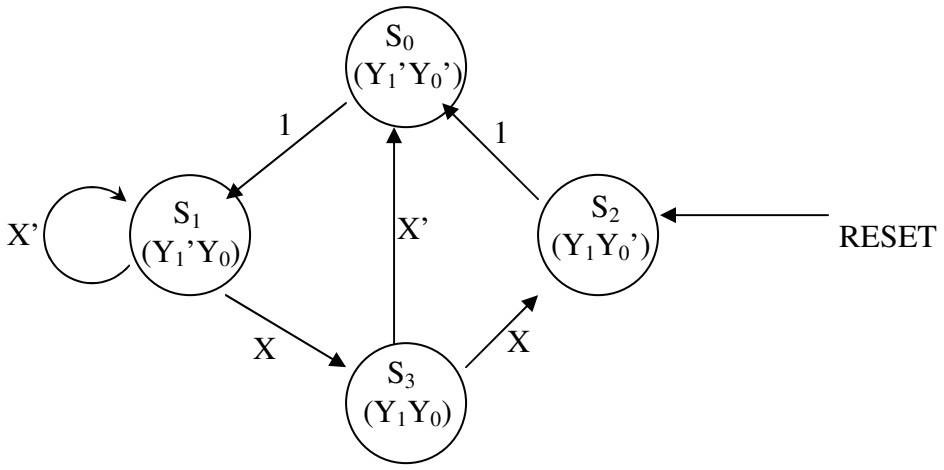
Output equations: $Y_1 = Q_1$ and $Y_0 = Q_0$
Output table

| $Q_1$ | $Q_0$ | $Y_1$ | $Y_0$ |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 |

State Encoding

| name | $Q_1$ | $Q_0$ |
|---|---|---|
| $S_0$ | 0 | 0 |
| $S_1$ | 0 | 1 |
| $S_2$ | 1 | 0 |
| $S_3$ | 1 | 1 |

State diagram




Note that I have indicated in the state circles also the outputs, and therefore the above diagram is in fact a state/ouput diagram.

RESET is an asynchronous input which makes the FSM go to state $S_2$ independent of the clock.
This FSM is a Moore FSM if do not consider RESET as a proper input.
It is a Mealy FSM if we consider RESET as a proper input as well.

**Q2. Write VHDL code for a 3-to-8 generic decoder (with enable), which has all inputs and and outputs active-high, using structural type programming using two instances of the following module which is already available in the library.**
SOLUTION:
library IEEE;

