

BM-311 Bilgisayar Mimarisi

Hazırlayan: M.Ali Akcayol
Gazi Üniversitesi
Bilgisayar Mühendisliği Bölümü

Konular

- **Bilgisayar Bileşenleri**
- Bilgisayarın Fonksiyonu
 - Instruction Cycle
 - Kesmeler (Interrupt'lar)
- Bus Yapıları
 - Bilgisayar Modülleri ve Bağlantıları
 - Bus Tasarım Kriterleri

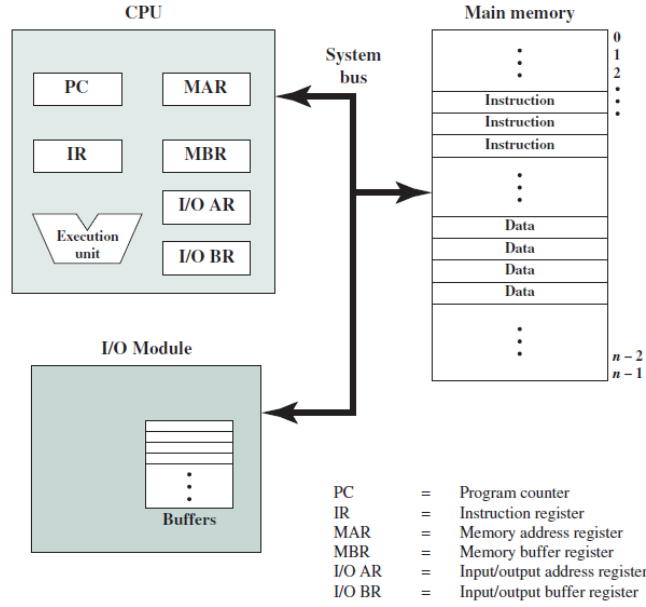
Bilgisayar Bileşenleri

- **Donanımsal sistemler** (hardwired system) **esnek değildir.**
- **Genel amaçlı donanımlar** kontrol sinyallerine göre farklı işler yapabilir.
- Donanımsal olarak bağlantıları yeniden oluşturmak yerine **yazılım kullanarak sadece kontrol işaretleri oluşturulur.**
- Bir program sıralı komut kümesidir.
 - Her adımda **aritmetik veya mantık bir işlem** yapılır.
 - Her işlem için çok sayıda **sıralı/sırasız kontrol işareti** üretilir.

Bilgisayar Bileşenleri

- Her işlem için **tekil bir kod sağlanır** (ADD, MOVE).
- **Donanım** kodu alır ve **kontrol işaretlerini üretir.**
- **Kontrol birimi** ve **aritmetik mantık birim** mikroişlemciyi oluşturur.
- Veriler ve komutlar sisteme dışarıdan alınır veya elde edilen sonuç dışarı aktarılır (input/output-I/O).
- **Program ve verilerin geçici saklanması için** geçici bir depolama birimine ihtiyaç duyulur (**main memory**).

Bilgisayar Bileşenleri



Konular

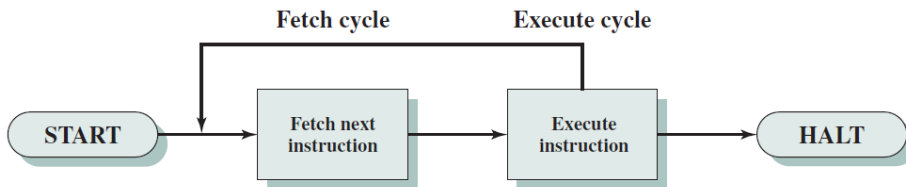
- Bilgisayar Bileşenleri
- **Bilgisayarın Fonksiyonu**
 - Instruction Cycle
 - Kesmeler (Interrupt'lar)
- Bus Yapıları
 - Bilgisayar Modülleri ve Bağlantıları
 - Bus Tasarım Kriterleri

Bilgisayarın Fonksiyonu

- **Mikroişlemci** gerçekleştireceği işlemleri programdaki **komutları kullanarak yapar.**
- En temel olarak **mikroişlemci iki adımda bir işlemi gerçekleştirir.**
- Birinci adımda **komutlar mikroişlemciye alınır (fetch).**
- İkinci adımda **komut çalıştırılır (execute).**
- Bu iki adıma **komut döngüsü (instruction cycle)** denilmektedir.

