**Lab9 Final Report**

**20160410 윤석훈**

library ieee;  
use ieee.std\_logic\_1164.all;  
use ieee.std\_logic\_unsigned.all;  
entity lab9 is  
port(sw : in std\_logic\_vector(2 downto 0);  
 key : in std\_logic\_vector(0 downto 0);  
 CLOCK\_50 : in std\_logic;  
 ledr : out std\_logic\_vector(2 downto 0);  
 HEX3, HEX2, HEX1, HEX0 : out std\_logic\_vector(6 downto 0));  
end entity;  
architecture lb9 of LAB9 is  
function matching(binary : in integer range 0 to 9)   
 return std\_logic\_vector is  
 begin  
  if binary=0 then return "1000000";  
  elsif binary=1 then return "1111001";  
  elsif binary=2 then return "0100100";  
  elsif binary=3 then return "0110000";  
  elsif binary=4 then return "0011001";  
  elsif binary=5 then return "0010010";  
  elsif binary=6 then return "0000010";  
  elsif binary=7 then return "1011000";  
  elsif binary=8 then return "0000000";  
  else return "0011000";  
 end if;  
end function;  
Type State\_type is (initial, run, stop);  
Signal Q : State\_type :=initial;  
Signal integersignal: integer :=0;  
Signal clk\_cnt : integer :=0;

Signal T:integer:=0;   
Signal clear,r :std\_logic;  
Signal A,B,C,D: integer;  
begin  
process(clock\_50, Q)  
begin  
 if (clock\_50'event and clock\_50 = '1') then   
  if (Q = run) then  
   clk\_cnt<=clk\_cnt+1;  
   clear<='0';  
   r <='1';   
  elsif (Q = initial) then   
   clear <= '1';  
   r <='0';   
   T <=0;  
  else  
   clear <= '0';  
   r<= '0';  
  end if;  
  if (clk\_cnt > 500000) then   
  clk\_cnt <= 0;  
  T<=T+1;  
  end if;  
 end if;  
end process;  
process(sw,key)  
begin  
 if sw(0)='1' and key(0)= '0' then  
  Q<= initial;  
 elsif  sw(1)='1' and key(0)='0' then  
  Q<= stop;  
 elsif sw(2)='1' and key(0) = '0' then   
  Q<= run;  
 end if;  
end process;  
D <= T/1000;  
C <= T/100- D\*10;  
B <= T/10 - D\*100 -C\*10;  
A <= T - D\*1000-C\*100-B\*10;  
hex3<= matching(D);  
hex2<= matching(C);  
hex1 <= matching(B);  
hex0 <= matching(A);  
end architecture;

**Explanation:**

2개의 Process문으로 구성하였다. Clock\_50이 50MHz의 Oscillator이므로, clk\_cnt 는 1초에 5\*10^7번 증가한다. 0.01초 단위로 숫자가 커져야 하므로, clk\_cnt 가 5\*10^7\*0.01=500000 보다 커질 때 다시 0으로 초기화하고, 숫자를 표시하기 위한 T를 1씩 올려나간다. T가 0.01초에 1씩 증가하므로, 10초에 1씩 증가하는 HEX3(Signal D)은 T를 1000으로 나눈 몫으로 설정한다. 1초에 1씩 증가하는 HEX2(Signal C)은 T를 100으로 나눈 몫에서 10초 자리의 10배를 빼면 되므로 T/100-D\*10으로 설정한다. 마찬가지로 B는 T/10에서 100\*D과 10\*C의 합을 뺀다. 이는, A,B,C,D모두 한자릿수이기 때문이다.

Initial, stop, run 3가지를 State-Type로 설정하였고, 각 시점의 State는 Signal Q가 받는다.