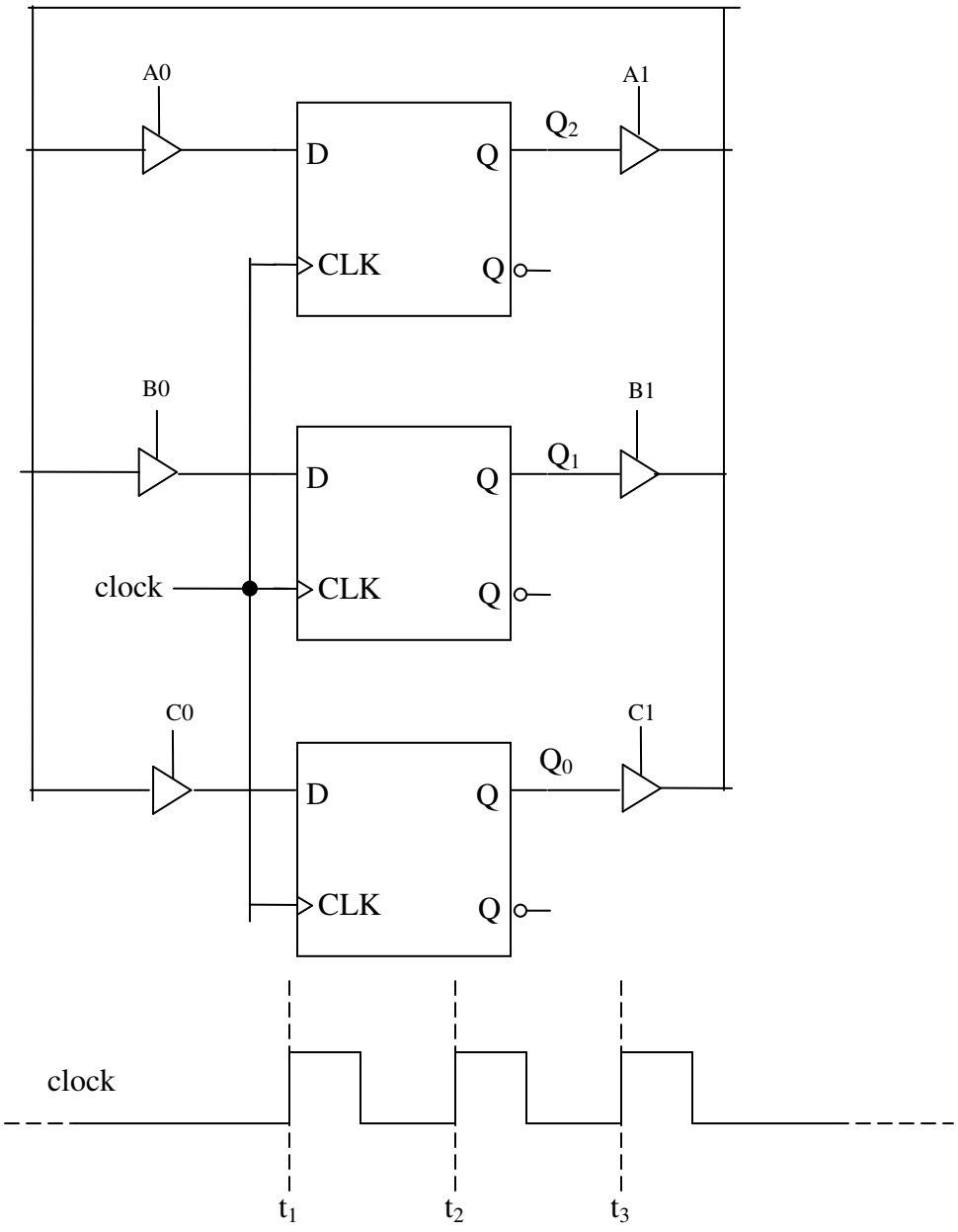
```
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity cat is
    Port ( E : in std_logic;
           S : in std_logic_vector(1 downto 0);
           Y : out std_logic_vector(0 to 3));
end cat;

architecture Behavioral of cat is
  signal temp:std_logic_vector(0 to 3);
begin
  with S select   temp <=
                "1000" when "00",
                "0100" when "01",
                "0010" when "10",
                "0001" when "11",
                "0000" when others;
     Y <= temp when E = '1' else "0000";
end Behavioral;
```

Solution:

