TRAFFIC LIGHT CONTROL SYSTEM

BY: Chinsung Kuo, Zhiwei Li, Mark Tarnicki, Jijian Chen

INTRODUCTION
HISTORY

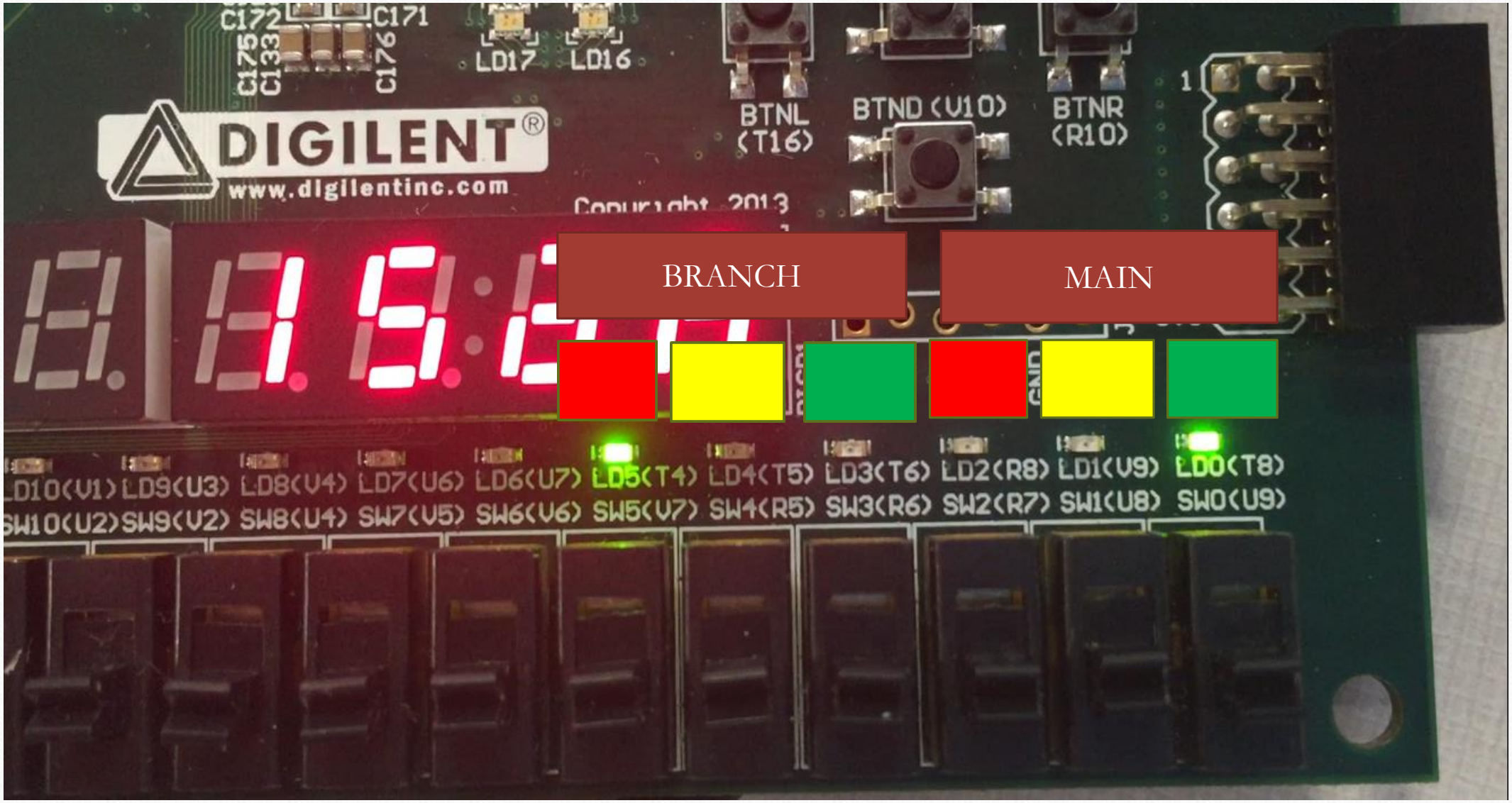
- Traffic light is a device that its main function is to control the flow of traffic by using mainly three colored lights which are green, yellow, and red.
- The first traffic controller had arms that extended out to show drivers what to do when approaching an intersection. This was designed in London, England in the 1860s.
- In December of 1920 in Detroit, the first four way three color traffic light was invented.

