



RECEP TAYYİP
ERDOĞAN
ÜNİVERSİTESİ

Elektrik-Elektronik Mühendisliği
Sayısal Lojik

Deney Raporu-4

Yakup Demiryürek
180711049

(Bahar 2023)

Amaç

FPGA kartı üzerinde 7-parçalı gösterge kurulumu amaçlanmıştır.

Ekipmanlar

- Xilinx yüklü bilgisayar
- FPGA kartı Deney Çalışması

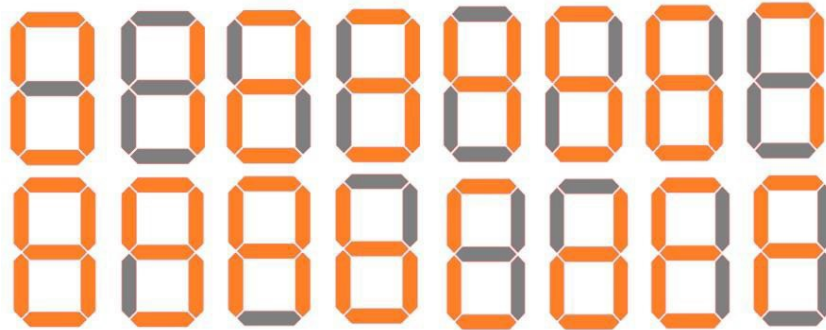
DÇ1

Girilen 4 bit sayının 10'luk tabandaki 7-parçalı gösterge üzerindeki ledlerin yanarak oluşturacakları şekiller **Şekil 1**'deki gibidir. 4-bitlik A_3, A_2, A_1, A_0 sayısı için uygun LED'ler yanacak şekilde $S_0, S_1, S_2, S_3, S_4, S_5, S_6$ çıkışlarının doğruluk tablosu **Tablo 1**'de gösterilmiştir.

Tablo 1. Doğruluk Tablosu

A_3	A_2	A_1	A_0	S_0	S_1	S_2	S_3	S_4	S_5	S_6	
0	0	0	0	1	1	1	1	1	1	0	0
0	0	0	1	0	1	1	0	0	0	0	1
0	0	1	0	1	1	0	1	1	0	1	2
0	0	1	1	1	1	1	1	0	0	1	3
0	1	0	0	0	1	1	0	0	1	1	4
0	1	0	1	1	0	1	1	0	1	1	5
0	1	1	0	1	0	1	1	1	1	1	6
0	1	1	1	1	1	1	0	0	0	0	7
1	0	0	0	1	1	1	1	1	1	1	8
1	0	0	1	1	1	1	1	0	1	1	9
1	0	1	0	1	1	1	0	1	1	1	A
1	0	1	1	0	0	1	1	1	1	1	b
1	1	0	0	1	0	0	1	1	1	0	C
1	1	0	1	0	1	1	1	1	0	1	d
1	1	1	0	1	0	0	1	1	1	1	E
1	1	1	1	1	0	0	0	1	1	1	F

DÇ2



Şekil 1. 7-parçalı gösterge ile sayıların ve harflerin gösterimleri

Karnough haritaları kullanılarak $S_0, S_1, S_2, S_3, S_4, S_5, S_6$ çıkışlarının en sade formülleri **Tablo 2**'de verilmiştir.

Tablo 2. Karnough haritaları

S ₀						S ₁						S ₂						S ₃					
		00	01	11	10			00	01	11	10			00	01	11	10			00	01	11	10
	00	1	0	1	1		00	1	1	1	1		00	1	1	1	0		00	1	0	1	1
	01	0	1	1	1		01	1	0	1	0		01	1	1	1	1		01	0	1	0	1
	11	0	1	0	0		11	0	1	0	0		11	0	1	0	0		11	1	1	0	1
	10	1	1	0	1		10	1	1	0	1		10	1	1	1	1		10	1	1	1	0
S ₄						S ₅						S ₆						S ₇					
		00	01	11	10			00	01	11	10			00	01	11	10			00	01	11	10
	00	1	0	0	1		00	1	0	0	0		00	0	0	1	1		00	0	0	0	0
	01	0	0	0	1		01	1	1	0	1		01	1	1	0	1		01	0	0	0	0
	11	1	1	1	1		11	1	0	1	1		11	0	1	1	1		11	0	0	0	0
	10	1	0	1	1		10	1	1	1	1		10	1	1	1	1		10	0	0	0	0

