

IOT / OT

Senior Master Computer Engineer
Hamit Mızrak

IOT/OT



KONULAR

- ▶ IoT (Internet of Things) ve OT (Operational Technology) teknolojileri, birbirleriyle ilişkili ancak farklı alanları kapsamaktadır. İşte bu iki teknolojiyle ilgili ana konu başlıklarını:
- ▶ ### IoT (Internet of Things)
- ▶ 1. **Temel Kavramlar ve Tanımlar**
 - ▶ - IoT nedir?
 - ▶ - IoT bileşenleri: Sensörler, Aktüatörler, Ağ Bağlantıları, IoT Platformları
 - ▶ - IoT ekosistemi ve mimarisi

KONULAR

- ▶ 2. **IoT Uygulama Alanları**
 - ▶ - Akıllı Evler
 - ▶ - Akıllı Şehirler
 - ▶ - Sağlık Hizmetleri
 - ▶ - Endüstriyel IoT (IIoT)
 - ▶ - Tarımda IoT (Akıllı Tarım)
 - ▶ - Taşımacılık ve Lojistik

KONULAR

- ▶ 3. **IoT Protokolleri ve İletişim Teknolojileri**
 - ▶ - MQTT, CoAP, AMQP
 - ▶ - Bluetooth, Zigbee, Z-Wave, LoRaWAN, NB-IoT

KONULAR

- ▶ 4. **Veri Toplama ve Analiz**
 - ▶ - IoT veri yönetimi
 - ▶ - Büyük veri ve IoT
 - ▶ - Veri analitiği ve makine öğrenimi

KONULAR

- ▶ 5. **Güvenlik ve Gizlilik**
 - ▶ - IoT güvenlik tehditleri ve zorlukları
 - ▶ - Güvenlik çözümleri ve en iyi uygulamalar
 - ▶ - Veri gizliliği ve kullanıcı verilerinin korunması

KONULAR

- ▶ 6. **IoT Platformları ve Yazılım**
 - ▶ - IoT platformları: AWS IoT, Google Cloud IoT, Azure IoT
 - ▶ - IoT cihaz yönetimi
 - ▶ - Edge Computing ve IoT

KONULAR

- ▶ ### OT (Operational Technology)
- ▶ 1. **Temel Kavramlar ve Tanımlar**
- ▶ - OT nedir?
- ▶ - OT bileşenleri: PLC, SCADA, DCS
- ▶ - OT sistemlerinin genel mimarisi

KONULAR

- ▶ 2. **OT Uygulama Alanları**
 - ▶ - Üretim ve Endüstriyel Otomasyon
 - ▶ - Enerji Yönetimi ve Güç Sistemleri
 - ▶ - Su ve Atık Su Yönetimi
 - ▶ - Madencilik ve Metalurji
 - ▶ - Ulaşım ve Altyapı

KONULAR

- ▶ 3. **OT Protokoller ve İletişim Teknolojileri**
 - ▶ - Modbus, Profibus, OPC UA
 - ▶ - Endüstriyel Ethernet ve saha veriyolu (Fieldbus) sistemleri

KONULAR

- ▶ 4. **Güvenlik ve Dayanıklılık**
 - ▶ - OT güvenlik tehditleri ve zorlukları
 - ▶ - Siber güvenlik ve fiziksel güvenlik önlemleri
 - ▶ - Endüstriyel kontrol sistemleri (ICS) güvenliği

KONULAR

- ▶ 5. **Entegrasyon ve Interoperabilite**
 - ▶ - IT/OT entegrasyonu
 - ▶ - Endüstri 4.0 ve akıllı fabrikalar
 - ▶ - OT sistemlerinin modernizasyonu ve dijital dönüşüm

KONULAR

- ▶ 6. **Veri Toplama ve Analiz**
 - ▶ - Sensör verileri ve gerçek zamanlı izleme
 - ▶ - Büyük veri ve analitik uygulamaları
 - ▶ - Veri entegrasyonu ve yönetimi

KONULAR

- ▶ 7. **Standartlar ve Düzenlemeler**
 - ▶ - OT standartları ve protokoller
 - ▶ - Düzenleyici gereklilikler ve uyumluluk
- ▶ IoT ve OT arasındaki temel farklar ve bu alanların nasıl kesiştiği, özellikle Endüstri 4.0 gibi modern teknolojik yaklaşımlar içinde büyük önem taşımaktadır. IoT, genellikle daha geniş bir veri toplama ve analiz ağının hizmet ederken, OT, belirli operasyonel süreçlerin yönetimi ve kontrolüne odaklanır.

IoT (Internet of Things)



Bölüm-1 (IoT (Internet of Things))

- ▶ ### IoT (Internet of Things)
- ▶ 1. **Temel Kavramlar ve Tanımlar**
 - ▶ - IoT nedir?
 - ▶ - IoT bileşenleri: Sensörler, Aktüatörler, Ağ Bağlantıları, IoT Platformları
 - ▶ - IoT ekosistemi ve mimarisi

IoT (Internet of Things)

- ▶ ### IoT (Internet of Things)
- ▶ ##### 1. Temel Kavramlar ve Tanımlar
- ▶ **IoT Nedir?**
- ▶ IoT, yani Nesnelerin Interneti, fiziksel nesnelerin internete bağlanarak veri toplaması, paylaşması ve belirli işlemleri gerçekleştirmesi anlamına gelir. Bu nesneler, sensörler, cihazlar, araçlar, yapılar ve diğer günlük nesneleri içerebilir. IoT, bu nesnelerin ağ üzerinden birbirleriyle iletişim kurmasını ve işbirliği yapmasını sağlar. Bu sayede, çeşitli uygulama alanlarında daha akıllı ve verimli sistemler oluşturulur. Örneğin, akıllı evler, akıllı şehirler, sağlık hizmetleri, endüstriyel otomasyon gibi birçok alanda kullanılabilir.

IoT (Internet of Things)

- ▶ **IoT Bileşenleri**
- ▶ 1. **Sensörler:**
 - Sensörler, fiziksel dünyadaki çeşitli parametreleri ölçen ve bu bilgileri dijital formata dönüştüren cihazlardır. Örneğin, bir sıcaklık sensörü ortamın sıcaklığını ölçer, bir nem sensörü havadaki nem oranını belirler.
 - Sensörler, IoT sistemlerinde kritik bir rol oynar çünkü çevresel verileri toplar ve analiz edilmesi için merkezi sistemlere iletir.

IoT (Internet of Things)

- ▶ 2. **Aktüatörler:**
 - ▶ - Aktüatörler, sensörlerden gelen verilere dayanarak fiziksel işlemler gerçekleştiren cihazlardır. Örneğin, bir motoru çalıştırmak, bir vanayı açıp kapatmak veya bir ısıtıcıyı açıp kapatmak gibi işlevler üstlenirler.
 - ▶ - Aktüatörler, IoT sistemlerinin çevreyle etkileşime geçmesini ve belirli işlemleri otomatik olarak gerçekleştirmesini sağlar.

IoT (Internet of Things)

- ▶ 3. **Ağ Bağlantıları:**
 - ▶ - IoT cihazları arasındaki iletişimini ve veri transferini sağlamak için kullanılan ağ bağlantılarıdır. Bu bağlantılar, Wi-Fi, Bluetooth, Zigbee, LoRa, NB-IoT gibi çeşitli kablosuz iletişim protokollerini içerebilir.
 - ▶ - Ağ bağlantıları, IoT cihazlarının verilerini merkezi bir sisteme veya birbirlerine iletmesini sağlar. Bu sayede, uzaktan erişim ve kontrol mümkün hale gelir.

IoT (Internet of Things)

- ▶ 4. **IoT Platformları:**
 - ▶ - IoT platformları, cihazların bağlantısını, veri yönetimini ve analitiği kolaylaştıran yazılım ve hizmetler bütünüdür. Bu platformlar, cihaz yönetimi, veri toplama, işleme ve analiz için gerekli araçları sağlar.
 - ▶ - Örnek olarak AWS IoT, Google Cloud IoT, Microsoft Azure IoT gibi bulut tabanlı platformlar gösterilebilir. Bu platformlar, geniş kapsamlı hizmetler sunarak IoT projelerinin hızlı ve verimli bir şekilde geliştirilmesini sağlar.

IoT Ekosistemi ve Mimarisi

- ▶ **IoT Ekosistemi ve Mimarisi**
- ▶ IoT ekosistemi, çeşitli bileşenlerin ve katmanların birlikte çalıştığı kompleks bir yapıdadır. Genel olarak, IoT mimarisi şu katmanları içerir:
 - ▶ 1. **Cihaz Katmanı:**
 - Sensörler ve aktuatörler bu katmanda bulunur. Bu cihazlar, çevresel verileri toplar ve belirli işlemleri gerçekleştirir.

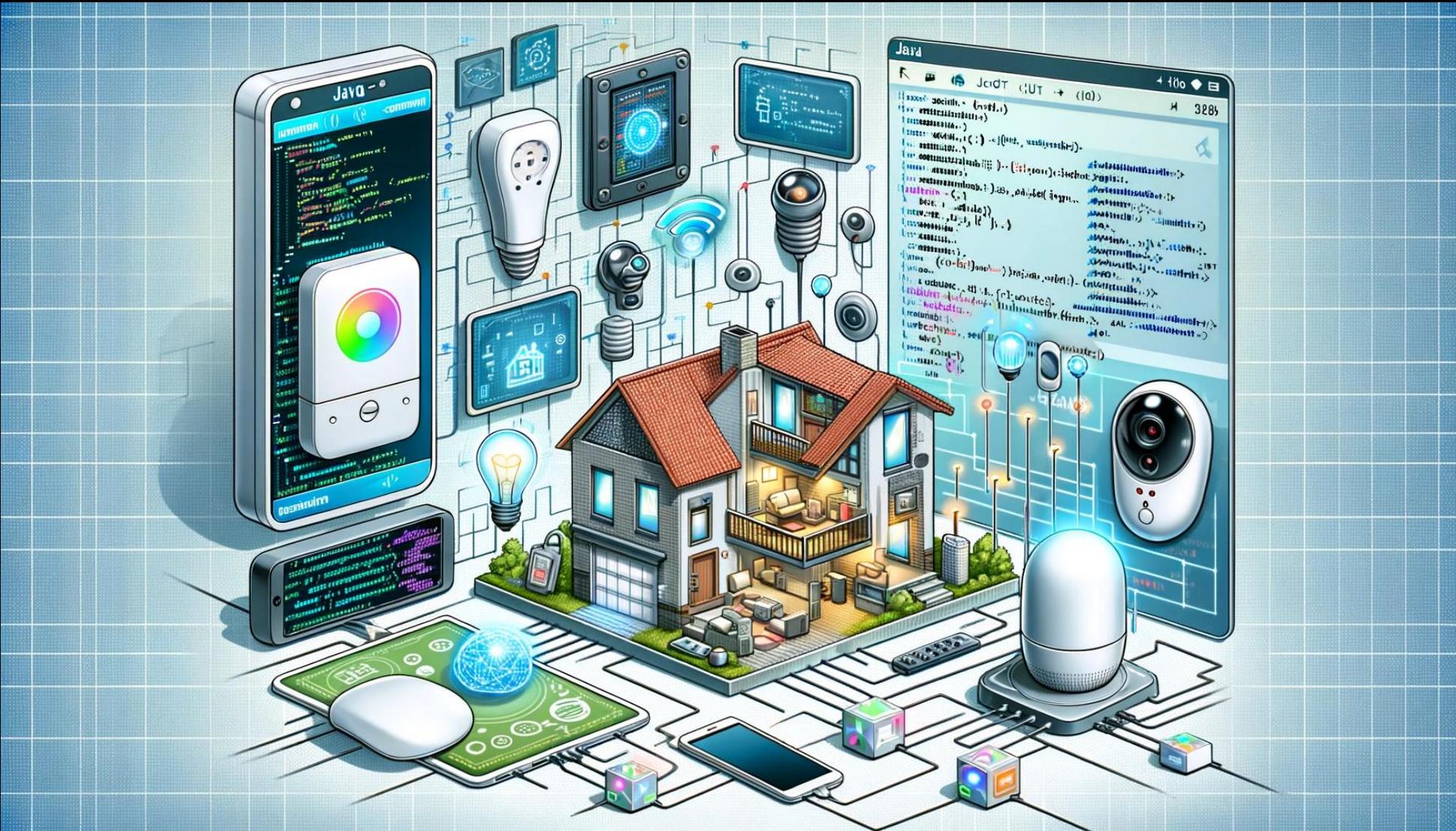
IoT (Internet of Things)

- ▶ 2. **Bağlantı Katmanı:**
 - Cihazları ağ üzerinden bağlayan iletişim protokollerini ve ağ altyapılarını bu katmanda yer alır. Veri iletimi ve cihazlar arası iletişim bu katman üzerinden sağlanır.
- ▶ 3. **Veri Yönetim Katmanı:**
 - Toplanan verilerin depolandığı, işlendiği ve yönetildiği katmandır. Bu katmanda, veri tabanları, veri işleme motorları ve analiz araçları bulunur.

IoT (Internet of Things)

- ▶ 4. **Uygulama Katmanı:**
 - Kullanıcıların IoT verilerine eriştiği, yönetim ve kontrol uygulamalarının çalıştığı katmandır. Bu katman, kullanıcı arayüzleri, kontrol panelleri ve diğer uygulama yazılımlarını içerir.
- ▶ 5. **Güvenlik Katmanı:**
 - IoT sistemlerinin güvenliğini sağlayan protokoller, şifreleme yöntemleri ve erişim kontrol mekanizmaları bu katmanda yer alır. Veri güvenliği, cihaz güvenliği ve ağ güvenliği bu katman tarafından sağlanır.
- ▶ IoT ekosistemi, bu katmanların entegrasyonu ve uyumlu çalışmasıyla etkin bir şekilde işlev görür. Bu entegrasyon, farklı cihazların ve sistemlerin birbirleriyle iletişim kurmasını ve işbirliği yapmasını sağlar, böylece akıllı ve bağlı bir dünya oluşturur.

2. **IoT Uygulama Alanları**



IoT (Internet of Things)

- ▶ 2. **IoT Uygulama Alanları**
 - ▶ - Akıllı Evler
 - ▶ - Akıllı Şehirler
 - ▶ - Sağlık Hizmetleri
 - ▶ - Endüstriyel IoT (IIoT)
 - ▶ - Tarımda IoT (Akıllı Tarım)
 - ▶ - Taşımacılık ve Lojistik

Smart House



Smart House

- ▶ Akıllı ev sistemlerini gösteren ayrıntılı bir çizim.
- ▶ Bu diyagram, akıllı ışıklar, akıllı termostat, akıllı güvenlik kameraları, akıllı kilitler, akıllı cihazlar (buzdolabı, fırın) ve merkezi bir akıllı ev merkezi gibi çeşitli akıllı cihazları ve sistemleri içermektedir.
- ▶ Her cihaz merkezi hub'a bağlı olup, hub buluta bağlanmaktadır.
- ▶ Akıllı ışıklar hareket sensörleri içerir ve mobil uygulama ile kontrol edilir.
- ▶ Akıllı termostat, sıcaklık kontrolünü mobil uygulama üzerinden gösterir.
- ▶ Güvenlik sistemi, uzaktan izleme ve kontrol gösteren kameralar ve akıllı kilitler içerir. Cihazlar otomasyon özellikleri ile tasvir edilmiştir.

2. **IoT Uygulama Alanları**

- ▶ IoT (Nesnelerin İnterneti) uygulama alanları hakkında bilgi verebilirim:
- ▶ **### 1. Akıllı Evler**
- ▶ Akıllı evler, IoT teknolojisinin en yaygın uygulama alanlarından biridir. Bu sistemler, çeşitli cihazların ve ev içi sistemlerin internet üzerinden kontrol edilmesine olanak tanır. Örneğin, akıllı termostatlar, aydınlatma sistemleri, güvenlik kameraları ve akıllı kilitler gibi cihazlar, kullanıcıların evlerini daha verimli ve güvenli bir şekilde yönetmelerini sağlar.

2. **IoT Uygulama Alanları**

- ▶ **### 2. Akıllı Şehirler**
- ▶ Akıllı şehirler, şehir altyapısını ve hizmetlerini optimize etmek için IoT teknolojilerini kullanır. Bu uygulamalar arasında akıllı trafik yönetimi, akıllı sokak lambaları, çevresel izleme sistemleri ve atık yönetimi gibi sistemler bulunur. Amaç, şehir yaşamını daha sürdürülebilir, verimli ve yaşanabilir hale getirmektir.

2. **IoT Uygulama Alanları**

- ▶ ### 3. Sağlık Hizmetleri
- ▶ Sağlık sektöründe IoT, hasta takibi, ilaç yönetimi ve uzaktan sağlık hizmetleri gibi birçok alanda kullanılır. Giyilebilir cihazlar, hastaların sağlık durumlarını sürekli izleyerek, doktorlara gerçek zamanlı veri sağlar. Bu, kronik hastalıkların yönetimi ve acil durum müdahaleleri için kritik öneme sahiptir.

2. **IoT Uygulama Alanları**

- ▶ ### 4. Endüstriyel IoT (IIoT)
- ▶ Endüstriyel IoT, üretim ve endüstriyel süreçlerin otomasyonu ve optimizasyonu için kullanılır. Sensörler ve bağlı cihazlar, makinelerin performansını izler, bakım ihtiyaçlarını önceden tespit eder ve üretim verimliliğini artırır. Bu, maliyetleri düşürür ve operasyonel verimliliği artırır.

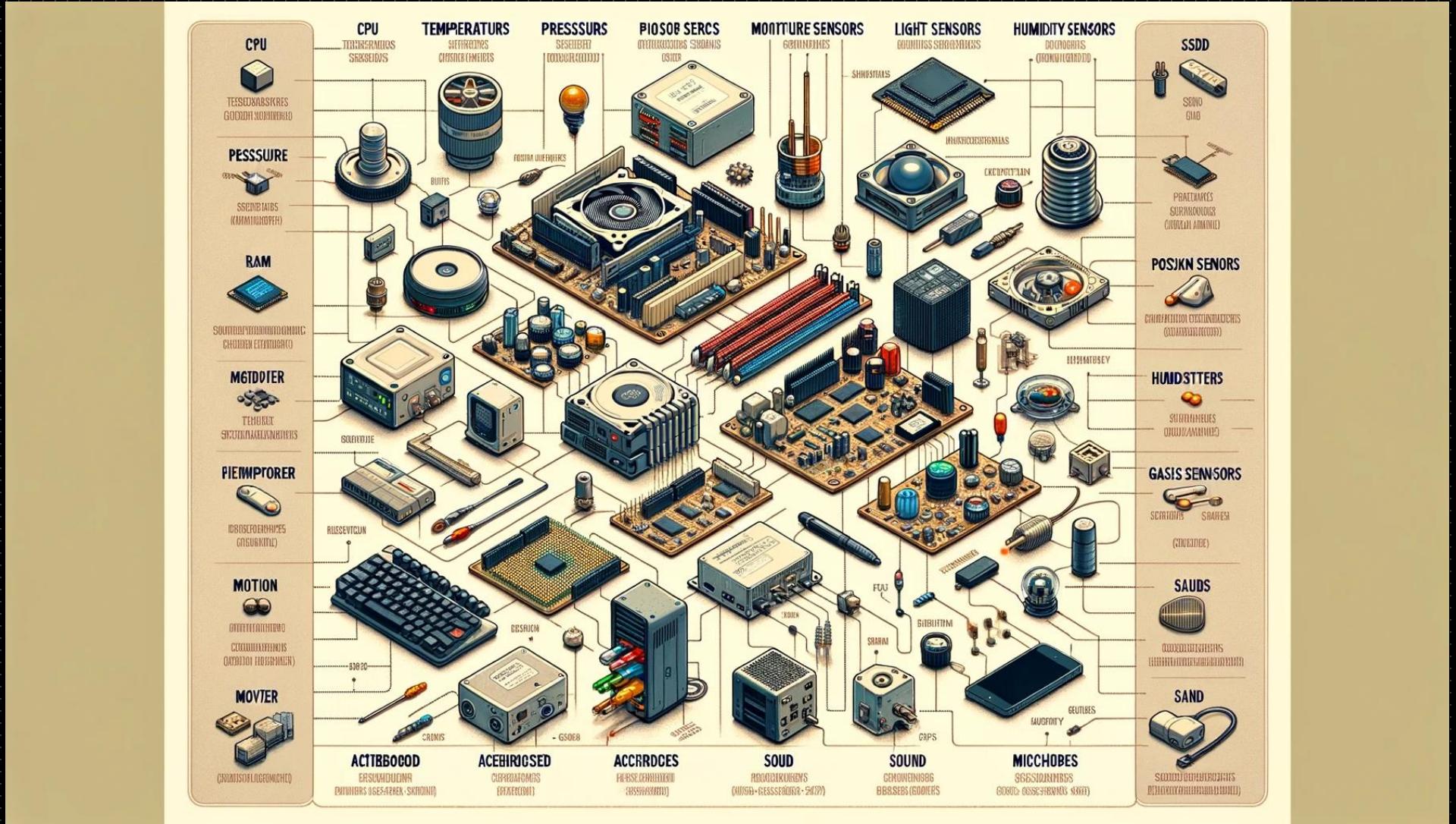
2. **IoT Uygulama Alanları**

- ▶ ### 5. Tarımda IoT (Akıllı Tarım)
- ▶ Tarım sektöründe IoT, tarım operasyonlarının daha verimli ve sürdürülebilir olmasına yardımcı olur. Sensörler ve bağlı cihazlar, toprağın nem seviyesini, hava koşullarını ve bitki sağlığını izleyerek, çiftçilere doğru veriler sağlar. Bu, sulama, gübreleme ve hasat süreçlerini optimize eder.

2. **IoT Uygulama Alanları**

- ▶ **### 6. Taşımacılık ve Lojistik**
- ▶ Taşımacılık ve lojistik sektöründe IoT, nakliye araçlarının izlenmesi, rota optimizasyonu ve kargo takibi gibi uygulamalar için kullanılır. Bu teknolojiler, operasyonel verimliliği artırır, teslimat sürelerini kısaltır ve maliyetleri düşürür. Ayrıca, araçların bakım ihtiyaçlarını önceden tespit ederek, arıza sürelerini azaltır.
- ▶ Bu alanların her biri, IoT teknolojisinin sunduğu veri toplama, analiz ve otomasyon imkanlarından faydalananarak, daha verimli, güvenli ve sürdürülebilir çözümler sunar.

Donanım Sensörler



Donanım ve Sensörler

- ▶ Donanım (hardware) ve sensörler, modern teknoloji ve bilgisayar sistemlerinin temel bileşenleridir. Donanım, fiziksel bileşenleri ifade ederken, sensörler çevresel verileri algılamak ve ölçmek için kullanılan cihazlardır. İşte bu iki kavramın detaylı açıklaması:

Donanım ve Sensörler

- ▶ ### Donanım (Hardware)
- ▶ Donanım, bilgisayar sistemlerini oluşturan fiziksel bileşenlerdir. Donanım bileşenleri, bilgisayarın çalışması ve belirli görevleri yerine getirmesi için gerekli olan tüm fiziksel parçaları içerir. Başlıca donanım bileşenleri şunlardır:
- ▶ 1. **Merkezi İşlem Birimi (CPU):**
 - ▶ - Bilgisayarın beynidir.
 - ▶ - Tüm işlemleri gerçekleştirir ve diğer bileşenlerle iletişim kurar.
 - ▶ - İşlemci çekirdekleri ve saat hızları performansı belirler.

Donanım ve Sensörler

- ▶ 2. **Bellek (RAM):**
 - ▶ - Geçici veri depolama alanıdır.
 - ▶ - İşlemci tarafından hızlı erişim gerektiren veriler burada tutulur.
 - ▶ - Veri saklama kapasitesi ve hız, sistem performansını etkiler.

Donanım ve Sensörler

- ▶ 3. **Sabit Disk (HDD) veya Katı Hal Sürücüsü (SSD):**
 - ▶ - Kalıcı veri depolama birimleridir.
 - ▶ - SSD'ler HDD'lere göre daha hızlıdır ve mekanik parçalar içermez.

Donanım ve Sensörler

- ▶ 4. **Anakart (Motherboard):**
 - Tüm bileşenleri birbirine bağlayan ana devre kartıdır.
 - CPU, RAM, depolama birimleri ve diğer bileşenler anakart üzerinden iletişim kurar.

Donanım ve Sensörler

- ▶ 5. **Grafik İşlem Birimi (GPU):**
 - Görüntü işlemleri ve grafiklerin oluşturulmasında kullanılır.
 - Özellikle oyun, video düzenleme ve yapay zeka uygulamalarında önemlidir.

Donanım ve Sensörler

- ▶ 6. **Güç Kaynağı Birimi (PSU):**
 - Bilgisayara gerekli elektrik gücünü sağlar.
 - Çeşitli bileşenlere uygun voltaj ve akım dağıtımını yapar.

Donanım ve Sensörler

- ▶ 7. **Giriş/Çıkış Aygıtları:**
 - Klavye, fare, monitör, yazıcı gibi cihazlardır.
 - Kullanıcıların bilgisayar ile etkileşimde bulunmasını sağlar.

Donanım ve Sensörler

- ▶ ### Sensörler
- ▶ Sensörler, fiziksel veya çevresel koşulları algılayan ve bu bilgileri elektriksel sinyallere dönüştüren cihazlardır. Sensörler, günlük yaşamdan endüstriyel uygulamalara kadar birçok alanda kullanılır. Başlıca sensör türleri şunlardır:

Donanım ve Sensörler

- ▶ 1. **Sıcaklık Sensörleri:**
 - Ortam veya nesnelerin sıcaklığını ölçer.
 - Termistörler, termokupplar ve entegre devre (IC) sıcaklık sensörleri bu kategoriye girer.

Donanım ve Sensörler

- ▶ 2. **Basınç Sensörleri:**
 - ▶ - Gaz veya sıvıların basıncını ölçer.
 - ▶ - Barometreler, manometreler ve piezoelektrik sensörler yaygın örnekleridir.

Donanım ve Sensörler

- ▶ 3. **Işık Sensörleri:**
 - ▶ - Ortamın ışık seviyesini algılar.
 - ▶ - Fotodiyotlar, fotoresistörler ve ışık algılayıcı IC'ler bu tür sensörlerdir.

Donanım ve Sensörler

- ▶ 4. **Hareket ve Konum Sensörleri:**
 - ▶ - Nesnelerin hareketini veya pozisyonunu tespit eder.
 - ▶ - İvmeölçerler, jiroskoplar ve manyetometreler bu tür sensörlere örnektir.

Donanım ve Sensörler

- ▶ 5. **Nem Sensörleri:**
 - ▶ - Havanın veya maddelerin nem oranını ölçer.
 - ▶ - Kapasitif ve direnç tabanlı nem sensörleri yaygındır.

Donanım ve Sensörler

- ▶ 6. **Ses Sensörleri:**
 - ▶ - Ses dalgalarını algılar.
 - ▶ - Mikrofonlar, akustik sensörler ve ultrasonik sensörler bu kategoriye girer.

Donanım ve Sensörler

- ▶ 7. **Gaz Sensörleri:**
 - Çeşitli gazların varlığını ve konsantrasyonunu algılar.
 - Karbon monoksit sensörleri, metan sensörleri ve oksijen sensörleri örnek olarak verilebilir.

Donanım ve Sensörler

- ▶ Donanım ve sensörler, birlikte çalışarak modern bilgisayar sistemleri ve akıllı cihazların işlevsellliğini sağlar. Donanım bileşenleri, sistemin işlem kapasitesini ve hızını belirlerken, sensörler çevresel verileri toplar ve bu verileri işlemek üzere donanıma iletir. Bu sayede, çeşitli uygulamalar ve sistemler, gerçek dünya verilerini kullanarak daha akıllı ve duyarlı hale gelir.

Donanım ve Sensörler

Donanım ve Sensörler

3. **IoT Protokoller ve İletişim Teknolojileri**

- ▶ 3. **IoT Protokoller ve İletişim Teknolojileri**
 - ▶ - MQTT, CoAP, AMQP
 - ▶ - Bluetooth, Zigbee, Z-Wave, LoRaWAN, NB-IoT

3. **IoT Protokoller ve İletişim Teknolojileri**

- ▶ **### IoT Protokoller ve İletişim Teknolojileri**
- ▶ IoT (Nesnelerin İnterneti) sistemleri, cihazların birbirleriyle ve merkezi bir sistemle etkili ve güvenilir bir şekilde iletişim kurmalarını sağlamak için çeşitli protokoller ve iletişim teknolojileri kullanır. İşte bazı yaygın IoT protokoller ve iletişim teknolojileri:
- ▶ **### IoT Protokoller**
- ▶ 1. **MQTT (Message Queuing Telemetry Transport)**
 - ▶ - **Nedir? :** MQTT, hafif bir mesajlaşma protokolüdür. Az yer kaplayan ve bant genişliği açısından verimli bir protokol olduğundan, özellikle düşük güçlü cihazlar ve düşük bant genişliği olan ağlar için uygundur.
 - ▶ - **Kullanım Alanları :** Sensör veri iletimi, uzaktan izleme sistemleri, mobil cihazlar.

3. **IoT Protokoller ve İletişim Teknolojileri**

- ▶ 2. **CoAP (Constrained Application Protocol)**
 - ▶ - **Nedir?**: CoAP, kısıtlı cihazlar için tasarlanmış bir web transfer protokolüdür. HTTP'nin basitleştirilmiş bir versiyonu olarak düşünülebilir ve küçük veri paketleri gönderip almayı sağlar.
 - ▶ - **Kullanım Alanları**: Akıllı evler, endüstriyel kontrol sistemleri, sağlık izleme sistemleri.
- ▶ 3. **AMQP (Advanced Message Queuing Protocol)**
 - ▶ - **Nedir?**: AMQP, mesaj odaklı middleware için bir açık standart protokoldür. Güvenli ve güvenilir mesaj iletimi sağlar.
 - ▶ - **Kullanım Alanları**: Finansal sektör, iş süreçleri otomasyonu, veri merkezleri.

3. **IoT Protokollerleri ve İletişim Teknolojileri**

- ▶ **### İletişim Teknolojileri**
- ▶ 1. **Bluetooth**
 - **Nedir?**:** Kısa mesafeli kablosuz iletişim teknolojisidir. Düşük enerji tüketimi ve kolay cihaz eşleştirme özellikleriyle bilinir.
 - **Kullanım Alanları**:** Giyilebilir cihazlar, sağlık izleme cihazları, akıllı ev ürünleri.
- ▶ 2. **Zigbee**
 - **Nedir?**:** Düşük güç tüketimi ve düşük veri hızı gerektiren IoT uygulamaları için tasarlanmış bir kablosuz iletişim protokolüdür.
 - **Kullanım Alanları**:** Akıllı aydınlatma, güvenlik sistemleri, enerji yönetimi.

3. **IoT Protokolleri ve İletişim Teknolojileri**

- ▶ 3. **Z-Wave**
 - ▶ - **Nedir?**: Ev otomasyonu ve IoT uygulamaları için kullanılan düşük güçlü bir radyo frekansı iletişim teknolojisidir.
 - ▶ - **Kullanım Alanları**: Akıllı kilitler, akıllı termostatlar, güvenlik sistemleri.
- ▶ 4. **LoRaWAN (Long Range Wide Area Network)**
 - ▶ - **Nedir?**: Uzun mesafeli, düşük güç tüketimli, geniş alan ağları için tasarlanmış bir iletişim protokolüdür. Büyük coğrafi alanlarda az miktarda veri iletmek için idealdir.
 - ▶ - **Kullanım Alanları**: Tarım, akıllı şehirler, çevresel izleme.

3. **IoT Protokoller ve İletişim Teknolojileri**

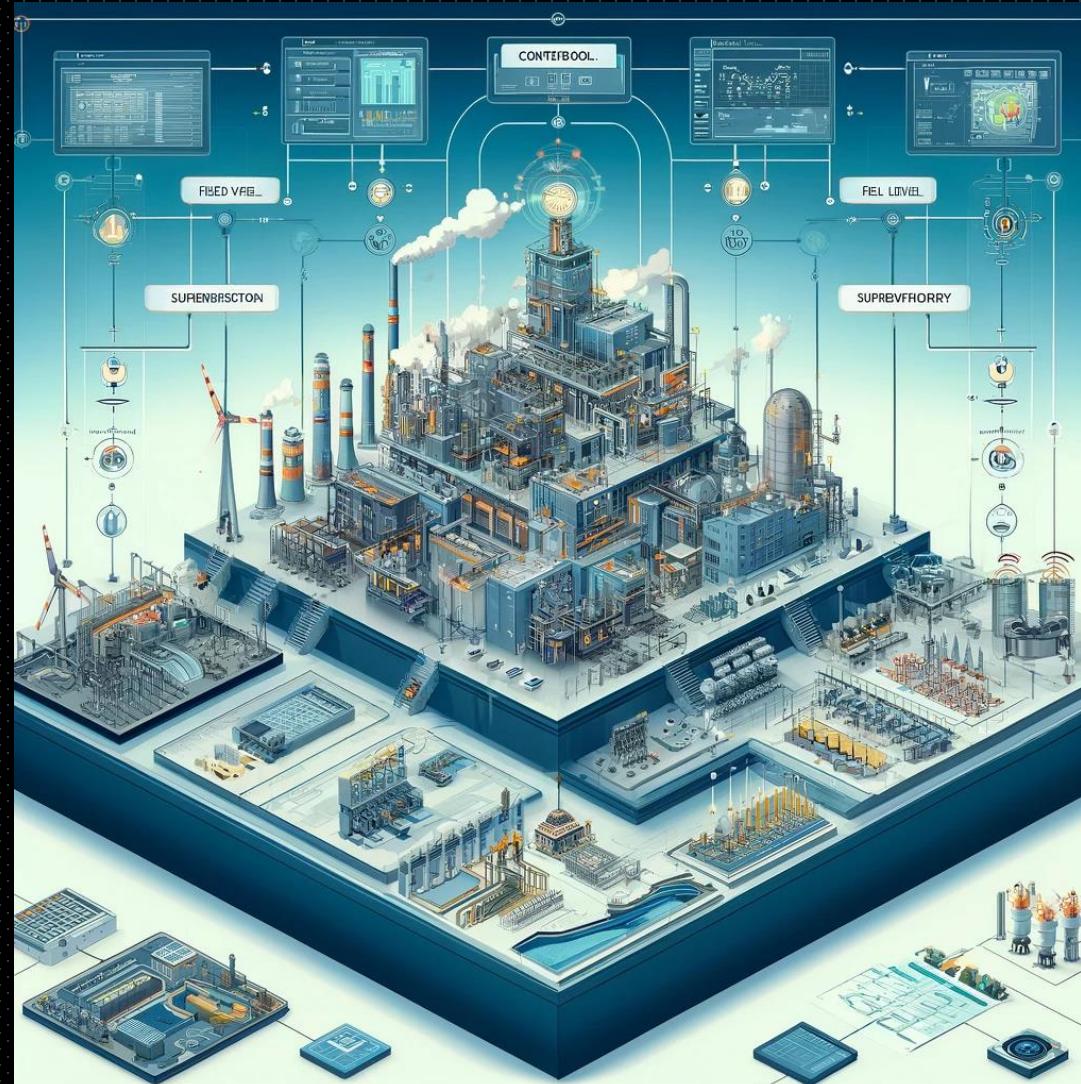
- ▶ 5. **NB-IoT (Narrowband IoT)**
 - ▶ - **Nedir?**: Hücresel ağlar üzerinde çalışan düşük güçlü geniş alan (LPWA) teknolojisidir. Uzun pil ömrü ve geniş kapsama alanı sağlar.
 - ▶ - **Kullanım Alanları**: Akıllı ölçüm sistemleri, şehir altyapısı izleme, sağlık cihazları.
- ▶ Bu protokoller ve iletişim teknolojileri, IoT ekosisteminin temelini oluşturur ve cihazların güvenilir ve verimli bir şekilde veri alışverişi yapmalarını sağlar.

4. **Veri Toplama ve Analiz**

- ▶ 4. **Veri Toplama ve Analiz**
 - ▶ - IoT veri yönetimi
 - ▶ - Büyük veri ve IoT
 - ▶ - Veri analitiği ve makine öğrenimi

- ▶ 5. **Güvenlik ve Gizlilik**
 - ▶ - IoT güvenlik tehditleri ve zorlukları
 - ▶ - Güvenlik çözümleri ve en iyi uygulamalar
 - ▶ - Veri gizliliği ve kullanıcı verilerinin korunması

OT (Operational Technology)



OT (Operational Technology)

- ▶ ### OT (Operational Technology)
- ▶ 1. **Temel Kavramlar ve Tanımlar**
- ▶ - OT nedir?
- ▶ - OT bileşenleri: PLC, SCADA, DCS
- ▶ - OT sistemlerinin genel mimarisi

OT (Operational Technology)

- ▶ OT (Operational Technology), fiziksel cihazların ve endüstriyel süreçlerin izlenmesi, kontrol edilmesi ve yönetilmesi için kullanılan donanım ve yazılım sistemlerini ifade eder.
- ▶ OT, fabrikalar, enerji santralleri, su arıtma tesisleri, ulaşım sistemleri ve diğer endüstriyel ortamlarda yaygın olarak kullanılır.
- ▶ OT sistemleri, üretim süreçlerinin verimli, güvenli ve kesintisiz bir şekilde işletilmesini sağlar.

OT (Operational Technology)

- ▶ ****OT'nin Temel Özellikleri ve İşlevleri:****
- ▶ 1. ****Gerçek Zamanlı Kontrol:****
 - OT sistemleri, endüstriyel süreçleri gerçek zamanlı olarak izler ve kontrol eder. Bu, süreçlerin anında ve doğru bir şekilde yönetilmesini sağlar. Örneğin, bir fabrikada üretim hattındaki bir arıza anında tespit edilip düzeltilebilir.
- ▶ 2. ****Endüstriyel Proseslerin Yönetimi:****
 - OT sistemleri, üretim hatları, enerji dağıtım ağları, su ve atık su arıtma tesisleri gibi endüstriyel proseslerin yönetiminde kullanılır. Bu sistemler, süreçlerin optimize edilmesi ve verimliliğin artırılması için kritik öneme sahiptir.

OT (Operational Technology)

- ▶ 3. **Fiziksel Cihazlarla Etkileşim:**
 - ▶ - OT, sensörler, aktüatörler, motorlar ve diğer fiziksel cihazlarla doğrudan etkileşim kurar. Bu cihazlardan gelen verileri toplar, analiz eder ve uygun kontrol komutlarını gönderir.
- ▶ 4. **Güvenlik ve Güvenilirlik:**
 - ▶ - OT sistemleri, endüstriyel süreçlerin güvenliğini ve güvenilirliğini sağlamak için tasarlanmıştır. Bu sistemler, güvenlik risklerini minimize etmek, operasyonel hataları önlemek ve kesintisiz bir çalışma sağlamak için sürekli izleme ve kontrol yapar.

OT (Operational Technology)

- ▶ 5. **Veri Toplama ve Analiz:**
- ▶ - OT sistemleri, geniş bir veri toplama ve analiz yeteneğine sahiptir. Bu, işletmelerin süreçlerini izlemelerini, performansı değerlendirmelerini ve iyileştirme alanlarını belirlemelerini sağlar. Örneğin, bir enerji santralinde üretilen enerjinin verimliliği ve tüketimi sürekli olarak izlenir ve analiz edilir.

OT (Operational Technology) Bileşenleri

- ▶ **OT'nin Bileşenleri:**
- ▶ 1. **PLC (Programmable Logic Controller):**
 - Programlanabilir Lojik Kontrolörler, endüstriyel otomasyon süreçlerini kontrol etmek için kullanılan özel amaçlı bilgisayarlardır. PLC'ler, giriş ve çıkış modülleri aracılığıyla çeşitli sensör ve aktuatörlerle etkileşim kurarak makinelerin ve süreçlerin otomatik kontrolünü sağlar.

OT (Operational Technology)

- ▶ 2. **SCADA (Supervisory Control and Data Acquisition):**
 - SCADA sistemleri, geniş ölçekli endüstriyel süreçlerin izlenmesi ve kontrol edilmesi için kullanılır. SCADA, veri toplama, uzaktan kontrol ve analiz yeteneklerine sahiptir. Bu sistemler, operatörlerin süreçleri izlemelerine, kontrol etmelerine ve gerekiğinde müdahale etmelerine olanak tanır.
- ▶ 3. **DCS (Distributed Control System):**
 - Dağıtılmış Kontrol Sistemleri, büyük ve karmaşık endüstriyel tesislerdeki süreçleri kontrol etmek için kullanılır. DCS, merkezi olmayan bir yapıya sahip olup, birden fazla kontrol cihazı ve bilgisayar arasında dağıtılmıştır. Bu sistemler, özellikle kimya, petrol rafinerisi ve enerji santralleri gibi sektörlerde yaygındır.

OT (Operational Technology)

- ▶ ****OT Sistemlerinin Kullanım Alanları:****
- ▶ - **Üretim:** Otomotiv, gıda ve içecek, elektronik gibi sektörlerde üretim hatlarının otomatik kontrolü ve izlenmesi.
- ▶ - **Enerji:** Elektrik üretim ve dağıtım sistemlerinin yönetimi ve izlenmesi.
- ▶ - **Ulaşım:** Demiryolu, metro, havaalanı ve liman gibi ulaşım altyapılarının kontrolü.
- ▶ - **Su ve Atık Su Yönetimi:** Su arıtma tesisleri ve atık su yönetim sistemlerinin izlenmesi ve kontrolü.
- ▶ - **Petrokimya:** Petrol ve gaz rafinerileri, kimyasal üretim tesislerinin yönetimi.

OT (Operational Technology)

- ▶ OT, endüstriyel operasyonların verimli, güvenli ve sürdürülebilir bir şekilde yönetilmesini sağlar.
- ▶ Modern endüstriyel süreçlerde OT, işletmelerin rekabet avantajı elde etmeleri ve operasyonel mükemmeliyeti sağlamaları için vazgeçilmez bir bileşendir.

OT (Operational Technology)

OT (Operational Technology)

OT (Operational Technology)

OT (Operational Technology)

OT (Operational Technology)

OT (Operational Technology)

OT (Operational Technology)

OT (Operational Technology)

OT (Operational Technology)

OT (Operational Technology)

OT Uygulama Alanları

- ▶ **OT Uygulama Alanları**
 - ▶ - Üretim ve Endüstriyel Otomasyon
 - ▶ - Enerji Yönetimi ve Güç Sistemleri
 - ▶ - Su ve Atık Su Yönetimi
 - ▶ - Madencilik ve Metalurji
 - ▶ - Ulaşım ve Altyapı

OT Protokoller ve İletişim Teknolojileri

- ▶ 9. **OT Protokoller ve İletişim Teknolojileri**
 - ▶ - Modbus, Profibus, OPC UA
 - ▶ - Endüstriyel Ethernet ve saha veriyolu (Fieldbus) sistemleri

OT Protokollerı ve İletişim Teknolojileri

- ▶ ### Modbus
- ▶ Modbus, Modicon sistemleri tarafından geliştirilen bir iletişim protokolüdür. Elektronik cihazlar arasında seri hatlar üzerinden bilgi传递i için kullanılır. Bilgi talep eden cihaza Modbus Master, bilgi sağlayan cihazlara ise Modbus Slave denir. Endüstriyel ortamlarda çeşitli enstrümantasyon, kontrol cihazları ve diğer otomasyon sistemlerini bağlamak için yaygın olarak kullanılır.

OT Protokoller ve İletişim Teknolojileri

- ▶ ### Profibus
- ▶ Profibus (Process Field Bus), otomasyon teknolojisinde fieldbus iletişimini için bir standarttır. Otomatik üretim ve süreç otomasyon sistemlerindeki cihazlar arasında iletişim için kullanılır. Profibus, geniş bir ağ topolojileri yelpazesinde çalışabilir ve yüksek hızlı iletişim destekler, bu da onu gerçek zamanlı uygulamalar için uygun hale getirir.

OT Protokoller ve İletişim Teknolojileri

- ▶ ### OPC UA
- ▶ OPC UA (OPC Unified Architecture), endüstriyel otomasyon için OPC Vakfı tarafından geliştirilen bir makineden makineye iletişim protokolüdür. Endüstriyel sistemlerde güvenli ve güvenilir veri değişimi için birleşik bir çerçeve sağlamak üzere tasarlanmıştır. OPC UA platform bağımsızdır ve farklı donanım ve yazılım platformlarında kullanılabilir. Karmaşık veri yapıları, gerçek zamanlı veri değişimi ve güçlü güvenlik özelliklerini destekler.

OT Protokoller ve İletişim Teknolojileri

- ▶ ### Endüstriyel Ethernet
- ▶ Endüstriyel Ethernet, endüstriyel ortamlar için uyarlanmış bir Ethernet standartıdır. Geleneksel Ethernet teknolojisine dayanır, ancak endüstriyel otomasyon sistemlerinde kullanılmak üzere iyileştirilmiştir. Endüstriyel Ethernet, yüksek hızlı veri iletimi, düşük gecikme süresi ve güvenilirlik sağlar. Ayrıca, endüstriyel uygulamalar için gereklili olan dayanıklılık ve sertifikasyonlara sahiptir.

OT Protokoller ve İletişim Teknolojileri

- ▶ ### Saha Veriyolu (Fieldbus) Sistemleri
- ▶ Fieldbus, endüstriyel otomasyon sistemlerinde cihazlar arasında veri iletişimini için kullanılan bir ağ sistemidir. Geleneksel noktadan noktaya bağlantılar yerine, bir bus topolojisi kullanarak birden fazla cihazın aynı iletişim hattını paylaşmasına olanak tanır. Fieldbus sistemleri, daha az kablolama, kolay kurulum ve bakım ile birlikte yüksek güvenilirlik sağlar. Çeşitli fieldbus protokolleri arasında Profibus, Foundation Fieldbus ve DeviceNet yer alır.
- ▶ Bu protokoller ve sistemler, endüstriyel otomasyonun temel taşılarıdır ve cihazlar arasında verimli, güvenli ve güvenilir veri iletişimini sağlarlar.

OT Güvenlik ve Dayanıklılık

- ▶ 10. **Güvenlik ve Dayanıklılık**
 - ▶ - OT güvenlik tehditleri ve zorlukları
 - ▶ - Siber güvenlik ve fiziksel güvenlik önlemleri
 - ▶ - Endüstriyel kontrol sistemleri (ICS) güvenliği

Entegrasyon ve Interoperabilite

- ▶ 11. **Entegrasyon ve Interoperabilite**
 - ▶ - IT/OT entegrasyonu
 - ▶ - Endüstri 4.0 ve akıllı fabrikalar
 - ▶ - OT sistemlerinin modernizasyonu ve dijital dönüşüm

Entegrasyon ve Interoperabilite

- ▶ Operational Technology (OT) sistemleri, özellikle endüstriyel kontrol sistemleri (ICS), günümüzde hem siber güvenlik hem de fiziksel güvenlik tehditlerine maruz kalmaktadır. Bu tehditlerin yönetilmesi ve güvenliğin sağlanması kritik öneme sahiptir. İşte OT güvenlik tehditleri ve zorlukları ile ilgili ayrıntılar:

OT Güvenlik Tehditleri ve Zorlukları

- ▶ ### OT Güvenlik Tehditleri ve Zorlukları
- ▶ ##### Siber Güvenlik Tehditleri
- ▶ 1. ****Malware ve Ransomware****: OT sistemlerine yönelik zararlı yazılımlar ve fidye yazılımları, üretim süreçlerini durdurarak büyük maddi zararlara yol açabilir.
- ▶ 2. ****İleri Seviye Kalıcı Tehditler (APT)****: Uzun süreli ve gizli siber saldırılar, OT sistemlerinin zayıf noktalarını hedefleyerek hassas bilgileri çalabilir veya sistemleri sabote edebilir.
- ▶ 3. ****Ağ Tabanlı Saldırılar****: OT ağlarına yönelik DDoS (Distributed Denial of Service) saldırıları, ağ trafiğini kesintiye uğratarak sistemlerin işleyişini bozabilir.

OT Güvenlik Tehditleri ve Zorlukları

- ▶ 4. **İç Tehditler**: İçeriden gelen tehditler, yetkili personelin kasıtlı veya kazara yapabileceği eylemlerden kaynaklanabilir.

OT Güvenlik Tehditleri ve Zorlukları

- ▶ ##### Fiziksel Güvenlik Tehditleri
- ▶ 1. **Yetkisiz Erişim**: Yetkisiz kişilerin fiziksel olarak tesislere ve kritik altyapılara erişim sağlamaşı.
- ▶ 2. **Sabotaj ve Hırsızlık**: Fiziksel bileşenlerin sabote edilmesi veya çalınması, OT sistemlerinin çalışmasını ciddi şekilde etkileyebilir.
- ▶ 3. **Doğal Afetler ve Kazalar**: Depremler, yangınlar, sel gibi doğal afetler ve diğer kazalar, OT sistemlerinin fiziksel altyapısına zarar verebilir.

OT Güvenlik Tehditleri ve Zorlukları

- ▶ **### Siber Güvenlik Önlemleri**
- ▶ 1. ****Ağ Segmentasyonu****: OT ağlarının, IT ağlarından ayrılması ve OT içinde kritik sistemlerin segmentasyon ile korunması.
- ▶ 2. ****Güvenlik Duvarları ve VPN'ler****: OT sistemlerine erişimi kontrol etmek için güvenlik duvarları ve sanal özel ağlar (VPN) kullanımı.
- ▶ 3. ****Düzenli Güvenlik Güncellemeleri****: Yazılım ve donanım bileşenlerinin düzenli olarak güncellenmesi ve yamalanması.
- ▶ 4. ****Saldırı Tespit ve Önleme Sistemleri (IDS/IPS)****: OT ağlarında şüpheli aktiviteleri tespit eden ve engelleyen sistemler.

OT Güvenlik Tehditleri ve Zorlukları

- ▶ **### Fiziksel Güvenlik Önlemleri**
- ▶ 1. ****Erişim Kontrolü****: Yetkilendirilmiş personel dışında kimsenin OT alanlarına girmemesini sağlamak için erişim kontrol sistemleri kullanımı.
- ▶ 2. ****Gözetim ve İzleme****: Kritik altyapı ve tesislerin güvenlik kameraları ve diğer izleme teknolojileri ile sürekli gözetim altında tutulması.
- ▶ 3. ****Fiziksel Bariyerler ve Kilit Sistemleri****: Hassas bölgelerin fiziksel bariyerler ve güçlü kilit sistemleri ile korunması.
- ▶ 4. ****Acil Durum Protokolleri ve Eğitimler****: Personelin acil durumlarda nasıl hareket edeceğini belirleyen protokoller ve düzenli güvenlik eğitimleri.

OT Güvenlik Tehditleri ve Zorlukları

- ▶ ### Endüstriyel Kontrol Sistemleri (ICS) Güvenliği
- ▶ 1. **Zayıflık Yönetimi**: ICS bileşenlerinde bulunan zayıflıkların tespit edilmesi ve giderilmesi.
- ▶ 2. **Güvenli Yazılım Geliştirme**: ICS yazılımlarının güvenli bir şekilde geliştirilmesi ve test edilmesi.
- ▶ 3. **Olay Müdahale Planları**: Güvenlik ihlalleri durumunda hızlı ve etkili bir şekilde müdahale edebilmek için detaylı olay müdahale planları oluşturulması.
- ▶ 4. **Sürekli İzleme ve Analiz**: ICS sistemlerinin sürekli olarak izlenmesi ve güvenlik olaylarının analiz edilmesi.
- ▶ Bu önlemler ve stratejiler, OT sistemlerinin güvenliğini sağlamak için kritik öneme sahiptir ve endüstriyel işletmelerin operasyonlarını güvenli bir şekilde sürdürmelerine yardımcı olur.

OT Veri Toplama ve Analiz

- ▶ 12. **Veri Toplama ve Analiz**
 - ▶ - Sensör verileri ve gerçek zamanlı izleme
 - ▶ - Büyük veri ve analitik uygulamaları
 - ▶ - Veri entegrasyonu ve yönetimi

OT Standartlar ve Düzenlemeler

- ▶ 13. **Standartlar ve Düzenlemeler**
 - ▶ - OT standartları ve protokoller
 - ▶ - Düzenleyici gereklilikler ve uyumluluk

OT Standartlar ve Düzenlemeler

- ▶ ### OT Standartları ve Protokoller
- ▶ OT (Operational Technology) sistemlerinin güvenli ve verimli bir şekilde çalışabilmesi için belirli standartlar ve protokoller geliştirilmiştir. Bu standartlar ve protokoller, cihazlar ve sistemler arasında uyumluluğu ve güvenliği sağlamak amacıyla kullanılır. İşte bazı önemli OT standartları ve protokoller:

OT Standartlar ve Düzenlemeler

- ▶ ##### OT Standartları
- ▶ 1. **IEC 62443**: Endüstriyel otomasyon ve kontrol sistemleri (IACS) için siber güvenlik standartları sağlayan bir dizi standarttır. Bu standartlar, tüm yaşam döngüsü boyunca güvenliğin sağlanması için gereklilikleri belirler.
- ▶ 2. **NIST SP 800-82**: Ulusal Standartlar ve Teknoloji Enstitüsü (NIST) tarafından yayınlanan bu kılavuz, endüstriyel kontrol sistemleri için bilgi güvenliği yönetimi konusunda rehberlik sağlar.
- ▶ 3. **ISO/IEC 27001**: Bilgi güvenliği yönetim sistemleri (ISMS) için uluslararası bir standarttır ve OT sistemlerinin bilgi güvenliği yönetimi ile ilgili gereklilikleri belirler.
- ▶ 4. **ISA-95**: Üretim operasyonları yönetimi ve iş süreçleri arasında entegrasyon için bir standarttır. Bu standart, OT ve IT sistemlerinin birlikte çalışabilirliğini sağlar.

OT Standartlar ve Düzenlemeler

- ▶ ##### OT Protokoller
- ▶ 1. **Modbus**: Seri ve TCP/IP üzerinden veri iletimi sağlayan yaygın bir endüstriyel protokoldür.
- ▶ 2. **Profibus**: Otomasyon ve kontrol sistemleri için yüksek hızlı iletişim sağlayan bir fieldbus protokolüdür.
- ▶ 3. **OPC UA**: Platform bağımsız, güvenli ve güvenilir veri değişimi sağlayan bir iletişim protokolüdür.
- ▶ 4. **EtherCAT**: Yüksek performanslı, Ethernet tabanlı bir fieldbus protokolüdür ve özellikle otomasyon uygulamalarında kullanılır.
- ▶ 5. **HART**: Analog ve dijital sinyallerin aynı anda iletilmesini sağlayan bir protokoldür, genellikle proses kontrolü için kullanılır.

OT Standartlar ve Düzenlemeler

- ▶ ### Düzenleyici Gereklilikler ve Uyumluluk
- ▶ OT sistemlerinin güvenliğini ve verimliliğini sağlamak için birçok düzenleyici gereklilik ve uyumluluk standartları belirlenmiştir. Bu gereklilikler, işletmelerin belirli güvenlik ve operasyonel standartlara uymalarını sağlar.

OT Standartlar ve Düzenlemeler

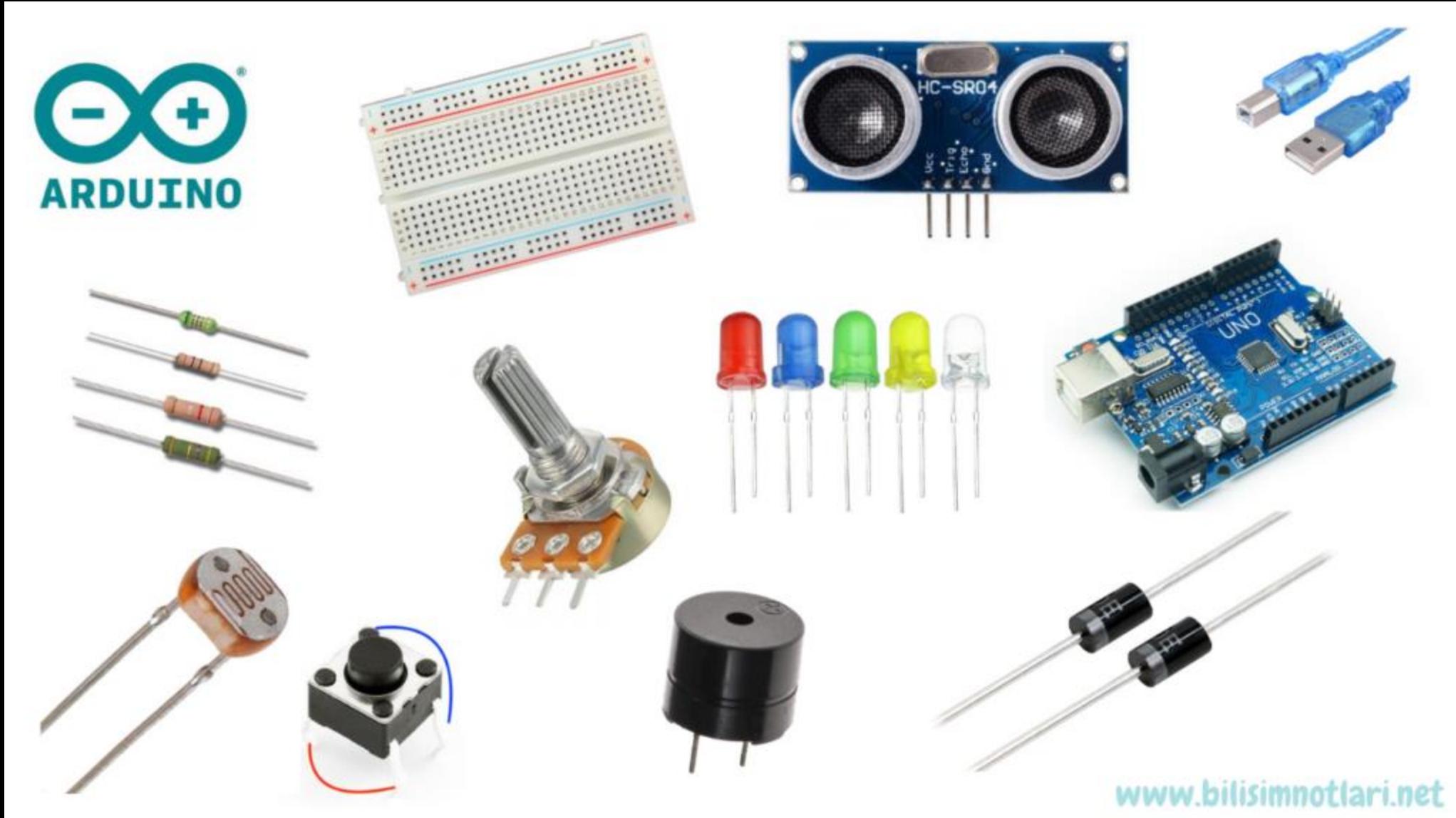
- ▶ ##### Dözenleyici Gereklikler
- ▶ 1. **NERC CIP**: Kuzey Amerika Elektrik Güvenilirlik Kurumu (NERC) tarafından belirlenen kritik altyapı koruma (CIP) standartları, elektrik şebekelerinin güvenliğini sağlamak için gereklilikler belirler.
- ▶ 2. **GDPR**: Genel Veri Koruma Yönetmeliği, Avrupa Birliği'nde veri koruma ve gizlilik konularında uygulanması gereken düzenlemeleri kapsar ve OT sistemlerinde veri işleme süreçleri için de geçerlidir.
- ▶ 3. **HIPAA**: Sağlık sigortası taşınabilirlik ve hesap verebilirlik yasası, sağlık sektöründe veri güvenliği ve gizliliği sağlamak için belirlenmiş düzenlemeler içerir.
- ▶ 4. **FISMA**: Federal Bilgi Güvenliği Yönetim Yasası, ABD federal kurumlarının bilgi güvenliği yönetimi ve koruma gerekliliklerini belirler.

OT Standartlar ve Düzenlemeler

- ▶ ##### Uyumluluk
- ▶ 1. ****Güvenlik Denetimleri ve Sertifikasyonlar****: Düzenleyici gerekliliklere uyumu sağlamak için düzenli güvenlik denetimleri yapılmalı ve gerekli sertifikasyonlar alınmalıdır.
- ▶ 2. ****Eğitim ve Farkındalık Programları****: Personelin güvenlik farkındalığını artırmak ve uyumluluk gereklilikleri hakkında bilgi sahibi olmalarını sağlamak için düzenli eğitim programları düzenlenmelidir.
- ▶ 3. ****Güvenlik Politikaları ve Prosedürleri****: Kurum içi güvenlik politikaları ve prosedürleri oluşturulmalı ve bu politikalar düzenleyici gerekliliklere uyumlu olmalıdır.
- ▶ 4. ****Risk Yönetimi ve Değerlendirmeleri****: OT sistemlerinde risk yönetimi süreçleri ve düzenli risk değerlendirmeleri yapılmalıdır.

- ▶ Bu standartlar ve düzenlemeler, OT sistemlerinin güvenli, uyumlu ve verimli bir şekilde çalışmasını sağlamak için kritik öneme sahiptir. İşletmeler, bu gerekliliklere uyararak hem güvenliklerini artırabilir hem de yasal sorumluluklarını yerine getirebilirler.

Elektronik Devre Kartları

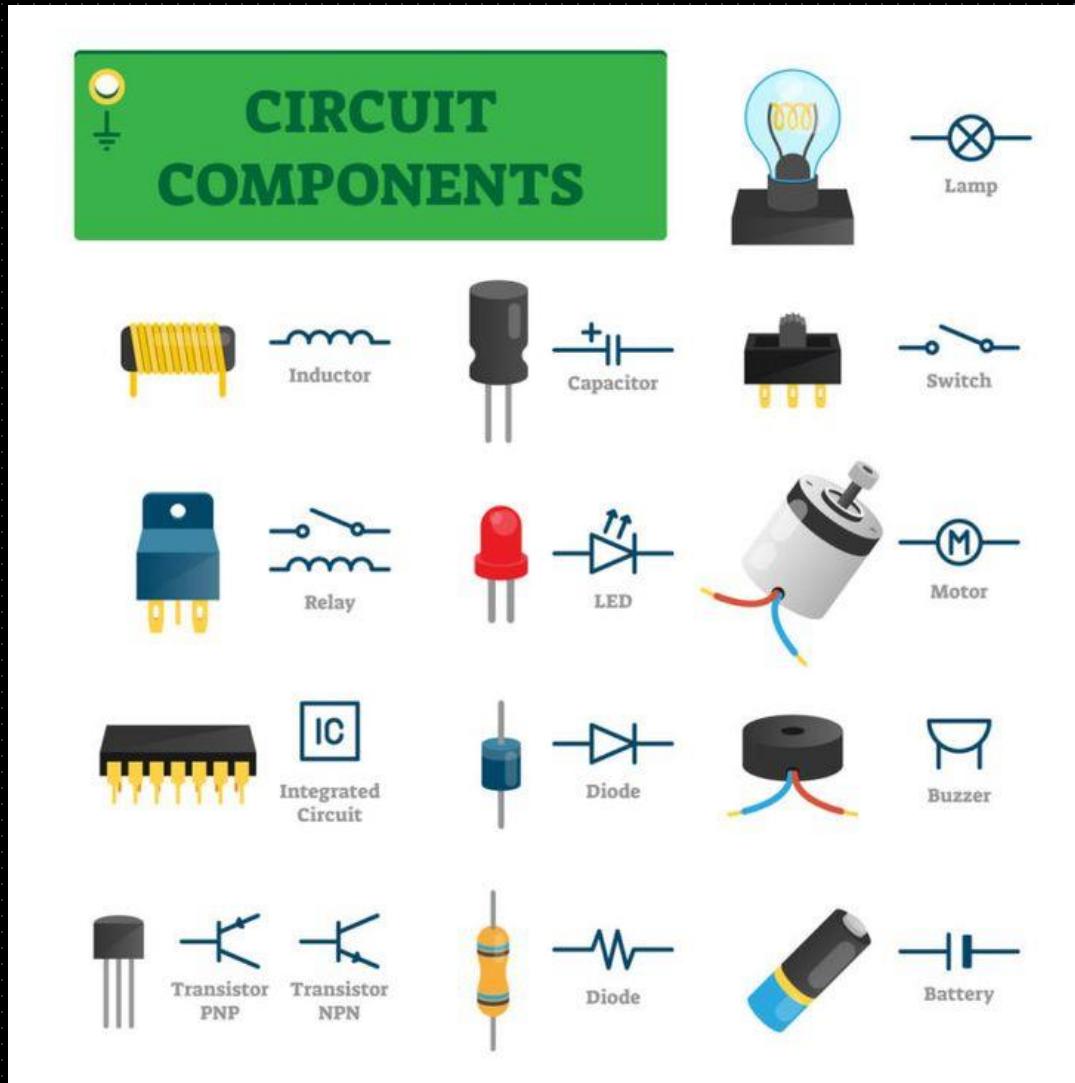


www.bilisimnotlari.net

Elektronik Devre Kartları

- ▶ ****Elektronik Devre Kartları Nedir?****
- ▶ Elektronik devre kartları, çeşitli elektronik bileşenlerin (dirençler, kapasitörler, transistörler, entegre devreler, vb.) belirli bir düzen içinde bir araya getirilerek, belirli işlevleri gerçekleştiren devrelerin oluşturulduğu fiziksel platformlardır. Elektronik devre kartları, elektronik cihazların ve sistemlerin temel yapı taşlarını oluşturur ve birçok uygulamada kullanılır.

Elektronik Devre Kartları



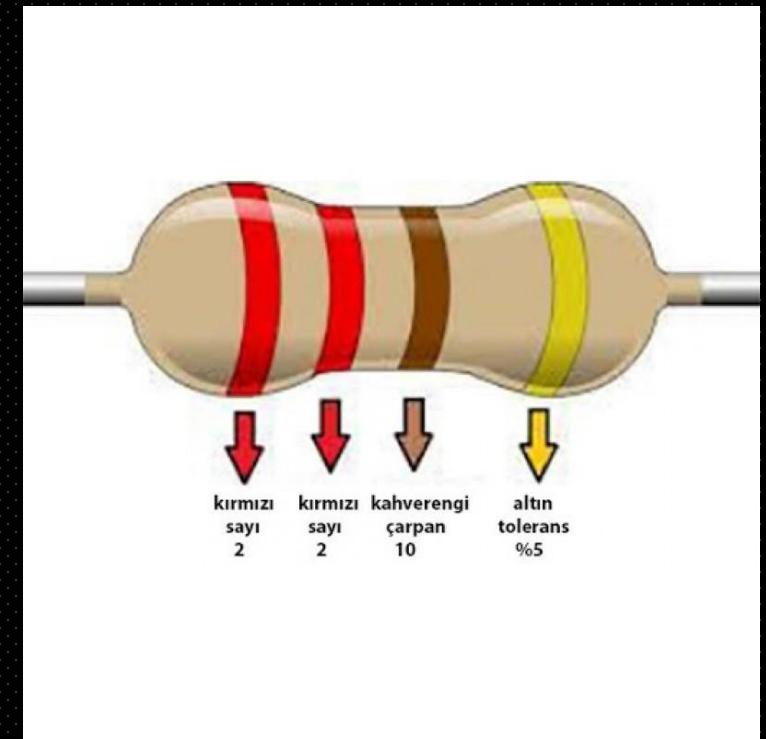
Elektronik Devre Kartlarının Temel Bileşenler

- ▶ ### Elektronik Devre Kartlarının Temel Bileşenleri:
- ▶ 1. **Baskı Devre Kartı (PCB - Printed Circuit Board):**
 - **Açıklama:** Baskı devre kartı, elektronik bileşenleri birbirine bağlamak için kullanılan yalıtkan bir levhadır. PCB'ler, bakır izlerle kaplanmış katmanlara sahiptir. Bu izler, bileşenler arasındaki elektriksel bağlantıları sağlar. PCB'ler, tek katmanlı, çift katmanlı veya çok katmanlı olabilir.
 - **Malzemeler:** Genellikle fiberglas (FR4) veya fenolik reçine gibi yalıtkan malzemelerden yapılır. Üstünde bakır izler bulunur.

Elektronik Devre Kartlarının Temel Bileşenler

- ▶ 2. **Elektronik Bileşenler:**
- ▶ - **Dirençler:** Elektrik akımını sınırlamak için kullanılır. Farklı direnç değerlerine ve güç derecelerine sahip olabilirler.

Renk	1. Şerit (ilk basamak)	2. Şerit (ikinci basamak)	3. Şerit (üçüncü basamak)	4. Şerit (katsayı)	5. Şerit (tolerans)	6. Şerit (sıcaklık katsayısı)
Siyah	0	0	0	$\times 10^0$		
Kahve	1	1	1	$\times 10^1$	$\pm \%1$	100
Kırmızı	2	2	2	$\times 10^2$	$\pm \%2$	50
Turuncu	3	3	3	$\times 10^3$	$\pm \%3$	15
Sarı	4	4	4	$\times 10^4$	$\pm \%4$	25
Yeşil	5	5	5	$\times 10^5$	$\pm \%0.5$	
Mavi	6	6	6	$\times 10^6$	$\pm \%0.25$	10
Mor	7	7	7	$\times 10^7$	$\pm \%0.1$	5
Gri	8	8	8	$\times 10^8$	$\pm \%0.05$	
Beyaz	9	9	9	$\times 10^9$	$\pm \%1$	
Altın					$\pm \%5$	
Gümüş					$\pm \%10$	



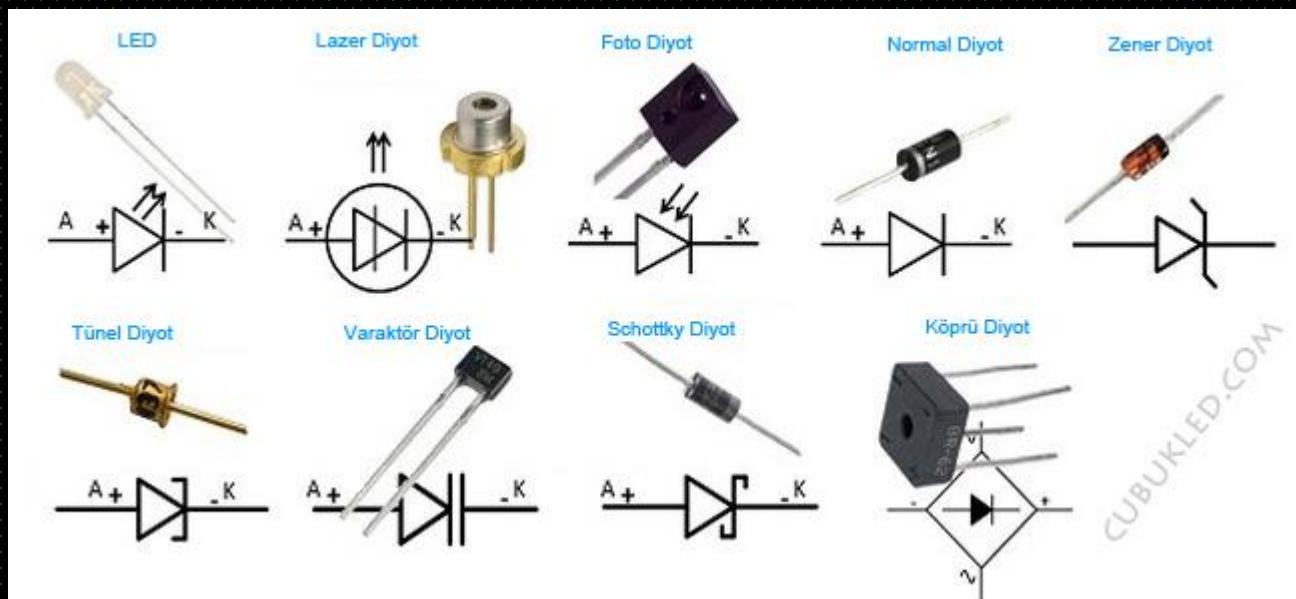
Elektronik Devre Kartlarının Temel Bileşenler

- ▶ 2. **Elektronik Bileşenler:**
- ▶ - **Kapasitörler:** Elektrik enerjisini depolamak ve salmak için kullanılır. Farklı kapasitans değerlerine ve gerilim derecelerine sahip olabilirler.



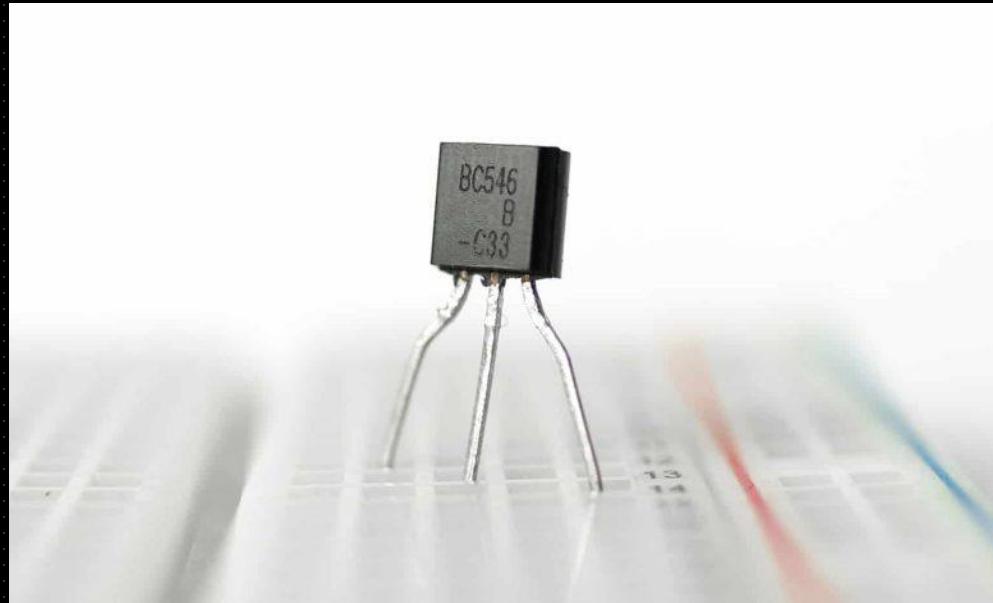
Elektronik Devre Kartlarının Temel Bileşenler

- ▶ 2. **Elektronik Bileşenler:**
- ▶ - **Diyotlar:** Elektriğin sadece bir yönde akmasına izin veren bileşenlerdir. LED'ler (Işık Yayan Diyotlar) ve zener diyotlar gibi çeşitli tipleri vardır.



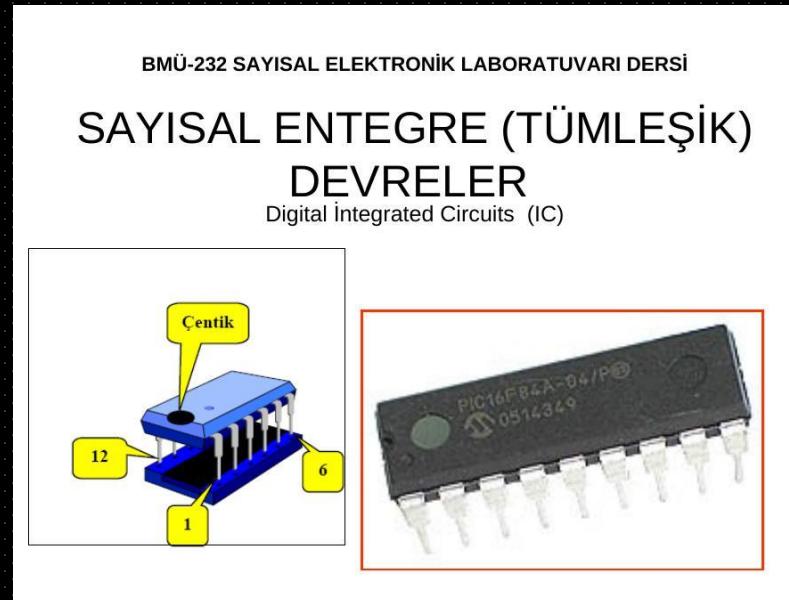
Elektronik Devre Kartlarının Temel Bileşenler

- ▶ 2. **Elektronik Bileşenler:**
- ▶ - **Transistörler:** Elektrik sinyallerini güçlendirmek veya anahtarlamak için kullanılır. Bipolar bağlantı transistörleri (BJT) ve alan etkili transistörler (FET) gibi çeşitleri vardır.



Elektronik Devre Kartlarının Temel Bileşenler

- ▶ 2. **Elektronik Bileşenler:**
- ▶ - **Entegre Devreler (IC'ler):** Birçok elektronik bileşenin tek bir çip içinde birleştirildiği bileşenlerdir. Mikrodenetleyiciler, işlemciler ve op-amp'ler gibi çeşitli türleri vardır.
- ▶



Elektronik Devre Kartlarının Temel Bileşenler

- ▶ 2. **Elektronik Bileşenler:**
- ▶ - **Bağlantı Elemanları:** Pin headerlar, konektörler ve lehim noktaları gibi bileşenler, dış cihazlarla bağlantı kurmak için kullanılır.



Elektronik Devre Kartlarının Temel Bileşenler

- ▶ 3. **Lehim:**
- ▶ - **Açıklama:** Elektronik bileşenlerin PCB üzerindeki bakır izlere fizikalı ve elektriksel olarak bağlanması sağlar. Lehim genellikle kalay ve kurşun合金asımlı veya çevre dostu olarak kurşunsuz合金asımlardan yapılır.



Elektronik Devre Kartlarının Çeşitleri:

- ▶ ### Elektronik Devre Kartlarının Çeşitleri:
- ▶ 1. **Tek Taraflı PCB:**
 - **Açıklama:** Sadece bir yüzünde bakır izler bulunan devre kartlarıdır. Genellikle basit ve düşük maliyetli uygulamalar için kullanılır.
 - **Uygulama Alanları:** Basit elektronik oyuncaklar, düşük maliyetli cihazlar.

Elektronik Devre Kartlarının Çeşitleri:

- ▶ 2. **Çift Taraflı PCB:**
 - **Açıklama:** Her iki yüzünde de bakır izler bulunan devre kartlarıdır. Bileşenlerin yerleştirilmesi ve izlerin düzenlenmesi açısından daha fazla esneklik sağlar.
 - **Uygulama Alanları:** Orta karmaşıklıkta elektronik cihazlar, endüstriyel ekipmanlar.

Elektronik Devre Kartlarının Çeşitleri:

- ▶ 3. **Çok Katmanlı PCB:**
 - ▶ - **Açıklama:** Üç veya daha fazla katmandan oluşan devre kartlarıdır. İç katmanlar, daha karmaşık devre tasarımları ve daha yüksek yoğunluklu bileşen yerlesimi sağlar.
 - ▶ - **Uygulama Alanları:** Bilgisayar anakartları, iletişim ekipmanları, gelişmiş elektronik cihazlar.

Elektronik Devre Kartlarının Üretim Süreci:

- ▶ ### Elektronik Devre Kartlarının Üretim Süreci:
 - ▶ 1. **Tasarım:**
 - **Açıklama:** Elektronik devre kartları, bilgisayar destekli tasarım (CAD) yazılımları kullanılarak tasarılanır. Devre şeması oluşturulur ve bileşenlerin yerlesimi belirlenir.
 - **Yazılımlar:** Eagle, KiCAD, Altium Designer gibi PCB tasarım yazılımları.

Elektronik Devre Kartlarının Üretim Süreci:

- ▶ 2. **Üretim:**
 - **Açıklama:** Tasarım dosyaları, üreticiye gönderilir ve PCB üretim süreci başlar. Bu süreçte bakır kaplamalı levhalar, tasarım izlerine göre kazınır ve bileşen delikleri açılır.
 - **Aşamalar:** Fotolitografi, kimyasal kazıma, delik açma, bakır kaplama, lehim maskesi uygulama ve baskı.
- ▶ 3. **Montaj:**
 - **Açıklama:** Elektronik bileşenler, PCB'ye yerleştirilir ve lehimlenir. Montaj işlemi manuel veya otomatik makinelerle yapılabilir.
 - **Yöntemler:** Yüzey montaj teknolojisi (SMT) ve delikli montaj (THT).

Elektronik Devre Kartlarının Üretim Süreci:

- ▶ 4. **Test ve Kalite Kontrol:**
 - **Açıklama:** Üretilen devre kartları, işlevsellik ve kalite açısından test edilir. Hatalı bağlantılar veya kusurlu bileşenler tespit edilip düzelttilir.

Elektronik Devre Kartlarının Uygulama Alanları:

- ▶ ### Elektronik Devre Kartlarının Uygulama Alanları:
 - ▶ 1. **Tüketiciler Elektroniği:**
 - ▶ - **Açıklama:** Akıllı telefonlar, tabletler, televizyonlar, oyun konsolları gibi cihazlarda kullanılır.
 - ▶
 - ▶ 2. **Endüstriyel Otomasyon:**
 - ▶ - **Açıklama:** Üretim makineleri, otomasyon sistemleri, robotik uygulamalar gibi alanlarda kullanılır.
 - ▶

Elektronik Devre Kartlarının Üretim Süreci:

- ▶ 3. **Otomotiv Elektroniği:**
 - ▶ - **Açıklama:** Araç içi eğlence sistemleri, motor kontrol üniteleri, güvenlik sistemleri gibi uygulamalarda kullanılır.
 - ▶
- ▶ 4. **Tıbbi Cihazlar:**
 - ▶ - **Açıklama:** Medikal görüntüleme cihazları, teşhis ekipmanları, hasta izleme sistemleri gibi alanlarda kullanılır.
 - ▶

Elektronik Devre Kartlarının Üretim Süreci:

- ▶ 5. **Havacılık ve Savunma:**
 - ▶ - **Açıklama:** Uçak elektronik sistemleri, radar sistemleri, askeri iletişim ekipmanları gibi uygulamalarda kullanılır.
- ▶ Elektronik devre kartları, modern teknolojinin vazgeçilmez bir parçasıdır ve birçok endüstride kritik rol oynar. Bu kartlar, elektronik cihazların işlevsellliğini ve performansını belirler ve doğru tasarım ve üretim süreçleri, yüksek kaliteli ve güvenilir ürünlerin üretilmesini sağlar.

Direnç

- Direnç hesaplama, genellikle Ohm Kanunu kullanılarak yapılır. Ohm Kanunu, elektrik devrelerindeki gerilim (V), akım (I) ve direnç (R) arasındaki ilişkiyi tanımlar. Ohm Kanunu'na göre:

Direnç

- ▶ $\boxed{V = I \cdot R}$
- ▶ Bu denklem, herhangi iki değişken bilindiğinde üçüncüyü hesaplamak için kullanılabilir. Direnci hesaplamak için gerilim ve akım değerlerini bilmemiz gereklidir. Direnç şu formülle hesaplanır:
- ▶ $\boxed{R = \frac{V}{I}}$

Direnç

- ▶ ### Adım Adım Direnç Hesaplama

- ▶ 1. **Gerilim ve Akımı Belirleme:**
 - ▶ - Gerilim (V) volt cinsinden ölçülür.
 - ▶ - Akım (I) amper cinsinden ölçülür.

Direnç

- ▶ 2. **Direnci Hesaplama:**
 - ▶ - Direnç (R) ohm cinsinden ölçülür.
 - ▶ - Direnci hesaplamak için gerilimi akıma bölün:
$$R = \frac{V}{I}$$

Direnç

- ▶ ### Örnek Hesaplama
- ▶ 1. **Örnek Gerilim ve Akım Değerleri:**
 - ▶ - Gerilim: 10 volt
 - ▶ - Akım: 2 amper
- ▶ 2. **Direnç Hesaplama:**
 - ▶ - $R = \frac{10 \text{ V}}{2 \text{ A}} = 5 \text{ ohm}$

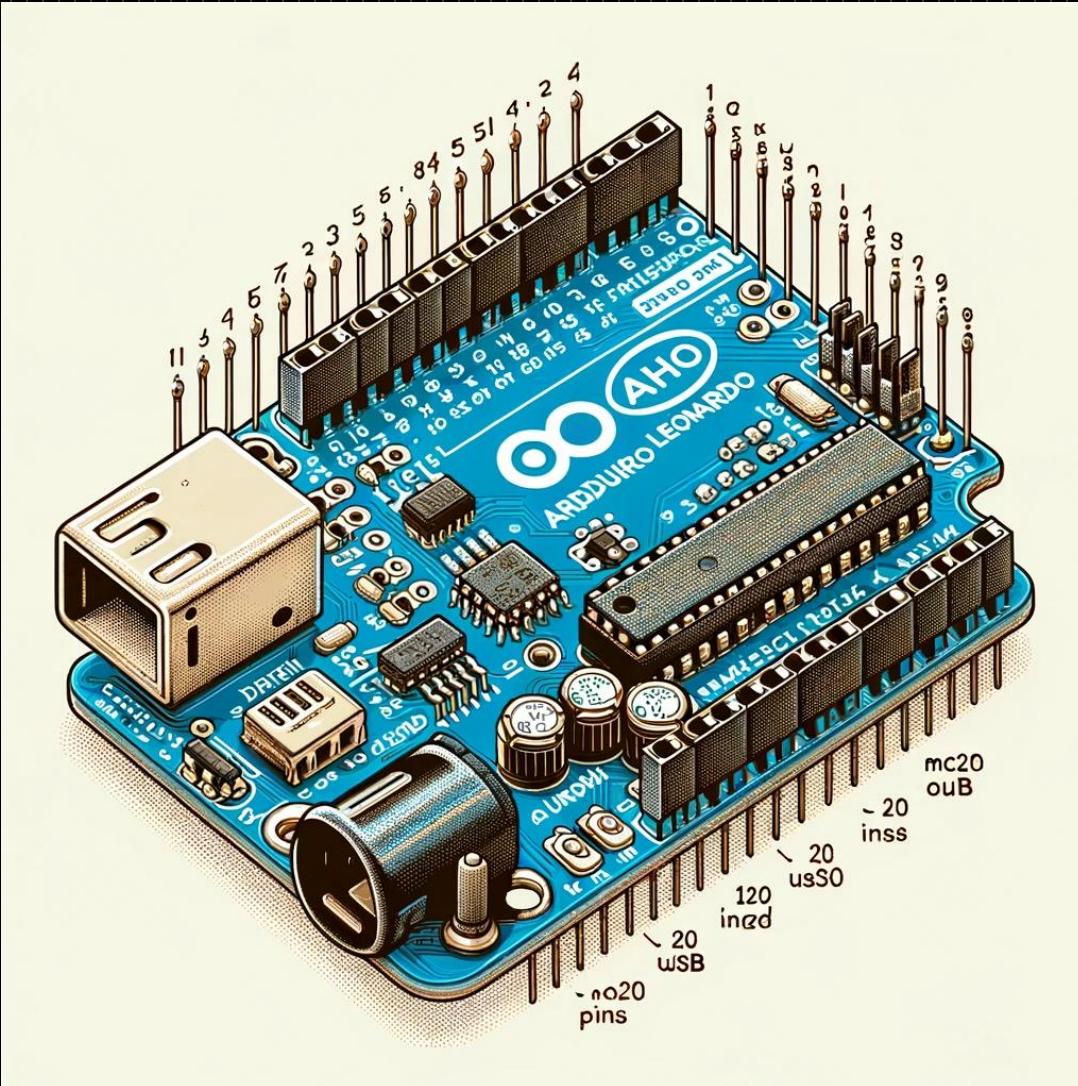
Direnç

- ▶ ### Seri ve Paralel Devrelerde Direnç Hesaplama
- ▶ **Seri Devre:**
 - Seri bağlı dirençlerin toplamı, tüm dirençlerin toplamına eşittir.
 - ▶
$$R_{\text{toplam}} = R_1 + R_2 + R_3 + \dots$$
- ▶ **Paralel Devre:**
 - Paralel bağlı dirençlerin toplamı, bireysel dirençlerin terslerinin toplamına eşittir.
 - ▶
$$\frac{1}{R_{\text{toplam}}} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} + \dots$$

Direnç

- ▶ ### Örnekler:
- ▶ 1. **Seri Devre:**
 - ▶ - $R_1 = 2 \text{ ohm}$, $R_2 = 3 \text{ ohm}$, $R_3 = 5 \text{ ohm}$
 - ▶ - $R_{\text{toplam}} = 2 + 3 + 5 = 10 \text{ ohm}$
- ▶ 2. **Paralel Devre:**
 - ▶ - $R_1 = 2 \text{ ohm}$, $R_2 = 3 \text{ ohm}$, $R_3 = 6 \text{ ohm}$
 - ▶ - $\frac{1}{R_{\text{toplam}}} = \frac{1}{2} + \frac{1}{3} + \frac{1}{6} = 1$
 - ▶ - $R_{\text{toplam}} = 1 \text{ ohm}$
- ▶ Bu temel prensipler, elektrik devrelerinde direnç hesaplamayı anlamak ve uygulamak için kullanılır.

Arduino



Arduino

- ▶ **Arduino Nedir?**
- ▶ Arduino, açık kaynaklı bir elektronik prototip geliştirme platformudur. Hem yazılım (IDE - Integrated Development Environment) hem de donanım (mikrodenetleyici kartları) içerir. Amacı, elektronik projeler oluşturmayı ve programlamayı kolaylaştırmak ve kullanıcı dostu hale getirmektir. Arduino platformu, hem amatör hem de profesyonel kullanıcılar tarafından geniş bir yelpazede projeler geliştirmek için kullanılır.

Arduino Temel Özellikleri

- ▶ **Arduino'nun Temel Özellikleri:**
- ▶ 1. **Donanım (Hardware):**
 - **Mikrodenetleyici:** Arduino kartlarının merkezinde genellikle Atmel AVR mikrodenetleyiciler (örneğin, ATmega328) bulunur. Bu mikrodenetleyiciler, kart üzerindeki giriş ve çıkış pinleri aracılığıyla çeşitli sensörler ve aktüatörlerle etkileşim kurar.
 - **Giriş/Çıkış Pinleri:** Dijital ve analog pinler, sensörler, motorlar, LED'ler ve diğer bileşenlerle bağlantı kurmak için kullanılır. Pinlerin sayısı ve türü, Arduino modeline göre değişir.
- ▶ - **Güç Kaynağı:** Arduino kartları, USB bağlantısı veya harici bir güç kaynağı (pil veya adaptör) aracılığıyla beslenir.
- ▶ - **Programlama Bağlantısı:** Arduino kartları, bilgisayara USB aracılığıyla bağlanır ve bu bağlantı üzerinden programlanır.

Arduino Temel Özellikleri

- ▶ 2. **Yazılım (Software):**
 - ▶ - **Arduino IDE:** Arduino projelerinin geliştirilmesi için kullanılan yazılım ortamıdır. Bu ortam, basit ve kullanıcı dostu bir arayüze sahiptir ve Arduino kartlarını programlamak için gereken tüm araçları sağlar. Arduino IDE, Windows, macOS ve Linux işletim sistemlerinde çalışır.
 - ▶ - **Arduino Programlama Dili:** C ve C++ dillerine dayanan basit bir programlama dilidir. Arduino IDE, bu dilde yazılan kodları derleyip mikrodenetleyiciye yükler. Arduino kodları, genellikle `setup()` ve `loop()` olmak üzere iki ana fonksiyondan oluşur:
 - ▶ - `setup()`: Programın başlangıcında bir kez çalıştırılan ayarları içerir.
 - ▶ - `loop()`: Sürekli olarak tekrar eden ana program döngüsünü içerir.

Arduino Temel Özellikleri

- ▶ 3. **Kütüphaneler (Libraries):**
- ▶ - Arduino, çeşitli sensörler, ekranlar, motorlar ve diğer bileşenlerle çalışmayı kolaylaştıran birçok hazır kütüphane içerir. Bu kütüphaneler, kullanıcıların karmaşık donanım kontrol işlemlerini basit kodlarla gerçekleştirmelerine olanak tanır.

Arduino'nun Kullanım Alanları

- ▶ ****Arduino'nun Kullanım Alanları:****
- ▶ 1. **Eğitim:**
 - Arduino, elektronik ve programlama öğrenmek için mükemmel bir araçtır. Okullarda ve üniversitelerde yaygın olarak kullanılır ve öğrencilerin pratik yaparak öğrenmelerine olanak tanır.
- ▶ 2. **Prototipleme:**
 - Mühendisler ve tasarımcılar, fikirlerini hızlı bir şekilde test etmek ve geliştirmek için Arduino'yu kullanır. Prototipleme sürecini hızlandırır ve maliyetleri düşürür.

Arduino'nun Kullanım Alanları

- ▶ 3. **Ev Otomasyonu:**
 - ▶ - Arduino, akıllı ev projeleri oluşturmak için kullanılabilir. Örneğin, aydınlatma sistemlerini kontrol etmek, güvenlik sistemleri oluşturmak veya iklim kontrol sistemleri geliştirmek için kullanılabilir.
- ▶ 4. **Robotik:**
 - ▶ - Arduino, robotik projelerde yaygın olarak kullanılır. Motor kontrolü, sensör verileri işleme ve karar verme süreçlerini yönetmek için ideal bir platformdur.
- ▶ 5. **Sanat ve Etkileşimli Projeler:**
 - ▶ - Sanatçılar ve tasarımcılar, interaktif sanat eserleri ve enstalasyonlar oluşturmak için Arduino'y'u kullanır. Sensörler ve aktuatörler aracılığıyla izleyicilerle etkileşime giren projeler yapılabilir.

Arduino Kart Modelleri

- ▶ ****Arduino Kart Modelleri:****
- ▶ - ****Arduino Uno:**** En yaygın ve popüler modeldir. Genellikle eğitim ve basit projeler için kullanılır.
- ▶ - ****Arduino Mega:**** Daha fazla giriş/çıkış pinine sahip olan bu model, daha karmaşık projeler için uygundur.
- ▶ - ****Arduino Nano:**** Küçük boyutlu bir model olup, kompakt projeler için idealdir.
- ▶ - ****Arduino Leonardo:**** USB bağlantısını emüle edebilen bu model, klavye veya fare gibi USB cihazları geliştirmek için kullanılabilir.

Örnek Proje: Basit LED Yanıp Sönme Projesi

- ▶ **Örnek Proje: Basit LED Yanıp Sönme Projesi**
- ▶ `` `cpp
- ▶ void setup() {
- ▶ // LED pinini çıkış olarak ayarla
- ▶ pinMode(13, OUTPUT);
- ▶ }
- ▶
- ▶ void loop() {
- ▶ // LED'i yak
- ▶ digitalWrite(13, HIGH);
- ▶ delay(1000); // 1 saniye bekle
- ▶ // LED'i söndür
- ▶ digitalWrite(13, LOW);
- ▶ delay(1000); // 1 saniye bekle
- ▶ }
- ▶ `` `

Örnek Proje: Basit LED Yanıp Sönme Projesi

- ▶ Bu basit program, Arduino'nun üzerindeki yerleşik LED'i bir saniye aralıklarla yakıp söndürür.
- ▶ Arduino, kullanıcı dostu yapısı, geniş kütüphane desteği ve güçlü topluluk desteği sayesinde elektronik projeler geliştirmek için ideal bir platformdur. Hem başlangıç seviyesindeki kullanıcılar hem de deneyimli mühendisler tarafından yaygın olarak kullanılmaktadır.

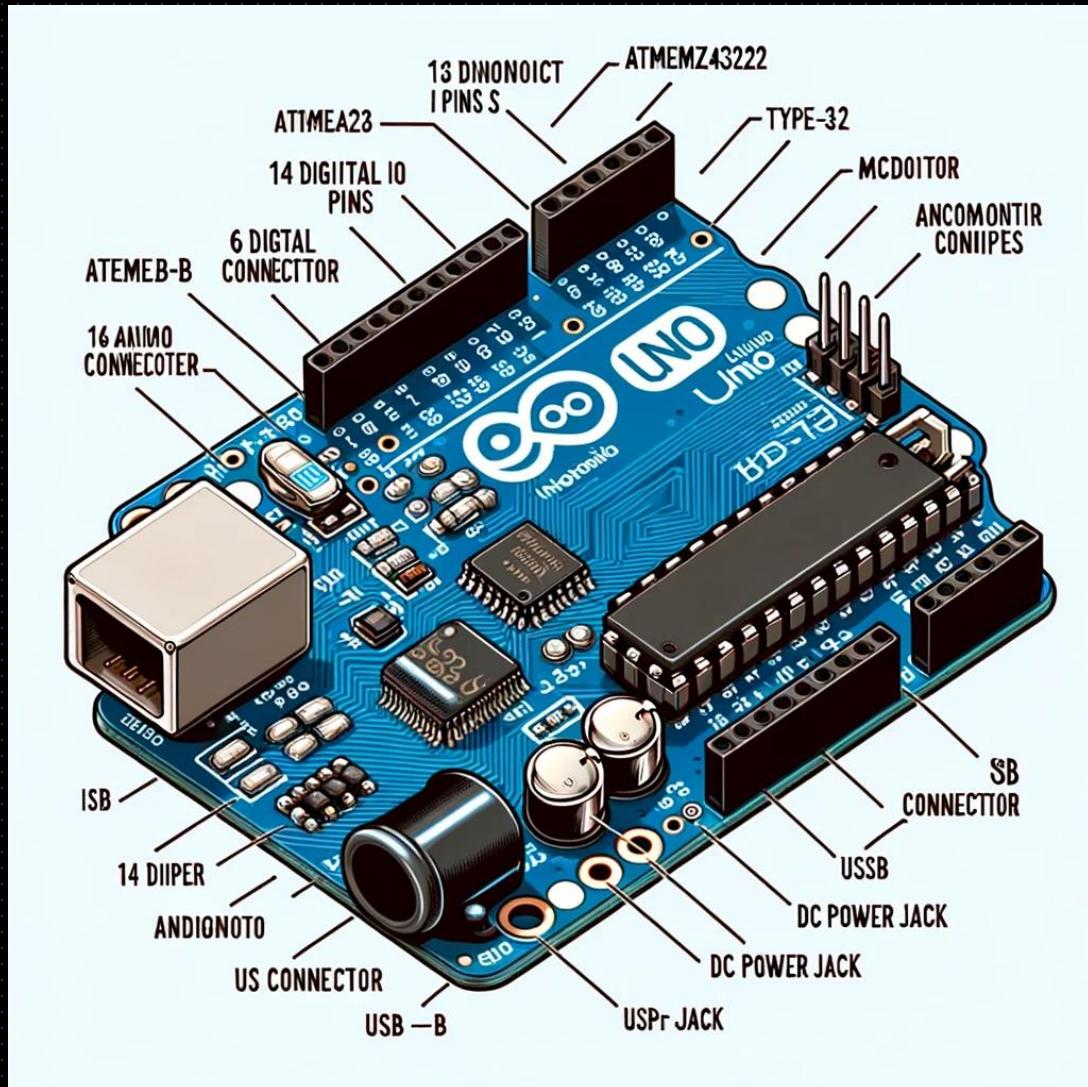
Arduino Modelleri

- ▶ Arduino platformu, çeşitli projeler ve uygulamalar için farklı modeller sunar. İşte en popüler Arduino modelleri ve her birinin ayrıntılı açıklamaları:

1. Arduino Uno

- ▶ ### 1. Arduino Uno
- ▶ - **Microcontroller:** ATmega328P
- ▶ - **Digital I/O Pins:** 14 (6 PWM çıkışı)
- ▶ - **Analog Input Pins:** 6
- ▶ - **Clock Speed:** 16 MHz
- ▶ - **Flash Memory:** 32 KB
- ▶ - **SRAM:** 2 KB
- ▶ - **EEPROM:** 1 KB
- ▶ - **Power Supply:** USB bağlantısı veya harici adaptör (7-12V)
- ▶ - **Özellikler:** Başlangıç seviyesi için ideal, geniş kullanım alanı.

1. Arduino Uno



1. Arduino Uno

- ▶ ### 1. Arduino Uno
- ▶ ##### Microcontroller: ATmega328P
- ▶ - **Açıklama:** ATmega328P, AVR mikrodenetleyici ailesinin bir üyesidir ve Arduino Uno'nun merkezi işlem birimidir. Bu mikrodenetleyici, 8-bit işlemci çekirdeğine sahiptir ve çeşitli kontrol ve izleme uygulamaları için idealdir. Düşük güç tüketimi ve yüksek performans özellikleriyle öne çıkar.

1. Arduino Uno

- ▶ ##### Digital I/O Pins: 14 (6 PWM çıkışı)
- ▶ - **Açıklama:** Arduino Uno'da 14 digital giriş/çıkış pini bulunur. Bu pinler, digital sinyalleri alabilir veya verebilir. Örneğin, bir butonun durumunu okumak veya bir LED'i açıp kapatmak için kullanılabilir. Bu pinlerin 6 tanesi, PWM (Pulse Width Modulation) çıkışı sağlayabilir. PWM, motor hız kontrolü, LED parlaklık ayarı gibi uygulamalarda kullanılır ve digital sinyallerin analog benzeri bir davranış sergilemesini sağlar.

1. Arduino Uno

- ▶ ##### Analog Input Pins: 6
- ▶ - **Açıklama:** 6 adet analog giriş pini, sensörlerden gelen analog sinyalleri okuyabilir. Bu pinler, 10-bit çözümürlükte ADC (Analog to Digital Converter) kullanır ve 0 ile 1023 arasında bir değer döndürür. Örneğin, bir sıcaklık sensöründen veya bir potansiyometreden gelen voltajı ölçmek için kullanılabilir.

1. Arduino Uno

- ▶ ##### Clock Speed: 16 MHz
- ▶ - **Açıklama:** Arduino Uno, 16 MHz hızında çalışan bir kristal osilatöre sahiptir. Bu hız, mikrodenetleyicinin işlem yapma kapasitesini belirler. 16 MHz, çoğu hobi ve prototip projeleri için yeterli bir hızdır, bu sayede işlemler hızlı ve verimli bir şekilde gerçekleştirilebilir.

1. Arduino Uno

- ▶ ##### Flash Memory: 32 KB
- ▶ - **Açıklama:** Flash bellek, program kodunu depolamak için kullanılır. Arduino Uno'da 32 KB flash bellek bulunur ve bu bellek, kullanıcıların yazdığı programların saklanması sağlar. Ancak, bu belleğin 2 KB'lık kısmı bootloader için ayrılmıştır. Bootloader, mikrodenetleyiciyi programlamak için gerekli olan küçük bir yazılımdır.

1. Arduino Uno

- ▶ ##### SRAM: 2 KB
- ▶ - **Açıklama:** SRAM (Static Random-Access Memory), çalışma sırasında veri depolamak için kullanılır. Değişkenler ve geçici veriler burada saklanır. Arduino Uno'da 2 KB SRAM bulunur, bu da çoğu temel ve orta düzey projeler için yeterlidir.

1. Arduino Uno

- ▶ ##### EEPROM: 1 KB
- ▶ - **Açıklama:** EEPROM (Electrically Erasable Programmable Read-Only Memory), kalıcı veri depolamak için kullanılır. Bu bellek, güç kapatıldığında bile verileri saklar. Arduino Uno'da 1 KB EEPROM bulunur, bu da kullanıcıların cihaz kapandıktan sonra bile kalıcı olarak saklanması gereken verileri depolamalarını sağlar.

1. Arduino Uno

- ▶ ##### Power Supply: USB bağlantısı veya harici adaptör (7-12V)
- ▶ - **Açıklama:** Arduino Uno, iki farklı şekilde güç alabilir. USB bağlantısı aracılığıyla bilgisayardan veya harici bir adaptör (7-12V) ile beslenebilir. USB bağlantısı, aynı zamanda Arduino'yu programlamak için de kullanılır. Harici adaptör ise, bağımsız çalışması gereken projeler için idealdir.

1. Arduino Uno

- ▶ ##### Özellikler: Başlangıç seviyesi için ideal, geniş kullanım alanı
- ▶ - **Açıklama:** Arduino Uno, kolay kullanımını ve geniş destek topluluğu ile özellikle yeni başlayanlar için idealdir. Çok sayıda kütüphane ve örnek proje sayesinde öğrenmeyi ve projeler geliştirmeyi kolaylaştırır. Eğitim, prototipleme, hobi projeleri ve daha fazlası için yaygın olarak kullanılır.
- ▶ Bu özellikler, Arduino Uno'yu çeşitli projeler için çok yönlü ve güçlü bir araç haline getirir. Hem amatörler hem de profesyoneller tarafından tercih edilir ve geniş bir kullanım alanına sahiptir.

1. Arduino Uno

- ▶ Arduino Uno, elektronik projeler geliştirmek için ideal olan popüler bir mikrodenetleyici kartıdır. İşte Arduino Uno'nun özelliklerinin ayrıntılı açıklaması:

1. Arduino Uno

- ▶ ### 1. Arduino Uno
- ▶ - **Microcontroller: ATmega328P**
- ▶ - **Açıklama:** ATmega328P, AVR mikrodenetleyici ailesinden bir çaptır. Arduino Uno'nun merkezinde yer alır ve kart üzerindeki tüm işlevleri yönetir. Bu mikrodenetleyici, düşük güç tüketimi ve yüksek performans sunar.
- ▶
- ▶ - **Digital I/O Pins: 14 (6 PWM çıkışı)**
- ▶ - **Açıklama:** Arduino Uno'da 14 dijital giriş/çıkış pini bulunur. Bu pinler, dijital sensörler, LED'ler, butonlar ve diğer bileşenlerle etkileşim kurmak için kullanılır. Bu pinlerden 6 tanesi, PWM (Pulse Width Modulation) çıkışı sağlayabilir. PWM, motor hız kontrolü veya LED parlaklık ayarı gibi uygulamalarda kullanılır.
- ▶

1. Arduino Uno

- ▶ - **Analog Input Pins: 6**
- ▶ - **Açıklama:** 6 analog giriş pini, sensörlerden gelen analog sinyalleri okuyabilir. Bu pinler, 0 ila 1023 arasında bir değer döndüren 10-bit ADC (Analog to Digital Converter) kullanır. Örneğin, bir potansiyometrenin konumunu ölçmek için analog girişler kullanılır.
- ▶
- ▶ - **Clock Speed: 16 MHz**
- ▶ - **Açıklama:** Arduino Uno, 16 MHz hızında çalışan bir kristal osilatör kullanır. Bu hız, mikrodenetleyicinin işlemleri ne kadar hızlı gerçekleştirebileceğini belirler. 16 MHz, çoğu hobi ve prototip projeleri için yeterlidir.
- ▶

1. Arduino Uno

- ▶ - **Flash Memory: 32 KB**
- ▶ - **Açıklama:** Flash bellek, program kodunu depolamak için kullanılır. Arduino Uno'nun 32 KB flash belleği vardır ve bu, karmaşık projeler için yeterli program alanı sağlar. Ancak, 2 KB'luk bir kısmı bootloader için ayrılmıştır.
- ▶
- ▶ - **SRAM: 2 KB**
- ▶ - **Açıklama:** SRAM (Static Random-Access Memory), çalışma sırasında veri depolamak için kullanılır. Değişkenler ve çalışma zamanında kullanılan veri bu bellekte tutulur. Arduino Uno'da 2 KB SRAM bulunur.
- ▶

1. Arduino Uno

- ▶ - **EEPROM: 1 KB**
- ▶ - **Açıklama:** EEPROM (Electrically Erasable Programmable Read-Only Memory), kalıcı veri depolamak için kullanılır. Bu bellek, güç kapatıldığında bile verileri saklar. Arduino Uno'da 1 KB EEPROM bulunur ve bu, küçük veri parçalarını depolamak için idealdir.
- ▶
- ▶ - **Power Supply: USB bağlantısı veya harici adaptör (7-12V)**
- ▶ - **Açıklama:** Arduino Uno, iki farklı şekilde güç sağlanabilir. USB bağlantısı aracılığıyla bilgisayardan veya harici bir adaptör (7-12V) ile beslenebilir. Bu, projenin ihtiyaçlarına göre esneklik sağlar.
- ▶

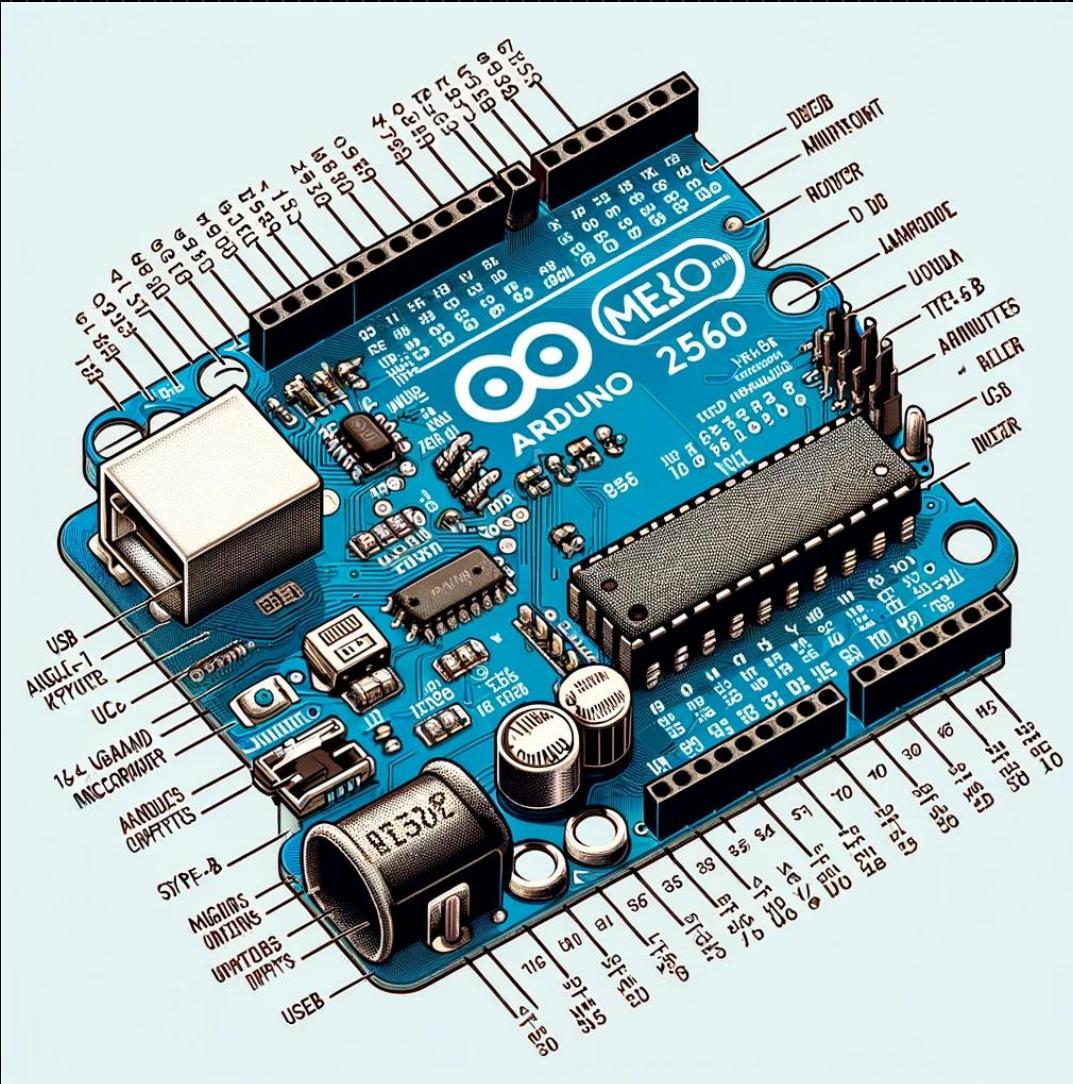
1. Arduino Uno

- ▶ - **Özellikler:** Başlangıç seviyesi için ideal, geniş kullanım alanı**
- ▶ - **Açıklama:** Arduino Uno, kolay kullanım ve geniş destek topluluğu ile başlangıç seviyesi kullanıcılar için idealdir. Çok sayıda kütüphane ve örnek proje mevcuttur, bu da öğrenmeyi ve projeler geliştirmeyi kolaylaştırır. Eğitim, prototipleme ve hobi projeleri için yaygın olarak kullanılır.
- ▶ Arduino Uno, esnekliği, kullanıcı dostu yapısı ve geniş destek topluluğu ile elektronik projeler geliştirmek için mükemmel bir platformdur. Bu özellikler, kullanıcıların geniş bir yelpazede projeler oluşturmasına olanak tanır.

2. Arduino Mega 2560

- ▶ ### 2. Arduino Mega 2560
- ▶ - **Microcontroller:** ATmega2560
- ▶ - **Digital I/O Pins:** 54 (15 PWM çıkışı)
- ▶ - **Analog Input Pins:** 16
- ▶ - **Clock Speed:** 16 MHz
- ▶ - **Flash Memory:** 256 KB
- ▶ - **SRAM:** 8 KB
- ▶ - **EEPROM:** 4 KB
- ▶ - **Power Supply:** USB bağlantısı veya harici adaptör (7-12V)
- ▶ - **Özellikler:** Daha fazla I/O pinine ve daha fazla hafızaya ihtiyaç duyan projeler için uygun.

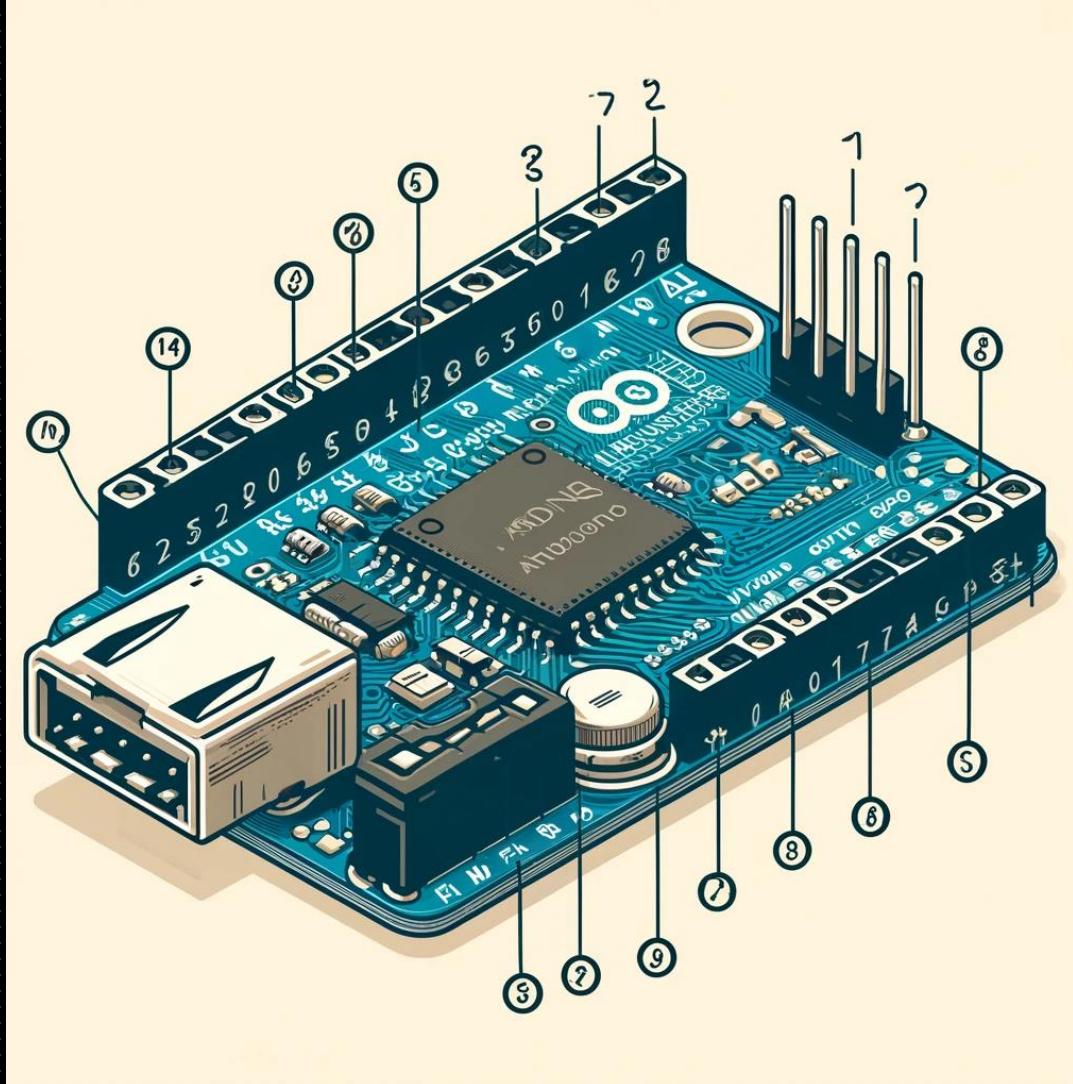
2. Arduino Mega 2560



3. Arduino Nano

- ▶ ### 3. Arduino Nano
- ▶ - **Microcontroller:** ATmega328
- ▶ - **Digital I/O Pins:** 14 (6 PWM çıkışlı)
- ▶ - **Analog Input Pins:** 8
- ▶ - **Clock Speed:** 16 MHz
- ▶ - **Flash Memory:** 32 KB
- ▶ - **SRAM:** 2 KB
- ▶ - **EEPROM:** 1 KB
- ▶ - **Power Supply:** Mini-USB bağlantısı veya harici adaptör (7-12V)
- ▶ - **Özellikler:** Küçük boyutlu projeler için kompakt ve taşınabilir.

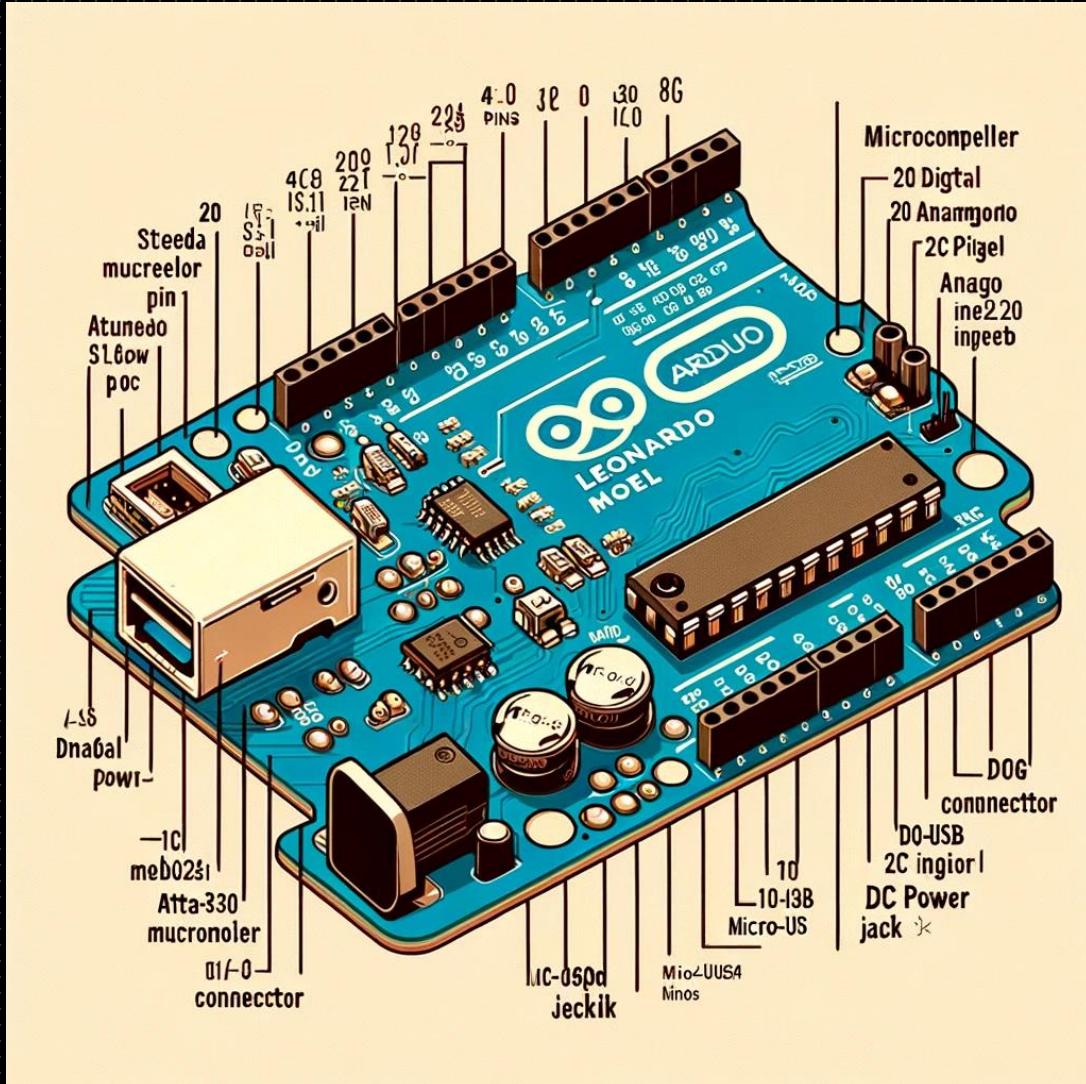
3. Arduino Nano



4. Arduino Leonardo

- ▶ ### 4. Arduino Leonardo
- ▶ - **Microcontroller:** ATmega32u4
- ▶ - **Digital I/O Pins:** 20 (7 PWM çıkışı)
- ▶ - **Analog Input Pins:** 12
- ▶ - **Clock Speed:** 16 MHz
- ▶ - **Flash Memory:** 32 KB
- ▶ - **SRAM:** 2.5 KB
- ▶ - **EEPROM:** 1 KB
- ▶ - **Power Supply:** USB bağlantısı veya harici adaptör (7-12V)
- ▶ - **Özellikler:** USB HID (Human Interface Device) desteği ile klavye veya fare olarak kullanılabilir.

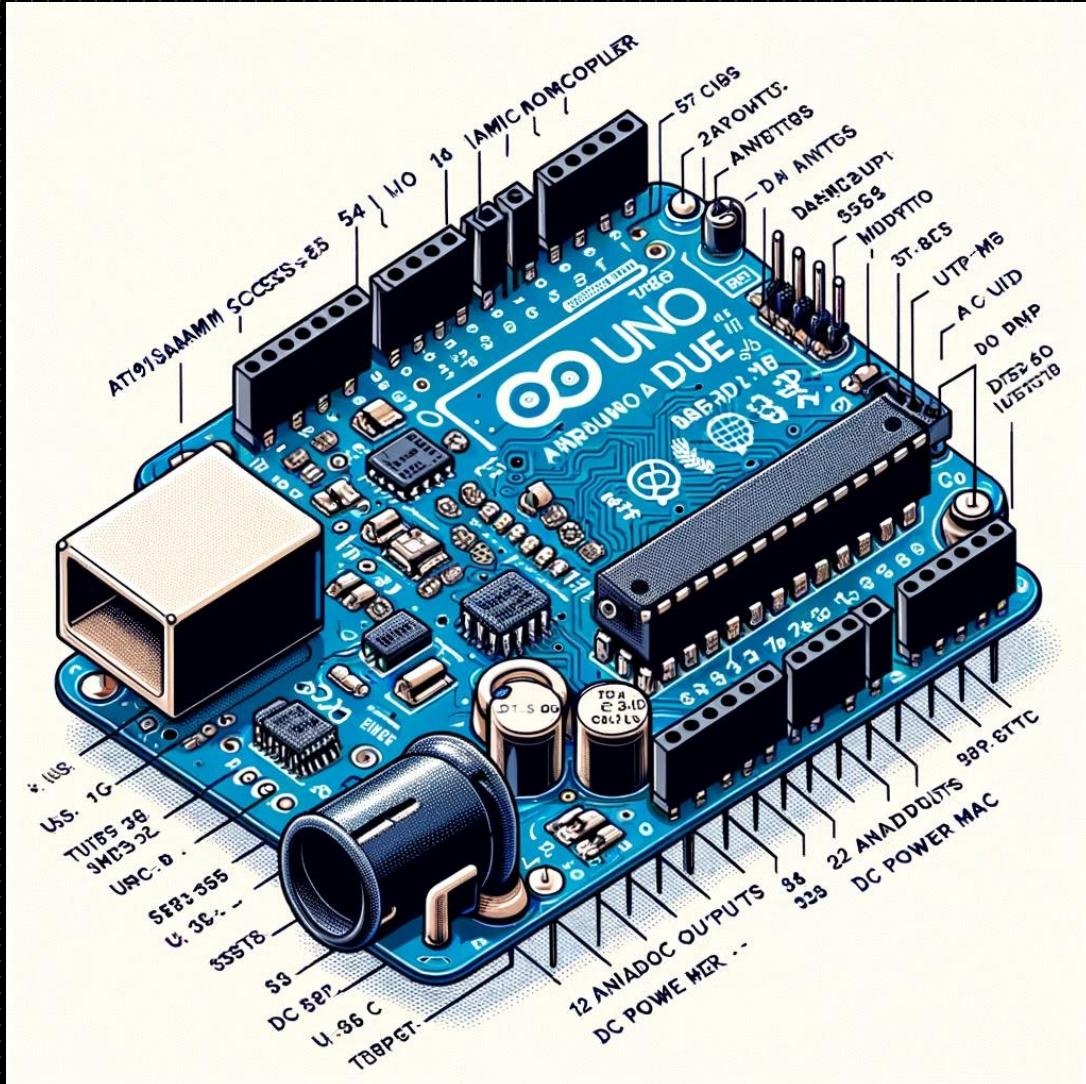
4. Arduino Leonardo



5. Arduino Due

- ▶ ### 5. Arduino Due
- ▶ - **Microcontroller:** AT91SAM3X8E (ARM Cortex-M3)
- ▶ - **Digital I/O Pins:** 54 (12 PWM çıkışı)
- ▶ - **Analog Input Pins:** 12
- ▶ - **Analog Output Pins:** 2 (DAC)
- ▶ - **Clock Speed:** 84 MHz
- ▶ - **Flash Memory:** 512 KB
- ▶ - **SRAM:** 96 KB
- ▶ - **EEPROM:** Yok
- ▶ - **Power Supply:** USB bağlantısı veya harici adaptör (7-12V)
- ▶ - **Özellikler:** Daha hızlı işlemci ve daha fazla bellek gerektiren projeler için uygun.

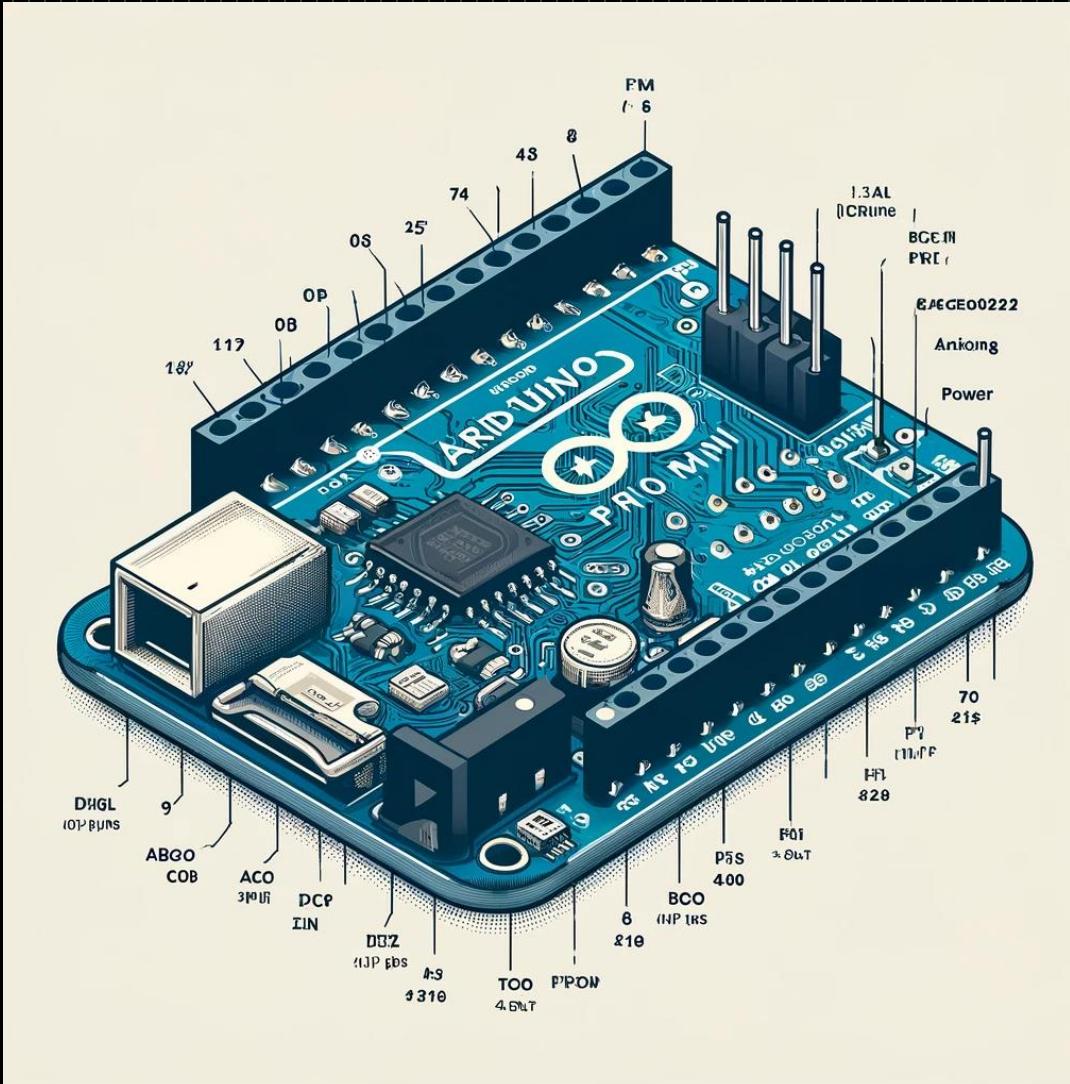
5. Arduino Due



6. Arduino Pro Mini

- ▶ ### 6. Arduino Pro Mini
- ▶ - **Microcontroller:** ATmega328
- ▶ - **Digital I/O Pins:** 14 (6 PWM çıkışlı)
- ▶ - **Analog Input Pins:** 8
- ▶ - **Clock Speed:** 16 MHz (5V modeli) veya 8 MHz (3.3V modeli)
- ▶ - **Flash Memory:** 32 KB
- ▶ - **SRAM:** 2 KB
- ▶ - **EEPROM:** 1 KB
- ▶ - **Power Supply:** 3.3V veya 5V
- ▶ - **Özellikler:** Minimum boyut ve düşük güç tüketimi gerektiren projeler için ideal.

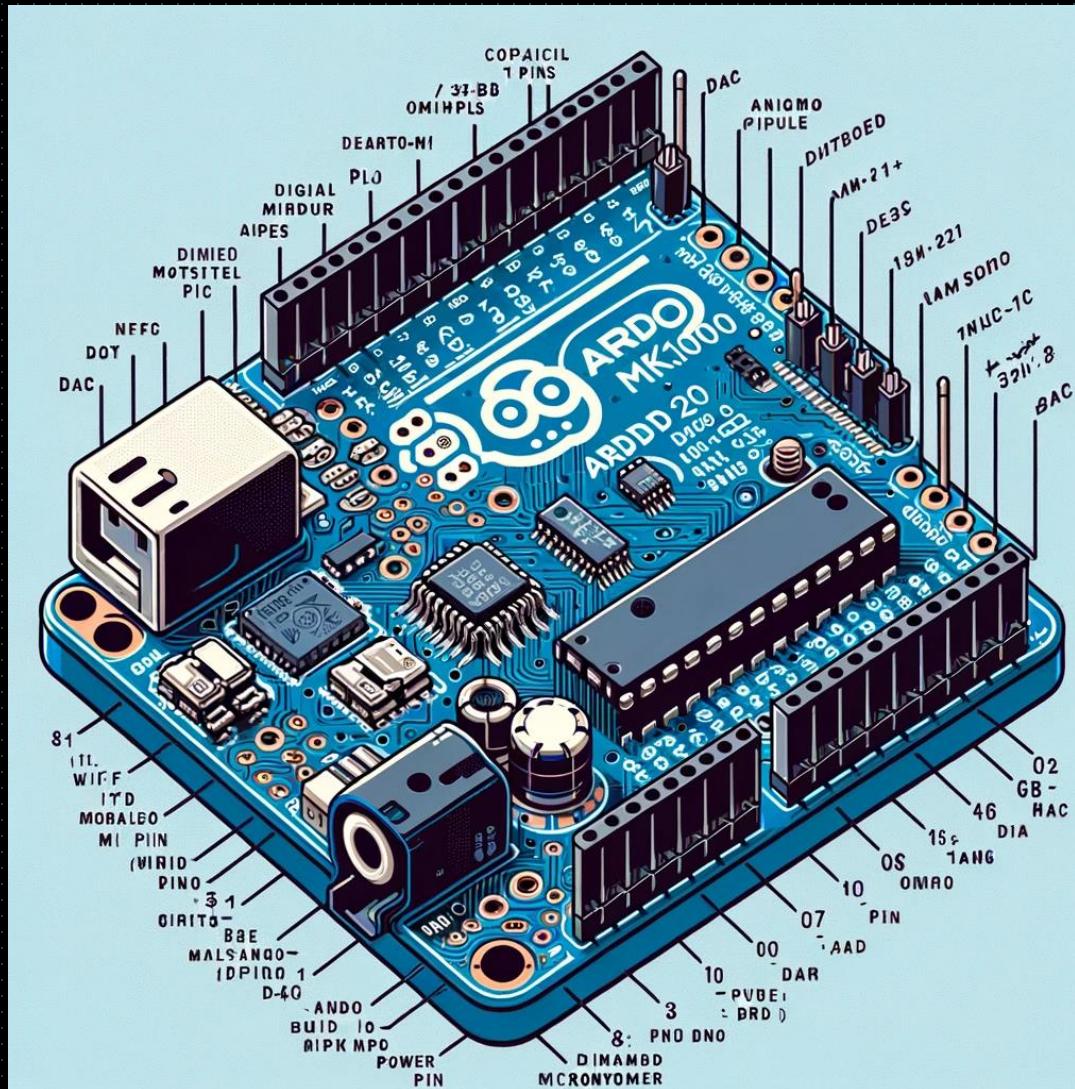
6. Arduino Pro Mini



7. Arduino MKR1000

- ▶ **### 7. Arduino MKR1000**
- ▶ - **Microcontroller:** SAMD21 Cortex-M0+ 32 bit ARM
- ▶ - **WiFi Module:** WINC1500
- ▶ - **Digital I/O Pins:** 8 (12 PWM çıkışı)
- ▶ - **Analog Input Pins:** 7
- ▶ - **Analog Output Pins:** 1 (DAC)
- ▶ - **Clock Speed:** 48 MHz
- ▶ - **Flash Memory:** 256 KB
- ▶ - **SRAM:** 32 KB
- ▶ - **EEPROM:** Yok
- ▶ - **Power Supply:** Li-Po pil veya USB
- ▶ - **Özellikler:** IoT (Internet of Things) projeleri için WiFi bağlantısı ve düşük güç tüketimi.

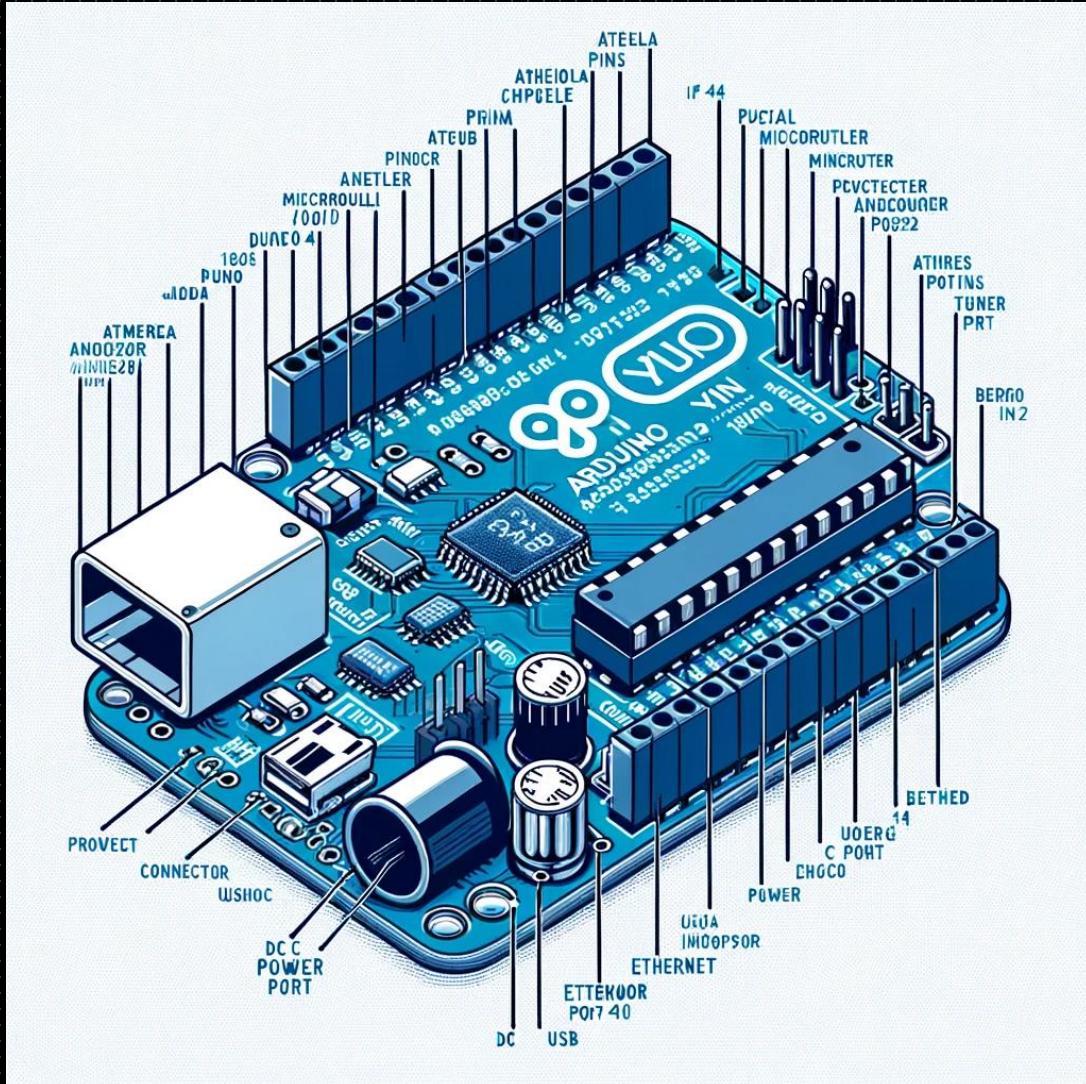
7. Arduino MKR1000



8. Arduino Yun

- ▶ **### 8. Arduino Yun**
- ▶ - **Microcontroller:** ATmega32u4 + Atheros AR9331
- ▶ - **Digital I/O Pins:** 20
- ▶ - **Analog Input Pins:** 12
- ▶ - **Clock Speed:** 16 MHz (ATmega) / 400 MHz (AR9331)
- ▶ - **Flash Memory:** 32 KB (ATmega) / 16 MB (AR9331)
- ▶ - **SRAM:** 2.5 KB (ATmega) / 64 MB DDR2 (AR9331)
- ▶ - **EEPROM:** 1 KB
- ▶ - **Power Supply:** USB veya harici adaptör (5V)
- ▶ - **Özellikler:** Linux tabanlı işletim sistemi ve WiFi desteği ile ağ bağlantılı projeler için uygundur.

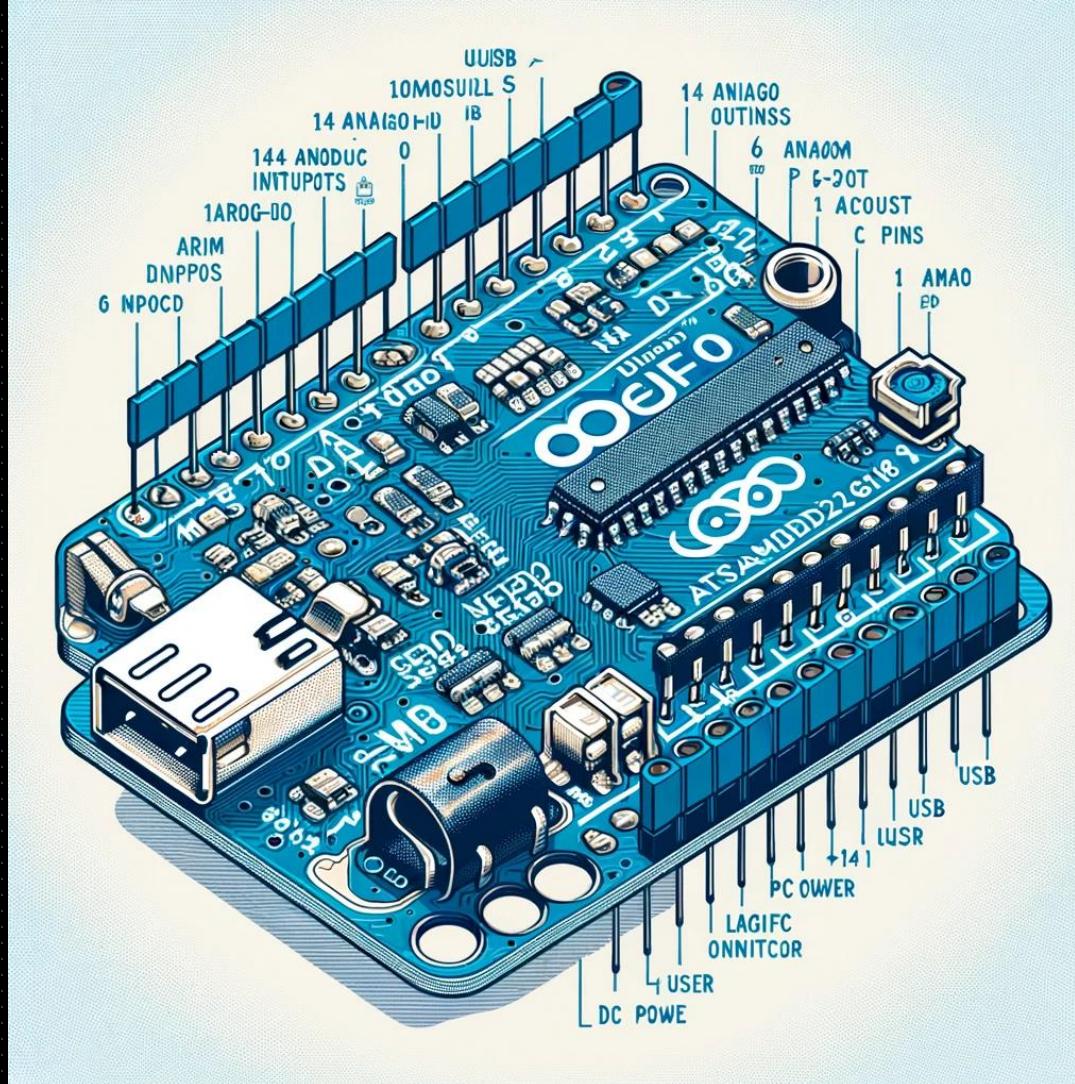
8. Arduino Yun



9. Arduino Zero

- ▶ ### 9. Arduino Zero
- ▶ - **Microcontroller:** ATSAMD21G18 (ARM Cortex-M0+)
- ▶ - **Digital I/O Pins:** 14 (12 PWM çıkışı)
- ▶ - **Analog Input Pins:** 6
- ▶ - **Analog Output Pins:** 1 (DAC)
- ▶ - **Clock Speed:** 48 MHz
- ▶ - **Flash Memory:** 256 KB
- ▶ - **SRAM:** 32 KB
- ▶ - **EEPROM:** Yok
- ▶ - **Power Supply:** USB bağlantısı veya harici adaptör (7-12V)
- ▶ - **Özellikler:** Daha gelişmiş projeler için 32-bit ARM Cortex-M0+ işlemci.

9. Arduino Zero



Arduino Modelleri

- ▶ Bu modeller, farklı ihtiyaçlara ve projelere hitap eden çeşitli özellikler sunar. Arduino kartlarının geniş yelpazesi, hem basit projeler hem de daha karmaşık sistemler geliştirmek için uygundur.

Fritzing

- ▶ Fritzing, elektronik devreler tasarlamak ve prototip oluşturmak için kullanılan açık kaynaklı bir yazılım platformudur. Fritzing, özellikle elektronik projelerle uğraşan hobici ve eğitimciler için tasarlanmıştır. İşte Fritzing hakkında daha detaylı bilgi:



Fritzing Temel Özellikler

- ▶ ### Temel Özellikler
- ▶ 1. **Kullanıcı Dostu Arayüz**: Fritzing, kolay kullanılabilir bir grafik arayüz sunar. Bu sayede kullanıcılar devrelerini hızlıca tasarlatabilir ve görselleştirebilir.
- ▶ 2. **Breadboard Görünümü**: Fritzing, breadboard üzerinde devre kurmayı simüle eder. Bu, kullanıcıların fiziksel olarak kurdukları devreleri yazılım ortamında da oluşturmasına olanak tanır.

Fritzing Temel Özellikler

- ▶ 3. **Şema Görünümü**: Devrelerin daha teknik ve ayrıntılı bir şematik diyagramını oluşturmak için kullanılır. Bu görünüm, devrelerin anlaşılmasını ve analiz edilmesini kolaylaştırır.
- ▶ 4. **PCB Tasarımı**: Fritzing, devrelerin basılı devre kartı (PCB) tasarımını da destekler. Kullanıcılar, devrelerini PCB formatında tasarlayabilir ve üretim için gerekli dosyaları oluşturabilirler.

Fritzing Temel Özellikler

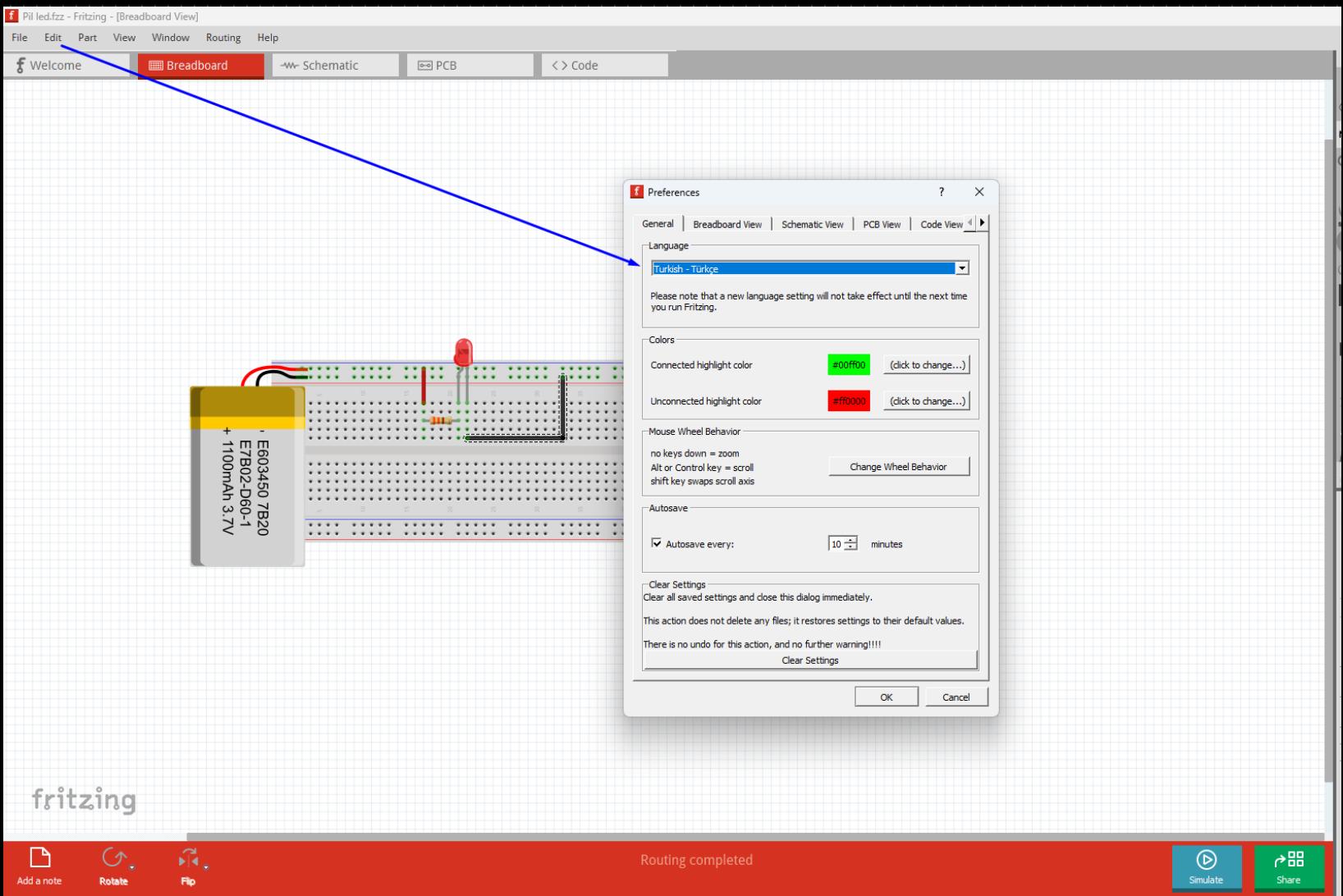
- ▶ 5. ****Parça Kütüphanesi****: Fritzing, geniş bir elektronik bileşen kütüphanesine sahiptir. Kullanıcılar, çeşitli elektronik bileşenleri kolayca projelerine ekleyebilirler.
- ▶ 6. ****Topluluk ve Paylaşım****: Fritzing, kullanıcıların projelerini paylaşabileceği bir topluluk platformuna sahiptir. Bu, diğer kullanıcıların projelerinden ilham almayı ve kendi projelerini paylaşmayı teşvik eder.

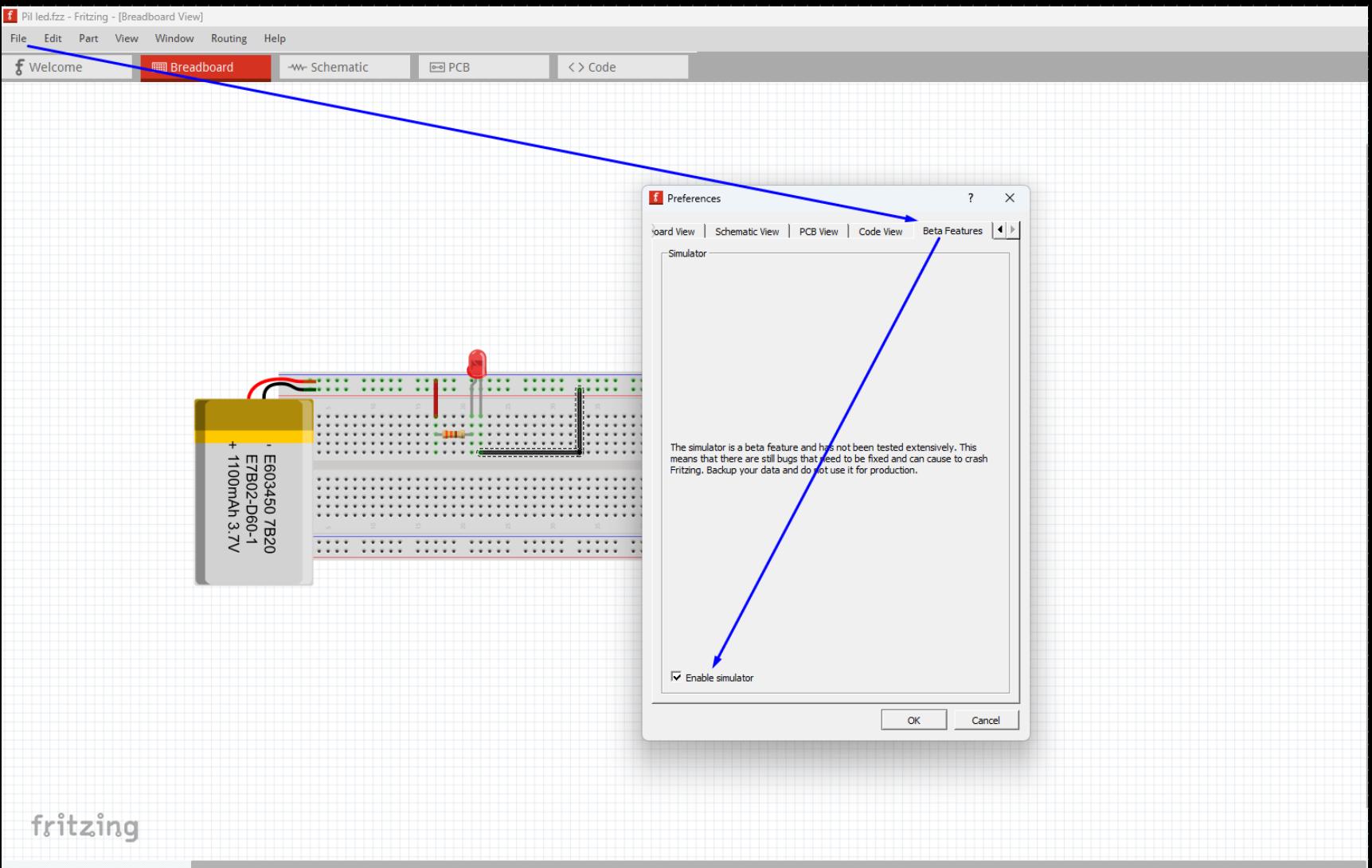
Fritzing Kullanım Alanları

- ▶ **### Kullanım Alanları**
- ▶ - **Eğitim**: Fritzing, okullarda ve üniversitelerde elektronik eğitimi için yaygın olarak kullanılır. Öğrenciler, devreleri sanal ortamda tasarlayarak pratik yapabilirler.
- ▶ - **Prototipleme**: Hobi projeleri ve küçük ölçekli profesyonel projeler için prototip oluşturma sürecinde kullanılır.
- ▶ - **Dökümantasyon**: Fritzing, projelerin dökümantasyonunu yapmak ve paylaşmak için etkili bir araçtır.

