ELE519 - Gömülü Sistemler

Ödev 2

TOBB ETÜ

Ekonomi ve Teknoloji Üniversitesi

Burak Can KARAKURT

161201049

Soru1)

a)

```
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 FA_and is
   Port ( a : in STD_LOGIC;
          b : in STD_LOGIC;
          sum : out STD_LOGIC;
          cin : in STD_LOGIC;
          cout : out STD_LOGIC);
end FA_and;

architecture Behavioral of FA_and is
signal sig1,sig2,sig3,sig4,sig5,sig6,sig7: std_logic;
```

begin

sig1<=(not a) and (not b) and (not cin);

sig2<=(not a) and b and (not cin);

sig3<=a and b and (not cin);

sig4<=a and b and cin;

sig5<=b and cin;

sig6<=a and cin;

sig7<= a and b;

sum<=sig1 or sig2 or sig3 or sig4;

cout<= sig5 or sig6 or sig7;

end Behavioral;

b) GAL16V8 üzerinde gerçeklemenin nasıl yapılacağını anlamadım. Dolayısı ile diğer şıkta sigorta yapısını da çizemedim.

Soru 2)

a) A6=1.

b)

A5	A4	A3	A2	A1	Out(sum)
0	0	0	0	0	0
0	0	0	0	1	1
0	0	0	1	0	1
0	0	0	1	1	0
0	0	1	0	0	1
0	0	1	0	1	0
0	0	1	1	0	0
0	0	1	1	1	1

A5	A4	A3	A2	A1	Out(carry)
0	0	0	0	0	0
0	0	0	0	1	0
0	0	0	1	0	0
0	0	0	1	1	1

0	**0**	**1**	**0**	**0**	**0**
0	**0**	**1**	**0**	**1**	**1**
0	**0**	**1**	**1**	**0**	**1**
0	**0**	**1**	**1**	**1**	**1**

Soru 3)

a)

HA.vhd

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity HA is
    Port ( a : in STD_LOGIC;
           b : in STD_LOGIC;
           sum : out STD_LOGIC;
           cout : out STD_LOGIC);
end HA;

architecture Behavioral of HA is

begin

sum <= a xor b;
cout <= a and b;

end Behavioral;
```

FA.vhd

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity FA is
    Port ( ain : in STD_LOGIC;
           bin : in STD_LOGIC;
           Cin : in STD_LOGIC;
           Sout : out STD_LOGIC;
           c : out STD_LOGIC);
end FA;

architecture Behavioral of FA is
    signal s1: STD_LOGIC;
    signal c1: STD_LOGIC;
    signal c2: STD_LOGIC;
    component HA is
        Port ( a : in STD_LOGIC;
               b : in STD_LOGIC;
               sum : out STD_LOGIC;
               cout : out STD_LOGIC);
    end component;
begin
HA1:HA
```

```
   port map(a=>ain,
            b=>bin,
            sum=>s1,
            cout=>c1);

HA2:HA
   port map(a=>s1,
        b=>Cin,
        sum=>Sout,
        cout=>c2);
c<= c1 or c2;
end Behavioral;
```

3.b)

ADDER_128.vhd

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity ADDER_128 is
   Port ( A_IN : in STD_LOGIC_VECTOR (127 downto 0);
          B_IN : in STD_LOGIC_VECTOR (127 downto 0);
          SUM_OUT : out STD_LOGIC_VECTOR (127 downto 0);
          CARRY_OUT : out STD_LOGIC);
end ADDER_128;

architecture Behavioral of ADDER_128 is
component FA is
   Port ( ain : in STD_LOGIC;
     bin : in STD_LOGIC;
```

```
      Cin : in STD_LOGIC;
      Sout : out STD_LOGIC;
      c : out STD_LOGIC);
   end component;
Signal Co: std_logic_vector(127 downto 0);
begin
F_A0: FA
   port map(ain=>A_IN(0),
            bin=>B_IN(0),
            Cin=>'0',
            Sout=>SUM_OUT(0),
            c=>Co(0));