Benefits of a Traffic Light Control System

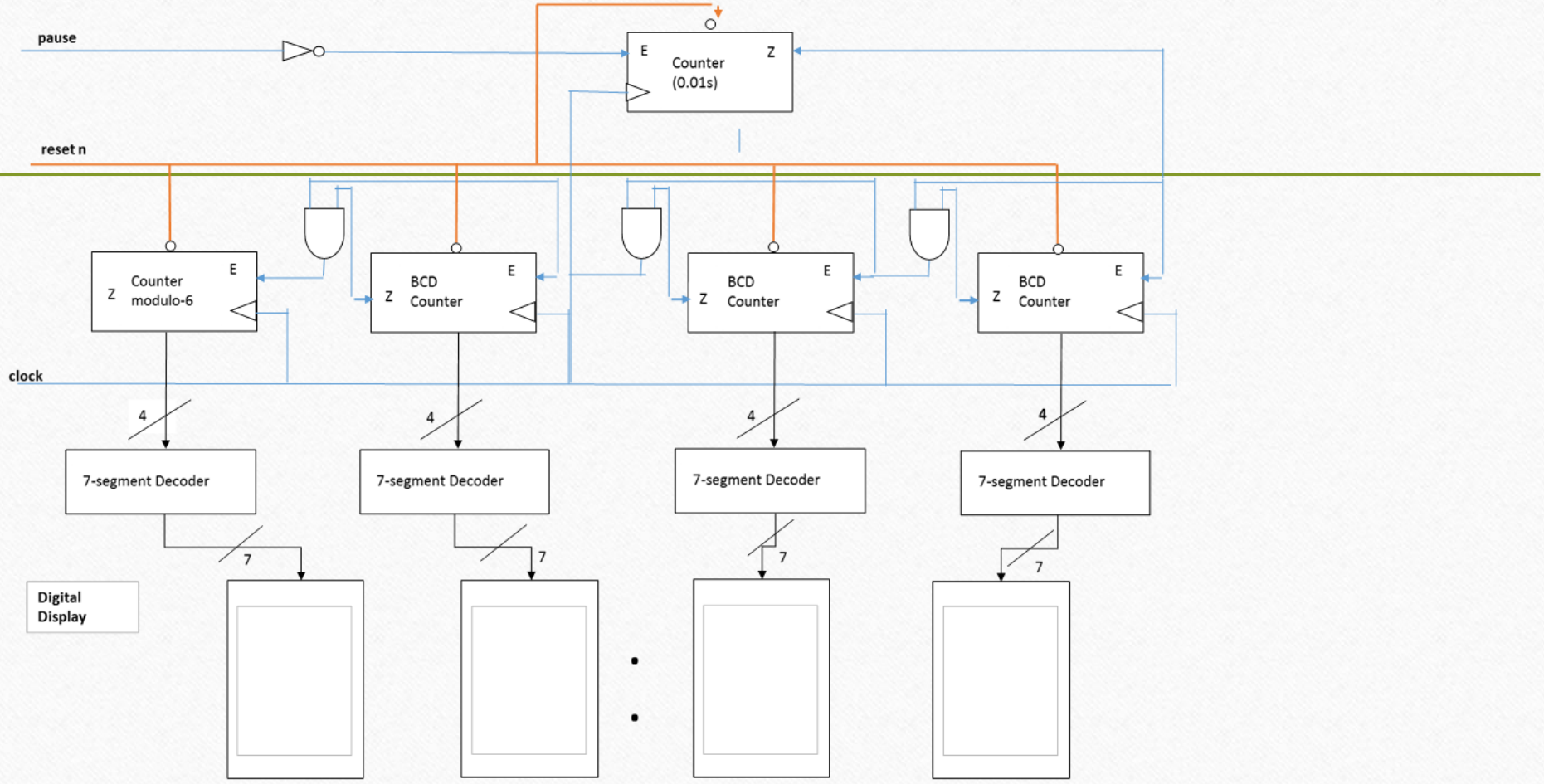
- Reduce Collisions
- Reduce the need to stop and start of traffic which in turn reduces fuel consumption, air pollution, and vehicle wear and tear.
- Increase the traffic capacity on the roads
- Improve time