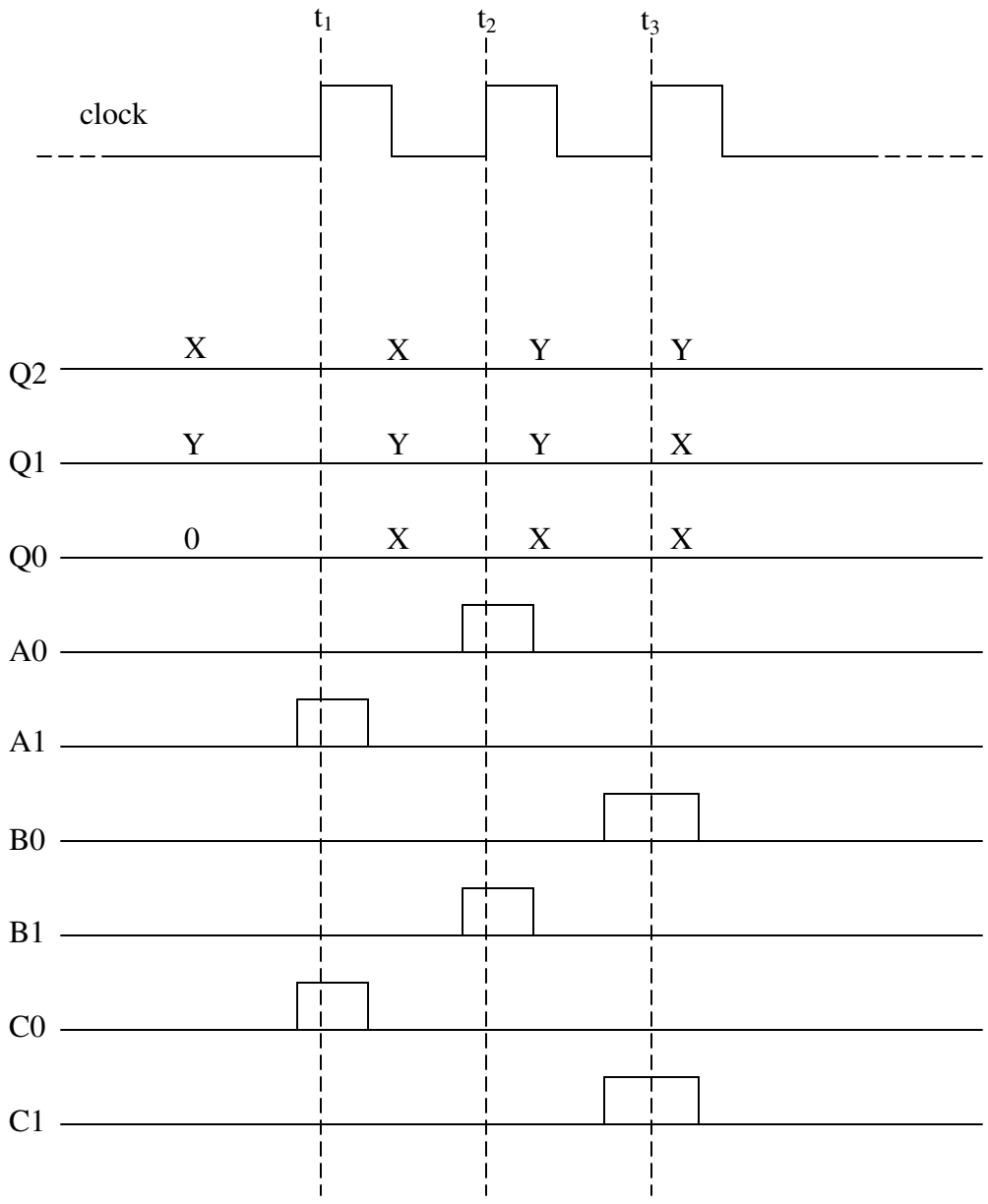
```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity top is
    Port ( EN : in std_logic;
           SEL : in std_logic_vector(2 downto 0);
           DECOUT : out std_logic_vector(0 to 7));
end top;

architecture Behavioral of top is
component cat is
    Port ( E : in std_logic;
           S : in std_logic_vector(1 downto 0);
           Y : out std_logic_vector(0 to 3));
end component;
signal G0,G1:std_logic_vector(0 to 3);
signal Z0,Z1:std_logic;
signal SS:std_logic_vector(1 downto 0);
begin
SS<= SEL(1)&SEL(0);
```

```
Z0<= EN and not SEL(2);
Z1<=EN and SEL(2);
L1: cat port map (Z0,SS,G0);
L2: cat port map (Z1,SS,G1);
DECOUT<=G0&G1;
end Behavioral;
```

