

**KARADENİZ TEKNİK ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI**

**SÜREKLİ MIKNATISLI TÜPSEL YAPILI DOĞRUSAL HAREKETLİ  
DOĞRU AKIM MOTORUNUN YAPAY SİNİR AĞLARIYLA DENETİMİ**

**YÜKSEK LİSANS TEZİ**

**Ezgi GÜNEY**

**MART 2015**

**TRABZON**

**KARADENİZ TEKNİK ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**

**ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI**

**SÜREKLİ MIKNATISLI TÜPSEL YAPILI DOĞRUSAL HAREKETLİ DOĞRU  
AKIM MOTORUNUN YAPAY SINIR AĞLARIYLA DENETİMİ**

**Ezgi GÜNEY**

**Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsünde  
“YÜKSEK LİSANS (ELEKTRİK MÜHENDİSLİĞİ)”  
Unvanı Verilmesi İçin Kabul Edilen Tezdir.**

**Tezin Enstitüye Verildiği Tarih : 05.03.2015**  
**Tezin Savunma Tarihi : 27.03.2015**

**Tez Danışmanı: Prof. Dr. Adem Sefa AKPINAR**

**Trabzon 2015**

**Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü**  
**Elektrik-Elektronik Mühendisliği Anabilim Dalında**  
**Ezgi GÜNEY Tarafından Hazırlanan**

**SÜREKLİ MIKNATISLI TÜPSEL YAPILI DOĞRUSAL HAREKETLİ DOĞRU  
AKIM MOTORUNUN YAPAY SİNİR AĞLARIYLA DENETİMİ**

**başlıklı bu çalışma, Enstitü Yönetim Kurulunun 10/03/2015 gün ve 1593 sayılı  
kararıyla oluşturulan jüri tarafından yapılan sınavda**

**YÜKSEK LİSANS TEZİ**  
**olarak kabul edilmiştir.**

**Jüri Üyeleri**

**Başkan : Prof. Dr. Adem Sefa AKPINAR** .....

**Üye : Prof. Dr. Cemal KÖSE** .....

**Üye : Doç. Dr. Halil İbrahim OKUMUŞ** .....

**Prof. Dr. Sadettin KORKMAZ**  
**Enstitü Müdürü**

## ÖNSÖZ

Bu tez, Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü Elektrik-Elektronik Mühendisliği Anabilim Dalı, Elektrik Mühendisliği Yüksek Lisans Programı'nda yapılan bir çalışmadır. “Sürekli Mıknatıslı Tüpsel Yapılı Doğrusal Hareketli Doğru Akım Motorunun Yapay Sinir Ağlarıyla Denetimi” konulu çalışmada, Doğrusal Hareketli Doğru Akım Motorların yapısı, kullanım yerleri, türleri hakkında bilgi verilmiştir. Sürekli Mıknatıslı Tüpsel Yapılı Doğrusal Hareketli Doğru Akım Motor tasarımı yapılmış, gerçekleştirilen model üzerinde deneyler yapılarak sürekli mıknatıslı rotorun denetimini sağlayacak kontrol sistemi tasarlanmıştır. Sürekli Mıknatıslı Tüpsel Yapılı Doğrusal Hareketli Doğru Akım Motorunun kontrolünde kullanılan Yapay Sinir Ağları yöntemi anlatılmış, gerçekleştirilen sistemin yapay sinir ağlarıyla denetimi sağlanmıştır. Yüksek lisans tezi danışmanlığımı üstlenerek, gerek konu seçimi ve gerekse çalışmaların yürütülmesi sırasında yardımlarını esirgemeyen hocam Prof. Dr. Adem Sefa AKPINAR' a en içten teşekkürlerimi sunuyorum. Katkılarından dolayı Doç. Dr. İsmail Hakkı ALTAŞ, Doç. Dr. Halil İbrahim OKUMUŞ ve Prof Dr. Cemal KÖSE' ye teşekkürü borç bilirim. Çalışmam boyunca yardımlarını esirgemeyen Öğr. Gör. Memnun DEMİR ve Öğr. Gör. Kurtuluş DEMİR arkadaşlarıma da en içten teşekkürlerimi sunarım.

Ezgi GÜNEY  
Trabzon 2015

## TEZ BEYANNAMESİ

Yüksek Lisans Tezi olarak sunduğum “Sürekli Mıknatıslı Tüpsel Yapılı Doğrusal Hareketli Doğru Akım Motorunun Yapay Sinir Ağlarıyla Denetimi” başlıklı bu çalışmayı baştan sona kadar danışmanım Prof. Dr. Sefa AKPINAR‘ın sorumluluğunda tamamladığımı, verileri kendim topladığımı, analizleri ilgili programlarda yaptığımı, başka kaynaklardan aldığım bilgileri metinde ve kaynakçada eksiksiz olarak gösterdiğimi, çalışma sürecinde bilimsel araştırma ve etik kurallara uygun olarak davrandığımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiğimi beyan ederim.27.03.2015

Ezgi GÜNEY

## İÇİNDEKİLER

	<u>Sayfa No</u>
ÖNSÖZ.....	III
TEZ BEYANNAMESİ.....	IV
İÇİNDEKİLER.....	V
ÖZET .....	VIII
SUMMARY .....	IX
ŞEKİLLER DİZİNİ.....	IX
TABLolar DİZİNİ.....	XII
KISALTMALAR DİZİNİ .....	XIII
1. GENEL BİLGİLER .....	1
1.1. Giriş.....	1
1.2. Doğrusal Hareketli Motorların Tarihçesi .....	1
1.3. Doğrusal Hareketli Motorların Yapısal Gelişimi.....	2
1.4. Doğrusal Hareketli Motorların Üstünlükleri.....	4
1.5. Doğrusal Hareketli Motorların Uygulama Alanları .....	4
1.6. Doğrusal Hareketli Motorların Sınıflandırılması.....	7
1.7. Doğrusal Hareketli Motorlar Üzerine Yapılmış Çalışmalar .....	9
2. SÜREKLİ MIKNATISLI TÜPSEL YAPILI DOĞRUSAL HAREKETLİ DOĞRU AKIM MOTORLARI (SMTYDHDAM) .....	12
2.1. SMTYDAM'ın Yapıları.....	12
2.2. SMTYDAM Özelliklerinin Karşılaştırılması.....	15
2.3. SMTYDAM' ın Üstünlükleri.....	17
2.4. SMTYDAM'ın Diğer Doğrusal Motorlarla Karşılaştırılması .....	17
2.5. SMTYDAM' ın Alan Dağılımı.....	18
2.6. Sürekli Miknatıslar.....	19
2.7. Sürekli Miknatıslı Motorların Üstünlükleri .....	21
2.8. Sürekli Miknatıslı Motorların Olumsuzlukları .....	21
3. YAPAY SİNİR AĞLARI .....	22
3.1. YSA'nın Tanımı.....	22
3.2. YSA'nın Tarihi Gelişimi.....	22

3.3.	YSA'nın Yapısı.....	23
3.3.1.	Biyolojik Sinir Hücresinin Yapısı.....	24
3.3.2.	Yapay Sinir Hücresinin Yapısı .....	26
3.3.2.1.	Girdiler( $x_n$ ).....	27
3.3.2.2.	Ağırlıklar( $w_n$ ).....	27
3.3.2.3.	Birleştirme Fonksiyonu( $\Sigma$ ) .....	27
3.3.2.4.	Aktivasyon Fonksiyonu(f) .....	27
3.3.2.4.1.	Doğrusal Aktivasyon Fonksiyonu.....	28
3.3.2.4.2.	Eşik Aktivasyon Fonksiyonu .....	28
3.3.2.4.3.	Sigmoid Aktivasyon Fonksiyonu.....	29
3.3.2.4.4.	Tanjant Sigmoid Aktivasyon Fonksiyonu.....	30
3.3.2.5.	Çıktılar .....	30
3.3.3.	Yapay Sinir Ağının Yapısı.....	31
3.3.3.1.	Giriş Katmanı.....	31
3.3.3.2.	Ara Katman (Gizli Katman).....	31
3.3.3.3.	Çıkış Katmanı .....	32
3.4.	YSA'nın Sınıflandırılması .....	32
3.4.1.	Yapılarına Göre YSA Çeşitleri .....	33
3.4.1.1.	İleri Beslemeli Ağlar .....	33
3.4.1.2.	Geri Beslemeli Ağlar .....	34
3.4.2.	Öğrenme Algoritmalarına Göre YSA Çeşitleri.....	34
3.4.2.1.	Danışmanlı Öğrenme .....	34
3.4.2.2.	Danışmansız Öğrenme .....	35
3.4.2.3.	Destekleyici Öğrenme.....	35
3.4.3.	Kullanılan Veri Türüne Göre YSA Çeşitleri.....	35
3.4.3.1.	Niteliksel Veri Kullanan Ağlar .....	36
3.4.3.2.	Niceliksel Veri Kullanan Ağlar.....	36
3.4.4.	Öğrenme Zamanına Göre YSA Çeşitleri .....	36
3.4.4.1.	Statik Öğrenme .....	36
3.4.4.2.	Dinamik Öğrenme.....	36
3.5.	YSA' da Öğrenme Kuralları .....	37
3.5.1.	Çevrimiçi Öğrenme Kuralları .....	37
3.5.2.	Çevrimdışı Öğrenme Kuralları.....	37

3.5.2.1.	Hebb Kuralı.....	37
3.5.2.2.	Backpropagation Net (Delta kuralı).....	39
3.5.2.3.	Hopfield Kuralı .....	42
3.5.2.4.	Khonen Kuralı.....	45
3.6.	Yapay Sinir Ağlarının Üstünlükleri .....	46
3.7.	Yapay Sinir Ağlarının Olumsuzlukları .....	48
3.8.	YSA' nın Kullanıldığı Alanlar .....	48
4.	SMTYDHDAM'IN GERÇEKLENMESİ VE YSA İLE DENETİMİ.....	50
4.1.	Mekanik Kısımın Gerçeklenmesi.....	50
4.1.1.	Stator Tasarımı.....	50
4.1.2.	Rotor Tasarımı .....	52
4.2.	Kontrol Kısımının Gerçeklenmesi .....	54
4.2.1.	Kontrol Kartı Ünitesi .....	55
4.2.2.	Mikrodenetleyici Ünitesi.....	56
4.2.3.	Sensör (Algılayıcı) Ünitesi.....	57
4.3.	Kapalı Çevrim YSA Kontrolünün Gerçeklenmesi.....	59
4.3.1.	Mikrodenetleyici Ünitesinde Kullanılan Denetleyicinin Yazılımı .....	60
4.3.2.	YSA ile Modelleme .....	64
4.3.3.	MATLAB Ortamında YSA Denetiminin Programlanması .....	67
5.	SONUÇ .....	73
6.	ÖNERİLER.....	75
7.	KAYNAKLAR .....	76
8.	EKLER.....	79

ÖZGÖEÇMİŞ



Yüksek Lisans Tezi

ÖZET

SÜREKLİ MIKNATISLI TÜPSEL YAPILI DOĞRUSAL HAREKETLİ  
DOĞRU AKIM MOTORUNUN YAPAY SİNİR AĞLARIYLA DENETİMİ

Ezgi GÜNEY

Karadeniz Teknik Üniversitesi  
Fen Bilimleri Enstitüsü  
Elektrik-Elektronik Mühendisliği Anabilim Dalı  
Danışman: Prof. Dr. Adem Sefa AKPINAR  
2015, 78 Sayfa, 12 Sayfa Ek

Bu çalışmada, endüstride daha çok kısa mesafeli hareket veya mekanik darbe uygulama alanına sahip olan SMTYDHDAM'ın yapay sinir ağlarıyla motor hız denetimini sağlamak amaçlandı.

Özgün olarak geliştirilen SMTYDHDAM'ın denetimi amacıyla elektronik kontrol sistemi oluşturuldu. Oluşturulan bu elektronik kontrol sistemi, evrensel seri veri yolu(USB) aracılığıyla, gerçek zamanlı çalışan MATLAB yazılımı ile yönetildi. Denetim, klasik yöntemlere bir alternatif olarak geliştirilen, karmaşık denetim algoritmalarına gerek duymayan yapay sinir ağları ile oluşturuldu.

Geliştirilen YSA denetleyici yapısı öncelikle bilgisayar ortamında test edildi, eğitim sonuçlarından ağıın öğrenme kabiliyetinde olduğu kabul edilerek geliştirilen sisteme uygulandı. Kullanılan algoritmalar ve devre düzeneğinin gerek rotor gerek frekans kontrolünde başarılı olunmasına yardımcı olduğu görüldü.

Geliştirilen sisteme ait devre şemaları, denetim algoritmaları ve elde edilen grafikler tezde sunulmuştur.

**Anahtar Kelimeler:** Doğrusal Hareketli Doğru Akım Motoru, Yapay Sinir Ağı, Sürekli Miknatıs, Gerçek Zamanlı Öğrenme

Master Thesis

SUMMARY

ARTIFICIAL NEURAL NETWORK CONTROL FOR PERMANENT-MAGNET  
TUBULAR LINEAR DIRECT CURRENT MOTOR

Ezgi GÜNEY

Karadeniz Technical University  
The Graduate School of Natural and Applied Sciences  
Elektric-Electronic Engineering Graduate Program  
Supervisor: Prof. Adem Sefa AKPINAR  
2015, 78 Pages, 12 Pages Appendix

In this thesis study, provide the Permanent-Magnet Tubular Linear Direct Current Motor's (PMTLDCM) speed control is aimed with an Artificial Neural Network Control.

An electronic control system was created for the PMTLDCM's control. Created this electronic control system was governed by real-time working MATLAB software, through universal serial bus (USB). Control, which was developed as an alternative to conventional methods, was created with artificial neural networks that do not require complex control algorithms them.

The developed ANN controller structure was tested primarily on computer, assuming that the training results in improved learning ability of the network was applied to the system. It seemed helping to be successful in the algorithms and circuit setting that is used in both rotor and frequency.

This control sistem's circuit diagrams, control algorithms, and graphs is presented in the thesis.

**Key Words:** Linear DC Motor, Artificial Neural Network, Permanent Magnet, Realtime Learning

## ŞEKİLLER DİZİNİ

### Sayfa No

Şekil 1.1.	Doğrusal hareket elde etme yöntemlerinin karşılaştırılması .....	3
Şekil 1.2.	Döner tip motordan doğrusal hareketli motora geçiş .....	3
Şekil 1.3.	Çift yanlı motorun elde edilmesi .....	3
Şekil 1.4.	Tüpsel yapıli motorlara geçiş.....	4
Şekil 1.5.	Sebze paketleme sistemi .....	5
Şekil 1.6.	Sıralama sistemleri (optik renk ayırma ve sıralama) .....	6
Şekil 1.7.	Gıda kesme-dilimleme işlemleri (Ekmek kesme makinesi) .....	6
Şekil 1.8.	Şişe dolum makinesi .....	7
Şekil 1.9.	Doğrusal hareketli motorların sınıflandırılması.....	8
Şekil 1.10.	Bir yapay kalp pompası kesiti .....	9
Şekil 1.11.	Nükleer reaktör kontrol motoru .....	10
Şekil 1.12.	x-y çizici motoru.....	10
Şekil 1.13.	Doğrusal hareketli motor akı geçişleri.....	10
Şekil 1.14.	Sürekli mıknatıslı doğrusal hareketli adım motor kesiti.....	11
Şekil 2.1.	Hava çekirdekli aksiyal akılı 3 fazlı bir tubular motorun 1 faz sargısı.....	12
Şekil 2.2.	Primeri ve sekonderi demir çekirdekli bir tubular motor .....	12
Şekil 2.3.	Tüpsel yapıli doğrusal motor .....	13
Şekil 2.4.	Döner tip makinadan doğrusal makinaya geçiş .....	14
Şekil 2.5.	Tüpsel yapıli makinanın elde edilişii .....	14
Şekil 2.6.	Tüp şeklinde doğrusal motorlar .....	15
Şekil 2.7.	Tüpsel yapıli motorların çeşitli eksenlerdeki hareketleri.....	16
Şekil.2.8.	Dahili ve harici mıknatıslar .....	18
Şekil 2.9.	Harici mıknatıslar .....	18
Şekil 2.10.	Eksenel mıknatısların topolojisi .....	19
Şekil 2.11.	İçsel mıknatısların topolojisi.....	19
Şekil 2.12.	Neodmium mıknatıs çeşitleri.....	20
Şekil 3.1.	Biyolojik sinir hücresi yapay sinir hücresinin yapısı.....	25
Şekil 3.2.	Yapay sinir hücresinin yapısı.....	26
Şekil 3.3.	Doğrusal aktivasyon fonksiyonunu .....	28

Şekil 3.4. (a) Esik aktivasyon fonksiyonunu b) Signum aktivasyon fonksiyonu .....	29
Şekil 3.5. Logaritma sigmoid aktivasyon fonksiyonu .....	29
Şekil 3.6. Tanjant sigmoid aktivasyon fonksiyonu .....	30
Şekil 3.7. Yapay sinir ağı modeli.....	32
Şekil 3.8. İleri beslemeli ağ yapısı .....	33
Şekil 3.9. Geri beslemeli ağ yapısı.....	34
Şekil 3.10. Hebb'in mimarisi .....	38
Şekil 3.11. Standart 3 katmanlı (x girdi, z saklı, y çıktı) Backpropagation ağı yapısı.....	40
Şekil 3.12. Hopfield net modeli .....	43
Şekil 3.13. Khonen' in 2 katmanlı (girdi, çıktı) SOM ileri beslemeli ağ yapısı .....	44
Şekil 4.1. Tüpsel yapılı doğrusal hareketli motorun stator gövdesinin yapısı ve ölçüleri .....	51
Şekil 4.2. Bobin bağlantı şeması.....	51
Şekil 4.3. Tüpsel doğrusal hareketli doğru akım motorunun rotor ölçüleri.....	52
Şekil 4.4. a) tek mıknatısın manyetik alan dağılımı b) birleştirilmiş mıknatısların manyetik alan dağılımı .....	53
Şekil 4.5. Gerçeklenen SMTYDHDAM sürücü devre şeması .....	54
Şekil 4.6. Gerçekleştirilen SMTYDHDAM sürücü sistem fotoğrafı.....	55
Şekil 4.7. Kontrol kartı ünitesi devre şeması .....	55
Şekil 4.8. PWM sürücü ünitesi devre şeması.....	57
Şekil 4.9. Sensör ünitesinin devre şeması.....	58
Şekil 4.10. Mikrodenetleyici "PIC_Setting (düzenleme)" algoritması.....	60
Şekil 4.11. Mikroişlemci "Interrupt (kesme)" algoritması .....	61
Şekil 4.12. a) Yön=0 durumunda bobin enerjilenme sırası, b) Yön=1 durumunda bobin enerjilenme sırası .....	61
Şekil 4.13. Mikroişlemci "motoradım" düzenleme algoritması .....	62
Şekil 4.14. Çok katmanlı ileri beslemeli-geri yayımlı YSA modeli.....	64
Şekil 4.15. Geri yayılım algoritmasının ağı uygulanması .....	66
Şekil 4.16. Geliştirilen YSA' nın test sinyali ile eğitimine ait grafik.....	68
Şekil 4.17. İstenen_frekans = 1.19, nu = 4 için elde edilen çıkış-hata-frekans grafiği.....	69
Şekil 4.18. İstenen_frekans = 1.19, nu = 24 için elde edilen çıkış-hata-frekans grafiği.....	70
Şekil 4.19. İstenen_frekans = 1.19, nu = 48 için elde edilen çıkış-hata-frekans grafiği.....	70
Şekil 4.20. İstenen_frekans = 1.08, nu = 4 için elde edilen çıkış-hata-frekans grafiği.....	71
Şekil 4.21. İstenen_frekans = 1.24, nu = 4 için elde edilen çıkış-hata-frekans grafiği.....	72

## TABLolar DİZİNİ

	<b><u>Sayfa No</u></b>
Tablo 1.1. DHM'lerin kullanım alanları .....	7
Tablo 1.2. İlave soğutma olmaksızın sürekli çalışma durumunda DHM'lerin nominal güçleri .....	9
Tablo 2.1. SMTYDHDAM motorların özellikleri .....	15
Tablo 4.1. Bobinlerin enerjilenme sırası kural dizilimi .....	59

## KISALTMALAR DİZİNİ

ÇKP	Çok Katlı Perseptron
DAM	Doğru Akım Motoru
DC	Doğru Akım
DHM	Doğrusal Hareketli Motor
DHSMDA	Doğrusal Hareketli Sürekli Mıknatıslı Doğru Akım Motoru
GZHGYYSA	Gerçek Zamanlı Hatayı Geri Yayımlı Yapay Sinir Ağı
LCD	Sıvı Kristal Ekranı
PWM	Darbe Genlik Modülasyonu
SMTYDHDAM	Sürekli Mıknatıslı Tüpsel Yapılı Doğrusal Hareketli Doğru Akım Motoru
TYSMDAM	Tüpsel Yapılı Sürekli Mıknatıslı Doğru Akım Motoru
USB	Evrensel Seri Veri Yolu
YSA	Yapay Sinir Ağı

## **1. GENEL BİLGİLER**

### **1.1. Giriş**

Modern teknolojik dünyada her geçen gün doğrusal hareketli motorlar gelişmekte ve ekstra çözümler ilave edilmektedir bu gelişimlere. Ayrıca diğer doğrusal hareket sistemleri olan hidrolik ve diğer sistemler her geçen gün bu motorların kullanımını artırmaktadır [23]. Kullanım alanı olarak örnek vermek gerekirse, hidrolik altyapılı bu tip motorlar ağır iş makineleri ve bezer sistemlerde kullanılırlar. Hidroliğin tercih edilme amacı ise motor gücüyle hidrolik basıncının kolay bir şekilde elde edilebilmesidir. Doğrusal hareket elde etmenin en uygun yolu doğrusal hareketli motorlardır. Hiç şüphe yok ki, doğrusal hareketli motorlar diğer sistemlere karşı daha avantajlıdır ve bu yüzden birçok kullanıcı tarafından alternatif olarak düşünülüp kullanılmaktadır. Bunlara ek olarak, doğrusal hareketli motorların tercih edilmesini sağlayan iki ana sebep vardır. Bunlardan ilki, maliyetin eskiye göre çok daha uygun olması ve manyetik malzemelerin işlenmesi, diğeri ise kontrol ve güç elektroniği alanındaki gelişmelerde rahatlıkla uygulanabilir olmasıdır.

Diğer taraftan doğrusal hareketli makinelerde kullanılmasına sıcak bakılmayan sistemler sürekli arızaya neden olurlar ve bu arıza sonucunda da bakım ihtiyaçları oluşur.

Yukarıda da belirtildiği gibi DHM'lerin diğer sistemlere göre oldukça fazla üstünlükleri vardır. Bunlar:

- Bağlantı aşamasında sadece kablo yeterlidir.
- Daha az karmaşıktır özellikle de hidrolik sistemlere karşı.

Bütün bu bilgilerin eşliğinde karar verilen bu tezin amacı, Sürekli Mıknatıslı Tüpsel Yapılı Doğrusal Hareketli Doğru Akım Motor (SMTYDHDAM) tasarımı yapmak ve bu motorun yapay sinir ağlarıyla denetimini sağlamaktır.

### **1.2. Doğrusal Hareketli Motorların Tarihçesi**

Doğrusal hareketli motorların çok eski bir tarihi bulunmaktadır. Birçok kaynakta bu tarih net olarak bilinmemekte olup ilk doğrusal hareketli motor 1917 yılında imal edilmiştir. Birkeland'ın bombardıman silahı olarak bilinen bu motor fırlatıcı olarak

tasarlanmış tüpsel yapılı relüktans motordur [2]. Başlangıç zamanından sonra yaklaşık 50-60 yıl bu motorlar çeşitli sebeplerden dolayı uygulamada kullanılamamışlardır. Bilim adamları bu 50 yıllık zaman zarfında çeşitli araştırmalar yaparak DHM'lerin gelişimine katkı sağlamışlardır [17].

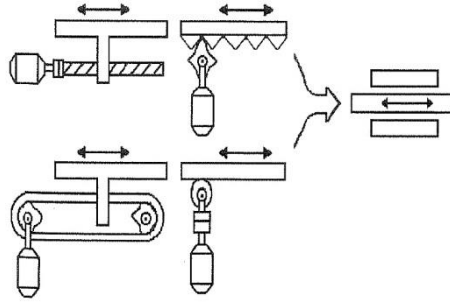
Teknolojik gelişmelerle birlikte DHM'lerin topolojisi tam olmamakla birlikte son 40 yıldır bilinmektedir. Bu sebepten dolayı genel olarak çalışmalar DHM'ler üzerinde yoğunlaştırılmıştır. Trenlerde, yürüyen merdivenlerde, yazıcılarda, asansörlerde, fotokopi makinalarında, fabrikalardaki tüm ürün işleme, ayırma ve paketleme işlemlerinde kısacası hayatımızın neredeyse her alanında artık DHM'ler kullanılmaktadır.

### **1.3. Doğrusal Hareketli Motorların Yapısal Gelişimi**

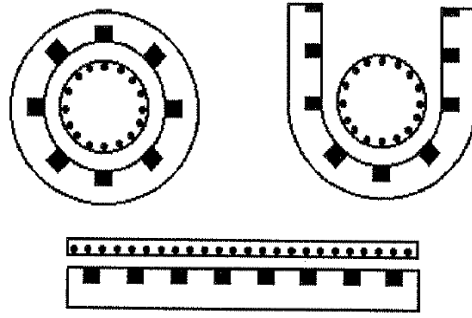
Elektrik motorlarının büyük çoğunluğu döner hareket elde etmek için tasarlanmıştır. Doğrusal hareket, dönel motorlarda dişli mekanizma düzeneği ile sağlanırken, doğrusal motorlarda elektromanyetizma gücünü kullanarak direk elde edilmektedir. Doğrusal motorlar uyartım türlerine göre alternatif akım ve doğru akım motorları olmak üzere ikiye ayrılır. Bu iki tip motorun da farklı avantajları vardır. Örneğin AA Doğrusal Hareketli motorlar uzun mesafeler için ideal kullanıma sahipken Doğrusal Hareketli DA motorlar ise kısa mesafelerde üstün çalışma performansına sahiptir [2].

Aşağıdaki Şekil 1.1.'de doğrusal hareket elde etme yöntemlerinden birkaçı görülmektedir. Şeklin sol kısmında döner tip motor kullanılmıştır. Bu yöntemin karmaşıklığı ve buna bağlı olarak arıza oluşturma riski şekilden kolaylıkla anlaşılabilir. Ancak şeklin sağ kısmında gösterilen DHM' nin kullanımı ile oluşan bütün aksaklıklar ortadan kalkar. Bu tip motorlarda direk hiçbir ara eleman kullanılmadan doğrusal hareket elde edilebilir.



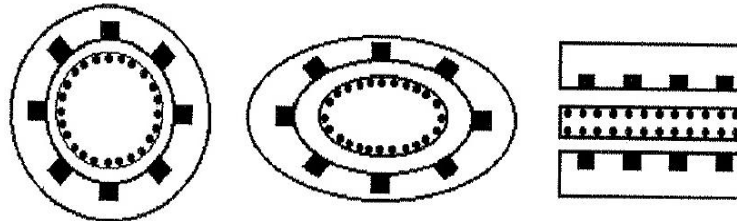


Şekil 1.1. Doğrusal hareket elde etme yöntemlerinin karşılaştırılması [17]

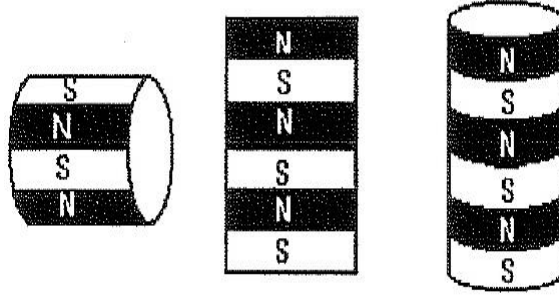


Şekil 1.2. Döner tip motordan doğrusal hareketli motora geçiş [18]

DHM'lerin yapısı döner bir makinanın kesilip açılmasıyla rahatlıkla açıklanabilir. Bunun açıklaması Şekil 1.2.' de görülmektedir. Başka yöntemlerle de DHM elde etmek mümkündür. Mesela Şekil 1.3.' deki gibi makinanın statorunun iki parçaya bölünmesi ve araya rotorun yerleştirilmesi ile yeni bir tasarım elde edilir. Bu tip motorlara Çift Yanlı Motor denir. Bu şekilde elde edilen motorlara Tüpsel Yapılı Motorlar denir. Döner tip makinadan bu yapının elde edilme aşamaları Şekil 1.4.' da gösterilmektedir.



Şekil 1.3. Çift yanlı motorun elde edilmesi [18]



Şekil 1.4. Tüpsel yapılı motorlara geçiş [18]

#### 1.4. Doğrusal Hareketli Motorların Üstünlükleri

DHM'ler maliyetleri bakımından ucuzdur ve kullanıldıkları sistemlerde verimi artırır. DHM'lerin üstünlükleri aşağıda belirtilmiştir [43]. Verimin yüksek olmasındaki ana neden mekanik bağlantılar yoktur.

- Yüksek tepki hızı mekanik iletimli sistemlerin tepki hızından daha fazladır.
- Mekanik iletim olmadığından motorda boşluk oluşmaz.
- Besleme gerilimi ve kontrol devresine bağlı olarak yüksek hıza sahiptir.
- Yüksek hassasiyete sahiptir.
- Yapı olarak basit olduğundan az bakım gerektirir.
- Uzun ömürlüdür.
- Sistemde bazı noktalarda işlemin tekrar gerçekleşmesi durumunda sistemden daha olumlu sonuçlar alınır.

DHM'ler bazı olumsuzluklara sahiptir. Bu olumsuzluklar aşağıda belirtilmiştir.

- Sıcaklıkla motor performansı ters orantılıdır.
- Motor ile kontrol elaman arasında bir ara yüze ihtiyaç vardır.
- Manyetik alan demiri çeker.

#### 1.5. Doğrusal Hareketli Motorların Uygulama Alanları

DHM'ler düz veya tüpsel yapıları olabilirler. Günümüzde DHM'lerin uygulama alanları hakkında birçok çalışma yapılmıştır. Bu çalışmaların sonucunda döner hareketli motorların doğrusal hareketli motorlara göre daha verimli olduğu anlaşılmıştır. DHM'ler ile ilgili yapılmış bazı çalışmalar aşağıda belirtilmiştir.

- İplik sarmada sürücü olarak,
- Kuyu açmada çekik olarak,
- Hızlı trenlerde sürücü olarak,
- Krenlerin doğrusal hareketinde,
- Kapı açma ve kapamada,
- Taşıma bantlarının sürülmesinde,
- Paketleme sistemlerinde,
- Düzenleme ve sıralama işlemlerinde,
- Şişe dolum işlemlerinde,
- Gıda kesme-dilimleme işlemlerinde kullanılır.

