

BLM312 Mikroişlemciler

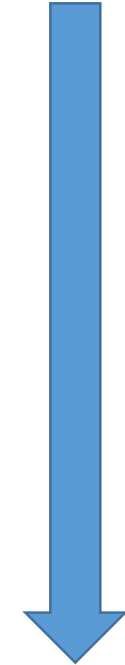
Microprocessor Architecture - Bus Systems

Outline

- İşlemci Mimarisi
 - Programlama İlerlemesi (*Programming Advancement*)
 - Temel İşlemci Mimarisi ve Zamanlama(*Basic Processor Architecture and Timing*)
 - Mimari Hususlar(*Architectural Considerations*)
- Kişisel Bilgisayar Mimarisi (Personal Computer Architecture)
- Bus Mimarisi

Programming Advancement

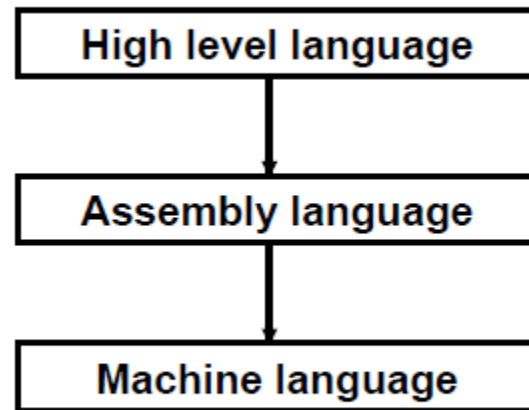
Yaklaşım	Açıklama
Circuit rewiring	Programlamak için devreleri manuel olarak yeniden bağlama
Machine language	İkili kodlar kullanılarak birler ve sıfırlardan oluşturulmuştur
Assembly language	İkili sayılar yerine ADD gibi anımsatıcı (mnemonics) kodların kullanılması
High level language	Fortran, COBOL, BASIC, C, vb.
Scripting language	Perl, Python, Ruby vb.



1. Daha Fazla İlerleme

2. Programcılar için Daha Kolay

Programming Advancement (cont.)



$A = A + B$

Load A

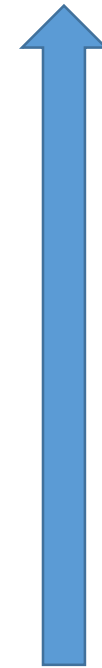
Load B

Add A,A,B

1110 1110 1101 0000

1110 1110 1101 0001

1001 1101 1110 0001



Programlama kolaylığı

Major Categories of Programming Languages

- **Machine Language** (First-generation Language)
 - En temel programlama dilleri seviyesi. İkili kodlu komutları (emirleri) kullanır.
- **Assembly Language** (Second-generation Language)
 - Bunlar bazen sembolik diller olarak adlandırılır, çünkü semboller işlem kodlarını ve bellek lokasyonlarını temsil etmek için kullanılır.

Major Categories of Programming Languages (cont.)

Assembly Language:

1. Makine dili komutları için anımsatıcılara ve bellek konumları için sembolik adlara izin veren sembolik bir kod

Örnek:

"1100 000" kullanmak yerine, aynı anlama gelen ADD sembolünü yazıyoruz.

2. Komutların, belirli bir bilgisayara ait tekil makine komutlarına yakın bir şekilde karşılık geldiği bir programlama dilidir. **Assembly dili, makine diline benzer şekilde, makineye bağlıdır (*machine dependent*).**

Major Categories of Programming Languages (cont.)

3. Assembly dili, farklı adresleme modlarını temsil etmek için özel bir notasyon kullanırken, işlem kodlarını temsil etmek için semboller kullanır.
4. Assembly kodunu makine koduna çevirmek için bir «*translator*» kullanılır. Bu translator'a **assembler** denir.

High-Level Languages(Third-generation Languages)

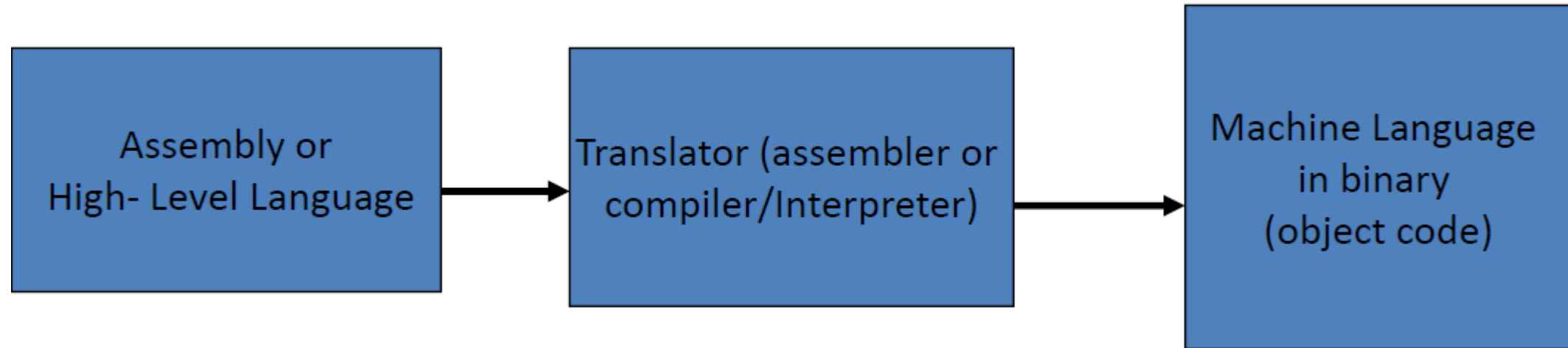
- İnsan diline veya matematiğin standart gösterimine çok benzeyen ifadeler adı verilen komutları kullanır

Fourth-generation Languages:

- Daha çok yordamsal olmayan ve etkileşimli olan çok çeşitli programlama dilleri.

Programcılar istedikleri sonucu belirtir, bilgisayar ise bu sonuçları verecek komutlar dizisine karar verir. **Örnek: SQL**

Major Categories of Programming Languages (cont.)

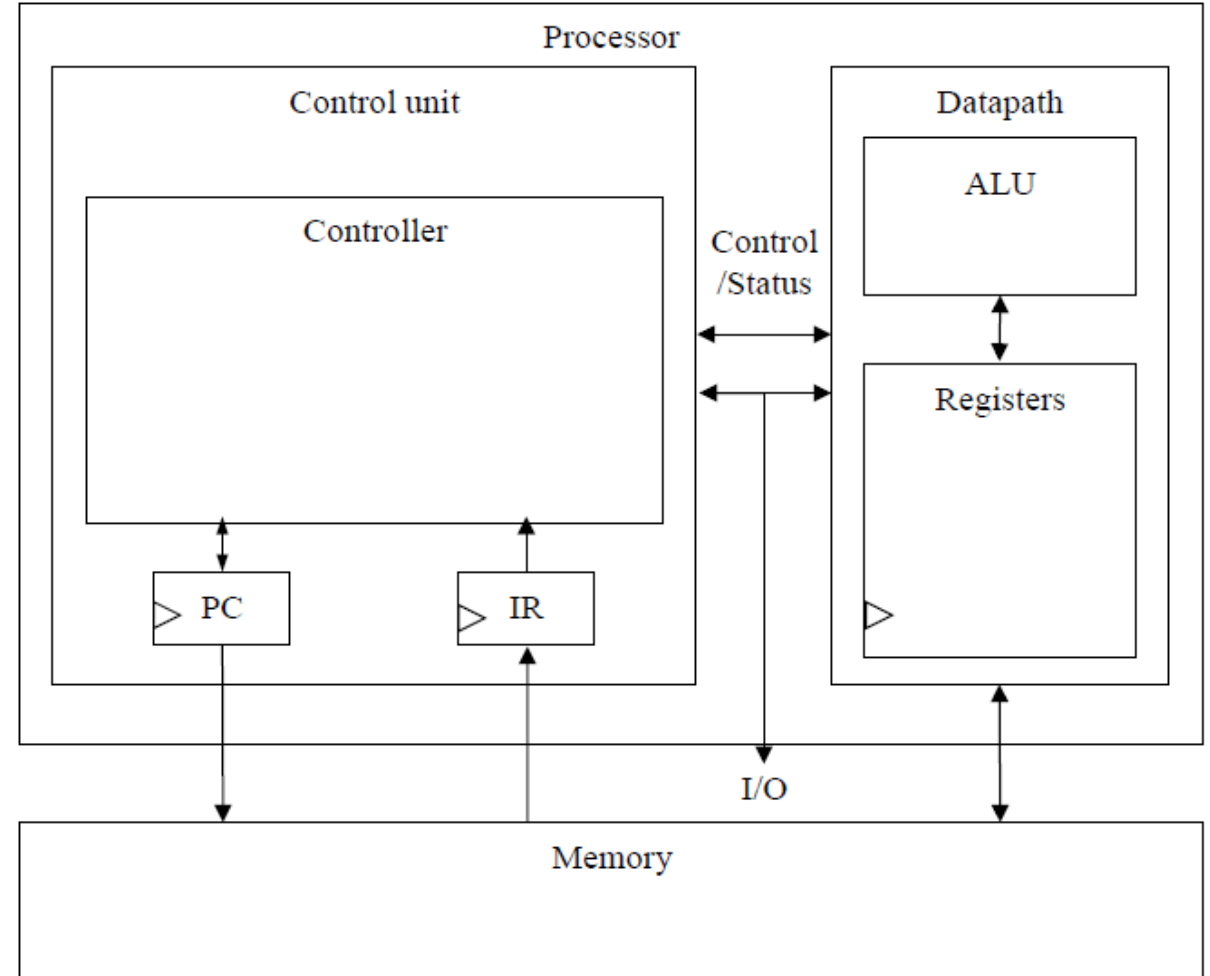


Outline

- İşlemci Mimarisi
 - Programlama İlerlemesi (*Programming Advancement*)
 - Temel İşlemci Mimarisi ve Zamanlama(*Basic Processor Architecture and Timing*)
 - Mimari Hususler(*Architectural Considerations*)
- Kişisel Bilgisayar Mimarisi (Personal Computer Architecture)
- Bus Mimarisi

