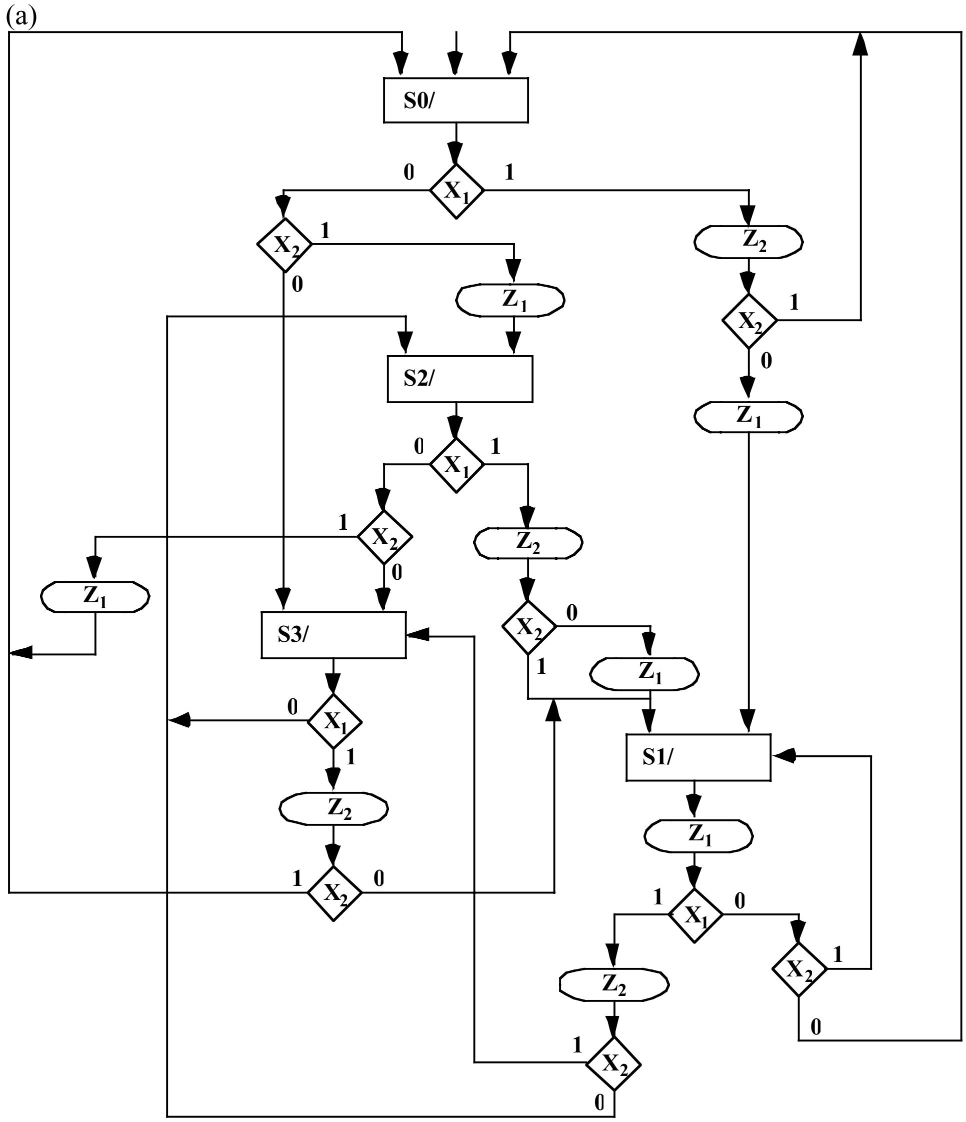
The University of Alabama in Huntsville
Electrical & Computer Engineering
CPE/EE 422/522
Spring 2004
Homework #5 Solution

5.1 (a) Construct an SM chart equivalent to the following state table. Test only one variable in each decision box. Try to minimize the number of decision boxes.
(b) Write a VHDL description of the state machine based on the SM chart.

Present State		Next State					Output Z_1Z_2			
	X_1X_2=	00	01	10	11	X_1X_2=	00	01	10	11
S0		S3	S2	S1	S0		00	10	11	01
S1		S0	S1	S2	S3		10	10	11	11
S2		S3	S0	S1	S1		00	10	11	01
S3		S2	S2	S1	S0		00	00	01	01

(a)



(b)