Yukarıda belirtilen uygulama alanları DHM'ler ile ilgili yapılmış çalışmalardan sadece bazılarıdır. Aşağıdaki şekillerde DHM'lerle oluşturulmuş bazı sistemler gösterilmiştir.



Şekil 1.5. Sebze paketleme sistemi [43]



Şekil 1.6. Sıralama sistemleri (optik renk ayırma ve sıralama) [43]



Şekil 1.7. Gıda Kesme-Dilimleme İşlemleri (Ekmek kesme makinesi) [43]



Şekil 1.8. Şişe dolum makinesi [43]

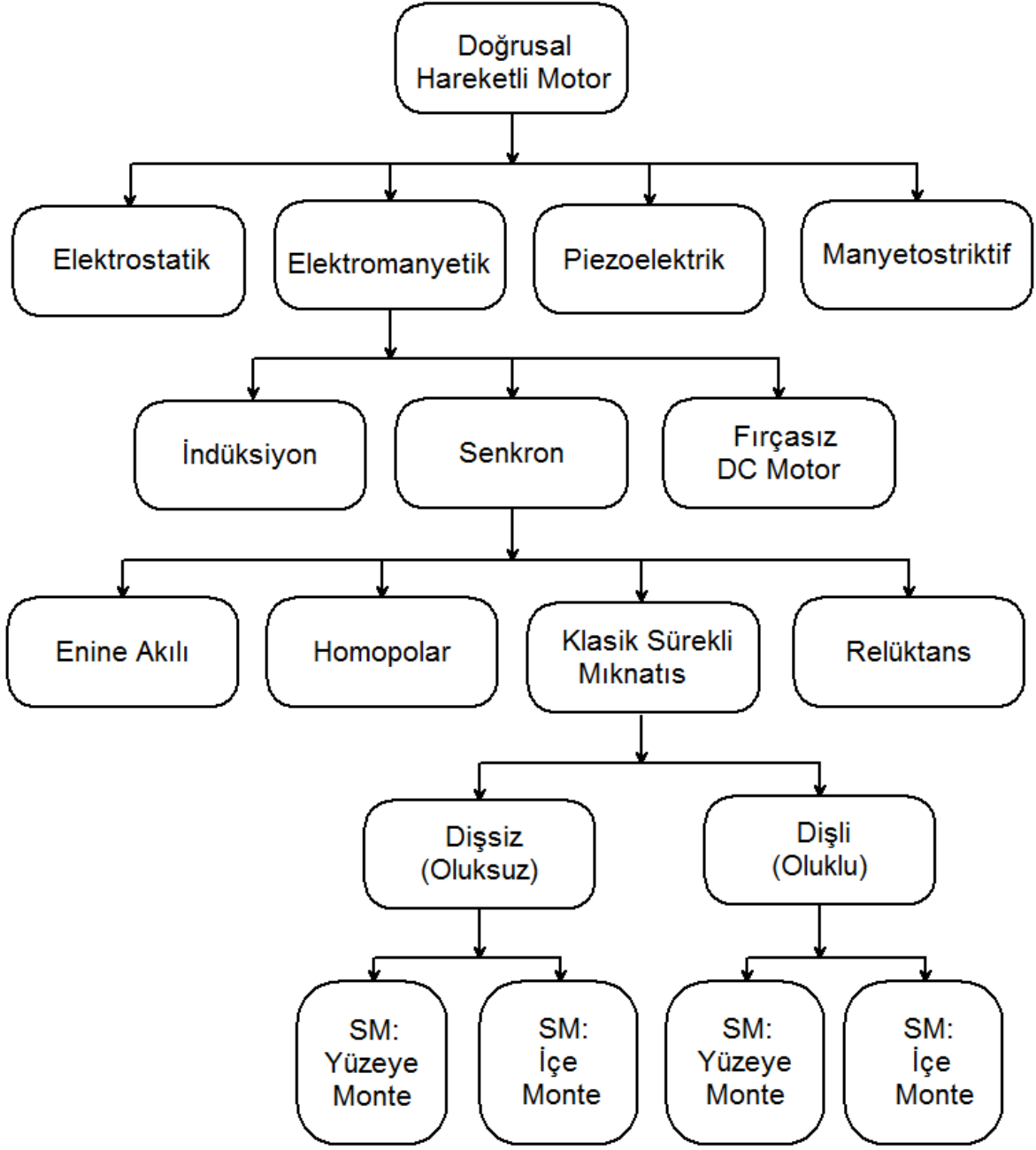
Genel olarak DHM'lerin kullanım alanlarına ait bilgi Tablo 1.' de verilmiştir.

Tablo 1.1. DHM'lerin kullanım alanları [26]

Kayan Kapılar	Asansörlerde	Depo Otomasyon
Krenlerde Sürücü Sistem Olarak	Sahne, Perde Hareketleri	Karıştırma Sürücü Olarak
Hurda Taşımada	Bagaj Taşımada	Döner Kapılarda
Doğrusal Hızlandırıcılarda	Yolcu Taşımacılığında	Esnek İmalat Sistemlerinde
Gemi Test Tanklarında	Hedef Hareketlerinde	Vagonların Alt Düzeneklerinde
Çelik Tüp Hareketlerinde	Metal Levha Hareketlerinde	Sarım İşlemlerinde
Hidrolik Silindirlerin Yerine	Robot Sistemlerinde	Konveyör Sistemlerinde
Çip ve Yonga Yarıiletken İmalatında	Pnömatik Silindirlerin Yerine	Hacimsel Pompalarda
Medikal Ekipmanlarda	Parça Kesme İşlemlerinde	Paketleme Makinelerinde
Eğlence Sektöründe	Hassa Öğütücülerde	Biçimlendirme İşlemlerinde
Kalıpçılıkta	Havalandırma Sistemlerinin Kontrolünde	Hassas Vana Kontrolünde
Simülasyonlarda	Test Düzeneklerinde	Parça Kesme İşlemlerinde

## 1.6. Doğrusal Hareketli Motorların Sınıflandırılması

DHM'lerin sınıflandırılması Şekil 1.9.'da gösterilmiştir. Şekilden de anlaşılacağı gibi en yaygın olan motor sınıfı elektromanyetiktir.



Şekil 1.9. Doğrusal hareketli motorların sınıflandırılması [6]

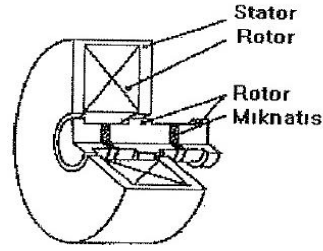
Doğrusal hareketli motorların güç üretmeleri büyüklüklerine, yapılarına ve çalışma sistemlerine bağlıdır. Aşağıdaki Tablo 1.2.'de DHM'lerin nominal güçleri gösterilmiştir.

Tablo 1.2. İlave soğutma olmaksızın sürekli çalışma durumunda DHM'lerin nominal güçleri [26]

Doğrusal Hareketli İndüksiyon Motoru	1-2 N/cm <sup>2</sup>
Oluklu Sürekli Mıknatıslı Senkron DHM	6 N/cm <sup>2</sup> 'ye kadar
Oluksuz Sürekli Mıknatıslı Senkron DHM	3 N/cm <sup>2</sup> 'ye kadar
Doğrusal Hareketli Relüktans Motor	1.5 N/cm <sup>2</sup>
Enine Akıllı DHM	3 N/cm <sup>2</sup>
Doğrusal Hareketli Piezoelektrik Motor	Motor Yapısına Göre
Doğrusal Hareketli Manyetosriktif	Motor Yapısına Göre
Doğrusal Hareketli Elektrostatik Motor	<sup>16</sup> /cm <sup>2</sup>

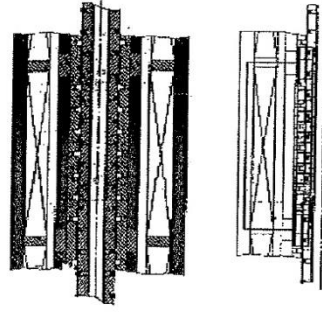
### 1.7. Doğrusal Hareketli Motorlar Üzerine Yapılmış Çalışmalar

DHM'ler kullanımları sayesinde sistemin verimi ve emniyeti artmaktadır. Özellikle yüksek güçlerdeki ve küçük boyuttaki tasarımlar sağlık sektöründe gelişme göstermiştir. Örneğin, yapay kalp pompası olarak kullanılmaktadır. Bu tasarım farklı sürekli mıknatıs kullanılarak elde edilmiştir. Şekil.1.10.'da bu tasarım gösterilmiştir. Bu tasarımın en belirgin özelliği ise az parçaya sahip olmasıdır ve bu yüzden de basit ve güvenilirdir [5].

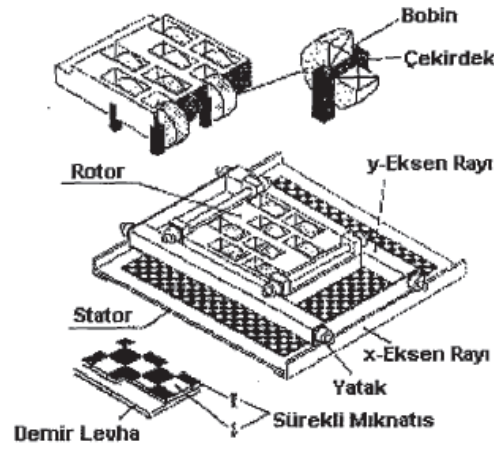


Şekil 1.10. Bir yapay kalp pompası kesiti [1]

DHM'ler alışılmış kullanım alanları dışında yüksek verim istenen yerlerde de gün geçtikçe yaygın olarak kullanılmaya başlamıştır. Bir model üzerine nükleer reaktörde nötron yutan malzemeden yapılmış olan kontrol çubuğunu denetlemek için geliştirilmiş bir doğrusal hareket motor yapısı Şekil. 1.11.'de gösterilmektedir [10]. x-y koordinat ekseninde çizim yapmak için tasarlanmış düzlemsel bir doğrusal hareketli motor yapısı ise Şekil 1.12.'dedir [39].

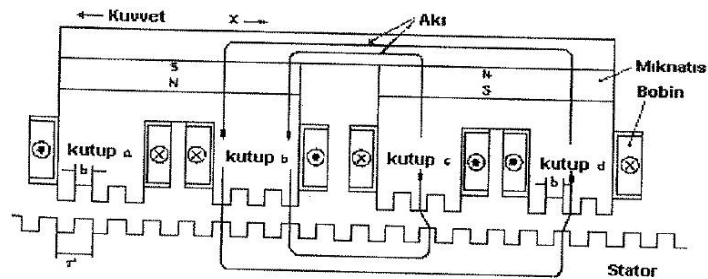


Şekil 1.11. Nükleer reaktör kontrol motoru [10]



Şekil 1.12. x-y çizici motoru [39]

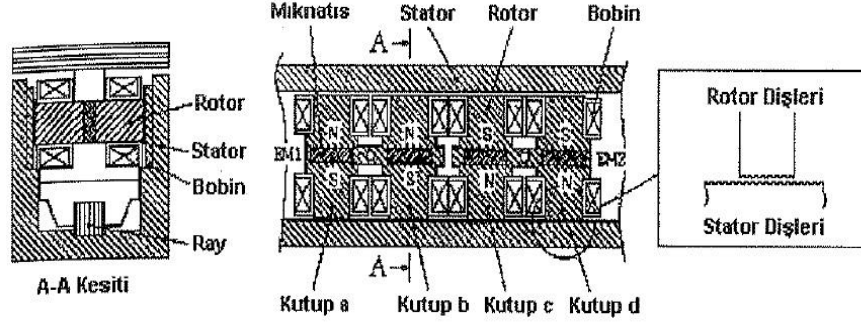
DHM'ler genelde kendi türleri arasında karşılaştırılırlar. Bu karşılaştırmanın ana nedeni döner yapılı motorların daha verimli olduğu bilinmektedir çünkü aynı malzemeler kullanılarak yaklaşık aynı boyutlarda motorlar elde edilir. Oluşturulan bu tip motorlarda da döner yapılı motorlar daha faydalıdır. Bahsedilen bu şekildeki doğrusal hareketli motor çalışması Şekil 1.13.'de görülmektedir [39]. Karşılaştırma yapılırken etkin manyetik akı, akı yoğunluğu, elektromotor kuvvet ve kuvvet arasında bir ilişki türetilir.



Şekil 1.13. Doğrusal hareketli motor akı geçişleri [38]



DHM'lerde alışılmıřın aksine karřılařtırmalar sadece tür bakımından deęil aynı zamanda kullanılan yöntemler bakımından da yapılmaktadır. Sürekli analiz ile sonlu elemanlar yöntemimin kıyaslandıęı bir DHM'nin yapısı Şekil 1.14.'de verilmiřtir.



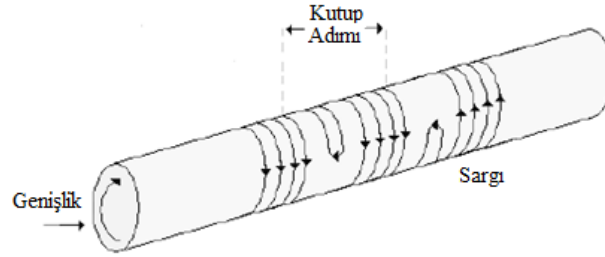
Şekil 1.14. Sürekli mıknatıslı doğrusal hareketli adım motor kesiti [39]

Doęrusal hareketli makinalar hakkında verilmiř olunan bu temel bilgilerden sonra alıřma konum olan tüpsel yapı sürekli mıknatıslı doğrusal akım makinalarına geilecektir.

## 2. SÜREKLİ MIKNATISLI TÜPSEL YAPILI DOĞRUSAL HAREKETLİ DOĞRU AKIM MOTORU (SMTYDHDAM)

### 2.1. SMTYDHDAM'nun Yapıları

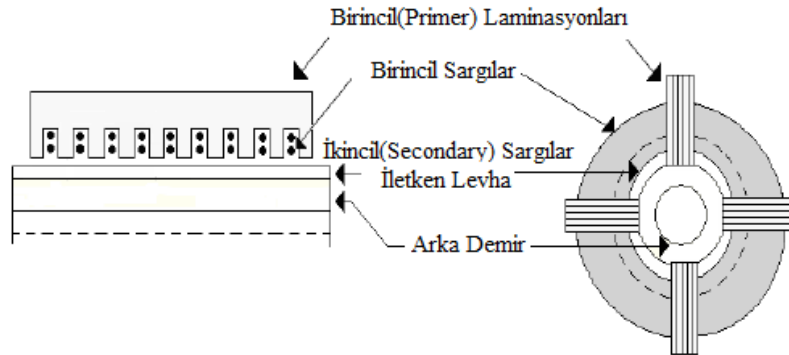
Tüpsel yapılı motorlar, DHM'ler içerisinde en kolay imal edilendir. Bazı uygulamalarda primer demir çekirdek bile içermez. Tüpsel yapılı motorlar mekanik nedenlerden dolayı kısa mesafeli hareket veya mekanik darbe(vuruş) için kullanılırlar. Şekil 2.1.' de hava çekirdekli aksiyal akılı tüpsel yapılı bir DAM'nin üç faz sargısından biri gösterilmektedir.



Şekil 2.1. Hava çekirdekli aksiyal akılı 3 fazlı bir tubular motorun 1 faz sargısı [45]

Primerin hava çekirdekli olması durumunda sekonder mutlaka demir çekirdekli olmalıdır. Demir çekirdek üzerine Al veya Cu'dan yapılmış bir gömlek giydirilerek kuvvet artırılabilir.

Primeri demir çekirdekli bir Tubular motorun radyal ve aksiyal kesiti Şekil 2.2.' de gösterilmektedir.



Şekil 2.2. Primeri ve sekonderi demir çekirdekli bir tubular motor [1]

DHSMDAM'da ya rotor hareket eden stator duran kısımdır ya da rotor duran stator hareket eden kısımdır. Bu durum ihtiyaç durumuna göre değişir [1]. Yani ya rotor dış kısımda stator iç kısımdadır ya da stator dış kısımda rotor iç kısımdadır. Bu çalışmada rotor hareket eden kısım olarak tasarlanmıştır.

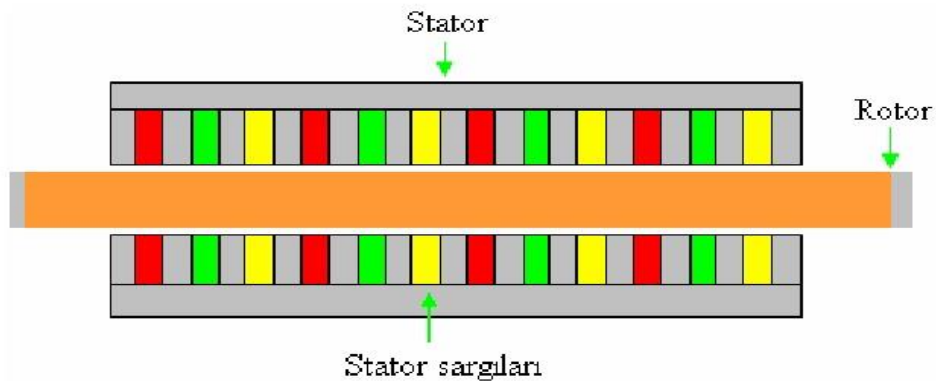
Bilindiği üzere motorlardaki uyarma sürekli mıknatıslarla sağlanır. Uyarma ihtiyacına gereksinim duymadan manyetik alan üreten malzemelerdir sürekli mıknatıslar. Modern dünyada en yaygın olanı Ferrit mıknatıslar olup kullanım alanı oldukça geniştir. Bunun birçok sebebi vardır.

- Ferrit mıknatıslar sert ve kırılmandır ayrıca hammaddeleri oldukça ucuzdur.
- İşletme sıcaklıkları yaklaşık  $250^{\circ}\text{C}$ 'dir.
- Ferrit mıknatısların en önemli özelliği ise paslanmazlar ve düzgün yüzeyleri vardır.

Genel olarak sürekli mıknatısların dirençleri(manyetik) büyüktür fakat manyetik geçirgenlikleri küçüktür. Sürekli mıknatısların bu özellikleri sayesinde kaçak akılar rahatlıkla azaltılabilir fakat mıknatısların yerleşimleri ve boyutları büyük önem taşır bu makinelerde.

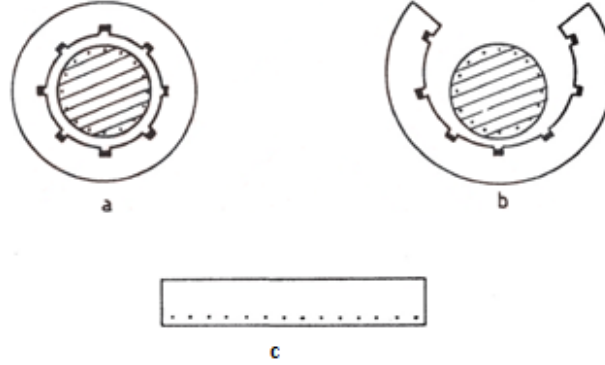
Tüpsel yapıya doğrusal motorların statoru silindriktir. Bu tip motorların sargıları yuvarlak (disk) şekilde bobinlerden oluşur. Kullanımları çok kolay ve yapısal olarak oldukça basittir [17]. Rotor olarak çelik mıknatıs çubuk kullanılabilir. Bir ve üç fazlı olarak tasarlanabilir. Bu çalışmada rotor olarak alüminyum gövde içerisine neodmium mıknatıs yerleştirilmesiyle elde edilen gövde kullanılmıştır.

Tüpsel yapıya doğrusal motorlar üstünlük bakımından diğer doğrusal motorların sahip olduğu üstünlüklerin hepsine sahiptir.



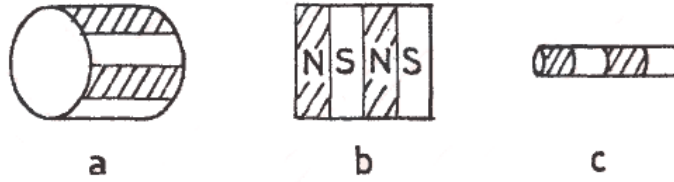
Şekil 2.3. Tüpsel yapıya doğrusal motor [17]

Döner makinanın radyal düzlem boyunca kesilip açılmasıyla doğrusal makinanın elde edilişi Şekil 2.4.'de göstermektedir.



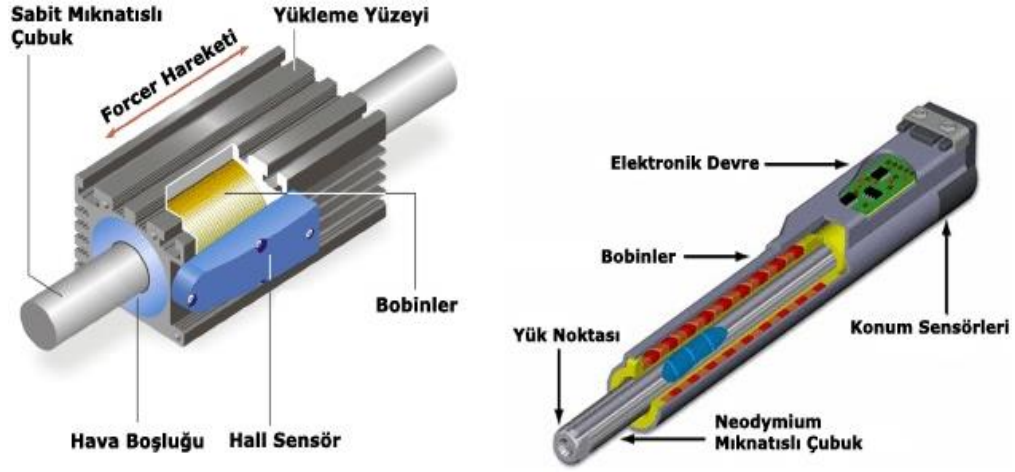
Şekil 2.4. Döner tip makinadan doğrusal makineye geçiş [17]

Tüpsel yapılı makinanın yapılışı doğrusal hareketli makinanın hareket eksenine paralel düzlem boyunca sarılmasıyla açıklanabilir. Şekil 2.5.'de döner makinadan tüpsel makineye geçiş işlemi görülmektedir.



Şekil 2.5. Tüpsel yapılı makinanın elde edilişi [17]

Yapılan bu işlemleri tüpsel yapılı makinanın bütün türleri için rahatlıkla uygulayabiliriz. Bu makinanın sekonder kısmı bakım gerektirmez ve oldukça ucuzdur ayrıca bu özelliklerden dolayı da en yaygın kullanılan makinedir [28]. Bunun sonucunda en az döner makine sayısı kadar doğrusal hareket yapan makine elde edilir.



Şekil 2.6. Tüp şeklinde doğrusal motorlar [28]

## 2.2. SMTYDAM Özelliklerinin Karşılaştırılması

Doğrusal motorların özellikleri karşılaştırılarak birbirlerine göre olan üstünlük ve olumsuzluklarını görebiliriz.

Tablo 2.1. SMTYDAM'ın özellikleri [11]

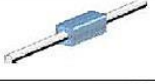
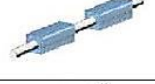
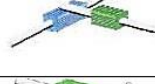
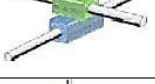


Özellikler	Demir Nüve	Hava Nüveli	Oluksuz	Tüp
Maliyet	Düşük	Yüksek	Düşük	Düşük
Çekim Kuvveti	En Yüksek	Yok	Orta	Yok
İtme Kuvveti	Çok İyi	Orta	İyi	Çok İyi
Isı Dağıtımı	İyi	Kötü	İyi	İyi

SMTYDAM'ın darbeli makine uygulamasında en fazla tercih edilmesinin ilk nedeni, diğer motorlara göre daha iyi itme karakteristiğine sahip olmasıdır. Bu motorların Tablo 2.1.' de belirtilen özelliklerine ek olarak aşağıdaki özellikleri de belirtebiliriz.

- Verimi oldukça yüksektir.
- Tasarım olarak simetriktir.
- Çok nadir hasar görürler.
- Dahili hareketli parça yoktur ve bakım gerektirmez.
- Doğrusal itmesi mükemmeldir.
- Sargılardan dolayı çok küçük bir manyetik gürültü oluşur.
- Hız aralıkları oldukça geniştir.

- Mekanik sağlamlığı oldukça yüksektir.
- Oldukça düzgün hareket eder.
- Yüksek hızlanma değerlerine sahiptir.
- İnce ayar gerektirmez.
- Hava aralığına duyarlı değildirler.
- Manyetik çekim kuvveti yoktur.
- Kurulumları çok basittir.
- Yapıları basit olduğundan kullanımları oldukça kolaydır.
- Kuvvet direkt olarak yüke transfer edilir.
- Direkt itmeli motordur, geri itmesi olmadığından aşınma olayı çok nadir gerçekleşir.
- Stator bobinlerin rotorun çevresini tamamen kaplamasından dolayı oluşan enerji optimum kullanılır.
- Hareket eden parça miktarı çok azdır ve mekanik ayar bakımından oldukça basittir.
- Manyetik kirliliğe minimum derecede neden olurlar fakat yok denecek kadar azdır.

Yukarıda belirtilen özelliklere ek olarak her türlü ekseninde çalışabilecek bir motor türüdür. Şekil 2.7.'de çalışabilecekleri eksenleri görebilirsiniz [11].

	X Tek eksenli tek itmeli blok.
	X1,X2 Tek eksenli çoklu itmeli blok sistemi.
	X,Y İki eksenli blok sistemi
	X,Y İki eksenli üst üste blok sistemi.
	3 veya 5 ekseninde hareket Üç boyutlu hareket
	X1, X2, X3, X4, Y1, Y2 Dört ayrı ekseninde birbirinden bağımsız hareket

Şekil 2.7. Tüpsel yapılı motorların çeşitli eksenlerdeki hareketleri [11]

### 2.3. SMTYDAM'ın Üstünlükleri

SMTYDAM'ın tercih edilmesinin temel nedenlerinden birisi, diğer doğrusal motorların sahip olduğu üstünlüklere sahiptir. Doğrusal motorlar diğer motorlarla kıyaslandığında daha sağlam ve daha hızlıdır. Bunun başlıca nedeni ise hareket alanları sınırsızdır, böylece hareket alanı boyunca rahatlıkla hareket edebilirler. Başlıca üstünlükleri aşağıda belirtilmektedir.

- Yüksek hız ve yüksek hassasiyeti bir arada sunar. Aynı noktalara tekrar hareketinde mükemmel sonuçlar verir. Duruma göre çok yüksek ve çok yavaş hızlarda çalışır.
- Motordaki tüm kuvvet yükün hareketinde kullanılır, doğrusal motorların kalkınma momentleri yüksektir.
- Doğrusal motorlar yumuşak hareket sağlarlar. Sistemde ark ve titreşim meydana gelmez bunun ana sebebi hareket boyunca yüzeyle temas olmamasıdır.
- Doğrusal motorlar daha doğru ve daha güvenlidir. Rotorlu sürücü sistemi ise düşük ayar ve düşük hareket performansına sahiptir.
- Doğrusal motorlar alışılmış mil yataklı sistemlere göre daha sağlamdır.
- Temiz oda uygulamaları için idealdir ve doğrusal motorlar temassız sistemlerdir. Düzenli olarak ayarları ve yağlanması yapıldığı zaman çok uzun süre bakım gerektirmez
- Malzemeler ve sargılar arasında temas olmadığından vakumlu sistemlerde rahatlıkla kullanılabilir [9].

### 2.4. SMTYDAM'ın Diğer Doğrusal Motorlarla Karşılaştırılması

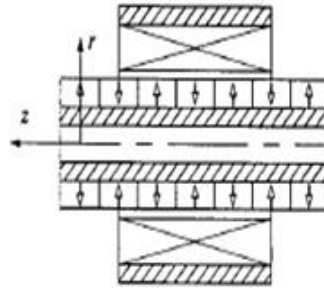
Doğrusal makinanın birkaç üstünlüğü olmasına rağmen ilk olarak bazı olumsuz yönlerden bahsetmek istiyorum. Bunlar [31];

- Büyük rotorlu sistemlerde kontrol zordur.
- Verimleri oldukça düşüktür.
- Doğrusal makinanın SMTYDAM'lara karşı üstünlükleri aşağıda belirtilmiştir. Bunlar [37];
- Konum hatası yok denecek kadar azdır.
- Açık çevrimli çalışmasında hiçbir problem yoktur, rahatlıkla çalışabilir.

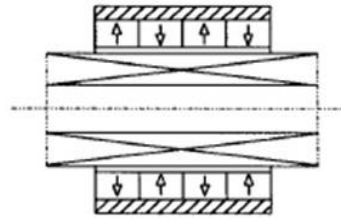
- Basit ve ucuzdur.
- Doğrusal hareket üretirken bağlantı mekanizmasına ihtiyaç duymaz.

## 2.5. SMTYDAM' nun Alan Dağılımı

Bu tip motorlarda matematiksel ifadenin çıkarılması manyetik alan dağılımı ile sağlanır ve bu ifade itme kuvvetinin tahminini gösterir. Şekil 2.8. ve 2.9.'da dairesel mıknatıslanmış dahili ve harici mıknatısların topolojisi gösterilmektedir. Burada sırasıyla hareket eden ya rotor ya da mıknatıstan herhangi biri olabilir. Başka bir ifadeyle içteki kısım rotor veya stator olabilir.



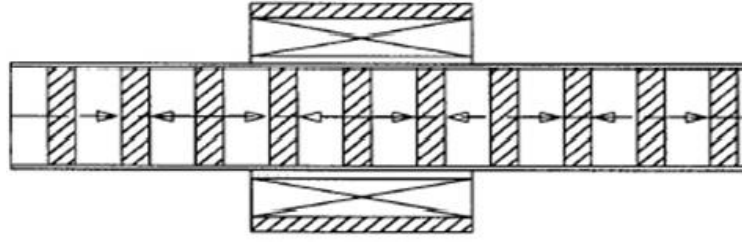
Şekil.2.8. Dahili mıknatıslar [7]



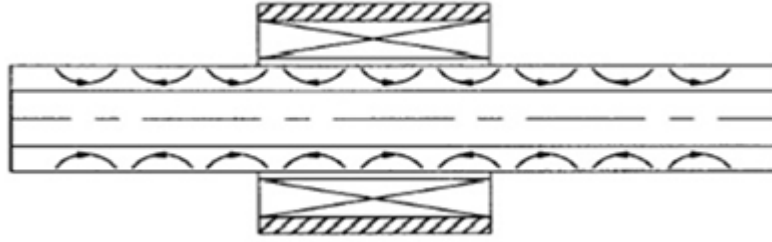
Şekil 2.9. Harici mıknatıslar [21]

Şekil 2.10.'de aksenal mıknatıslanmış mıknatısların topolojisi gösterilmekteyken Şekil 2.11.' de içsel mıknatısın halbach mıknatıslanmasının topolojisi gösterilmektedir.





Şekil 2.10. Aksel mıknaısların topolojisi [40]



Şekil 2.11. İçsel mıknaısların topolojisi [13]

Yukarıda gösterilen topolojilerde istenilen motor gücüne bağılı olarak rotor veya statordan herhangi biri diğlerinden daha uzun olabilir. Bu durum uygulanan sisteme bağılı olarak değışmektedir. Diğ taraftan da, oluklu demir çekirdek topolojileri daha güçlü akım şiddetine sahiptir fakat bu akım istenilmeyen dalgalanmalar oluşturabilir. Sonuç olarak, demir nüvede yüksek eddy akım kayıplarına neden olabilir. Bu durum eddy akım kayıplarına neden olurken aynı zamanda da motorun mevcut performansını azaltmaktadır.

## 2.6. Sürekli Mıknaıslar

Mıknaıs, özgül ağırlığı  $2.7 \text{ gr/cm}^3$  olan, manyetik alan üreten nesne veya malzeme olarak tanımlanmaktadır. Demir, nikel, kobalt gibi bazı metalleri çeker, bakır ve alüminyum gibi bazı metallere ve metal olmayan malzemelere etki etmez.

Günümüzde gelişmiş sanayide sürekli mıknaısın farklı malzemelerden yapılmış birçok çeşidi bulunmaktadır. Son zamanlarda ise nadir yer elementlerden yapılan mıknaıslar sayesinde önemi bir kat daha artmıştır. Bu durumun ana nedeni ise nadir yer elementlerinin aşırı derecede güç çarpanına sahip olmasıdır. Dolayısıyla aynı iş daha kolay ve küçük elde edilebilir. Sonuç olarak, bu tip sürekli mıknaıslar elektrik makinaları alanında geniş bir kullanıma sahip olmuştur.

Bu tez çalışmasında da nadir yer elementlerinden olan neodmium mıknatıslar kullanılmıştır. Neodmium mıknatısların manyetik kuvveti, normal mıknatıslarla kıyaslanmayacak derecede yüksektir. Bu manyetik kuvvet, çok özel sistemlerin yardımıyla ve özel cihazlarla sağlanmaktadır. Güçlü manyetik alanları ile dikkat çeken bu mıknatıslar farklı şekillerde ve boylarda üretilmekte olup kullanım alanına ve koşullarına göre, farklı elementlerle kaplanarak dayanıklılığı da sağlanmaktadır. Neodmium mıknatısların, küp, blok, disk, silindirik gibi farklı şekillerini görmek mümkündür.



Şekil 2.12. Neodmium mıknatıs çeşitleri[46]

Bu mıknatısın içerisinde bulunan neodmium, demir, samaryum, nikel ve kobalt gibi elementler topraktan çıkarılarak kullanılmaktadır. Elementlerin farklı oranlarda karışımıyla hazırlanan bileşimler, farklı niteliklere sahip olan mıknatısların elde edilmesini sağlar. Bunların kimisi daha güçlü, kimisi ısıya dayanıklı, bazısı da mukavemetli olarak üretilebilir. Mıknatısın dayanıklılığını arttırmak için, üzerleri uygun elementlerle kaplanır. Diğer mıknatıslar gibi manyetik güçlerini zamanla kaybetmeyerek, kalıcı bir manyetik alanın oluşmasını sağlamaktadırlar. Kendi ağırlıklarının yaklaşık olarak, 1300 katı kadar fazla ağırlığı taşıyabilme özelliği de, birçok alanda neodmium mıknatısların kullanılmasını gerektirmektedir.

Bu mıknatıslar bakır, altın, çinko, kalay ve gümüş gibi elementlerle kaplanarak, otomotiv, sağlık sektörü, elektrik motorları, petrokimya endüstrisi, oyuncak sektörü gibi alanlarda tüketicinin kullanımına sunulmaktadır.

## 2.7. Sürekli Mıknatıslı Motorların Üstünlükleri

- Rotordaki bakır kayıplarının azalmasından dolayı bu kayıplar statorda yoğunlaşır.
- Seçilen mıknatıs türüne göre aynı makinaların karakteristiği mıknatısın düzenlenme şekline bağlı olarak değişme gösterebilir.
- Kullanılan alanın verimli olması isteniyorsa mıknatıs kullanımına önem verilmesi gerekmektedir. Bu durum mıknatıs kullanımının önemini göstermektedir.
- Sürekli mıknatısların bakımı azdır ve verimi yüksektir. Bu durumun nedeni ise bilezik ve fırçaların olmamasıdır.

## 2.8. Sürekli Mıknatıslı Motorların Olumsuzlukları

- Harici güç elektroniği devreleri gerektirir.
- Rotor mıknatısları motor maliyetini artırır.
- Rotorun konumu bilinmelidir yoksa stator senkronize edilemez. Başka bir ifadeyle rotorun konumu hassas bir şekilde bilinmelidir.
- Sensör kullanılarak rotorun konumu kolaylıkla belirlenebilir fakat bu durum maliyeti artırır ve aynı zamanda da güvenilirliği düşürür.

### **3. YAPAY SİNİR AĞLARI**

Bu bölümde, çalışmanın denetimini sağlayan Yapay Sinir Ağları (YSA) teknolojisinin gelişimindeki aşamalar ile YSA'nın temel unsurları, çeşitli ağ yapıları ve ne tür sistemlerde kullanıldıkları ile YSA'nın özellikleri üzerinde durulmuştur.

#### **3.1. YSA' nın Tanımı**

Yapay sinir ağları (YSA), insan beyninin çalışma yapısını modelleyerek beynin öğrenme, hatırlama, genelleme yapma yolu ile yeni bilgiler ortaya çıkarma gibi temel işlevlerini gerçek yaşama uyarlayabilmek üzere geliştirilen mantıksal donanım ve yazılımlardır. Kısaca YSA; insan beyninin çalışma yapısını kullanarak bilgisayarlara düşünme yeteneği sağlamak üzere geliştirilmiş bir sistemdir.

YSA modelleme tekniği üzerine çalışmalar önceleri temel tıp birimlerinde insan beynindeki nöronların matematiksel modellenmesi çabaları ile başlamış, geçtiğimiz on sene içerisinde fizik, matematik, elektrik ve bilgisayar mühendisliği gibi çok farklı bilim dallarında araştırma konusu haline gelmiştir [33]. Özellikle son zamanlarda bilgisayar kullanımının hızla yaygınlaşması sonucunda yapay zeka çalışmaları da bir ivme kazanmıştır. Bu ivmenin en önemli nedeni, klasik tekniklerle çözümü zor olan doğrusal olmayan karmaşık problemler için ysa'nın çözüm kabiliyeti ve gerçek hayattaki verilerin eğitilmesi yeteneğine sahip olmasıdır [36].

#### **3.2. YSA' nın Tarihi Gelişimi**

YSA'nın temelleri, 1943 yılında bir nörobiyolog olan Mc Culloch ve istatistikçi Pitts tarafından yayınlanan makale ile atılmıştır. YSA'nın dayandığı ilk hesaplama modelinin ortaya koyulduğu bu makaleyi 1954 yılında Farley ve Clark' ın uyarılara adaptif ve tepkisel yaklaşım sergileyen bir ağ modelini konu alan çalışmaları takip etmiştir. Benzer bir diğer çalışma da 1956 yılında Rochester, Holland, Haibit ve Duda tarafından gerçekleştirilmiştir. Bu dönem 1958' de, Frank Rosenblatt ve Wightman ilk perceptron tipini önerene kadar böyle devam etmiştir.

1958 yılından sonra ysa arařtırmaları az sayıda bilim adamı tarafından sürdürülmüř ve kısmi ilerlemeler kaydedilmiřtir. Örneđin bu durgunluk döneminde Widrow tarafından 1960 yılında algılayıcıya getirilen farklı bir yaklaşımın ürünü olan Adaline (Adaptive Linear Element) ile YSA teknolojisini analog bir elektronik cihaz kadar somutlařtırmıřtır [19].

80'lerin ortalarında Sinir Ağları çalıřmaları tekrar gelişmeye bařlamıřtır. Bu Hopfield, Amari, Grossberg, Kohonen, Oji gibi ünlü birçok bilim adamının katkıları ile gerçekteřmiřtir. 1982' de Hopfield, hesaplama problemlerini çözmek için aktif bir şekilde kullanılabilen fiziksel sistemleri modelleyen yeni bir fikir ortaya çıkarmıřtır. Bu model belli bir enerji düzeyine yakınsayan birinci dereceden doğrusal olmayan diferansiyel denklem takımları ile ifade ediliyordu. Hopfield ağları bu yüzden bazen "iliřkili ağlar" olarak da anılmaktadırlar [34].

Takip eden yıllarda bir kısım bilim adamının birbirlerinden habersiz olarak çok katmanlı algılayıcıyı geliřtirmeleri ve 1986 yılında Rumelhart ve arkadaşlarının geri yayılım algoritmasını geliřtirmesi ile yapay sinir ağları, doğrusal olmayan problemlere de yanıt vermeye ve dolayısıyla daha fazla akademisyenin ilgisini çekmeye bařlamıřtır. 1985'ten günümüze kadar yapılan çalıřmalar ise YSA uygulamalarındaki çeřitliliđi arttırmıř ve literatüre pek çok kaynak kazandırmıřtır.

Günümüzde YSA ile gerçekteřtirilen uygulamalar arasında tahmin, sistem modelleme ve model sınıflandırma, iyileřtirme, otomasyon ve kontrol, adaptif filtreleme, veri iliřkilendirme, veri sıkıřtırma, örüntü tanıma ve eşleřtirme gibi pek çok karmařık problem yer almaktadır [19].

### **3.3. YSA'nın Yapısı**

YSA teknolojisi, canlılarda bulunan sinir sisteminin çalıřmasını modelleyerek elektronik ortama tařımaya yönelik bir programlama yaklaşımı olduđundan yapay sinir ağlarının çalıřmasını anlayabilmek için öncelikle biyolojik sinir sisteminin yapısına bakmak gerekir. Oysa insan beyni, hala sayısız bilinmeyen olan oldukça karmařık bir sistemdir ve günümüz teknolojisine rađmen çözümlenmesi zor bir sistemdir. Bu nedenden dolayı, bu kısımda sadece biyolojik sinir hücresinin yapısı ve çalıřmasına değinilmektedir. Biyolojik sinir sisteminin yapı tařı olan sinir hücreleri nöronlar, YSA'nın da yapı tařıdır.

### 3.3.1. Biyolojik Sinir Hücresinin Yapısı

Biyolojik sinir sisteminin temel yapı taşı olan nöronlar dört ana bölümden oluşmaktadır. Bunlar; soma(çekirdek), dendrit(sinir dalları), akson(eksen), sinapslardır.

Çekirdek(soma) ve hücrenin yaşamsal işlevlerini sağlayan mekanizmayı kapsar. Şekli farklı nöron çeşitlerine göre değişiklik gösteren çekirdek dentritler yoluyla iletilen tüm sinyalleri alıp toplayan merkezdir. Çekirdek gelen toplam sinyali diğer sinir hücrelerine göndermek üzere, bilgiyi aksone iletir.

Dendrit(sinir dalları) çok sayıda kısa uzantıdan oluşur ve hücrenin kökleri gibidir. Kısa olmalarına rağmen dendritler beynin işleyişinde oldukça önemli bir yere sahiptir. Görevi diğer sinir hücrelerinden iletilen sinyalleri, sinir hücresinin çekirdeğine iletmektir. Hücrenin çekirdeği ile her bir dendrit arasında farklı bir iletişim söz konusudur. Bu sebeple bazı dendritlerin etkileşimde ağırlıklı (dominant) pay sahibi, diğerlerinin de pasif (resesif) olduğu gözlenmektedir. Bu ise dışarıdan alınan sinyallerde seçicilik gibi önemli bir olgunun sinir hücresi tarafından gerçekleştirilmesi anlamını taşımaktadır [27,33].

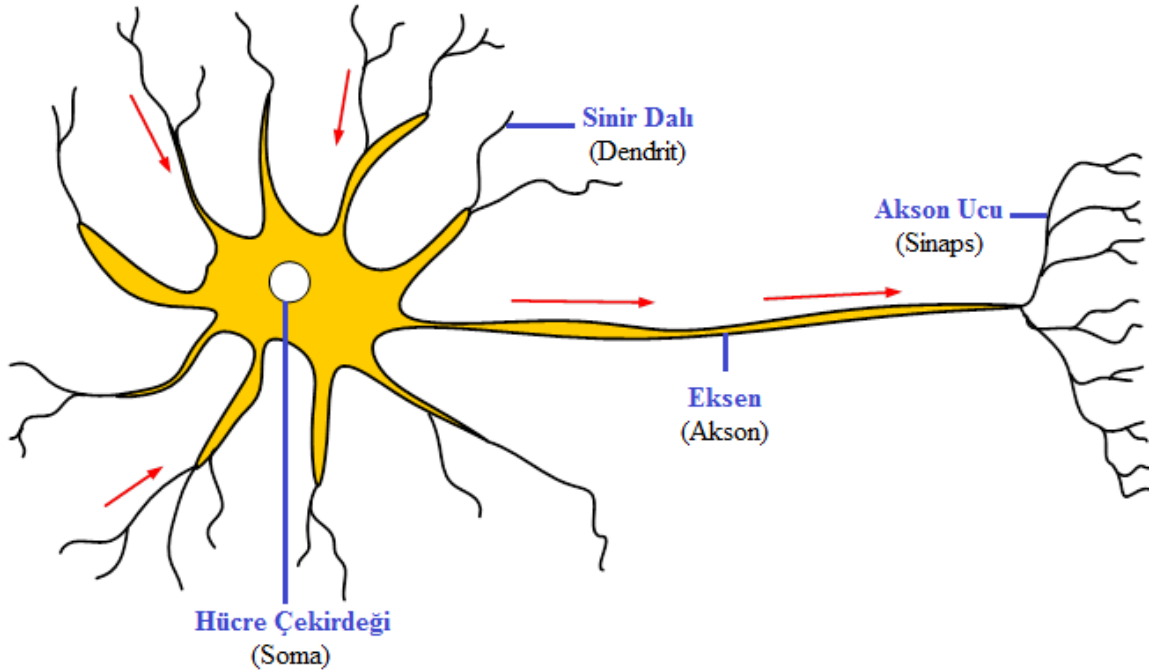
Akson(eksen) sinir hücresinin ana koludur ve her sinir hücresinde sadece bir tane akson bulunur. Aksonun görevi hücre çekirdeğinden aldığı toplam bilgiyi bir sonraki sinir hücresine dağıtmaktır. Ancak akson bu toplam sinyalin ön işlemde geçirilmeden diğer sinir hücresine aktarılmasına engel olur. Çünkü akson ucunda sinaps denilen ve aralarında ancak elektron mikroskobu ile görülebilen bir “boşluk” bulunan birimlere bilgiyi aktarır [24].

Sinaps(akson ucu) aksondan gelen toplam bilgiyi ön işlemde geçirdikten sonra diğer sinir hücrelerinin dendritlerine iletmekle görevlidir. Sinapsın ön işlem ile gerçekleştirdiği görev çok önem taşımaktadır. Bu ön işlem gelen toplam sinyalin, belli bir eşik değerine göre değiştirilmesinden ibarettir.

Böylece toplam sinyal olduğu gibi değil, belli bir aralığa indirgenerek diğer sinir hücrelerine iletilmiş olunur. Bu açıdan, her gelen toplam sinyal ile dendrite iletilen sinyal arasında bir korelasyon (ilişki) oluşturulur. Buradan yola çıkılarak “öğrenme” işleminin sinapslarda gerçekleştiği fikri ortaya atılmış ve bu hipotez, günümüz yapay sinir ağı dünyası için teori haline dönüşmüştür. Yapay sinir ağı modelleri üzerinde “öğrenme” bu teoriye dayanılarak, sinapslar ve dendritler arasında yer alan ağırlık katsayılarının güncellenmesi olarak algılanmaktadır.

Merkezi sinir sisteminin temel işlem elemanı sinir hücresidir ve insan beyinde yaklaşık 15 milyar sinir hücresi olduğu tahmin edilmektedir. Dendritler, diğer hücrelerden gelen sinyalleri hücre gövdesine yani çekirdeğe iletir. Çekirdek dendritten gelen sinyalleri bir araya toplar ve aksona iletir. Aksonlar, gelen bilgiyi hücreden dışarı taşıyan daha uzun bir yoldur. Toplanan bu sinyaller, akson tarafından işlenerek sinapslara gönderilir. Aksonların bitiminde sinapsa gelen ve dendritler tarafından alınan bilgiler belirli bir eşik değerine ulaştığında hücre bir tepki üretir. Hücrenin tepkisini artırıcı yöndeki girişler uyarıcı, azaltıcı yöndeki girişler ise önleyici girişler olarak söylenir ve bu etkiyi sinaps belirler. Sinapslar da yeni üretilen sinyalleri diğer sinir hücrelerine iletir ve bu şekilde sinyal aktarımı gerçekleşir.

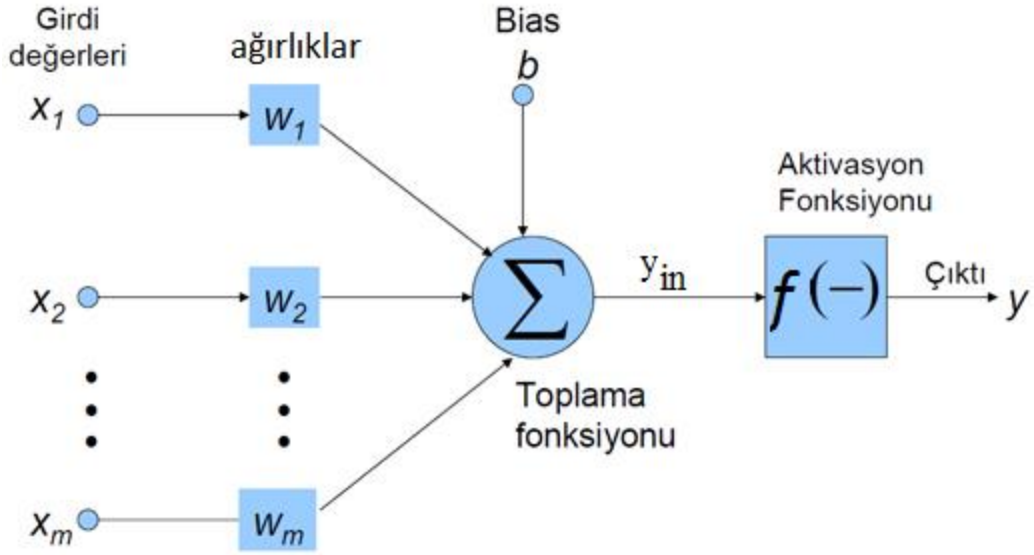
İnsan beyninin 60 trilyon sinaps bağlantısından oluştuğu düşünülürse son derece karmaşık ve etkin bir yapı olduğu anlaşılır. Diğer taraftan bir sinir hücresinin tepki hızı, günümüz bilgisayarlarına göre oldukça yavaş olmakla birlikte duyusal bilgileri son derecede hızlı değerlendirebilmektedir. Bu nedenle insan beyni; öğrenme, birleştirme, uyarılma ve genelleştirme yeteneği nedeniyle son derece karmaşık, doğrusal olmayan ve paralel dağılmış bir bilgi işleme sistemi olarak tanımlanabilir [12,27].



Şekil 3.1 Biyolojik sinir hücresi[41]

### 3.3.2. Yapay Sinir Hücresinin Yapısı

Yapay sinir ağları, biyolojik nöronlardan esinlenilerek modellendiğinden dolayı bir önceki kısımda biyolojik olan sinir hücresinin elemanları tanıtılmış, çalışma sistemi kısaca anlatılmıştır. Bu kısımda ise biyolojik sinir hücresinden yapay sinir hücre modeline geçiş yapılarak YSA'ların çalışma ilkeleri anlatılmıştır Yapay sinir ağlarının temel işlem elemanı olan yapay sinir hücreleri, doğal sinir hücrelerinin dört temel fonksiyonu ile şematize edilmektedir. Tasarlanan yapay sinir hücresi modellerinde çeşitli farklılıklar olabilmektedir. Ancak, genel olarak bir sinir hücresinin yapısı ve bileşenleri Şekil 3.2'de gösterildiği gibidir.



Şekil 3.2. Yapay sinir hücresinin yapısı[19]

En genel yapay sinir hücresine ait çıkışın matematiksel model denklemi:

$$y_{in} = \sum_{i=1}^n x_i \cdot w_i + b$$

$$y = f(\sum_{i=1}^n x_i \cdot w_i + b) \quad i = 1, 2, 3, \dots, n \quad (3.1.)$$

Bu gösterimde,  $y$  hücre çıktı değerini,  $f(.)$  transfer fonksiyonunu,  $y_{in}$  toplama fonksiyonunu,  $x_i$  hücrenin girdi değerlerini,  $w_i$  bağlantı ağırlıklarını göstermektedir,  $b$  ise esik değer olarak adlandırılır.



### 3.3.2.1. Girdiler ( $x_n$ )

Girdiler, dış çevreden ya da sistemdeki diğer nöronlardan ulaşan bilgilerdir. Girdiler yapay sinir hücresine bir diğer hücreden gelebileceği gibi direk olarak dış dünyadan da gelebilir. Bu bilgiler, ağın öğrenmesi istenen örnekler ile belirlenir, toplanmak üzere çekirdeğe gönderilir.

### 3.3.2.2. Ağırlıklar ( $w_n$ )

Yapay sinir hücresine gelen bilgiler çekirdeğe ulaşmadan önce geldikleri bağlantıların ağırlığıyla çarpılarak çekirdeğe iletilir. Ağırlıklar, bir sinir hücresine etki eden girdilerin matematiksel katsayılarıdır ve bağlantılı oldukları girdilerin sinir hücresini ne ağırlıkta etkileyeceğini belirler. Bu sayede girdilerin üretilecek çıktı üzerindeki etkisi ayarlanabilmektedir. Bu ağırlıkların değerleri pozitif, negatif veya sıfır olabilir. Ağırlığı sıfır olan girdilerin çıktı üzerinde herhangi bir etkisi olmamaktadır.

### 3.3.2.3. Birleştirme Fonksiyonu ( $\Sigma$ )

Birleştirme fonksiyonu, toplama fonksiyonu olarak da adlandırılmaktadır. Birleştirme fonksiyonları nörona gelen girdiler ile bu girdilere ait ağırlıkların çarpımlarının toplamını ifade eder.

### 3.3.2.4. Aktivasyon Fonksiyonu ( $f$ )

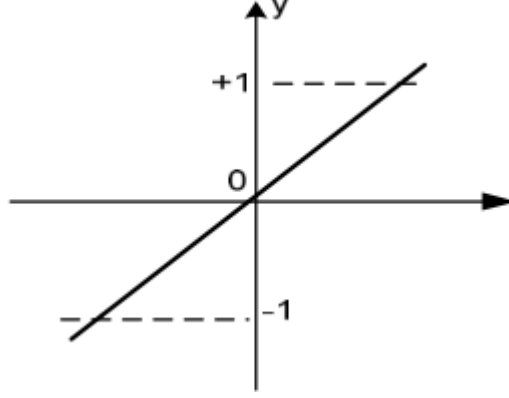
Bu fonksiyon, birleştirme fonksiyonu ile elde edilen sonucu seçilen bir aktivasyon işleminden geçirerek yapay sinir hücresinin çıktısını belirler.

Aktivasyon fonksiyonları karmaşık ve doğrusal olmayan problemlere çözüm getirilebilmesi için sıklıkla doğrusal olmayan yapıda da olabilirler. Tasarlanan yapıya en uygun aktivasyon fonksiyonu deneme-yanılma yoluyla en uygun sonucu verecek şekilde seçilmelidir. Çünkü bu fonksiyonlar, birleştirme fonksiyonu sonuçlarını seçilen aktivasyon fonksiyonuna uyarlayarak katmanların çıkışlarını belirli değerlere sınırlandırır ve atandıkları katmandaki tüm nöronlara aynı şekilde etki ederler.

Yaygın olarak kullanılan aktivasyon fonksiyonları adım(step) fonksiyonları, eşik (threshold), sigmoid fonksiyonu ile tanjant sigmoid fonksiyonu olup aşağıda detaylı olarak anlatılmaktadır [14,19].

#### 3.3.2.4.1. Doğrusal Aktivasyon Fonksiyonu

Doğrusal aktivasyon fonksiyonu doğrusal bir problemi çözmek amacıyla doğrusal hücre veya genellikle katmanlı yapay sinir ağlarının çıkış katmanında kullanılır. Hücre net girdisini doğrudan hücre çıkışı olarak verir. YSA'ların çıkış katmanında kullanılan doğrusal fonksiyon şekildeki gibidir.



Şekil 3.3. Doğrusal aktivasyon fonksiyonunu

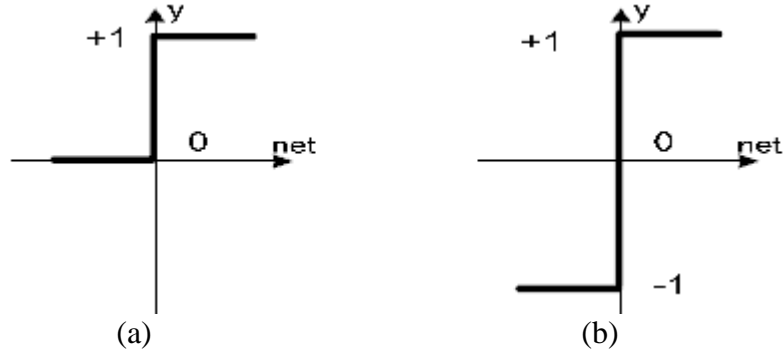
Doğrusal aktivasyon fonksiyonunun matematiksel ifadesi:

$$y = x \quad (3.2)$$

#### 3.3.2.4.2. Eşik Aktivasyon Fonksiyonu

Eşik aktivasyon fonksiyonu, eğer çıkış değeri sıfırdan küçükse sıfır, sıfırdan daha büyük bir değer ise çıkışında +1 değeri verir. Eşik aktivasyon fonksiyonunun -1 ile +1 arasında değişeni ise signum aktivasyon fonksiyonu olarak adlandırılır. Signum aktivasyon fonksiyonu, net giriş değeri sıfırdan büyükse +1, sıfırdan küçükse -1, sıfıra eşitse sıfır değerini verir.

Şekil 3.4'de eşik aktivasyon fonksiyonunun ve signum aktivasyon fonksiyonunun grafiği görülmektedir.



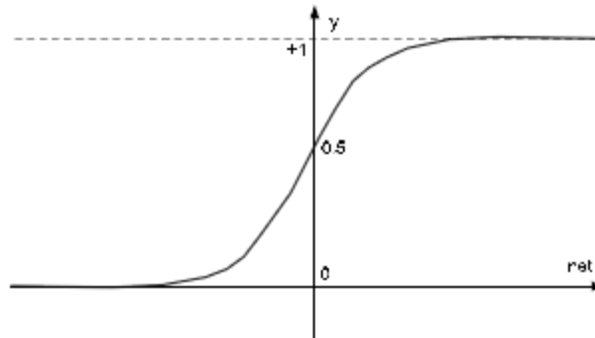
Şekil 3.4. (a) Eşik aktivasyon fonksiyonunu b) Signum aktivasyon fonksiyonu[24]

Eşik aktivasyon fonksiyonunun matematiksel ifadesi:

$$y = \begin{cases} -1, & x < 0 \\ 1, & x \geq 0 \end{cases} \quad (3.3)$$

### 3.3.2.4.3. Sigmoid Aktivasyon Fonksiyonu

Sigmoid aktivasyon fonksiyonu, türevi alınabilir ve sürekli bir fonksiyondur. Doğrusal olmayan bir fonksiyon olması nedeniyle uygulamada en çok kullanılan aktivasyon fonksiyonudur. Bu fonksiyon, girdinin her değeri için 0 ile 1 arasında bir değer üretir.



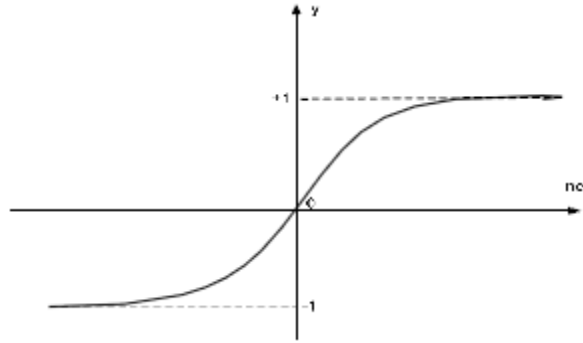
Şekil 3.5. Logaritma sigmoid aktivasyon fonksiyonu[24]

Sigmoid aktivasyon fonksiyonunun matematiksel ifadesi:

$$Y = \frac{1}{1+e^{(-x)}} \quad (3.4)$$

#### 3.3.2.4.4. Tanjant Sigmoid Aktivasyon Fonksiyonu

Tanjant sigmoid fonksiyonu, sigmoid fonksiyonu gibi türevi alınabilen bir fonksiyon olmasının yanında sigmoid fonksiyonunun biraz farklı seklidir. Sigmoid fonksiyonun çıktı aralığı 0 ve 1 değişirken, tanjant Sigmoid fonksiyonunun çıktısı -1 ve 1 aralığında değişmektedir.



Şekil 3.6. Tanjant sigmoid aktivasyon fonksiyonu[24]

Tanjant sigmoid aktivasyon fonksiyonunun matematiksel ifadesi:

$$y = \frac{1-e^{-2x}}{1+e^{-2x}} \quad (3.5)$$

#### 3.3.2.5. Çıktılar

Aktivasyon fonksiyonu uygulanmasıyla elde edilen değer, nöronun çıktı değeridir. Çıktılar, diğer nöronlara tekrar girdi olarak verilebileceği gibi dış ortama çıktı olarak da gönderilebilirler. Yapay sinir hücresinin çalışmasında dikkat edilmesi gereken husus, bir sinir hücresinin birden fazla girdisi olabilmesine rağmen sadece bir çıktısının olmasıdır.

Şekil 3.2’de görünen yapay sinir hücresinin girdi değerleri  $x_i$  ve her bir girdi değerlerinin ağırlık katsayısı (önemlilik derecesi)  $w_i$  ile belirtilmiştir. Bu girdi değerlerin her biri bir bağlantı ağırlığıyla çarpılmaktadır. Çekirdek ise tüm bu girdi sinyallerinin ağırlıklı toplamlarını elde etmektedir. Tüm bu toplam sinyal  $y_{in}$  ile gösterilmiş ve  $f$  ile belirtilen eşikleme fonksiyonuna girdi olarak yönlendirilmiştir. Eşikleme fonksiyonundan çıkan sonuç sinyali  $y$  ile belirtilmiş ve diğer hücreye beslenmek üzere yönlendirilmiştir.

