



MARMARA ÜNİVERSİTESİ
TEKNOLOJİ FAKÜLTESİ
ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ
BÖLÜMÜ



ARM Tabanlı Mikroişlemciler İçin Kablosuz Yazılım Güncellenmesi

Mahmut Ahmet ÜNAL
Recep ÇETİNKAYA

Tez Danışmanı
Dr. Öğr. Üyesi Ulvi BAŞPINAR

İSTANBUL, 2019



MARMARA ÜNİVERSİTESİ
TEKNOLOJİ FAKÜLTESİ
ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ
BÖLÜMÜ



ARM Tabanlı Mikroişlemciler İçin Kablosuz Yazılım Güncellenmesi

Mahmut Ahmet ÜNAL
Recep ÇETİNKAYA

Tez Danışmanı
Dr. Öğr. Üyesi Ulvi BAŞPINAR

İSTANBUL, 2019

MARMARA ÜNİVERSİTESİ
TEKNOLOJİ FAKÜLTESİ
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ BÖLÜMÜ

Marmara Üniversitesi Teknoloji Fakültesi Elektrik-Elektronik Mühendisliği Öğrencileri Mahmut Ahmet ÜNAL ve Recep ÇETİNKAYA'nın “**ARM Tabanlı Mikroişlemciler İçin Kablosuz Yazılım Güncellenmesi**” başlıklı bitirme projesi çalışması 11.01.2019 tarihinde jüriye sunulmuş ve jüri üyeleri tarafından başarılı bulunmuştur.

Jüri Üyeleri

Dr. Öğr. Üyesi ULVİ BAŞPINAR (Danışman)

Marmara Üniversitesi (İMZA).....

Prof. Dr. Hasan ERDAL (Üye)

Marmara Üniversitesi (İMZA).....

Doç. Dr. Hayriye KORKMAZ (Üye)

Marmara Üniversitesi (İMZA).....

TEŞEKKÜR

Tez çalışmamız sırasında bize her türlü konuda yardımcı olan, bu bilimsel tezin ortaya çıkması için engin tecrübelerini bizden esirgemeyen danışman hocamız Dr. Öğr. Üyesi ULVİ BAŞPINAR'a, sonsuz teşekkürlerimizi sunarız.

Manevi desteklerinden dolayı ailelerimize teşekkürü borç biliriz.

Mahmut Ahmet ÜNAL

Recep ÇETİNKAYA

İçindekiler

SEMBOLLER VE KISALTMALAR LİSTESİ	7
ŞEKİLLER LİSTESİ	8
TABLOLAR LİSTESİ	9
ÖZET	10
ABSTRACT	11
BÖLÜM 1. GİRİŞ.....	12
1.1 Literatür Özeti.....	12
1.2 Bitirme Projesinin Amacı	12
BÖLÜM 2. TEZ İÇİN KULLANILAN TEORİ VE YÖNTEMLER	13
2.1 Mikrodenetleyici Nedir?	13
2.1.1 STM32F407VG Discovery Kartı	14
2.1.2 STM32F401RE Nucleo Kartı	15
2.2 Geliştirme Kartlarının Kullanıldığı Platformlar	16
2.2.1 Keil uVision Derleyicisi	16
2.2.2 STM32 CubeMX Programı	18
2.2.3 STM32 ST-LINK Utility Programı	19
2.2.4 Arduino IDE	19
2.3 Bluetooth Haberleşme Teknolojisi.....	20
2.3.1 Hc-05 Bluetooth Modülü Genel Özellikleri	20
2.4 Wi-Fi Haberleşme Teknolojisi	22
2.4.1 ESP8266 Wi-Fi Modülü	22
2.4.2 ESP8266 Wi-Fi Modülü Genel Özellikleri	22
2.4.3 TCP/IP Protokolü	23
2.4.4 Sunucu (Server).....	23
2.4.5 İstemci (Client)	24
2.5 Sistem Mimarisi.....	25
2.5.1 Bootloader Nedir?	25
2.5.2 Firmware Nedir?	25
2.5.3 Hafıza(Memory) Nedir?	25
2.5.4 STM32F4 Discovery Kartının Hafıza Yapısı.....	26
2.5.5 STM32F401RE Nucleo Kartının Hafıza Yapısı	27
2.5.6 Seri Veri İletişimi	27

BÖLÜM 3. SONUÇLAR	30
3.1 Bluetooth Modülü ile Yazılım Güncellenmesi	30
3.2 Wi-Fi Modülü ile Yazılım Güncellenmesi	32
KAYNAKÇA	38
ÖZGEÇMİŞ	39

SEMBOLLER VE KISALTMALAR LİSTESİ

IoT: Internet of Things(İngilizce, Nesnelerin İnterneti)

ARM: Advanced RISC Machine(İngilizce, Gelişmiş RISC Makinesi)

RISC: Reduced Instruction Set Computer(İngilizce, İndirgenmiş Komut Seti Bilgisayarı)

Wi-Fi: Wireless Fidelity(İngilizce, Kablosuz Bağlantı Alanı)

MCU: Microcontroller Unit(İngilizce, Mikrodenetleyici Birimi)

CPU: Central Processing Unit(İngilizce, Merkezi İşlem Birimi)

RAM: Random Access Memory(İngilizce, Rastgele Erişilebilir Hafıza)

ROM: Read Only Memory(İngilizce, Yalnızca Okunabilir Hafıza)

TCP: Transmission Control Protocol(İngilizce, Geçiş Kontrol Protokolü)

IP: Internet Protocol(İngilizce, İnternet Protokolü)

OSI: Open Systems Interconnection(İngilizce, Açık Sistemler Arası Bağlantı)

HTTP: Hyper Text Transfer Protocol(İngilizce, Hiper Metin Transfer Protokolü)

OTA: Over the Air(İngilizce)

HAL: Hardware Abstraction Layer(İngilizce, Donanım Soyutlama Katmanı)

ŞEKİLLER LİSTESİ

Şekil 2.1 Mikrodenetleyici İç Yapısı

Şekil 2.2 Stm32f4 Discovery Kartı

Şekil 2.3 Stm32f401re Nucleo Kartı

Şekil 2.4 Keil uVision’da geliştirme kartının seçilmesi

Şekil 2.5 Keil uVision’da oluşturulan örnek proje

Şekil 2.6 STM32 CubeMX’te Stm32f4 Discovery Kartına Ait Pin Çıkışları

Şekil 2.7 STM32 ST-LINK Utility Programıyla Görülen Geliştirme Kartının Hafızası

Şekil 2.8 Arduino IDE ile oluşturulan Proje Örneği

Şekil 2.9’da HC-05 Bluetooth Modülü

Şekil 2.10 ESP8266 Wi-Fi Modülü

Şekil 2.11 Sunucu – İstemci İlişkisi

Şekil 2.12 UART Seri Haberleşmesi

Şekil 3.1 Python Dilinde Yazılmış Arayüz

Şekil 3.2 Bluetooth ile Yazılım Güncellenmesinin Akış Diyagramı

Şekil 3.3 PHP Diliyle hazırlanan Web Sitesi

Şekil 3.4 Wi-Fi ile Yazılım Güncellenmesinin Akış Diyagramı

Şekil 3.5 Wi-Fi Yazılım Güncelleme de Bootloader ve Firmware İlişkisi

Şekil 3.6 Güncel Yazılımın Yüklenmesi

TABLÖLAR LİSTESİ

Tablo 2.1 STM32F4 Discovery Kartının Flash Hafıza Yapısı

Tablo 2.2 STM32F401RE Nucleo Kartının Flash Hafıza Yapısı

ÖZET

Sistem üreticileri, sistem işlevselliğini arttırmak, hata düzeltmeleri yapmak veya güvenlik tehditlerine yanıt vermek için sahada çalışan cihazlarının yazılımlarını güncelleme ihtiyacı hissedecektir. Bu işlem için genel olarak sistem üreticileri üretilen ve sahada kullanılan cihazları geri çağırarak gerekli düzeltmeler yapıldıktan sonra tekrar sahaya gönderir ya da güncel yazılıma sahip cihazlar bir önceki versiyonları ile değiştirilir ki bu ekonomik bir yöntem değildir. Diğer bir yöntem ise sahaya teknik bir ekip yollanarak güncelleme işleminin yapılması sağlanır. Sahadaki cihazların yazılımını güncelleme uzak mesafedeki veya hareketli gömülü sistemler ya da büyük miktarlarda üretim yapıp sahada kullanılan cihazlar için çok daha zor olabilir. Bu problemin giderilmesi sahada çalışan tüm cihazları uzak mesafeden kablosuz bir şekilde programlama ile sağlanabilir.

Tezde Gömülü aygıtları ortak bir bulut sistemi üzerinden kablosuz olarak programlayarak bu sorunun giderilmesi amaçlanmıştır. Bu tezde, gömülü aygıtlar için kablosuz güvenli yazılım güncellemesi yapılabilmesine olanak sağlayacak özelleştirilebilir bootloader mekanizması ile cihazların bir bulut sistemi üzerinden yazılımlarının güncellenmesi becerisi üzerine odaklanılmıştır.

Anahtar kelimeler: Bootloader, özelleştirilebilir bootloader, yazılım güncelleme, kabosuz yazılım güncelleme

ABSTRACT

System builders will need to update the software of their field-working devices to improve system functionality, make bug fixes, or respond to security threats. Generally, system manufacturers are called back to the field after the necessary corrections are made and the devices with the current software are replaced with their previous versions, which is not an economical method. Another method is to send a technical team to the site and update process is provided. Updating software in the field can be much more difficult for long-distance or mobile embedded systems or for large-scale devices. Eliminating this problem can be achieved by wirelessly programming all devices operating in the field at a long distance.

In this thesis, it is aimed to solve this problem by wirelessly programming embedded devices in a common cloud system. This thesis focuses on the ability of the devices to update software through a cloud system with a customizable bootloader mechanism that will enable wireless secure software updates for embedded devices.

Key words: Bootloader, customizable bootloader, firmware update, wireless firmware update

BÖLÜM 1. GİRİŞ

1.1 Literatür Özeti

Nesnelerin interneti (IoT) alanının yakın gelecekte çok yönlü büyümesi bekleniyor. Gartner'ın yaptığı araştırmaya göre 2020 yılında yaklaşık 26 milyar ürünün bir şekilde internete bağlanacağı tahmin ediliyor.[1] Akıllı bileklikler, akıllı saatler, akıllı gözlükler, akıllı t-shirtler, akıllı raketler, ev otomasyon sistemleri, akıllı arabalar ve birçoğu. Nesneler üzerine yapılan yatırımlarla, hepsi yakın bir zamanda Wi-Fi ve Bluetooth teknolojisi ile internete bağlanıyor olacak. Bu, internete bağlı cihazların daha geniş bir yelpazeye yayılmasına yol açacaktır. Sistem üreticileri, sistem işlevselliğini arttırmak, hata düzeltmeleri yapmak veya güvenlik tehditlerine yanıt vermek için sahada çalışan cihazlarının yazılımlarını güncelleme ihtiyacı hissedecektir. Bu işlem için Genel olarak sistem üreticileri üretilen ve sahada kullanılan cihazları geri çağırarak gerekli düzeltmeler yapıldıktan sonra tekrar sahaya gönderir ya da güncel yazılıma sahip cihazlar bir önceki versiyonları ile değiştirilir ki bu ekonomik bir yöntem değildir. Diğer bir yöntem ise sahaya teknik bir ekip yollanarak güncelleme işleminin yapılması sağlanır. Sahadaki cihazların yazılımını güncelleme uzak mesafedeki veya hareketli gömülü sistemler ya da büyük miktarlarda üretim yapıp sahada kullanılan cihazlar için çok daha zor olabilir. Bu problemin giderilmesi sahada çalışan tüm cihazları uzak mesafeden kablosuz bir şekilde programlama ile sağlanabilir.