Bulunan formüller şöyledir;

$$S_0 = (A.D.(C \oplus B) + A'.C'.(B \oplus D))'$$

$$S_1 = B.D.(A \oplus C) + A'.C'.D' + B'.(C.D' + C'.A + D.A')$$

$$S_2 = D.(A.C)' + A'.C' + (A \oplus B)$$

$$S_3 = A.C' + B.(D \oplus C) + A'.B'.(C + D') + D.A.B'$$

$$S_4 = A.(B + C) + D'.(C + B'.C')$$

$$S_5 = D'.(B + C') + A.(B' + C) + A'.B.C'$$

$$S_6 = C.(D' + B') + A.(B' + D) + B.A'.C'$$

DÇ3

DÇ2'de bulunan formüller kullanılarak 4-bit girişli 8-bit çıkışlı vhdl kodu yazılmıştır. FPGA kartında '0' durumunda '1', '1' durumunda '0' olmaktadır bu yüzden **DÇ1'**de bulunan formüller çeviriciden geçirilmiştir. S7'ninher zaman yanması sağlanmıştır. Giriş ve çıkışlar vektörel olarak tanımlanmıştır.

```
Port ( A : in STD_LOGIC_VECTOR (3 downto 0);
```

```
      S : out STD_LOGIC_VECTOR (7 downto 0));
```

```
S(0)<= not(((A(3)and A(0))and(A(1)xor A(2)))or(((not A(3))and(not A(1)))and(A(2)xor A(0)))));
```

```
S(1)<= not((((not A(2))and((A(1)and(not A(0)))or((not A(1))and A(3))or(A(0)and(not A(3)))))or((A(2)and A(0))and(A(3)xor A(1)))or((not A(0)) and (not A(3)) and (not A(1)))));
```

```
S(2)<= not(((A(3)xor A(2))or((not A(1))and A(0))or((not A(3))and(not A(1)))or((not A(3))and A(0)))));
```

```
S(3)<= not((((not A(1))and A(3))or((A(0)xor A(1))and A(2))or((not A(3))and(not A(2))and (A(1)or(not A(0))))or(A(0)and A(3)and(not A(2)))));
```

```
S(4)<= not(((A(3)and(A(2)or A(1)))or((not A(0))and(A(1)or((not A(2))and(not A(1)))))));
```

```
S(5)<= not((((not A(0))and(A(2)or(not A(1))))or(A(3)and(((not A(2))or A(1)))or((not A(3))and(not A(1))and A(2)))));
```

```
S(6)<= not(((A(1) and ((not A(0)) or (not A(2)))) or (A(3) and (A(0) or (not A(2)))) or (A(2) and (not A(3)) and (not A(1)))));
```

```
S(7)<=not('1');
```

DÇ4

DÇ3’de yazılan vhdl kodun simülasyonu hazırlanıp, **Şekil 2**’de verildiği gibi test edilmiştir.

```
constant period : time := 50 ns;
```

```
    A3process: PROCESS
```

```
    BEGIN
```

```
    A(3) <= '0';
```

```
    wait for period*8;
```

```
    A(3)<= '1';
```

```
    wait for period*8;
```

```
    end process;
```

```
    A2process: PROCESS
```

```
    BEGIN
```

```
    A(2) <= '0';
```

```
    wait for period*4;
```

```
    A(2) <= '1';
```

```
    wait for period*4;
```

```
    end process;
```

```
    A1process: PROCESS
```

```
    BEGIN
```

```
    A(1) <= '0';
```

```
    wait for period*2;
```

```
    A(1) <= '1';
```

```
    wait for period*2;
```

```
    end process;
```

```
    A0process: PROCESS
```

```
    BEGIN
```

```
    A(0) <= '0';
```

```
    wait for period;
```

```
    A(0) <= '1';
```

```
    wait for period;
```

```
end process;
```



Şekil 2.Simülasyon

Sonuç

S1-S2

DÇ1 ve DÇ2’de elde edilmiştir.

S3

DÇ1’de verilen ve DÇ4’de elde edilen değerler karşılaştırıldığında testimizin doğruluğu kanıtlanmıştır.