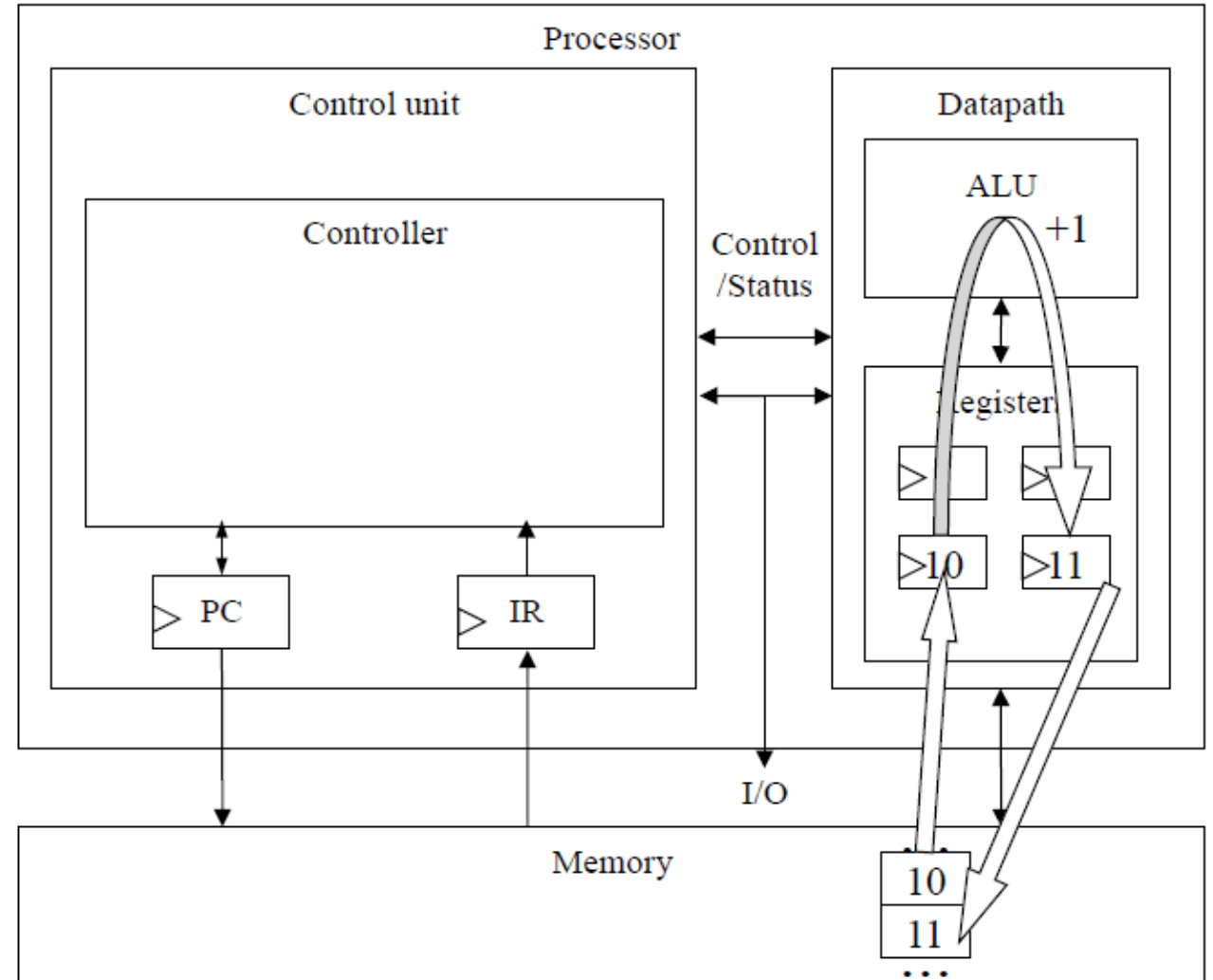
Basic Processor Architecture and Timing

- İşlemci içindeki bileşenler
 - Control Unit
 - Datapath Unit
- Programı ve verileri saklamak için bellek
- Çevre birimlerle iletişim kurmak için I / O



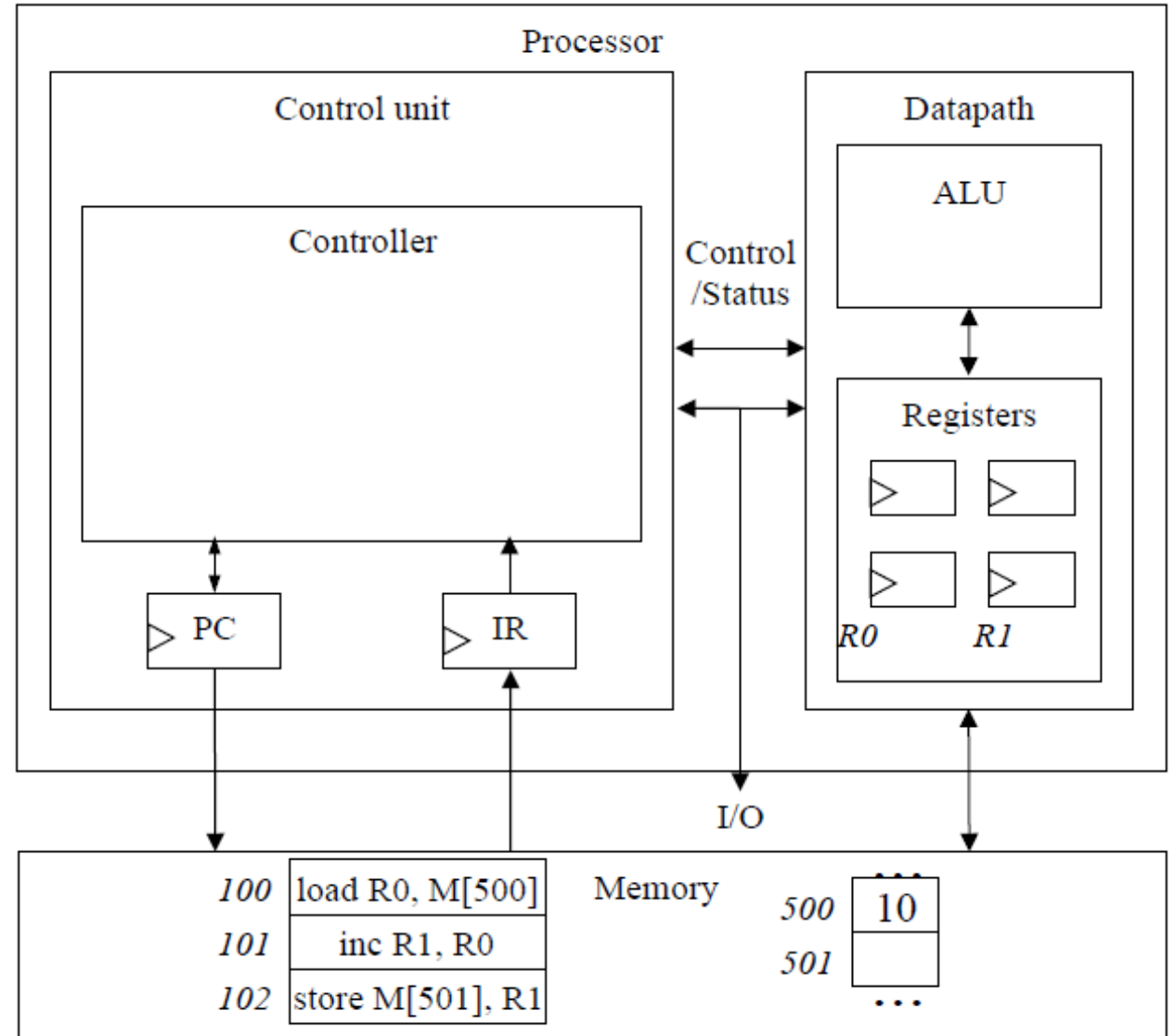
Datapath Operation

- Load
 - Bellek hücresinden kaydediciye veri okur
- ALU operation
 - ALU aracılığıyla belirli kaydedicilerden girdi alır, kaydedicilere geri depolar
- Store
 - Kaydedicileri bellek hücrelerine yazar