G1:for i in 1 to 127 generate
F_A: FA
   port map(ain=>A_IN(i),
            bin=>B_IN(i),
            Cin=>Co(i-1),
            Sout=>SUM_OUT(i),
            c=>Co(i));
   end generate
CARRY_OUT <= Co(127);

end Behavioral;
```

3.c)

ADDER_CARRY_LOOKAHEAD.vhd

```
entity FA_WS is
   Port ( a : in STD_LOGIC;
          b : in STD_LOGIC;
          c_in : in STD_LOGIC;
          s : out STD_LOGIC;
          c_out : out STD_LOGIC);
end FA_WS;

architecture Behavioral of FA_WS is
   Signal axorb : STD_LOGIC;
begin
   axorb <= a xor b;
   s <= axorb xor c_in;
   with axorb select
      c_out <= c_in when '1',
               a when others;

end Behavioral;



entity ADDER_CARRY_LOOKAHEAD_WS is
   Port ( A_IN : in STD_LOGIC_VECTOR (63 downto 0);
          B_IN : in STD_LOGIC_VECTOR (63 downto 0);
          SUM_OUT : out STD_LOGIC_VECTOR (63 downto 0);
          CARRY_OUT : out STD_LOGIC);
end ADDER_CARRY_LOOKAHEAD_WS;

architecture Behavioral of ADDER_CARRY_LOOKAHEAD_WS is
   component  FA_WS is
     Port ( a : in STD_LOGIC;
            b : in STD_LOGIC;
            c_in : in STD_LOGIC;
            s : out STD_LOGIC;
            c_out : out STD_LOGIC);
   end component;
```

```
   Signal C : STD_LOGIC_VECTOR (64 downto 0); -- CARRY
begin

   C(0) <= '0';

   GEN1 : for i in 0 to 63 generate begin
      U : FA_WS port map(
         a=>A_IN(i),
         b=>B_IN(i),
         c_in=>C(i),
         s=>SUM_OUT(i),
         c_out=>C(i+1));
   end generate GEN1;

   CARRY_OUT <= C(64);

end Behavioral;
```

3.d)

ADDER_CARRY_PROPAGATE_MACRO.vhd

```
entity ADDER_CARRY_PROPAGATE_MACRO is
   Port ( A_IN : in STD_LOGIC_VECTOR (127 downto 0);
          B_IN : in STD_LOGIC_VECTOR (127 downto 0);
          SUM_OUT : out STD_LOGIC_VECTOR (127 downto 0);
          CARRY_OUT : out STD_LOGIC);
end ADDER_CARRY_PROPAGATE_MACRO;

architecture Behavioral of ADDER_CARRY_ PROPAGATE _MACRO is
   Signal CARRY : STD_LOGIC_VECTOR (127 downto 0);
   Signal aXorB : STD_LOGIC_VECTOR (127 downto 0);
begin

   G : for i in 0 to 127 generate begin
      AXORB(i) <= A_IN(i) xor B_IN(i);
   end generate G;

   U0 : CARRY4 port map (
      CO => CARRY(3 downto 0),
      O => SUM_OUT(3 downto 0),
      CI => '0',
      CYINIT => '0',
      DI => B_IN(3 downto 0),
```

```
      S => aXorB(3 downto 0)
   );

   G2 : for i in 1 to 15 generate begin
      U : CARRY4 port map (
         CO => CARRY((4*(i+1)-1) downto (4*i)), -- 4-bit carry out
         O => SUM_OUT((4*(i+1)-1) downto (4*i)), -- 4-bit carry chain XOR data out
         CI => CARRY(4*i-1), -- 1-bit carry cascade input
         CYINIT => '0', -- 1-bit carry initialization
         DI => B_IN((4*(i+1)-1) downto (4*i)), -- 4-bit carry-MUX data in
         S => aXorB((4*(i+1)-1) downto (4*i)) -- 4-bit carry-MUX select input
      );
   end generate G2;

   CARRY_OUT <= CARRY(127);

