

KARADENİZ TEKNİK ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ



QUADCOPTER UÇUŞ KONTROLCÜSÜNÜN TASARLANMASI

TASARIM PROJESİ

HAZIRLAYANLAR
MUSTAFA TOPSAKAL
ERHAN YILMAZ
TAYFUR YÜCEL

279965

280031

290253

2015-2016 GÜZ DÖNEMİ

KARADENİZ TEKNİK ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

QUADCOPTER UÇUŞ KONTROLCÜSÜNÜN TASARLANMASI

TASARIM PROJESİ

HAZIRLAYANLAR
MUSTAFA TOPSAKAL
ERHAN YILMAZ
TAYFUR YÜCEL

279965

280031

290253

Bu projenin teslim edilmesi ve sunulması tarafımdan uygundur.

Danışman : Yrd. Doç. Dr. Sedat GÖRMÜŞ

2015-2016 GÜZ DÖNEMİ



IEEE Etik Kuralları

IEEE Code of Ethics



Mesleğime karşı şahsi sorumluluğumu kabul ederek, hizmet ettiğim toplumlara ve üyelerine en yüksek etik ve mesleki davranışta bulunmaya söz verdiğimi ve aşağıdaki etik kurallarını kabul ettiğimi ifade ederim:

1. Kamu güvenliği, sağlığı ve refahı ile uyumlu kararlar vermenin sorumluluğunu kabul etmek ve kamu veya çevreyi tehdit edebilecek faktörleri derhal açıklamak;
2. Mümkün olabilecek çıkar çatışması, ister gerçekten var olması isterse sadece algı olması, durumlarından kaçınmak. Çıkar çatışması olması durumunda, etkilenen taraflara durumu bildirmek;
3. Mevcut verilere dayalı tahminlerde ve fikir beyan etmelerde gerçekçi ve dürüst olmak;
4. Her türlü rüşveti reddetmek;
5. Mütenasip uygulamalarını ve muhtemel sonuçlarını gözeterek teknoloji anlayışını geliştirmek;
6. Teknik yeterliliklerimizi sürdürmek ve geliştirmek, yeterli eğitim veya tecrübe olması veya işin zorluk sınırları ifade edilmesi durumunda ancak başkaları için teknolojik sorumlulukları üstlenmek;
7. Teknik bir çalışma hakkında yansız bir eleştiri için uğraşmak, eleştiriyi kabul etmek ve eleştiriyi yapmak; hatları kabul etmek ve düzeltmek; diğer katkı sunanların emeklerini ifade etmek;
8. Bütün kişilere adilane davranmak; ırk, din, cinsiyet, yaş, milliyet, cinsi tercih, cinsiyet kimliği, veya cinsiyet ifadesi üzerinden ayrımcılık yapma durumuna girişmemek;
9. Yanlış veya kötü amaçlı eylemler sonucu kimsenin yaralanması, mülklerinin zarar görmesi, itibarlarının veya istihdamlarının zedelenmesi durumlarının oluşmasından kaçınmak;
10. Meslektaşlara ve yardımcı personele mesleki gelişimlerinde yardımcı olmak ve onları desteklemek.

IEEE Yönetim Kurulu tarafından Ağustos 1990'da onaylanmıştır.

ÖNSÖZ

“Raspberry Pi ile Quadcopter uçuş kontrolcüsünün tasarlanması” konulu bu çalışma, Karadeniz Teknik Üniversitesi, Bilgisayar Mühendisliği Bölümü’nde “Tasarım projesi” olarak hazırlanmıştır.

Tasarım projesi çalışması sırasında bilgisini ve desteğini bizden esirgemeyen Danışman Hocamız Yrd. Doç. Dr. Sedat GÖRMÜŞ’e teşekkür ederiz.

MUSTAFA TOPSAKAL

ERHAN YILMAZ

TAYFUR YÜCEL

Trabzon 2015

İÇİNDEKİLER

Sayfa No

ÖNSÖZ.....	II
İÇİNDEKİLER.....	III
ÖZET.....	IV
1. QUADCOPTER.....	1
1.1 Quadcopter Tarihi.....	1
1.2 Quadcopter Parçaları.....	7
2. RASPBERRY Pİ VE RASBİAN.....	8
2.1 Raspberry Pi	8
2.2 Raspbian Kurulumu	10
2.3 Raspbian Güncelleştirmeleri	12
2.4 I2C Protokolünün Aktif Edilmesi.....	12
2.5 WiringPi Kütüphanesinin Güncelleştirilmesi.....	17
3. ANDROID UYGULAMA VE HABERLEŞME.....	19
3.1 Quadcopter Kontrolü için Android Uygulaması.....	19
3.2 Raspberry Pi ile Telefon Haberleştirilmesi.....	19
4. QUADCOPTER HAREKETİ.....	22
4.1 Jiroskop	22
4.2 I2C Protokolü.....	22
4.3 ESC İle Motor Kontrolü.....	24
4.4 Raspberry Pi GPIO Programlama.....	25
5. SONUÇLAR... ..	26
6. KAYNAKLAR.....	27

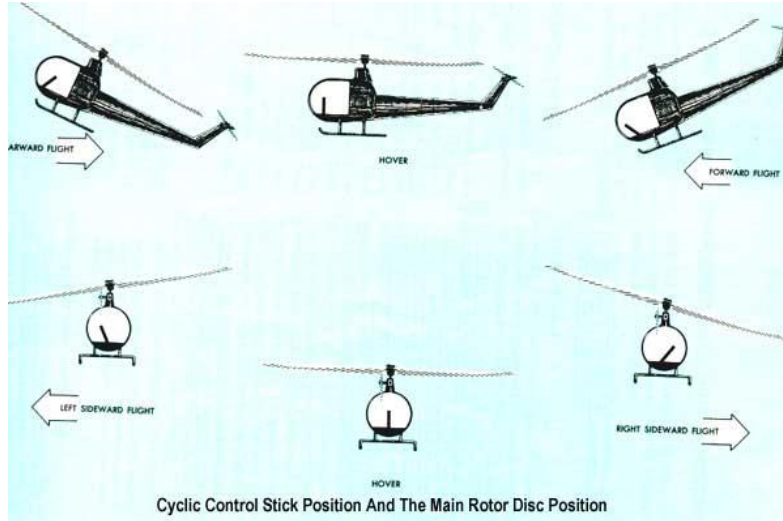
ÖZET

Raspberry Pi, Raspberry Pi Foundation tarafından okullarda çocuklara küçük yaşta temel bilgisayar eğitimini vermek amacı ile yola çıkmış, kredi kartı büyüklüğünde tek board'lu bir bilgisayardır. Seçeceğiniz herhangi bir SD kart üzerine ücretsiz bir işletim sistemi yükleyebilir ve bunun ardından cihazı boot ederek dakikalar içinde tam teşekküllü taşınabilir bir mini bilgisayara sahip olabilirsiniz. Projede Raspberry Pi ile Quadcopter Uçuş Kontrolünün Tasarlanmasını ele alındı. Android bir cihazla motorların güçlerinin artırıp azaltma işlemleri ve uçağın farklı yönlere doğru uçuşması sağlandı. Gyroscope sensörü ile orta kısmında dönen mil üzerindeki algılayıcılar ile 3 boyutlu ve ivme ölçebilen hareket sağlanacaktır. Motorların Raspberry Pi'de aktif olması için terminal üzerinde I2C kütüphaneleri Raspberry Pi'ye yüklendi ve I2C fonksiyonları ile motor hareketleri sağlandı. Raspberry Pi foundation tarafından kurulum kolaylaşması amacı için NOOBS(New Out Of the Box) dağıtım sistemi kuruldu. Bu dağıtımın bir karta birden fazla işletim sistemi kurabilme, açılış yöneticisi ayarlayabilme gibi çeşitli artıları vardır. Bu dağıtım, yeni kullanıcılara başlangıç için tüm işletim sistemi alternatiflerini görmeleri için güzel bir ortam sağlar. Raspberry Pi'de bulunan pinler üzerinde temel veri giriş ve çıkış yapabilmemiz için C kütüphanesine ek olarak WiringPi kütüphanesi eklendi.

1. QUADCOPTER

1.1 QUADCOTER TARİHİ

Quadcopter, aynı zamanda quadrotor helikopter, quadrocopter, quadrotor ve quadraopter isimleri verilen quadcopter, dört motor tarafından kaldırılan ve yürütülen bir multicopterdir. Sabit kanatlı hava araçlarının aksine quadcopterler kaldırma kuvvetini dönen kirislerden elde ettiği için rotorcraft sınıfına girer. Helikopterlerin aksine, quadcopterler genellikle simetrik olarak yerleştirilmiş pervaneler kullanır. Bunlar kollektif olarak ayarlanır ancak helikopterdeki gibi “cyclic” değildir. Cihazın kontrolü motorların dönüş hızını değiştirerek, yani tork yükünü ve itme/kaldırma karakteristiklerinde farklılık yaratarak sağlanır. [1]



Şekil 1.1

Quadcopter konfigürasyonları uçuş tarihinde sürekli görülen torka bağlı kontrol sorunlarını ve kuyruk motorundan kaynaklanan verim kaybını bertaraf etmek amacıyla ortaya çıkmıştır. 1920 ve 30larda insanlı uçuş için tasarımlar yapılmıştır. Bu araçlar havadan ağır olup dikey kalkış ve iniş(VTOL) yapabilen cihazların öncülerindendir. Ancak ilk prototipler düşük performanslı, arkasından gelenler ise pilota düşen iş yükü, düşük stabilite ve sınırlı kontrol imkanından dolayı kullanılmamıştır.