Bilgisayarın Fonksiyonu - Instruction Cycle

- **Execute** adımı **birden fazla alt adımdan oluşabilir.**
- Örneğin, komut operand gerektiriyorsa operand'ların alınması execute adımında gerçekleştirilebilir.



Bilgisayarın Fonksiyonu - Fetch Cycle

- **Program counter (PC)** bir sonraki komutun adresini tutar.
- **İşlemci** PC ile gösterilen **adresten komutu alır.**
- **PC** alınan komut boyutu kadar **artırılır** (Atlama komutu çalışırsa farklı bir adrese geçilir).
- Hafızadan alınan komut **Instruction Register (IR)**'a aktarılır.
- **İşlemci** alınan **komutu yorumlar** ve gereken **işlemleri yapar.**

Bilgisayarın Fonksiyonu - Execute Cycle

- Execute aşamasında farklı işlemler yapılabilir.
- **CPU-Hafıza**
 - CPU ile hafıza arasında veri aktarılır.
- **CPU-I/O**
 - CPU ile I/O cihazları arasında veri aktarılır.
- **Veri üzerinde işlem**
 - Aritmetik ve mantık işlemlerden birisi yapılır.
- **Kontrol**
 - Programdaki komutların çalışma sırasında değişiklik yapılabilir.
- Yukarıdakilerden birkaç tanesi birlikte yapılabilir.

Bilgisayarın Fonksiyonu-Örnek Program

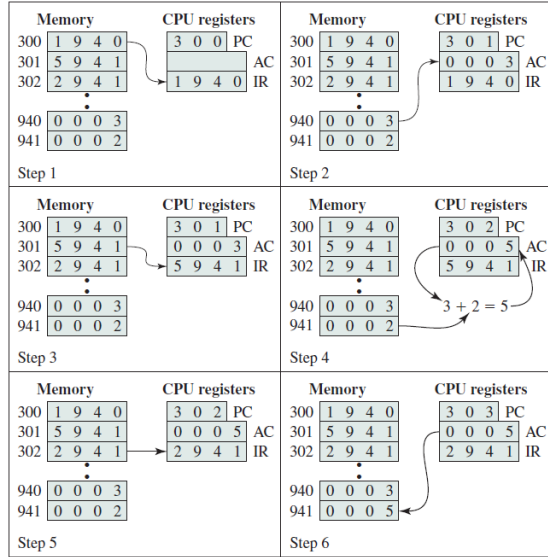
```
1940 LOAD MEM ; AC <- MEM
0001 1001 0100 0000
```

```
2941 STORE MEM ; MEM <- AC
0010 1001 0100 0001
```

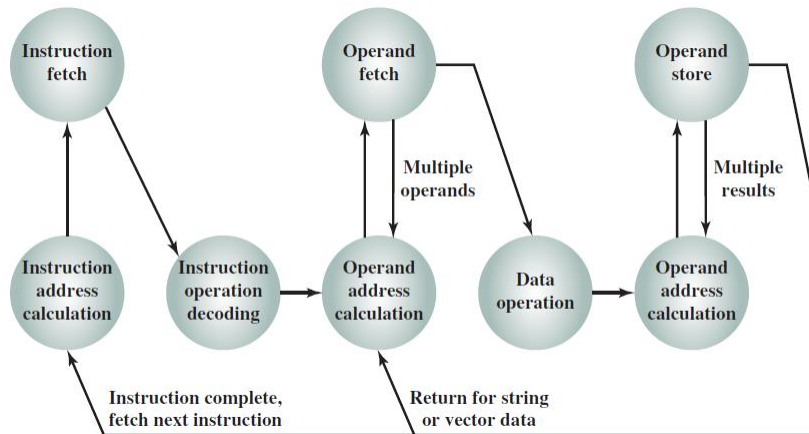
```
5941 ADD MEM ; AC <- MEM + AC
0101 1001 0100 0001
```

