

Bilgisayar Mimarisi

- **Bilgisayar Mimarisi;** bir bilgisayar sisteminin nasıl tasarlandığını ve çalıştığını belirleyen kavramlar bütünüdür. Bilgisayar mimarisi, hem donanım hem de yazılım bileşenlerini kapsar ve bu bileşenlerin birbiriyle nasıl etkileşim kurduğunu tanımlar.
- **Bilgisayar mimarisi;** Bellek, ALU, kaydediciler, I/O birimi, yol arabirimi, kontrol birimi gibi birimlerin tasarlanması, çalışma prensiplerinin öğrenilmesi, verilerin bu birimler arasında taşınması gibi işlemlerin nasıl yapıldığının öğrenilmesi ve bu birimlerin programlanması ile ilgilenir.

Bilgisayar Mimarisi Ana Bileşenleri

- Merkezi İşlem Birimi (CPU):
 - Bilgisayarın beyni olarak kabul edilir.
 - Komutları işler, hesaplamaları yapar ve kontrol mekanizmalarını yönetir.
 - CPU, aşağıdaki temel bileşenlerden oluşur:
 - **Kontrol Birimi (CU):** Komutları yorumlar ve diğer bileşenlere ne yapması gerektiğini söyler.
 - **Aritmetik ve Mantık Birimi (ALU):** Matematiksel ve mantıksal işlemleri gerçekleştirir.
 - **Kayıtlar (Registers):** Geçici veri saklama birimleridir.
- Bellek:
 - İşlemci tarafından hızlı bir şekilde erişilebilen verilerin saklandığı yerdir.
 - Türleri:
 - **RAM (Rastgele Erişimli Bellek):** Geçici depolama birimi.
 - **ROM (Salt Okunur Bellek):** Değiştirilemeyen, kalıcı veriler içerir.

- Giriş/Çıkış (I/O) Sistemleri:
 - Kullanıcı ile bilgisayar arasında etkileşimi sağlar.
 - Klavye, fare, monitör gibi cihazlar bu gruba dahildir.
- Veriyolları (Buses):
 - Bilgisayar bileşenleri arasındaki veri iletişim kanallarıdır.
 - Türleri:
 - **Adres Veri Yolu:** Bellekteki yerlerin adreslenmesini sağlar.
 - **Veri Yolu:** Verilerin taşınmasını sağlar.
 - **Kontrol Yolu:** İşlemleri koordine eder.

Mikroişlemci - Mikrodenetleyici

- **Mikroişlemci**, bir bilgisayarın veya elektronik cihazın merkezi işlem birimi (CPU) olarak görev yapan, entegre devre (IC) üzerine yerleştirilmiş bir elektronik çiptir. Mikroişlemciler, matematiksel ve mantıksal işlemleri gerçekleştiren, veri işleyen ve bir sistemin işleyişini kontrol eden temel bileşenlerdir.
- **Mikrodenetleyici**, bir mikroişlemcinin yanı sıra bellek (RAM, ROM), giriş/çıkış birimleri (I/O) ve diğer çevresel birimlerin tek bir entegre devre (IC) üzerinde birleştirildiği kompakt bir elektronik bileşendir. Genellikle gömülü sistemlerde kullanılır ve bir sistemi baştan sona kontrol edebilecek şekilde tasarlanmıştır. Bir **mikrodenetleyici**, mikroişlemcinin, kristal osilatör, zamanlayıcılar, seri ve analog giriş çıkışlar, programlanabilir hafıza gibi bileşenlerle tek bir tümleşik devre üzerinde üretilmiş halidir (Pic, STM32).

Mikroişlemcinin Temel Görevleri

- **Komut Yürütme:** İşlemciye verilen talimatları (komut setini) işler.
- **Veri İşleme:** Matematiksel ve mantıksal işlemleri gerçekleştirir.
- **Kontrol:** Sistemin diğer bileşenlerini (bellek, giriş/çıkış birimleri vb.) koordine eder.
- **Veri Depolama:** Geçici olarak işlenen verileri kayıtlarda saklar.

Mikroişlemcinin Temel Bileşenleri

- Aritmetik ve Mantık Birimi (ALU):
 - Matematiksel (toplama, çıkarma) ve mantıksal (AND, OR, NOT) işlemleri yapar.
- Kontrol Birimi (CU):
 - Mikroişlemci içindeki ve çevresindeki işlemleri denetler.
 - Komutları yorumlar ve işlem sırasını belirler.
- Kayıtlar (Registers):
 - Verilerin ve adreslerin geçici olarak saklandığı yüksek hızlı depolama alanlarıdır.
- Saat (Clock):
 - Mikroişlemcinin çalışma hızını belirleyen sinyal üreticisidir. Saat hızı genellikle MHz veya GHz cinsinden ifade edilir.
- Veriyolları (Buses):
 - Veri, adres ve kontrol sinyallerini taşır.
 - Üç temel türü vardır:
 - **Veri Yolu:** Veri transferi sağlar.
 - **Adres Yolu:** Bellek veya cihazların adreslenmesi için kullanılır.
 - **Kontrol Yolu:** İşlemci ve diğer bileşenler arasındaki kontrol sinyallerini taşır.

- Günümüzde genel kullanım için hemen hemen tüm CPU'lar mikroişlemcilerdir denilebilir. Onun haricinde akıllı telefonlar, gömülü sistemler ve endüstriyel otomasyon sistemlerinde kullanılmaktadır.
- İlk mikroişlemci olan **Intel 4004**, 1971 yılında Intel tarafından geliştirilmiştir.
- 4-bitlik bir işlemciydi ve hesap makinelerinde kullanıldı.
- Daha sonraki yıllarda **Intel 8086** (16-bit), **Motorola 68000** ve diğer gelişmiş modeller piyasaya sürülerek, mikroişlemciler bilgisayarlarda, elektronik cihazlarda ve endüstriyel uygulamalarda yaygın olarak kullanılmaya başlandı.

