

**T.C.**

**DÜZCE ÜNİVERSİTESİ**

**MÜHENDİSLİK FAKÜLTESİ**

**OTONOM İMHA ARACI**

**BİTİRME PROJESİ**

**CİHAN AYTAÇOĞLU**

**MEVLÜT ÖZTÜRK**

**BİTİRME TEZİ**

**Danışman:Dr.Öğr.Üyesi Enver KÜÇÜKKÜLAHLI**

**DÜZCE, 2022**

[1. GİRİŞ 4](#_Toc1905634920)

[1.1 TARİHİ SÜREÇ 6](#_Toc2135967705)

[1.3.1.1 Template Matching (Şablon Eşleştirme) 8](#_Toc12776479)

[1.3.1.2 HAAR Cascade 9](#_Toc1273579204)

[1.3.1.3 LBP (LOCAL BİNARY PATTERNS) 9](#_Toc1284612104)

[1.3.1.4 Meanshift (Ağırlıklı Ortalama Kaydırma) 9](#_Toc58142327)

[2.1 RASPBERRY Pİ KURULUMU 11](#_Toc102917552)

[2.2 OPENCV KURULUMU 13](#_Toc108838918)

[2.3 ESP KURULUM SÜRECİ 15](#_Toc109299212)

[2.4 STEP MOTOR KÜTÜPHANESİ HAZIRLAMA 16](#_Toc620408246)

[2.5 SİLAH SİSTEMİ YAZILIMI 19](#_Toc359585601)

[2.6 OTONOM ARACIN YÖNETİMİ VE KODLARI 22](#_Toc1905494962)

[2.7 MOBİL UYGULAMA İLE OTONOM İMHA ARACININ YÖNETİLMESİ 28](#_Toc85628534)

[2.8 KULLANILAN MATERYALLER 31](#_Toc2082706311)

[2.9 SİLAH SİSTEMİNİN DEVRE ŞEMASI 38](#_Toc1764171311)

[3 SONUÇ 39](#_Toc1430332215)

[KAYNAKÇA 39](#_Toc501087450)

[EKLER 41](#_Toc619040524)

[Ek1 : Projeyle İlgili Görseller 41](#_Toc1810172048)

BEYAN

Bu tez çalışmasının kendi çalışmamız olduğunu, tezin planlanmasından yazımına kadar bütün aşamalarda etik dışı davranışımın olmadığını, bu tezdeki bütün bilgileri akademik ve etik kurallar içinde elde ettiğimi, bu tez çalışmasıyla elde edilmeyen bütün bilgi ve yorumlara kaynak gösterdiğimi ve bu kaynakları da kaynaklar listesine aldığımızı, yine bu tezin çalışılması ve yazımı sırasında patent ve telif haklarını ihlal edici bir davranışımın olmadığını beyan ederiz

**27 Mayıs2022**

**Cihan AYTAÇOĞLU**

**Mevlüt ÖZTÜRK**

TEŞEKKÜR

Lisans öğrenimimizde ve bu tezin hazırlanmasında gösterdiği her türlü destek ve yardımdan dolayı çok değerli hocamız Dr. Öğretim Üyesi Enver KÜÇÜKKÜLAHLI’ a en içten dileklerimizle teşekkür ederiz. Bu çalışma boyunca yardımlarını ve desteklerini esirgemeyen sevgili ailelerimize ve çalışma arkadaşlarımıza sonsuz teşekkürlerimi sunarız.

27Mayıs2022

Cihan AYTAÇOĞLU

Mevlüt ÖZTÜRK

# **GİRİŞ**

Otonom sistemler, kendi kararlarınca hedeflere yönelebilen ve çeşitli akıllı varlıklara uyarlanabilen sistemler ile kendi kararlarını verebilmekte olan sistemler haline getirilerek günümüz teknolojisinde çeşitli hükumetlerin ve devletlerin can ve mal kayıplarını minimalize etmek amacıyla parmak bastıkları projeler arasında rehavet ile otonom sistemler gelmektedir. Asıl en büyük avantajları ise insan hayatının değeri paha biçilemez olduğundan insan kayıplarını önlemekteki başarısıdır, aynı zamanda bu sistemler çeşitli hesaplamaları ve lokasyonları kestirimi açısından insanlardan daha hızlı yaptıkları için de önemle bir yer arz etmektedir. Günümüzde otonom sistemler yalnızca savunma sanayi ya da askeri alanda kullanılmakla kalmayıp aynı zamanda tarım, endüstri ve daha nice alanlarda başarıyla görevlerini icra ettirmektedirler. Sanayi devriminden bu yana süregelen hızlı üretim ve tüketim ihtiyaçlarına karşılık artık insan yükünün de bir kenara bırakılıp otomatikleştirme ve devamlı üretim yolunda atılan en büyük atılımlardandır. Otonom sistemler çeşitli departmanlara ayrılırlar bunlardan birisi de kavrayabilme ve karar iyileştirme yeteneği olan otonom sistemlerdir. Günümüzde ise bu sistemler kabaca robot olarak tasvir edilmektedir. Her farklı sektörde kullanılan otonom sistemlerin sektöre bağımlı olarak tasarım hedefleri vardır. Sanayi sistemlerinde bu hızlı üretim iken, savunma sanayi de ise minimum can kaybı ve başarı hassasiyetidir.

Geliştirilen projemiz ise tam olarak bu sektöre uyarlı olarak hayata geçirilmek istenen bir prototiptir. Bu projede uzaktan (lokal) yönlendirilebilen bir araç tasarlandı ve buna eklenti olarak nesne tespiti yapabilen ve bu tespite paralel olarak ateşlenebilen bir silah sistemi tasarlanmıştır. Savunma sanayinde kullanılan askeri ağ yapısına benzer bir mantıkla farklı ağ parazitlerinden bu otonom sisteminin etkilenmemesi adına prototipin çalışma aralığı lokal ağ bazında kısıtlı tutulmuştur.

**ÖZET**

Hedeflenen tasarım olan otonom imha aracı savunma sanayi ‘ne yönelik olarak tasarlanan askeri bir modeldir. Bu modelin asıl amacı insanın can kaybını minimalize etmek ve çeşitli lokasyon hesaplamalarında hassasiyetinin çok yüksek olmasıdır. Belirlenen hedefin ya da nesnenin hata payının oldukça düşük olması önem arz etmektedir. Geliştirilen bu prototip tamamen otonom ve kısmi otonom özelliğine sahiptir.

Prototip geliştirilirken iki ayrı parça olarak tasarlandı;

Bunlar:

Araç sistemi ve silah sistemi olmaktadır. Araç sisteminin çalışma mantığı operasyon ya da keşif yapılacak bölgeye gönderilir ve bölgesel ağ bazında bağlı olduğu ağda kullanıldığından parazitlerden etkilenme olasılığı oldukça düşüktür. Manuel olarak kontrol edilen araç istenilen koordinatlara doğru hareket eder. Silah sistemi ise bölgeye gönderilen aracın üzerinde yer alıp, keşif ya da operasyon esnasında karşılaşılan tehditlere karşı bir nesne tespiti yapar ve görüntü verileri kullanıcı tarafına iletilir. Tehdit durumunda silah sistemi ateşlenir ve hedef imha edilir.

# **1.1 TARİHİ SÜREÇ**

Sanayi devriminden bu yana insanlık daha hızlı gelişmekte olup durağan bilmeyen bir tüketim haline geldiğinden bu insanlık için büyük bir sorun haline dönüşmüştür. Ortaya çıkan kıtlıklar, meydana gelen doğal afetler ve daha nice harici ya da dahili arızlar neticesinde ortaya çıkan can ve mal kayıpları ,insanların tüketime olan eğilimleri, üretimde beklentiden daha düşük bir sonuç elde edilmesi gibi faktörler insanların en büyük sorunları haline gelmiştir. İşte tam da bu sorunlardan ötürü insanlık farklı çağ dilimlerine geçiş yapacak önemli adımlar atmıştırlar. Bu adımlardan en önemli ise sanayi devrimidir. Bu devrimden sonra otomasyon, robot ,makine gibi kavramlar hayata geçip seri üretim amaçlı insanların hayatını kolaylaştırmıştır. Sürekli geçen zaman ve gelişen teknolojiye bağımlı olarak bu sistemler de yerini iyileştirilmiş üst modellerine devretmişlerdir. İlk olarak insanlık tarihine yarı otonom sistemler girmiştir. Henüz makineler ya da kablolar yok iken tekerleğin icadı bile günümüzdeki otonom araçların atası olarak anılmaktadır. Basit makinalar, kaldıraçlar gibi fizik dersinde görülen konular da aslında bir otonom sistemdir. Günümüz yüzyılına bakıldığında ise otonom sistemlere gelişmiş yapay zeka, makine öğrenmesi gibi modern algoritmalar entegre edilerek farklı işlevsellikler ile donatılmışlardır. Eklenen bu modern algoritmalar sayesinde otonom sistemler karar verebilme, görüntü işleme, sınıflandırma, kümeleme, insanımsı düşünebilen bir hal alabilmektedirler aynı zamanda geçmiş verilerden yararlanarak anlamlı ve faydalı bir bilgi ortaya çıkarabilen bu sistemler insanımsı sistemler unvanına kavuşmuştur. Daha önce de belirtildiği üzere zaman ve teknolojiye bağımlı olarak gelişimi hızla artan otonom sistemler hakkında uzun vadeli düşüncelerin insan üzerindeki olumsuz etkilerinin çoğalacağı düşünülmektedir. Günümüzde otonom sistemlerin favorisi olarak robotlar (insanımsı robotlar) anılmaktadırlar. Tıpkı bir insan gibi akıllıca karar verebilme mekanizmasına sahip, konuşabilen ve algılayabilen burobotlar insanı bir hayli heyecanlandırıyor. Diğer taraftan ise teknolojinin gelişiminden süre gelen araçların meydana getirdiği kazalar sonucu oluşan can ve mal kayıplarının dur durakbilmeyendöngüsü günümüz teknolojisinin otonom araçları ile neredeyse optimal çözüme ulaşmıştır.

**1.2 OTONOM SİSTEMLERİN SAVUNMA SANAYİNDEKİ ROLÜ**

Savunma sanayinde kullanılan otonom sistemler savunma ve saldırı stratejisinde insan gücünü ve insan gücünden kaynaklı hataları minimum seviyeye indirerek az zamanda daha çok savunma ve saldırı stratejisi geliştirilmesini sağlamaktadır. Görüntü işleme teknikleri sayesinde saldırı stratejisinde hedeflerin konumu ve doğrultusu maksimum hassasiyetle bulunmakta ve hassas atışlar yapılarak hedefler kısa sürede imha edilebilmektedir. Savunma stratejisinde radarlar yardımıyla tespit edilen tehditler savunma sistemini uyararak insan gücüne bağlı kalmadan yapay zeka yardımıyla tehditler yok edilmekte ve kritik önem taşıyan bölgeler güvenli hale getirilmektedir. Gelişen otonom sistemler sayesinde saldırılması gereken bölgelere insan göndermeden hedefleri otomatik olarak tespit edilip yok edilmektedir. Bu gelişmeler sayesinde otonom sistemlere gösterilen ilgi ve ihtiyaç artmaktadır.

