

23/05/2018

1	2	3	4	5	6	T

IÜ Mühendislik Fakültesi
Bilgisayar Mühendisliği Bölümü

Ad-Soyad:
No:
İmza:

Bilgisayar Organizasyonu ve Tasarımı Final Sınavı

- I. a) Komut kümesinin tam olarak kabul edilebilmesi için hangi kriterlerin sağlanması gereklidir?
b) Kesme servis rutinin 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 çalıştırınca 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 FFFF				
Z, HEX 0				
END				

4. Aşağıda assembly kodu verilmiş olan programda toplama komutu yürütülürken kesme oluşmuştur. Bu göre, eksik olan (...) satırları tamamlayınız, altı çizili yerbeler gelecek olan komutları ve etiketleri yazı ve "/" işaretini olan satırların açıklamasını yazınız. Kesmenin olduğu adresi ve kesmeden sonra 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			
E, 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 beştiilen adresleme moduna göre AC'nin içeriğine kaynaktaki veriyi yüklemektedir. Buna göre aşağıdaki beş işlemin her birinde kullanılan adresleme
- a) Load #300
 b) Load 50(XR)
 c) Load \$100
 d) Load (R1)+
 e) Load 100

2.b) (Yer kalmadıysa isterseniz 2. Sorunun b şikkinin 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 mikrokomut alanı değildir?

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

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

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

- i.* Aşağıdaki mikroişlemelerden hangisinde sayıının işaret biti korunur?
- CIL
 - CIR
 - SHR
 - ASHL
 - 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) 8 olduğuna göre bu bilgisayarda toplam kaç saklayıcı vardır?
- 160
 - 180
 - 200
 - 220
 - 240

Temel bilgisayar için kontrol fonksiyonu ve mikroişlemeleri içeren komut listesi

	Fetch	
Decode	R'T0: R'T1: R'T2:	AR ~ PC IR ~ M[AR], PC ~ PC + 1 D0, ..., D7 ~ Decode IR(12 ~ 14),
Indirect	D7'IT3:	AR ~ IR(0 ~ 11), I ~ IR(15)
Interrupt	T0'T1'T2'(IEN)(FGI + FGO): RT0: RT1: RT2:	AR ~ M[AR] R ~ 1 AR ~ 0, TR ~ PC M[AR] ~ TR, PC ~ 0
Memory-Reference	RT3: ADD LDA STA BUN BSA ISZ	PC ~ PC + 1, IEN ~ 0, R ~ 0, SC ~ 0 DR ~ M[AR] AC ~ AC + DR, SC ~ 0 DR ~ M[AR] AC ~ AC + DR, E ~ Cout, SC ~ 0 DR ~ M[AR] AC ~ DR, SC ~ 0 M[AR] ~ AC, SC ~ 0 PC ~ AR, SC ~ 0 M[AR] ~ PC, AR ~ AR + 1 PC ~ AR, SC ~ 0 DR ~ M[AR] DR ~ DR + 1 M[AR] ~ DR, if(DR=0) then (PC ~ PC + 1), SC ~ 0
Register-Reference	D7'I'T3 = r IR(i) = Bi r:	(tüm saklayıcı referanslı komutlar için ortak) (i = 0,1,2, ..., 11) 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	D7'IT3 = p IR(i) = Bi p:	(tüm I/O referanslı komutlar için ortak) (i = 6,7,8,9,10,11) 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