

Ad-Soyad :
No :

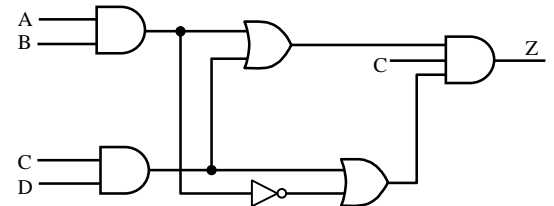
Email :
İmza :

(Final-18.01.2010)
0112611 – Lojik Devreler

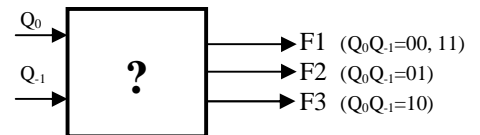
S1. Bozuk bir hard diskten kurtarılan IEEE-754 floating point formatındaki dosyadan okunan verilerden biri (C1960000)₁₆ olarak çözümlenmiştir. Bu kurtarılan sayının ne olduğunu bulunuz. $N = (-1)^s * 1.M * 2^E (E = 127)$ (10)

S2. Yandaki devre için;

- Z ifadesini yazınız. (5)
- Literal (L), Gate (G), ve Gate with Not (GN) cost değerlerini bulunuz. (5)
- Z yi çarpımların toplamı (mintermler) olarak ifade ediniz. (5)
- Z ifadesini sadeleştiriniz. (5)



S3. İki n bitlik işaretli integer sayının çarpımı için geliştirilen Booth algoritmasında toplam 4 register tanımlanmıştır: A (n bit akümülatör, başlangıçta sıfır), Q (n bit çarpan), Q_{-1} (1 bit, başlangıçta sıfır), M (n bit çarpılan). Çarpma işlemi için Q registerinin en az ağırlıklı biti Q_0 ve Q_{-1} e bakılarak üç farklı işlemden biri yürütülmektedir. Bu işlemlerin hangisinin yürütüleceği yandaki lojik devre tarafından üretilen F1, F2, ya da F3 işaretleri tarafından belirlenmektedir.



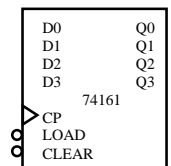
- (a) Bu işaretleri üreten logic devreyi tasarlamak için gereken doğruluk tablosunu oluşturunuz. (4)
- (b) Devreyi sadece NOR kapıları kullanarak gerçekleştiriniz. (7)
- (c) Aynı devreyi sadece 1-to-2 kod çözücü (decoder) ve OR kapıları kullanarak gerçekleştiriniz. (7)
- (d) Aynı devreyi sadece 2-to-1 çoğullayıcı (multiplexer) kullanarak gerçekleştiriniz. (7)

S4. Ardışıl bir devrenin durum tablosu yandaki gibidir.

- (a) Devrenin türünü (nedenleriyle) belirleyiniz. (5)
- (b) Durum diagramını çiziniz. (5)
- (c) Devreyi D-FF kullanarak gerçekleştiriniz. (8)
- (d) Aynı devreyi JK-FF kullanarak gerçekleştiriniz. (7)

Present state	Next state Input (XY =)				Output (Z =)
	00	01	10	11	
a	a	a	b	b	0
b	a	b	b	a	1

S5. Yandaki 4 bit sayıcı entegre devresini kullanarak **3** ten **10** a kadar sayan bir sayıcı devresi tasarlayınız. (10)



S6. $A=15$ ve $B=-10$ sayıları verildiğine göre;

- (a) Bu sayıların 2s complement sayı sistemindeki karşılıklarını 5 bit olarak bulunuz. (4)
- (b) $A+B$ ve $A-B$ işlemleri sonucunda taşma (overflow) ve eldevar (carry) değerleri ne olur? (6)

S7. Seri olarak gelen n bitlik bir sayının ikiye tümleyenini (2s complement) alan ardışıl (sequential) devreyi tasarlayınız. (10)

CEVAPLAR

C1.

$(C1960000)_{16} = (1\ 1000001\ 1\ 001011000000000000000000)_{\text{floating point}}$

$S = 1$, $BE = 10000011$, $M = 001011000000000000000000$

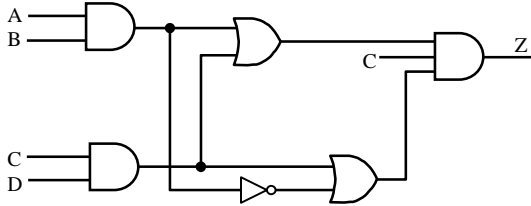
$$N = (-1)^S \times 1.M \times 2^{BE-127}$$

$$N = (-1)^1 \times 1.001011000000000000000000 \times 2^{10000011-01111111}$$

$$N = -(1.001011 \times 2^{00000100}) = -(10010.11)$$

$$N = -(10010.11)_2 = -(18.75)_{10}$$

C2.



