

İÜ Mühendislik Fakültesi
Bilgisayar Mühendisliği Bölümü

Ad-Soyad:
No:
İmza:

23/05/2018

1	2	3	4	5	6	T

Bilgisayar Organizasyonu ve Tasarımı Final Sınavı

1. a) Komut kümesinin tam olarak kabul edilebilmesi için hangi kriterlerin sağlanması gerekir?
b) Kesme servis rutin görevleri nelerdir?
c) Mikro-programlanmış (yazılımsal) kontrolde kaç tip bellek vardır. Belleklerin türlerini ve yapısını adres & data hattı sayılarını da belirterek veriniz.

- a) Temel bilgisayara ait olan PC saklayıcısına ilişkin kontrol işaretlerini ve RTL ifadelerini yazarak bu saklayıcıya ilişkin tüm işlemleri gerçekleştiren devreleri ortak veri yolunu da içerecek şekilde tasarlayıp çiziniz.
b) Temel bilgisayarda bulunan kesme flip-flop'u R'nin kontrol kapı devrelerini bir JK tipi ff ve minimum sayıda kapı kullanarak çiziniz.

3. Aşağıda assembly programı verilmiş olan kodun yan tarafına onaltılık biçimde makine kodu karşılığını adres bilgisini de göstererek yazınız ve her komut çalıştıktan sonra onaltılık olarak AC ve PC'nin içeriğini yazarak tabloyu doldurunuz. Program ne iş yapmaktadır?

	Adres	MakineKod	AC	PC
ORG 500				
LDA Y				
CMA				
INC				
ADD X				
BUN V				
A, INC				
STA Z				
HLT				
V, AND W				
BUN A				
X, DEC 67				
Y, DEC -12				
W, HEX FFF0				
Z, HEX 0				
END				

4. Aşağıda assembly kodu verilmiş olan programda toplama komutu yürütülürken kesme oluşmuştur. Bu göre, eksik olan (...) satırları tamamlayınız, altı çizili yerlere gelecek olan komutları ve etiketleri yazı ve "/" işareti olan satırların açıklamasını yazınız. Kesmenin oluştuğu adresi ve kesmeden so programın döndüğü adresi belirtiniz.

```

ORG 0
SFR, HEX 0      /
BUN SER

ORG 150
CLA
ION
LDA X
.....
STA Z

ORG 350
SER, STA _____
CIR          /
_____ SE
SKI          /
.....
INP
OUT
STA PT1 I
ISZ PT1      /
SNR, SKO     /
BUN CIK
LDA PT2 I
OUT
ISZ PT2
.....
CIL
LDA SAC
.....
BUN SFR I    /
AC, HEX 400
HEX 450
I, HEX 500
HEX 550

```

5. Herhangi bir anda bir işlemcideki saklayıcılar PC = 300, RI = 150, XR = 50 ve RAM'deki bazı bellek gözleri de M[100] = 80, M[150] = 500, M[401] = 700 değerlerine sahiptir. Load komutu belirtilen adresleme moduna göre AC'nin içeriğine kaynaktaki veriyi yüklemektedir. Buna göre aşağıdaki beş işlemin her birinde kullanılan adresleme modunun adını belirtip AC'ye yüklenecek değerleri bulunuz.

a) Load #300

b) Load 50(XR)
index adresleme 100

c) Load \$100
dolaylı adresleme 80

d) Load (R1)+
öbölüm artırım 650

e) Load 100
değerden adresleme 100

2.b) (Yer kalmadıysa isterseniz 2. Sorunun b şıkkının çözümünü burada gösterebilirsiniz.)

6. Her bir soru için doğru cevabı işaretleyiniz.

(9 / 3) - (4 x 2) şeklinde verilen infix ifade, postfix(sonek) gösterime çevrilerek yığın organizasyonlu bir yapıda hesaplanacaktır. Buna göre dördüncü adımda yığına atılan sayı kaçtır?

- a) 2 b) 3 c) 4 d) 8 e) 9

ii. Hangisi bir mikrokmut alanı değildir?

