Universidad Nacional Autonoma de Mexico Facultad de ingeniería

Diseño Digital VLSI

Grupo : 4

Práctica: 5

DISEÑO DEL CONTROL DE SENSORES ULTRASÓNICO

Edgar Daniel Barcenas Martinez

OBJETIVO:

El alumno aprenderá a diseñar mediante la utilización de atributos a señales (´HIGH) y tipos de variables (UNSIGNED) el control de un sensor ultrasónico (HC-SR04).

ESPECIFICACIONES:

Diseñar un circuito utilizando FPGA que se encargue de calcular la distancia de un obstáculo por medio de un sensor ultrasónico (HC-SR04), y observar los resultados de distancia por medio de 2 displays de 7 segmentos. La figura 7.1 muestra el diagrama a bloques del sistema.

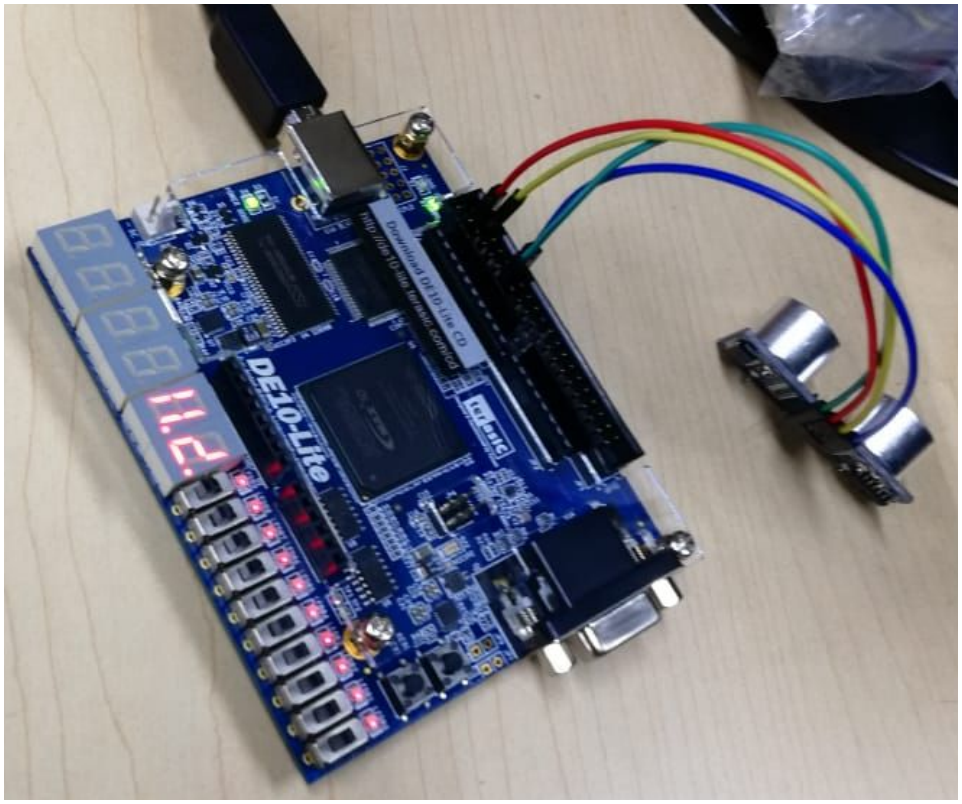
DESARROLLO:

```
 1   library IEEE;
 2   use IEEE.STD_LOGIC_1164.ALL;
 3   use IEEE.NUMERIC_STD.ALL;
 4   entity sonicos is
 5    Port (clk: in STD_LOGIC;
 6    sensor_disp: out STD_LOGIC;
 7    sensor_eco: in STD_LOGIC;
 8    anodos: out STD_LOGIC_VECTOR (3 downto 0);
 9    segmentos: out STD_LOGIC_VECTOR (7 downto 0));
10   end sonicos;
11   architecture Behavioral of sonicos is
12    signal cuenta: unsigned(16 downto 0) := (others => '0');
13    signal centimetros: unsigned(15 downto 0) := (others => '0');
14    signal centimetros_unid: unsigned(3 downto 0) := (others => '0');
15    signal centimetros_dece: unsigned(3 downto 0) := (others => '0');
16    signal sal_unid: unsigned(3 downto 0) := (others => '0');
17    signal sal_dece: unsigned(3 downto 0) := (others => '0');
18    signal digito: unsigned(3 downto 0) := (others => '0');
19    signal eco_pasado: std_logic := '0';
20    signal eco_sinc: std_logic := '0';
21    signal eco_nsinc: std_logic := '0';
22    signal espera: std_logic:= '0';
23    signal siete_seg_cuenta: unsigned(15 downto 0) := (others => '0');
24   begin
25    anodos(1 downto 0)<= "11";
26    siete_seg: process(clk)
```

```
27    begin
28    if rising_edge(clk) then
29    if siete_seg_cuenta(siete_seg_cuenta'high) = '1' then
30    digito <= sal_unid;
31    anodos(3 downto 2) <= "01";
32    else
33    digito <= sal_dece;
34    anodos(3 downto 2) <= "10";
35    end if;
36    siete_seg_cuenta <= siete_seg_cuenta +1;
37    end if;
38    end process;
39
40
41    Trigger:process(clk)
42    begin
43    if rising_edge(clk) then
