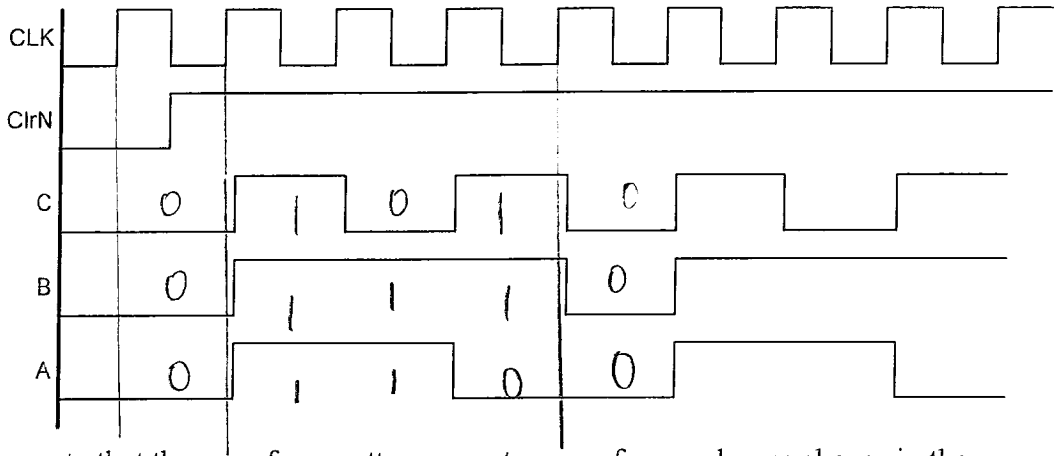
1. (10%) Implement a positive-edge triggered J-K flip-flop with only S-R Latches, AND, OR, and NOT gates. (Please draw the circuits.)

2. (20%) Design a waveform generator which generates the following waveforms:



Please note that the waveform pattern repeats every four cycles, as shown in the above figure.

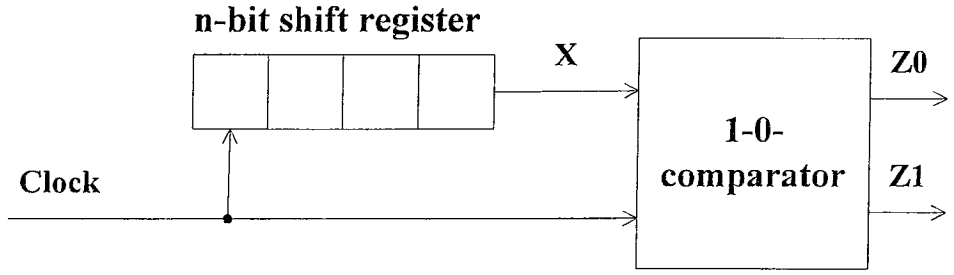
(a) (5%) This waveform generator can be designed as a sequential circuit with three positive-edge triggered flip-flops. Derive the state graph of this circuit. (Hint: please present the state as CBA. For example, CBA=000 can be presented as (000) in the state graph.)

(b) (10%) Flip-flop C is a T flip-flop, flip-flop B is an S-R flip-flop, and flip-flop A is a D flip-flop. Please design the excitation table, which is shown as follows:

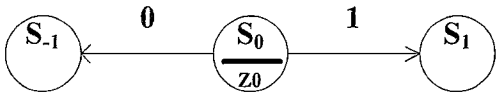
CBA	T_C	S_B	R_B	D_A
000				
001				
⋮				

(c) (5%) Realize it with one positive-edge triggered T flip-flop, one positive-edge triggered S-R flip-flop, one positive-edge triggered D flip-flop, and basic logic gates. (AND, OR, NOT, NAND, NOR) (Please draw the circuits.)

3. (15%) In this figure, an n-bit number is initially loaded in the shift register. For each clock, the number is shifted to the right by one bit. The *1-0-comparator* has one input (X) and two outputs (Z_0, Z_1). If the total number of zeros in X is larger than or equal to the total number of ones , then $Z_0 = 1$ and $Z_1=0$. Otherwise, $Z_0=0$ and $Z_1=1$.

