

ESP32 ile Tanışma

Temel Kavramlar, Donanım ve Proje Örnekleriyle
Uygulamalı Bir Başlangıç Kaynağı

Hazırlayan

Emre GÜZEL
Elektrik Elektronik Mühendisi

İçindekiler

Bölüm 1 — ESP32 Nedir?	3
Bölüm 2 — ESP32 Donanımı ve Pin Yapısı	6
Bölüm 3 — ESP32'yi Programlamaya Giriş	10
Bölüm 4 — Genel Kullanım Alanları	14
Bölüm 5 — Uygulamalı Proje Örnekleri	16
Bölüm 6 — Sık Karşılaşılan Sorunlar ve İpuçları	21
Bölüm 7 — Özet ve Sonraki Adımlar	23
Bölüm 8 — Alıştırmalar	24
Terimler Sözlüğü	28

■ Bu Kitapçık Hakkında

Bu kaynak, ESP32 geliştirme kartını ilk kez tanıyacak lise son sınıf ve üniversite birinci sınıf öğrencileri için hazırlanmıştır. Konular basit bir dille, günlük hayattan benzetmelerle anlatılmıştır. Her bölümün sonunda konuyu pekiştirmeye yönelik kısa "Hızlı Tekrar Notları" bulunur. Kitapçığın sonunda ise öğrendiklerinizi ölçebileceğiniz doğru-yanlış, boşluk doldurma ve soru-cevap alıştırmaları ile bir cevap anahtarı yer alır.

ESP32 Nedir?

ESP32; içerisinde işlemci, hafıza, kablosuz iletişim donanımı (Wi-Fi ve Bluetooth) ve çok sayıda giriş/çıkış pini barındıran, tek bir çip üzerine yerleştirilmiş küçük bir bilgisayar gibi düşünülebilecek bir mikrodenetleyicidir. Çin merkezli Espressif Systems firması tarafından üretilir ve dünya genelinde hobi projelerinden endüstriyel uygulamalara kadar çok geniş bir alanda kullanılır.

Basitçe ifade etmek gerekirse ESP32; bir sensörden veri okuyabilen, bu veriyi işleyebilen, bir LED'i yakıp söndürebilen, bir motoru sürebilen ve tüm bunları internet üzerinden uzaktan takip edilebilir hâle getirebilen küçük ve uygun fiyatlı bir elektronik karttır.

■ Günlük Hayattan Benzetme

ESP32'yi küçük ve akıllı bir asistan gibi düşünebilirsiniz. Bu asistan; çevresini sensörleriyle "görür" (sıcaklık, ışık, hareket gibi bilgileri okur), bir karar verir (örneğin "sıcaklık çok yüksek, fanı çalıştırmalıyım") ve bu kararı bir çıkışa (örneğin bir röle veya motor) ileterek uygular. Üstelik bu asistan, internete bağlanarak size telefonunuzdan haber de verebilir.

1.1 Mikrodenetleyici Kavramı

Bir mikrodenetleyici; işlemci, hafıza ve giriş/çıkış birimlerinin tek bir çip üzerinde birleştirildiği entegre devredir. Bilgisayarınızdaki işlemci (CPU) yalnızca hesaplama yapar; oysa bir mikrodenetleyici hem hesaplama yapar hem de dış dünyayla (sensörler, motorlar, ekranlar, butonlar) doğrudan haberleşebilir. ESP32 da bu tanıma uyan, üzerinde kablosuz iletişim özelliği de bulunan gelişmiş bir mikrodenetleyicidir.

Mikroişlemci ile Mikrodenetleyici Farkı

Bilgisayarlarda kullanılan mikroişlemciler (örneğin bir dizüstü bilgisayarın işlemcisi) tek başına çalışamaz; hafıza, ekran kartı ve giriş/çıkış birimleri gibi birçok ek parçaya ihtiyaç duyar. Mikrodenetleyiciler ise bu birimlerin çoğunu kendi içinde barındırdığından tek başına, küçük ve bağımsız bir cihaz olarak çalışabilir. Bu yüzden mikrodenetleyiciler; ev aletlerinden otomobillere, oyuncaklardan endüstriyel makinelere kadar pek çok cihazın içinde "gizli beyin" olarak görev yapar.

Görsel 1.1 — ESP32 Geliştirme Kartı (Genel Görünüm)

Kartın üzerinde USB bağlantı noktası, iki yanda sıralanmış pin dizileri, ortada metal kılıflı ESP32 modülü ve küçük butonlar (EN ve BOOT) görülmektedir.

1.2 ESP32'nin Kısa Tarihçesi

Espressif Systems, ilk olarak Wi-Fi özellikli ESP8266 adlı çipi üretmiş ve bu çip, düşük maliyeti sayesinde kısa sürede meraklılar arasında yaygınlaşmıştır. ESP8266'nın sunduğu imkânları daha da genişletmek isteyen firma, 2016 yılı civarında hem Wi-Fi hem de Bluetooth desteği sunan, çift çekirdekli işlemciye sahip ESP32 ailesini duyurmuştur. Günümüzde ESP32; eğitimden endüstriyel otomasyona kadar birçok alanda tercih edilen popüler bir platform hâline gelmiştir.

1.3 Öne Çıkan Özellikler

- Çift çekirdekli işlemci: Genellikle 240 MHz hızında çalışan iki adet işlem birimi bulunur; bu sayede birden fazla görev aynı anda yürütülebilir.
- Wi-Fi desteği: Kart, doğrudan bir Wi-Fi ağına bağlanabilir; internet üzerinden veri gönderip alabilir.
- Bluetooth desteği: Klasik Bluetooth ve Bluetooth Low Energy (BLE) ile kısa mesafeli kablosuz iletişim kurulabilir.
- Çok sayıda GPIO pini: Genel amaçlı giriş/çıkış pinleri sayesinde sensör, buton, LED gibi birçok bileşen bağlanabilir.
- Analog-dijital çevirici (ADC): Sıcaklık, ışık, nem gibi analog sinyal üreten sensörlerden veri okunabilir.
- Düşük güç tüketimi modları: Pille çalışan projelerde enerji tasarrufu sağlayan uyku modları bulunur.
- Uygun fiyat: Sahip olduğu özelliklere göre oldukça ekonomik bir karttır.

1.4 ESP32 ile Arduino Uno Arasındaki Fark

Arduino Uno, eğitim amaçlı en yaygın kullanılan kartlardan biridir; ancak üzerinde kablosuz iletişim donanımı bulunmaz ve işlem gücü ESP32'ye kıyasla oldukça sınırlıdır. ESP32 ise Arduino Uno'nun sunduğu giriş/çıkış imkânlarına ek olarak Wi-Fi, Bluetooth, daha fazla işlem gücü ve daha fazla hafıza sunar. Bu nedenle internete bağlı (IoT) projelerde genellikle ESP32 tercih edilir.

Özellik	Arduino Uno	ESP32
İşlemci Hızı	16 MHz	240 MHz (çift çekirdek)
Wi-Fi	Yok	Var
Bluetooth	Yok	Var (Klasik + BLE)
GPIO Sayısı	Yaklaşık 20	Yaklaşık 34
Çalışma Voltajı	5 V	3.3 V

❏ Bilgi

ESP32'nin çalışma voltajı 3.3 V'tur. 5 V ile çalışan Arduino sensörlerini doğrudan ESP32'ye bağlamadan önce voltaj seviyelerinin uyumlu olduğundan emin olunmalıdır.

1.5 ESP32 Ailesinin Farklı Modelleri

"ESP32" adı aslında tek bir ürünü değil, benzer özelliklere sahip bir kart ailesini ifade eder. Piyasada en sık karşılaşılan modeller şunlardır:

- ESP32-WROOM: En yaygın kullanılan, çift çekirdekli, Wi-Fi ve Bluetooth destekli temel modeldir. Bu kitapçıkta örnekler bu model esas alınarak hazırlanmıştır.
- ESP32-S2: Tek çekirdekli işlemciye sahiptir; USB üzerinden doğrudan iletişime odaklanan projelerde tercih edilir.
- ESP32-S3: Yapay zekâ tabanlı basit görüntü ve ses işleme özellikleri eklenmiş, daha güçlü bir modeldir.
- ESP32-C3: Daha küçük boyutlu ve daha düşük maliyetlidir; farklı bir işlemci mimarisi (RISC-V) kullanır.

Yeni başlayanlar için genellikle ESP32-WROOM tabanlı geliştirme kartları önerilir, çünkü hem yaygındır hem de internette bolca örnek ve kaynak bulunur.

□ Hızlı Tekrar Notları

- ESP32, Wi-Fi ve Bluetooth özellikli, çift çekirdekli bir mikrodenetleyicidir.
- Mikrodenetleyiciler; işlemci, hafıza ve giriş/çıkış birimlerini tek çipte birleştirir.
- ESP32, Espressif Systems firması tarafından üretilir.
- ESP32; Arduino Uno'ya kıyasla daha hızlı çalışır ve kablosuz iletişim sunar.
- ESP32 ailesinde WROOM, S2, S3 ve C3 gibi farklı modeller bulunur.

ESP32 Donanımı ve Pin Yapısı

Bir ESP32 geliştirme kartını ilk elinize aldığınızda üzerinde çok sayıda pin, birkaç buton ve bir USB bağlantı noktası görürsünüz. Bu bölümde kartın temel donanım bileşenlerini ve en sık kullanılan pin tiplerini ayrıntılı biçimde tanıyacağız.

2.1 Kart Üzerindeki Temel Bileşenler

- USB Bağlantı Noktası: Kartı bilgisayara bağlamak, kod yüklemek ve kartı güçlendirmek için kullanılır.
- EN (Enable) Butonu: Kartı yeniden başlatmaya yarar; bilgisayardaki "reset" tuşuna benzer.
- BOOT Butonu: Kart programlanırken bazı modellerde bu butona basılı tutmak gerekebilir.
- GPIO Pinleri: Kartın iki yanında sıralanmış, dış dünya ile bağlantı kurmaya yarayan genel amaçlı pinlerdir.
- Güç Pinleri (3V3, GND, VIN): Harici bileşenlere enerji sağlamak veya kartı harici bir kaynaktan beslemek için kullanılır.

Görsel 2.1 — ESP32 Pin Yerleşim Şeması (Pinout)

Kartın üst kısmında BOOT ve EN butonları, sol ve sağ kenarlarda numaralandırılmış GPIO pinleri, alt kısımda ise 3V3, GND ve VIN güç pinleri şematik olarak gösterilmektedir.

2.2 Sık Kullanılan Pin Türleri

Dijital Giriş/Çıkış (GPIO)

Bir GPIO pini, yazılım aracılığıyla giriş veya çıkış olarak ayarlanabilir. Çıkış olarak ayarlandığında bir LED'i yakıp söndürebilir; giriş olarak ayarlandığında bir butonun basılıp basılmadığını algılayabilir.

■ Günlük Hayattan Benzetme

GPIO pinlerini evinizin kapıları gibi düşünebilirsiniz. Bir kapıyı "çıkış" olarak kullanırsanız oradan dışarı bir şey gönderirsiniz (örneğin LED'e enerji göndermek); "giriş" olarak kullanırsanız dışarıdan içeri bir bilgi alırsınız (örneğin bir butonun durumu).

Analog Giriş (ADC)

ESP32 üzerindeki bazı pinler analog sinyalleri okuyabilir. Örneğin bir sıcaklık sensöründen veya bir potansiyometreden (döner direnç) gelen değişken gerilim, bu pinler aracılığıyla sayısal bir değere dönüştürülerek okunur. ADC, gerçek dünyadaki sürekli değişen

sinyalleri, bilgisayarın anlayabileceği sayılara çeviren bir tür "tercüman" gibi çalışır.

PWM (Darbe Genişlik Modülasyonu)

PWM, bir çıkışın açık ve kapalı kalma sürelerini hızlıca değiştirerek analog bir etki oluşturmayı sağlar. Bir LED'in parlaklığını kademeli olarak ayarlamak veya bir motorun hızını kontrol etmek için kullanılır.

□ Günlük Hayattan Benzetme

Bir odanın ışığını çok hızlı şekilde açıp kapatırsanız, insan gözü bunu fark edemez ve ışığın daha soluk yandığını algılar. PWM de tam olarak buna benzer bir mantıkla çalışır: LED'i çok hızlı açıp kapatarak parlaklığını ayarlar.

I2C, SPI ve UART (Haberleşme Protokolleri)

- I2C: İki kablo (SDA ve SCL) ile birden fazla sensörün aynı hatta bağlanmasını sağlayan bir haberleşme yöntemidir.
- SPI: Daha hızlı veri aktarımı gerektiren ekran veya hafıza kartı gibi bileşenlerde tercih edilir.
- UART (Seri Haberleşme): Kartın bilgisayarla veya başka bir kartla, TX ve RX pinleri üzerinden basit metin tabanlı veri alışverişi yapmasını sağlar.

⚠ Dikkat — Pin Sınırlamaları

Bazı GPIO pinleri kartın açılışında (boot) özel görevler üstlenir ve bu pinlerin yanlış kullanımı kartın düzgün başlamamasına neden olabilir. Bir projeye başlamadan önce kullanılan kart modelinin resmî pinout şemasına mutlaka bakılmalıdır.

2.3 Güç Kaynağı ve Besleme

ESP32, genellikle bilgisayarın USB portundan gelen 5 voltluk gerilimi, kart üzerindeki bir voltaj regülatörü sayesinde kendi ihtiyaç duyduğu 3.3 volta düşürür. Kart; bir bilgisayardan, bir USB şarj adaptöründen veya uygun bir pil/batarya biriminden beslenebilir.

- Taşınabilir projelerde genellikle şarj edilebilir lityum polimer (Li-Po) piller kullanılır.
- Motor gibi yüksek akım çeken bileşenler kullanılıyorsa kart, yalnızca bilgisayarın USB portundan değil; harici ve yeterli akım sağlayabilen bir kaynaktan da beslenmelidir.
- Pille çalışan projelerde ESP32'nin düşük güç tüketimi modları, pil ömrünü uzatmak için aktif hâle getirilebilir.

2.4 Breadboard ve Jumper Kablo Kullanımı

Elektronik devre denemeleri yaparken bileşenleri lehimlemeden, geçici olarak birbirine bağlamak için breadboard adı verilen delikli deney tahtaları kullanılır. Breadboard üzerindeki delikler, belirli sıralar hâlinde birbirine elektriksel olarak bağlıdır.

- Jumper kablolar: Breadboard üzerindeki bileşenleri ESP32'nin pinlerine bağlamak için kullanılan kısa, esnek kablolardır.
- Direnç: Devredeki akımı sınırlayarak LED gibi hassas bileşenlerin zarar görmesini engeller.
- Devreyi kurarken sıra: Önce kart enerjisiz durumdayken bağlantılar yapılmalı, ardından kart bilgisayara bağlanmalıdır.

Görsel 2.2 — Breadboard Üzerinde Basit Bir LED Bağlantısı

ESP32'nin bir GPIO pininden çıkan jumper kablo, breadboard üzerindeki bir direnç ve LED üzerinden GND pinine geri bağlanmaktadır.

⚠ Dikkat — Elektrostatik Deşarj (ESD)

ESP32 gibi hassas elektronik kartlarla çalışırken ellerin ve çalışma ortamının statik elektrik yükünden arındırılmış olmasına dikkat edilmelidir. Kart bağlıyken pinlere doğrudan dokunmaktan kaçınılmalıdır.

📌 Hızlı Tekrar Notları

- GPIO pinleri yazılımla giriş veya çıkış olarak ayarlanabilen genel amaçlı pinlerdir.
- ADC, analog sinyalleri sayısal değere çevirir.
- PWM, bir çıkışı hızlıca açıp kapatarak analog bir etki oluşturur.
- I2C, SPI ve UART; ESP32'nin diğer bileşenlerle haberleşmesini sağlayan protokollerdir.
- ESP32, 3.3 volt ile çalışır ve breadboard üzerinde lehimsiz devreler kurulabilir.

ESP32'yi Programlamaya Giriş

ESP32'yi programlamak için en yaygın kullanılan yöntem, Arduino IDE adlı ücretsiz yazılımdır. Bu bölümde kurulum adımlarını, temel kod yapısını ve basit programlama kavramlarını adım adım ele alacağız.

3.1 Arduino IDE Kurulumu

- Arduino IDE, resmî internet sitesinden bilgisayara indirilip kurulur.
- Program açıldıktan sonra ayarlar bölümünden ESP32 kart desteği eklenir.
- Kart yöneticisinden kullanılan ESP32 modeli seçilir.
- Kart, USB kablosu ile bilgisayara bağlanır ve doğru bağlantı noktası (port) seçilir.
- Kurulumun başarılı olduğunu kontrol etmek için basit bir LED yakıp söndürme örneği yüklenebilir.

3.2 Temel Kod Yapısı: setup() ve loop()

Arduino tabanlı her programda iki temel fonksiyon bulunur: setup() fonksiyonu kart açıldığında yalnızca bir kez çalışır ve genellikle pin ayarları burada yapılır. loop() fonksiyonu ise kart çalıştığı sürece sürekli tekrar eder.

❏ Örnek: Basit Kod İskeleti

```
void setup() {  
  // Bu bölüm sadece bir kez çalışır  
  pinMode(2, OUTPUT);  
}  
  
void loop() {  
  // Bu bölüm sürekli tekrar eder  
  digitalWrite(2, HIGH);  
  delay(1000);  
  digitalWrite(2, LOW);  
  delay(1000);  
}
```

❏ Günlük Hayattan Benzetme

setup() fonksiyonunu sabah evden çıkmadan önce yapılan hazırlıklara (ayakkabı bağlama, çanta alma) benzetebiliriz; yalnızca bir kez yapılır. loop() fonksiyonu ise okulda her gün tekrarlanan ders programına benzer; kart çalıştığı sürece aynı adımlar tekrar eder.

3.3 Değişkenler ve Veri Tipleri

Bir programda değişen bilgileri (bir sıcaklık değeri, bir pin numarası, bir metin) saklamak için değişkenler kullanılır. Her değişkenin bir veri tipi vardır ve bu tip, değişkenin ne tür bir bilgi tutacağını belirler.

Veri Tipi	Ne Tutar?	Örnek
int	Tam sayı	int sıcaklik = 25;
float	Ondalıklı sayı	float nem = 45.6;
bool	Doğru / Yanlış (true / false)	bool ledDurumu = true;
String	Metin	String mesaj = "Merhaba";

3.4 Yorum Satırları (Açıklama Ekleme)

Kod içine, çalışmayan fakat kodu okuyan kişiye bilgi veren açıklamalar eklenebilir. Tek satırlık açıklamalar // ile başlar; birden fazla satırlık açıklamalar ise /* ve */ arasına yazılır. Yorum satırları, hem kendi kodunuzu daha sonra hatırlamanıza hem de başkalarının kodunuzu anlamasına yardımcı olur.

3.5 Koşul İfadeleri: if / else

Bir programın belirli bir duruma göre farklı davranması gerektiğinde koşul ifadeleri kullanılır. Örneğin bir sıcaklık değeri belli bir eşiği aştığında bir fanı çalıştırmak istenebilir.

❏ Örnek: if / else Kullanımı

```
int sıcaklik = 32;

if (sıcaklik > 30) {
    Serial.println("Sıcaklık yüksek, fan çalışsın!");
} else {
    Serial.println("Sıcaklık normal.");
}
```

3.6 Sık Kullanılan Hazır Fonksiyonlar

- pinMode(pin, mod): Bir pinin giriş mi yoksa çıkış mı olacağını belirler.
- digitalWrite(pin, deger): Bir çıkış pinine HIGH (yüksek) veya LOW (düşük) değerini gönderir.
- digitalRead(pin): Bir giriş pininin o anki değerini (HIGH veya LOW) okur.
- analogRead(pin): Bir analog pinden okunan değeri sayısal olarak döndürür.
- delay(milisaniye): Programın belirtilen süre kadar beklemesini sağlar.
- Serial.println(mesaj): Bilgisayar ekranındaki Seri Port İzleyici'ye bir mesaj yazdırır.

3.7 Alternatif: MicroPython

Arduino IDE dışında, ESP32 üzerinde Python diline benzer bir dil olan MicroPython ile de kod yazılabilir. MicroPython, özellikle Python bilenler için daha kolay bir başlangıç sunar; ancak bu kitapçıkta örnekler Arduino IDE üzerinden anlatılacaktır.

İpucu

İlk kez ESP32 ile çalışacaksanız, kartın doğru tanınıp tanınmadığını görmek için önce basit bir LED yakıp söndürme (blink) örneğini yükleyerek başlamanız önerilir.

Görsel 3.1 — Arduino IDE Ekran Görünümü

Pencerenin üst kısmında yükle ve doğrula butonları, sol tarafta kod düzenleme alanı, alt kısımda ise yükleme durumunu gösteren mesaj kutusu yer almaktadır.

Hızlı Tekrar Notları

- setup() bir kez, loop() ise sürekli olarak çalışır.
- int, float, bool ve String en sık kullanılan veri tipleridir.
- // işareti ile başlayan satırlar açıklama (yorum) satırlarıdır.
- if / else ifadeleri, programın koşula göre farklı davranmasını sağlar.
- pinMode, digitalWrite, digitalRead ve delay en sık kullanılan hazır fonksiyonlardır.

Genel Kullanım Alanları

ESP32'nin kablosuz iletişim özelliği, onu özellikle "Nesnelerin İnterneti" (IoT) olarak adlandırılan alanda çok popüler hâle getirmiştir. Bu bölümde ESP32'nin sıkça kullanıldığı başlıca alanlara göz atacağız.

4.1 Ev Otomasyonu

Akıllı prizler, uzaktan kontrol edilebilen aydınlatma sistemleri ve oda sıcaklığını izleyen sistemler genellikle ESP32 tabanlıdır. Kullanıcı, telefon uygulaması üzerinden evindeki cihazları internet üzerinden kontrol edebilir.

4.2 Sensör Ağları, Tarım ve Çevre İzleme

Sıcaklık, nem, hava kalitesi gibi verileri toplayan sensörler ESP32'ye bağlanarak bu veriler düzenli aralıklarla bir sunucuya veya mobil uygulamaya gönderilebilir. Bu sayede bir sera, bir bahçe ya da bir depo uzaktan izlenebilir. Örneğin toprak nem sensörüyle donatılmış bir ESP32, toprak kurduğunda otomatik olarak sulama sistemini çalıştırabilir.

4.3 Giyilebilir Teknoloji ve Küçük Robotik Projeler

ESP32'nin küçük boyutu ve düşük güç tüketimi, giyilebilir cihazlarda ve basit robot projelerinde tercih edilmesini sağlar. Örneğin bir robotun motorlarını kontrol ederken aynı anda telefonla kablosuz haberleşmesi mümkündür.

4.4 Uzaktan İzleme ve Güvenlik Sistemleri

Kapı sensörleri, hareket algılayıcılar ve basit kamera modülleriyle birleştirilen ESP32 kartları, düşük maliyetli güvenlik ve izleme sistemlerinin temelini oluşturabilir.

4.5 Eğitim ve Hobi Projeleri

ESP32'nin uygun fiyatı ve geniş kaynak desteği, onu okullarda ve hobi amaçlı projelerde de yaygın bir tercih hâline getirmiştir. Basit bir hava durumu istasyonu, uzaktan kumandalı bir araba ya da ödev amaçlı küçük bir oyun konsolu gibi birçok eğlenceli proje, ESP32 ile öğrenciler tarafından kolaylıkla geliştirilebilir.

❏ Bilgi

ESP32'nin kullanım alanları burada sayılanlarla sınırlı değildir. Aynı temel prensipler; sağlık, eğitim ve endüstriyel otomasyon gibi birçok farklı alana da uyarlanabilir.

Görsel 4.1 — ESP32 Tabanlı Örnek Bir IoT Sistemi

Şemada; sıcaklık sensörünün ESP32'ye bağlandığı, ESP32'nin ev Wi-Fi ağına bağlanarak bu veriyi internet üzerinden bir telefon uygulamasına ilettiği gösterilmektedir.

□ Hızlı Tekrar Notları

- ESP32; ev otomasyonu, tarım, giyilebilir teknoloji ve güvenlik sistemlerinde kullanılır.
- Nesnelerin İnterneti (IoT), cihazların internet üzerinden veri alışverişi yapmasıdır.
- ESP32'nin uygun fiyatı, onu eğitim ve hobi projeleri için de ideal kılar.

Uygulamalı Proje Örnekleri

Bu bölümde, öğrencilerin ESP32 ile ilk deneyimlerini kazanabileceği, kolaydan zora doğru sıralanmış beş temel proje örneği yer almaktadır. Her proje için gerekli malzemeler, kısa açıklama, bağlantı adımları ve örnek kod verilmiştir.

Proje 1 — LED Yakıp Söndürme (Blink)

Amaç: ESP32'nin bir çıkış pinini kullanarak bağlı bir LED'i belirli aralıklarla yakıp söndürmek.

Gerekli Malzemeler: ESP32 kartı, LED, 220 Ω direnç, breadboard, jumper kablolar.

- LED'in uzun bacağı (anot), direnç üzerinden ESP32'nin 2 numaralı GPIO pinine bağlanır.
- LED'in kısa bacağı (katot), doğrudan GND pinine bağlanır.
- Kart USB kablosuyla bilgisayara bağlanır ve aşağıdaki kod yüklenir.

□ **blink.ino**

```
int ledPin = 2;

void setup() {
  pinMode(ledPin, OUTPUT);
}

void loop() {
  digitalWrite(ledPin, HIGH);
  delay(500);
  digitalWrite(ledPin, LOW);
  delay(500);
}
```

□ **Bunu Deneyin**

delay() fonksiyonundaki 500 değerini 100 veya 2000 gibi farklı sayılarla değiştirip LED'in yanma hızının nasıl değiştiğini gözlemleyin.

Proje 2 — Buton ile LED Kontrolü

Amaç: Bir butona basıldığında LED'in yanmasını, bırakıldığında sönmesini sağlamak. Bu proje, dijital giriş okumanın temel mantığını öğretir.

Gerekli Malzemeler: ESP32 kartı, LED, buton, 220 Ω direnç, breadboard, jumper kablolar.

- LED, Proje 1'deki gibi 2 numaralı pine bağlanır.
- Butonun bir ucu 4 numaralı GPIO pinine, diğer ucu GND pinine bağlanır.
- Kodda kullanılan INPUT_PULLUP ayarı sayesinde ekstra bir direnç kullanmaya gerek kalmaz.

□ **buton_led.ino**

```
int ledPin = 2;
int butonPin = 4;

void setup() {
  pinMode(ledPin, OUTPUT);
  pinMode(butonPin, INPUT_PULLUP);
}

void loop() {
  if (digitalRead(butonPin) == LOW) {
    digitalWrite(ledPin, HIGH);
  } else {
    digitalWrite(ledPin, LOW);
  }
}
```

□ **Günlük Hayattan Benzetme**

Bu proje, bir kapı zili için çalışma mantığına benzer: Butona (zile) bastığınız sürece bir sonuç (ses ya da bu örnekte ışık) üretilir; bıraktığınızda sonuç durur.

Proje 3 — Sıcaklık ve Nem Sensöründen Veri Okuma

Amaç: DHT11 sensörü ile ortam sıcaklığını ve nemini ölçüp bilgisayar ekranında (Seri Port izleyici) görüntülemek.

Gerekli Malzemeler: ESP32 kartı, DHT11 sensörü, jumper kablolar, "DHT sensor library" adlı kütüphane.

- DHT11 sensörünün veri (data) pini, ESP32'nin 15 numaralı GPIO pinine bağlanır.
- Sensörün besleme ve toprak (GND) uçları sırasıyla 3V3 ve GND pinlerine bağlanır.
- Arduino IDE üzerinden "DHT sensor library" kütüphanesi kurulur.

`dht_okuma.ino`

```
#include <DHT.h>

#define DHTPIN 15
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(115200);
  dht.begin();
}

void loop() {
  float sicaklik = dht.readTemperature();
  float nem = dht.readHumidity();
  Serial.print("Sicaklik: ");
  Serial.print(sicaklik);
  Serial.print(" C   Nem: ");
  Serial.println(nem);
  delay(2000);
}
```

`İpucu`

Kütüphaneler, Arduino IDE içerisindeki "Kütüphane Yöneticisi" (Library Manager) üzerinden aratılıp tek tıkla kurulabilir.

Proje 4 — Wi-Fi Ağına Bağlanma

Amaç: ESP32'yi ev veya okul Wi-Fi ağına bağlayarak internete erişim sağlamak. Bu, ileride kurulacak birçok IoT projesinin ilk adımıdır.

Gerekli Malzemeler: ESP32 kartı, 2.4 GHz destekli bir Wi-Fi ağı (isim ve şifre bilgisi).

❏ wifi_baglanti.ino

```
#include <WiFi.h>

const char* ssid = "AG_ADI";
const char* sifre = "AG_SIFRESI";

void setup() {
  Serial.begin(115200);
  WiFi.begin(ssid, sifre);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("Baglanti basarili!");
  Serial.println(WiFi.localIP());
}

void loop() {
}
```

⚠ Dikkat — Ağ Uyumluluğu

ESP32, yalnızca 2.4 GHz frekans bandındaki Wi-Fi ağlarına bağlanabilir; 5 GHz ağlarına doğrudan bağlanamaz.

Proje 5 — Basit Bir Web Sunucusu ile LED Kontrolü

Amaç: ESP32 üzerinde küçük bir web sayfası çalıştırarak, aynı Wi-Fi ağındaki bir telefon veya bilgisayardan tarayıcı üzerinden LED'i açıp kapatmak. Bu proje, önceki projelerde öğrenilen dijital çıkış ve Wi-Fi bağlantısı konularını bir araya getirir.

Gerekli Malzemeler: ESP32 kartı, LED, 220 Ω direnç, breadboard, jumper kablolar.

▢ web_sunucu_led.ino

```
#include <WiFi.h>
WiFiServer sunucu(80);
int ledPin = 2;

void setup() {
  pinMode(ledPin, OUTPUT);
  Serial.begin(115200);
  WiFi.begin("AG_ADI", "AG_SIFRESI");
  while (WiFi.status() != WL_CONNECTED) delay(500);
  sunucu.begin();
  Serial.println(WiFi.localIP());
}

void loop() {
  WiFiClient istemci = sunucu.available();
  if (istemci) {
    String istek = istemci.readStringUntil('\r');
    if (istek.indexOf("/ac") != -1) digitalWrite(ledPin, HIGH);
    if (istek.indexOf("/kapat") != -1) digitalWrite(ledPin, LOW);
    istemci.println("HTTP/1.1 200 OK");
    istemci.println("LED durumu guncellendi");
    istemci.stop();
  }
}
```

⚠ Dikkat — Ağ Bilgisi Güvenliği

Örnek kodlardaki Wi-Fi adı ve şifresi bilgileri, gerçek projelerde başkalarıyla paylaşılan dosyalarda açık şekilde bırakılmamalıdır.

Görsel 5.1 — Tarayıcı Üzerinden LED Kontrol Arayüzü

Basit bir web sayfasında "LED'i Aç" ve "LED'i Kapat" yazılı iki buton ve altında LED'in o anki durumunu gösteren kısa bir metin yer almaktadır.

□ Hızlı Tekrar Notları

- Blink projesi, GPIO çıkışlarının en temel kullanımını gösterir.
- INPUT_PULLUP, ekstra direnç kullanmadan buton okumayı kolaylaştırır.
- DHT11 sensörü, sıcaklık ve nem verisini okumak için kullanılır.
- ESP32 yalnızca 2.4 GHz Wi-Fi ağlarına bağlanabilir.
- Web sunucusu projesi; GPIO, Wi-Fi ve haberleşme konularını bir araya getirir.

Sık Karşılaşılan Sorunlar ve İpuçları

6.1 Kart Bilgisayar Tarafından Tanınmıyor

Bu durumda öncelikle USB kablusunun yalnızca şarj değil, veri aktarımını da destekleyen bir kablo olduğundan emin olunmalıdır. Ayrıca bilgisayara uygun USB sürücüsünün (genellikle CP210x veya CH340 sürücüsü) kurulu olup olmadığı kontrol edilmelidir.

6.2 Kod Yüklenirken Hata Alınıyor

Bazı ESP32 kartlarında, kod yükleme işlemi başlamadan hemen önce kart üzerindeki BOOT butonuna kısa süreliğine basılı tutmak gerekebilir. Ayrıca Arduino IDE üzerinde doğru kart modeli ve doğru bağlantı noktasının (port) seçili olduğundan emin olunmalıdır.

6.3 Wi-Fi Ağına Bağlanamıyor

ESP32'nin yalnızca 2.4 GHz frekans bandındaki Wi-Fi ağlarını desteklediği unutulmamalıdır. 5 GHz bandında yayın yapan ağlara ESP32 doğrudan bağlanamaz. Ağ adı ve şifrenin doğru yazıldığından da emin olunmalıdır.

6.4 Kart Aşırı Isınıyor veya Kendiliğinden Yeniden Başlıyor

Bu genellikle yetersiz güç kaynağından kaynaklanır. Özellikle motor veya güçlü LED şeritleri gibi fazla akım çeken bileşenler kullanılırken kartın harici, yeterli akım sağlayabilen bir kaynaktan beslenmesi önerilir.

6.5 Sensörden Anlamsız veya Sabit Değerler Okunuyor

Bu sorun genellikle yanlış pin bağlantısından, gevşek bir kablodan veya yanlış kütüphane kullanımından kaynaklanır. Bağlantıların şemaya uygun olduğu ve kullanılan kütüphanenin sensör modeliyle eşleştiği kontrol edilmelidir.

⚠ Dikkat — Kısa Devre Riski

Bağlantı yaparken güç (3V3 veya VIN) ve toprak (GND) uçlarının yanlışlıkla doğrudan birbirine temas etmemesine dikkat edilmelidir; aksi hâlde kart zarar görebilir.

📌 İpucu

Bir sorunla karşılaşıldığında, Arduino IDE'nin alt kısmındaki mesaj penceresinde çıkan hata metni genellikle sorunun kaynağı hakkında doğrudan bilgi verir; bu mesajı dikkatle okumak çoğu zaman çözüme en hızlı yoldur.

□ Hızlı Tekrar Notları

- Kart tanınmıyorsa önce USB kablosu ve sürücüler kontrol edilmelidir.
- Kod yüklenirken hata alınırsa BOOT butonu ve port seçimi kontrol edilmelidir.
- ESP32 yalnızca 2.4 GHz Wi-Fi ağlarına bağlanabilir.
- Ani yeniden başlamalar genellikle yetersiz güç kaynağından kaynaklanır.

Özet ve Sonraki Adımlar

Bu kitapçıkta ESP32'nin ne olduğunu, temel donanım yapısını, nasıl programlandığını, hangi alanlarda kullanıldığını ve uygulamalı proje örnekleriyle nasıl çalıştığını inceledik. Öğrenilen temel kavramlar şu şekilde özetlenebilir:

- ESP32, Wi-Fi ve Bluetooth özellikli, çift çekirdekli bir mikrodenetleyicidir.
- GPIO, ADC ve PWM pinleri sayesinde birçok sensör ve bileşenle çalışabilir.
- Arduino IDE üzerinden setup() ve loop() fonksiyonları ile programlanır.
- Değişkenler, koşul ifadeleri ve hazır fonksiyonlar temel programlama araçlarıdır.
- Ev otomasyonundan sensör ağlarına kadar geniş bir kullanım alanına sahiptir.
- Basit LED kontrolünden Wi-Fi tabanlı web sunucusuna kadar kademeli olarak daha karmaşık projeler geliştirilebilir.

Sonraki Adımlar İçin Öneriler

- Bu kitapçıkta ki proje örneklerini kendi ESP32 kartınızda tek tek uygulayın.
- Farklı sensör türlerini (ışık, mesafe, hareket) ESP32 ile denemeye başlayın.
- Basit bir mobil uygulama veya web paneli ile ESP32'nizi kontrol etmeyi öğrenin.
- Zaman içinde birden fazla sensörü aynı anda kullanan daha kapsamlı bir IoT projesi tasarlamayı hedefleyin.

□ Kapanış Notu

Elektronik ve programlama, uygulama yaptıkça pekişen alanlardır. Bu kitapçıkta ki örnekleri birebir uygulamanız, ESP32'yi gerçekten kavramanın en etkili yoludur.

Alıştırmalar

Bu bölümdeki alıştırmalar, kitapçıkta öğrendiğiniz kavramları pekiştirmeniz için hazırlanmıştır. Önce soruları kendi başınıza cevaplamayı deneyin, ardından bölüm sonundaki cevap anahtarından kendinizi kontrol edin.

8.1 Doğru / Yanlış Soruları

	D	Y
1. ESP32, yalnızca kablolu iletişim ile çalışan bir mikrodenetleyicidir.	☐	☐
2. ESP32 üzerinde Wi-Fi ve Bluetooth özellikleri birlikte bulunur.	☐	☐
3. Arduino Uno, ESP32'ye göre daha yüksek işlem hızına sahiptir.	☐	☐
4. GPIO pinleri, yazılım aracılığıyla giriş veya çıkış olarak ayarlanabilir.	☐	☐
5. ESP32'nin çalışma voltajı 5 V'tur.	☐	☐
6. PWM, bir çıkışın açık/kapalı sürelerini değiştirerek analog bir etki oluşturur.	☐	☐
7. ESP32, yalnızca 5 GHz Wi-Fi ağlarına bağlanabilir.	☐	☐
8. setup() fonksiyonu, kart çalıştığı sürece sürekli tekrar eder.	☐	☐
9. DHT11 sensörü, sıcaklık ve nem ölçmek için kullanılır.	☐	☐
10. Breadboard, elektronik bileşenlerin lehimlenerek kalıcı biçimde bağlandığı bir tahtadır.	☐	☐

Aşağıdaki cümlelerdeki boşlukları, kitapçıkta geçen doğru kavramlarla tamamlayınız.

1. ESP32, _____ ve Bluetooth özelliklerine sahip bir mikrodnetleyicidir.
2. Bir mikrodnetleyicinin dış dünyayla bağlantı kurmasını sağlayan genel amaçlı pinlere _____ denir.
3. Analog sinyalleri sayısal değere çeviren birime _____ denir.
4. Arduino tabanlı bir programda yalnızca bir kez çalışan fonksiyon _____ fonksiyonudur.
5. Kart çalıştığı sürece sürekli tekrar eden fonksiyon _____ fonksiyonudur.
6. ESP32'yi üreten firmanın adı _____ 'tir.
7. Sıcaklık ve nem ölçmek için kullanılan yaygın sensör _____ 'dir.
8. Elektronik bileşenlerin lehimsiz olarak bağlanabildiği deney tahtasına _____ denir.
9. ESP32'nin çalışma voltajı _____ voltur.
10. İki kablo (SDA ve SCL) ile birden fazla sensörün bağlanmasını sağlayan haberleşme protokolüne _____ denir.

Aşağıdaki soruları kendi cümlelerinizle, kısaca cevaplayınız.

1. ESP32'nin Arduino Uno'ya göre iki temel üstünlüğü nedir?

Cevap: _____

2. GPIO pini ne işe yarar? Kısaca açıklayınız.

Cevap: _____

3. PWM nerede kullanılır? Bir örnek veriniz.

Cevap: _____

4. ESP32 ile kurulan bir web sunucusu günlük hayatta ne işe yarayabilir? Bir örnek üzerinden açıklayınız.

Cevap: _____

5. Kod yüklerken kart bilgisayar tarafından tanınmıyorsa ilk olarak ne kontrol edilmelidir?

Cevap: _____

6. ESP32'nin pille çalışan projelerde düşük güç modlarına neden ihtiyaç duyduğunu açıklayınız.

Cevap: _____

Doğru / Yanlış Soruları — Cevaplar

1-Yanlış 2-Doğru 3-Yanlış 4-Doğru 5-Yanlış 6-Doğru 7-Yanlış 8-Yanlış 9-Doğru 10-Yanlış

Boşluk Doldurma — Cevaplar

1. Wi-Fi
2. GPIO
3. ADC (Analog-Dijital Çevirici)
4. setup()
5. loop()
6. Espressif Systems
7. DHT11
8. Breadboard
9. 3.3
10. I2C

Kısa Cevaplı Sorular — Örnek Cevaplar

1. ESP32'nin en önemli iki üstünlüğü; üzerinde Wi-Fi/Bluetooth desteğinin bulunması ve Arduino Uno'ya kıyasla çok daha yüksek işlem gücüne sahip olmasıdır.
2. GPIO pinleri, yazılım aracılığıyla giriş veya çıkış olarak ayarlanabilen genel amaçlı pinlerdir; bir LED'i sürmek veya bir butonun durumunu okumak için kullanılabilir.
3. PWM, bir çıkışın açık/kapalı sürelerini hızlıca değiştirerek analog bir etki oluşturur; örneğin bir LED'in parlaklığını veya bir motorun hızını ayarlamak için kullanılır.
4. Örneğin bir öğrenci, ESP32 ile kurduğu web sunucusu sayesinde evindeki bir lambayı, aynı Wi-Fi ağına bağlı telefonundan tarayıcı üzerinden açıp kapatabilir.
5. Öncelikle USB kablusunun veri aktarımını desteklediğinden ve gerekli USB sürücülerinin bilgisayara kurulu olduğundan emin olunmalıdır.
6. Pille çalışan projelerde enerji sınırlıdır; düşük güç modları sayesinde kart, kullanılmadığı anlarda daha az enerji harcayarak pilin daha uzun süre dayanmasını sağlar.

Terimler Sözlüğü

Mikrodenetleyici	İşlemci, hafıza ve giriş/çıkış birimlerini tek bir çipte birleştiren, dış dünyayla doğrudan etkileşim kurabilen entegre devre.
Mikroişlemci	Yalnızca hesaplama yapan, çalışabilmesi için hafıza ve giriş/çıkış birimleri gibi ek parçalara ihtiyaç duyan işlem birimi.
GPIO	Genel amaçlı, yazılımla giriş veya çıkış olarak ayarlanabilen dijital pin.
ADC	Analog bir sinyali (örneğin bir sensörden gelen değişken gerilimi) sayısal bir değere dönüştüren birim.
PWM	Bir çıkışın açık/kapalı sürelerini hızlıca değiştirerek analog bir etki oluşturan sinyal türü.
Wi-Fi	Kablosuz yerel ağ üzerinden internete bağlanmayı sağlayan iletişim teknolojisi.
Bluetooth / BLE	Kısa mesafeli kablosuz veri iletişimi sağlayan teknoloji; BLE, düşük enerji tüketen türüdür.
IoT (Nesnelerin İnterneti)	Günlük nesnelerin internet üzerinden birbirleriyle veya kullanıcıyla veri alışverişi yaptığı sistemlerin genel adı.
Arduino IDE	Mikrodenetleyicilere kod yazmak ve yüklemek için kullanılan ücretsiz yazılım geliştirme ortamı.
Değişken	Bir programda değişebilen bir bilgiyi (sayı, metin, doğru/yanlış değeri) saklamak için kullanılan isimlendirilmiş bellek alanı.
Fonksiyon	Belirli bir işi yapmak üzere tanımlanan ve gerektiğinde çağrılan kod parçası.
Koşul İfadesi (if / else)	Programın belirli bir duruma göre farklı davranmasını sağlayan yapı.
Seri Port İzleyici	Kart ile bilgisayar arasındaki metin tabanlı veri alışverişini ekranda gösteren Arduino IDE aracı.
Breadboard	Elektronik bileşenlerin lehim gerektirmeden geçici olarak bağlanabildiği delikli deney tahtası.
Jumper Kablo	Breadboard üzerindeki bileşenleri birbirine veya bir karta bağlamak için kullanılan kısa, esnek kablo.
I2C / SPI / UART	ESP32'nin diğer bileşenlerle veri alışverişi yapmasını sağlayan farklı haberleşme protokolleri.
ESD (Elektrostatik Deşarj)	Statik elektrik yükünün ani biçimde boşalarak hassas elektronik bileşenlere zarar verebilmesi durumu.