

# Bilgisayar Mimarisi ve Mantık Kapıları

Bursa Teknik Üniversitesi  
BLM0101 - Bilgisayar Mühendisliğine Giriş Proje Sunumu

25360859432  
Kürşat Şahin

# Bilgisayar Mimarisi Nedir?

- Temel olarak bir sistemin uçtan uca yapısını ifade eder.
- İçerisinde bulunan bileşenler, bilgisayarın amacına göre işlemleri sırasıyla bir düzen içerisinde gerçekleştirir.
- Bir bilgisayar mimarisi bilgisayarın bilgi işlemesini, saklamasını ve geri almasını sağlar.
- İki çeşit mimari temel alınır. Von Neumann Mimarisi ve Harvard Mimarisi.

# Bilgisayar Mimarisi Bileşenleri

- 4 Adet temel bileşen vardır.
- Giriş, Çıkış, Depolama, Merkezi İşlem Birimi (CPU)
- Bu birimler sistematik olarak alt bölümlere ayrılabilir.
- Bu bileşenler, adres, veri ve kontrol yolundan oluşan sistem veri yolu (Sistem veri yolları) ile birbirine bağlanır.

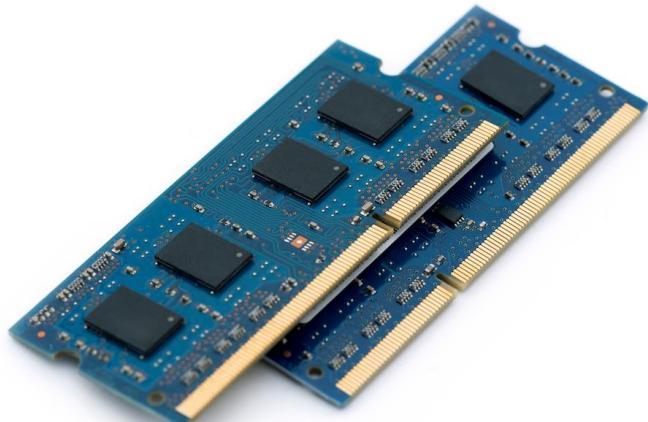
# Giriş ve Çıkış birimi

- Bilgisayarda işlenecek veriler Giriş biriminden alınır.
- Alınan veri sistem veri yolu ile önce depolanmaya sonrasında ise CPU'ya gönderilir.
- CPU da işleme alınan veri ise Çıkış birimine çıktı olarak iletilir.
- Giriş ve çıkış birimlerine örnek vermek gereklse yandaki tablo en çok bilinenlerdir.

| Giriş Birimleri (Input Devices) | Çıkış Birimleri (Output Devices) |
|---------------------------------|----------------------------------|
| Klavye                          | Monitör                          |
| Fare                            | Yazıcı                           |
| Tarayıcı                        | Hoparlör                         |
| Dokunmatik Ekran (giriş amaçlı) | Projektör                        |
| Mikrofon                        | Kulaklık                         |

# Depolama birimi/bellek

- Veriyi depolamak için kullanılır. Birincil ve İkincil depolama olarak ikiye ayrılır.



RAM Bellek



SSD Bellek

# Birincil Depolama Birimi

- RAM ve ROM olmak üzere iki farklı bellek vardır.
- RAM, gerekli bilgileri doğrudan işlemciye (CPU) iletir. Verileri ve talimatları depolayan geçici bellektir.
- ROM, önceden yüklenmiş talimatları içeren bir bellektir. Bu belleğin içeriği kalıcıdır ve sonradan değiştirilemez.
- Makinedeki önyükleme işlemleri için kullanılır. Bilgisayar başlarken mimarinin nasıl kurulacağını, işletim sisteminin başlatılabilmesi için sabit diskin nasıl bulunacağı gibi işlemlerden sorumludur.

# İkincil Depolama Birimi

- İkincil depolama veya harici depolama birimleri denebilir.
- Doğrudan CPU'ya erişimleri yoktur.
- Büyük miktarda veriyi kalıcı olarak saklayabilirler. (HDD, SSD, CD)
- Veriyi kullanmadan önce ana depolama birimine aktarmaları gereklidir.

# Merkezi İşlem Birimi (CPU)

- Yazmaçları, Aritmetik Mantık Birimini ve derleme dili komutlarını yürüten birimdir.
- Bilgisayarın tüm parçaları ile iletişim halindedir.
- 5 Adet alt bileşeni vardır.