SCOPE

- The following topics will be covered in our project that we covered and learned in the classroom.
  - Counter Cycle
  - Finite State Machine (FSM)
  - 7-Segment Decoder


BRANCH
MAIN
DIGILENT
www.digilentinc.com
BTNL (T16)
BTND (V10)
BTNR (R10)
LD10(V1) LD9(U3) LD8(V4) LD7(U6) LD6(U7) LD5(T4) LD4(T5) LD3(T6) LD2(R8) LD1(V9) LD0(T8)
SW10(U2) SW9(V2) SW8(U4) SW7(V5) SW6(V6) SW5(V7) SW4(R5) SW3(R6) SW2(R7) SW1(U8) SW0(U9)

counter


pause
reset n
clock
E
Counter
(0.01s)
Z
Counter
modulo-6
BCD
Counter
BCD
Counter
BCD
Counter
4
7-segment Decoder
7-segment Decoder
7-segment Decoder
7-segment Decoder
7
Digital
Display

Timing 40 seconds

```
begin
Q_3(3) <= '0';Q_3(2) <= '0';
    --40 seconds stopwatch
    -- Counter: 0.01s
    gz: my_genpulse generic map (COUNT => 10**6)
     port map (clock => clock, resetn => resetn, E => start, z => z);
    --    z <= '1'; -- only for simulation

    -- Counter: 10   0.1
    g0: my_genpulse generic map (COUNT => 10)
        port map (clock => clock, resetn => resetn, E => z, Q => Q_0, z => z_0);

    -- Counter: 10    1
    g1: my_genpulse generic map (COUNT => 10)
        port map (clock => clock, resetn => resetn, E => E_1, Q => Q_1, z => z_1);
        E_1 <= z and z_0;

     -- Counter: 10   10
    g2: my_genpulse generic map (COUNT => 10)
        port map (clock => clock, resetn => resetn, E => E_2, Q => Q_2, z => z_2);
        E_2 <= E_1 and z_1;

    -- Counter: 4
    g3: my_genpulse generic map (COUNT =>4)
        port map (clock => clock, resetn => resetn, E => E_3, Q => Q_3 (1 downto 0),z => z_3);
        E_3 <= E_2 and z_2;
        done <= z_3 and E_3;
```