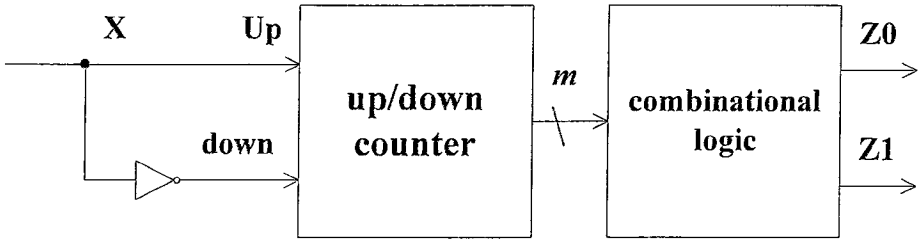


(a) (5 %) For n=4, please complete the following Moore state graph of the 1-0-comparator. Assume that S_0 is the initial state (Z_0=1, Z_1=0), which also represents equal number of zeros and ones. (hint: 9 states)

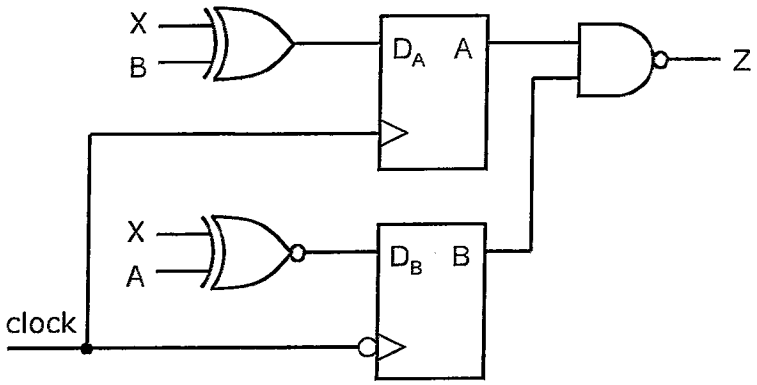


(b) (5 %) For n = 4, please reduce the number of states in part **A**. Show your state graph of minimum states.

(c) (5 %) Suppose that we want to design an n-bit 1-0-comparator using an m-bit *up/down counter*. The *Up* input of the counter is connected to X and the *Down* input is connected to X'. Please specify the requirement for m (in terms of n) such that the comparator can function correctly.



4. (15%) Consider the following circuit with two D flip-flops (positive-edge triggered for D_A and negative-edge triggered for D_B). Let A=B=0 be the initial state and the clock start at a low level. Assume the flip flops have propagation delay 2ns and setup time 1ns, and all the gates have delay 3ns. Also assume that the input X of this circuit can be given every half clock cycle and settles to its correct value fast enough.



$A^+ = XB' + X'B$

$B^+ = XA' + X'A$

$Z = (AB)'$

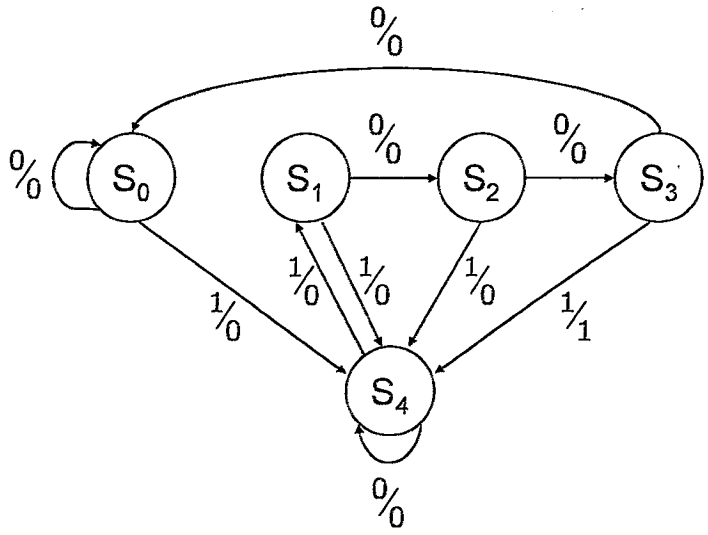
(a) (5%) Please draw the state graph of the circuit. (Suggestion: A good way, among other possibilities, to understand the circuit is to have 8 states.)