**1.3 NESNE TESPİTİ**

Nesne tespiti kameradan alınan görüntülerin belirli algoritmalar ve modeller yardımıyla görüntü üzerindeki canlı ve ya cansız varlıkların tespit edilmesidir.

**1.3.1 Nesne Tespiti İçin Kullanılan Algoritmalar**

Günümüzde kullanılan bir çok nesne tespit edebilen algoritmalar geliştirilmiştir. Bu algoritmaların kimisi gerçek zamanlı tespit işlemleri için kullanılırken kimisi ise yüksek doğrulukta veri elde edebilmek için kullanılmaktadır. Nesne tespit etmenin bazı yolları vardır ; eğitilmiş model üzerinden nesne tespit edebilme , görüntü üzerindeki piksel değerlerinden yola çıkarak çeşitli hesaplamalarla nesne tespiti.

### **1.3.1.1 Template Matching (Şablon Eşleştirme)**

Şablon eşleştirme teorisi, insan örüntü tanımaya yönelik en temel yaklaşımı tanımlar. Algılanan her nesnenin uzun süreli belleğe bir "şablon" olarak depolandığını varsayan bir teoridir. Tam bir eşleşme bulmak için gelen bilgiler bu şablonlarla karşılaştırılır.

### **1.3.1.2 HAAR Cascade**

Bu yöntem görüntü üzerinde çeşitli haar-like özellikler tarayan ve nesne tespit eden algoritmadır.

Görüntü üzerinde taranan haar-like özellikler şunlardır :

Kenar özelliği: Görüntü üzerinde belirli bir alan koyu alandan oluşuyor ve belirli bir alan açık renklerden oluşuyor ise kenar özelliği olduğunu belirtmektedir.

Çizgi özelliği: Görüntü üzerinde sırasıyla açık, kapalı, açık renklerden oluşuyor ise çizgi özelliği vardır.

Dört-Kare Özelliği: Çaprazlama olarak kare şeklinde koyu ve açık tonlar çapraz bir şekilde bulunuyor ise dört kare özelliğini belirtmektedir.

### **1.3.1.3 LBP (LOCAL BİNARY PATTERNS)**

LBP doku analizi operatörü, gri-seviyeden bağımsız bir doku ölçümü yöntemidir. Yerel ikili örüntüler, bilgisayar görüşünde sınıflandırma için kullanılan bir tür görsel tamamlayıcıdır. LBP görüntü analizi, doku analizi, yüz analizi ve hareket analizi gibi alanlarda kullanılmaktadır.

LBP kullanılması oldukça kolay ve ucuz maliyetli olan bir doku operatörüdür. Her bir pikselin komşuluklarını esas almaktadır. Operatörün yaptığı iş, seçilen merkez noktanın önceden belirlenmiş yarıçap parametresine göre, yarıçap uzaklığındaki her bir komşusunun kıyasına dayanmaktadır.

### **1.3.1.4 Meanshift (Ağırlıklı Ortalama Kaydırma)**

Bu algoritmada veri kümesi üzerindeki veri (bazı kaynaklarda nokta olarak geçer) dağılımının en yüksek olduğu yeri bulmayı sağlar. Yani elinizdeki bir görüntü ve devam edecek görüntü üzerinde belirlenen bir alan içerisinde yer alan tepe noktaları belirlemesi sürecidir. Belirlenen tepe noktalar takip edilmek istenilen nesne olacaktır.

**1.4 ARAÇ YÖNLENDİRİLMESİ VE NESNELERİN İNTERNETİ**

Geliştirilen modelin araç entegre kısmında aracın yönlendirilmesi manuel olarak yerel ağda yapılmaktadır. Akıllı araç modülündeki entegre de geliştirilen sistem yerel ağ merkezli bir dağıtım noktası olarak kullanılır ve belirlenen parametrelerde bir ağ yayınlar aracın kontrol mekanizması için oluşturulmuş uygulama ile sisteme giriş yapan yetkili kullanıcı aracı hedeflenen bölgeye kendi verdiği komutlar ile yönlendirir. Nesnelerin interneti konusunda ise kullanılan geliştirici kart modeli sayesinde bir ağ yayınlanır ancak bu ağa birden fazla kullanıcı bağlandığı takdirde kullanıcı adayları aracı kullanamayacaklardır bu sorundan ötürür araç tekil olarak tasarlanmıştır. En yetkili kullanıcı tarafından yönlendirilir ve veriler bir protokol üzerinden görüntülenebilir bir haldedir. Keşif ya da operasyon bölgesine ulaşan araç artık diğer temel model olan silah sistemine görevini icra ettirmesi için uygun koşulları sağlar. Olası parazitlenmelere karşı yerel ağda üretilen proje dıştan gelecek olumsuz etmenlere karşı da dirençlidir. Bölgede işlevini yerine getiren araç daha sonrasında ise tekrar en yetkili kullanıcı tarafından kontrol edilerek geri dönüşü sağlanır.

**2. OTONOM İMHA ARACININ TASARIMI ve GELİŞTİRİLMESİ**

## **2.1 RASPBERRY Pİ KURULUMU**

Raspberry pi projemizde nesne tespitini yapıp silah sitemini hedefe yönlendirecek olan cihazdır. Raspberry pi üzerine linux tabanlı olan Raspian işletim sistemi kuracağız. Raspberry pi için geliştirilen bir çok işletim sistemi mevcut olsa da kullanılacak olan kütüphaneler ve gerekli konfigürasyonların bir arada bulunduğu işletim sistemi Raspian işletim sistemidir.

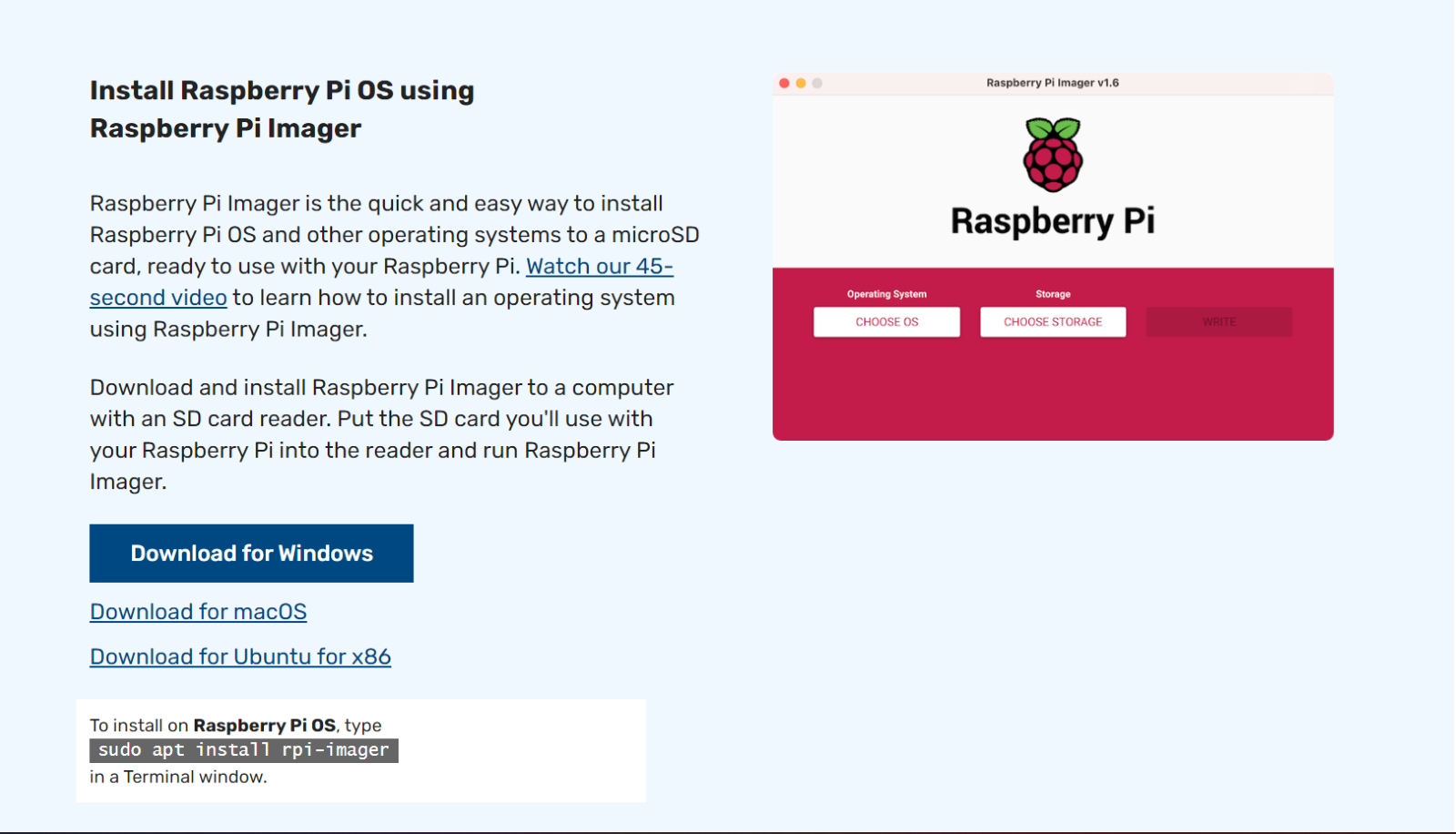
Raspberry pi üzerinde 40 adet GPIO pini barındırmaktadır. Axis kontrolü , silah kontrolü ve sensör okuma işlemleri bu pinler üzerinden gerçekleştirilecektir.

Kurulum aşamaları :

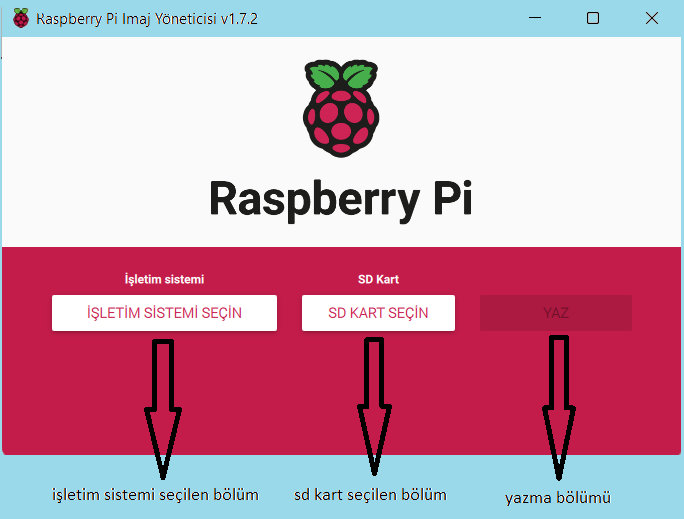
* Raspberry pi imager programı indirme:

Bu yazılım sd kartımıza işletim sistemini kuracak olan yazılımdır.