Günümüzde quadcopter tasarımları insansız hava aracı(UAV) olarak popülerleşmiştir. Bu araçlar elektronik kontrol sistemleri ve elektronik sensörlerle stabilize olur. Küçük tasarımları ve çevik manevra kabiliyeti sayesinde hem kapalı hem açık alanda uçurulabilirler.

Quadcopterlerin boyut olarak benzer helikopterlere karşı bazı avantajları vardır. İlki, quadcopterlerin pervane açısını değiştirmek için kullanılan mekanik bağlantılara ihtiyacı yoktur.

Böylelikle tasarım ve bakımı basitleşir. İkincisi, dört motor kullanıldığı için motorların çapının küçük olması, dolayısıyla uçuşta daha uçuş sırasında daha az kinetik enerjiye sahip olması, böylelikle olası bir çarpışma halinde motorların daha az hasar alması. Yapım ve kontrol kolaylığı sayesinde amatör model uçuş projelerinde quadcopter şaseleri sıklıkla kullanılmaktadır.

Tarihi

İlk Denemeler

Oehmichen No.2 (1920)

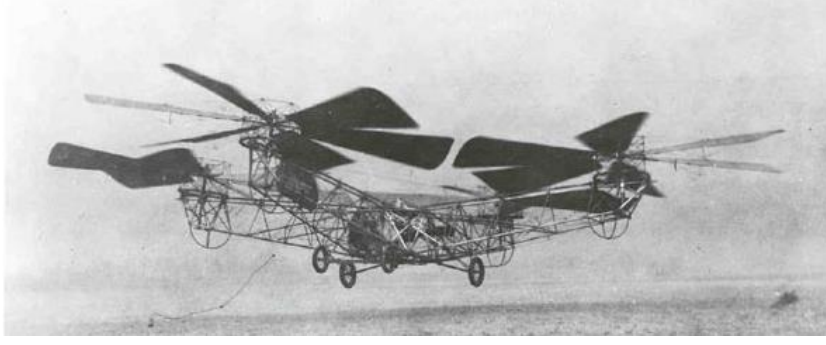
1920lerde Etienne Oehmichen rotorcraft araçlarla denemeler yapmıştır. Deneme yaptığı altı araçtan bir tanesi dört motorlu sekiz pervaneli No.2'dir. Oehmichen No.2 dört kol ucundaki iki pervaneli motorlarla birlikte çelik şase kullanmıştır. Beş pervane araca dikey olarak stabilize edilmiş olup, burun kısmında bir diğer pervane yönlendirme sağlarken, son iki pervane ise yatay olarak bağlanıp, ileriye itme sağlamak için kullanılmıştır. Zamanına göre önemli miktarda stabilite sunan bu araç, 1920lerin ortalarında binden fazla test uçuşu yapmıştır. 1923 yılında havada birkaç dakika kalabilmiş, 14 Nisan 1924'te ise 360 metre giderek FAI(Uluslararası Havacılık Federasyonu) helikopterler için mesafe rekorunu kırmıştır. Dairesel bir rotayı tamamlayıp daha sonra ilk 1 kilometrelik kapalı devre uçuşu bir rotorcraftla tamamlamıştır.

de Bothezat helicopter(1922)

Dr. George de Bothezat ve Ivan Jerome tarafından geliştirilen bu araç, X şeklinde şaseye bağlı adet pervaneden oluşur. Değişken yükseklikli iki küçük pervane itme ve sapmayı kontrol etmek için kullanılmıştır. Amerikan ordusu tarafından yapılan bu araç ilk uçuşunu Ekim 1922'de gerçekleştirmiştir ve 1923'ün sonuna kadar 100 uçuşu tamamlamıştır. Ulaştığı maksimum yükseklik ise 5 metredir. Gösteri imkanı olmasına rağmen, yeterince güçlü olmayan, tepkisiz, mekanik olarak karmaşık ve hassastır. Ayrıca havada sabit dururken yana hareket etmek için pilota çok yük düşüyordu.

Convertawings Model A Quadroter (1956)

Bu özgün helikopterin çok daha büyük bir sivil ve askeri quadrotora prototip olması planlanmıştı. Tasarımı iki adet motorun v kayışıyla dört rotoru döndürmesi şeklindedir. Kontrol ise kuyruk motoruna ihtiyaç duyulmadan, motorların itme gücünü değiştirerek sağlanmıştır. 1950lerin ortasında birçok kez başarılı bir şekilde uçmuştur. Ayrıca ileri uçuş yapabilen ilk dört rotorlu hava taşıtıdır. Sipariş düşüklüğü sebebiyle proje iptal edilmiştir. Convertawings ayrıca 19ton maksimum kapasiteli, 4.9ton taşıyıp 278 km/s hızıyla 300 mil menzile gidebilen bir Model E tasarlamıştır.



Şekil 1.2

Curtiss-Wright VZ-7 (1958)

Curtiss-Wright VZ-7, Curtiss-Wright firması tarafından Amerikan Ordusu için dikey iniş ve kalkış(VTOL) yapabilecek şekilde tasarlanmıştır. Kontrolü dört motorun itme gücünü değiştirerek sağlanmıştır.

Güncel Değişiklikler

İnsansız Hava Araçları(UAV) son yıllarda birçok uygulama için yoğun olarak kullanılmaktadır. Hava araçlarındaki manevra kabiliyetine ve havada sabit kalma özelliğine olan ihtiyaç günümüzdeki quadcopter araştırmalarına sebep olmuştur. Dört motorlu tasarım hem basit hem de güvenilir ve manevra kabiliyetinin yüksek olmasına olanak sağlar. Son teknoloji araştırma çalışmaları quadcopterlerin çoklu-araç iletişimi, çevre araştırmaları ve manevra kabiliyeti için geliştirmeleri uygulanabilir hale getirmektedir. Eğer bu gelişen teknoloji birleşebilirse, quadcopterler başka araçlar tarafından yapılması mümkün olmayan gelişmiş otonom görevleri gerçekleştirebilecek. Şu anki programlardan bazıları:

- Bell Boeing Quad TiltRotor sabit quadcopter konseptini tilt rotolarla kombine ederek C-130 boyunda bir konseptle ileriye götürüyor.
- Aermatica Spa'dan Anteos ilk resmi izinli sivil hava sahasında uçabilen rotary kanatlı RPA(uzaktan kontrol edilen hava aracı) olup, ileride ayrılmamış hava sahasında uçuş yapabilecek ilk hava aracıdır.
- AeroQuad ve ArduCopter açık kaynaklı donanıma ve yazılıma sahip, Arduino tabanlı DIY quadcopter projesidir.
- Parrot AR.Drone üstünde Parrot SA tarafından üretilmiş kameralar olan küçük RC bir quadcopterdir. Akıllı telefon veya tabletlerle kontrol edilmek için tasarlanmıştır. Haziran 2013'te Paris Air Show'da 500.000 den fazla satıldığı açıklanmıştır.



Şekil 1.3



Şekil 1.4

Parrot.AR Drone'un uçan prototipi. Parrot.AR Drone 2.0 kalkarken. Nevada, 2012

Kullanım

Araştırma Platformu

Quadcopterler üniversite araştırmalarında, uçuş kontrol teorisi, navigasyon, gerçek zamanlı sistemler ve robot teknolojisi gibi birçok alanda test ve değerlendirme de kullanılır. Son yıllarda çoğu üniversitede giderek artan bir şekilde karışık manevralar yapan quadcopterler görülmüştür. Sürü halindeki quadcopterler formasyon halinde havada durarak takla, halkadan geçme ve grup olarak bir pencereden geçme gibi karışık görev görevleri uygulayabilir.

Quadcopterlerin çok amaçlı test platformu olarak kullanılmasında birçok avantaj vardır. Fiyatlarının düşüklüğü, boyut çeşitliliği ve basit mekanik tasarımı sayesinde amatörler tarafından toplanıp bakımları yapılabilir. Çok dallı yapısı sayesinde birçok farklı alandan akademisyenin bir araya girerek geliştirme yapmasına imkan tanır. Quadrocopter projeleri genel olarak bilgisayar mühendisliği, elektrik mühendisliği ve mekanik mühendisliği uzmanlarının bir araya gelmesiyle oluşur.

Manevra kabiliyetleri çok yüksek olduğu için quadcopterler her durumda ve ortamda kullanılabilir. Ayrıca otonom şekilde insana ihtiyaç olmadan uçabilirler. Bunlar yıllar boyunca artan araştırmaların başlıca sebebidir.

Dünya çapında bazı araştırma laboratuvarları daha gelişmiş kontrol teknikleri ve araştırmalar için çalışmaktadır. Bunların başlıcaları MIT Aerospace Controls Lab, ETH Flying Machine Arena ve Pennsylvania General Robotics, Automation, Sensing and Perception (GRASP)'dır.

Asker ve Güvenlik Güçleri

Quadcopter insansız hava araçları asker ve güvenlik güçleri tarafından takip, keşif ve kentsel alandaki arama kurtarma çalışmaları için kullanılır. Bunun bir örneği küçük ve sessiz bir

UAV olan, Kanadalı Aeryon Labs tarafından üretilen Aeryon Scout'tır. Şirket Orta Amerika'daki uyuşturucu kaçakçısının ormanın derinliklerinde kısıtılması sırasında önemli bir rol oynadığını iddia etmiştir.

