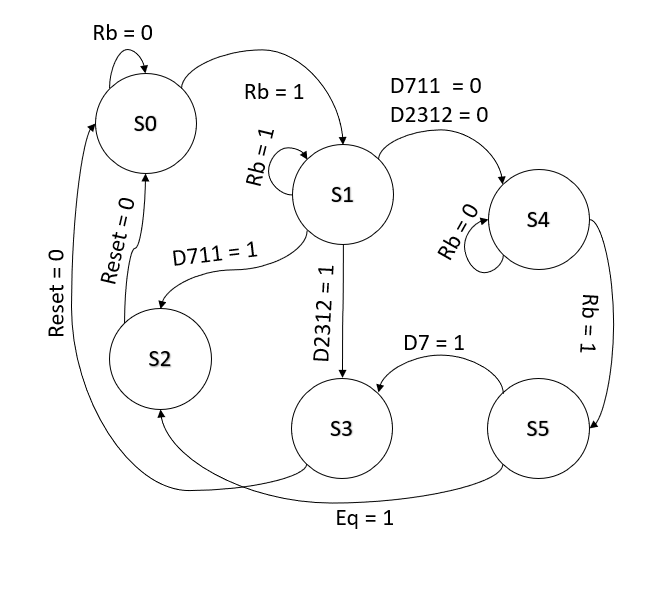
**Report for Dice Game**

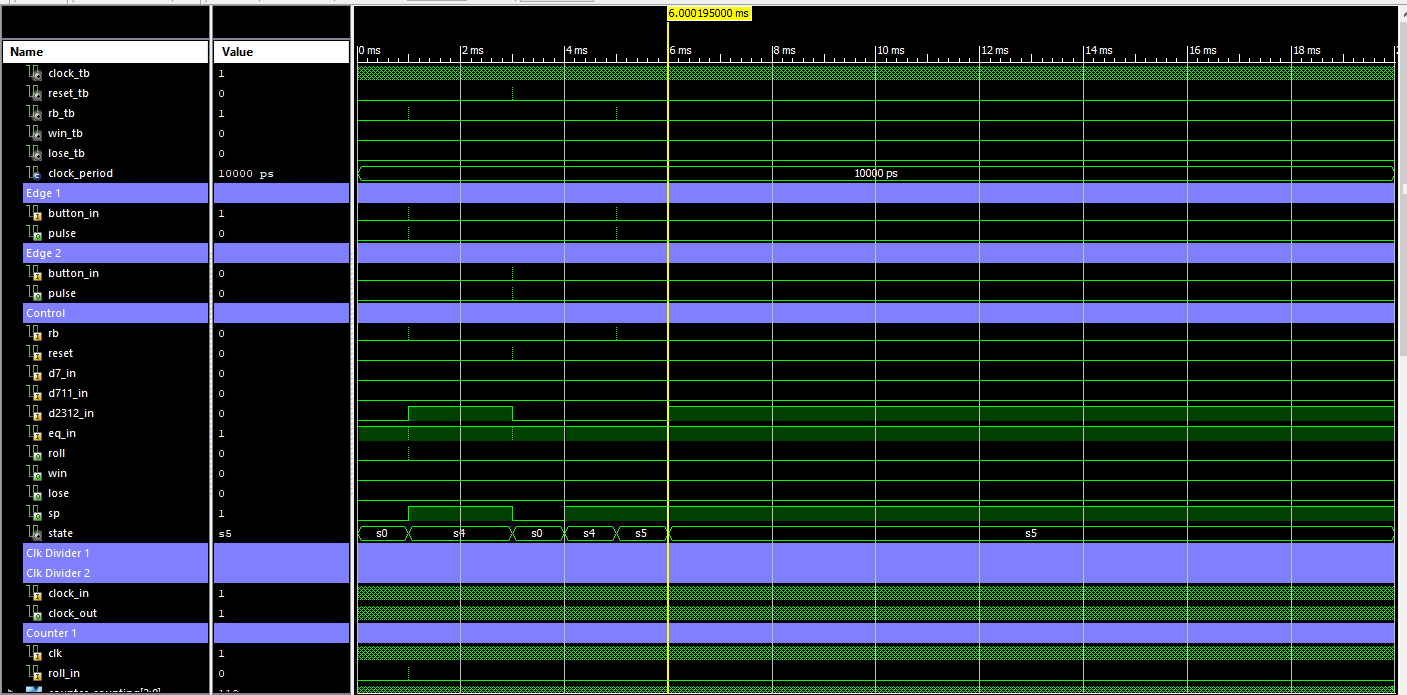
**Question1:**

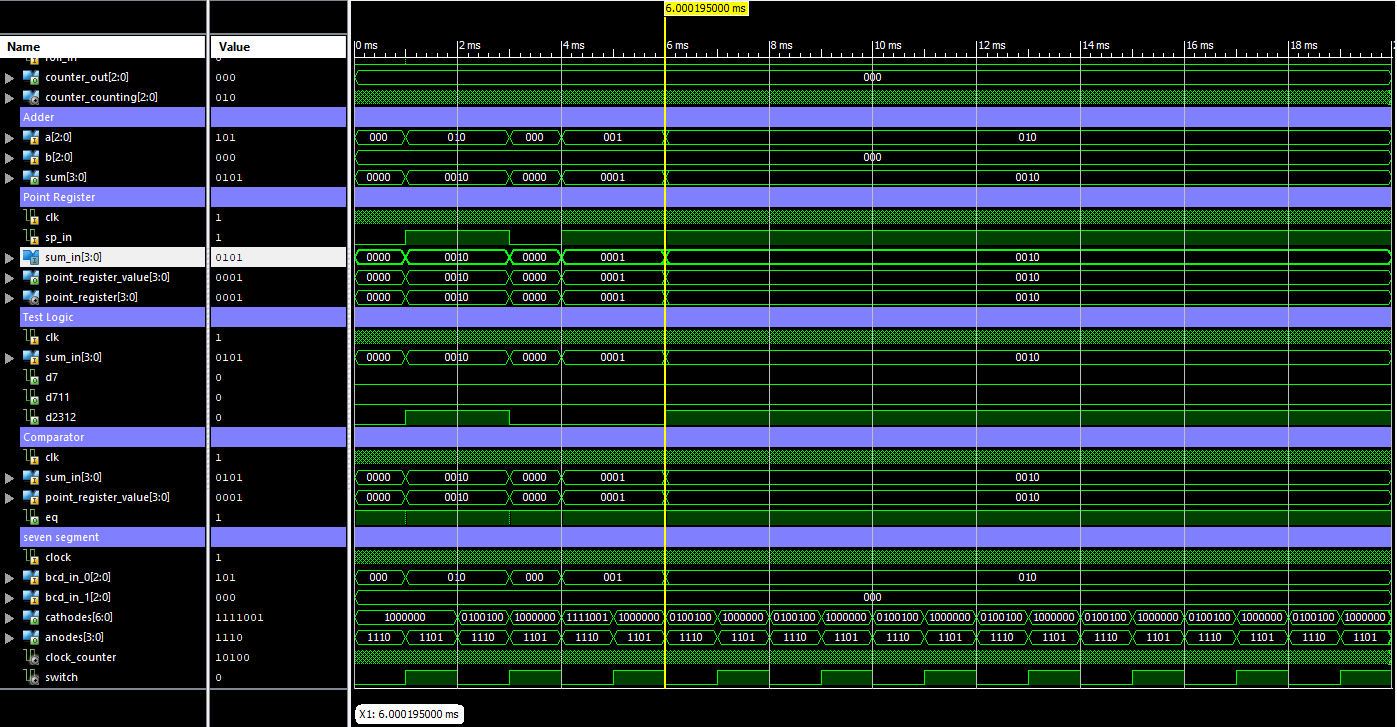
The FSM of the flow chart is as follows



**Question 2:**

The Simulation are as follows:







Win and Lose conditions

The Top Module is as follows:

|  |
| --- |
| ----------------------------------------------------------------------------------  -- Company:  -- Engineer:  --  -- Create Date: 21:32:06 10/25/2024  -- Design Name:  -- Module Name: top - Behavioral  -- Project Name:  -- Target Devices:  -- Tool versions:  -- Description:  --  -- Dependencies:  --  -- Revision:  -- Revision 0.01 - File Created  -- Additional Comments:  --  ----------------------------------------------------------------------------------  library IEEE;  use IEEE.STD\_LOGIC\_1164.ALL;  -- Uncomment the following library declaration if using  -- arithmetic functions with Signed or Unsigned values  use IEEE.NUMERIC\_STD.ALL;  -- Uncomment the following library declaration if instantiating  -- any Xilinx primitives in this code.  --library UNISIM;  --use UNISIM.VComponents.all;  entity top is  Port(  clock : in std\_logic;  Reset : in std\_logic;  Rb : in std\_logic;  Win : out std\_logic := '0';  Lose : out std\_logic := '0';  cathodes\_out : out std\_logic\_vector(6 downto 0) := (others => '0');  anodes\_out : out std\_logic\_vector(3 downto 0) := (others => '1')  );  end top;  architecture Behavioral of top is  component Edge\_Detector  Port(  CLK\_SYS : in std\_logic;  BUTTON\_IN : in std\_logic;  PULSE : out std\_logic  );  end component;    component Control is  Port(  clk : in std\_logic;  Rb : in std\_logic;  Reset : in std\_logic;  D7\_in : in std\_logic;  D711\_in : in std\_logic;  D2312\_in : in std\_logic;  Eq\_in : in std\_logic;  Roll : out std\_logic;  Win : out std\_logic;  Lose : out std\_logic;  Sp : out std\_logic  );  end component;    component onetosix\_counter  Port(  clk : in std\_logic;  reset : in std\_logic;  Roll\_in : in std\_logic;  counter\_out : out std\_logic\_vector(2 downto 0)  );  end component;    component clock\_divider is  generic(  FPGA\_CLK\_FREQUENCY : integer := 100000000;  REQUIRED\_FREQUENCY : integer := 50000000  );  Port (  clock\_in : in STD\_LOGIC;  clock\_out : out STD\_LOGIC  );  end component;    component Adder  Port (  A : in STD\_LOGIC\_VECTOR(2 downto 0); -- 4-bit input A  B : in STD\_LOGIC\_VECTOR(2 downto 0); -- 4-bit input B  SUM : out STD\_LOGIC\_VECTOR(3 downto 0) -- 5-bit Sum Output  );  end component;    component point\_register  Port(  clk : in STD\_LOGIC;  Sp\_in : in STD\_LOGIC;  SUM\_IN : in std\_logic\_vector(3 downto 0);  point\_register\_value : out std\_logic\_vector(3 downto 0)  );  end component;    component comparator  Port(  clk : in std\_logic;  sum\_in : in std\_logic\_vector(3 downto 0);  point\_register\_value : in std\_logic\_vector(3 downto 0);  Eq : out std\_logic  );  end component;    component Test\_Logic  Port(  clk : in std\_logic;  sum\_in : in std\_logic\_vector(3 downto 0);  D7 : out std\_logic;  D711 : out std\_logic;  D2312 : out std\_logic  );  end component;    component bcdto7seg  Port (  clock : in std\_logic;  bcd\_in\_0 : in std\_logic\_vector (2 downto 0);  bcd\_in\_1 : in std\_logic\_vector (2 downto 0);  cathodes : out std\_logic\_vector (6 downto 0);  anodes : out std\_logic\_vector(3 downto 0)  );  end component;    signal start\_button : std\_logic := '0';  signal reset\_button : std\_logic := '0';  signal Roll\_out : std\_logic := '0';  signal Win\_out : std\_logic := '0';  signal Lose\_out : std\_logic := '0';  signal counter\_one\_out : std\_logic\_vector(2 downto 0) := (others => '0');  signal counter\_two\_out : std\_logic\_vector(2 downto 0) := (others => '0');  signal clock\_divided\_1 : std\_logic := '0';  signal clock\_divided\_2 : std\_logic := '0';  signal Sum\_out : std\_logic\_vector(3 downto 0) := (others => '0');  signal point\_register\_val : std\_logic\_vector(3 downto 0) := (others => '0');  signal Eq\_out : std\_logic := '0';  signal D7\_out : std\_logic := '0';  signal D711\_out : std\_logic := '0';  signal D2312\_out : std\_logic := '0';  signal Sp\_out : std\_logic := '0';  begin    uut : Edge\_Detector  port map(  CLK\_SYS => clock,  BUTTON\_IN => Rb,  PULSE => start\_button  );    uut2 : Edge\_Detector  port map(  CLK\_SYS => clock,  BUTTON\_IN => Reset,  PULSE => reset\_button  );    uut3 : Control  port map(  clk => clock,  Rb => start\_button,  Reset => reset\_button,  D7\_in => D7\_out,  D711\_in => D711\_out,  D2312\_in => D2312\_out,  Eq\_in => Eq\_out,  Roll => Roll\_out,  Win => Win,  Lose => Lose,  Sp => Sp\_out  );    --uut4 : clock\_divider  --generic map(  -- FPGA\_CLK\_FREQUENCY => 100000000,  -- REQUIRED\_FREQUENCY => 100000000  --)  --port map(  -- clock\_in => clock,  -- clock\_out => clock\_divided\_1  --);    uut5 : onetosix\_counter  port map(  clk => clock,  reset => reset\_button,  Roll\_in => Roll\_out,  counter\_out => counter\_one\_out  );    uut6 : clock\_divider  generic map(  FPGA\_CLK\_FREQUENCY => 100000000,  REQUIRED\_FREQUENCY => 50000000  )  port map(  clock\_in => clock,  clock\_out => clock\_divided\_2  );    uut7 : onetosix\_counter  port map(  clk => clock\_divided\_2,  reset => reset\_button,  Roll\_in => Roll\_out,  counter\_out => counter\_two\_out  );    uut8 : Adder  port map(  A => counter\_one\_out,  B => counter\_two\_out,  SUM => Sum\_out  );    uut9 : point\_register  port map(  clk => clock,  Sp\_in => Sp\_out,  SUM\_IN => Sum\_out,  point\_register\_value => point\_register\_val  );    uut10 : comparator  port map(  clk => clock,  sum\_in => Sum\_out,  point\_register\_value => point\_register\_val,  Eq => Eq\_out  );    uut11 : Test\_Logic  Port map(  clk => clock,  sum\_in => Sum\_out,  D7 => D7\_out,  D711 => D711\_out,  D2312 => D2312\_out  );    uut12 : bcdto7seg  Port map (  clock => clock,  bcd\_in\_0 => counter\_one\_out,  bcd\_in\_1 => counter\_two\_out,  cathodes => cathodes\_out,  anodes => anodes\_out  );      end Behavioral; |