# Aritmetik Mantık Birimi (ALU)

- İşlemciye verilen verilere hızlı olarak aritmetik ve mantıksal işlemler uygular.
- Aritmetik işlemler (toplama, çıkarma, çarpma, bölme)
- Mantıksal işlemler (<, >, VE (AND), VEYA (OR) vb.)
- Ortaya çıkan sonuçları saklamak için yazmaç (Register) kullanır.

# Yazmaçlar (Register)

- Yüksek hızlı ve amaca yönelik geçici belleklerdir.
- CPU tarafından erişilir ve değiştirilir.
- Hangi işlemin yapılacağı, hangi adrese yapılacağı ve işlem içeriğini barındırır.
- Bu bilgileri Aritmetik Mantık Biriminden almaktadır.

# Kontrol Birimi

- Veri ve komut işlemeyi düzenler.
- ALU ve kayıtlarla iletişim kurarak bilgisayara depolanmış program komutlarını yürütmesi için talimat verir.

# Veri Yolları (Dahili veri yolları)

- Bileşenler arasındaki iletişimini sağlarlar.
- Aynı zamanda bu veri yolları üzerinde CPU bellek hücresi adresi ve o adressteki veriyi almak istedığını elektronik sinyal ile iletir ve bellekten veri çekilebilir. Bu duruma okuma adı verilir.
- Bu işlemin tam tersi olarak da CPU bellekteki hücrenin adresini ve hangi veriyi oraya yerleştirmek istedığını elektronik sinyal ile iletебilir. Bu duruma da yazma adı verilir.

# Önbellek

- Fiziksel olarak CPU içerisinde yer alan yüksek hızlı bellek parçasıdır.
- İşlem sırasında ana belleğin o an kullandığı bölümün bir kopyası saklanır. Bu sayede yazmaçlar ana bellek ile CPU arasında git gel yapmak yerine daha hızlı bir şekilde verilere ulaşabilir.
- Buradaki veriler üzerinde yapılan değişiklikler işlemin bittiğine karar verildikten sonra ana belleğe aktarılır.

# Von Neumann Mimarisi

- Diğer mimarilerden ayıran en önemli özelliği verileri ve komutları aynı bellekte saklamasıdır.
- Ana bileşenleri şu şekildedir: Bellek Birimi, Merkezi İşlem Birimi (CPU) ve Giriş/Çıkış birimleri.
- Bu mimari temelinde bir döngü şeklinde çalışır. Bu döngü Fetch-Decode-Execute döngüsüdür.
- CPU, komutun adresini belirler ve komutu ister (Fetch), gelen komutun hangi işlemi yapacağı kontrol birimi tarafından çözülür (Decode), son olarak da Aritmetik Mantık Birimi tarafından komut işlenir (Execute)
- En zayıf noktası ise veriler ve komutlar aynı veriyolunda taşındığı için darboğaz olma riski fazladır.

# Harvard Mimarisi

- Von Neumann mimarisindeki en büyük sorunu çözmek için geliştirilmiştir.
- Bu mimarinin Von Neumann mimarisinden en büyük farkı verilerin ve komutların farklı veri yolları üzerinden taşınmasıdır.
- Bu işlem ile darboğaz engellenerek işlemlerin daha hızlı yapılması amaçlanmıştır.
- Bu mimarinin dezavantajı ise maliyet ve boşça çıkan alandır. İki ayrı bellek kontrolcüsü, daha fazla pin sayısı vb. işlemlerden dolayı maliyet artmaktadır. Boşça çıkan alan ise şu şekildedir: Veri belleği doluyken komut belleği boş olsa bile, boş alan veri için kullanılamaz. Bu da bellek kullanım verimliliğini düşürür.

# Neumann Mimarisi ile Harvard Mimarisi Arasındaki Farklar

| Özellik        | Von Neumann | Harvard                  |
|----------------|-------------|--------------------------|
| Bellek         | Tek         | Aynı komut ve veri       |
| Veri Yolu      | Tek         | İki (komut ve veri ayrı) |
| Paralel Erişim | Yok         | Var                      |
| Performans     | Orta        | Yüksek                   |
| Kullanım       | PC CPU'ları | Mikrodenetleyici, DSP    |
| Esneklik       | Yüksek      | Düşük                    |

# Temel Mantık Kapıları

- Transistör ve diyonların bir araya getirilmesiyle oluşturulan cihazlardır.
- Mantıksal olarak çalışırlar ve 0 ile 1 ikili sonucunu verirler.
- Elektronik sistemlerin en temelidirler.
- Dijital elektronik sistemlerin ve bilgisayar biliminin alfabesi gibidir.