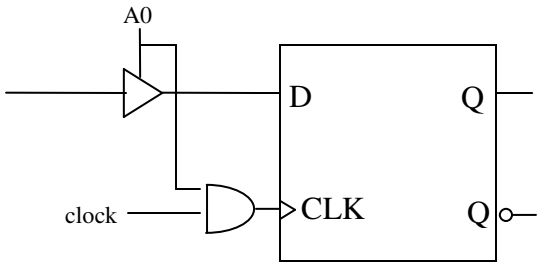
**Q3. For the circuit and the clock signal shown below, $Q_2$ = X, $Q_1$ = Y, and $Q_0$ = '0' before time $t_1$. Draw time waveforms for A0,A1,B0,B1,C0, and C1 so that $Q_2$ = Y, and $Q_1$ = X after time $t_3$ (in other words the values of $Q_2$ and $Q_1$ are switched).**



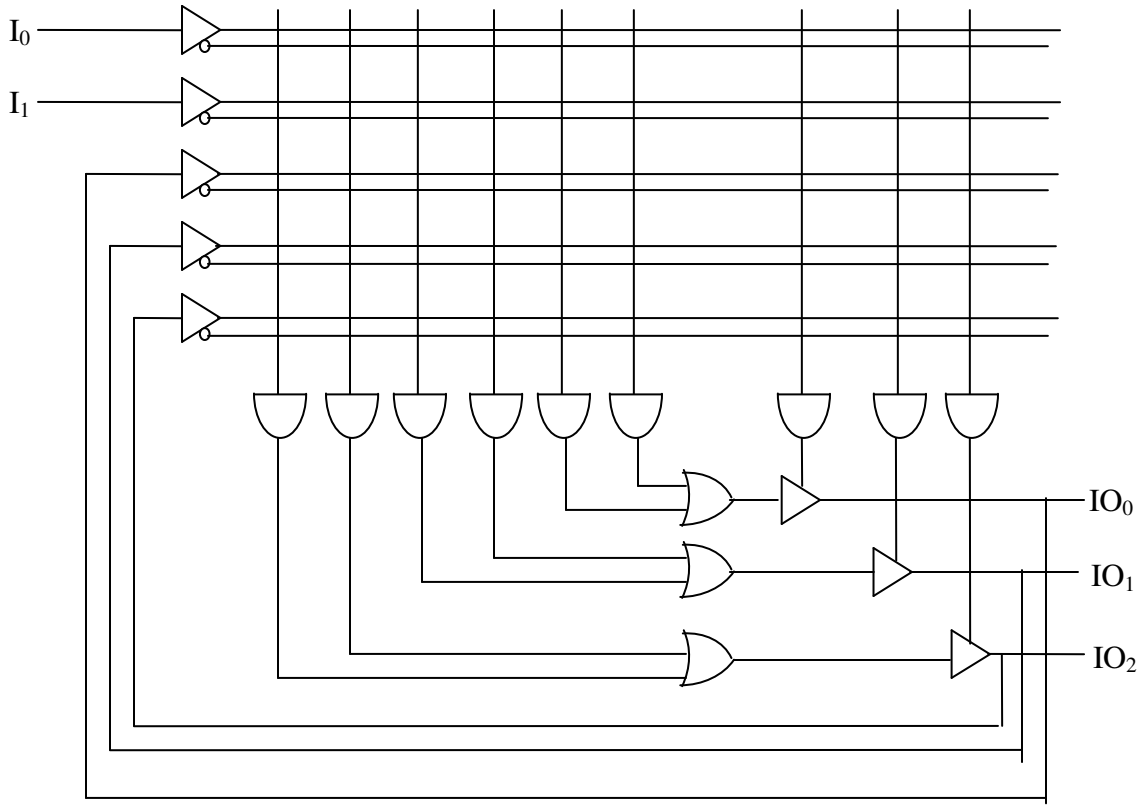
SOLUTION:
As announced during the exam we assume if a ff's input is at high Z then its output is not changed at a clock tick. (Of course it is possible to include circuitry to guarantee this assumption but it makes the circuit look very complicated, therefore it was not drawn).