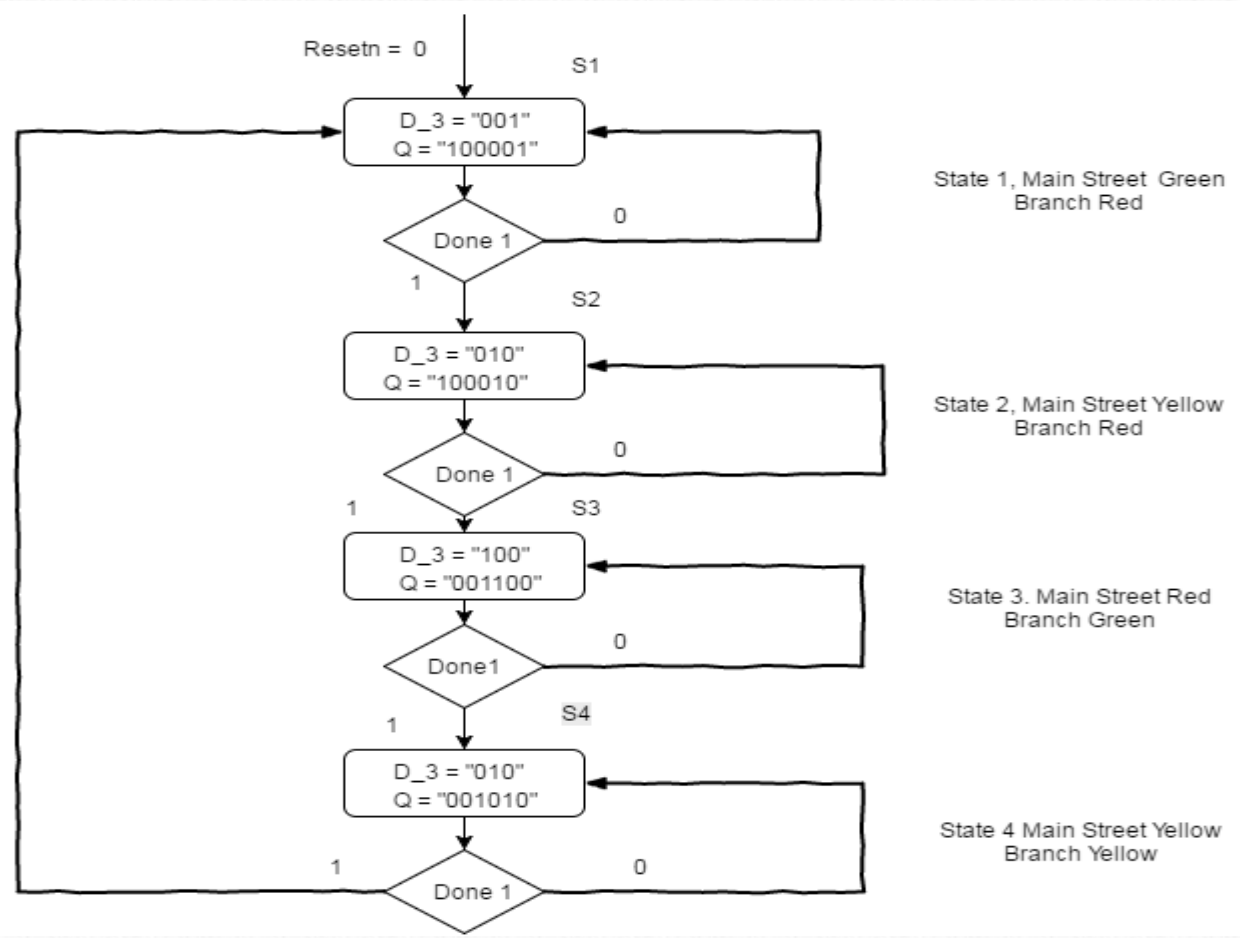
Timing 4 seconds

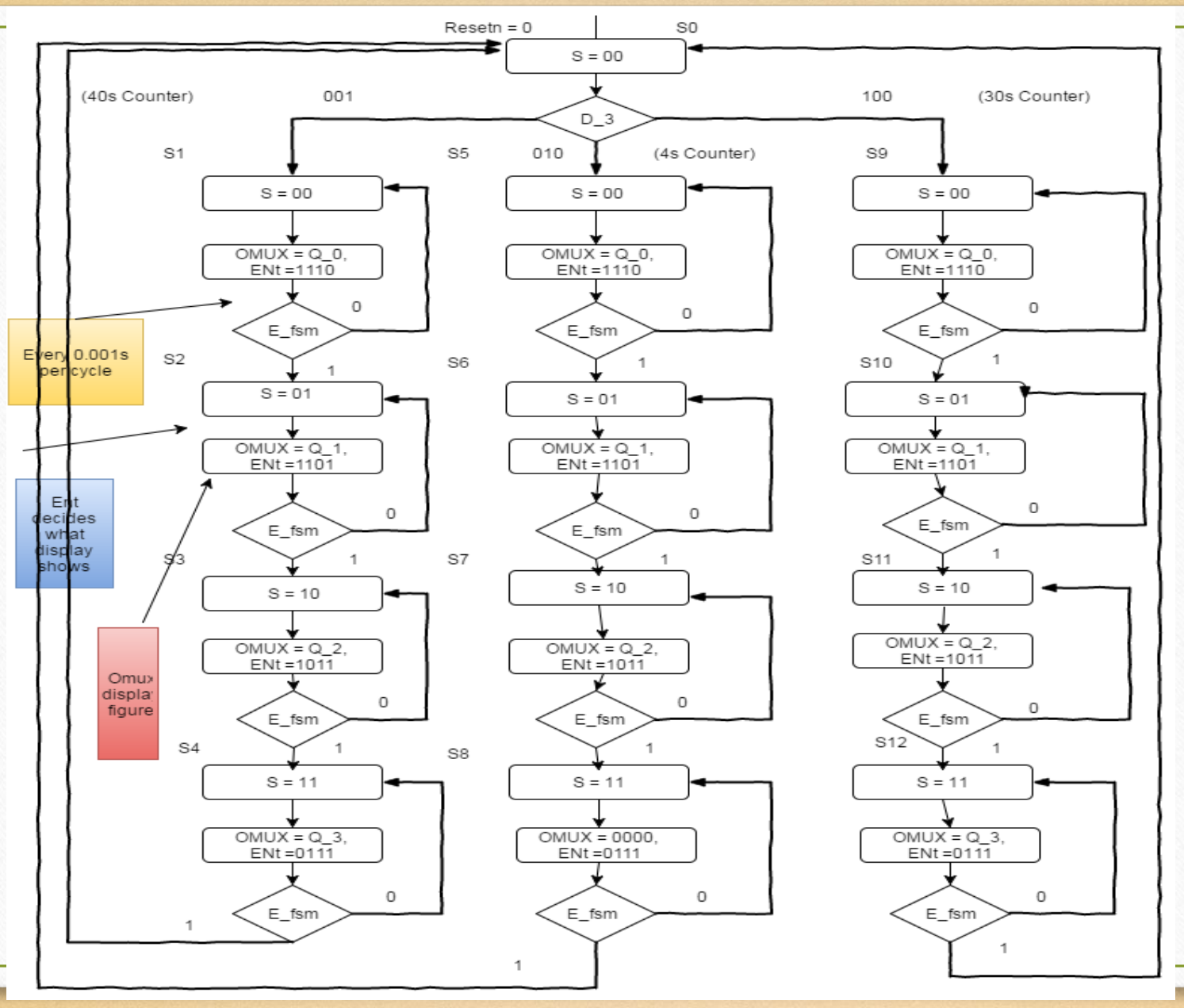
```
begin
Q_2(3) <= '0';Q_2(2) <= '0';
    --4 seconds stopwatch
    -- Counter: 0.01s
    gz: my_genpulse generic map (COUNT => 10**6)
     port map (clock => clock, resetn => resetn, E => start, z => z);
    --    z <= '1'; -- only for simulation

    -- Counter: 10    0.1
    g0: my_genpulse generic map (COUNT => 10)
        port map (clock => clock, resetn => resetn, E => z, Q => Q_0, z => z_0);

    -- Counter: 10    1
    g1: my_genpulse generic map (COUNT => 10)
        port map (clock => clock, resetn => resetn, E => E_1, Q => Q_1, z => z_1);
        E_1 <= z and z_0;

     -- Counter: 4    1*4
    g2: my_genpulse generic map (COUNT => 4)
        port map (clock => clock, resetn => resetn, E => E_2, Q => Q_2(1 downto 0), z => z_2);
        E_2 <= E_1 and z_1;
        done <= z_2 and E_2 ;
```

