T.C.
İSTANBUL AYDIN ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

RF İLETİŞİMLİ MOBİL ROBOT KOLU

YÜKSEL LİSANS TEZİ

Gökhan EŞGİ

Mekatronik Mühendisliği Ana Bilim Dalı

Mekatronik Mühendisliği Programı

EYLÜL-2016

T.C.
İSTANBUL AYDIN ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

RF İLETİŞİMLİ MOBİL ROBOT KOLU

YÜKSEL LİSANS TEZİ

Gökhan EŞGİ

Y1313.110001

Mekatronik Mühendisliği Ana Bilim Dalı
Mekatronik Mühendisliği Programı

Tez Danışmanı: Prof. Dr. Osman Nuri UÇAN

EYLÜL 2016

T.C.
İSTANBUL AYDIN ÜNİVERSİTESİ
FEN BİLİMLER ENSTİTÜSÜ MÜDÜRLÜĞÜ

Yüksek Lisans Tez Onay Belgesi

Enstitümüz Mekatronik Mühendisliği Ana Bilim Dalı Mekatronik Mühendisliği Tezli Yüksek Lisans Programı **Y1313.110001** numaralı öğrencisi **Gökhan EŞGİ**'nın **"RF İLETİŞİMLİ MOBİL ROBOT KOLU"** adlı tez çalışması Enstitümüz Yönetim Kurulunun 07.09.2016 tarih ve 2016/23 sayılı kararıyla oluşturulan jüri tarafından oy birliği ile Tezli Yüksek Lisans tezi olarak Kabul edilmiştir.

Öğretim Üyesi Adı Soyadı	İmzası
Tez Savunma Tarihi :29/09/2016	
1)Tez Danışmanı: Prof. Dr. Osman Nuri UÇAN	.....
2) Jüri Üyesi : Prof. Dr. M. Emin TACER	..... 29/09/2
3) Jüri Üyesi : Prof. Dr. Sedef KENT	.....

Not: Öğrencinin Tez savunmasında **Başarılı** olması halinde bu form **imzalanacaktır**. Aksi halde geçersizdir.

YEMİN METNİ

Yüksek Lisans Tezi olarak sunduğum " Rf iletişimli mobil robot kolu " adlı çalışmamın, tezin proje safhasından sonuçlanmasına kadar ki bütün süreçlerde bilimsel ahlak ve geleneklere aykırı düşecek bir yardıma başvurulmaksızın yazıldığını ve yararlandığım eserlerin Bibliyografya'da gösterilenlerden oluştuğunu, bunlara atıf yapılarak yararlanılmış olduğunu belirtir ve onurumla beyan ederim.

29/09/2016
Gökhan EŞGİ

ÖNSÖZ

Bu çalışma İstanbul Aydın Üniversitesi Fen Bilimleri Enstitüsü Mekatronik Mühendisliği Bölümünde hazırlanmış olup, rf iletişim ile mobilite kazandırılmış bir robot kol sisteminin geliştirilmesi üzerine yüksek lisans tezi olarak tamamlanmıştır.

Tez çalışmalarım boyunca gösterdiği her türlü destek ve paylaştığı görüşlerinden dolayı çok değerli hocam Prof. Dr. Osman Nuri Uçan'a en içten dileklerimle teşekkür ederim.

Eğitimim boyunca çalışmamı destekleyen İstanbul Aydın Üniversitesi ve Mekatronik Mühendisliği Bölümüne teşekkürü borç bilirim. Çalışmamın tüm ilgililere yararlı olmasını dilerim.

Eylül 2016 Gökhan EŞGİ

İÇİNDEKİLER **Sayfa**

ÖNSÖZ.....v

İÇİNDEKİLER.....vi

ÇİZELGE LİSTESİ.....x

ŞEKİL LİSTESİ.....xi

SİMGELER VE KISALTMALAR.....xiii

ÖZET.....xv

ABSTRACT.....xvi

1.GİRİŞ.....1

2.HABERLEŞME.....2

2.2. Haberleşme sistemi.....2

2.2.1. Verici.....2

2.2.2. İletim Ortamı.....2

2.2.3. Alıcı.....3

2.3. Frekans, Periyot, Dalga Boyu.....3

2.4. Modülasyon.....4

2.4.1. Modülasyon Türleri.....4

2.4.2. Genlik Modülasyonu.....5

2.4.2.1. Çift Yan Bant Genlik Modülasyonu.....5

2.4.2.2. Çift Yan Bant Genlik Modülasyonu Elde Edilmesi.....5

2.4.2.2.1. Bilgi İşareti.....5

2.4.2.2.2. Taşıyıcı işaret.....6

2.4.2.2.3. Modüleli işaret.....6

2.4.2.3. Tek Yan Bant Modülasyon.....6

2.4.3. Frekans Modülasyonu.....6

2.4.3.1.Frekans Modülasyon İhtiyacı.....6

2.4.3.2. Frekans Modülasyonunda Bant Genişliği.....7

1.3.4. PLL Faz Kilitli Döngü Faz Dedektörü.....7

2.5. SAYISAL HABERLEŞME.....8

2.5.1. Temel Kavramlar.....8

2.5.1.1. Bit.....8

2.5.1.2. BPS (Bit Per Second).....8

2.5.1.3. Baud.....8

2.5.1.4. Baud Rate (Oran).....8

2.5.1.5. BER: Bit Error Rate (Bit Hata Oranı).....9

2.5.1.6. Kanal.....9

2.5.1.7. Kanal Kapasitesi.....9

2.5.1.8. Gürültü.....9

2.5.1.8.1. Sistem İçi Gürültü Kaynakları.....9

2.5.1.8.2. Sistem Dışı Gürültü Kaynakları.....9

2.5.2. Örnekleme Teoremi.....9

2.5.3. Kodlama.....10

2.5.3.1. Kodları İletim Formatları.....11

2.5.4. Data Gönderilmesi.....11

2.5.4.1. Asenkron Seri Data Gönderim.....12

2.5.4.2. Senkron Seri Data Gönderim.....14

2.5.5. Darbe Kod Modülasyonu Ve Kodlama Teknikleri.....14

2.5.5.1. Kuantalama İşlemi.....15

2.6. RF Modüller.....15

2.6.1. Tezde kullanılan rf alıcı-verici modülleri.....17

2.6.1.1. ATX-34S UHF ASK verici modülü.....17

2.6.1.2.ARX-34C UHF ASK alıcı modülü.....19

3.MOTORLAR.....21

3.1. Alternatif Akım (AC) Motorları.....21

3.2. Doğru Akım (DC) Motorları.....22

3.2.1. Fırçalı DC Motor.....22

3.2.1.1Endüvi (Rotor).....22

3.2.1.2.Gövde (Stator).....23

3.2.1.3.Fırçalar ve donanımları.....23

3.2.1.4. Motor yan kapakları.....23

3.2.2. Sabit Mıknatıslı Motor.....23

3.2.3. Elektro Mıknatıslı Motor.....23

3.2.4. Fırçasız DC Motor.....23

3.2.5. Adım (Step) Motor.....24

3.3. Servo Motorlar.....24

3.3.1 Servo Motorun Temel Fonksiyonları.....26

3.3.2 Servo Motor Kontrolü.....26

3.3.3. Darbe Genişlik Modülasyonu.....27

3.3.4. Donanım Kontrolörleri.....27

3.4. Tezde Mobilite İçin Kullanılan DC Motorun Kontrolü.....27

3.4.1. L298 entegresi.....27

4. MPU-6050 SENSÖRÜ.....28

4.1.Donanımsal Özellikler……..34

5. LCD (Liquid Crystal Display)……37

5.1 LCD'nin yapısı……..37

6.UYGULAMA……39

7.SONUÇLAR……..41

KAYNAKLAR……...……...43

EKLER……...……...45

ÖZGEŞMİŞ……..75

ÇİZELGE LİSTESİ

Çizelge 2.1: Pin özellikleri.....18

Çizelge 2.2: Pin özellikleri.....19

Çizelge 3.1: Pin özellikleri.....29

Çizelge 5.1: Ayak pinleri.....33

ŞEKİL LİSTESİ

Şekil 2.1 : Verici blok şeması.....2

Şekil 2.2: Genel bir alıcının blok şeması.....3

Şekil 2.3: Saykıl ve periyot.....3

Şekil 2.4: Modülasyon türleri.....4

Şekil 2.5: Genlik modüleli sinyal frekans tayfı.....5

Şekil 2.6: Çift yan bant genlik modülasyonlu verici blok şeması.....5

Şekil 2.7: Bilgi işareti.....6

Şekil 2.8: FM sinyali frekans spektrumu.....7

Şekil 2.9: PLL faz detektörü blok diyagramı.....8

Şekil 2.10: Ortak mod iletim formatları.....11

Şekil 2.11: Dijital sinyallerin iletilmesi.....12

Şekil 2.12: Asenkron haberleşme.....13

Şekil 2.13: simplex veri iletişimi.....13

Şekil 2.14: Half dublex seri veri iletişimi.....13

Şekil 2.15: Full dublex seri veri iletişimi.....13

Şekil 2.16: Senkron haberleşme.....14

Şekil 2.17: Verici modül pin yapısı.....17

Şekil2.18: Alıcı modül pin yapısı.....19

Şekil 3.1 Servo motor.....24

Şekil 3.2 Servo motorun iç görünümü.....25

Şekil 3.3 Servo motor çalışma şeması.....26

Şekil 3.4: H bridge (köprü) motor kontrol devresi.....28

Şekil 4.1 R Vektörü Koordinat Sisteminde Görünümü.....30

Şekil 5.1 Lcd Bağıntı Şekli.....34

Şekil 6.1: Çalışma Blok Şeması.....35

SİMGELER VE KISALTMALAR

Rf: radyo frekansı

W: watt

KW: kilowatt

LNB:Low Noise Block, dişik gürültü bloğu

Hz: hertz

MHz: Mega hertz

KHz: Kilo Hertz

GHz: giga Hertz

f: frekans

T: periyot

λ: dalga boyu

c: ışık hızı

fc: taşıyıcı sinyal

fm: Bilgi sinyali

VSB: Vestigial Side Band, Artık Yan Band Modulasyonu

SSB:Sıngle Sıde Band, Tek Yan Band Modulasyonu

DSB: Duble Side Band ,Çift Yan Bant Modülasyonu Modulasyonu

PM: Phase Modulatıon, Faz Modulasyonu

FM: Frequency Modulation Frekans Modülasyonu

PCM: Pulse Code Modulation Darbe Kod Modülasyonu

PPM: Pulse Position Modulation Darbe Konum Modülasyonu

PWM: Pulse Width Modulation Darbe Genişlik Modülasyonu

PAM: Pulse Amplitude Modulation Darbe Genlik Modülasyonu

V: volt,gerilim

Vc: taşıyıcı frekans gerilimi

Vm: bilgi frekans gerilimi

t: zaman

FET: Field Effect Transistor Alan Etkili Transistör

PLL: Phase Locked Loops faza kitlenmiş döngü

GM: genlik modulasyonu

VCO: Voltage controlled Oscillator, gerilim kontrollü osilatör

DC: direct current, dogru akım

AC: alternative currrent alternatif akım

BPS: Bit Per Second saniyedeki bit sayısı

sn: saniye

BER: Bit Error Rate Bit Hata Oranı

NRZ: non return zero, sıfıra donüşsüz kodlama

RZ: return to zero, sıfıra dönüşlü

RS-232: Recomended Standard 232

Ω: ohm

KΩ: kilo ohm

RF İLETİŞİMLİ MOBİL ROBOT KOLU

ÖZET

Bilgisayarlar insanların uzunca zaman harcayarak yaptıkları işleri saniyeler içerisinde yaparak işlerimizi kolaylaştıran cihazlardır. Robotlar, bilgisayarın gelişimi ile ön plana çıkmaya başlamıştır. Robotlar günümüzde, daha az insan gücü kullanımı, insanların tehlike altına girmemesi, daha az maliyet, daha hızlı ve kaliteli üretim gibi amaçlarla kullanılmaktadır. Bu çalışmada, bir robot koluna mobilite özelliği eklenmiştir. Ardından, rf iletişim ile robot kolu ve mobilite kontrolü yapılmıştır. Robotun tutucusu da belli eksenlerde yere paralel olması sağlanmıştır.

Anahtar Kelimeler : Robot kol, radyo frekansı, mobilite

MOBILE ROBOT ARM VIA RF COMMUNICATION

ABSTRACT

Computers are devices that facilitate our work in making people spend a long time by their jobs seconds. Robots have started to come to the forefront in the development of the computer. Robots today, less manpower utilization, prevent the entry of people in jeopardy, less cost, faster, and are used for purposes such as quality production. In this study, the mobility feature was added to a robot arm. Then, RF communications with the robotic arm and mobility check is performed. the holder of the robot is also provided to be parallel to the ground in certain axis.

Key Words : Robotic arm, radio frequency, mobility

1.GİRİŞ

İnsanlık tarih boyunca yeni arayışlar içerisinde çevresini incelemiştir. Tarihsel gelişimi içinde de birçok yenilik ortaya çıkarmıştır. Bu yeniliklerin temel amacı, insanoğlu için daha yaşanabilinir bir dünyada yaşam kalitesini arttırmaktır. Günümüzde de bu arayışlar devam etmektedir. Bu uğraşların bir sonucu olarak da hayatımızın birçok alanında robotlar yer almaktadır. Robotlar üzerine yapılan araştırmalar oldukça fazla ve önemlidir. Artık insanlığı ilgilendiren alanlarda, sanayide, askeriyede, uzay çalışmalarında, tıbbi alanda, arama kurtarma faaliyetlerinde, laboratuarlar ve daha birçok alanda geniş araştırmalarda devam etmektedir. Robotlar, daha az insan gücü kullanımı, insanların tehlike altına girmemesi, daha az maliyet, daha hızlı ve kaliteli üretim gibi amaçlarla kullanılmaktadır.[1-3]

Laboratuarda kullanılması amacı ile mobil robot kolu yapılmıştır. Bu tezde de RF iletişim ile mobilite sağlanmış robot kolunun uzaktan kontrolü sağlanmıştır. Bu robot koluna ilave olarak gyro (denge) sensörü ile gripperın (tutucunun) dengesi sağlanmıştır. Böylece robot dururken ve hareket ederken gripperda (tutucuda) duran nesnenin pitch ve roll (yunuslama ve yuvarlanma) ekseninde yere paralel durması sağlanmıştır.

2.HABERLEŞME

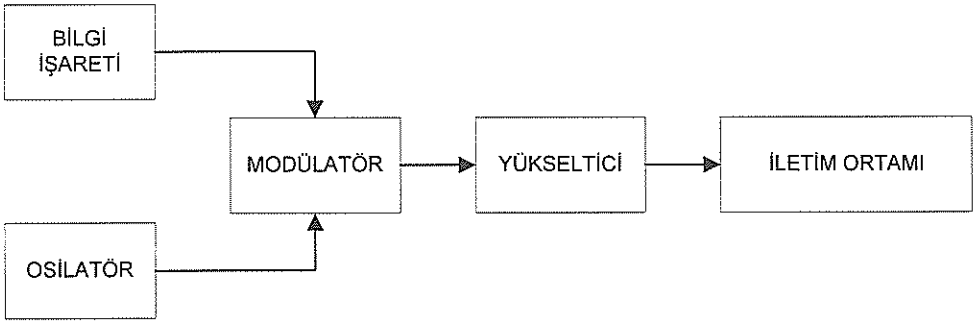
Karşılıklı bilgi alışverişi haberleşmedir. Haberleşme işleminin elektronik temelli ortamda yapılması telekomünikasyon olarak adlandırılır. Teknolojinin hızla ilerlemesiyle elektroniğin, internet ve kablosuz haberleşme de kolayca ulaşılabilinir kılmıştır. Bu da iletişimin ağırlıklı olarak artık elektronik ortamda yapılmaktadır.

2.2. Haberleşme sistemi

Haberleşme sistemleri verici, iletim ortamı ve alıcı ile oluşmaktadır.

2.2.1. Verici

Verici, bilgi işaretini osilatör ile alıcının anlayacağı şekilde modüle ederek iletim ortamına bırakan devrelerdir. Tüm verici istasyonları için farklı frekans aralıkları seçilmiş ve standart olarak belirlenmiştir. Buradaki amaç, iletim ortamındaki işaretlerin birbirine karışmaması ve frekans bandlarını daha fazlaca kullanmaktır. Verici, kendine ayrılan frekans bandında gönderdiği işaretleri ilgili frekans bandında kaydırır. Yapılan bu işlem ile birçok radyo istasyonu işareti birbirleriyle karışmaz. Vericilerin çıkış gücü ile iletim mesafeleri değişmektedir. Örneğin, televizyon vericileri 1W-20KW, gemi istasyon vericileri 400W-1.5KW, uçaklarda kullanılan radyo altimetre 0.5W çıkış gücüne sahiptir.



Şekil 2.1 : Verici blok şeması

2.2.2. İletim Ortamı

Alıcı ile verici arasında bilginin taşındığı ortam iletim ortamıdır. Örneğin, uydu ile LNB(Low Noise Block) alıcısı arasında bilgi, havayı iletim ortamı olarak kullanır.

Uydu alıcısı ve televizyon arasındaki bilgi iletimi, koaksiyel kablonun iletim ortamı olarak kullanılması ile sağlanır. İletim ortam çeşitlerini aşağıdaki gibi sıralayabiliriz:

i. Bakır tel kablo
ii. Çift bükümlü kablo
iii. Koaksiyel kablo
iv. Dalga kılavuzu
v. Fiber optik kablo
vi. Hava, boşluk ve su gibi doğal ortamlar

2.2.3. Alıcı

Verici devresinden iletim ortamına bırakılan modüleli sinyali alarak bilgi sinyalini elde eden elektronik devrelerdir.