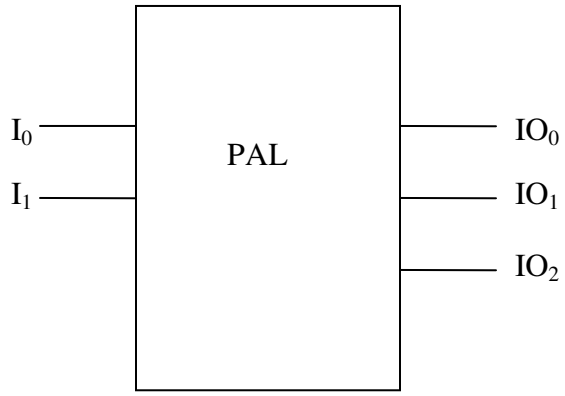

For your information: One possible way of guaranteeing the above assumption is to gate the clock. For example (shown only for the first ff) the following circuit does it.



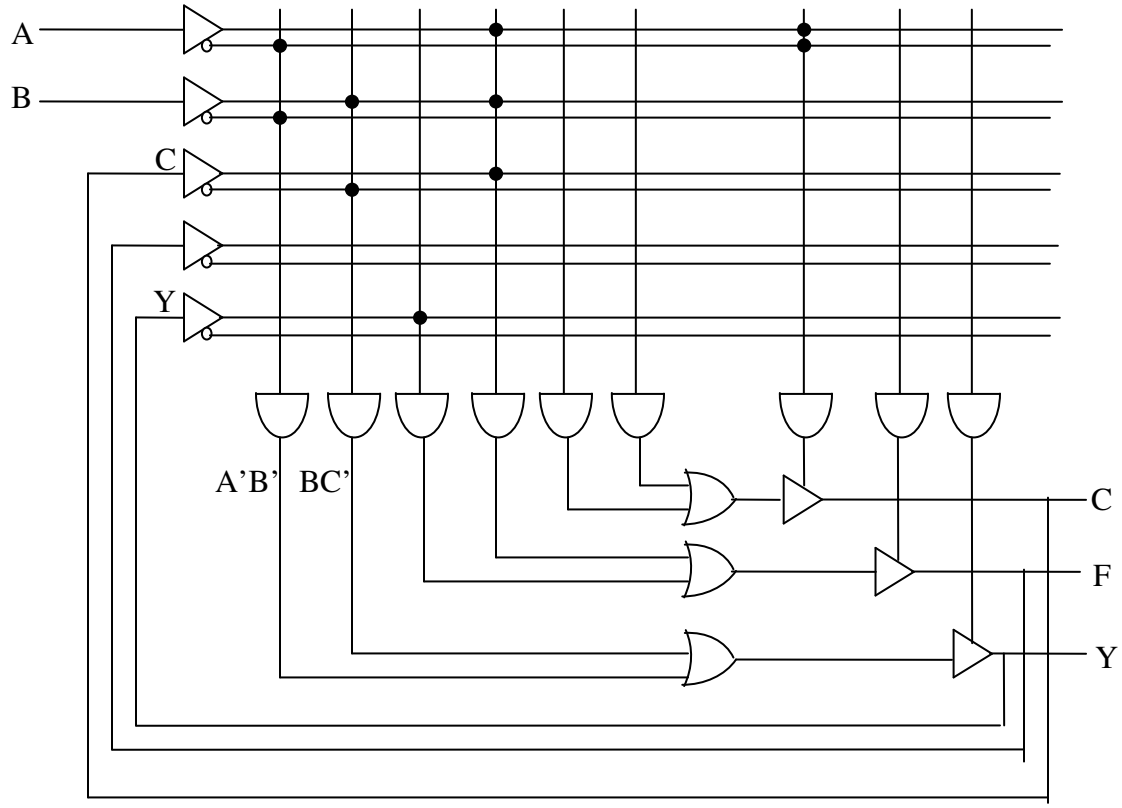

**Q4. The internal circuit of a PAL which has 2 input pins and 3 input-output pins is drawn below. In this PAL, AND gate inputs are '1' if they are not connected. Implement the function F = A'B' + BC' + ABC using this PAL. Do not simplify or modify the expression. Do not use any external wire or component.**
SOLUTION:



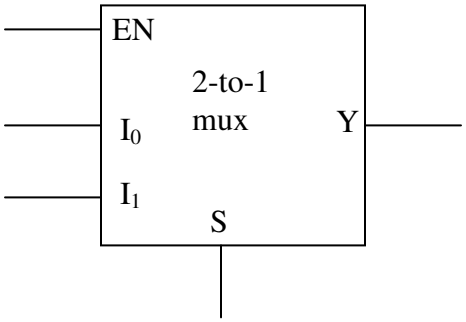

To avoid any confusion, the pin diagram (excluding ground and supply connections) of the above PAL is given below:




Solution: Say Y = A'B' + BC'. Then F = Y + ABC



**Q5. We have 2-to-1 multiplexers in stock. These have active-high pins and the block diagram is as follows:**




**a) Design and draw a circuit which implements this 2-to-1 multiplexer, using NAND and NOT gates.**

**b) Use 3 of these 2-to-1 multiplexers, a 2-to-4 decoder, and additional minimum number of simple gates to realize a 6-to-1 multiplexer which has the following properties: It has three control inputs S2, S1, S0, six data inputs D0, D1, D2, D3, D4, D5 and two outputs OUT and ERROR. It has the following function table:**

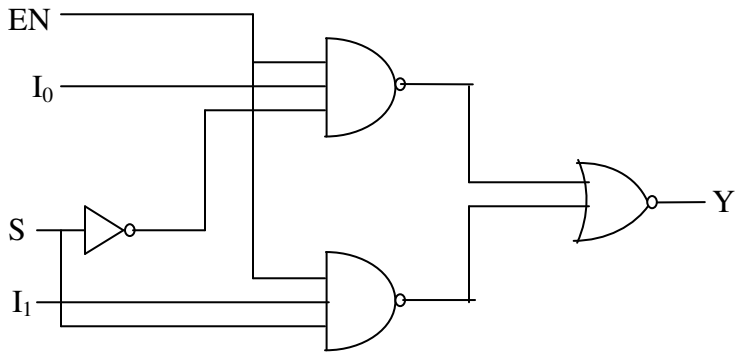
| Function | S2 | S1 | S0 | OUT | ERROR |
|---|---|---|---|---|---|
| (choose) | 0 | 0 | 0 | D0 | 0 |
| (choose) | 0 | 0 | 1 | D1 | 0 |
| (choose) | 0 | 1 | 0 | D2 | 0 |
| (choose) | 0 | 1 | 1 | D3 | 0 |
| (choose) | 1 | 0 | 0 | D4 | 0 |
| (choose) | 1 | 0 | 1 | D5 | 0 |
| (disable) | 1 | 1 | 0 | 0 | 0 |
| (error) | 1 | 1 | 1 | 0 | 1 |

SOLUTION:

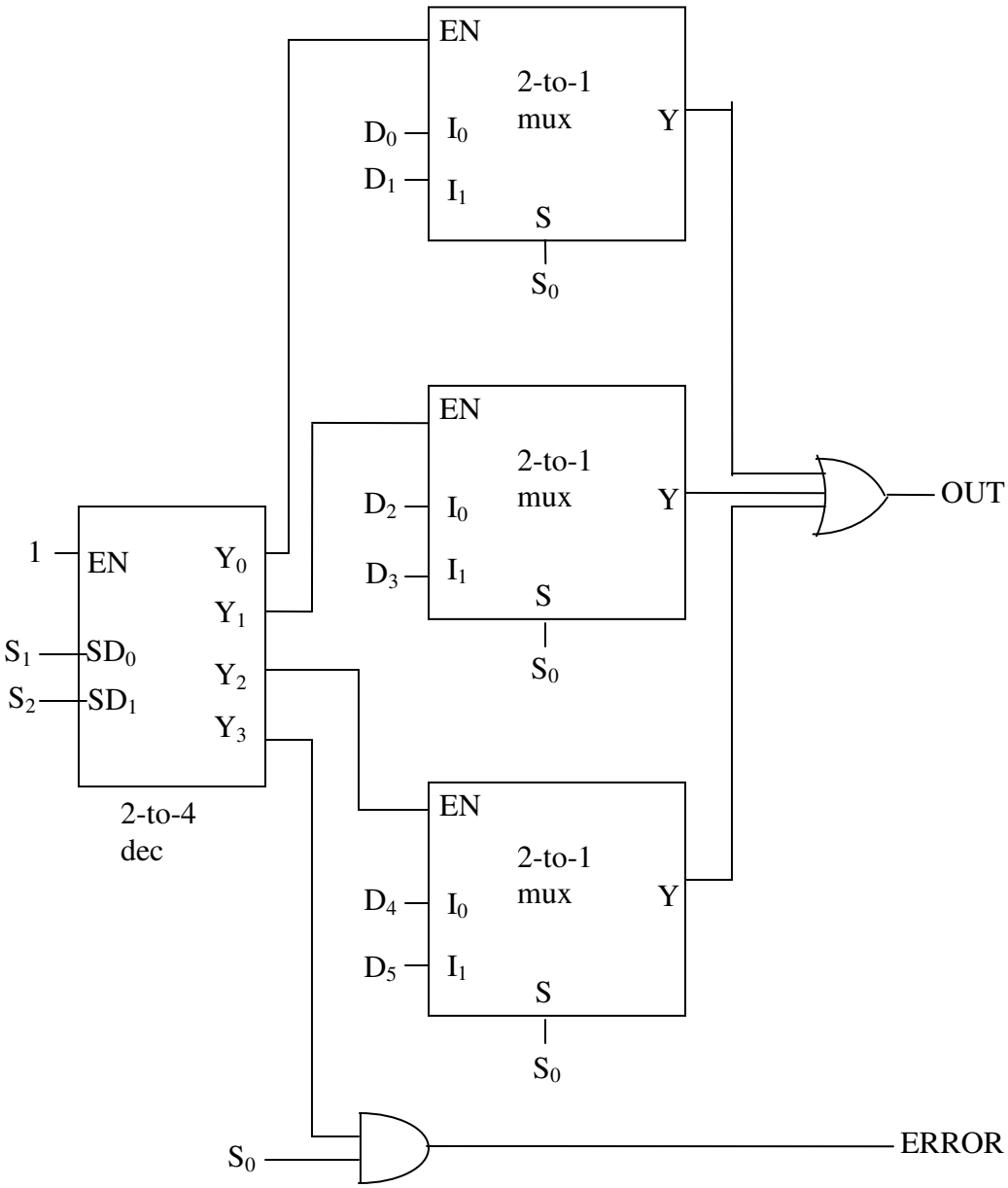
a)

| EN | S | Y |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | $I_0$ |
| 1 | 1 | $I_1$ |

Thus $Y = EN.S'.I_0 + EN.S.I_1$




b)


EN
2-to-1
mux
$D_0$
$I_0$
Y
$D_1$
$I_1$
S
$S_0$
EN
2-to-1
mux
$D_2$
$I_0$
Y
OUT
$D_3$
$I_1$
S
$S_0$
1
EN
$Y_0$
$Y_1$
$S_1$
$SD_0$
$Y_2$
$S_2$
$SD_1$
$Y_3$
2-to-4
dec
EN
2-to-1
mux
$D_4$
$I_0$
Y
$D_5$
$I_1$
S
$S_0$
$S_0$
ERROR