<https://www.raspberrypi.com/software/> web adresinden bu yazılımı indirerek bilgisayarımıza kuruyoruz



şekil 1



şekil 1.1

Kurulumu yaptıktan sonra eğer monitör vs. Kullanmak yerine bilgisayarımızdan ssh ile bağlanmak istiyorsak sd karttaki boot bölümüne ssh isminde boş ve uzantısı olmayan bir dosya oluşturuyoruz. Raspberry ile bağlantı kurmak için aynı ağda bulunmamız gerekiyor dolayısıyla Raspberry pi’nin bizim ağımıza otomatik bağlanması gerekir. Bunun için yine sd karta wpa\_supplicant.conf isminde dosya oluşturuyoruz. Dosyanın içine şunları ekliyoruz :

country=TR

update\_config=1

ctrl\_interface=/var/run/wpa\_supplicant

network={

scan\_ssid=1

ssid="wifi ismi"

psk="wifi şifresi"

}

Bu aşamaları yaptıktan sonra raspberry pi’yi çalıştırdıktan sonra ağımıza otomatik olarak bağlanacaktır. Bağlantı kurulduktan sonra terminal ekranımızla raspberry cihazımıza ssh ile bağlanıp bazı konfigürasyonlar yapması gerekir. Terminal ekranında ssh [pi@raspberrypi.local](mailto:pi@raspberrypi.local) komutu ile bağlanıp “sudo raspi-conig" ile konfigürasyon ayarlarından interface bölümünden vnc server ve camera seçeneklerini aktif etmeliyiz.

## **2.2 OPENCV KURULUMU**

Opencv kütüphanesi gerçek zamanlı nesne takibi yapmamızı sağlayan kütüphanedir. Bu kütüphaneyi cihaza kaynak kurulumu yaparak entgre edilmelidir.

Kurulum :

Kurulum aşaması raspberry pi cihazının terminali ile yapılır.

* İlk adımda geliştirici paketlerini “sudo apt-get install build-essential cmake cmake-curses-gui pkg-config" komutu ile yüklüyoruz
* İkinci adımda opencv bağımlılıklarını “sudo apt-get install libtiff5-dev \  
   libjasper-dev \  
   libpng12-dev \  
   libavcodec-dev \  
   libavformat-dev \  
   libswscale-dev \  
   libeigen3-dev \  
   libxvidcore-dev \  
   libx264-dev \”

Komutu ile yüklüyoruz

* Üçüncü adımda opencv paketlerini “wget <https://github.com/opencv/opencv/archive/4.x.zip> -O opencv\_source.zip “ , “wget <https://github.com/opencv/opencv_contrib/archive/4.x.zip> -O opencv\_contrib.zip “ komutu ile indiriyoruz ve “unzip opencv\_source.zip” , “unzip opencv\_contrib.zip” komutları ile dosyaları çıkartıyoruz.
* Dördüncü adımda opencv-4.x klasörünün içine build isminde klasör oluşturuyoruz ve bu klasörün içine giriyoruz
* Beşinci adımda cmake işlemini

“cmake -D CMAKE\_BUILD\_TYPE=RELEASE \

-D CMAKE\_INSTALL\_PREFIX=/usr/local \

-D BUILD\_WITH\_DEBUG\_INFO=OFF \

-D BUILD\_DOCS=OFF \

-D BUILD\_EXAMPLES=OFF \

-D BUILD\_TESTS=OFF \

-D BUILD\_opencv\_ts=OFF \

-D BUILD\_PERF\_TESTS=OFF \

-D INSTALL\_C\_EXAMPLES=ON \

-D INSTALL\_PYTHON\_EXAMPLES=ON \

-D OPENCV\_EXTRA\_MODULES\_PATH=../../opencv\_contrib-4.x/modules \

-D ENABLE\_NEON=ON \

-D WITH\_LIBV4L=ON \ ../ “

komutu ile gerçekleştiriyoruz

* Altıncı adımda build işlemine “make -j4” komutu ile başlıyoruz
* Yedinci adımda “sudo make install” ve “sudo ldconfig” komutu ile işlemi sonlandırıyoruz.

## **2.3 ESP KURULUM SÜRECİ**

Otonom imha aracında kullanılan önemli yazılım geliştirme ekipmanlarından birisi ESP model geliştirici kart olmaktadır. Bu geliştirici kart sayesinde otonom, iot(nesnelerin interneti) gibi günümüzün gözde projeleri tasarlanabilmektedir. Genellikle c dili tabanlı olan Arduino ide ile çalıştırılır, derlenir ve karta yazılım yüklenir kartın kurulum sürecinde öncelikli olarak karta uyumlu olarak tasarlanmış yazılımlar ara yüz uygulama programına kütüphaneleri tespit edilerek indirilir ve kart sisteme tanıtılır. Kullanmış olduğumuz geliştirici kart modelimiz sisteme tanıtıldıktan sonra yapılacak olan projemizin gerekliliğine göre ihtiyaç duyulan kütüphaneler yine aynı şekilde sisteme yüklenip tanıtılır. Esp geliştirici kart modelinde sisteme karta yüklenen yazılım akım bağlantısı kesilene yada yeterlilik kriteri sağlanana kadar bir döngü içerisinde çalışır , akım kesildiği takdirde çalışmasını durdurur ancak mevcut kartın içerisinde mevcut bulunan yazılım eeprom bellekte tutulmaya devam edilir yani tekrar karta besleme uygulandığında geliştirici kartımıza daha önce yüklemiş olduğumuz proje yazılımı çalışacaktır. Ancak kartımıza farklı bir yazılım yüklediğimiz takdirde önceden hafızaya alınmış olan yazılım silinir ve sistemin hafızası yeni yazlımı tanır. Geliştirdiğimiz modelde tam bahsedildiği üzere işlemler yapılmıştır ve gerekli yazılım dokümanı karta yüklenmiştir. İstenildiği takdirde ise farklı projeler için gerekli kütüphane manuel olarak tasarlanıp yazılım geliştirme ortamına kütüphane olarak entegre edilerek geliştirici kart üzerinde kullanılabilir. Geliştirici kartımızın kablolama pinleri de verimli bir şekilde kullanılmıştır. ESP8266 üzerinde işlevli olarak analog ve dijital pinler bulunmaktadır bu pinler kullanılan diğer donanım ekipmanların uygunluğuna göre bağlanır ve entegre edilir. Yanlış pinlerin kullanılması yada desteklenmeyen donanım parçasının pin’ler de kullanılması durumunda ise geliştirici kartımız yanabilir.

## **2.4 STEP MOTOR KÜTÜPHANESİ HAZIRLAMA**

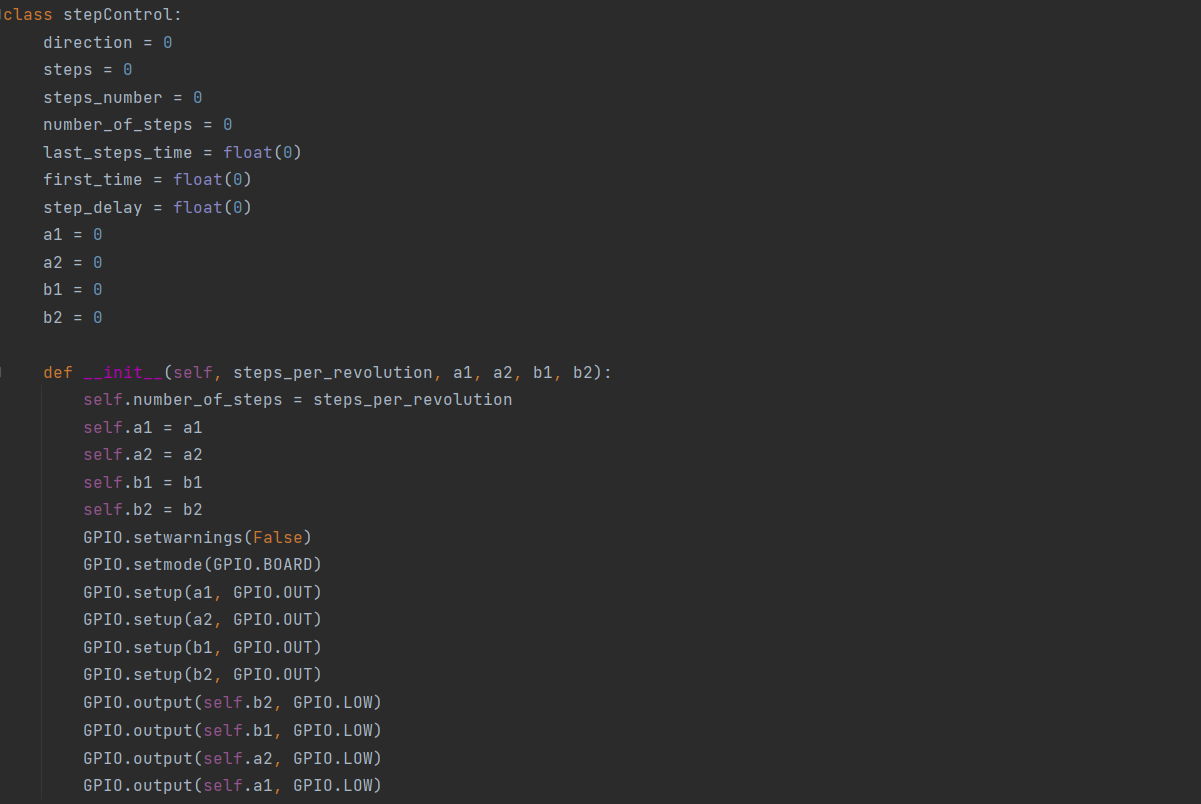
Bu projede eksen hareketlerinden ve sistemin ağırlığından ötürü motorlar yüksek akım çekmektedir. Bu sebepten ötürü yüksek akım verebilen bir step motor sürücü kartı olan l298n motor sürücü kartı kullanılmıştır. Raspberry pi’nin içerisinde bu sürücü kartını çalıştırabilen hazır bir kütüphane yoktur dolayısıyla kendimiz oluşturmalıyız.