Şekil 2.2: Genel bir alıcının blok şeması

2.3. Frekans, Periyot, Dalga Boyu

Frekans, bir işaretin birim zamanda yaptığı titreşim sayısıdır. Birimi Hertz (Hz) ve f ile gösterilir.

$$f=1/T$$

Formülü ile hesaplanır. Burada;

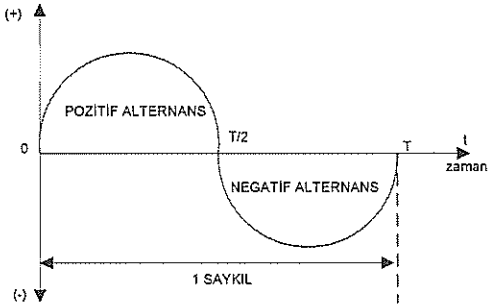
f = Frekans

T= Periyot'tur.

İşaretin bir saykıl yapması için geçen süre ise periyottur. Birimi saniye olarak ifade edilir.

$$T=1/f$$

Formülü ile hesaplanır.



Şekil 2.3: Saykıl ve periyot

Dalga boyu, işaretin bir saykılda aldığı yoldur. Birimi metredir. λ ile ifade edilir. Elektromanyetik sinyallerin hızı boşlukta ışık hızı olarak ele alınmaktadır. Dalga boyu, ışık hızı ve o sinyalin periyodunun matematiksel çarpımı ile bulunur.

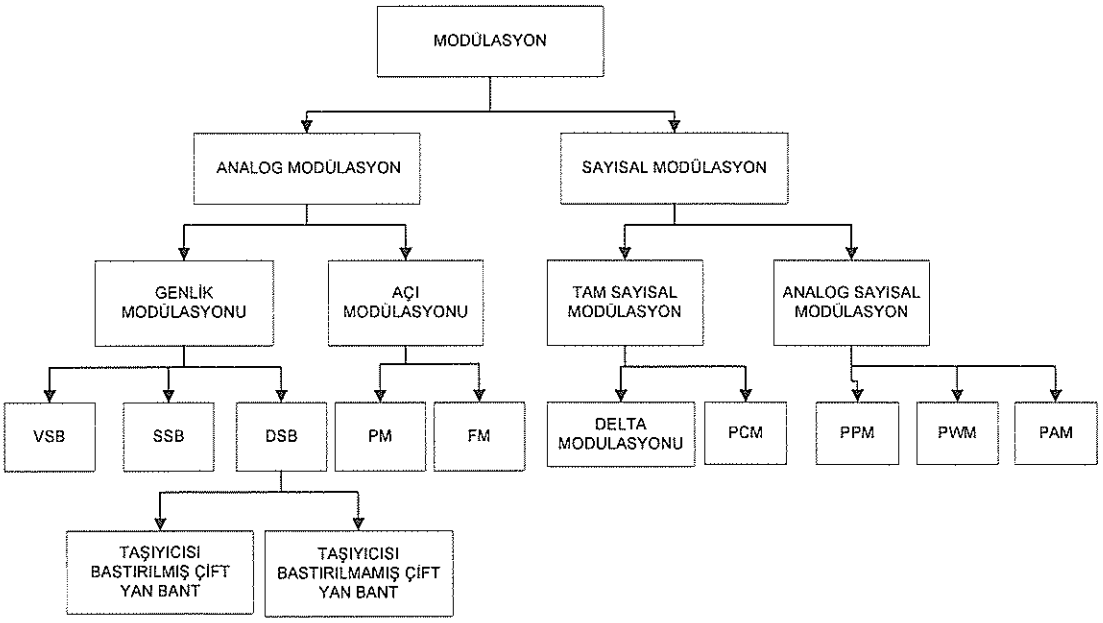
Dalga Boyu (λ) = Işık Hızı (C) *Periyot (T)

2.4. Modülasyon

Modülasyon, bilgi sinyali ile taşıma sinyalinin birleştirilmesi işlemidir. Bu işlem genellikle düşük genlik ve frekansa sahip bilgi sinyali ile yüksek frekanslı bir sinyale bindirilmesi ile yapılır. Bilgi sinyaline göre en az bir parametresi değiştirilen yüksek frekanslı sinyal de taşıyıcı sinyaldir(fc). Anlamlı bilgiye sahip (veri, görüntü, dosya vb.) düşük güce ve frekansa sahip sinyal, bilgi sinyalidir(fm).

2.4.1. Modülasyon Türleri

Modülasyon analog ve sayısal modülasyondur. Analog ve sayısal modülasyon da içlerinde farklılıklar göstermektedir.

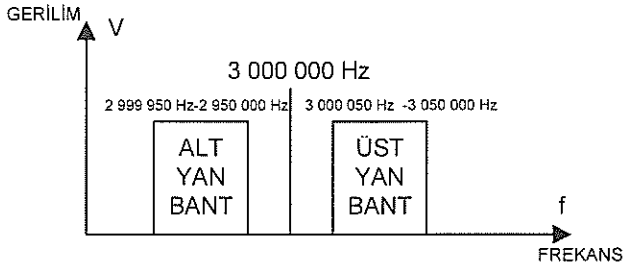


Şekil 2.4: Modülasyon türleri

2.4.2. Genlik Modülasyonu

2.4.2.1. Çift Yan Bantlı Genlik Modülasyonu

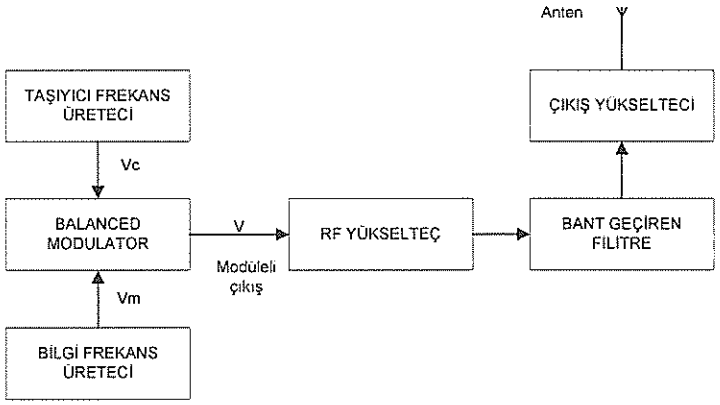
Taşıyıcı işaret genliğinin bilgi işaretinin durumuna göre değişmesi genlik modülasyonudur. Modülasyon yapılırken bilgi sinyalindeki frekanslar üst yan bant ve alt yan bant şeklinde meydana gelir. Şekil 2.5 'te örnek olarak 50Hz ile 50KHz arasındaki bilgi sinyali ile 3 MHz taşıyıcı sinyalin modülasyonundan sonra oluşan yan bantlar gösterilmektedir. Veri iletilmesinde yan bantların her ikisi de kullanılıyorsa, bu genlik modülasyonuna çift yan bantlı genlik modülasyonu adı verilir.



Şekil 2.5: Genlik modüleli sinyal frekans tayfı

2.4.2.2. Çift Yan Bantlı Genlik Modülasyonu Elde Edilmesi

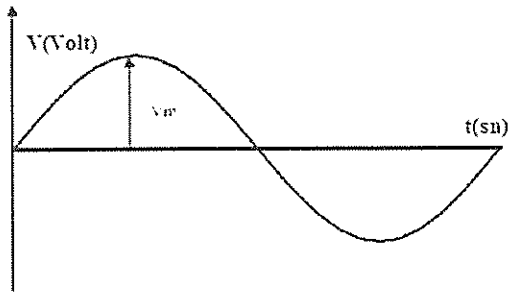
Genlik modülasyonunu oluşturmak için kullanılan devre modülatör devresidir. Modülatör bilgi sinyali ile taşıyıcı sinyali uygun şekilde birleştirerek modüleli sinyali oluşturur.



Şekil 2.6: Çift yan bantlı genlik modülasyonlu verici blok şema

2.4.2.2.1. Bilgi İşareti

Bilgi işareti, düşük frekanslı işarettir.



Şekil 2.7: Bilgi işareti

2.4.2.2.2. Taşıyıcı işaret

Taşıyıcı işaret, kosinüs ya da sinüs işareti olup yüksek frekanslıdır.

2.4.2.2.3. Modüleli işaret

Taşıyıcı işareti ve bilgi işaretin belli kurallarla birleştirilmiş halidir.

2.4.2.3. Tek Yan Bant (SSB-Single Side Band)Modülasyon

Tek yan bant modülasyonunda, genlik modülasyonunda ihtiyaç duyulan bant genişliğini yarıya indirerek işaretin alt ya da üst yan bantlarından birisinin filtre ile seçilip gönderilir. Genellikle uzak mesafelere bilgi gönderirken kullanılır. Tek yan bant sinyalinde genellikle dengeli modülatör devreleri tercih edilir. Dengeli Köprü Modülatörleri, FET (**Field Effect Transistor**)'e sahip Push-Pull Dengeli Modülatörleri, Dengeli Halka Modülatör devreleri SSB sinyali oluşturmak için kullanılan modülatörlerden bazılarıdır.

2.4.3. Frekans Modülasyonu

2.4.3.1.Frekans Modülasyon İhtiyacı

Verici devrelerinde sinyal/gürültü oranının mümkün oldukça düşük olması aranır. Yüksek güce sahip genlik modülasyonlu vericilerin sinyal/gürültü oranı, oldukça yüksektir. Bu problemi aşmak için de frekans modülasyonu geliştirilmiştir. FM devrelerinde GM devrelerine farkla olarak PLL (**Phase Locked Loops**) sentezör devresi, vurgu (emphasis) devresi ve limiter devresi kullanılır. Frekans modülasyonu, bilgi işareti genliğine göre taşıyıcı işareti frekansının değişmesi ile modüleli sinyal elde edilir. [4]

Avantajları;

1. Sinyal üzerindeki gürültünün seviyesi azaltıldığı için sesin kalitesi yüksektir.
2. Frekans modülasyonu gürültü hassasiyeti genlik modülasyonuna kıyasla daha iyidir.

3. FM in yakalama etkisine sahiptir. Bu etkiye maruz kalmaktan sebeple istenmeyen sinyaller rahatlıkla bertaraf edilir. Benzer frekansta iki sinyalden herhangisinin çıkış gücü yüksek ise o sinyali alıcı tarafından alınması *yakalama etkisi (Capture)* olarak tanımlanmıştır.
4. PLL devrelerine sahiptir.

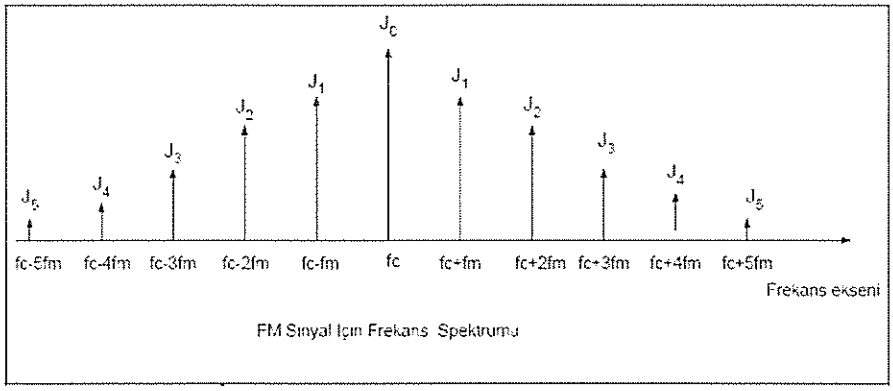
Dikkat edilmesi gereken durumlar;

1. FM bant genişliği oldukça fazladır.
2. FM devreleri pahalıdır.

2.4.3.2. Frekans Modülasyonunda Bant Genişliği

Frekans modülasyonunda bütün modüle edici tüm sinyallerde birer çift yan bant meydana gelir. Teorik olarak, bu durum frekans modülasyonunda sonsuz sayıda yan bantın var olmasıdır.

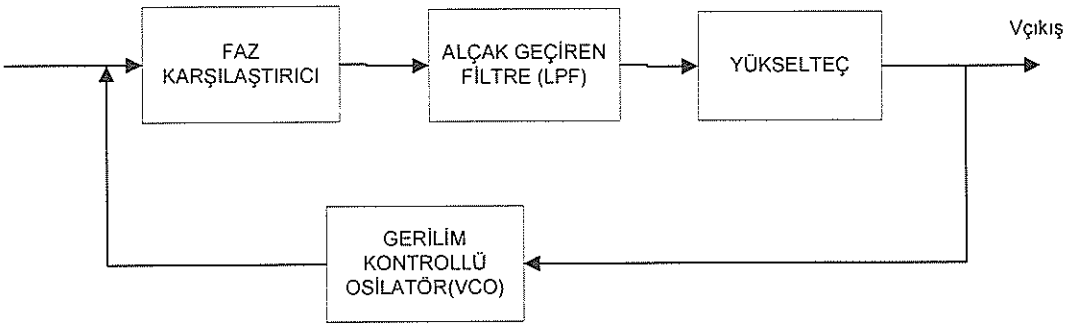
Örneğin 20MHz.lik taşıyıcı sinyali 200 KHz lik bir sinyali ile frekans modülasyonu yapılırsa, 20200-19800 KHz, 20400-19600 KHz, 20600-19400 KHz yan bant sinyalleri meydana gelir. Ancak, frekans değişimini arttırdıkça yan bant sinyallerinde güç azalmaları başlar. Taşıyıcı sinyal genliğinin yaklaşık %1'inden düşük yan bantların genliği tolere edilebilir. Frekans modülasyonu, ortalama ±75 KHz lik bant genişliğinde yapılır. Belirtilen bant değerinin altında yapılmış olan FM yayınlar dar bant frekans modülasyonu, bant değerinin üstünde yapılan yayınlar ise geniş bantlı frekans modülasyonu olarak adlandırılır. Bir FM sinyalinin frekans spektrumu Şekil 2.13'de gösterilmiştir.



Şekil 2.8: FM sinyali frekans spektrumu

1.3.4. PLL Faz Kilitli Döngü Faz Dedektörü

Faz dedektör devreleri; fm alıcı devrelerinde, fm verici devrelerinde, frekans kaymalı anahtarlama devrelerinde, modemlerde, izleme filtre gibi devrelerde kullanılır ve de PLL frekanslı geri beslemeye sahip kapalı döngülü kontrol devresidir. PLL devreleri yüksek maliyeti ve karmaşık yapısı olmasına rağmen gelişen entegre teknolojisi ile daha kullanışlı bir hal almışlardır.



Şekil 2.9: PLL faz detektörü blok diyagramı

Faz karşılaştırıcı: dışarıdan gelen sinyal ve Gerilim Kontrollü Osilatör 'den gelen sinyali çarpar. Yani çarpma işlemi yapan devredir. Girişindeki iki işaretin arasındaki faz veya frekans farkından orantılsal bir şekilde çıkışında DC gerilim meydan gelir.

VCO: İngilizce Gerilim Kontrollü Osilatör kelimelerinin ilk harflerini alan devre, gerilim kontrolü ile frekans üreten osilatör devresidir.

Alçak geçiren filtre: Faz Karşılaştırıcı devre çıkışını kullanarak fark frekansı bileşeni geçirir.

2.5. SAYISAL HABERLEŞME

2.5.1. Temel Kavramlar

2.5.1.1. Bit

Dijital elektronikte temel olarak 0 ve 1 vardır. Bütün işlemler 0 ve 1 üzerinden yapılır. Bu değerler bilgisinin her biri bit olarak adlandırılır.

2.5.1.2. BPS (Bit Per Second)

Sayısal veri iletişiminde saniyedeki iletimi yapılan bit sayısıdır, BPS olarak adlandırılır.

2.5.1.3. Baud

Bir alıcı ya da vericinin bir iletişim sırasında gönderdiği bilginin ölçüsüdür. Örneğin; bir cihaz her bir iletişim esnasında 10 bitle kodlanmış bir bilgi alıyor veya gönderiyor ise 1 baud, 10 bitdir.

2.5.1.4. Baud Rate (Oran)

Bir alıcı ya da vericinin çıkışında saniyede oluşan sembol (baud) değişikliği baud hızı olarak adlandırılır. Baud hızının gösterimi baud/saniye iledir.

2.5.1.5. BER: Bit Hata Oranı (Bit Error Rate)

İletişim sırasında gönderilmiş olan verinin yanlış algılanması veya veri içindeki bozulan bit oranıdır.

2.5.1.6. Kanal

Sinyallerin geçtiği, frekanslardan oluşturulan banda verilen addır.

2.5.1.7. Kanal Kapasitesi

Mevcut kanalın saniyede en fazla iletilebileceği bit miktarına verilen addır.

2.5.1.8. Gürültü

Asıl sinyalleri olumsuz olarak etkileyen durumlardır. Etkin gürültü çeşitlerini sistem içi ve sistem dışı olarak ayırabiliriz;

2.5.1.8.1. Sistem İçi Gürültü Kaynakları

Isıl gürültü: Sıcaklığın artması ile serbest elektronların enerji seviyelerinin artması, iletken içindeki elektronların rastgele hareketini arttırır. Bu da sistemi etkiler.

Atış gürültüsü: P-N eklemlerine sahip devre elemanlarında elektronların hareketlerinden kaynaklı oluşan beklenmeyen elektriksel değişmelerdir.

2.5.1.8.2. Sistem Dışı Gürültü Kaynakları

Sistem dışı gürültü kaynakları şu şekilde sıralanabilir;

1. Güneş,
2. Şimşek ve yıldırım kaynaklılar
3. Lambalar
4. Elektrik motoru kaynaklılar

2.5.2. Örnekleme Teoremi

Haberleşme devrelerinde temel hedef hızlı ve sağlıklı bir şekilde veri iletişimini oluşturmaktır. Analog haberleşmedeki sakıncalardan dolayı sayısal haberleşme teknikleri daha çok tercih edilmektedir. Bu nedenle analog işaretler sayısal biçime dönüştürülür. Analog işaretin sayısal işarete dönüştürülmesi, analog işaret düzgün bir örnekleme frekansı ile örnekleme işlemi uygulanmasıdır. Bilgi işareti, belirlenen darbe dizisiyle çarpılır. Örnekleme için, A Bant genişliği olan işaret ile en az 2A

hızına örnekleme sinyali ile iletim ortamına verildiğinde; alıcıda da orijinal bilgi sinyali elde edilinebilinir.