1.2 Bitirme Projesinin Amacı

Bu tezde gömülü aygıtları ortak bir bulut sistemi üzerinden kablosuz olarak programlayarak literatür özeti kısmında zikredilen sorunun giderilmesi amaçlanmıştır. Bu tezde, gömülü aygıtlar için kablosuz güvenli yazılım güncellemesi yapılabilmesine olanak sağlayacak özelleştirilebilir bootloader mekanizması ile cihazların bir bulut sistemi üzerinden yazılımlarının güncellenmesi becerisi üzerine odaklanılmıştır.

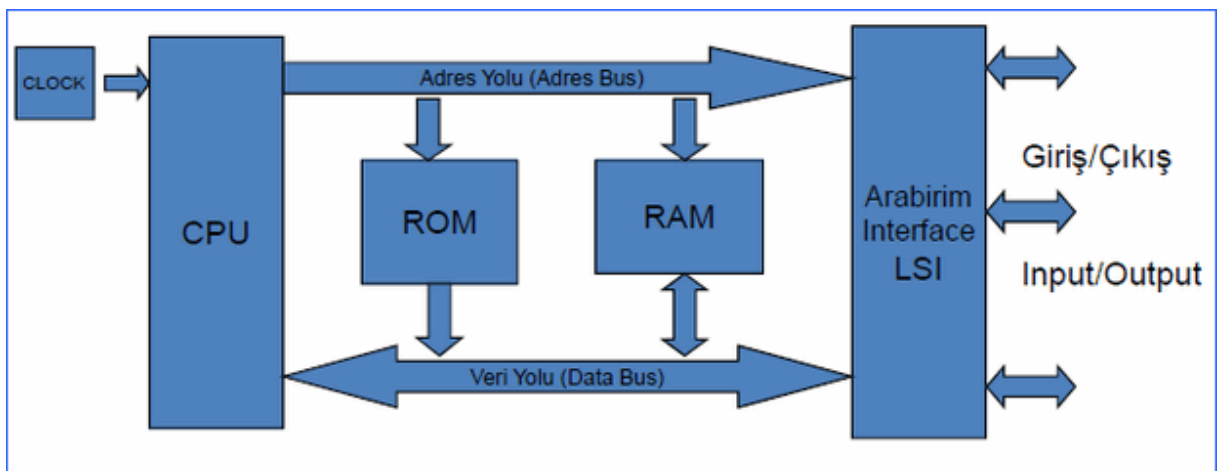
Bu tezde Arm Cortex-M4 mimarisini wi-fi ve bluetooth üzerinden programlamaya yarayan bir bootloader kodu geliştirilmiştir ve wi-fi bulut sistemi için Sunucu-istemci mekanizması oluşturulmuştur. Geliştirilen sistem Stm32 mimarisi için olmasına rağmen geliştirilen mekanizma gömülü sistemler alanında kullanılan diğer MCU mimarileri için de kullanılabilir.

BÖLÜM 2. TEZ İÇİN KULLANILAN TEORİ VE YÖNTEMLER

2.1 Mikrodenetleyici Nedir?

Mikrodenetleyici bir bakımda bir bilgisayardır. Her ne kadar bir klavyesi, monitörü, kasası ve bunun gibi çevre birimleri olmasa da bir bilgisayarın yaptığı her şeyi yapabilir. Örneğin her bilgisayarın bir merkezi işlem ünitesi vardır ve bu ünite makine kodlarını bizim anlayabileceğimiz karakterlere dönüştürür, programları yorumlar, işler, düzenler, bilgisayarın çeşitli birimleriyle irtibat kurar. Ve bu işlemleri yaparken bazı değişkenleri ve geçici olarak elde ettiği bilgileri sakladığı bir rasgele erişimli hafızaya ihtiyaç duyar. Ayrıca bilgisayarların dış dünyayla bilgi alış ve verişlerinde kullandıkları bazı giriş ve çıkış üniteleri bulunmaktadır. Örnek olarak fare ve klavye giriş yaptığımız elemanlara, monitör ve yazıcı çıkış aldığımız elemanlara birer örnektir. Bilgilerimizi kaydettiğimiz hard diskler ise hem giriş hem de çıkış elemanı olarak çalışmaktadırlar. Aynen bilgisayarda olduğu gibi mikrodenetleyiciye de fare ve klavye gibi çevre elemanlarının işlemlerini nispeten de olsa yerine getirecek elemanlar ekleyerek küçük bir bilgisayar gibi kullanmamız mümkündür. Bir mikrodenetleyici genel olarak aşağıdaki birimlerden oluşur: [2]

1. CPU
2. RAM
3. EPROM/PROM/ROM (Erasable Programmable Read Only Memory-Silindir, yazılır sadece okunur bellek)
4. I/O (giriş/çıkış-input/output)
5. Timers (zamanlayıcılar)
6. Interrupt (kesmeler)



Şekil 2.1 Mikrodenetleyici İç Yapısı

2.1.1 STM32F407VG Discovery Kartı

Mikrodenetleyici kontrol kartı olarak STM32F4 Discovery kartı kullanılmıştır. ST firmasının üretmiş olduğu kontrol kartı ARM Cortex - M4 serisi STM32F407VG işlemciye sahiptir. Cortex - M4 serisi 32 bit işlem yapabilen yüksek performans sağlayan yeni nesil ARM tabanlı işlemcidir.

ARM mimarisi pek çok gömülü tasarımda kullanılan 32-bit işlemci mimarisidir. Güç tasarruf özelliklerinden dolayı, ARM işlemciler mobil elektronik gibi düşük güç tüketiminin kritik bir parametre olduğu pazarda en fazla tercih edilen mikrodenetleyicilerdir. Günümüzde ARM işlemci ailesi tüm yeryüzündeki 32-bit gömülü işlemcilerin %75' ini oluşturmaktadır. ARM işlemciler taşınabilir cihazlardan (PDA, cep telefonu, medya oynatıcılar, avuç içi oyun üniteleri, hesap makineleri v.b.) bilgisayar parçalarına kadar (disk sürücüler, masaüstü modemler v.b.) tüketici elektroniğinin her alanında yoğun olarak kullanılmaktadır.

Gömülü tasarım uygulamalarında en sık kullanılan ARM mimarisi komut kümeleri, 32-bit'lik ARM ve 16-bit'lik Thumb komut kümeleridir. Her Thumb komutunun bir ARM komut karşılığı vardır fakat bunun tersi doğru değildir. Bu sorun bu iki komut kümesinin bir arada çalışmasının mümkün olması ile aşılmıştır. Bu sayede, 16-bit'lik komut kümesinin daha az bellek kullanımı ve 32-bit'lik komut kümesinin üstün işlevsellik özellikleri bir arada kullanılabilmektedir.

Şekil 2.2’de kullanılan mikrodenetleyici kontrol kartı görülmektedir. ST Microelectronics firmasının üretmiş olduğu kart yeni nesil ARM Cortex-M4 serisi mikrodenetleyicisi kullanmaktadır [3].



Şekil 2.2 Stm32f4 Discovery Kartı

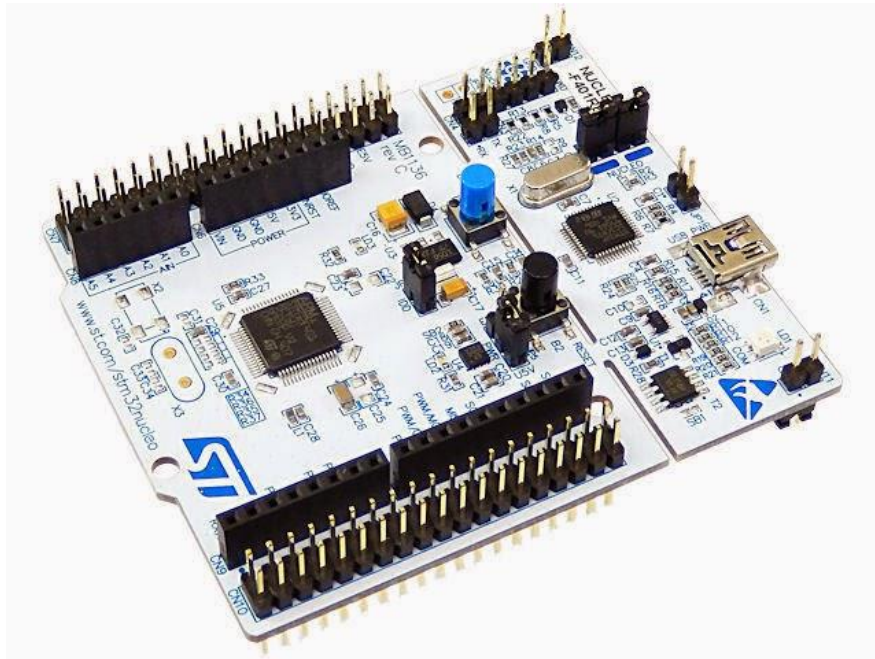
Kullanılan mikrodnetleyici kontrol kartı ařağıdaki özellikleri içermektedir:

- 168 MHz/210 DMIPS Cortex-M4 CPU tek çevirim DSP MAC birimi
- 1 MB flash hafızası
- 192 KB RAM bellek
- 140 adet giriş/çıkış (I/O) birimi
- 24 adet ADC birimi - 14 adet zamanlayıcı
- USB Host ve Slave özelliğı
- Ethernet kontrolü
- Dahili hata ayıklama özelliğı
- 3 eksen hareket sensörü - Çok yönlü dijital mikrofon
- Hoparlör sürücü entegre edilmiş DAC birimi [3]

2.1.2 STM32F401RE Nucleo Kartı

Mikrodnetleyici kontrol kartı olarak STM32F4 Discovery kartının yanında STM32F401RE Nucleo kartı da kullanılmıştır. ST firmasının üretmiş olduğı kontrol kartı ARM Cortex - M4 serisi STM32F401RET6işlemciye sahiptir. Cortex - M4 serisi 32 bit işlem yapabilen yüksek performans sağlayan yeni nesil ARM tabanlı işlemcidir.

Şekil 2.3’de kullanılan mikrodnetleyici kontrol kartı görölmektedir. ST Microelectronics firmasının üretmiş olduğı kart yeni nesil ARM Cortex-M4 serisi mikrodnetleyicisi kullanmaktadır.