Yapay sinir hücresinin görevi kısaca;  $x_i$  girdi örüntüsüne karşılık  $y$  çıktısı sinyalini oluşturmak ve bu sinyali diğer hücrelere iletmektir. Her  $x_i$  ile  $y$  arasındaki korelasyonu temsil eden  $w_n$  ağırlıkları, her yeni girdi örüntüsü ve çıktı sinyaline göre tekrar ayarlanır. Bu ayarlama süreci öğrenme olarak adlandırılır. Öğrenmenin tamamlandığının belirtilebilmesi için; girdi örüntüleri,  $w_i$  ağırlıklarındaki değişim stabilize olana dek sistemi beslemektedir. Stabilizasyon (durağanlık) sağlandığı zaman hücre öğrenmesini tamamlamıştır [12,24,27].

### **3.3.3. Yapay Sinir Ağının Yapısı**

Yapay nöronların birbirleriyle bağlantılar aracılığıyla bir araya gelmeleriyle yapay sinir ağı oluşmaktadır. Nöronların aynı doğrultu üzerinde bir araya gelmeleriyle katmanlar, katmanların değişik şekilde birbirleriyle bağlanmalarıyla da değişik ağ mimarileri oluşur. YSA’lar üç ana katmadan oluşur. Bu katmanlar sırasıyla; giriş, ara ve çıkış katmanlarıdır.

#### **3.3.3.1. Giriş Katmanı**

Giriş katmanı, yapay sinir ağına dış dünyadan gelen bilgilerin alındığı katmandır. Bu katmanda dış dünyadan gelecek giriş sayısı kadar nöron bulunmasına rağmen girdiler herhangi bir işleme uğramadan ara katmanlara iletilmektedir. Bu katmanda bilgi işleme yapılmamaktadır.

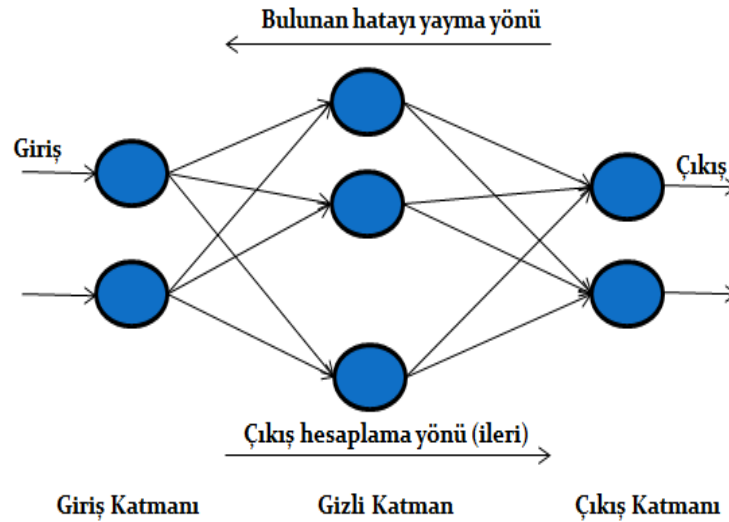
#### **3.3.3.2. Ara Katman (Gizli Katman)**

Giriş katmanından çıkan bilgiler ara katmana gelmektedir. Giriş katmanından gelen bilgiler, ara katmanlar boyunca çeşitli algoritmalarla işlenerek çıkış katmanına gönderilir.

Ara katman sayısı ağdan ağa değişebilir. Bir ağ içinde, gerçekleştirilmek istenen uygulamanın niteliğine göre bir veya daha fazla ara katman bulunabilir. Ara katmanlar çeşitli kaynaklarda gizli katman veya ara katman olarak da adlandırılmaktadır. Ara katmanlardaki hücre sayıları, girdi ve çıktı sayılarından bağımsızdır.

### 3.3.3.3. Çıkış Katmanı

Çıkış katmanı, ara katmanlardan gelen bilgileri işleyerek ağın girdi katmanından gelen verilere karşılık olan çıktıları üreten katmandır. Bu katmanda üretilen çıktılar dış dünyaya gönderilir. Bir YSA'da tek çıkış katmanı bulunur.



Şekil 3.7. Yapay sinir ağı modeli [27]

### 3.4. YSA'nın Sınıflandırılması

YSA işleyiş olarak benzer olmalarına rağmen herhangi bir tasarım ve işleyiş standardı bulunmamaktadır. Ağ mimarisi, kullanılan öğrenme yöntemi, nöronların bağlantı stratejileri vb. farklılıklara sahip çok sayıda YSA yapısı vardır. Nöron dizilimlerine, nöronların ağırlıklarının düzenlenmesi için yapılan hesaplamaların türüne ve zamanına göre YSA'lar üç ana kritere göre sınıflandırılabilir [30].

### 3.4.1. Yapılarına Göre YSA Çeşitleri

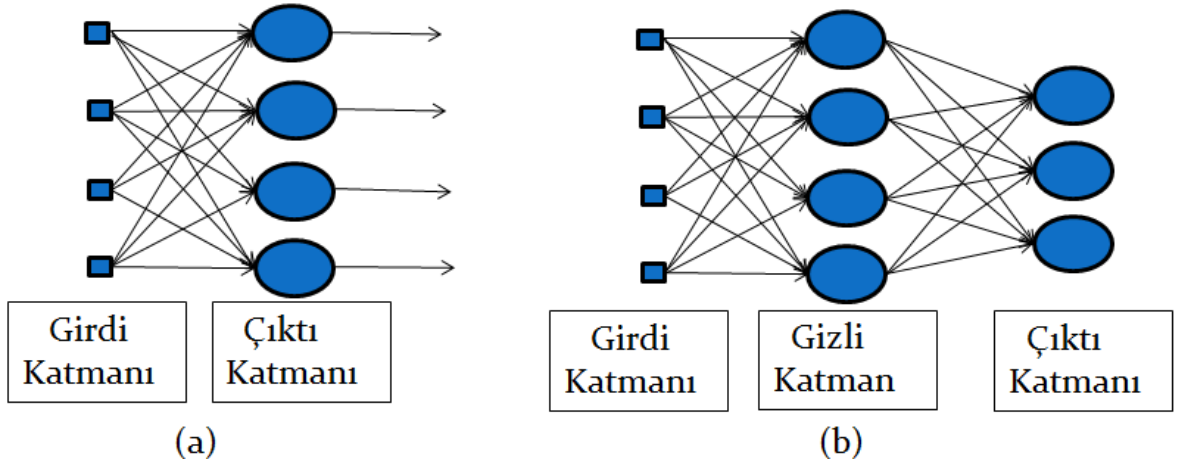
Yapay sinir ağları, yapılarına göre, ileri beslemeli (feedforward) ve geri beslemeli (feedback) ağlar olmak üzere iki şekilde sınıflandırılırlar.

#### 3.4.1.1. İleri Beslemeli Ağlar

İleri beslemeli ağlarda nöronlar girişten çıkışa doğru düzenli katmanlar şeklindedir. Bir katmandan sadece kendinden sonraki katmanlara bağ bulunmaktadır. Bu yapılarda döngülere yer verilmez ve bu sayede YSA çıktısı hızla sonuçlandırılabilir. Bir katmandaki nöronların çıktıları belirli oranlarda ağırlıklandırılarak bir sonraki katmana iletilir. Bu şekilde saklı katmanlar boyunca iletilen bilgiler çıkış katmanında işlenerek ağın çıktıları oluşturulur ve daha sonra dış dünyaya aktarılır. İleri beslemeli yapay sinir ağlarında gecikmeler yoktur. Kendi aralarında:

- Tek katmanlı ileri beslemeli ağlar
- Çok katmanlı ileri beslemeli ağlar olarak ayrılırlar.

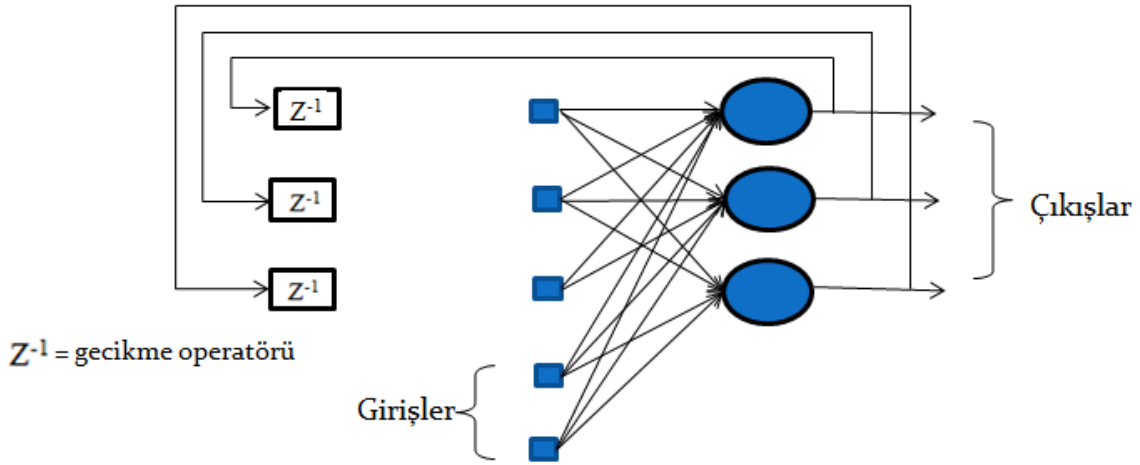
En çok bilinen geriye yayılım öğrenme algoritması, bu tip YSA'ların eğitiminde etkin olarak kullanılmakta ve bazen bu ağlara geriye yayılım ağları da denmektedir.



Şekil 3.8 a) Tek katmanlı ileri beslemeli ağlar b) Çok katmanlı ileri beslemeli ağlar[19]

### 3.4.1.2. Geri Beslemeli Ağlar

Geri beslemeli ağlarda, bir nöronun çıktısı sadece kendinden sonra gelen nöron katmanına girdi olarak verilmez. Kendinden önceki katmanda veya kendi katmanında bulunan herhangi bir nörona girdi olarak bağlanabilir. Böylece girdiler hem ileri hem de geri yönde iletilmiş olur. Geri besleme, bir katmandaki hücreler arasında olduğu gibi katmanlar arasındaki hücreler arasında da olabilir. Bu tür ağ yapılarının önemli bir özelliği bölgesel dağıtılmış hafızaya sahip olmalarıdır. Yani ağın belirli bir andaki çıktısı hem o andaki hem de daha önceki anlardaki girdilere ilişkin veriyi içermektedir[12,24,30].



Şekil 3.9. Geri beslemeli ağ yapısı [ 19]

### 3.4.2. Öğrenme Algoritmalarına Göre YSA Çeşitleri

YSA' da öğrenme, ağın verilen girdilere göre nöronlar arasındaki ağırlıkları çıktı üretebilecek şekilde ayarlanmasıdır. Bu ağırlıkların ne şekilde değiştirileceği öğrenme kuralları ile belirlenir. YSA öğrenme algoritmalarına göre danışmanlı, danışmansız ve destekleyici öğrenme olarak üçe ayrılır.

#### 3.4.2.1. Danışmanlı Öğrenme

Danışmanlı öğrenme sırasında ysa, kullanılmaya başlanmadan önce eğitilir. Eğitim sırasında, hem girdi değerleri hem de o girdi değerleri için istenen çıktı değerleri sisteme



verilir. İstenen çıktı ile ağın ürettiği çıktı karşılaştırılarak hata hesabı yapılır, ağırlıklar güncellenir. Hata payı hesaplanırken ağın bütün çıktıları ile beklenen çıktıları arasındaki fark hesaplanır ve bu farka göre her nörona düşen hata payı bulunur. Daha sonra her nöron kendine gelen ağırlıkları günceller. Bu sebeple danışmanlı öğrenme algoritmasının bir “öğretmene” veya “danışmana” ihtiyacı vardır [3,20].

#### **3.4.2.2. Danışmansız Öğrenme**

Danışmansız öğrenme, danışman gereksinimi olmadan, ağın gerçekleştirilmesi istenen davranışı kendi kendine öğrenebildiği bir algoritmadır. Yani sistemin öğrenmesine yardımcı olan herhangi bir danışman yoktur. Sisteme sadece örnek girdiler verilir, örneklerdeki parametreler arasındaki ilişkileri sistemin kendi kendine öğrenmesi beklenir. Girişte verilen bilgilere göre ağ her bir örneği kendi arasında sınıflandıracak şekilde kendi kurallarını oluşturur. Ağ bağlantı ağırlıklarını aynı özellikte olan dokuları ayırabilecek şekilde düzenleyerek öğrenme işlemini tamamlar.

Danışmansız öğrenen ağlarda girdiler, ağ tarafından daha detaylı yorumlanır ve aralarındaki benzerlikler ortaya çıkarılır. Bu tür ağlar, söz konusu özelliklerinden dolayı genellikle sınıflandırma problemlerinin çözümünde tercih edilir. Danışmansız öğrenme algoritmasında, sadece YSA'nın girdileri üzerinde işlem yapıldığı için danışmansız öğrenen ağların hızı, danışmanlı öğrenen ağlara göre daha yüksek olmaktadır [20,30].

#### **3.4.2.3. Destekleyici Öğrenme**

Destekleyici öğrenme yaklaşımında ağın her iterasyonu sonucunda elde ettiği sonucun ne ölçüde doğru olduğunu belirten bir bilgi verilir. Ağ bu bilgilere göre kendini yeniden düzenler. Bu sayede ağ herhangi bir girdi dizisiyle hem öğrenerek hem de sonuç çıkararak işlemeye devam eder [20,30].

#### **3.4.3. Kullanılan Veri Türüne Göre YSA Çeşitleri**

Kullanılan veri türüne göre ysa, niteliksel veri kullanan ağlar ve niceliksel veri kullanan ağlar olarak iki grupta toplanırlar.

### **3.4.3.1. Niteliksel Veri Kullanan Ağlar**

Niteliksel veri kullanan ağlar, YSA girdilerinde dış görünüş, renk, tat, biçim gibi sayısal olarak ölçülemeyen özelliklere ilişkin niteliksel verinin kullanıldığı ağ türüdür. Bu ağ türü, desen oluşturma ve sınıflandırma problemlerinde kullanılmaktadır.

### **3.4.3.2. Niceliksel Veri Kullanan Ağlar**

Niceliksel veri kullanan ağlar, YSA girdilerinde birimlerin ölçüm ve tartım sonucu değerleri saptanan sayısal özelliklerini belirten, aralıklı ölçekli veya orantılı ölçekli değişkenler gibi niceliksel verilerin kullanıldığı ağ türüdür. Bu ağ türü, genelleme ve tahmin problemlerinde kullanılmaktadır.

### **3.4.4. Öğrenme Zamanına Göre YSA Çeşitleri**

Yapay sinir ağları öğrenme zamanına göre de statik ve dinamik öğrenme olarak ikiye ayrılır.

#### **3.4.4.1. Statik Öğrenme**

Statik öğrenme kuralıyla çalışan yapay sinir ağları kullanmadan önce eğitilmektedir. Eğitim tamamlandıktan sonra ağ istenilen şekilde kullanılabilir. Ancak bu kullanım sırasında ağın üzerindeki ağırlıklarda herhangi bir değişiklik olmaz.

#### **3.4.4.2. Dinamik Öğrenme**

Dinamik öğrenme kuralı ise yapay sinir ağlarının çalıştığı süre boyunca öğrenmesini öngörerek tasarlanmıştır. Yapay sinir eğitim aşaması bittikten sonra da daha sonraki kullanımlarında çıkışların onaylanmasına göre ağırlıklarını değiştirerek çalışmaya devam eder.

### 3.5. YSA' da Öğrenme Kuralları

Yapay sinir ağlarında ağı eğitimi hangi algoritmada çalışırsa çalışsın bir kurala göre dayandırılmaktadır. Nöronlar arasındaki ağırlıkların değiştirilme yöntemi, öğrenme kurallarından hangisinin uygulandığına göre farklılıklar gösterir. . Bu öğrenme kuralı temel olarak iki grupta incelenebilmektedir:

#### 3.5.1. Çevrimiçi Öğrenme Kuralları

Gerçek zamanlı olarak çalışabilen bu tip kurallar uygulanırken sistemde bir taraftan işlem fonksiyonları yerine getirilirken diğer taraftan öğrenmeye devam edilmektedir. Backpropagation Net (Delta kuralı) ve Khonen öğrenme kuralı bu sınıfa örnek olarak verilebilir [25].

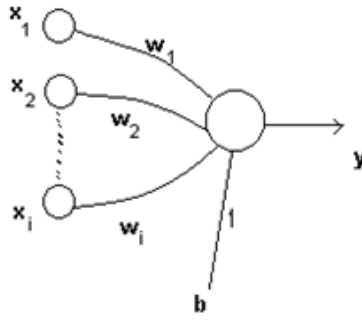
#### 3.5.2. Çevrimdışı Öğrenme Kuralları

Bu tip öğrenmede öncelikle ağı eğitilir, daha sonra ağı gerçek hayatta kullanıma alınır. Ağı kullanımdayken öğrenme gerçekleşmemektedir. Yeni verilerin ve durumların ağı öğretilmesi istendiğinde ağı hizmetten çıkarılır ve çevrimdışı halde yeniden eğitilip tekrar kullanıma sunulur [25].

Yapay sinir ağlarında sıklıkla kullanılan dört temel öğrenme kuralı şu başlıklar altında sıralanabilir: Hebb kuralı, Hopfield kuralı, Delta kuralı ve Khonen kuralıdır.

##### 3.5.2.1. Hebb Kuralı

1949 yılında Hebb tarafından geliştirilen bu öğrenme kuralı diğer tüm kuralların temelini oluşturmaktadır. Daha sonra geliştirilen çeşitli öğrenme kurallarına esin kaynağı olan bu öğrenme kuralında, eğer art arda gelen iki hücrenin matematiksel işareti aynı ise, bu iki hücrenin bağlantısı kuvvetlendirilir. Bağlantının kuvvetlendirilmesi, iki hücre arasındaki ağırlık değerinin artırılması ile yapılır. Eğer hücrelerin matematiksel işaretleri farklı ise bu durumda da hücreler arası bağlantılar zayıflatılır, diğer bir ifade ile ağırlıklar azaltılır.



Şekil 3.10 Hebb'in mimarisi[27]

Hebb'in öğrenme formülü aşağıda verilmiştir.

$$w_i(\text{yeni}) = w_i(\text{eski}) + x_i \cdot y \quad (3.6)$$

Hebb'in algoritması aşağıdaki gibidir;

Adım 0: Tüm ağırlıkları başlangıç değerlerine ata

$$w_i = 0 \quad i = 1, 2, 3, \dots, n$$

Adım 1: Her bir girdi örüntüsü (s) ve o örüntünün istenen sonucu (t) gereken değer ikilisi için

Adım 2-4 arasını tekrarla.

Adım 2: Her bir girdi elemanını girdi örüntüsünden al:

$$x_i = s_i \quad i = 1, 2, 3, \dots, n$$

Adım 3: Çıkması gereken sonu ata

$$y = t$$

Adım 4: Ağırlıkları ve denge unsurunu ayarla

$$w_i(\text{yeni}) = w_i(\text{eski}) + x_i * y \quad (i = 1, 2, 3, \dots, n)$$

$$b(\text{yeni}) = b(\text{eski}) + y$$

Hebb'in basit yapısı güncel olarak basit sınıflandırma problemlerinde kullanılmaktadır. Sınıflandırma rol oynayan “y” (3.6) formülündeki gibi hesaplandığında çıkan sonuç öğrenmenin tamamlandığını ve girdinin hangi sınıfa ait olduğunu belirtir.

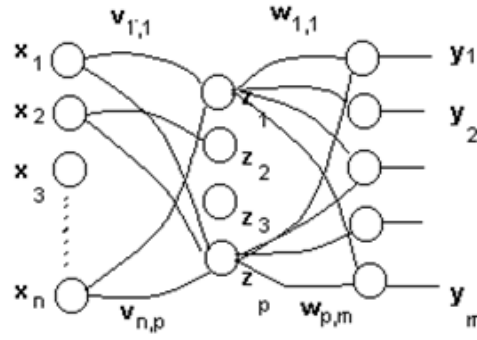
### 3.5.2.2. Delta Kuralı (Backpropagation Net)

Backpropagation net modeli yapay sinir ağları içerisinde hemen hemen her probleme rahatlıkla uyarlanabilir bir yapıyı oluşturmaktadır. Backpropagation geribeslemeli bir öğrenme mekanizması kullanır. Burada yapay sinir ağının yapısı ileri beslemeli olmasına karşın, hatanın geriye doğru ket vurmasından kaynaklanan bir öğrenme olduğu için geri besleme söz konusu edilmektedir. Bu kurala göre, ağırlık değerleri, yapay sinir ağının ürettiği çıkış değeri ile üretilmesi istenen hedef değer arasındaki hatanın karesel ortalamasını en az yapacak şekilde değiştirilir.

Backpropagation öğrenmede sürekli (continuous) girdi tipini kullanır. Aktivasyon için ise türevi alınabilecek bir fonksiyonu işleme sokmaktadır. Genellikle sigmoid fonksiyonunu kullanır.

$$w_{i,j}(new) = w_{i,j}(old) + (\mu * [t - f(y_{in})] * f'(y_{in})) \quad (3.7)$$

Denklem 3.7 'e göre türevi alınabilen bir fonksiyon kullanılması backpropagation tipi bir yapay sinir ağı modeli için önem taşımaktadır. Çünkü türev bir eğri üzerinde değişim olarak tarif edilmektedir. Bir başka deyişle; hatanın minimize edilmesi demek, hatanın türevinin 0 olması anlamına gelmektedir. Bu yüzden backpropagation hatanın türevini “wij” ağırlıkları üzerinde öğrenmekte bu sayede hatalar her bir iterasyon (döngü) sonunda 0'a doğru yaklaşmaktadır. Backpropagation bu sebepten dolayı başarılı kullanım sahasına sahip olup mimarisi Şekil 3.11' deki gibidir [27].



Şekil 3.11. Standart 3 katmanlı (x girdi, z saklı, y çıktı) Backpropagation ağı yapısı[27]

Backpropagation mimarisi için kullanılan algoritma, ilk adımda “y” ler üzerinde oluşan hatayı saklı ve çıktı katmanı arasında yer alan “ $w_{p,m}$ ” ler üzerine yansıtırlar. Ancak güncelleme hemen gerçekleştirilmez. “ $w_{p,m}$ ” lerin güncelleme yapılmamış olan hallerinde oluşan hatalar ise girdi ve saklı katman arasında yer alan “ $v_{p,n}$ ” ler üzerine yansıtılır. “ $w_{p,m}$ ” ve “ $v_{p,n}$ ” ler aynı anda güncellenerek backpropagation algoritmasının paralel bir yapı arz ederek öğrenmesi sağlanmaktadır. Bu paralel yapı sayesinde backpropagation mimarisinin performansı diğer yapılara oranla daha yüksek olacaktır. Backpropagation algoritmasında kullanılan sigmoid fonksiyonu ve bu fonksiyonun türevi (3.8) ve (3.9) formüllerinde verilmiştir [27].

$$f_{sigmoid}(x) = \frac{2}{1 + e^{-x}} - 1 \quad (3.8)$$

$$f'_{sigmoid}(x) = \frac{1}{2} * |1 + f_{sigmoid}(x)| * |1 - f_{sigmoid}(x)| \quad (3.9)$$

Backpropagation algoritması aşağıdaki gibidir;

Adım 0: Ağırlıklara başlangıç değeri ata.

$$0 < w_{jk} < 1, 0 < v_{ij} < 1$$

Adım 1: Enerji stabilize olan dek 2-9.Adımlar arasını tekrarla

Adım 2: Her bir girdi örüntüsü (s) ve ona ait olan çıktı değeri (t) için 3-8. arasını tekrarla.

İleri beslemeli Kısım:

Adım 3: Girdi sinyalini al ve üst katmana ilet.

$$x_i = s_i$$

Adım 4: Her bir gizli katman için toplam sinyalleri hesapla.

$$z_{in\_j} = S \cdot x_i * v_{ij}$$

Her bir gizli katman için çıkış değeri hesapla.

$$z_j = f_{sigmoid}(z_{in\_j})$$

Adım 5: Her bir çıktı katmanı için toplam sinyalleri hesapla.

$$y_{in\_k} = S z_j * w_{jk}$$

Her bir çıktı hücresinin çıkış değerini hesapla.

$$y_k = f_{sigmoid}(y_{in\_k})$$

Hatanın Geri beslenmesi Kısım:

Adım 6: Çıktı katmanında oluşan hatayı hesapla

$$d_k = (t_k - y_k) * f_{sigmoid}(y_{in\_k})$$

Çıktı-saklı katmanları arası yapılacak ağırlık değişimini bul.

$$Dw_{jk} = m * d_k * z_j$$

Adım 7: Her bir saklı katman üzerindeki hatayı hesapla

$$d_{in\_j} = S d_k * w_{jk}$$

Girdi katmanına yansıtılacak hatayı hesapla

$$d_j = d_{in\_j} * f_{\phi sigmoid}(z_{in\_j})$$

Girdi-saklı katmanları arası yapılacak ağırlık değişimini bul

$$Dv_{ij} = m * d_j * x_i$$

Adım 8: Simdi tüm ağırlıkları aynı anda güncelle.

$$w_{jk}(yeni) = w_{jk}(eski) + Dw_{jk}$$

$$w_{jk}(yeni) = w_{jk}(eski) + Dw_{jk}$$

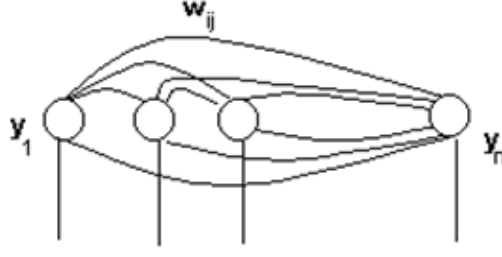
Adım 9: iterasyonu kontrol et. Bitir.

### 3.5.2.3. Hopfield Kuralı

Hopfield kuralı, Hebb kuralına benzer bir algoritmadır. Ancak Hebb kuralından farklı olarak iki hücrenin bağlantılarının, ne kadar kuvvetlendirileceği ya da zayıflatılacağı, öğrenme katsayısına göre belirlenir. Her yinelemede ne derece değişiklik yapılacağı ise tasarımcının belirlediği bir öğrenme katsayısı oranınca olur. Girdi ve çıktı nöronlarının her ikisi de aynı polariteye sahipse bağlantı ağırlıkları öğrenme katsayısı kadar arttırılır, farklı polariteye sahipse öğrenme katsayısı kadar azaltılır.

Mimari olarak Hopfield tek katmanlı bir yapıya sahiptir. Bu tek katman hem girdi hem de çıktı katmanı olarak görev yapar. Simetrik olarak tüm sinir hücreleri birbirine bağlanmıştır. Sadece hücrelerin kendi üzerlerine bağlantıları mevcut değildir. Şekil 3.12' de Hopfield modelinin mimarisi açık biçimde görülmektedir.





Şekil 3.12. Hopfield net modeli[27]

Hopfield modelinin öğrenme sürecinde kullandığı kural en basit kural olarak bilinen-Hebb kuralıdır. Yukarıda belirtildiği üzere ağırlıkların öğrenilmesi iki sinir hücresi arasında ki değerin aynı olup olmadığına bağlıdır. Girdi örüntülerinde binary (ikili) değerlerden ziyade bipolar(iki kutuplu) değerlerin kullanılması daha doğru olan yaklaşımdır. Buna göre her girilen örüntü için öğrenme şöyle ifade edilir;

$$W = \{w_{ij}\} \text{ ve } w_{ij} = \sum_p s_i(p) * s_j(p) \quad , i \neq j \quad (3.10)$$

Öğrenme Hopfield’da çok basit olan Hebb kuralı ile sağlanmaktadır. Yapılan “m” tane (p=m) girdi için(her bir girdi “n” hücreden oluşmaktadır. ‘i=j=1,2,...n.’) ağırlık matrisini oluşturmaktadır.

Örneğin y1 hücresi -1, y3 hücresi -1 iken,  $w_{13} = w_{13} + (-1)*(-1) = w_{13} + 1$ . Tüm girdi örüntüleri kullanılarak oluşturulduğunda “wij” üzerinde öğrenmenin tamamlanmış olduğu belirtilir.

Hopfield modelinin öğrenim süreci tamamlandıktan sonra asil önemli olan, girdi örüntüsünün sisteme beslendiğinde istenen sonuca ulaşp ulaşamayacağıının (convergence) kontrolünü yapacak enerji fonksiyonunun kurulmasıdır. Hopfield; Lyapunov fonksiyonu ile sistemin enerjisinin hedefe ulaşabileceği (convergence) ve o hedefte stabilize halinde kalabileceğinin kontrolünü yine bu fonksiyon yardımıyla sağlayabileceğini ispat etmiştir. Hopfield modelinin en önemli parçası olan enerji fonksiyonu formül (3.11)’de belirtilmiştir.

$$E = -\left(\frac{1}{2}\right) * \sum_{i \neq j} \sum_j y_i * y_j * w_{ij} - \sum_i x_i * y_i + \sum_j \varphi_i * y_i \quad (3.11)$$

Enerji formülüne dikkat edilecek olunursa, burada her bir hücrenin aktifliğinin kontrolünün yapıldığı ve aktif olanların ağırlık değerlerinin enerjiye katıldığı görülecektir. Bu noktada önemli olan girdi örüntüsünün  $\Delta y_i$  kadar bir değişikliğe uğratıldığında (yönlendirildiğinde) enerji değişiminin ne olacağını hesaplanmasıdır. Enerjide oluşan bu değişim aşağıdaki formül ile belirtilmiştir:

$$\Delta E = -[\sum_j (y_j * w_{ij}) + x_i - \phi_i] * \Delta y_j \quad (3.12)$$

Hopfield modelinin algoritması ise aşağıda sunulmaktadır;

Adım 0: Ağırlıkların değerlerini öğren. (Hebb kuralı kullanarak)

$$w_{ij} = S_p s_i(p) * s_j(p) \quad i \neq j \text{ iken}$$

Enerji lokal minimumu yakalayana kadar 1-7. Adımlar arasını tekrarla

Adım 1: Her bir girdi vektörü için 2-6. Adımlar arasını tekrarla.

Adım 2: Dışarıdan gelen girdi vektörünü sisteme besle

$$y_i = x_i$$

Adım 3: Her bir sinir hücresi için 4-6. Adımlar arasını tekrarla

Adım 4: Her hücrenin girdi toplam sinyalini bul

$$y_{in_i} = x_i + S_j (y_j * w_{ij})$$

Adım 5: Aktivasyonu hesapla

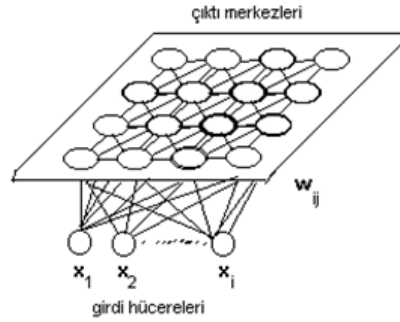
Adım 6: Çıktıları diğer hücrelere dağıt.