Anahtarlama hızının fazla olması örnekleme yapılan işaretin, özgün işarete daha da benzemesini sağlar. fm band genişlikli bilgi işaretinde, örnekleme frekansı (fs=1/Ts) Nyquist ile sağlanan şarttı sağlamalıdır.

fs ≥ 2fm ; $fs = 2fm$ frekansı da Nyquist frekansı olarak adlandırılır.

Nyquist koşulu oluştuğunda, $f(t)$ işareti örnek değerleri fm olan uygun alçak filtreden geçirilmesiyle bilgi kaybı olmadan yeniden oluşturulabilinir; bu işleme ayrıca interpolasyon denir. Teorik olarak $fs = 2fm$'lik örnekleme frekansı yeterlidir, pratikte ise örnekleme frekansı, alıcıda bulunan devrelerin ideal olmamasından dolayı Nyquist frekansından büyük seçilir.

Analog bilgi işareti, sonlu süreli darbeler ve sonlu genlik ile çarpılması doğal örnekleme adını alır. Bu işlemde amaç, darbe katarı genliği işaretin biçimini korumaktır. Başka bir örnekleme şekli ise düz tepeli örneklemedir. Bu örneklemede, analog işaret örneklenirken örnek değeri darbe süresince sabit tutar.

2.5.3. Kodlama

Belirlenmiş sabit değerlere göre sinyalin değiştirilmesi kodlama olarak adlandırılır. Haberleşme sistemlerinde genellikle bilgi işreti kodlanarak iletişim sağlanır. Kodlamaya en bilindik yaygın örnek olarak Samuel Mors'un bulduğu telgraf sistemidir. Günümüzde ses iletimi çoğunlukla dijital haberleşme ile çözümlenir. Bilgi işaretinin kodlanarak ve dijital format ile iletilmesinin kazanımları olarak sıralayacak olursak başlıca;

1. Gürültüye karşı daha az hassastır.
2. Distorsiyon seviyesi daha azdır.
3. Kaybolan bilgi işaretinin tekrar oluşturulması daha kolaydır.
4. İletim verimliliği daha fazladır.

Bilgi işareti önce dijital forma dönüştürülür. Kodlama ile bilgi işareti sıkıştırılır ve iletim verimliliği arttırılır. K sayıda karakter, bit olarak kodlanacaksa gerekli bit sayısı n = log2 K formülü ile bulunur.

Burada:

n = Kodlama yapmak için gereken 2 li bit sayısı,

K= Karakter Sayısı'dır.

Kod Etkinliği:

Kodlama işlemi sonucunda gereken bit ve kullanılmış olan bitler arasındaki oranıdır. Oranın değerinin yüksek olması kodlamanın etkinliğini artırır.

Parite: Gönderilmiş veride hata kontrolü parite biti ile kontrol edilir. Parite biti ile kontrol sağlanan sistemlerde, gönderilen karakterlerin sonunda ilave bir bit eklenir. İlave olan bit parite bitidir. Bu bit parite üreteci aracılığı ile üretilir.

Tek parite: Verilerdeki bilginin içerdiği bitlerde bulunan 1' ler toplanır ve sonuç tek ise parite biti 0 yapılır.

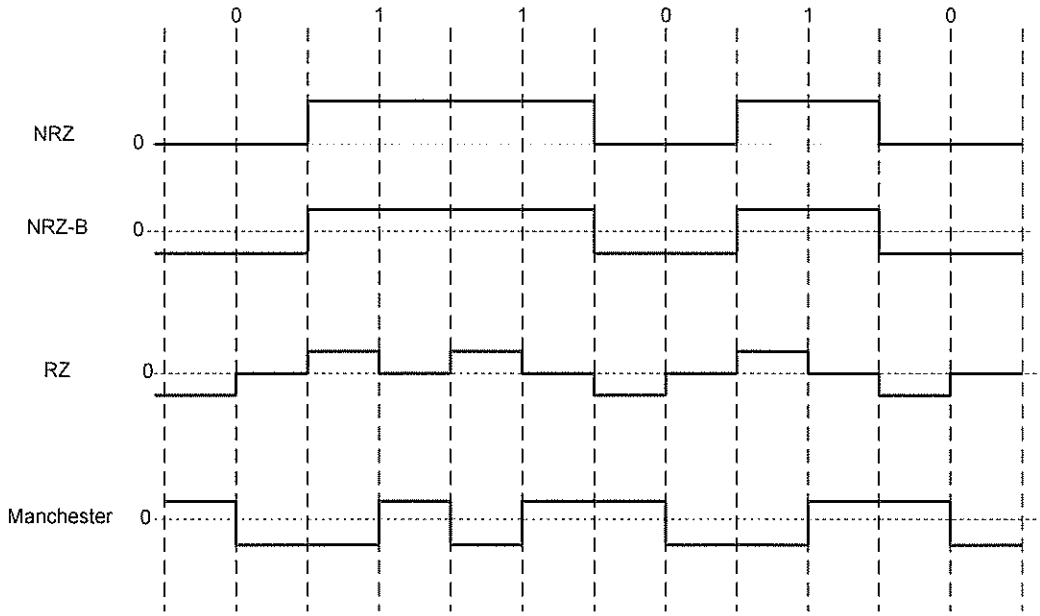
Çift parite: Verilerdeki bilginin içerdiği bitlerde bulunan 1' ler toplanır ve sonuç tek olursa parite biti 1 yapılır, eğer çift olursa 0 yapılır.

2.5.3.1. Kodları İletim Formatları

Verilerin kodlanmasında bir çok kodlama yöntemleri vardır.[5]

Bu kodlama tekniklerinden bazıları ise;

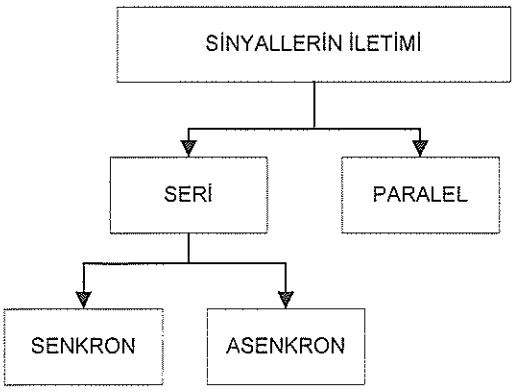
1. NRZ,
2. NRZ-B,
3. RZ,
4. Manchester.



Şekil 2.10: Ortak mod iletim formatları

2.5.4. Data Gönderilmesi

Sayısal haberleşme seri iletişim ve paralel iletişim olarak ikiye ayrılır.



Şekil 2.11: Dijital sinyallerin iletilmesi

Seri iletişim, paralel iletişime kıyasla daha yavaş ve yazılımsal olarak da daha uğraştırıcıdır. Seri port üzerinden haberleşen bazı cihazlar ile haberleşme yapabilmek için iletişimin paralel iletişime çevrilmesi gerekebilir. Çevirim işlemi için de UART (Universal Asynchronous Receive Transmit, Evrensel Asenkron İletim Alma) devreleri kullanılır. Seri iletişimin sakıncalarına rağmen kullanılmaktadır. Sakıncaları ise;

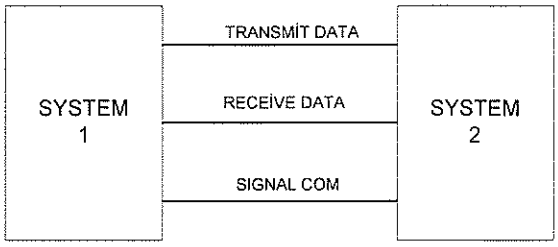
1. Seri iletişimde kullanılan kablolar paralel iletişimde kullanılan kablolara göre daha uzundur. Seri iletişim devrelerinde, lojik 1 seviyesi 3-25V aralığındadır. Paralel haberleşmede lojik 1 seviyesi 5 V'tur. Seri haberleşmede kablo kayıpları çok fazla etkilemez.
2. Seri iletişimde daha az bağlantı kabloları kullanılır.
3. İnfrared (kızıl ötesi) iletişimde seri haberleşme kullanmaktadır.
4. Mikrodenetleyiciler dış ortamla haberleşme işlemleri için seri iletişimi kullanılır. Seri iletişim ile entegrenin ayakları daha az olur.

Seri veri iletişimi kendi içinde asenkron ve senkron olarak ayrılır.

2.5.4.1. Asenkron Seri Data Gönderim

İletimin aynı zaman diliminde olmamasından (asynchronous) dolayı alıcının ve göndericinin eşgüdümlü olmasına gerek yoktur. Gönderici belli bir dizinde hazırladığı veriyi iletim hattına aktarır. Alıcı ise hattı sürekli dinler, verinin geldiğini bildiren işareti ya da bloğu alındıktan sonra gelen veriyi tutar ve toplayarak

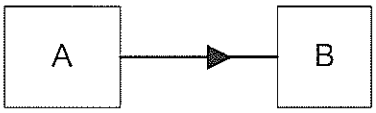
karakterleri oluşturur. Gönderilen her karakter sekiz bitten meydana gelmesi bilginin işlenmesinde kolaylık oluşturur.



Şekil 2.12: Asenkron haberleşme

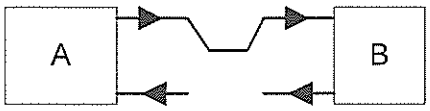
Asenkron seri data gönderiminde lojik1 5 Volt,lojik0 0 Volt seviyedir. Seri veri, asenkron RS-232 standardında gönderilirse voltaj polariteleri lojik1 -12 Volt, lojik0 +12 Volt olarak tanımlıdır. Ayrıca kullanılan iletişim kanallarına göre *simplex, half duplex* ve *full duplex* olarak üçe ayrılır.

Seri veri iletişimi tek yönde oluşuyorsa, veri iletimi *simplexdir*.



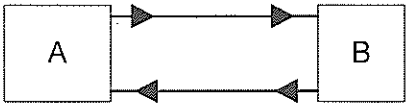
Şekil 2.13: simplex veri iletişimi

Burada verici ve alıcı karşılıklı iletişimdedir ama aralarında tek bir hat kullanılmaktadır. Yani, veri gönderiliyor ve de alınabiliniyor ise *duplexdir*. Bir tarafın gönderdiği veri bitirilmeden diğer taraf gönderim yapamaz, bir iletişim hattı üzerinden haberleşeme *half duplex haberleşmedir*.



Şekil 2.14: Half dublex seri veri iletişimi

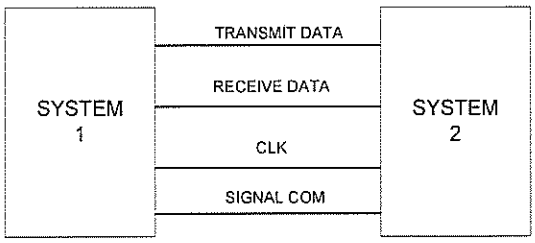
İki tarafın aynı anda veri gönderilip ve de alınabildiği iki ayrı iletişim hattı üzerinden haberleşme sağlayan iletişim *full duplex haberleşme*dir.



Şekil 2.15: Full dublex seri veri iletişimi

2.5.4.2. Senkron Seri Data Gönderim

Verinde Start–Stop bitleri kullanılmadan byte blokları halinde gönderilmesi senkron seri veri iletimi olarak adlandırılmaktadır. Başlangıçta, alıcı ve gönderici arasında senkronizasyonun sağlanması amacı ile senkronizasyon bitleri gönderilir.



Şekil 2.16: Senkron haberleşme

Data blokları gönderilirken sonlarına blok sonu karakteri eklenir. Gönderilen her blok için parite kontrolleri sağlanır blok son blok ise de yazı sonu karakteri sonuna eklenir. İletilecek bilginin bitmesi ile de önceden belirlenmiş iletim sonu karakteri gönderilir. Blok parite kontrolünü yapmak amacı ile blok karakter kontrol kullanılır. BCC (Blok Karakter Kontrol) gönderilen data bloklarındaki karakterler için yatay parite ve dikey parite kontrolü yaparak, diğer data bloğunun gönderilirken Acknowledge-izin-(ACK) karakteri gönderir. Parite kontrolü sonucunda hata tespit edildiğinde önceki data bloğunun tekrar gönderilmesi amacıyla Not Acknowledge-(NAK) karakteri üretilir ve gönderilir. [6]

2.5.5. Darbe Kod Modülasyonu Ve Kodlama Teknikleri

Bilgi işareti frekansından, belirlenmiş aralıklarda ve de iki katı ya da daha fazla frekansta örnekler alıp, belirli dizinler arasına yerleştirilerek ikili sayı sistemini kullanarak kodlama işlemi darbe kod modülasyonu (PCM – Pulse Code Modulation) olarak tanımlanır ve aşamaları;

1. Örnekleme
2. Kuantalama
3. Kodlama

Darbe kod modülasyonu (PCM), darbe modülasyon tekniklerinde sayısal (dijital) iletim tekniği olarak tektir. Darbe kod modülasyonunda, darbeler sabit uzunluk ve genliğe sahiptir. Darbe kod modülasyonunda, analog sinyal örneklenerek iletim için sabit uzunluğa sahip seri binary sayıya çevirilir. Binary sayı da analog sinyalin

gerilim değerine göre farklılık gösterir. Örnekleme, bilgi sinyalinden örnekler alıp, işlenerek örneklerin gönderilmesidir. Devamında da örnek değerler önceden belirlenmiş seviyelerden en yakın olanına yaklaştırma yapılarak bir binary kod sözcüğü ile kodlama işlemi yapılır. Böylece, 0-1 dizisi bir darbe dizisi oluşturulur. Burada "1" darbeyi, "0" ise darbenin olmadığını belirtir.

Kodlamada yöntemi üç aşamalıdır:

Birincisi: Vericiden alınan sinyal örneklenip kuantalama işlemi yapıldıktan sonra sinyalin pozitif alternas veya negatif alternas olduğuna bakılıp; pozitif alternasta ise 1, negatif alternasta ise 0 örnek işareti ile kodlanır.

İkincisi: Örneklenen sinyal eşit olmayan hangi bölüm aralığında ise ikili sayı ile ve bit olarak kodlanır.

Üçüncüsü: Örnekleme kuantalama işlemine uğtamış sinyalin segment aralığındaki değerine bakılıp segment aralıklarında bulunan değerlerden hangisi ise verilen ikili kod ile kodlanır. Kodlama işleminden sonra alınan örnek kodlar PCM sisteminin içerisinde zaman aralıklarına konumlandırılır.

2.5.5.1. Kuantalama İşlemi

Analog sinyalin dijital sinyale dönüştürülmesine kuantalama adı verilir. Kuantalanma ile işaretin en küçük değeri ile en büyük değeri arasını basamaklara ayırarak işaretin belirlenen basamaklar ile yaklaşık değerini oluşturmayı amaçlanmaktadır. Kuantalama, lineer ve lineer olmayan kuantalama olarak ikiye ayrılır. Burada en önemli nokta örnekleme frekansının doğru seçilmesidir. Örnekleme frekansı Nyquist oranıdır. Nyquist ölçütlerinde örnekleme frekansı, maksimum giriş frekansının 2 katı veya daha fazla olmasıdır. fs örnekleme frekansı olarak; fs>=2f olmalıdır.

2.6. RF Modüller

Kullanıcı düşük maliyete sahip yüksek performans ve kolay kullanılabilen ürünleri kullanır. Bundan dolayı Radyo haberleşme ürünleri, düşük maliyet, lisans gerektirmeden basit bir şekilde kullanım ile yüksek iletişim performansı gösterirler. Bu özelliklerin hepsine sahip çok az ürün mevcuttur; üreticiler tarafında ise bu tür ideal ürünleri ortaya koymak ciddi bir problemdir. Yaygın olarak yüksek frekansta çalışan devrelerin, teorideki kadar başarılı çalışmadığı bir gerçektir. Ayrıca kablosuz sistemlerde yasal mevzuatlar da işin içine girdiğinde, işin zorluk seviyesi bir kat daha

artmaktadır. Bu sorunların oluştuğu durumlarda yüksek frekansta çalışan ve devreye entegre edilebilecek bir devre elemanı ya da modülün ihtiyacı ortaya çıkmaktadır. Özellikleri ve çeşitlerine göre piyasada olan RF modülleri;

A. **Çeşitlerine göre :** üç gruba ayrılırlar. Bunlar; sadece sinyal ileten Vericiler, sadece sinyal alan Alıcılar ve de hem alıcı hem verici görevi gören Alıcı vericileridir. Alıcı ve vericiler tek yönlü iletişim için kullanılırlar. Ayrıca alıcılar ve vericilerin tek modülde olması çift yönlü haberleşmeye imkan sağlamaktadır.

B. **Frekans Bandına göre :**

1. **434 MHz Bandına göre:** iletişim frekans bandı aralığında sağlanır. 434 MHz bandındaki radyo dalgalarının kullanımı geniş bir alana yayılmıştır.
2. **868 MHz Bandına göre:** iletişim frekans bandı aralığında sağlanır.434 MHz bandı ile kıyaslandığında, kullanılan ekipman miktarı az olduğu için daha az parazit ile karşılaşılır. Kullanım alanı 434 MHz modülleri ile kıyasla daha dardır.
3. **2.4 GHz Bandına göre:** yüksek oranlı düzlüğe sahiptirler. 400 MHz e kıyaslandığında dalgalar daha az kırılmaya sahiptirler. Fakat yüksek bit oranı ve güvenliği, daha az parazit karışımı olması ile daha yaygındır.

C. **Dijital modülasyon sistemine göre :** Bu modüllerde kullanılan modülasyon sistemleri FSK (CPFSK, MSK, GMSK), PSK (BPSK, QPSK, DQPSK) ve ASK dır.

D. **Kullanılan frekans kanallarına göre:** sentezleyici sistemler belirli frekans ve sabit genlikte frekans kanalı desteklenir ve de daha düşük enerji sarfiyatına sahiptirler. Kullanıcı eğer kullanımı için planladığı kanalın meşgul olduğunu görürse iletişim için başka kanallara da geçiş yapabilir. Otomatik olarak kanallar arasında geçiş yapabilen hazır fonksiyonlu modüller vardır. Frekans kanallarına bağlı olarak, iletimdeki zaman limitleri oluşur.