Ticari

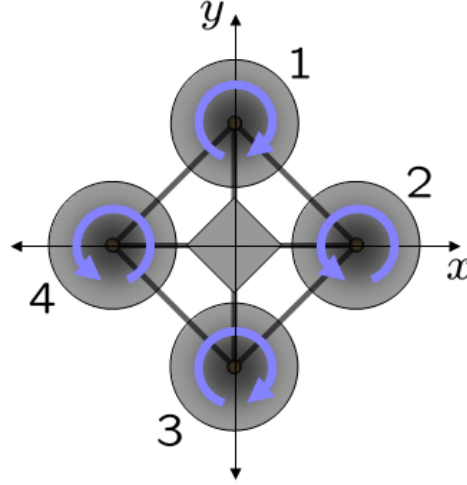
Quadcopterlerin en çok kullanıldığı alan havadan çekimlerdir. Maliyet düşüklüğü ve otonom uçuş özelliği sayesinde quadcopterler bu işler için çok uygundur. Quadcopterler ile havadan fotoğraf çekmek GPS koordinatlarını girip bir tuşa basmak kadar basittir. On-board kameraları sayesinde kullanıcılar görüntüyü canlı olarak yere aktarabilir. Emlak fotoğraflamadan endüstriyel sistem denetimine kadar birçok firma tarafından kullanılır. Çeşitli kuruluşlar quadcopterlerin kapalı devre televizyon özelliklerinden yararlanarak yerdeki hareketliliği takip eder.

Uçuş Kontrolü

Her motor merkezinden bir itme gücü ve tork üretir, bunun yanında cihazın uçuş yönünün aksine bir sürüklenme kuvveti oluşur. Bütün motorlar eşit açısız hızda dönüyorsa, birinci ve üçüncü motorla ikinci ve dördüncü motor ters yönlerde dönüyorsa, toplam aerodinamik tork ve bundan dolayı yaw axise etkiyen açısız hızlanma sıfırdır. Yani konvansiyonel helikopterlerde kullanılan sapmayı stabilize eden motora ihtiyaç yoktur. Sapmaya aerodinamik torkların arasındaki uyuşmazlık sebep olur.

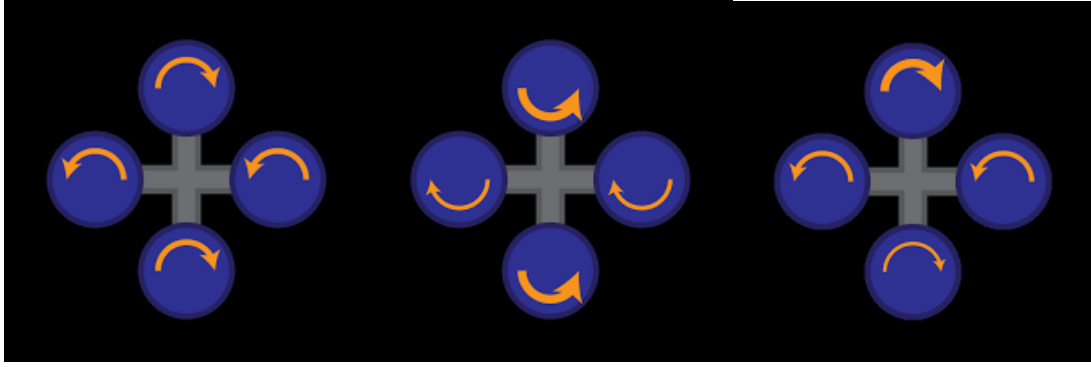
Pitch ve roll eksenlerine etkiyen açısız hızlanma yaw eksenine etki etmeden uygulanabilir. Aynı yöne dönen her pervane pitch veya roll ekseninden birini kontrol eder ve bir motorun itme kuvvetini arttırırken diğerini azaltmak yaw stabilitesi için gerekli olan tork dengesini koruyarak roll veya pitch eksenine tork uygulanmasını sağlar. Böylelikle, sabit pervaneler quadrotorların her eksene manevrasını sağlar. Öteleme ivmesi ise sıfır olmayan bir pitch veya roll açısıyla sağlanır. Üç, altı veya başka bir sayıda motor yerine dört motor kullanılmasının sebebi dört motorun iki uygun dönöl simetriye sahip olmasıdır. Dört motorla itmeyi yanlara verip roll hareketi yapmak kolaydır. İki aynı dönen motorun birinin hızının artıp diğerinin azalmasıyla toplam tork ve yaw'a uygulanan güç sıfır olarak kalır. Klasik bir helikopteri kontrol ederken ana motordan gelen torku dengeleyip yaw kontrolü uygulamak zor bir işlemdir ve hatırı sayılır ölçüde pratik gerektirir. Quadcopter'in ise yaw kontrolü doğuştan dengededir. Ana kontroller değişse de öğrenmesi daha kolaydır. Günümüzde kaliteli quadcopterler kendinden gyroscope ile olup yaw'ı daha eksiksiz stabilize eder.

Quadcopterlerin motorları “elmas” veya “kare” şeklinde dizilebilir.



Şekil 1.5

Torku dengelemek için 1 ve 3 numaralı motorlar saat yönüne dönerken 2 ve 4 numaralı motorlar saat yönünün tersine dönüyor.



Şekil 1.6

Şekil 1.7

Şekil 1.8

1.6-Quadcopter her motora eşit itme uygulayarak havada sabit durur.

1.7-Aynı yönde dönen motorlara daha çok itme uygulayarak yaw'unu ayarlar.

1.8-Pitch veya rollunu tek motora uygulanan itmeyi yükselterek karşısındaki motorun itmesini düşürerek sağlar.

Yapım için gerekli olan mekanik bileşenler şase, pervane ve motorlardır. En yüksek performans ve basit kontrol algoritmaları için motorlar ve pervaneler eşit uzaklıkta yerleştirilmelidir. Günümüzde karbonfiber yapılar hafif yapısı ve yüksek yapısal sertliği sayesinde popülerleşmiştir.

Çalışan bir quadcopter için gerekli elektronik içerik ise RC helikopterler gibi ESC modülü, kontrol kartı ve pildir. Genel olarak tipik bir hobi uzaktan kumandası kullanılır.

1.2 QUADCOTER PARÇALARI

Şekil 1.9 Frame Kitler



Şekil 1.10 4 adet motor



Şekil 1.11 HP 30A ESC Brushless



Şekil 1.12 4 adet pervane



2.1 RASPBERRY PI

Raspberry Pi Özellikleri: [2]

	NEW! Raspberry Pi 2 Model B 1GB	Model B+ Learn More
Processor Chipset	Broadcom BCM2836 ARMv7 Quad Core Processor powered Single Board Computer running at 900 MHz	Broadcom BCM2835 ARMv6 SoC full HD multimedia applications processor
RAM	1GB SDRAM @ 450 MHz	512 MB SDRAM @ 400 MHz
Storage	MicroSD	MicroSD
USB 2.0	4x USB Ports	4x USB Ports
Power Draw / voltage	1.8A @ 5V	1.8A @ 5V
GPIO	40 pin	40 pin
Ethernet Port	Yes	Yes

8

Resmi Dağıtımlar

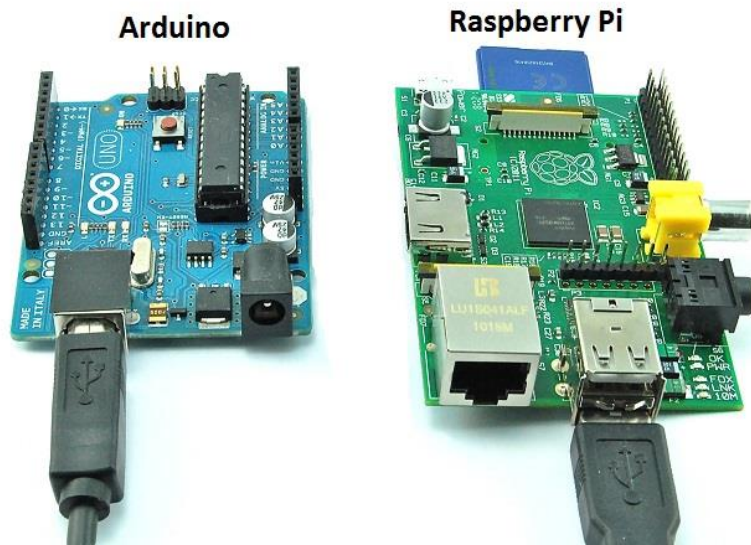
NOOBS(New Out Of the Box): Raspberry Pi foundation tarafından kurulum kolaylaşması amacı ile hazırlanan bir dağıtım çeşididir. Bu dağıtımın bir karta birden fazla işletim sistemi kurabilme, açılış yöneticisi ayarlayabilme gibi çeşitli artıları vardır. Bu dağıtım, yeni kullanıcılara başlangıç için tüm işletim sistemi alternatiflerini görmeleri için güzel bir ortam sağlar.