Mikroişlemci Çeşitleri

- RISC İşlemciler (Reduced Instruction Set Computing):
 - Az sayıda basit komut seti kullanır.
 - Örnek: ARM işlemciler.
- CISC İşlemciler (Complex Instruction Set Computing):
 - Daha geniş ve karmaşık komut seti kullanır.
 - Örnek: Intel x86 işlemciler.
- Gömülü Mikroişlemciler:
 - Belirli bir işlevi yerine getirmek için özelleştirilmiştir.
 - Örnek: Arduino, Raspberry Pi gibi platformlarda kullanılan mikroişlemciler.

Mikrodenetleyicinin Önemi

- Entegre Yapı:
 - Mikroişlemci (CPU), bellek (RAM ve ROM), zamanlayıcılar, giriş/çıkış birimleri ve çevresel arabirimler tek bir çipte birleştirilmiştir.
- Düşük Güç Tüketimi:
 - Genellikle düşük enerji tüketir ve batarya ile çalışabilen cihazlar için idealdir.
- Gerçek Zamanlı Çalışma:
 - Zamanlayıcı ve kesme (interrupt) mekanizmaları sayesinde gerçek zamanlı işlemleri yönetebilir.
- Özelleştirilebilirlik:
 - Programlanabilir bir yapıya sahiptir; belirli bir uygulama için tasarlanabilir ve optimize edilebilir.

Mikrodenetleyicinin Temel Bileşenleri

- Mikroişlemci (CPU):
 - Verileri işler ve karar verme mekanizmasını sağlar.
- Bellek:
 - **RAM:** Geçici veri depolama alanı.
 - **ROM/Flash:** Program kodlarını ve kalıcı verileri saklar.
- Giriş/Çıkış Portları (I/O):
 - Sensörler, motorlar, ekranlar gibi dış cihazlarla etkileşim kurar.
- Zamanlayıcılar ve Sayaçlar:
 - Zaman kontrolü ve ölçüm işlemleri için kullanılır.
- Kesme (Interrupt) Sistemi:
 - Acil durumlarda işlemcinin ana işini durdurup belirli bir görevi yerine getirmesini sağlar.
- ADC/DAC (Analog-Dijital Çeviriciler):
 - Analog sinyalleri dijitale dönüştürmek veya dijital sinyalleri analog hale getirmek için kullanılır.

Mikrodenetleyicinin Çalışma Yapısı

- Programlama:
 - Mikrodenetleyici, özel bir programlama dili (genellikle C veya Assembly) ile programlanır.
 - Programlanan kod ROM veya Flash belleğe yüklenir.
- Veri İşleme:
 - Sensörlerden gelen veriler mikroişlemci tarafından işlenir.
 - İşleme sonucunda motorlar, LED'ler veya diğer çıkış cihazları kontrol edilir.
- Gerçek Zamanlı Kontrol:
 - Sistem gereksinimlerine göre belirli bir süre içinde işlemler gerçekleştirilir.

Mikrodenetleyici Çeşitleri

- 8-Bit Mikrodenetleyiciler:
 - Basit uygulamalar için uygundur (örneğin: ATmega328 - Arduino Uno'da kullanılır).
- 16-Bit Mikrodenetleyiciler:
 - Orta düzey uygulamalarda kullanılır (örneğin: MSP430).
- 32-Bit Mikrodenetleyiciler:
 - Daha karmaşık ve yüksek performans gerektiren uygulamalara uygundur (örneğin: ARM Cortex-M serisi).

Mikrodenetleyici Kullanım Alanları

- Tüketici Elektroniği:
 - Televizyonlar, uzaktan kumandalar, çamaşır makineleri, klimalar.
- Endüstriyel Otomasyon:
 - Robotik kollar, sensör tabanlı kontrol sistemleri. (PIC Serisi)
- Otomotiv:
 - Hız sabitleyiciler, hava yastıkları, motor kontrol sistemleri.
- IoT Cihazları (Nesnelerin İnterneti):
 - Akıllı ev cihazları, giyilebilir teknoloji. (ESP8266/ESP32)
- Tıbbi Cihazlar:
 - Kan şekeri ölçüm cihazları, kalp atışı monitörleri.

Özellik	Mikrodenetleyici	Mikroişlemci
Bileşenler	CPU, bellek, I/O birimleri entegredir	Yalnızca CPU, diğer bileşenler ayrı
Kullanım Alanı	Gömülü sistemler	Genel amaçlı bilgisayarlar
Maliyet	Daha ucuz	Daha pahalı
Boyut	Daha küçük	Daha büyük
Enerji Tüketimi	Daha düşük	Daha yüksek

Sayı Sistemleri

- Günümüz bilgisayarları dijital sistemlerdir. Dijital sistemdeki her sinyal AÇIK (ON) veya KAPALI (OFF) şekilde ifade edilir.
- Sayıları temsil ederken de açık ve kapalı olma durumu kullanılır. Bu sayede kapalı ve açık 0 ve 1 rakamlarını temsil edebilir.
- Bit (Binary Digit) ise ikili sayı olup, bir bit 0 veya 1 değerine sahip olabilir. İkili sayı sisteminde (**Binary**) bir örnek şu şekildedir:

