SCROLLING LED DISPLAY

GURAMRITPAL BAL

ANJA JAEGER

KAROLOS MISHREKY

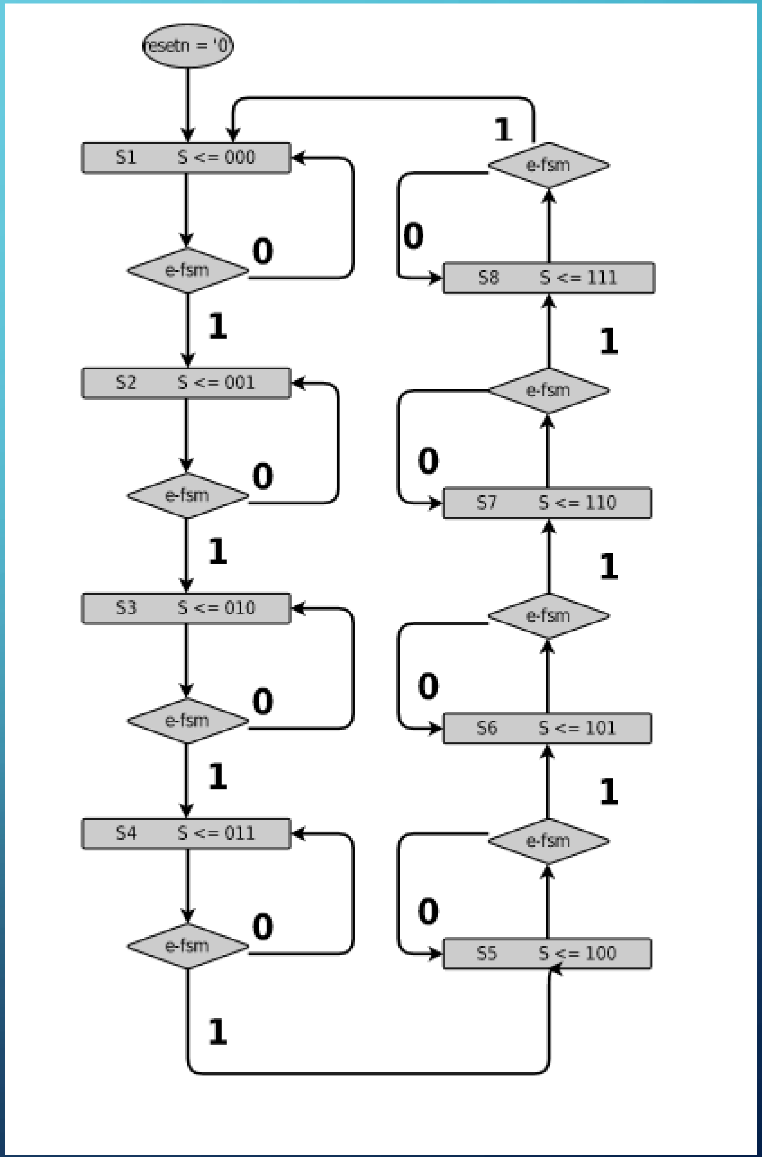
SAPAN PATEL

MOTIVATIONS

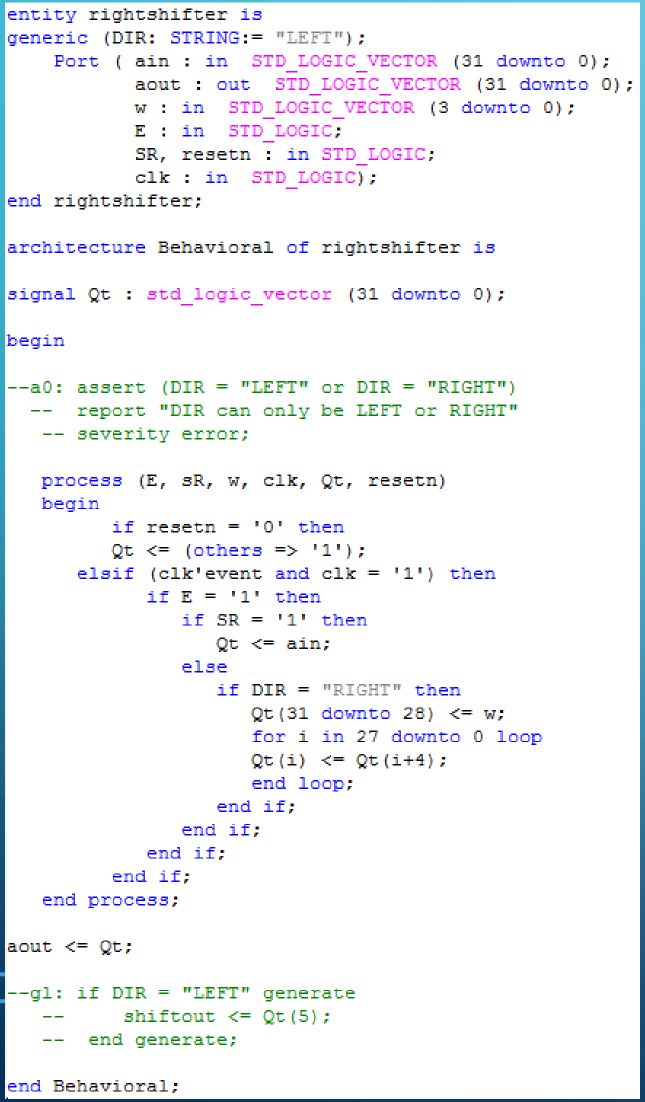
- Displaying information for people to see at a glance.
- Utilizing a display technology that can be seen at all times of the day.
- Further our knowledge in how specific parts from the class interact with each other.