E. **Osilasyon devrelerinin tiplerine göre:** Kristal ve rezonatör olmak üzere iki tür osilasyon vardır. Kristal devresin yüksek frekans kararlılığa sahiptir. Ancak rezonatör devresi daha düşük seviyede frekans kararlılığına sahiptir. Genellikle kristal osilatör, devrede düşük maliyetinden dolayı tercih edilir.

F. Data giriş/çıkış metodlarına göre: Seri data ve paralel data girişi olarak iki tip format mevcutur. NRZ, NRZI, DMI, Manchester kodları ve benzeri kodlar seri data formatlarına uygun kodlardır.

G. Data iletim hızına göre: veri gönderim hızları geniş bir alanda yapılabilir. Veri iletimi için yapılan radyo modülleri için yüksek hızlarda veri aktarımı yapılabilir.

H. Yasal Prosedürlere göre: kullanılan modül Kısa Mesafe Erişimli Telsiz Cihazlarının Temel Standartları İle Kurma Ve Kullanma Esasları Hakkında Yönetmeliğ'ine (TGM-STK-001) uyumlu olmak durumundadır.

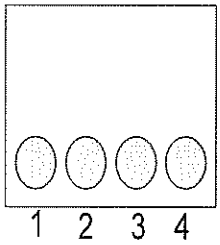
I. Uygulamalarına göre: Pil ile çalışan modüller küçük ebatlarda ve düşük enerji kullanan devreler olmalıdır. Endüstriyel alanlarda kullanılan modüllerde dış etmenler ağırlık kazanmaktadır.

2.6.1. Tezde kullanılan rf alıcı-verici modülleri

2.6.1.1. ATX-34S UHF ASK verici modülü

ATX-34S UHF ASK verici modülü, Kısa Mesafe Erişimli Telsiz Cihazlarının Temel Standartları ile Kurma ve Kullanma Esasları Hakkında Yönetmelik (TGM-STK-001) 'in 433-434MHz. ISM bandı ile ilgili bölümüne uygun olarak tasarlanmıştır.

ATX-34S kısa mesafelerde uzaktan erişim devrelerinde uygun fiyatı ile kolay bir tercihtir. Modülün anten haricinde hiç bir RF komponentine ihtiyacı yoktur. Basit bir kablo da anten bağlantısı yapılarak kullanılabilinir. ATX-34S modülü kullanıcının tasarımına rahatça uyabilecek şekilde yapılmıştır. Modülün temel özellikleri 433.920 MHz. UHF bandında olması, yüksek frekans kararlılığı ve düşük akım ile çalışması sıralanabilinir.[7]



Şekil 2.17: Verici modül pin yapısı

Çizelge 2.1: Pin özellikleri

Pin No	Pin-İsim	I/O	Açıklama
1	Vcc	-	+DC Besleme
2	DIN	I	DIGITAL INPUT
3	GND	-	Kontrol kartının toprak ucu
4	ANT	O	Anten bağlantı noktası

Besleme Voltajı: modül içerisinde bir voltaj regülatorü olmadığından dolayı besleme voltajına dikkat edilmesi gerekmektedir. Eğer modül, belirlenen değerlerin altında beslenir ise kararsız çalışır. Eğer ki modül belirtilen değerlerin üstünde veya besleme uçları ters bağlanırsa modülde kalıcı hasarlar oluşur. Düşük maliyet elde edebilinmesi için modül içerisine ters polarizasyon ve yüksek voltajdan koruyacak devre eklenmemiştir. Besleme voltajında ±100 mV değişimler üstü değişimler modülde kararsız çalışmalara sebep olur. Besleme voltajında sabitleme yapmak için regülatör devresi kullanılması daha uygun olacaktır.

Data Format: dijital datanın gönderilmesi için DIN pini mevcuttur. DIN pini gönderilen sinyallerin verildiği giriştir. Data protokolün ise aşağıdaki gibi yapılmıştır.

TX: (5*preamble)+sencron+data

Preamble, sencron ve data sinyal bloklarında veriler gönderilirken Manchester kodlama tekniği ile kodlanarak vericiye gönderilmiştir. Haberleşme sistemlerinde mesaj başlangıcında preamble kullanılması neredeyse zorunludur. Preamble ardışık 1 ve 0 lardan oluşan (01010101...) bir dizindir. Gönderilen tüm 1 ve 0'ların zamanları eşittir. Preamble donanım eşzamanlamayı sağlar. Aslında preamble alıcıya gönderilen bilgi için bir uyandırma sinyali görevi görmektedir.

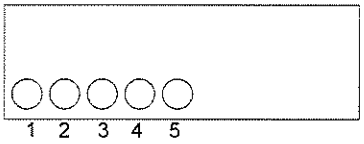
Sencron senkronizasyonunun sağlanmasına ve mesaj başlangıcının doğru tayin edilmesine olanak sağlar.

Data gönderiminde araya boşluk girmeden gönderilmelidir. Eğer arada boşluk oluşursa ise tekrardan preamble ve sencron gönderilmelidir. RX preamble'a bakmaz. Genellikle sencron aranıp, data okunur. Devremizde preamble gönderilirken beş kez, sencron ve data ise birer kez gönderilmiştir. Burada preamble'ın beş kez

gönderilmesi ile güvenliğin arttırılarak başka sinyallerin araya girmemesi amaçlanmıştır.

2.6.1.2.ARX-34C UHF ASK alıcı modülü

ARX-34C UHF ASK alıcı modülü, Kısa Mesafe Erişimli Telsiz Cihazlarının Temel Standartları ile Kurma ve Kullanma Esasları Hakkında Yönetmelik (TGM-STK-001) 'in 433-434MHz. ISM bandı ile ilgili bölümüne uygun olarak tasarlanmıştır. ATX-34C kısa mesafelerde uzaktan erişim devrelerinde uygun fiyatı ile kolay bir tercihtir. Modülün anten haricinde hiç bir RF komponentine ihtiyacı yoktur. Basit bir kablo da anten bağlantısı yapılarak kullanılabilinir. ARX-34C modülü kullanıcının sistemine entegre olabilecek şekilde yapılmıştır. Modülün temel özellikleri 433.920 MHz. UHF bandında olması, yüksek frekans kararlılığı ve düşük akım ile çalışması sıralanabilinir. [8]



Şekil2.18: Alıcı modül pin yapısı

Çizelge 2.2: Pin özellikleri

Pin No	Pin-İsim	I/O	Açıklama
1	ANT	I	Anten bağlantı noktası
2	GND	-	Kontrol kartının toprak ucu
3	Vcc	-	+DC Besleme
4	Aout	O	ANALOG OUTPUT
5	Dout	O	DIGITAL OUTPUT

Besleme Voltajı: modül içerisinde bir voltaj regülatorü olmadığından dolayı besleme voltajına dikkat edilmesi gerekmektedir. Eğer modül, belirlenen değerlerin altında beslenir ise kararsız çalışır. Eğer ki modül belirtilen değerlerin üstünde veya besleme uçları ters bağlanırsa modülde kalıcı hasarlar oluşur. Düşük maliyet elde edebilinmesi için modül içerisine ters polarizasyon ve yüksek voltajdan koruyacak devre eklenmemiştir. Besleme voltajında ±100 mV değişimler üstü değişimler modülde kararsız çalışmalara sebep olur. Besleme voltajında sabitleme yapmak için regülatör devresi kullanılması daha uygun olacaktır.

Data Format: dijital datanın alınması için DOUT pini mevcuttur. DOUT pininden alınan sinyallerin demodüleli olarak alındığı pindir.

Analog out: Analog out çıkışı test içindir. Çıkışındaki sinyal 1,5 Vdc seviyenin üstüne eklenmiş olarak ölçülür.

3.MOTORLAR

Elektrik motorları için klasik bir tanımla yapacak olursak, elektrik enerjisinin mekanik enerjiye dönüşümünü sağlayan elektrik makineleridir. Elektrik motorlarının çokça bir kısmı döner düzeneğe sahiptir. Bir sabit (stator), hareketli silindirsel(rotor), eş eksenli iki ferromanyetik armütürden oluşur. Yaygın kullanılan bir aygıt olan elektrik motorları bulaşık makinesinin pompasını, oda havalandırma modülü pervanesini, oyuncak arabaların hareketli parçalarını çalıştırır. Artık günümüzde elektrik motorları genellikle mikro işlemcilerle donatılmış veya hazır sürücü devreleri kullanarak kullanıcıların ihtiyaçlarına göre ayarlanabilir duruma gelmiştir. Elektrik motorları DC ve AC olarak iki ana başlık altında toplanabilir.[9]

3.1. Alternatif Akım (AC) Motorları

Alternatif akım motorlarını iki ana başlık altında toplayabiliriz; asenkron motorlar ve senkron motorlar. Bu iki gruptaki motorların temel ilkesi, metalden yapılan kütlenin, döner elektromanyetik alan sayesinde sürüklenmesine dayanır. Bu motorlardan, asenkron tipler standarttırlar. Senkron tipleri de büyük güç gerektiren yerlerde kullanılabilir. Bu iki grup motorlar eksenli iki armütüre sahiptir; stator sabit, rotor da hareketlidir. Senkron motor ve asenkron motorların statorları aynı şekil ve yapıya sahiptir.

Senkron motorun manyetik alanı, rotorun sargısını besleyen bağımsız bir doğru akım oluşturur. Senkron motorlarda başlıca yetersizlik, rotorun kendini harekete geçirememesidir. Özsenkron motorlarda, rotor sargısı yerine ise sabit mıknatıslar bulunmaktadır. Asenkron motor çalışması ise farklıdır; rotorun sargısı çok fazladır. Rotora yalnız statorda oluşan tek alan akım indüklenir. Rotor başka hiçbir enerji kaynağına ise bağlanmamıştır. Sincap kafesli motorların sargısı, yapraklı bir rotorun yivlerine yerleştirilmiş genellikle bakır çubuklardan meydana gelir. Bu yapı dayanıklı, sade ve ekonomiktir.

3.2. Doğru Akım (DC) Motorları

Hareketleri net, keskin ve güçlüdür. Hızları rahatça değiştirilir. Motor sürekli durup çalışacak, hassas hız ayarlı olacak ve de yük altındayken ani frenlemelere maruz kalacaksa, motorun seçimi zordur. Manyetik alan içerisinde duran iletken telden akım geçerse, iletken tel üzerinde hareket gözlemlenir.

DC motorlar da iletken telden geçen akım ile birlikte hareket enerjisini oluşturur. Hareketin yönü, manyetik alanın ve iletkenden geçen akımın yönü ile alakalıdır. Eğer manyetik alan ve iletkenden geçen akımın yönünün değiştirilmesi ile motorların dönme yönü de değiştirilebilir.

Doğru akım motorları şu şekilde gruplandıra biliriz;

1. **Fırçalı DC Motor**
2. **Fırçasız DC Motor**
3. **Step Motor**
4. **Servo Motor**

3.2.1. Fırçalı DC Motor

DC motorlardaki hareketli parçalardaki manyetik alan, elektrik akımı etkisiyle oluşur. Hareketli motor parçasına akım, sabit bir iletkenden verilemez. Dönme hareketiyle iletken motor miline sarılır. Fırça ve kolektör aracılığı ile motorun hareketli olan bu bölümüne akım aktarılır. Fırça ve kolektör kullanılan motorlar fırçalı dc motor olarak adlandırılır. Fırçalı dc motorlar gövdelerinde kullanılan manyetik alan kaynağına göre ikiye ayrılırlar. Bunlar; sabit mıknatıslı ve elektro mıknatıslıdır. Tüm motor tiplerinde kullanılan parçalar benzerdir. Bu parçalar ise endüvi(rotor), gövde(stator), fırça ve koruma aksesuarlarıdır.

3.2.1.1Endüvi (Rotor)

Endüvi; motorun dönen kısmıdır. Bu parça, silindirik gövde üzerindeki oluklara sarılmış sargılardan oluşur. Endüvi sargılarının uçları, fırçaların temas ettiği yere (kolektöre) bağlıdır.

3.2.1.2.Gövde (Stator)

N-S kutuplarını oluşturmak için yapılmış olan sargıların yerleştirildiği kısım indüktör olarak adlandırılır. Küçük motorlarda indüktörler, doğal mıknatıs ile yapılır; yüksek güçlü motorlarda ise indüktörleri bobinlerden oluşur

3.2.1.3.Fırçalar ve donanımları

Kolektöre yapışarak elektrik akımının iletilmesini fırçalar (kömür) sağlar. Fırçalar, motorun akım ve gerilim değerlerine göre farklı parçalardan üretilir. Fırçaların kolektöre düzgünce tutunmasını sağlamak için ise baskı yayları vardır.

3.2.1.4. Motor yan kapakları

Rotorun stator içinde merkezi olarak tutunmasını yan kapaklar yaparlar.

3.2.2. Sabit Mıknatıslı Motor

Fırçalı motorlarda stator kısmı sabit mıknatıslı olanlarına sabit mıknatıslı motor denir. DC motorların yapı olarak genel özellikleri sabit mıknatıslı motorlarda da görülmektedir. Yapısındaki fark, stator (gövde) içindeki sabit olan mıknatıslardandır. İki sargı ucu da nüve üzerinde birkaç turla iki kolektör dilimine bağlanır. Simetrik olan bu kolektör dilimleri motor içinde bulunan iki fırçaya aynı anda temas eder. Fırçanın birinden kolektör dilimine giriş yapan akım, rotor üzerinden manyetik alan oluşturarak diğer kolektör dilimine teması bulunan fırçadan devresini tamamlar.

3.2.3. Elektro Mıknatıslı Motor

Fırçalı motorlarda stator kısmı elektro mıknatıslı olanlarına sabit mıknatıslı motor denir. Endüstriyel sistemlerde kullanılan dc motorlar, genellikle elektro mıknatıslı motorlardır. DC motorların yapı olarak genel özellikleri elektro mıknatıslı motorlarda da görülmektedir. Yapısındaki fark, stator (gövde) içindeki manyetik alan oluşturan kısımdır. İndüktör elektro mıknatıstandır. Elektro mıknatısın oluşması için statorda kutup başları oluşturularak bobin sarılmıştır. Motora gerilim uygulandığında statordaki bir kutuptan(stator sargıdan) giren akım çıktığı stator kutbuna doğru bir manyetik alan oluşturur. Oluşan bu manyetik alan, rotora bir kuvvet uygular ve de bu kuvvetle motorda dönme etkisi oluşur. Akımın yönü değiştiğinde motorun devir yönü değişmez.

3.2.4. Fırçasız DC Motor

Dc Motorların rotor bölümünde manyetik alanın doğal yoldan karşılandığı motorlardır.

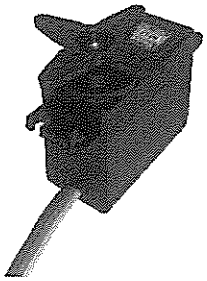
Rotor bölümünde akım yoktur. Bundan dolayı da akım taşıyacak fırça ve kolektör düzenekleri yoktur. Bu sebepten de bu motorlara fırçasız motor denir. Temel prensip ve çalışması bakımından fırçalı DC motorlarla aynıdır.

3.2.5. Adım (Step) Motor

Step motor, kontrollü dönüş hareketlerinin gerekli olduğu uygulamalarda kullanılan bir motordur. Dönme hareketinin gerçekleştirilmesi için sürekli bir gerilime ihtiyacı yoktur. Step motor sargılarına uygulanan elektrik palsine göre istenen miktarda döner. Mikrodenetleyiciler ile rahatlıkla sürülebilir. Step motorda, motorun dönmesi girişe uygulanan palsler ile bağıntılıdır. Palslerin sürekli ve sıralı olmasına dikkat edilmelidir. Palsler arasında geçen süre ayarlanarak motorun hızı ayarlanabilinir. Step motorların gücü ve momenti sınırlıdır. Step motor statoru çok sayıda kutupludur. Bu kutuplar birbirlerine simetriktir. Böylece, bobinler aynı akım yolu üzerinde olup aralarında bir manyetik alan oluştururlar. Manyetik alan değişimleri anahtarlama ile olur. Anahtarlama ile statorun güney ve kuzey kutupları döndürülür. Rotordaki manyetik alan, stator üzerinde oluşan manyetik alanla etkileşime girer. Adımları (stepler) ile stator manyetik alanı döndürür. Rotor da buna benzer adımlarla (steplerle) stator manyetik alanını takip eder.

3.3. Servo Motorlar

Konum kontrollü uygulamalarda genellikle adım motoru ve servo motorlar kullanılmaktadır. Adım motorlarında kullanımı alanları düşük güç ve moment gereken sistemlerdir. Servo motorlar ise yüksek güç ve moment ve de hızlı tepki gereksinimi olan sistemlerde kullanılır. Servo motorların kalkış ve duruş anları daha yumuşak gerçekleşir.[10]

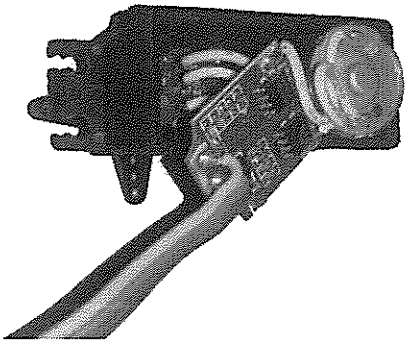


Şekil 3.1 Servo motor

Bu durum ise sistemde çift yönlü faydası vardır. Birincisi motor milinde yük olduğunda motorun herketi sırasında yükün zarar görmesi önlenir. İkincisi de motorun ilk hareketi sırasında yüksek akım çekmesini engeller. Servo motorların çok

çeşitli uygulamalarda kullanılmasının nedenleri arasında, yüksek torklu olması, doğru konumlanması, kolay kurulması ve kontrol edilmesi, ekonomik olmasıdır. Servo motor, bir DC motorun şaftının dişli sisteminin bir potansiyometre ile bağlanması ile meydana gelmektedir. Servo motorlar, üzerindeki kontrol devresi ve potansiyometrede gerilim bölünmesine göre pozisyonunda bilgi edinir ve gelen sinyalle karşılaştırarak doğru pozisyonuna hareket eder. Servo motorlar açılı dönerler. Bundan dolayı geniş kullanım alanları mevcuttur. Gömülü kontrol devrelerine sahip olan servo motorlar birkaç watt ile birkaç yüz watt olarak farklı güçlerde üretilirler. Servo motorların en yaygın kullanım alanları; radarlar, robot kolları, izleme sistemleri gösterilebilir.