Adım 7: Enerji fonksiyonu hesapla, “converge” noktasındaysa dur, değilse rastgele bir “ $y_i$ ” değerini  $\Delta y_i$  kadar değiştir.

### 3.5.2.4. Khonen Kuralı

Khonen tarafından geliştirilen bir öğrenme kuralıdır. Bu kurala göre yapay sinir ağındaki hücreler, ağırlıklarını değiştirmek için birbirleri ile yarışır. Çıkış değeri en büyük olan hücre yarışı kazanır ve ağırlık değerlerinin değiştirilmesine izin verilir. Eğitim setinin beklenen çıktısına en uygun sonucu üreten nöronların ağırlıkları ayarlanarak sonuca olan etkileri kuvvetlendirilir.

Khonen geliştirdiği mimaride, tamamen kendi kendine öğrenmeyi hedeflediği için ele aldığı “k” adetlik girdi popülasyonunun ilk “n” adedini “n” cluster (öbek, merkez noktası) olarak yerleştirir. Ve tüm popülasyon elemanlarını bu merkez noktalarına göre öğrenir. Öğrenmek için alınan girdi, tüm cluster’lar ile kıyaslanır ve girdi hangi cluster’a (gruba) yakın görünüyorsa oraya adapte edilir (güncellenir). Ve her nokta yakın bulunduğu grup merkezini kendine doğru çekerek grup merkezlerinin daha iyi grup temsilcisi olabilmesini sağlamaya çalışır. Eğer bu işlem 100, 1000, 10000 iterasyon ile tekrarlanırsa, görünen sonuç; cluster olarak belirlenen grup merkezlerinin popülasyonu en iyi temsil eden grup merkezleri olduğu görülecektir. Bir başka deyişle, her grup merkezi örnek uzayın ayrı ayrı sınıflarını temsil edecektir. Bu sayede girdi örüntüleri kendilerini ait oldukları sınıfın merkezine yakın bulacaklardır. Elbette ki “komşuluk” kavramı sayesinde birbirine yakın olan sınıflar yine birbirleriyle ilişkili hareket edeceklerdir. Khonen geliştirdiği yapay sinir ağı modeli için Şekil 3.13’ deki mimariyi temel aldı:



Şekil 3.13. Khonen' in 2 katmanlı (girdi, çıktı) SOM ileri beslemeli ağ yapısı[27]

Khonen yapısının algoritması aşağıda belirtilmiştir;

Adım 0: Ağırlıklara başlangıç değerlerinin rastgele küçük sayılarla ataması yap.

$$0 < w_i < 1$$

Öğrenme katsayısını da ata

$$0 < m < 1$$

Adım 1: Yeterli iterasyona ulaşıncaya dek 2-8. Adımlar arasını tekrarla

Adım 2: Her bir girdi örüntüsü için 3-5. Adımlar arasını tekrarla

Adım 3: Her bir merkez noktası (j) için uzaklık hesapla

$$\text{uzaklık}(j) = S (w_{ij} - x_i)^2$$

Adım 4: Minimum uzaklığa sahip olan j. cluster' i bul.

$$\text{index} = \min \{j, \text{uzaklık}(j)\}$$

Adım 5: index' in gösterdiği cluster' in ve komsularının ağırlıklarını güncelle.

$$w_{i,\text{index}(\text{yeni})} = w_{i,\text{index}(\text{eski})} + m * (x_{i,\text{index}} - w_{i,\text{index}(\text{eski})})$$

Komsuları için ise “r” uzaklık katsayısı gözeterek güncelleme yap.

Adım 6: Öğrenme katsayısını hassaslaştır.

$$m = m * 0.6 \text{ (0.6 rastgele bir sayıdır)}$$

Adım 7: komşuluk derecesini azalt

Adım 8: Durusu kontrol et [30,33].

### 3.6. Yapay Sinir Ağlarının Üstünlükleri

Doğrusal olmama: Doğrusal yöntemler, karmaşık problemlerin çözümünde yetersiz kalmaktadır. Ancak YSA teknolojisi gücünü karmaşık problemlere getirdiği basit

çözümlerden almaktadır. Bu avantajı sağlayan temel unsur ise YSA'nın temel bilgi işleme elemanı olan nöronların ve dolayısıyla ağın bütününe doğrusal olmayan yapısıdır.

**Öğrenme:** Alışlagelmiş yöntemlerle çözüm aranan problemler çoğunlukla kendi özelinde çözümlenmeye çalışılmıştır. Oysa bileşenlerinin ilişkileri tam anlamıyla tanımlanamayan problemler de olasıdır. Bu tür durumlarda insan beyninin öğrenme ve öğrenilenler doğrultusunda karar verme yetisi öne çıkmaktadır. YSA da benzer şekilde problemleri eğitim aşamasında ağa verilen bilgilerle çözer. Dolayısıyla çözülecek problemdeki değişiklikler için ağın değil, eğitim verisinin yani YSA'ya öğretilenlerin değiştirilmesi yeterlidir. Öğrenme, gerek uygulama kolaylığı gerekse çözüm kabiliyeti olarak ortaya çıkan önemli bir özelliktir.

**Genelleme:** Öğrenme, yukarıda bahsedilenlerin yanı sıra YSA'ya genelleme yapılabilmesi gibi bir diğer önemli özellik daha kazandırmaktadır. Genellemenin önemi ise YSA'ya girilen veride gürültü, hata, eksiklik olması gibi durumlarda dahi YSA çıkışında doğru sonuçlar elde edilebilmesidir. Bu özellik sayesinde örüntü tanıma ve sınıflandırma problemleri için başarılı bir teknik olmuştur.

**Paralellik:** Bilgi işleme sistemlerinde gözetilmesi gereken önemli bir husus ilsem hızıdır. Ancak problem bütününe tüm parçalarının sırasıyla işlendiği seri yöntemler karmaşık işlemler için hayli yavaşlatıcı olmaktadır. Bu sebepten YSA'nın paralel bilgi işleme özelliği problemlere getirdiği hızlı çözümler sayesinde büyük önem taşır, çünkü gerçek zamanlı sistemlerin temeli paralel bilgi işleme yöntemlerinde yatmaktadır.

Paralel bilgi işleme sistemlerinde problem, parçalar halinde ele alındığı ve sistemin her bir elemanı ayrı bir parçanın çözümünü gerçekleştirdiği için olası hatalara karşı daha büyük tolerans vardır. Tasarımdaki bu esneklik sayesinde, ağın herhangi bir kısmının zarar görmesi ağın tamamının işlevini yerine getirememesi anlamına gelmediği gibi sadece, işlevselliğini kaybeden nöronlar oranında bir performans kaybıyla tatminkâr sonuçlar elde edilebilir.

**Tasarım ve analiz kolaylığı:** Tüm YSA tasarımlarının temelinde, YSA'nın en küçük yapay bilgi işleme elemanı olan nöronlar yer alır. Daha önce de bahsedildiği gibi nöronlar yaklaşık olarak aynı yapıdadır ve bu benzerlik sayesinde farklı uygulama alanlarında standart ağ yapılarının ve öğrenme algoritmalarının kullanım imkanı vardır. Teoride aynı olan yöntemlerin ise uygulamada tasarım ve analiz kolaylığı sağlaması olağandır [19].

### 3.7. Yapay Sinir Ağlarının Olumsuzlukları

Günümüzde makinelerin çoğu seri şekilde çalışmakta olup üzerlerinde paralel işlemleri yapmak zaman kaybına yol açmaktadır. YSA'ların gerçek zamanlı veri işleyebilmeli paralel işlemciler üzerinde çalışması ile sağlanabilmektedir. Üstelik bir ağın nasıl oluşturulması gerektiği tam olarak bilinmemektedir. Her problem için farklı sayıda işlemci gerekebilmektedir.

Problemlere uygun ağ yapısı deneme yanılma yöntemi ile bulunur. Ağlara göre parametre değerleri (öğrenme katsayısı, her katmanda olması gereken işlem elemanı sayısı vb.), katman sayısı da değişmektedir, standart bir formülizasyon bulunmamaktadır. Dolayısı ile YSA'ları kesin çözümü değil en iyi çözümü bize sunar.

Problem ağlara tanıtılırken YSA'ların doğası gereği problemlerin sayısal şekilde ifade edilmesi gerekmektedir. Bu süreçte sürekli değerler kullanılabileceği gibi kesikli (binary) değerler de kullanılabilir. Fakat kullanıcının tecrübesizliği çoğu olayın YSA'ları ile çözülememesinin nedeni olarak görülmektedir.

Ağların eğitimine ne zaman son verileceğine karar vermek için de belirlenmiş bir yöntem yoktur. YSA'ların en büyük dezavantajı ise ağların çözüm üretirken bunu nasıl ve neden ürettiklerinin tam olarak bilinmemesidir [12].

### 3.8. YSA'nın Kullanıldığı Alanlar

Yapay sinir ağlarının kullanılabileceği birçok alan vardır. Dolayısıyla yapay sinir ağlarının kullanıldığı ve başarılı sonuçlar elde edilen yüzlerce uygulama sayılabilir. Fakat herhangi bir problemin çözümü için yeterli etkinlikte ve verimlilikte bir yöntem varsa, bu problemi yapay sinir ağı ile çözmek mantıklı olmayabilir.

Başarılı uygulamalar incelendiğinde, yapay sinir ağlarının, doğrusal olmayan, çok boyutlu, gürültülü ve hata olasılığı yüksek sensör verilerinin bulunduğu, kesin olmayan, karmaşık, verilerin eksik olduğu, ortaya çıkan çözümde matematiksel modele ve algoritmaya ihtiyaç duyulmayan hallerde yaygın olarak kullanıldıkları görülmektedir [32].

#### a) Yapay Sinir Ağlarının Kullanıldığı Teorik Uygulamalar

- Doğrusal olmayan sistem modelleme
- Akıllı kontrol
- Sinyal filtreleme ve doğrusal olmayan sinyal işleme

- Optimizasyon
- Probabilistik fonksiyon kestirimleri
- Sınıflandırma
- Örüntü tanıma, ilişkilendirme ve eşleştirme

#### b) Yapay Sinir Ağlarının Kullanıldığı Pratik Uygulamalar

- Kimyasal proseslerin modellenmesi
- Akıllı araçlar ve robotlar için optimum rota belirleme
- Robotlarda görme sistemlerinin ve hareket mekanizmalarının kontrol edilmesi
- Makine, sistem ve süreçler için arıza tespiti
- İletişim kanallarındaki ekoların filtrelenmesi, anahtarlama ve yoğunluğun kontrolü
- Hedef tanıma ve takip sistemleri
- Radar ve sonar sinyallerinin sınıflandırılması
- Radar ve görüntü sinyalleri işleme
- Güvenlik sistemlerinde konuşma ve parmak izi tanıma
- Mekanik parçaların ömürlerinin ve kırılmalarının tahmin edilmesi
- Endüstriyel ürünlerin görsel kalite kontrolü ve imalatta meydana gelen hataların

tespiti

- Kredi kartı hilelerinin tespiti
- Döviz kuru tahminleri, risk analizleri

## **4. SMTYDHDAM'IN GERÇEKLENMESİ VE YSA İLE DENETİMİ**

### **4.1. Mekanik Kısımın Gerçeklenmesi**

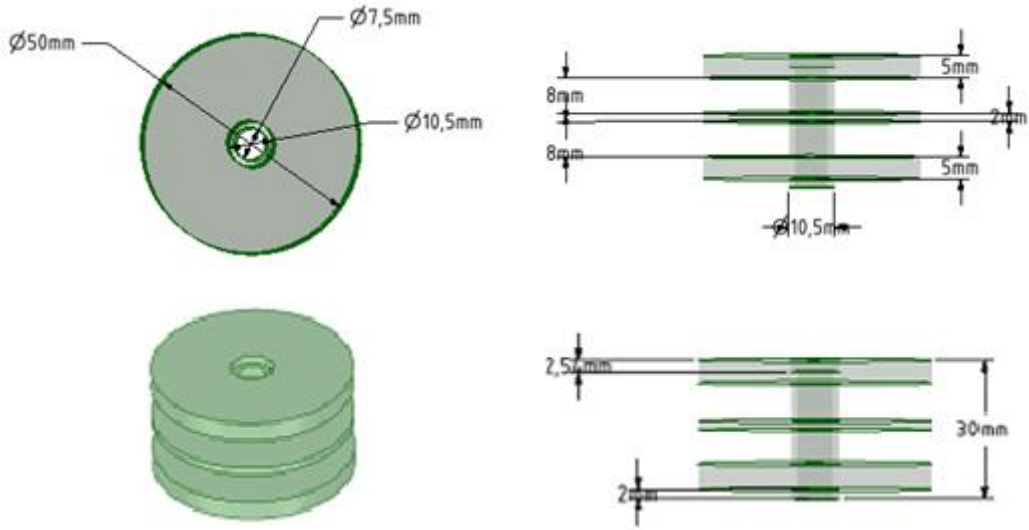
#### **4.1.1. Stator Tasarımı**

Makinenin durağan kısmı olan statoru oluşturan temel parçalar farklı zaman dilimlerinde enerjilenen dört adet bobin ve bu bobin sargılarının üzerine yerleştirildiği kestamit malzemeden yapılmış gövdedir. İş makinelerinde, yüksek darbeye karşı dayanıklılık gerektiren yerlerde kullanılan kestamit, döküm yoluyla imal edilen, sıkı bir dokuya ve sertliğe sahip bir malzemedir. Düşük sürtünme katsayısı, ekonomik olması, kolay işlenebilirliği ve aşınma mukavemetinin çok yüksek olması başlıca özellikleridir.

Elektrik makinelerinin gövdeleri genellikle akı yolu relüktansını düşürmek, fuko-histerezis kayıplarını azaltmak, manyetik akı yoğunluğunu yükseltmek için sac paketlerden yapılır. Fakat bu tasarımda stator gövdesi olarak manyetik histerezisliğin öneminden dolayı kestamit tercih edilmiştir. Doğrusal hareketin kontrollü ve sürekli olabilmesi için gövdede manyetik alan oluşmamalı, bobinlerin sadece belli aralıklarda elektrikleşmesi ve birbirini manyetik olarak etkilememesi gerekmektedir. Bu nedenle yukarıda belirtilmiş olan üstünlüklere bağlı olarak kestamit kullanılmıştır.

Stator modeline, çeşitli gövde modelleri üzerinde yapılan denemeler ile ulaşılan en verimli model bulunarak karar verilmiştir. Söz konusu stator model Şekil 4.12'de verilmiştir. Bu modelde, sargılar arası kısa devre oluşumunu engellenmek amacıyla sargılar kendi aralarında aynı malzemeyle yalıtılmıştır. Tasarımda bu gövdeden dört tane yapılmıştır. Gövde üzerindeki oluklara 0.30 mm çapına sahip emaye kaplı bakır tel sarılmıştır. Tel çapı, motorun sürekli olarak ileri-geri yönde yapacağı hareket durumunda, sargılarında oluşacak akım göz önüne alınarak belirlenmiştir. Bahsedilmiş olan stator gövdeleri 2, 3, 4'lü kombinasyonlarda denenerek denetim açısından ciddi farklılıklar gözlenmediğinden son olarak 2'li kombinasyonda karar kılınmıştır. Böylece, istenen çalışmaya daha hafif, daha az yer kaplayan ve daha kısa bir stator ile ulaşılması sağlanmıştır.

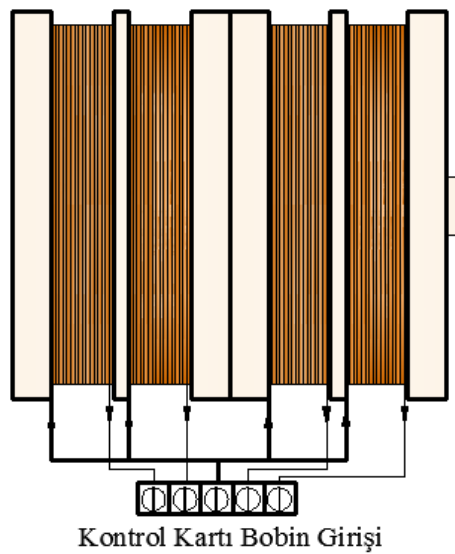




Şekil 4.1. Tüpsel yapılı doğrusal hareketli motorun stator gövdesinin yapısı ve ölçüleri

Stator gövdesinin parçalar halinde ve herhangi bir ara malzemeye ihtiyaç duymaksızın birbirine monte edilebilir yapılmasının temel nedeni, olası bir arıza durumunda gövdenin bütüne dokunulmadan sadece arızalı bobininin bulunduğu gövdenin kısa sürede değiştirilebilmesine imkan sağlamaktır.

Bobin sargıları Şekil 4.2.'de gösterildiği düzende bağlanmış olup, sargı genişliği 7.5mm ve sargı yükseklikleri 48mm'dir. Sarım sayısı yaklaşık olarak 1770 sarımdır.



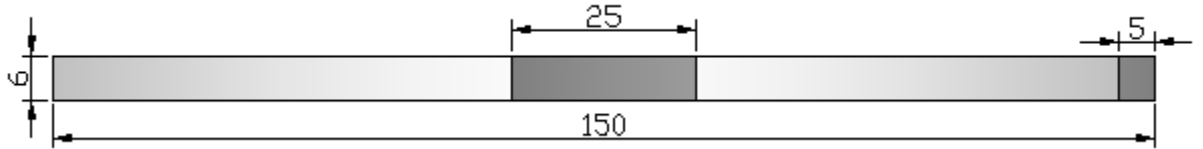
Şekil 4.2. Bobin bağlantı şeması

#### 4.1.2. Rotor Tasarımı

Sürekli mıknatıslı doğrusal hareketli DA motorların çalışma ilkesi sabit mıknatıslar arasında oluşturulmuş kalıcı manyetik alan ile motor sargıları üzerinden akan akımının etkileşimiyle ortaya çıkan itme veya çekme kuvvetlerine dayanır. Sürekli mıknatıslı motorlarda rotor tamamen sabit mıknatıstan yapılmaz. Çünkü sabit mıknatısın manyetik alan geçirgenliği havaya yakındır yani manyetik alan geçirgenliği açısından demire (saca) göre oldukça kötüdür. Tamamen sabit mıknatıstan oluşacak bir rotor, stator sargı endüktans değerinin düşmesine neden olmaktadır. Bu nedenle stator manyetik alanını kuvvetlendirmek için rotor, sürekli mıknatıs ve silisli sacların çeşitli yapılarda beraber kullanılmasıyla oluşturulur [41].

Bölüm 2.6.'da ve yukarıda verilen bilgilerden yola çıkılarak sabit mıknatıslı rotor tasarlanmıştır. Rotor tasarımında 10mm çapında, 20 mm kalınlığında bir adet neodmium mıknatıs ve 150mm uzunluğunda, 6mm çapında içi boş alüminyum çubuk kullanılmıştır. Mıknatıslar, kullanıldığı yere tutturularak veya yapıştırılarak kullanılmaktadır. Yapılan çalışmada bu işlem, ferromanyetik olmayan bir metal olan alüminyum çubuk içerisine yerleştirilerek gerçekleştirilmiştir.

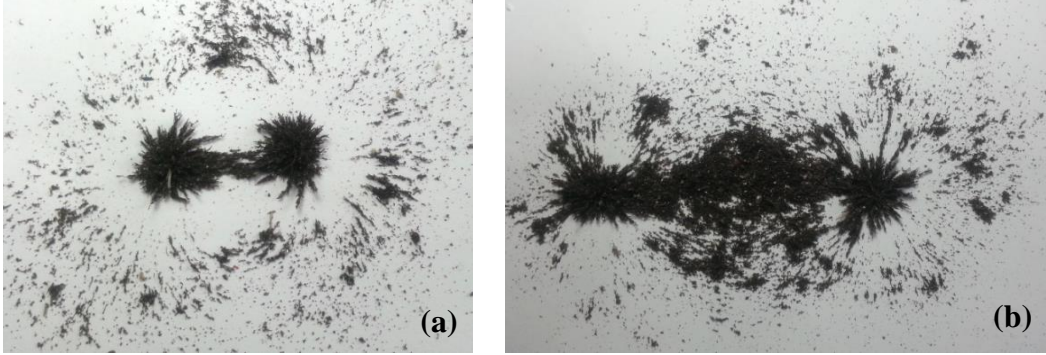
Statorun senkronize edilebilmesi için rotor konumunun hassas bir şekilde bilinmesi gerektiğinden sensör devresi kullanılmıştır. Kullanılan sensör devresi Bölüm 4.2.3.'de detaylı olarak anlatılmaktadır. Sensörlerin rotor konumunu algılayabilmeleri için rotorun sensör devresine karşılık gelen uç kısmına 5mm kalınlığında, 5mm çapında bir neodmiyum mıknatıs eklenmiştir. Rotor tasarımı Şekil 4.3'te verilmektedir.



Şekil 4.3. Tüpsel doğrusal hareketli doğru akım motorunun rotor ölçüleri

Neodmiyum mıknatıslar belli standartlarda üretilmektedir. Bu nedenle rotorun tamamı bir bütün mıknatıstan oluşturulamamıştır. Bu duruma alternatif olarak özdeş mıknatıslar birbirine eklenerek rotor oluşturulmuştur.

Fakat bu yapıdaki rotor çalışmada verimsiz olmuştur. Şöyle ki, rotor içine yerleştirilen mıknatıs bekleme sürelerinde sargı gruplarının altına gelmişse dışarıdan müdahale edilene kadar hareketsiz kalmakta, ileri geri hareketini gerçekleştirirken tutukluk yapmaktadır. Bu durumun nedenini tespit etmek için mıknatıslar ve manyetik alanları ile ilgili deney yapılmıştır.



Şekil 4.4. a) Tek mıknatısın manyetik alan dağılımı b) Birleştirilmiş mıknatısların manyetik alan dağılımı

Deney sonucunda tek bir mıknatısın manyetik alanı ile birleştirilmiş mıknatısın manyetik alan dağılımının aynı olmadığı, birleştirilmiş mıknatısların bir bütün mıknatıs gibi davranmadığı gözlemlenmiştir. Deney sonucu elde edilen manyetik alan dağılımları Şekil 4.4'te gösterildiği gibidir.

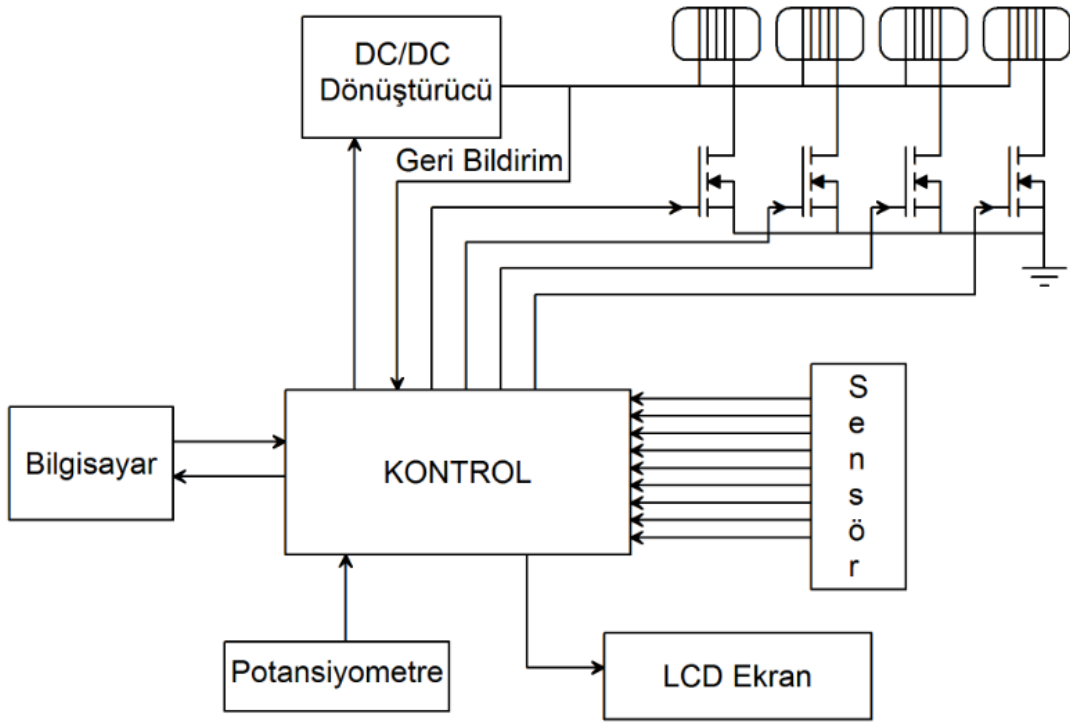
Rotorun elektromıknatıs değil de doğal mıknatıstan oluşması enerjinin stator sargılarına verilmesini gerektirmektedir. Stator sargılarına sabit bir doğru gerilimin verilmesi durumunda döner alan oluşmayacaktır. Döner alanın oluşabilmesi için kutup sargılarının sırası ile enerjilendirilmesi gerekir.

Doğrusal hareketli motor uygulamalarında doğrusal hareket rotorun ileri-geri hareketi ile oluşturulur. Rotor hareketini sağlayan statorun istenildiği kadar uzun yapılabilmesinden dolayı da rotorun hareket mesafesi için herhangi bir kısıtlama yoktur. Bu çalışmada stator uzunluğu 58mm, rotor uzunluğu 150mm olarak kullanılmıştır. Rotor uzunluğu netleştirilmeden önce farklı uzunluklarda çalışmalar yapılmış, uzunlukla ivmenin ters orantılı olarak arttığı, buna paralel olarak yön değiştirme süresinin uzadığı ve tasarlanan kontrol düzeneği ile denetimi sağlamanın zorlaştığı gözlemlenmiştir.

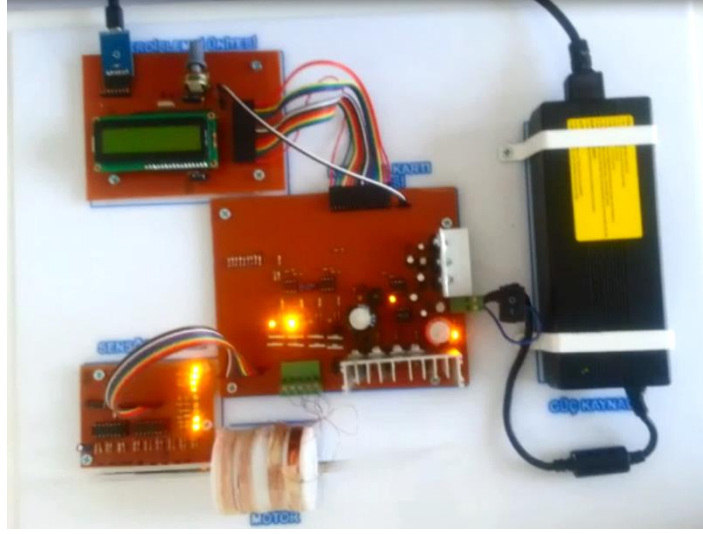
Bu nedenlerden dolayı rotor, statora göre az bir uzunlukta, ileri-geri hareketi gözlemlenmeye yetecek kadar kısa olacak şekilde gerçekleştirilmiştir.

## 4.2. Kontrol Kısımının Gerçeklenmesi

Gerçekleştirilmiş olan SMTYDHDAM'ın kontrolü amacıyla Kontrol Kartı Ünitesi, Mikrodenetleyici Ünitesi ve Sensör Ünitesi olmak üzere üç ayrı ünitenin birleşiminden meydana gelen bir elektronik kontrol sistemi oluşturulmuştur. Oluşturulan bu elektronik kontrol sistemi, Evrensel Seri Veri Yolu (**Universal Serial Bus**) aracılığıyla, gerçek zamanlı çalışan matlab yazılımı ile yönetilmektedir. Sistemin kontrolü Kapalı Çevrim YSA Denetimli Kontrol ile sağlanmaktadır. Sistem genel olarak motor bobinlerinin belirlenen kural diziliminde, Darbe Genlik Modülasyonu Tekniği (**Puls Width Modulation**) ile verilen güçte aktifleştirilmesi durumunda rotorun ileri-geri hareketini gerçekleştirmesini sağlamaktadır.



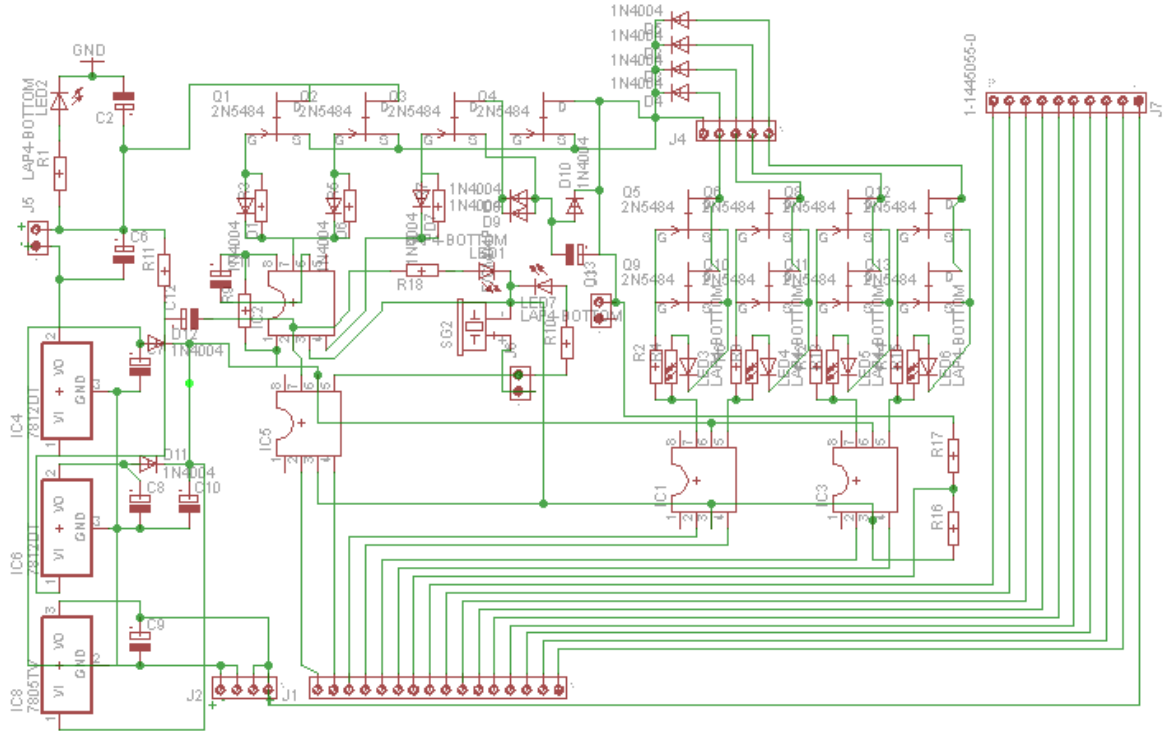
Şekil 4.5. Gerçeklenen SMTYDHDAM sürücü devre şeması



Şekil 4.6 Gerçekleştirilen SMTYDHDAM sürücü sistem fotoğrafı

#### 4.2.1. Kontrol Kartı Ünitesi

Kontrol kartı ünitesi üç kısımdan oluşmaktadır. Bunlar; Step Down DA-DA Dönüştürücü(Kıyıcı), 5V-12V'luk Gerilim Regülatörleri ve Motor Sürücüdür.