FINITE STATE MACHINE

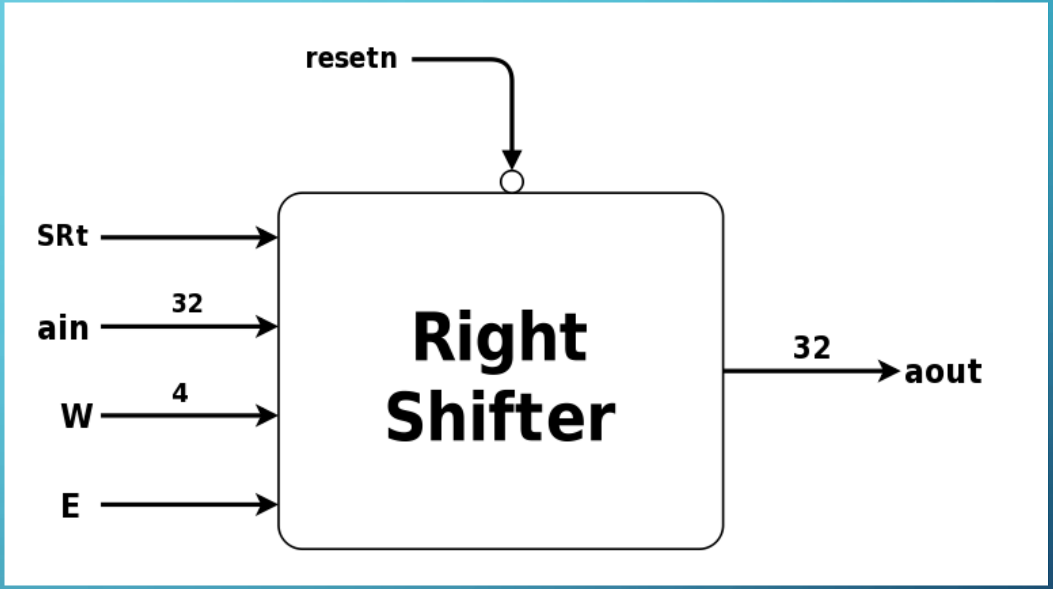
```vhdl
Transitions: process (resetn, clock, E_fsm)
begin
   if resetn = '0' then
      y <= S1;
   elsif (clock'event and clock = '1') then
      if E_fsm = '1' then
               case y is
                  when S1 => y <= S2;
                  when S2 => y <= S3;
                  when S3 => y <= S4;
                  when S4 => y <= S5;
                  when S5 => y <= S6;
                  when S6 => y <= S7;
                  when S7 => y <= S8;
                  when S8 => y <= S1;
                  when others => y <= S1;
               end case;
            end if;
   end if;
end process;

Outputs: process (y)
begin
   case y is
       when S1 => s <= "000";
       when S2 => s <= "001";
       when S3 => s <= "010";
       when S4 => s <= "011";
       when S5 => s <= "100";
       when S6 => s <= "101";
       when S7 => s <= "110";
       when S8 => s <= "111";
      when others => s <= "000";
   end case;
end process;
```