Servo motorlar içlerinde DC motor, konum algılayıcı olarak potansiyometre, motor çıkışında olması istenilen açının oluşması amacı ile kontrol devresi vardır. 180 derece dönüş açısı mevcuttur. Potansiyometre, servo motorun miline bağlı olup milin o anki açısını kontrol devresine gönderir. Kontrol devresi de motoru istenilen açıya döndürür.



Şekil 3.2 Servo motorun iç görünümü

Servo motorlar daha çok model arabalarda, uçaklarda ve küçük güçteki robot uygulamalarında kullanılan motor türlerindendir. İçinde bir DC motor, kod çözücü, motoru sürücü devresi ve motor gücünü arttırmak amaçlı plastik ya da metal dişliler bulunmaktadır. Servo motorun elektronik sürücüsü fazla akım çekmediğinden dolayı mikrodenetleyici tarafından direkt olarak sürülebilir.

Genellikle plastik ya da metal dişli sistemler mevcuttur. Servo motorlar için çalışma gerilimi 4,8-6V aralığındadır. En sık kullanılan servo motorun gucu 1 cm/3.5kg'dır.

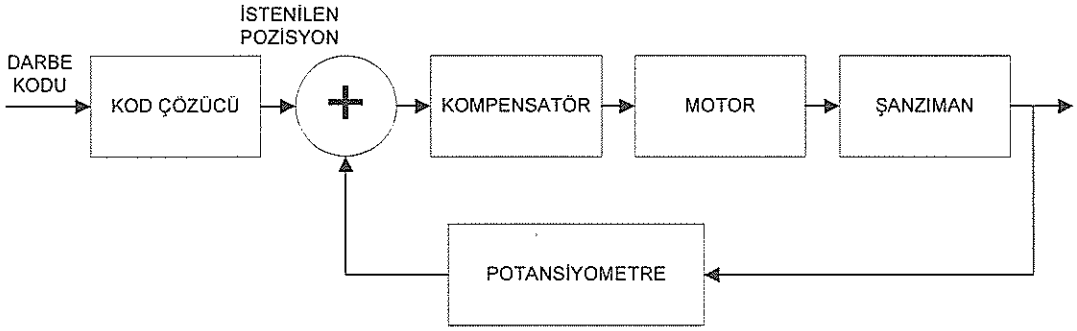
Servo motorun devreye üç bağlantı noktasından bağlanır. Bu noktalardan ikisi enerji için kullanılır. Diğeri de sinyal girişi içindir. Artı (+) hat genellikle kırmızı renk; eksi (-) hat genellikle siyah veya kahverengi renkli kablo ve sinyal hattı da sarı, turuncu ya da beyaz renkli kablo ile iletilir.

Servo motorun sahip olduğu DC motor, çift mıknatıslı bir statoru ve fırçalı bobin rotoru vardır.

Motor mili 1:200-1:300 arası dönüş oranı olan bir dişli sisteme bağlıdır. Böylece yüksek tork değeri elde edilir. Dişli sistemin çıkışı da 5kΩ'luk bir potansiyometreye bağlanıp, mil konumu elektronik kumanda devresine iletir. Elektronik devrende amaç, mil konumunu gelecek olan veri konumuna göre motorun dönüşünü sağlayıp olması gereken yerinde durdurmaktır.

3.3.1 Servo Motorun Temel Fonksiyonları

Servo motorun çalışma özelliği, alınan darbe kodu ile konum değiştirmektir. Servo motorun gerçekleştirdiği işlem aşağıda verilen blok diyagramda görülmektedir. Alınması istenilen konum ve şaftın pozisyonu karşılaştırılır. Kompensatör de gelen bilgiyi düzenleyerek giriş işareti olarak ayarlar. Motorun şanzımana bağlı olması, çıkışta düşük hız olsa bile yüksek tork gücü elde edilmesini sağlar. Potansiyometrenin görevi, geri besleme sinyali vermektir.



Şekil 3.3 Servo motor çalışma şeması

3.3.2 Servo Motor Kontrolü

Servo motorların kontrolü için PWM sinyali gönderilir. PWM, bir çok mikrodenetleyicinin donanımsal olarak desteklediği bir özelliktir. Servo motor konumları PWM (Darbe Genişlik Modülasyonu) yöntemi ile kontrol edilir.

Darbenin değeri sabit tutulup 20 ms'de bir tekrarlanırsa servo motor aynı konumda kalmaya devam eder. Servo motorun darbe değerleri markadan markaya farklılık gösterir. Darbe değeri bilinmeyen bir servo motorun süresi ya katalog ya da deneme yanılma ile bulunabilir.

3.3.3. Darbe Genişlik Modülasyonu (Pulse Width Modulation – PWM)

Darbe genişlik modülasyonu(PWM), üretilen darbenin genişliğinin kontrol edilerek çıkışta üretilen analog değer veya sinyalin elde edilmesidir. PWM, elektronikte birçok alanda farklı durumlar için kullanılmaktadır. Telekomünikasyon, güç, ses yükselteçleri gibi çeşitli uygulamalarda kullanılmaktadır. Üretilen kare dalga sinyalleri genişliklerinin ortalaması, üretilmesi istenilen analog değeri karşımıza çıkarır.

3.3.4. Donanım Kontrolörleri

Daha önce de bahsedildiği gibi mikrodenetleyicilerde dahili PWM modülü vardır. PWM uygulamaları yapılmadan önce, mikrodenetleyicide şu ayarlar yapılmalıdır;

Dahili sayıcılar ayarlanıp, kare dalga modülasyonunda kullanılacak çalışma periyodu belirlenmelidir,

PWM kontrolleri ayarlanmalıdır,

PWM çıkışının sağlanacağı pin belirlenmelidir,

Sayıcı başlatılmalıdır

PWM kontrolü aktif hale getirilmelidir.

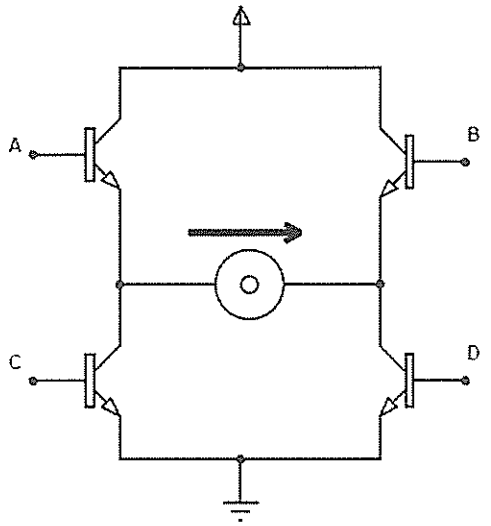
Biz programımızda mikrodenetleyicideki dahili PWM modülünü kullanmayıp, programda önceden belirlenen değerler kullanılacaktır.

3.4. Tezde Mobilite İçin Kullanılan DC Motorun Kontrolü

Tezde mobilite kontrolünü sağlamak için 12 volt DC redüktörlü motorlar kullanıldı. Motorların redüktörlü olması ile motorlarda düşük enerji sarfiyatı ile yükten etkilenmemesi sağlandı. Devre içerisinde ise motorların kontrolü L298 motor sürücü entegresi ile sağlandı.

3.4.1. L298 entegresi

DC Motorun uçlarına gerilim uygulandığında, motor bir yönde dönmeye başlar. Uygulanan gerilim ters olarak uygulanırsa motor bir önceki yönün tersine hareket edecektir. DC motorun yön kontrolünü sağlamak için çeşitli uygulamalar geliştirilmiştir. Uygulamalardan en yaygın olanı ise H brige(h köprü) 'dür.



Şekil 3.4: H bridge (köprü) motor kontrol devresi

H brige genellikle 4 transistör ile motorun iki yönde kontrolünü sağlar. Motorun gösterilen yönde dönmesi istenildiği zaman A ve D taransistörleri iletime geçirilir ve motor dönmeye başlar. Tersi istikamette dönmesi istenildiğinde ise B ve C transistörleri iletime geçirilir. Durması istenildiğinde ise bu dört uca gerilim uygulanmaması yeterli olacaktır. [11]

H brige yöntemi için en çok kullanılan entegre ise L298 entegresidir. L298 akım kontrolü yapabilmekte olup 2.5 Ampere kadar olan uygulamalarda da kullanılabilmektedir. Ayrıca entegrede iki adet H brige bulunmaktadır. Böylece iki DC motor kontrolünü tek entegreden sağlanır.

L298 entegresinde 15 bacak mevcuttur. Bu bacaklar IN1, IN2, IN3, IN4, OUT1, OUT2, OUT3, OUT4, ENB, ENA, SENSA, SENSB, VS, VSS, GND olarak adlandırılır. Bu uçlardan IN1, IN2, OUT1, OUT2, ENA, SENSA A köprüsünün kontrolüne, IN3,IN4, OUT3, OUT4, ENB, SENSB B köprüsünün kontrolüne ayrılmıştır.

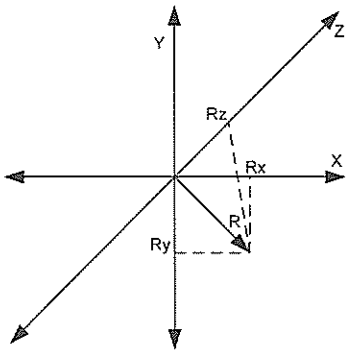
Çizelge 3.1: Pin özellikleri

Bacak Adı	İşlevi
IN1, IN2	A köprüsünün kontrolü
IN3, IN4	B köprüsünün kontrolü
OUT1, OUT2	A köprüsü çıkışı
OUT3, OUT4	B köprüsü çıkışı
ENA	A köprüsünü aktifleştirme
ENB	B köprüsünü aktifleştirme
SENSA	A köprüsündeki akım kontrolü
SENSB	B köprüsündeki akım kontrolü
VS	Çıkışlardan alınmak istenilen gerilim
VSS	Entegrenin Çalışma voltajı
GND	Entegrenin şase beslemesi

4. MPU-6050 SENSÖRÜ

MPU-6050 entegresi robot kolundaki Gripper'ı istenilen açıda tutmak için kullanılan Gimbal'da kullanıldı. MPU-6050 entegresi bir adet ivme ölçer (accelerometer) ve bir adet jiroskopa sahiptir. İvmeölçer, kendisi üzerlerindeki yerçekimi kuvvetini ve ani hızlanma ya da durma ivmesini ölçmektedir. Alınan değer m/s^2 türünden ifade edilebilir. Sensör, herhangi bir çekim alanının kapsamında değilse üzerine 1g lık yerçekimi kuvveti etki eder. Yer çekim kuvveti yaklaşık $9.8m/s^2$ ve de dünyada bulunulan noktaya göre değişir. Yer çekim etkisi bulunulan noktada ani bir değişime uğramadığı için, eğimölçer veya hareket algılayıcı olarak kullanırken etkisi göz ardı edilebilir. İvme ölçerin bir, iki ve de üç eksende ölçüm yapan türevleri mevcuttur. MPU-6050 sensörü üç eksende ölçüm yapabilir ama iki eksendeki ölçüm değerleri kullanıldı, ama sensörün özelliği nedeni ile üç ekseninin özelliği de anlatılacaktır.

Sensörün yeryüzüne olan açısının değişmesi, sensörün eksenlerine etki eden kuvvetleri de değiştirecektir. Yeni değerin okunması ile yeryüzü ile oluşan açı trigonometri ile hesaplanabilinir. R vektörünü ivme ölçer üzerine düşen kuvvet vektörü olarak kabul edelim. R vektörü 3 bileşenden oluşur ve R=[Rx,Ry,Rz]' dir. R vektörü, pisagor teoremi ($R^2 = Rx^2 + Ry^2 + Rz^2$) ile hesaplanabilir.[12-14]



Şekil 4.1 R Vektörü Koordinat Sisteminde Görünümü

Rx, Ry ve Rz bilinir ise, trigonometrik fonksiyon ile R vektörünün X, Y ve Z eksenleri ile kesiştiği açılar hesaplanır ve sensörün yeryüzü ile konum tespiti yapılabilir.

İvme ölçerden ADC yardımı ile her eksende -32768 ile +32768 arasında bir değer alınabilinir. Elde edilen bu değerler ile işlem yapılamaz. İşlem yapılabilmesi için g değeri bilinmelidir. g değeri ise ADC ile okunan değerler volt ile ifade edilir.

Değerleri volt olarak ifadede de en küçük değer için gerilim değeriyle (VDD 3.3 Volt) çarpılması gerekir. Okunan değer besleme gerilimi ile çarpılıp $2^{\text{ADC çözünürlügü}}$ ile bölünür.

$VoltRx = AdcRx * Vdd / 2^{16}$
$VoltRy = AdcRy * Vdd / 2^{16}$
$VoltRz = AdcRz * Vdd / 2^{16}$

İvme ölçerin 0 g deki değerine zero-g denilir. Değer sensörün kullanım kılavuzunda (datasheet) bulunabilinir ve de genellikle Vdd/2 olarak görülmektedir. Okunan değerden zero-g değeri çıkarılıp sensörün hassasiyet değerine bölündüğünde hangi eksene ne kadar kuvvet uygulandığı hesaplanabilir. Hassasiyet değeri de kullanım kılavuzunda (datasheet) bulunmaktadır.

$Rx = (AdcRx * Vdd / 2^{16} - VzeroG) / Hassasiyet$
$Ry = (AdcRy * Vdd / 2^{16} - VzeroG) / Hassasiyet$
$Rz = (AdcRz * Vdd / 2^{16} - VzeroG) / Hassasiyet$

Bu şekilde ölçülen değerler trigonometrik fonksiyonlar ile açısal değerlere çevrilir. Şekil 4.1'de gösterilen R vektörünün eksenler ile yaptığı açıları AR_x, AR_y, AR_z olarak adlandırırsak. AR_x açısının kosinüs değeri R_x /R değerini verir. R_x ve R bilindiğine göre arkkosinüs (arccosinüs) ile AR_x açısı elde edilir. Bu hesaplama AR_y, AR_z için de geçerlidir.

$\cos(AR_x) = R_x/R$, $AR_x=\arccos(Rx/R)$
$\cos(AR_y) = R_y/R$, $AR_y=\arccos(Ry/R)$
$\cos(AR_z) = R_z/R$, $AR_z=\arccos(Rz/R)$

MPU-6050 sensörü içinde ayrıca jiroskop bulunmaktadır.

4.1.Donanımsal Özellikler

Besleme voltajı	: 2.4V–3.46V
Lojik voltaj	: 1.8V±5% veya besleme gerilimi(VDD)

3 eksen ivme ölçer : Programlanabilir tam ölçek aralıklı ±2g, ±4g, ±8g ve ±16g

Bekleme modu besleme akımı : 8μA

İvme ölçer düşük güç mod akım değeri : 10uA 1Hz'de, 20uA 5Hz'de,70uA 20Hz'de, 140uA 40Hz'de.

Bağlantı ara yüzü :400kHz hızlı I^2C seri sunucu

Darbe tolerans :10,000g

5. LCD (Liquid Crystal Display)

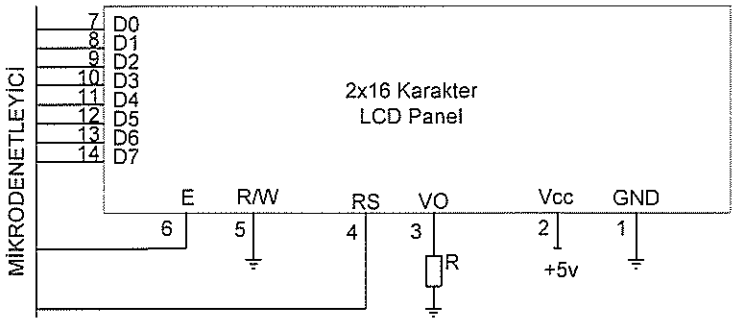
5.1 LCD'nin yapısı

LCD (Liquid Crystal Display) 'ler günlük hayatta sıkça kullanılmaktadır. Burada kullanılan 2x16 (2 satır ve 16 sütunlu) bir LCD ekrandır. Farklı fiyatlar ve ebatlarda ekranlar bulmaktadır. Bu ekran tipinin kullanılmamda, fiyatının uygun olması ve bu proje için ebatlarının yeterli olmasıdır. Standart LCD'ler 8 bit ile programlanır. Ayrıca 5Volt çalışma voltajına sahiptirler. Standart 2x16'lık LCD paneller 14 pinlidir (Şekil 5.2). Bu pinlerden 8 tanesi paralel portun DATA pinlerine bağlanır. Aşağıdaki tabloda LCD ekranın pinleri ve görevleri verilmektedir. 15 ve 16 numaralı ayaklar ise LCD panelinin ekran parlaklığının ayarlanması için kullanılır.[15]

Çizelge 5.1: Ayak pinleri

Pin Numarası	Sembol	İşlev
1	Vss	Besleme ucu (GND)
2	Vdd/Vcc	Besleme ucu (+5V)
3	Vee/Vo	Parlaklık ayarı
4	RS	0=Bilgilendirme girişi/1=Veri girişi
5	R/W	0=LCD' ye yazma /1=LCD'den okuma
6	E	Sinyal erişim
7	DB0	0 nolu bilgi pini
8	DB1	1 nolu bilgi pini
9	DB2	2 nolu bilgi pini
10	DB3	3 nolu bilgi pini
11	DB4	4 nolu bilgi pini
12	DB5	5 nolu bilgi pini
13	DB6	6 nolu bilgi pini
14	DB7	7 nolu bilgi pini

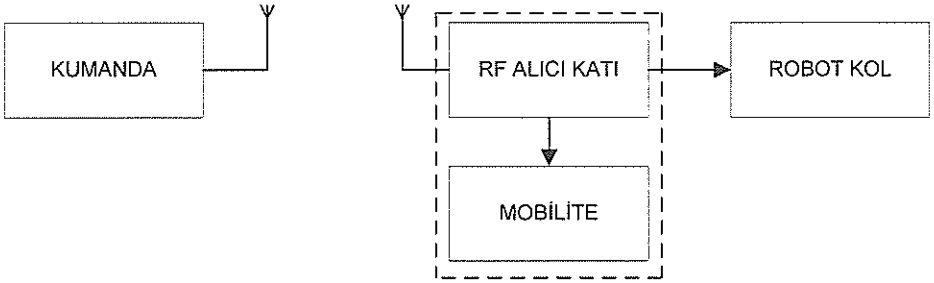
Tabloda açıklanan bağlantılar çok detaylı olmayabilir. Dolayısı ile şekil5.1 de bağlantı şeması ile tekrar belirtilmiştir.



