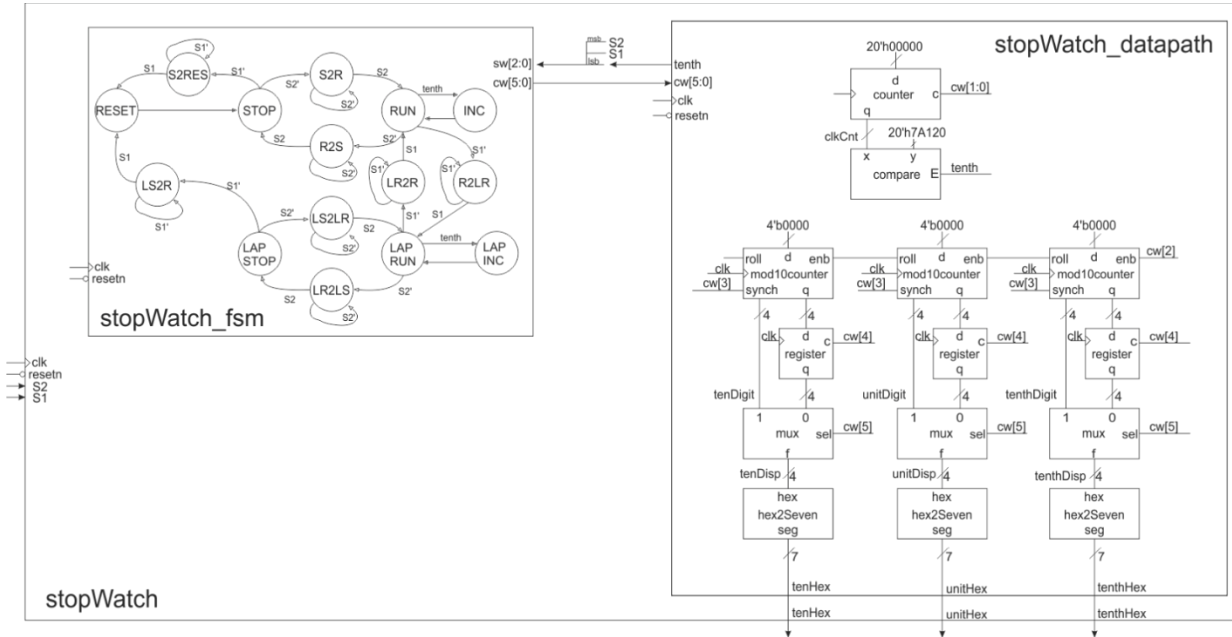




EENG 498 – Embedded Systems II

Stopwatch Datapath and Control



	cw[5] 2x1 mux	cw[4] lap register	cw[3] mod reset	10	cw[2] mod10 count	cw[1:0] timer counter
	0 = mod10	1 = load	1 = reset		1 = count up	11 = load
	1 = register	0 = hold	0 = hold		0 = hold	10 = count up
						01 = not used
						00 = hold
RESET						
STOP						
S2RES						
S2R						
RUN						
R2LR						
R2S						
INC						
LAPRUN						
LR2R						
LR2LS						
LAPINC						
LAPSTOP						
LS2R						
LS2LR						



Complete the VHDL code for the top-level stopwatch component.

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use work.stopwatch_package.all;

entity stopWatch is
  Port (   clk:    in STD_LOGIC;
          resetn: in STD_LOGIC;
          S1, S2: in STD_LOGIC;
          tenHex, unitHex, tenthHex:out STD_LOGIC_VECTOR(6 downto 0));

end stopWatch;

architecture Structure of stopWatch is

  signal cw: STD_LOGIC_VECTOR(CW_VECTOR_LENGTH-1 downto 0);
  signal sw: STD_LOGIC_VECTOR(SW_VECTOR_LENGTH-1 downto 0);
  signal tenth : STD_LOGIC;

begin

  sw(0) <= tenth;
  sw(1) <= S1;
  sw(2) <= S2;

  dpsw: stopWatch_datapath
    port map (   clk => _____,
               resetn => _____,
               cw => _____,
               tenth => _____,
               tenHex => _____,
               unitHex => _____,
               tenthHex => _____ );

  cusw: stopWatch_fsm
    port map(   clk => _____,
              resetn => _____,
              cw => _____,
              sw => _____ );

end Structure;
```