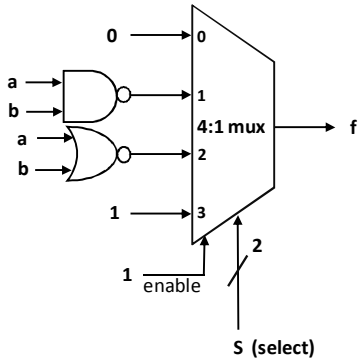
NAME:	ENTRY No.:

1. Write a Boolean expression to represent the function implemented by the circuit shown. There is no need to draw truth table or Karnaugh map.



[2]

Solution:

Denoting the multiplexer inputs as x0, x1, x2 and x3, we can express f as follows.

f = enable.(s1'.s0'.x0 + s1'.s0.x1 + s1.s0'.x2 + s1.s0.x3),
where s1 and s0 are the two bits of input s.

We can replace x0, x1, x2, x3 and enable by '0', (a.b)', (a+b)', '1' and '1' respectively.
This leads to the following expression for f.

f = 1.(s1'.s0'.0 + s1'.s0.(a.b)' + s1.s0'.(a+b)' + s1.s0.1)
= s1'.s0.(a.b)' + s1.s0'.(a+b)' + s1.s0
= s1'.s0.a' + s1'.s0.b' + s1.s0'.a'.b' + s1.s0

This may be simplified to (not required by the question) the following.

f = s0.a' + s0.b' + s1.a'.b' + s1.s0

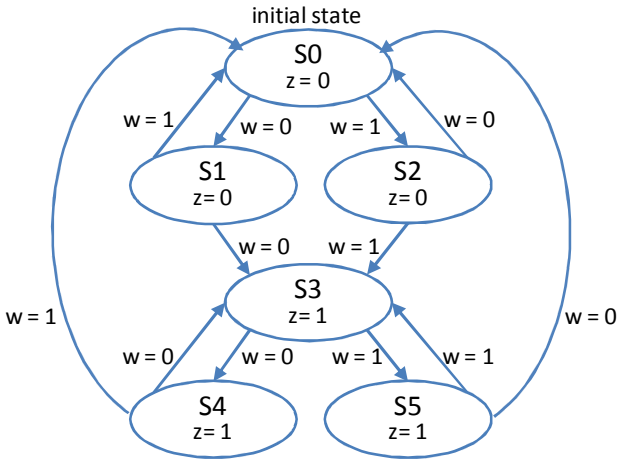
NAME:	ENTRY No.:

2. Construct state diagram and state table of a synchronous sequential circuit that looks at input w in frames or sequences of 2 bits. In each frame, it checks if both the bits are identical (both 1's or both 0's). The result of this check is shown on output z during the next frame in both its clock cycles. Output z is made 1 if the bits in the previous frame were identical and made 0 otherwise. Assume that in the initial state, the circuit is ready to start looking at a new frame (i.e., the first 2 bits form the 1st frame, the next 2 bits form the 2nd frame and so on) and z is to be kept 0 during the first frame. The input is synchronized with the clock.

[2]

Solution:

State transition diagram.

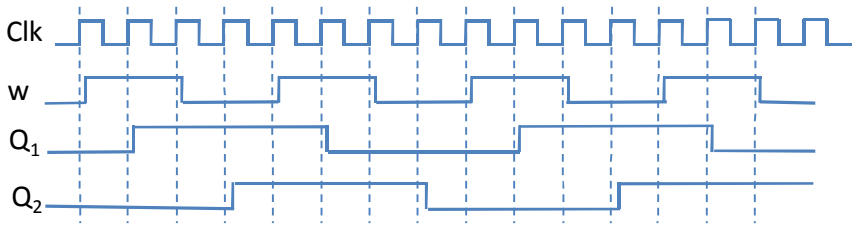


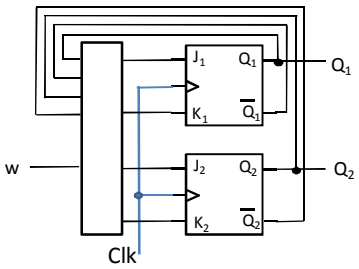
State table

Present state	Next state for w = 0	Next state for w = 1	Output z
S0	S1	S2	0
S1	S3	S0	0
S2	S0	S3	0
S3	S4	S5	1
S4	S3	S0	1
S5	S0	S3	1

NAME:	ENTRY No.:

3. Behaviour of a circuit with one input w and two outputs Q_1, Q_2 is depicted by the waveforms shown below. The circuit consists of two JK flip-flops and a combinational circuit that drives their JK inputs. Inputs to this combinational circuit are w and flip-flop outputs. Derive expressions for the functions implemented by this combinational circuit.