Kütüphane kodları :



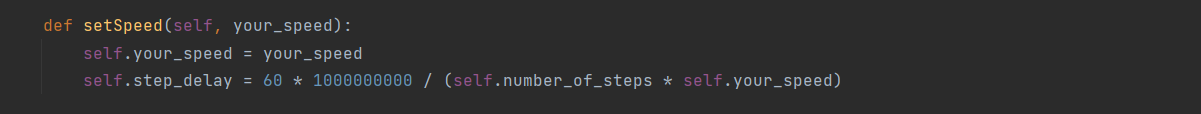
şekil 2.0

Bu kütüphaneyi yapabilmek için GPIO pinlerini kullanmamız gerekir ve adım arasında gecikme oluşturmak için time kütüphanesine ihtiyaç duymaktayız



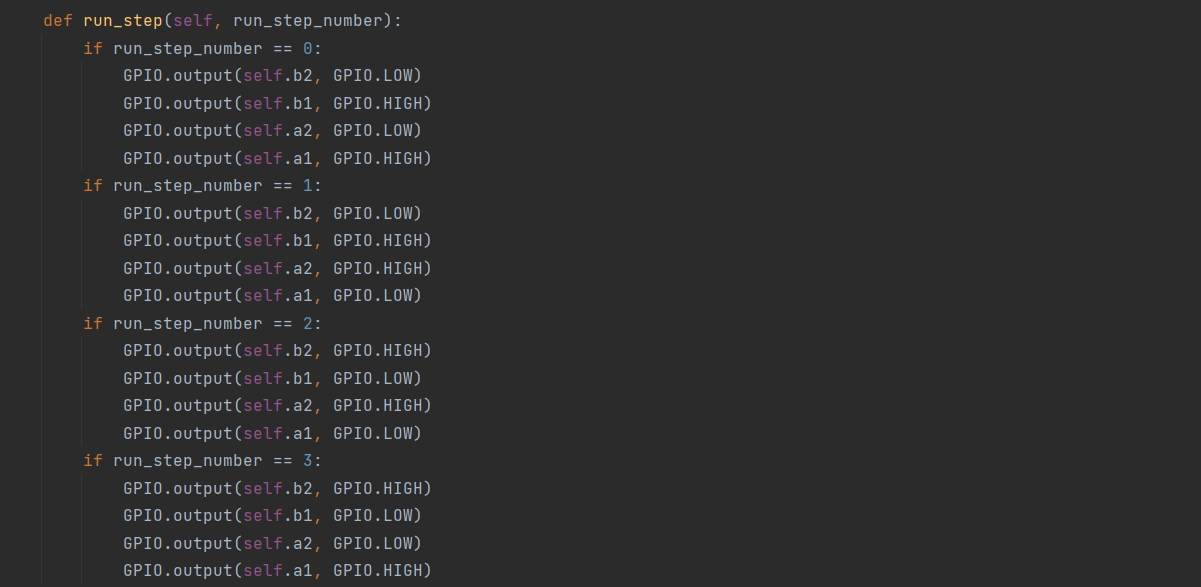
şekil 2.1

Bu kısımda pin tanımlamaları step motorun tam tur sayısı belirlenmektedir



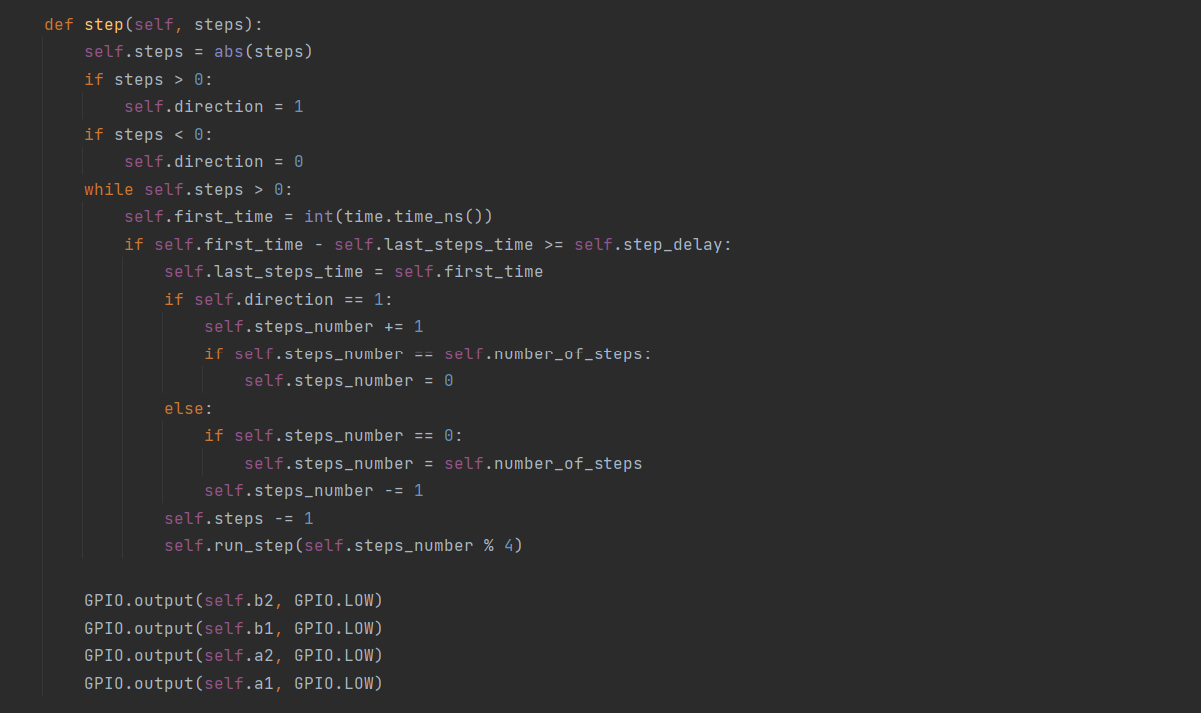
şekil 2.2

Bu fonksiyonda motor hızını RPM(Round Per Minute) olarak belirlenir.



şekil 2.3

Bu fonksiyon motorların belirli bir senkronizasyon içinde çalışmasını sağlar.



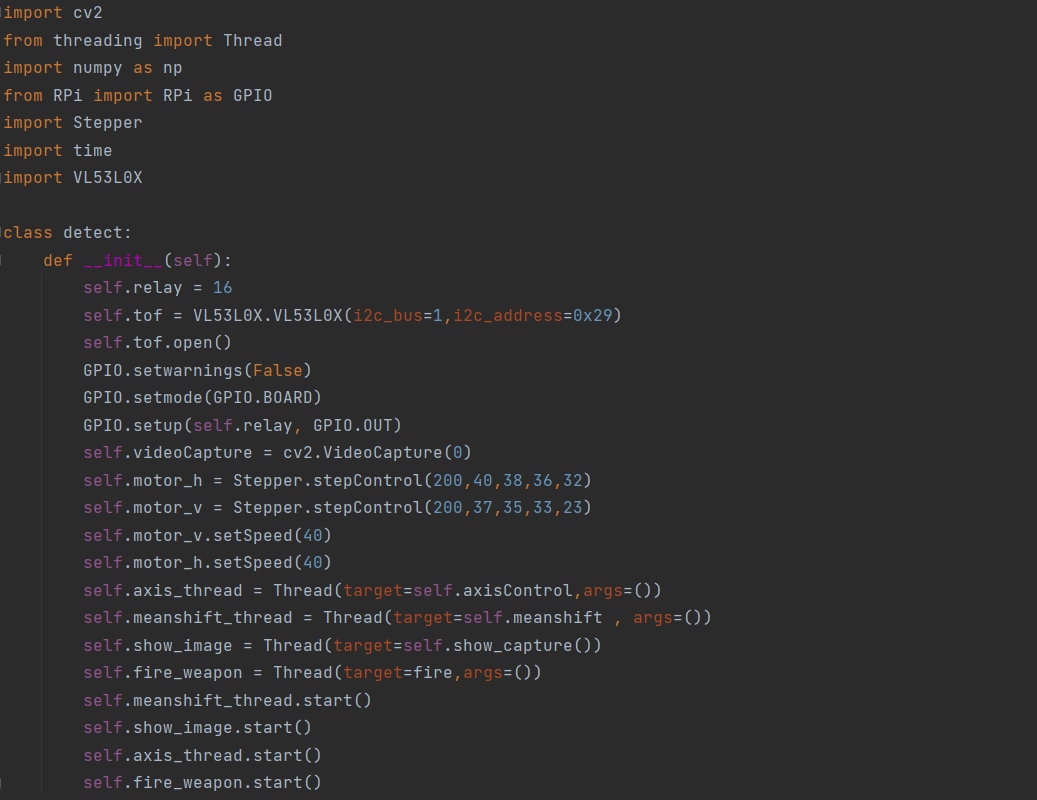
şekil 2.4

Bu fonksiyon step motorun bir sonraki adımını ve o adıma ne kadar süre sonra geçeceğini hesaplar.

## **2.5 SİLAH SİSTEMİ YAZILIMI**

Bu silah siteminde çalışma hızından ötürü mean shift algoritması kullanılmıştır. Sistem nesneyi tespit ettikten sonra gelen koordinat bilgileri doğrultusunda motorları hedefe yönlendirmekte ve hedef sensörün görüş uzaklığına ve algılama mesafesine girdiği zaman atış yapmaktadır.

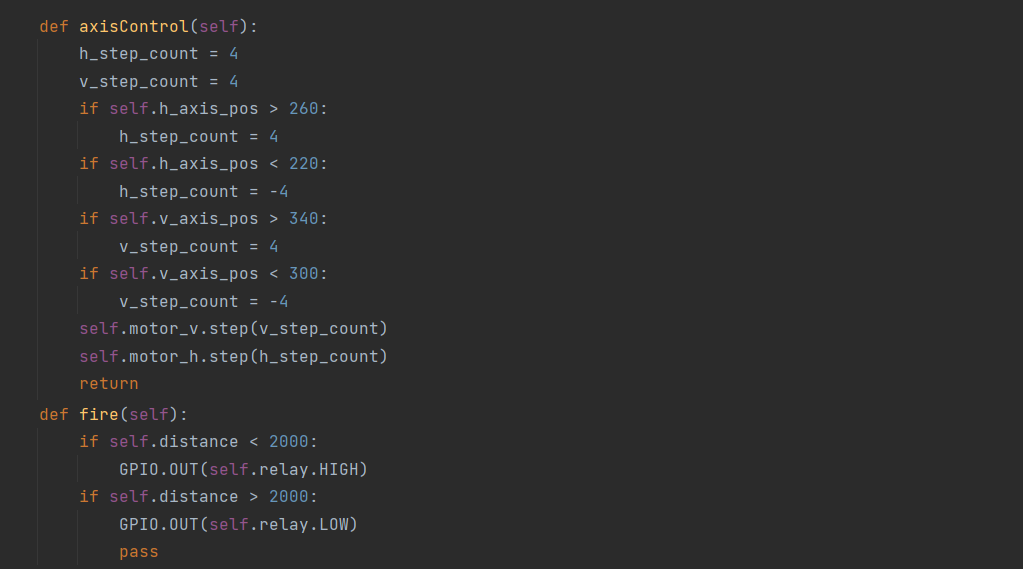
Kodlar :

 şekil 3.0

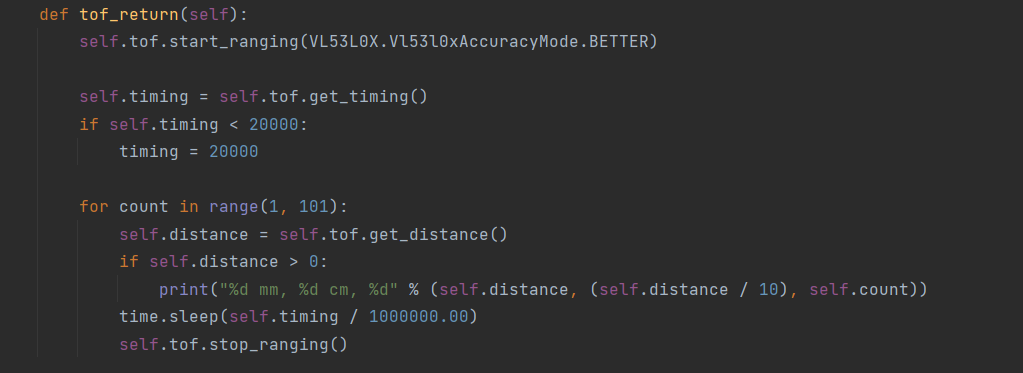
Bu bölümde sistemin çalışması için gereken kütüphaneleri dahil ettik ve yapıcı metotta sensörü , motorları ve eşzamanlı çalışabilmek için gerekli Thread’ları tanımladık.