Raspbian: En popüler Raspberry Pi dağıtımdır. Debian ARM 'den Raspberry Pi için özel olarak modifiye edilmiş bu dağıtımı Raspberry Pi'yi genel bir bilgisayar olarak, veya eğitim aracı olarak kullanacak kullanıcılar için önerilen bir dağıtımdır.

Arduino İle Raspberry Pi'nin Farkı [4]

- Arduino, 8-bit'lik Atmega Mikrokontrolcü'ye sahipken, Raspberry Pi'de 32-bit ARM tabanlı Mikroişlemci bulunmaktadır,
- Arduino 8-16 Mhz saat hızına sahipken, Raspberry Pi de bu 1 Ghz'e kadar çıkmaktadır,
- Arduino 2-8 KB, Raspberry Pi ise 512 MB Ram kapasitesine sahiptir,
- Raspberry Pi de GPU (Grafik işlemci), Ses, USB ve Ethernet çıkışı bulunurken Arduino'da bunlar bulunmamaktadır.
- Arduino, Raspberry Pi'ye göre çok daha kolay programlama imkanına sahiptir. Raspberry Pi'yi iyi bir şekilde kullanabilmek için Linux komutlarına hakim olmak gerekmektedir.

Şekil 2.4



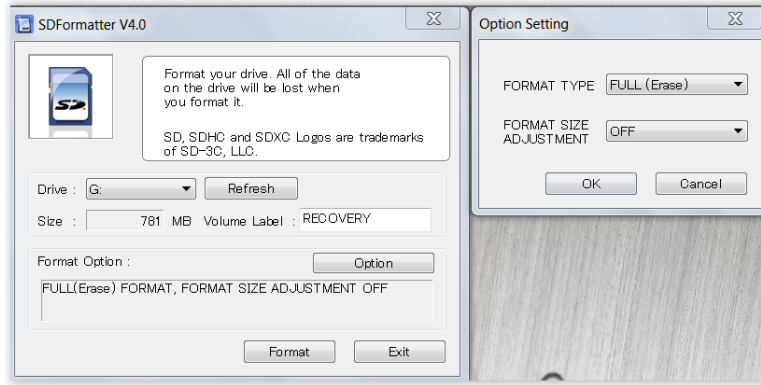
2.2 RASPBIAN KURULUMU

Raspberry Pi için gerekli malzemeler:

Raspberry Pi Board, SD Kart (Minimum 8 GB), SD Kart Okuyucu (Realtek Card Reader), 5V-2A'lık Raspberry Pi Adaptörü, Klavye ve Mouse, HDMI kablosu, HDMI destekli ekran

İlk önce NOOBS programı indirilir. İndirme tamamladıktan sonra Minimum 8 GB'lık SD kart için SD Formatter programı kurularak önce kart formatlanır, sonra da NOOBS paketi SD kartına yapıştırılır. [5]

Şekil 2.5



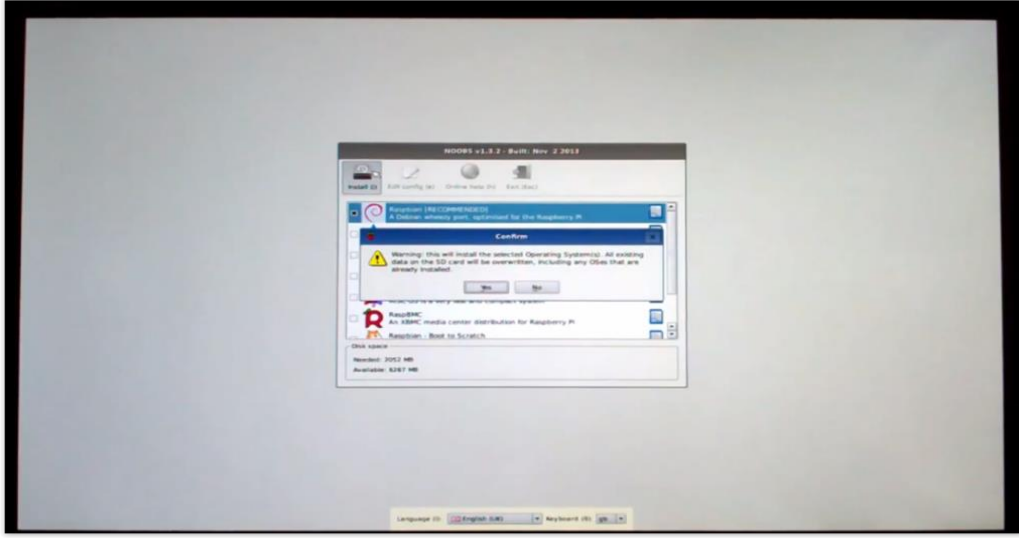
SD kartı formatlamak için FORMAT TYPE: FULL (Erase) olmalı ve FORMAT SIZE kısmı ise ON olmalıdır. SD kartı işletim sistemi hazır bir şekilde Raspberry Pi 'ye takılır.

Şekil 2.6



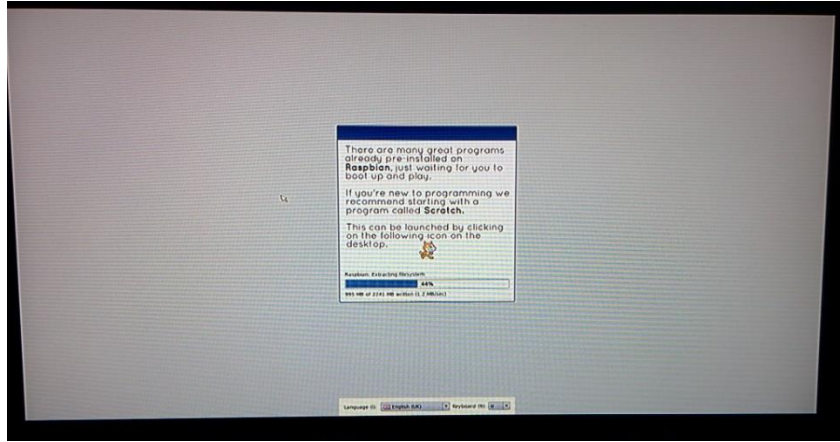
SD kartı taktıktan sonra adaptöre enerji verildiğinde ekranda büyük kırmızı ahududu resmi görülür. Bir süre sonra ise başlatım menüsü gelir.

Şekil 2.7



Install kısmına Yes dedikten sonra kurulum başlatılır ve şu şekilde aşamalar bekletilir.

Şekil 2.7



Şekil 2.8



Kurulum tamamlanınca ise şu şekilde tamamlandığına dair mesaj gelir.

Şekil 2.9



2.3 RASPBERRY PI GÜNCELLEŞTİRME

Raspberry Pi’de örneğin GPIO kullanmak için Raspbian dağıtımı önerilir, çünkü Raspbian dağıtımı GPIO modüllerini destekler. Verimli bir kullanım ve kütüphane kurulumları için Raspbian dağıtımında olduğundan emin olmak gerekir.

İlk olarak tüm paketleri ve paket listesi güncellenir: [6]

```
sudo apt-get update  
sudo apt-get upgrade  
sudo apt-get dist-upgrade
```

2.4 I2C KURULUMU

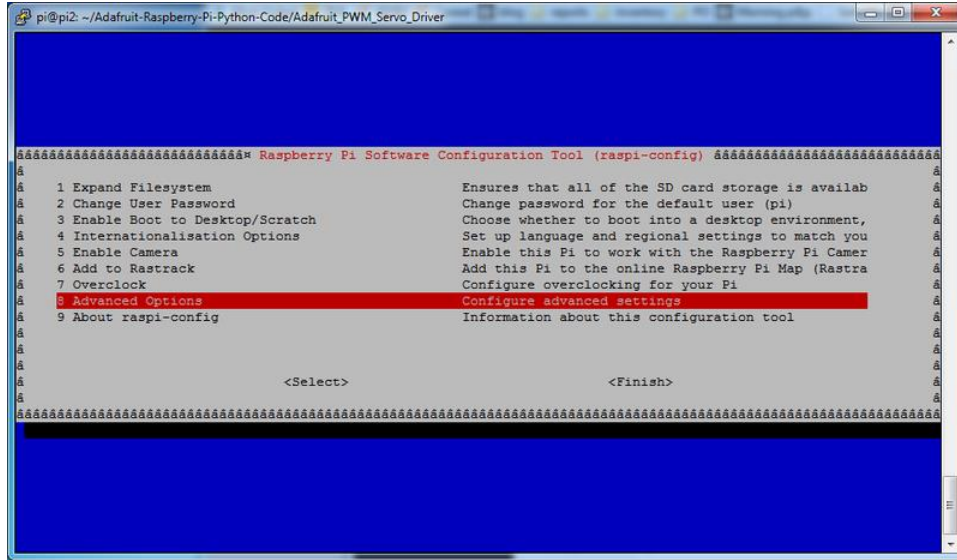
I2C, bir çip ile diğerleriyle konuşmayı sağlayan çok kullanışlı standarttır. Birden fazla Raspberry Pi’nin bağlanmasına izin verir ve benzersiz adresin her birini sık sık jumper modül ayarlarını değiştirerek ayarlayabilir. I2C araçlarını terminale yüklemek için önce aşağıdaki komutlar yazılır:

```
sudo apt-get install i2c-tools libi2c-dev  
sudo apt-get install python-smbus
```

