A copy of our code is pasted below.  Comments are written in red, describing the function of parts of the code.

----------------------------------------------------------------------------------

-- Company:

-- Engineer:

--

-- Create Date: 11/30/2016 12:41:01 PM

-- Design Name:

-- Module Name: main - Behavioral

-- Project Name:

-- Target Devices:

-- Tool Versions:

-- Description:

--

-- Dependencies:

--

-- Revision:

-- Revision 0.01 - File Created

-- Additional Comments:

--

----------------------------------------------------------------------------------

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

-- Uncomment the following library declaration if using

-- arithmetic functions with Signed or Unsigned values

use IEEE.NUMERIC\_STD.ALL;

-- Uncomment the following library declaration if instantiating

-- any Xilinx leaf cells in this code.

--library UNISIM;

--use UNISIM.VComponents.all;

entity main is

   Port ( ClkIn : in STD\_LOGIC;

          switch : in STD\_LOGIC\_VECTOR (15 downto 0);

          stop : in STD\_LOGIC;

          freq : out STD\_LOGIC;

          lit : out STD\_LOGIC\_VECTOR (15 downto 0);

          gain : out STD\_LOGIC;

          shutdown : out STD\_LOGIC;

          an : out STD\_LOGIC\_VECTOR (3 downto 0);

          cat : out STD\_LOGIC\_VECTOR (6 downto 0));

end main;

architecture Behavioral of main is

   --initialize the variables

   signal count15, count14, count13, count12, count11, count10, count9, count8, count7, count6, count5, count4, count3, count2, count1, count0 : integer := 0;

   signal output : std\_logic := '1';

   signal counter : integer := 0;

begin

   gain <= '1';  --this sets the adapter to a 6 dB gain, a '0' will set it to a 12 dB gain

   shutdown <= '1'; --this will restrict current to the speaker if set to a '0'

   an <= "0111"; --allows only the first digit of the 7 segment display to light up

   process (clkin) begin

   if stop = '0' then --makes it so that nothing can occur when the stop button is pushed

       if rising\_edge(clkin) then --everything runs off of the rising edge of a 100 MHz clock (the system clock)

       case (switch) is

       when "1000000000000000" => --when SW15 is flipped up

           lit <= "1000000000000000"; --LED 15 will illuminate

           cat <= "1110010"; --the 7 seg display will display a C

               count15 <= count15 + 1; --counter 15 will increment

               if count15 = 191110 then --if the counter reaches 191110 (.0019 s)

                   output <= not output; --output signal will flip high to low or low to high

                   count15 <= 0; --counter 15 will reset

               end if;

       when "0100000000000000" => --same thing but new frequencies, LEDs, and digits displayed for switches 14 through 0

           lit <= "0100000000000000";

           cat <= "1110010";

                   count14 <= count14 + 1;

                   if count14 = 180388 then

                       output <= not output;

                       count14 <= 0;

                   end if;

       when "0010000000000000" =>

           lit <= "0010000000000000";

           cat <= "1000010";

                   count13 <= count13 + 1;

                   if count13 = 170265 then

                       output <= not output;

                       count13 <= 0;

                   end if;

       when "0001000000000000" =>

           lit <= "0001000000000000";

           cat <= "1000010";

               count12 <= count12 + 1;

               if count12 = 160705 then

                   output <= not output;

                   count12 <= 0;

               end if;

       when "0000100000000000" =>

           lit <= "0000100000000000";

           cat <= "0110000";

               count11 <= count11 + 1;

               if count11 = 151685 then

                   output <= not output;

                   count11 <= 0;

               end if;

       when "0000010000000000" =>

           lit <= "0000010000000000";

           cat <= "0111000";

               count10 <= count10 + 1;

               if count10 = 143172 then

                   output <= not output;

                   count10 <= 0;

               end if;

       when "0000001000000000" =>

           lit <= "0000001000000000";

           cat <= "0111000";

               count9 <= count9 + 1;

               if count9 = 135139 then

                   output <= not output;

                   count9 <= 0;

               end if;

       when "0000000100000000" =>

           lit <= "0000000100000000";

           cat <= "0000100";

               count8 <= count8 + 1;

               if count8 = 127551 then

                   output <= not output;

                   count8 <= 0;

               end if;

       when "0000000010000000" =>

           lit <= "0000000010000000";

           cat <= "0000100";

               count7 <= count7 + 1;

               if count7 = 120395 then

                   output <= not output;

                   count7 <= 0;

               end if;

       when "0000000001000000" =>

           lit <= "0000000001000000";

           cat <= "0001000";

               count6 <= count6 + 1;

               if count6 = 113636 then

                   output <= not output;

                   count6 <= 0;

               end if;

       when "0000000000100000" =>

           lit <= "0000000000100000";

           cat <= "0001000";

               count5 <= count5 + 1;

               if count5 = 107259 then

                   output <= not output;

                   count5 <= 0;

               end if;

       when "0000000000010000" =>

           lit <= "0000000000010000";

           cat <= "1100000";

               count4 <= count4 + 1;

               if count4 = 101239 then

                   output <= not output;

                   count4 <= 0;

               end if;

       when "0000000000001000" =>

           lit <= "0000000000001000";

           cat <= "1110010";

               count3 <= count3 + 1;

               if count3 = 95560 then

                   output <= not output;

                   count3 <= 0;

               end if;

       when "0000000000000100" =>

           lit <= "0000000000000100";

           cat <= "1110010";

               count2 <= count2 + 1;

               if count2 = 90192 then

                   output <= not output;

                   count2 <= 0;

               end if;

       when "0000000000000010" =>

           lit <= "0000000000000010";

           cat <= "1000010";

               count1 <= count1 + 1;

               if count1 = 85131 then

                   output <= not output;

                   count1 <= 0;

               end if;

       when "0000000000000001" =>

           lit <= "0000000000000001";

           cat <= "1000010";

               count0 <= count0 + 1;

               if count0 = 80354 then

                   output <= not output;

                   count0 <= 0;

               end if;

       when others =>

           cat <= "1111111";

           lit <= "0000000000000000";

       end case;

       end if; --for rising edge clock

   end if; --for stop input

   freq <= output; --freq is set to the output signal used above

   end process; --ends process

end Behavioral;