Opcode Operand

0001	1001 0100 0000
------	----------------



Bilgisayarın Fonksiyonu-Komut Döngüsü



Bilgisayarın Fonksiyonu-Kesmeler

- Kesmeler I/O cihazları tarafından üretilebilirler ve işlemcinin normal çalışmasını keserler.
- **Program kesmeleri**
 - Overflow, division by zero.
- **Timer kesmeleri**
 - CPU'nun içindeki timer'lar tarafından üretilirler.
- **I/O kesmeleri**
 - I/O denetleyicileri tarafından oluşturulur.
- **Donanım kesmeleri**
 - Memory parity hatası, pil uyarısı, disk okuma hatası.

Bilgisayarın Fonksiyonu-Kesmeler

Program akışı

USER PROGRAM

{
<statement>
<statement>
⋮
<statement>
}

Code segment 1

WRITE

{
<statement>
<statement>
⋮
<statement>
}

Code segment 2

WRITE

{
<statement>
<statement>
⋮
<statement>
}

Code segment 3

I/O PROGRAM

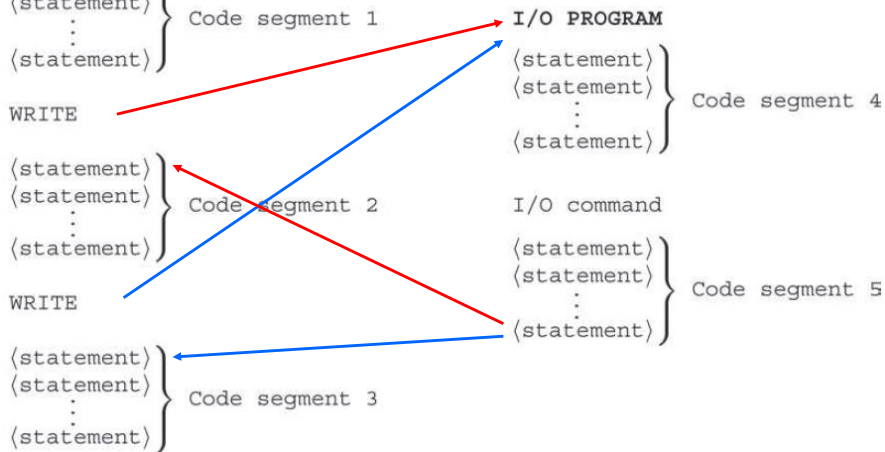
{
<statement>
<statement>
⋮
<statement>
}

Code segment 4

I/O command

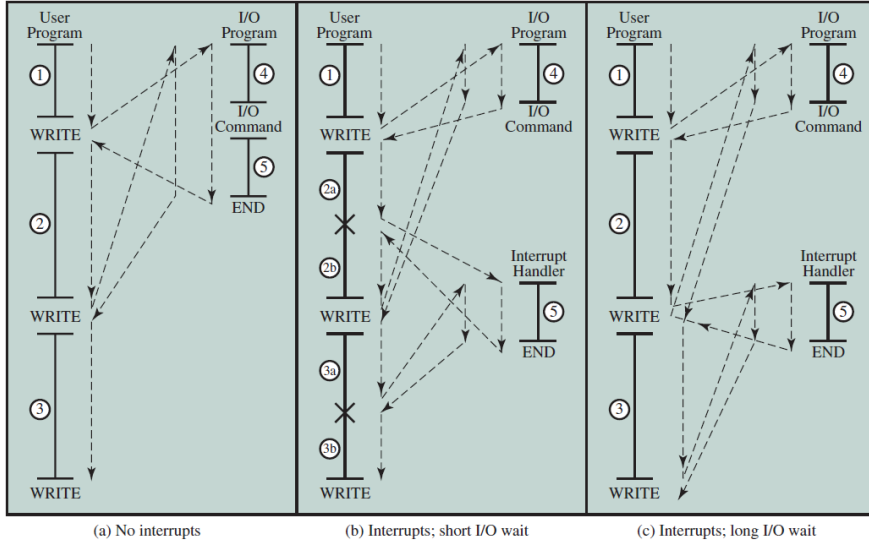
{
<statement>
<statement>
⋮
<statement>
}