$$11_{10} = 1*2^3 + 1*2^1 + 1*2^0 = 1011_2$$

$$11_{10} = 8 + 2 + 1$$

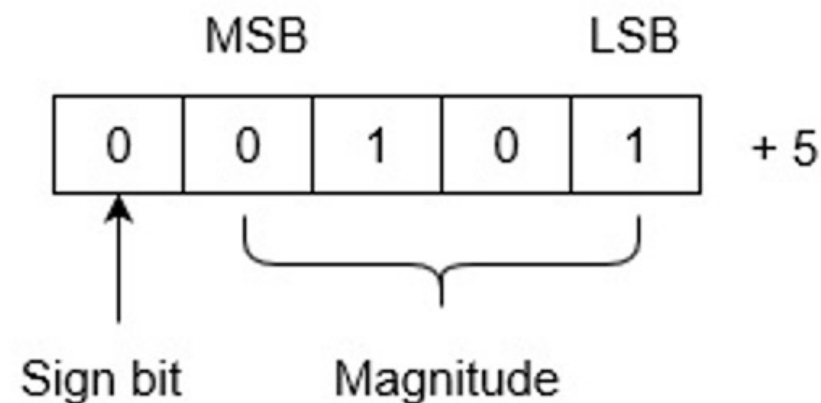
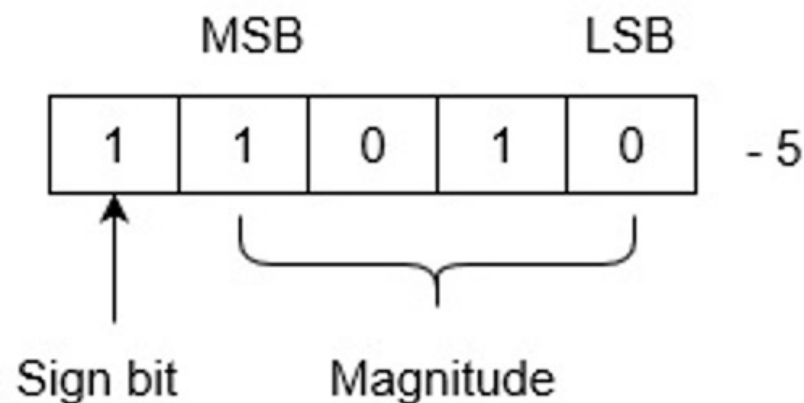
Hexadecimal Sayı Sistemleri

Hexadecimal	Binary	Decimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
A	1010	10
B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15

$$\begin{aligned} 5A3_{16} &= 5 \cdot 16^2 + 10 \cdot 16^1 + 3 \cdot 16^0 \\ &= 5 \cdot 256 + 10 \cdot 16 + 3 \cdot 1 \\ &= 1280 + 160 + 3 \\ &= 1443_{10} \end{aligned}$$

Binary Sayılarda İşaret

- $010110_2 = 22_{10}$
- $101001_2 = -22_{10}$
- **1'in Tümleyeni Yaklaşımı (1's Complement):** 1'in tümleyen ikili sayıları İşaretli sayı gösteriminde çok faydalıdır. **Eski Bilgisayar Sistemlerinde:** Negatif sayı temsili için kullanılmıştır. **Mantıksal Operasyonlarda:** Bit düzeyinde işlemlerde yaygındır.
- Pozitif sayılar basitçe İkili sayı (binary) olarak temsil edilir. Pozitif ikili sayı için yapılacak hiçbir şey yoktur. Ancak negatif ikili sayı gösterimi durumunda 1'in tümleyeniyle temsil ederiz. Önce sayıyı pozitif işaretle temsil edip, sonra bu sayının 1'e tümleyeni alınır. Kolay olarak; Pozitif gösterimdeki **bitlerin tersini al**.



- **1'e tümlleme** ve **2'ye tümlleme** işlemleri, özellikle bilgisayar sistemlerinde **negatif sayıların temsil edilmesi** ve **aritmetik işlemlerin basitleştirilmesi** amacıyla yapılır.
- Bilgisayarlar toplama işlemini kolayca yapabilir, ancak **çıkarma** işlemi daha karmaşıktır. Negatif sayılar **1'e tümlleme** veya **2'ye tümlleme** yöntemiyle temsil edilirse, çıkarma işlemi şu şekilde basitleşir:

$$A - B = A + (-B)$$

- Bu durumda -B, ya 1'e ya da 2'ye tümlleme yöntemiyle bulunur ve toplama işlemi yapılır.

- İşaret (Sign) Biti:
 - İşaret biti (en yüksek dereceli bit, MSB), sayının pozitif mi negatif mi olduğunu belirtir.

0 → Pozitif

1 → Negatif

- İşaret biti dikkate alınmazsa sayının pozitif ya da negatif olduğu yanlış yorumlanabilir.
- Örneğin, 1010 hem negatif bir sayı hem de pozitif bir sayı olarak yorumlanabilir.

- Taşıma (Carry) Biti:
- İkili aritmetikte, toplama işlemi sırasında bir sonraki haneye taşan 1'dir.
- 1'in tümleyenini aritmetiğinde taşma sonucu oluşan "carry out" genellikle geri döndürülerek toplamın doğru olması sağlanır (**end-around carry**).
- Taşıma biti kullanılmazsa sonuç hatalı olur çünkü taşma, toplamın doğruluğunu etkiler.

- 4 Bitlik Sistem - (+5) + (-3)

+5: 0101

-3: 1100 (1'in tümleyeni)

Toplama işlemi:

```
  0101  (+5)
+ 1100  (-3)
-----
 10001  (Sonuç = Taşmalı)
```

```
  0001  (4 bit toplam)
+    1  (end-around carry)
-----
  0010  (Sonuç = +2)
```

- Negatif Sayıların Toplamı - $(-2) + (-3)$

-2: 1101 (1'in tümleyeni)

-3: 1100 (1'in tümleyeni)

Toplama işlemi:

$$\begin{array}{r} 1101 \quad (-2) \\ + 1100 \quad (-3) \\ \hline 11001 \quad (\text{Sonuç} = \text{Taşmalı}) \end{array}$$

$$\begin{array}{r} 1001 \quad (4 \text{ bit toplam}) \\ + 1 \quad (\text{end-around carry}) \\ \hline 1010 \quad (\text{Sonuç} = -5) \end{array}$$

İkinin Tümleyeni (Two's Complement)

- Modern bilgisayar sistemlerinde negatif sayıları temsil etmek ve aritmetik işlemleri kolaylaştırmak için yaygın olarak kullanılan bir yöntemdir. **1'in tümleyeni** yöntemindeki bazı dezavantajları giderir ve toplama-çıkarma işlemlerini daha basit hale getirir.
- - **1**'in tümleyeninde hem **$^{**}+0^{**}$** (**0000**) hem de **$^{**-0^{**}}$** (**1111**) olduğundan, iki sıfır durumu oluşur. Bu durum ikinin tümleyeninde yoktur. İkinin tümleyeninde sadece **** tek bir sıfır durumu vardır **** : **0000**.
- Bir sayının **ikinin tümleyeni**, aşağıdaki iki adımla hesaplanır:
 - **1'in tümleyeni** alınır (her bit ters çevrilir: $1 \rightarrow 0, 0 \rightarrow 1$).
 - Sonuca 1 eklenir.