# Mantık Kapıları ne işe yarar?

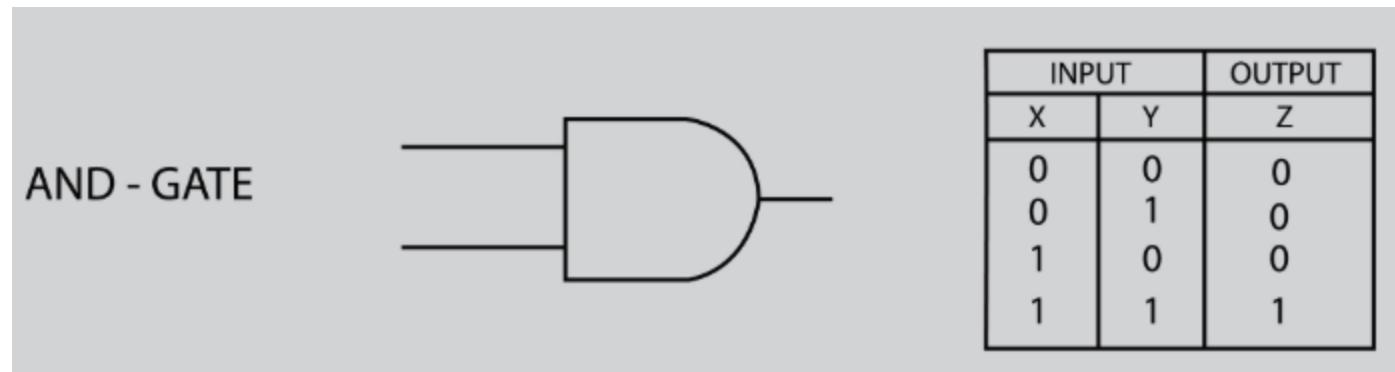
- Adından da anlaşılacağı üzere bilgisayar üzerindeki mantıksal işlemlerin tamamını gerçekleştirirler.
- Karar verme mekanizması oluşturma, karşılaştırma yapma, koşulları işlemleri gerçekleştirmeye gibi işlevleri vardır.
- Bu işlevler sayesinde bilgisayarlar hesaplama yapabilir, kararlar alabilir ve elektronik sistemler otomatik çalışabilir.

# Temel Mantık Kapıları

- 3 Ana temel mantık kapıları vardır.
- Bunlar AND, OR ve NOT kapılarıdır.
- Karmaşık hesaplamaları hesaplamak için birbirine bağlı şekilde kullanılabilirler.
- Daha sonrasında bu kapılardan türetilmiş kapılar da mevcuttur.

# AND Gate (Ve Kapısı)

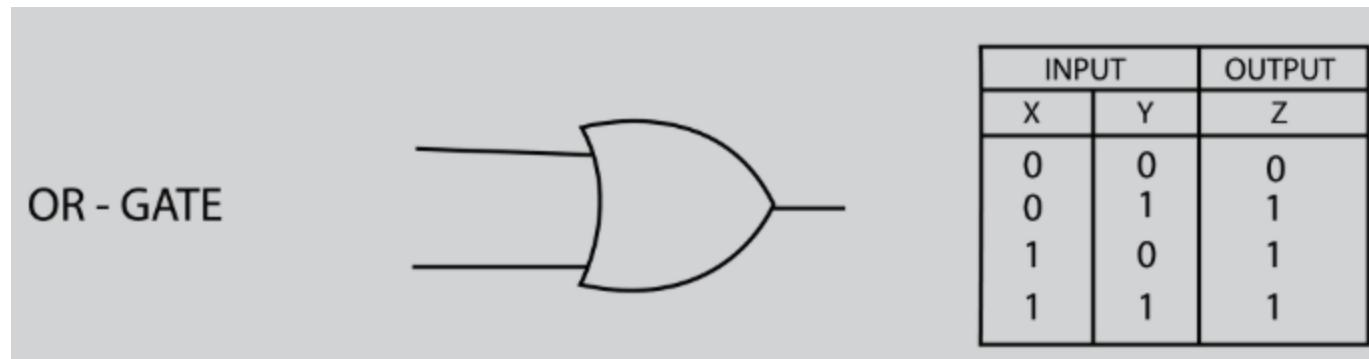
- En titiz kapı olarak düşünülebilir. Çıkışın 1 olabilmesi için tüm girişlerin 1 olması gereklidir.
- Matematiksel olarak Çıktı =  $\text{Girdi1} * \text{Girdi2}$  olarak düşünülebilir.
- Herhangi bir girişin 0 olduğu durumda 0 çıktısı verir.



