

## BİLGİSAYAR MİMARİSİ

### FİNAL SINAVI – 2018 (Örgün + İkili)

1) Aşağıdaki soruları MIPS Assembler programlama ile yanıtlayınız?

A) **abs** Rdest, Rsrc komutu  $Rdest \leftarrow \| Rsource \|$  Rsrc register içeriğinin mutlak değeri  
abs pseudo komutunu gerçek (real) makine komutları kullanarak gerçekleyiniz?

B) Aşağıdaki ifadeyi MIPS assember dili karşılığını yazınız?

a, b, c, ve d sırasıyla \$s0, \$s1, \$s2, ve \$s3 register içinde tutulmuş olsunlar.

Bütün sayılar 32 bit işaretli sayılar olduğu kabul edilecektir... || or işlemi

**if ((a > b) || (b > c)) {d = 1;}**

2) Aşağıdaki MIPS Assembler programı bir dizi üzerinde işlem yaparak sonuçları \$v0 ve \$v1 registerler içinde tutmaktadır. Dizin 5000 words içermektedir (0 ile 4999 olarak indekslenen), ve dizin taban adresi \$a0 içinde , ve kelime sayısı (5000) \$a1 içinde tutulduğunu düşünün. .  
1 Cümle halinde programın ne yaptığını tanımlayın, \$v0 ve \$v1 içinde tutulan değerlerin ne olduğunu belirtiniz?

```
add $a1, $a1, $a1
add $a1, $a1, $a1
add $v0, $zero, $zero
add $t0, $zero, $zero
outer: add $t4, $a0, $t0
      lw $t4, 0($t4)
      add $t5, $zero, $zero
      add $t1, $zero, $zero
inner: add $t3, $a0, $t1
      lw $t3, 0($t3)
      bne $t3, $t4, skip
      addi $t5, $t5, 1
skip:  addi $t1, $t1, 4
      bne $t1, $a1, inner
      slt $t2, $t5, $v0
      bne $t2, $zero, next
      add $v0, $t5, $zero
      add $v1, $t4, $zero
next:  addi $t0, $t0, 4
      bne $t0, $a1, outer
```

3) Aşağıda özellikleri verilmiş 2 farklı Cache bellek için a-d şıklarını yanıtlayınız?

**Cache 1 :**

Türü: Direct-mapped cache.

Cache satır uzunluğu (line) 1 byte.

Index = 10-bit; Takı (tag): 6-bit; hit time = 1 cycle

**Cache 2:**

Türü: 2 way set associative cache.

Cache satır uzunluğu (line) 1 word=4 byte

Index= 7-bit; Takı (tag): 7-bit; hit time = 2 cycle

- a) Her 2 bellek için Cache Size (kapasitesini ) hesaplayınız?
- b) Takıları saklamak için ne kadarlık Cache alanına ihtiyaç bulunmaktadır (her 2 Cache için ayrı ayrı hesaplanacak)?
- c) Cache vuru oranı (hit rate) 70 % ve 50% olduğu bilindiği, ancak hangi cache hangi vuru oranına sahip olduğu bilinmediğini kabul ediniz.  
Sizce Cache 1 ve Cache 2 için Vuru oranlarını belirlemek mümkünmüdür, nedenini belirterek yazınız?
- d) Her 2 cache için ıskala zamanı=20 cycle ise ve c) şıkında belirlenen Vuru Oranları kullanılarak her 2 bellek için Efektif Bellek Varış Zamanını  $T_E$  bulunuz?

4) A) Aşağıdaki MIPS Assembler programını pipelined MIPS mimarisi üzerinde çalıştırıldığında oluşan problemleri (hazards) belirtiniz, hangi satırlarda oluştuğunu gösteriniz?

lw \$t1, 0(\$t0)

lw \$t2, 4(\$t0)

add \$t3, \$t1, \$t2

sw \$t3, 12(\$t0)

lw \$t4, 8(\$t0)

add \$t5, \$t1, \$t4

sw \$t5, 16(\$t0)

B) Bu problemi gidermek için alınan önlemleri bir pipeline zamanlama diyagramı çizerek gösteriniz? Kaç saat çevriminde programın icra edildiğini yazınız?

