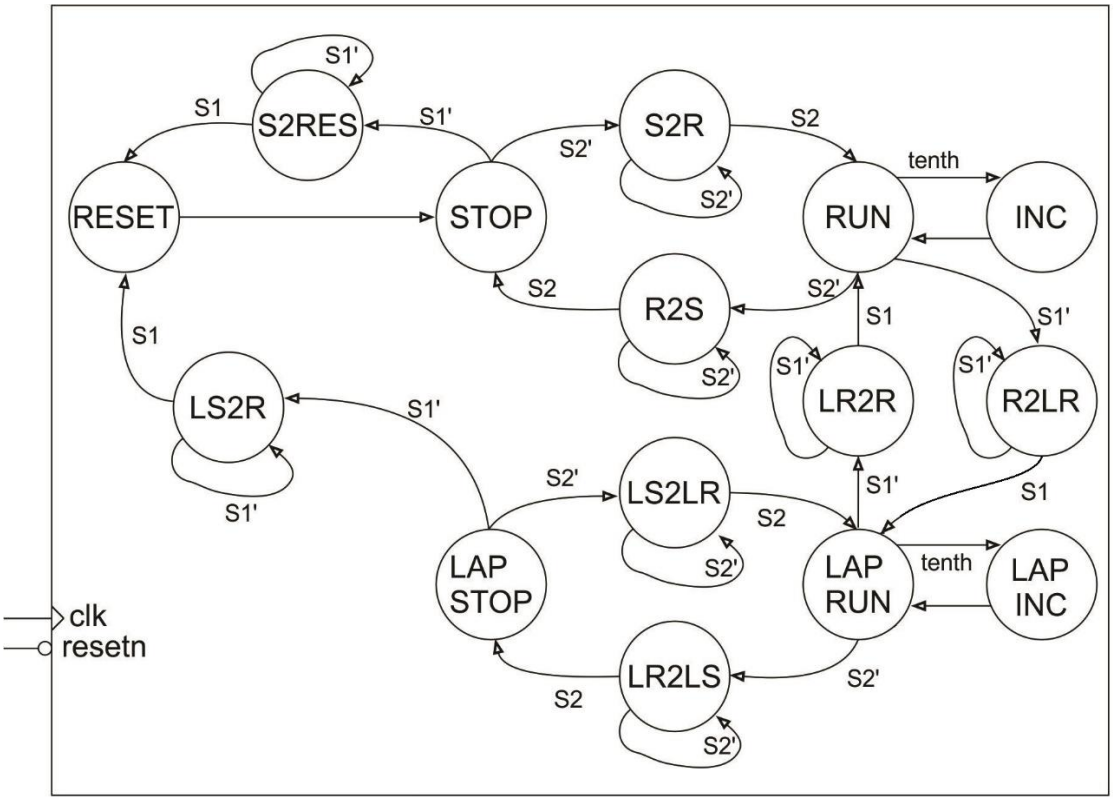
EENG 498 – Embedded Systems II

Stopwatch Control Unit

The control unit has a lot of wait states that allow a (slow) human to interface to the speedy digital circuit. These wait states, wait for the user to release the button. The buttons are active low.




```
--The following declaration is from the package file
type state_type is (       RESET_STATE, S2RESET_STATE, STOP_STATE, S2R_STATE, RUN_STATE,
                           R2LR_STATE, R2S_STATE, INC_STATE, LAPINC_STATE, LAPRUN_STATE, LR2R_STATE,
                           LR2LS_STATE, LAPSTOP_STATE, LS2R_STATE, LS2LR_STATE);

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use work.stopWatch_package.all;

entity stopWatch_fsm is
    PORT (  clk : in  STD_LOGIC;
            resetn : in  STD_LOGIC;
            sw: in STD_LOGIC_VECTOR(_____________________ - 1 downto 0);
            cw: out STD_LOGIC_VECTOR (_____________________ - 1 downto 0));
end stopWatch_fsm;
```

```
architecture Behavioral of stopWatch_fsm is
    signal tenth, S1, S2: STD_LOGIC;
    signal state: state_type;

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

    state_proces: process(clk)
    begin
       if (rising_edge(clk)) then
           if (resetn = '0') then
              state <= RESET_STATE;
           else
              case state is
                  when RESET_STATE =>
                     state <= STOP_STATE;

                  when STOP_STATE =>









              end case;
           end if;
       end if;
    end process;

    output_process: process (state)
    begin
       case state is
           when RESET_STATE    => cw <= "001011";
              . . .
       end case;
    end process;
end structure;
```