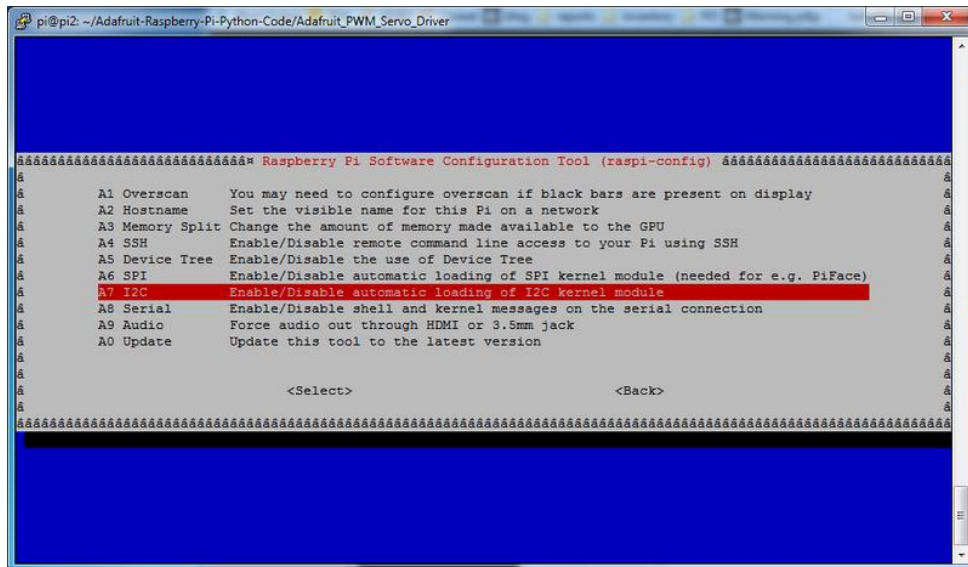
[7]

sudo raspi-config ve ARM çekirdeği ile Linux çekirdeği desteği için i2c desteğini yüklemek için şu yönergeler izlenir: [7]

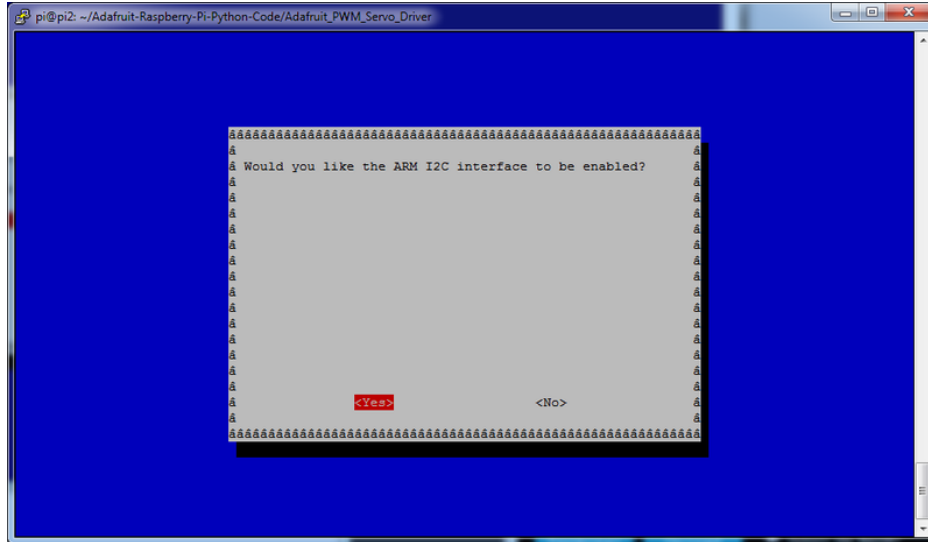
Şekil 2.10



Şekil 2.11



Şekil 2.12



Şekil 2.13



Bu işlemlerde sonra cihaz şu komutla yeniden başlatılır:

```
sudo reboot
```

veya

```
sudo shutdown -r now
```

Raspbian için SSH açmak için aşağıdaki komut yazılır:

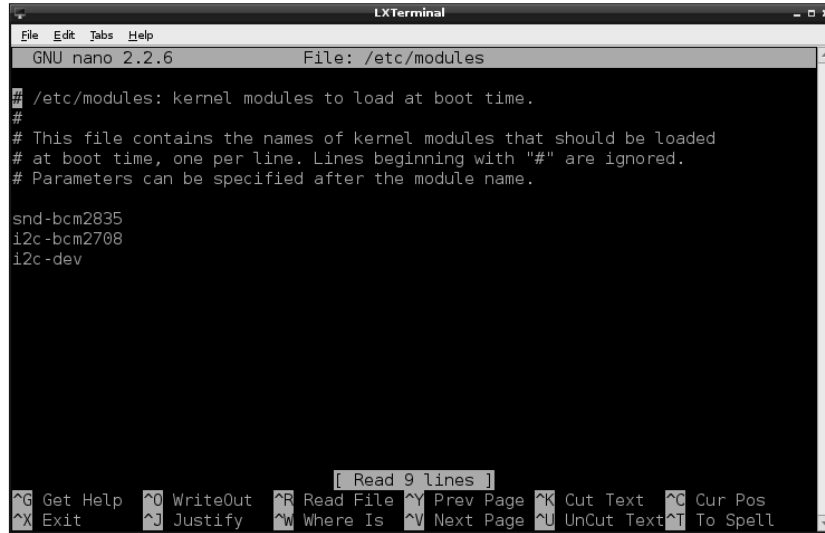
```
sudo nano/etc / modules
```

ve dosya sonuna aşağıdaki 2 satır eklenir:

```
i2c - bcm2708
```

```
i2c -dev
```

Şekil 2.14



```
GNU nano 2.2.6 File: /etc/modules

# /etc/modules: kernel modules to load at boot time.
#
# This file contains the names of kernel modules that should be loaded
# at boot time, one per line. Lines beginning with "#" are ignored.
# Parameters can be specified after the module name.

snd-bcm2835
i2c-bcm2708
i2c-dev
```

Değişimler CTRL + X yapıldıktan sonra Y yazıp Enter tuşuna basılarak kaydedilir.

Daha sonra aşağıdaki komutlar ile kara listeler belirlenir ve listeye alınacakların başına # işareti konur.

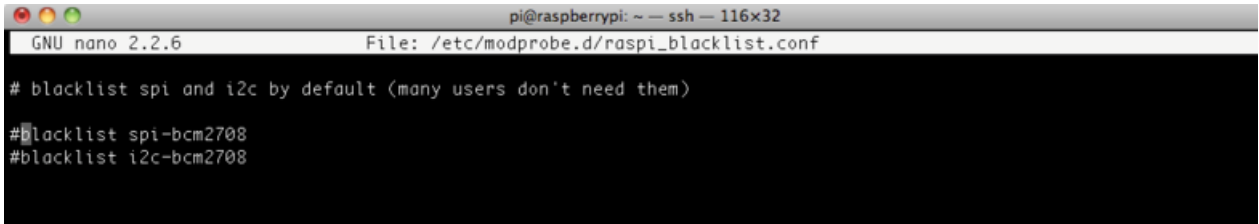
```
sudo nano/etc/modprobe.d/raspi-blacklist.conf
```

Başına # işareti konulacak komutlar :

```
blacklist spi-bcm2708
```

```
blacklist i2c-bcm2708
```

Şekil 2.15



```
pi@raspberrypi: ~ -- ssh -- 116x32
GNU nano 2.2.6 File: /etc/modprobe.d/raspi_blacklist.conf

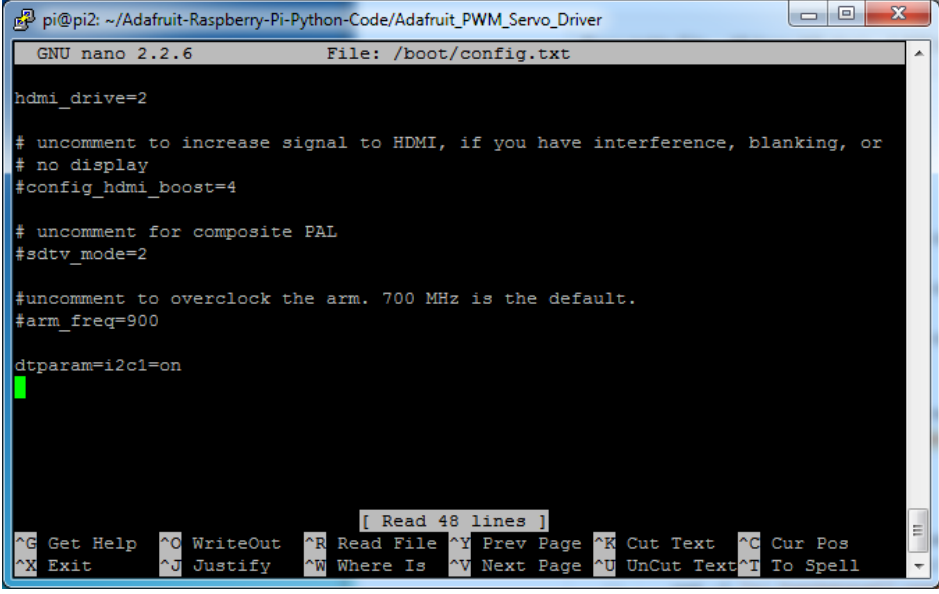
# blacklist spi and i2c by default (many users don't need them)
#blacklist spi-bcm2708
#blacklist i2c-bcm2708
```

komutu ile Raspberry Pi boot /config.txt dosyası güncellenir ve aşağıdaki satır bu dosyaya eklenir:

```
Sudo nano/boot/config.txt
```

```
dtoverlay=i2c-rtc,ds1307
```