- İkkinin tümleyeninde, negatif sayılar doğrudan toplama işlemine dahil edilebilir. Ayrı bir çıkarma devresine gerek yoktur.
- Örneğin:

$$+5 = 0101$$

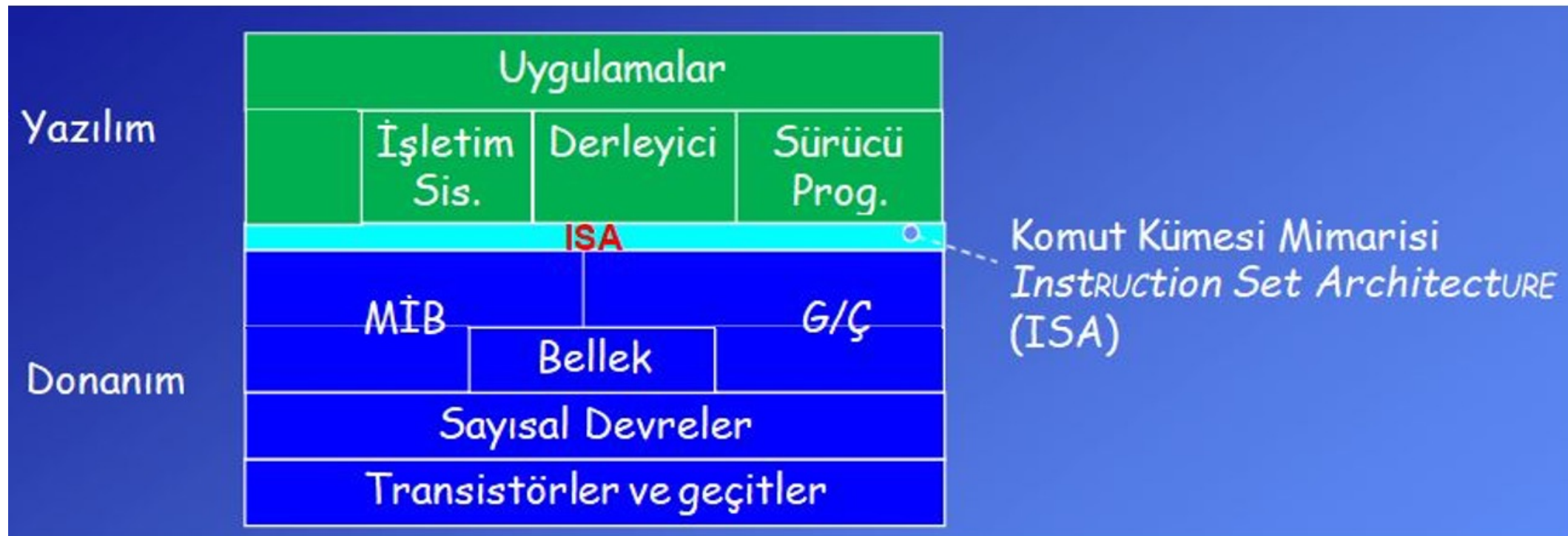
$$-3 = 1101 \text{ (ikinin tümleyeni)}$$

Toplam:

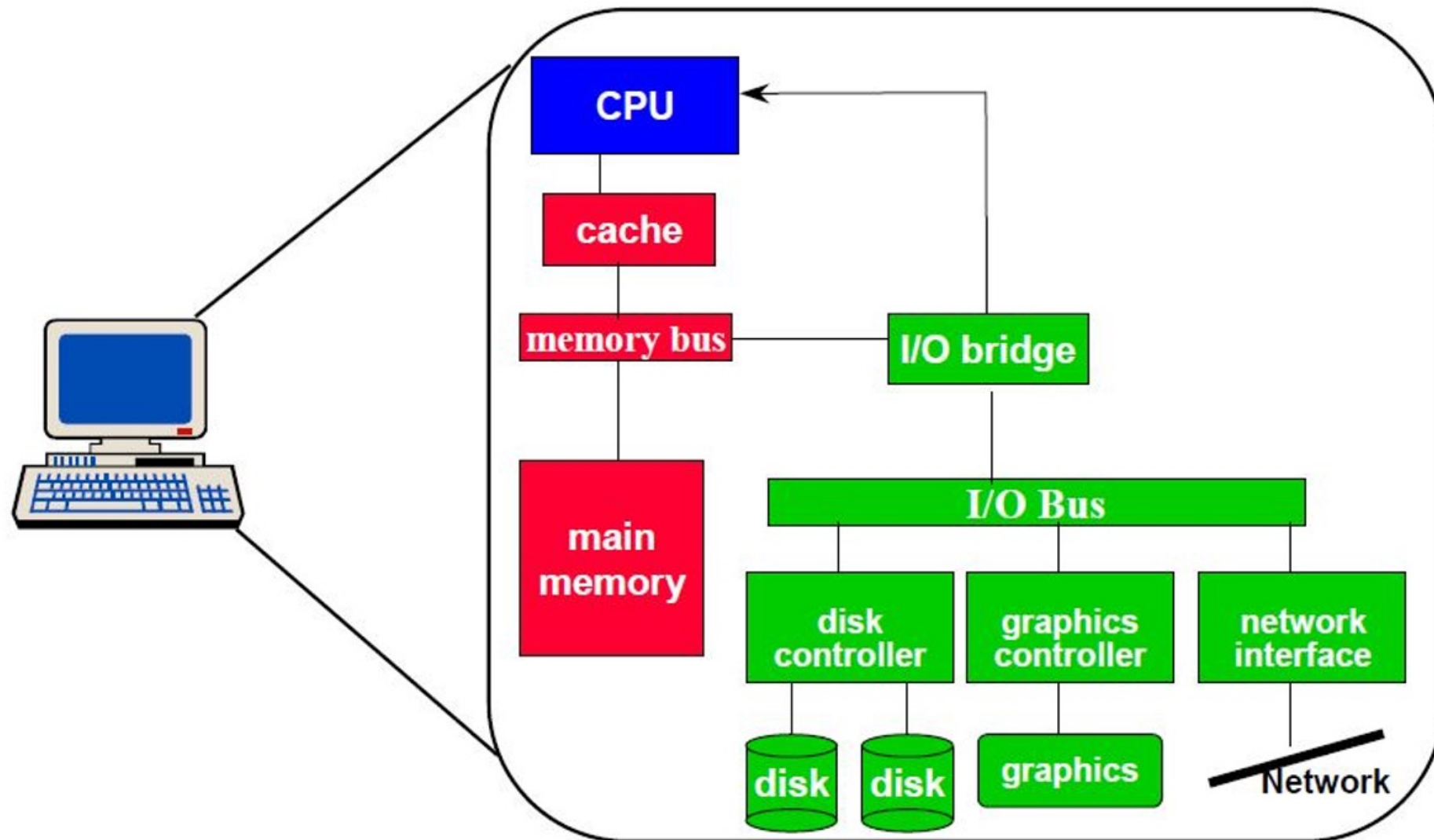
$$\begin{array}{r} 0101 \\ + 1101 \\ \hline 0010 \end{array} \quad (\text{Sonuç: } +2)$$