AND Kapısı Doğruluk Tablosu

# OR Gate (Veya Kapısı)

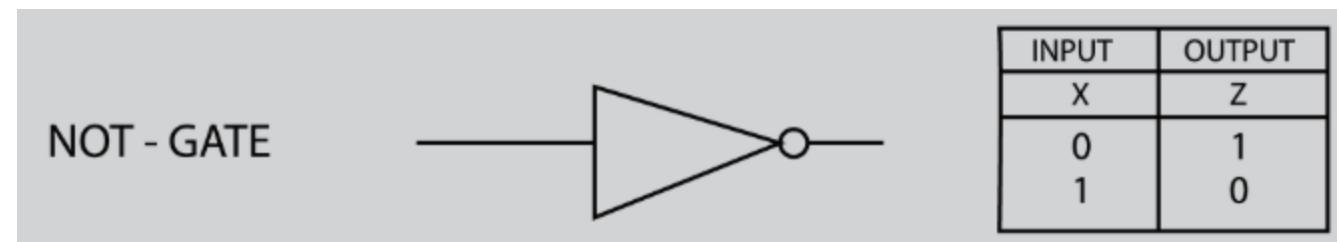
- Esnek kapı olarak düşünülebilir. Çıkışın 1 olabilmesi için And kapısının aksine iki girişten birinin veya ikisinin de 1 olması yeterlidir.
- Matematiksel olarak Çıktı = Girdi1 + Girdi2 olarak düşünülebilir. Burada 1 + 1 sonucunun 1 çıkacağını dikkate almalıyız.



OR Kapısı Doğruluk Tablosu

# NOT Gate (Değil Kapısı)

- Kendisine gelen girişlerin hepsini tersine çevirir.
- Diğer kapıların aksine tek girişi vardır.



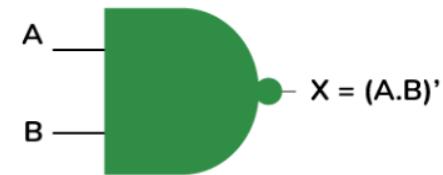
**NOT Kapısı Doğruluk Tablosu**

# Türetilmiş Mantık Kapıları

- Temel mantık kapılarının kombinasyonları ile daha karmaşık işlemler gerçekleştirilebilir.
- NAND, NOR, XOR ve XNOR olmak üzere 4 adet türetilmiş kapı vardır.

# NAND Gate (VE DEĞİL)

- AND ve NOT kapısının birleşimi ile oluşturulur.
- Sonucun 0 olması için iki girişin de 1 olması gereklidir. Aksi takdirde her sonuç 1 olacaktır.
- Dijital sistemler sadece NAND kapısı kullanılarak inşa edilebilir. Bu yüzden evrensel kapı denebilir.



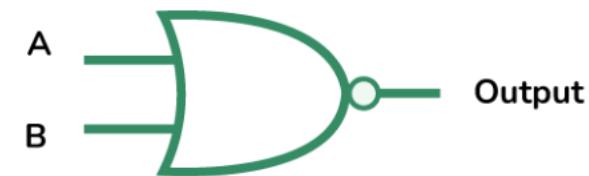
Truth Table

| Input A | Input B | X = (A.B)' |
|---------|---------|------------|
| 0       | 0       | 1          |
| 0       | 1       | 1          |
| 1       | 0       | 1          |
| 1       | 1       | 0          |

NAND Kapısı Doğruluk Tablosu

# NOR Gate (VEYA DEĞİL)

- OR ve NOT kapısının birleşimi ile oluşturulur.
- Kapıdan çıkacak çıktıının hesaplanması için öncelikle girişler OR işlemine tabi tutulur. Daha sonrasında NOT işlemi ile son çıktı belirlenir.