(a) $Z = (AB + CD)(\overline{AB} + \overline{CD})C$

(b) $L = 9$; $G = 11$; $GN = 12$

(c) $Z = \overline{A}\overline{B}CD + \overline{A}B\overline{C}D + A\overline{B}\overline{C}D + ABCD = m_3 + m_7 + m_{11} + m_{15} = \sum m(3, 7, 11, 15)$

(d) $Z = (AB + CD)(\overline{AB} + \overline{CD})C = (AB\overline{A}\overline{B} + ABCD + \overline{A}\overline{B}CD + CDCD)C = (ABCD + \overline{A}\overline{B}CD + CD)C$
 $Z = ((AB + \overline{A}\overline{B} + 1)CD)C = (CD)C = CD$

C3.

(a)

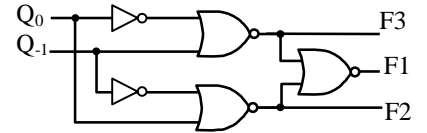
Q_0	Q_{-1}	F1	F2	F3
0	0	1	0	0
0	1	0	1	0
1	0	0	0	1
1	1	1	0	0

(b)

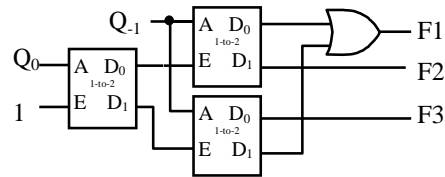
$$F1 = (Q_0 + \overline{Q}_{-1})(\overline{Q}_0 + Q_{-1})$$

$$F2 = \overline{Q}_0 Q_{-1}$$

$$F3 = Q_0 \overline{Q}_{-1}$$

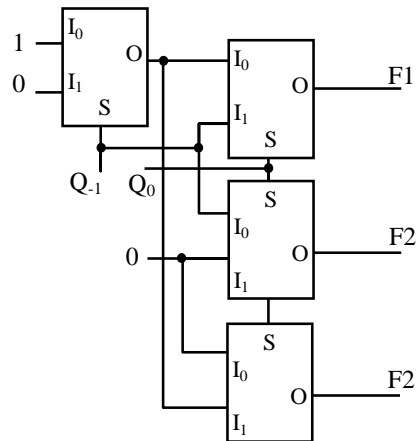


(c) 1-to-2 line decoder implementation



(d) 2-to-1 multiplexer implementation

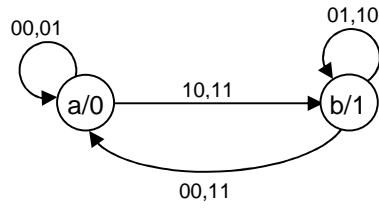
Q_0	Q_{-1}	F1	F2	F3
0	0	1	0	0
0	1	0	1	0
1	0	0	0	1
1	1	1	0	0



C4.

(a) Bu devre Moore tipi bir ardışıl devredir. Çünkü çıkış girişten bağımsızdır (sadece durumun bir fonksiyonudur).

(b) Durum diagramı:



(c) İki giriş (XY), bir çıkış (Z), ve iki durum vardır. Buna göre gereken flip-flop sayısı 1 dir ($\log_2 2 = 1$). Durum tablosu yeniden düzenlenir ve gerekli durum ataması yapılırsa;

Present state	Next state Input (XY =)				Output (Z =)
	00	01	10	11	
a	a	a	b	b	0
b	a	b	b	a	1

Present state	Next state Input (XY =)				Output (Z =)
	00	01	11	10	
0	0	0	1	1	0
1	0	1	0	1	1

Ya da aşağıdaki şekilde düzenlenirse

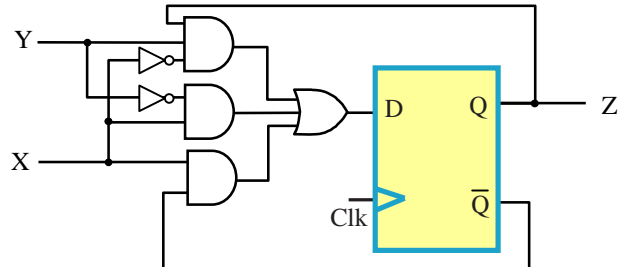
Present state	Input		Next state	Out
	X	Y	Q ⁺	
a	0	0	a	0
a	0	1	a	0
a	1	0	b	0
a	1	1	b	0
b	0	0	a	1
b	0	1	b	1
b	1	0	b	1
b	1	1	a	1