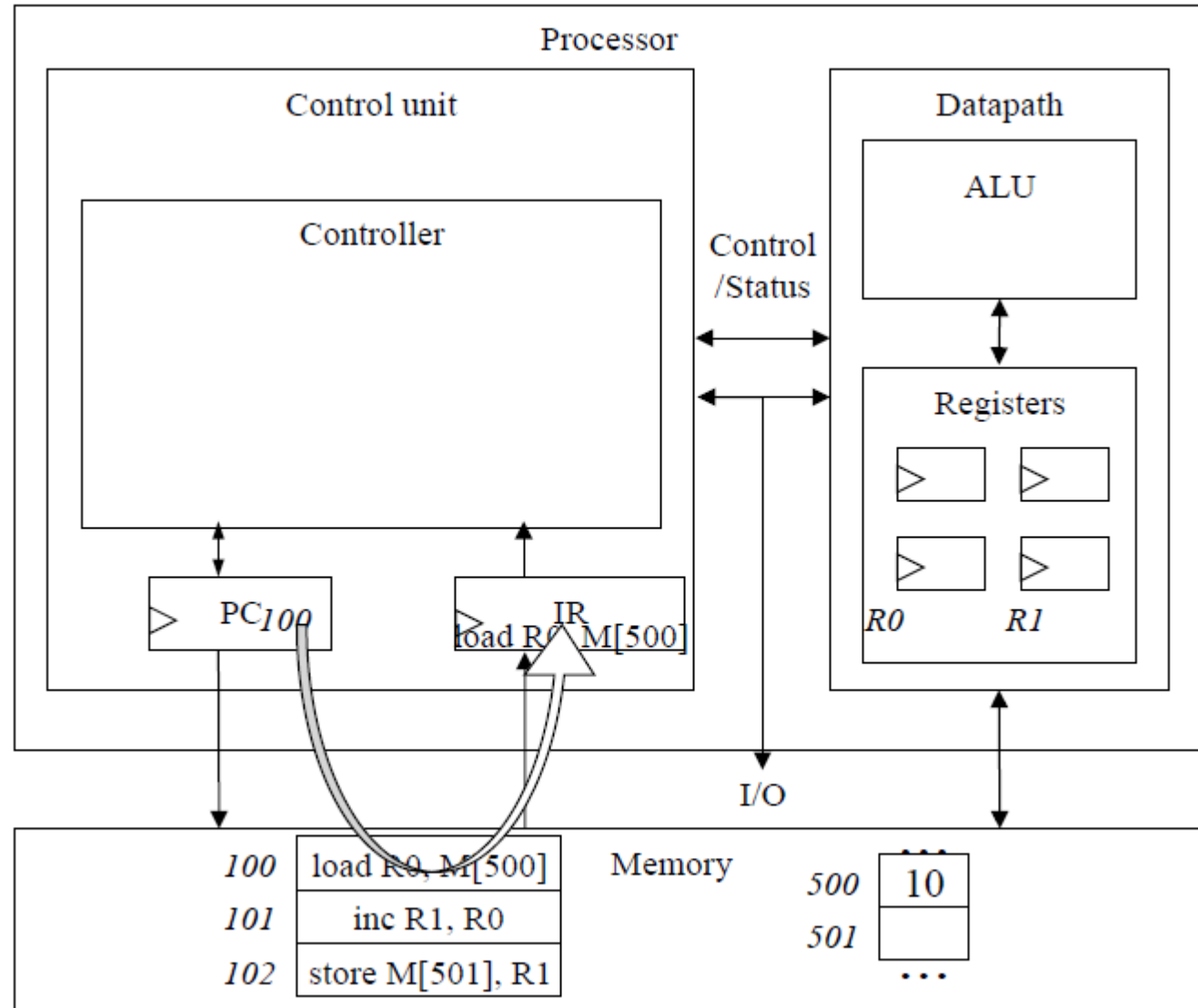
Control Unit Operation

- Control unit: veri yolu işlemlerini yapılandırır
 - Bellekte saklanan icrası istenen işlemler dizisi ("komutlar") - "program"
- Instruction cycle – her biri bir saat (clock) döngüsü olmak üzere birkaç alt işleme ayrılmıştır, örneğin:
 - **Fetch:** Sıradaki komutu IR'ye alır
 - **Decode:** Komutun ne anlama geldiği belirlenir
 - **Fetch operands:** Verileri bellekten veri yolu kaydedicilerine taşır
 - **Execute:** Veriler ALU'da işlenir
 - **Store results:** Kaydedicilerdeki veriler belleğe yazılır



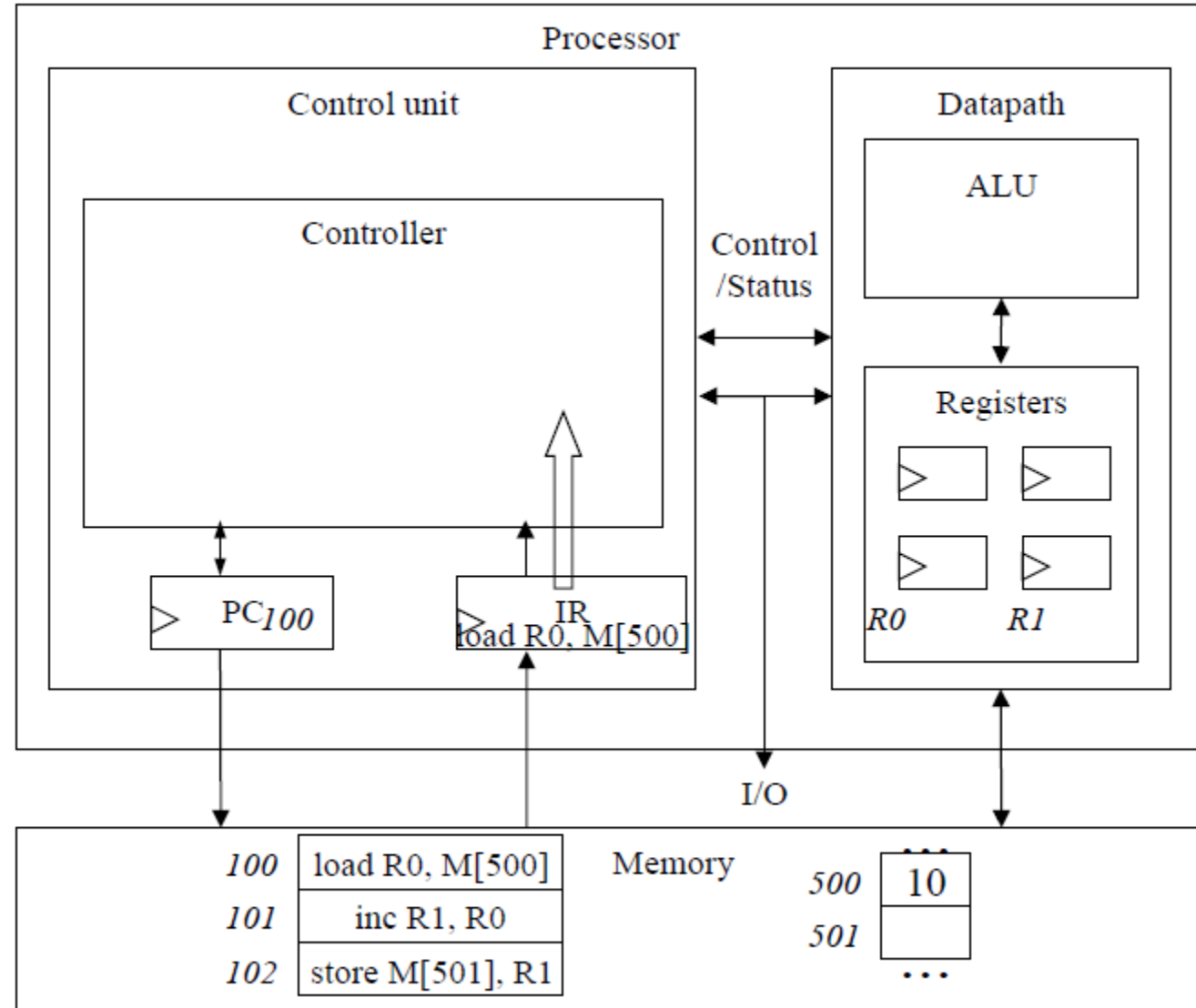
Control Unit Sub-Operations

- Fetch
 - IR'ye bir sonraki komutu al
 - PC: program counter, her zaman sonraki komutu gösterir
 - IR: getirilen komutları tutar



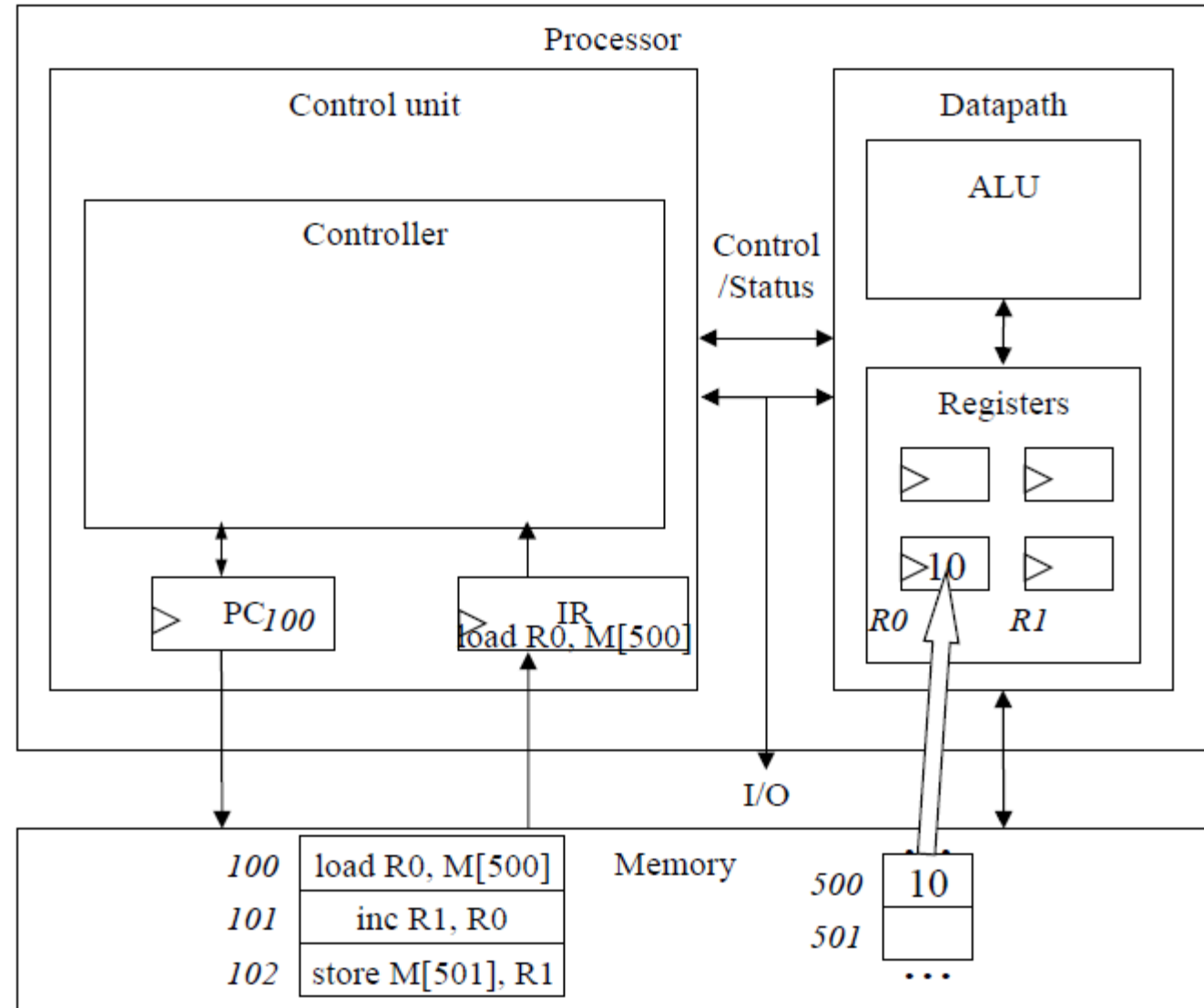
Control Unit Sub-Operations

- Decode
 - Komutun ne anlama geldiği belirlenir



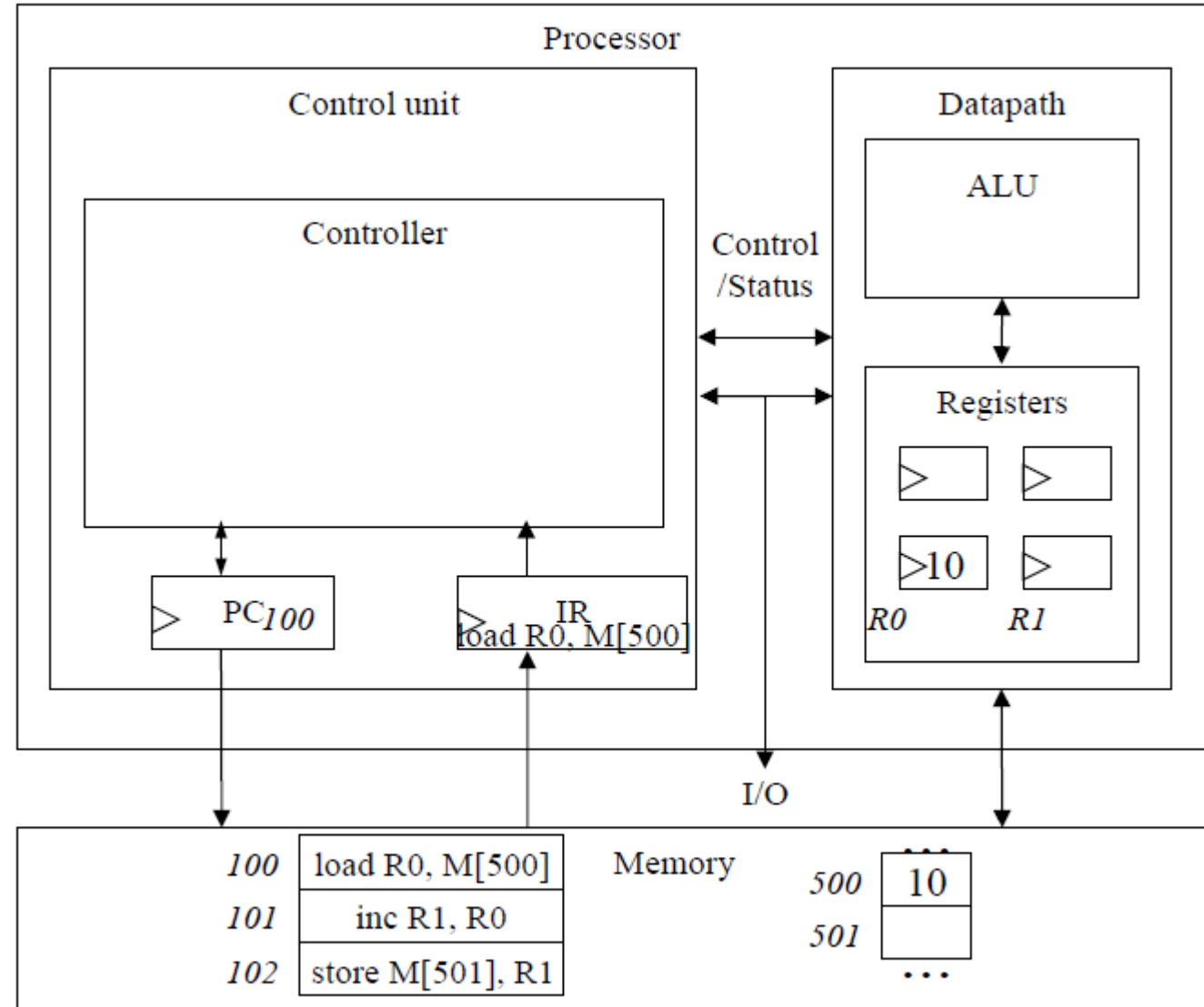
Control Unit Sub-Operations

- Fetch operands
 - verileri bellekten veri yolu kaydedicilerine taşır



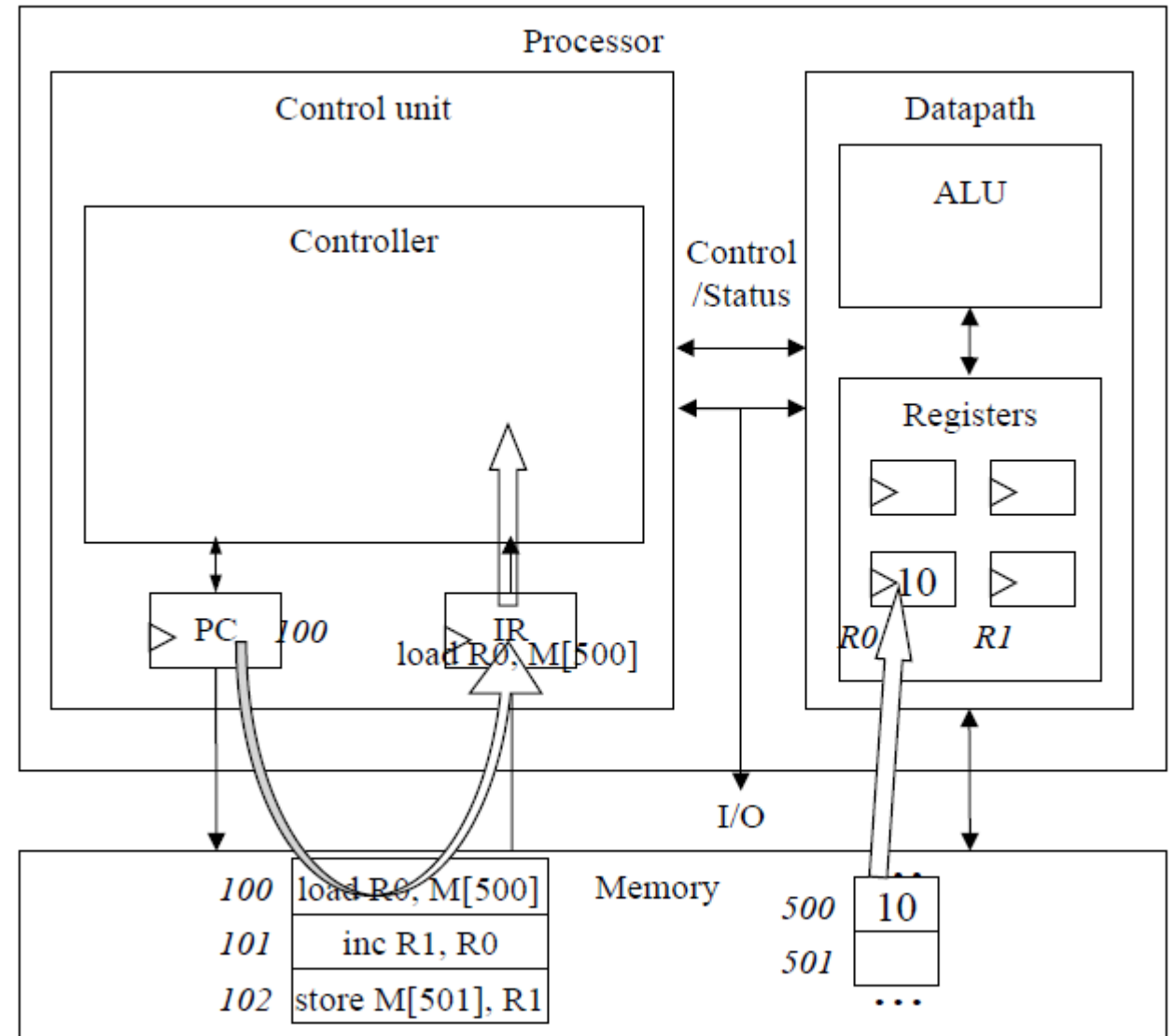
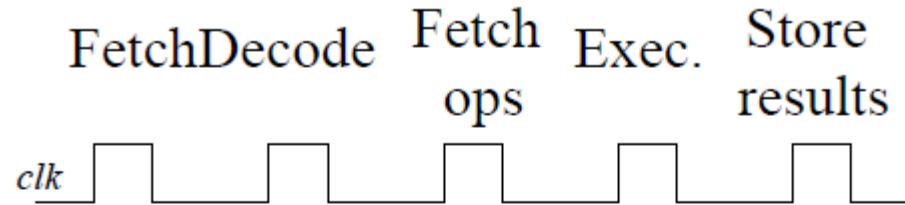
Control Unit Sub-Operations

- Execute
 - Veriler ALU'dan geçirilir.
- Store results
 - Kaydedicilerdeki veriler belleğe yazılır



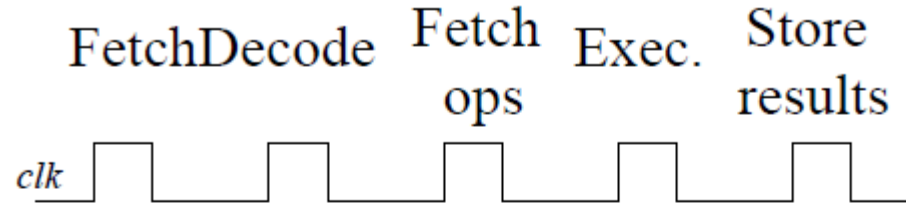
Instruction Cycles

PC=100

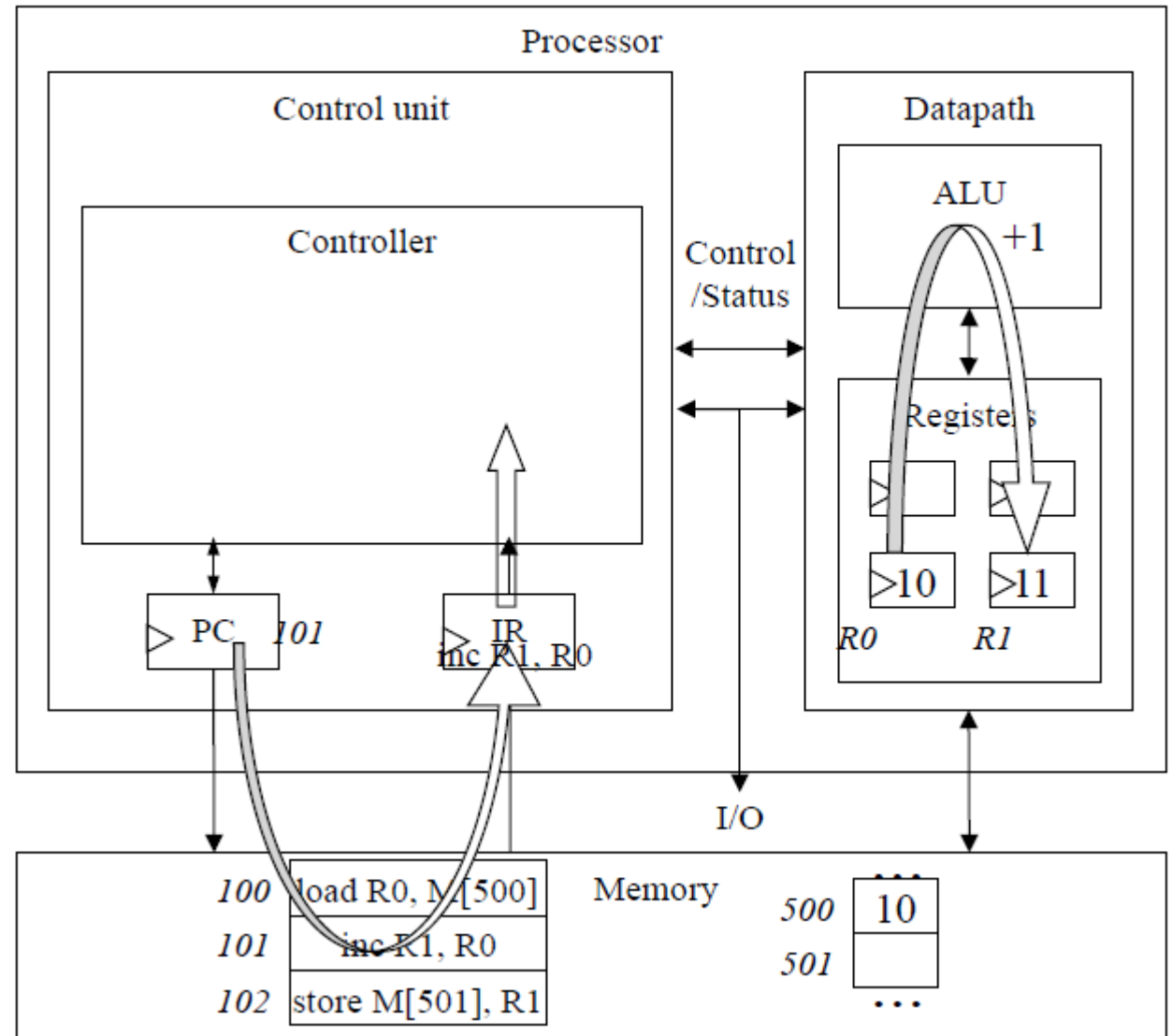
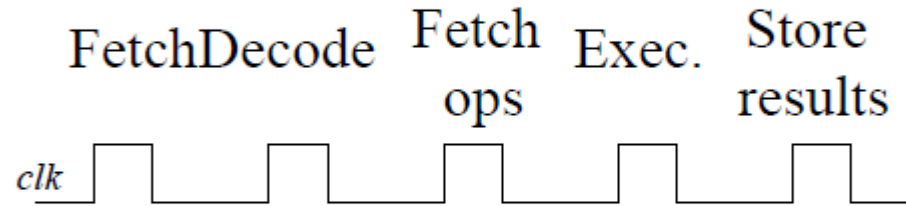


Instruction Cycles

PC=100



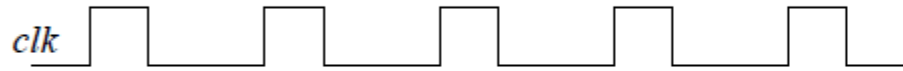
PC=101



Instruction Cycles

PC=100

FetchDecode Fetch ops Exec. Store results



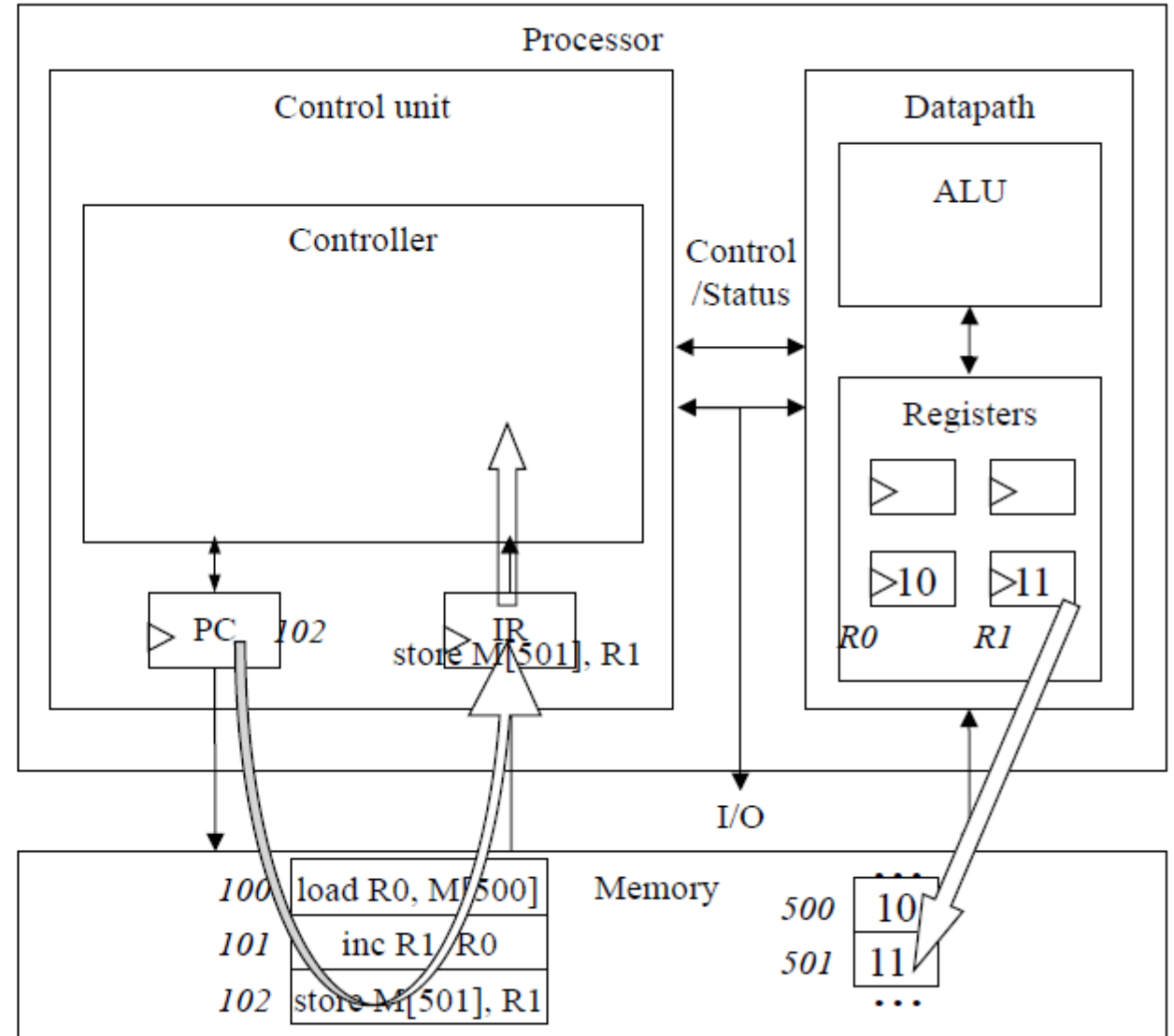
PC=101

FetchDecode Fetch ops Exec. Store results



PC=102

FetchDecode Fetch ops Exec. Store results

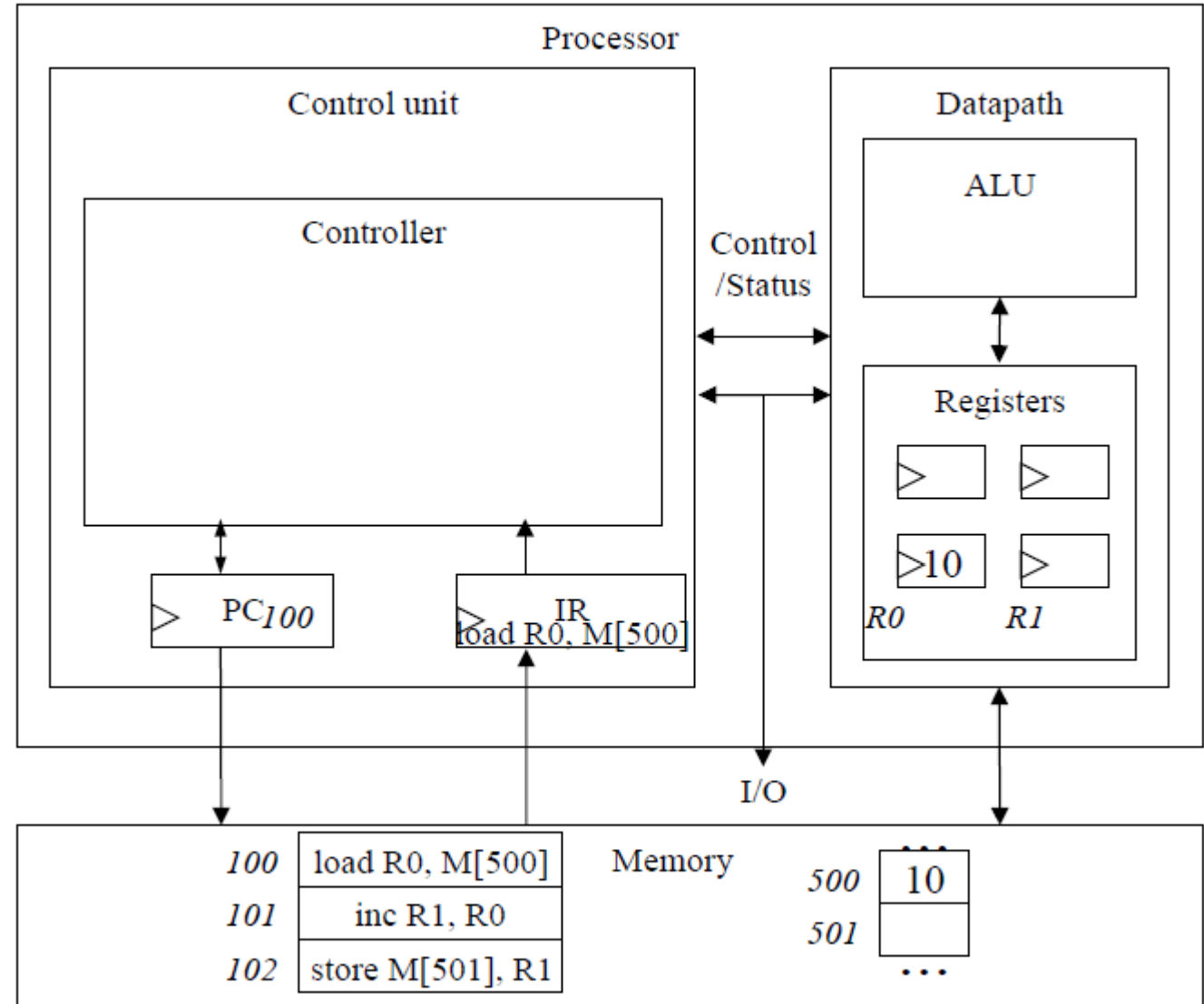


Outline

- İşlemci Mimarisi
 - Programlama İlerlemesi (*Programming Advancement*)
 - Temel İşlemci Mimarisi ve Zamanlama(*Basic Processor Architecture and Timing*)
 - Mimari Hususlar(*Architectural Considerations*)
- Kişisel Bilgisayar Mimarisi (Personal Computer Architecture)
- Bus Mimarisi