Truth Table

| Input A | Input B | $0 = (A + B)'$ |
|---------|---------|----------------|
| 0       | 0       | 1              |
| 0       | 1       | 0              |
| 1       | 0       | 0              |
| 1       | 1       | 0              |

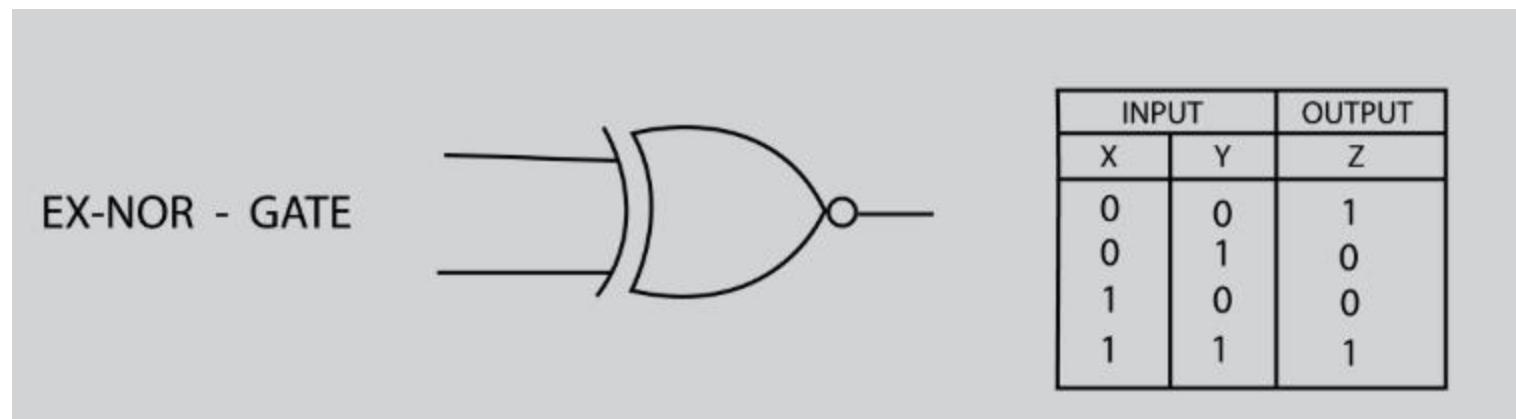
NOR Kapısı Doğruluk Tablosu

# XOR Gate (Özel VEYA)

- Girişler birbirinden farklıysa 1 sonucu verilir.
- Veya kapısından farklı iki giriş 1 olsa bile sonuç yine de 0 olur.
- Bilgisayarların toplama işlemi yapabilmesi için kritik bir öneme sahiptir.

# XNOR Gate (Özel VEYA DEĞİL)

- XOR ve NOT kapısının birleşimidir.
- XOR işleminden gelen sonuçları NOT ile tekrardan işleme sokarak son çıktıyi verir.



XNOR Kapısı Doğruluk Tablosu

# Mantık Kapıları Neden Bu Kadar Önemlidir?

- Fiziksel dünya ile Dijital dünya arasında köprü görevi görür.
- Fiziksel dünyadan gelen bilgiler analog olarak gelir ve bunların anlaşılabilmesi için dijitalleştirilmesi gereklidir.  
Mantık kapıları bu işlemi yapabilirler.
- Aritmetik işlemlerin yapılabilmesi için mantık kapılarına ihtiyaç vardır.
- Bilgisayarın hafızasındaki işlemleri, verilerin saklanması gibi işlemleri mantık kapıları sayesinde yapabiliriz.
- Karar verme mekanizmaları mantık kapıları üzerinden dönmektedir.
- Özetlemek gerekirse bilgisayarı bir sistemin beyni olarak düşünürsek, mantık kapıları da o beynin nöronlarıdır.  
Yaptıkları iş basit bir sinyal değişikliği olarak gözükse de tüm sistemin düzgün ve işlevsel çalışmasını sağlarlar.

# Kaynakça

- -<https://www.spiceworks.com/tech/tech-general/articles/what-is-computer-architecture/>
- Yapay zeka araçları (Gemini, ChatGPT)
- Bilgisayar Bilimine Giriş (J. Glenn Brookshear) Chapter 2 (2.1 Computer Architecture, 2.2 Machine Language)
- <https://www.geeksforgeeks.org/digital-logic/logic-gates/>