(b) (5%) What is the minimum clock period of the circuit?

(c) (2%) Consider the circuit same as above expect for making D_B positive-edge triggered. What is the minimum clock period of this new circuit? (Assume the input X is given every clock cycle and settles to its correct value fast enough.)

(d) (3%) Does the new circuit of (c) starting with A=B=0 behave the same as the prior circuit (in terms of input/output sequences)? If not, please provide an input sequence that differentiates these two circuits. That is, the corresponding output sequences produced by these two circuits are different under this input sequence.

5. (10%) Consider the Mealy machine M with the following state graph. Assume state S_0 is the initial state. We say that a binary input string (or input sequence) is **accepted** by M if and only if the last bit of the corresponding output sequence is 1.



0 0 0 0. 0 | 0.

Example

Time:	0 1 2 3 4 5 6
X =	0 1 1 0 0 1 0
Z =	0 0 0 0 0 1 0

The input sequence "0110010" is not accepted by M because the last output is 0.

(a) (5%) Can M accept any (finite) input sequence with an even number of 1's? What kind of input sequences does M accept?

(b) (5%) Please draw the state graph of a Moore machine that accepts the same set of input sequences as M.

6. (30%) Design a sequential circuit to decode Huffman code. The Huffman codeword tree is given in Figure a.

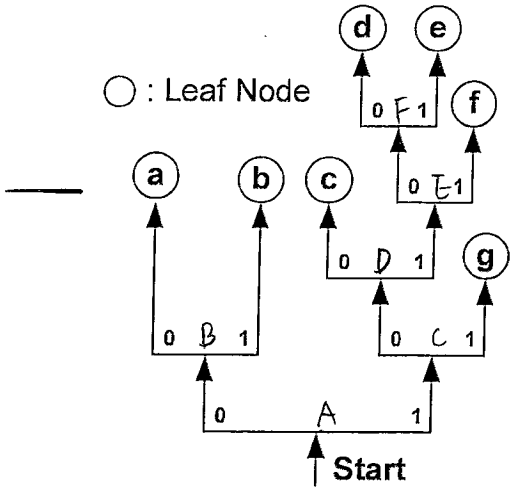


Figure a. Huffman tree.

The decoding process consists of the following steps:

I. Read the input sequence X and traverse the given Huffman tree until a leaf node is reached.

II. When the leaf node of tree is reached, the output Z is 1. Then, back to start point.

For Example:

X	1	0	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	1	0	0
Z	0	0	0	1	0	1	0	1	0	1	0	0	1	0	1	0	0	0	0	1

(a) (5%) Fill the blank of state table.

Time	Input Sequence Received (Least Significant Bit First)	Present State	Next State		Present Output (Z)	
			X= 0	X=1	X=0	X=1
t_0	reset	A				
t_1	0	B				
	1	C				
t_2	10	D				
t_3	101	E				
t_4	1010	F				

(b) (10%) Reduce the (a) state table to minimum number of states and draw Mealy State Graph.

(c) (15%) Use D flip-flop to realize the circuit with minimum SOP for combinational circuit. Please also draw the circuits. Assign states A, B, C,... in the order of 000, 001, 010,

7. (bonus 5 %) The following verilog code describes a 4-bit Gray-code-to-binary-code converter. Draw the circuit and label each gate corresponding to the gate name in the code.

```
module Gray2Bin ( g, b );
    input [3:0] g;
    output [3:0] b;

    buf b3 (b[3], g[3]);
    xor x2 (b[2], b[3], g[2]);
    xor x1 (b[1], b[2], g[1]);
    xor x0 (b[0], b[1], g[0]);
endmodule
```