- a) Adres ☒ b) Şart bitleri c) İşlem kodu d) Dallanma durumu ☒ e) Mikroişlemler

7. "Komutun içinde, operandın saklandığı saklayıcının adresi bulunur." şeklinde tanımlanan adresleme modu hangisidir?

- a) Doğrudan adresleme b) Bağlı adresleme c) Sıralı adresleme
d) Saklayıcı adresleme e) Saklayıcı dolaylı adresleme

- iv. Aşağıdaki mikroişlemlerden hangisinde sayının işaret biti korunur?
- a) CIL b) CIR c) SHR d) ASHL e) ASHR
- v. RISC mimarisine sahip bir bilgisayarın her penceresi, 20 yerel saklayıcı (L), 2x5 ortak saklayıcı (2C) ve 20 global saklayıcı (G) kullanabilmektedir. Pencere sayısı (w) & olduğuna göre bu bilgisayarda toplam kaç saklayıcı vardır?
- a) 160 b) 180 c) 200 d) 220 e) 240

Temel bilgisayar için kontrol fonksiyonu ve mikroişlemleri içeren komut listesi

Fetch	R'T0:	AR ← PC
Decode	R'T1:	IR ← M[AR], PC ← PC + 1
Indirect	R'T2:	D0, ..., D7 ← Decode IR(12 ~ 14),
Interrupt	D7'IT3:	AR ← IR(0 ~ 11), i ← IR(15)
		AR ← M[AR]
	T0'T1'T2'(IEN)(FGI + FGO):	R ← 1
	RT0:	AR ← 0, TR ← PC
	RT1:	M[AR] ← TR, PC ← 0
	RT2:	PC ← PC + 1, IEN ← 0, R ← 0, SC ← 0
Memory-Reference		
AND	D0T4:	DR ← M[AR]
ADD	D0T5:	AC ← AC + DR, SC ← 0
LDA	D1T4:	DR ← M[AR]
	D1T5:	AC ← AC + DR, E ← Cout, SC ← 0
STA	D2T4:	DR ← M[AR]
BUN	D2T5:	AC ← DR, SC ← 0
BSA	D3T4:	M[AR] ← AC, SC ← 0
	D4T4:	PC ← AR, SC ← 0
	D5T4:	M[AR] ← PC, AR ← AR + 1
ISZ	D5T5:	PC ← AR, SC ← 0
	D6T4:	DR ← M[AR]
	D6T5:	DR ← DR + 1
	D6T6:	M[AR] ← DR, if(DR=0) then (PC ← PC + 1), SC ← 0
Register-Reference		
	D7I'T3 = r	(tüm saklayıcı referanslı komutlar için ortak)
	IR(i) = Bi	(i = 0, 1, 2, ..., 11)
	r:	SC ← 0
CLA	rB11:	AC ← 0
CLE	rB10:	E ← 0
CMA	rB9:	AC ← AC'
CME	rB8:	E ← E'
CIR	rB7:	AC ← shr AC, AC(15) ← E, E ← AC(0)
CIL	rB6:	AC ← shl AC, AC(0) ← E, E ← AC(15)
INC	rB5:	AC ← AC + 1
SPA	rB4:	If(AC(15)=0) then (PC ← PC + 1)
SNA	rB3:	If(AC(15)=1) then (PC ← PC + 1)
SZA	rB2:	If(AC = 0) then (PC ← PC + 1)
SZE	rB1:	If(E=0) then (PC ← PC + 1)
HLT	rB0:	S ← 0
Input-Output	D7IT3 = p	(tüm I/O referanslı komutlar için ortak)
	IR(i) = Bi	(i = 6, 7, 8, 9, 10, 11)
	p:	SC ← 0
INP	pB11:	AC(0-7) ← INPR, FGI ← 0
OUT	pB10:	OUTR ← AC(0-7), FGO ← 0
SKI	pB9:	If(FGI=1) then (PC ← PC + 1)
SKO	pB8:	If(FGO=1) then (PC ← PC + 1)
ION	pB7:	IEN ← 1
IOF	pB6:	IEN ← 0