44    if espera = '0' then
45    if cuenta = 500 then
46    sensor_disp <= '0';
47    espera <= '1';
48    cuenta <= (others => '0');
49    else
50    sensor_disp <= '1';
51    cuenta <= cuenta+1;
52    end if;
53
```

```
 52     end if;
 53
 54   elsif eco_pasado = '0' and eco_sinc = '1' then
 55     cuenta <= (others => '0');
 56     centimetros <= (others => '0');
 57     centimetros_unid <= (others => '0');
 58     centimetros_dece <= (others => '0');
 59   elsif eco_pasado = '1' and eco_sinc = '0' then
 60     sal_unid <= centimetros_unid;
 61     sal_dece <= centimetros_dece;
 62   elsif cuenta = 2900*2-1 then
 63   if centimetros_unid = 9 then
 64     centimetros_unid <= (others => '0');
 65     centimetros_dece <= centimetros_dece + 1;
 66   else
 67     centimetros_unid <= centimetros_unid + 1;
 68     end if;
 69     centimetros <= centimetros + 1;
 70     cuenta<= (others => '0');
 71   if centimetros = 3448 then
 72     espera <= '0';
 73     end if;
 74   else
 75     cuenta <= cuenta + 1;
 76     end if;
 77     eco_pasado<= eco_sinc;
 78     eco_sinc <= eco_nsinc;
 79     eco_nsinc <= sensor_eco;
 80     end if;
 81     end process;
 82
```

```
 75     cuenta <= cuenta + 1;
 76     end if;
 77     eco_pasado<= eco_sinc;
 78     eco_sinc <= eco_nsinc;
 79     eco_nsinc <= sensor_eco;
 80     end if;
 81     end process;
 82
 83   Decodificador: process (digito)
 84     begin
 85   if digito=X"0" then segmentos <= X"81";
 86   elsif digito=X"1" then segmentos <= X"F3";
 87   elsif digito=X"2" then segmentos <= X"49";
 88   elsif digito=X"3" then segmentos <= X"61";
 89   elsif digito=X"4" then segmentos <= X"33";
 90   elsif digito=X"5" then segmentos <= X"25";
 91   elsif digito=X"6" then segmentos <= X"05";
 92   elsif digito=X"7" then segmentos <= X"F1";
 93   elsif digito=X"8" then segmentos <= X"01";
 94   elsif digito=X"9" then segmentos <= X"21";
 95   elsif digito=X"a" then segmentos <= X"11";
 96   elsif digito=X"b" then segmentos <= X"07";
 97   elsif digito=X"c" then segmentos <= X"8D";
 98   elsif digito=X"d" then segmentos <= X"43";
 99   elsif digito=X"e" then segmentos <= X"0D";
100   else
101     segmentos<= X"1D";
102     end if;
103     end process;
104   end Behavioral;
```



CONCLUSIONES: