



Bilgisayar Mühendisliğine Giriş
Bilgisayar Temel Kavramlar

Dr. Cahit Karakuş
Esenyurt Üniversitesi

Today's Computer & Software Engineering

- 1- Quantum Computer (Quantum Physics) and Quantum Computing (Mathematics, Probability, Statistical analysis)
- 2- Artificial Intelligence, Data Science, Database management, Data preparation
- 3- Artificial Intelligence, Machine learning Algorithms
- 4- Automation / Autonomous: Machines with computer systems (Computer Organization, Microprocessor, Assembly, Intelligence Algorithms)
- 5- Algorithm and Mathematical models
- 6- Software Languages: Python, Java Script, C++, Matlab, Assembly

Tanımlar

- İşlemci veriyi işlerken ana bellek (RAM-ROM) ve işlenecek komutların içeriğinin yazıldığı geçici saklayıcı (register) adlı devrelerle birlikte çalışır.
- **RAM:** Okunup yazılan bellektir, elektrik kesildiğinde içerik kaybolur.
- **ROM:** Salt okunur bellektir. Elektrik kesildiğinde içeriği kaybolmaz. Önceden bilgisayar mühendisleri tarafından içeriği hazırlanır ve yüklenir.
- **Data width (veri genişliği):** 8,16,32,64, 128 bit olarak karşımıza çıkar.
- Örneğin veri genişliği 128 bit demek; 1 clock periyodunda 128 bit (16byte) veri yazmak ya da okumak demektir. Clock: Darbe katarı.
- **Byte:** 8 bit veriyi temsil eder ya da 8 bitlik temel bellek birimini işaret eder.
- **MIPS (millions of instructions per second):** saniyede işlenen işlem sayısını ifade etmektedir ve CPU performansını ölçmek için kullanılan bir birimdir.
- **Assembler:** doğrudan doğruya CPU'nun yönetildiği yazılımdır, makine dilidir.

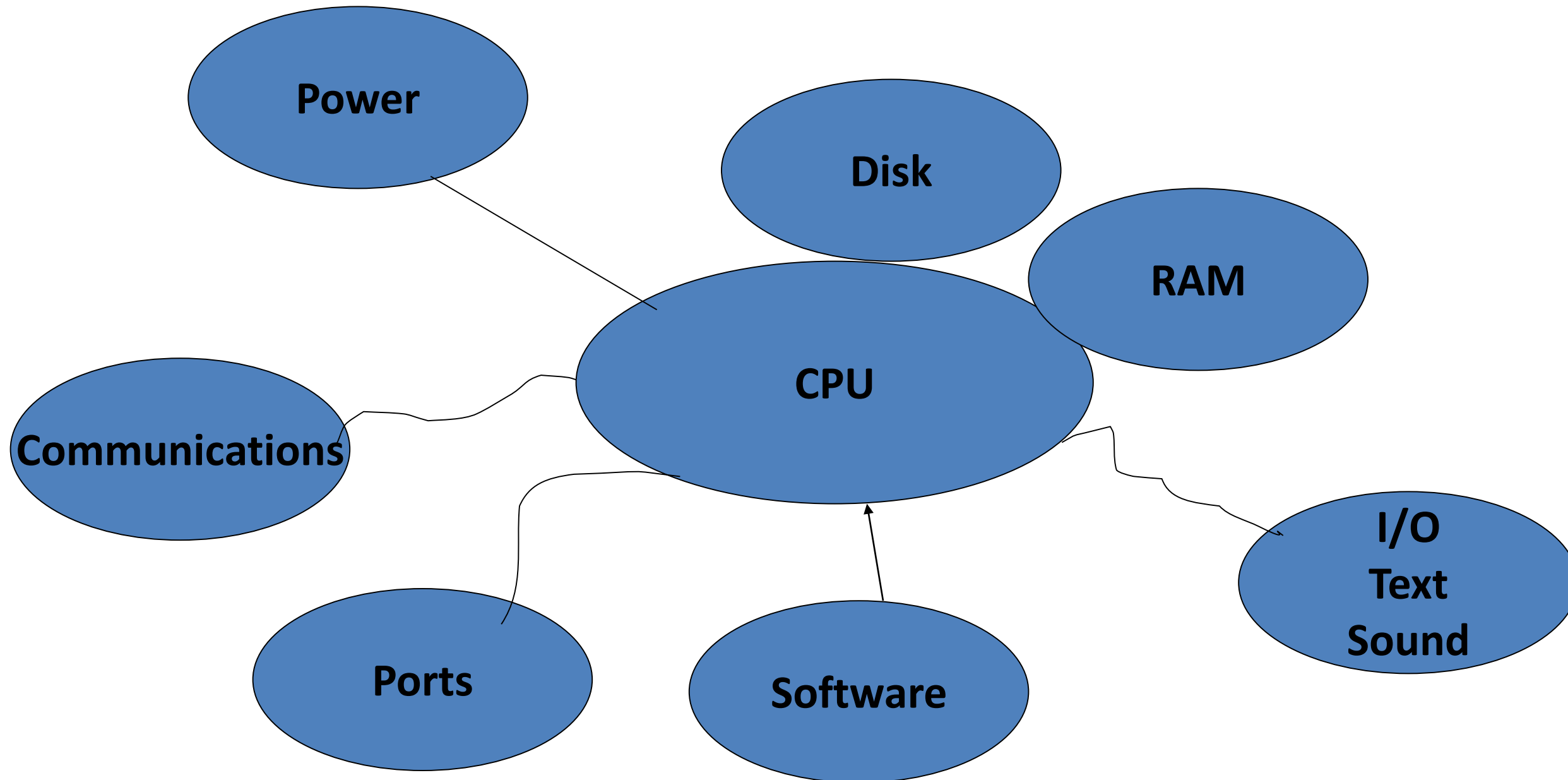
What Is A Computer?

- Bilgisayar veri işleyebilen bir yapıdır.
- Bilgisayar, bir dizi aritmetik veya mantıksal işlemi bağımsız olarak gerçekleştirmek üzere programlanmış genel amaçlı bir cihazdır.
- Yakın gelecekte binlerce işlemci, verileri çok hızlı bir şekilde işlemek için birlikte çalışırken kuantizasyon hesaplamasını kullanacak. Bitler elektronlar veya fotonlarla temsil edilecektir.
- Bilgi Sistemi: Veriyi alan, saklayan, işleyen ve bilgiyi çıktı olarak sağlayan sistemdir.

What is a computer?

- a computer is a sophisticated electronic calculating machine that:
 - **Accepts** input information,
 - **Processes** the information according to a list of internally stored instructions and
 - **Produces** the resulting output information.
- Functions performed by a computer are:
 - **Accepting** information to be processed as **input**.
 - **Storing** a list of **instructions** to process the information.
 - **Processing** the **information** according to the list of instructions.
 - **Providing** the results of the processing as **output**.
- What are the functional units of a computer?

What's in a Computer?



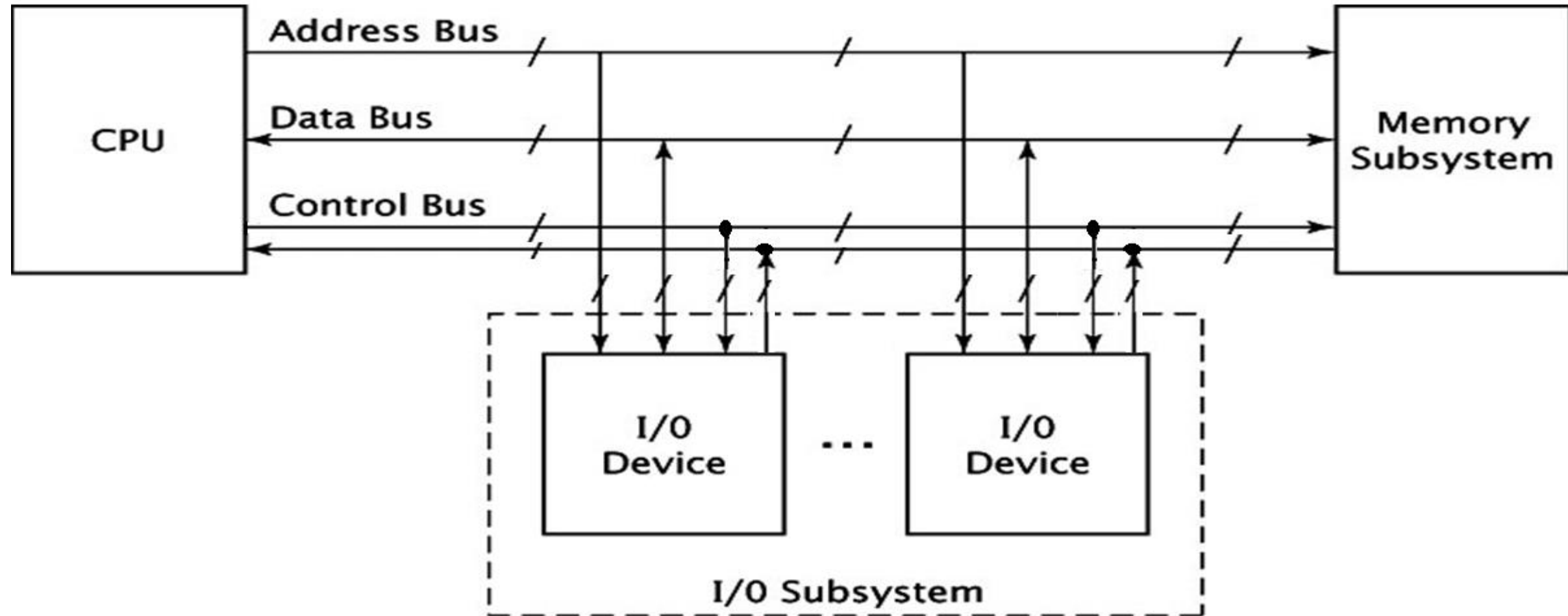
Bilgisayar - Mikroişlemci

- **Bilgisayar:** Girdi olarak verileri ikili sayı (bit: 0/1) sisteminde alan, verileri depolayan, işleyen ve kullanışlı bir biçimde çıktı verisi sağlayan programlanabilir bir makinedir.
 - Input: Data
 - Instructions: Software, Programs
 - Output: Information (numbers, words, sounds, images)
- **Mikroişlemci:** Bellekten komutları alan, kod çözen, yürüten, bellek ya da I/O birimleri için çıkış üreten program kontrollü yarı iletken cihazdır (Semiconductor Device, IC: Integrated Circuit). Bilgisayarlarda CPU (Central Processing Unit: Merkezi İşlem Birimi) olarak kullanılır.

Bilgisayar Organizasyonu

- Bilgisayar Organizasyonu: Bir bilgisayar sistemini tasarlamak için bilgisayar sisteminin bilinmesi gereken detayları içerir. İşlemci, bellek ve giriş / çıkış devrelerinin tasarımını içeren bir bilgisayarın sisteminin düzenlemesidir.
- Bilgisayar sistemini oluşturan temel bileşenler: Mikroişlemci, Bellekler, I/O birimleri, Clock & Timing ve Sistem Bus (Birimler arasında veri haberleşmesinin yapıldığı hatlardır).
- Eğer tüm bileşenler ve yazılım tek bir entegre denilen elektronik devre içinde ise bu tür bilgisayar sistemlerine gömülü sistemler denir.
 - Sürücüsüz araba tasarlarken nasıl bir bilgisayar sistemine ihtiyacım var?
 - Mikroişlemci ne olmalı?
 - Bellek ve boyutları ne olmalı?
 - I/O birimleri nelerdir?
 - Bellekler ile mikroişlemci veri haberleşmesi hatlarının sayısı ne olmalı?

Basic Computer Organization



CPU Internal Structure

Internal structure and basic operation of a microprocessor

- Arithmetic and logic unit (ALU),
- Control unit,
- Clock and Timing,
- Register sets,
 - accumulator,
 - condition code register,
 - program counter,
 - stack pointer,
 - segment registers
 - **Flags**

İşlemcinin organizasyonunu anlamak

○ Fetching and execution cycles and Pipelining

- Komut Getirme/Götürme talimatı: İşlemci bellekten (Register, önbellek, ana bellek) bir komutu alıp getirir ya da götürür.
- Komutun yorumlanması: Hangi eylemin gerekli olduğunu belirlemek için komutun kodu çözülür.
- Verileri getir: Bir komutun yürütülmesi aşamasında, bellekten veya bir G/Ç modülünden veri okunmasını gerektirebilir.
- İşlem verileri: Bir komutun yürütülmesi, veriler üzerinde bazı aritmetik veya mantıksal işlemlerin gerçekleştirilmesini gerektirebilir.
- Veri yazma: Bir komutun sonuçları olarak veriler belleğe veya bir G/Ç modülüne yazılması gerekebilir.
- Bunları yapabilmek için işlemcinin bazı verileri geçici olarak saklaması gerektiği açık olmalıdır. Bir sonraki komutun nereden alınacağını bilmesi için son komutun yerini hatırlaması gerekir. Bir komut yürütülürken talimatları ve verileri geçici olarak saklaması gerekir. Yani işlemcinin küçük bir dahili belleğe (önbellek) ihtiyacı vardır.

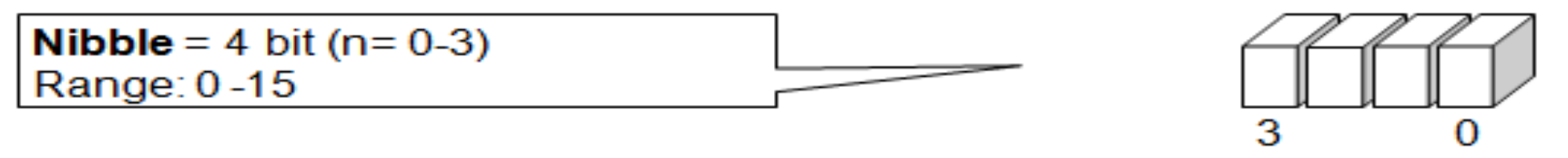

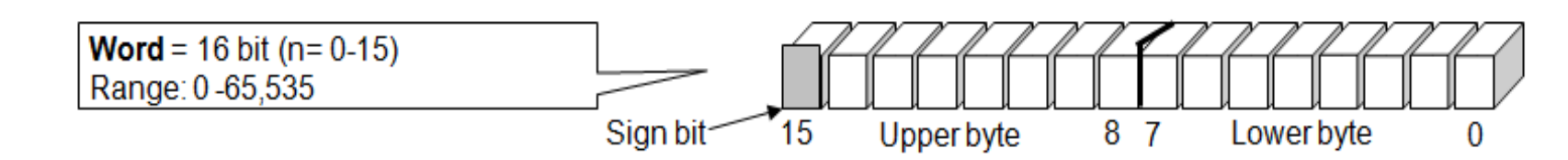
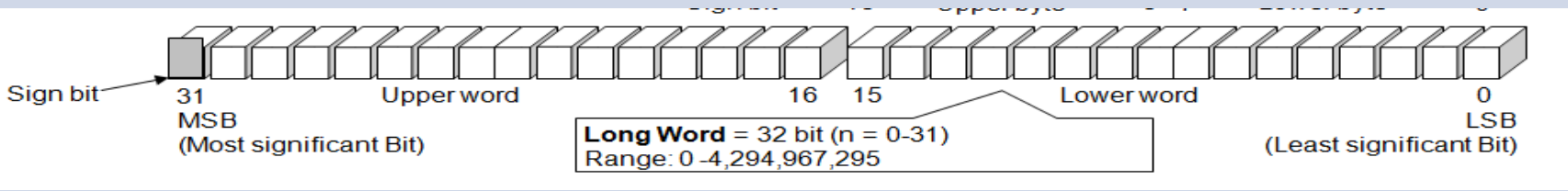
Instruction cycle

- Instruction are fetched and executed by the control unit one by one. The sequences involved for the fetch of one instruction and its execution are known as instruction cycle.
- In all computer languages, expressions consist of two types of components: operands and operators. Operands are the objects that are manipulated and operators are the symbols that represent specific actions. For example, in the expression
- $5 + x$
- x and 5 are operands and $+$ is an operator. All expressions have at least one operand.

Evolution of Data Lines

- Number of data lines: 4, 8, 16, 32, 64, 128 bit
- Number of address lines: 20 bit to 50 bit
- Nibble (4 bit), byte (8bit), word(16bit), longword(32bit)

DATA SIZE

<p>Nibble</p>	<p>4 bit</p>	<p>Nibble = 4 bit (n= 0-3) Range: 0 -15</p> 
<p>Byte</p>	<p>8 bit</p>	<p>Byte = 8 bit (n = 0-7) Range: 0 -255</p> 
<p>Word</p>	<p>16 bit</p>	<p>Word = 16 bit (n= 0-15) Range: 0 -65,535</p> 
<p>Long word</p>	<p>32 bit</p>	<p>Long Word = 32 bit (n = 0-31) Range: 0 -4,294,967,295</p> 

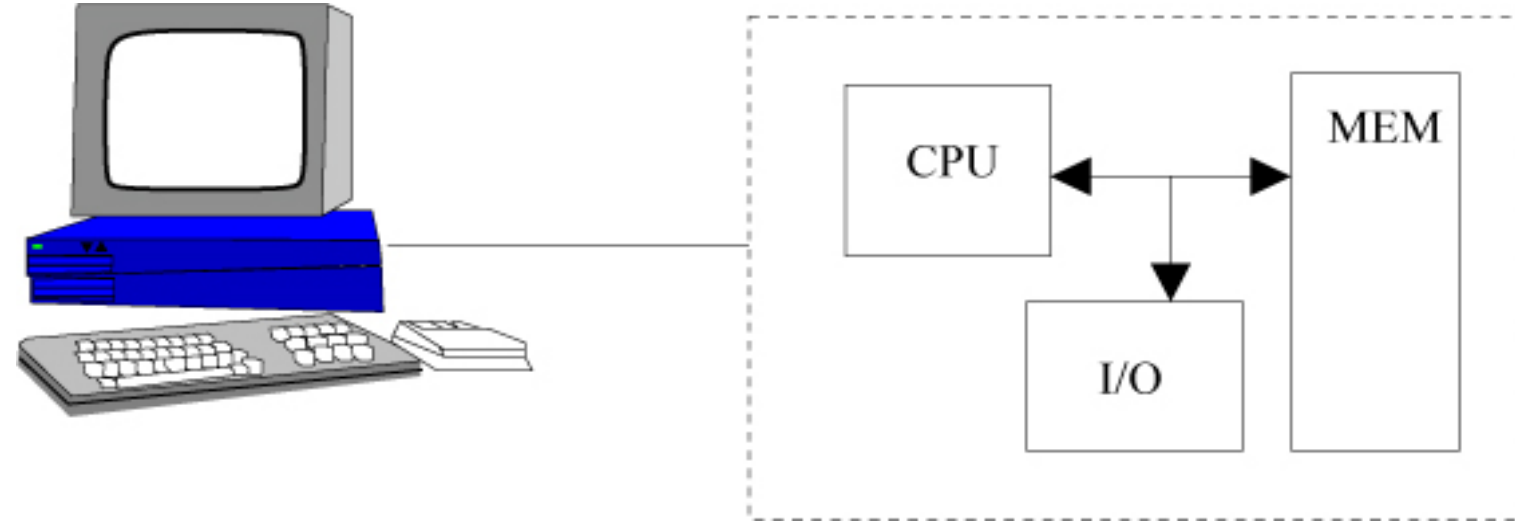


Basic Components of the Computer

System Components

- **CPU/Microprocessor**
- **Memory Subsystem**
- **I/O Subsystem**
- **System Bus**
- **Clock & Timing**

Bilgisayar Temel Bileşenleri



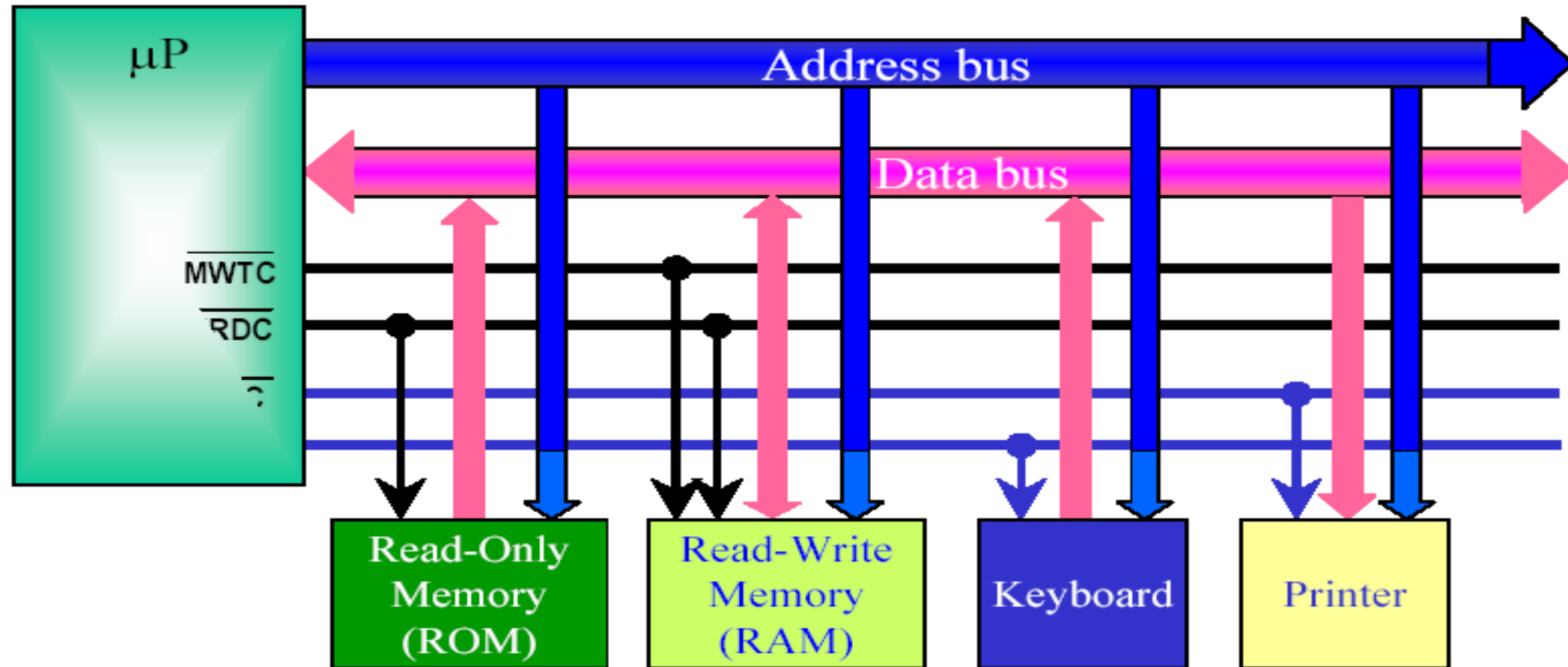
Bilgisayar Sisteminin Temel Bileşenleri:

- CPU – Central Processing Unit (Mekezi İşlemci Birimi - Mikroişlemci)
- Memory Subsystem (Main Memories): Yazılıp okunacak verileri ve programları saklar.
- Input and output (I/O) system
- System Bus: data, address and control.
- Clock and Timing (Microprocessor clock system)
-

Bir Bilgisayar Sisteminin İşlevleri:

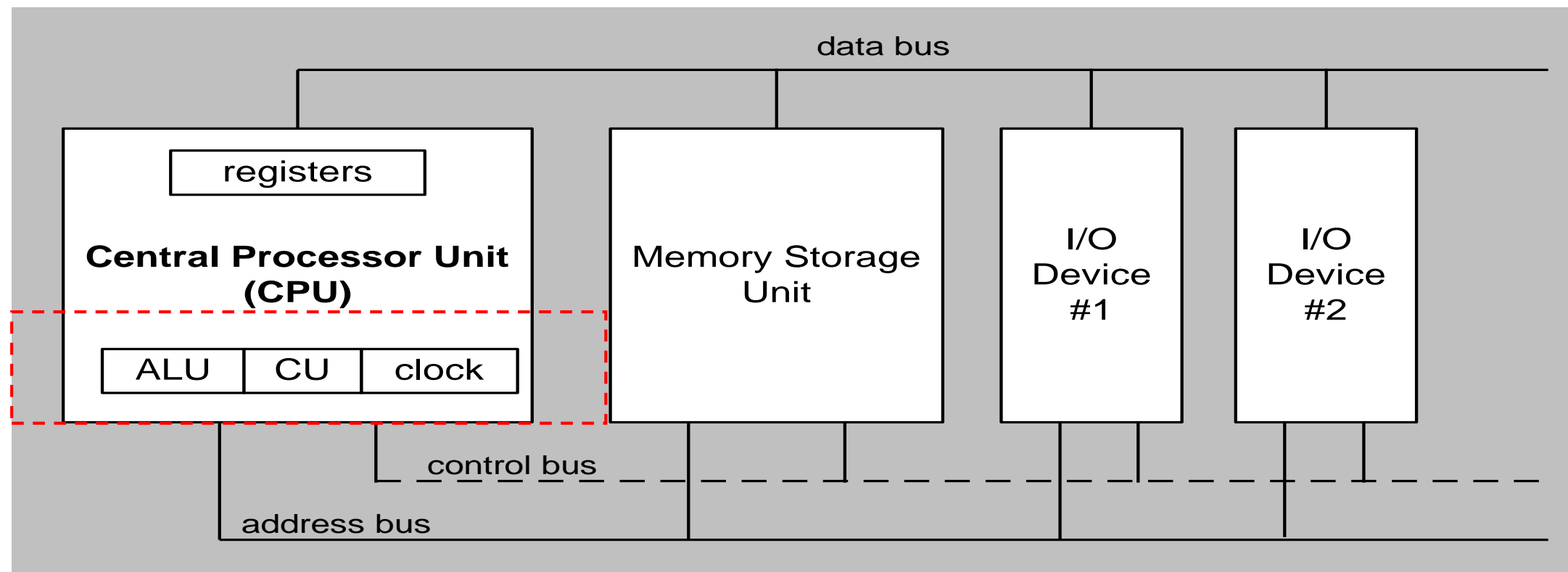
- İşlemci Ana İşlev Döngüsü (Fetching and execution cycles)
- Address Decoding: Bellek gözlerinin çakışmasını ya da üst üste gelmesini önlemek amacıyla bellek seçer.
- Clock and Timing Signals: Verilerin senkronize (eş zamanlı) işlenmesini sağlar.
- Pipelining – Verimi artırmak amacıyla, komutlar dizisinin ardışık düzenlendiği ve eşzamanlı yürütüldüğü işlevlerdir.

Block Diagram of a Microprocessor-based computer system



Components of CPU

- Registers
- **control unit (CU)**: controls operation timing / sequencing. control unit (CU) coordinates sequence of execution steps
- **arithmetic logic unit (ALU)**: performs all arithmetic, bitwise & logic operations, receives command sequence from CU
- **clock**: synchronizes CPU operation
- **System Bus**: Address, Data, Control



A Computer System

○ Blok diyagramları kullanan bir bilgisayar sisteminin temel bileşenleri

- CPU – Central Processing Unit (Mekezi İşlemci Birimi - Mikroişlemci)
- Memory
- Input and output unit
- System Bus: data bus, address bus and control bus.

○ Evolution of microprocessor address lines: 8,16,32 dan 64 bit; Adres hatları bellek gözü veya I/O birimi seçmede kullanılır. Paralel hatlardır. Grup olarak çalışırlar. Üzerlerinde bitleri (1 / 0) temsil eden elektrik sinyalleri taşınır.

- Toplam kapasite= 2^n byte olarak hesaplanır. Buradaki n ifadesi adres hattı sayısıdır.
- Örneğin adres hattı sayısı, $n=30$ ise bilgisayar sistemin adresleme kapasitesi= 2^{30} byte =1 GByte'dır.

○ Data lines: Nibble, byte, word ... Longword. Bellekler ve I/O birimleri ile veri yazma ve okumada kullanılır. Paralel hatlardır. Grup olarak çalışırlar. Üzerlerinde 1 ya da 0'ı temsil eden elektrik sinyalleri taşınır.

- Data bus hat sayısı taşınacak veri kapasitesini byte cinsinden belirler. Örneğin data bus hat sayısı 64 bit ise, data bus kapasitesi= $64/8=8$ byte'dır.

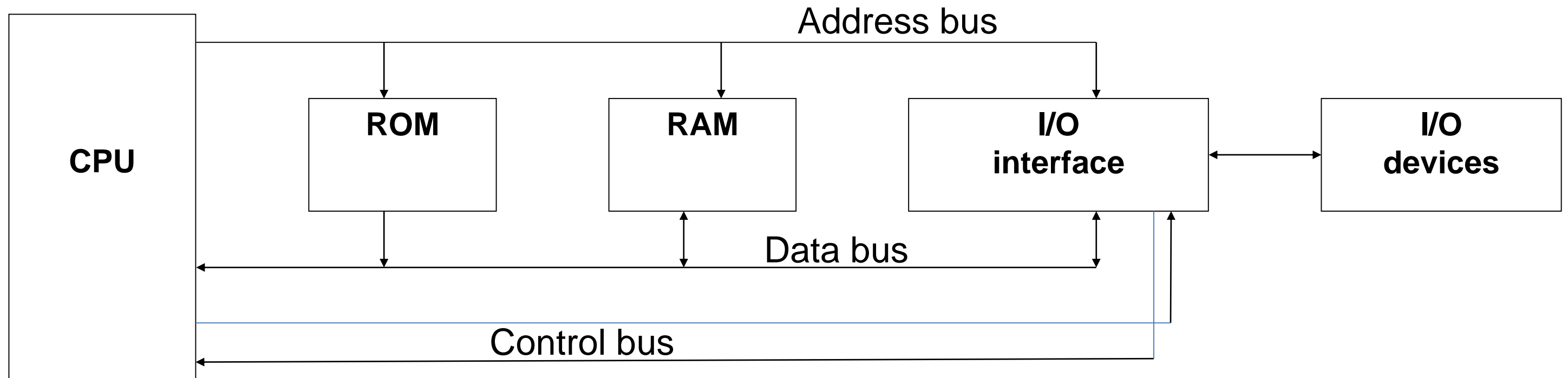
○ Address Decoding Unit

○ Fetching and execution cycles

○ Microprocessor clock system

Block diagram of a basic microcomputer

Basic computer system consist of a Central processing unit (CPU), memory (RAM and ROM), input/output (I/O) unit.



CPU - Central Processing Unit

- the portion of a computer system that carries out the instructions of a computer program
- the primary element carrying out the computer's functions. It is the unit that reads and executes program instructions.
- The data in the instruction tells the processor what to do.
- Bilgisayar sistemindeki bir bilgisayar programının komutlarını yerine getiren kısımdır.
- Bilgisayarın işlevlerini yerine getiren birincil öğedir.
- Program komutlarını okuyan, yürüten ve yazan birimdir.
- Komut verler (Program) işlemciye ne yapması gerektiğini söyler.

Pentium D dual core processors

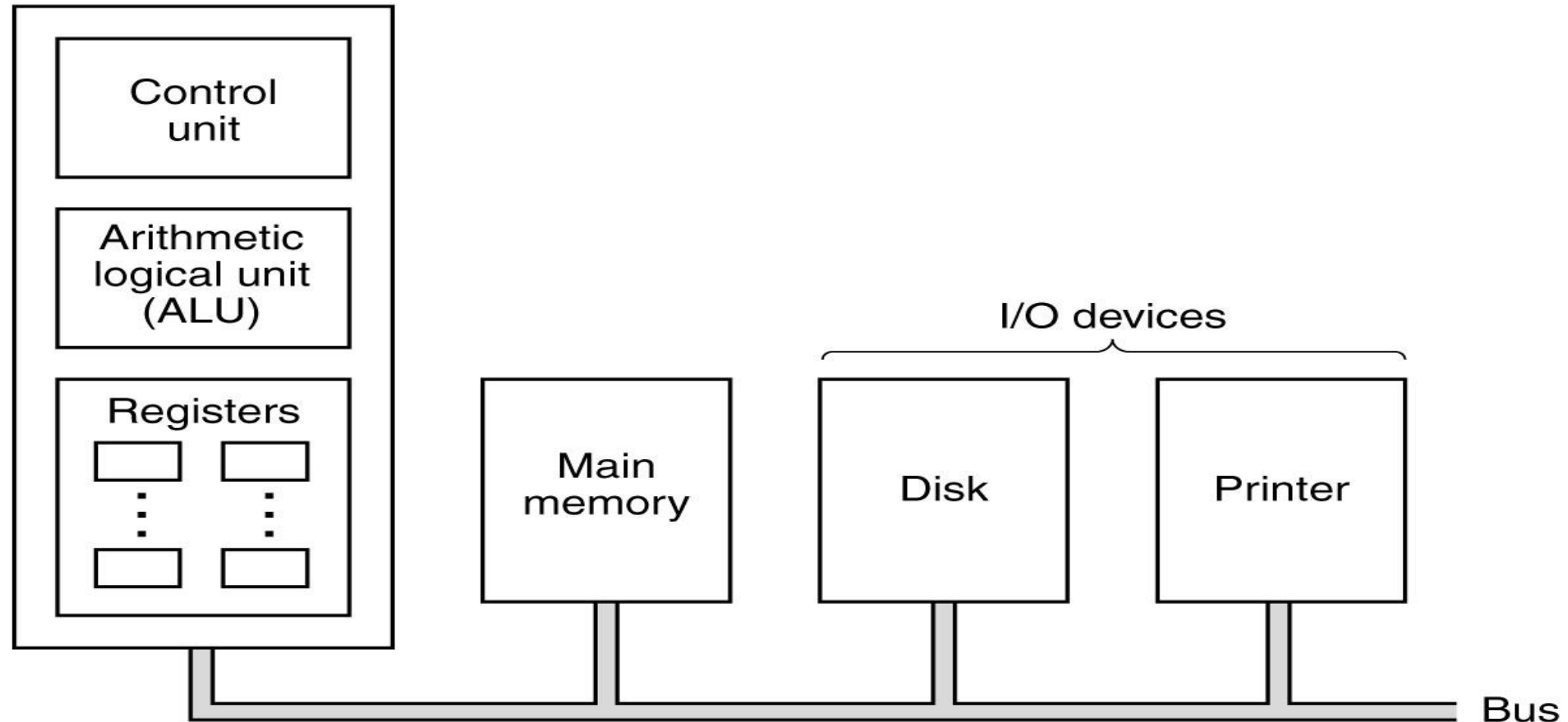


Central Processing Unit (CPU)

- CPU veya mikroişlemci (veya kısaca işlemci), bir bilgisayar sisteminin beyni olarak adlandırılır.
- CPU üç ana alt sistemden oluşur: Kontrol Birimi (CU), Aritmetik Mantık Birimi (ALU) ve Registers (Geçici Kayıt Alanları).
- Bilgisayar sisteminin hızı, kullanılan işlemcinin mimarisine göre belirlenir.

Central Processing Unit

Central processing unit (CPU)



The organization of a simple computer with one CPU and two I/O devices

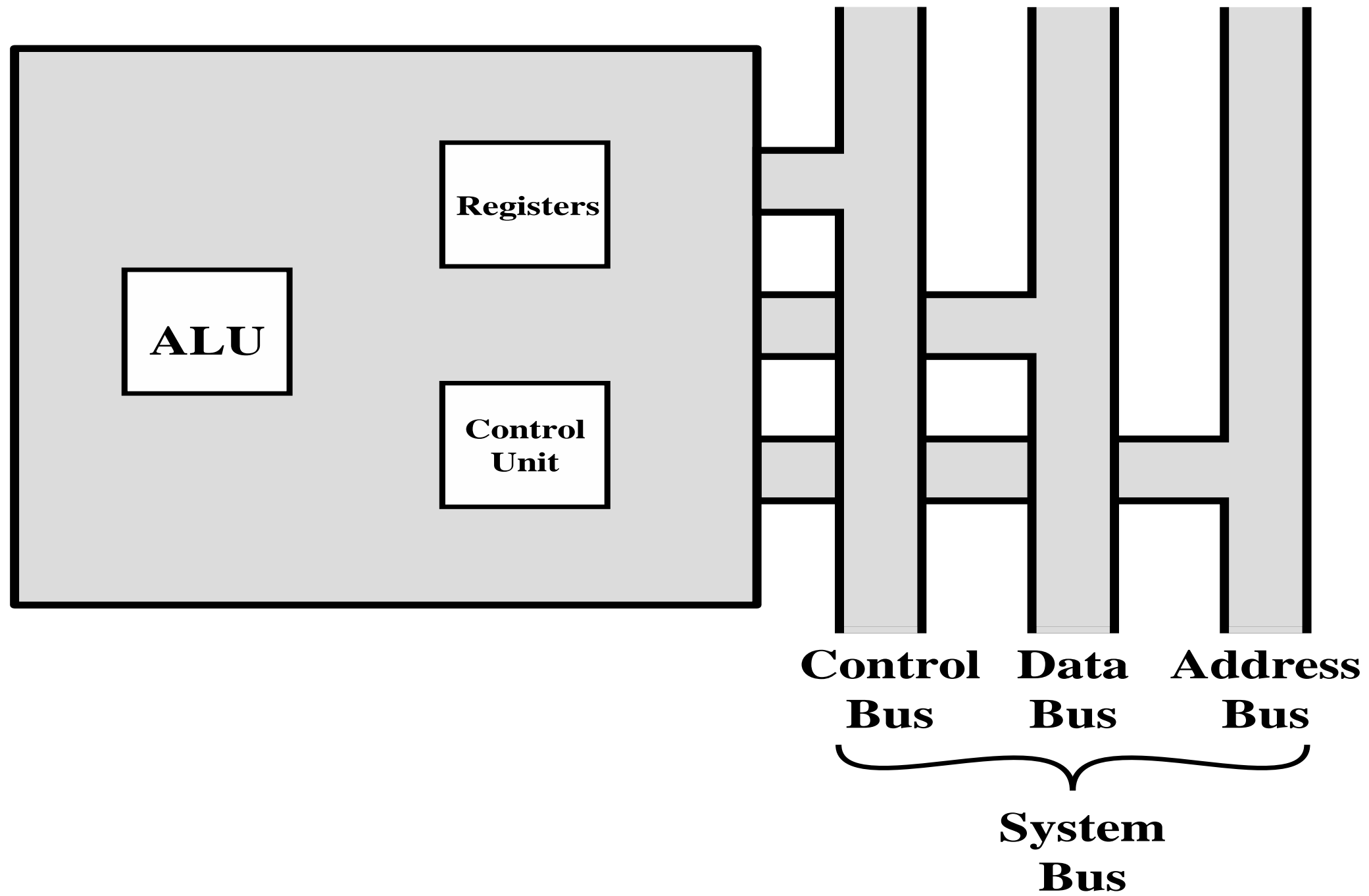
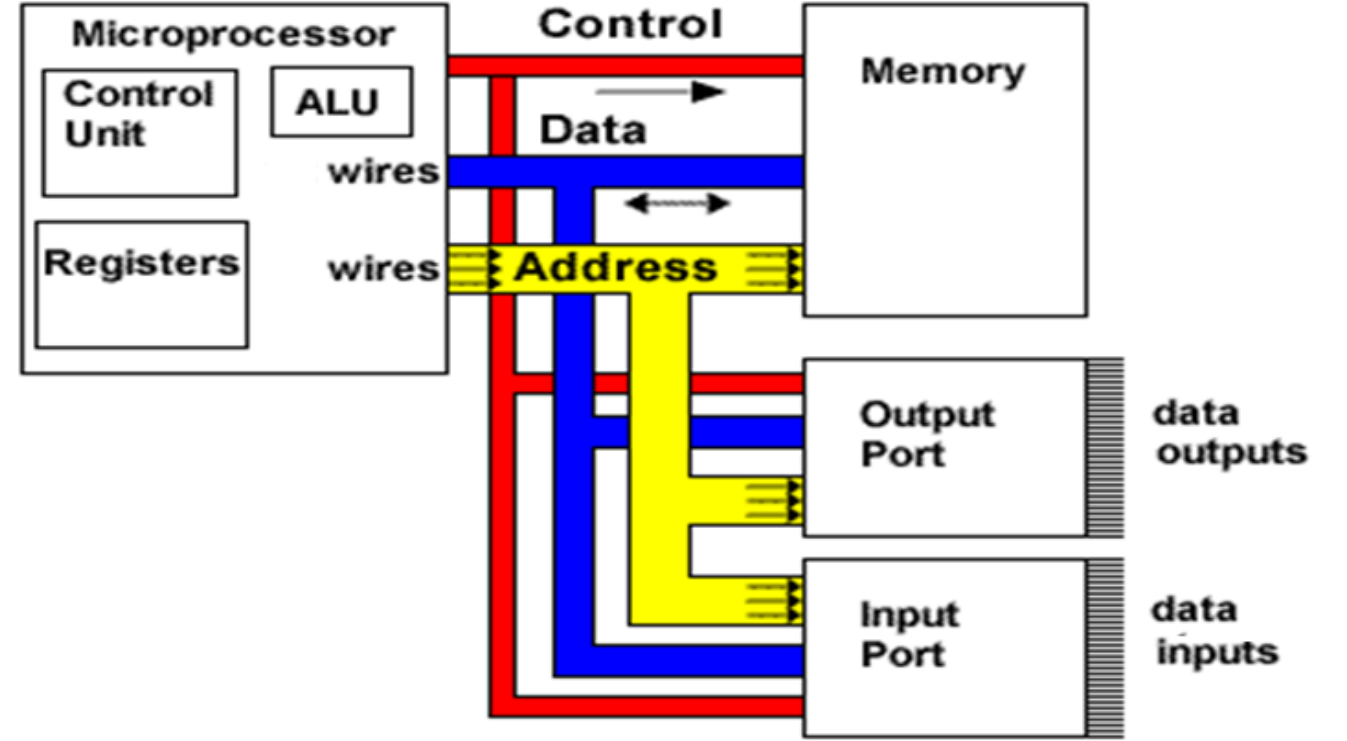


Figure 14.1 The CPU with the System Bus

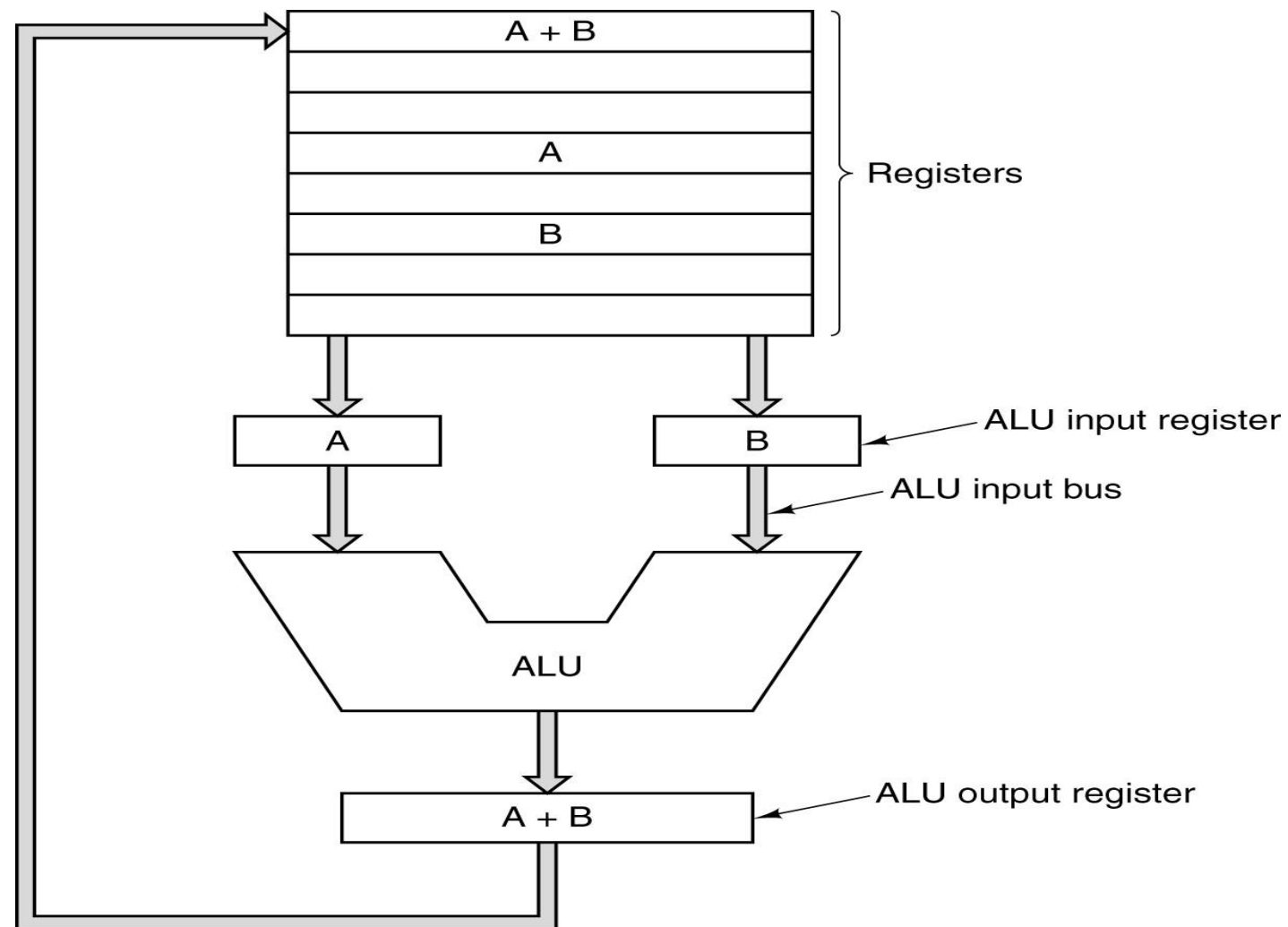
Bir mikroişlemcinin yapısı

- System Bus
- Arithmetic and logic unit
- Control Unit
- Registers (Özel Amaçlı Geçici Kayıt Alanları): İşlenecek komutların transfer edildiği geçici saklayıcılardır:
 - Data Registers: Verinin işlenmesinde ve transferinde kullanılır.
 - Segment registers: Belleklerin başlangıç adreslerini saklar. Fiziksel adres hesaplanırken sonuna 4 bit 0 eklenir.
 - Pointer and Index registers: Belleklerin başlangıç adresinden sonra verinin nerede olduğunu belirtir.
 - Program counter: İşlenecek bir sonraki komutun yerini işaret eder.
 - Flags: İşlem sonucunda değişen durumları saklar.

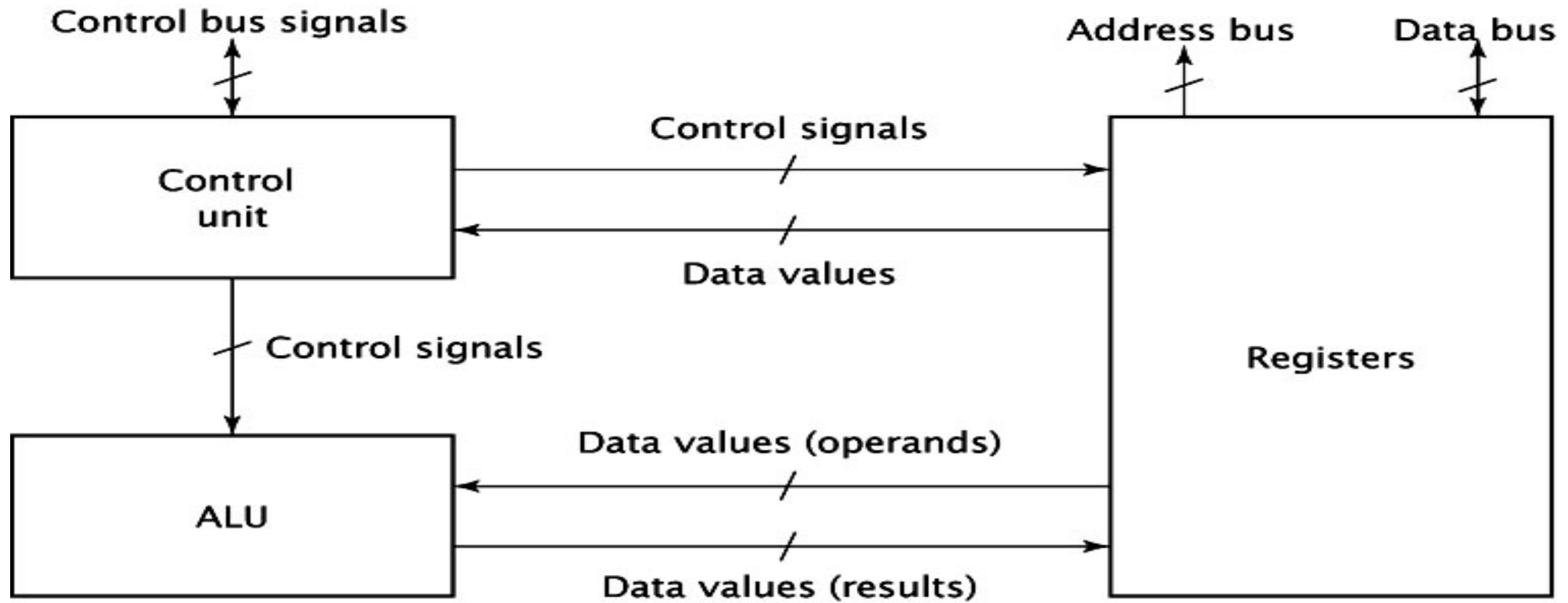


CPU Organization

The data path of a typical Von Neumann machine.



CPU Organization





System Bus

Communication inside a computer..

- Bir bilgisayar programı hem komutlardan hem de verilerden oluşur. Program giriş ünitesi aracılığıyla bilgisayar komutlar ile beslenir ve hafızada saklanır.
- Programın çalışabilmesi için komutların teker teker hafızadan (Rom /HD) getirilmesi gerekmektedir.
- Bu komutların getirilmesi kontrol ünitesi tarafından yapılır.
- Bir komut alındıktan sonra kontrol ünitesi komutun kodunu çözümler.
- Komuta göre kontrol ünitesi diğer ünitelere kontrol sinyalleri verir.

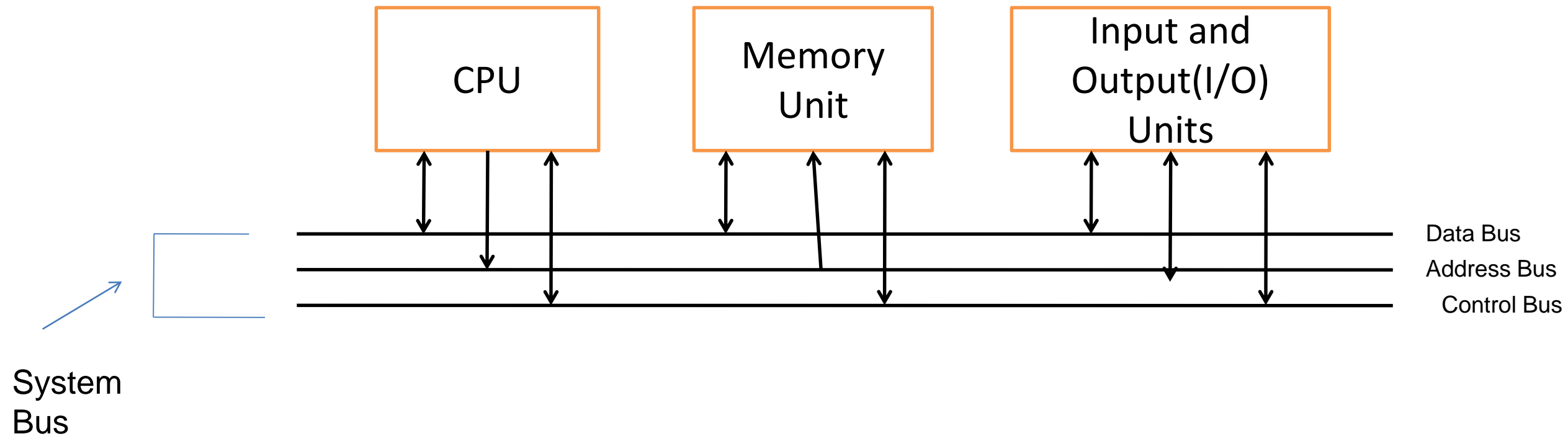
Communication inside a computer..

- Bir komut yürütüldükten sonra, komutun sonucu hafızada (Ram) saklanır veya geçici olarak kontrol ünitesinde veya ALU'da saklanır, böylece bir sonraki komut tarafından kullanılabilir.
- Bir programın sonuçları çıktı birimi aracılığıyla bilgisayardan alınır.
- Kontrol ünitesi, ALU ve Geçici özel kayıt alanları toplu olarak Merkezi İşlem Birimi (CPU) olarak bilinir.

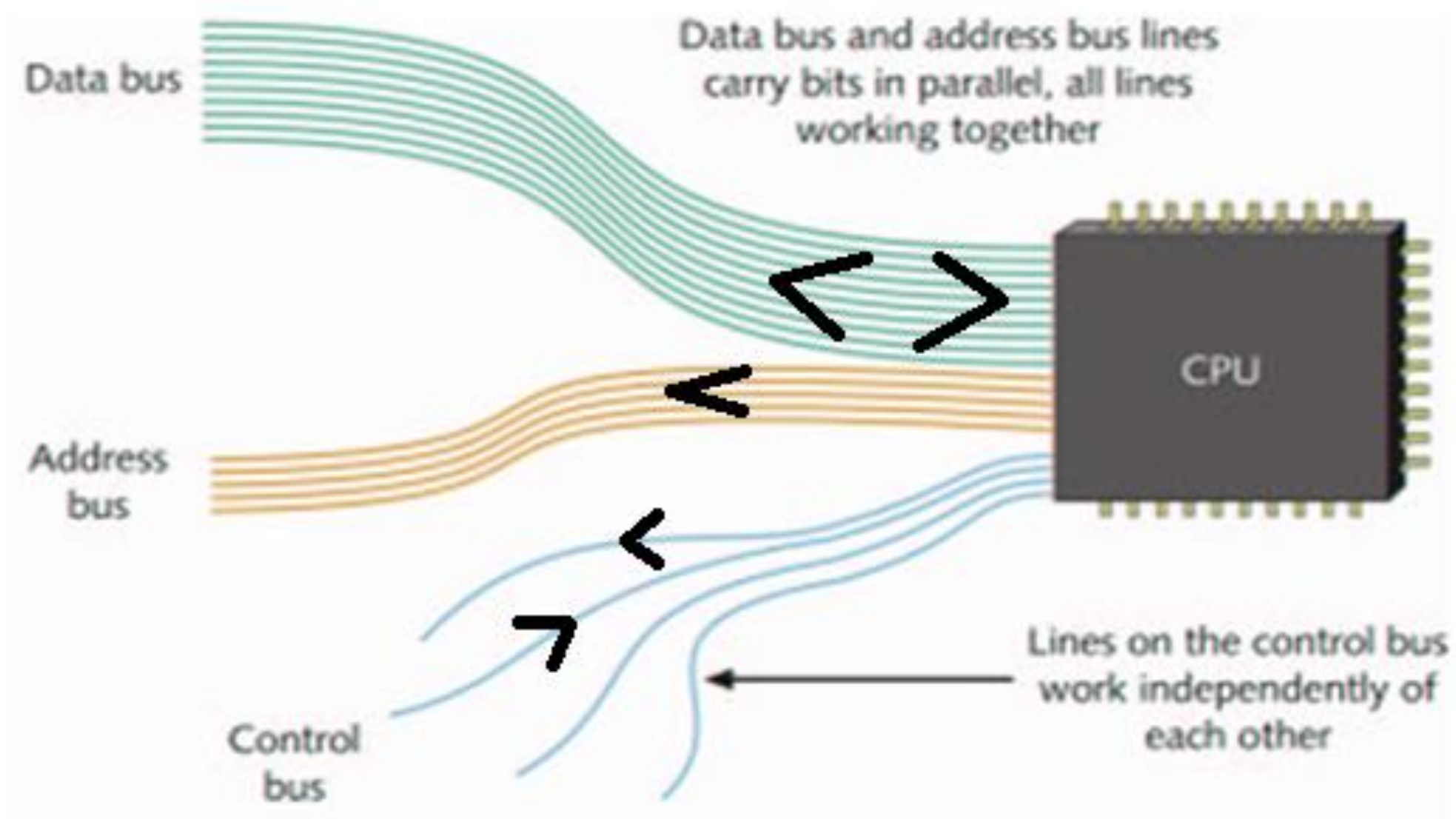
Interconnection Of Units

- A computer program consists of both instructions and data.
- The program is fed into the computer through the input unit and stored in the memory.
- In order to execute the program, the instructions have to be fetched from memory one by one and store it into registers (working memory) for processing.
- This fetching of instructions is done by the control unit.

Interconnection of Computers Units via System Bus



System Bus Components



- **System Bus – wires connecting memory & I/O to microprocessor**
 - **Address Bus**
 - Unidirectional
 - Identifying peripheral or memory location
 - **Data Bus**
 - Bidirectional
 - Transferring data
 - **Control Bus**
 - Synchronization signals
 - Timing signals
 - Control signal

- Bus is data path. Adres Bus ve Data Bus, grup olarak çalışırlar. Üzerlerinde bitleri (1 / 0) temsil eden elektrik sinyalleri taşınır.
- Bus: Herbir hat üzerinde 0 /1 (bit) bulunur. Elektriksel sinyal ile temsil edilir. Dijital sayı sistemidir.
- Data Bus: Grup çalışır. İki yönlüdür. Aynı anda ya yazar ya da okur. Sayısını belirlemede aynı anda yazılacak ya da okunaca bellek içeriği, I/O register boyut ile belirlenir. Maksimum olan seçilir. (8,16,32,64,128, 256, ...)
- Adres Bus tek yönlüdür. Hat sayısı belirlemede: bellek sayısı, bellek kapasitesi ve I/O sayısı göz önüne alınır.
- Control Bus: Bir kısmı gidiş yönünde bir kısmı ise geliş yönündedir. Sinyalleri kontrol eder, zamanlama ve kontrol amaçlıdır. Control bus checks the data to the destination.

Sistem Bus

Sistem Bus: Bir bilgisayar sisteminde CPU ile diğer temel bileşenler (CPU'nun ve Bilgisayar sisteminin) arasında haberleşmenin organize edildiği hatlardır. Herbir hat üzerinde bitler (1 / 0) elektrik sinyalleri olarak taşınır.

○ Adres Bus:

- Bir bilgisayar sisteminde bellek ve bellek gözü veya I/O birimini seçmede kullanılan tek yönlü paralel hatlardır. Grup olarak çalışırlar.

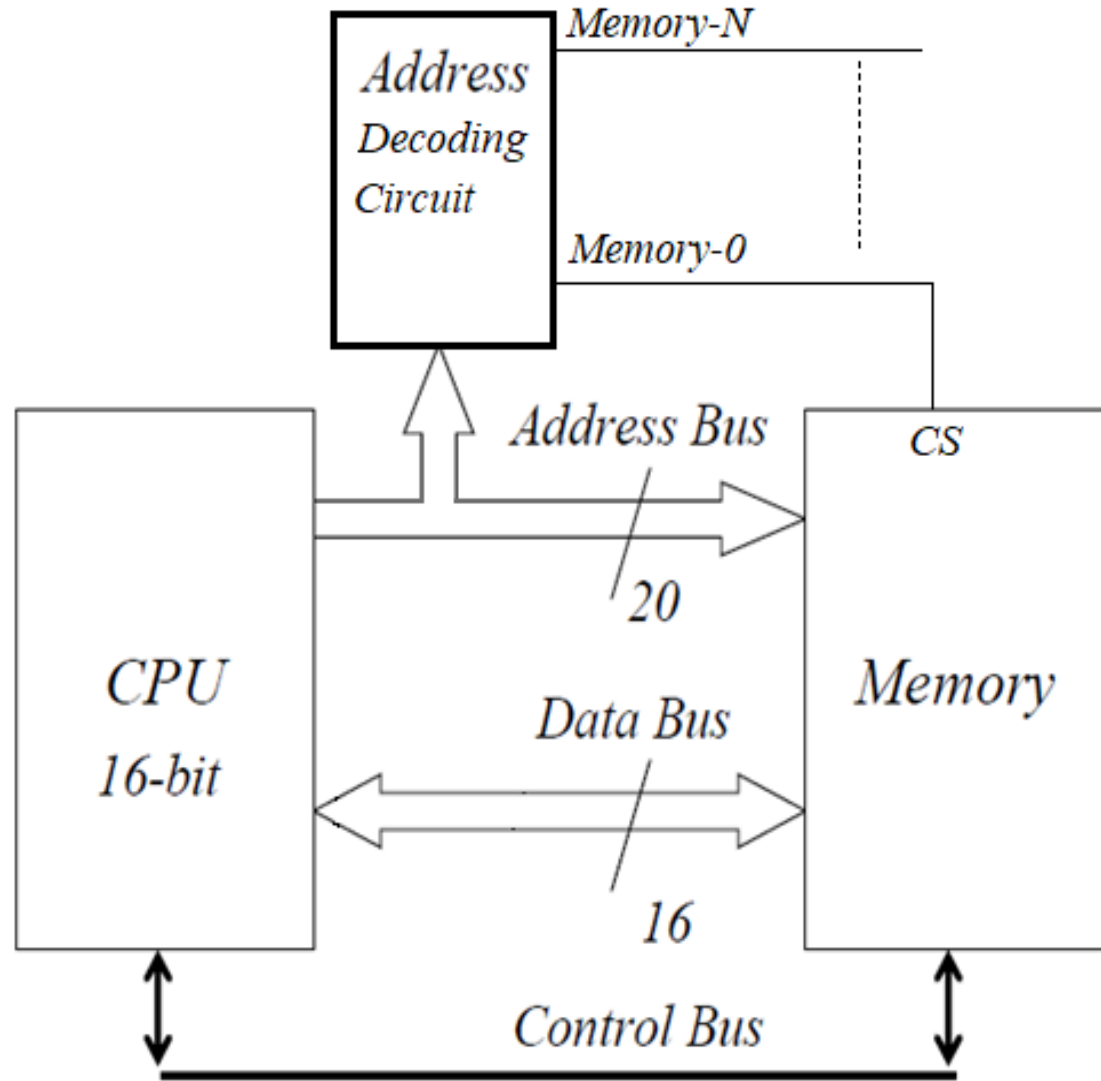
○ Data Bus:

- Bellekler ve I/O birimlerine veri yazmada ya da okumada kullanılan iki yönlü paralel hatlardır. Grup olarak çalışırlar.

○ Control Bus:

- CPU ile Bellek ya da I/O birimleri arasındaki iş akışını yönetir. Verileri hedefe kadar kontrol ederler.
- Yaygın kullanılan Control Bus hatları: Yaz, Oku, Dur, Clock, Reset (Yeniden başlat), ...
- Tek yönlüdür. Bir kısmı CPU'dan bellek veya I/O birimlerine; bir kısmı ise bellek veya I/O birimlerinden CPU'ya gider. Grup olarak çalışmazlar. Bireysel çalışırlar. Paralel hatlardır.

İşlemcinin Organizasyonu: CPU – Bellek Arayüzü



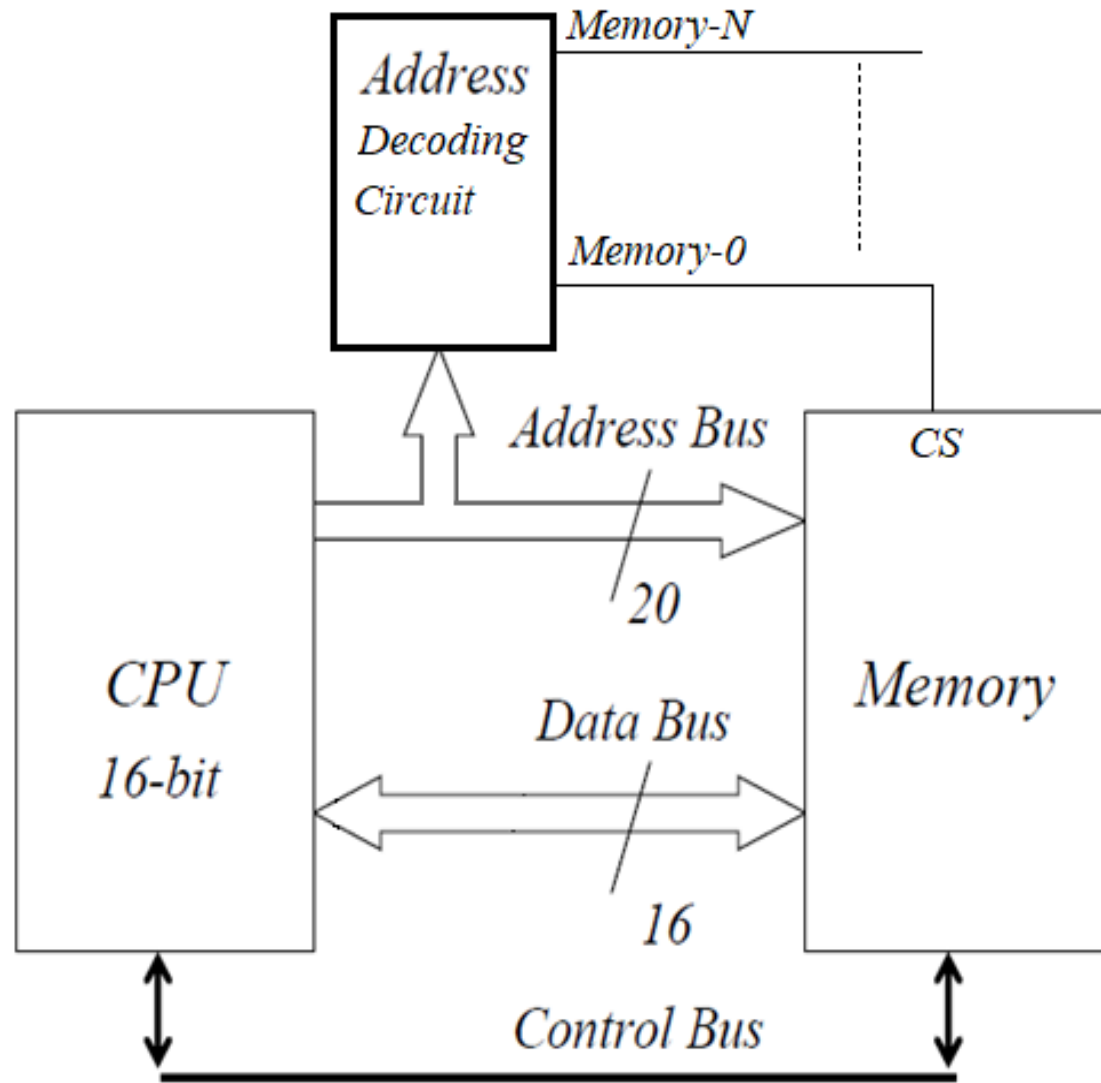
- Adres Bus CPU'dan belleklere ve Address Decoding Devresine giden adres bus'a ait hatlardan oluşur.

Adres Bus Hatları paraleldir, grup halinde çalışır; her bir hat üzerinde elektriksel sinyal olarak ikili durum (bit) 0 ya da 1 değeri mevcuttur.

CPU'dan çıkan Adres Bus hat sayısı CPU'nun erişebileceği maksimum bellek kapasitesini verir. Buradaki 20 adet adres bus hat sayısı, $n=20$ ise adreslenecek maksimum bellek kapasitesi $=2^{20} = 1\text{Mbyte}$ dir.

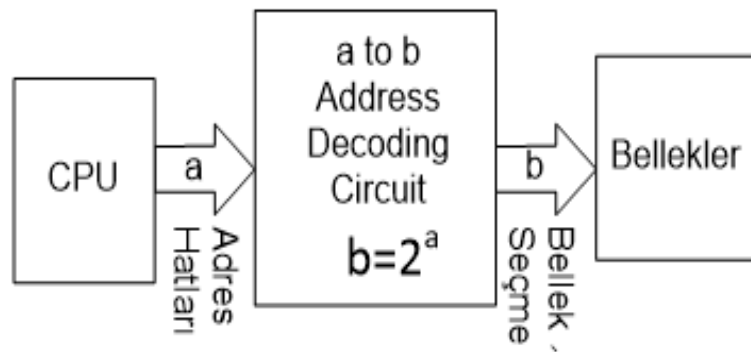
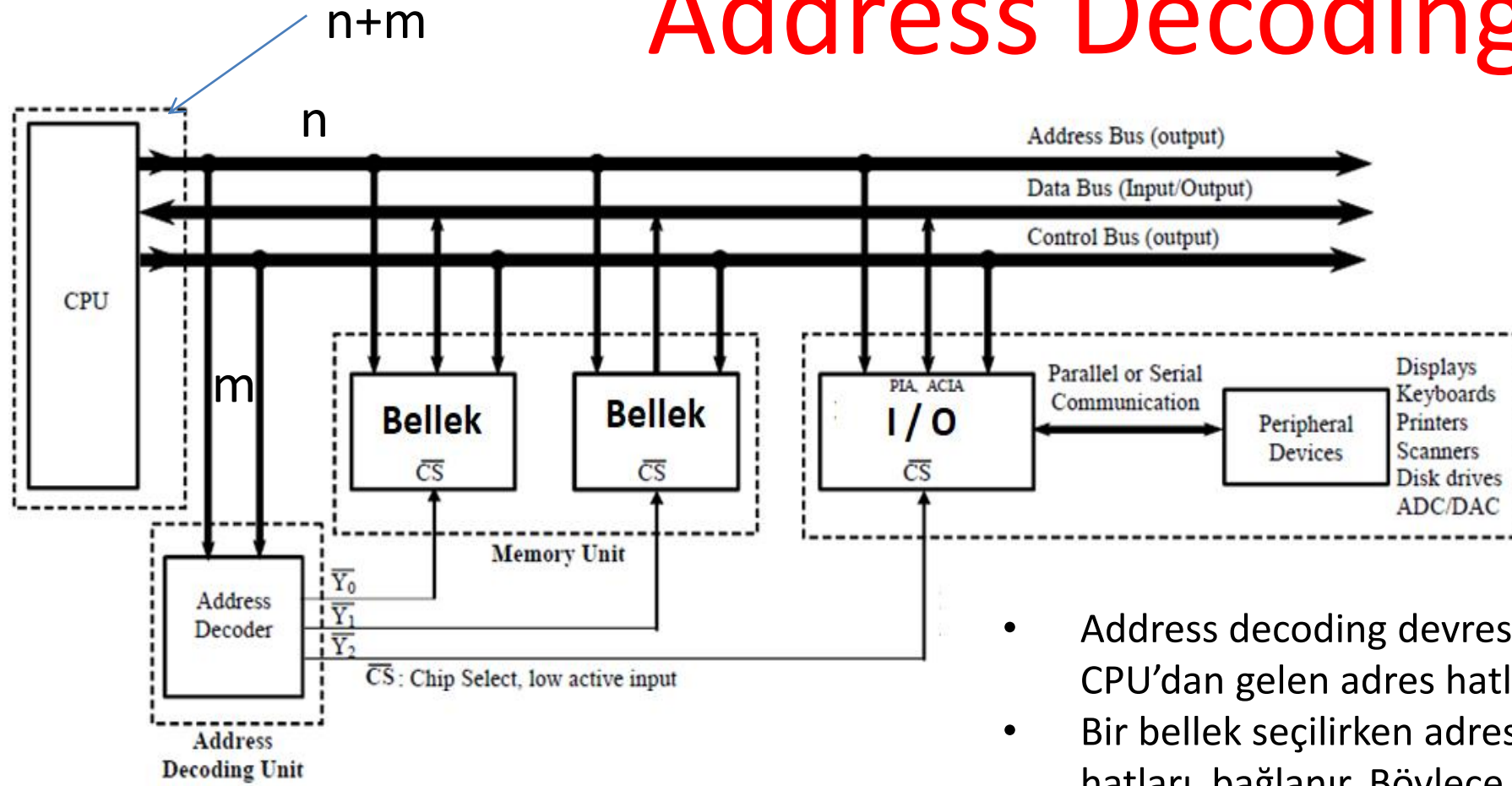
Address Decoding Devresi erişilecek bellekleri seçer.

İşlemcinin Organizasyonu: CPU – Bellek Arayüzü



- CPU'dan çıkıp belleklere gelen Adres Bus hat sayısı, belleklerin kapasitesini belirler ve verinin kayıt edildiği bellek gözüne erişimi sağlar. Data bus, bellek gözüne yazılıp ya da okunacak veridir. Hatlar paraleldir ve grup olarak çalışır. Herbir hat üzerinde 0 ya 1 ikili durum mevcuttur. Data Bus hat sayısı CPU'nun veri transfer özelliğini verir. CPU, 16 bit dendiğinde Data Bus hat sayısı 16 olan bir CPU'dan bahseder. Word olarak 16 bit yazılıp okunur. 2 Byte'a denk gelir. Control bus hatları da paraleldir. Grup olarak çalışmaz. Ayrık çalışır. Herbir hat üzerinde 0 ya 1 ikili durum mevcuttur.

Address Decoding Unit



- Address decoding devresinin çıkışları bellek seçer. Seçilen belleğin gözüne CPU'dan gelen adres hatları yardımıyla erişilir.
- Bir bellek seçilirken adres decoding devresinin girişine CPU'dan gelen adres hatları bağlanır. Böylece adres decoding devresinin çıkışları bellekleri seçmede kullanılır.
- Seçilecek bellek sayısı=Address decoding devresi çıkış sayısı= 2^m dir. Burada m CPU'dan address decoding devresinin girişine gelen adres hattı sayısıdır.
- Herbir belleğin kapasitesini belirleyen (Kapasite= 2^n) n adet hat CPU'dan gelir.
- Amaç aynı anda bir belleğin ilgili veri gözünün seçilmesidir.
- Adres decoding devresinin çıkış sayısı bellek sayısına eşit ya da büyük olmak zorundadır. (2 'nin üssüne eşit olmayan bellek sayılarında)

Örnek

- I/O sayısı: 21, Bellek sayısı: 32, Maksimum bellek kapasitesi: 37Gbyte ise CPU'dan çıkacak adres hattı sayısı nedir?
- Toplam birim sayısı, $N=21+32=53$, U0, ..., U52
- Bir anda I/O ve Belleklerden birini seçmek için CPU'dan çıkacak Adres Hattı Sayısı m ise, toplam birim sayısı, $N=2^m$ dir. Bu ifadeden $m=6$ bulunur.
- Maksimum Bellek Kapasitesi=37Gbyte, 64Gbyte alınır.
- $64\text{Gbyte}=2^6 * 2^{30}=2^{36}$ Byte
- Maksimum belleğe bellek gözlerini seçmek için gidecek adres hattı sayısı, $k=36$ dır.
- İndisleme: A35, A34, ... , A1, A0. Not: Diğer bellekler ne kadar adres hattına ihtiyaç duyarlarsa o kadar adres hattı paralel gider.
- Toplam adreshattı sayısı = Maksimum belleğe bellek gözlerini seçmek için gidecek adres hattı sayısı + Bir anda I/O ve Belleklerden birini seçmek için CPU'dan çıkacak Adres Hattı Sayısı
- Toplam adres hattı sayısı = $k+m= 36 + 6 =42$
- İndisleme: A41, A40, ... , A1, A0

A41	-	-	-	A36	A35	-	-	-	A1	A0
Toplam 6 adet					Toplam 36 adet					
Bir anda I/O ve Belleklerden birini seçmek					Maksimum bellek ve diğer bellek gözlerini seçmek					

Adres Dekoding Circuit:

- Giriş - Çıkış
- 000000 – U0
- 000001 – U1
-
- 110100 – U52
-
- 111111 – U63

Number of Address Bus

Number of Address Bus, N

- ↓
- Parallel lines
- Every line has bit: 0/1
- 2^N cases.

↳ Every case select

- Memory - memory cell
or I/O units.

a 0/1
b 0/1
c 0/1

a	b	c	$2^3 = 8$
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Total memory capacity is - 2^N
(Number of memory cell)

Example, $N = 44$ how much total capacity of memory. 1 cell has 8 bit → byte.

$$\text{Memory Capacity / Number of memory cell} = 2^{44} = 2^4 \cdot 2^{40} = 16 \text{ TByte}$$

↓ Tern

Example, Memory Capacity is 37 Mbyte, what is the

Number Address Bus Lines = 26 (Number of address Bus Lines)

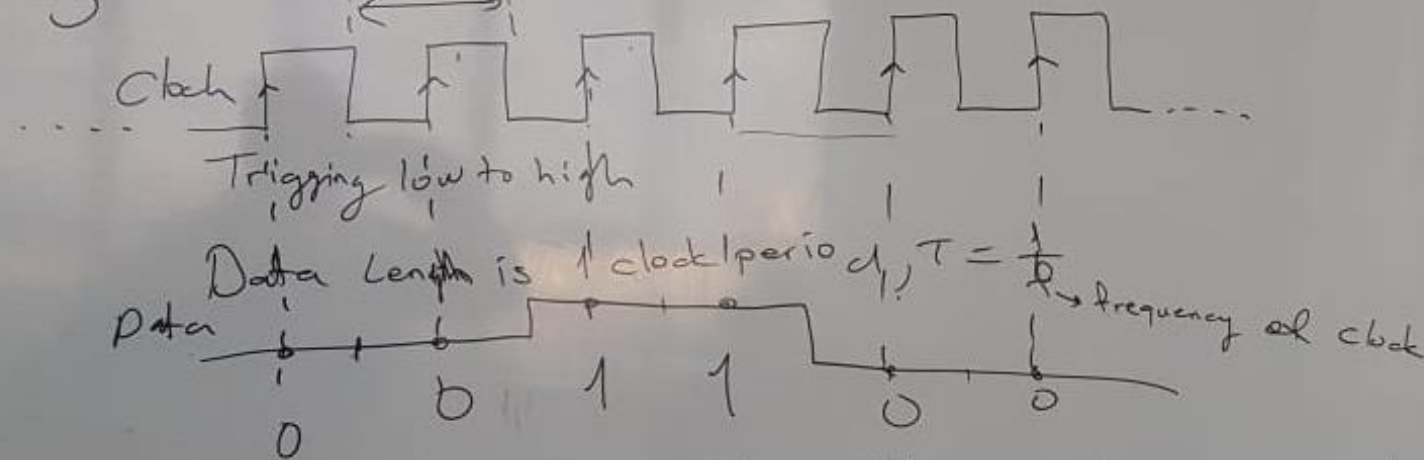
$$37 \text{ Mbyte} \rightarrow 64 \text{ Mbyte} = 2^6 \cdot 2^{20} = 2^{26}$$

Number of Data Bus

Data bus:

- 1- Parallel lines, every line has bit (0/1)
- 2- Carry - data (write to memory cell, two directional (R/W))
every memory cell has 8 bit = 1 byte.

3 - Clock & Timing Synchronize the data as bit (0/1)



- 4- Parallel lines carry the data, write or read to memory cell. at 1 clock period.
for example, if number of lines are 64, what is means that?
as 1 clock period. write or read 64 bits.
64 bits = 8 byte.



CPU: Mikroişlemci

Mikroişlemci Nedir?

- Mikroişlemciler, bilgisayar sisteminin kalbidir. Bilgisayar operasyonlarını kontrol ederek veri işleme işlevlerini yerine getirir. Kısaca işlemci veya CPU (Central Process Unit-Merkezi İşlem Birimi), kullanıcı ya da programcı tarafından yazılan programları meydana getiren komutları veya bilgileri yorumlamak ve yerine getirmek için gerekli olan tüm mantıksal devreleri kapsar.
- İlk mikroişlemci 1971 yılında hesap makinası amacıyla üretilen Intel firmasının 4004 adlı ürünüdür. Bir defada işleyebileceği verinin 4-bit olmasından dolayı 4-bitlik işlemci denilmekteydi.

Mikroişlemcilerin İşlevsel Bölümleri

- **Sayıcılar (Counter):** Sayıcılar işlemi yapılacak komut ve verilerin adreslerini taşıyarak bilgisayarın çalışması sırasında hangi verinin hangi sırayla kullanılacağını belirlerler.
- **Giriş-Çıkış Devreleri:** Bu devreler mikroişlemcinin, yalnız giriş ve yalnızca çıkış yapan veya giriş-çıkış yapan birimleri ile bağlantı kurduğu devrelerdir.
- **Kayan Nokta Birimi (FPU):** Matematik işlemcisi olarak da bilinir. Mikroişlemcide yoğun matematik işlemleri yapan birimdir. Mikro işlemcinin işlem gücünü belirlemektedir.
- Grafik İşlem Birimlerinin (GPU).

Mikroişlemcilerin İşlevsel Bölümleri

- **Ön Bellek (Cache):** Mikroişlemcinin sonraki adımlarda işleyeceği verilerin önceden transfer edilip hazırlandığı kendi ön belleğidir.
- **Clock Darbe Katarı:** Mikro işlemcide temel parametre clock hızıdır. Clock darbe katarı, CPU işleminde bitleri senkronize (eş zamanlı) eder. Her bilgisayarın bir sistem clock sinyali vardır. Clock sinyali 1 ve 0 lardan oluşan darbe katarıdır. Bit olarak tanımlanan verileri tetiklemede yani işlemede kullanılır. Kişisel bilgisayarın hızları genellikle gigahertz (GHz) cinsinden ifade edilir. Clock'un periyodu veri uzunluğunda olmalıdır. Clock süreklidir, sonsuzdur. 1 ve 0 in süresi (periyodu), 1 bit uzunluğundadır. Darbe katarı, ezelden ebede gider.

Mikroişlemcileri Birbirinden Ayıran Özellikler

- 1) Mikro işlemcinin her saat darbesinde işlem yapabileceği bit sayısı: İşlenen veriler işlemcinin özelliğine göre 4-bit, 8-bit, 16-bit, 32-bit ve 64-bit uzunluğunda olabilir. Veri yolu uzunluğuna eşittir.
- 2) Komut İşleme Hızı: Mikro işlemcilerin çalışması için saat sinyallerine ihtiyaç vardır. İşlemci (CPU) her saat sinyalinde bir sonraki işlem basamağına geçer. Saat frekansı mikro işlemciye dışardan uygulanan ya da işlemcinin içinde bulunan osilatörün frekansıdır. Komut çevrim süresi ise herhangi bir komutun görevini tamamlayabilmesi için geçen süredir.
- 3) Adresleme Kapasitesi: Bir işlemcinin adresleme kapasitesi, adresleyebileceği veya doğrudan erişebileceği bellek alanının büyüklüğüdür. Bu büyüklük işlemcinin adres hattı sayısına bağlıdır. Bu hattın sayısı tasarlanacak sistemde kullanılacak bellek miktarını da belirlemektedir.
- 4) Kaydedici Sayısı: Mikro işlemcilerde kaydediciler, genel amaçlı kaydediciler ve özel amaçlı kaydediciler olmak üzere iki grupta toplanır. Bu kaydediciler 8, 16, 32 ve 64-bitlik olabilir. Kaydedicilerin sayısının programcının işinin kolaylaştırmasının yanında programın daha sade ve anlaşılır olmasını da sağlar. Her mikro işlemcinin kendine has yapısı ve kaydedici isimleri vardır. Herhangi bir mikro işlemciyi programlamaya başlamadan önce mutlaka bu kaydedicilerin isimlerinin ve ne tür işlevlere sahip olduklarının iyi bilinmesi gerekir.

Mikroişlemcileri Birbirinden Ayıran Özellikler

Farklı Adresleme Modları: Bir komutun işlenmesi için gerekli verilerin bir bellek bölgesinden alınması veya bir bellek bölgesine konulması ya da bellek–kaydedici veya kaydedici–kaydedici arasında değiştirilmesi için farklı erişim yöntemleri kullanılır. Mikro işlemcinin işleyeceği bilgiye farklı erişim şekilleri, "adresleme yöntemleri" olarak ifade edilir. Kısaca adresi tarif yollarıdır.

Adresleme türleri;

- Doğrudan adresleme
- Dolaylı adresleme
- Veri tanımlı adresleme
- Kaydedici adresleme
- Mutlak adresleme
- Göreceli adresleme
- İndisli adresleme
- Akümülatör ve imalı adresleme

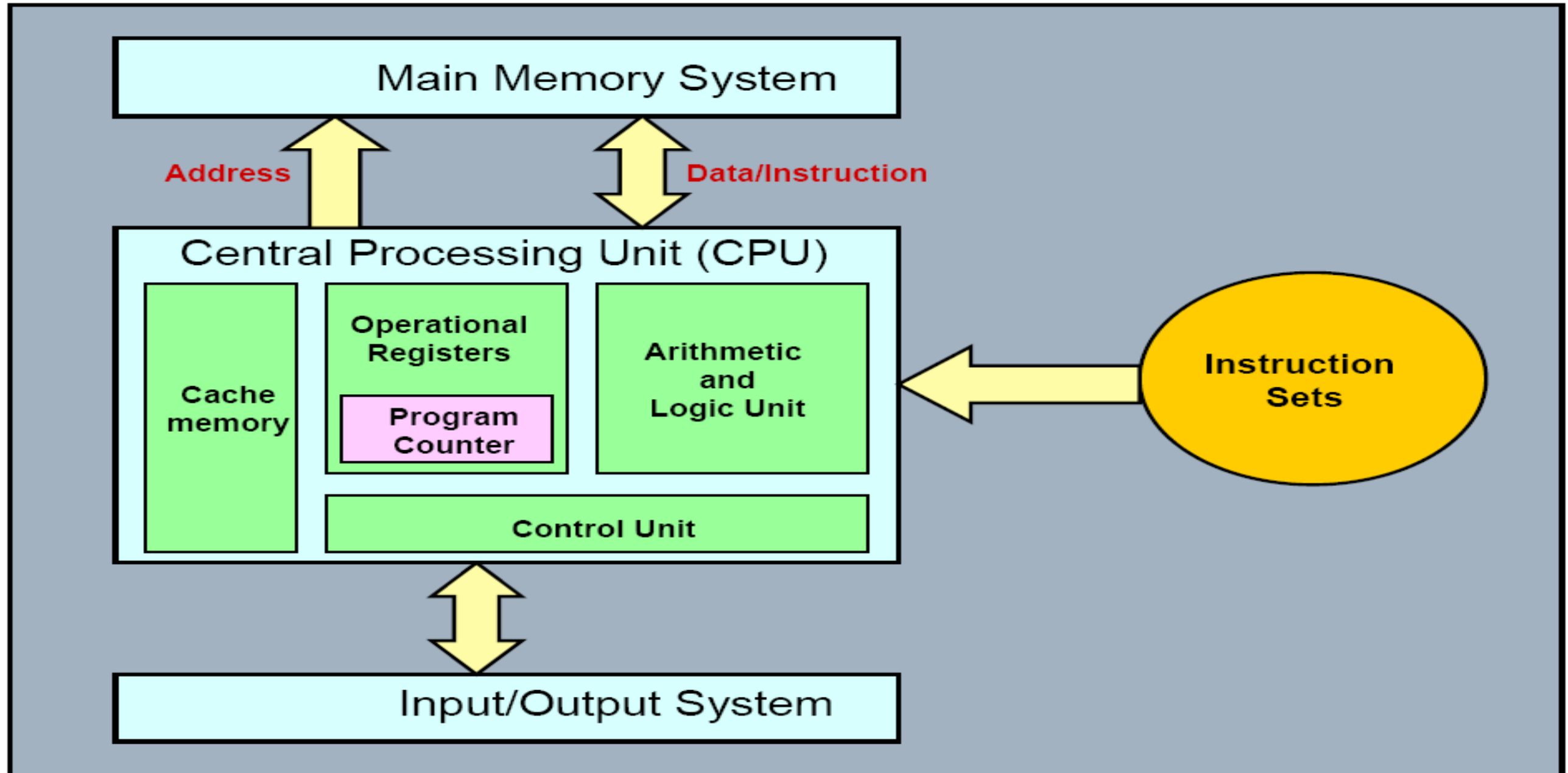
Mikroişlemcileri Birbirinden Ayıran Özellikler

İlave Edilecek Devrelere Uyumluluk: Mikro işlemcili sisteme eklenecek devrelerin en azından işlemci hızında çalışması gerekir. Sisteme ilave edilecek bellek entegrelerinin hızları işlemci ile aynı hızda olması tercih edilmelidir. Aynı şekilde sisteme takılan giriş çıkış birimlerinin hızları ve performansları mikro işlemci ile aynı veya çok yakın olmalıdır. Sisteme takılan birimlerin hızları mikro işlemciye göre düşükse mikro işlemcinin hızı diğer elemanlardaki yavaşlıktan dolayı düşer. Örneğin bir sensör ile ortam sıcaklığını ölçtüğümüzü düşünelim. Eğer sensörümüz geç ısınıp, soğuyorsa mikroişlemci ile sensörümüzü hızlı okumanın bir anlamı olmaz.

Specialized Processors

- DSP - Digital Signal Processors
 - Image processing; sound, speech
- Math co-processors
 - Real number arithmetic
- ASICs - Application-Specific Integrated Circuits
 - Microwave controller
 - Engine management controller

Content Coverage

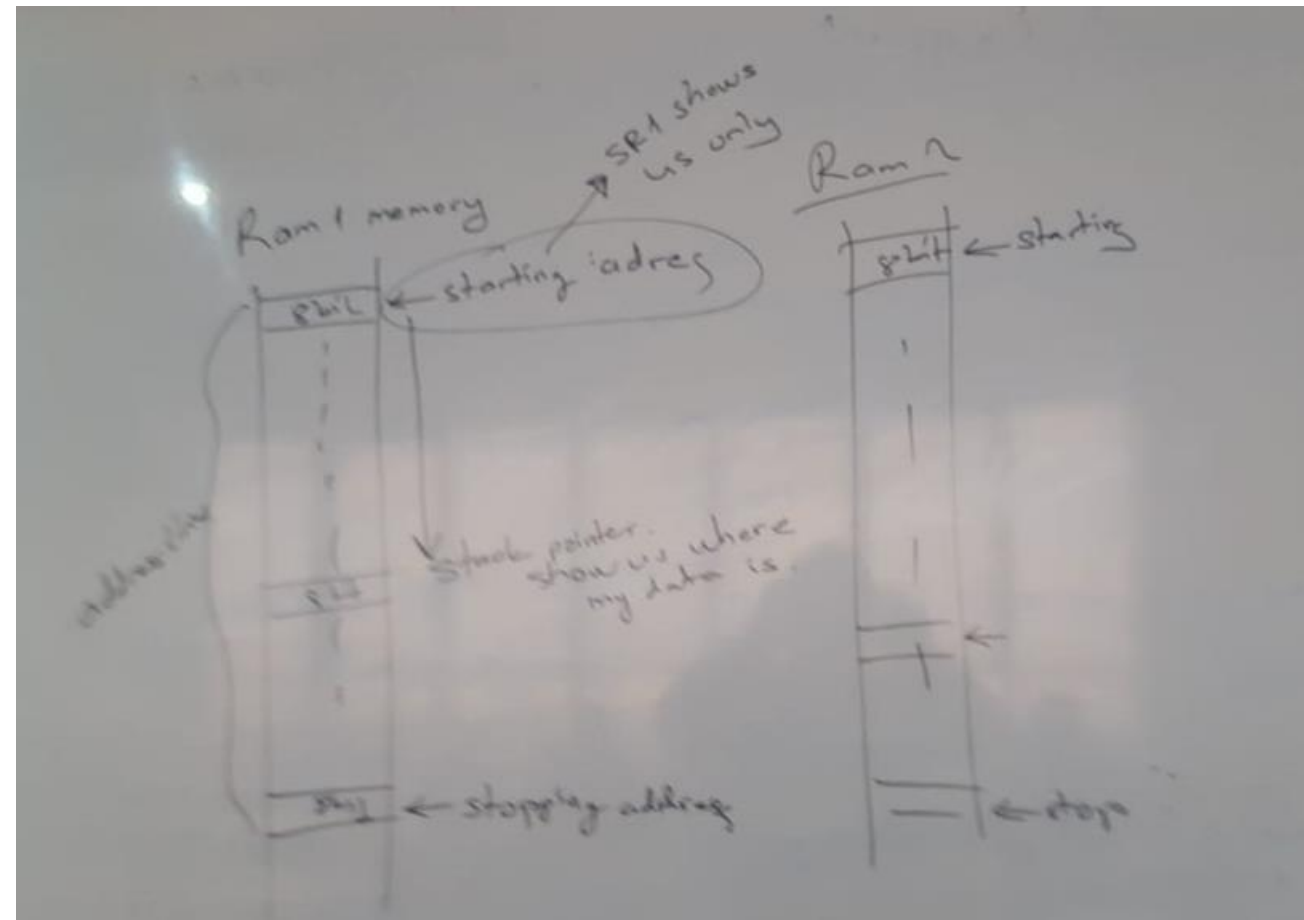


CPU Internal Structure

Internal structure and basic operation of a microprocessor

- Arithmetic and logic unit (ALU), Flag
- Control unit,
- Clock and Timing,
- Register sets,
 - accumulator,
 - condition code register,
 - program counter,
 - stack pointer,
 - segment registers

Starting address of the segment register



CPU'yu Oluşturan Ana Bileşenler

İşlemcilerin içerisindeki donanım blokları, komutları uygun sırayla bellekten okur ve gerekli verilere erişip, bu veriler üzerinde komutta tanımlanan işleri yürütür.

- **Önbellekler (Cach):** İşlenecek verinin önceden getirilip CPU'da hazır edilmesinde kullanılır. Performans artırıcı özelliği vardır. Çünkü yazma ve okuma sürelerini hızlandırır. İkinci önceliklidir.
- **Özel Amaçlı Saklayıcılar (Register):** Mikroişlemcide işlenecek ya da transfer edilecek verinin geçici olarak saklandığı (RAM), Kontrol ve ALU birimlerinin doğrudan bağlandığı bellek birimidir. Birinci önceliklidir. Özel Amaçlı Saklayıcılar verinin manevrasında ve geçici olarak tutulmasında görevlidirler. Veri işleme ve veri iletişim ara yüzünde kullanılır. Yüksek hızlı veri işlemede ve transfer etmede hızlı rol oynar. Geçici depolama alanıdır.

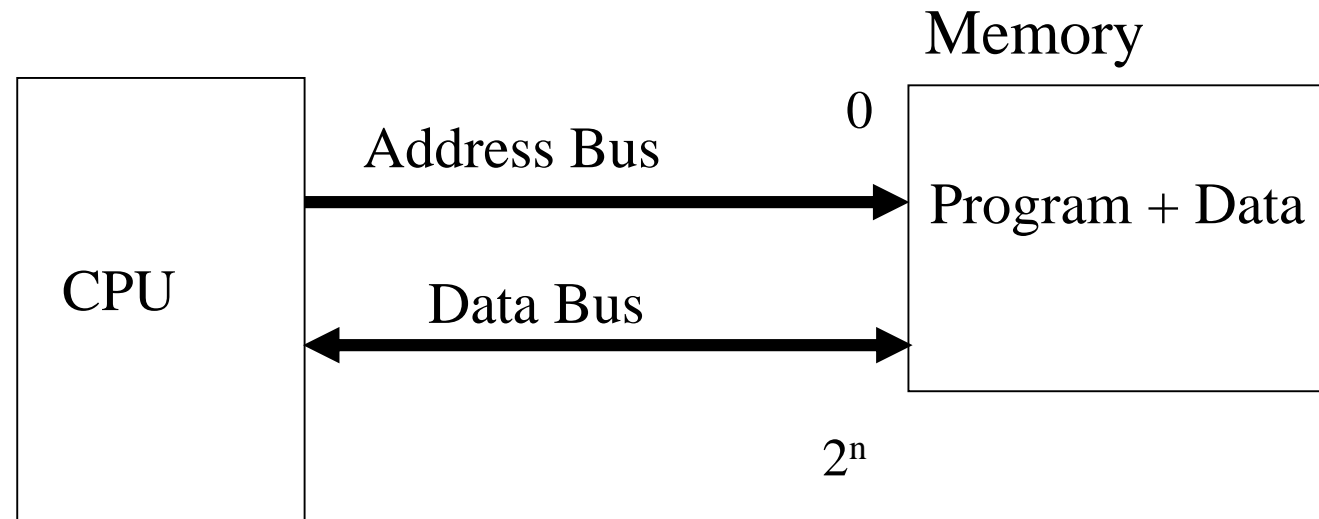
CPU'yu Oluşturan Ana Bileşenler

- **Aritmetik ve Mantık Birimi (Arithmetic Logic Unit -ALU):** Tüm matematik ve mantık işlemlerini gerçekleştirir, CU'dan komut dizisini alır. Mikro işlemcinin birinci derecede önem taşıyan bir birimidir
- **Kontrol Ünitesi (Control Unit -CU):** Donanımların çalışmasını düzenler. İşlem akışını düzenler, komutları yorumlar ve bu komutların yerine getirilmesini sağlar. CU, işlenen komuta göre mikro işlemci içerisindeki operasyon zamanlamasını / sıralamayı kontrol eder.
- **Haberleşme Yolları:** İşlemci iç mimarisindeki blokları birbirine bağlayan veri yolu yapılarıdır. Mikro işlemci ile bilgisayarın diğer birimleri arasındaki bağlantıları sağlayan iletkenlerdir. Veri yolları (data bus), Adres yolları (address bus), Denetim yolları (control bus)

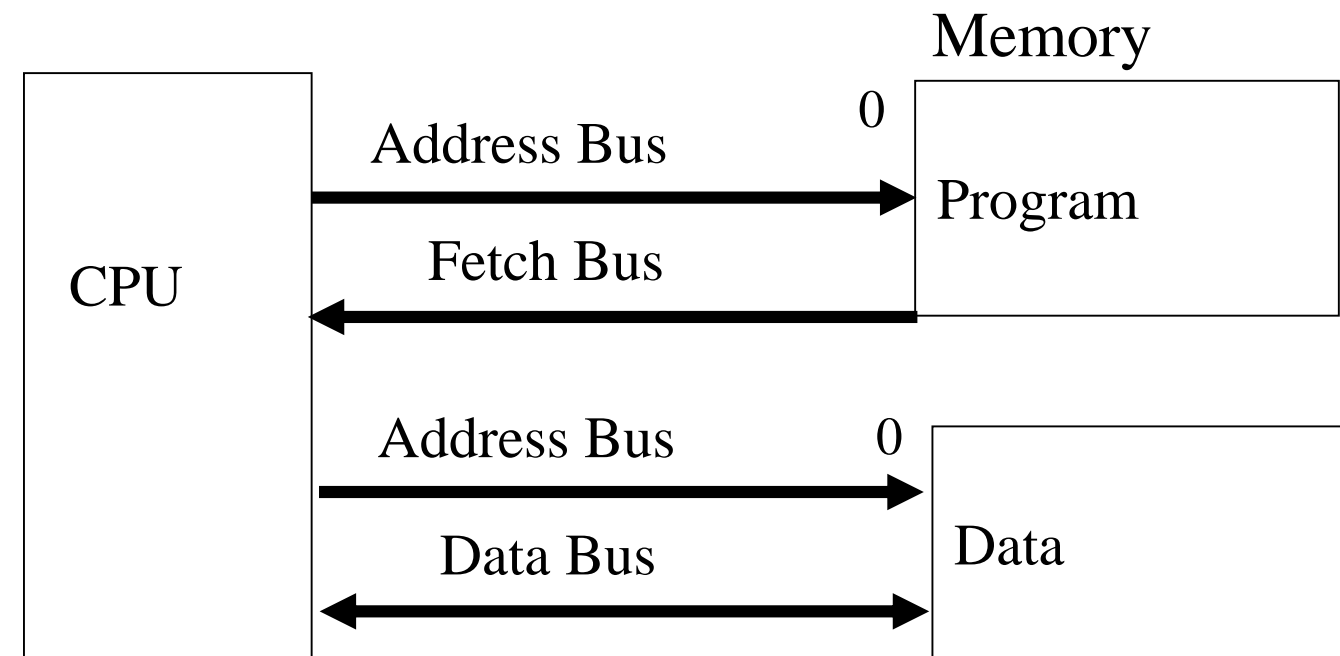
Von Neumann and Harvard architectures

- Von Neumann
 - Allows instructions and data to be mixed and stored in the same memory module
 - More flexible and easier to implement
 - Suitable for most of the general purpose processors
- Harvard:
 - Uses separate memory modules for instructions and for data
 - It is easier to pipeline and there are no memory alignment problems
 - Higher memory throughput
 - Suitable for DSP (Digital Signal Processors)

CPU Architectures



Von Neumann
Architecture



Harvard
Architecture

The structure of a microprocessor

○ Arithmetic and logic unit

○ Control Unit

○ Clock

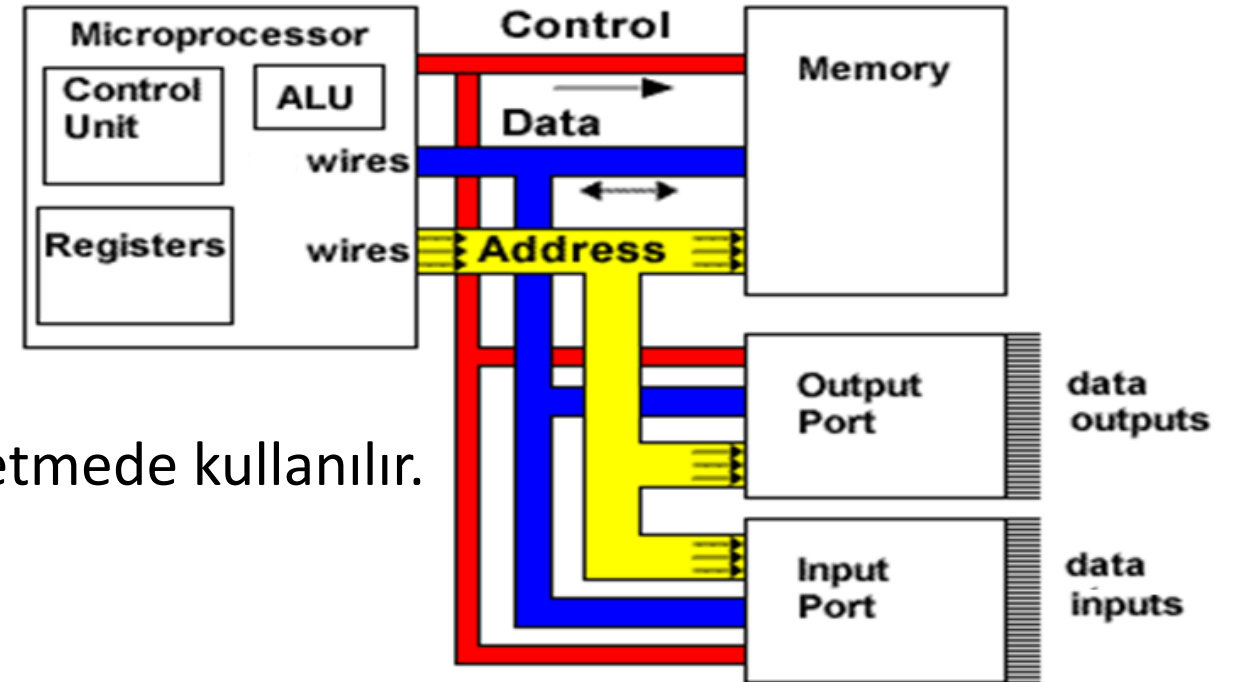
○ Registers:(Özel Amaçlı Saklayıcılar):

Özel amaçlı saklayıcıdır. Yüksek hızlı veri işlemde ve transfer etmede kullanılır.

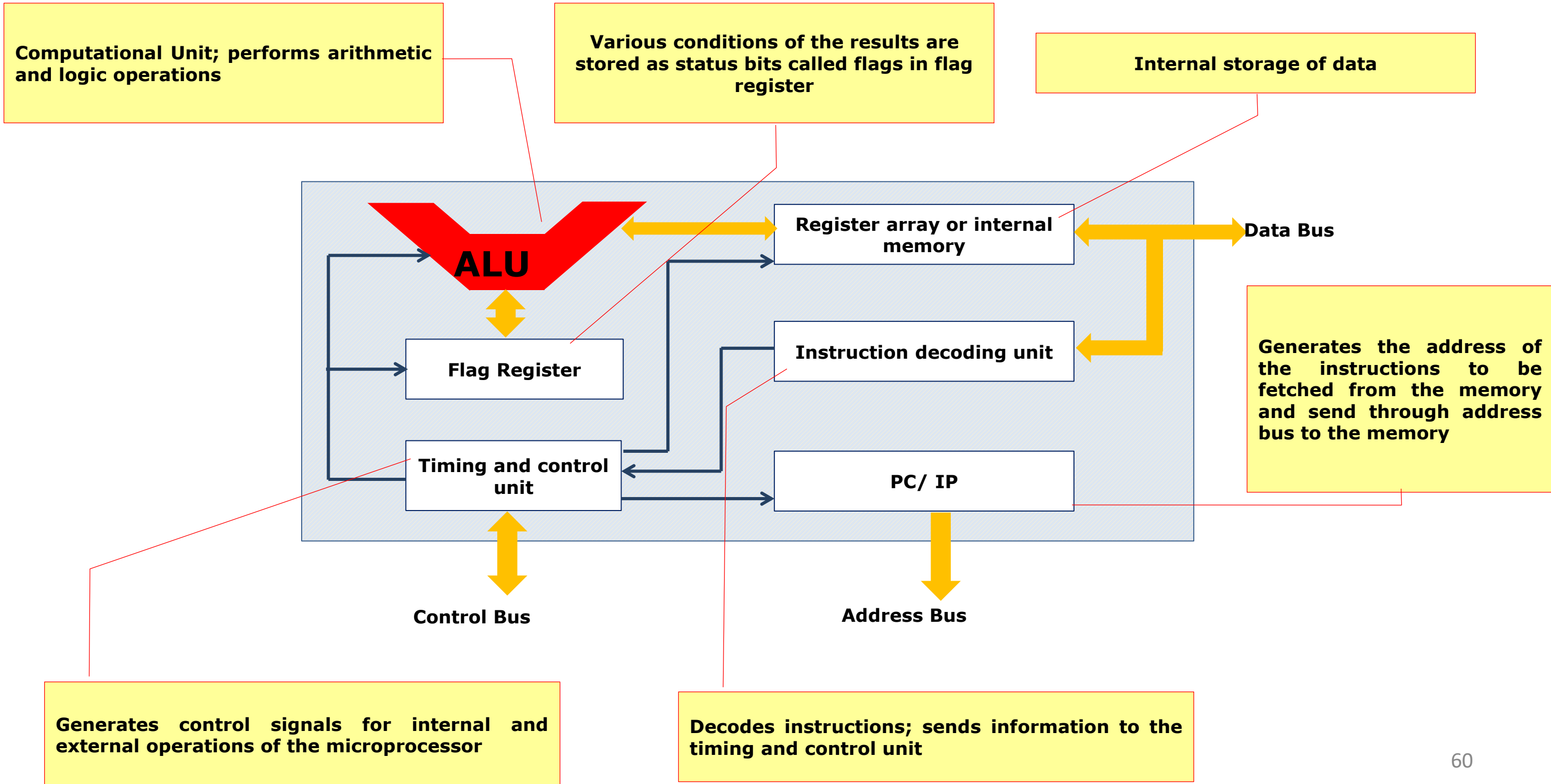
Geçici depolama alanıdır. CPU içinde bulunur.

İşlenecek komutların transfer edildiği geçici saklayıcılardır:

- Data Registers
 - Pointer and Index registers,
 - Segment registers,
 - Program counter,
 - Flags
- Kontrol birimi ve aritmetik/mantık birimi. Bu iki bileşen işlemci içinde sistem veriyolu adı verilen bir çeşit elektronik bağ ile birleştirilir. Veriyolu aynı zamanda bu bileşenleri bilgisayar sisteminin bellekleri ve I/O birimleri ile birleştirir.



Functional blocks

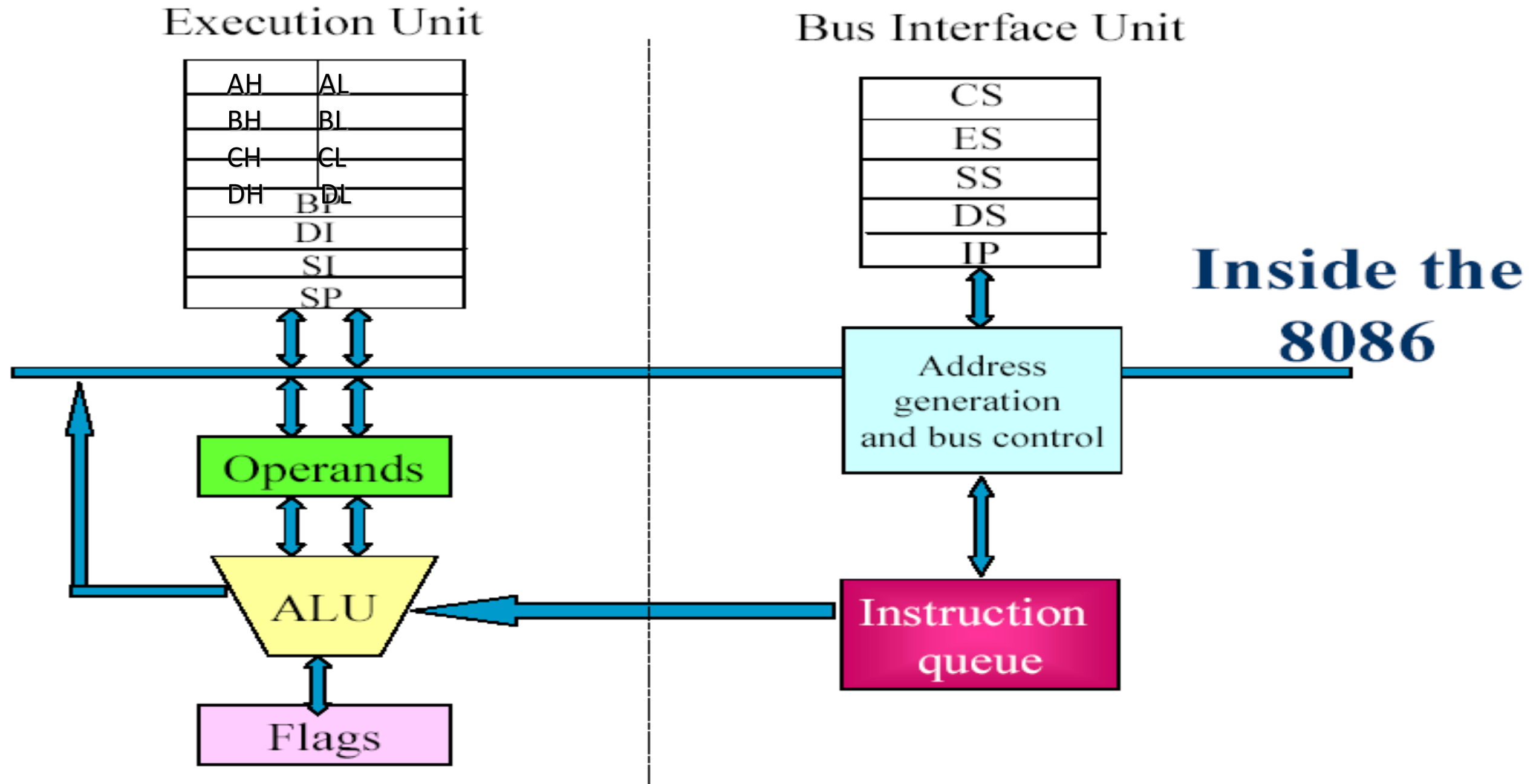


Inside X86...*pipelining*

Intel implemented the concept of pipelining by splitting the internal structure of the X86 into two sections that works simultaneously:

- **Execution Unit (EU)** – executes instructions previously fetched
- **Bus Interface Unit (BIU)** – accesses memory and peripherals

Inside X86



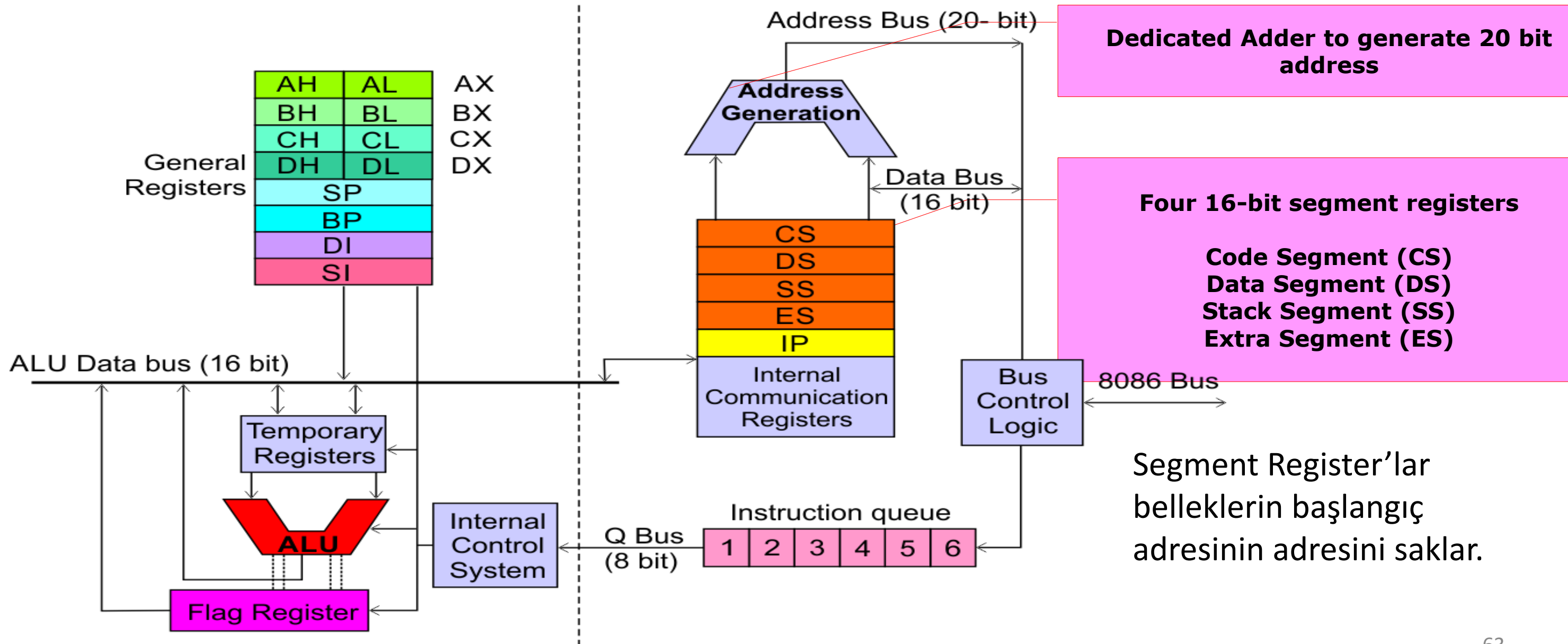
Execution Unit (EU)

EU executes instructions that have already been fetched by the BIU.

BIU and EU functions separately.

Bus Interface Unit (BIU)

BIU fetches instructions, reads data from memory and I/O ports, writes data to memory and I/O ports.





Registers

Data registers

D0	
D1	
D2	
D3	
D4	
D5	
D6	
D7	

Address registers

A0	
A1	
A2	
A3	
A4	
A5	
A6	
A7	

Program status

Program counter
Status register

(a) MC68000

General registers

AX	Accumulator
BX	Base
CX	Count
DX	Data

Pointers & index

SP	Stack ptr
BP	Base ptr
SI	Source index
DI	Dest index

Segment

CS	Code
DS	Data
SS	Stack
ES	Extrat

Program status

Flags
Instr ptr

(b) 8086

General Registers

EAX		AX
EBX		BX
ECX		CX
EDX		DX

ESP		SP
EBP		BP
ESI		SI
EDI		DI

Program Status

FLAGS Register
Instruction Pointer

(c) 80386 - Pentium 4

Figure 14.3 Example Microprocessor Register Organizations

- It is instructive to examine and compare the register organization of comparable systems. In this section, we look at two 16-bit microprocessors that were designed at about the same time: the Motorola MC68000 [STR179] and the Intel 8086 [MORS78]. Figures 14.3a and b depict the register organization of each; purely internal registers, such as a memory address register, are not shown.
- The MC68000 partitions its 32-bit registers into eight data registers and nine address registers. The eight data registers are used primarily for data manipulation and are also used in addressing as index registers. The width of the registers allows 8-, 16-, and 32-bit data operations, determined by opcode. The address registers contain 32-bit (no segmentation) addresses; two of these registers are also used as stack pointers, one for users and one for the operating system, depending on the current execution mode. Both registers are numbered 7, because only one can be used at a time. The MC68000 also includes a 32-bit program counter and a 16-bit status register.
- The Intel 8086 takes a different approach to register organization. Every register is special purpose, although some registers are also usable as general purpose. The 8086 contains four 16-bit data registers that are addressable on a byte or 16-bit basis, and four 16-bit pointer and index registers. The data registers can be used as general purpose in some instructions. In others, the registers are used implicitly. For example, a multiply instruction always uses the accumulator. The four pointer registers are also used implicitly in a number of operations; each contains a segment offset. There are also four 16-bit segment registers. Three of the four segment registers are used in a dedicated, implicit fashion, to point to the segment of the current instruction (useful for branch instructions), a segment containing data, and a segment containing a stack, respectively. These dedicated and implicit uses provide for compact encoding at the cost of reduced flexibility. The 8086 also includes an instruction pointer and a set of 1-bit status and control flags.
- The point of this comparison should be clear. There is no universally accepted philosophy concerning the best way to organize processor registers [TOON81]. As with overall instruction set design and so many other processor design issues, it is still a matter of judgment and taste.
- A second instructive point concerning register organization design is illustrated in Figure 14.3c. This figure shows the user-visible register organization for the Intel 80386 [ELAY85], which is a 32-bit microprocessor designed as an extension of the 8086. The 80386 uses 32-bit registers. However, to provide upward compatibility for programs written on the earlier machine, the 80386 retains the original register organization embedded in the new organization. Given this design constraint, the architects of the 32-bit processors had limited flexibility in designing the register organization.
-

Registers

- It is a special temporary storage location within the CPU.
- Registers quickly accept, store and transfer data and instructions that are being used immediately.
- To execute an instruction, the control unit of the CPU retrieves it from main memory and places it onto a register.
- The typical operations that take place in the processing of instruction are part of the instruction cycle or execution cycle.
- The instruction cycle refers to the retrieval of the instruction from main memory and its sub sequence at decoding.
- The time it takes to go through the instruction cycle is referred to as I-time.

Registers

- Special-purpose
- High-speed
- Temporary storage
- Located inside CPU

Mikroişlemcinin iç yapısında,

- Veri işleme ve Veri iletişim ara yüzünde kullanılır.
- Özel amaçlı saklayıcıdır.
- Yüksek hızlı veri işleme ve transfer etme hızlı rol oynar.
- Geçici depolama alanıdır.
- CPU içinde bulunur. CPU'nun ana bileşenidir.

Registers

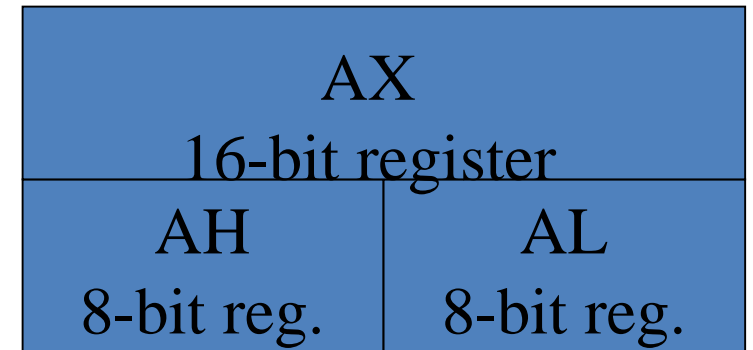
- Registers are used to store information temporarily
- 8086, 8088, 80286 contains 8-bit, 16-bit registers
- 80386, 80486, Pentium, Pentium Pro, and Pentium II contain 8, 16, and 32-bit registers.
- How about Pentium 4?

Overview

- Registers
 - General purpose registers (8)
 - Operands for logical and arithmetic operations
 - Operands for address calculations
 - Memory pointers
 - Segment registers (6)
 - EFLAGS register
 - The instruction pointer register
- The stack

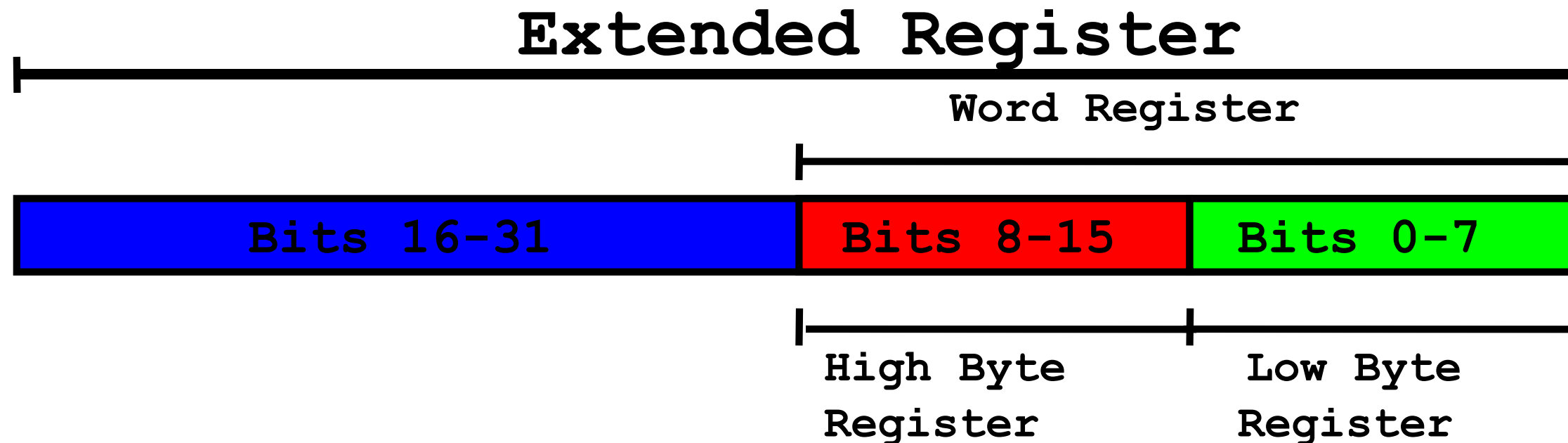
Inside X86...registers

- **Registers**
 - To store information temporarily



Category	Bits	Register Names
General	16	AX, BX, CX, DX
	8	AH, AL, BH, BL, CH, CL, DH, DL
Pointer	16	SP (stack pointer), BP (base pointer)
Index	16	SI (source index), DI (destination index)
Segment	16	CS (code segment), DS (data segment) SS (stack segment), ES (extra segment)
Instruction	16	IP (instruction pointer)
Flag	16	FR (flag register)

Anatomy of a Register



In today's 32-bit architecture, each register has 32 bits that can be used. However, some registers can be accessed as a single byte, and all registers can be accessed as a 16-bit value, or a 32-bit value. This is further explained on the next slide.

General Registers

32 bit Registers		16 bit Registers		8 bit Registers	
EAX	EBP	AX	BP	AH	AL
EBX	ESI	BX	SI	BH	BL
ECX	EDI	CX	DI	CH	CL
EDX	ESP	DX	SP	DH	DL
Bits 16-31		Bits 8-15		Bits 0-7	

Registers

6 Category

- General
- **Pointer**
- Index
- Segment
- Instruction
- Flag

EAX		AH	AX	AL
EBX		BH	BX	BL
ECX		CH	CX	CL
EDX		DH	DX	DL
EBP		BP		
ESI		SI		
EDI		DI		
ESP		SP		
EIP		IP		
EFLAGS		FLAGS		
		CS		
		DS		
		ES		
		SS		
		FS		
		GS		

Register names

- Accumulator
 - Base index
 - Count
 - Data
 - Stack Pointer
 - Base Pointer
 - Destination index
 - Source index
 - Instruction Pointer
 - Flags
- Segment registers
- Code
 - Data
 - Extra
 - Stack

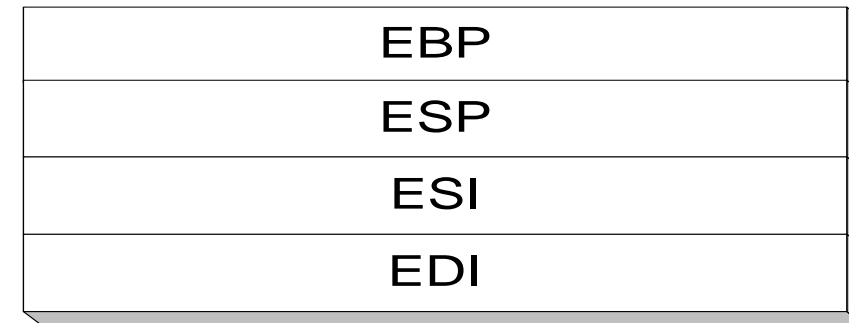
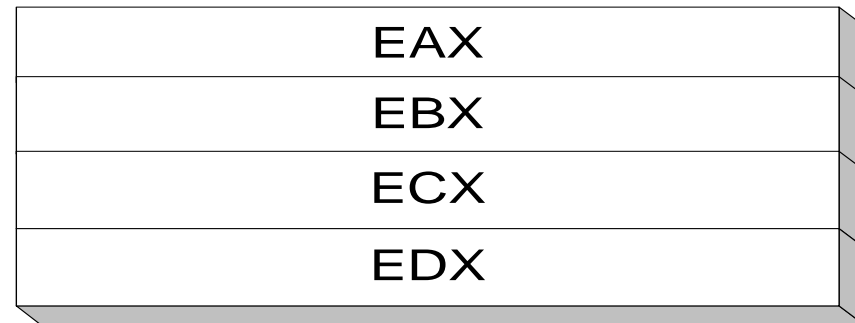
Some Specialized Register Uses (1 of 2)

- General-Purpose
 - EAX – accumulator
 - ECX – loop counter
 - ESP – stack pointer
 - ESI, EDI – index registers
 - EBP – extended frame pointer (stack)
- Segment
 - CS – code segment
 - DS – data segment
 - SS – stack segment
 - ES, FS, GS - additional segments

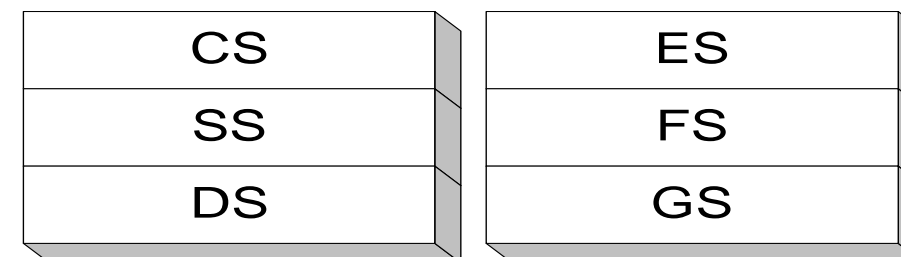
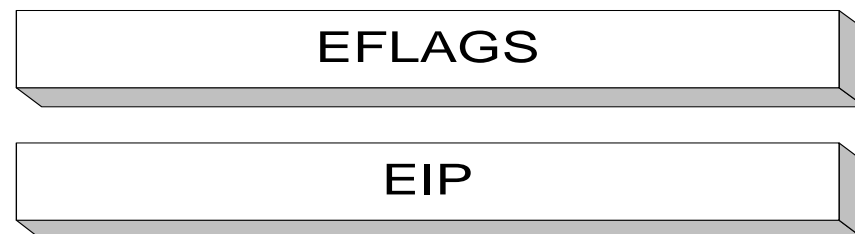
General-Purpose Registers

Named storage locations inside the CPU, optimized for speed.

32-bit General-Purpose Registers

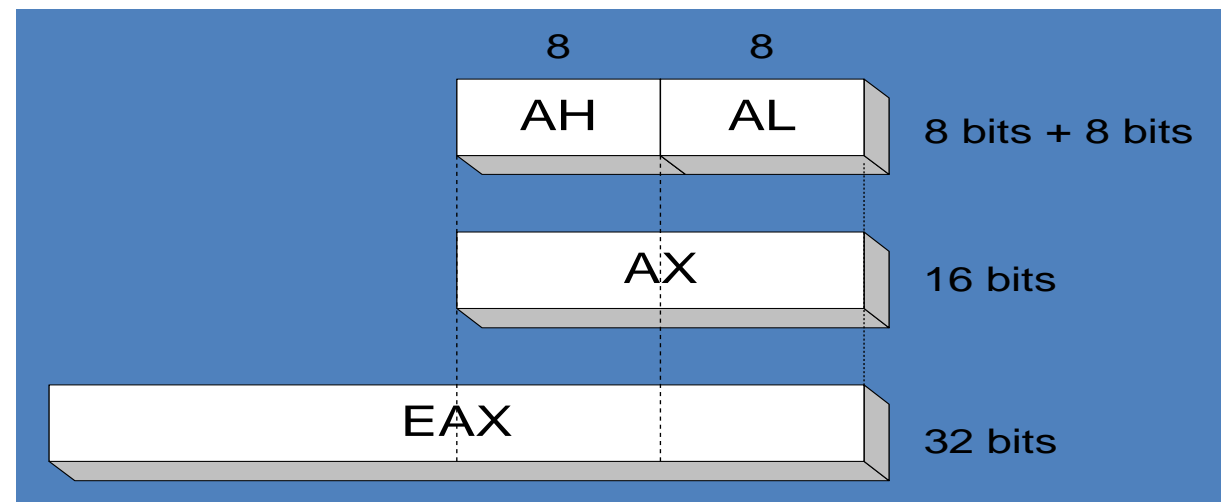


16-bit Segment Registers



Accessing Parts of Registers

- Use 8-bit name, 16-bit name, or 32-bit name
- Applies to EAX, EBX, ECX, and EDX



32-bit	16-bit	8-bit (high)	8-bit (low)
EAX	AX	AH	AL
EBX	BX	BH	BL
ECX	CX	CH	CL
EDX	DX	DH	DL

General Registers I

- **EAX** – ‘Accumulator’
 - accumulator for operands and results data
 - usually used to store the return value of a procedure
- **EBX** – ‘Base Register’
 - pointer to data in the DS segment
- **ECX** – ‘Counter’
 - counter for string and loop operations
- **EDX** – ‘Data Register’
 - I/O pointer

Index and Base Registers

- Some registers have only a 16-bit name for their lower half:

32-bit	16-bit
ESI	SI
EDI	DI
EBP	BP
ESP	SP

General Registers II

- **ESI** – ‘Source Index’
 - source pointer for string operations
 - typically a pointer to data in the segment pointed to by the DS register
- **EDI** – ‘Destination Index’
 - destination pointer for string operations
 - typically a pointer to data/destination in the segment pointed to by the ES register

General Registers III

- **EBP** – ‘Base Pointer’
 - pointer to data on the stack
 - points to the current stack frame of a procedure
- **ESP** – ‘Stack Pointer’
 - pointer to the top address of the stack
 - holds the stack pointer and as a general rule should not be used for any other purpose

Segment Registers

- **CS** – ‘Code Segment’
 - contains the segment selector for the code segment where the instructions being executed are stored
- **DS (ES , FS , GS)** – ‘Data Segment’
 - contains the segment selectors for the data segment where data is stored
- **SS** – ‘Stack Segment’
 - contains the segment selector for the stack segment, where the procedure stack is stored

The EFLAGS Register I

- Carry Flag – CF (bit 0)
 - **Set** if an arithmetic operation generates a carry or a borrow out of the most-significant bit of the result; **cleared** otherwise.
- Parity Flag – PF (bit 2)
 - **Set** if the least-significant byte of the result contains an even number of 1 bits; **cleared** otherwise.
- Adjust Flag – AF (bit 4)
 - Set if an arithmetic operation generates a carry or a borrow out of bit 3 of the result; **cleared** otherwise.

Flags

- The status flags (bits 0, 2, 4, 6, 7, and 11) of the EFLAGS register indicate the results of arithmetic instructions, such as the ADD, SUB, MUL, and DIV instructions. The EFLAGS register is modified automatically by the CPU after mathematical operations. Generally you cannot access these registers directly.
- We will see that some instructions such as conditional jumps rely on the contents of the EFLAGS register to make decisions.
- The carry flag indicates an overflow condition for unsigned-integer arithmetic. It is also used in multiple-precision arithmetic.
- The adjust flag is used in binary-coded decimal (BCD) arithmetic.
- Of the six status flags we define on the slides, only the CF flag can be modified directly, using the STC, CLC, and CMC instructions. The bit instructions (BT, BTS, BTR, and BTC) copy a specified bit into the CF flag. See the Intel manual for details.

The EFLAGS Register II

- Zero Flag – ZF (bit 6)
 - **Set** if the result is zero; **cleared** otherwise
- Sign Flag – SF (bit 7)
 - **Set** equal to the most-significant bit of the result, which is the sign bit of a signed integer
- Overflow Flag – OF (bit 11)
 - **Set** if the integer result is too large a positive number or too small a negative number (excluding the sign-bit) to fit in the destination operand; **cleared** otherwise

Flags

- With respect to the SF flag, 0 indicates a positive value and 1 indicates a negative value.
- The OF flag indicates an overflow condition for signed-integer (two's complement) arithmetic.
- There are other flags contained in the EFLAGS register are:
- Bit 9 – Interrupt enable flag
- Bit 10 – Direction flag. This flag determines whether the EDI/ESI registers will be incremented or decremented during string operations.
- Bit 11 – Overflow flag
- Bits 12-13 – IOPL I/O Privilege level
- Bit 14 – NT Nested task flag
- Bit 16 – RF Resume flag
- Bit 17 – VM Virtual 8086 mode flag
- Bit 18 – AC Alignment check flag (486+)
- Bit 19 – VIF Virtual interrupt flag
- Bit 20 – VIP Virtual interrupt pending flag
- Bit 21 – ID flag
- Bits in the EFLAGS register that we have not listed are reserved by Intel.

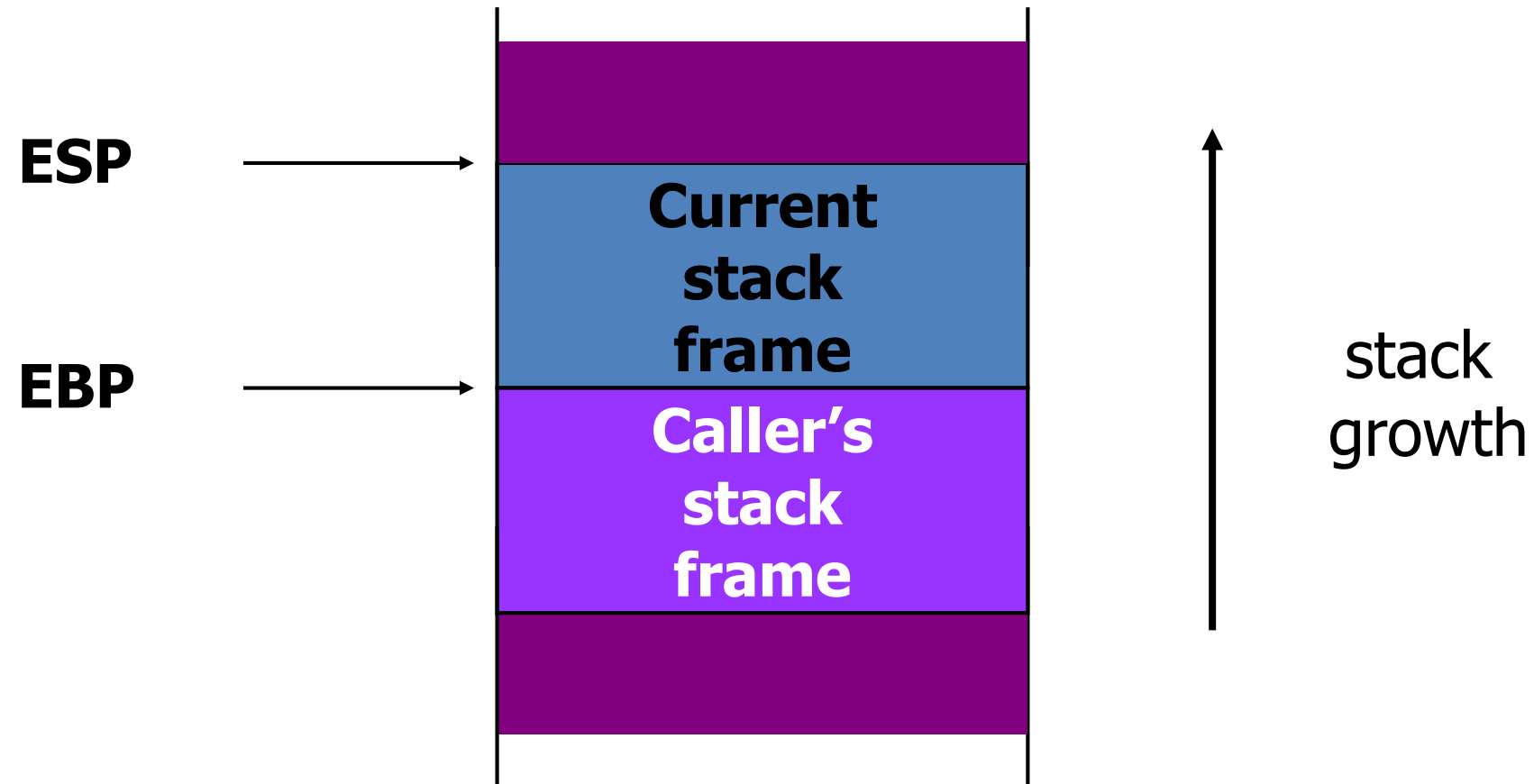
Instruction Pointer

- **EIP**
 - ‘Instruction Pointer’
 - Contains the offset within the code segment of the next instruction to be executed
 - Cannot be accessed directly by software
 - The processor fetches instructions from the code segment, using a logical address that consists of the segment selector in the CS register and the contents of the EIP register. The EIP register contains the offset within the code segment of the next instruction to be executed.
 - The EIP register cannot be accessed directly by software; it is controlled implicitly by control transfer instructions (e.g., procedure calls or jump statements), interrupts, and exceptions.
 - The only way to read the EIP register is to execute a CALL instruction and then read the value of the return instruction pointer from the procedure stack.
 - The EIP register can be loaded indirectly by modifying the value of a return instruction pointer on the procedure stack and executing a return instruction.

The Stack

The stack starts in high memory and grows toward low memory

- In the Intel architecture, the stack grows toward low memory (i.e., from addresses with larger numbers to addresses with smaller numbers).
- At the beginning of a routine, the base pointer is saved on the stack and then it is set to equal the current stack pointer. The stack pointer is then adjusted to make room for the subroutine's local variables. Such an area on the stack is called a "stack frame".





ALU

Arithmetic Logic Unit (ALU)

- ALU performs all the arithmetic and logical functions.
- It performs arithmetic as well as logical functions.
- The speed of the computer system is defined by the architecture of the processor being used.

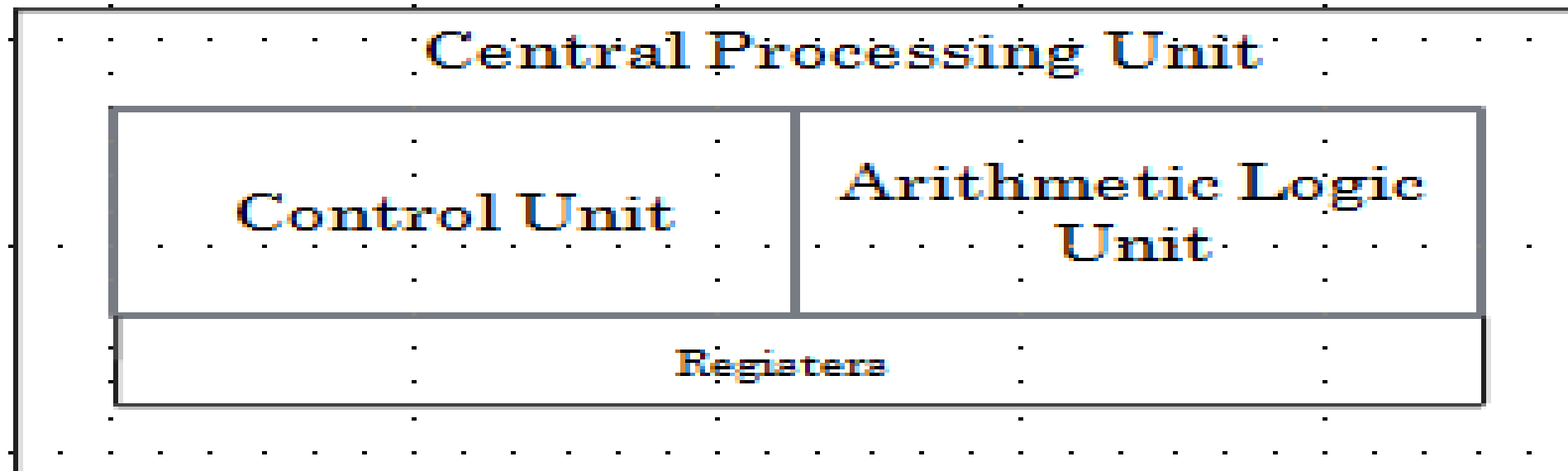
ALU

- Indicating a simplified view of a processor, connection to the rest of the system via the system bus.
- A similar interface would be needed for any of the interconnection structures.
- The reader will recall that the major components of the processor are an arithmetic and logic unit (ALU) and a control unit (CU).
- The ALU does the actual computation or processing of data. The control unit controls the movement of data and instructions into and out of the processor and controls the operation of the ALU.
- In addition, the figure shows a minimal internal memory, consisting of a set of storage locations, called registers.

Arithmetic Logic unit

- The ALU contains electronic circuits necessary to perform arithmetic and logical operations.
- The arithmetic operations are ADD, SUBSTRACT, MULTIPLY, DIVIDE, etc.
- The logical operations include COMPARE, SHIFT, ROTATE, AND, OR, etc
- The control unit analyses each instruction in the program and sends the relevant signals to all other units – ALU, Memory, Input unit and Output unit

Central Processing Unit (CPU)



Control Unit

- It is responsible for directing and coordinating most of the computer system activities.
- It does not execute instructions by itself. It tells other parts of the computer system what to do.
- It determines the movement of electronic signals between the main memory and arithmetic logic unit as well as the control signals between the CPU and input/output devices.
- To complete an event i.e. processing, control unit repeats a set of four basic operations:
- Fetching is the process of obtaining a program instruction or data item from the memory
- Decoding is the process of translating the instruction into commands the computer can execute.
- Executing is the process of carrying out the commands.
- Storing is the process of writing the result to memory.

Control Unit

- The internal communication inside a computer that transforms raw data into useful information is called **processing**.
- To perform this transformation, the computer uses two components- processor and memory
- The program is fed into the computer through the input unit and stored in the memory
- To execute the program, the instructions have to be fetched from memory one by one which is done by control unit
- Then the control unit decodes the instruction.
- According to instruction, control unit issues signals to other units.
- After instruction is executed, the result of the instruction is stored in memory or stored temporarily in the registry, so that this can be used by the next instruction.
- The results of a program are taken out of the computer through the output unit.



Main Memory

Real-Address mode

- 1 MB RAM maksimum adreslenebilir
- Uygulama programları hafızanın herhangi bir alanına erişebilir
- Tek görev
- MS-DOS işletim sistemi tarafından desteklenir

Byte - Memory sizes

- Byte: 8bit'lik veriyi temsil eder.
- 8-bitlik bellek gözünü refere (işaret) eder.
- We refer to this as a $k \times n$ memory.
 - There are k *address lines*, which can specify one of 2^k addresses.
 - Each address contains an n -bit word.

For example: a 26×16 RAM contains $2^{26} = 64\text{M}$ words, each memory cell is 16 bits long.

- The RAM would need 26 address lines.
- The total bit *storage capacity* is $2^{26} \times 16 \text{ bit} = 2^{27} \times 8 \text{ bit} = 2^{27} \text{ byte} = 2^{30} \text{ bits}$.
- Adres bus hat sayısı=26 adet.
- Hücresel uzunluğu word (16bit) olan belleğin indislenmesi : A25, A23, A22, ..., A1, A0

Bellek

- **Bellek:** Mikroişlemcinin işlediği veya manipüle ettiği verileri saklayan transistör devre elemanlarından oluşan veri saklama özelliği olan ardışıl lojik kapılardır. Bellekler ikili sayı sisteminde 1 veya 0 formatlarından oluşan komut ya da veri gibi bilgileri saklar.
- **Bellekler: Verileri saklamak için kullanılır.** Mikroişlemcilere komutları ya da verileri transfer etmek için gerekli devrelerle birlikte bir depolama hücreleri koleksiyonudur.
- **Transistör:** Farklı bir noktadan elektron akışını kontrol ederek (akım, gerlimi) sinyalleri anahtarlama yada kuvvetlendirme işlevlerini yerine getiren, yarı iletken teknolojisinde üretilen bir devre elemanıdır. Transistörler atomik yapıda üretilebilmektedir.
- **Register:** Geçici özel amaçlı saklayıcılardır. Mikroişlemcinin içinde en temel saklayıcı birimdir.
- **ROM bellek:** Değiştirilmeyen komut ve verileri saklamak için kullanılır. Sadece okunur bellektir.
- **RAM bellek:** Hem yazılan hem de okunan bellektir.
- **Ön Bellek (Cache):** Mikroişlemcinin sonraki adımlarda işleyeceği verilerin önceden transfer edilip hazırlandığı kendi ön belleğidir (SRAM).

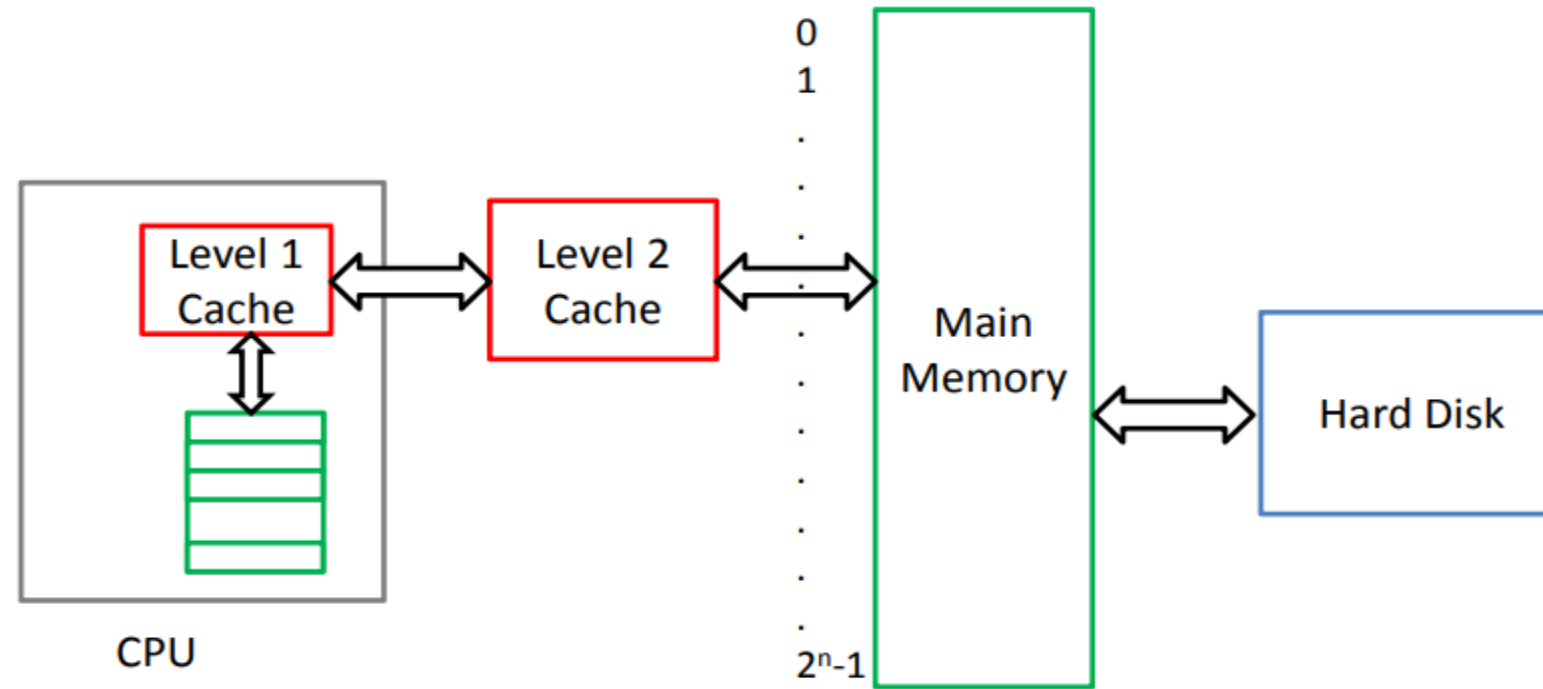
Memory

- ROM
 - read-only memory
- EPROM
 - erasable programmable read-only memory
- Dynamic RAM (DRAM)
 - inexpensive; must be refreshed constantly
- Static RAM (SRAM)
 - expensive; used for cache memory; no refresh required
- Video RAM (VRAM)
 - dual ported; optimized for constant video refresh
- CMOS RAM
 - complimentary metal-oxide semiconductor
 - system setup information

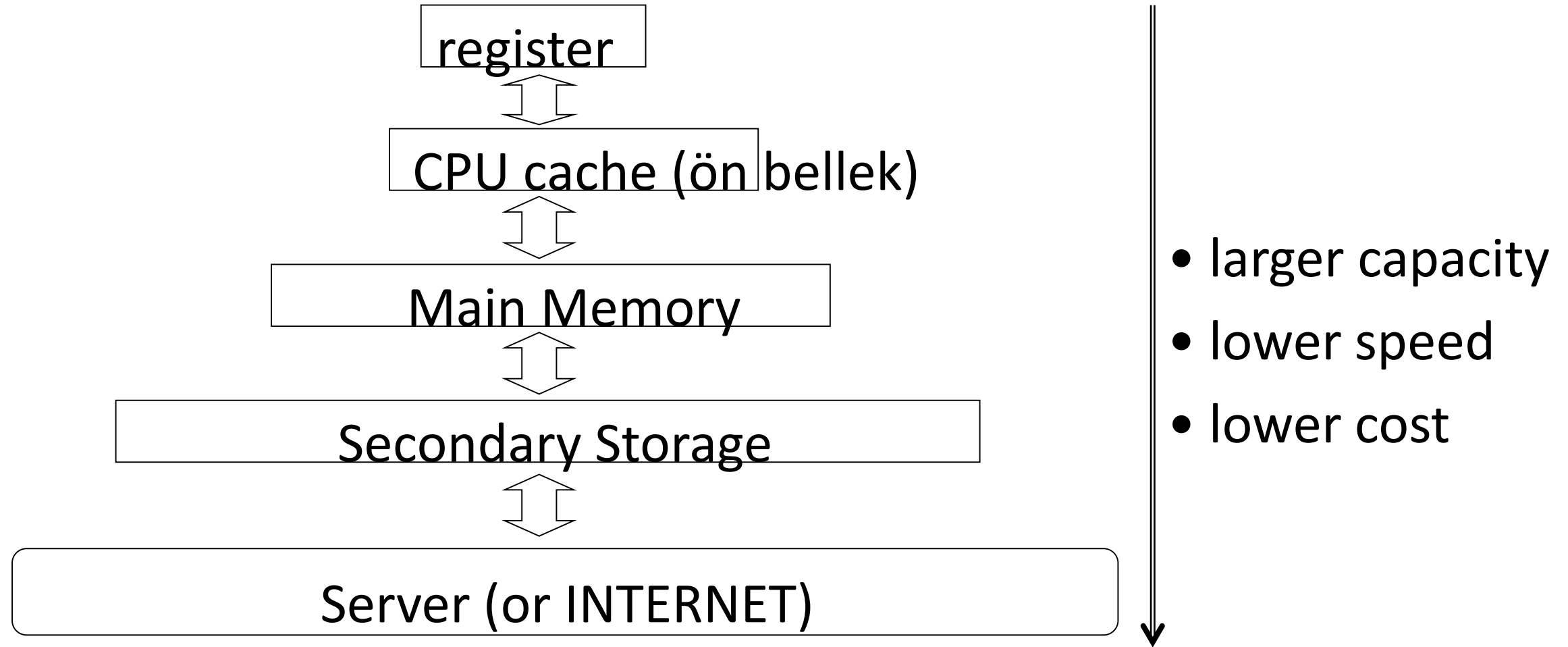
Memory Types

- Ana Bellek: Ram, Rom, CMOS (BIOS)
 - Cache
 - Dynamic ram
 - Static ram
 - Flash memory
 - Memory sticks
 - Virtual memory
 - Video memory
 - Bios
 - Hard Disk
- Belleklerin üç görevi vardır:
 - İşlenecek veriyi depolar.
 - Veriyi işleyen komutları (programları) depolar.
 - İşlenmiş, iletişim veya çıkış aygıtlarına gönderilmek için bekleyen veriyi depolar.

Memory Hierarchy



CPU – Bellek erişim özelliğine göre belleklerin sıralanması



Cache Memory

- High-speed, expensive, static RAM both inside and outside of the CPU.
 - Level-1 cache: inside the CPU
 - Level-2 cache: outside the CPU
- Cache hit: when data to be read is already in cache memory
- Cache miss: when data to be read is not in cache memory.

Memory Segmentation

- A memory segment is a block of 2^{16} (64K) bytes.
- Each segment is identified by a segment number
 - Segment number is 16 bits .
- A memory location is specified by an offset within a segment.
- Logical address: segment:offset
 - A000:4872h means offset 4872h within segment A000h.
- Physical address: segment * 10H + offset
 - $A000h * 10h + 4872h = A4872h$ (20-bit address)

Paging

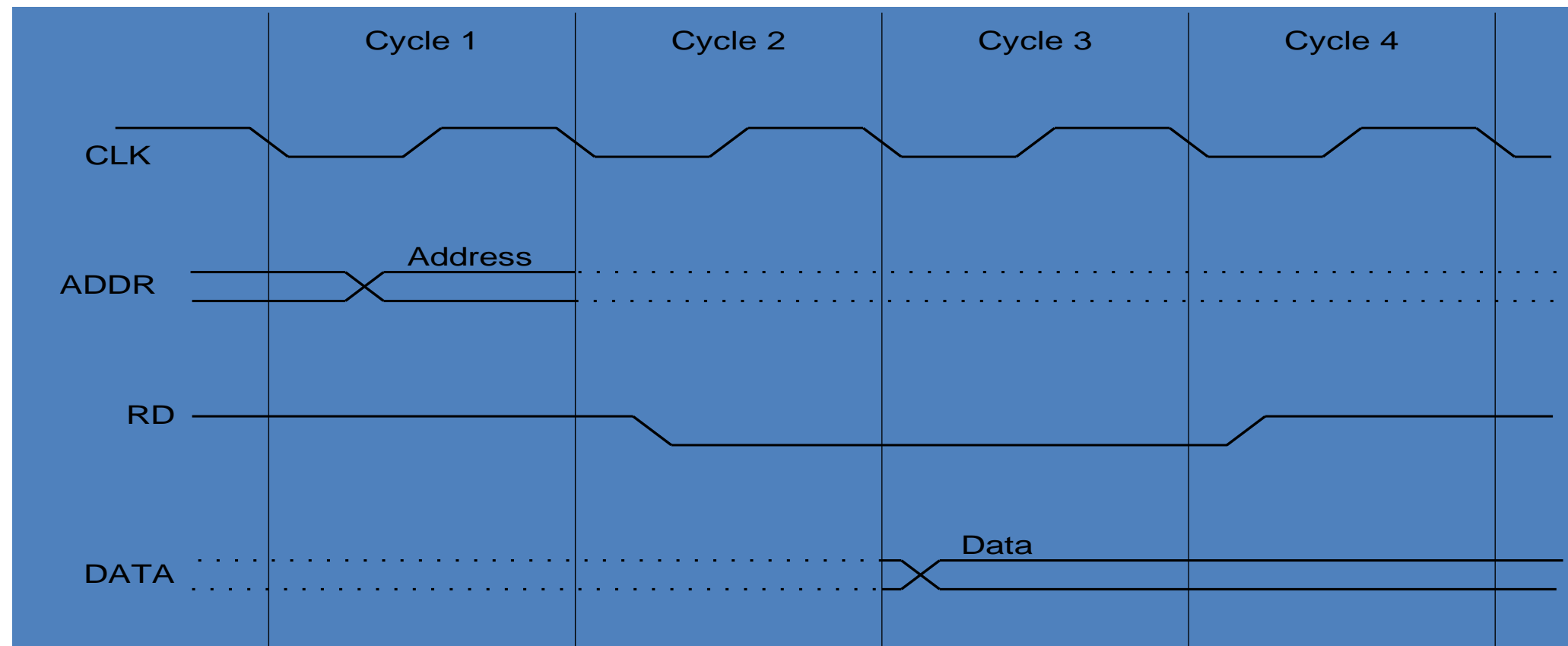
- Supported directly by the CPU
- Divides each segment into 4096-byte blocks called **pages**
- Sum of all programs can be larger than physical memory
- Part of running program is in memory, part is on disk
- **Virtual memory manager** (VMM) – OS utility that manages the loading and unloading of pages
- **Page fault** – issued by CPU when a page must be loaded from disk

IA-32 Memory Management

- Real-address mode
- Calculating linear addresses
- Protected mode
- Multi-segment model
- Paging
- Gerçek adres modu
- Doğrusal adresleri hesaplama
- Korumalı mod
- Çok segmentli model
- sayfalama

Reading from Memory

- Multiple machine cycles are required when reading from memory, because it responds much more slowly than the CPU. The steps are:
 - address placed on address bus
 - Read Line (RD) set low
 - CPU waits one cycle for memory to respond
 - Read Line (RD) goes to 1, indicating that the data is on the data bus

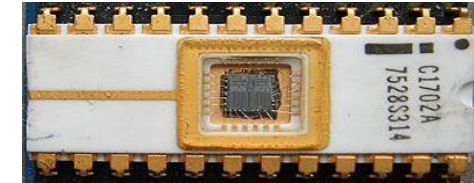


Types of Memory

- **Static RAM**
- **Dynamic RAM**
- **ROM**
- **PROM**
- **EPROM**
- **EEPROM**

Memory

- Elektronik dijital bilgisayarda kullanım için verileri veya programları (talimat dizileri) geçici veya kalıcı olarak depolamak için kullanılan fiziksel cihazlar.
- Bilgisayar ana belleği iki ana çeşitte gelir: rasgele erişimli bellek (RAM) ve salt okunur bellek (ROM).
- RAM, CPU komut verdiği her an okunabilir ve yazılabilir, ancak ROM, hiçbir zaman değişmeyen veri ve yazılımla önceden yüklenmiştir, bu nedenle CPU yalnızca ondan okuyabilir.
- ROM, genellikle bilgisayarın ilk başlatma talimatlarını depolamak için kullanılır.
- Genel olarak, bilgisayarın gücü kapatıldığında RAM içeriği silinir, ancak ROM verilerini süresiz olarak tutar.
- Bir PC'de, ROM, bilgisayar açıldığında veya sıfırlandığında bilgisayarın işletim sisteminin sabit disk sürücüsünden RAM'e yüklenmesini düzenleyen BIOS adlı özel bir program içerir.



Computer Memory

- Memory contains instructions for the processor to execute or data it operates on
- **Address Locations** - Memory consists of a sequential number of locations, each of which are a specific number of bits wide.
 - byte wide memory 8 bits (PC-8088)
 - 16 bits (XT-8086, AT-80286)
 - 32 bits (386DX, 486SX, 486DX)
 - 64 bits (Modern systems – Pentium and up)
- Each memory location is referred to as an **address**, and generally expressed in hexadecimal notation (using base 16 numbers).
- The size is denoted as the number of locations times the number of bits in each location
- The processor selects a specific address in memory by placing the address on the **address bus** . The value on this address bus is used by the memory system to find the data at the specific location

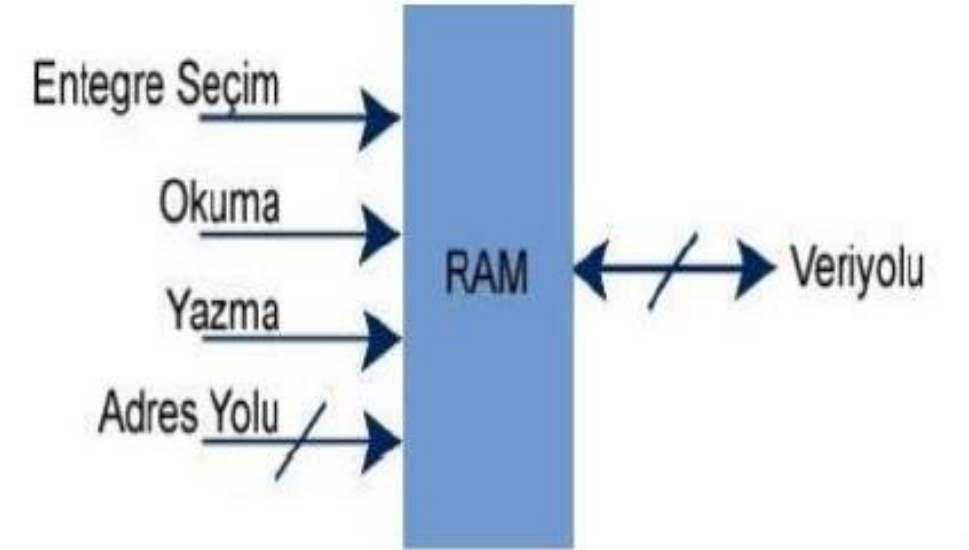
Computer Memory

- The total number of address locations which can be accessed by the processor is known as its **physical address space**. How large this is determined by the size of the address bus, and is often expressed in terms of Kilobytes (x1024), Megabytes or Gigabytes.
 - 16 bits address bus = 64K (65536 locations)
 - 20 bits address bus = 1MB (IBM PC)
 - 32 bits address bus = 4GB (486DX)
- **Access Times** - Access time refers to how long it takes the processor to read or write to a specific memory location within a chip. The limiting factor is the type of technology used to implement the memory cells inside the chip.
- **Volatility** - This refers to whether or not the contents of the memory is lost when power is turned off. If the contents are lost, the memory is *volatile*. If the contents are retained, then the memory is *non-volatile*.

Bellekler

1)RAM Bellekler : Mikro işlemcinin çalışması esnasında her türlü değişkenin üzerinde yer aldığı ve

- geçici işlemlerin yapıldığı birimi RAM belleklerdir. Özel bir sıra takip etmeden herhangi bir adrese erişildiği için **rastgele erişimli bellek** (Random Access Memory)–RAM olarak isimlendirilir. RAM tipi entegreler hem yazmada hem okumada kullanıldıklarından CPU, bu entegreleri kontrol ederken okuma R (Okuma) ve W (yazma) sinyalleri göndermesi gerekir. Ayrıca entegrenin istendiği zaman aktif duruma geçmesinin sağlayacak entegre seçimi (CS =Chip Select) pini bulunmaktadır ve active low (aktif düşük, 0 Volt) ile çalışır. Her bir biti bir flip-flop devresi olan bu bellekler, yeni bir tetikleme işareti gelinceye
- kadar içindeki bilgiyi (0 veya 1'i) saklayabilme özelliği sebebiyle **çok düşük güç tüketimi** ile çalışmaktadır. Dışardan devreye bağlanan bir pil yardımıyla içindeki bilgileri çok uzun süreler boyunca saklayabilme imkânı vardır. Yüksek maliyetli olmaları sebebiyle çok yüksek kapasitelerde üretilmez.



Bellekler

2)ROM Bellekler : **Yalnız okunabilen** birimlere ROM (Read Only Memory) bellekler denir. Bu bellek elemanlarının en büyük özelliği enerjisi kesildiğinde içindeki bilgilerin silinmemesidir. ROM belleklere bilgiler üretim aşamasında yüklenir. Kullanıcıların bellek içindeki bilgileri değiştirmesi mümkün değildir.

3)PROM (Programlanabilir ROM) Bellekler : PROM'lar **bir kez programlanabilir**. Bu bellek elemanı entegre şeklindedir. Kaydedilen bilgiler enerji kesildiğinde silinmez. Üzerine program kodlarını veya verileri yazmak için PROM programlayıcı cihazlara ihtiyaç vardır. Bu bellek elemanının yapısında küçük sigorta telleri bulunur. Bellek hücrelerinde hepsi sağlam durumda bulunan sigortalar "1"i temsil eder. Yazılacak olan bilginin bit düzeninde "0"lara karşılık gelen hücredeki sigorta, küçük bir elektrik akımı ile aktarılır. Bu şekilde PROM programlanır.

4)EPROM (Silinebilir Programlanabilir ROM) Bellekler : "EPROM"lar bellek hücrelerine elektrik sinyali uygulanarak programlama işlemi yapılır. Kaydedilen bilgiler enerji kesildiğinde silinmez. "EPROM" içindeki programın silinmemesi için cam pencereli kısım ışık geçirmeyen bantla örtülmelidir. EPROM belleğe yeniden yazma işlemi yapmak için "EPROM" üzerindeki bant kaldırılıp ultraviyole altında belirli bir süre tutmak gerekir. Bu şekilde içindeki bilgiler silinebilir. Böylece tekrar programlanabilir hâle gelen ürün tekrar tekrar farklı programların denenmesi ve cihazın çalıştırılması için kullanılabilir.

5)EEPROM (Elektriksel Yolla Değiştirilebilen ROM) Bellekler : Üzerindeki bilgiler, **elektriksel olarak yazılabilen ve silinebilen** bellek elemanlarıdır. "EEPROM"u besleyen enerji kesildiğinde üzerindeki bilgiler kaybolmaz. "EEPROM"daki bilgilerin silinmesi ve yazılması için özel silme ve yazma cihazlarına gerek yoktur. Programlayıcılar üzerinden gönderilen elektriksel sinyalle programlanır. "EEPROM"la aynı özellikleri taşıyan fakat yapısal olarak farklı ve daha hızlı olan, elektriksel olarak değiştirilebilir "ROM"lara Flash Bellek denir.

Storage Systems

- Units of Storage:
 - 1 bit
 - 8 bits = 1 byte
 - 1kbyte = 2^{10} = 1024 bytes
 - 1Mbyte = 2^{20} = 1048576 bytes
- Memory (RAM, ROM)
- Optical Disks
- CD ROM
- Magnetic Disk
- Floppy Disk
- Flash Bellek
- Hard Disk

Memory

- Memory is the computer's electronic scratchpad or local store in computer terminology.
- Used for temporary storage of calculations, data, and other work in progress.
- Two types: Primary and Secondary
- Primary memory or the main memory is part of the main computer system. The primary memory itself is of two types.
- The first is called random access memory (RAM) and the other is read only memory (ROM).

Random Access Memory (RAM)

- The processor directly stores and retrieves information from it.
- Memory is organized into locations. Each memory location is identified by a unique address. The access time is same for all location.
- It is volatile: when turned off, everything in RAM disappears.
- Two types:

Types of RAM

- **Dynamic Random Access Memory (DRAM):**
This type RAM retain the content of any location only for a few milliseconds. Within that period, each location must be written again with the same contents. This is known as refreshing.
- **Static Random Access Memory (SRAM):**
This type of RAM preserves the contents of all the locations as long as the power supply is present. SRAM is generally included in a computer system by the name of cache.

Read Only Memory (ROM)

- Data stored in ROM cannot be modified, or can be modified only slowly or with difficulty, so it is mainly used to distribute.
- The instructions in ROM are built into the electronic circuits of the chip which is called firmware.
- Random access in nature and non-volatile.

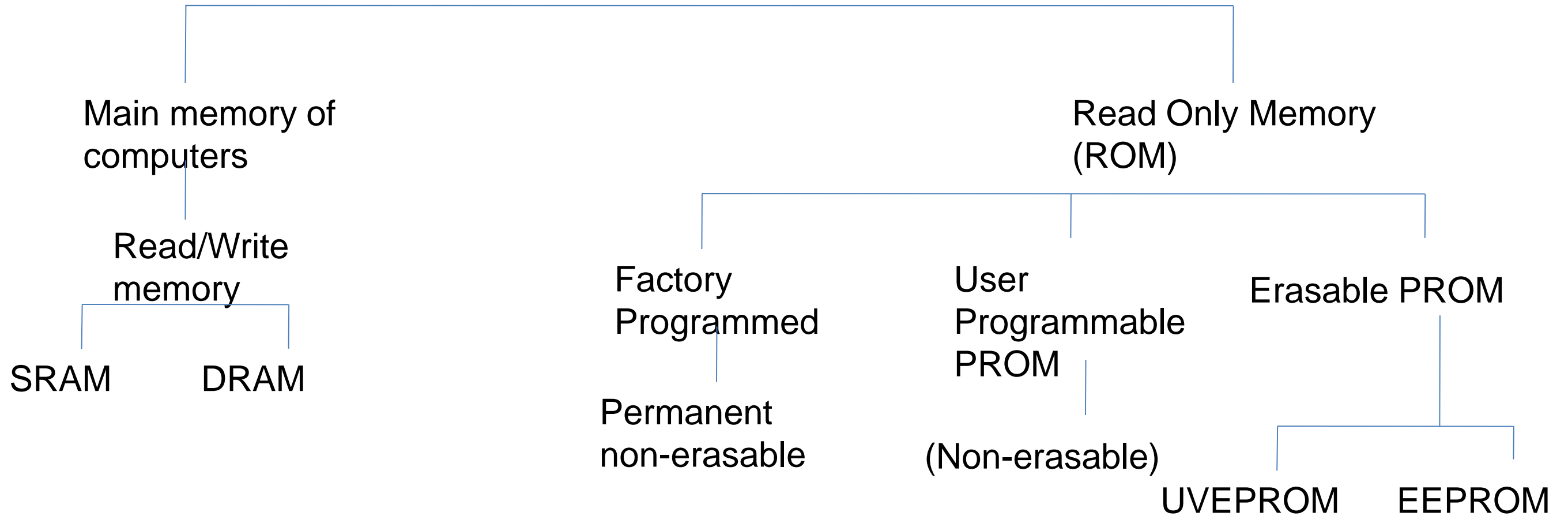
Types of ROM

- Programmable read-only memory (PROM), or one-time programmable ROM can be written to or programmed via a special device called a PROM programmer.
- Erasable programmable read-only memory (EPROM) can be erased by exposure to strong ultraviolet light then rewritten with a process that again needs higher than usual voltage applied.
- Electrically erasable programmable read-only memory (EEPROM) is based on a similar semiconductor structure to EPROM, but allows its entire contents (or selected banks) to be electrically erased, then rewritten electrically, so that they need not be removed from the computer

Flash Memory

- Modern type of EEPROM invented in 1984.
- Random access memories and are non-volatile.
- Use one transistor per memory cell and come in capacities ranging from 1 MB to 32 GB by the year 2007.
- The read time is much smaller (tens of nanoseconds) compared write time (tens of microseconds).

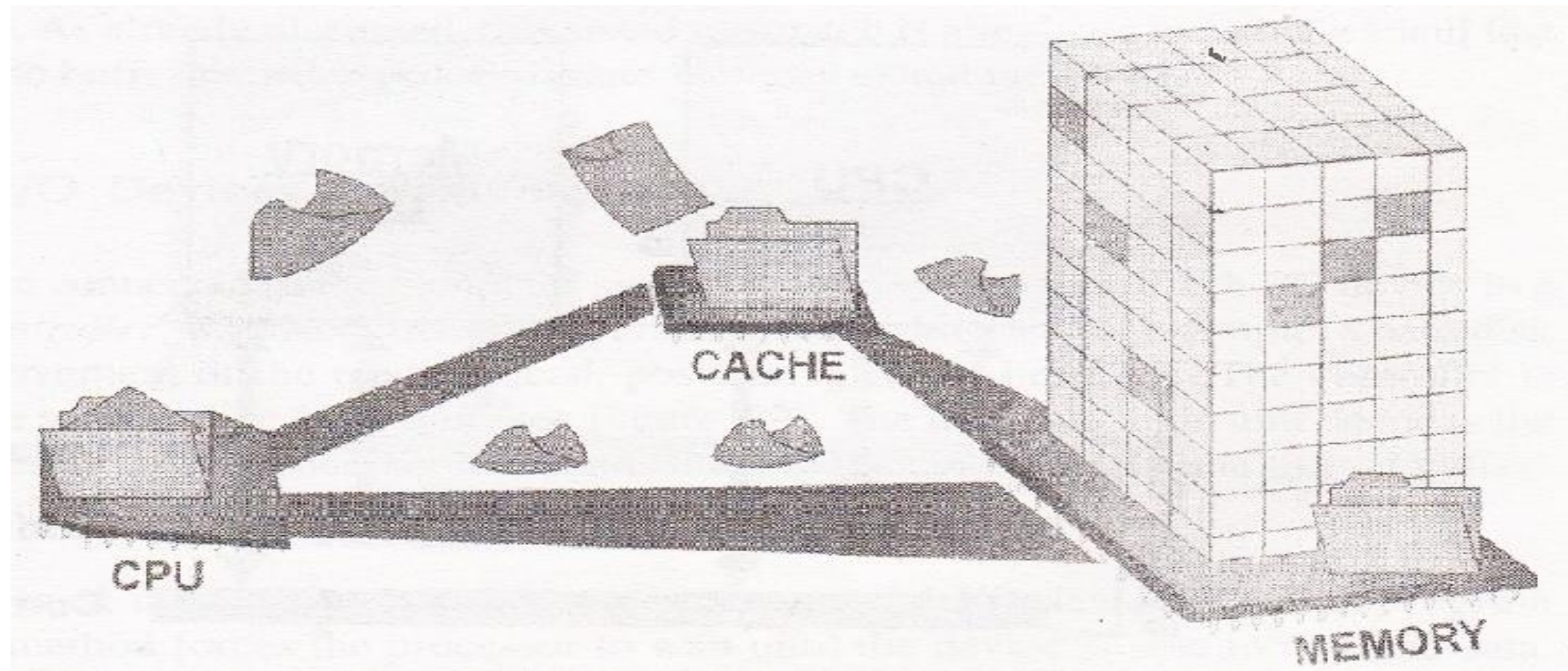
Varieties of Semiconductor Random Access Memories



Cache Memory

- High speed memory kept in between processor and RAM to increase the data execution speed.
- Kept near to the processor.
- Major reason for incorporating cache in the system is that the CPU is much faster than the DRAM and needs a place to store information that can be accessed quickly.
- Cache fetches the frequently used data from the DRAM and buffers (stores) it for further processor usage.

Cache Memory



Comparison of Memory Types

- DRAM
 - very high density → cheap data cache in computers
 - must be periodically refreshed → slower than SRAM
 - volatile; no good for program (long term) storage
- SRAM (basically a Latch)
 - fastest type of memory
 - low density → more expensive
 - generally used in small amounts (L2 cache) or expensive servers
- EEPROM
 - slow/complex to write → not good for fast cache
 - non-volatile; best choice for program memory
- ROM
 - hardware coded data; rarely used except for bootup code
- Register (flip flop)
 - functionally similar to SRAM but less dense (and thus more expensive)
 - reserved for data manipulation applications

İşlemci Bellek İlişkisi

- İşlemci içerisinde önbellekte fazla miktarda bilgi saklanamaz ve geçici saklayıcılarda (kayıt ediciler) ise veri manipule (Transfer, Kontrol, ALU) işlevleri yerine getirilir.
- İşlemci gerekli verileri RAM'den okur ve işlem sonuçlarını RAM belleğe yazar.
- İşlemcinin işleyeceği verilerin RAM'in hangi bellek gözünden okunacağı ve işlem sonuçlarının hangi bellek gözüne yazılacağı programda verilmiş adreslerle belirlenir.
- Ram bellekte herbir veri programlarda belirlenen bir adreste bulunur.
- Ram bellek işlemcinin hesap defteri gibidir. Devamlı buradan bir şeyler okur, siler, yazar...
- CPU, hangi verinin hangi RAM bellek ile ilişkili olduğu yazılan program tarafından verilen adres hatları üzerinden Address Decoding Unit tarafından belirlenir.

Latch

- Yazma anında Clock sinyalini yüksel kenar (Tetikleme) gelmediği sürece çıkış değerini korur.
- Bir elektronik devreye çalışma gerilimi uygulandığı sürece durumunu ve buna bağlı olarak çıkışındaki değeri devamlı olarak koruyan multivibratör çeşidi Flip Flop olarak isimlendirilir. FF olarak sembolize edilir. Lojik kapılar ile oluşturduğumuz flip floplar lojik devrelerde en önemli bellek elemanlarıdır.
- FF'ler için çift kararlı multivibratör (bistable multivibratör) terimi de kullanılır. FF'lerin tetikleme girişine uygulanan kare veya dikdörtgen şeklindeki sinyaller, tetikleme sinyali/palsi olarak adlandırılır. FF devresi tetikleme sinyalinin pozitif kenarında tetikleniyorsa pozitif kenar tetikleme negatif kenar tetiklemeli devre olarak tanımlanır.
- Birçok FF türü vardır. Bunlardan en çok kullanılanları:
 - R-S(reset-set) tipi FF
 - Tetiklemeli(clocked) R-S FF
 - J-K Tipi FF
 - Master Slave tipi FF
 - D (data) tipi FF
 - T(Toggle) tipi FF'dir

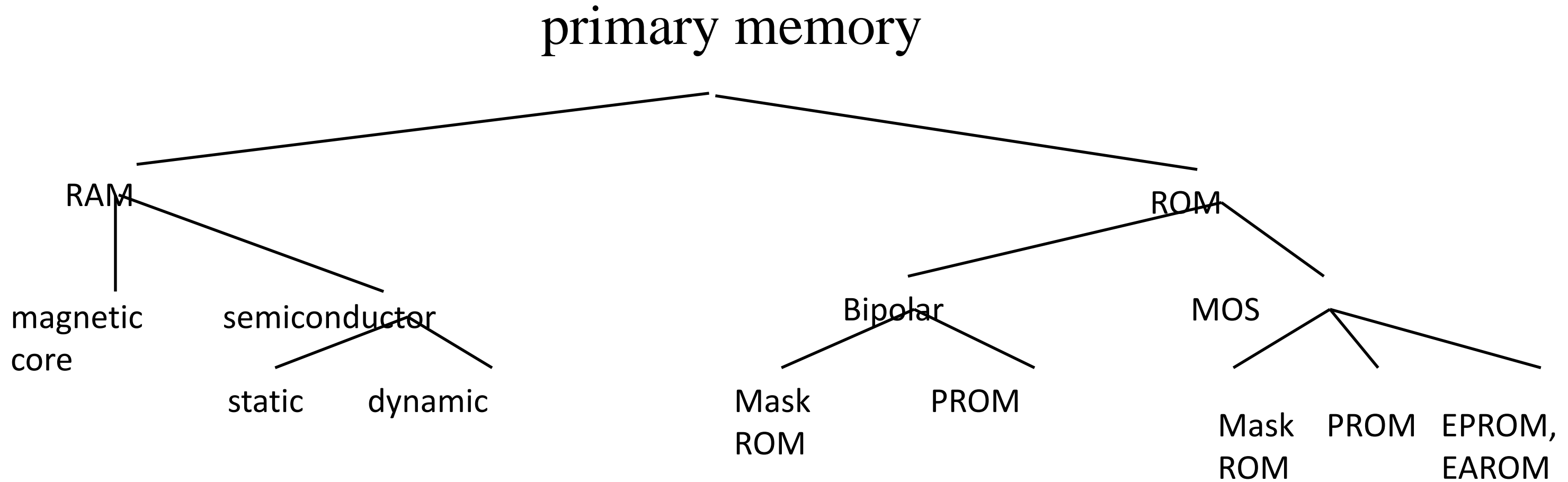
Memory speed

- **Access time (T_a)**
 - the average time taken to read a unit of information
e.g., 100 ns (100×10^{-9} s)
- **Access rate (R_a) = $1/T_a$ (bits/second)**
 - e.g., $1/100\text{ns} = 10 \text{ Mb/s}$
- **Cycle time (T_c)**
 - the average time lapse between two successive read operations
e.g., 500 ns (500×10^{-9} s)
- **Bandwidth or transfer rate (R_c) = $1/T_c$ (bits/second)**
 - e.g., $1/500\text{ns} = 2 \text{ Mb/s}$

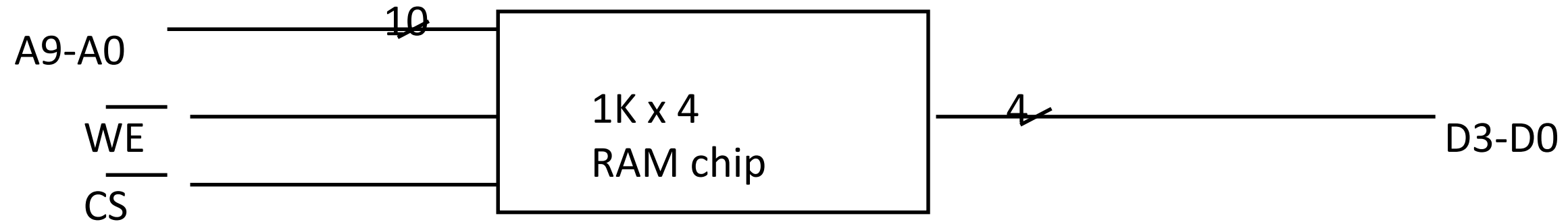
Classes of Memory

- RAM (“normal memory”)
- Direct-access storage: HD, CD ROM, DVD
- Sequential access storage tapes: DAT
- Associative (content-addressable) memory: searches for data via bit patterns
 - CAM (Content Addressable Memory)
 - Includes comparison logic with each bit of storage.
 - A data value is broadcast to all words of storage and compared with the values there.
 - Words which match are flagged.
 - Subsequent operations can then work on flagged words.
 - (computing-dictionary.thefreedictionary.com)
- ROM

Categories of RAM and ROM



Main Memory Design



\overline{CS}	\overline{WE}	MODE	Status of the Bi-directional Datelines D3-D0	Power
H	X	not selected	High impedance	Standby
L	L	Write	Acts as input bus	Active
L	H	Read	Acts as output bus	Active

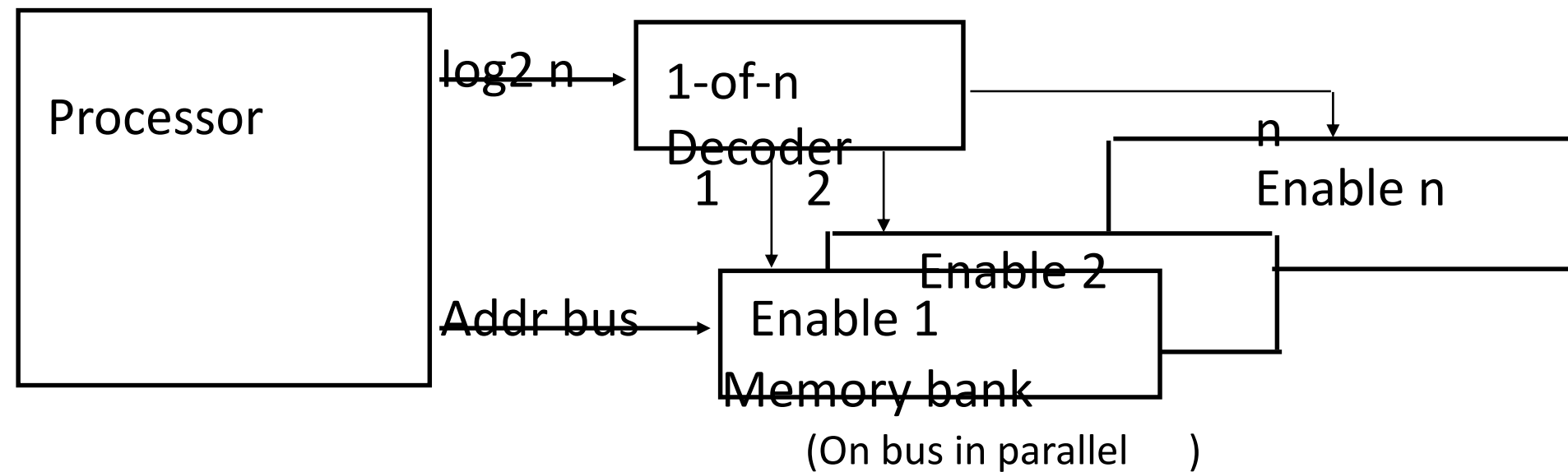
Main Memory Design

Q. How do we build a 4K x 4 RAM using four 1K x 4 RAM chips?

Chip	A11	A10	A9	A8	A7 ... A0	Range
0	0	0	x	x	x ... x	0000 to 1023
1	0	1	x	x	x ... x	1024 to 2047
2	1	0	x	x	x ... x	2048 to 3071
3	1	1	x	x	x ... x	3072 to 4096

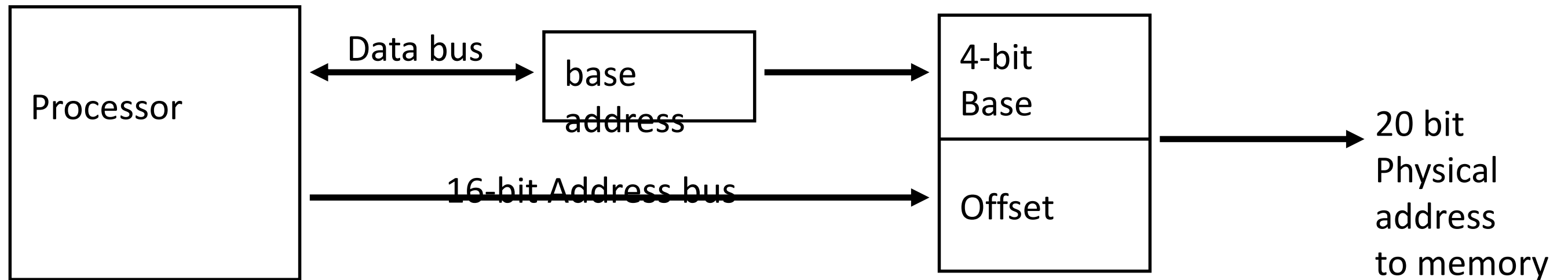
Main Memory Design

- Q. How do we build a 256KB RAM system with an 16-bit address bus and four 64KB RAM chips?
- Memory band-switching

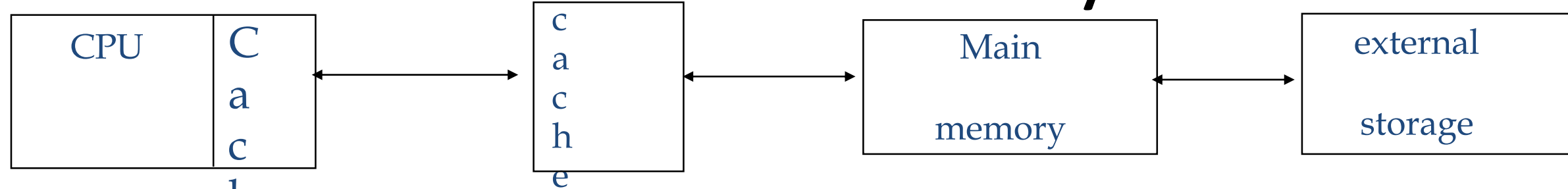


Main memory design

- Memory address extension



Cache Memory



- Cache: fast-access memory buffer
- locality principle: programs usually use limited memory areas, in contrast to totally random access
 - spatial: location, address
 - temporal: time accessed
 - if commonly used memory can be buffered in high-speed cache, overall performance enhanced
 - cache takes form of small amount of store, with hardware support for maintenance and lookup
 - each cache cell saves a *cache line* - block of main memory (4-64 words)
- **cache hit:**
 - requested memory resides in cache

Cache

- *cache miss:*
 - requested memory not in cache, and must be fetched from main memory and put into cache
- *unified cache:*
 - instns, data share same cache
- *split cache:*
 - separate instn, data caches
- *parallel access:*
 - double the bandwidth
- *level 2 cache:*
 - between instn/data cache and main memory
- Cache maintenance algorithms similar in spirit to virtual memory ideas at operating system level; main difference is that cache is hardware-supported, whereas v.m. is software implemented

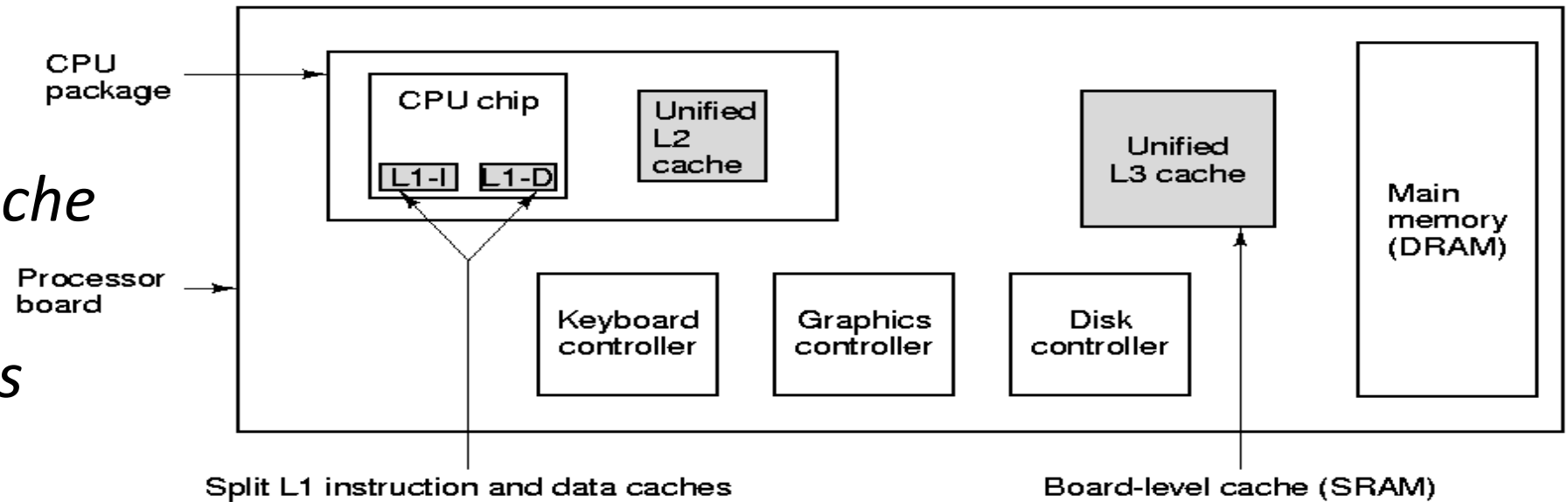


Figure 4-37. A system with three levels of cache.



I/O Units

I/O Unit

- Girdi/çıkıtı (I/O), bir bilgi işleme sistemi (bilgisayar gibi) ile dış dünya, muhtemelen bir insan veya başka bir bilgi işleme sistemi arasındaki iletişimi ifade eder.
- Girişler, sistem tarafından alınan sinyaller veya verilerdir ve çıkışlar, sistemden gönderilen sinyaller veya verilerdir.
- Bilgisayara giriş veya çıkış sağlayan aygıtlara çevre birimleri denir.
- Tipik bir kişisel bilgisayarda çevre birimleri, klavye ve fare gibi giriş aygıtlarını ve ekran ve yazıcı gibi çıktı aygıtlarını içerir.
- Sabit disk sürücüleri, disket sürücüleri ve optik disk sürücüleri, hem giriş hem de çıkış aygıtı olarak işlev görür.
- Bilgisayar ağı, başka bir G/Ç biçimidir.

Input Devices

- Any peripheral used to provide data and control signals to an information processing system such as a computer or other information appliance.
- Common input devices: Keyboard, Mouse
- Other devices: microphone, digital camera, scanner.

Output Devices

- Any piece of computer hardware equipment used to communicate the results of processed data to the user.
- Examples: Monitors, Printers, Speakers, etc.

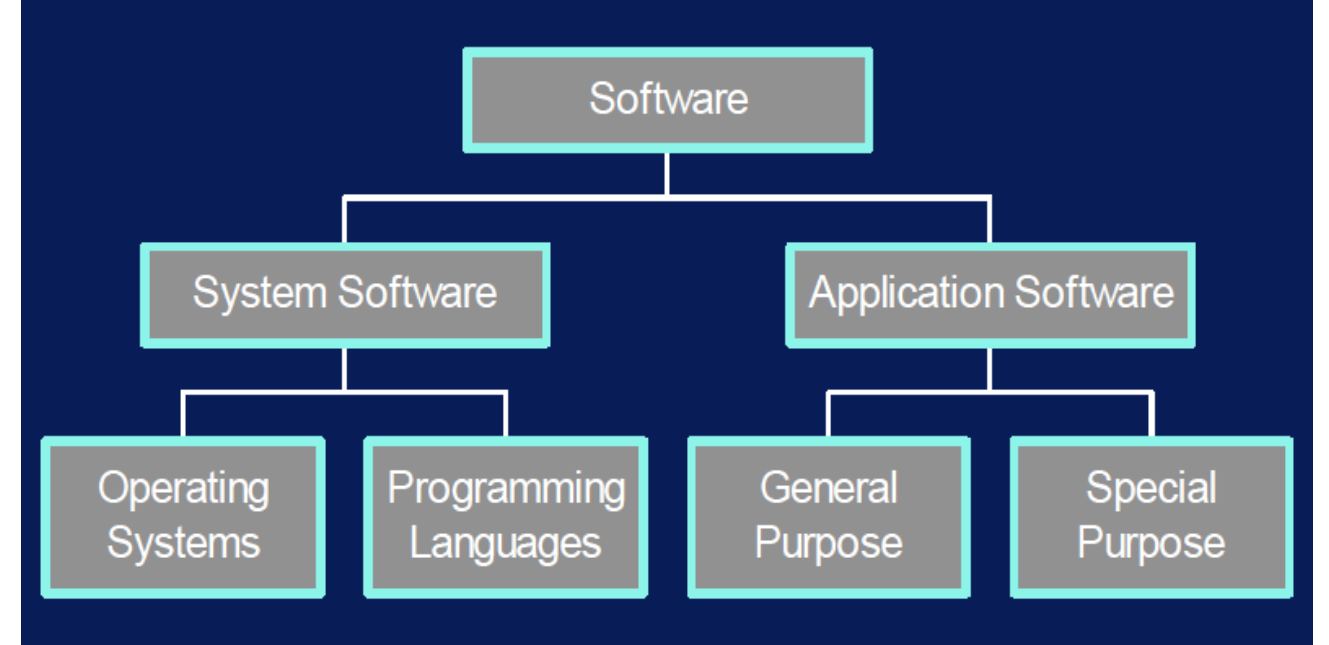


Operating Systems

Operating Systems

Operating Systems:

- İşletim Sistemi, bilgisayar sisteminin genel işleyişini yöneten yazılımdır.
- Temel amaç uygulama programlarını desteklemektir.
- Uygulama programlarından cihazların ayrıntıları gizlenir.
- DOS (Disk İşletim Sistemi) Windows: Intel 80X86 işlemciler için tasarlandı
- Macintosh OS: Motorola ve PowerPC işlemciler için tasarlanmıştır.
- Kullanıcı arayüzleri: Programı kullanan kişiden (kullanıcı) bilgi iletmekten sorumlu yazılım. Bilgisayarla iletişim kurar ve bilgisayarı kontrol eder. Üç tür kullanıcı arayüzü:
 - Graphic user interfaces
 - Menu driven interfaces
 - Command driven interfaces





Components of Computer

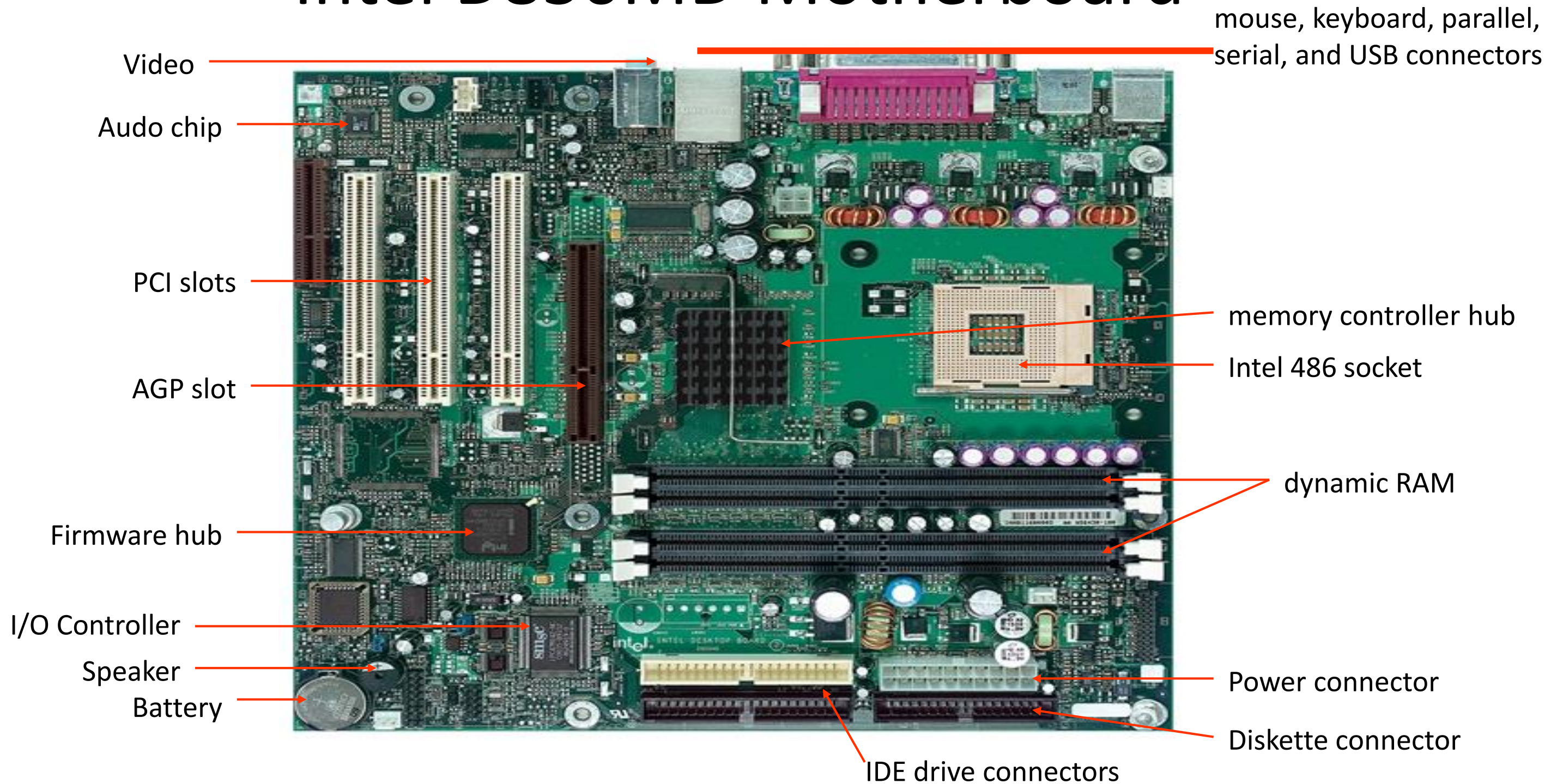
Components of an IA-32 Microcomputer

- Motherboard
- Video output
- Memory
- Input-output ports

Motherboard

- CPU socket
- External cache memory slots
- Main memory slots
- BIOS chips
- Sound synthesizer chip (optional)
- Video controller chip (optional)
- IDE, parallel, serial, USB, video, keyboard, joystick, network, and mouse connectors
- PCI bus connectors (expansion cards)

Intel D850MD Motherboard



Input-Output Ports

- USB (universal serial bus)
 - intelligent high-speed connection to devices
 - up to 12 megabits/second
 - USB hub connects multiple devices
 - *enumeration*: computer queries devices
 - supports *hot* connections
- Parallel
 - short cable, high speed
 - common for printers
 - bidirectional, parallel data transfer
 - Intel 8255 controller chip

Input-Output Ports (cont)

- Serial
 - RS-232 serial port
 - one bit at a time
 - uses long cables and modems
 - 16550 UART (universal asynchronous receiver transmitter)
 - programmable in assembly language



Clock & Timing

Processor Speed

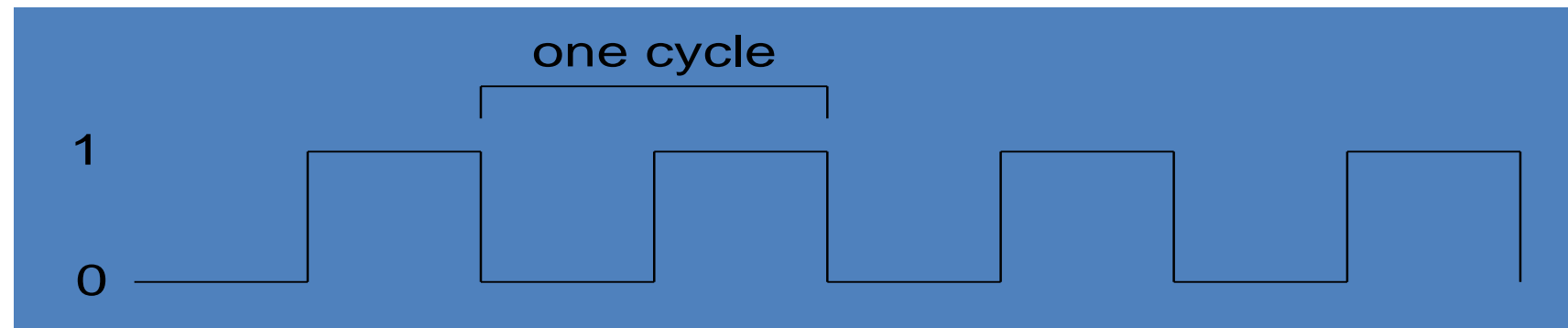
- Speed of a computer system is determined by several factors, clock speed of the processor and the speed and size of the data bus.
- Clock speed is the rate at which the processor processes information and this is measured in millions of cycles per second(Megahertz)
- The more the number of hertz, the faster is the processing speed
- The larger the bus width and the faster the bus speed, the greater the amount of data can travel on it in a given amount of time.

Microprocessor clock

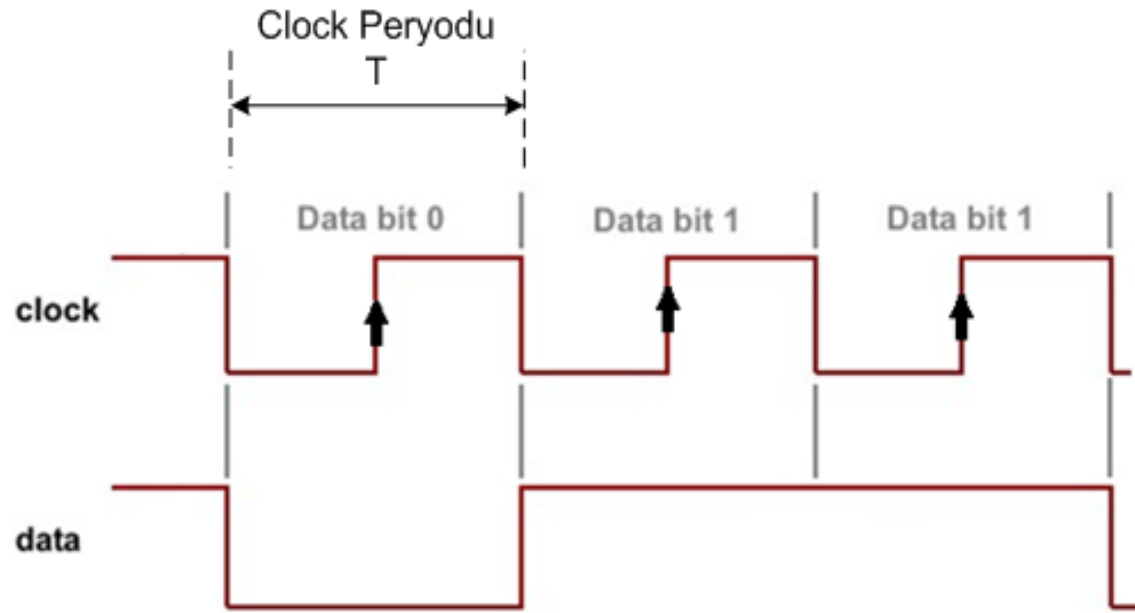
- Also called clock rate, the speed at which a microprocessor executes instructions. Every computer contains an internal clock that regulates the rate at which instructions are executed and synchronizes all the various computer components.
- The CPU requires a fixed number of clock ticks (or clock cycles) to execute each instruction. The faster the clock, the more instructions the CPU can execute per second. Clock speeds are expressed in megahertz (MHz) or gigahertz ((GHz).
- Some microprocessors are superscalar, which means that they can execute more than one instruction per clock cycle.
- Like CPUs, expansion buses also have clock speeds. Ideally, the CPU clock speed and the bus clock speed should be the same so that neither component slows down the other. In practice, the bus clock speed is often slower than the CPU clock speed, which creates a bottleneck. This is why new local buses, such as AGP, have been developed.

Clock

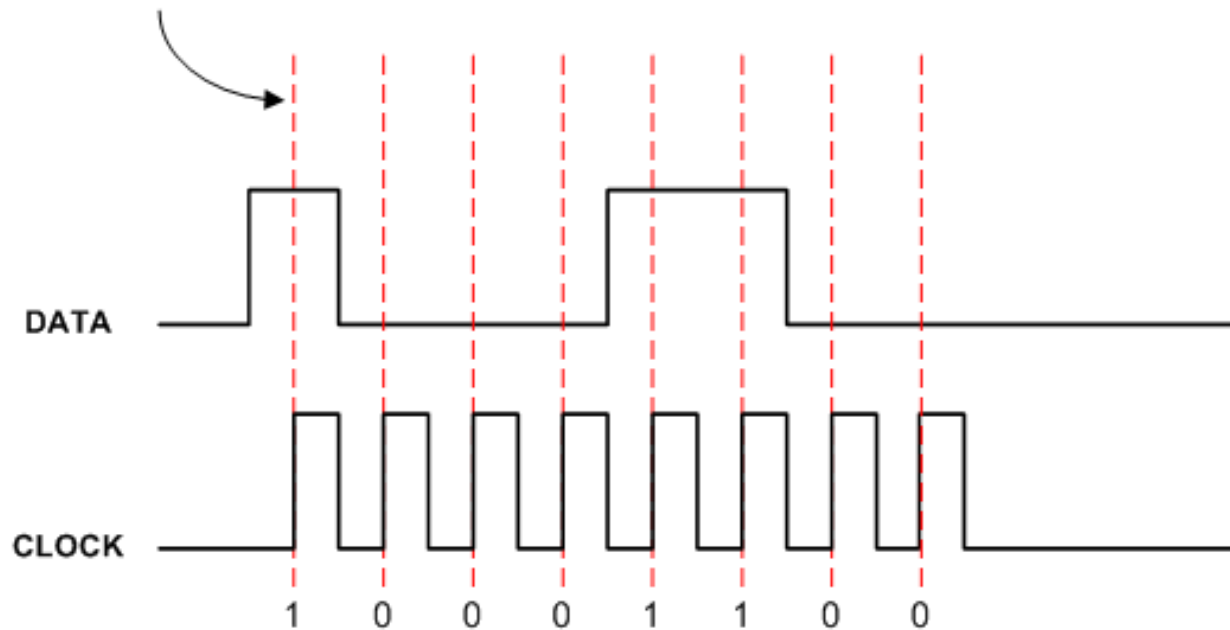
- synchronizes all CPU and BUS operations
- machine (clock) cycle measures time of a single operation
- clock is used to trigger events



Clock and Data



Data sampled on rising edge of clock

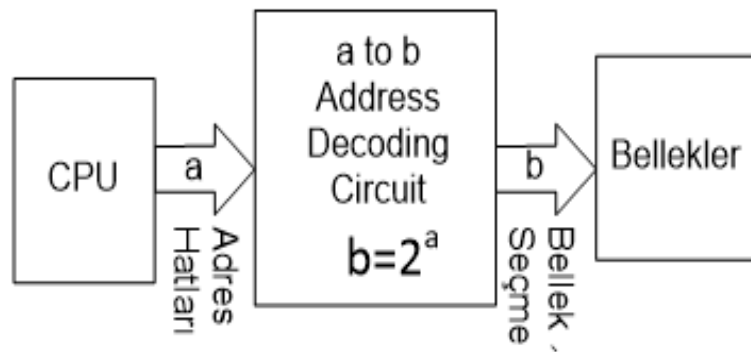
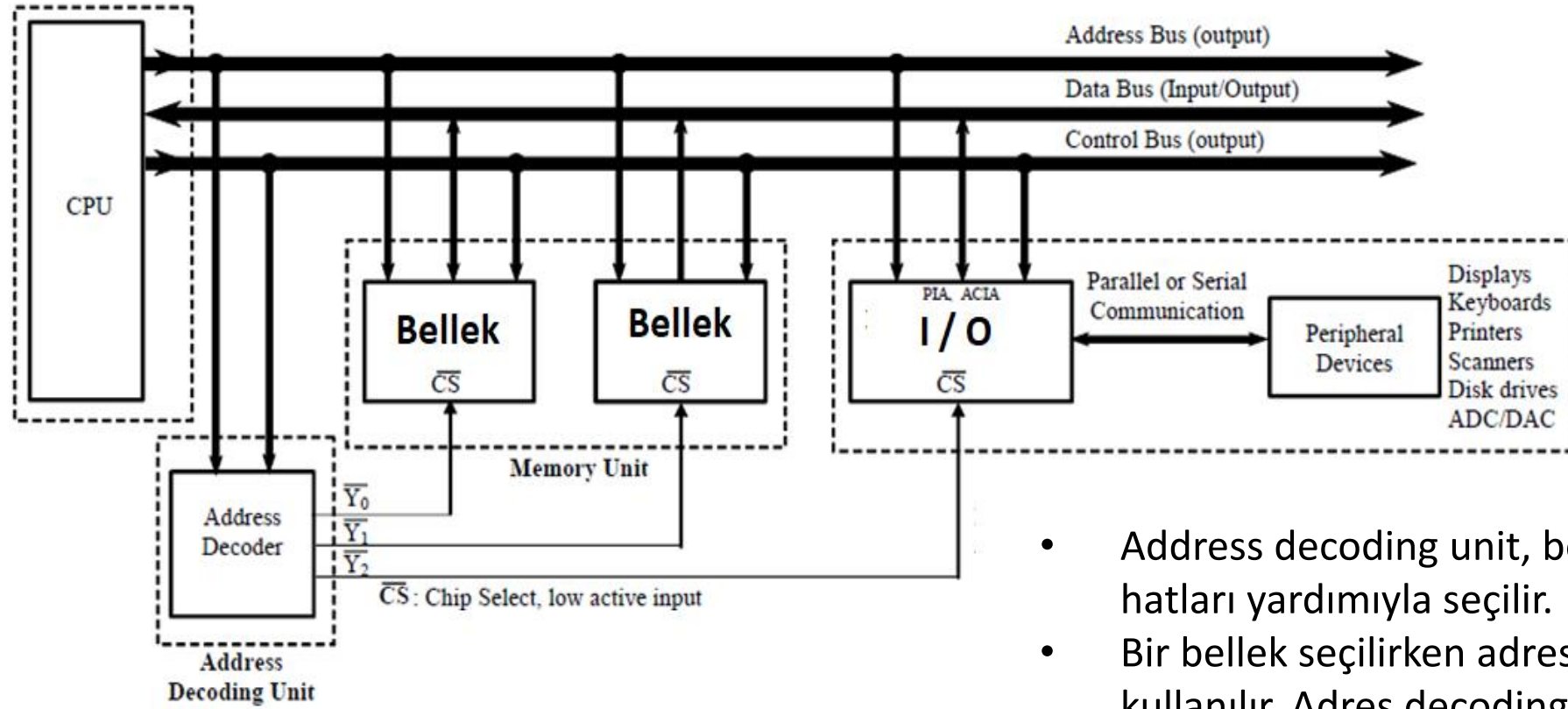


- Her bilgisayarın bir sistem clock darbe sinyali vardır.
- Sonsuzdan gelip sonsuza gider; sürekli dir.
- Clock darbe sinyali 1 ve 0 lardan oluşan darbe katarıdır (Elektriksel sinyaldir).
- Clock periyodu, data uzunluğuna eşittir.
- Bit olarak tanımlanan verileri tetiklemede yani işlemede kullanılır.
- Kişisel bilgisayar hızları genellikle gigahertz (GHz) cinsinden ifade edilir.



Address Decoding Unit

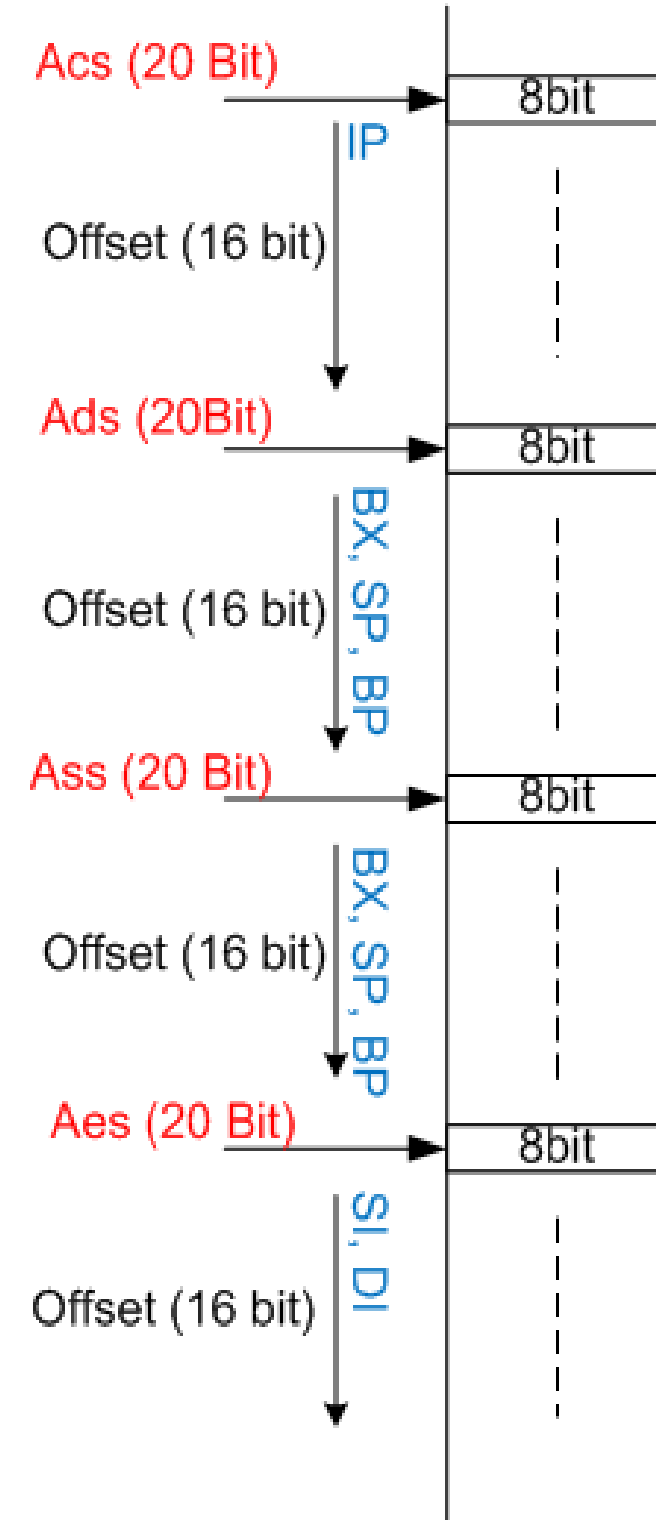
Address Decoding Unit



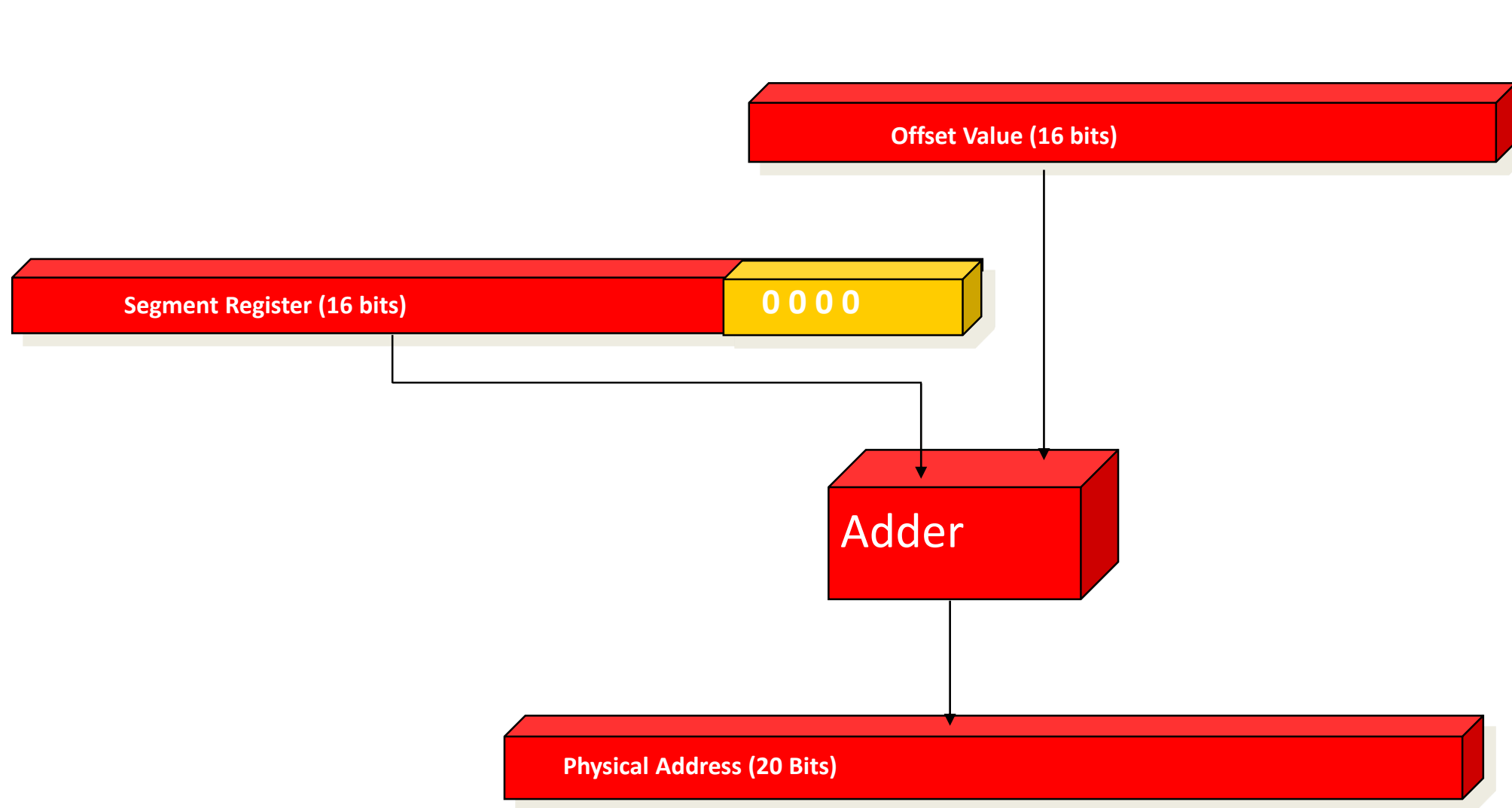
- Address decoding unit, bellek seçer. Belleğin gözü ise CPU'dan gelen adres hatları yardımıyla seçilir.
- Bir bellek seçilirken adres decoding devresine CPU'dan gelen adres hatları kullanılır. Adres decoding devresinin çıkışları bellekleri seçmede kullanılır.
- Seçilecek bellek sayısı=Address decoding devresi çıkış sayısı= 2^m dir. Burada m CPU'dan gelen adres hattı sayısıdır.
- Herbir belleğin kapasitesini belirleyen (Kapasite= 2^n) n adet hat CPU'dan gelir.
- Amaç aynı anda bir belleğin ilgili veri gözünün seçilmesidir.
- Adres decoding devresinin çıkış sayısı bellek sayısına eşit ya da büyük olmak sorunudur.
- Adres decoding devresini girişi, CPU dan gelecek adres hattı sayısı belirlenir. Özeldir.

Bellek Gözünün Fiziksel Adresi

- Belleklerin başlangıç konumlarına erişim
 - Segment Registers: 16 bit
 - CS: ROM
 - DS, SS, ES: RAM
 - Segment register'ın içeriği 10h ile çarpıldığında ilgili belleğin fiziksel başlangıç adresi 20 bit olarak hesaplanmış olur.
- Bellek başlangıç adresi belirlendikten sonra ilgili bellek gözünün adresi, belleğin 20bitlik fiziksel başlangıç adresine ilgili ofsset register ya da değişken ilavesi ile hesaplanır



Addressing Modes : Memory Access



	15	8	7	0
AX	AH		AL	
BX	BH		BL	
CX	CH		CL	
DX	DH		DL	



15		0
SP		
BP		
DI		
SI		
Flag Register		

EU

15		0
CS		
DS		
SS		
ES		

BIU

CPU Registers

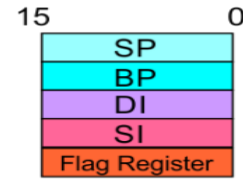
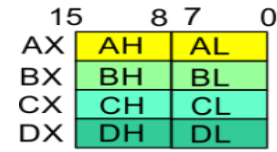
- **Fourteen 16-bit registers**
- **Data Registers**
 - AX (Accumulator Register): AH and AL
 - BX (Base Register): BH and BL
 - CX (Count Register): CH and CL
 - DX (Data Register): DH and DL
- **Pointer and Index Registers**
 - SI (Source Index)
 - DI (Destination Index)
 - SP (Stack Pointer)
 - BP (Base Pointer)
 - IP (Instruction Pointer)

CPU Registers

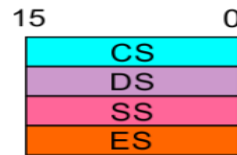
- **Segment Registers**
 - CS (Code Segment)
 - DS (Data Segment)
 - SS (Stack Segment)
 - ES (Extra Segment)
- **FLAGS Register**
 - Zero flag
 - Sign flag
 - Parity flag
 - Carry flag
 - Overflow flag

Architecture

8086 registers categorized into 4 groups



EU



BIU



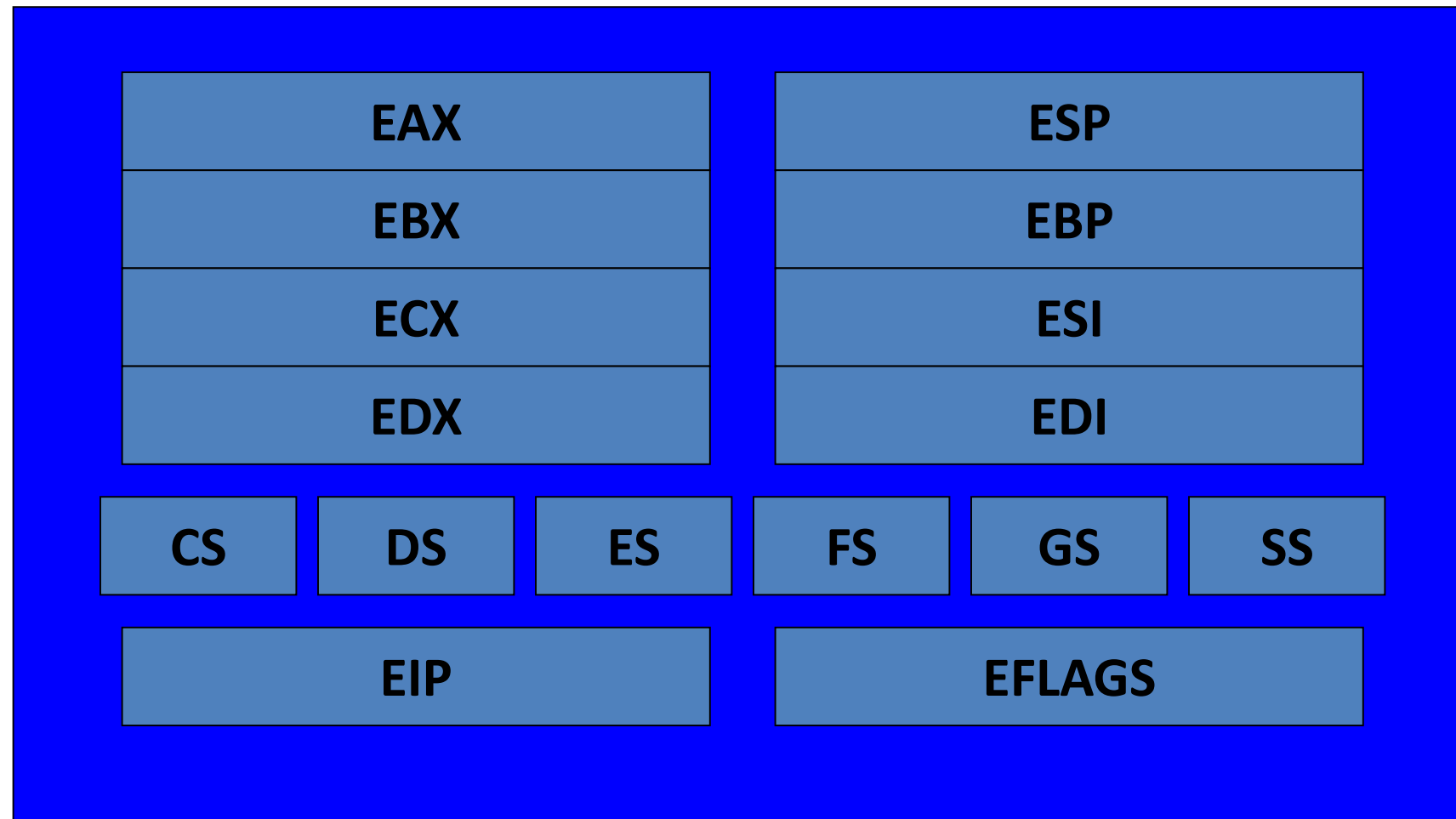
Sl.No.	Type	Register width	Name of register
1	General purpose register	16 bit	AX, BX, CX, DX
		8 bit	AL, AH, BL, BH, CL, CH, DL, DH
2	Pointer register	16 bit	SP, BP
3	Index register	16 bit	SI, DI
4	Instruction Pointer	16 bit	IP
5	Segment register	16 bit	CS, DS, SS, ES
6	Flag (PSW)	16 bit	Flag register

Register	Name of the Register	Special Function
AX	16-bit Accumulator	Stores the 16-bit results of arithmetic and logic operations
AL	8-bit Accumulator	Stores the 8-bit results of arithmetic and logic operations
BX	Base register	Used to hold base value in base addressing mode to access memory data
CX	Count Register	Used to hold the count value in SHIFT, ROTATE and LOOP instructions
DX	Data Register	Used to hold data for multiplication and division operations
SP	Stack Pointer	Used to hold the offset address of top stack memory
BP	Base Pointer	Used to hold the base value in base addressing using SS register to access data from stack memory
SI	Source Index	Used to hold index value of source operand (data) for string instructions
DI	Data Index	Used to hold the index value of destination operand (data) for string operations

Pentium's visible "registers"

- Four general-purpose registers:
 - 32 bit: eax, ebx, ecx, edx
 - 16 bit: ax, bx, cx, dx
 - 8 bit: ah-al, bh-bl, ch-cl, dh-dl
- Four memory-addressing registers:
 - 32 bit: esp, ebp, esi, edi
- Six memory-segment registers: Belleklerin başlangıç adreslerini gösterir. Segment register'lar 16 bittir, 20 bit fiziksel adresi belirlemek için 10h ile çarpılır ya da sağ tarafına 4 adet 0 bit eklenir.
 - 16 bit: cs, ds, es, fs, gs, ss
 - cs: rom, program bellek, ds, es, fs, gs, ss: RAM bellek
- An instruction-pointer and a flags register:
 - 32 bit: eip, eflags
 - 16 bit: IP, Flags

The sixteen x86 registers



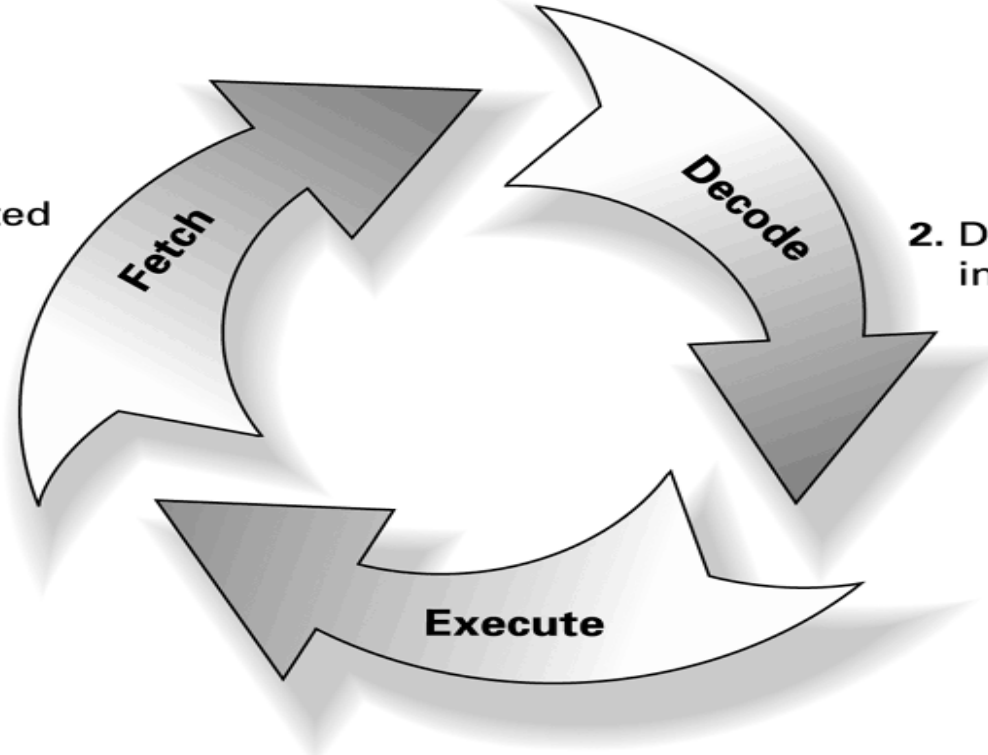
Intel Pentium processor



Instruction Execution Cycle

Mikroişlemci işlev döngüsü

1. Retrieve the next instruction from memory (as indicated by the program counter) and then increment the program counter.



2. Decode the bit pattern in the instruction register.

3. Perform the action required by the instruction in the instruction register.

- İşlev döngüsünü yerine getiren birimler ayrı ayrı çalışır.

Mikroişlemci ana işlev döngüsünde,

Fetch Process:

- Veriyi bellek ya da I/O biriminden Git – Al – Getir.
- Veriyi bellek ya da I/O birimine Al - Götür.

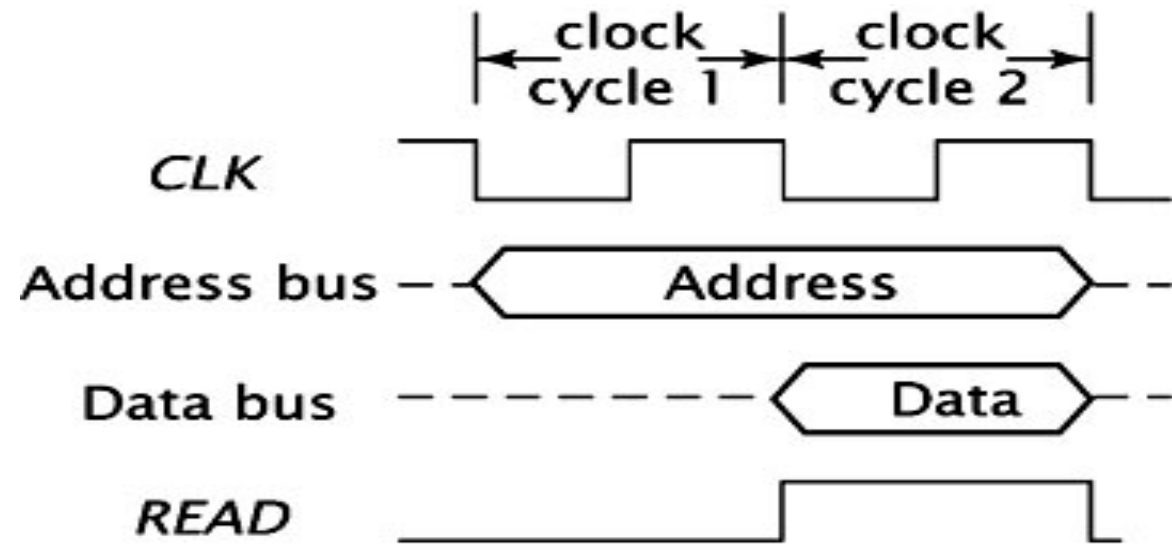
Execute process : Uygula, işle

Decode: Bellek ve ilgili gözünü seçer.

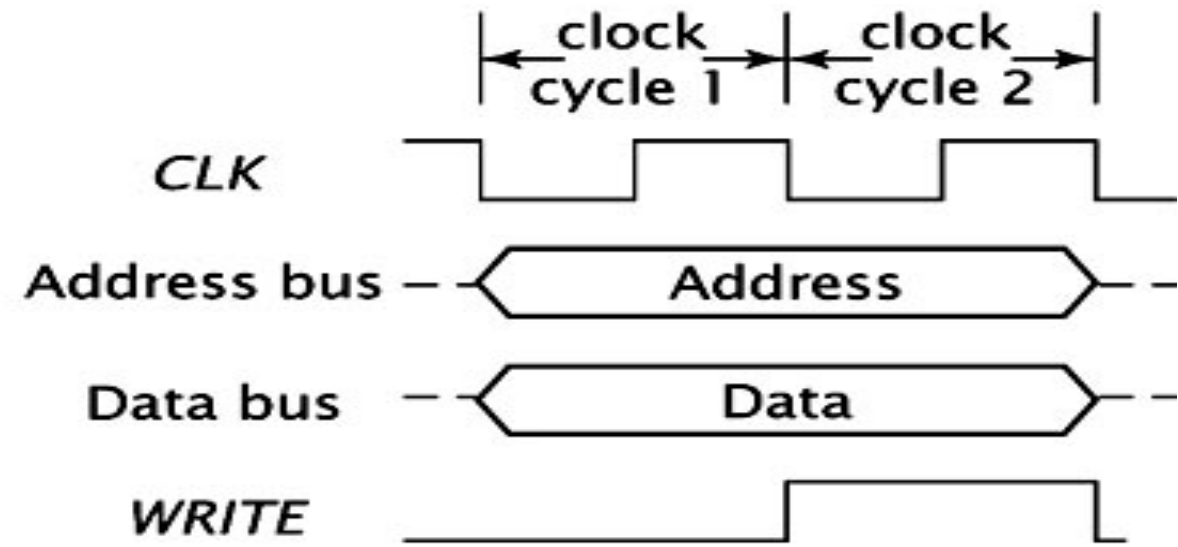
Instruction Cycle

- **Fetch**
- **Decode**
- **Execute**

Instruction Fetch



(a)



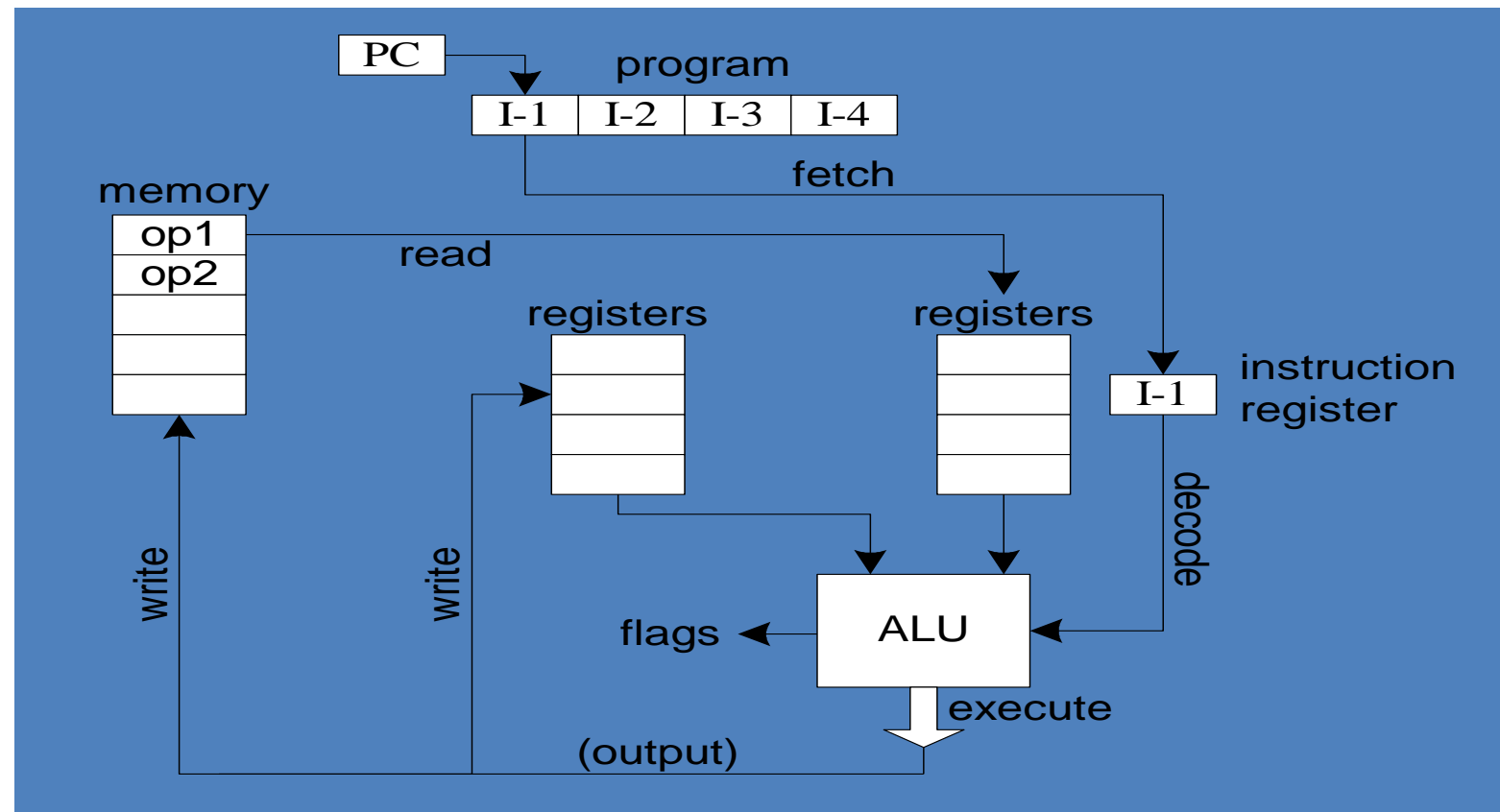
(b)

Fetch-Execute Process

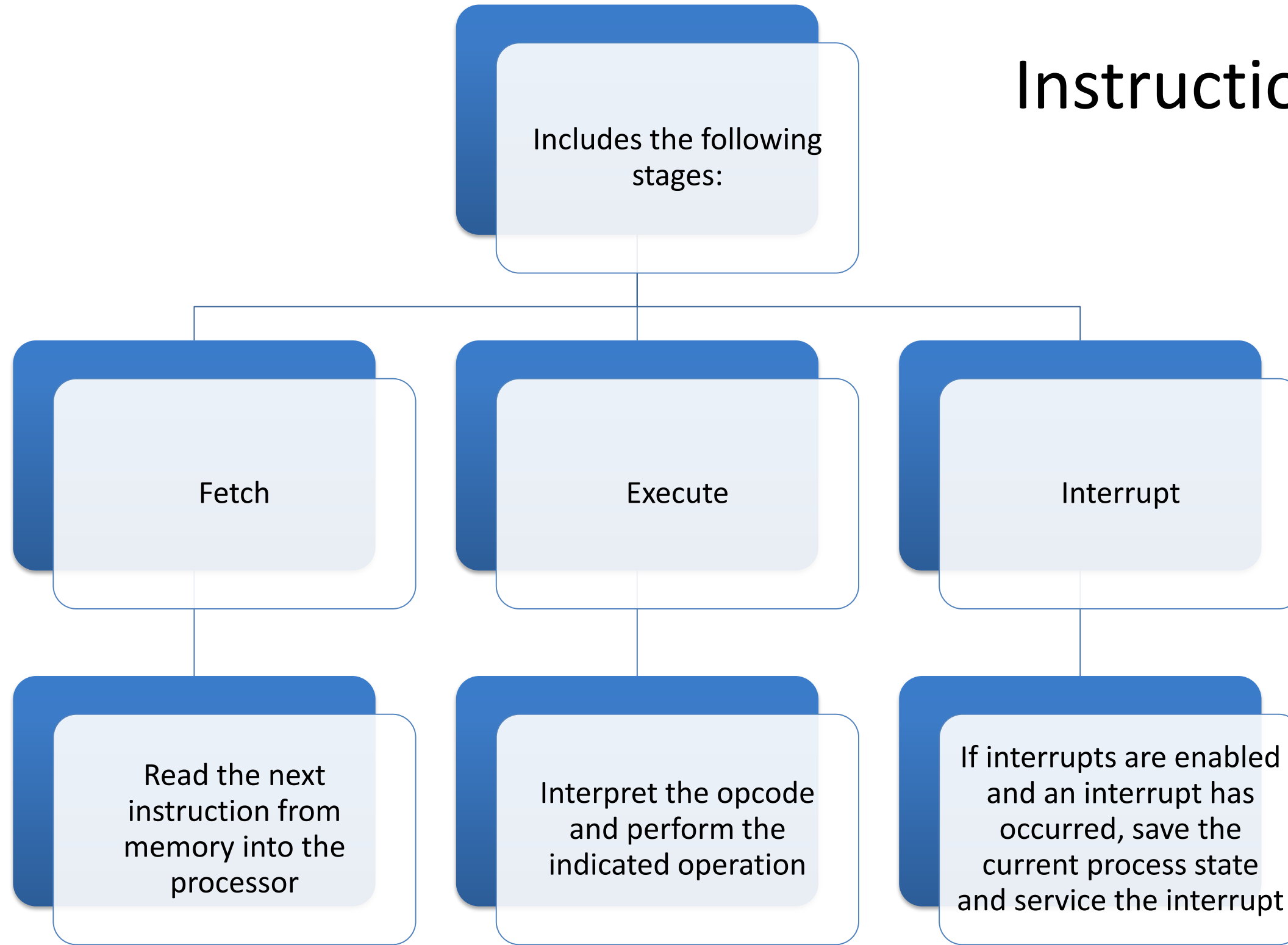
- **Program Counter (PC) or Instruction Pointer (IP)**
 - Holds address of next instruction to fetch
- **Instruction Register (IR)**
 - Stores the instruction fetched from memory
- **Fetch-Execute process**
 - Read an instruction from memory addressed by PC
 - Increment program counter
 - Execute fetched instruction in IR
 - Repeat process

Instruction Execution Cycle

- Fetch
- Decode
- Fetch operands
- Execute
- Store output



Instruction Cycle



Instruction Cycle

- To recall, an instruction cycle includes the following stages:
- Fetch: Read the next instruction from memory into the processor.
- Execute: Interpret the opcode and perform the indicated operation.
- Interrupt: If interrupts are enabled and an interrupt has occurred, save the current process state and service the interrupt.
- We are now in a position to elaborate somewhat on the instruction cycle. First, we must introduce one additional stage, known as the indirect cycle.

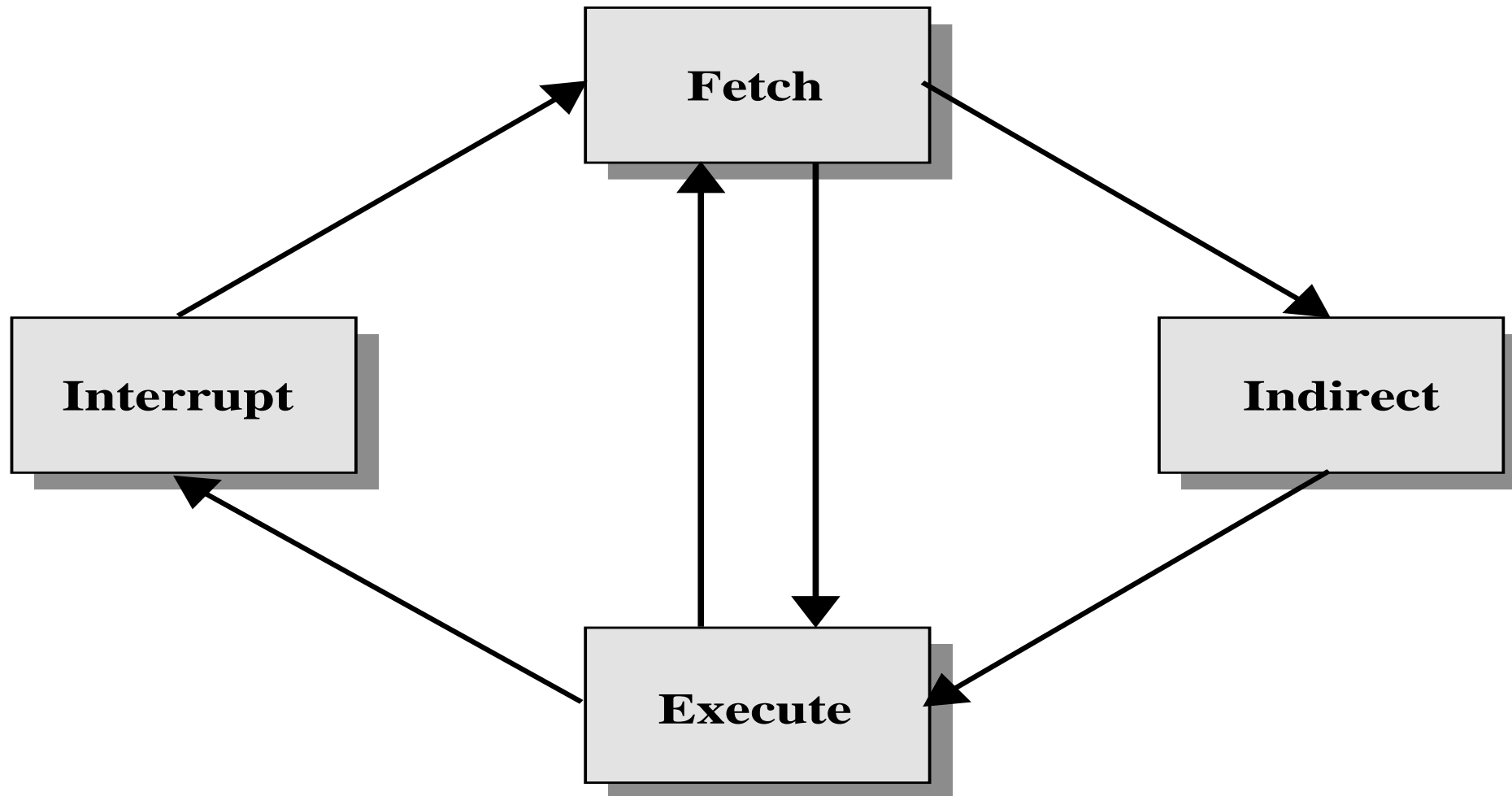


Figure 14.4 The Instruction Cycle

Instruction Cycle

- The execution of an instruction may involve one or more operands in memory, each of which requires a memory access. Further, if indirect addressing is used, then additional memory accesses are required.
- We can think of the fetching of indirect addresses as one more instruction stages. The main line of activity consists of alternating instruction fetch and instruction execution activities. After an instruction is fetched, it is examined to determine if any indirect addressing is involved. If so, the required operands are fetched using indirect addressing. Following execution, an interrupt may be processed before the next instruction fetch.

Additional Stages

- Fetch instruction (FI)
 - Read the next expected instruction into a buffer
- Decode instruction (DI)
 - Determine the opcode and the operand specifiers
- Calculate operands (CO)
 - Calculate the effective address of each source operand
 - This may involve displacement, register indirect, indirect, or other forms of address calculation
- Fetch operands (FO)
 - Fetch each operand from memory
 - Operands in registers need not be fetched
- Execute instruction (EI)
 - Perform the indicated operation and store the result, if any, in the specified destination operand location
- Write operand (WO)
 - Store the result in memory

Instruction Cycle

- While these factors reduce the potential effectiveness of the two-stage pipeline, some speedup occurs. To gain further speedup, the pipeline must have more stages. Let us consider the following decomposition of the instruction processing.
- Fetch instruction (FI): Read the next expected instruction into a buffer.
- Decode instruction (DI): Determine the opcode and the operand specifiers.
- Calculate operands (CO): Calculate the effective address of each source operand. This may involve displacement, register indirect, indirect, or other forms of address calculation.
- Fetch operands (FO): Fetch each operand from memory. Operands in registers need not be fetched.
- Execute instruction (EI): Perform the indicated operation and store the result, if any, in the specified destination operand location.
- Write operand (WO): Store the result in memory.
- With this decomposition, the various stages will be of more nearly equal duration.



Pipelining

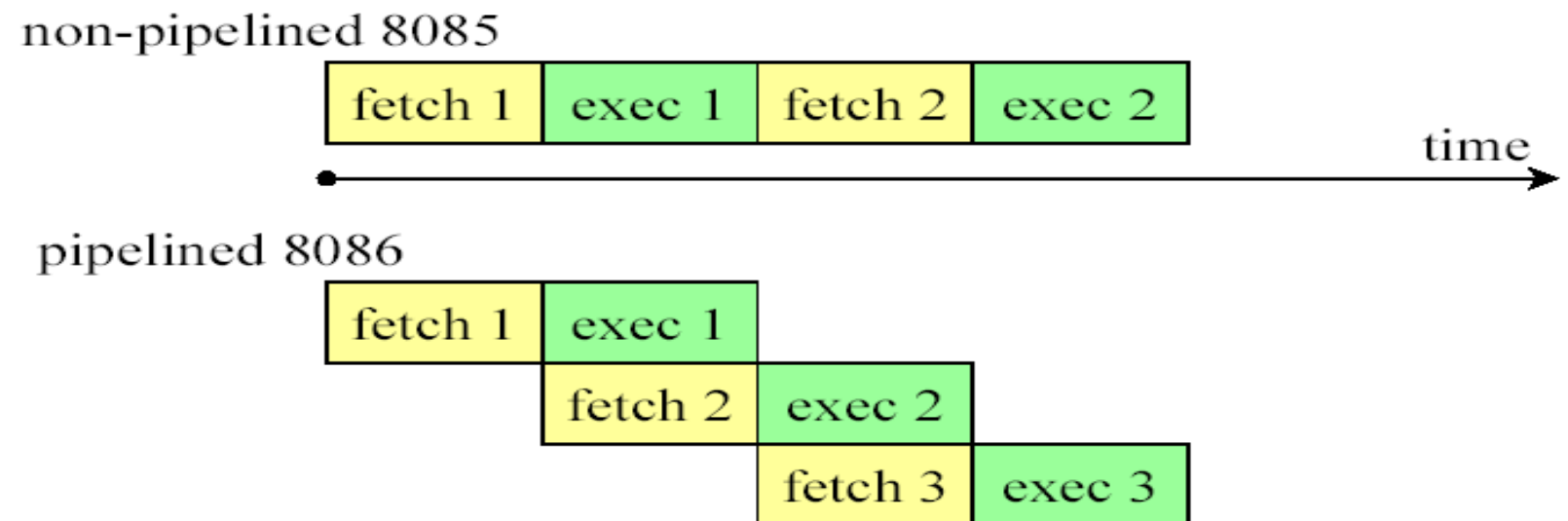
Inside X86

Concepts important to the internal operation of X86

- Pipelining: İşlevsel döngüleri yerine getiren tüm birimlerin aynı anda işlem yapması, çalışmasıdır.
- Registers

Inside X86...*pipelining*

- **Pipelining**
 - Two ways to make CPU process information faster:
 - Increase the working frequency – technology dependent
 - Change the internal architecture of the CPU
 - Pipelining is to allow CPU to fetch and execute at the same time



Without Pipelining

- Pipeline, ardışık düzende, işlemcinin komutları paralel olarak yürütmesini mümkün kılar
- S1 .. S6: Cpu Birimleri
- Cycles: Clock periyotlarından oluşur. Herbir işlevin yerine getirildiği süreyi tanımlar.
- Ayrık aşamalara bölünmüş komut yürütme
- Sağda, ardışık düzenlenmemiş bir işlemci örneği. Birçok boşa giden döngü.

		Stages					
		S1	S2	S3	S4	S5	S6
Cycles	1	I-1					
	2		I-1				
	3			I-1			
	4				I-1		
	5					I-1	
	6						I-1
	7	I-2					
	8		I-2				
	9			I-2			
	10				I-2		
	11					I-2	
	12						I-2

Pipelined Execution

- Döngülerin (Cycles) daha verimli kullanımı, daha fazla talimat çıktısı:

		Stages					
		S1	S2	S3	S4	S5	S6
Cycles	1	I-1					
	2	I-2	I-1				
	3		I-2	I-1			
	4			I-2	I-1		
	5				I-2	I-1	
	6					I-2	I-1
	7						I-2

k durum ve n komut için
gerekli döngü sayısı:

$$k + (n - 1)$$

- Pipelining uygulamasında tüm birimler aynı anda farklı işlevlerde (Komutlar) çalışmaya başlar.

Wasted Cycles (pipelined)

- Komut işleme döngüsü (Fetching, Decoding, Executing, ...) aşamalardan biri, iki veya daha fazla saat döngüsü gerektirdiğinde (Clock periyodu), saat döngülerinin bir kısmı boşa harcanır.

		Stages					
		S1	S2	S3	S4	S5	S6
Cycles	1	I-1					
	2	I-2	I-1				
	3	I-3	I-2	I-1			
	4		I-3	I-2	I-1		
	5			I-3	I-1		
	6				I-2	I-1	
	7				I-2		I-1
	8				I-3	I-2	
	9				I-3		I-2
	10					I-3	
	11						I-3

For k states and n instructions, the number of required cycles is:

$$k + (2n - 1)$$

Superscalar

Bir süperskalar işlemcinin birden çok yürütme işlem hattı vardır. Aşağıda, Aşama S4'ün sol ve sağ boru hatlarına (u ve v) sahip olduğuna dikkat edin.

		Stages						
		S1	S2	S3	S4		S5	S6
					u	v		
Cycles	1	I-1						
	2	I-2	I-1					
	3	I-3	I-2	I-1				
	4	I-4	I-3	I-2	I-1			
	5		I-4	I-3	I-1	I-2		
	6			I-4	I-3	I-2	I-1	
	7				I-3	I-4	I-2	I-1
	8					I-4	I-3	I-2
	9						I-4	I-3
	10							I-4

For k states and n instructions, the number of required cycles is:

$$k + n$$

pipeline

- For the sake of illustration, let us assume equal duration. Using this assumption, Figure 14.10 shows that a six-stage pipeline can reduce the execution time for 9 instructions from 54 time units to 14 time units.
- Several comments are in order: The diagram assumes that each instruction goes through all six stages of the pipeline. This will not always be the case. For example, a load instruction does not need the WO stage. However, to simplify the pipeline hardware, the timing is set up assuming that each instruction requires all six stages. Also, the diagram assumes that all of the stages can be performed in parallel. In particular, it is assumed that there are no memory conflicts. For example, the FI, FO, and WO stages involve a memory access. The diagram implies that all these accesses can occur simultaneously. Most memory systems will not permit that. However, the desired value may be in cache, or the FO or WO stage may be null. Thus, much of the time, memory conflicts will not slow down the pipeline.

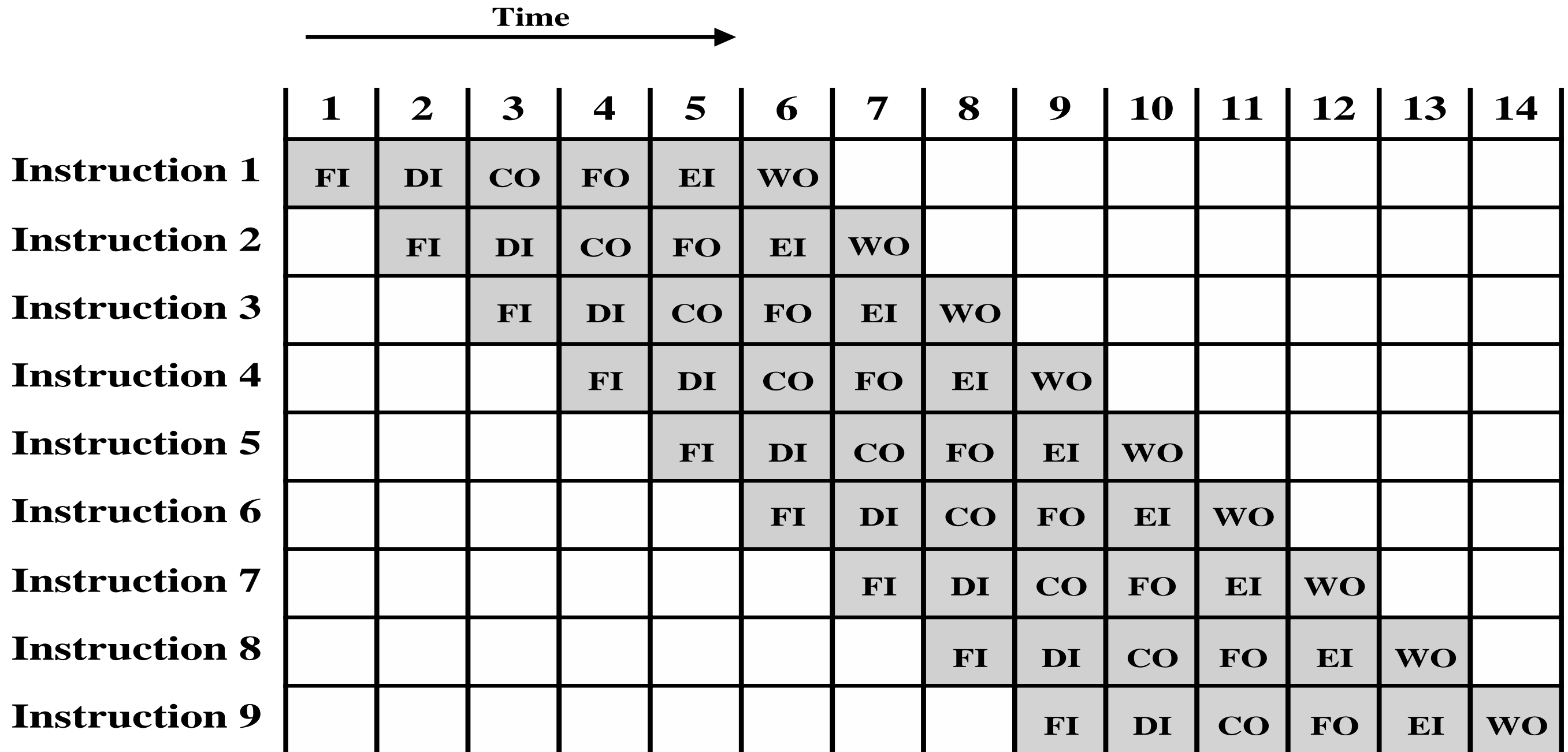


Figure 14.10 Timing Diagram for Instruction Pipeline Operation

Parallel Computing

Measure of Performance

1 CPU, Units in MFLOPS ($\times 10^6$)

Machine/CPU Type	LINPACK Performance	Peak Performance
Intel Pentium 4 (2.53 GHz)	2355	5060
NEC SX-6/1 (1proc. 2.0 ns)	7575	8000
HP rx5670 Itanium2 (1GHz)	3528	4000
IBM eServer pSeries 690 (1300 MHz)	2894	5200
Cray SV1ex-1-32(500MHz)	1554	2000
Compaq ES45 (1000 MHz)	1542	2000
AMD Athlon MP1800+(1530MHz)	1705	3060
Intel Pentium III (933 MHz)	507	933
SGI Origin 2000 (300 MHz)	533	600
Intel Pentium II Xeon (450 MHz)	295	450
Sun UltraSPARC (167MHz)	237	333

Mega FLOPS ($\times 10^6$)

Giga FLOPS ($\times 10^9$)

Tera FLOPS ($\times 10^{12}$)

Peta FLOPS ($\times 10^{15}$)

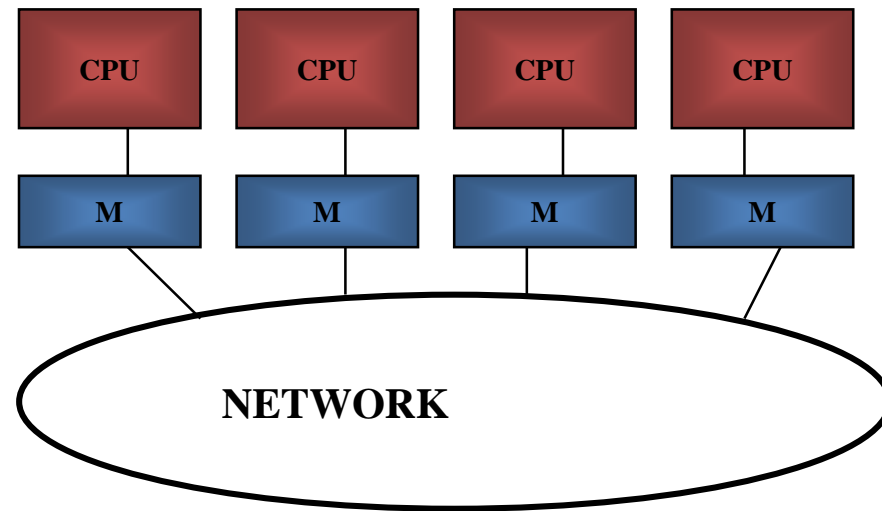
Exa FLOPS ($\times 10^{18}$)

Zetta FLOPS ($\times 10^{21}$)

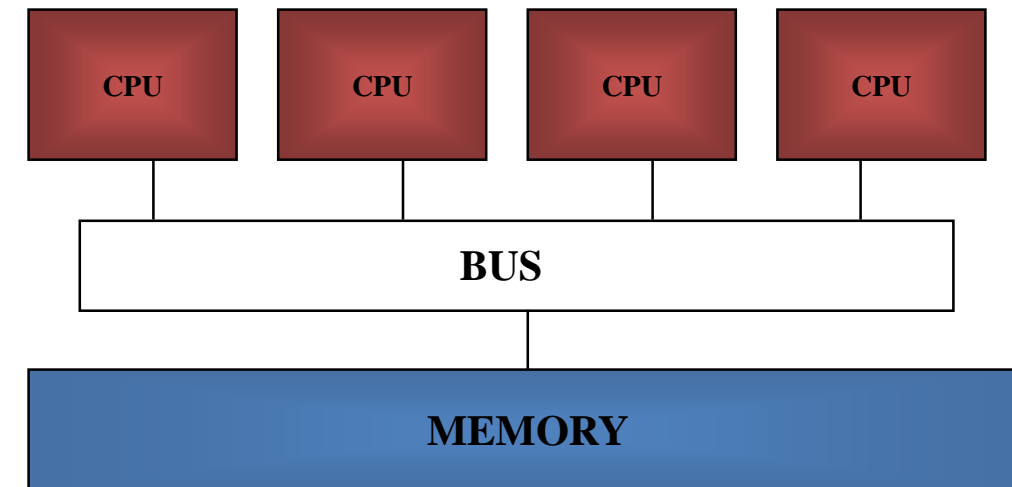
Yotta FLOPS ($\times 10^{24}$)

<http://en.wikipedia.org/wiki/FLOPS>

Shared/Distributed-Memory Architecture



Distributed memory - each processor has its own local memory. Must do message passing to exchange data between processors.
(examples: Emerald, Topsail Clusters)



Shared memory - single address space. All processors have access to a pool of shared memory. (examples: Chastity/zephyr, happy/yatta, cedar/cypress, sunny)
Methods of memory access : Bus and Crossbar

What is a Beowulf Cluster?

- A Beowulf system is a collection of [personal computers](#) constructed from commodity-off-the-shelf hardware components [interconnected](#) with a system-area-network and configured to operate as a single unit, [parallel computing](#) platform (e.g., MPI), using an open-source network operating system such as [LINUX](#).
- Main components:
 - PCs running LINUX OS
 - Inter-node connection with Ethernet, Gigabit, Myrinet, InfiniBand, etc.
 - MPI (message passing interface)

What is Parallel Computing ?

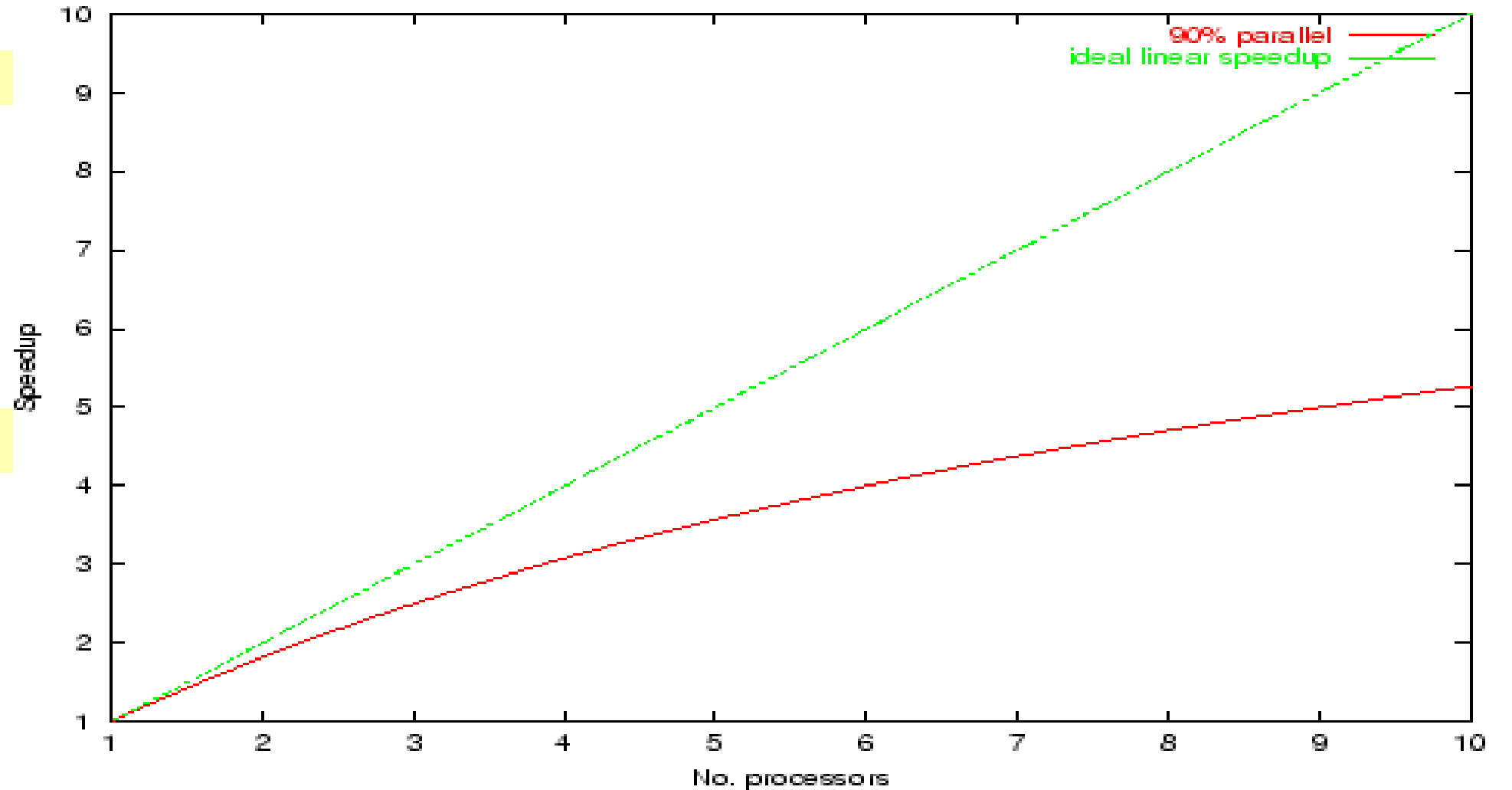
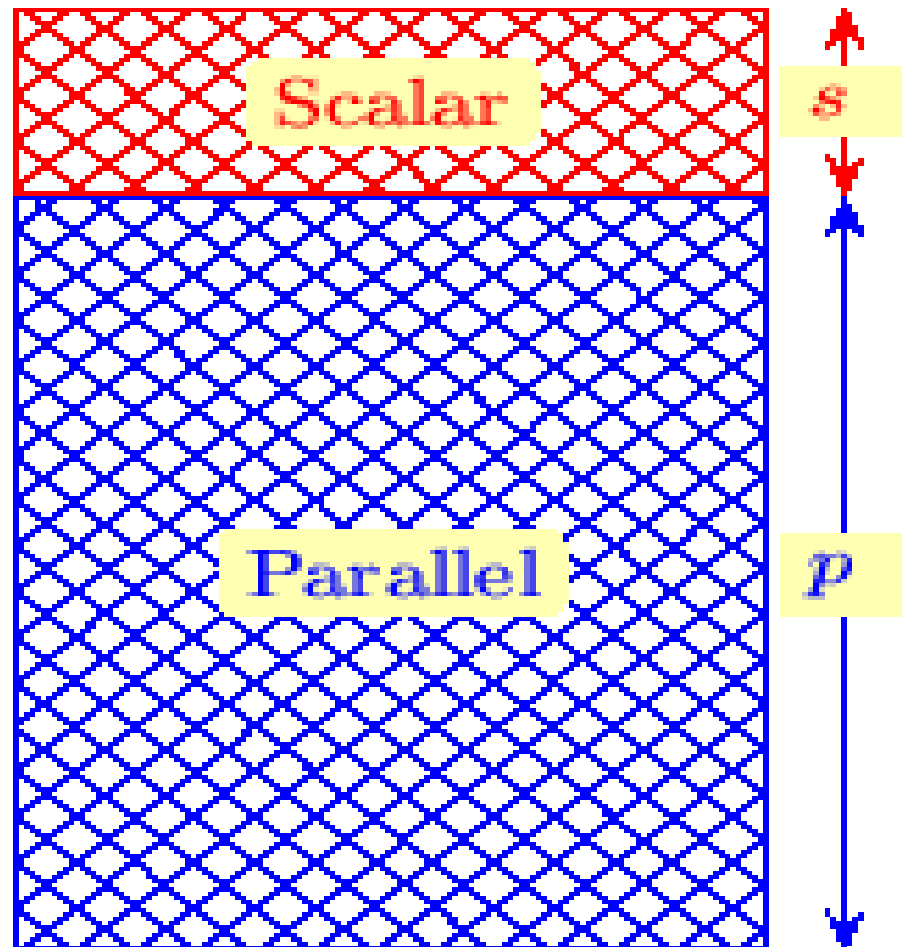
- Concurrent use of multiple processors to process data
 - Running the *same program on many processors*.
 - Running *many programs on each processor*.

Advantages of Parallelization

- **Cheaper**, in terms of Price/Performance Ratio
- **Faster** than equivalently expensive uniprocessor machines
- Handle **bigger** problems
- **More scalable**: the performance of a particular program may be improved by execution on a large machine
- **More reliable**: In theory if processors fail we can simply use others

Catch: Amdahl's Law

$$\text{Speedup} = \frac{1}{s + p/n}$$



Other Parallelization Models

- VIA: Virtual Interface Architecture -- Standards-based Cluster Communications
- PVM: a portable message-passing programming system, designed to link separate host machines to form a "virtual machine" which is a single, manageable computing resource. It's largely an academic effort and there has been no much development since 1990s.
- BSP: Bulk Synchronous Parallel Model, a generalization of the widely researched PRAM (Parallel Random Access Machine) model
- Linda: a concurrent programming model from Yale, with the primary concept of "tuple-space"
- HPF: PGI's first standard parallel programming language for shared and distributed-memory systems.



Intel x86 Evolution

1960s

Development of digital integrated circuits

The Microprocessor Age

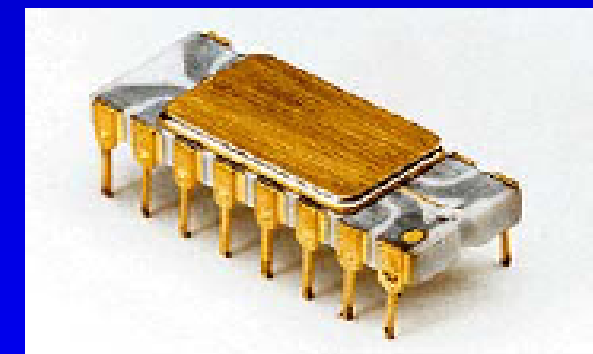
<http://www.computerhistory.org/exhibits/microprocessors/index.page>

November 15, 1971

First Advertisement for Microprocessor Appears.

Intel 4004 has 2,250 transistors,
weighted less than an ounce,

handling data in 4-bit chunks, 45 instructions,
and could perform 50-60K instructions per second.



Year	name	Data size	memory size	#instructions	
1971	4004	4	4096 4-bit	45	first microprocessor
1973	8008	8	16K bytes	48	1st 8-bit μ P
1973	8080	8	64K bytes		10 times faster than 8008
1973	MC6800	8	64K bytes		1st Motorola μ P
1977	8085	8	64K bytes	246	Intel's most successful 8-bit general-purpose μ P due to its low cost
	Z80	8			Zilog's most successful microprocessor
1978	8086	8,16	1M bytes	>20,000	1st 16-bit μ P
1979	8088	8,16	1M bytes		prefetch instruction using cache
1981	IBM decided to use 8088 in its personal computer				
1983	80286	8,16	16M		
1986	80386	8,16,32	4G		
1989	80486	8,16,32	4G		
1993	Pentium	8,16,32	4G		
1995	Pentium Pro	64	64G		
1997	Pentium II	64	64G		
1999	Pentium III	?	?		
2000	Pentium 4	?	?		

Intel x86 Evolution: Milestones

<i>Name</i>	<i>Date</i>	<i>Transistors</i>	<i>MHz</i>
• 8086	1978	29K	5-10
– First 16-bit Intel processor. Basis for IBM PC & DOS			
– 1MB address space			
• 386	1985	275K	16-33
– First 32 bit Intel processor , referred to as IA32			
– Added “flat addressing”, capable of running Unix			
• Pentium 4F	2004	125M	2800-3800
– First 64-bit Intel processor, referred to as x86-64			
• Core 2	2006	291M	1060-3500
– First multi-core Intel processor			
• Core i7	2008	731M	1700-3900
– Four cores (our shark machines)			

Intel IA-32 Family

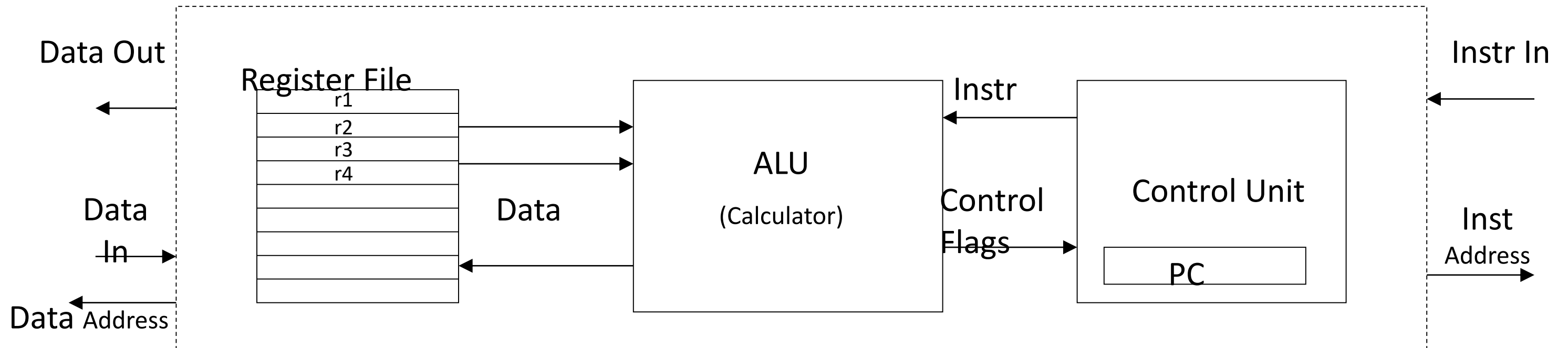
- Intel386
 - 4 GB addressable RAM, 32-bit registers, paging (virtual memory)
- Intel486
 - instruction pipelining
- Pentium
 - superscalar, 32-bit address bus, 64-bit internal data path

Intel P6 Family

- Pentium Pro
 - advanced optimization techniques in microcode
- Pentium II
 - MMX (multimedia) instruction set
- Pentium III
 - SIMD (streaming extensions) instructions
- Pentium 4
 - NetBurst micro-architecture, tuned for multimedia

Internal Structure of the Processor

- Control Unit
 - Fetches instructions from memory, Interprets them, Controls ALU
- ALU
 - Does all computations
- Register File
 - Stores variables



Internal Components of Microprocessors

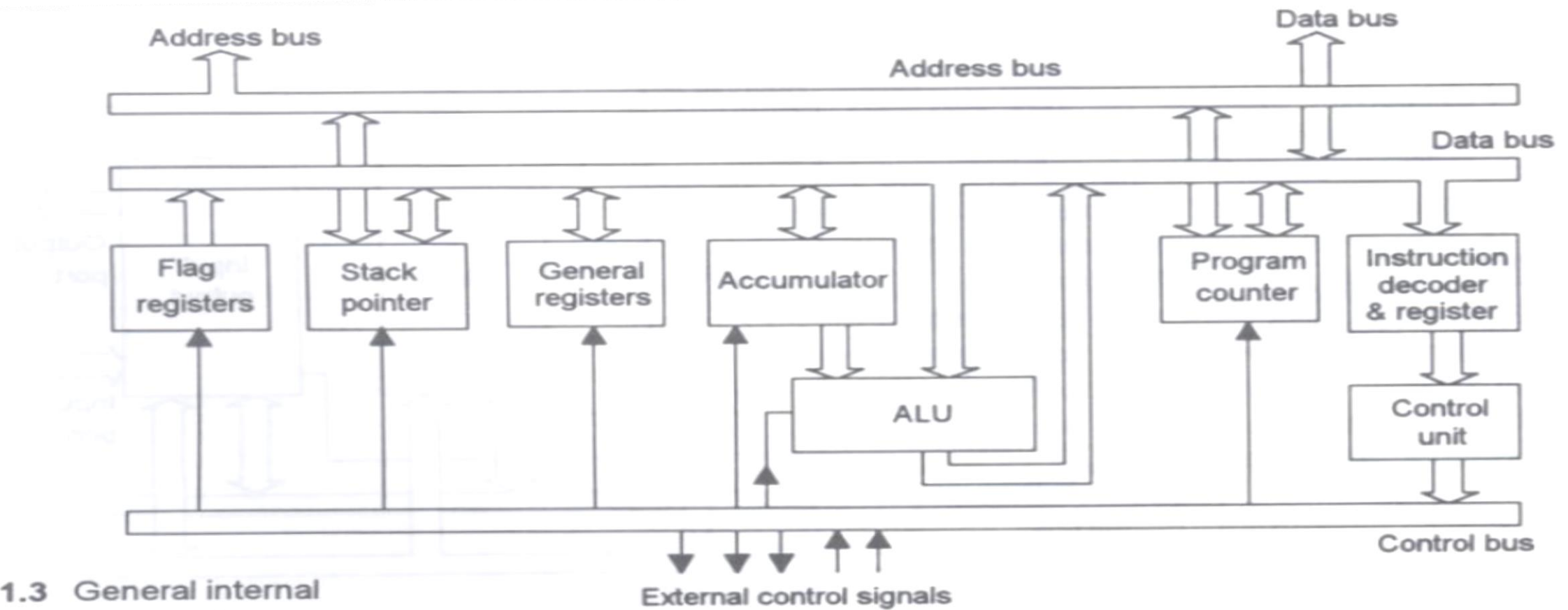


Fig. 1.3 General internal architecture of a microprocessor

CISC – Complex Instruction Set Computers

- Refers to number and complexity of instructions
- Improvements was: Multiply and Divide
- The number of instruction increased from
 - 45 on 4004 to:
 - 246 on 8085
 - 20,000 on 8086 and 8088

RISC – Reduced Instruction Set Computer

- Executes one instruction per clock

Newer RISC - Superscaler Technology

- Execute more than one instruction per clock

CISC and RISC

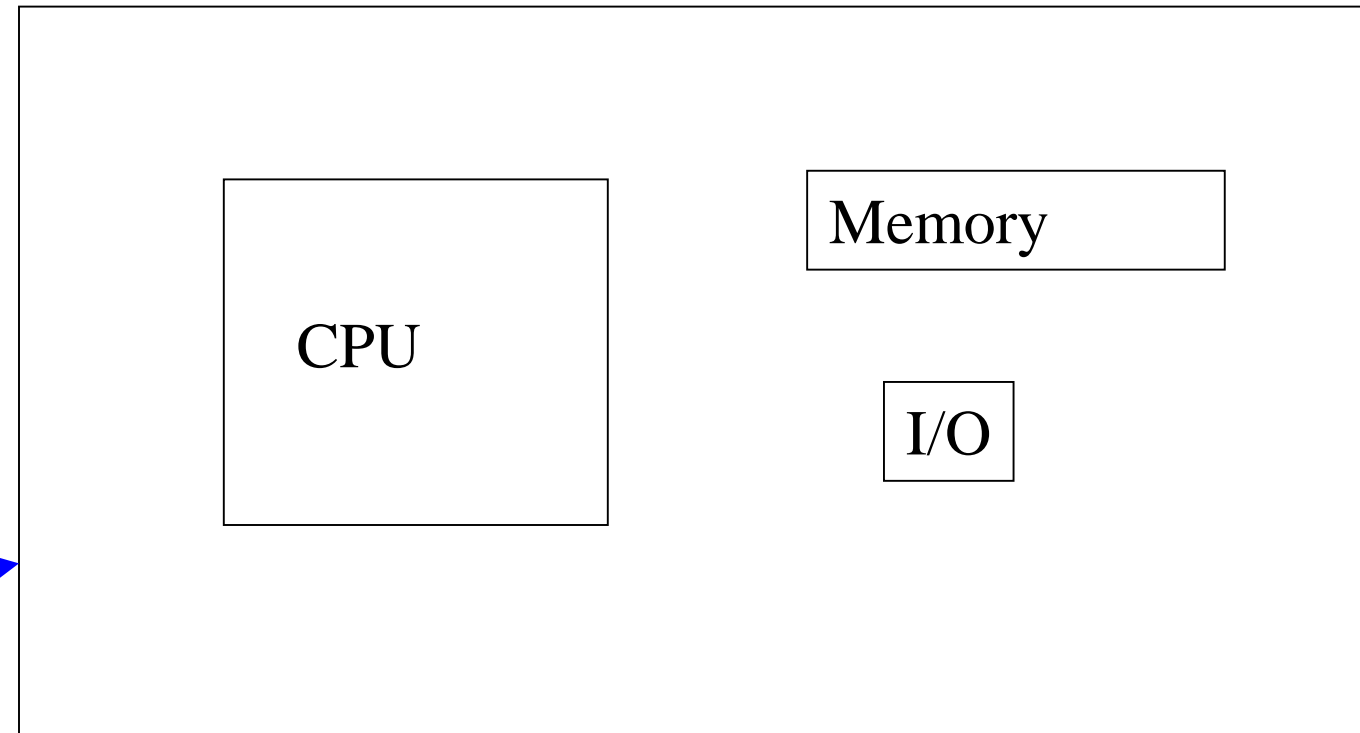
- CISC – complex instruction set
 - large instruction set
 - high-level operations
 - requires microcode interpreter
 - examples: Intel 80x86 family
- RISC – reduced instruction set
 - simple, atomic instructions
 - small instruction set
 - directly executed by hardware
 - examples:
 - ARM (Advanced RISC Machines)
 - DEC Alpha (now Compaq)



Embedded System

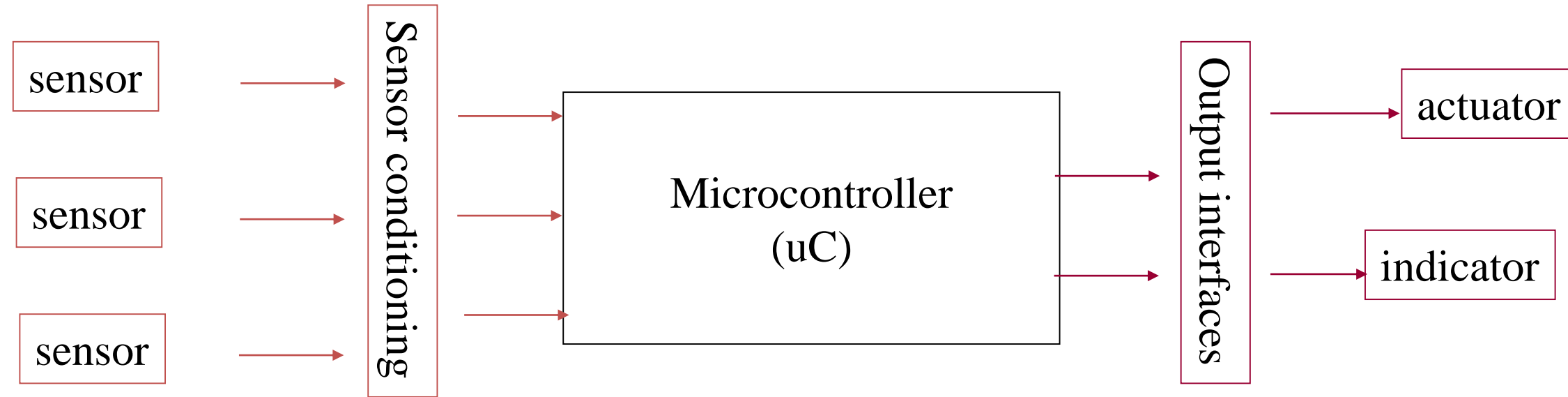
Basic Components of Digital Computer

- CPU
- Memory
- I/O



Could be a chip, a board, or several boards

Embedded System General Block Diagram



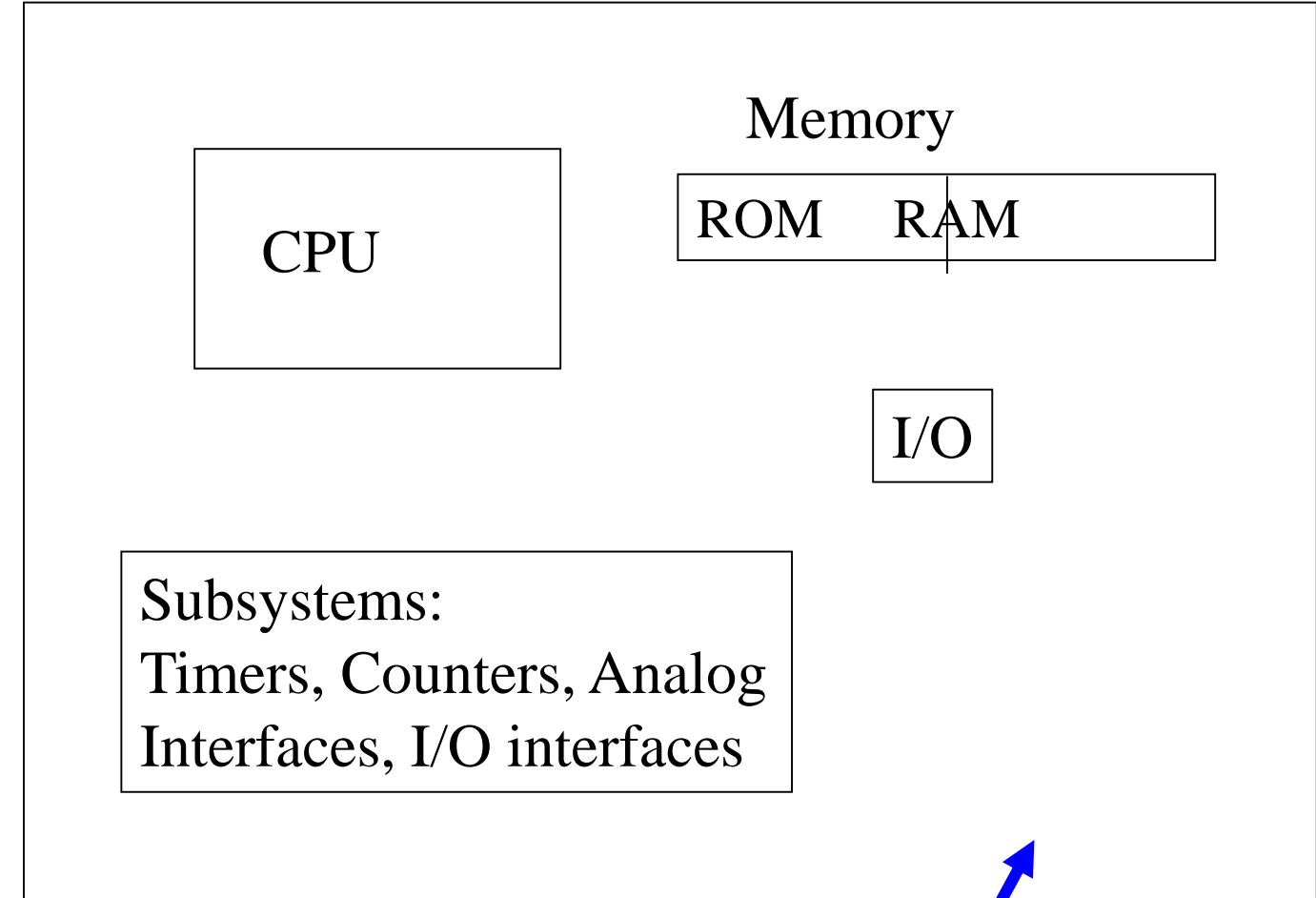
Aktüatör ya da eyleyici: Bir mekanizmayı veya sistemi kontrol eden veya hareket ettiren bir tür motordur. Bir enerji kaynağı tarafından çalıştırılır. Bu kaynak genellikle elektrik akımı, hidrolik akışkan basıncı veya pnömatik basınçtır ve bazı tür hareketlerle enerjiye dönüşür.

Mikrodenetleyiciler

Bir mikroişlemci çekirdeğine ilave olarak, ortak bellek alanlarını kullanan, özelleştirilmiş görevler ile donatılmış çevrebirimlerin eklenmesi ile ortaya çıkan yapıya mikrodenetleyici denir. Denetim teknolojisi gerektiren uygulamalarda kullanılmak üzere tasarlanmış olan mikrodenetleyiciler, mikro işlemcilere göre çok **daha basit ve ucuzdur**.

Mikroişlemcili sistemin tasarımı ve kullanımı mikrodenetleyicili sisteme göre daha karmaşık ve masraflıdır.

Mikrodenetleyicili bir sistemin çalışması için elemanın kendisi ve bir osilatör kaynağının olması yeterlidir. Mikrodenetleyicilerin küçük ve ucuz olmaları, bunların tüm elektronik kontrol devrelerinde kullanılmasını sağlamaktadır.



A single chip

Mikrodenetleyici Seçimi

Mikrodenetleyiciler ile tasarım yapmadan önce tasarlanan sisteme uygun bir denetleyici seçmek için o denetleyicinin taşıdığı özelliklerin bilinmesi gereklidir. Mikrodenetleyicinin hangi özelliklere sahip olduğu kataloglarından anlaşılabilir. Aşağıda sıralanan özellikler bunlardan bazılarıdır;

- Programlanabilir dijital paralel giriş/çıkış.
- Programlanabilir analog giriş/çıkış.
- Seri giriş/çıkış (senkron, asenkron ve cihaz yönetimi).
- Motor veya servo kontrol için pals sinyali çıkışı.
- Harici giriş vasıtasıyla kesme.
- Harici bellek arabirimi.
- Harici veri yolu arabirimi.
- Dahili bellek tipi seçenekleri (ROM, EPROM, PROM, EEPROM).
- Dâhilî RAM seçeneği.
- Kayan nokta hesaplaması.

Mikroişlemci ve Mikrodenetleyiciler Arasındaki Farklar

- Bir mikroişlemci görevini yerine getirebilmesi için mutlaka, verilerin saklanacağı bellek birimine, dış dünyadan veri alışverişinin düzenli yapılmasını sağlayan giriş/çıkış birimine ihtiyaç duyar. Bunlar bir mikroişlemcili sistemde ayrı ayrı birimler (entegreler) şeklinde yerini alır. Bundan dolayı mikroişlemcili sistemlere çok entegreli sistemler denilir.
- Bilgisayar gibi mikroişlemcili sistemlere verilen bir örnekte, bir bilgisayarın bir çamaşır makinesinde veya cep telefonunda kullanılması elbette mümkün olmayacaktır.
- Bilgisayar aynı anda milyonlarca işi yapabildiğinden ve çok yer kapladığından böyle yerlerde kullanılması mantıklı olmaz ve maliyetli olur. Bundan dolayı, sistemi meydana getiren elemanların birçok özelliklerinden feragat edilerek ve bir entegrede birleştirilerek mikroişlemcilerin yeni türevleri (mikrodenetleyiciler) oluşturulmuştur. Bir saydırma veya PWM sinyali üreteceğimizi düşünürsek mikroişlemci ile bunu yazılımsal olarak yapmamız gerekecektir. Ancak mikrodenetleyicinin özelleştirilmiş modülleri sayesinde bu işlemleri programa paralel olarak modüllerle yapabiliriz. Böylece ana programdaki yoğunluk azalır ve işlemcimiz hızlanmış olur.



Quantum Computing

What is a Quantum Computer?

- Quantum Computer (Electron – Foton)
 - A computer that uses the phenomena of quantum mechanics to perform operations on data through devices such as quantum superposition and entanglement.
- Classical Computer (Binary)
 - A computer that uses voltage passing through circuits and gates, which can be calculated entirely through classical mechanics.

Quantum Computing -1

- While silicon-based chips are used in classical computers, information is carried, stored and processed by electrical signals.
- In quantum computers, quantum systems such as subatomic particles, photons or electrons are used. When the processor performs quantization calculations so that it can process data very quickly, bits are represented by electrons or photons.
- Electrons flow from one atom to another in conductors, creating electric current. In order for a current of 1 ampere to occur, 6.25×10^{18} electrons must flow from any point of the conductor in 1 second.
- Transistor is a circuit element produced in semiconductor technology that controls the flow of electrons. Microwave tubes speed up and slow down the flow of electrons.
- In quantum computers, electrical conductors called nanowires are just one atom thick, and a data bit is represented by the superposition and entanglement of an electron.
- Quantum computing behaves according to the laws of quantum mechanics and makes use of concepts such as probabilistic computing, superlocation, and entanglement. These concepts form the basis of quantum algorithms that harness the power of quantum computing to solve complex problems. As probability turns into decision making by guessing, algorithms that provide skills and experience that will increase performance are needed.

Quantum Computing -2

- A quantum computer is a computer that takes advantage of quantum mechanical phenomena. Quantum computing exploits this behavior, particularly quantum superposition and entanglement, by using special hardware that supports the preparation and manipulation of quantum states.
- The basic unit of information in quantum computing is the qubit, similar to bits in traditional digital electronics. Unlike a classical bit, a qubit can exist in a superposition of two "ground" states; This roughly means that it is in both states at the same time. When measuring a qubit, the result is the probabilistic output of a classical bit, so quantum computers are generally non-deterministic. If a quantum computer manipulates the qubit in a certain way, wave interference effects can amplify desired measurement results. The design of quantum algorithms involves creating procedures that allow a quantum computer to perform calculations efficiently and quickly.

Quantum Computing -3

Computer engineers typically describe a modern computer's operation in terms of classical electrodynamics. Within these "classical" computers, some components (such as semiconductors and random number generators) may rely on quantum behavior, but these components are not isolated from their environment, so any quantum information quickly decoheres. While programmers may depend on probability theory when designing a randomized algorithm, quantum mechanical notions like superposition and interference are largely irrelevant for program analysis.

Quantum programs, in contrast, rely on precise control of coherent quantum systems. Physicists describe these systems mathematically using linear algebra. Complex numbers model probability amplitudes, vectors model quantum states, and matrices model the operations that can be performed on these states. Programming a quantum computer is then a matter of composing operations in such a way that the resulting program computes a useful result in theory and is implementable in practice.

Quantum Computing

- Communication between machine and machine (M2M – Machine to Machine)
- Internet of things (IoT: Internet of Things)
- Quantum Computing (QC - Quantum Computing)
- Machine Learning (ML – Machine Learning)
- QC-supported ML and Quantum ML (QML)
- IoT
- Mobile autonomous machines
- 5G, 6G, ... communication

Quantum Machine Learning

- There has been a strong reemergence of research on data analysis applications and intelligent machines in recent years.
- This growing interest is partly due to advances in classical computing methods and partly due to the enormous parallelism potential offered by Quantum Computing (QC) and related quantum technologies.
- These advances in computational methods, such as Machine Learning (ML), data-driven learning and quantum-assisted computational methods, have a strong potential to realize the aspirations of a service-oriented fully intelligent communication network.
- In the emerging paradigm of increasing human-machine connectivity, a significant increase in the number of network nodes and data traffic is expected.
- Machine Learning (ML) and Quantum Computing (QC) methods will offer a new framework that enables Quantum ML (QML) technologies for efficient processing of voluminous data.

Dirac Gösterimi (The Dirac Notation)

- Quantum hesaplama ile birlikte, kubit (qubit) kavramının ihtiyaç duyduğu notasyon Dirac tarafından geliştirilen bir gösterimle karşılanabilmektedir. Bra-ket olarak da geçer.
- Bra-ket gösterimi $\langle | \rangle$ şeklinde sembolize edilebilir. Buradaki **bra** kısmı $\langle |$ olurken **ket** kısmı $| \rangle$ olmuş olur. Yani İngilizcedeki parantez anlamına yakın bir kelimeyi parçalara bölerek (aslında bracket kelimesi, İngilizcede parantez anlamına gelir), parantez iki alt parçada gösterilir.
- $\langle |$, bra gösterimi ulaşmak istediğimiz hali, veya beklediğimiz durumu göstermeye yarar. Örneğin $\langle x=1.5 |$ gösterimi bize, parçacığın, 1.5 konumunda bitmesini istediğimizi veya böyle bir beklentimiz olduğunu gösterir. Bu durumda, örneğin $\langle x=1.5 | x=3 \rangle$ gösterimi, parçacığın 3 konumunda başlayarak 1.5 konumunda bitmesi anlamına gelir.
- $\psi \rangle$ gösterimi, mevcut durumun ψ vektörü olduğunu ifade eder.
- Kubitlerin, klasik bitlerden farklı değerler alabileceğidir. Örneğin kubitler, 0 ve 1 arasındaki herhangi bir doğrusal değeri alabilir.
- $\psi \rangle = \alpha |0\rangle + \beta |1\rangle$ şeklindeki gösterimde, ψ değeri, yukarıda verilen α değeri kadar 0 ve β değeri kadar 1'dir. Yani bu iki değer arasında bir yerde kabul edilen bir vektördür. Bu vektörün uzunluğunu 1 olarak kabul edersek, Pisagor bağlantısından $|\alpha|^2 + |\beta|^2 = 1$ olmalıdır.
- Ket gösterimi, vektörel bir gösterimdir. Diğer bir deyişle, $|v\rangle$ gösterimi aslında $[v]$ şeklinde gösterilebilen bir kolon vektördür. Bra gösterimi ise satır vektördür.

Qubit

- Quantum hesaplamada en küçük bilgi birimine quantum bit denir, klasik sayısal hesaplamadaki bit'e biraz benzemesinden dolayı **qubit** olarak isimlendirilmiştir. Quantum hesaplamasının en temel yapı taşına quantum bit, qubit denir.
- Quantum hesaplamada, temel bilgi birimi Qubittir.
- Qubit(Quantum bit): Bilgiyi depolamak için "quantum" etkisinin (dolaşıklık) üst üste binmesini kullanan temel bir bellek birimi. Bir "Qubit" bilgi olasılığını depolar. Aynı anda hem belirli olasılıkla "1" hem de belirli olasılıkla "0" ı temsil eder.
- Qubit'in iki durumu vardır. Bunlar bit'in
 - Klasik bit'te 0'a karşılık gelen $|0\rangle$ durumu
 - Klasik bit'te 1'e karşılık gelen $|1\rangle$ durumu
- Bunlara Dirac fonksiyonu denir. Vektördürler.
- Qubitler birer fiziksel duruma karşılık gelirler.

What is a qubit?

- A bit has two possible states $|0\rangle$ or $|1\rangle$
- Unlike bits, a qubit can be in a **state other than**
 $|0\rangle$ or $|1\rangle$
- We can form linear combinations of states
 $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$
- A qubit state is a unit vector in a two-dimensional *complex vector space*

Let's Go Buy a Computer

Computer Configuration

- Processor (Pentium IV, 2.8 GHz; Core i5, i7, i9)
- RAM (16GB of SDRAM (expandable))
- Disk (1TB)
- CD ROM/ CD RW/DVD/...
- 17" XGA TFT Display (1280 x 1024 res.)
- S3 Savage IX 128-bit AGP 2x graphics
 - 8MB memory, 3D Hardware acceleration, composite TV-Out support,
 - ...
- 16-bit Sound
- HDMI
- Ethernet
- 2 USB Ports
- Built-in 10/100 Ethernet Adapter and 802.11b/g wireless