Şekil 2.3 Stm32f401re Nucleo Kartı

Kullanılan mikrodnetleyici kontrol kartı ařağıdaki özellikleri içermektedir:

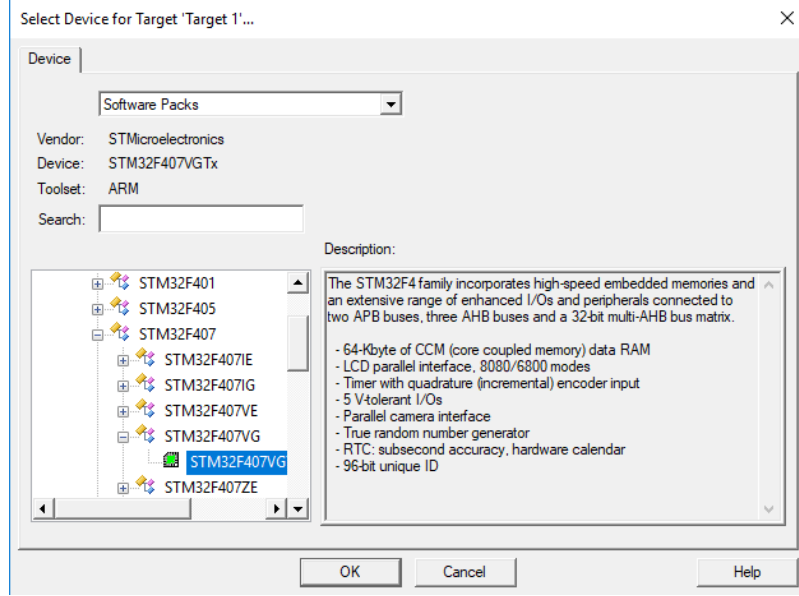
- 84MHz'de FPU'lu ARM Cortex-M4 CPU
- 512 KByte Flash
- LQFP64 paketinde STM32 mikrodnetleyici
- Vcore mantık kaynağı oluşturmak için harici SMPS (sadece '-P' eklenmiş boardlarda kullanılabilir)
- Arduino ile paylaşılan 1 kullanıcı LED'i
- 1 kullanıcı ve 1 sıfırlama butonu
- 32.768 kHz LSE kristal osilatör[4].

2.2 Geliştirme Kartlarının Kullanıldığı Platformlar

2. Bölümün ilgili kısımlarında bahsedilen geliştirici kartlarının veya tezde kullanılan modüllerin programlanmasında veya ilgili işlemlerin yapılmasında kullanılan platform veya uygulamalar bu kısımda tanıtılmıştır.

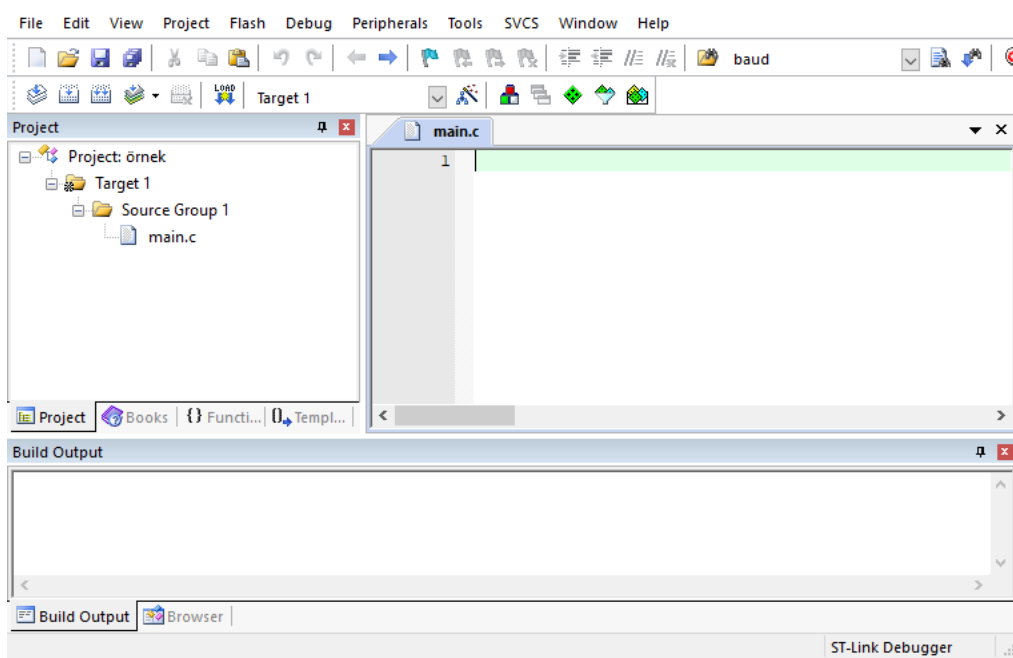
2.2.1 Keil uVision Derleyicisi

Geliştirme kartları için yazılımın oluşturulup derlenildiği platform Keil'in uVision derleyicisidir. Derleyicinin benzetim özelliğinin olması hiçbir donanım gerektirmeden yazılan kodları deneme imkânı sunmaktadır. Ayrıca donanım üzerindeki programda gerçek zamanlı olarak debug edebilme imkânı sunmaktadır. Bu platformda yazılım geliştirmek için içerisinde desteklenen kart veya işlemci seçilir ve gerekli konfigürasyonlar yapıldıktan sonra program yazılıp derlenir ve oluşturulan hex uzantılı makine kodu dosyasını program sol üstte bulunan "Download" butonuyla geliştirme kartına gömülür. Şekil 2.4'te Keil uVision programında geliştirme yapılacak kartın seçilebileceği platform gösterilmiştir[5].



Şekil 2.4 Keil uVision’da geliştirme kartının seçilmesi

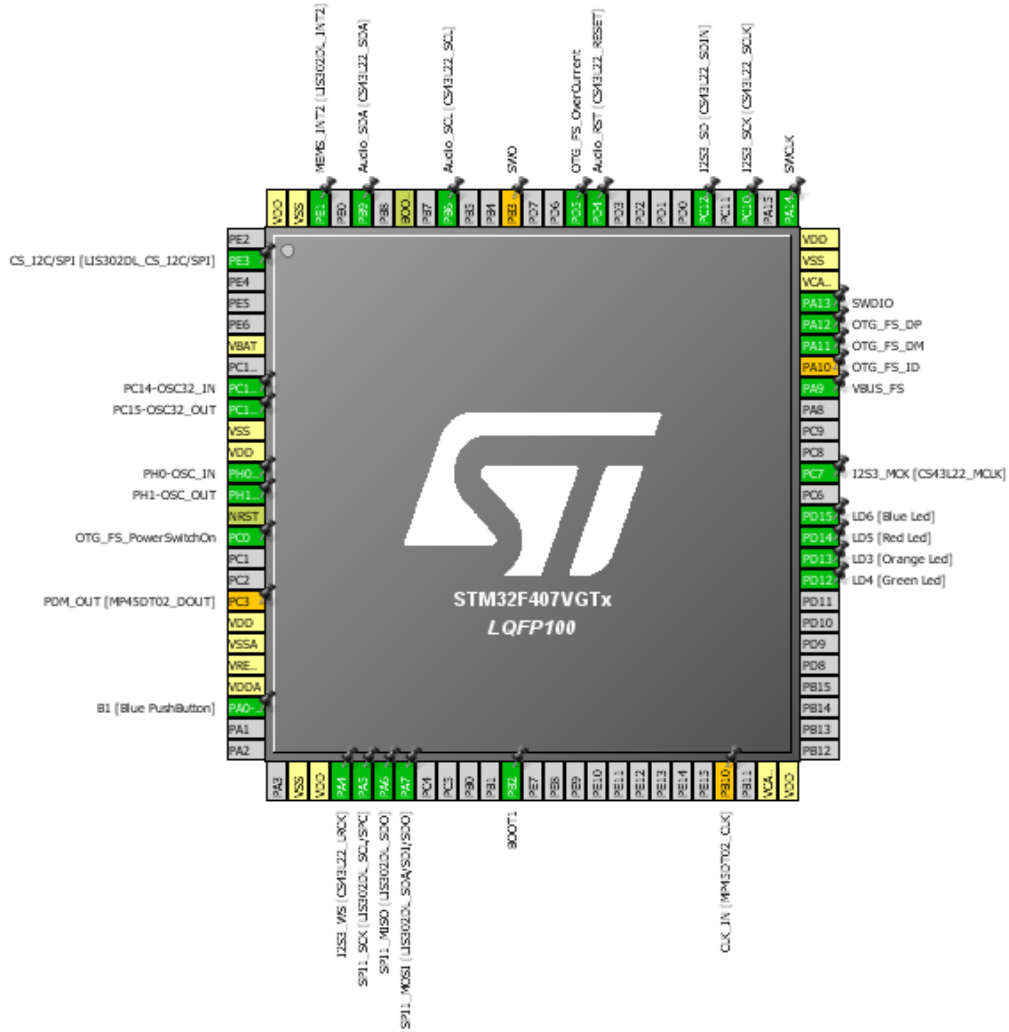
Geliştirme kartı seçilir, gerekli konfigürasyonlar ve Target menüsünden gerekli ayarlamalar yapılır. Program tarafından oluşturulan projeye gerekli kodlar ve yüklemeler yapılır. Şekil 2.5’te Keil uVision programında oluşturulan boş bir proje gösterilmektedir.