Şekil 4.7. Kontrol kartı ünitesi elektrik devre şeması

Sistemde motorun hız denetimi ysa ile sabit frekans altında darbe genişliği değiştirilerek yapılmaktadır. Bu nedenle en kolay kontrol yöntemi olarak bilinen DA PWM kontrolü kullanılmıştır. DA kıyıcılar, bir endüktans gerektirmeden, bir DA gerilimi yüksek frekansta kıyarak çıkıştaki ortalama DA gerilimi giriş geriliminin altında olmak üzere ayarlayan, temel olarak düşürücü türe giren ve yaygın olarak DA motor kontrolünde kullanılan dönüştürücüler olarak bilinir. DA kıyıcı prensip olarak bir kontrollü güç elemanı (transistör, aktif eleman) ve bir kontrolsüz güç elemanı (diyot, pasif eleman) ile elde edilir[4].

Gerçekleştirilen sistemin çalışmasında ihtiyaç duyulan akım ve yüksek frekansları karşılama kapasitesinden dolayı IRF630N mosfet, kontrolsüz güç elemanı olarak da SBLI040CT diyot kullanılmıştır. Kullanılan mosfetlerin ilettime geçmesi için gate ucundan verilen gerilim 10V civarı olmalıdır. Sistemin ana beslemesi 24V'luk DA güç kaynağı ile sağlanmakta olduğundan bu gerilimi karşılamak için 7812 gerilim regülatörü kullanılmıştır.

Gerilim regülatörlerine paralel bağlı diyotlar ile ihtiyaç duyulan akım regülatörlere paylaştırılarak regülatörlerin aşırı ısınması problemi ortadan kaldırılmıştır.

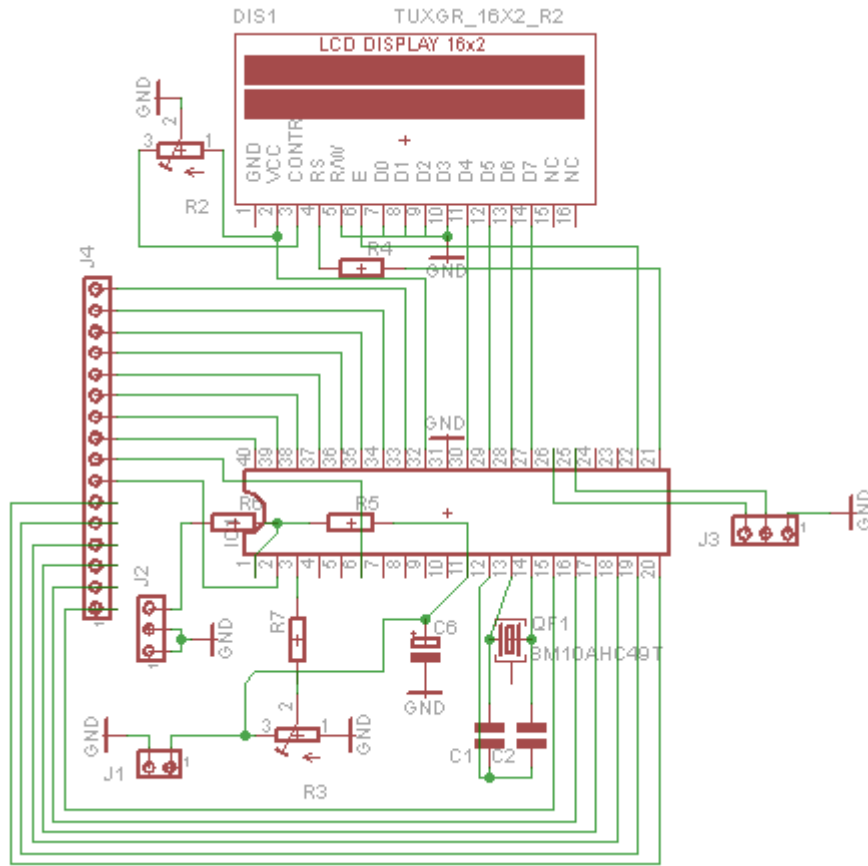
DA-DA dönüştürücünün anahtarlama elemanları olan mosfetlerin sürülmesinde IR2111 entegresi kullanılmıştır. Bu entegrenin Logic 0 seviyesi besleme geriliminden 2 volt aşağısı olup Logic 1 seviyesi ise bunun üstüdür.

Mikrodenetleyici karttan gelen 5V ile bu entegre sürülemeyeceğinden, IR2111 entegresini sürmek için hızlı ve kaliteli bir sürüş sağlayan TC4427 mosfet sürücü entegresi kullanılmıştır.

Motor bobinlerini sürmek için IRF630N mosfetler ikişerli paralel bağlanmıştır. Böylece akımı paylaşarak daha yüksek akımlara karşı dayanıklı hale getirilmişlerdir. Bu mosfetleri sürmek için de TC4427 mosfet sürücü entegre kullanılmıştır. Ayrıca iletimde olan mosfetlerin takibinde kolaylık olması açısından bunlara paralel ledler kullanılmıştır.

#### **4.2.2. Mikrodenetleyici Ünitesi**

Mikrodenetleyici olarak MICROCHIP firmasına ait Pic18F452 kullanılmıştır. Bu denetleyicilerde analog-sayısal dönüştürücü ve pwm ara yüzleri mikrodenetleyicinin içinde bulunduğundan tercih edilmiştir.

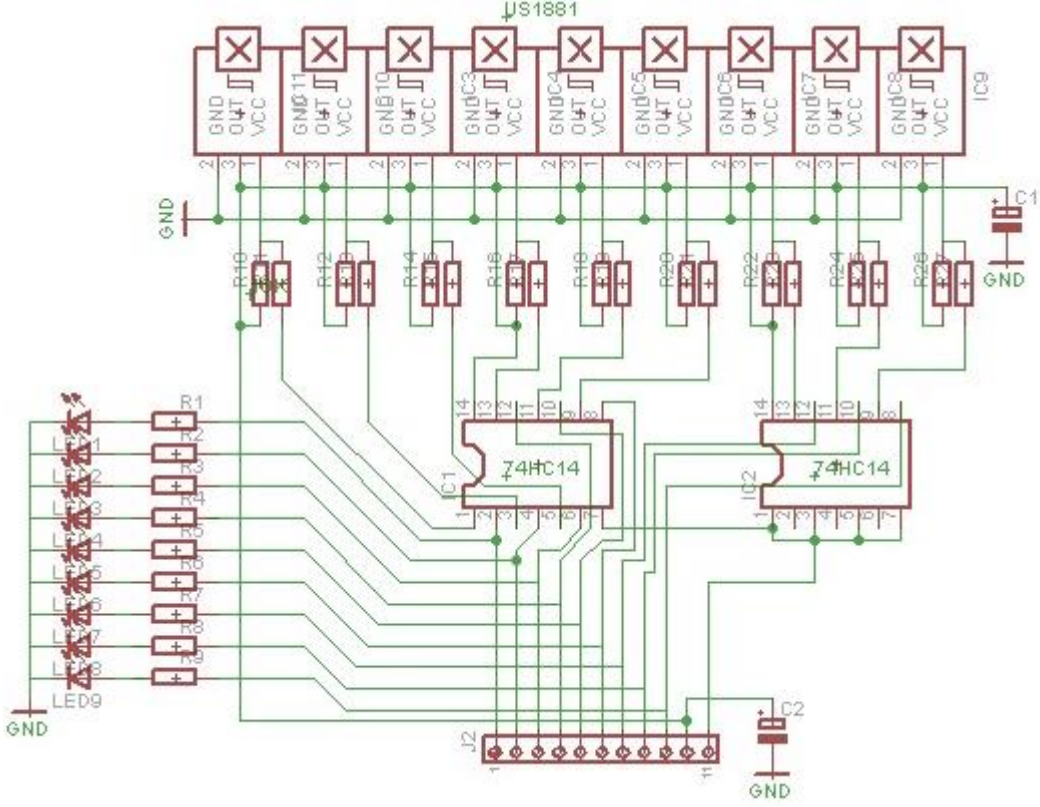


Şekil 4.8. PWM sürücü ünitesi elektrik devre şeması

Mikrodenetleyici ünite, kontrolün kullanıcı modunda veya otomatik yani matlab kontrollü modda gerçekleştirilmesine Sıvı Kristal Ekran(Liquid Crystal Display) aracılığıyla olanak vermektedir. Mod seçimi manuel olarak ayarlı potansiyometre vasıtasıyla yapılmaktadır. Bu mikrodenetleyici ile bilgisayar arasında USB- Serial Convertor üzerinden haberleşme yapılmaktadır.

#### 4.2.3. Algılayıcı (Sensör) Ünitesi

Algılayıcı ünitesi dokuz adet US1881 hall effect sensör (alan etkili algılayıcı) iki adet SN74HC14N Schmitt Trigger entegre, her bir sensörün çalışırılığını gösteren dokuz adet led ve diğer devre elemanlardan oluşmaktadır.



Şekil 4.9. Sensör ünitesinin elektrik devre şeması

Hall effect sensörler genel olarak manyetik alana sahip cisimlerin uzaklığını, pozisyonunu ya da dönüşünü algılayan hassas sensörlerdir. Çoğunlukta yarı iletken malzemeler kullanılarak elde edilmektedirler. Üç bacaklı olarak üretilmektedirler.

Bütün mıknatıslarla (doğal-elektromıknatıs) uyumlu olan bu sensörler fiziksel darbelerle karşı dayanıklılık, sıcaklık değişiminden etkilenmeme gibi üstünlüklere sahiptirler.

Bazı Hall sensörler tek çıkışlı olarak imal edilirken bazı sensörler iki çıkışlı yapılmaktadır. Bu çalışmada iki çıkışlı olanı kullanılmıştır. Hall sensörü ön yüzüne mıknatısın S kutbunun veya arka yüzüne mıknatısın N kutbunun yaklaşmasıyla birlikte ON konumuna gelmektedir. Sensör bir kez ON olduktan sonra, mıknatıs ortamdaki uzaklaştırılsa dahi ON konumunda kalmaya devam etmektedir. Mıknatısın ters kutbu yaklaştırıldığında ise OFF konumuna geçmektedir[44]. Shimitt trigger (tetikleyici) entegre ise sensörlerin ürettiği çıkış sinyalinin kalitesini artırarak kontrol kartına ve ledlere göndermektedir.

Rotor ileri hareketini gerçekleştirirken uç kısmında yer alan mıknatısın manyetik alanına giren sensörler manyetik akıya göre giriş gerilimine bağlı doğrusal değişen bir çıkış



gerilimi üretmekte yani iletken konuma geçmektedirler. Rotor geri hareketinde ise sensörler mıknatısın diğer kutbunun manyetik etkisiyle yalıtkan konuma geçmektedir. Böylelikle iletim-yalıtım durumları ile rotor konumunun dijital olarak okunması gerçekleşmektedir.

### 4.3. Kapalı Çevrim YSA Kontrolünün Gerçeklenmesi

Gerçek zamanlı çalışan ysa ile sistemin kontrolünde rotorun konumuna bağlı olarak ‘X sensörü aktif olursa Y sensörü aktif olana kadar A bobini enerjilenecektir’ şeklinde kural dizilimi oluşturulmuştur.

Tablo 4.1. Bobinlerin enerjilenme sırası kural dizilimi

kuraldizisi[0]	0b00000011
kuraldizisi[1]	0b00000111
kuraldizisi[2]	0b00011111
kuraldizisi[3]	0b00111111

Sistemin kapalı çevrim ‘Kullanıcı Mod’ kontrolünün yapılması durumunda bu kural rotor konumuna bağlı olarak işlemekte, enerjilenen bir bobinden sonra diğer bobinin enerjilenmesi için geçen süre rotorun hızına göre değişmektedir. Sensörlerden gelen veriye göre motor konumu değişmektedir. Yani frekans açık çevrim, rotor ise kapalı çevrim olarak çalışmaktadır.

Kapalı çevrim ‘Otomatik Mod’ kontrolünde bu kural basamakları arasındaki ilerleme süresi sensörlerden gelen bilgiye göre değişmektedir. Hem frekans hem de rotor konumu kapalı çevrim olarak çalışmakta, yani geri bildirimli bir kontrol sağlanmaktadır.

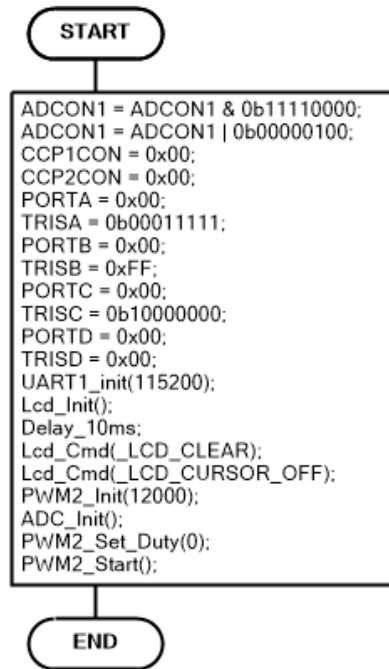
Bu sürenin değişken olması frekansın da değişmesine neden olmaktadır. Süre ne kadar uzarsa frekans da o kadar azalmakta, süre ne kadar kısalsın frekans da o oranda artmaktadır. Frekansı en çok etkileyen faktör ise sisteme uygulanan güçtür. Bobine ne kadar fazla güç verilirse bobinde o kadar fazla manyetik kuvvet oluşmaktadır. Bu manyetik kuvvetin etkisiyle rotor o kadar fazla güçle çekilmektedir. Bu gücü ayarlamak için ise YSA kullanılmıştır. YSA, PWM oranını arttırıp azaltarak kullanıcının belirlediği rotor hareket

frekansını yakalamaya çalışmaktadır. Bu kısımda mikrodenetleyici yazılımı ve çalışmada kullanılan YSA yapısı hakkında bilgi verilecektir.

#### 4.3.1. Mikrodenetleyici Ünitesinde Kullanılan Denetleyicinin Yazılımı

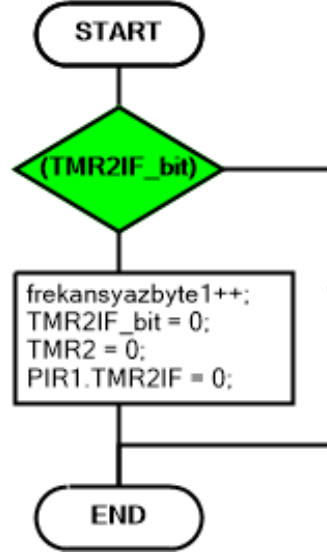
Mikrodenetleyici ünitesinin beyni olan mikroişlemci içerisinde Pic\_setting, interrupt(kesme), motoradım ve main olmak üzere dört ayrı metot bulunmaktadır.

Pic\_setting metodu, sistem ilk çalıştığında main metodunun ilk çağırdığı metottur. Pic\_setting metodunda mikroişlemcinin çalışması için gerekli olan ayarlar yapılmaktadır. Analog portların ayarlanması, kullanılan(PortA, PortB, PortC, PortD) portların girişi-çıkışlarının belirlenmesi, LCD ekranın çalışmaya hazırlanması, DA-DA Dönüştürücü için gereken PWM2 kanalının aktifleştirilmesi işlemleri bu metotta yapılmaktadır.



Şekil 4.10. Mikrodenetleyici “PIC\_Setting (düzenleme)” algoritması

Interrupt metodu, işlemci rutin işlerken Tmr2(Timer-Zamanlayıcı)’nin taşması durumunda çalışmaktadır. Sekiz bitlik bir zamanlayıcı olan Tmr2’nin her taşmasında bir interrupt üretilmektedir. Interrupt kesmesinden elde edilen verilerle süre ve frekans hesaplanmaktadır.



Şekil 4.11. Mikroişlemci “interrupt(kesme)” algoritması

Motoradım metodunda ise sensörler kontrol edilerek gerekli durumlarda motorun bir adım ileri gitmesi, yani bir sonraki işleme gitmesi durumu gerçekleşmektedir. Motorun hangi bobininin enerjilendireceği ise sensörlerden gelen konum bilgisiyle main metodunda belirtilmiş kural dizilimine göre gerçekleşmektedir.

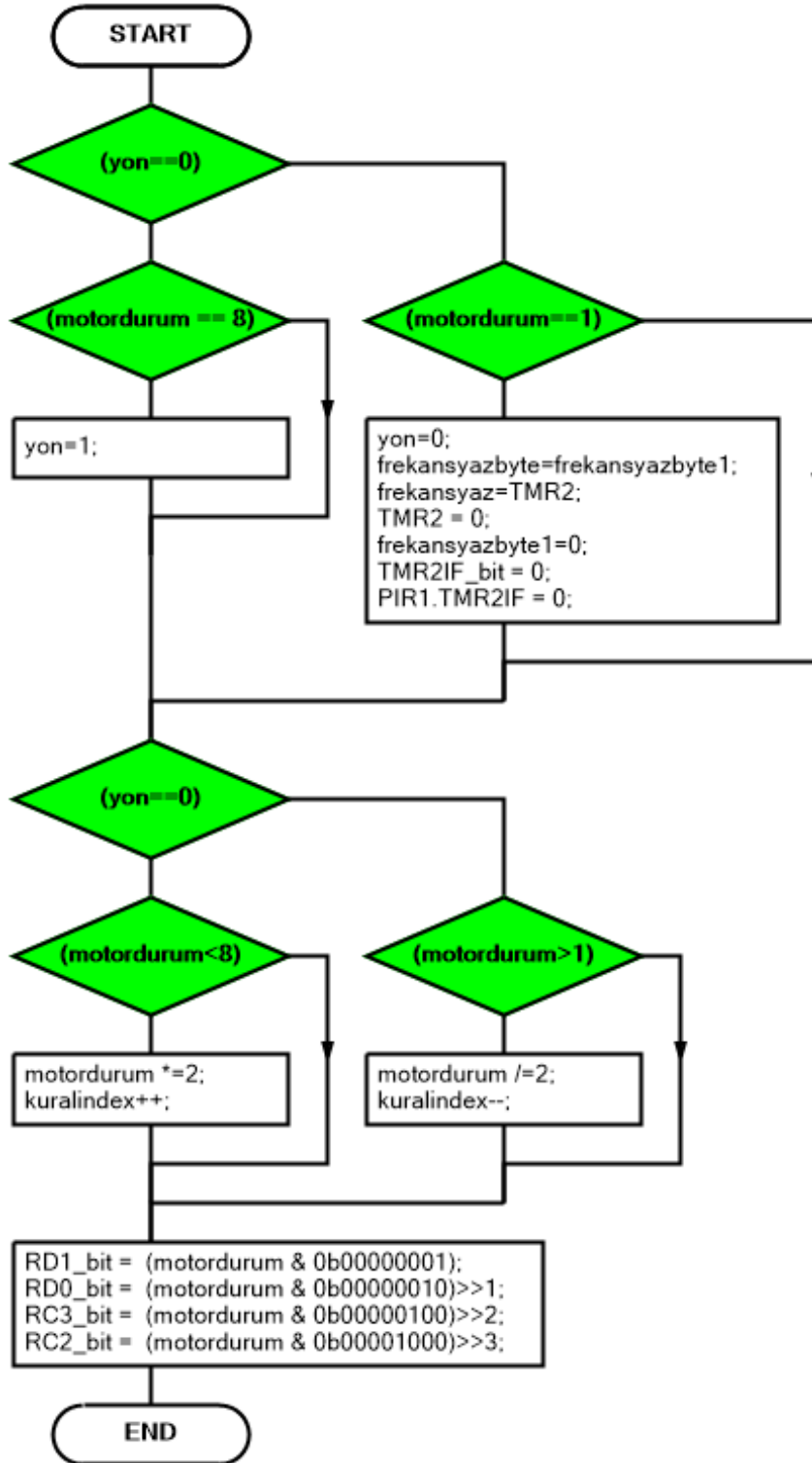
Tablo 4.1’de verilen kural dizilimi için yön=0 ve yön=1 olmak üzere iki yön tayin edilmiştir. Rotorun sensör ünitesine bakan uç kısmında bulunan mıknatıs ile hizasında bulunduğu sensör grubundan okunan veriye göre;

Yön=0 ise ve o an 1.bobin aktif ise 2.bobin enerjilenmekte ve bu işlem 4.bobinin enerjilenmesine kadar bu düzende devam etmektedir.



Şekil 4.12 a) Yön=0 durumunda bobin enerjilenme sırası, b) Yön=1 durumunda bobin enerjilenme sırası

4.bobinin enerjilenmesinden sonra artık yön değişmekte ve yön=1 olarak çalışmaktadır. 4.bobin aktifken ve yön=1 ise 3.bobin enerjilenmekte ve bu işlem de 1.bobinin enerjilenmesine kadar devam etmektedir.



Şekil 4.13 Mikroişlemci “motoradım” düzenleme algoritması

Main metodu; uygulamanın ana metodudur. Uygulama içerisinde diğer metotlar bu metottan çağrılmaktadır. Fakat interrupt metotları hariçtir.

İnterrupt metotları tetiklenmelerini donanımsal ya da yazılımsal başka kaynaklardan almakta, main metodundan bağımsız olarak çalışmaktadır. Bu çalışmada interrupt metodu tetiklemeyi mikroişlemcinin içerisindeki sayaç register'ı olan Tmr2'nin taşmasından almaktadır.

Sistem ilk çalıştığında LCD ekranına menü seçeneği gelmektedir. Menü ekranında ilk etapta kullanıcının potansiyometreyi ortalama bir değere getirmesi beklenmektedir.

Potansiyometre üzerinde analog/dijital çevirici vasıtasıyla okunan değer yaklaşık 2.5V civarındaki aralığa gelmesiyle 'Menü'(kullanıcı seçim ekranı) gelmektedir. Potansiyometre sağa doğru çevrildiğinde 'Kul Mod'(kullanıcı modu), sola doğru çevrildiğinde ise 'Oto Mod'(otomatik çalışma modu) seçilebilmektedir.

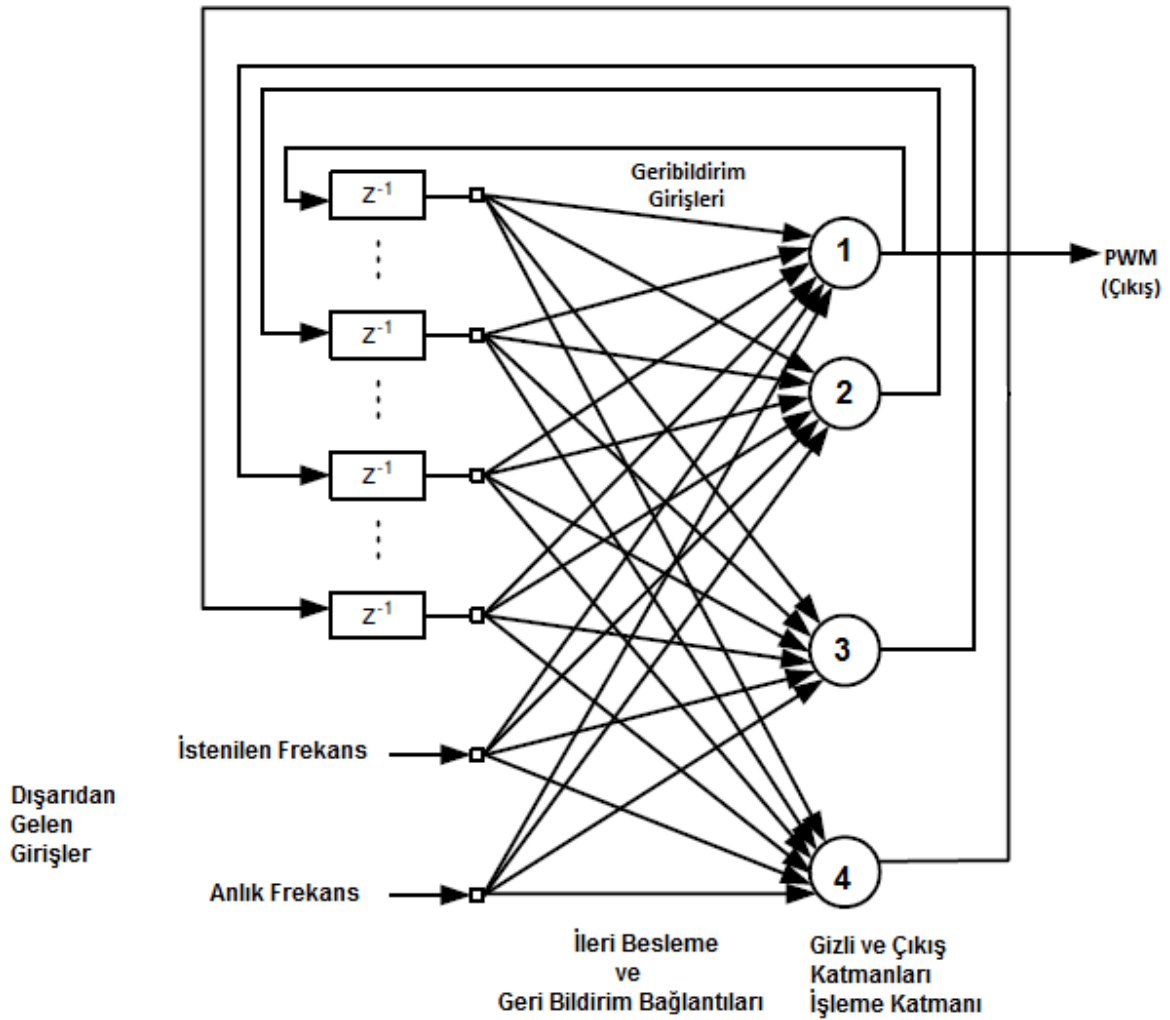
Kullanıcı modunda, sistemin çalışması için bilgisayar bağlantısına ihtiyaç duyulmamaktadır. Sistem, hangi bobinlerin tetikleneceğini sensörlerden aldığı veriye göre kendi yapmakta, yani mikroişlemci içerisine yazılmış olan kural dizilimine göre çalışmaktadır. Kullanıcı yine potansiyometreyi kullanarak DA-DA dönüştürücünün PWM sinyalindeki genliğini değiştirebilmektedir. Bu sayede farklı güç değerlerinde rotor hızındaki değişimleri gözlemleyebilmektedir. Kapalı çevrim rotor konum kontrollü çalışma modu diyebileceğimiz bu modun kullanılmasının nedenlerinden biri ysa denetimli çalışma yapısını matlab ortamında oluşturmada gerekli olan minPwm, maxPwm, gibi çalışma aralıklarını belirlemektir. Bir diğer nedeni ise bilgisayar bağlantısı olmaksızın doğrusal motorun değişken güç aralıklarındaki çalışmasının gözlemlenebilir olmasını sağlamaktır.

Kullanıcı otomatik çalışma modunu seçtiğinde ise yine sistem sensörlerden gelen veriye göre hangi bobinin tetiklenmesi gerektiğine kendi karar vermekte fakat DA-DA dönüştürücüye uygulanacak olan PWM sinyalinin genliği ise matlab yazılımından gelen frekans bilgisine göre seri port vasıtasıyla atanmaktadır.

Genel olarak mikrodenetleyici ünitesi içerisindeki pic18F452'nin içindeki yazılım, sensöre göre hangi bobinin elektriklenmesi gerektiğini belirlemiş olduğumuz kural dizilimine göre ayarlamaktadır. Aynı zamanda frekansı hesaplayarak frekans verisini bilgisayara göndermektedir. Frekans, rotorun bir tam hareketini tamamlamasında geçen sürenin 1 saniyeye bölünmesi ile hesaplanmıştır. Mikroişlemci içerisine yazılmış olan main metoduna ait algoritma şeması Ek 1' de verilmektedir.

### 4.3.1. YSA ile Modelleme

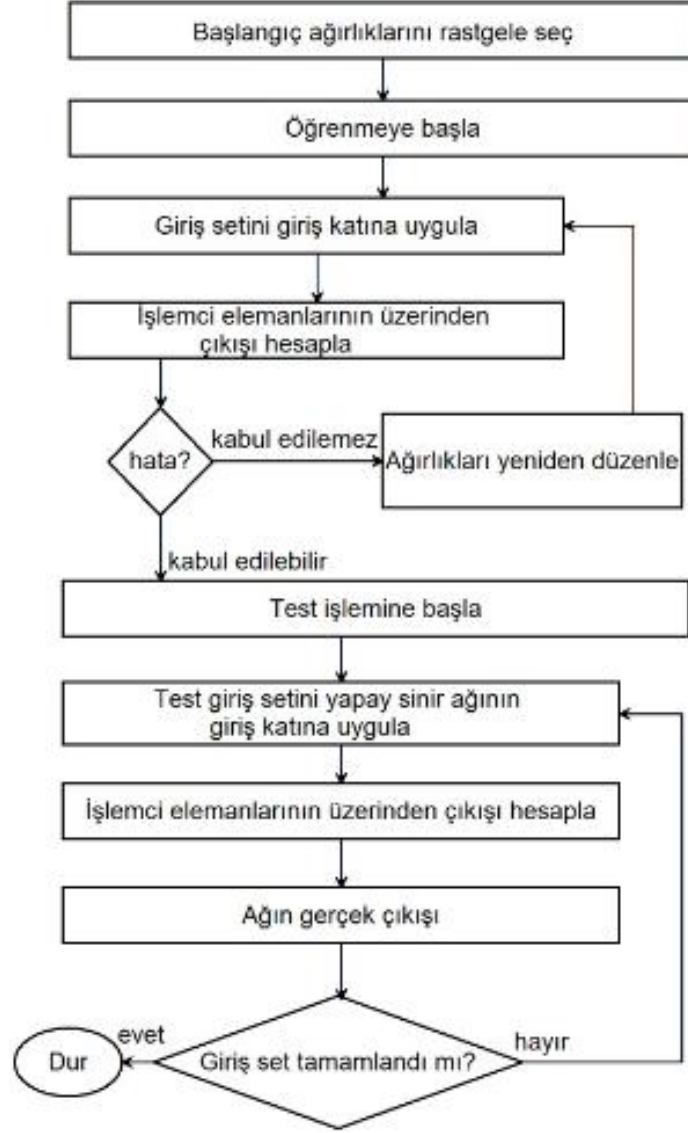
Yapay sinir ağlarında öğrenme tamamen insan zekası gibi edinilen bilgi ve tecrübeye dayanmaktadır. YSA yapısının seçiminde kabul edilmiş herhangi bir kural yoktur. Bu nedenle YSA yapısındaki katman sayısı ve katmanlardaki düğüm sayısı (neuron sayısı) deneme-yanılma yoluyla bulunmaktadır. Bu çalışmada öğrenme algoritmasının bu ağı eğitmede kullanılabilir olması nedeniyle yaygın olarak kullanılan Çok Katlı Perseptron (ÇKP) modeli tercih edilmiştir. Şekil 4.14'de verilen ÇKP modeli, iki giriş, dört hücreli bir ara katman ve bir de çıkış katmanından oluşturulmuştur.