Bilgisayar Mimarisi (Computer Architecture)

- Bilgisayar Mimarisi, donanım ve yazılım arasındaki arayüzü oluşturur.
- Komut kümesi mimarisi (ISA) bir bilgisayar donanımının, alt düzey programcıya (sistem programlarına) görünen yüzüdür.
- ISA, makine dili komutları, programcıya görünen saklayıcıları ve işlemcinin doğal veri yapılarını içerir.



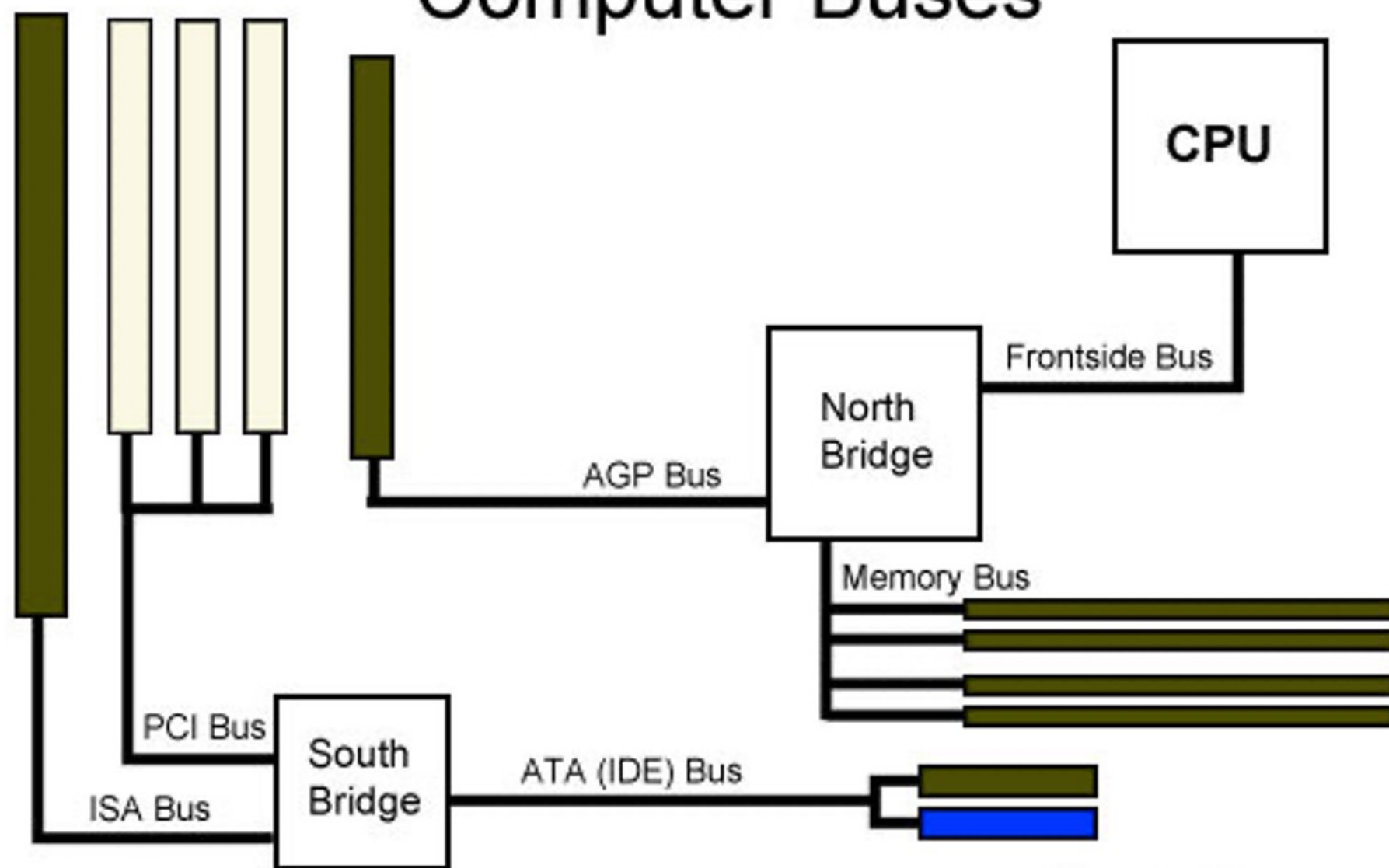
System Organization



Veriyolu (BUS)

- Veriyolu, bir bilgisayara bağlı bileşenler veya cihazlar arasındaki bağlantıdır.
- Bilgisayar bileşenlerinin birbirlerine bağlanmalarındaki en önemli yöntem “veriyolu (bus)” olarak adlandırılan birimlerin kullanımudur.
- Veriyolu, üzerine değişik bileşenlerin bağlanabilmelerine imkan verebilen, birbirine paralel elektrik iletkenlerinin biraraya gelmeleri ile oluşmaktadır.
- Veriyolları:
 - Veri sinyallerini,
 - Verilerin adres sinyallerini,
 - Kontrol sinyallerini
 - Gücüiletirler.
- İki temel veriyolu türü vardır:
 - İç Veriyolu (*Internal bus*): İşlemcinin içinde bulunmakta ve işlemci tasarımının bir parçasıdır.
 - Dış Veriyolu (*External bus*): Ayrı donanım elemanlarını birbirine bağlamak için kullanılmaktadır.