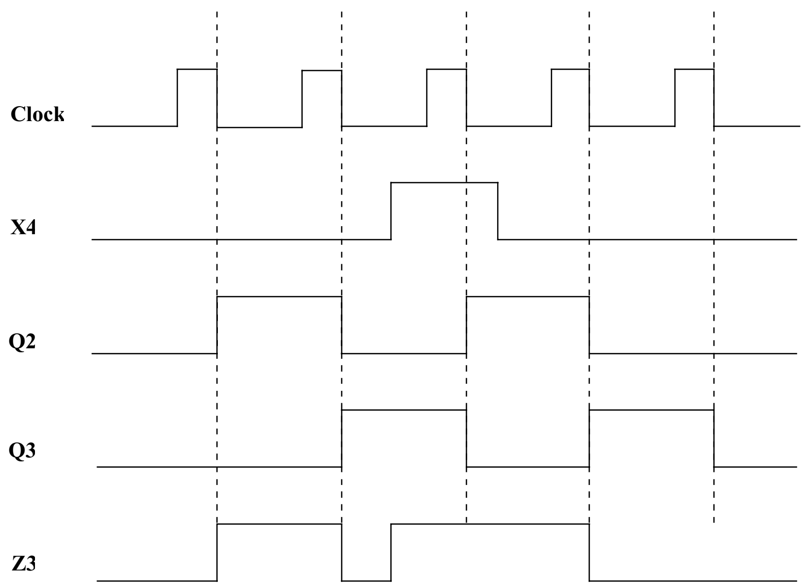
```
entity 5_1 is
  port (CLK, X1, X2 : in bit;
        Z1, Z2 ; out bit);
end 5_1;

architecture Smbehave of 5-1 is
  type states is {S0, S1, S2, S3};
  signal state, next_state : states;
begin
  process (X1, X2, state)
  begin
    Z1 <= ‘0’;  Z2 <= ‘0’;
    case state is
      when S0 => if (X1 = ‘0’) then
                   if (X2 = ‘0’) then
                     next_state <=  S3;
                   else
                     next_state <= S2;
                   end if;
                 else
                   if (X2 = ‘0’) then
                     Z1 <= ‘1’;
                     Z2 <= ‘1’;
                     next_state <= S1;
                   else
                     Z2 <= ‘1’;
                     next_state <= S0;
                   end if
                 end if;
      when S1 => if (X1 = ‘0’) then
                   Z1 <= ‘1’;
                   if (X2 = ‘0’) then
                     next_state <=  S0;
                   else
                     next_state <= S1;
                   end if;
                 else
                   Z1 <= ‘1’; Z2 <= ‘1’;
                   if (X2 = ‘0’) then
                     next_state <= S2;
                   else
                     next_state <= S3;
                   end if
                 end if;
      when S2 => if (X1 = ‘0’) then
                   if (X2 = ‘0’) then
                     next_state <=  S3;
                   else
                     Z1 <= ‘1’;
                     next_state <= S0;
                   end if;
                 else
                   next_state <= S1;
                   Z2 <= ‘1’
                   if (X2 = ‘0’) then
                     Z1 <= ‘1’;;
```

```
                 end if
               end if;
    when S3 => if (X1 = '0') then
                 next_state <= S2;
               else
                 if (X2 = '0') then
                   Z2 <= '1';
                   next_state <= S1;
                 else
                   Z2 <= '1';
                   next_state <= S0;
                 end if
               end if;
  end case;
end process;
process (CLK)
begin
  if (CLK = '1' and CLK'event) then
    state <= next_state;
  end if;
end process;
end Smbehave;
```

5.6 For the given SM chart:

(a) Complete the following timing diagram (assume that X1 = 1, X2 = 0, X3 = 0, X5 = 1, and X4 is as shown)/ Flip-flops change state on falling edge of clock.



10.1 (a) Determine the necessary inputs to the following network to test for u stuck-at-0.
(b) For this set of inputs, determine which other stuck-at faults can be tested.
(c) Repeat (a) and (b) for r stuck-at-1.

(a) & (b)	A	B	C	D	Faults Tested
	d	1	0	1	u s-a-0, b s-a-0, d s-a-0, w s-a-0
	d	1	1	0	u s-a-0, b s-a-0, c s-a-0, w s-a-0
	d	1	1	1	u s-a-0

(c)	A	B	C	D	Faults Tested
	0	1	0	0	a s-a-1, c s-a-1, d s-a-1, q s-a-1, r s-a-1, u s-a-1, v s-a-1, w s-a-1, p s-a-0

10.3 For the following network, find a minimum number of test vectors that will test all s-a-0 and s-a-1 faults at the AND and OR gate inputs. For each test vector, specify the values of A, B, C, and D, and the stuck-at faults that are tested.

A	B	C	D	Faults Tested
0	1	d	1	k s-a-0, l s-a-0, m s-a-0, r s-a-0, z s-a-0
1	1	0	d	e s-a-0, f s-a-0, g s-a-0, p s-a-0, z s-a-0
1	0	1	d	h s-a-0, i s-a-o, j s-a-0 , q s-a-0, z s-a-0
0	0	1	1	h s-a-1, l s-a-1, p s-a-1, q s-a-1, r s-a-1, z s-a-1
0	1	0	0	e s-a-1, m s-a-1, p s-a-1, q s-a-1, r s-a-1, z s-a-1
1	0	0	d	f s-a-l, j s-a-1, p s-a-1, q s-a-1, r s-a-1, z s-a-1
1	1	1	1	g s-a-1, i s-a-1, k s-a-1, p s-a-1 , q s-a-1, r s-a-1, z s-a-1

10.7 State graphs for two sequential machines are given below. The first graph represents a correctly functioning machine, and the second represents the same machine with a malfunction. Assuming that the two machines can be reset to their starting states (S0 and T0), determine the shortest input sequence that will distinguish the two machines.

Input:	0	0	0	1	1	1	1
Correct Output:	1	1	1	0	0	0	1
Incorrect Output:	1	1	1	0	0	0	0