**Constraints**

set\_property PACKAGE\_PIN W5 [get\_ports ClkIn]

   set\_property IOSTANDARD LVCMOS33 [get\_ports ClkIn]

   create\_clock -add -name sys\_clk\_pin -period 10.00 -waveform {0 5} [get\_ports ClkIn]

set\_property PACKAGE\_PIN R2 [get\_ports {switch[15]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {switch[15]}]

set\_property PACKAGE\_PIN T1 [get\_ports {switch[14]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {switch[14]}]

set\_property PACKAGE\_PIN U1 [get\_ports {switch[13]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {switch[13]}]

set\_property PACKAGE\_PIN W2 [get\_ports {switch[12]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {switch[12]}]

set\_property PACKAGE\_PIN R3 [get\_ports {switch[11]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {switch[11]}]

set\_property PACKAGE\_PIN T2 [get\_ports {switch[10]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {switch[10]}]

set\_property PACKAGE\_PIN T3 [get\_ports {switch[9]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {switch[9]}]

set\_property PACKAGE\_PIN V2 [get\_ports {switch[8]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {switch[8]}]

set\_property PACKAGE\_PIN W13 [get\_ports {switch[7]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {switch[7]}]

set\_property PACKAGE\_PIN W14 [get\_ports {switch[6]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {switch[6]}]

set\_property PACKAGE\_PIN V15 [get\_ports {switch[5]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {switch[5]}]

set\_property PACKAGE\_PIN W15 [get\_ports {switch[4]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {switch[4]}]

set\_property PACKAGE\_PIN W17 [get\_ports {switch[3]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {switch[3]}]

set\_property PACKAGE\_PIN W16 [get\_ports {switch[2]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {switch[2]}]

set\_property PACKAGE\_PIN V16 [get\_ports {switch[1]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {switch[1]}]

set\_property PACKAGE\_PIN V17 [get\_ports {switch[0]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {switch[0]}]

set\_property PACKAGE\_PIN U18 [get\_ports {stop}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {stop}]

set\_property PACKAGE\_PIN L1 [get\_ports {lit[15]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {lit[15]}]

set\_property PACKAGE\_PIN P1 [get\_ports {lit[14]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {lit[14]}]

set\_property PACKAGE\_PIN N3 [get\_ports {lit[13]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {lit[13]}]

set\_property PACKAGE\_PIN P3 [get\_ports {lit[12]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {lit[12]}]

set\_property PACKAGE\_PIN U3 [get\_ports {lit[11]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {lit[11]}]

set\_property PACKAGE\_PIN W3 [get\_ports {lit[10]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {lit[10]}]

set\_property PACKAGE\_PIN V3 [get\_ports {lit[9]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {lit[9]}]

set\_property PACKAGE\_PIN V13 [get\_ports {lit[8]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {lit[8]}]

set\_property PACKAGE\_PIN V14 [get\_ports {lit[7]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {lit[7]}]

set\_property PACKAGE\_PIN U14 [get\_ports {lit[6]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {lit[6]}]

set\_property PACKAGE\_PIN U15 [get\_ports {lit[5]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {lit[5]}]

set\_property PACKAGE\_PIN W18 [get\_ports {lit[4]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {lit[4]}]

set\_property PACKAGE\_PIN V19 [get\_ports {lit[3]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {lit[3]}]

set\_property PACKAGE\_PIN U19 [get\_ports {lit[2]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {lit[2]}]

set\_property PACKAGE\_PIN E19 [get\_ports {lit[1]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {lit[1]}]

set\_property PACKAGE\_PIN U16 [get\_ports {lit[0]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {lit[0]}]

set\_property PACKAGE\_PIN A14 [get\_ports {freq}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {freq}]

set\_property PACKAGE\_PIN A16 [get\_ports {gain}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {gain}]

set\_property PACKAGE\_PIN B16 [get\_ports {shutdown}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {shutdown}]

set\_property PACKAGE\_PIN W4 [get\_ports {an[3]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {an[3]}]

set\_property PACKAGE\_PIN V4 [get\_ports {an[2]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {an[2]}]

set\_property PACKAGE\_PIN U4 [get\_ports {an[1]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {an[1]}]

set\_property PACKAGE\_PIN U2 [get\_ports {an[0]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {an[0]}]

set\_property PACKAGE\_PIN W7 [get\_ports {cat[6]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {cat[6]}]

set\_property PACKAGE\_PIN W6 [get\_ports {cat[5]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {cat[5]}]

set\_property PACKAGE\_PIN U8 [get\_ports {cat[4]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {cat[4]}]

set\_property PACKAGE\_PIN V8 [get\_ports {cat[3]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {cat[3]}]

set\_property PACKAGE\_PIN U5 [get\_ports {cat[2]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {cat[2]}]

set\_property PACKAGE\_PIN V5 [get\_ports {cat[1]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {cat[1]}]

set\_property PACKAGE\_PIN U7 [get\_ports {cat[0]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {cat[0]}]