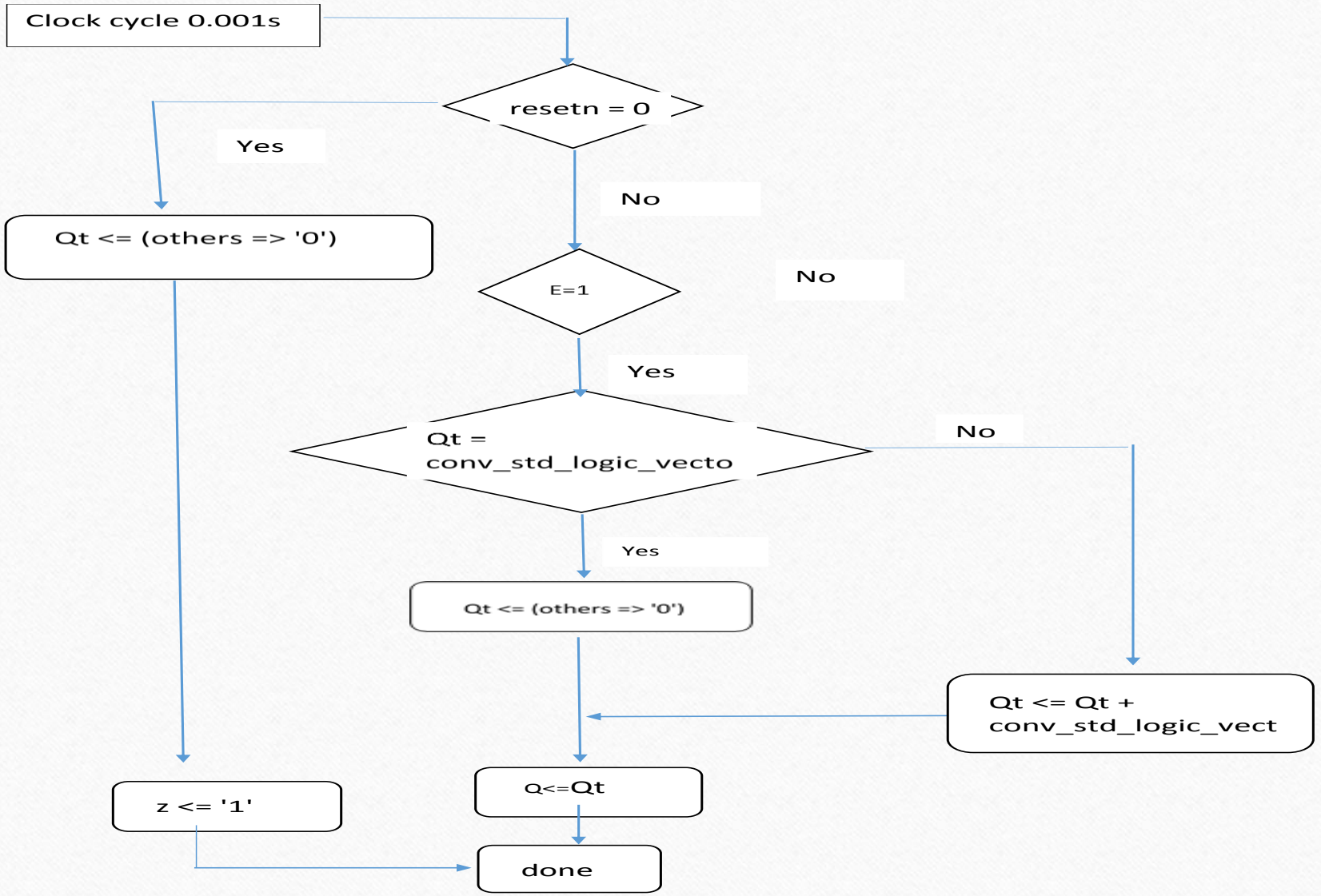
Timing 30 seconds

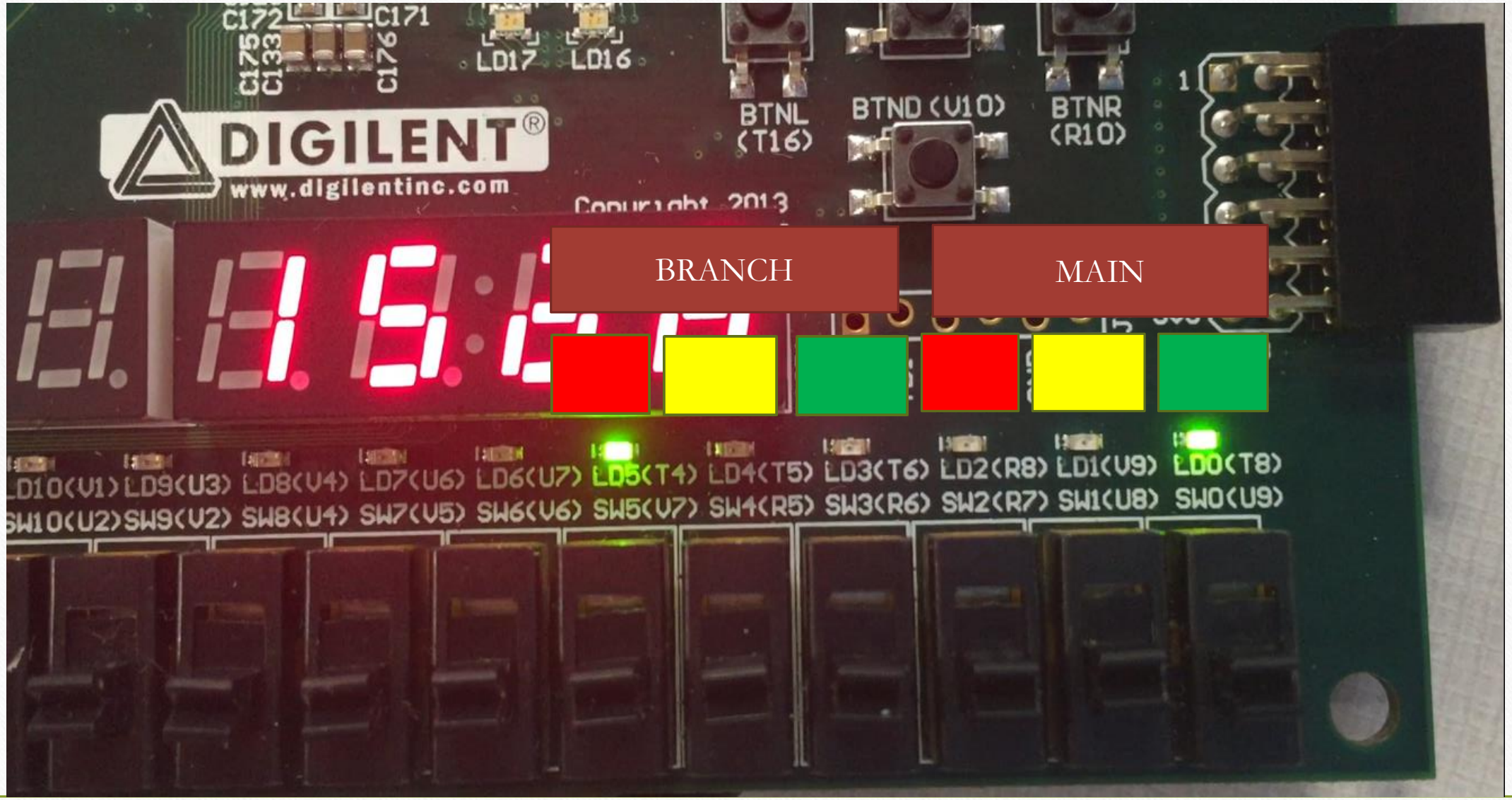
```
begin
Q_3(3) <= '0';Q_3(2) <= '0';--Q_3(1) <= '0';Q_3(0) <= q3;
    --30 seconds stopwatch
    -- Counter: 0.01s
    gz: my_genpulse generic map (COUNT => 10**6)
     port map (clock => clock, resetn => resetn, E => start, z => z);
    --    z <= '1'; -- only for simulation

    -- Counter: 10   0.1
    g0: my_genpulse generic map (COUNT => 10)
        port map (clock => clock, resetn => resetn, E => z, Q => Q_0, z => z_0);

    -- Counter: 10    1
    g1: my_genpulse generic map (COUNT => 10)
        port map (clock => clock, resetn => resetn, E => E_1, Q => Q_1, z => z_1);
        E_1 <= z and z_0;

     -- Counter: 10   10
    g2: my_genpulse generic map (COUNT => 10)
        port map (clock => clock, resetn => resetn, E => E_2, Q => Q_2, z => z_2);
        E_2 <= E_1 and z_1;

    -- Counter: 3
    g3: my_genpulse generic map (COUNT =>3)
        port map (clock => clock, resetn => resetn, E => E_3, Q => Q_3(1 downto 0),z => z_3);
        E_3 <= E_2 and z_2;
        done <= z_3 and E_3;
--      q_3(3 downto 2)<="00";q_3(1)<=q_31(0);q_3(0)<='0';
```