Şekil 2.16



```
pi@pi2: ~/Adafruit-Raspberry-Pi-Python-Code/Adafruit_PWM_Servo_Driver
GNU nano 2.2.6 File: /boot/config.txt

hdmi_drive=2

# uncomment to increase signal to HDMI, if you have interference, blanking, or
# no display
#config_hdmi_boost=4

# uncomment for composite PAL
#sdtv_mode=2

#uncomment to overclock the arm. 700 MHz is the default.
#arm_freq=900

dtparam=i2c1=on

[ Read 48 lines ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

Daha sonra `sudo reboot`

veya `sudo shutdown -r now`

ile tekrar yeniden başlatılır.

Daha sonra ise I2C testi yapılarak cihazın bağlı olup olmadığı algılanır ve bu da aşağıdaki komutlar ile sağlanır:

Revizyon 1 cihazlar için (256 MB RAM’li model için)

```
sudo i2cdetect -y 0
```

Revizyon 2 cihazlar için (512 MB RAM’li model için)

```
sudo i2cdetect -y 1
```

Şekil 2.17



```
root@raspberrypi:~# sudo i2cdetect -y 1
    0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40: 40  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70: 70  --  --  --  --  --  --  --  --  --  --  --  --  --  --
root@raspberrypi:~#
```

Bu şekilde bir çıktı alınıyorsa cihaz çalışıyordur.

2.5 WIRINGPI KÜTÜPHANESİNİ GÜNCELLEŞTİRME

Raspberry Pi'nin beyni ve omurgası sayılan BCM2835'in P1 portu 26 pine sahip P1, BCM2835'in bazı pinlerine erişilmesini sağlayan bir porttur. 26 pinin 8 tanesi Genel Amaçlı Giriş Çıkış (GPIO) için programlanırken biri donanımsal PWM çıkışı olmak üzere I2C, SPI ve UART gibi birçok porta sahiptir. Ayrıca 3.3V ve 5V'luk beslemeleri vardır.

Raspberry Pi'nin B modeli ayrıca bir konnektör lehimleyebileceğimiz 4 pinli ayrı bir GPIO portuna da sahiptir. Bu portun adı da P5'dir. P5'i kullanarak 2mm'lik 2*4 erkek header konnektör bağlayarak 4 GPIO pini daha elde edilebilir.

WiringPi, birçok dolaylı GPIO kontrol projesine de temel teşkil etmiş durumdadır. Bu özelliği C'de kodlanmış olmasına bağlıdır.

WiringPi GPIO Komut Satırı Aracı

WiringPi'nin asıl bahsedilmek istenilen kısmı beraberinde gelen GPIO adındaki komut satırı aracıdır. Bu komut ile Raspberry Pi'nin portlarını komut satırından kolayca kontrol edilebilir.

WiringPi kütüphanesi kurulumunun en sağlıklı yolu Git üzerinde indirip kurmaktır. Aşağıdaki komut ile kütüphane Raspberry Pi'ye indirilir:

```
git clone git:// git.drogon.net/wiringPi
```

Ardından inen klasöre gidip root hakları ile derleme ve kurulum başlatılır.

```
cd wiringPi  
sudo ./ build
```

En son olarak da

```
gpio -v
```

komutu ile kodun board'da düzgün okunup okunmadığı görülür.

Terminal üzerinden kullanım için aşağıdaki komut yazılır:

```
gpio readall
```

Komut yazıldıktan sonra oluşan çıktı şu şekildedir: [8]

Şekil 2.18

```
pi@raspberrypi ~$ cd wiringPi & gpio readall
```

BCM	wPi	Name	Mode	V	Physical	V	Mode	Name	wPi	BCM
		3.3v			1	2		5v		
2	8	SDA.1	IN	1	3	4		5v		
3	9	SCL.1	IN	1	5	6		0v		
4	7	GPIO. 7	IN	0	7	8	1	ALT0	TxD	15
		0v			9	10	1	ALT0	RxD	16
17	0	GPIO. 0	IN	0	11	12	0	IN	GPIO. 1	1
27	2	GPIO. 2	IN	0	13	14		0v		18
22	3	GPIO. 3	IN	0	15	16	0	IN	GPIO. 4	4
		3.3v			17	18	0	IN	GPIO. 5	5
10	12	MOSI	IN	0	19	20		0v		23
9	13	MISO	IN	0	21	22	0	IN	GPIO. 6	6
11	14	SCLK	IN	0	23	24	0	IN	CE0	10
		0v			25	26	0	IN	CE1	11
0	30	SDA.0	IN	0	27	28	0	IN	SCL.0	31
5	21	GPIO.21	IN	0	29	30		0v		1
6	22	GPIO.22	IN	0	31	32	0	IN	GPIO.26	26
13	23	GPIO.23	IN	0	33	34		0v		12
19	24	GPIO.24	IN	0	35	36	0	IN	GPIO.27	27
26	25	GPIO.25	IN	0	37	38	0	IN	GPIO.28	28
		0v			39	40	0	IN	GPIO.29	29
										21

Mesela 7. pin modülünü aktif edelim. Aktif etme şu şekilde olacaktır:

```
gpio mode 7 out
```

Aşağıdaki komut ile lamba söndürülür:

```
gpio write 7 1
```

Bu komut ile 7 numaralı wiringPi pinine 1 değeri yazılır. Aynı şekilde geri yakmak için de şu şekilde komut yazılır:

```
gpio write 7 0
```

3. ANDROID UYGULAMA VE HABERLEŞME

3.1 QUADCOPTER KONTROLÜ İÇİN ANDROID UYGULAMASI

Quadcopter'in güç ve yön kontrolü için bir kumanda uygulaması geliştirildi. Yapılan uygulamanın görsel kısmında Linear Layout, SeekBar ve TextView araçları kullanıldı.

Uygulamanın güç kontrolü için SeekBar aracı kullanıldı. Bu araç ile bir işlem çubuğuna kaydırılabilir bir nesne eklendi ve kullanıcının dokunup tek bir ekseninde sürüklemesiyle seviye değiştirildi.

Yön kontrolü için genellikle mobil oyunlarda kullanılan kumanda kolu(joystick) oluşturuldu. Kullanıcı bir daire şeklinde olan kumanda düğmesine dokunuyor ve dokunduğu anda bu düğmeden daha küçük ikinci bir düğme açılıyor. İkinci düğme birinci düğmenin içerisinde 2 ekseninde hareket edebiliyor ve bu düğmenin X-Y değerleri ölçülüyor.

Bu uygulama ile güç değerini ve quadcopterin yönünü kontrol etmekte kullanılacak olan x, y değerlerini kullanıcının kontrol etmesi sağlanmaktadır. [9]

3.2 RASPBERRY PI İLE TELEFON HABERLEŞMESİ

Yapılan uygulama ile Raspberry Pi'nin haberleşebilmesi için Transmission Control Protocol (TCP) kullanıldı.

TCP, veri güvenliğinin sağlandığı ve bağlantı temelli (connection-oriented) bir iletişim protokölüdür. TCP protokolü *stream* soketleri üzerinden sağlanmaktadır. Haberleşecek taraflardan bir tanesi gelecek istekleri dinlemekte diğer taraf ise istekte bulunmaktadır.

Dinleyen taraf sunucu (server), istekte bulunan taraf ise istemci (client) olarak isimlendirilmektedir. Projenin sunucu-istemci modeline uyarlanması, sunucu görevini Android uygulaması istemci görevini ise Raspberry Pi üstlenmektedir. [10]

Sunucu tarafında gerçekleşen olaylar kısaca şu şekilde:

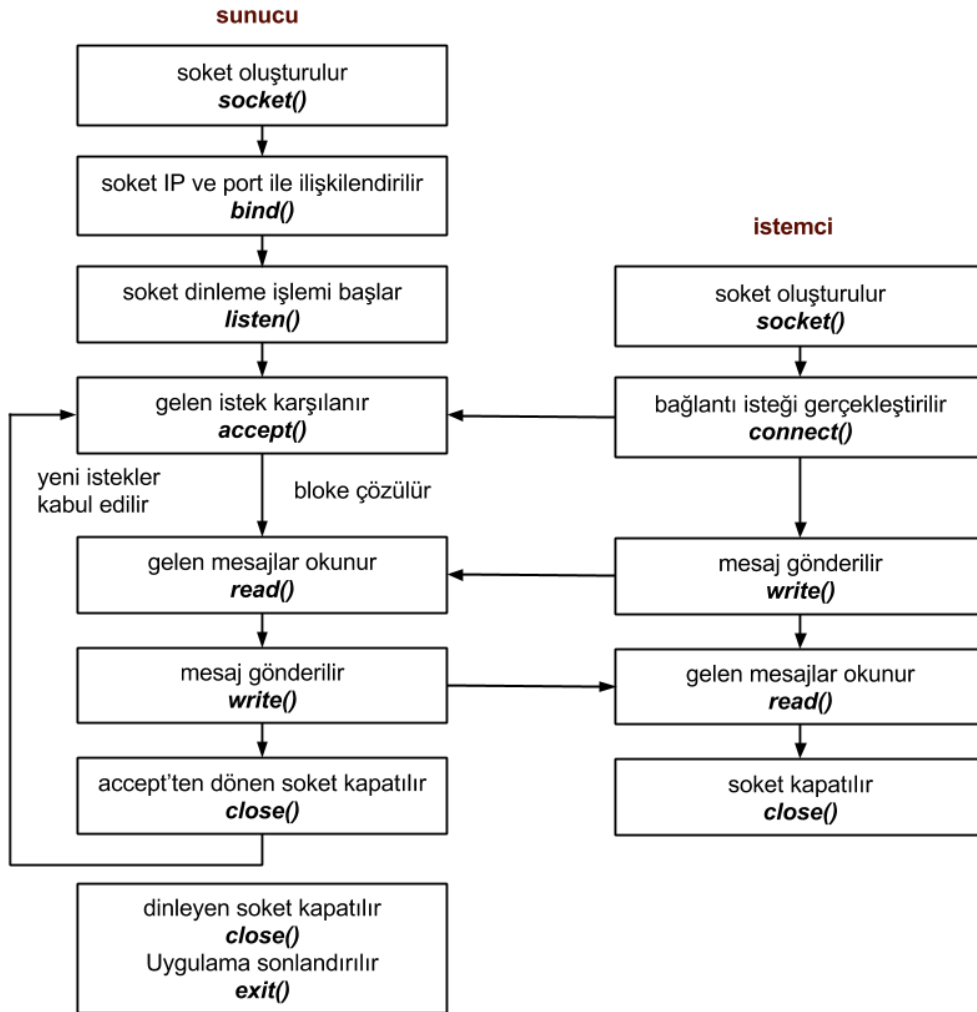
1. *socket* ile haberleşme alt yapısı oluşturulur.
2. *bind* ile soket bir IP adresi ve port ile ilişkilendirilir.
- 3 *listen* ile soket üzerinde dinleme işlemi yapılacağı belirtilir.
4. *accept* ile gelen bağlantı isteği kabul edilir.
5. *read* ve *write* fonksiyonlarıyla çift yönlü okuma ve yazma işlemleri yapılır. Bu durum karşılıklı konuşmaya denk gelmektedir. Son olarak *close* ile bağlantı sonlandırılır.

İstekte bulunan tarafta ise geçen olaylar kısaca şöyledir:

Sunucu tarafında olduğu gibi ilk olarak socket ile haberleşme alt yapısı oluşturulur.

1. connect ile karşı tarafa bağlanma isteği gönderilir.
2. read ve write fonksiyonlarıyla çift yönlü okuma ve yazma işlemleri yapılır.
3. close ile bağlantı sonlandırılır. [11]

Şekil 3.1

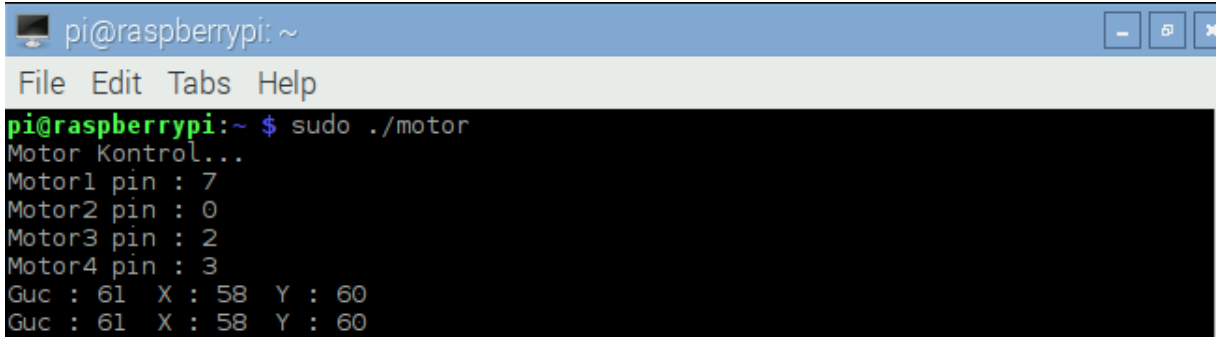


Sunucunun yaşam döngüsü boyunca birden çok istemciye yanıt verebilmek için bir döngü içerisinde yeniden `accept` fonksiyonunu çağırdığını görmekteyiz. Bu örnekte bizim uygulamamıza karşı düşer yani sunucu tek bir istemciye hizmet verebilmektedir.

Şekil 3.2



Şekil 3.3



4. QUADCOPTER HAREKETİ

4.1 JİROSKOP

Jiroskop, ([İngilizce](#): Gyroscope, Gyro) veya **Cayroskop**, **Cayro**, yön ölçümü veya ayarlamasında kullanılan, açısal dengenin korunması ilkesiyle çalışan bir alet. Jiroskopik hareketin temeli [fizik](#) kurallarına ve [açısal momentumun](#) korunumu ilkesine dayalıdır. [12]

Jiroskop olarak bilinen alet ilk olarak 1817’de [J. Bohnenberger](#) tarafından icat edilmiştir ve jiroskop adı [1852](#)’de Dünya’nın dönüş hareketini incelemek üzere yaptığı deneyler sırasında [J. Foucault](#) tarafından verilmiştir. Bir jiroskop [presesyon](#) ve [nutasyon](#) olarak bilinen hareketleri de içine alan çeşitli hareketler yapar. Günlük hayatta, [uçak](#) ve [gemilerde](#) yön bulmak için, uzay [teleskoplarında](#) yörünge kararlılığını sağlayabilmek için yaygın olarak jiroskoplardan yararlanılmaktadır. [Bisiklete](#) binen herkes, bir bisiklet hızlı gittiği vakit dengeyi sağlamanın, yavaş gittiği vaktinkine göre çok daha kolay olduğunu bilir. Bir [topaç](#), dönme hızı büyükse, dik kalarak dönmeye devam eder, fakat yavaşladıkça yana yatmaya başlar ve sonunda devrilir. Bu örneklerin her ikisinde de, kararsız olan (yani kolayca düşebilecek olan) cisimler, yeterli hızla hareket halinde oldukları vakit dik durabilmektedir.

Bunlarda gördüğümüz, bir defa bir düzlemde dönmeye başlatılan bir cismin o düzlemde dönmeye devam etmesi özelliğinden jiroskoplarda ve denizcilik ile havacılıkta kullanılan başka çeşitli seyir yardımcılarında faydalanılır. Bu özellik, ağırlığının büyük bir kısmı çevresine yakın toplanmış bulunan tekerleklerde daha açıktır. Bu cins ağır tekerleklerin hepsine jiroskop denir.

4.2 I2C HABERLEŞME PROTOKOLÜ

Dijital sistemlerde kablolu seri haberleşme ile ilgili birçok standart vardır. SPI, I²C bunlara örnek olarak verilebilir. Bu standartların kullandıkları uç sayısı, ulaşabilecekleri maksimum hızlar birbirinden farklı olmakla beraber I²C protokolü oldukça hızlı veri aktarımına olanak tanımaktadır. Bir arada çalışan, belirli aralıklarla birbiriyle haberleşen, yavaş çeşitli çevresel cihazların minimum harici donanım gereksinimiyle haberleşmelerini sağlar. Basit, düşük bant genişliğine sahip, kısa-mesafe protokolüdür. Mevcut I2C cihazlarının çoğu 400kbps'ye kadar hızlarda çalışmaktadır. I2C ile birden fazla cihazı haberleştirmek kolaydır çünkü içerisinde adresleme planı da bulunmaktadır. uçları bulunabilir. Ancak bu kontroller bu protokol için bir standart olmadığından kullanıcı bunları kendisi düzenlemek zorundadır. [13]

I²C protokolünde temel olarak iki hat vardır. Bunlar SCL ve SDA olup, SCL (SerialCLock) veri senkronizasyonu için kullanılan clock darbeleri hattı, SDA (SerialData) ise veri hattıdır. Ayrıca elemanların kendine has yazma koruma(WP-Write Protect), çıkış izin(OE-Output Enable) gibi I²C protokolünde veri alışverişi Start(başla) komutuyla başlar ve Stop(bitir) komutuyla biter. Bu protokolda bilgi veya komut taşıyan veri alışverişi saat darbesinin lojik düşük(low) seviyelerinde gerçekleşir. Cihazlar arasında haberleşmenin başladığını veya tamamlandığını belirten Start(başla) ve Stop(bitir) durum komutları ise sadece SCL hattının yani saat sinyalinin lojik yüksek(high) olduğu durumlarda gönderilir. SCL hattı lojik yüksek iken SDA hattında lojik1'den lojik0'a geçiş start komutu anlamına gelir ve veri transferinin başlayacağını bildirir. Benzer şekilde SCL hattı lojik yüksek iken SDA hattında lojik0'dan lojik1'e geçiş stop(bitir) komutu anlamına gelir ve haberleşen iki cihaz arasındaki haberleşmenin tamamlandığını bildirir.

I²C protokolünde gönderilen veri ve okunan veri aynı hat üzerinden, yani SDA hattından transfer edilir. SDA pini sürekli olarak bir pull-up direnciyle V_{cc} 'ye (TTL için +5V besleme gerilimi) bağlı olduğundan hattaki start ve stop bitlerinin anlaşılması hatta lojik 0 olup olmamasıyla anlaşılır. Ayrıca hattın sürekli yüksek(high) lojik olarak kurulması bazı elemanların yüksek empedans konumlarında bile hattan lojik 1 okunmasını sağlar.

I²C protokolünde 'Master' denilen ana bir kontrol birimi cihazı -ki genelde mikrodenetleyiciler kullanılır- ve 'Slave' denilen cihazlar vardır. Bu protokole göre, I²C veri yoluna bağlı her bir slave durumundaki cihazın, en fazla 7 bittten oluşan, kendisine ait bir adresi vardır. Bu cihaz bir LCD, EEPROM, mikrodenetleyici veya herhangi bir sayısal sistem olabilir. Veri yolundaki haberleşme iki yönlü de olabilir. SDA hattında eğer bu adres bilgisi master tarafından gönderilmişse, o adrese ait slave tarafından SDA hattına ACK yani 'alındı' anlamına gelen bir cevap gönderilir. Bu aşamadan sonra slave birimi kullanılmaya hazırdır.

I²C veriyolu çok master'lı (multi-master) veri yoludur. Yani, birden fazla sayıda data transferini başlatabilme özelliğindeki tümdevre, veri yoluna bağlanabilir. Bu durumda data ("veri") transferini başlatan tümdevre(IC-Integrated Circuit) Bus-Master(Veriyolu Efendisi) durumunda olacak, veri yolundaki tüm diğer IC'ler ise Bus-Slave(Veriyolu Köleleri) durumunda olacaktır. Master durumundaki eleman genelde bir mikrodenetleyici olduğundan, bir I²C haberleşmesinin nasıl çalıştığını şu şekilde özetleyebiliriz:

1. Öncelikle mikrodenetleyici SDA hattından "Start" biti gönderir. Bu durumda tüm slave IC'ler kendilerini alınacak adres bilgisi için hazırlarlar.

2. Mikrodenetleyici, haberleşmek istediği slave IC'nin adresini ve yapılacak işlemin yazma veya okuma olup olmadığını 1 bayt halinde SDA hattından I2C veri yoluna aktarır. Slave IC'ler SDA hattından gelen adres bilgisini alırlar ve bunun kendi adresleri olup olmadığına karar verirler.
3. Masterın gönderdiği adrese sahip IC kendisinin veri yolunda olduğunu ve haberleşmeye hazır olduğunu bildirmek üzere SDA yoluna bir ACK sinyali gönderir. Diğer IC'ler mikrodenetleyiciden gelecek "Stop" bitini beklemeye koyulurlar ki, bu haberleşme tamamlanana kadar müdahalede bulunmasınlar ve bir sonraki haberleşmede kendileri çağrılabilirler.
4. Veri transferi gerçekleşir. İşlem yazma ise master'dan slave'e doğru, okuma ise tersi yönde veri transferi gerçekleşir. Transfer tamamlandıktan sonra Master(yani örnekte mikrodenetleyici) Stop bitini SDA hattından aktarır ve haberleşme sona erer.

4.3 ESC İLE MOTOR KONTROLÜ

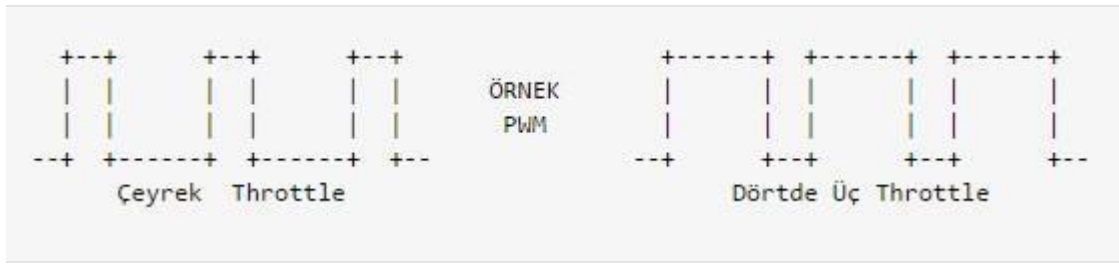
Şekil 4.1



ESC'ler motorun dönüş hızını elektronik olarak kontrol eder. Elektrik motorlarını anlatırken de DC motorların voltaj-devir ilişkisinden bahsedilir. İşte ESC'ler motora giden elektrik basıncını, yani **voltajı** düşürerek motorun hangi devirde döneceğini ayarlar. Ancak motora giden voltajı lineer regülasyon ile tamamen orantılı bir biçimde düşürmenin büyük zorlukları ve dezavantajları vardır. Lineer regülatörler elektrik akımının fazla voltajını bir geçiş transistörü üzerinde ısı enerjisine çevirir. Bu nedenle de verimlilikleri sadece %30-40 civarındadır. Dahası ortaya büyük miktarda ısı çıkar. Bu yüzden bugün kullanılan modern ESC'ler, motora giden gücü **Pulse Width Modulation (PWM)** metodu ile düşürür. Çok basitçe ESC, giriş akımını hızla "kapatıp açar". Akımın, birim sürenin kaçta kaçında açık, kaçta kaçında kapalı olduğuna bağlı olarak motora ulaşan ortalama voltaj belli bir değere inecektir. [14]

Bir saniyelik süreyi dikkate alarak bir örnekle verilirse; Varsayalım ki 10 voltluk bir batarya kullanılıyor ve çeyrek gazla uçuruluyor. Bu durumda ESC akımı saniyenin onda 2'sinde açacak, onda 8'sinde kapatacaktır. Motora ulaşan voltaj ortalaması da 2V olacaktır. Aslında devre açıkken motora 10V gelmektedir ancak arada devre kapandığından ve bu işlem çok hızlı olduğundan motor ne 10V hızına çıkmaya, ne de durmaya vakit bulabilir, ortalama değer olan 2V hızında döner.

Şekil 4.2



Modern yüksek frekanslı ESC'ler, bu akımı kesip verme işini saniyede binlerce defa yapar, bu sayede de son derece yumuşak bir motor devir kontrolü elde edilir. Eğer ESC varsa muhtemelen bir yerlerinde KHz cinsinden bir rakam görülür. İşte bu rakam, ESC'nin hangi frekansta, yani hangi hızla motora giden gücü kesip verebildiğini gösterir.

4.3 RASPBERRY PI İLE GPIO PROGRAMLAMA

Şekil 4.3