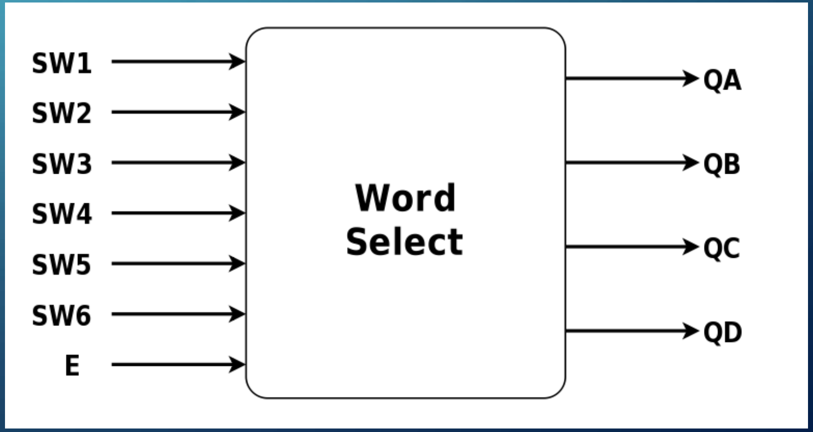


SHIFTER AND WORD SELECT

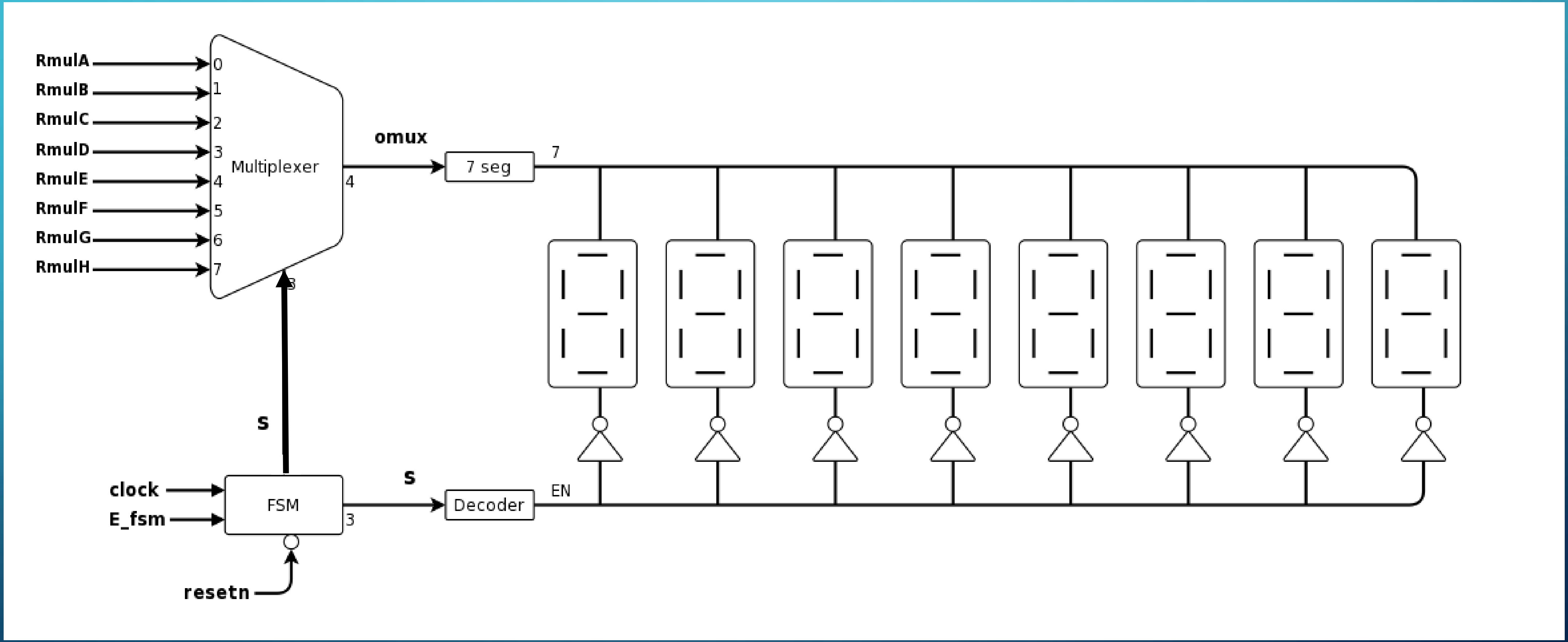


```
entity rightshifter is
generic (DIR: STRING:= "LEFT");
    Port ( ain : in  STD_LOGIC_VECTOR (31 downto 0);
           aout : out  STD_LOGIC_VECTOR (31 downto 0);
           w : in  STD_LOGIC_VECTOR (3 downto 0);
           E : in  STD_LOGIC;
           SR, resetn : in STD_LOGIC;
           clk : in  STD_LOGIC);
end rightshifter;

architecture Behavioral of rightshifter is

signal Qt : std_logic_vector (31 downto 0);

begin

--a0: assert (DIR = "LEFT" or DIR = "RIGHT")
  --  report "DIR can only be LEFT or RIGHT"
   -- severity error;

   process (E, sR, w, clk, Qt, resetn)
   begin
         if resetn = '0' then
         Qt <= (others => '1');
      elsif (clk'event and clk = '1') then
            if E = '1' then
               if SR = '1' then
                  Qt <= ain;
               else
                  if DIR = "RIGHT" then
                     Qt(31 downto 28) <= w;
                     for i in 27 downto 0 loop
                     Qt(i) <= Qt(i+4);
                     end loop;
                  end if;
               end if;
            end if;
         end if;
   end process;

aout <= Qt;

--gl: if DIR = "LEFT" generate
   --     shiftout <= Qt(5);
   --  end generate;

end Behavioral;
```