FSM


Resetn = 0
S1
D_3 = "001"
Q = "100001"
Done 1
0
1
S2
D_3 = "010"
Q = "100010"
Done 1
0
1
S3
D_3 = "100"
Q = "001100"
Done1
0
1
S4
D_3 = "010"
Q = "001010"
Done 1
1
0
State 1, Main Street Green
Branch Red
State 2, Main Street Yellow
Branch Red
State 3. Main Street Red
Branch Green
State 4 Main Street Yellow
Branch Yellow


Resetn = 0
S0
S = 00
(40s Counter)
001
D_3
100
(30s Counter)
S1
S5
010
(4s Counter)
S9
S = 00
OMUX = Q_0,
ENt =1110
E_fsm
0
1
S2
S6
S10
S = 01
OMUX = Q_1,
ENt =1101
S3
S7
S11
S = 10
OMUX = Q_2,
ENt =1011
S4
S8
S12
S = 11
OMUX = Q_3,
ENt =0111
OMUX = 0000,
ENt =0111
Every 0.001s
1 cycle
Ent
decides
what
display
shows
Omux
displa
figure


Clock cycle 0.001s
resetn = 0
Yes
No
Qt <= (others => '0')
E=1
No
Yes
Qt = conv_std_logic_vecto
No
Yes
Qt <= (others => '0')
Qt <= Qt + conv_std_logic_vect
Q<=Qt
z <= '1'
done


BRANCH
MAIN
DIGILENT
www.digilentinc.com
BTNL (T16)
BTND (V10)
BTNR (R10)
LD17
LD16
LD10(V1) LD9(U3) LD8(V4) LD7(U6) LD6(U7) LD5(T4) LD4(T5) LD3(T6) LD2(R8) LD1(V9) LD0(T8)
SW10(U2) SW9(V2) SW8(U4) SW7(V5) SW6(V6) SW5(V7) SW4(R5) SW3(R6) SW2(R7) SW1(U8) SW0(U9)

Conclusion

- This design uses the VHDL hardware language which we learned in class. Like FSM, Counter.
- The program's data can be set base on actual conditions with flexible modifications.
- The countdown time display allows drivers and pedestrians to pass safely.