 şekil 3.1

Bu kod nesne tespit etmek için kullanılan mean shift algoritmasının kodlarıdır. Tespit edilen nesneler in konumuna motorların yönlendirilmesi için koordinat bilgisi bu fonksiyonda belirlenir.

 şekil 3.2

Bu noktada step motorların hareket etmesi ve fire(self) fonksiyonu ile eğer atışa uygun durum var ise atış yapılır.

 şekil 3.3

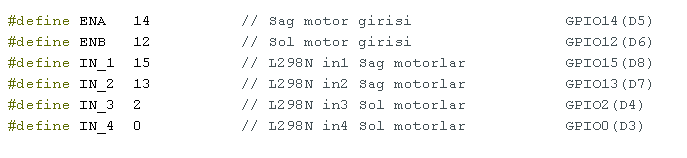
Bu fonksiyon ile kameranın hemen yanında olan hedefin uzaklığının belirlenmesini sağlayan ToF(Time of Flight) sensöründen uzaklık verisi okunur.

 şekil 3.4

Bu adım main metodu ile ilgili sınıfa ait nesne oluşturulup nesne algılama fonksiyonu çalıştırılır.

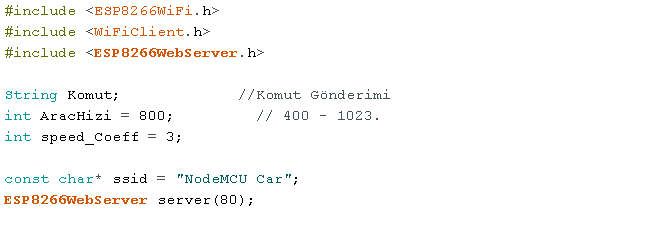
## **2.6 OTONOM ARACIN YÖNETİMİ VE KODLARI**

Daha önceki başlıklarda da belirtildiği üzere aracımızın yönlendirilmesi ve hareketi konusunda gerekli malumatlar verilmişti. Gelin birde kodlar üzerinde bu işlemlerin yazılım işleyişine göz atalım.



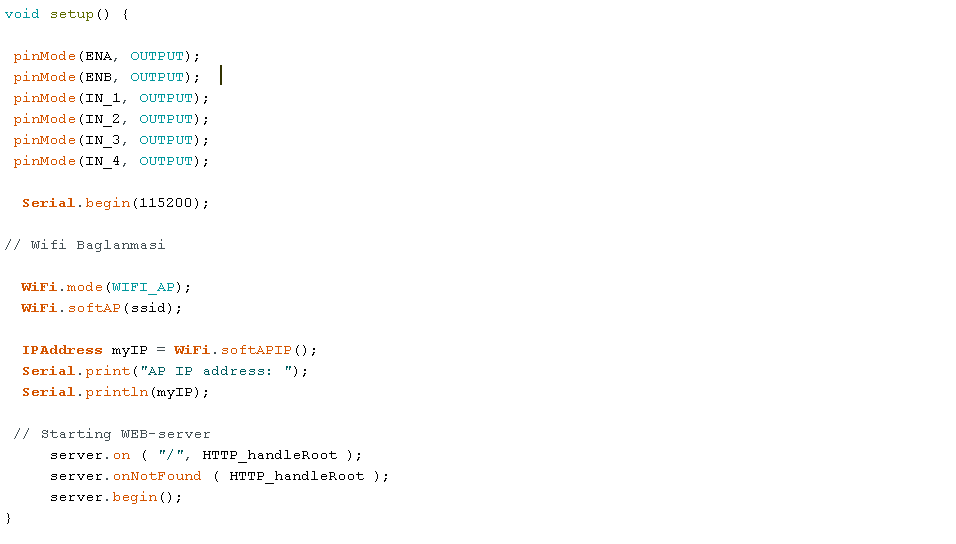
şekil 4.0

Burada yazılan tanımlamalar motor sürücü için gerekli pin tanımlamaları olmaktadır.

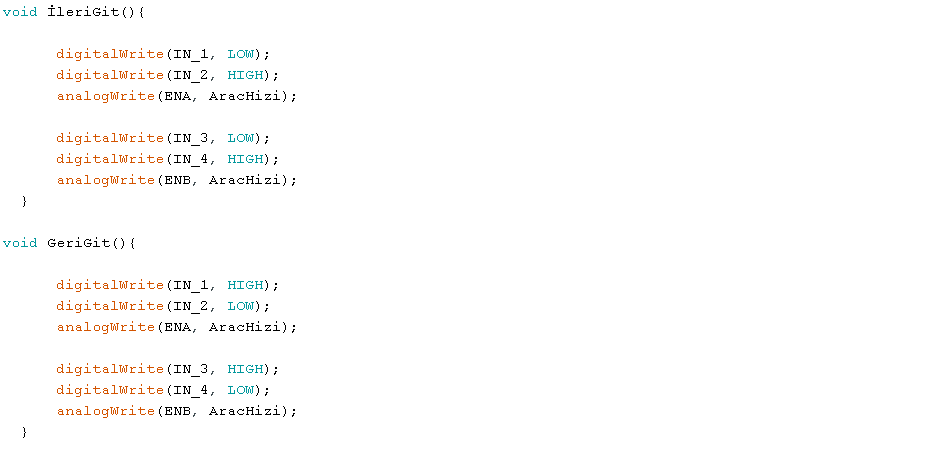


Şekil 4.1

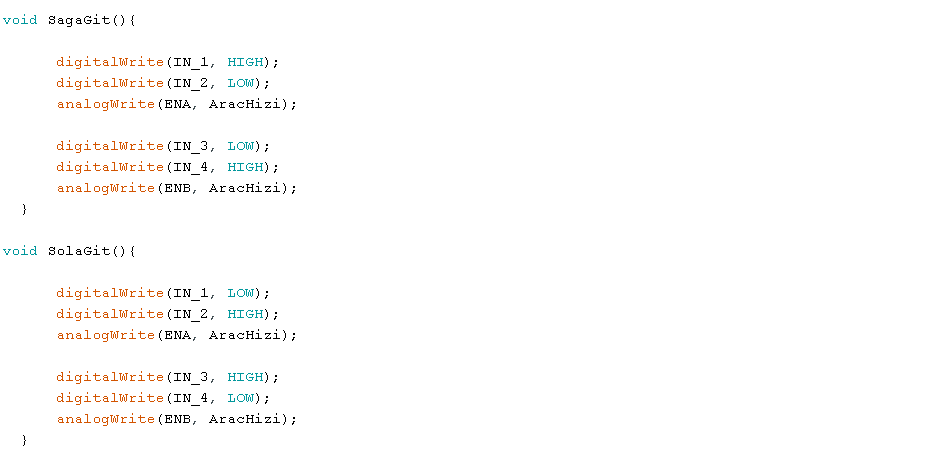
Bu kısımda ise model için gerekli olan kütüphaneler sisteme tanıtıldı ve kütüphaneler haricinde gerekli tanımlamalar belirtildi.

 Şekil 4.2

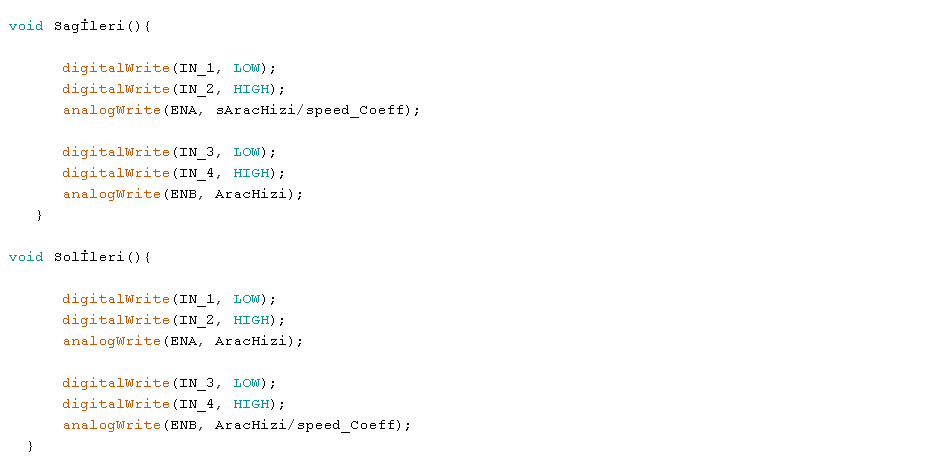
Verilen kod bloğunda programın en temel iki döngüsünden biri olan setup döngüsü yer almaktadır. Setup döngüsünde geliştirilen yazılımın en önemli tanımlamaları belirtilir ve bu döngü içerisinde yalnızca bir kez çalıştırılır. Bu döngümüz içerisinde yalnızca bir kez çalışacak komutlarımız ise motor sürücümüzün entegre edildiği pinlerin çıkış pinleri olarak tanıtılmasıdır.

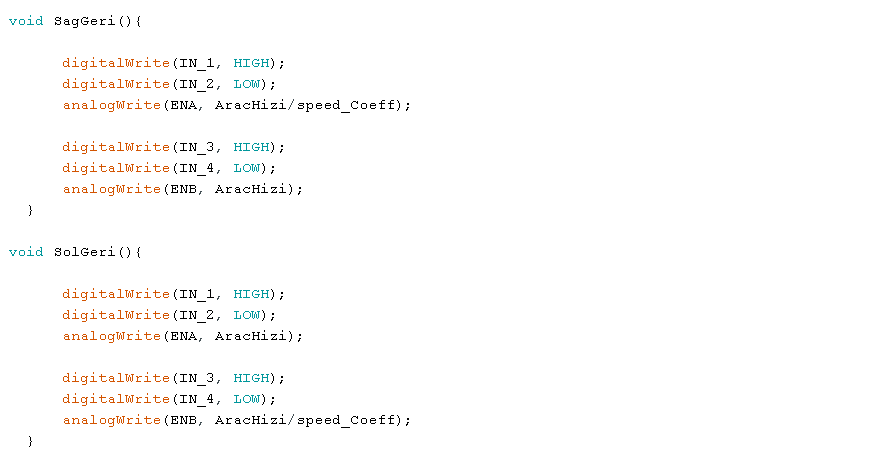
 şekil 4.3

Verilen fonksiyon bloklarında ise aracımızın yönlendirme komutlarını yerine getirecek olan fonksiyonlar tanımlanmıştır.