Present state	Input		Next state	Out
	X	Y	Q ⁺	
0	0	0	0	0
0	0	1	0	0
0	1	0	1	0
0	1	1	1	0
1	0	0	0	1
1	0	1	1	1
1	1	0	1	1
1	1	1	0	1

XY \ Q	0	1
00	0	0
01	0	1
11	1	0
10	1	1

XY \ Q	0	1
00	0	1
01	0	1
11	0	1
10	0	1

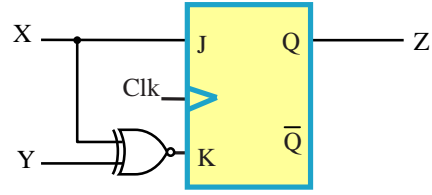
$$D_Q = X\bar{Y} + X\bar{Q} + \bar{X}YQ \quad Z = Q$$



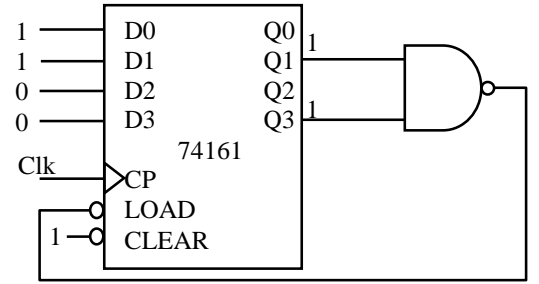
(c)

$$J = X \quad K = \overline{X \oplus Y} \quad Z = Q$$

PS	Input		NS	FF inputs		Out
Q	X	Y	Q ⁺	J	K	Z
0	0	0	0	0	x	0
0	0	1	0	0	x	0
0	1	0	1	1	x	0
0	1	1	1	1	x	0
1	0	0	0	x	1	1
1	0	1	1	x	0	1
1	1	0	1	x	0	1
1	1	1	0	x	1	1



C5. 3 ten 10 a kadar sayan devrenin başlangıç değeri 3 = 0011, bitiş değeri 10 = 1010. Buna göre;
 $D0 = 1, D1 = 1, D2 = 0, D3 = 0$ olmalı.
 $Q0 = 0, Q1 = 1, Q2 = 0, Q3 = 1$ olduğunda LOAD girişi etkinleştirilerek yukarıdaki giriş değeri (0011) devrenin başlangıç değeri olarak yüklenmeli. Girişleri Q1 ve Q3 e bağlı bir NAND kapısı gerekli işareti oluşturmak için yeterlidir. Yandaki devre 3 ten 10 a kadar sayma işlemini gerçekleştirir.



C6.

(a) $A = (+15)_{10} = (01111)_{2s \text{ comp. sayı sistemi}}$
 (b) $A+B$

$B = (-10)_{10} = (10110)_{2s \text{ comp. sayı sistemi}}$
 $A-B = A + B$ nin 2s comp.

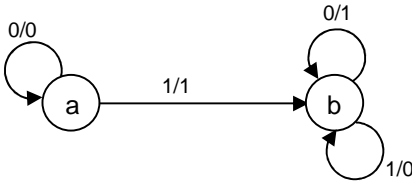
$(+15) \rightarrow 01111$
 $(-10) \rightarrow 10110$
 $1 \ 00101$

$(+15) \rightarrow 01111$
 $(+10) \rightarrow 01010$
 $0 \ 11001$

taşma (overflow) = 0
 elde var (carry) = 1

taşma (overflow) = 1
 elde var (carry) = 0

C7. Giriş x çıkış y olarak alınırsa; durum diagramı, durum tablosu ve devrenin gerçekleştirimi aşağıdaki gibidir:
 $a = 0, b = 1$ seçilirse;



PS (Q)	NS(Q ⁺)		Output (y)	
	x=0	x=1	x=0	x=1
a	a	b	0	1
b	b	a	1	0

PS (Q)	NS(Q ⁺)		Output (y)	
	x=0	x=1	x=0	x=1
0	0	1	0	1
1	1	1	1	0

D-FF kullanılırsa;

$$D_Q = x + Q, \quad y = x \oplus Q$$

PS (Q)	NS(Q ⁺)		J		K	
	x=0	x=1	x=0	x=1	x=0	x=1
0	0	1	0	1	*	*
1	1	1	*	*	0	0

JK-FF kullanılırsa;

$$J = x, \quad K = 0, \quad y = x \oplus Q$$