Şekil 4.14. Çok katmanlı ileri beslemeli-geri yayımlı YSA modeli[29]

ÇKP birçok alana uygulanmış olan bir YSA yapısıdır. Bir katmandaki bütün işlem elemanları bir üst katmandaki bütün işlem elemanlarına bağlıdır. Giriş katındaki nöronlar tampon gibi davranırlar ve giriş sinyalini ara kattaki nöronlara dağıtırlar. Ara kattaki her bir nöronun çıkışı, kendine gelen bütün giriş sinyallerini takip eden bağlantı ağırlıkları ile çarpımlarının toplanması ile elde edilir. Elde edilen bu toplam, çıkışın toplam bir fonksiyonu olarak hesaplanmaktadır. Gerçekleştirilen uygulamada, YSA mimarisi olarak çok katmanlı ileri beslemeli sinir ağı, öğrenme tipi olarak gerçek zamanlı öğrenme, öğrenme algoritması olarak da hatayı geri yayma(back propagation) algoritması kullanılmaktadır.

Hatayı geri yayma, genellikle çok katmanlı perseptronların, ağı gizli katmanlarına bağlı olan ağırlıklarını değiştirmek için kullanılan bir öğrenme algoritmasıdır. Geri yayılım algoritması ağırlıkları ters yönde değiştirmek için hesaplanmış hata değerlerini kullanır. Bu veriler hazırlandıktan sonra ağı eğitim kümesinden girdiler sunulur. Bu girdi değerleri gizli katmana iletilir. Gizli katmandaki her bir nöronun çıkışı, kendine gelen bütün giriş sinyallerini takip eden bağlantı ağırlıkları ile çarpımlarının toplanmasıyla elde edilir. Elde edilen bu toplam, çıkışın toplam bir fonksiyonu olarak hesaplanmaktadır. Bu çalışmada sigmoid fonksiyonu kullanılmıştır. Diğer katlardaki nöronların çıkışları da aynı şekilde hesaplanır. Kullanılan eğitime algoritmasına göre, ağı çıkışı ile arzu edilen çıkış arasındaki hata tekrar geriye doğru yayılarak hata minimuma düşüncüye kadar YSA ağı ağırlıklarını değiştirir [38]. Hatayı geri yayma algoritması ile ağı öğrenme aşamasına ait algoritma şeması Şekil 4.15’de verilmiştir.



Şekil 4.15. Geri yayılım algoritmasının ağa uygulanması[35]

Eğitim esnasında hata daima azalmamakta, bazen arttığı da gözlemlenmektedir. Hatanın artması genellikle çözümden uzaklaşıldığını işaret etmektedir. Bu nedenle hata değerinin artması durumunda eğitime son verilmektedir.

YSA'ların doğrusal olmama özelliğinden dolayı verilerin bir normalizasyon işleminden geçmesi gerekmektedir. Normalizasyon işlemi YSA logaritmik(YSA LOG) fonksiyonu için  $[0 - 1]$ , YSA tanjant(YSA TAN) fonksiyonu için  $[-1,0 - 1,0]$  aralığında yapılmaktadır. Bu çalışmada normalizasyon işleminde YSA logaritmik fonksiyonu kullanılmıştır. Logaritmik fonksiyonun matematiksel ifadesi 4.1'de verildiği gibidir.

$$F(x) = 1/[1 + e^{-x}] \quad (4.1)$$



Yapay sinir ağların eğitimi, ham veri setine normalizasyon yöntemi uygulanmadan çok yavaş olabilir. Çünkü YSA girdileri arasında aşırı büyük veya küçük değerler görülebilir ve Bunlar yanlışlıkla girdi setine girmiş olabilir. Bu durumda net girdiler hesaplanırken bu değerler aşırı büyük veya küçük değerlerin doğmasına neden olarak ağı yanlış yönlendirebilirler. Bütün girdilerin belirli aralıkta (çoğunlukla 0-1 aralığında) ölçeklendirilmesi hem farklı ortamlardan gelen bilgilerin aynı ölçek üzerine indirgenmesine hem de yanlış girilen çok büyük ve küçük şekildeki değerlerin etkisinin ortadan kalkmasına imkan sağlar.

Literatürde normalleştirme işlemlerinde kullanılan farklı teknikler mevcuttur. Bunların başlıcaları Min kuralı, Max kuralı, Medyan, Sigmoid olarak sıralanabilir. Bu çalışmada 'Min-Max Normalizasyonu' yöntemi kullanılmıştır. Veri kümesinin [0 1] arasında bir ölçeklendirmeye tabi tutulabilmesi için o kümenin  $X_{min}$  - $X_{max}$  aralığı 4-80 olarak tespit edilmiş olup 4.2'deki eşitliğe göre ölçeklendirme yapılmıştır.

$$X_{yeni} = (X - x_{min}) / (x_{max} - x_{min}) \quad (4.2)$$

Bu eşitlikte;

$X_{yeni}$  = Normalize edilmiş veriyi,

$X$  = Girdi değerini,

$X_{min}$  = Girdi seti içerisinde yer alan en küçük sayıyı,

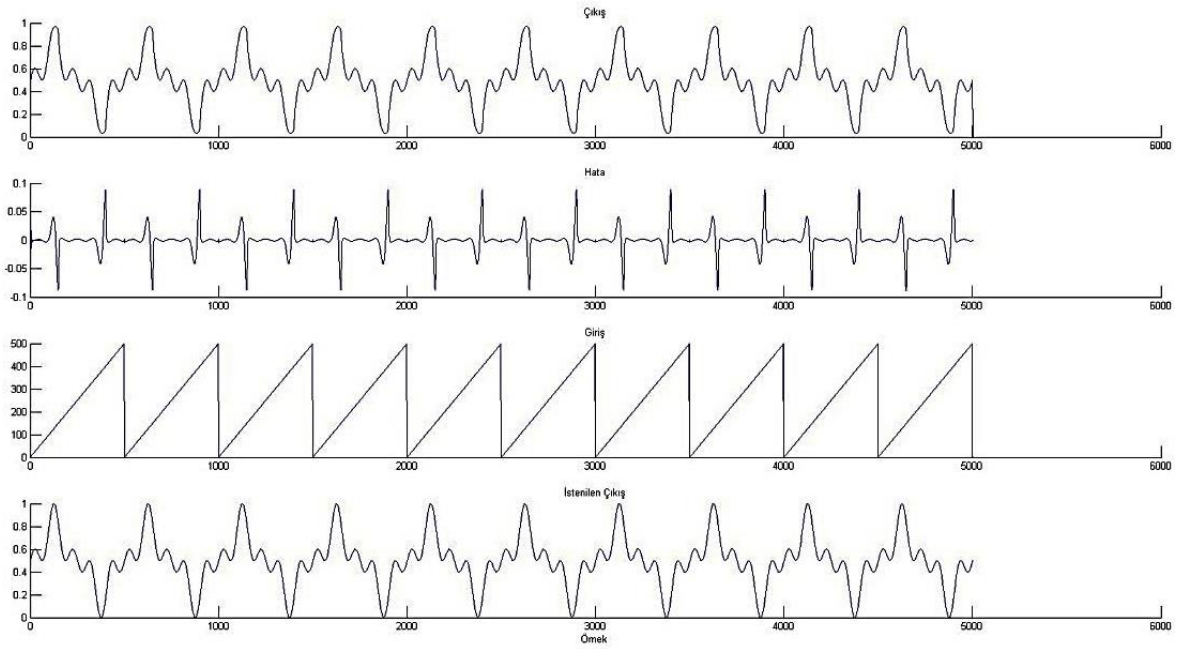
$x_{max}$  = Girdi seti içerisinde yer alan en büyük sayıyı,

ifade etmektedir [22].

### 4.3.3. MATLAB Ortamında YSA Denetiminin Programlanması

MATLAB (MATris LABORatuvarı) grafik çizimlerdeki kolaylıklar, mühendislik ve fen uygulamalarının hemen her sahasında kullanılabilecek hazır fonksiyonlar, kullanıcıya programlama imkanı sağlaması, programlamanın kolay olması ve harici program ve dosyalarla çalışabiliyor olması gibi üstünlüklere sahiptir [15]. Bu üstünlüklerinden dolayı gerçekleştirilen sistemin YSA denetimli çalışmasını sağlamak için MATLAB programı kullanılmıştır.

YSA kontrollü çalışma tasarımında en önemli nokta, kontrol düzeneğine uygun ağ yapısını belirlemektir. Gerçek zamanlı deneysel ortamda çalışabilecek uygun yöntemin bulunması için bilgisayar ortamında çalışmalar yapılmış, bu çalışmalar sonucu uygun görülen ağ yapısı gerçek zamanlı çalışan sistemin kontrolünde kullanılmıştır.



Şekil 4.16. Geliştirilen YSA' nın test sinyali ile eğitime ait grafik

Geliştirilen YSA denetleyici yapısını test etmek amacıyla ağın girişine testere dişli sinyal uygulanıp eğitim sinyali olarak Şekil 4.16'da gösterilen istenilen çıkış sinyali kullanılmıştır. Ağın çıkışı ve hata sinyalleri şekil üzerinde gösterilmiştir. Grafikten edinilen bilgilerden ağın öğrenme kabiliyeti olduğu kabul edilmiş ve gerçekleştirilen sisteme uygulanmıştır.

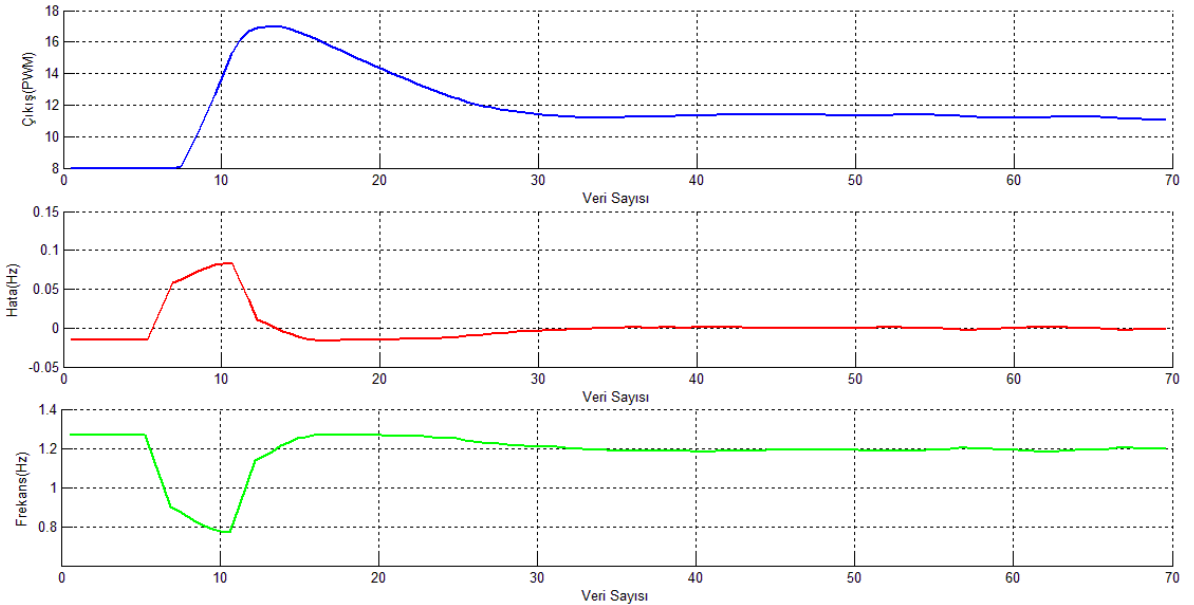
Sisteme enerji verildiğinde mikrodenetleyici ünitesi kullanıcının menü seçimi yapmasını beklemektedir. YSA denetimli çalışma için LCD ekran üzerinde görülen 'OTOMOD' yani otomatik çalışma modu seçilir. Sistem ilk olarak mikrodenetleyici yazılımı vasıtasıyla sensörlerden gelen veriye göre enerjilenmesi gereken bobini tetikler. Bobinlerin besleme gerilimi DA-DA dönüştürücüye uygulanacak olan PWM değeriyle ayarlanmaktadır. Bu değer Matlab ortamında geliştirilmiş olan yazılım ile usb seri dönüştürücü üzerinden mikrodenetleyiciye aktarılmaktadır. Mikrodenetleyici doğrusal hareketli motorun anlık çalışma frekansını seri port üzerinden Matlab'a aktarmakta, bu veri

kullanıcı tarafından belirlenen ‘istenilen frekans’ değerini kullanarak hata verisini üretmektedir. Üretilmiş olan hata verisi ağırlık eğitimi için kullanılmaktadır. YSA hataya göre geri dağılım algoritmasına bağlı olarak hücrelerin ağırlıklarını değiştirmekte, yani öğrenmeyi gerçekleştirmektedir. YSA ilk çalışmasında rastgele ağırlıklar atamaktadır. YSA’lar hangi yapıda olursa olsun eğitilirken ilk olarak rasgele bir ağırlık değeri ile çalışmaktadır. Rotor ilk turunu gerçekleştirdikten sonra ysa öğrenmeye başlamaktadır.

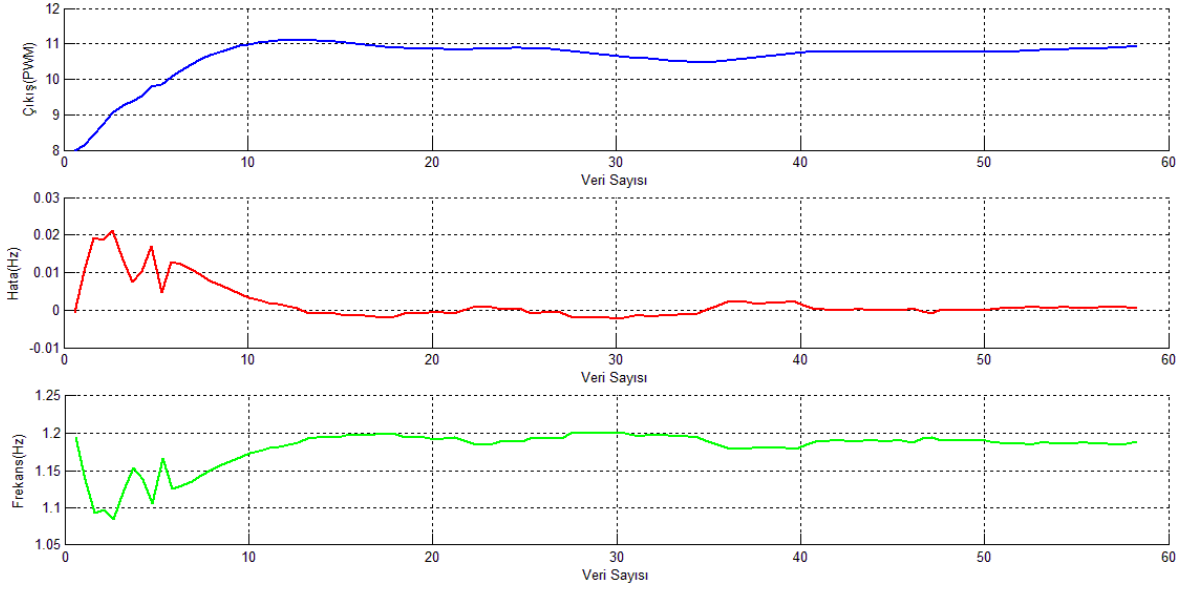
Denetimde kullanılan YSA’nın girişi olarak sistemin göndermiş olduğu frekans değeri ile kullanıcı tarafından istenen frekans değeri kullanılmaktadır. Aktivasyon fonksiyonu olarak Sigmoid Fonksiyonu kullanılmıştır. Tanjant Fonksiyonu da denemiş, benzer sonuçlar elde edilmiştir. Fakat daha geç sonuç verdiği için Sigmoid Fonksiyon tercih edilmiştir. YSA ’nın çıkışı ise sisteme uygulanacak PWM değerini vermektedir.

Yapay sinir ağlarında bir diğer önemli husus da eğitimin ne kadar süreceğidir. Eğitime işlemi iki şekilde tamamlanabilir; belli bir değere kadar hata toleransı kullanmak ya da eğitim sayısı belirlemek. Gerçeklenen sistemde hata toleransı kullanılmadığından programın çalışması durdurulana kadar ağ eğitime devam etmektedir.

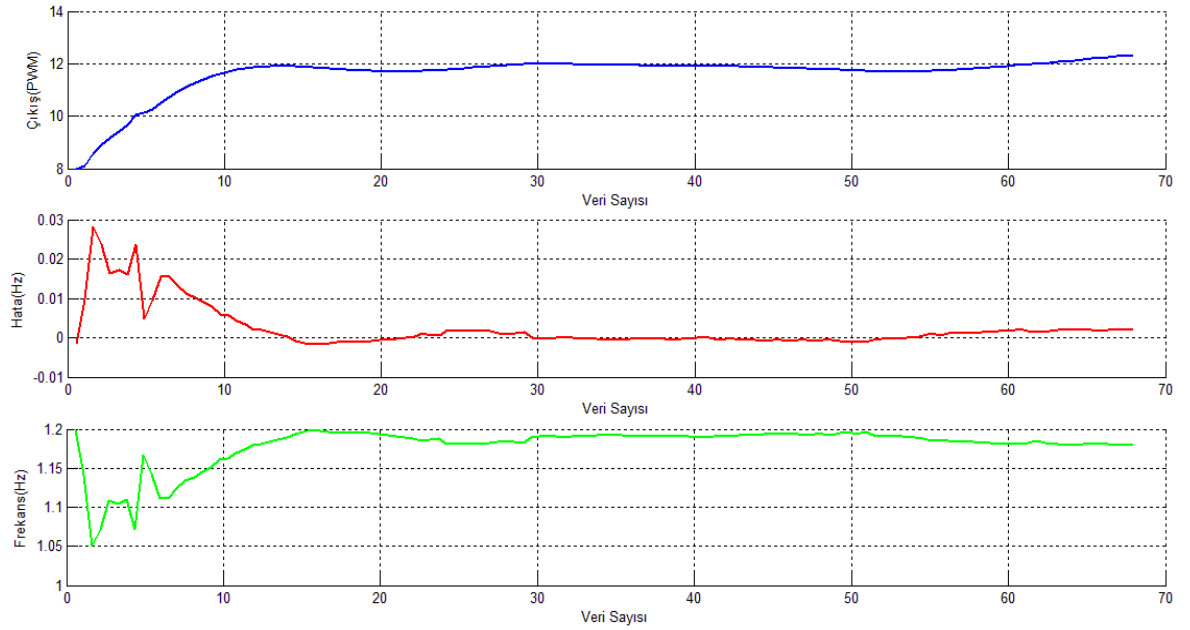
Gerçek zamanlı olarak çalışan sistemin farklı hücre değerlerindeki ysa denetim verilerine ait grafikler Şekil 4.17-4.18-4.19’da verilmektedir.



Şekil 4.17. İstenen\_frekans = 1.19, nu = 4 için elde edilen çıkış-hata-frekans grafiği



Şekil 4.18. İstenen\_frekans = 1.19, nu = 24 için elde edilen çıkış-hata-frekans grafiği



Şekil 4.19. İstenen\_frekans = 1.19, nu = 48 için elde edilen çıkış-hata-frekans grafiği

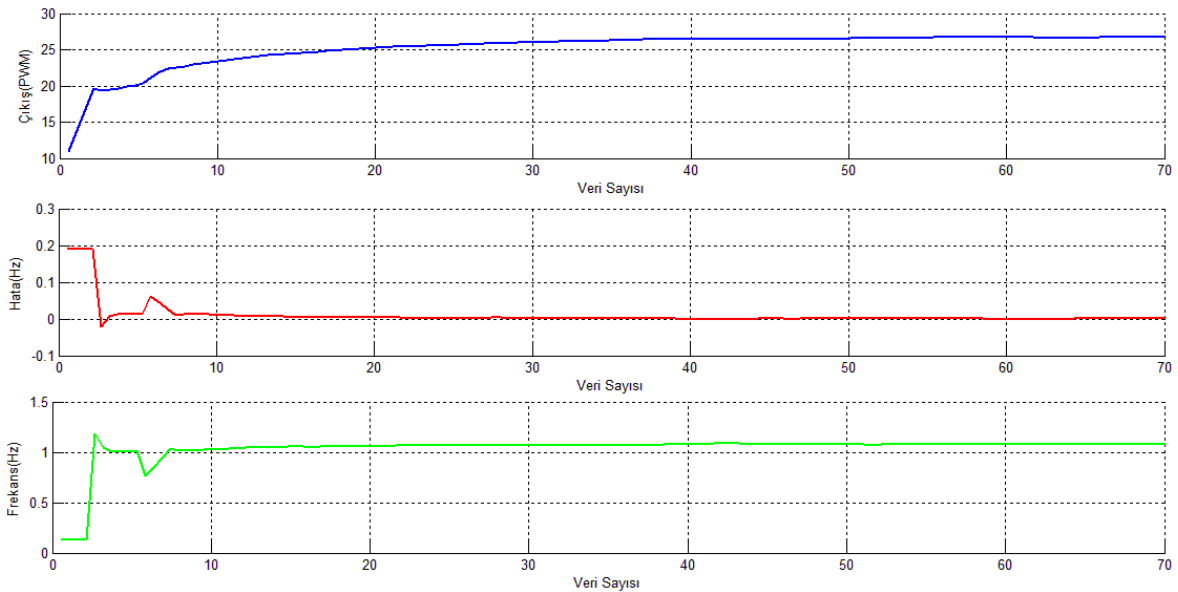
Şekil 4.17-4.18 ve 4.19'da üç farklı parametre için grafik görülmektedir. İlk parametre 'Çıkış' sistemin istenen frekans değerine ulaşması için gereken PWM tetikleme değerini vermektedir. İkinci parametre olan 'Hata' ise istenen frekans ile sistemin anlık frekans değeri arasındaki farkı vermektedir. Son parametre 'Frekans' ise sistemin anlık çalışma frekans bilgisidir. Sistemin denetimi için anlık frekans ile istenen frekans

arasındaki fark sifira indirgenecek şekilde ağ ağırlıkları güncellenmekte ve çıkış PWM üreterek sistemin anlık frekansının istenilen frekansa ulaşması sağlanmaktadır.

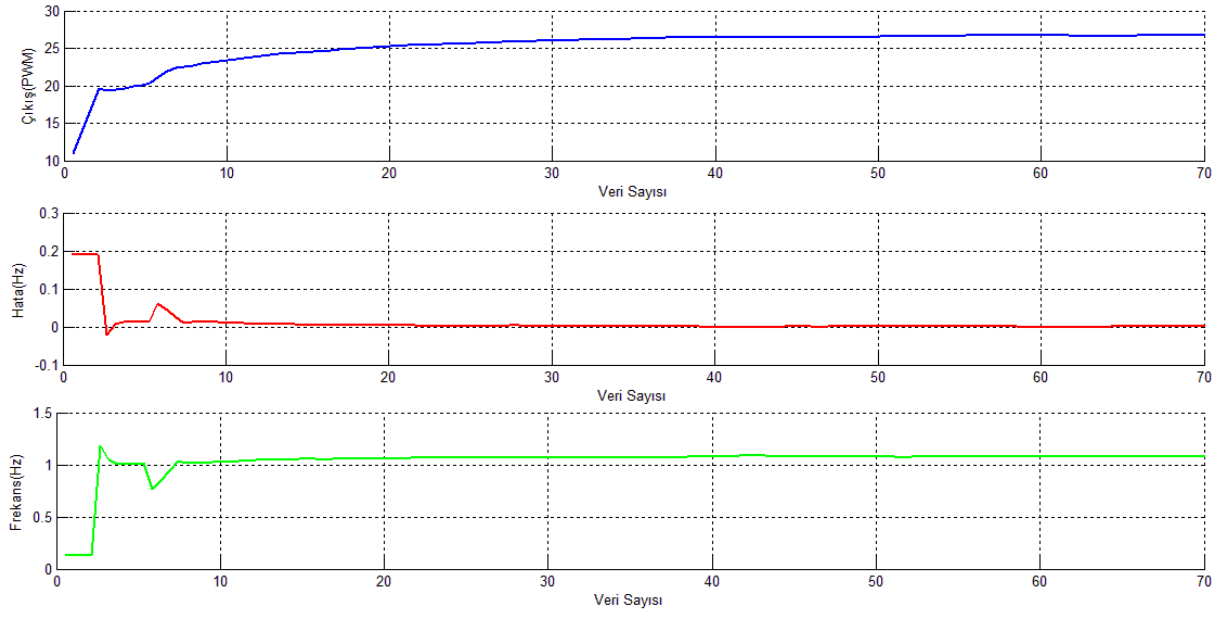
Grafiklerinden görüldüğü üzere yaklaşık yirmi örneğe kadar çıkış sinyali hızla yükselmektedir. Bu durumun nedeni yukarıda bahsedildiği üzere kullanılan ysa modelinin gerçek zamanlı öğrenmeye dayalı olmasıdır. Önceden edinilmiş bilgi olmadığından ağ rastgele atanmış ağırlık değerlerine göre sisteme çıkış verisi göndermektedir. Sistem çalışmaya devam ettiği sürece geliştirilen YSA, sistemin göndermiş olduğu frekans değerine bağlı olarak çıkarım yapmakta ve ağırlık değerlerini kullanılan öğrenme algoritması ile en uygun çıkışı elde edebilecek şekilde yeniden düzenleyerek öğrenmeyi gerçekleştirmektedir. Ağın öğrenmesi ortalama otuz veriden sonra gerçekleştirmektedir.

Şekil 4.17-4.18 ve 4.19’ de ‘istenilen frekans’ değeri 1.19 olarak belirlenmiş olup farklı ara katman hücre sayılarına göre çalışmasına ait çıkış grafikleri alınmıştır. Bu sistemde en iyi sonuç  $nu=4$  alındığı zaman elde edilmiştir. Diğer grafiklerde ise ‘nu’ değeri arttıkça öğrenme daha geç sürede gerçekleşmekte ve sistem daha az karalı çalışmaktadır.

$Nu=4$  iken farklı ‘İstenilen\_frekans’ değerleri için ağ eğitime ait grafikler Şekil 4.20-4.21’de verilmektedir.



Şekil 4.20. İstenilen\_frekans = 1.08,  $nu = 4$  için elde edilen çıkış-hata-frekans grafiği



Şekil 4.21. İstenen\_frekans = 1.24, nu = 4 için elde edilen çıkış-hata-frekans grafiği

Grafiklerden görüldüğü üzere özgün olarak geliştirilen SMTYDHDAM, YSA denetim algoritması yardımıyla gerçek zamanlı olarak kullanıcının istediği frekans değerlerine kısa sürede ulaşmayı başarmıştır.

## 5. SONUÇ

Bu tez çalışmasında özgün olarak gerçekleştirilen doğrusal hareketli, tüpsel yapılı, sürekli mıknatıslı doğru akım motorunun yapay sinir ağları ile denetimli olarak çalışması araştırılmıştır.

Kullanıcı modu kontrolü ile kullanıcı potansiyometre aracılığı ile da-da dönüştürücüye uygulanan PWM sinyalinin genliği değiştirilebilmektedir, böylelikle motor frekansına sisteme uygulanan gücün etkisi gözlenmiştir.

Otomatik mod kontrolü ile sistem seri port üzerinden kontrolü gerçekleştirilebilmektedir. Bu çalışmada MATLAB üzerinde Gerçek Zamanlı Hatayı Geri Yayımlı Yapay Sinir Ağı (GZHGYSA) ile kontrol gerçekleştirilmiş ve ağın öğrenme süresinin ara katman hücre sayısına bağlı olarak değişmekte olduğu gözlenmiştir. Gerçeklenen sistemde en iyi öğrenme dört hücreli ağ modeli ile elde edilmiştir. Bu sistemin GZHGYSA ile kontrolünün gerçekleştirilebileceği görülmüştür.

Böylelikle 'Otomatik Mod'da sisteme uygulanan pwm değerinin matlab ortamında veya herhangi bir yazılım ile ayarlanabilmesinin farklı kontrol algoritmalarının test edilip gözlemlenmesinde faydalı bir araç olarak kullanılabilmesi öngörülmektedir.

Uzun süreli çalışmada statorda bobin sıcaklığının artışı, sıcaklıkla bakır kabloların dirençlerinin arttığı ve üzerinden geçen akımın azaldığı gözlemlenmiştir. Böylelikle sıcaklığın motor çalışma frekansını düşüren bir etki olduğu tespit edilmiştir.

Rotor tasarımı aşamasında tek ve blok mıknatısların manyetik alan içerisindeki davranışlarındaki farklılıklar deneysel olarak gözlemlenmiş ve verimlilik açısından yapıyı şekillendirecek bilgiye ulaşılmıştır. Tek bir mıknatıstan oluşan rotor belirtilen kural dizilimine her iki kontrol yöntemiyle uyumlu çalışmaya adapte olmuştur. Blok mıknatıslı gövde ise bir bütün mıknatıs gibi davranmayıp gövde boyunca N-S,N-S manyetik alan oluşturduğundan enerjilenen bobin grubunun bir kutbu tarafından itilirken diğer kutbu tarafından çekildiğinden hareketsiz kalmıştır. Yani rotor manyetik alanı, stator manyetik alanı tarafından kenetlenmiştir. Bu gözlemler kullanıcıya rotor gövde yapısı ve manyetizma hakkında fikir vermektedir.

Stator tasarımı aşamasında denenmiş olan gövde modelleri de gövde uzunluğu ve bobin kesitindeki farklılıklar hakkında fikir vermiştir. Tek bir bobin sarımı kullanıldığında rotor hızla bir yönde itilmekte ve geri çağırılana kadar yuvadan fırlamaktadır. Bu durum

rotorun çok daha uzun yapılması ve anahtarlamının da daha hızlı olmasını gerektirmektedir. Bu yapı maliyeti arttıran ve denetimi zorlaştıran bir tasarımıdır. Bir başka gövde yapısı olarak birbirinden uzak sabitlenmiş iki sargı grubu denenmiş ve rotorun ileri geri hareketi sağlanmıştır. Bu yapıda rotora etki eden manyetik kuvvetin stator ve rotor arasındaki uzaklığın karesi ile ters orantılı olması nedeni ile verimin düşük olduğu görülmüştür.