Şekil 5.1 Lcd Bağlantı Şekli

6.UYGULAMA

Bu yüksek lisans tezi çalışmasında, literatürde yer alan çalışmaların ayrıntılı olarak gözden geçirilmesinin ardından özgün olarak geliştirilmiştir. Tez çalışmasının ana blok şeması şekil 6.1 'de görülmektedir.



Şekil 6.1: Çalışma Blok Şeması

Çalışmada kablosuz iletişimi sağlamak amacı ile bir kumanda devresi yapılmıştır. Kumandada ve alıcı devrede radyo frekans modülleri kullanılmıştır. Radyo frekans modüleri, kanundaki ilgili yönetmeliğe uygun olarak,433-434 MHz. ISM bandı ile ilgili bölümüne uygun olarak tasarlanmıştır. Önceki bölümlerde anlatıldığı için tekrar detaylara girilmeyecektir. Kumandadan gönderilen veri RF alıcı katında ayrıştırılarak robot kol ve mobilite katlarına iletilir. Kumandayı kontrol eden kullanıcının istediği hareket sağlanır. Mobilite katı, robot kolunu taşıyan düzeneğin hareketini kontrol etmektedir. Robot kol katı adından da anlaşılacağı üzere robot kolunu kontrol etmektedir. Kumanda katı, rf alıcı katı, mobilite katı ve robot kol katlarının devre şemaları, denetleyici algoritmaları ve denetleyicilerindeki programlar eklerde mevcuttur.[16-18]

7.SONUÇLAR

Bu çalışmada, bir robot koluna mobilite özelliği kazandırılarak 433.920 MHz radyo frekansı bandında kendi tasarladığım kumanda ile uzaktan kontrolü sağlanmıştır. Ayrıca, robot kolunun tutucusunun gimbal aracılığı ile pitch ve roll (yunuslama ve yuvarlanma) ekseninde yere paralel durması sağlanmıştır. RF vericisi, kodlama yöntemi, anten verimi ve kumanda beslemesi gibi durumlar göz önünde bulundurulduğunda, kumanda ve robot kolunun 100 metrelik mesafede sorunsuz iletişim sağlanacaktır. Kullanılan rf band aralığı yasal band aralığı olduğu için, bu frekansta birçok aynı bilgi mevcut olacaktır. Bunlar da iletişimizin sorunlu olmasına neden olacaktır. Bu sorunu çözmek için de iletişimde Manchester Kodlama tekniği ile kodlama yapılmıştır. Bu teknik ile birlikte bilgi gönderimi, 5 kere preamble ardından sencron ve data gönderimi yapılarak daha doğru bir şekilde sağlanmıştır.
Gelen bilgi durumuna göre de mobiliteyi sağlayan motorların hareketi sağlanmıştır. Robot kolu ise, enerji verildiğinde ilk belirlenen durumda bekler. Daha sonra istenilen konumlara gelmektedir.

Bu çalışma, geliştirilebilir niteliktedir. Daha farklı bir kablosuz iletişim modeli ile mesafe aralığı değiştirilebilir. Farklı ara yüzler geliştirilerek robot kolun kontrolü daha eğlenceli bir hal alabilir. Farklı varyasyonları oluşturulabilir.

Bu çalışmada, tasarımın insanları etkileyecek ortamdan uzak tutarak yapılması gerekli olan bir işin yapılmasını sağlamaktı. Çalışma ile sağlanmıştır.

KAYNAKLAR

[1] GÜNDOĞDU K., ÇALHAN A., "İnsansız Askeri Kara Aracı Tasarımı", İleri Teknoloji Bilimleri Dergisi Cilt 2, Sayı 1, 36-45, 2013

[2] ŞAHİN İ., YALVAÇ M., "KABLOSUZ KONTROL EDİLEBİLEN MOBİL ARAŞTIRMA ROBOTU", e-Jornal New World Sciences Academy 2012, Volume 7, Number: 1, Article Number 1C0499, January 2012

[3] CORA A., TAFLAN N., SARAÇ B., YASEMİN Ö., "Telsiz Bomba İmha Robotunun Tasarımı ve Gerçekleştirilmesi ", Eleco 2014 Elektrik, Elektronik, Bilgisayar ve Biyomedikal Mühendisliği Sempozyumu, 27 – 29 Kasım 2014, Bursa

[4] T.C. MEB, MEGEP, 2007, Elektrik Elektronik Teknolojisi Analog ve Sayısal Haberleşme

[5] PASTACI H., 1996, Modern Elektronik Sistemler

[6] KARAKAŞ H., 2005,İleri Pic 16f84 Uygulamaları

[7] "www.udea.com.tr/home_assets/documents/ATX-34S%20KILAVUZ.pdf"

[8] "www.udea.com.tr/home_assets/documents/NRX-34U-E%20KILAVUZ.pdf"

[9] T.C. MEB, MEGEP, 2013, BİLİŞİM TEKNOLOJİLERİ MOTORLAR

[10] TAŞKAYA H.O., " Uzak Bilgisayar Kontrollü Robot Kol", Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Elektronik ve Haberleşme Mühendisliği, İstanbul, 19-25, (2012)

[11] ÇİÇEK S., 2007, CCS C İle Pic Programlama

[12] "http://www.alldatasheet.com/datasheet-pdf/pdf/517744/ETC1/MPU6050.html"

[13] SONGÜL S., "TANK NAMLUSU STABİLİZASYON SİSTEMİNİN ARDUİNO İLE UYGULANMASI VE DENEYSEL DÜZENEĞİNİN HAZIRLANMASI", Yüksek Lisans Tezi, Trakya Üniversitesi Fen Bilimleri Enstitüsü Makine Mühendisliği, Edirne, 23-31, (2014)

[14] ÖZGÜR O., AÇAN E., AKSOY Y., OĞUZTÜZÜN H., "Yazılım Tabanlı MIL-STD-1553B Veri Yolu Simülasyonu Üzerinden Sensör Verilerinin Aktarılması", Eleco 2014 Elektrik, Elektronik, Bilgisayar ve Biyomedikal Mühendisliği Sempozyumu, 27 – 29 Kasım 2014, Bursa

[15] ŞAHİN H., DEDEOĞLU K.S., 2013, Mikroc İle Pic Programlama

[16]"http://www.basecamelectronics.com/files/v3/SimpleBGC_32bit_Connection_Diagram.pdf"

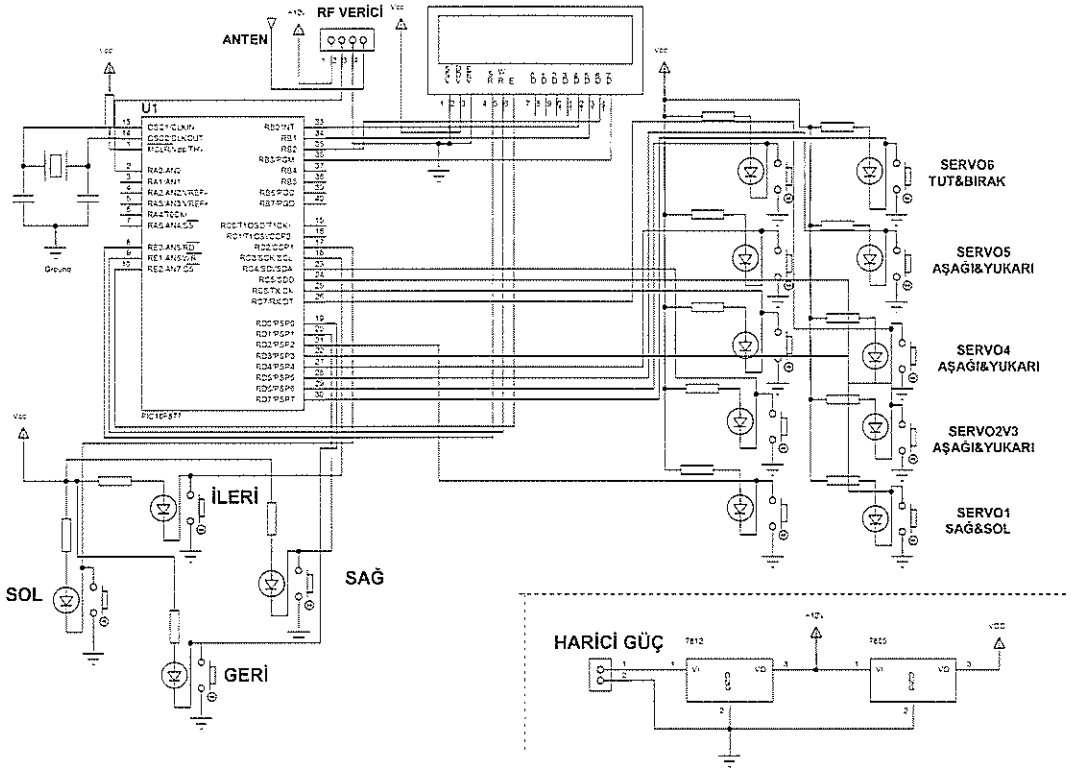
[17] "http://www.altaskitap.com"

[18] "http://www.microchip.com/"

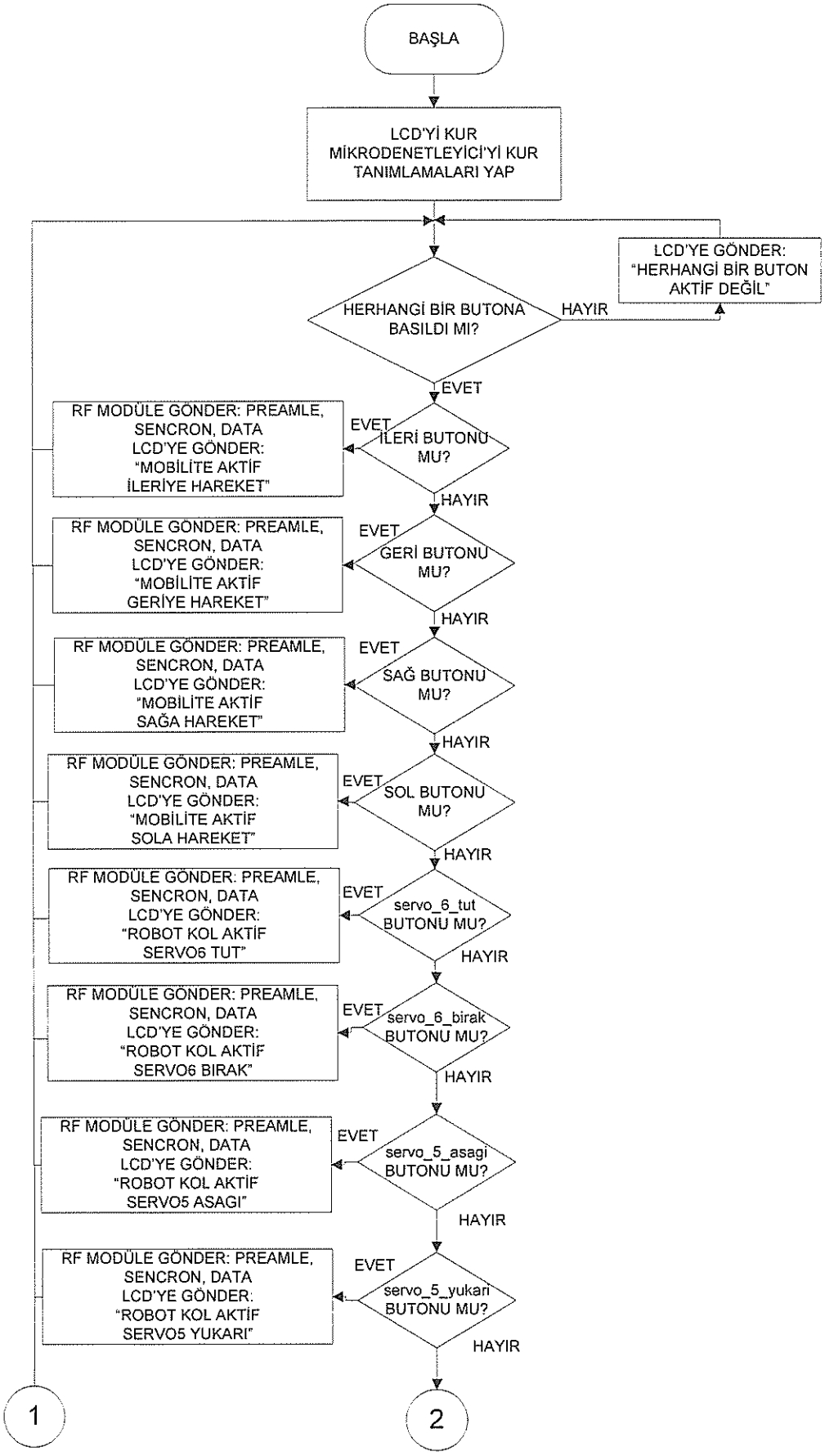
EKLER

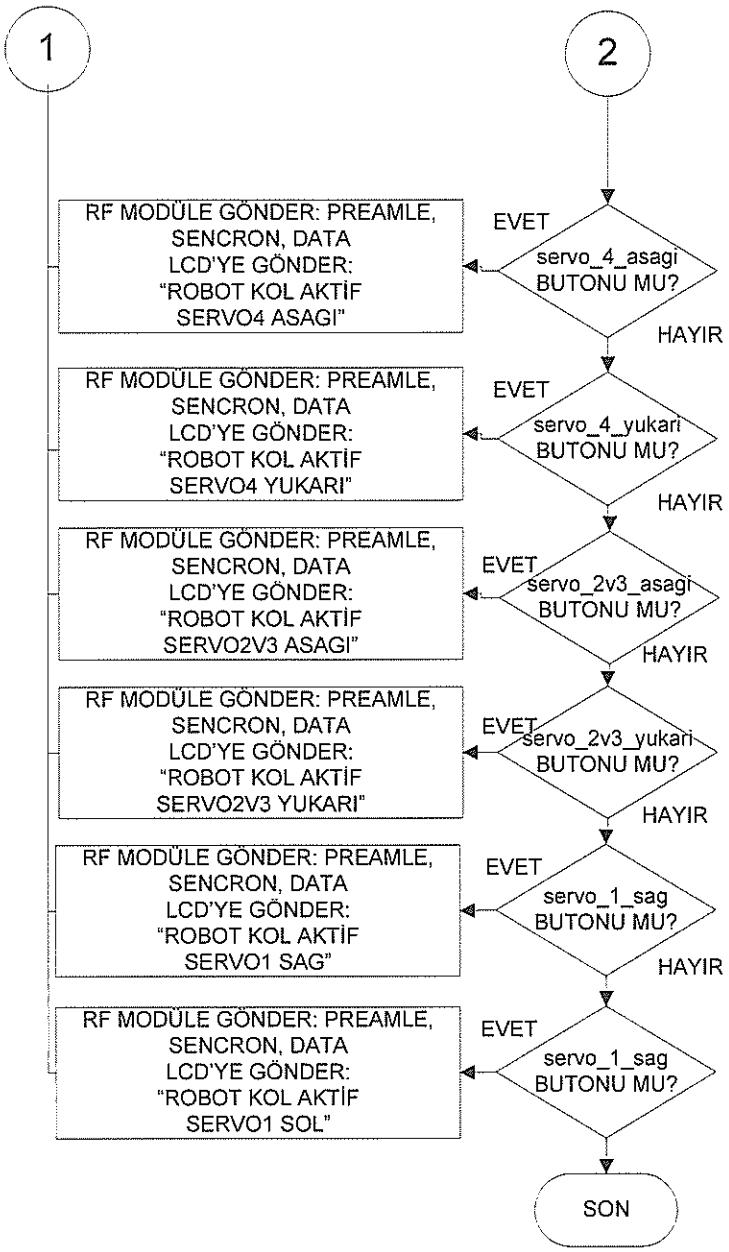
EK A: Kumanda devre şeması
EK B: Kumanda mikrodenetleyici algoritması
EK C: Kumanda mikrodenetleyici programı
EK D: Rf alıcı katı ve mobilite kontrol devre şeması
EK E:Rf alıcı katı ve mobilite kontrol devresi mikrodenetleyici algoritması
EK F:Rf alıcı katı ve mobilite kontrol devresi mikrodenetleyici programı
EK G: Robot kol devre şeması
EK H: Robot kol mikrodenetleyici algoritması
EK I: Robot kol mikrodenetleyici programı

EK A: Kumanda devre şeması



EK B: Kumanda mikrodenetleyici algoritması




1
2
RF MODÜLE GÖNDER: PREAMLE, SENCRON, DATA LCD'YE GÖNDER: "ROBOT KOL AKTİF SERVO4 ASAGI"
EVET
servo_4_asagi BUTONU MU?
HAYIR
RF MODÜLE GÖNDER: PREAMLE, SENCRON, DATA LCD'YE GÖNDER: "ROBOT KOL AKTİF SERVO4 YUKARI"
EVET
servo_4_yukari BUTONU MU?
HAYIR
RF MODÜLE GÖNDER: PREAMLE, SENCRON, DATA LCD'YE GÖNDER: "ROBOT KOL AKTİF SERVO2V3 ASAGI"
EVET
servo_2v3_asagi BUTONU MU?
HAYIR
RF MODÜLE GÖNDER: PREAMLE, SENCRON, DATA LCD'YE GÖNDER: "ROBOT KOL AKTİF SERVO2V3 YUKARI"
EVET
servo_2v3_yukari BUTONU MU?
HAYIR
RF MODÜLE GÖNDER: PREAMLE, SENCRON, DATA LCD'YE GÖNDER: "ROBOT KOL AKTİF SERVO1 SAG"
EVET
servo_1_sag BUTONU MU?
HAYIR
RF MODÜLE GÖNDER: PREAMLE, SENCRON, DATA LCD'YE GÖNDER: "ROBOT KOL AKTİF SERVO1 SOL"
EVET
servo_1_sag BUTONU MU?
SON

