

-- *Blinking LED System*

LIBRARY ieee;

USE ieee.std_logic_1164.ALL;

ENTITY blinkvhdl IS

PORT(

clk : IN STD_LOGIC;

q : OUT integer Range 0 to 1);

END blinkvhdl;

ARCHITECTURE a OF blinkvhdl IS

BEGIN

PROCESS (clk)

VARIABLE cnt : INTEGER RANGE 0 TO 67108863;

BEGIN

IF (clk'EVENT AND clk = '1') THEN

cnt := cnt + 1;

if (cnt < 25175000) then q <= 0; else q <= 1; END IF;

if (cnt > 50350000) then cnt := 0; end if;

END IF;

END PROCESS;

END a;

-- *Parking Garage System*

-- Two sensors count cars entering and leaving the parking garage.
-- INpb counts the cars entering. OUTpb counts the cars leaving.
-- Each sensor is debounced.
-- LotFull simulates a closing gate blocking cars from entering the lot once it is full.
-- The lot is full when the car count is 9.
-- clk is the UP-1 oscillator input. q is the car counter.

LIBRARY ieee;

USE ieee.std_logic_1164.ALL;

ENTITY park IS

PORT(

INpb, OUTpb, clk : IN STD_LOGIC;

q : OUT INTEGER RANGE 0 TO 15;

LotFull : OUT STD_LOGIC);

END park;

ARCHITECTURE a OF park IS

SIGNAL SHIFT4in : STD_LOGIC_VECTOR(3 DOWNTO 0);

SIGNAL SHIFT4out : STD_LOGIC_VECTOR(3 DOWNTO 0);

BEGIN

PROCESS (clk, INpb, OUTpb)

VARIABLE debcnt : INTEGER RANGE 0 TO 131071; -- approx. 5 millisecond interval for debounce

VARIABLE cnt : INTEGER RANGE 0 TO 15; -- car counter

Variable debup, debdown : INTEGER RANGE 0 TO 1; -- used as a flag bit.

BEGIN

IF (clk'EVENT AND clk = '1') THEN

debcnt := debcnt + 1;

-- Car entering garage (INpb is the car entering pushbutton)

IF (debcnt = 131071) THEN

SHIFT4in(3 DOWNTO 1) <= SHIFT4in(2 DOWNTO 0);

SHIFT4in(0) <= INpb;

END IF;

IF SHIFT4in(3 DOWNTO 0) = "1111" THEN

if (debup = 1) then cnt := cnt + 1; debup := 0; end if; -- car enters. Clear debup flag when PB is up or 1.

else

debup := 1; -- set debup flag while PB is being held down (0).

End if;

-- Car leaving garage (OUTpb is the car leaving pushbutton)

IF (debcnt = 131071) THEN

SHIFT4out(3 DOWNTO 1) <= SHIFT4out(2 DOWNTO 0);

SHIFT4out(0) <= OUTpb;

END IF;

IF SHIFT4out(3 DOWNTO 0) = "1111" THEN

if (debdown = 1) then cnt := cnt - 1; debdown := 0; end if; -- car exits. Clear debdown flag when PB is up or 1.

else

debdown := 1; -- set debdown flag while PB is being held down (0).

End if;

if (cnt = 10) then cnt := 9; END IF;

if (cnt = 15) then cnt := 0; END IF;

if (cnt = 9) then LotFull <= '1'; else Lotfull <= '0'; END IF;

END IF;

q <= cnt;

END PROCESS;

END a;

-- *Stopwatch System*

LIBRARY ieee;

USE ieee.std_logic_1164.ALL;

ENTITY stopw IS

PORT(

clk : IN STD_LOGIC;

start, stop, lap : IN STD_LOGIC;

q : OUT INTEGER RANGE 0 TO 63);

END stopw;

ARCHITECTURE a OF stopw IS

BEGIN

PROCESS (clk)

VARIABLE freqd : INTEGER RANGE 0 TO 33554431;

VARIABLE cnt : INTEGER RANGE 0 TO 63;

VARIABLE lapcnt : INTEGER RANGE 0 TO 63;

VARIABLE flag : integer Range 0 to 1;

BEGIN

IF (clk'EVENT AND clk = '1') THEN

freqd := freqd + 1;

IF (stop = '0') then flag := 0; end if;

IF (start = '0') then cnt := 0; freqd := 0; flag := 1; end if;

IF (freqd = 25175000) then

if (flag = 1) then cnt := cnt + 1; end if;

if (cnt = 60) then cnt := 0; end if;

freqd := 0;

end if;

if (lap = '1') then

q <= cnt; lapcnt := cnt;

else

q <= lapcnt;

end if;

end if;

END PROCESS;

END a;