C) Program komutlarının sırasını değiştirerek (sadece 1 komut) a. şıkındaki pipeline hazard probleminin gecikme (stall) konulmadan giderilebileceğini gösteriniz ? (pipeline zamanlama diyagramı çizilebilir) Yeni durumda kaç saat çevriminde programın icra edildiğini yazınız?

5) a) Bir I/O Arabirim Ünitesinin temel görevlerini (fonksiyonlarını) maddeler halinde yazınız?

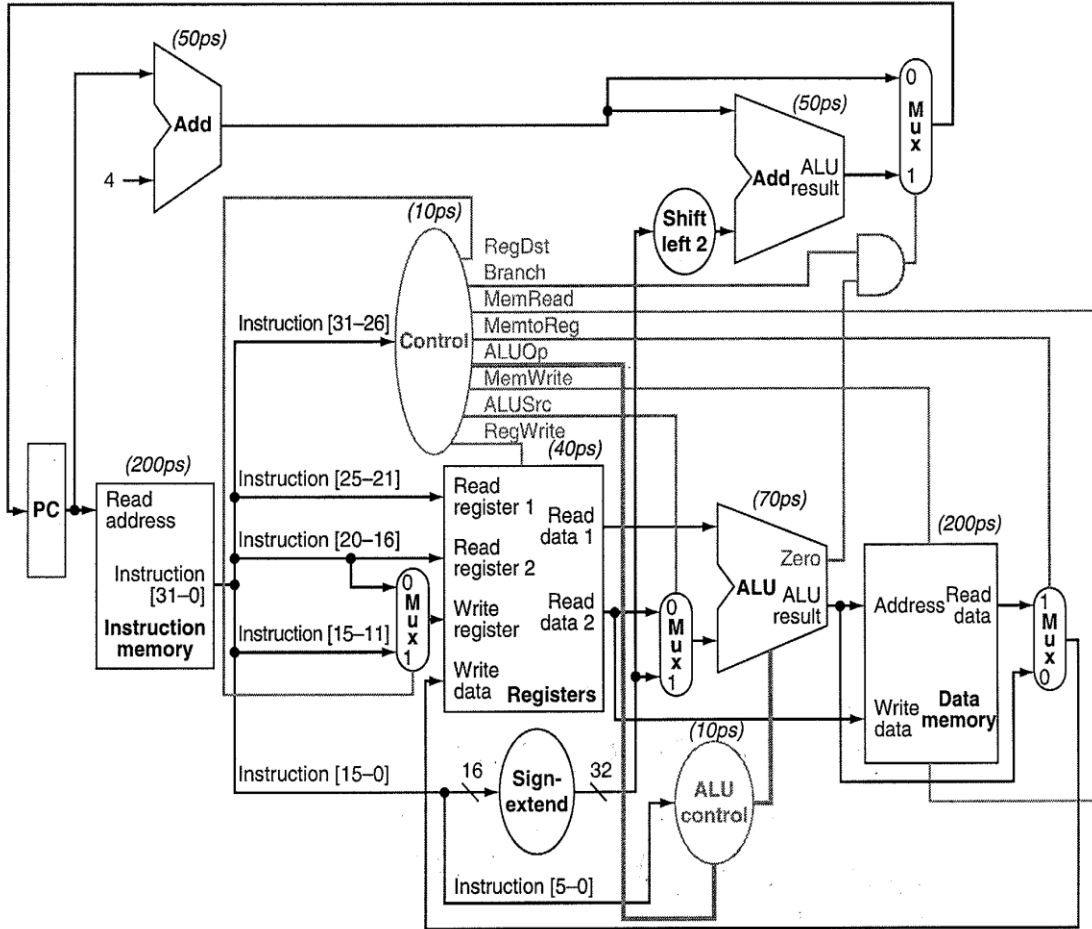
b) I/O ünitesinin blok diyagramını çiziniz?

c) Asenkron data transfer yöntemlerini 1 cümle ile kısaca açıklayınız?