EK C: Kumanda mikrodenetleyici programı

```
#include <htc.h>
#include "delay.h"
#include "lcd.h"
__CONFIG( 0x3FFD & 0x3FBF & 0x3FFF & 0x3F7F & 0x3FFB );

#define ileri                RC3
#define geri                 RD0
#define sag                  RD1
#define sol                  RC2
#define servo_6_tut          RD6
#define servo_6_birak        RD7
#define servo_5_asagi        RD4
#define servo_5_yukari       RD5
#define servo_4_asagi        RC6
#define servo_4_yukari       RC7
#define servo_2v3_asagi      RC4
#define servo_2v3_yukari     RC5
#define servo_1_sag          RD2
#define servo_1_sol          RD3
#define cikis                RA0
#define m1                   500
#define m2                   1000
//--------------------data-------------------------------------------------------------
void tx_data(char data)
        {       unsigned char i=8,veri=0;
                        while(i--)
                        {       veri=data&0x80;
                                if (veri==0){
                                                cikis=1;
                                                _delay(m2);
                                                cikis=0;
                                                _delay(m2);}
```

```
                                        else{
                                                cikis=0;
                                                _delay(m2);
                                                cikis=1;
                                                _delay(m2);}
                data=(data<<1);}}
//----------------------sencron-----------------------------------------------------------
void tx_sencron(void){
                unsigned char pr=0x0F;//******sencron sayısı=pr'dir******
                unsigned char bt=8,veri=0;
                        while(bt--)
                        {
                                veri=pr&0x80;
                                if (veri==0)
                                {       cikis=1;
                                        _delay(m2);
                                        cikis=0;
                                        _delay(m2);}
                                        else {          cikis=0;
                                                        _delay(m2);
                                                        cikis=1;
                                                        _delay(m2);}
                                pr=(pr<<1);}}
//----------------------preamble----------------------------------------------------------
void tx_preamble(void){
unsigned char tekrar=5,pr=0x55,_pr;//******preamble sayısı=pr'dir******
                while(tekrar--){
                        _pr=pr;
                        unsigned char bt=8,veri=0;
                        while(bt--){    veri=_pr&0x80;
                                        if (veri==0){   cikis=1;
                                                        _delay(m2);
                                                        cikis=0;
```

```
                                        _delay(m2);}
                        else {          cikis=0;
                                        _delay(m2);
                                        cikis=1;
                                        _delay(m2);}
                _pr=(_pr<<1);}}}
void main()
        {       ADCON1=0x07;
                TRISE=0x00;
                TRISA=0x00;
                TRISB=0x00;
                TRISD=0xFF;
                TRISC=0xFF;
                PORTB=0X00;
                PORTE=0x00;
                PORTA=0x00;
                lcd_init();
                lcd_yaz(" herhangi buton ");
                lcd_gotoxy(2,1);
                lcd_yaz("   aktif degil  ");
        while(1){
                if(ileri==0){   tx_preamble();//ileri butona basıldı
                                tx_sencron();
                                tx_data(0x00);
                                lcd_gotoxy(1,1);
                                lcd_yaz("mobilite aktif ");
                                lcd_gotoxy(2,1);
                                lcd_yaz("ileriye hareket");
                                } else
                if(geri==0){    tx_preamble();//geri butona basıldı
                                tx_sencron();
                                tx_data(0x01);
                                lcd_gotoxy(1,1);
```

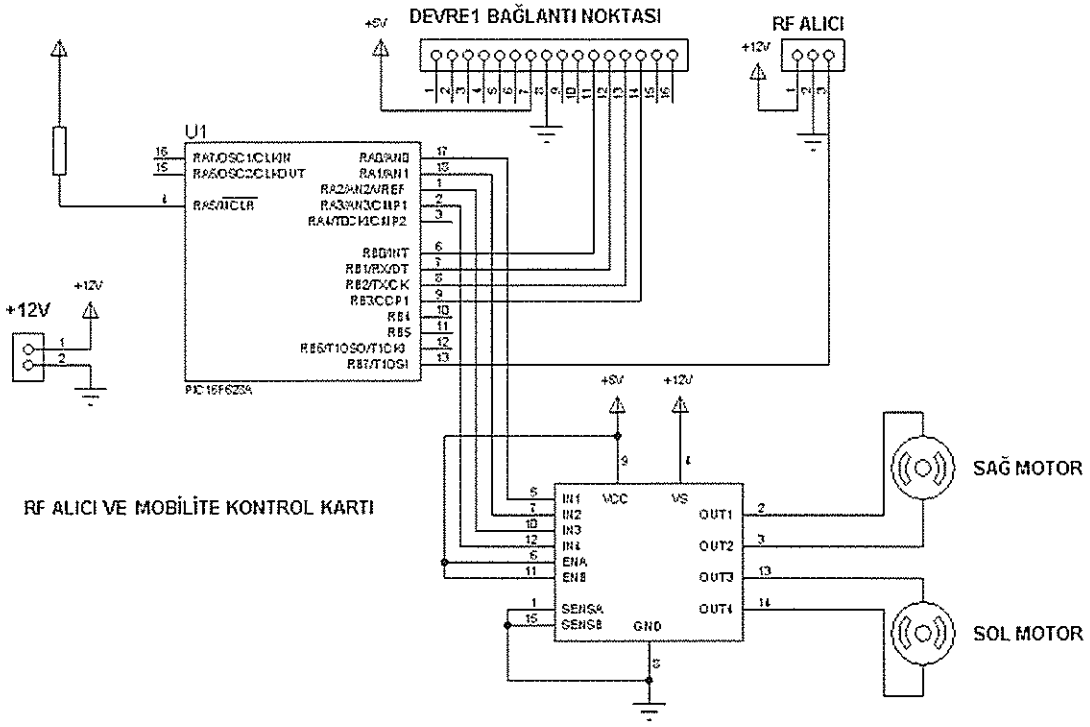
```
                    lcd_yaz("mobilite aktif ");
                    lcd_gotoxy(2,1);
                    lcd_yaz("geriye hareket ");
                    } else
          if(sag==0){ tx_preamble();//sag butona basıldı
                    tx_sencron();
                    tx_data(0x02);
                    lcd_gotoxy(1,1);
                    lcd_yaz("mobilite aktif ");
                    lcd_gotoxy(2,1);
                    lcd_yaz("saga hareket   ");
                    } else
          if(sol==0){ tx_preamble();//sol butona basıldı
                    tx_sencron();
                    tx_data(0x03);
                    lcd_gotoxy(1,1);
                    lcd_yaz("mobilite aktif ");
                    lcd_gotoxy(2,1);
                    lcd_yaz("sola hareket   ");
                    }else
   if(servo_6_tut==0){ tx_preamble();//servo_6_tut butona basıldı
                    tx_sencron();
                    tx_data(0x04);
                    lcd_gotoxy(1,1);
                    lcd_yaz("robotkol aktif ");
                    lcd_gotoxy(2,1);
                    lcd_yaz("servo6 tut     ");
                    } else
if(servo_6_birak==0){ tx_preamble();//servo_6_birak butona basıldı
                    tx_sencron();
                    tx_data(0x05);
                    lcd_gotoxy(1,1);
                    lcd_yaz("robotkol aktif ");
```

```
                    lcd_gotoxy(2,1);
                    lcd_yaz("servo6 birak   ");
                    } else
if(servo_5_asagi==0){ tx_preamble();//servo_5_asagi butona basıldı
                    tx_sencron();
                    tx_data(0x06);
                    lcd_gotoxy(1,1);
                    lcd_yaz("robotkol aktif ");
                    lcd_gotoxy(2,1);
                    lcd_yaz("servo5 asagi   ");
                    } else
if(servo_5_yukari==0){ tx_preamble();//servo_5_yukari butona basıldı
                    tx_sencron();
                    tx_data(0x07);
                    lcd_gotoxy(1,1);
                    lcd_yaz("robotkol aktif ");
                    lcd_gotoxy(2,1);
                    lcd_yaz("servo5 yukari  ");
                    } else
if(servo_4_asagi==0){ tx_preamble();//servo_4_asagi butona basıldı
                    tx_sencron();
                    tx_data(0x08);
                    lcd_gotoxy(1,1);
                    lcd_yaz("robotkol aktif ");
                    lcd_gotoxy(2,1);
                    lcd_yaz("servo4 asagi   ");
                    } else
if(servo_4_yukari==0){ tx_preamble();//servo_4_yukari butona basıldı
                    tx_sencron();
                    tx_data(0x09);
                    lcd_gotoxy(1,1);
                    lcd_yaz("robotkol aktif ");
                    lcd_gotoxy(2,1);
```

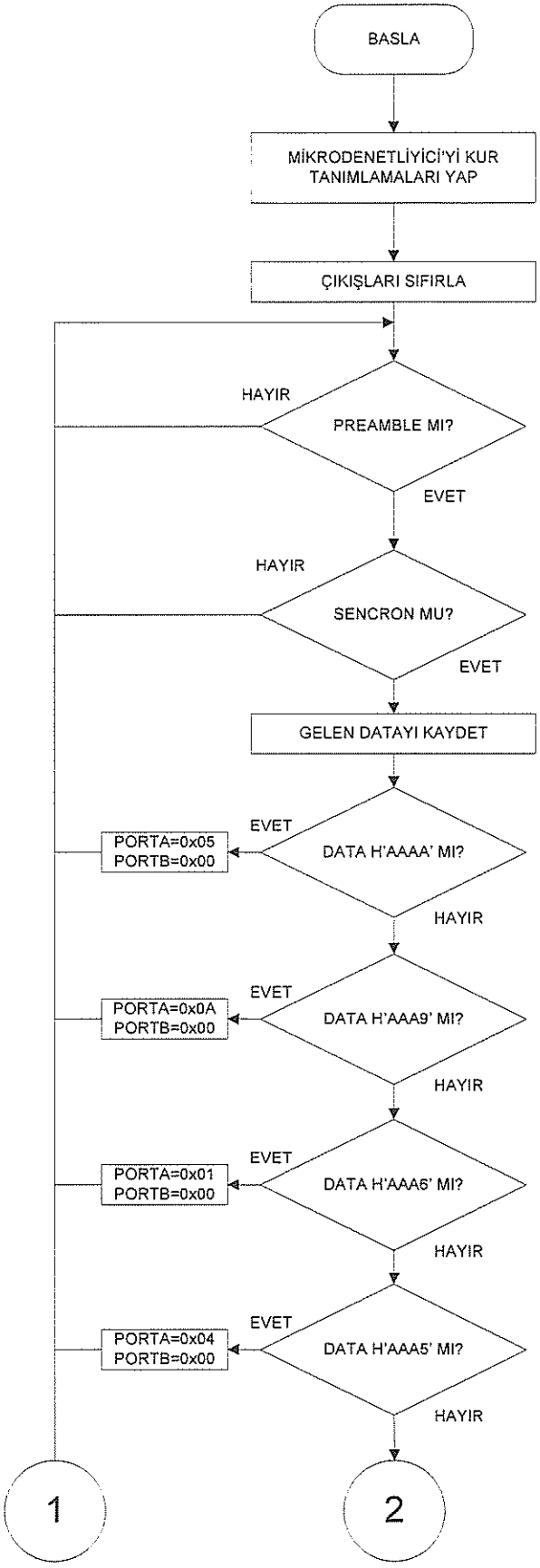
```
                    lcd_yaz("servo4 yukari  ");
                    } else
if(servo_2v3_asagi==0){   tx_preamble();//servo_2v3_asagi butona basıldı
                    tx_sencron();
                    tx_data(0x0A);
                    lcd_gotoxy(1,1);
                    lcd_yaz("robotkol aktif");
                    lcd_gotoxy(2,1);
                    lcd_yaz("servo2v3 asagi");
                    } else
if(servo_2v3_yukari==0){  tx_preamble();//servo_2v3_yukari butona basıldı
                    tx_sencron();
                    tx_data(0x0B);
                    lcd_gotoxy(1,1);
                    lcd_yaz("robotkol aktif ");
                    lcd_gotoxy(2,1);
                    lcd_yaz("servo2v3 yukari");
                    } else
        if(servo_1_sag==0){ tx_preamble();//servo_1_sag basıldı
                    tx_sencron();
                    tx_data(0x0C);
                    lcd_gotoxy(1,1);
                    lcd_yaz("robotkol aktif ");
                    lcd_gotoxy(2,1);
                    lcd_yaz("servo1 sag     ");
                    } else
        if(servo_1_sol==0){ tx_preamble();//servo_1_sol butona basıldı
                    tx_sencron();
                    tx_data(0x0D);
                    lcd_gotoxy(1,1);
                    lcd_yaz("robotkol aktif ");
                    lcd_gotoxy(2,1);
                    lcd_yaz("servo1 sol     ");
```

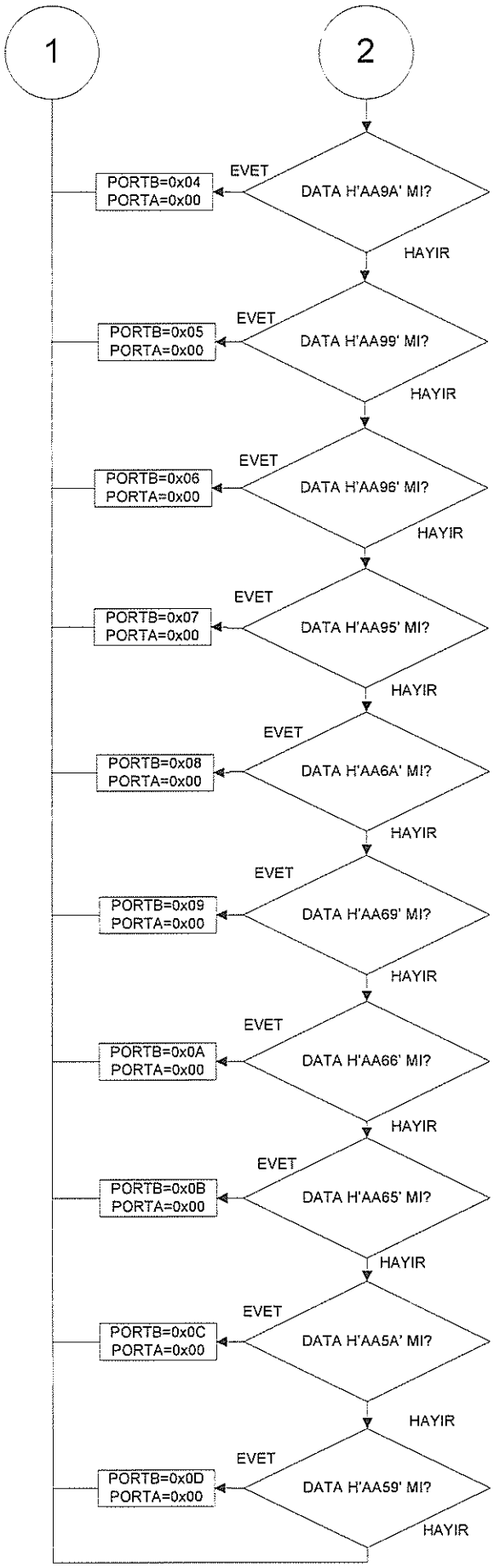
```
} else {
        lcd_gotoxy(1,1);
        lcd_yaz(" herhangi buton ");
        lcd_gotoxy(2,1);
        lcd_yaz("   aktif degil  ");
                        }}}
```

EK D: Rf alıcı katı ve mobilite kontrol devre şeması



EK E: Rf alıcı katı ve mobilite kontrol devresi mikrodenetleyici algoritması




1
2
DATA H'AA9A' MI?
EVET
PORTB=0x04
PORTA=0x00
HAYIR
DATA H'AA99' MI?
EVET
PORTB=0x05
PORTA=0x00
HAYIR
DATA H'AA96' MI?
EVET
PORTB=0x06
PORTA=0x00
HAYIR
DATA H'AA95' MI?
EVET
PORTB=0x07
PORTA=0x00
HAYIR
DATA H'AA6A' MI?
EVET
PORTB=0x08
PORTA=0x00
HAYIR
DATA H'AA69' MI?
EVET
PORTB=0x09
PORTA=0x00
HAYIR
DATA H'AA66' MI?
EVET
PORTB=0x0A
PORTA=0x00
HAYIR
DATA H'AA65' MI?
EVET
PORTB=0x0B
PORTA=0x00
HAYIR
DATA H'AA5A' MI?
EVET
PORTB=0x0C
PORTA=0x00
HAYIR
DATA H'AA59' MI?
EVET
PORTB=0x0D
PORTA=0x00
HAYIR

