

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	5
Bölüm 3 — ESP32'yi Programlamaya Giriş	8
Bölüm 4 — Genel Kullanım Alanları	11
Bölüm 5 — Uygulamalı Proje Örnekleri	13
Bölüm 6 — Sık Karşılaşılan Sorunlar ve İpuçları	17
Bölüm 7 — Özet ve Sonraki Adımlar	19
Terimler Sözlüğü	20

□ 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 örneklerle anlatılmıştır. Amaç; ESP32'nin ne olduğunu, temel donanım yapısını, nasıl programlandığını ve hangi projelerde kullanılabileceğini kavratmaktır.

ESP32 Nedir?

ESP32; içerisinde işlemci, hafıza, kablosuz iletişim donanımı (Wi-Fi ve Bluetooth) ve çok sayıda giriş/çıkış pini barındıran, tek bir çip üzerine entegre edilmiş 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 ucuz bir elektronik karttır.

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.

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 Öne Çıkan Özellikleri

- 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: Pil ile çalışan projelerde enerji tasarrufu sağlayan uyku modları bulunur.
- Uygun fiyat: Sahip olduğu özelliklere göre oldukça ekonomik bir karttır.

1.3 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	~20	~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.

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 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.

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.

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.

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.

ESP32'yi Programlamaya Giriş

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

3.1 Arduino IDE Kurulumu (Özet)

- 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.

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);  
}
```

3.3 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.

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 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.

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.

❏ Bilgi

ESP32'nin kullanım alanları burada sayılanlarla sınırlı değildir. Aynı temel prensipler; tarım, 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.

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 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.

□ blink.ino

```
int ledPin = 2;

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

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

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.

□ 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);
  }
}
```

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.

□ 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, 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() {
}
```

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.

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.

⚠ 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.

📌 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.

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.
- 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.

Terimler Sözlüğü

Mikrodenetleyici	İşlemci, hafıza ve giriş/çıkış birimlerini tek bir çipte birleştiren, dış dünyayla doğrudan etkileşim kurabilen entegre devre.
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.
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.