6) Mips komut setine addm komutu ilave edilmek isteniliyor..  
 $\text{addm rd, SBT(rs)} ; \text{rd} = \text{M}[\text{r(s)}] + \text{SBT}$

a) (10p) Bu işlemi yapabilmek için datapath yapısına eklenecek komponentleri ve kontrol çıkışı olarak ekstra ilaveleri de ekleyerek MIPS datapath üzerinde gösteriniz?

b) (10p) Aşağıda verilen Tek Çevrimli MIPS mimarisi için verilen unite gecikmeleri gözönüne alınarak mux ve kontrol işaret gecikmeleri ihmal edilirse normal **add** komutu ve ilave edilen **addm** komut gecikmelerini bulunuz?



Süre 90 dakika

Puanlar: 1) a)10p b) 5p 2) 15p 3) 20p 4) 15p 5) 15p 6) 20p

Başarılar....

## 2018 MİMARİ FİNAL ÇÖZÜMLERİ

1)a-

**abs** Rdest, Rsrc

bgez Rsrc,skip

sub Rdest,\$0,Rsrc

skip: -----

**b) a) if ((a > b) || (b > c)) {d = 1;}**

Solution: bgt \$s0, \$s1, L1

ble \$s1, \$s2, next

L1: ori \$s3, \$zero, 1 next:

**2)** This code compares every element in the array against all elements for identical matches.

It counts the frequency of occurrence of each value in the array.

The count of the most frequently used value is returned in \$v0 and the value itself is returned in \$v1.

**3) a)** cache 1: 1024 1B lines = 1KB.

Cache 2: 128 4B lines \* 2 ways = 1KB

b) cache 1: 1024 x 6-bit tags = 6Kb

cache 2: 256 x 7-bit tags = 1792 bits

c) Her 2 Cache Size eşit olduklarından ve 2 way set-associative yapıda aynı satırda 2 farklı data bulunduğundan vuru oranı daha yüksek olacaktır.

Buna göre Cache 1= 50%, Cache 2= 70% Vuru Oranına sahip olduğu söylenebilir..

d) etkin bellek varış süresi = (hit rate)\*(hit time) + (miss rate)\*(miss time)).

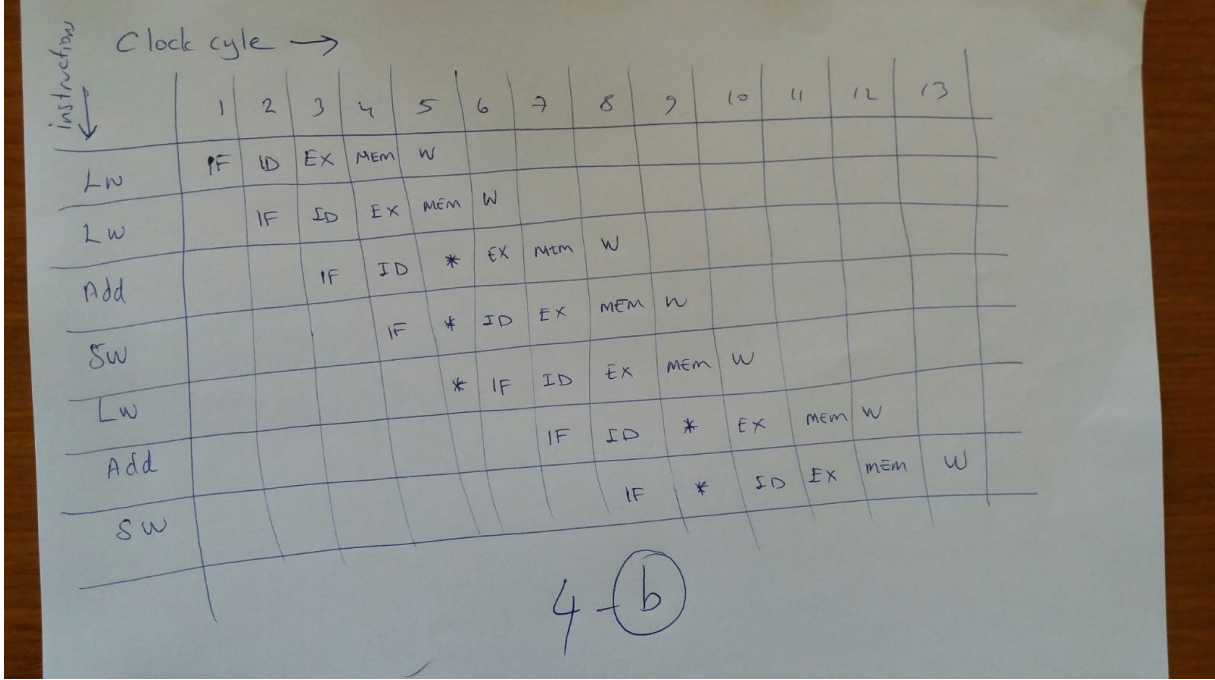
Cache 1  $T_e = 0.5 * 1 + 0.5 * 20 = 10.5$  cycles