Şekil 2.5 Keil uVision’da oluşturulan örnek proje

2.2.2 STM32 CubeMX Programı

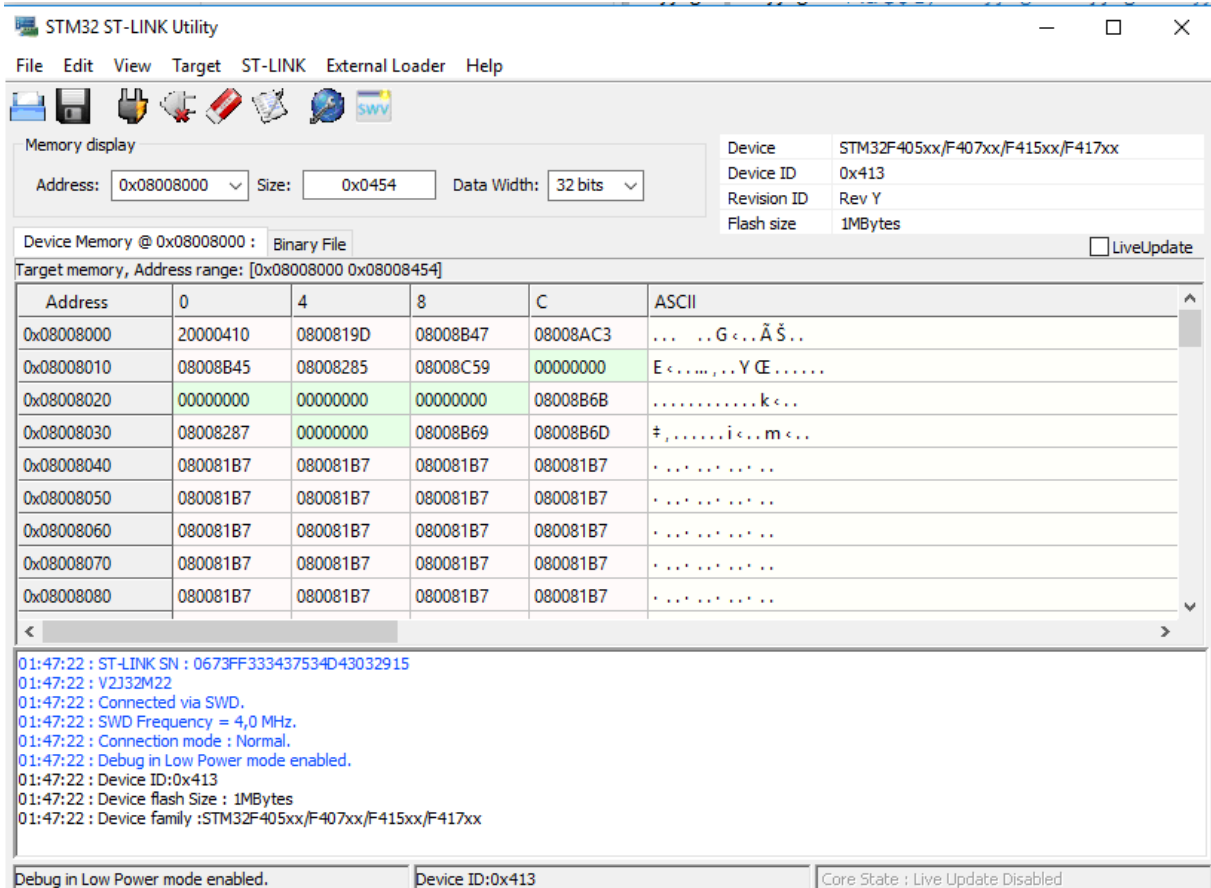
ST firmasının geliştirmiş olduğu ve seçilen geliştirme kartının pin çıkışlarının program bünyesinde hazır olarak sunulduğu platformdur. Gerekli pin çıkışlarının izin verilen birimler için konfigüre edildiği ve yapılan ayarlamaları C dilinde yazılmış kodunu oluşturan bir uygulamadır. Bu tezde STM32 CubeMX programından faydalanılmıştır. Şekil 2.6’da CubeMX programı tarafından STM32F4 Discovery kartına ait oluşturulan, pin çıkışları gösterilmiştir.



Şekil 2.6 STM32 CubeMX’te Stm32f4 Discovery Kartına Ait Pin Çıkışları

2.2.3 STM32 ST-LINK Utility Programı

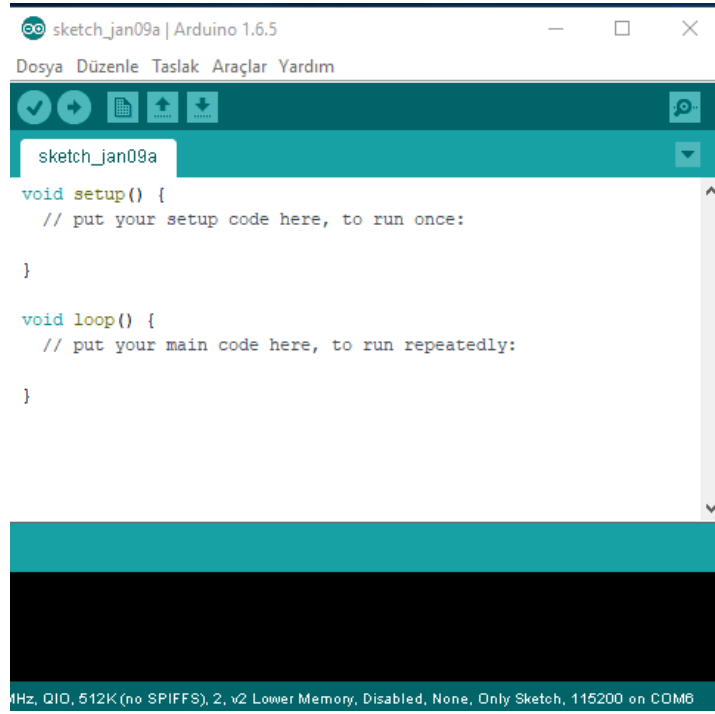
ST firması tarafından geliştirilen bu program bilgisayara bağlı bulunan geliştirme kartının hafızasını göstermektedir. Bu program ile kartın ilgili sektöründe bulunan veri görülebilir ve silinebilir. Sadece ilgili sektör silinebildiği gibi tüm hafıza da silinebilir. Bu tezde STM32 ST-LINK Utility programından faydalanılmıştır. Şekil 2.7’de STM32 ST-LINK Utility programında geliştirme kartının hafızası gösterilmiştir.



Şekil 2.7 STM32 ST-LINK Utility Programıyla Görülen Geliştirme Kartının Hafızası

2.2.4 Arduino IDE

Arduino IDE, kod yazmak ve kartlara yüklemek için kolay kullanımlı açık kaynak bir yazılımdır. Arduino yazılımı (IDE) "Processing", Arduino programlama dili ise "Wiring" tabanlıdır. Bu tezde Arduino IDE’de bulunan ESP8266 kütüphanesi kullanılarak ESP8266 Wi-fi modülü programlanmıştır. Şekil 2.8’de Arduino IDE’de oluşturulan boş bir proje örneği gösterilmiştir[6].



Şekil 2.8 Arduino IDE ile oluşturulan Proje Örneği

2.3 Bluetooth Haberleşme Teknolojisi

Bluetooth kablosuz iletişim teknolojisi, küçük boyutlu cihazlar için, mobil bilgisayarları, mobil telefonları, elde taşınabilen küçük cihazları birbirlerine ve internete bağlayabilen dünya çapında belirlenmiş düşük maliyetli bir radyo iletişim çözümüdür.

Tipik özellikteki bir Bluetooth 20-100 metrede (cihazın sınıfına göre mesafe değişebilmektedir) veri gönderme hızı maksimum 723Kbps(Kilobit per second) iletim oranı sağlar.

2.3.1 Hc-05 Bluetooth Modülü Genel Özellikleri

Bluetooth modülü devletlerce insanların kullanımına açık frekans bandında çalışmaktadır. Genel olarak bazı özelliklere sahiptir bu özelliklere aşağıdaki gibi sıralanmıştır.

- I. 2,4 GHz haberleşme frekansı
- II. Hassasiyet: ≤ -80 dBm(desibel miliwatt)
- III. Çıkış gücü: $\leq +4$ dBm
- IV. Asenkron hız: 2,1 Mbps(Megabit per second) / 160 Kbps
- V. Senkron hız: 1 Mbps / 1 Mbps
- VI. Çalışma gerilimi: 1,8 - 3,6 V
- VII. Akım: 50 mA(mili Amper)
- VIII. Kimlik doğrulama ve şifreleme özelliğine sahiptir

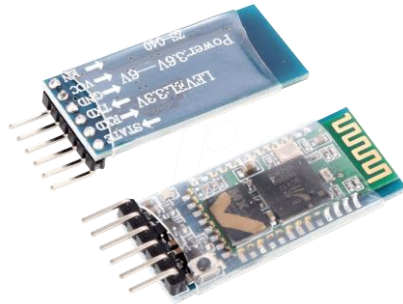
- IX. UART(Universal Asynchronous Receiver/Transmitter), IC2(Inter-Integrated Circuit), USB 1.2 iletişim protokollerine sahip
- X. Master/Slave (Yönetici/köle) ya da Loopback Modları
- XI. AT(Attention) komutları ile çalışabilme[7].

HC-05 bluetooth modülü, seri haberleşme protokolüne göre çalışmaktadır. Alıcı Rx(receiver) ucu veriyi alan uç iken, verici Tx (tranceiver) ucu veriyi gönderen uçtur. Bluetooth modülüne bağlanmak için modülün bir isim ve şifresi bulunmaktadır ama ilk kullanımda bir engel oluşmaması için bu şifre aktif değildir. Bluetooth bağlantısı yapılırken modülün adı HC-05, şifresi 1234 veya 0000 olmaktadır bunun yanı sıra baud hızı default olarak 9600 ayarlanmış şekilde hazır gelir.

Kablosuz haberleşme bağlantı yapabilmek için bir modülün yönetici, diğer modülün ise yönetilen modda olması gerekmektedir. Hc-06 bluetooth modülü sadece yönetilen modda çalışmaktadır bundan dolayı bu modül kullanılmamıştır. Bu yüzden bu uygulamada hem yöneten hem de yönetilebilen modda çalışabilen Hc-05 bluetooth modülü kullanılmıştır.

Hc-05 bluetooth modülünün en önemli avantajlarından birisi de kolay şekilde yöneten moda geçebilmesidir. Yöneten veya yönetilen moda geçmek ve diğer ayarlamaları yapmak için AT komutları kullanılmaktadır. AT komutlarını kullanmak için HC-05 bluetooth modülünü konfigürasyon moduna geçirilmesi gerekmektedir. Bunun için modülün 5V bağlantısının yapıldığı sırada modül üzerindeki butonu basılı tutulur. Konfigürasyon moduna girildiğini, modül üzerinde bulunan ve yanmakta olan LED'in yanıp sönme sıklığından anlaşılabilmektedir. Eğer 3'er saniyelik aralıklarla yanıp sönüyorsa, modül konfigürasyon modundadır.

LED'in yanıp sönmesi sıklığı bu şekilde ise bu bluetooth modülün iletişim modunda olduğunu göstermektedir. Modül iletişim modunda iken diğer bluetooth cihazları tarafından yapılan taramalarda listelenmektedir. İletişim modunda bir cihaz bluetooth modülüne bağlandığında ise LED, 3 saniyede bir kere kısa yanıp sönmektedir. Şekil 2.9'da HC-05 Bluetooth modülü gösterilmiştir[8].



Şekil 2.9'da HC-05 Bluetooth Modülü

2.4 Wi-Fi Haberleşme Teknolojisi

IEEE 802.11 standardı olan Wi-Fi teknolojisi hemen hemen tüm akıllı cihazlarda, bilgisayarlarda ve mobil sistemlerde kullanılmaktadır. Kapsama alanı diğer teknolojilere oranla daha fazla olmasına karşın ağı yeni aygıtın eklenip çıkarılması daha zordur ve bu durum tüm sistemi etkilemektedir. Altyapı sistemi geniştir ve büyük boyutlu veri aktarımı yapılabilir[9].

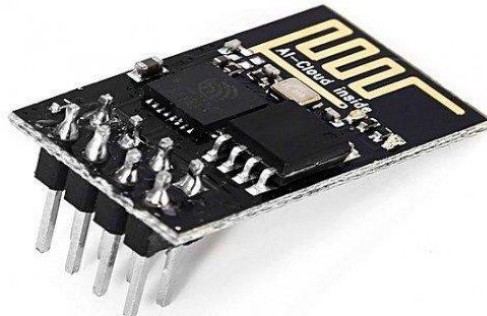
2.4.1 ESP8266 Wi-Fi Modülü

ESP8266, Espressif Systems tarafından üretilen, Wi-Fi teknolojisine sahip, yüksek performanslı, küçük boyutlu ve düşük güç tüketimli bir yongada sistemdir. Wi-Fi teknolojisinin yeteneklerini başka cihazlara kazandırmak ya da bu yetenekleri kullanan bağımsız düşük maliyetli uygulamalar yapmak için uygundur[10].