Code segment 5



Bilgisayarın Fonksiyonu-Kesmeler

Program akışı



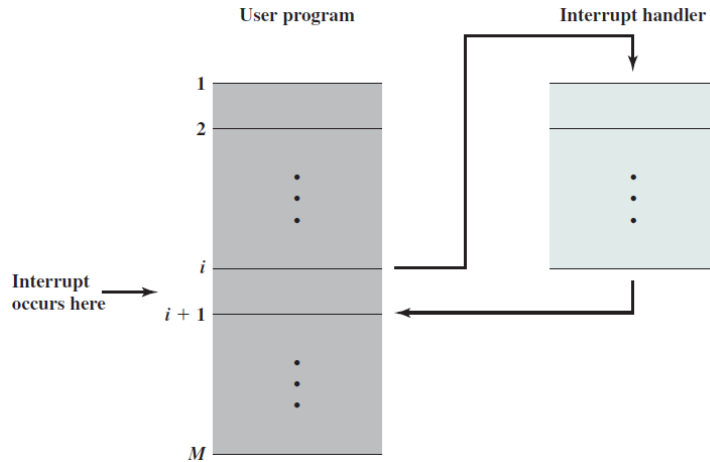
Bilgisayarın Fonksiyonu-Kesmeler

Interrupt cycle

- **Komut döngüsünün sonuna eklenir.**
 - İşlemci keme gelip gelmediğini kontrol eder.
- **Keme gelmemişse** sonraki komut fetch edilir.
- **Keme gelmişse,**
 - Çalışmakta olan program beklemeye alınır.
 - Register içerikleri saklanır.
 - PC'ye yeni adres değeri aktarılır.
 - Keme için gerekli işlemler gerçekleştirilir.
 - Önceki programa dönülür ve register değerleri yeniden yüklenir.