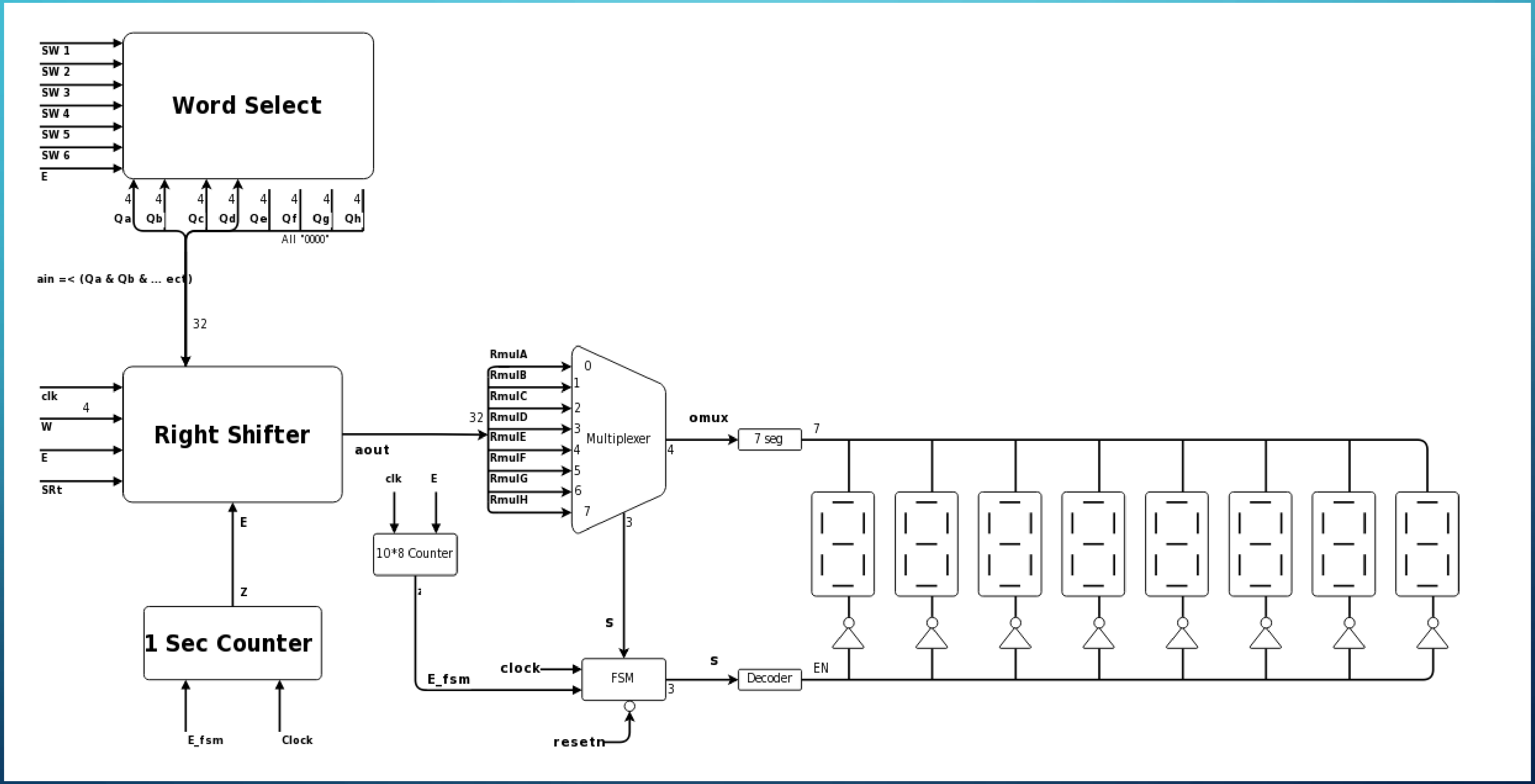




TOP LEVEL DISPLAY OUTPUT

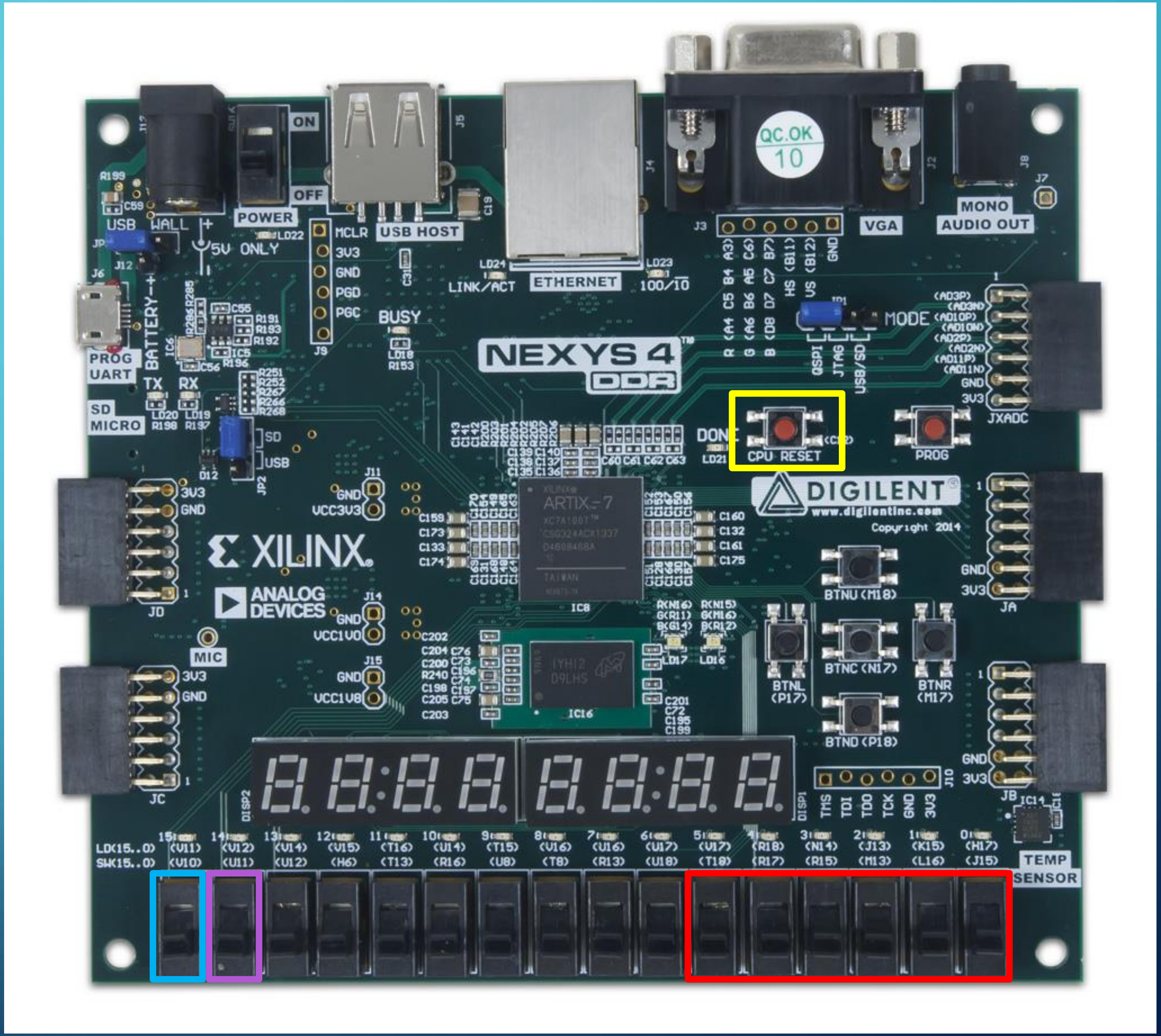


TOP LEVEL DIAGRAM



LAYOUT

- Reset (Yellow)
- Enable (Blue)
- Load (Purple)
- Words (Red)



PROBLEMS

- Displaying the words at the correct location.
- Actually getting the words to scroll across the 7-segment displays.
- Correctly scrolling the words off the screen and back onto the screen continuously.
- Incorrect finite state machine.

CONCLUSION

- Successful use of a large shifter.
- Continuous scrolling of output word.
- Improvements
  - Could be more intricate while utilizing a better display.
  - Could not use the full alphabet due to the drawbacks of the 7-segment display.