Cache 2  $T_e = 0.7 * 2 + 0.3 * 20 = 7.4$  cycles

4- a) Data hazard problem var.. Her 2 add komut satırında data bağımlılığı bulunmaktadır..

b) Add komutu satırlarında pipeline diyagramında STALL konulmalıdır..

Toplam 13 cc de program icra olur.



c) lw \$t1, 0(\$t0)

lw \$t2, 4(\$t0)

lw \$t4, 8(\$t0)

add \$t3, \$t1, \$t2

sw \$t3, 12(\$t0)

add \$t5, \$t1, \$t4

sw \$t5, 16(\$t0)

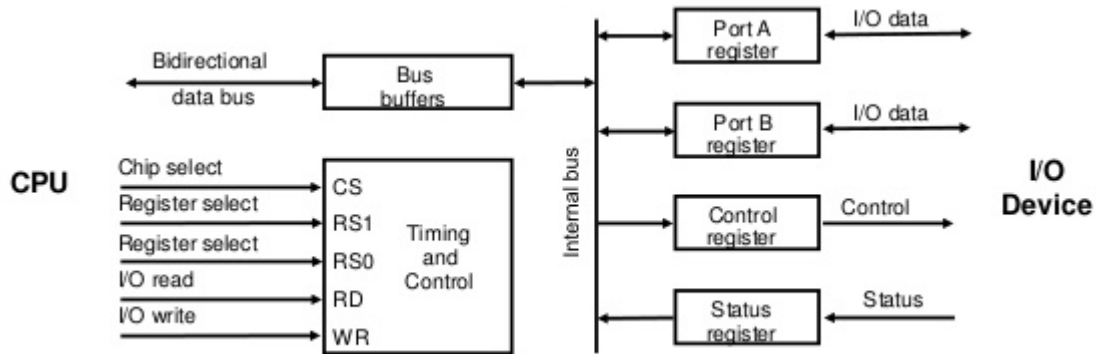
Pipeline diyagramı çizilirse 11 cc de programın icra edildiği görülür.

5) a)

- Eleman adreslerinin kodçözümü (eleman kodu)
- Komutların kodçözümü (işlem)
- Çevresel kontrolör için gerekli işaretleri sağlar.
- Data akışını senkronize eder ve yönlendirir.

Çevresel elemanlar ve CPU veya Bellek arasında transfer hızını ayarlar.

b)



c) 2 Asenkron Data Transfer Metodu

1) Strobe darbesi :

Bir strobe darbesi, transfer meydana gelmek zorunda olduğunda diğer bir birimi göstermek için bir birimin ürettiği işarettir.

Herbir transferi zamanlamada, tek bir kontrol işareti kullanır.

Strobe işaretini kaynak veya hedef ünite gönderebilir.

2) El Sıkışma :

- Bir kontrol işareti datanın varlığını göstermek için iletilmekte olan her bir dataya eşlik etmesidir.

- Alıcı birim datayı kabul ettiğini onaylamak için diğer bir kontrol işareti gönderir.

#### Kaynak-Etkili iletişim

Transferi başlatan Kaynak Ünitesinin hedef ünitesinin gerçekten datayı alıp almadığından bilgisi yoktur.