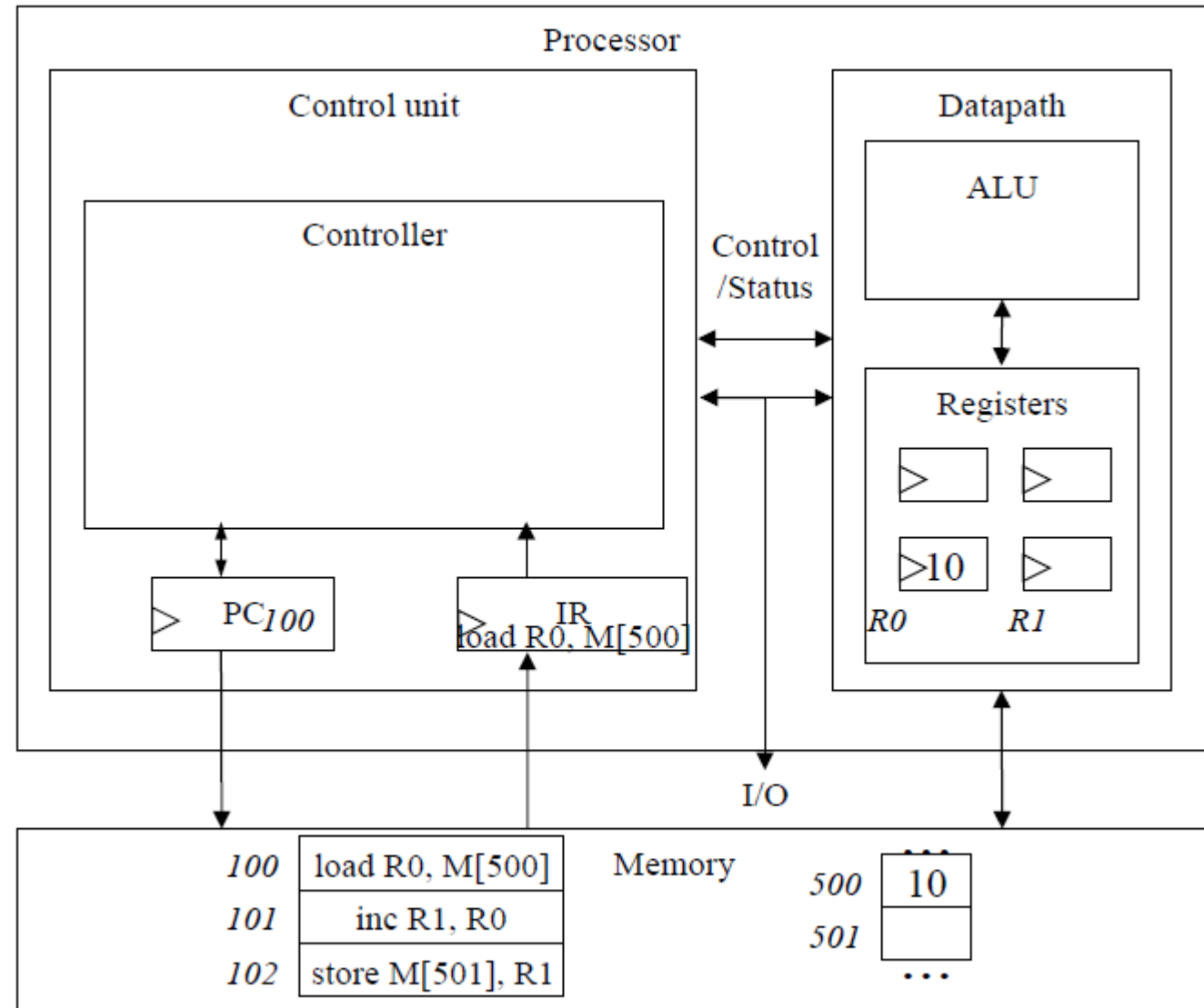
Architectural Considerations

- *N-bit* processor
 - N-bit ALU, registers, buses, memory data interface
 - Embedded: 8-bit, 16-bit, 32-bit common
 - Desktop/servers: 32-bit, even 64
- PC (*Program counter*) boyutu adres uzayını (*address space*) belirler



Architectural Considerations

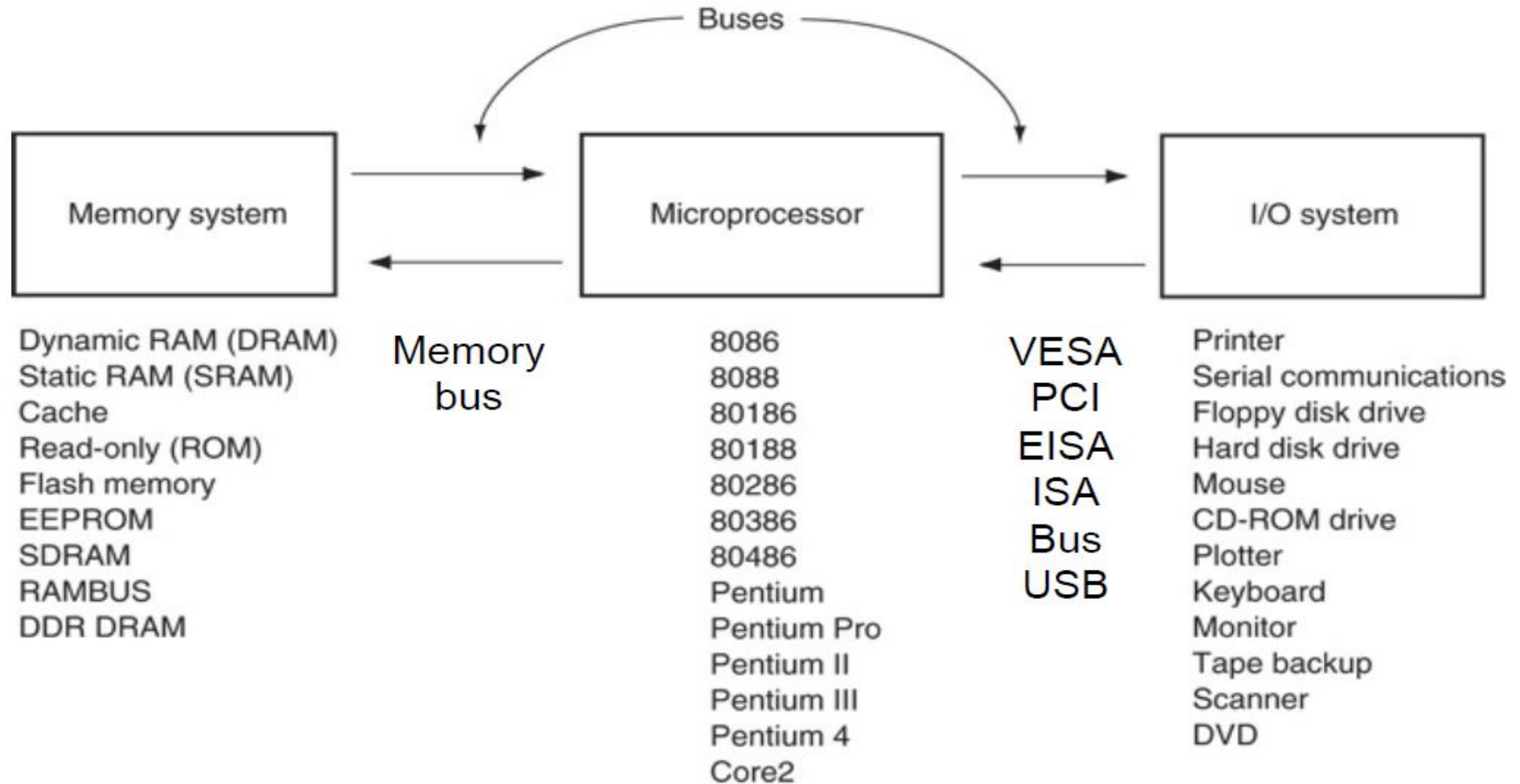
- Clock frequency
 - Saat periyodunun tersi



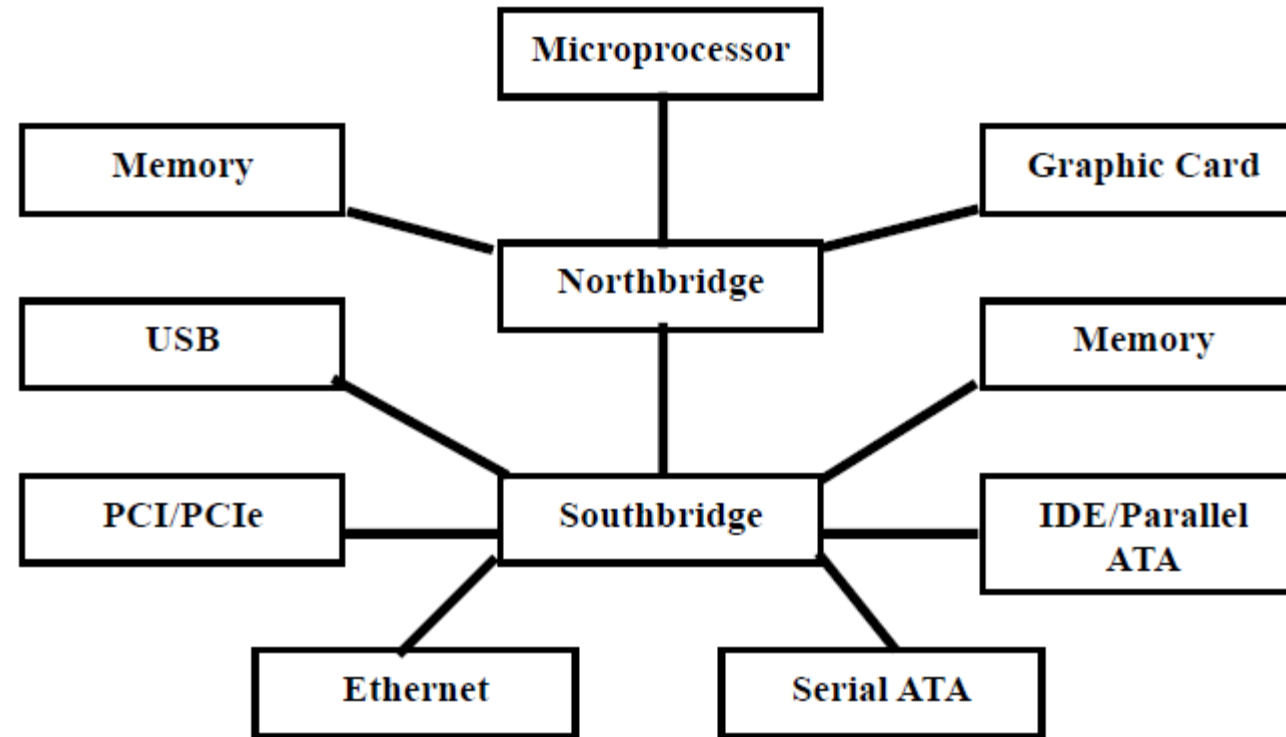
Outline

- İşlemci Mimarisi
 - Programlama İlerlemesi (*Programming Advancement*)
 - Temel İşlemci Mimarisi ve Zamanlama(*Basic Processor Architecture and Timing*)
 - Mimari Hususler(*Architectural Considerations*)
- Kişisel Bilgisayar Mimarisi (Personal Computer Architecture)
- Bus Mimarisi

Basic Architecture: Personal Computer



Basic Architecture: Personal Computer (cont.)