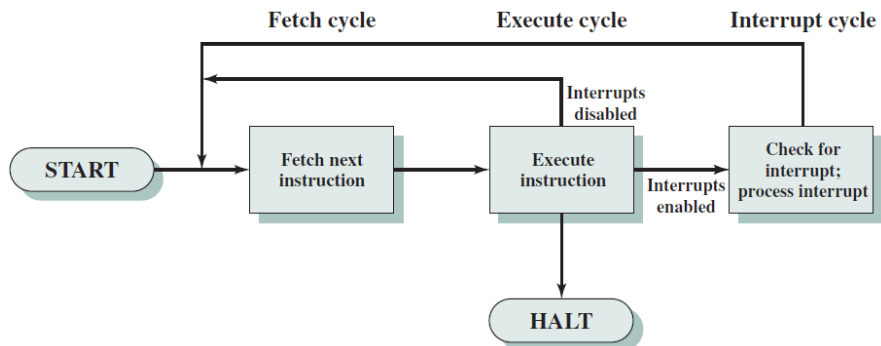
Bilgisayarın Fonksiyonu-Kesmeler

Kesmenin gerçekleştirilmesi



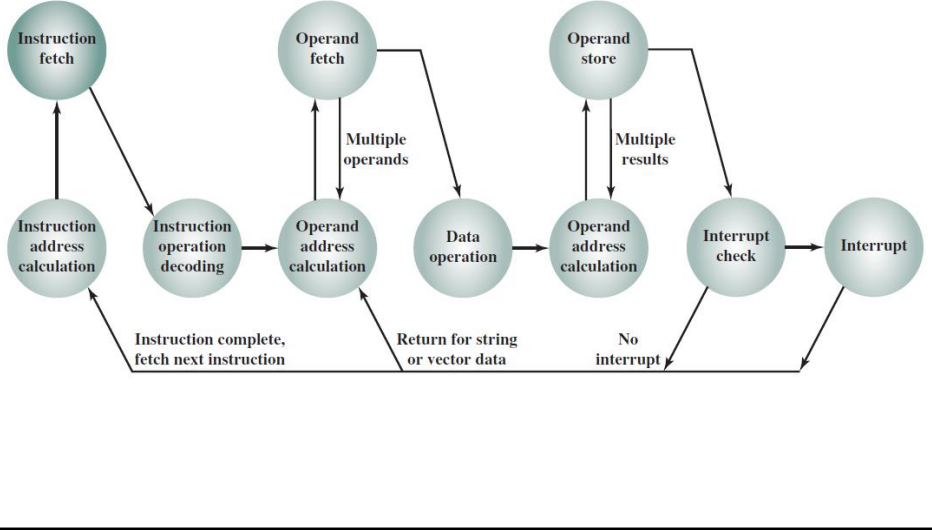
Bilgisayarın Fonksiyonu-Kesmeler

Kesmeyle birlikte komut döngüsü



Bilgisayarın Fonksiyonu-Kesmeler

Kesme ile birlikte komut döngüsü



Bilgisayarın Fonksiyonu-Kesmeler

Çoklu kesmeler

■ Yeni gelen kesmeler etkisiz kılınır:

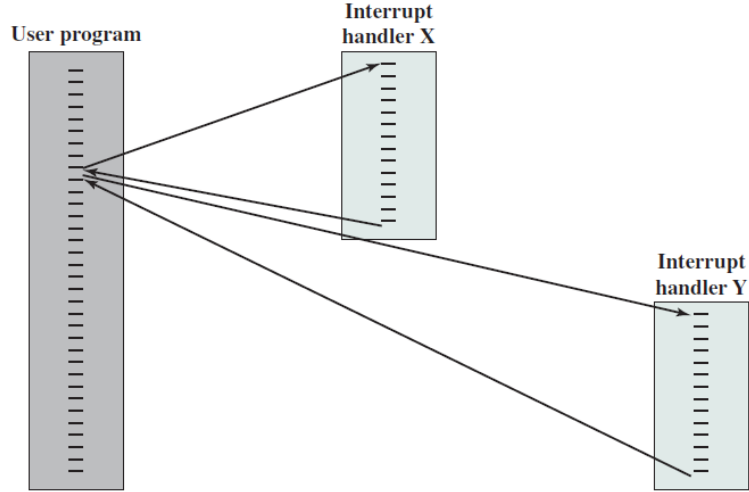
- İşlemci bir kesmeyi çalıştırırken yeni geleni beklemeye alır.
- Önceki kesme bitirildiğinde bekleyen kesmeler oluştuğu sırayla işlenir.

■ Önceliklendirme yapılır:

- Bir kesme çalışırken yeni gelen kesmenin önceliği daha düşükse yeni gelen beklemeye alınır.
- Bir kesme çalışırken yeni gelen daha öncelikli ise çalışan kesme olduğu yerde bırakılır ve yeni gelene geçilir.
- Öncelikli kesme bitirildikten sonra yarıda kesilen kesmeye kaldığı yerden devam edilir.

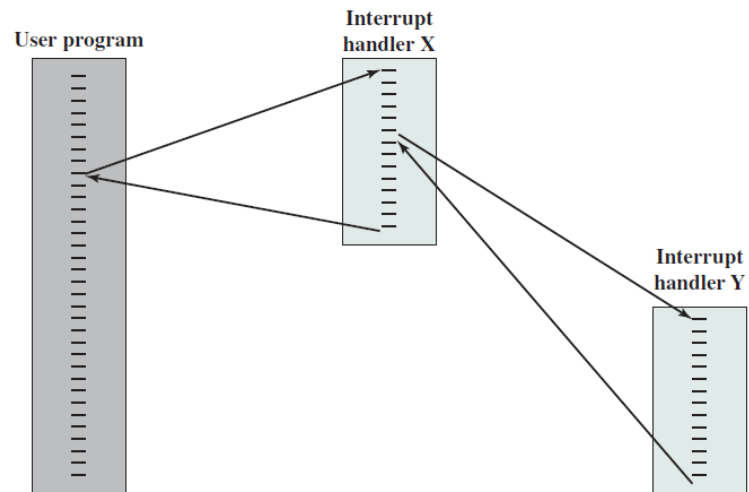
Bilgisayarın Fonksiyonu-Kesmeler

Kesmelerin sıralı çalışması



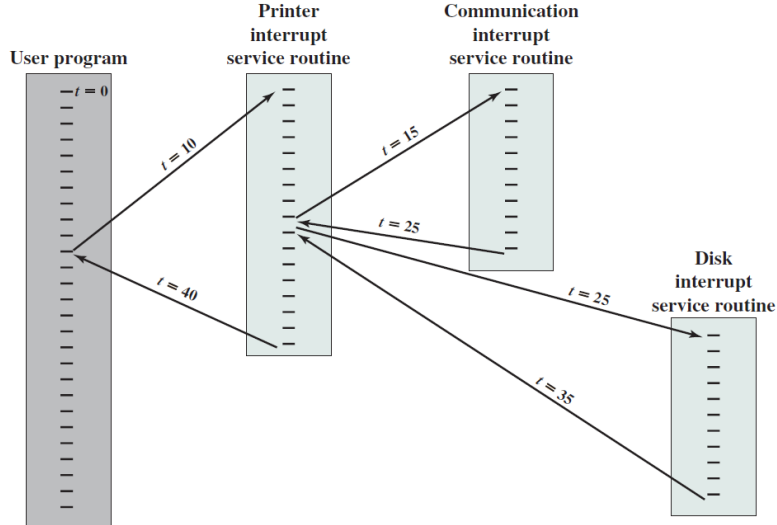
Bilgisayarın Fonksiyonu-Kesmeler

Kesmelerin öncelikli çalışması



Bilgisayarın Fonksiyonu-Kesmeler

Çoklu kesmelerin çalışma sırası



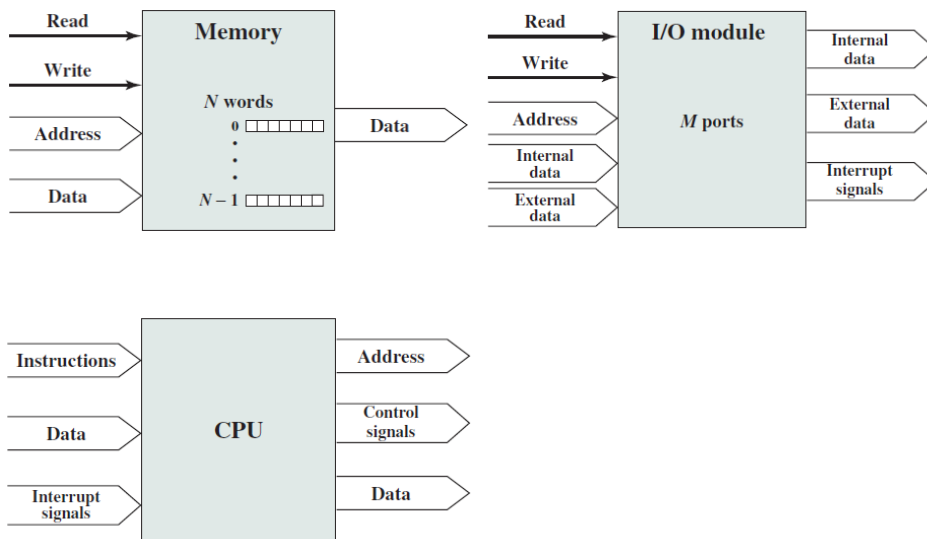
Konular

- Bilgisayar Bileşenleri
- Bilgisayarın Fonksiyonu
 - Instruction Cycle
 - Kesmeler (Interrupt'lar)
- **Bus Yapıları**
 - Bilgisayar Modülleri ve Bağlantıları
 - Bus Tasarım Kriterleri

Bus Yapıları

- Tüm birimler birbiriyle iletişim yapabilmelidir.
 - İletişim farklı verileri aktarmak için kullanılır.
 - Taşınan veri **kontrol işareti**, **data** veya **adres** bilgisi olabilir.
- Farklı birimlerin bağlantıları da farklıdır.