Computer Buses



- İşlemcinin data veri yolu (**data bus**) ve adres yolu (**address bus**) genişliği önemli kavramlar arasında gelmektedir. Veri yolu (bus); sinyalleri taşımak için tasarlanmış bağlantılar dizisidir.
- Mikroişlemcinin yolu denildiğinde ilk akla gelen, veri göndermek ve almak için kullanılan tel kümesidir. Birim zamanda ne kadar çok sinyal gönderilebilirse, o kadar çok veri transfer edilebilir ve **veri yolu** o kadar hızlı olur.

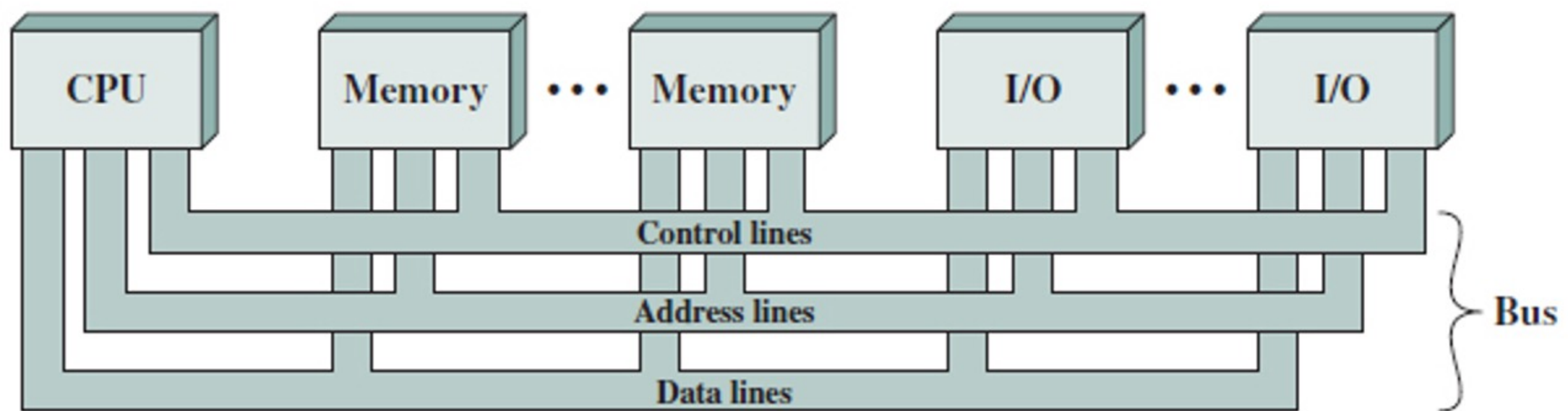


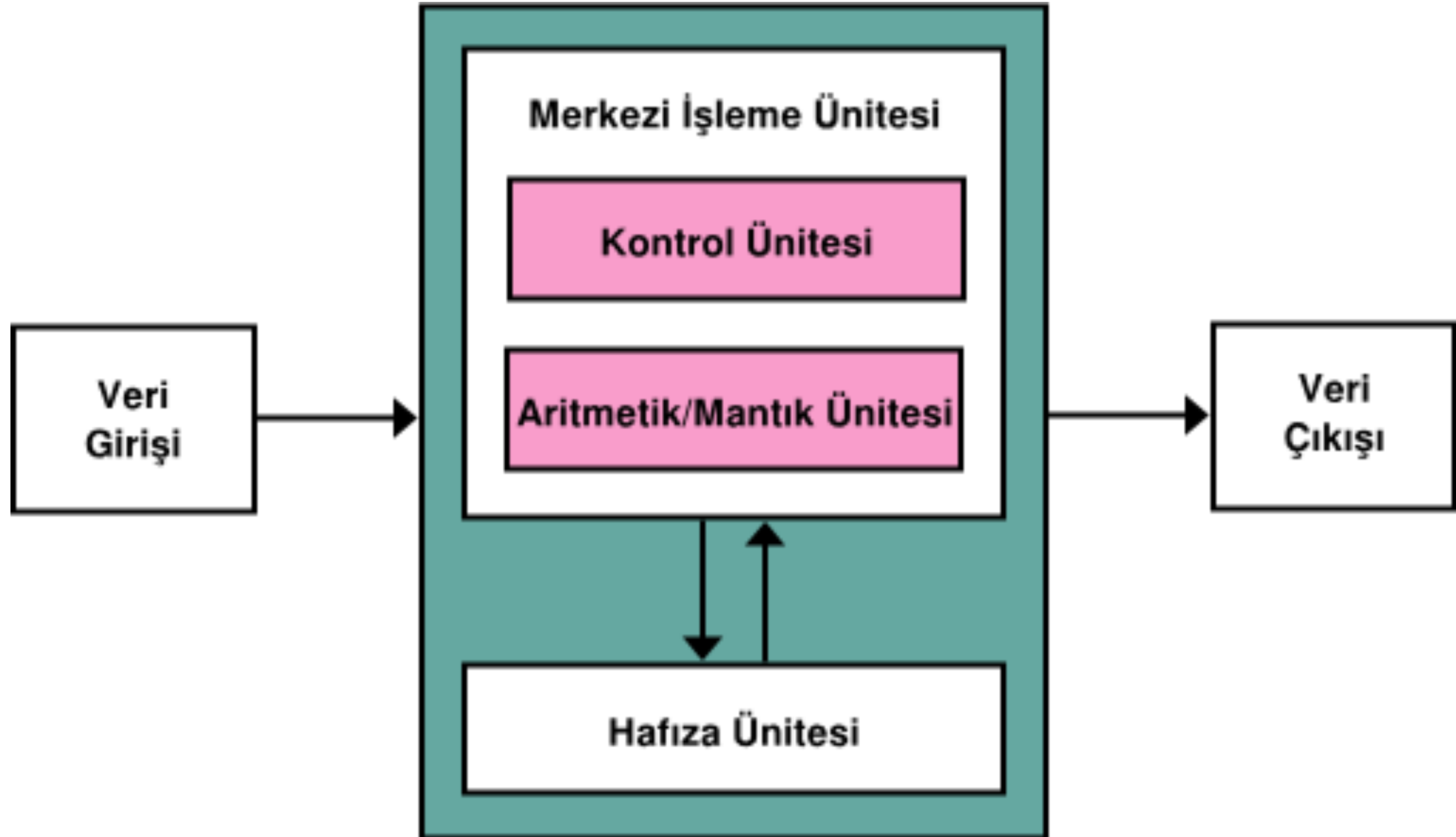
Figure Bus Interconnection Scheme

- Veri yolundan tamamen farklı olan **adres yolu**, verinin gönderileceği veya alınacağı bellek konumunu bildiren adresleme bilgisini taşıyan bir dizi telden oluşur.
- Veri yolunda olduğu gibi adres yolundaki her bir telde yalnızca tek bir bitlik bilgi taşır. Bu tek bit, söz konusu adresin tek bir basamağını oluşturur.
- Bu nedenle, tel sayısı arttıkça adreslenecek toplam bellek miktarı da artar. Adres yolunun genişliği, bir yonganın adresleyebileceği maksimum RAM miktarını belirler.