end Behavioral;
```

3.e)

FA_ele519hw23e.vhd

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity FA_ ele519hw23e is
   Port ( A_IN : in STD_LOGIC_VECTOR (15 downto 0);
      B_IN : in STD_LOGIC_VECTOR (15 downto 0);
```

```
      CARRY_IN : in STD_LOGIC;
      SUM_OUT : out STD_LOGIC_VECTOR (15 downto 0);
      CARRY_OUT : out STD_LOGIC);
end FA_ele519hw23e;

architecture Behavioral of FA_ele519hw23e is
Signal A : integer;
Signal B : integer;
Signal CARRY : integer;
Signal SUM : integer;
Signal TEMP_SUM : STD_LOGIC_VECTOR (32 downto 0);
begin
A <= CONV_INTEGER(A_IN);
B <= CONV_INTEGER(B_IN);
CARRY <= CONV_INTEGER(CARRY_IN);
SUM <= A + B + CARRY;
TEMP_SUM <= CONV_STD_LOGIC_VECTOR(SUM, 17);
SUM_OUT <= TEMP_SUM(15 downto 0);
CARRY_OUT <= TEMP_SUM(16);

end Behavioral;
---------------------------------------- ----------------------------

entity xFA_ele519hw23e is
   Port ( A_IN : in STD_LOGIC_VECTOR (127 downto 0);
          B_IN : in STD_LOGIC_VECTOR (127 downto 0);
          SUM_OUT : out STD_LOGIC_VECTOR (127 downto 0);
          CARRY_OUT : out STD_LOGIC);
end xFA_ele519hw23e;

architecture Behavioral of xFA_ele519hw23e is
   component FA_ele519hw23e is
      Port ( A_IN : in STD_LOGIC_VECTOR (15 downto 0);
             B_IN : in STD_LOGIC_VECTOR (15 downto 0);
             CARRY_IN : in STD_LOGIC;
```

```
            SUM_OUT : out STD_LOGIC_VECTOR (15 downto 0);
            CARRY_OUT : out STD_LOGIC);
    end component;
    Signal TEMP_CARRY : STD_LOGIC_VECTOR (7 downto 0);
begin
    FA0 : FA_ele519hw23e port map(
        A_IN=>A_IN(15 downto 0),
        B_IN=>B_IN(15 downto 0),
        CARRY_IN=>'0',
        SUM_OUT=>SUM_OUT(15 downto 0),
        CARRY_OUT=>TEMP_CARRY(0));
--
    FA1 : FA_ele519hw23e port map(
        A_IN=>A_IN(31 downto 16),
        B_IN=>B_IN(31 downto 16),
        CARRY_IN=>TEMP_CARRY(0),
        SUM_OUT=>SUM_OUT(31 downto 16),
        CARRY_OUT=>TEMP_CARRY(1));
--
    FA2 : FA_ele519hw23e port map(
        A_IN=>A_IN(47 downto 32),
        B_IN=>B_IN(47 downto 32),
        CARRY_IN=>TEMP_CARRY(1),
        SUM_OUT=>SUM_OUT(47 downto 32),
        CARRY_OUT=>TEMP_CARRY(2));
--
    FA3 : FA_ele519hw23e port map(
        A_IN=>A_IN(63 downto 48),
        B_IN=>B_IN(63 downto 48),
        CARRY_IN=>TEMP_CARRY(2),
        SUM_OUT=>SUM_OUT(63 downto 48),
        CARRY_OUT=>TEMP_CARRY(3));
--
     FA4 : FA_ele519hw23e port map(
        A_IN=>A_IN(79 downto 64),
        B_IN=>B_IN(79 downto 64),
        CARRY_IN=>TEMP_CARRY(4),
        SUM_OUT=>SUM_OUT(79 downto 64),
        CARRY_OUT=>TEMP_CARRY(4));
--
     FA5 : FA_ele519hw23e port map(
        A_IN=>A_IN(95 downto 80),
        B_IN=>B_IN(95 downto 80),
        CARRY_IN=>TEMP_CARRY(5),
        SUM_OUT=>SUM_OUT(96 downto 80),
        CARRY_OUT=>TEMP_CARRY(45);
--
```

```
      FA6 : FA_ele519hw23e port map(
       A_IN=>A_IN(111 downto 96),
       B_IN=>B_IN(111 downto 96),
       CARRY_IN=>TEMP_CARRY(6),
       SUM_OUT=>SUM_OUT(111 downto 96),
       CARRY_OUT=>TEMP_CARRY(6));
--
      FA7 : FA_ele519hw23e port map(
       A_IN=>A_IN(127 downto 112),
       B_IN=>B_IN(127 downto 112),
       CARRY_IN=>TEMP_CARRY(7),
       SUM_OUT=>SUM_OUT(127 downto 112),
       CARRY_OUT=>TEMP_CARRY(7));



   CARRY_OUT <= TEMP_CARRY(7);
end Behavioral;
```

f)

Adder_128	25.3968
ADDER_CARRY_PROPAGATE_MACRO	15.6182
xFA_ele519hw23e	14.638

Hızlar yukarıdaki tabloda verilmiştir. Adder 128'de carry nedeniyle bekleme süreleri artmakta dır. Diğer devreler ise bu iş için daha optimize edilmiş devrelerdir. Ayrıca çip üzerinde birbirl erine daha yakın oldukları ve haberleşmeleri için geçen süre daha kısa olduğu için hız artmakt adır.