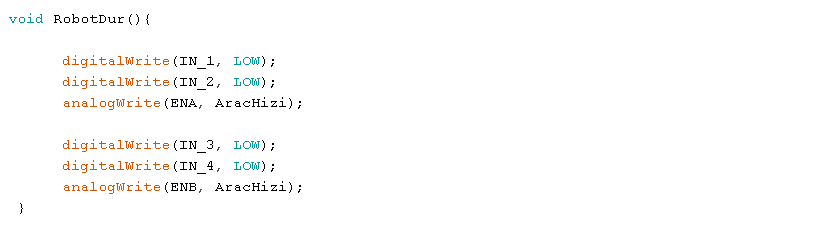
 şekil 4.4

Yukarıda belirtilen fonksiyonlar gibi bu fonksiyonlarda aracımızın yön komutuna göre hareket etmesini sağlayacak olan blokları içerir.

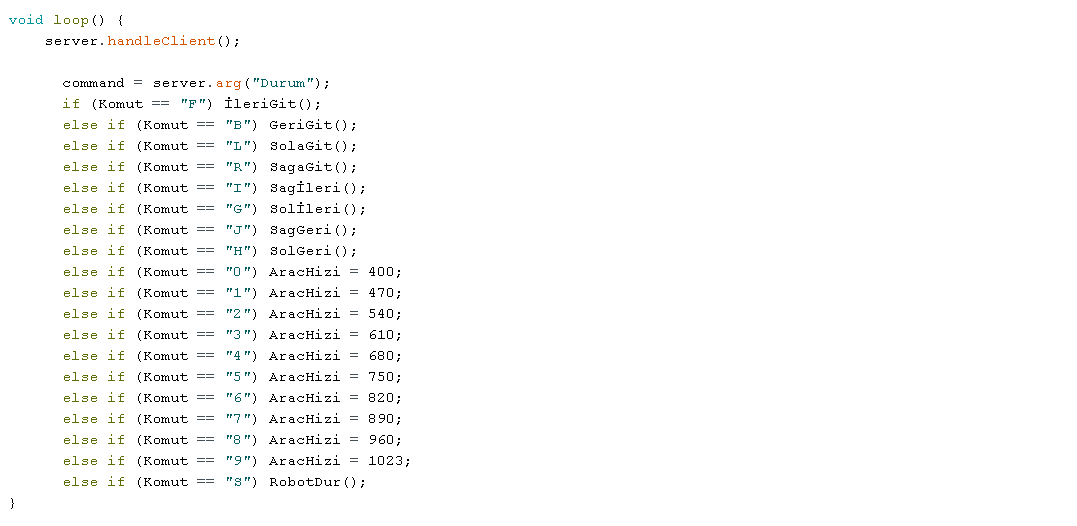
 şekil 4.5

 şekil 4.6

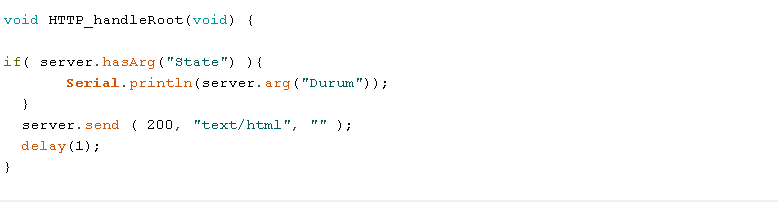
Bu bloklarımızda ise görüldüğü gibi sağ geri ve sol ileri komutları yer almakta ve bu komutlara göre motor sürücü girişlerinin dijital ve analog değerleri belirlenmekte.

 şekil 4.7

Bu fonksiyon bloğumuzda ise artık aracın durdurma yani yeterlilik kriteri komutu verildiği zaman motor sürücü parametrelerinin alacağı değerler belirlendi.

 şekil 4.8

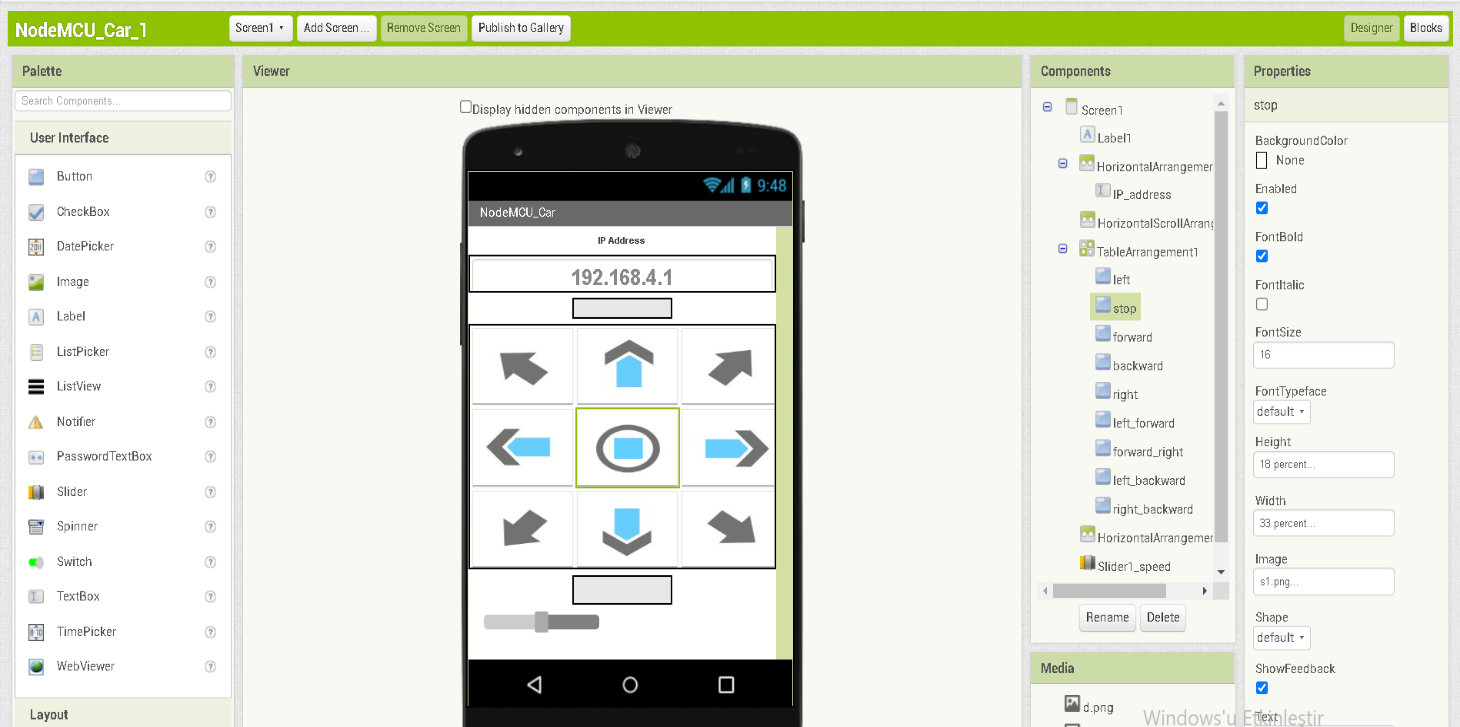
Verilen döngü bloğumuz en temel iki döngüden biri olan loop döngüsüdür bu döngünün içerisine geliştirici kartımızdan harici besleme kaynak gücü kesilene kadar görevini icra ettirecek kodları yazıyoruz ve görüldüğü üzere koşul yapıları ile yönlendirme pozisyonları belirtilmiş olarak sunulmuştur.

 şekil 4.9

Son olarak ise verilen bloğumuzda bir durum incelemesi yapılmaktadır.

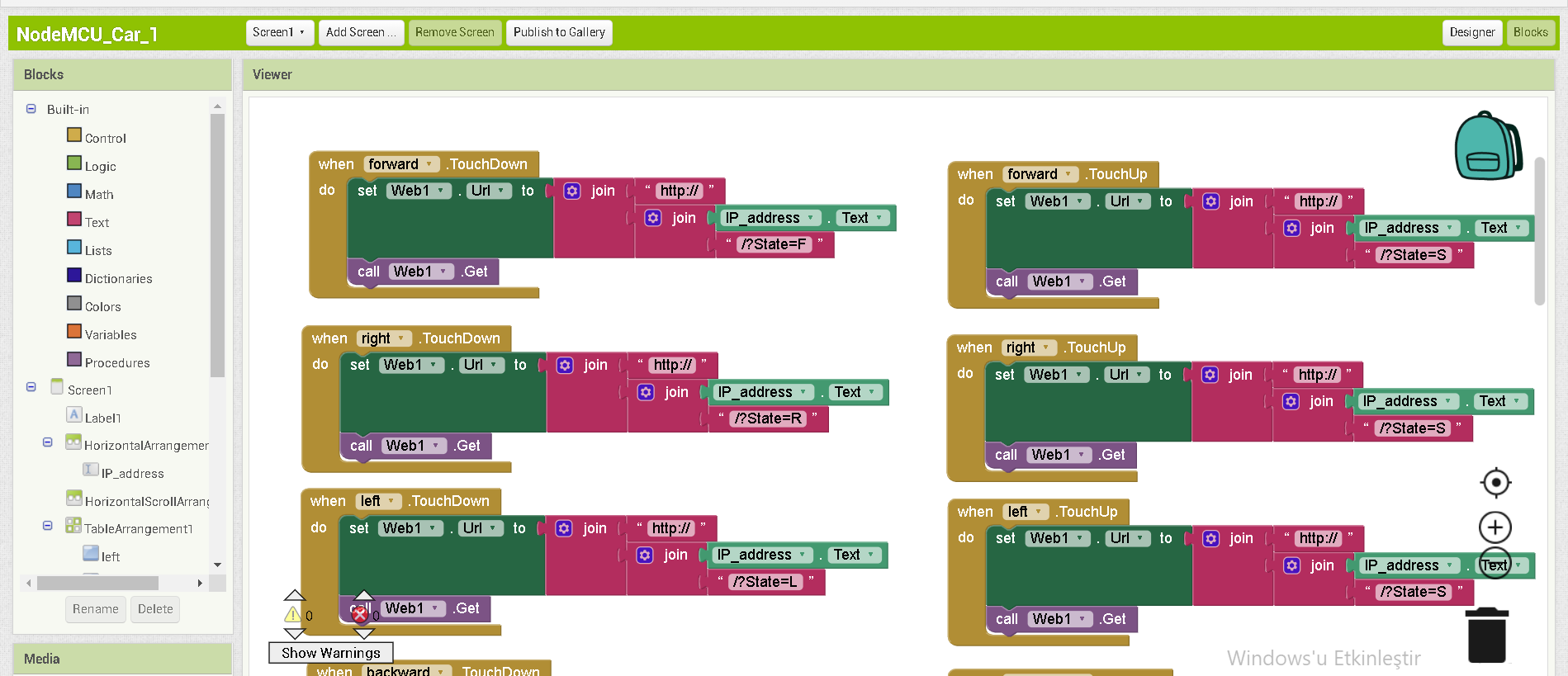
## **2.7 MOBİL UYGULAMA İLE OTONOM İMHA ARACININ YÖNETİLMESİ**