Outline

- İşlemci Mimarisi
 - Programlama İlerlemesi (*Programming Advancement*)
 - Temel İşlemci Mimarisi ve Zamanlama(*Basic Processor Architecture and Timing*)
 - Mimari Hususler(*Architectural Considerations*)
- Kişisel Bilgisayar Mimarisi (Personal Computer Architecture)
- Bus Mimarisi

What is inside Computer?

- Tıpkı sokaktaki bir otobüsünün insanları bir yerden bir yere taşıması gibi, bir bilgisayarın içindeki veri yolu (**bus**) da bilgileri bir yerden bir yere taşır.
- Bilgisayarın bileşenleri **bus** ile bağlanır.
- En basit biçimdeki bir veri yolu (**bus**), CPU ile bellek ve CPU ile I / O arasında elektrik sinyalleri biçiminde bilgi taşımak için kullanılan bir dizi teldir.
- Örnek: VAX, 32 bit adres yoluna ve 32 bit veri yoluna sahiptir.

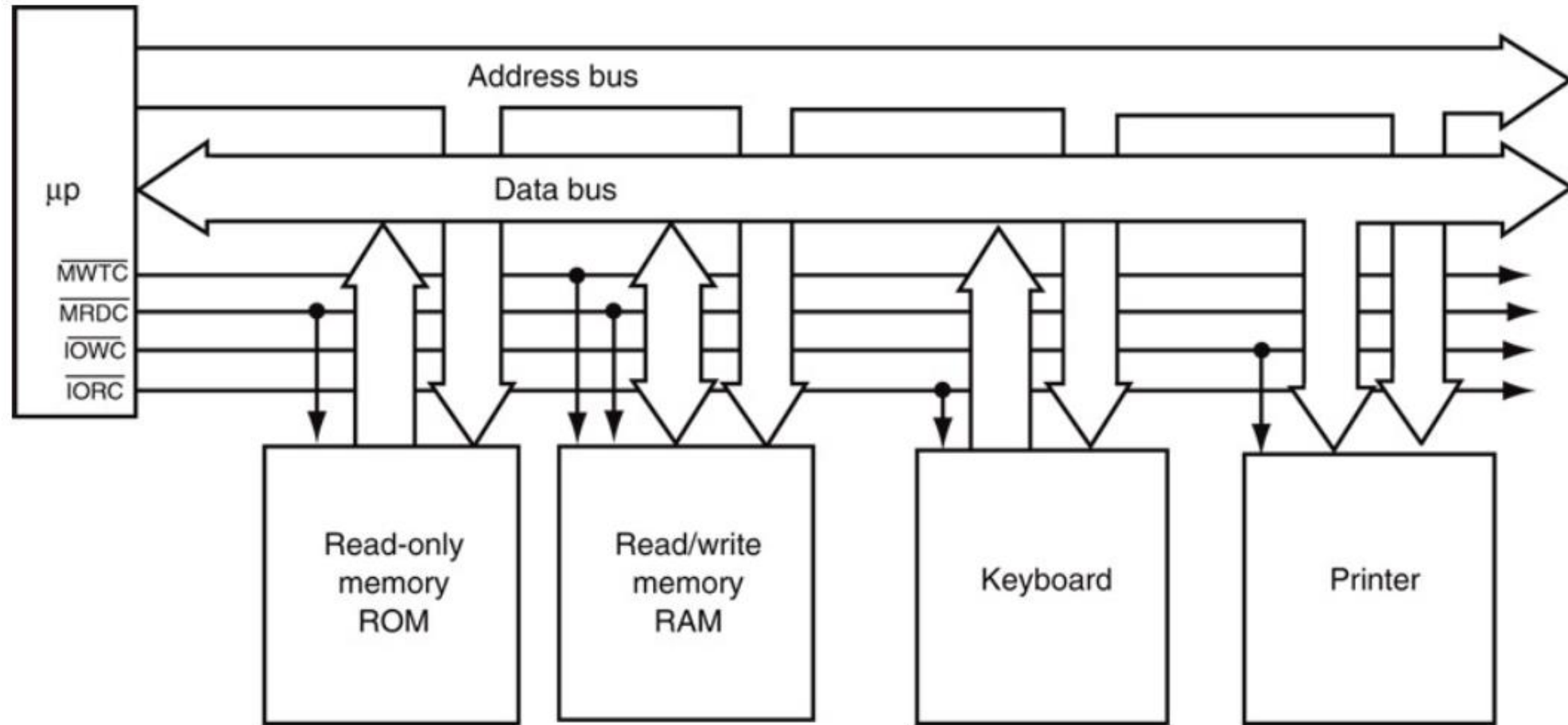
What is inside Computer.

Her bilgisayarda üç tür **bus** vardır:

1. Address bus
2. Data bus
3. Control (Signal) bus

- CPU tarafından tanınan bir cihaz (bellek veya I/ O) için, bir adres atanmalıdır. Cihaza atanan adres benzersiz olmalıdır; cihazların aynı adrese sahip olmasına izin verilmez.
- CPU adresi **address bus**'a koyar ve dekode devresi cihazı bulur. Daha sonra da CPU **data bus**'ı kullanarak cihaza veri gönderir veya alır.
- **Control bus** cihaza (**CPU nun veri alma veya gönderme isteklerini belirten**) okuma/yazma sinyallerini sağlamak için kullanılır.

Basic Architecture: Bus



***Note that Pentium system bus architecture is more complex than this.**

More about data bus:

- Data bus, CPU dan içeri/dışarı veri taşımak için kullanıldığından bus'lar ne kadar müsait ise CPU o kadar iyidir.
- Daha geniş **data bus**, daha pahalı CPU ve bilgisayar anlamına gelir.
- CPU'daki data bus'ların ortalama genişliği 8 ile 64 bit arasında değişir.
- Data bus iki yönlüdür (**bi-directional**). CPU veri gönderir ve alır.
- Bilgisayarın işleme gücü, veri yollarının genişliğiyle ilgilidir. Çünkü 8-bit genişliğindeki bir veri yolu (bus) aynı anda 1 byte gönderebilir. Fakat 16-bit genişliğindeki bir veri yolu (bus) aynı anda 2 byte gönderebilir. 2 kat daha hızlı.

More about data bus:

- iki tür veri yolu (data bus) vardır:

Internal data bus (back side bus): CPU ve önbellek (*cache*) arasındaki veri yolu

External data bus (front side bus): önbellek (cache) ve bellek arasındaki veri yolu

Not: backside ve frontside bus her zaman farklıdır. Birincisi CPU hızında çalışır. Diğeri ise bellek ve I/O gibi daha düşük hızlardaki cihazların ihtiyaçlarını karşılamak için tasarlanmıştır.

Basic Architecture: Bus (cont.)

- Address:
 - Eğer I/O ise, 0000H ile FFFFH arasında bir değer verilir
 - Eğer bellek ise mimariye göre değişir
 - 20-bits (8086/8088)
 - 24-bits (80286/80386SX)
 - 25-bits (80386SL/SLC/EX)
 - 32-bits (80386DX/80486/Pentium)
 - 36-bits (Pentium Pro/II/III)
- Data:
 - 8-bits (8088)
 - 16-bits (8086/80286/80386SX/SL/SLC/EX)
 - 32-bits (80386DX/80486/Pentium)
 - 64-bits (Pentium/Pro/II/III)
- Control:
 - Çoğu sistemin en az 4 kontrol veriyolu (control bus) bağlantısı vardır (aktif düşük).
 - MRDC (Memory Read Control), MWRC, IORC (I/O Read Control), IOWC

Basic Architecture: Bus Standards

ISA (Industry Standard Architecture): 8 MHz

- 8-bit (8086/8088)
- 16-bit (80286-Pentium)

• EISA: 8 MHz

- 32-bit (older 386 and 486 machines).

• PCI (Peripheral Component Interconnect): 33 MHz

- 32-bit or 64-bit (Pentiums)
- New: PCI Express and PCI-X 533 MTS

• VESA (Video Electronic Standards Association): 25–40 MHz

- 32-bit or 64-bit (Pentiums)
- VESA Yerel Veri Yolu, video (grafik) işlemlerini hızlandırmayı amaçlayan standartlaştırılmış bir yüksek hızlı kanal sağlamak
- Yalnızca disk ve video. PCI ile rekabet edebilir fakat popüler değil

Basic Architecture: Bus Standards

- USB (Universal Serial Bus): 1.5 Mbps, 12 Mbps, 480 Mbps and now 5Gbps.
 - Newest systems.
 - Serial connection to microprocessor.
 - For keyboards, the mouse, modems and sound cards.
 - To reduce system cost through fewer wires.
- AGP (Advanced Graphics Port): 66MHz
 - Newest systems.
 - Fast parallel connection: Across 64-bits for 533MB/sec.
 - For video cards.
 - To accommodate the new DVD (Digital Versatile Disk) players.
 - Latest AGP 3.0 with peak bandwidth of 2.1GB/s.

Case Study: Inside your iPhone...