2.4.2 ESP8266 Wi-Fi Modülü Genel Özellikleri

ESP8266 Wi-Fi modülü, seri haberleşme protokolüne göre çalışmaktadır. Alıcı Rx(receiver) ucu veriyi alan uç iken, verici Tx (transceiver) ucu veriyi gönderen uçtur. AT komutları ile modülün yazılımının güncel olup olmaması, modülün çalışır durumda olması gibi durumlar kontrol edilir. Ayrıca modülün kablosuz ağı bağlanması veya balantısının kesilmesi işlemleri yapılabilir. AT komutları ile modülün IP ve MAC adresleri gibi bilgileri de öğrenilebilir.

Modüle AT komutları gönderildiğinde bize cevap verirken üzerinde bulunan mavi LED yanıp söner ve enerji verildiğinde ise üzerinde bulunan kırmızı LED yanar. Bu tezde ESP8266 Wi-Fi modülü Arduino IDE kullanılarak programlanmıştır. Şekil 2.10'da ESP8266 Wi-Fi modülü gösterilmiştir.



Şekil 2.10 ESP8266 Wi-Fi Modülü

ESP8266'nın genel özellikleri:

- 802.11 b/g/n desteği
- Düşük güç tüketimine sahip 32 bit'lik dahili işlemci
- Dahili TCP/IP protokol yığını
- WiFi 2.4 GHz, WPA/WPA2 desteği
- STA/AP/STA+AP modlarında çalışma
- Uyanma ve veri paketi alma süresi 2ms'den az
- Bekleme durumunda güç tüketimi 1mW'tan az
- Uyku durumunda 10uA'den az akım tüketme

2.4.3 TCP/IP Protokolü

Farklı makineler arasında bağlantı kurulması işlemine internetworking adı verilir. Farklı tipte makine ağlarının entegrasyonu için standart bir protokol seti kullanılır. Genel olarak kullanılan internetworking mimarisi TCP/IP (Transmission control protocol/ Internet protokol) dir.

TCP/IP bu standartlaşmış internetworking mimarilerinden bir tanesidir. OSI standardına göre Network Tabakasının üzerindeki tabakaları standartlaştırılır. TCP/IP bu özelliği ile veri paketlerinin adreslendirilmesi ve yönlendirilmesi için standart bir yöntem sunar.

Network ortamında makinelerin birbirleriyle konuşması için tasarlanmış bir protokoldür[12]

2.4.4 Sunucu (Server)

Sunucu, web sitesine ait dosyaları barındıran ve bu dosyaların internette paylaşılmasına olanak tanıyan bilgisayar olarak tanımlanabilir.

Sunucularda birer bilgisayardır. Ev kullanıcılarının bilgisayarlarından farklı olarak bazı özellikler içerirler. Bu özellikler ise;

- Sunucularda programlama dillerini okuyup, çalıştırabilecek yazılımlar kuruludur.
- Sunucular kullanıcılar ile HTTP veya HTTPS adı verilen transfer protokelleri ile iletişim kurarlar bunun içinde sunucuda yüklü yazılımlar vardır.
- Sunucular datacenter (veri merkezleri) adı verilen yerlerde tutulur. Buralarda sunucular 7/24 açık kalarak web sitesi ulaşmak istenildiğinde her zaman ulaşılmasını sağlarlar. Datacenterlarda birçok sunucu bulunduğundan ağ yapısında ev kullanıcılarının kullanıldığı ağlardan farklıdır.

- Datacenterlar ađ yapılarında 100 Mbit – 1 Gbit arası internet hızlarını kullanırlar. Bu hız bilgisayarlar arasında eşit bir şekilde paylaşılır. Aynı şekilde ađ ve diđer donanım kaynakları bilgisayarlar arasında paylaşılır.
- Genel olarak bakılırsa sunucular iine atılan web sitesi dosyalarını iřlerler ve kullanıcılara web sitesini sunarlar. Kullanıcıların web sitesi üzerinde gerekleřtirdiđi iřlemlere gre gerekirse yeniden iřlem yapıp yeni sonular verirler[13].

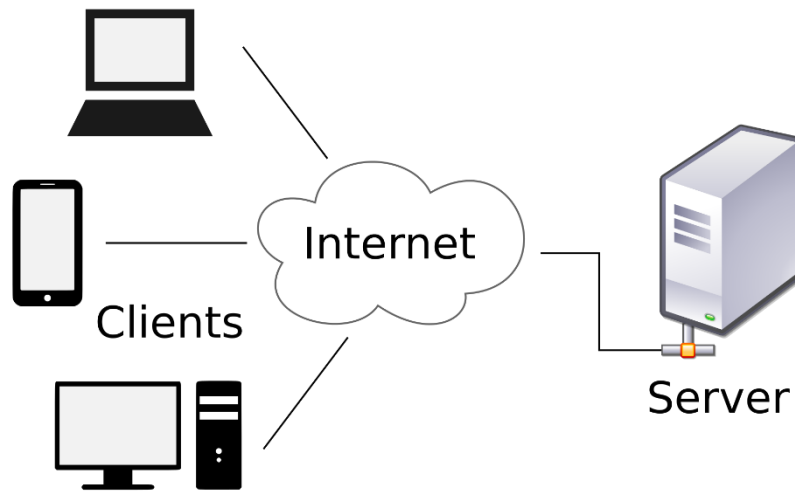
2.4.5 İstemci (Client)

Sunuculara dosyaları aması ve belli iřlemleri gerekleřtirmesi iin HTTP ve HTTPS protokelleri üzerinden istek gnderen kullanıcılara istemci denir. İstemci aslında kullanıcıdır.

Herhangi bir web sitesine girildiđi anda belli iřlemler bařlar.

- HTTP olarak GET isteđinde bulunulur. Girilen sayfayı getirmesi iin sunucuya bir mesaj gnderilir.
- Sunucu iřlemi gerekleřtirmeye bařlar. Eđer dosyayı bulursa sonucu 200 olarak dndrr. Bu sayfanın bulunduđu ve yklenmekte olduđu anlamına gelir. Bunun gibi farklı HTTP Request Kodları mevcuttur. Bu sayede sunucu ve isteci iletiřim kurar.
- Sunucu 200 cevabını verdikten sonra istenilen sayfadaki kodları iřlemeye bařlar
- Kodların iřlenmesinden sonra oluřan sonucu kullanıcı tarayıcı yardımı ile grr.
- Kodlar üzerinde kullanıcıya tekrar iřlem yapma seenekleri sunuluyorsa (ye olmak, yorum yapmak vb.) sunucu bu iřlemleri de gerekleřtirerek yeni sonular dndrmeye devam eder.

Btn bu iřlem srecini bařlatan sunucuya istek yollayan istemcidir. řekil 2.11’de sunucu ve istemci arasındaki iliřki gsterilmiřtir[13].



řekil 2.11 Sunucu – İstemci İliřkisi

2.5 Sistem Mimarisi

Tezin bu bölümünde kablosuz güncelleme işlemi oluşturmak için tasarlanan sistemin yapısından bahsedilmiştir.

2.5.1 Bootloader Nedir?

Bootloader, MCU flash hafızasında veya ROM'da depolanan küçük bir kod parçasıdır. Uygulama yükleyicisi olarak işlev görür ve gerektiğinde uygulamaları güncellemek için bir mekanizmadır. Bu tezde kablosuz yazılım güncellenebilmesi için teze ait bootloader kodu yazılmıştır. Bu bootloader kodu sayesinde yazılım güncellenebilmektedir.

2.5.2 Firmware Nedir?

Cihaz üzerinde çalışan belirli spesifik görevleri olan ana yazılımdır. Sistem işlevselliğini artırma, hata düzeltmeleri yapma ve yeni versiyon güncellemesi bu kısımda yapılan işlemlerdir. Bu tezde geliştirilen firmware geliştirme kartlarının 2. ve sonrasında bulunan sektörlerinde depolanır. Yazılan bootloader koduyla karta enerji verildiğinde otomatik olarak firmware kısmı çalıştırılır.

2.5.3 Hafıza(Memory) Nedir?

İşlemcinin çalıştırdığı programlar ve programlara ait bilgiler hafıza üzerinde saklanmaktadır. Hafıza geçici bir depolama alanıdır. Hafıza üzerindeki bilgiler güç kesildiği anda kaybolmaktadır. Bu nedenle bilgisayarlarda verileri daha uzun süreli ve kalıcı olarak saklamak için farklı birimler mevcuttur.

Hafızanın farklı çeşitleri vardır. Hafıza çeşitlerini; RAM'den veri okumak ya da RAM'e veri yazmak için kullanılan protokoller belirlemektedir. Genel anlamda belleklerin DRAM ve SRAM olmak üzere iki ana çeşidi vardır.

DRAM: Bu teknolojiye transistör ve kondansatörler birlikte kullanılmaktadır. Veriler kondansatörlerde tutulmakta, okuma ve yazma işlemleri için de transistörler kullanılmaktadır. Kondansatörler yapıları gereği çok hızlı enerji harcamaktadır. Bu nedenle üzerlerinde bulunan verileri koruyabilmek için enerjisinin sürekli yenilenmesi gerekmektedir.

SRAM: Bu tip RAM'lerde veriler yüklendikten sonra sabit kalmaktadır. Bu nedenle enerjisinin sürekli yenilenmesi gerekmemektedir. SRAM'ler; DRAM'den daha hızlı ve daha güvenilirdir ancak DRAM'ler kadar yaygın değildir.

EPROM: EPROM (Erasable Programmable Read Only Memory); Bu tip belleklerde elektrik kesintisinde bilgi kaybolmamaktadır. EPROM’ların içinde bulunan veriler temizlenerek yeniden programlanabilir ancak sadece okunabilir şekilde çalışmaktadır.

FLASH MEMORY: Elektrik enerjisinin olmadığı durumlarda verileri hafızasında tutabilen yapıya sahiptir. Flash Memory’ler EPROM’un bir çeşididir. Tek farkı Flash Memory’lere veri yazma işlemi de yapılmaktadır[14].

2.5.4 STM32F4 Discovery Kartının Hafıza Yapısı

STM32F4 Discovery geliştirme kartının genel özellikleri tezin ilgili kısımlarında verilmiştir. Bu kısımda kartın flash hafızasının sektör bilgileri Tablo 2.1’de gösterilmektedir. Bu tez kapsamında yazılan özelleştirilebilir bootloader kodu için 0 ve 1. Sektörler tarafımızdan rezerve edilmiş olup ilgili sektörlerde saklanmaktadır[15].