[4]

Solution:

Transitions required at various clock edges and corresponding J K values.

clock edge	w	present Q1 Q2	next Q1 Q2	Q1 transition	Q2 transition	J1 K1	J2 K2
1	0	0 0	0 0	0 => 0	0 => 0	0 -	0 -
2	1	0 0	1 0	0 => 1	0 => 0	1 -	0 -
3	1	1 0	1 0	1 => 1	0 => 0	- 0	0 -
4	0	1 0	1 1	1 => 1	0 => 1	- 0	1 -
5	0	1 1	1 1	1 => 1	1 => 1	- 0	- 0
6	1	1 1	0 1	1 => 0	1 => 1	- 1	- 0
7	1	0 1	0 1	0 => 0	1 => 1	0 -	- 0
8	0	0 1	0 0	0 => 0	1 => 0	0 -	- 1
9	0	0 0	0 0	0 => 0	0 => 0	0 -	0 -
10	1	0 0	1 0	0 => 1	0 => 0	1 -	0 -
11	1	1 0	1 0	1 => 1	0 => 0	- 0	0 -
12	0	1 0	1 1	1 => 1	0 => 1	- 0	1 -
13	0	1 1	1 1	1 => 1	1 => 1	- 0	- 0
14	1	1 1	0 1	1 => 0	1 => 1	- 1	- 0
15	1	0 1	0 1	0 => 0	1 => 1	0 -	- 0

Table for J1 K1

	Q1' Q2'	Q1' Q2	Q1 Q2	Q1 Q2'
w'	0 -	0 -	- 0	- 0
w	1 -	0 -	- 1	- 0

J1 = w Q2'
K1 = w Q2

Table for J2 K2

	Q1' Q2'	Q1' Q2	Q1 Q2	Q1 Q2'
w'	0 -	- 1	- 0	1 -
w	0 -	- 0	- 0	0 -

J2 = w' Q1
K2 = w' Q1'

NAME:	ENTRY No.:

4. Give entity and architecture description in VHDL for a 4 to 2 encoder, a purely combinational circuit. It has two output ports. If exactly one of the 4 inputs is 1, it outputs a 2 bit code on the first port. Otherwise, it indicates an error on the second port. The 2 bit code on the first port indicates which input is 1. In case of error, the code is ignored.

[2]

Solution:

```
ENTITY encoder_4_2 IS
  PORT (X: IN      bit_vector (3 DOWNTO 0);
           code: OUT  bit_vector (1 DOWNTO 0);
           error:  OUT  bit
          );
END encoder_4_2;

ARCHITECTURE casestmt OF encoder_4_2 IS
BEGIN
 PROCESS (X)
   BEGIN
     CASE X IS
       WHEN “1000”       => code <= “11”; error <= ‘0’;
       WHEN “0100”       => code <= “10”; error <= ‘0’;
       WHEN “0010”       => code <= “01”; error <= ‘0’;
       WHEN “0001”       => code <= “00”; error <= ‘0’;
       WHEN OTHERS    => code <= “00”; error <= ‘1’;
     END CASE;
   END PROCESS;
END ARCHITECTURE casestmt;

alternatively,

ARCHITECTURE ssa OF encoder_4_2 IS
BEGIN
   error <= ‘0’ WHEN (X = “1000” or X = “0100” or X = “0010” or X = “0001”) ELSE
          ‘1’;
  WITH X SELECT
    code <= "11" WHEN "1000",
          "10" WHEN "0100",
          "01" WHEN "0010",
          "00" WHEN OTHERS;
END ARCHITECTURE ssa;
```