- Assembly programlama dili düşük seviyeli bir dil olup Assembly dili ile program yazarken kullanılan bilgisayarın donanımsal özelliklerinin bilinmesi gerekir. Yazılan program kullanılan mikroişlemcinin yapısına bağlıdır.
- Assembly dili ile program yazarken programcı doğrudan bilgisayarın işlemcisi ve hafızası ile uğraşır. Anabellekteki (RAM'deki) ve işlemci kaydedicilerindeki (registers) değerleri doğrudan değiştirebilme imkanı vardır.
- Yüksek seviyeli dillerle karşılaştırıldığında assembly dilinde yazılan programlar daha hızlıdır ve bellekte daha az yer kaplar. Buna karşılık program yazmak yüksek seviyeli dillerde daha kolaydır.

Von Neumann Mimarisi

- Özellikleri:
 - Program ve veri, aynı bellek üzerinde depolanır.
 - İşlemci, bellekten komutları sıralı bir şekilde okur ve işler.
 - Bir veri yolu üzerinden işlem yapar.
- Avantajları:
 - Tasarımı basittir ve uygulaması kolaydır.
 - Genel amaçlı bilgisayarlar için uygundur.
- Dezavantajları:
 - Von Neumann darboğazı, bellek ve işlemci arasındaki veri iletimi sırasında ortaya çıkar. "Von Neumann darboğazı" olarak bilinen bellek ve işlemci arasındaki veri transferi sınırlaması performansı düşürür.