Diğer bir gözlem ise kullanılan farklı iletken kesitli gövdelerin davranışları üzerine olmuştur.  $0.30\text{mm}^2$  kesitli gövde yapısında  $0.8\text{A}$  akım çekerken,  $0.40\text{mm}^2$  kesitli gövde ise yaklaşık  $3\text{A}$  akım çektiği, fakat  $0.30\text{mm}^2$  kesitli gövdeye göre manyetik itme kuvvetinin daha büyük olduğu gözlemlenmiştir.

Bilgisayar benzetimi ortamında denetimin, gerçek zamanlı çalışmadaki denetime nazaran ideale daha yakın olarak gerçekleştiği gözlemlenmiştir.

Gerçekleştirilmiş olan sistemin, SMTYDHDAM' un çalışma yapısının öğretilmesinde eğitim kurumlarında ders materyali olarak kullanılabilir kabiliyette olduğu görülmüştür.



## 6. ÖNERİLER

- Bu çalışma kullanıcı tarafından seçilebilen kurallar, ayarlanabilen güç gibi özellikleri sayesinde farklı çalışma prensiplerinin anlaşılmasını kolaylaştırma açısından kullanıcıların doğrusal hareketli rotorlu bir sistemin kontrolünün nasıl gerçekleşeceği konusunda kendilerinin de deneyerek öğrenebileceği bir eğitim materyali olarak kullanılabilir.
- Çalışmada kullanılan rotor gövdesi yerine aynı uzunluğa sahip bir bütün mıknatıstan oluşturulan rotor kullanılarak çalışması gözlemlenebilir. Her iki yapıda da benzer sonuçlar elde edileceği gözlemlenecektir.
- Stator kısmını oluşturan bobin blokları arttırılarak itme-çekme kuvveti daha da arttırılabilir. Bu durumda elektronik devre tasarımını değiştirmek gerekecektir.
- Rotor konum bilgisi için kullanılan sensörler bobin bloklarının içerisine gömülerek de kullanılabilir. Bu tarz bir tasarım kontrolü için yazılımda küçük bir değişiklik yapmak gerekir. Geçiş işlemini rotorun sensörlerin bulunduğu konuma gelmesi olarak ayarlamak yeterli olacaktır.
- Mikrodenetleyicinin çalışma frekansının daha yüksek olması kontrolün daha kararlı bir şekilde çalışması sağlayacaktır.
- Rotor gövdesi ile stator arası uygun bir yağ uygulandığı takdirde çalışma frekansına olumsuz yönde etki eden sürtünme faktörü azaltılarak daha ideal bir çalışma sağlanacaktır.

## 7. KAYNAKLAR

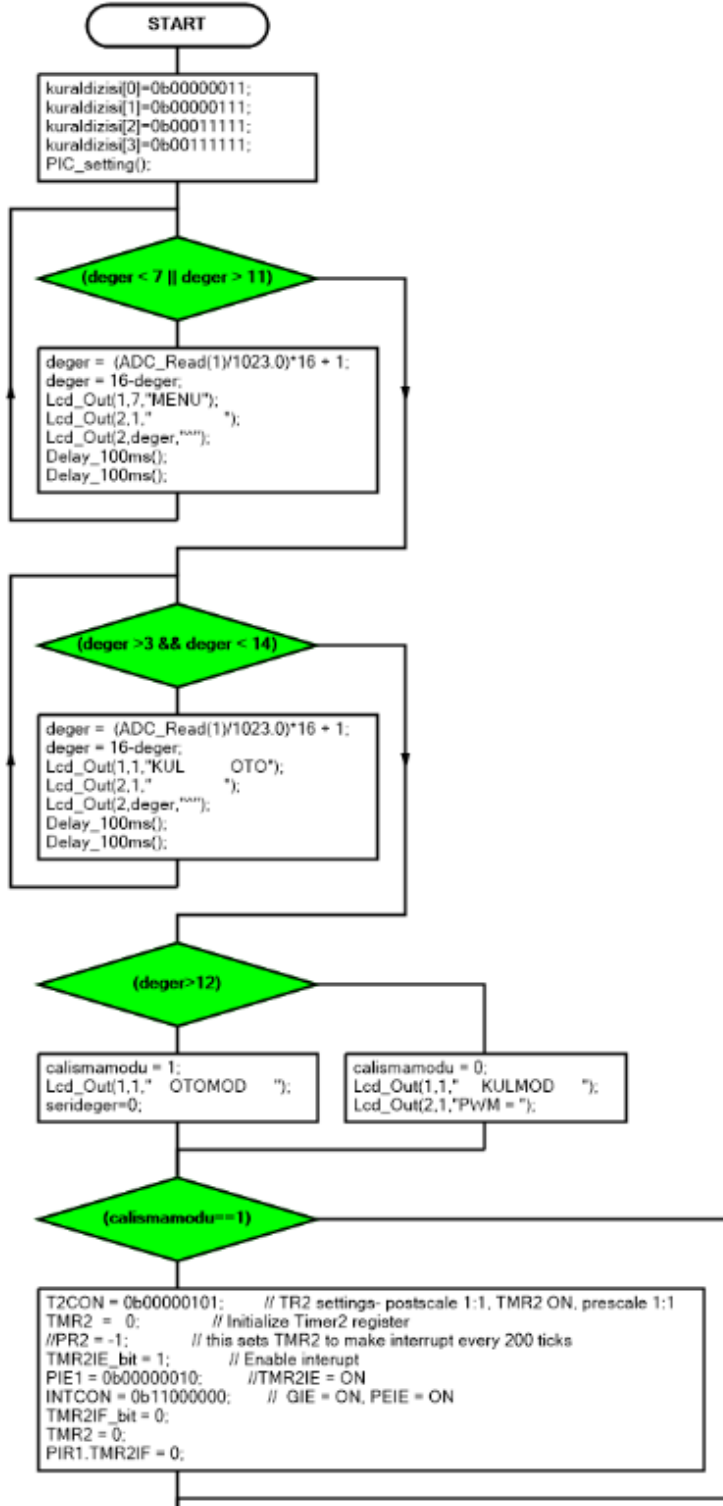
1. Akpınar, A.S., Doğrusal Hareketli Elektrik Makinaları Yüksek Lisans Ders Notları, KTÜ, 2004.
2. Basak, A., Permanent Magnet DC Linear Motors, 1994.
3. Bishop, M.,C., Neural Networks for Pattern Recognition, Clarendon Press. Oxford, 1995.
4. Bodur, H., Güç Elektroniği, Birsen Yayınevi, İstanbul, 2010.
5. Brown R.H. ve Jaroudi M., Torque Prediction and Maximization strategies for Bifilar-Wound Hybrid Step Motors, IEEE Trans. On Power Electr, PE-7, 3(1992) 535-541.
6. Chevailer S., Comparative Study and Selection Criteria Of Linear Motors, Doktora Tezi, Ecole Polytechnique Federale De Lausanne Presentee, la faculte Sciences Et Techniques de L'ingenieur, Suisse, 2006.
7. Clark, R. E., Simith, D. S., Mellor, P.H. and Howe, D., Design optimization of moving-magnet actuators for reciprocating electromechanical systems, IEEE Trans. Magn., 31(1995) 3746-3748.
8. Curram, S.P. and Mingers, J., Neural Networks, Decision Tree Induction and Discriminant Analysis: An Emprical Comparison, The Journal of the Operational Research Society, 45, 4(1994) 440-450.
9. Dixon, T., "Linear Induction Motors: An Alternative Desing For Use In Broader Applications", Graduation Project, Oregon State University, 6-7, 2004.
10. Ebihara D. and Kuno Y., The Development of Claw Pole Linear Pulse Motor With Permanent Magnet, IEEE Transactions on Magnetics, MAG-25, 5(1989) 3919-3921.
11. Emekci, M., Otomatik Açılıp Kapanan Kapılar İçin Tüp Tipi Lineer İndüksiyon Motor Tasarımı, Yüksek Lisans Tezi, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Ankara, 2006.
12. Elmas, Ç., Yapay Zeka Uygulamaları, Seçkin Yayıncılık, 2011.
13. Gu, Q. and Gao, H., Effect of slotting in PM electrical machines, Electric Machines And Power Systems., 10, 2 (1985) 273-284.
14. Gupta, R., Singh, H.P. ve Bailer- Jones, C.A.L., An Introduction to Artificial Neural Networks, Narosa Publishing House, New Delhi, India, 2001.

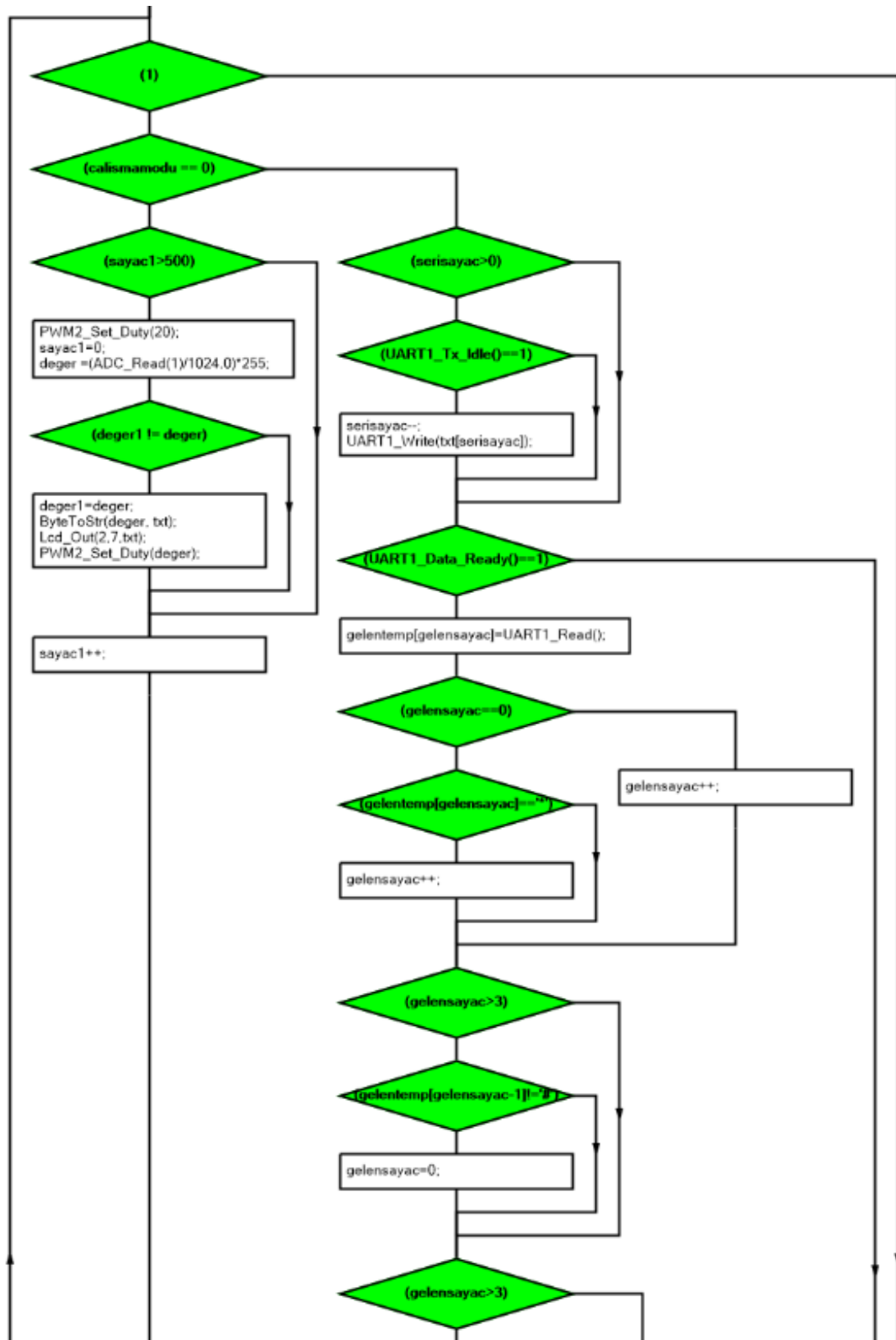
15. Gündoğdu, Ö., Kopmaz, O. ve Ceviz, M., Mühendislik ve Fen Uygulamalarıyla MATLAB, Nobel Yayın Dağıtım, Ankara, 2004.
16. Gürdal, O., Algılayıcılar ve Dönüştürücüler, Ankara, 2000.
17. Gürdal, O., Elektrik Makinalarının Tasarımı, Atlas Yayınları, İstanbul, 2001
18. Gürleyük, S.S, Doğrusal Hareketli Sürekli Mıknatıslı Tüpsel Yapılı Adım Motorunun Dinamik Davranışının İncelenmesi ve Bulanık Mantık Temelli Denetimi, Doktora Tezi, Fen Bilimleri Enstitüsü, KTÜ, Trabzon, 1997.
19. Haykin, S., Neural Networks, A Comprehensive Foundaiton, Second Edition, McMaster University Hamilton, Ontario, Canada, 1999.
20. He, Q., Neural Network and Its Application in IR, Graduate School of Library and Informaiton Science University of Illinois at Urbana- Champaing ,Spring, 1999.
21. He, Y., Zhu, Z.Q. and Howe, D. “A PWM controlled linear servo system for friction wiewlding,” in Proc. 2nd Int Conf. Elect. Mach. CICEM’95, Hangzhou, China, 2, 3(1995) 51-55.
22. Jayalakshmi, T., Santhakumaran, A., “Statistical Normalization and Back Propagation for Classification”, International Journal of Computer Theory and Engineering, 3, 1, 2011.
23. Karaçar, E., Hidrolik ve Pnömatiğe Alternatif Çözüm; Doğrusal Motorlar, II. Hidrolik Pnömatik Kongresi ve Sergisi, Kasım 2001, İzmir, Bildiriler Kitabı, 487-493.
24. Kaynak, O. ve Efe, Ö., Yapay Sinir Ağları ve Uygulamaları, Boğaziçi Üniversitesi Yayınevi, 2000.
25. Kim, M.H., Simoes, M.G. ve Bose, B.K., Neural Network-Based Estimation of Power Electronic Waveforms, IEEE Transactions on Power Electronics, 2, 1996.
26. Küçükali, M., Doğrusal Hareketli Asenkron Motor Sürücülü Mekanik Osilatör ve Evirici ile Kontrolü, Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü, KTÜ, Trabzon, 2008.
27. Lacher, R.C., Artificial Neural Networks, An Introduction to the Theory and Practice, Florida State University, 2009.
28. Laithwaite, E.R. and, Nasar, S.A., Linear Motion Electrical Machines, Proc. IEEE, 16 (1970) 531-539.
29. Mandic, D. and Goh, L.S., A Complex-Valued RTRL Algorithm for Recurrent Neural Networks, Massachusetts Institute of Technology, 2004.
30. Medsker, L.R. ve Jain, L.C., Recurrent Neural Networks Design and Applications, CRC Press, 2001.

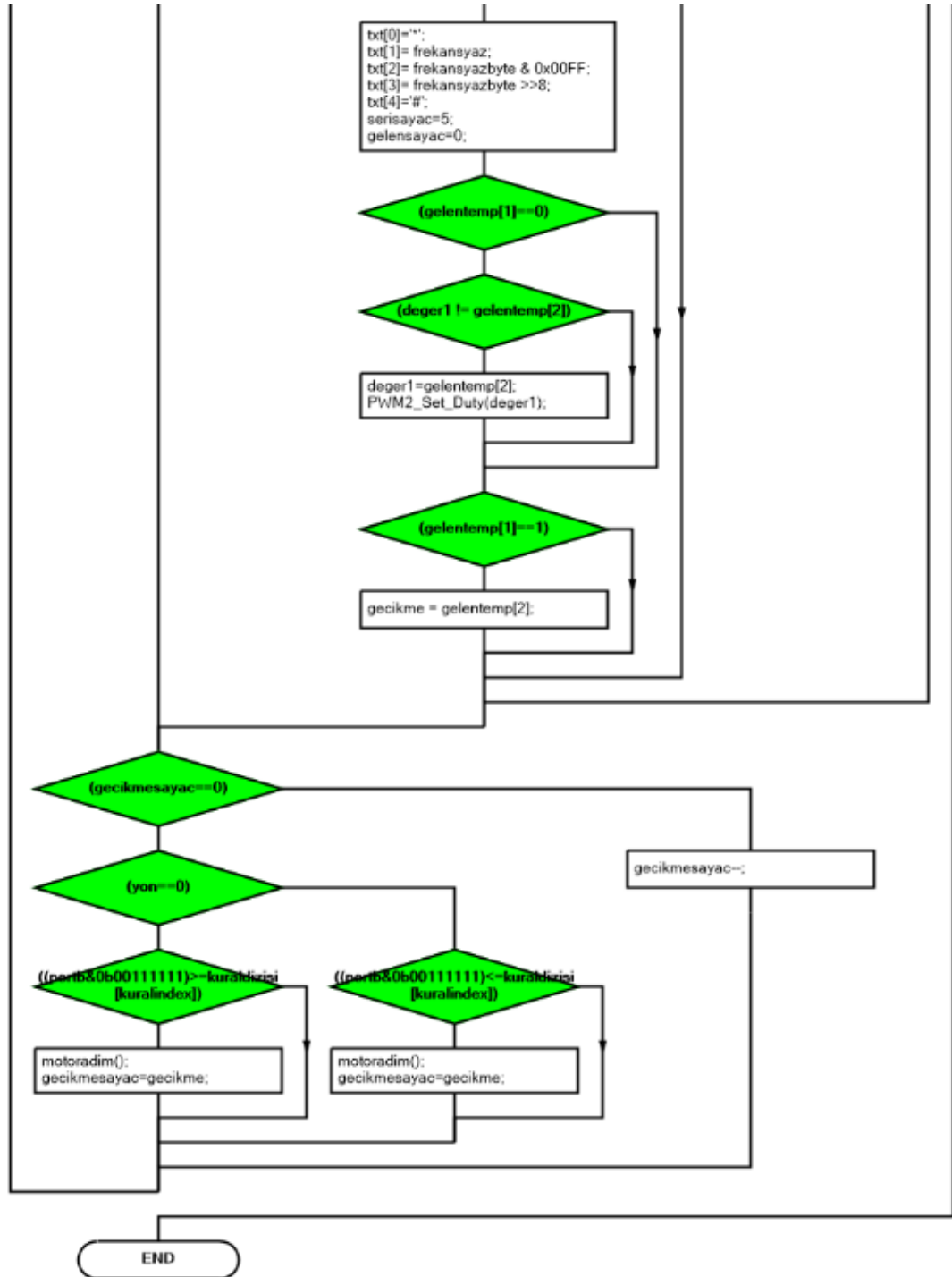
31. Nasar, S.A. and Boldea, I., Linear Motion Electric Machines , John Wiley & Sons , New York, 1976.
32. Öğücü, O., Yapay Sinir Ağları ile Sistem Tanıma, Yüksek Lisans Tezi, İ.T.Ü., Fen Bilimleri Enstitüsü, İstanbul, 2006.
33. Rojas, R., Neural Networks, Springer-Verlag, Berlin, 1996.
34. Rosenblatt, F., “Principles of Neurodynamics”, East Lansing, MI: Spartan Books, 1962
35. Sağiroğlu, Ş., Beşdok E. ve Erler, M., Mühendislikte Yapay Zeka Uygulamaları I: Yapay Sinir Ağları, Ufuk Kitap Kırtasiye-Yayıncılık, Kayseri, 2003.
36. Schott, J.C., Economic Analys Through Neuro-Fuzzy Time Series Analysis, A thesis submitted to the Graduate Faculty of the University of Colorado Springsin partial fulfillment of therequirements fort he degree of Master of Science Department of Computer Science, 2005.
37. Sen, P.C., Principles of Electric Machines and Power Electronics, John Wiley & Sons, Singapore, 1989.
38. Wilamowski, B.M., Neural Network Architectures and Learning Algorithms,IEEE Industrial Electronics Magazine, 2009, 56-63.
39. Yokozuko T. ve Baba E., Force-Displacement Characteristics of Linear Stepping Motors, IEEE Proceeding-B, 1 (1992) 37-42 .
40. Zhu, Z.Q., Xia, Z.P., Howe, D. and Mellor, P., H., Reduction of cogging force in slotless linear permanent magnet motors, Proc. Inst. Elect. Eng., B-144, 4, 227–282, 1997.
41. [www.androticsllc.com/](http://www.androticsllc.com/) 16 Ocak 2015
42. [www.belgeler.com/blg/2vua/firasiz-dc-motor-sabt-miknatisli-ac-motor](http://www.belgeler.com/blg/2vua/firasiz-dc-motor-sabt-miknatisli-ac-motor), 24 Eylül 2014
43. [www.compumotor.com/whitepages/linearmotorarticle.pdf](http://www.compumotor.com/whitepages/linearmotorarticle.pdf), Linear Motors Complement Today’s Linear Motion Technologies Parker Hannifin Corporation, 11 Mart 2007.
44. [www.elektroteknoloji.com/blog/alan-etkili-sensor-hall-effect-sensor](http://www.elektroteknoloji.com/blog/alan-etkili-sensor-hall-effect-sensor), 07 Ağustos 2014
45. <http://eemtech.net/yayinlar/dhm.pdf>, 25 Eylül 2014
46. [www.magnetdeposu.com/neodyum-magnet](http://www.magnetdeposu.com/neodyum-magnet), 7 Ağustos 2014

## 8. EKLER

### Ek 1 Mikroişlemci Main Metodu Algoritma Şeması







Mikroislemci Main Metodu Algoritma Şeması

**Ek 2 Ađın İlk Yapılandırılmasının Gerçekleřtirildiđi MATLAB® Programı (m-file)**

```
function net = rtrlinit(ni,no,nu,activationfunc,initwrange)
```

```
net.ni = ni;  
net.no = no;  
net.nu = nu;  
net.iwrange = initwrange;  
nic = ni + 1;
```

```
if strcmp(activationfunc,'sigmoid')  
    net.sig1tanh2 = 1;  
elseif strcmp(activationfunc,'tanh')  
    net.sig1tanh2 = 2;  
else  
    error('Activation function must be "sigmoid" or "tanh"');  
end
```

```
net.p = zeros(nu,nu,nu+nic);  
net.y = zeros(nu,1);  
net.w = (rand(nu,nu+nic)-0.5)*2*initwrange;
```

```
net.eyenu = eye(nu);  
net.dw = zeros(size(net.w));  
net.y = zeros(nu,1);
```



**Ek 3 Ağ Ağırlıklarını Değiştiren ve Öğrenmeyi Gerçekleştiren MATLAB® Programı (m-file)**

```
function net=hatauygula(net,hata)
    net.alpha= alpha;
    for j = 1:net.no
        net.error = hata(j);
        net.dw = net.dw + net.error * shiftdim(net.p(j,:,:),1);
    end
    net.w = net.w + net.alpha * net.dw;
end
```

**Ek 4 Ağın Kullanıldığı MATLAB® Programı (m-file)**

```

function net = rtrl(net,cycles,sample,alpha,haricihata,hata)
% net = rtrl(net,cycles,sample,alpha)
% net: produced by rtrlinit
% cycles: number of cycles per example. The partial derivative is
% accumulated for this many cycles between each weight update.
% sample: vector of inputs and target
% alpha: learning rate

net.alpha = alpha;
ni = net.ni;
no = net.no;
nu = net.nu;
nic = ni+1;

for cycle = 1:cycles
    net.u = [net.y; sample(1:ni)'; 1];
    net.y = activation(net, net.w * net.u );
    dyr = repmat(actderiv(net,net.y),1,nu);
    for i = 1:nic+nu
        net.p(:,i) = dyr .* (net.w(:,1:nu) * net.p(:,i) + ...
            diag(ones(1,nu)*net.u(i)));
    end
    net.output(cycle,:) = net.y';
end

net.dw(:,:) = 0;
if haricihata>0
    for j = 1:no
        if haricihata == 1
            net.error = sample(j+ni) - net.y(j);
        else
            net.error = hata(j);
        end
        net.dw = net.dw + net.error * shiftdim(net.p(j,,:),1);
    end
else
    net.error = 0;
end
net.w = net.w + alpha * net.dw;

%-----
function y = activation(net,s)
if net.sig1tanh2 == 1
    y = 1 ./ (1 + exp(-s));
else
    y = tanh(s);
end

```

end

```
%-----  
function dy = actderiv(net,y)  
if net.sig1tanh2 == 1  
    dy = y .* (1 - y);  
else  
    dy = 1 - y.^2;  
end
```

### Ek 5 Tüm Sistemi Çalıştıran MATLAB® Programı (m-file)

```

clear all;
clc
close all;
%-----Sistem Ayarları-----
%istenilen_Frekans
istenilen_frekans = 1.19;
%iletişim Yapılandırılması
Seriport_isim = 'COM12';
Seriport_BaudRate = 115200;
%Önceki Bilgileri Devralması için 1 değilse 0
dosyadanyukle = 0;
dosyayakaydet = 0;
%Sinir ağı Ayarları
Giris_Birimi_Sayisi = 1;
Cikis_Birimi_Sayisi = 1;
Birim_sayisi=4;
Akitvasyon_Fonksiyonu='sigmoid';
Agirlik_Araligi=0.1;
%Kontrol Ayarları 0 Pwm 1 Bobinler arasında gerçeken Gecikme
kontrolmodu = 0; %1 veya 0
minPwm=4;
maxPwm=100;
minGecikme=0;
maxGecikme=255;
maxfrekans=5;
%Kontrol Geçikmesi
bekleme =0.25;

dosya=0;
if(dosyadanyukle==1)
    dosya = exist('net.mat');
    if (dosya == 2)
        load('net.mat');
        Disp('Dosyadan Yuklendi');
    else
        Disp('Dosya Okunamadı');
    end
end
if(dosya ~= 2)
    ni = Giris_Birimi_Sayisi ; no = Cikis_Birimi_Sayisi; nu =Birim_sayisi;
    net = rtrlinit(ni,no,nu,Akitvasyon_Fonksiyonu,Agirlik_Araligi);

end
% Eğer açık port varsa onları kapatmak için
if(size(instrfind)>0)
fclose(instrfind);
end

```

```

s = serial(Seriport_isim);
set(s,'BaudRate',Seriport_BaudRate);
set(s,'Databit',8);
set(s,'Timeout',1);
fopen(s);

while s.BytesAvailable<(0)
    % Close & clean up
    fclose(s);
    delete(s);
    clear s;
    % redo the setup
    fclose(instrfind);
    foundTermination = 0;
    s = serial(Seriport_isim);
    set(s,'BaudRate',Seriport_BaudRate);
    set(s,'Databit',8);
    set(s,'Timeout',1);
    fopen(s);
end

```

```

figure(1);
subplot(3,1,1)
title('Çıkış');
lHandle0 = line(nan, nan);
subplot(3,1,2)
title('Hata');
lHandle1 = line(nan, nan);
subplot(3,1,3)
title('Frekans');
lHandle2 = line(nan, nan);

```

```

seriveri.mod=kontrolmodu;
seriveri.veri=0;
gelenveriler=[];
hatalar=[];
gelenornek=0;
Xa=[];
Ya=[];
Yb=[];
Yc=[];
tempfrekansa=[];
seriyaz=[];

```

```

while 1
    disp(['Çalışma Modu = ' num2str(seriveri.mod) ' Veri = ' num2str(seriveri.veri)]);
    seriyaz=['*' char(uint8(seriveri.mod)) char(uint8(seriveri.veri)) '#'];
    disp(['Yazılan Değer = ' seriyaz]);

```

```

fwrite(s,seriyaz,'char');
gelenornek = gelenornek + 1;
pause(0.5);
cc = 1;
gelenveri = 0;
serihatasayac=0;
gelenveriyaz=[];
while(cc<5)
    t="";
    try
        t = fread(s,1,'char');
    catch
        t=0;
    end
    d=size(t);
    if(d(2)>0)
        gelenveriyaz=[gelenveriyaz t]

        if(double(t)==35)
            cc=2;
            continue;
        end
        if(cc==2)
            cc=3;

            gelenveri = double(t)*256.0;
            continue;
        end
        if(cc==3)
            cc=4;
            gelenveri = gelenveri + double(t);
            continue;
        end
        if(cc==4)
            cc=5;
            gelenveri = gelenveri*256.0;
            gelenveri = gelenveri + double(t);
            serihatasayac=0;
            continue;
        end
    end
    serihatasayac = serihatasayac+1;
    if(serihatasayac>15)
        break;
    end
end
if(serihatasayac>15)
    disp('Seri iletişim hatası2');

```

```

    break;
end
disp(gelenveriyaz);
saniyedekitik=10000000/16;
gelenveri=1.0/(gelenveri/saniyedekitik);
ghkl=size(tempfrekans);
if(ghkl(2)>9)
    tempfrekans=[tempfrekans(end-8:end) gelenveri];
    gelenveri= sum(tempfrekans)/10;
else
    tempfrekans=[tempfrekans gelenveri];
end
disp(tempfrekans)
gelenveriler = [gelenveriler gelenveri];
disp(['Firekans = ' num2str(gelenveri)]);
net = rtrl(net,[istenilen_frekans/maxfrekans]);
if(seriveri.mod==0)
    seriveri.veri = net.y(1)*(maxPwm-minPwm) + minPwm;
else
    seriveri.veri = net.y(1)*(maxGecikme-minGecikme) + minGecikme;
end
seriveri.veri=uint8(seriveri.veri);
hata=(istenilen_frekans/maxfrekans)-(gelenveri/maxfrekans);
hatalar=[hatalar hata];

error = hata*100;
integral = integral + error*dt;
derivative = (error-previous_error)/dt;
u = Kp*error+Ki*integral+Kd*derivative;
yp = yp+(-b*yp/m + u/m)*dt
yp1=yp;
if(yp1<0)
    yp1=0;
end
if(yp1>1)
    yp1=1;
end
seriveri.veri = yp1*(maxPwm-minPwm) + minPwm;
net=hatauygula(net, yp1, 0.1);

Xa = [Xa toc];
Ya = [Ya seriveri.veri];
Yb = [Yb hata];
Yc = [Yc gelenveri];

set(IHandle0, 'XData', Xa, 'YData', Ya);
set(IHandle1, 'XData', Xa, 'YData', Yb);
set(IHandle2, 'XData', Xa, 'YData', Yc);
drawnow;

```

```
end  
fclose(s);  
save('net.mat','net','ni','no','nu');
```



## **ÖZGEÇMİŐ**

07.10.1982 tarihinde Sinop'ta doğdu. İlk, ortaokul ve lise öğrenimini Sinop'ta tamamladı. 2001 yılında başladığı Gazi Üniversitesi Teknik Eğitim Fakültesi Elektrik Eğitimi Bölümünden 2005 yılında Elektrik Öğretmeni unvanı ile mezun oldu. 2008 Aralık ayından itibaren Sinop Üniversitesi Meslek Yüksek Okulu Elektrik ve Enerji Programı'nda Öğretim Görevlisi olarak çalışmaya başladı. 2011 yılında Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü Elektrik-Elektronik Mühendisliği Anabilim Dalı'nda yüksek lisans öğrenimine başladı.