Bus Yapıları- Bilgisayar Modülleri ve Bağlantıları



Bus Yapıları-Hafıza Bağlantıları

- Veri alır ve gönderir.
- Adres alır.
- Kontrol işaretleri alır.
 - Write
 - Read

Bus Yapıları-I/O Bağlantıları

- I/O bağlantıları hafıza bağlantılarına benzer şekildedir.
- Çıkış
 - Çevre birimlerine veri gönderir.
 - Bilgisayara veri gönderir.
- Giriş
 - Çevre birimlerinden veri alır.
 - Bilgisayardan veri alır.
- Bilgisayardan kontrol işaretleri alır.
- Çevre birimlerine kontrol işareti gönderir (disk döndür).
- Bilgisayardan adres alır (Çevre birimleri port numarasıyla ifade edilir).
- Kesme sinyalleri gönderir.

Bus Yapıları-CPU Bağlantıları

- Hafızadan komut ve data okur.
- Veri yazar.
- Diğer birimlere kontrol işaretleri, adres ve data gönderir.
- Kesme isteklerini alır ve gerçekleştirir.

Bus Yapıları

- Bus iki veya daha fazla cihazı bağlar.
- Genellikle **broadcast** şeklinde çalışır.
- Birden fazla bağlantı bir iş için gruplandırılarak kullanılır.
- Örneğin **32 bit bus** için 32 adet tek bitlik bağlantı birlikte kullanılır.
- Genellikle **data bus**, **adres bus** ve **kontrol bus** şeklinde üç grup altında ifade edilir.

Bus Yapıları-Data Bus

- Veri veya komut taşır.
- **Performansı birinci derecede etkiler.**
- **8, 16, 32, 64 bit** genişliğinde bus kullanımı mevcuttur.
- Genişlik arttıkça bir seferde okunan veya yazılan bit sayısı artar, sonuçta performans artmış olur.

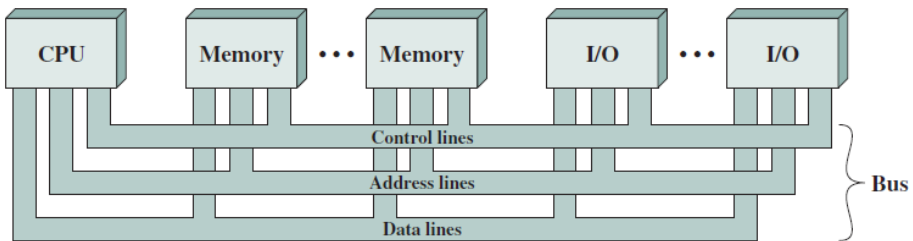
Bus Yapıları-Adres Bus

- **Kaynak** veya **hedef** verinin **adresini tanımlar.**
- CPU, hafızada adresi verilen yerden veri okur veya hafızada adresi verilen yere veri yazar.
- Adres bus genişliği **adreslenebilir alanın boyutunu belirler.**
- Adres **bus genişliği arttıkça** adreslenebilir **hafıza artar.**
- **8086** için $2^{16} = 64K$ adreslenebilir hafıza oluşturulur.

Bus Yapıları-Kontrol Bus

- Kontrol ve zamanlama bilgisini sağlar.
- **Hafızaya okuma** veya **yazma komutunu gönderir.**
- Kesme isteđi iletilir.
- Clock sinyalleri iletilir.