**Question 3:**

The Constraint Requirements for the FPGA of Nexys 3 are as follows:

|  |
| --- |
| ## This file is a general .ucf for Nexys3 rev B board  ## To use it in a project:  ## - remove or comment the lines corresponding to unused pins  ## - rename the used signals according to the project  ##Clock signal  Net "clock" LOC=V10 | IOSTANDARD=LVCMOS33;  ## 7 segment display  Net "cathodes\_out(0)" LOC = T17 | IOSTANDARD = LVCMOS33; #Bank = 1, pin name = IO\_L51P\_M1DQ12, Sch name = CA  Net "cathodes\_out(1)" LOC = T18 | IOSTANDARD = LVCMOS33; #Bank = 1, pin name = IO\_L51N\_M1DQ13, Sch name = CB  Net "cathodes\_out(2)" LOC = U17 | IOSTANDARD = LVCMOS33; #Bank = 1, pin name = IO\_L52P\_M1DQ14, Sch name = CC  Net "cathodes\_out(3)" LOC = U18 | IOSTANDARD = LVCMOS33; #Bank = 1, pin name = IO\_L52N\_M1DQ15, Sch name = CD  Net "cathodes\_out(4)" LOC = M14 | IOSTANDARD = LVCMOS33; #Bank = 1, pin name = IO\_L53P, Sch name = CE  Net "cathodes\_out(5)" LOC = N14 | IOSTANDARD = LVCMOS33; #Bank = 1, pin name = IO\_L53N\_VREF, Sch name = CF  Net "cathodes\_out(6)" LOC = L14 | IOSTANDARD = LVCMOS33; #Bank = 1, pin name = IO\_L61P, Sch name = CG  #Net "cathodes\_out(7)" LOC = M13 | IOSTANDARD = LVCMOS33; #Bank = 1, pin name = IO\_L61N, Sch name = DP  Net "anodes\_out(0)" LOC = N16 | IOSTANDARD = LVCMOS33; #Bank = 1, pin name = IO\_L50N\_M1UDQSN, Sch name = AN0  Net "anodes\_out(1)" LOC = N15 | IOSTANDARD = LVCMOS33; #Bank = 1, pin name = IO\_L50P\_M1UDQS, Sch name = AN1  Net "anodes\_out(2)" LOC = P18 | IOSTANDARD = LVCMOS33; #Bank = 1, pin name = IO\_L49N\_M1DQ11, Sch name = AN2  Net "anodes\_out(3)" LOC = P17 | IOSTANDARD = LVCMOS33; #Bank = 1, pin name = IO\_L49P\_M1DQ10, Sch name = AN3  ## Leds  Net "Win" LOC = U16 | IOSTANDARD = LVCMOS33; #Bank = 2, pin name = IO\_L2P\_CMPCLK, Sch name = LD0  Net "Lose" LOC = V16 | IOSTANDARD = LVCMOS33; #Bank = 2, pin name = IO\_L2N\_CMPMOSI, Sch name = LD1  ## Switches  Net "Rb" LOC = T10 | IOSTANDARD = LVCMOS33; #Bank = 2, pin name = IO\_L29N\_GCLK2, Sch name = SW0  Net "Reset" LOC = T9 | IOSTANDARD = LVCMOS33; #Bank = 2, pin name = IO\_L32P\_GCLK29, Sch name = SW1 |

These includes 2 switches, 2LED and 2 seven segments which are refreshing at 1ms response time.