EK F: Rf alıcı katı ve mobilite kontrol devresi mikrodenetleyici programı

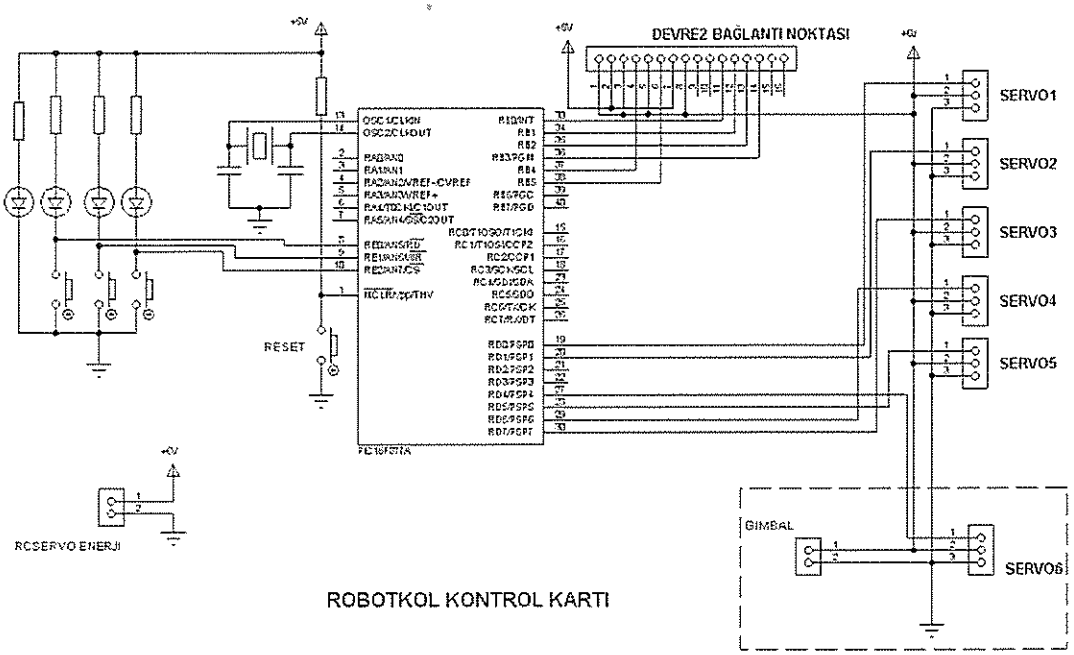
```
#include <pic.h>
__CONFIG(INTIO & WDTDIS & MCLREN & LVPDIS & PWRTDIS & BORDIS
& CPD & UNPROTECT);
        #define m1      500
        #define m2      1000
        #define m3      1500
        #define m4      2000
        #define m6      3000
        #define giris     RB7
        #define cikis1   RA0
        #define cikis2   RA1
        #define cikis3   RA2
        #define cikis4   RA3
unsigned char sayac;
//---------------rx_preamble-----------------------------
unsigned char rx_preamble(void)
    {       unsigned char i=8,veri;
            while(i--){
                    veri=(veri<<1);
                    if (giris) veri=veri|0x01;
                    _delay(m4);}
                return(veri);   }
void main()
    {unsigned int i,a=0,b=0,veri=0,wdt=0;
            TRISB=0xF0;
            TRISA=0x00;
            CMCON=0x07;
            OPTION=0x07;
            PORTA=0x00;
    while(1)
            { i=16;
yy:         wdt=0;
```

```
        if(giris) goto yy;
        _delay(m6);
        if(!giris)goto yy;
        _delay(m6);
        a=rx_preamble();
        if(a==0x55) goto xx;
        if (sayac!=11)          //butona basılmadıysa çıkışlar sıfır...
                {sayac++;
                    if (sayac>10) {
                            PORTA=0x00;
                            PORTB=0x00;
         } }
        goto yy;
xx:                                 //sencron bilgisini kontrol ediyor...
         sayac=0;
while(!giris){ _delay(100);wdt++; if (wdt>20) goto yy;} //burada rb7 yi yakalamaya
çalışırken ya da gelen sinyal kesilirse çıkışta kilitlenmemesini sağlıyor...
        _delay(m3);
        if(giris) goto xx;
        _delay(m2);
        if(!giris) goto xx;
        _delay(m2);
        if(giris) goto xx;
        _delay(m2);
        if(!giris) goto xx;
        _delay(m2);
        if(giris) goto xx;
        _delay(m2);
        if(!giris) goto xx;
        _delay(m2);
        if(giris) goto xx;
        _delay(m2);
        if(giris) goto xx;
```

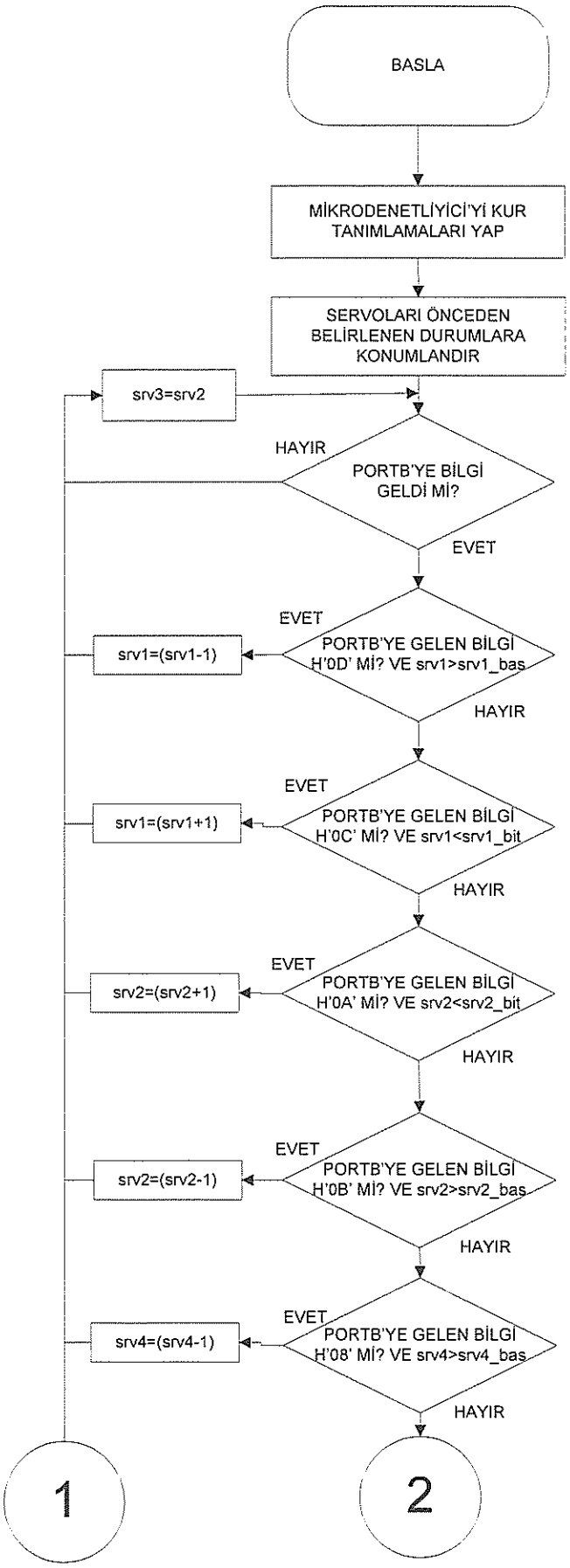
```
        _delay(m2);
        if(!giris) goto xx;
        _delay(m2);
        if(giris) goto xx;
        _delay(m2);
        if(!giris) goto xx;
        _delay(m2);
        if(giris) goto xx;
        _delay(m2);
        if(!giris) goto xx;
        _delay(m2);
        if(giris) goto xx;
        _delay(m2);
        if(!giris) goto xx;
        _delay(m2);

while(i--)        //data yani (veri) okunuyor...
        {       veri=(veri<<1);
                if (giris) veri=veri|0x01;
                _delay(m2);}

if ((veri==0xAAAA)) PORTA=0x05;PORTB=0x00;
                        //veri=h'00'ın manchester kodlu halidir İLERİ
if ((veri==0xAAA9)) PORTA=0x0A;PORTB=0x00;
                        //veri=h'01'ın manchester kodlu halidir GERİ
if ((veri==0xAAA6)) PORTA=0x01;PORTB=0x00;
                        //veri=h'02'ın manchester kodlu halidir SAG
if ((veri==0xAAA5)) PORTA=0x04;PORTB=0x00;
                        //veri=h'03'ın manchester kodlu halidir SOL
if ((veri==0xAA9A)) PORTB=0x04;PORTA=0x00;
                //veri=h'04'ın manchester kodlu halidir SERVO_6_TUT
if ((veri==0xAA99)) PORTB=0x05;PORTA=0x00;
                //veri=h'05'ın manchester kodlu halidir SERVO_6_BIRAK
```

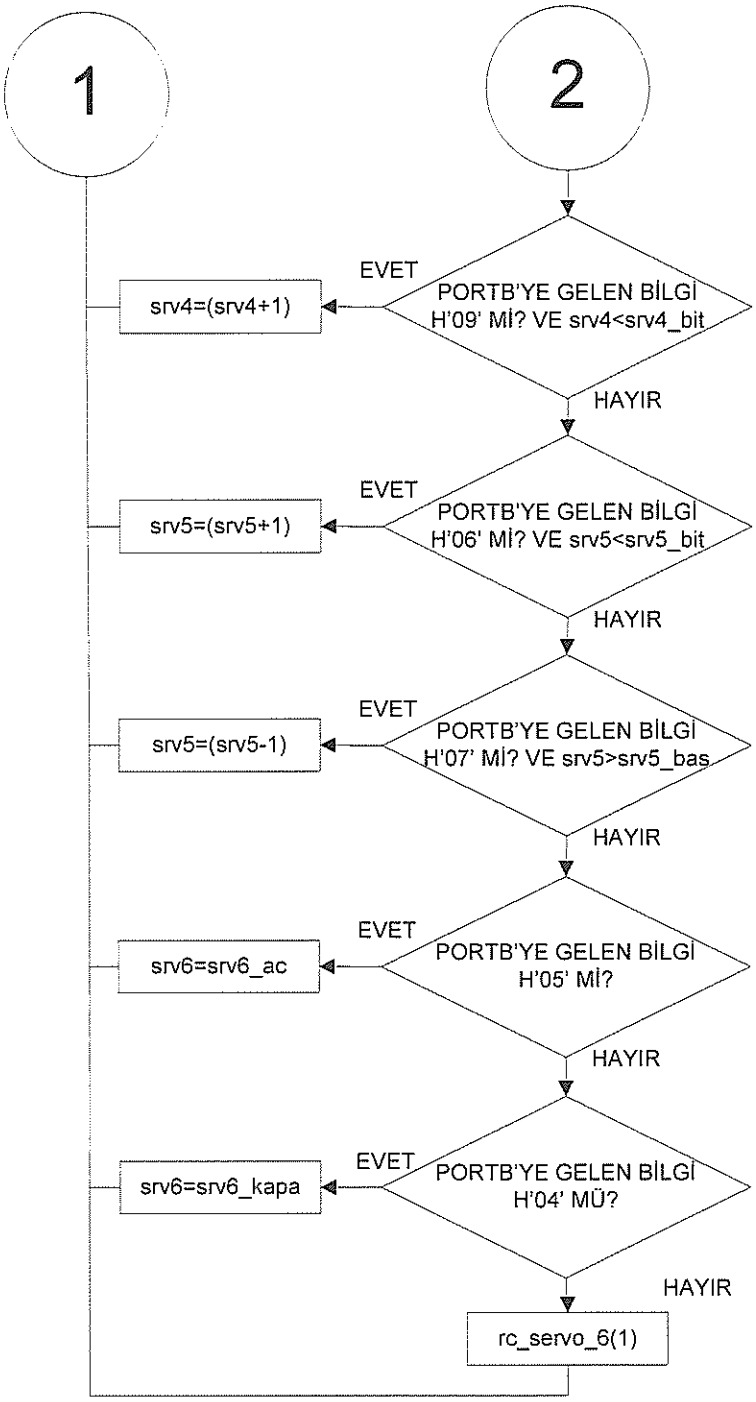
```
if ((veri==0xAA96)) PORTB=0x06;PORTA=0x00;
                //veri=h'06'ın manchester kodlu halidir SERVO_5_ASAGI
if ((veri==0xAA95)) PORTB=0x07;PORTA=0x00;
                //veri=h'07'ın manchester kodlu halidir SERVO_5_YUKARI
if ((veri==0xAA6A)) PORTB=0x08;PORTA=0x00;
                //veri=h'08'ın manchester kodlu halidir SERVO_4_ASAGI
if ((veri==0xAA69)) PORTB=0x09;PORTA=0x00;
                //veri=h'09'ın manchester kodlu halidir SERVO_4_YUKARI
if ((veri==0xAA66)) PORTB=0x0A;PORTA=0x00;
                //veri=h'0A'ın manchester kodlu halidir SERVO_2V3_ASAGI
if ((veri==0xAA65)) PORTB=0x0B;PORTA=0x00;
                //veri=h'0B'ın manchester kodlu halidir SERVO_2V3_YUKARI
if ((veri==0xAA5A)) PORTB=0x0C;PORTA=0x00;
                //veri=h'0C'ın manchester kodlu halidir SERVO_1_SAG
if ((veri==0xAA59)) PORTB=0x0D;PORTA=0x00;
                //veri=h'0D'ın manchester kodlu halidir SERVO_1_SOL
        }}
```

EK G: Robot kol devre şeması



EK H: Robot kol mikrodenetleyici algoritması




1
2
PORTB'YE GELEN BİLGİ H'09' Mİ? VE srv4<srv4_bit
EVET
srv4=(srv4+1)
HAYIR
PORTB'YE GELEN BİLGİ H'06' Mİ? VE srv5<srv5_bit
EVET
srv5=(srv5+1)
HAYIR
PORTB'YE GELEN BİLGİ H'07' Mİ? VE srv5>srv5_bas
EVET
srv5=(srv5-1)
HAYIR
PORTB'YE GELEN BİLGİ H'05' Mİ?
EVET
srv6=srv6_ac
HAYIR
PORTB'YE GELEN BİLGİ H'04' MÜ?
EVET
srv6=srv6_kapa
HAYIR
rc_servo_6(1)

EK I: Robot kol mikrodenetleyici programı

```
#include <htc.h>
#include "delay.h"
__CONFIG( 0x3FFD & 0x3FBF & 0x3FFF & 0x3F7F & 0x3FFB );
char    srv1_bas=50;
char    srv1_nrm=80;
char    srv1_bit=200;           // Robot kol servolarının

char    srv2_bas=80;                    // başlangıç, normal ve
char    srv2_nrm=100;                   // bitiş konumları
char    srv2_bit=250;                   // sabit olarak tanımlanır
char    srv4_bas=40;
char    srv4_nrm=150;
char    srv4_bit=190;
char    srv5_bas=30;
char    srv5_nrm=230;
char    srv5_bit=250;
char    srv6_ac=80;
char    srv6_nrm=45;
char    srv6_kapa=45;

char    srv;
char    srv1;
char    srv2;
char    srv3;
char    srv4;
char    srv5;
char    srv6;
char    srv7;
char    srv8;

#define servo1  RD0
#define servo2  RD1
```

```
        #define servo3  RD7
        #define servo4  RD6
        #define servo5  RD5
        #define servo6  RD4
void rc_servo_6(char x )
        {  // 1 tane RC servo kotrol etmek için
                char i,j,k;
                for (j=1;j<=x;j++){
               servo1=1;
        for(k=0;k<srv1;k++) DelayUs(3);
                servo1=0;
                servo2=1;
        for(k=0;k<srv2;k++) DelayUs(3);
                 servo2=0;
                 servo3=1;
         for(k=0;k<srv3;k++) DelayUs(3);
                  servo3=0;
                  servo4=1;
        for(k=0;k<srv4;k++) DelayUs(3);
                   servo4=0;
                   servo5=1;
        for(k=0;k<srv5;k++) DelayUs(3);
                    servo5=0;
                    servo6=1;
         for(k=0;k<srv6;k++) DelayUs(3);
                     servo6=0;
              DelayMs(10);}}
//-----------------main-----------------------------

void main(void){
        TRISD=0X00;                 //d portu cıkıs
        PORTD=0X00;                 //d portu silinir
        TRISE=0X07;          //e portu giris
```

```
ADCON1=7;          //adc kapalı
TRISB=0X0F;        //b portu cıkıs lcd+rf modul baglantıları var
                   // alt dort bit giris

srv1=srv1_nrm;         // Servo değişkenlerine servoların normal
srv2=srv2_nrm;         // konumlarının tutulduğu sabitlerin
srv3=srv2_nrm;          // değerleri aktarılır
srv4=srv4_nrm;
srv5=srv5_nrm;
srv6=srv6_nrm;
rc_servo_6(1);

while(1){
if ((PORTB==0x0D) && (srv1>srv1_bas)) srv1=(srv1-1); // 1.servo sol
if ((PORTB==0x0C) && (srv1<srv1_bit)) srv1=(srv1+1); // 1.servo sağ
if ((PORTB==0x0A) && (srv2<srv2_bit)) srv2=(srv2+1);
            // 2.servo ve 3.servo yukarı
if ((PORTB==0x0B) && (srv2>srv2_bas)) srv2=(srv2-1);
            // 2.servo ve 3.servo aşağı
if ((PORTB==0x08) && (srv4>srv4_bas)) srv4=(srv4-1); // 4.servo yukarı
if ((PORTB==0x09) && (srv4<srv4_bit)) srv4=(srv4+1); // 4.servo aşağı
if ((PORTB==0x06) && (srv5<srv5_bit)) srv5=(srv5+1); // 5.servo yukarı
if ((PORTB==0x07) && (srv5>srv5_bas)) srv5=(srv5-1); // 5.servo aşağı
if (PORTB==0x05) srv6=srv6_ac;              // 6.servo açık
if (PORTB==0x04) srv6=srv6_kapa;             // 6.servo kapalı

  srv3=srv2;
  rc_servo_6(1);
}}
```

ÖZGEŞMİŞ

Adı-Soyadı: Gökhan EŞGİ

Doğum Tarihi ve Yeri: 09.11.1989 – KADIKÖY

e-posta: gokhanesgi@gmail.com

Öğrenim Durumu:

Lisans: Gazi Üniversitesi, Teknik Eğitim Fakültesi, Elektronik Öğretmenliği

MESLEKİ DENEYİMLER:

(2012-Halen Devam Ediyor)-Türk Havayolları Teknik A.Ş. Hat Bakım Başkanlığı A Bakım Müdürlüğü (Avionik ve Mekanik Uçak Teknisyeni)

(2014-2015 Güz Dönemi)- İstanbul Aydın Üniversitesi Anadolu Bil Myo Uçak Teknolojisi Bölümü (Misafir Öğr. Gör.)

MAKALE ÇALIŞMALARI:

Project PISCES: Developing an In-Flight Entertainment System for Smart Devices (Springer Plus – Submit Edilmiş ve Beklemede)