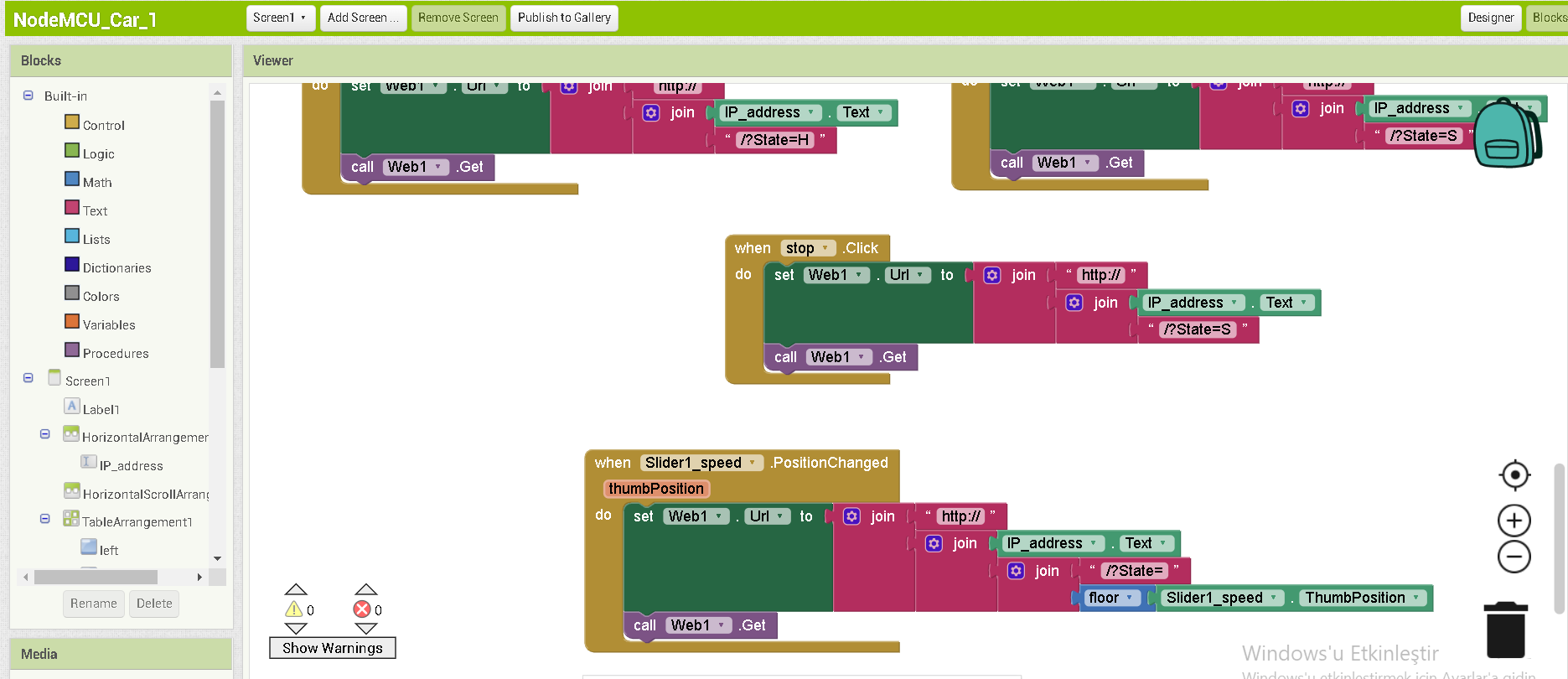
Geliştirilen modelimizde daha önce de bahsi geçtiği gibi aracımız yetkili bir kullanıcı tarafından kullanılır. Aracı kullanacak olan yetkili kullanıcı ise geliştirilmiş olan mobil uygulamayı akıllı cihazına indirerek gerekli yönlendirme işlemlerini yerine getirebilir şimdi gelin bu uygulama ara yüzünün tasarımına birlikte göz atalım.



şekil 5.0

Görüldüğü üzere mobil uygulamamız bir tasarım ara yüzü uygulamasında tasarlanmış olup gerekli apk uzantılı dosyalar daha sonra tasarlanmış olan araç uygulamamızın yönetimi için indirilir ve dış ortama aktarılır.

 şekil 5.1

 şekil 5.2

Verilen görsellerde ise uygulamamızın blok diyagram yapısı ele alınmıştır.

## **2.8 KULLANILAN MATERYALLER**

Projemizde kullandığımız materyaller şunlardır:

1-)Raspberrypi4

* Bu geliştirme kartı projemizde görüntü işleme ve silah sistemini yönlendirmesi için kullanılmıştır.

2-)L298N motor sürücüsü(3 adet)

* Bu motor sürücü kartı silah sisteminde eksenlerde bulunan step motorları ve aracın hareketini sağlayan DC motorları kontrol etmek için kullanılmıştır

3-)5adet redüktörlü DC motor

* Bu motorlardan 4 tanesi aracın hareketi bir tanesi de silah sisteminde mermiyi namluya itmek için kullanılmıştır

4-)esp8266 model geliştirici kart

* Bu geliştirme kartı aracın kontrolünü sağlamak için kullanılmıştır

5-)modeli beslemek için güç kaynağı

* Bu güç kaynağı sistemin çalışabilmesini sağlayan gücü üretmek için kullanılmıştır

6-)Raspberry pi CSI kamera modülü

* CSI kamera modülü raspberry pi cihazının görüntü alabilmesini sağlamaktadır

7-)4 adet tekerlek

* Tekerler aracın hareket etmesini , zemin ile cihaz arasındaki mesafeyi koruması için kullanılmıştır

8-)donanım entegrelerinin birleştirilmesi için kablolar

* Gerekli donanımlarının bağlantıları için kullanılmıştır

9-)1 adet akım ve voltaj ayarlı regülatör kartı

* Raspberry pi cihazının harici güç kaynağından bağımsız hale gelmesi için kullanılmıştır

10-)silah sistemi için bir kalıp

* Silahın genel tasarımı içindir

11-)1 adet VL53OX ToF sensör

* Bu sensör hedefin hassas uzaklık bilgisini elde etmek için kullanılmıştır

12-)2 adet Creality step motor

* Silah sisteminin yatay ve dikey eksende hareketi için kullanılmıştır

13-)16 adet 1.5 V AA pil

* Bu piller aracın hareketi ve raspberry pi cihazının güç beslemesi için kullanılmıştır

14-)1 adet 5V röle modülü

* Röle kartı silahın atış motorlarının çalışmasını kontrol etmek için kullanılmıştır

15-)2 adet 18650 1200 mA li-on pil

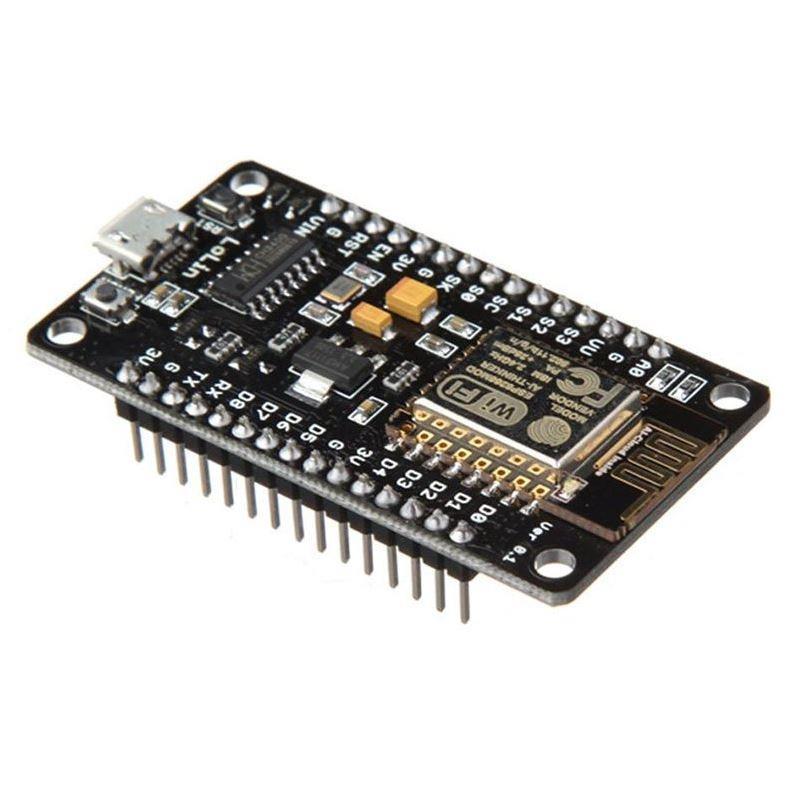
* li-on piller silahın atış motorlarının güç beslemesi için kullanılmıştır

16-)2 adet DC motor

* Bu motorlar silahın namlusuna gelen mermileri hızlandırılmasını sağlamaktadır

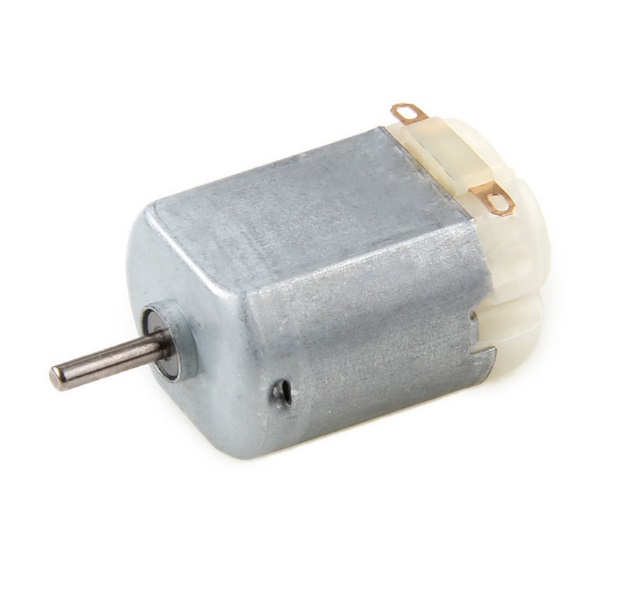
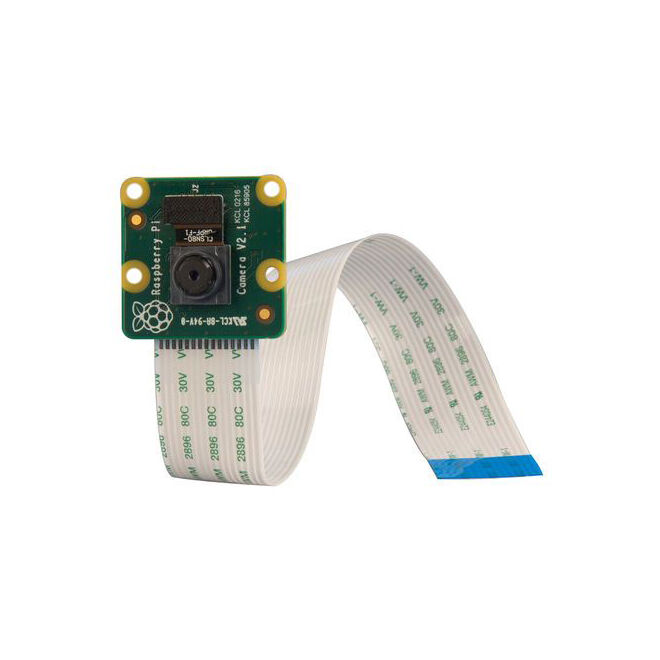