Block	Name	Block base addresses	Size
Main memory	Sector 0	0x0800 0000 - 0x0800 3FFF	16 Kbytes
	Sector 1	0x0800 4000 - 0x0800 7FFF	16 Kbytes
	Sector 2	0x0800 8000 - 0x0800 BFFF	16 Kbytes
	Sector 3	0x0800 C000 - 0x0800 FFFF	16 Kbytes
	Sector 4	0x0801 0000 - 0x0801 FFFF	64 Kbytes
	Sector 5	0x0802 0000 - 0x0803 FFFF	128 Kbytes
	Sector 6	0x0804 0000 - 0x0805 FFFF	128 Kbytes
	.	.	.
	Sector 11	0x080E 0000 - 0x080F FFFF	128 Kbytes
System memory		0x1FFF 0000 - 0x1FFF 77FF	30 Kbytes
OTP area		0x1FFF 7800 - 0x1FFF 7A0F	528 bytes
Option bytes		0x1FFF C000 - 0x1FFF C00F	16 bytes

Tablo 2.1 STM32F4 Discovery Kartının Flash Hafıza Yapısı

2.5.5 STM32F401RE Nucleo Kartının Hafıza Yapısı

STM32F401RE Nucleo geliştirme kartının genel özellikleri tezin ilgili kısımlarında verilmiştir. Bu kısımda kartın flash hafızasının sektör bilgileri Tablo 2.2’de gösterilmektedir. Bu tez kapsamında yazılan özelleştirilebilir bootloader kodu için 0 ve 1. Sektörler tarafımızdan rezerve edilmiş olup ilgili sektörlerde saklanmaktadır[15].

Block	Name	Block base addresses	Size
Main memory	Sector 0	0x0800 0000 - 0x0800 3FFF	16 Kbytes
	Sector 1	0x0800 4000 - 0x0800 7FFF	16 Kbytes
	Sector 2	0x0800 8000 - 0x0800 BFFF	16 Kbytes
	Sector 3	0x0800 C000 - 0x0800 FFFF	16 Kbytes
	Sector 4	0x0801 0000 - 0x0801 FFFF	64 Kbytes
	Sector 5	0x0802 0000 - 0x0803 FFFF	128 Kbytes
	Sector 6	0x0804 0000 - 0x0805 FFFF	128 Kbytes
	Sector 7	0x0806 0000 - 0x0807 FFFF	128 Kbytes
System memory		0x1FFF 0000 - 0x1FFF 77FF	30 Kbytes
OTP area		0x1FFF 7800 - 0x1FFF 7A0F	528 bytes
Option bytes		0x1FFF C000 - 0x1FFF C00F	16 bytes

Tablo 2.2 STM32F401RE Nucleo Kartının Flash Hafıza Yapısı

2.5.6 Seri Veri İletişimi

Seri veri iletimi, bir veri içindeki bitlerin aynı hat üzerinden ard arda gönderilmesidir. Bilgisayar ağlarında kullanılan iletişim seri iletişimidir. Seri veri iletiminde, bir kerede bir karakterin sadece bir biti iletilir. Alıcı makine, doğru haberleşme için karakter uzunluğunu, başla-bitir (start-stop) bitlerini ve iletim hızını bilmek zorundadır. Paralel veri iletiminde, bir karakterin tüm bitleri aynı anda iletildiği için başla-bitir bitlerine ihtiyaç yoktur. Dolayısıyla doğruluğu daha yüksektir. Paralel veri iletimi, bilginin tüm bitlerinin aynı anda iletimi sebebiyle çok hızlıdır[16].

Seri iletişim asenkron seri iletişim ve senkron seri iletişim olmak üzere iki çeşittir:

Asenkron seri iletişim:

Herhangi bir zamanda veri gönderilebilir. Veri gönderilmediği zaman hat boşta kalır. Senkron seri iletişimden daha yavaştır. Her veri grubu ayrı olarak gönderilir. Gönderilen veri bir anda bir karakter olacak şekilde hatta bırakılır. Karakterin başına başlangıç ve sonunda hata sezmek

için başka bir bit eklenir. Başlangıç için başla biti (0), veri iletişimini sonlandırmak için ise dur biti (1) kullanılır.

Senkron seri iletişim:

Senkron iletişim alıcı ve vericinin eş zamanlı çalışması anlamına gelir. Önce gönderici taraf belirli bir karakter gönderir. Bu her iki tarafça bilinen iletişime başlama karakteridir. Alıcı taraf bu karakteri okursa iletişim kurulur. Verici bilgileri gönderir. Transfer işlemi veri bloku tamamlanana ya da alıcı verici arasındaki eşleme kayboluncaya kadar devam eder. Bu tezde bir seri iletişim protokolü olan UART seri haberleşmesi kullanılmıştır[16].

UART Nedir

Universal Asynchronous Receiver/Transmitter (UART) bir seri haberleşme protokolüdür. UART byte olarak aldığı dataları bir diğer cihaza asenkron olarak iletir. RS-232,RS-422 ya da RS-485 haberleşme standartlarını destekler. Senkron haberleşme de gerçekleştirebilir. (USART) Temel olarak 4 kablo ile kontrol edilir. Bunlar Gnd, Vcc, TX(transmitter) ve RX(receiver) dir[16].

Önemli UART Sinyalleri

RXD: Haberleşmede verilerin alındığını gösteren sinyaldir.

TXD: Haberleşmede verilerin gönderildiğini gösteren sinyaldir. Cihazların TXD ve RXD bacakları bir birlerine terslenerek bağlanırlar. Çünkü birisi TXD ile gönderirken karşıdaki bunu RXD ile alacaktır.

RTS(Ready To Send): Veri almaya hazırım, veri göndermeye başlayabilirsin sinyalidir.

CTS(Clear To Send): Veri gönderebilirim sinyalidir. CTS ve RTS bacakları birbirlerine terslenerek bağlanırlar[16].

UART Özellikleri

Bir seferde 8-bit ya da 9-bit veri transferi yapılabilir.

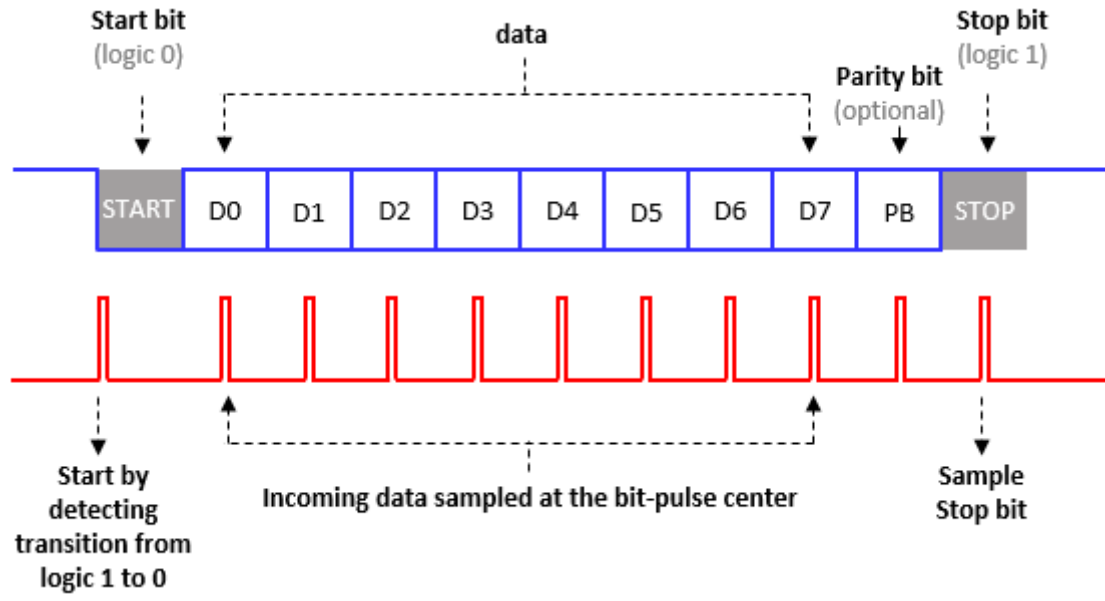
Gönderim hızı BAUD ayarı ile kontrol edilebilir. ÖRN: 115200 bps(bit per second) saniyede 115200 bit gönderilecek anlamını taşır.

Verinin doğruluğunu ve bozulmaya uğrayıp uğramadığını test etmek için parity-bit kullanılır.

Hardware Flow Control(Donanım akış kontrolü) ile iki cihaz arasında aynı hız elde edilebilir. Paket kaybı azaltılabilir.

Handshaking yöntemi ile cihazlar birbirlerine veri göndereceklerini önceden haber verebilirler. Bununla birlikte daha güvenli bir iletim sağlanmış olur.

İşlemci üzerinde bağlı bulunan pine interrupt bağlanabilir. Şekil 2.12’de UART seri haberleşmesi gösterilmiştir[16].



Şekil 2.12 UART Seri Haberleşmesi

BÖLÜM 3. SONUÇLAR

3.1 Bluetooth Modülü ile Yazılım Güncellenmesi

Bu tezde bluetooth modülü kullanılarak geliştirme kartlarının yazılım güncellenmesi yapılmıştır. Bunun için öncelikle karta enerji verildiğinde kullanıcı butonu yardımıyla bootloader moduna geçilir, bu işlem karta enerji verdikten sonra istenilen zamanda yapılabilir.

Bilgisayarla bluetooth modülüne bağlandıktan sonra bilgisayarda açılan port ile bilgisayar ve kart arasındaki iletişim sağlanır.

Python dilinde yazılmış bir arayüz sayesinde bootloader modundayken kartın versiyon numarasını, cihp numarasını, hafızanın silinmesini, ilgili sektörün silinmesini ve yazılım güncellenmesi işlemleri yapılabilir. Şekil 3.1’de python dilinde yazılmış arayüz gösterilmiştir.

Arayüz kodu : [www.github.com/Officializm/Bluetooth_User_Interface](https://github.com/Officializm/Bluetooth_User_Interface)

Bluetooth için bootloader kodu:

[www.github.com/Officializm/STM32F4_Bluetooth_Bootloader](https://github.com/Officializm/STM32F4_Bluetooth_Bootloader)

```
Lutfen cihazinizin port numarasini giriniz(Or: COM3):COM8

Port Acildi

+-----+
|          Menu          |
|    STM32F4 BootLoader v1    |
+-----+

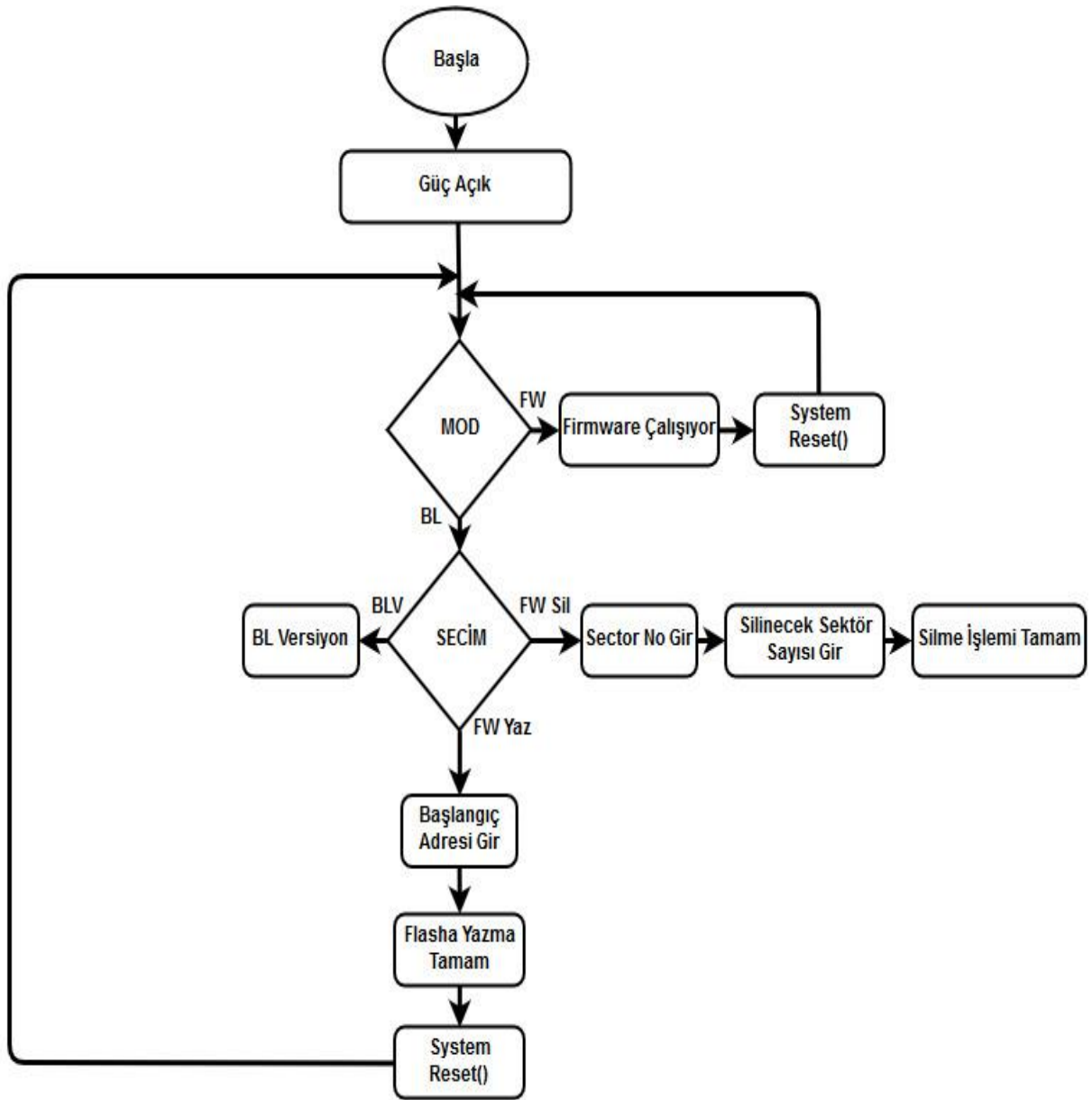
Bootloader ile Ne Yapmak İstersiniz?

BL_VERSİYON                --> 1
BL_CHIPID                  --> 2
BL_HAFIZAYI SIL            --> 3
BL_YAZILIM GÜNCELLE        --> 4
ÇIKIŞ                      --> 0

Yapmak istediğiniz işlemi seciniz :_
```

Şekil 3.1 Python Dilinde Yazılmış Arayüz

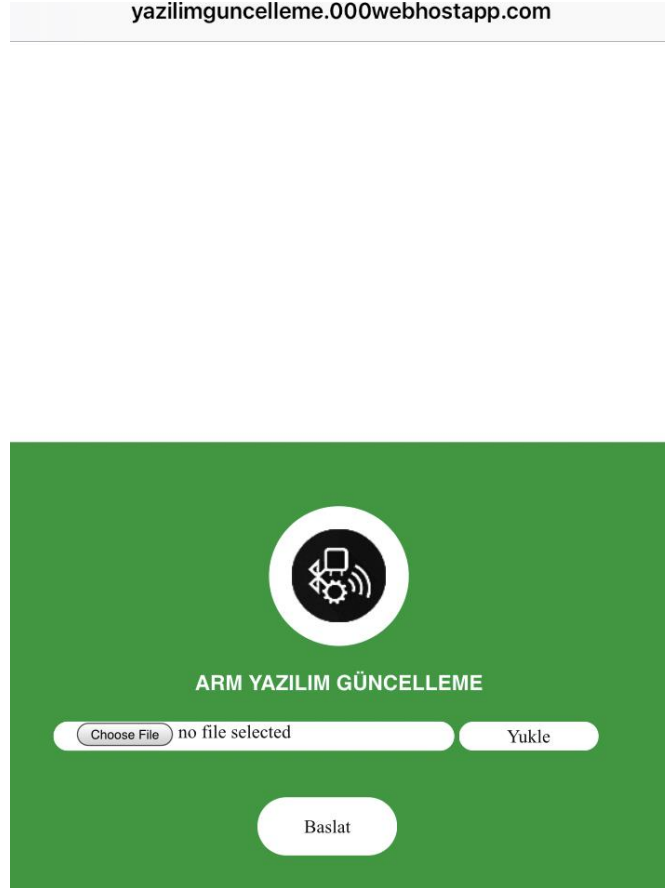
Şekil 3.2’de ise Bluetooth ile yazılım güncellemesinin akış diyagramı verilmiştir. Yazılıp güncellemesi yapılırken arayüzün bulunduğu klasörde oluşturulan bin uzantılı dosya güncelleme için kullanılır. Bu dosya güncellenmek için seçildiğinde ilk önce hafıza silinir ardından yazılım güncellenir.



Şekil 3.2 Bluetooth ile Yazılım Güncellenmesinin Akış Diyagramı

3.2 Wi-Fi Modülü ile Yazılım Güncellenmesi

Bu tezde bluetooth modülüne ek olarak ESP8266 Wi-Fi modülü kullanılarak geliştirme kartlarının yazılım güncellenmesi yapılmıştır. Wi-Fi modülüyle yazılım güncellenmesi yapılabilmesi için web sitesi hazırlandı. Web sitesinin kodları php diliyle hazırlandı. Şekil 3.3'te tez için hazırlanan web sitesi gösterilmiştir.



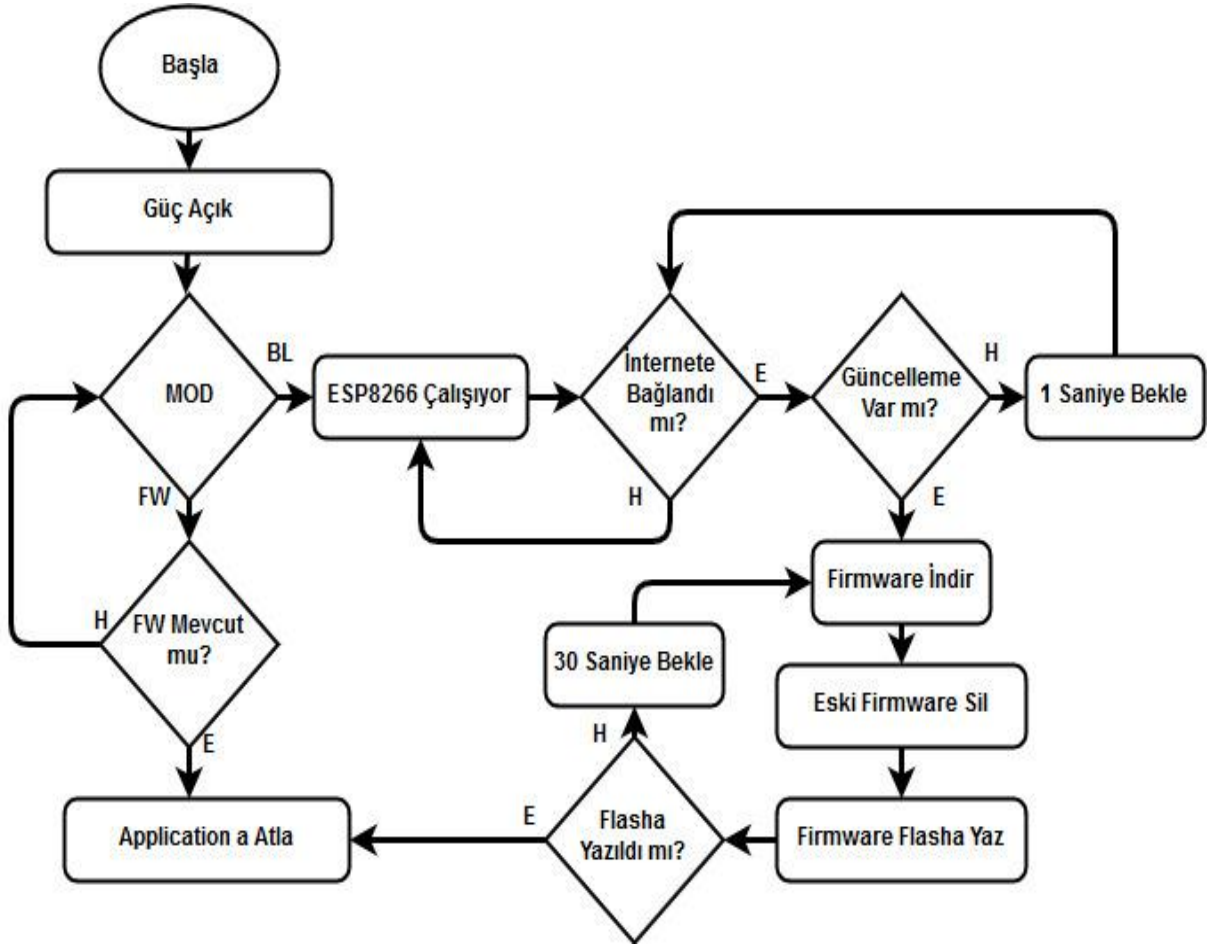
Şekil 3.3 PHP Diliyle hazırlanan Web Sitesi

Yazılım güncellenmesi yapılabilmesi için öncelikle karta enerji verildikten herhangi bir süre sonra kullanıcı butonu yardımıyla kart bootloader moduna alınır. Bu esnada güncellenmesi istenen bin uzantılı dosyayı “Gözet” butonuyla bilgisayardan veya mobil cihazdan seçtikten sonra “Yükle” butonuyla bin uzantılı dosya sisteme yüklenir. Bootloader moduna alınan kart, web sitesine yüklenen bin uzantılı dosyayı “Baslat” butonuna basılmasıyla flash hafızasının ilgili sektörüne yükler. Şekil 3.4'te Wi-Fi modülü kullanılarak yapılan yazılım güncellemesinin akış şeması gösterilmiştir.

Wifi için bootloader: [www.github.com/Officializm/STM32F4 Wifi Bootloader](http://www.github.com/Officializm/STM32F4_Wifi_Bootloader)

ESP8266 kodu: https://github.com/Officializm/ESP8266WIFI_CODE

Web arayüz: <http://yazilimguncelleme.000webhostapp.com/>



Şekil 3.4 Wi-Fi ile Yazılım Güncellenmesinin Akış Diyagramı

Bluetooth kullanılarak yapılan yazılım güncellenmesinde olduğu gibi güncel yazılım karta yüklenirken ilk olarak eski yazılım hafızadan silinir ve güncel yazılım ilgili sektöre yazılmış olur. Aşağıda sistemde kullanılan kodlar yardımıyla yazılım güncellenmesi olayı detaylandırılmıştır.

```

if(HAL_GPIO_ReadPin(GPIOC,GPIO_PIN_5) == GPIO_PIN_RESET)
{
    bootloader_uart_read_data();//Bootloader mod
}
else
{
    bootloader_jump_to_user_app();//Application mod
}
}

```

Sisteme enerji verildiğinde GPIO pininin durumuna bağlı olarak cihaz application modunda ya da bootloader modunda başlar.