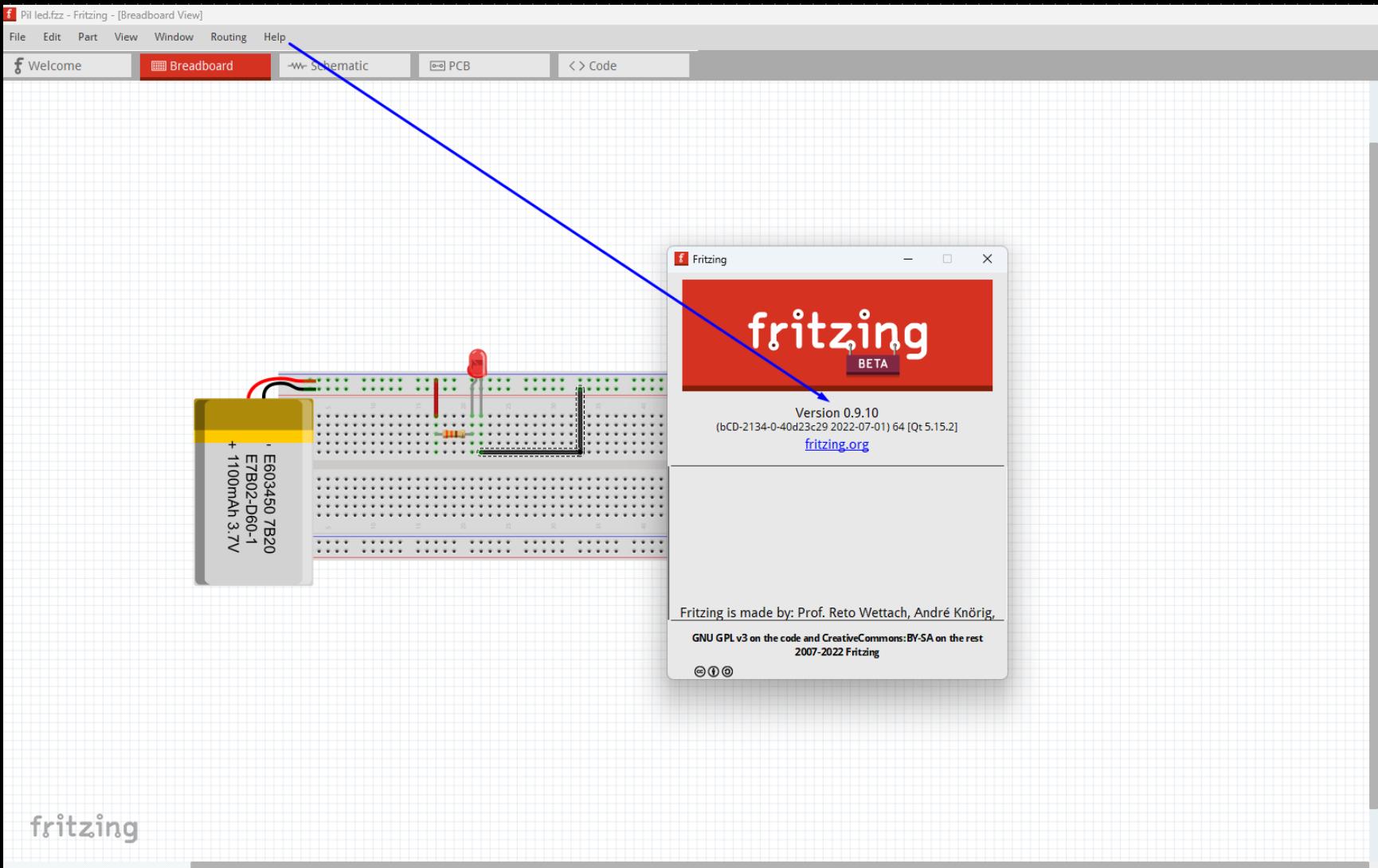
Fritzing Kullanım Alanları

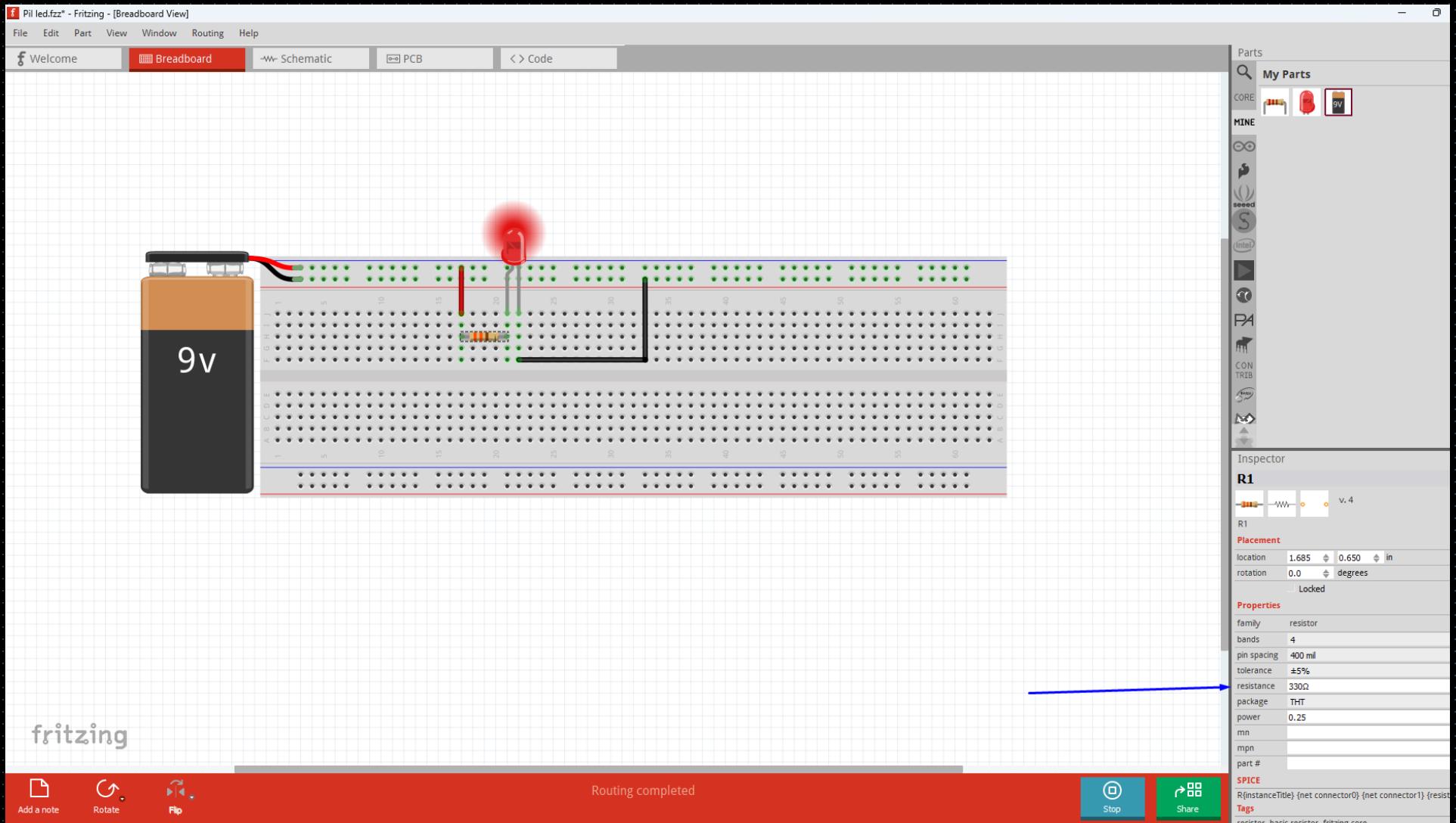
- ▶ Fritzing'in sağladığı bu özellikler, elektronik projelerle uğraşan herkes için güçlü ve erişilebilir bir araç olmasını sağlar. Eğer elektronikle ilgileniyorsanız, Fritzing'i denemek projelerinizi daha kolay ve eğlenceli hale getirebilir.

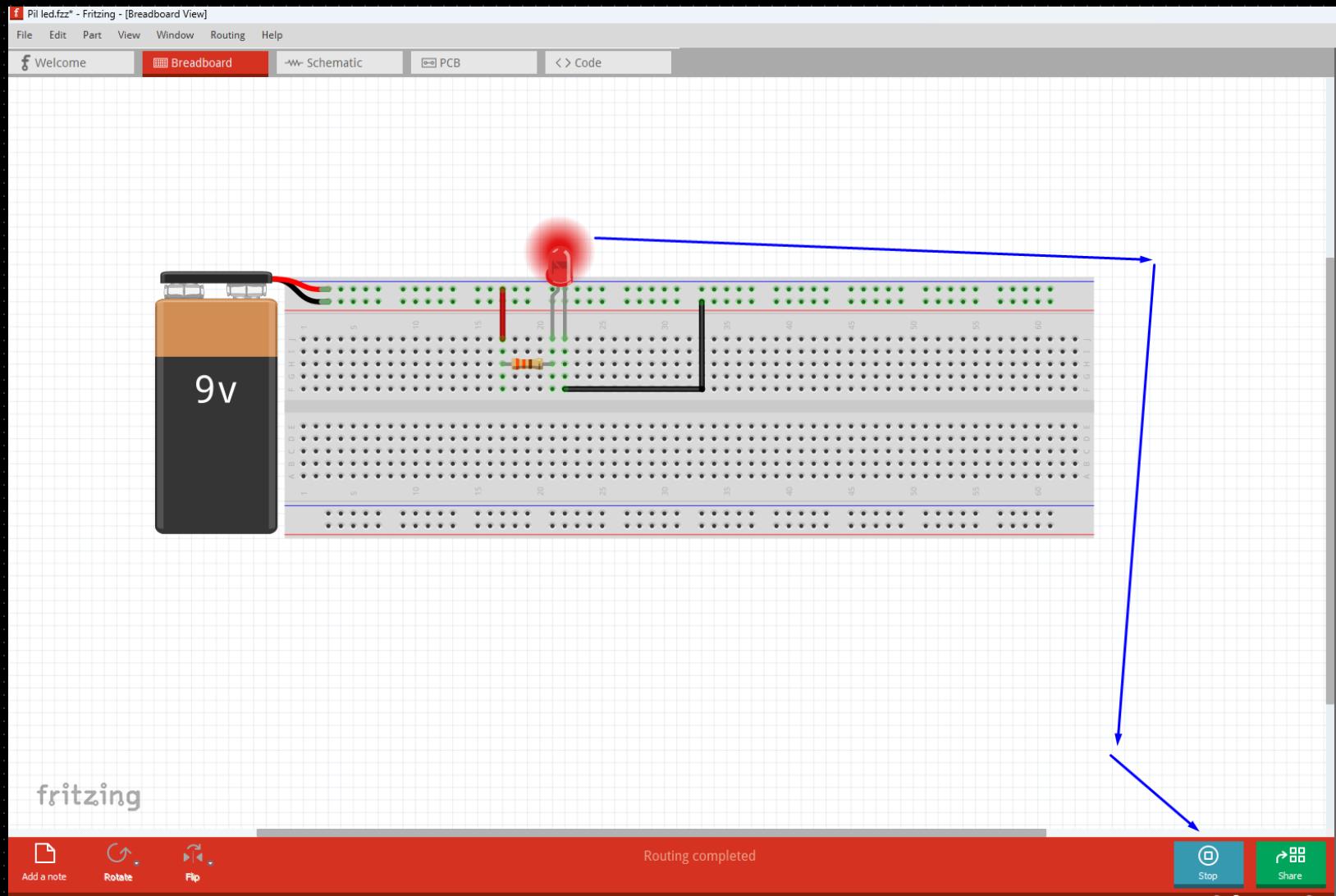


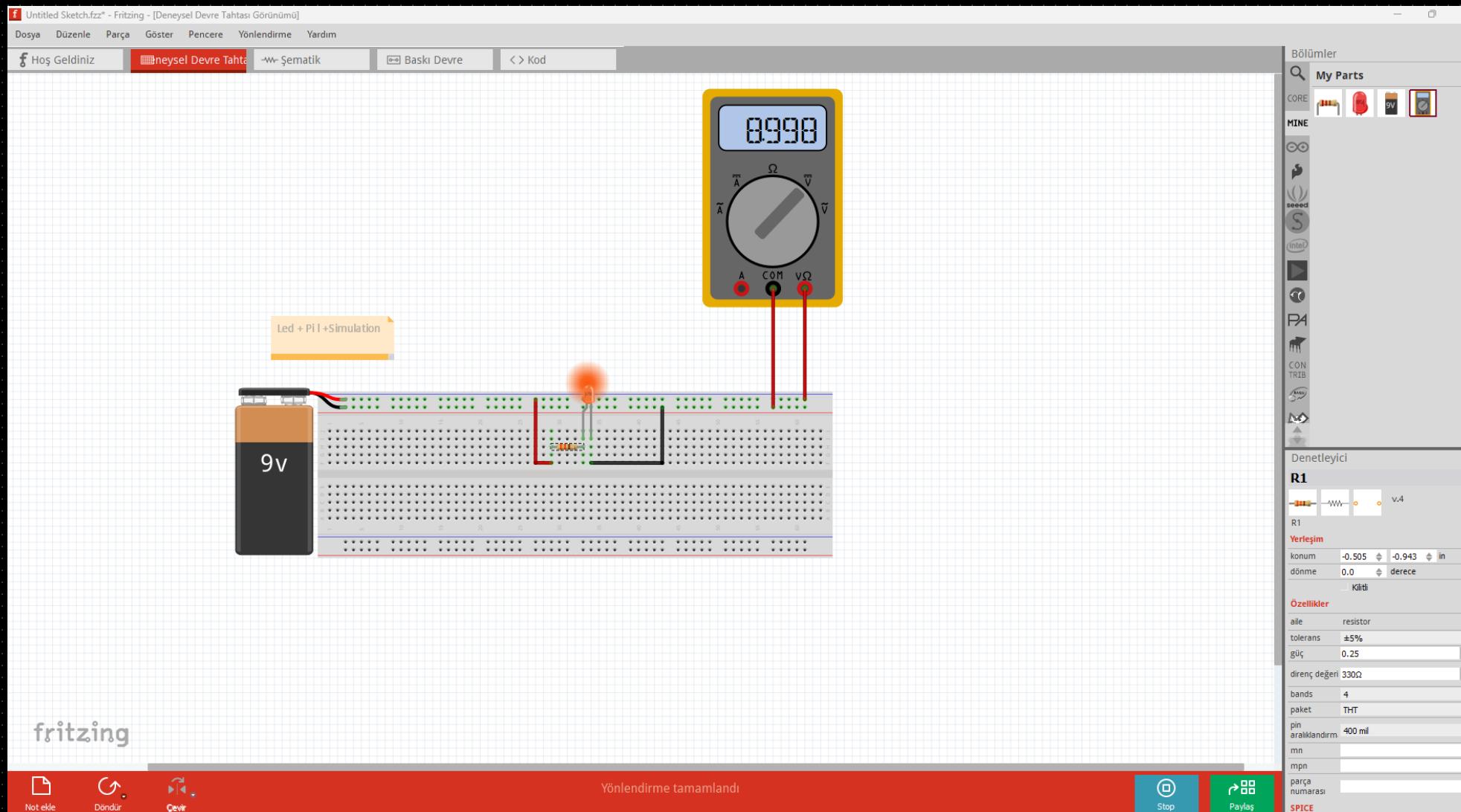


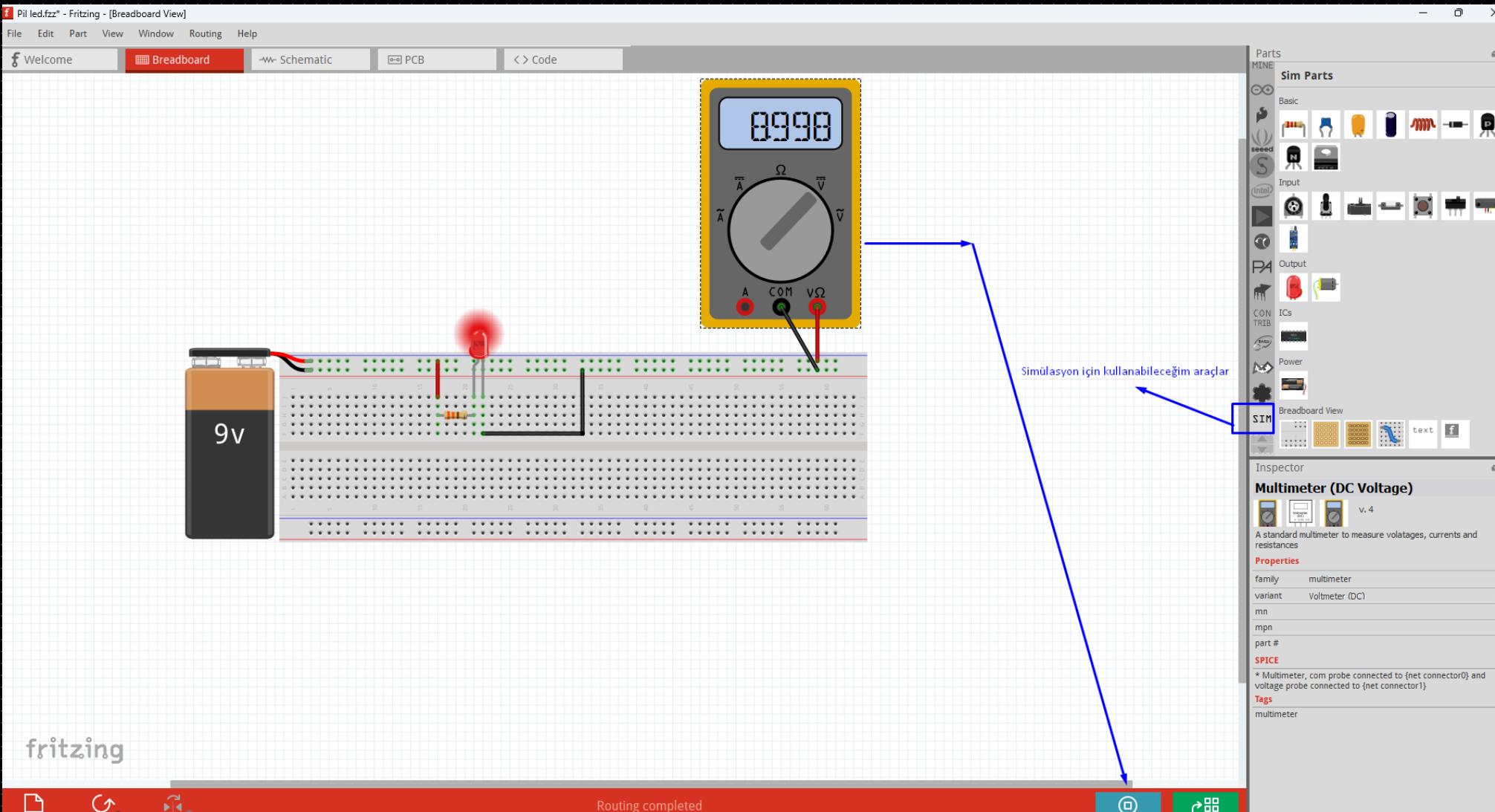
fritzing

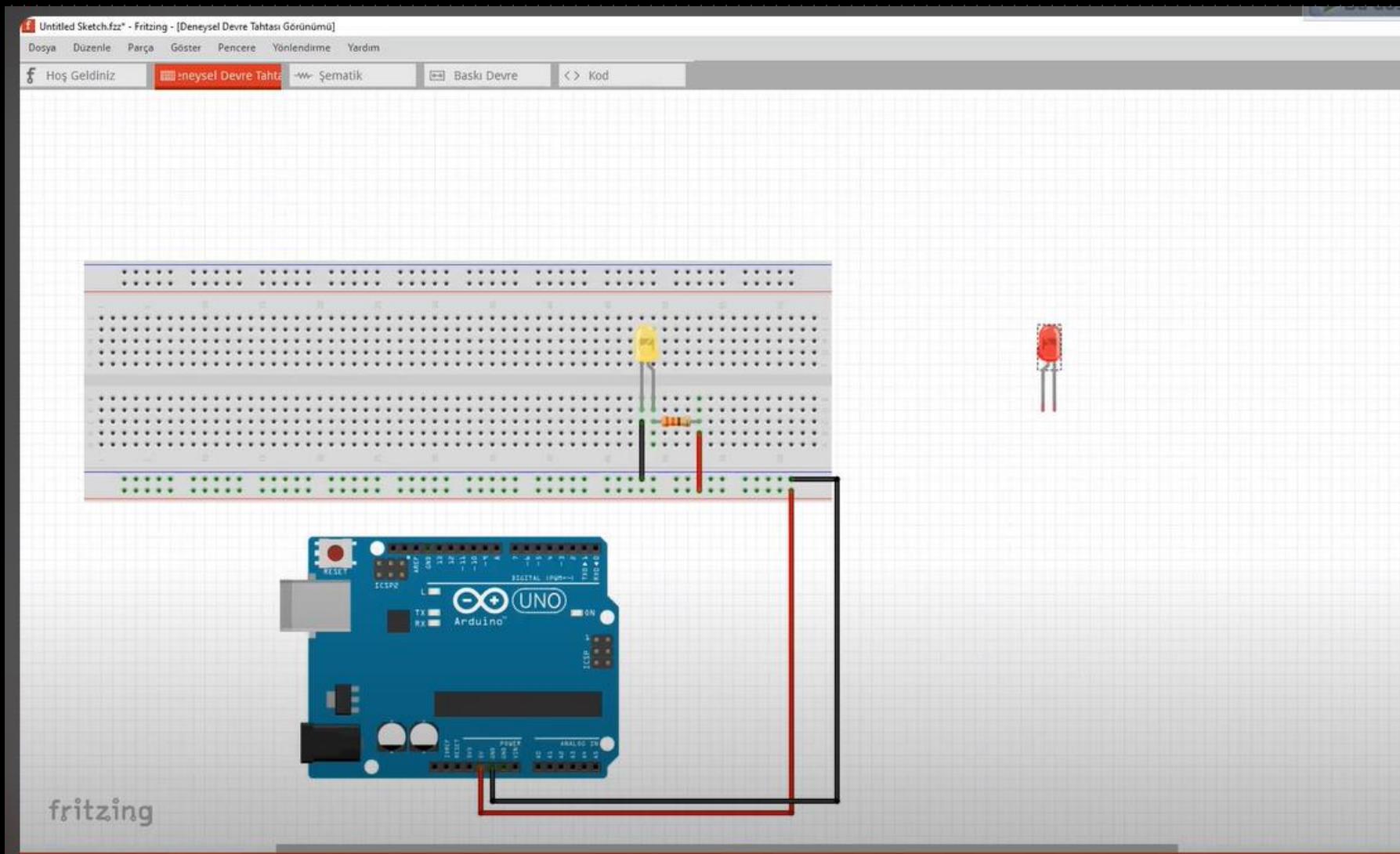




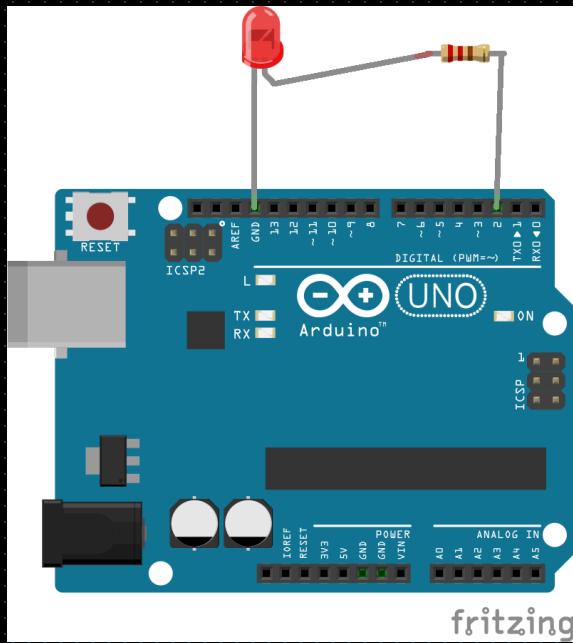








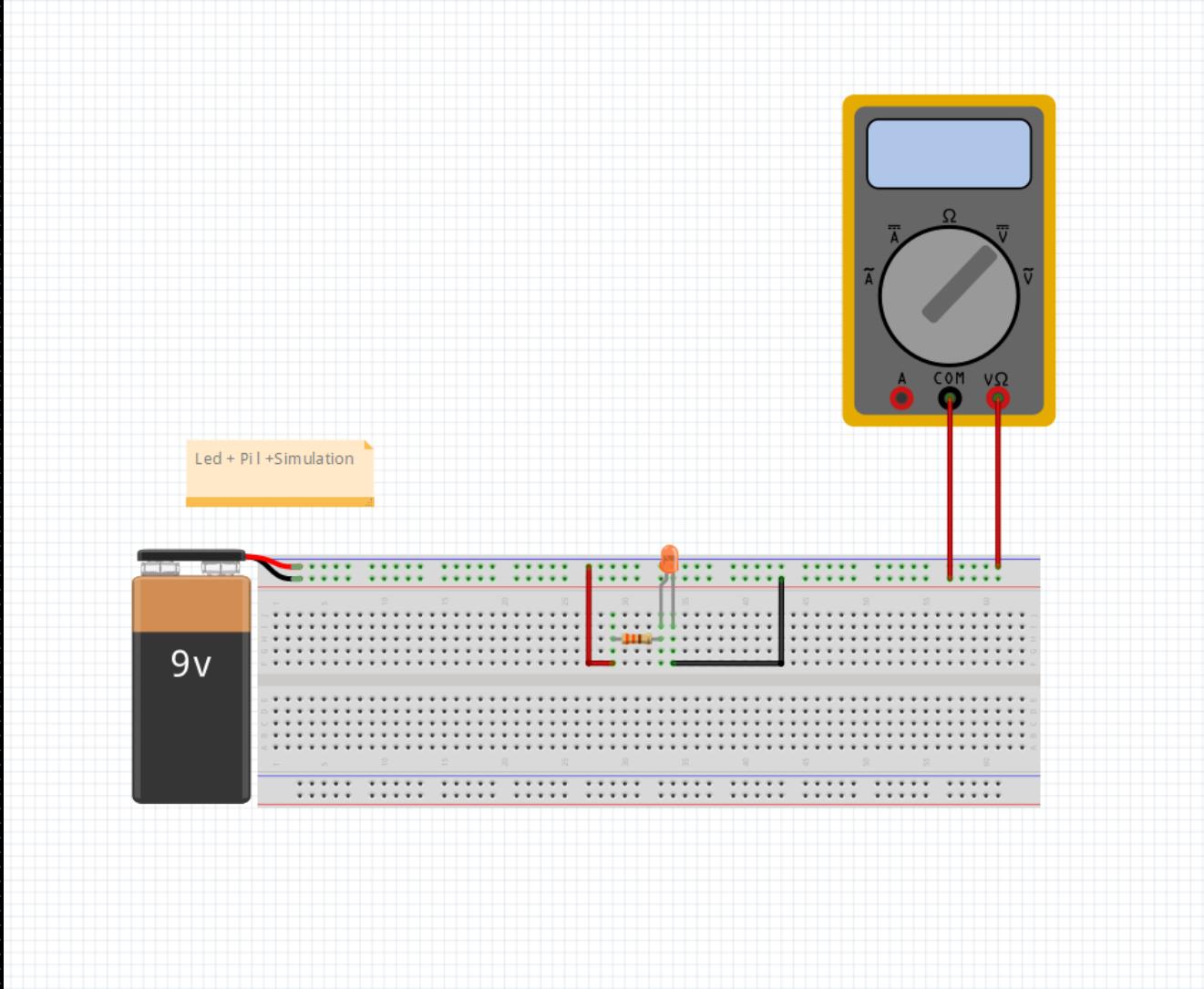
Simülasyon Arduino Led



Simülasyon Arduino Led

```
► void setup(){  
  
► // Digital pin seçmeliyim  
► pinMode(2,OUTPUT);  
► }  
  
► void loop(){  
► digitalWrite(2,HIGH);//lamba yan  
► delay(2000);  
► digitalWrite(2,LOW);// Lamba son  
► delay(1000);  
► }
```

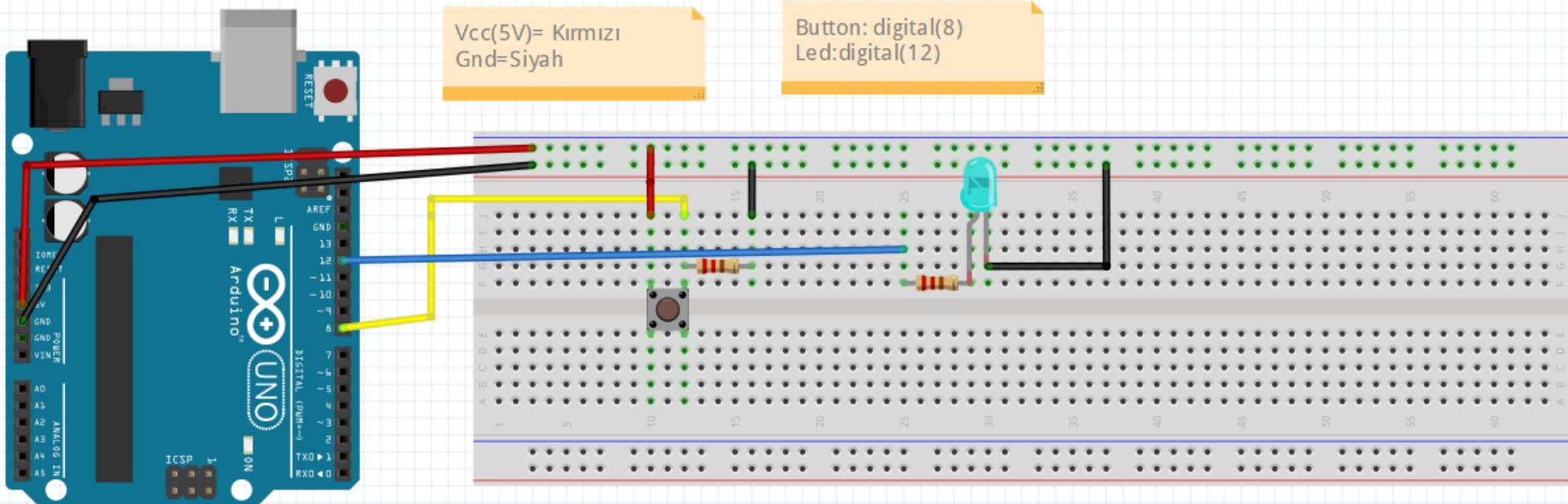
Simülasyon Arduino Led



Simülasyon Arduino Led

- ▶ void setup() {
- ▶ // put your setup code here, to run once:
- ▶ /*multiple*/
- ▶
- ▶
- ▶ // Digital 13 Pini ♦?k?? yapmak istiyorum
- ▶ pinMode(13, OUTPUT); // ; unutma
- ▶ }
- ▶

- ▶ void loop() {
- ▶ // put your main code here, to run repeatedly:
- ▶ digitalWrite(13, HIGH); // Lambay? yak
- ▶ delay(2000); // 2.saniye bekle
- ▶ digitalWrite(13, LOW); // Lambay? s♦nd♦r
- ▶ delay(1000); // 1. saniye bekle(uyu)
- ▶ }



fritzing

```
Untitled2.ino ×

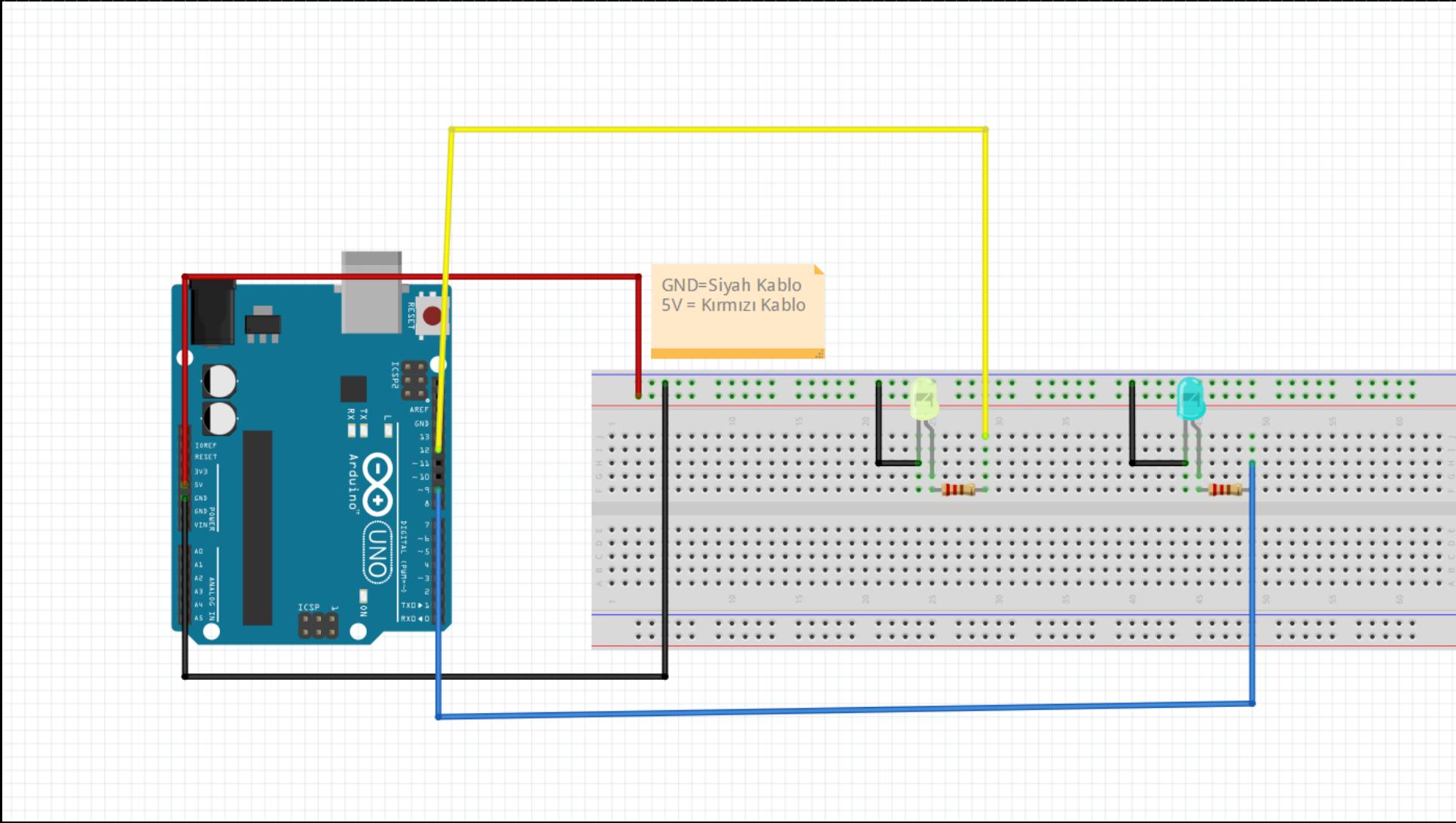
#define Button 8
#define Led 12

void setup(){
// Led
// pinMode(12,OUTPUT);
pinMode(Led,OUTPUT)

// Button
// pinMode(8,INPUT);
pinMode(Button,INPUT)

}

void loop(){
if(digitalRead(Button) == 1){
  delay(500);
  digitalWrite(Led,HIGH);
}else{
  digitalWrite(Led,LOW);
}
}
```

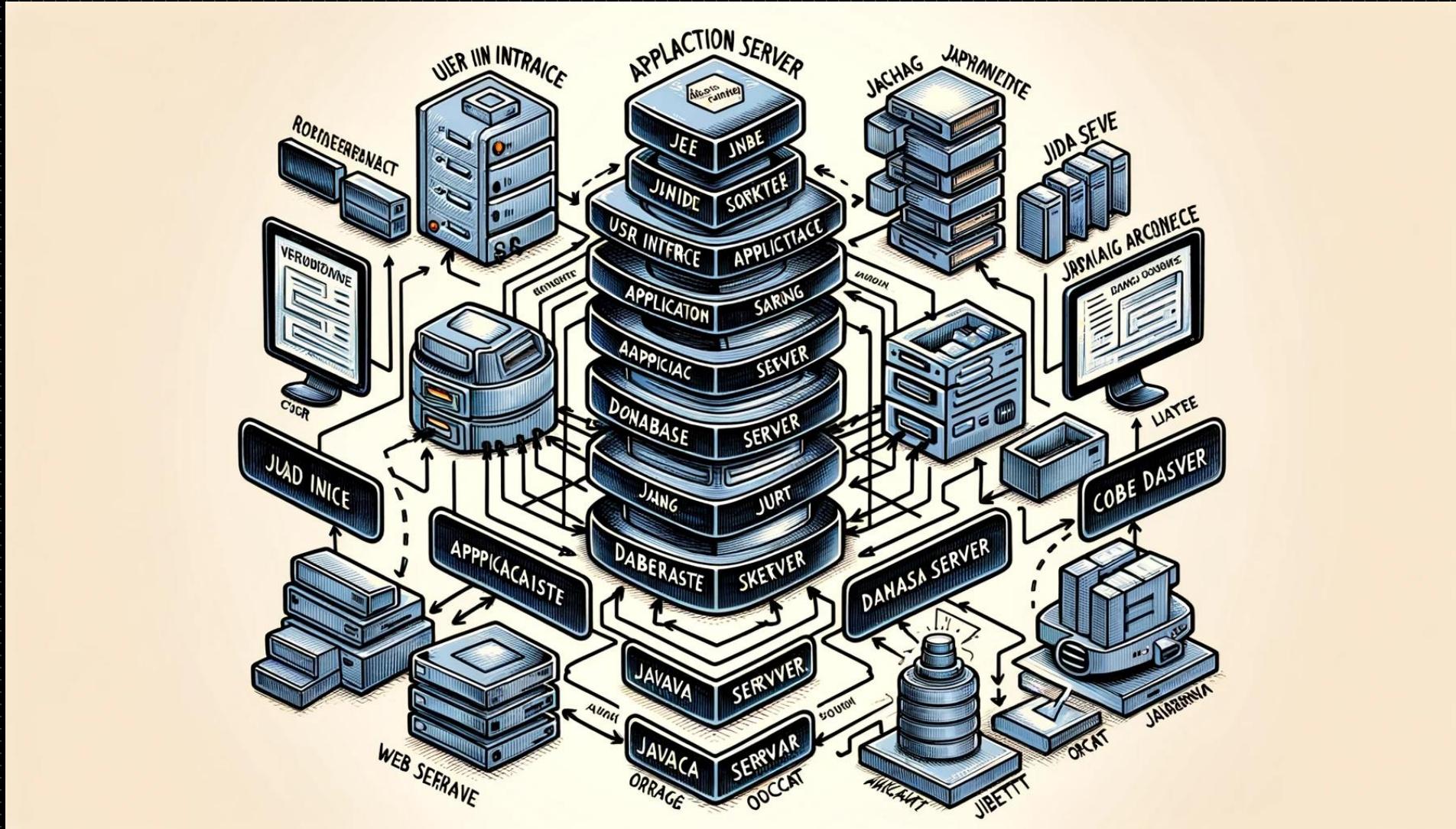


```
▶ void setup(){  
▶  
▶    // Sar? LED  
▶    pinMode(12,OUTPUT);  
▶  
▶    // Mavi LED  
▶    pinMode(9,OUTPUT);  
▶    }  
▶
```

```
▶ void loop(){
▶   // Mavi yans?n
▶   digitalWrite(9,HIGH);
▶   delay(500);
▶ 
▶   // Sar? Yans?n
▶   digitalWrite(12,HIGH);
▶   delay(1000);
▶ 
▶   // Mavi Sons?n
▶   digitalWrite(9,LOW);
▶   delay(1000);
▶ 
▶   // Sar? Sons?n
▶   digitalWrite(12,LOW);
▶   delay(1000);
▶ 
▶ }
```

- ▶ [Arduino Simulator \(lrusso.github.io\)](https://lrusso.github.io)

Java İle IOT



Hava Kalitesi Mysql , Java

- ▶ Mysql
- ▶ Java

Hava Kalitesi Mysql , Java

Bir hava kalitesi sensöründen veri okuyup bu verileri MySQL veritabanına kaydeden bir Java uygulaması örneği oluşturalım. Bu uygulamada, JDBC kullanarak MySQL veritabanına bağlanacak ve verileri kaydedeceğiz.

Gerekli Kütüphaneler

MySQL JDBC sürücüsünü kullanacağımız. Maven kullanıyorsanız, `pom.xml` dosyanıza şu bağımlılığı eklemeniz gerekecek:

```
```xml
<dependency>
 <groupId>mysql</groupId>
 <artifactId>mysql-connector-java</artifactId>
 <version>8.0.23</version>
</dependency>
```

```

Hava Kalitesi Mysql , Java

```
```sql
CREATE SCHEMA `iot_db` DEFAULT CHARACTER SET utf8 ;

use iot_db;

CREATE TABLE air_quality_data (
 id INT AUTO_INCREMENT PRIMARY KEY,
 air_quality INT NOT NULL,
 timestamp TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

SELECT * FROM iot.air_quality_data;
```

```

Hava Kalitesi Mysql , Java

Database Create

```
CREATE SCHEMA `iot_db` DEFAULT CHARACTER SET utf8 ;  
  
use iot_db;  
  
CREATE TABLE air_quality_data (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    air_quality INT NOT NULL,  
    timestamp TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

Java Class

```
package iot;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.util.Random;

public class IoTExample {

    private static final String DB_URL = "jdbc:mysql://localhost:3306/iot_db";
    private static final String USER = "root";
    private static final String PASS = "root";
```

```
public static void main(String[] args) {
    try (Connection conn = DriverManager.getConnection(DB_URL, USER, PASS)) {

        while (true) {
            // Sahte ışık seviyesi verisi üretimi
            Random rand = new Random();
            int lightLevel = rand.nextInt(1000); // 0 ile 1000 arasında rastgele ışık seviyesi

            // Veritabanına veri ekleme
            String insertSQL = "INSERT INTO air_quality_data (air_quality) VALUES (?)";
            try (PreparedStatement pstmt = conn.prepareStatement(insertSQL)) {
                pstmt.setInt(1, lightLevel);
                pstmt.executeUpdate();
                System.out.println("Inserted light level: " + lightLevel);
            } catch (SQLException e) {
                e.printStackTrace();
            }

            // 5 saniye bekleme
            try {
                Thread.sleep(5000);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
    }
}
```

Java İle IoT

Java ile bir IoT uygulaması geliştirmek, genellikle sensör verilerini toplamak, analiz etmek ve bu verileri bir merkezi sunucuya veya bulut platformuna göndermek anlamına gelir.

Aşağıda, basit bir IoT uygulaması örneği için bir Java kodu bulabilirsiniz.

Bu örnek, bir sıcaklık sensöründen verileri okur ve bu verileri bir MQTT sunucusuna (Message Queuing Telemetry Transport) gönderir.

Gerekli Kütüphaneler

Öncelikle, gerekli kütüphaneleri eklemeniz gerekecek. MQTT için Paho Java Client kütüphanesini kullanacağız. Maven kullanıyorsanız, `pom.xml` dosyanıza şu bağımlılığı ekleyin:

```
```xml
<dependency>
 <groupId>org.eclipse.paho</groupId>
 <artifactId>org.eclipse.paho.client.mqttv3</artifactId>
 <version>1.2.5</version>
</dependency>
````
```

Java İle IOT

Sensör Verilerini Okuma ve Gönderme

Aşağıda, sahte bir sıcaklık sensöründen verileri okuyup bu verileri bir MQTT broker'ına gönderen basit bir Java programı bulunmaktadır.

```
```java
import org.eclipse.paho.client.mqttv3.MqttClient;
import org.eclipse.paho.client.mqttv3.MqttException;
import org.eclipse.paho.client.mqttv3.MqttMessage;
import org.eclipse.paho.client.mqttv3.persist.MemoryPersistence;

import java.util.Random;

public class IoTExample {
 public static void main(String[] args) {
 String broker = "tcp://mqtt.eclipse.org:1883";
 // Enum oluştur
 String clientId = "JavaIoTClient";
 MemoryPersistence persistence = new MemoryPersistence();

 try {
 MqttClient sampleClient = new MqttClient(broker, clientId, persistence);
 sampleClient.connect();
 }
 }
}
```

# Java İle IOT

```
while (true) {
 // Sahte sıcaklık verisi üret
 Random rand = new Random();
 int temperature = rand.nextInt(100);

 // MQTT mesajı oluştur
 String content = "Temperature: " + temperature + "°C";
 MqttMessage message = new MqttMessage(content.getBytes());
 message.setQos(2);

 // Mesajı yayına
 sampleClient.publish("iot/sensors/temperature", message);

 System.out.println("Message published: " + content);

 // 5 saniye bekle
 Thread.sleep(5000);
}
}
}
``
```

### ### Açıklamalar

1. **Bağlantı Bilgileri**: MQTT broker'a bağlanmak için `tcp://mqtt.eclipse.org:1883` adresini kullandık. Bu adres, Eclipse'in herkese açık MQTT broker'ıdır.
2. **Sıcaklık Verisi Üretilimi**: 'Random' sınıfını kullanarak 0 ile 100 arasında rastgele bir sıcaklık değeri ürettik.
3. **MQTT Mesajı Yayınlama**: Üretilen sıcaklık verisini bir MQTT mesajı olarak "iot/sensors/temperature" konusuna yayınladık.
4. **Sonsuz Döngü**: Bu işlem sonsuz bir döngüde her 5 saniyede bir tekrarlanır.

Bu örnek, temel bir IoT uygulaması yapısı sunmaktadır. Gerçek dünyada, sıcaklık verilerini gerçek bir sensörden okuyabilir ve daha güvenilir bir MQTT broker kullanabilirsiniz. Ek olarak, hata yönetimi ve güvenlik konularını da dikkate almanız gerekecektir.

Tabii, Java ile IoT uygulamaları geliştirmek için farklı senaryolar oluşturabiliriz. Bu örnekte, bir nem sensöründen veri okuyarak bu veriyi bir RESTful web servisine gönderen bir uygulama oluşturacağız. Bu uygulama, HTTP POST istekleri alacak ve verileri kaydedecek.

### ### Gerekli Kütüphaneler

HTTP isteklerini yapmak için Apache HttpClient kütüphanesini kullanacağız. Maven kullanıyorsanız, `pom.xml` dosyanızda şu bağımlılıkları eklemeniz gerekecek:

```
```xml
<dependency>
    <groupId>org.apache.httpcomponents</groupId>
    <artifactId>httpclient</artifactId>
    <version>4.5.13</version>
</dependency>
```

```

### ### IoT Uygulaması Örneği

Aşağıda, sahte bir nem sensöründen verileri okuyup bu verileri bir RESTful web servisine gönderen basit bir Java programı bulunmaktadır.

```
```java
import org.apache.http.client.methods.CloseableHttpResponse;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.entity.StringEntity;
import org.apache.http.impl.client.CloseableHttpClient;
import org.apache.http.impl.client.HttpClients;
import org.apache.http.util.EntityUtils;

import java.io.IOException;
import java.util.Random;

public class IoTExample {
```

```
public static void main(String[] args) {
    String serverUrl = "http://example.com/iot/sensors/humidity"; // Sunucu URL'si

    while (true) {
        // Sahte nem verisi üretimi
        Random rand = new Random();
        int humidity = rand.nextInt(100); // 0 ile 100 arasında rastgele nem değeri

        // HTTP POST isteği ile veri gönderme
        try (CloseableHttpClient httpClient = HttpClients.createDefault()) {
            HttpPost postRequest = new HttpPost(serverUrl);
            String jsonPayload = "{\"humidity\":\"" + humidity + "\"}";
            StringEntity entity = new StringEntity(jsonPayload);
            postRequest.setEntity(entity);
            postRequest.setHeader("Accept", "application/json");
            postRequest.setHeader("Content-type", "application/json");

            CloseableHttpResponse response = httpClient.execute(postRequest);
            String responseString = EntityUtils.toString(response.getEntity());
            System.out.println("Response: " + responseString);

            response.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```

```
// 5 saniye bekleme
try {
    Thread.sleep(5000);
} catch (InterruptedException e) {
    e.printStackTrace();
}
}
```

Açıklamalar

1. **Sunucu URL'si**: Nem verisini göndermek için kullanılacak sunucu URL'sini `serverUrl` değişkenine tanımladık.
2. **Nem Verisi Üretilimi**: `Random` sınıfını kullanarak 0 ile 100 arasında rastgele bir nem değeri ürettik.
3. **HTTP POST İsteği**: Üretilen nem verisini JSON formatında bir HTTP POST isteği ile sunucuya gönderdik.
4. **Sonsuz Döngü**: Bu işlem sonsuz bir döngüde her 5 saniyede bir tekrarlanır.

Bu örnek, temel bir IoT uygulaması yapısı sunmaktadır. Gerçek dünyada, nem verilerini gerçek bir sensörden okuyabilir ve daha güvenilir bir web sunucusu kullanabilirsiniz. Ek olarak, hata yönetimi ve güvenlik konularını da dikkate

Java Uşık sensörü

Elbette, Java ile başka bir IoT uygulaması örneği yapalım. Bu örnekte, bir ışık sensöründen veri okuyarak bu veriyi bir veritabanına kaydeden bir uygulama oluşturacağız. Bu uygulama, JDBC kullanarak MySQL veritabanına bağlanacak.

Gerekli Kütüphaneler

Veritabanı bağlantısı için MySQL JDBC sürücüsünü kullanacağız. Maven kullanıyorsanız, `pom.xml` dosyanızda şu bağımlılığı eklemeniz gerekecek:

```
```xml
<dependency>
 <groupId>mysql</groupId>
 <artifactId>mysql-connector-java</artifactId>
 <version>8.0.23</version>
</dependency>
```

```

Java Uşık sensörü

IoT Uygulaması Örneği

Aşağıda, sahte bir ışık sensöründen verileri okuyup bu verileri MySQL veritabanına kaydeden basit bir Java programı bulunmaktadır.

MySQL Veritabanı Yapılandırması

Öncelikle, aşağıdaki SQL komutlarını kullanarak bir veritabanı ve tablo oluşturun:

Java Uşık sensörü

```
```sql
CREATE DATABASE iot_db;
USE iot_db;

CREATE TABLE light_sensor_data (
 id INT AUTO_INCREMENT PRIMARY KEY,
 light_level INT NOT NULL,
 timestamp TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

```

Java Uşık sensörü

```
#### Java Kodu

``java
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.util.Random;

public class IoTExample {

    private static final String DB_URL = "jdbc:mysql://localhost:3306/iot_db";
    private static final String USER = "root";
    private static final String PASS = "password";

    public static void main(String[] args) {
        try (Connection conn = DriverManager.getConnection(DB_URL, USER, PASS)) {

            while (true) {
                // Sahte ışık seviyesi verisi üretimi
            }
        }
    }
}
```

Java Uşık sensörü

```
while (true) {
    // Sahte ışık seviyesi verisi üretimi
    Random rand = new Random();
    int lightLevel = rand.nextInt(1000); // 0 ile 1000 arasında rastgele ışık seviyesi

    // Veritabanına veri ekleme
    String insertSQL = "INSERT INTO light_sensor_data (light_level) VALUES (?)";
    try (PreparedStatement pstmt = conn.prepareStatement(insertSQL)) {
        pstmt.setInt(1, lightLevel);
        pstmt.executeUpdate();
        System.out.println("Inserted light level: " + lightLevel);
    } catch (SQLException e) {
        e.printStackTrace();
    }

    // 5 saniye bekleme
    try {
        Thread.sleep(5000);
    } catch (InterruptedException e) {
        e.printStackTrace();
    }
} catch (SQLException e) {
    e.printStackTrace();
}
}
```
}
```

# Java Uşık sensörü

## ### Açıklamalar

1. \*\*Veritabanı Bağlantısı\*\*: `DB\_URL`, `USER` ve `PASS` değişkenleri ile veritabanı bağlantı bilgilerini tanımladık.
2. \*\*İşık Seviyesi Verisi Üretilimi\*\*: `Random` sınıfını kullanarak 0 ile 1000 arasında rastgele bir ışık seviyesi değeri ürettik.
3. \*\*Veritabanına Veri Ekleme\*\*: Üretilen ışık seviyesini `INSERT INTO` SQL komutu ile `light\_sensor\_data` tablosuna ekledik.
4. \*\*Sonsuz Döngü\*\*: Bu işlem sonsuz bir döngüde her 5 saniyede bir tekrarlanır.

Bu örnek, temel bir IoT uygulaması yapısı sunmaktadır. Gerçek dünyada, ışık seviyelerini gerçek bir sensörden okuyabilir ve daha güvenilir bir veritabanı bağlantısı kullanabilirsiniz. Ek olarak, hata yönetimi ve güvenlik konularını da dikkate almanız gerekecektir.

# Java Uşık sensörü