Bus Yapıları-Bađlantı Őeması



Bus Yapıları-Bus Tasarım Kriterleri

- **Tür**
 - Dedicated (Adanmış)
 - Multiplexed (Çoğullanmış)
- **Kullanım yöntemi**
 - Cenraliad (Merkezi)
 - Distributed (Dağıtık)
- **Zamanlama**
 - Senkron
 - Asenkron
- **Bus genişliği**
 - Adres
 - Data
 - Kontrol
- **Veri aktarım türü**
 - Read
 - Write
 - Read-modify-write
 - Read-after-write
 - Block

Bus Yapıları-Bus Tasarım Kriterleri

- **Tür**
 - **Dedicated (Adanmış)**
 - ✓ Her cihaz için ayrı bağlantı yolları kullanılır.
 - ✓ Trafik problemi olmaz.
 - ✓ Cihazlar istediği anda veri gönderebilir.
 - **Multiplexed (Çoğullanmış)**
 - ✓ Paylaşılmış bağlantılar kullanılır.
 - ✓ Adres doğrulaması gerekir.
 - ✓ Daha az bağlantı gerektirir.
 - ✓ Karmaşık kontrol yöntemleri gerekir.

Bus Yapıları-Bus Tasarım Kriterleri

- Kullanım yöntemi
 - **Centralized (Merkezi)**
 - ✓ Bus denetimi bir cihaz tarafından yapılır.
 - ✓ Kontrol daha kolaydır.
 - **Distributed (Dağıtık)**
 - ✓ Her modül bus'a erişebilir.
 - ✓ Her cihazın ayrı denetleyicisi vardır.

Bus Yapıları-Bus Tasarım Kriterleri

- Zamanlama
 - **Senkron**
 - Olaylar clock sinyalle belirlenir.
 - Kontrol bus clock bağlantısına sahiptir.
 - 1-0 veya 0-1 geçişi bir clock cycle'dır.
 - Genellikle senkronizasyon yükselen kenarla yapılır.
 - **Asenkron**
 - Clock kullanılmaz.
 - Her olay bir öncekinin bitmesine bağlı çalışır.
 - Kontrol daha karmaşıktır.
 - Tüm cihazları tam performansta çalıştırabilir.

Bus Yapıları-Bus Tasarım Kriterleri

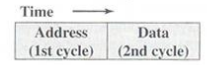
- Bus genişliđi
 - **Adres**
 - ✓ Adres bus genişledikçe adreslenebilir alan artar.
 - **Data**
 - ✓ Data bus genişledikçe bir seferde okunup yazılabilecek veri boyutu artar.
 - **Kontrol**
 - ✓ Kontrol bus genişledikçe kontrol işareti sayısı artar.

Bus Yapıları-Bus Tasarım Kriterleri

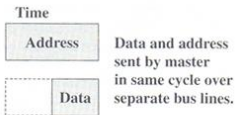
- Veri aktarım türü
 - **Read**
 - ✓ Adres bilgisi gönderilerek ilgili veri okunur.
 - **Write**
 - ✓ Adres bilgisi gönderilerek ilgili veri yazılır.
 - **Read-modify-write**
 - ✓ Adres bilgisi gönderilerek veri okunur ve deđiştirildikten sonra aynı adrese yazılır.
 - **Read-after-write**
 - ✓ Adres bilgisi gönderilerek yazılır ardından kontrol amaçlı okunarak karşılaştırılır.
 - **Block**
 - ✓ Adres bilgisi gönderilerek art arda yazma veya okuma yapılır.

Bus Yapıları-Bus Tasarım Kriterleri

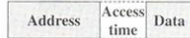
- Veri aktarım türleri



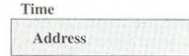
Write (multiplexed) operation



Write (non-multiplexed) operation



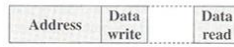
Read (multiplexed) operation



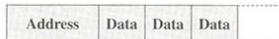
Read (non-multiplexed) operation



Read-modify-write operation



Read-after-write operation



Block data transfer