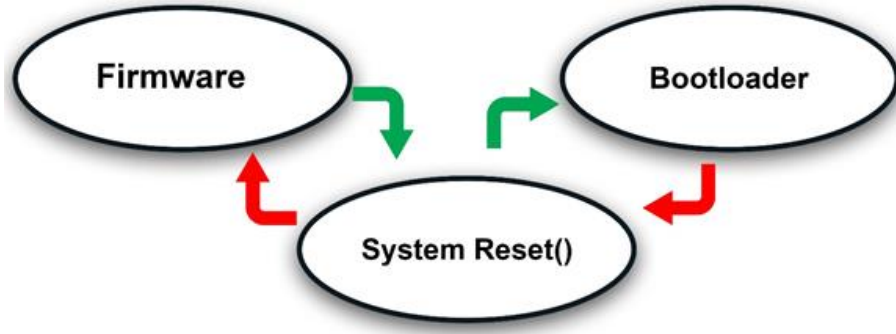
```
void bootloader_uart_read_data(void)
{
    while(1){
        memset(bl_rx_buffer,0,BL_RX_LEN); //Dizinin tüm
        elemanlarını sıfırlar
        HAL_UART_Receive(C_UART,bl_rx_buffer,BL_RX_LEN,HAL_MAX_DELAY); //Uart1
        dinlemeye geçer.
        bootloader_handle_mem_write_cmd(bl_rx_buffer);
    }
}
```

GPIO pini reset durumunda ise cihaz bootloader moduna geçerek uart dinlemeye geçer.

```
void bootloader_jump_to_user_app(void)
{ //Cihaz application moodunda başladığında bu fonksiyona dallanır.
  //App reset handlerin belirtilen adresten başlatılması sağlanarak
  // istenilen adresten itibaren firmwarenin ayaga kalkması sağlanır
  void (*app_reset_handler)(void);
  uint32_t msp_value = *(volatile uint32_t
*)FLASH_SECTOR2_BASE_ADDRESS;
  __set_MSP(msp_value);
  uint32_t resethandler_address = *(volatile uint32_t *)
(FLASH_SECTOR2_BASE_ADDRESS + 4);
  app_reset_handler = (void*) resethandler_address;
  app_reset_handler();
}
```

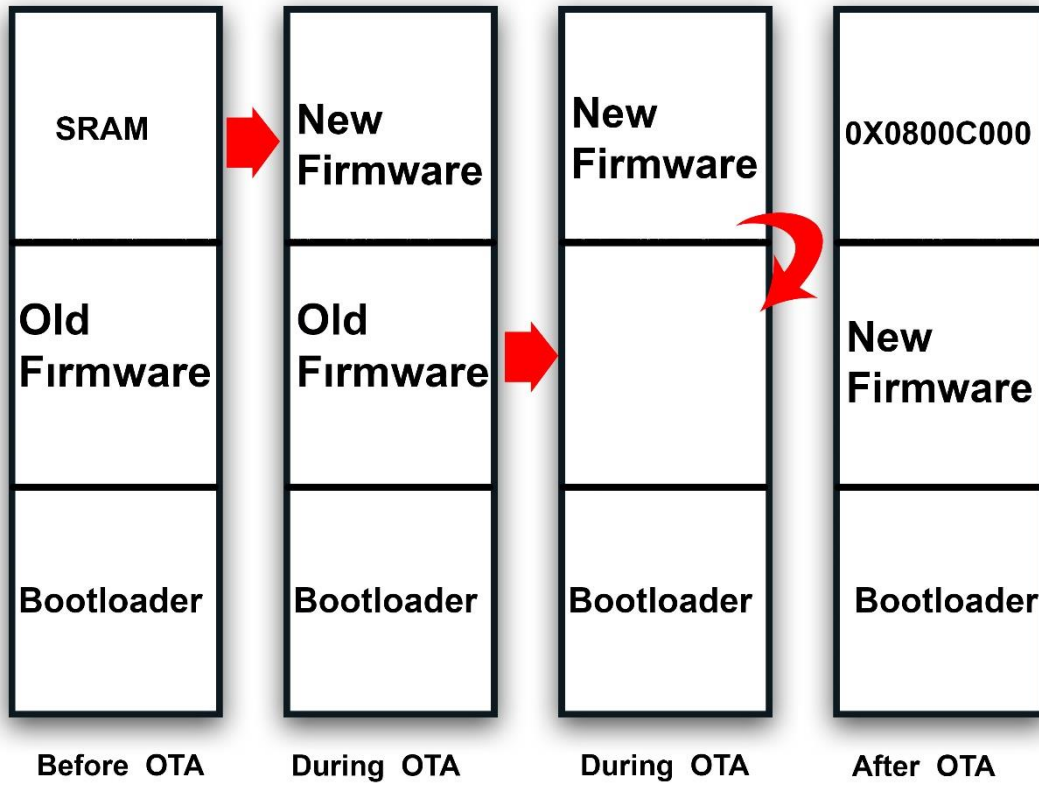
GPIO pini set durumunda ise yazılım belirtilen adresten itibaren ilk olarak app_reset_handler mekanizmasını çalıştırır. Böylece belirtilen adrese yazılan firmware kodu anlamlı bir hale getirilmesi sağlanarak çalıştırılır.

Bootloader veya firmware modu seçildikten sonra yazılım belirtilen adreslere dallanarak bootloader veya firmware kodunu çalıştırır. Wifi kısmında bootloader ve firmware ilişkisi Şekil 3.5'te gösterildiği gibidir.



Şekil 3.5 Wi-Fi Yazılım Güncelleme de Bootloader ve Firmware İlişkisi

Application katmanı çalışmaya başladığı andan itibaren esp8266 wifi modülü server tarafında bulunan veritabanını dinlemeye geçer. Application katmanında bootloader moduna geçme işlemi için bir interrupt(kesme) mekanizması kullanılmıştır. Web arayüzünde güncelleme işlemini başlatmak için “Baslat” butonuna basıldığında bu işlem veritabanına kaydedilir. Esp8266 wifi modülü belirli zaman aralıklarıyla veritabanını kontrol eder ve güncelleme işlemi başladığında GPIO2 pinini aktif eder. GPIO2 pini aktif olduğunda application tarafında bu değişiklik interrupta düşer. Interrupt içinde cihazın kendini yeniden başlatılması için NVIC_System_reset() fonksiyonu çalışır. Böylece cihaz yazılımsal olarak kendini yeniden başlatır ve Bootloader moduna girer. Şekil 3.6’da yeni yazılımın indirilip güncellenmesiyle ilgili bir şekil gösterilmiştir.



Şekil 3.6 Güncel Yazılımın Yüklenmesi

Yeni yazılım indirildikten sonra RAM’de saklanır. Eski yazılım aşağıda görüldüğü gibi flash bellekten silinir ve yeni yazılım eski yazılımın adresine yazılır bu işlemden sonra cihaz NVIC_System_Reset() fonksiyonunu tekrar çalıştırarak yeniden başlar ve application moduna girer. Böylece yeni yazılım çalışmaya başlar.

```
void bootloader_handle_mem_write_cmd(uint8_t *pBuffer)
{
    execute_flash_erase(2,1); //Firmware kodunun bulunduğu
    //flash bellekte 2. sektörün 1. kısmı silinir
    execute_mem_write(pBuffer,0x08008000, BL_RX_LEN); //Yeni Firmware flasha
    yazılır.
    HAL_NVIC_SystemReset(); //İşlem tamamlandıktan sonra cihaz kendini
    yeniden başlatır.
```


KAYNAKÇA

- [1] (URL: <http://www.teknolo.com/internet-things-nesnelerin-interneti-nedir/>)
- [2] Rafat Ö., 2010, PIC16F877 mikrodnetleyicisiyle bir PLC tasarımı, Yüksek Lisans Tezi, Selçuk Üniversitesi Fen Bilimleri Enstitüsü, Konya, 21.
- [3] [PAMUKÇU P. 2014, Dizel motorların enjeksiyon sistemi test cihazının elektronik kontrolünün gerçekleştirilmesi, Yüksek Lisans Tezi, Gazi Üniversitesi Bilişim Enstitüsü, 20-21]
- [4] [URL: www.st.com – stm32f401re nucleo board reference manual]
- [5] [KANDEMİR V. 2011, Genel Amaçlı Programlanabilir Kontrolör Tasarımı, Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü, 63]
- [6] [URL: Arduino Software [https://www.arduino.cc/en/Main/ Software](https://www.arduino.cc/en/Main/Software)]
- [7] [<http://www.calismaprensibi.com/bluetooth-nedir-nasil-calisir-kablosuz-iletisimnedir.html>]
- [8] [AK, M. 2018, El Rehabilitasyon Sistemi için Akıllı Eldiven Tasarımı, Lisans Tezi, Marmara Üniversitesi Teknoloji Fakültesi, 10-11]
- [9] [Özköse, H., Bilişim Uzmanlığı Tezi, Makineler Arası Haberleşme (M2M) ve Türkiye İçin Düzenleyici Öneriler, Bilgi Teknolojileri ve İletişim Kurumu, 2014]
- [10] [Nisio D.A, Noia D.T, Carducci G.C, Spadavecchia M. "Design of a Low Cost Multipurpose Wireless Sensor Network" Measurements & Networking (M&N), 2015]
- [12] [ŞENER E. 2016, Geniş Bant Optik ve Kablosuz Haberleşme Ağları İçin Erişim Teknolojilerinin İncelenmesi, Lisans Tezi, Karabük Üniversitesi Mühendislik Fakültesi, 88]
- [13] [URL: <http://yazilimyazari.com/server-ve-client/>]
- [14] [BOLAT Y. 2015, Hafıza Analizi, Yüksek Lisans Tezi, İstanbul Bilgi Üniversitesi Sosyal Bilimler Enstitüsü, 8-11]
- [15] [URL: www.st.com – reference manual]
- [16] [KARAPINAR K., KAYA F. 2016, Gömülü Sistemlerin Haberleşmesi, Karabük Üniversitesi Mühendislik Fakültesi, 33-35]

ÖZGEÇMİŞ

Adı Soyadı: Mahmut Ahmet ÜNAL
Doğum Tarihi: 25.11.1996
Doğum yeri: Pendik/ İSTANBUL
Okuduğu Lise: Kartal Yüksel-İlhan Alanyalı Anadolu Öğretmen Lisesi / 2010 - 2014
Staj Yaptığı Yerler: İSBAK A.Ş – Ar-Ge Stajyeri – İSTANBUL
TÜBİTAK – UME – Medikal Metroloji Laboratuvarı - KOCAELİ

Adı Soyadı: Recep ÇETİKAYA
Doğum Tarihi: 01.08.1995
Doğum yeri: Beşiri/ BATMAN
Okuduğu Lise: Pendik Türk Telekom Anadolu Teknik Lisesi / 2009 - 2013
Staj Yaptığı Yerler: Geobilgi Yazılım A.Ş. – Ar-Ge Stajyeri - İSTANBUL
Pavotek Tasarım A.Ş. – Ar-Ge Stajyeri - İSTANBUL