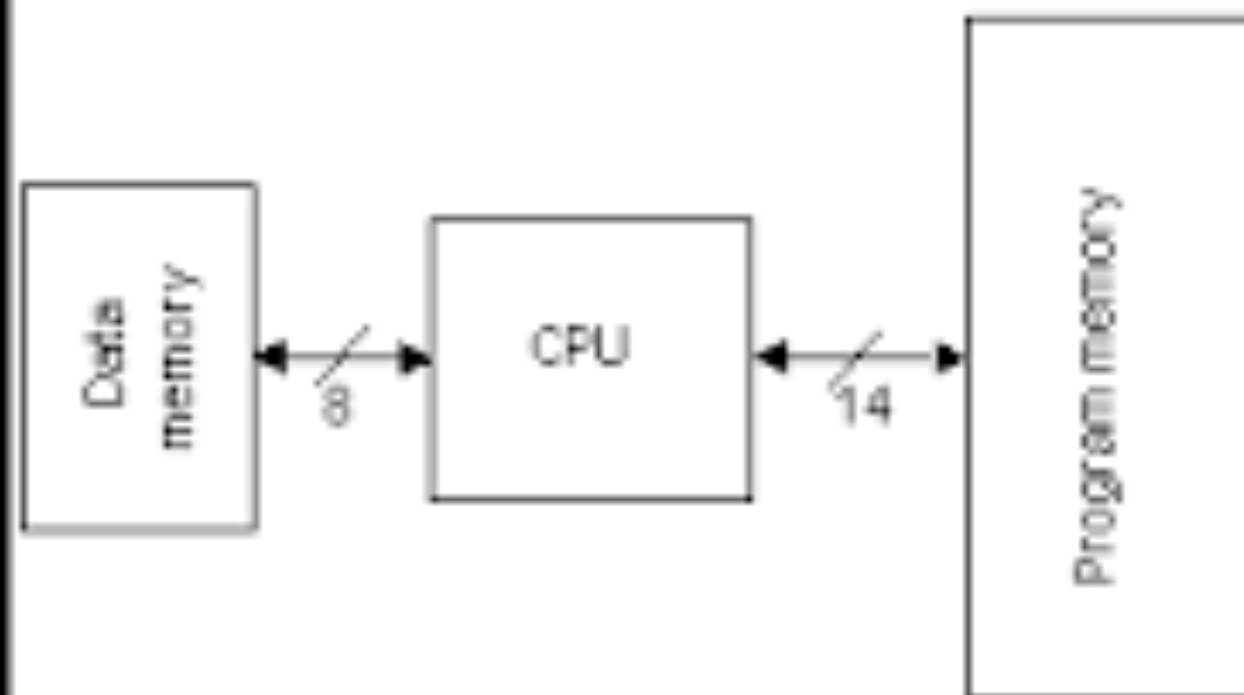


Harvard Mimarisi

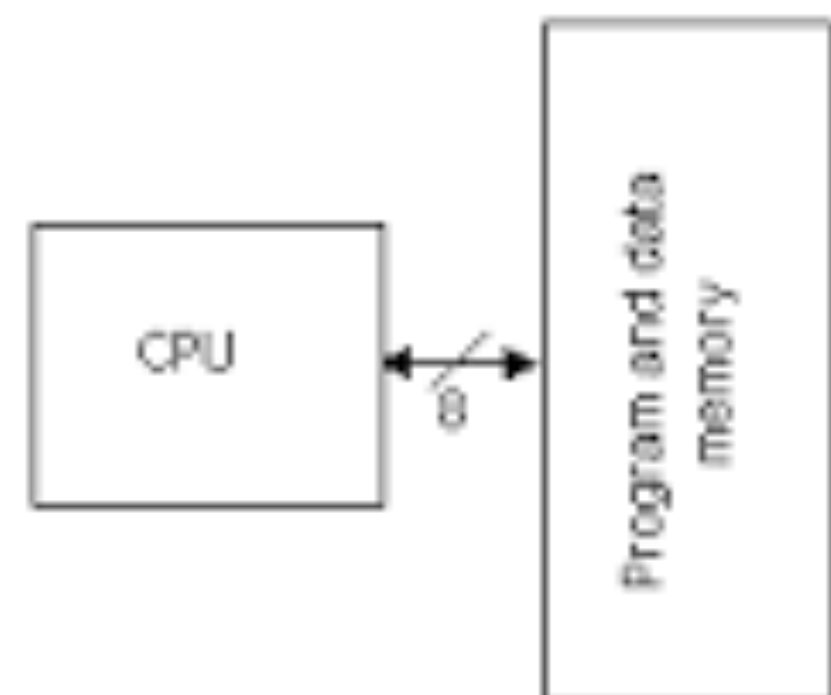
- Özellikleri:
 - Veri ve komutlar için ayrı bellek ve veri yollarına sahiptir.
 - İşlemci aynı anda hem veri hem de komut üzerinde çalışabilir.
 - Mikrodenetleyicilerde bu mimari sıkça kullanılır. Örneğin, **ARM Cortex-M serisi mikrodnetleyiciler**, gerçek zamanlı uygulamalarda bu mimariden yararlanır.
- Avantajları:
 - Daha yüksek işlem hızı sağlar.
 - Bellek erişim çatışmalarını önler.
- Dezavantajları:
 - Karmaşık bir tasarıma sahiptir.
 - Maliyet açısından daha yüksektir.



Harvard



von-Neumann



Harvard vs. von Neuman Block Architectures

CISC (Complex Instruction Set Computer) Mimarisi

- Özellikleri:
 - Çok sayıda ve karmaşık komut kümesine sahiptir.
 - Tek bir komut, birden fazla işlemi gerçekleştirebilir.
 - Intel x86 işlemci ailesi CISC mimarisine bir örnektir. Bu işlemciler, yazılım geliştiriciler için kolaylık sağlayan geniş bir komut seti sunar.
- Avantajları:
 - Kod yoğunluğu daha düşüktür.
 - Programlama kolaylığı sağlar.
- Dezavantajları:
 - İşlemcinin tasarımı daha karmaşıktır.
 - Yüksek güç tüketimi gerektirir.

RISC (Reduced Instruction Set Computer) Mimarisi

- Özellikleri:
 - Daha az ve basit komut kümesine sahiptir.
 - İşlemci başına bir komut işler.
 - ARM işlemciler, RISC mimarisinin başarılı bir örneğidir. Bu işlemciler genellikle mobil cihazlarda ve gömülü sistemlerde kullanılır.
- Avantajları:
 - Daha hızlı işlem yapılmasını sağlar.
 - Enerji verimliliği yüksektir.
- Dezavantajları:
 - Daha büyük bir bellek gerektirebilir.
 - Programlama karmaşıklığı olabilir. Yazılım geliştirme süreci daha karmaşık olabilir.

CISC Modeli



RISC Modeli



Paralel Mimariler

- Özellikleri:
 - Aynı anda birden fazla işlem yapılabilir.
 - SIMD (Tek Komut Çoklu Veri), MIMD (Çoklu Komut Çoklu Veri) gibi alt türlere ayrılır.
 - Süper bilgisayarlar, paralel mimarilerle çalışır. Örneğin, **NVIDIA'nın CUDA teknolojisi**, grafik işleme birimlerinde (GPU) paralel işleme yeteneklerini kullanır.
- Avantajları:
 - Yüksek performanslı uygulamalar için uygundur.
 - Büyük veri setleri üzerinde verimli çalışır.
- Dezavantajları:
 - Karmaşık bir tasarıma sahiptir.
 - Paralellikten tam anlamıyla yararlanmak için uygun yazılımlar gerekir.

Mobil Mimariler

- Özellikleri:
 - Düşük güç tüketimi ve enerji verimliliği odaklıdır.
 - Mobil cihazlara özel optimize edilmiştir.
 - Qualcomm Snapdragon işlemcileri, mobil mimarinin bir örneğidir. Bu işlemciler, yüksek performans ve düşük güç tüketimi sağlar.
- Avantajları:
 - Pil ömrünü artırır.
 - Küçük boyut ve düşük ısı üretimi sağlar.
- Dezavantajları:
 - Daha düşük işlem gücüne sahip olabilir.

Açık ve Kapalı Mimariler

- Açık Mimari:
 - Donanım ve yazılım geliştiricileri tarafından kolayca değiştirilebilir ve uyarlanabilir.
 - Örneğin: IBM PC.
- Kapalı Mimari:
 - Sahiplenilen ve yalnızca üretici tarafından değiştirilebilen mimari.
 - Örneğin: Apple Mac.