Şekil 6.0



Şekil 6.1

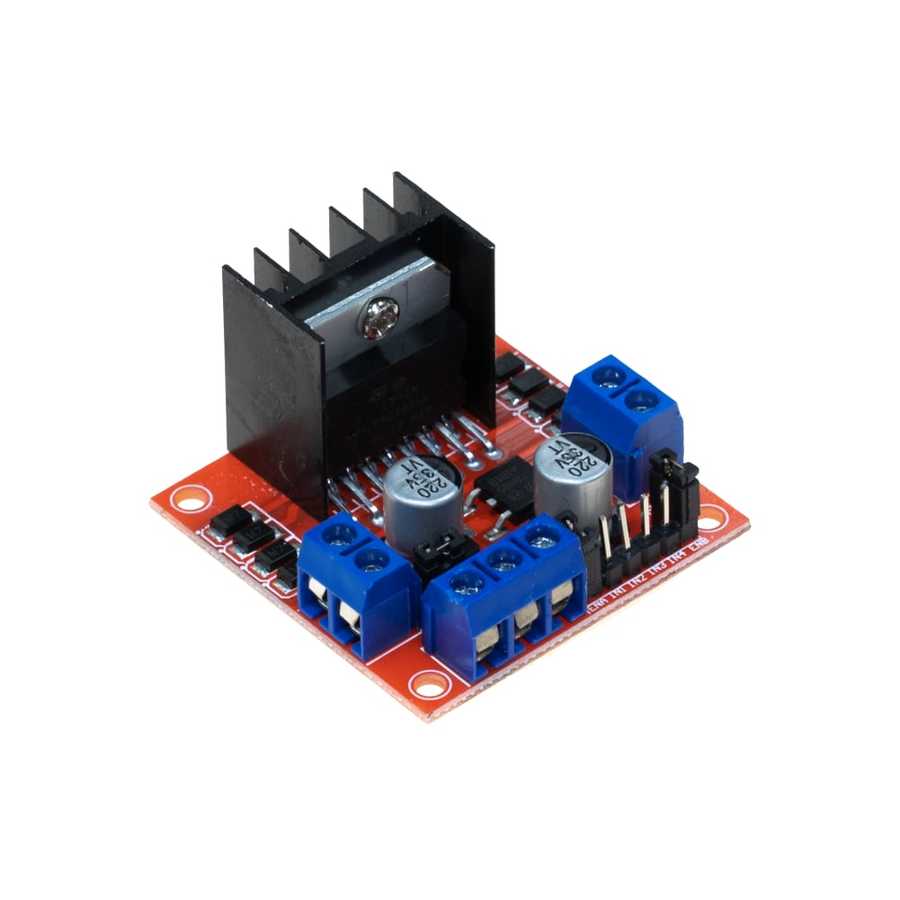


Şekil 6.2 Şekil 6.3 

Şekil 6.4

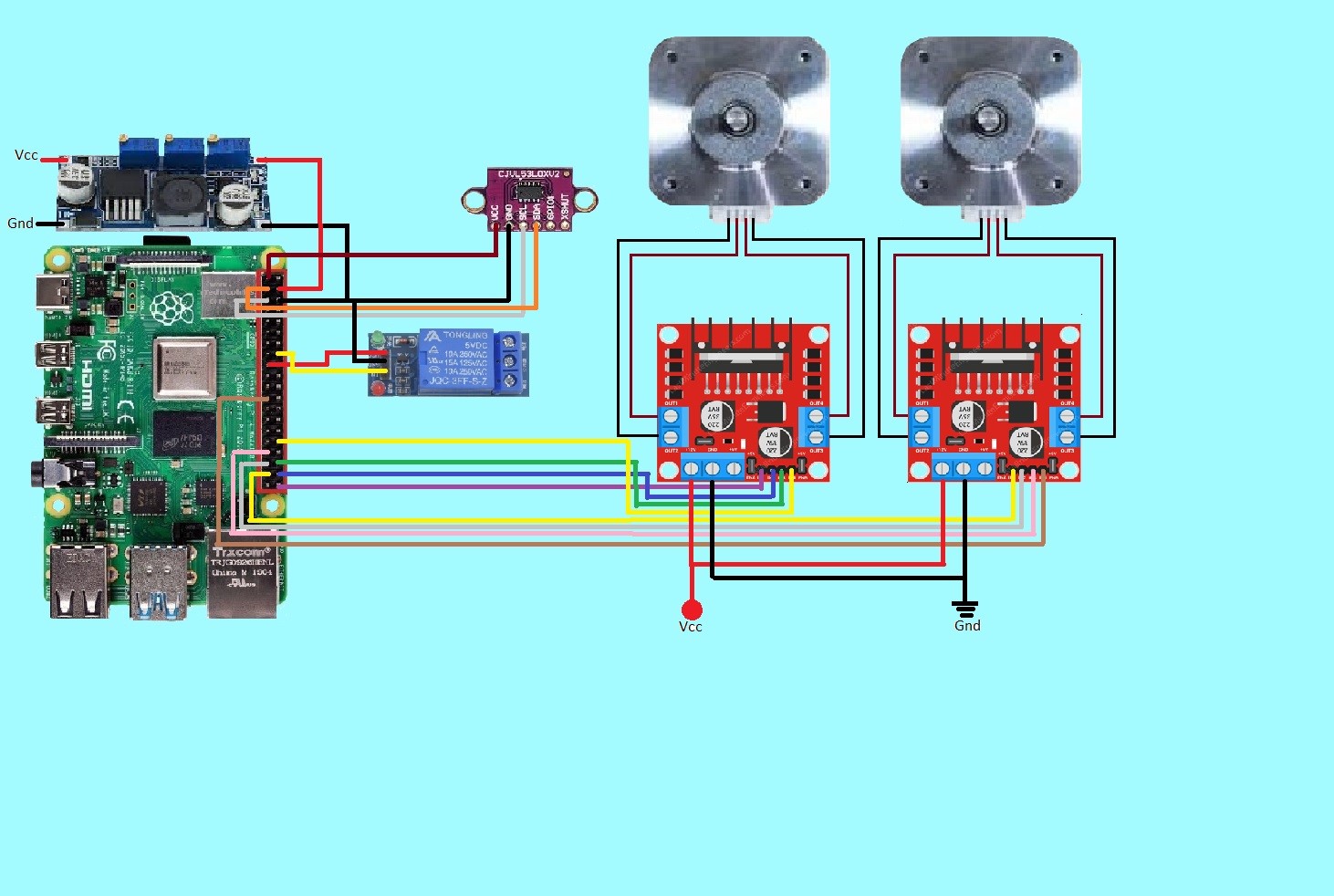
Şekil 6.5 

Şekil 6.6



Şekil 6.7

## **2.9 SİLAH SİSTEMİNİN DEVRE ŞEMASI**

 şekil 7.0

# **3 SONUÇ**

Otonom imha sistemin hedefleri tespit etme ve hedefleri takip etmesi için test edilmiştir. Test sonucunda otonom imha sistemimiz belirlenen bölgedeki hedefe odaklanıp başarıyla takip etmiştir. Takip sonucunda odaklanan hedef mesafe sensörünün mesafesine ve görüş açısına girdiği noktada hedeflere atış yapılıp başarı ile vurulmuştur. Otonom imha sisteminin aracı istenilen bölgelere ulaşarak uzaktan kontrol ile minimum insan gücü ile hedef imhası olağan hale gelmiştir.

# **KAYNAKÇA**

<https://www.researchgate.net/profile/Okan-Topcu/publication/279449032_Intelligent_Autonomous_Systems_Akilli_Otonom_Sistemler/links/5592f3a508aed7453d464cbe/Intelligent-Autonomous-Systems-Akilli-Otonom-Sistemler.pdf>

<http://earsiv.anadolu.edu.tr/xmlui/handle/11421/18722>

<https://www.ulakbilge.com/makale/pdf/1624967182.pdf>

<https://www.researchgate.net/profile/Mehmet-Caliskan-3/publication/309160134_Design_of_shock_absorbers_for_weapon_systems_by_autonomous_software/links/5801f3a008ae23fd1b670037/Design-of-shock-absorbers-for-weapon-systems-by-autonomous-software.pdf>

<https://dergipark.org.tr/en/pub/gazimmfd/issue/6706/89218>

<https://dergipark.org.tr/en/pub/ajit-e/issue/62376/867383>

<https://tusside.tubitak.gov.tr/sites/images/Tusside/insansiz_lojistik_aglarinin_geleceginin_degerlendirilmesi_io.pdf>

<https://dergipark.org.tr/en/pub/politeknik/issue/33364/931490>

<https://docs.opencv.org/4.x/>

<https://www.raspberrypi.com/documentation/>

<https://docs.arduino.cc/>

<https://mesutpiskin.com/blog/tag/mean-shift>

# **EKLER**

## **Ek1 : Projeyle İlgili Görseller**

Şekil1.0 : Raspberry pi indirme bağlantısı

Şekil1.1 : Raspberry pi kurulum aşaması

Şekil2.0 : Step motor kontrolü için gerekli kütüphane

Şekil2.1 : Tanımlamalar ve yapıcı metot

Şekil2.2 : Motor hızı belirleme

Şekil2.3 : Motorları çalıştırma

Şekil2.4 : Adım sırasını ve adım arasındaki süreyi belirleme

Şekil3.0 : Kütüphaneleri dahil etme ve tanımlama

Şekil3.1 : Mean shift algoritması

Şekil3.2 : axis kontrolü ve ateş etme

Şekil3.3 : sensörden veri okuma

Şekil3.4 : Sistemi çalıştırma

Şekil4.0 : esp için pin tanımlamaları

Şekil4.1 : kütüphaneleri dahil etme

Şekil4.2 : pin modları ve wifi modu belirleme

Şekil4.3 : ileri ve geri gitme fonksiyonları

Şekil4.4 : sağa ve sola gitme fonksiyonları

Şekil4.5 : sağ ileri ve sol ileri gitme fonksiyonları

Şekil4.6 : sağ geri ve sol geri gitme fonksiyonları

Şekil4.7 : motorları durdurma fonksiyonları

Şekil4.8 : yön bilgisi elde etme

Şekil4.9 : durum bilgisi elde etme

Şekil5.0 : mobil uygulama arayüzü

Şekil5.1 : mobil uygulama tasarımı

Şekil5.2 : mobil uygulama tasarımı

Şekil6.0 : raspberry pi

Şekil6.1 : esp8266

Şekil6.2 : jumper kablo

Şekil6.3 : DC motor

Şekil6.4 : raspberry pi CSI kamera

Şekil6.5 : redüktörlü DC motor

Şekil6.6 : teker

Şekil6.7 : L298n motor sürücü kartı

Şekil7.0 : silah sistemi devresi