```
#include <wiringPi.h>
#include <stdlib.h>
#include <stdio.h>

int pinOut = 7;
int pinInput = 11;

int main(void){
    printf("Gpio Programlama...\n");

    wiringPiSetup(); //Modül aktif edildi.
    pinMode(pinOut,OUTPUT); //pinOut çıkış olarak belirlendi.
    pinMode(pinInput,INPUT); //pinInput giriş olarak belirlendi.

    while(True)
    {
        if(digitalRead(pinInput) == 0) //Giriş pinine giriş uygulanmış ise.
        {
            digitalWrite(pinOut,HIGH); //Çıkış pinini YÜKSEK seviye yap.
        }
        delay(10); //10 ms'ye bekle;
    }
    return 0;
}
```

5.SONUÇLAR

Proje genel olarak Quadcopterin dengeli bir biçimde hareket etmesi, Android uygulamasından ve Jiroskoptan gelecek verilerin koordineli bir biçimde Raspberry Pi'ye aktarılarak motor sürücüye bu verilere göre gerekli frekansları göndermekle alakalıdır. Proje geliştirilirken jiroskoptan gelen verilerin filtrelenmesi ve seçilen motorlar fazla akıma gerek duyduğundan güç sorunu yaşandı. Sorunun çözülmesi için iki motor bir güç kaynağı ile diğer iki motorda başka bir güç kaynağı ile beslendi. Proje geliştirilirken Raspberry Pi kullanımı ve genel ayarlamaların yapılması, Raspberry Pi 'de GPIO arayüzü kullanımı, Linux ortamında yazılım geliştirme, Android Programlama, Client-Server Modeli, I2C Protokolü gibi konular incelendi.

6. KAYNAKLAR

Raspberry PI DIKEY EKSEN YAYINLARI (ARDA KILIÇDAĞI)

- [1]<http://www.multicopterturkiye.com/>
- [2]<http://www.samm.com/>
- [3] <http://www.raspi.gen.tr/2012/08/01/raspberry-pi-nedir/>
- [4]<http://www.elektrikport.com/teknik-kutuphane/raspberry-pi-nedir-arduino-ile-farklari-nelerdir-/8305#ad-image-0>
- [5]<http://www.utkuolcar.com/>
- [6] <https://www.raspberrypi.org/documentation/raspbian/updating.md>
- [7]<https://learn.adafruit.com/adafruits-raspberry-pi-lesson-4-gpio-setup/configuring-i2c>
- [8]<http://blog.calik.me/raspberry-pi/raspberry-pi-wiringpi-kurulumu.html>
- [9]<http://www.akexorcist.com/2012/10/android-code-joystick-controller.html>
- [10]<http://androidsrc.net/android-client-server-using-sockets-server-implementation/>
- [11]<https://demirten.gitbooks.io/linux-sistem-programlama/content/sockets/tcp.html>
- [12] <https://tr.wikipedia.org/wiki/Jiroskop>
- [13] <http://ramazansural.blogspot.com.tr/2010/11/i2c-haberlesme-protokolu.html>
- [14] <http://www.freepist.com/esc-electronic-speed-controller>