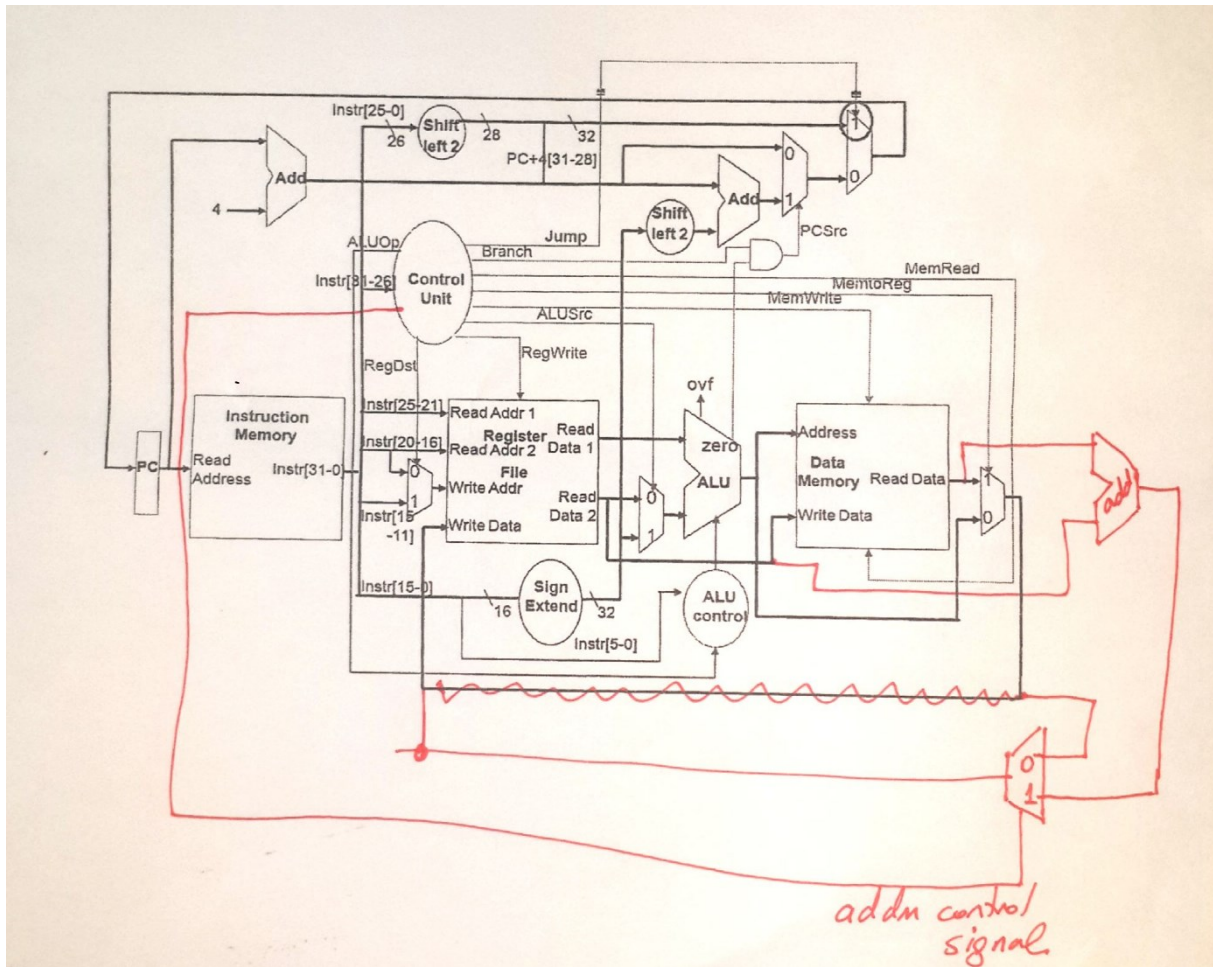
#### Hedef-Etkili iletişim

Transferi başlatan Hedef Ünitesinin kaynağın datayı bus üzerine yerleştirip yerleştirmedeği bilgisi yoktur.

Bu problemi çözmek için, ELSIKİŞMA Metodu kullanılır.

Transferi başlatan birime bir cevap sağlamak için 2. bir kontrol işareti gönderilir.

6) a)



b)

	IM	REG	ALU	DM	REGW	ilave ADD	Toplam
Add	200	40	70	----	40		350
Addm	200	40	70	200	40	50	600