

**KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

**ÇİFT YANLI DOĞRUSAL HAREKETLİ SÜREKLİ MİKNATISLI SENKRON
MOTORUN TASARIMI, ÜRETİMİ VE YAPAY SİNİR AĞ TABANLI
SKALER HIZ DENETİMİ**

DOKTORA TEZİ

Özcan OTKUN

**ŞUBAT 2015
TRABZON**

**KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

**ÇİFT YANLI DOĞRUSAL HAREKETLİ SÜREKLİ MIKNATISLI SENKRON
MOTORUN TASARIMI, ÜRETİMİ VE YAPAY SİNİR AĞ TABANLI
SKALER HIZ DENETİMİ**

Özcan OTKUN

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

**Tezin Enstitüye Verildiği Tarih : 05.01.2015
Tezin Savunma Tarihi : 05.02.2015**

Tez Danışmanı : Prof. Dr. A. Sefa AKPINAR

Trabzon 2015

Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü
Elektrik-Elektronik Mühendisliği Ana Bilim Dalında
Özcan OTKUN Tarafından Hazırlanan

**ÇİFT YANLI DOĞRUSAL HAREKETLİ SÜREKLİ MIKNATISLI SENKRON
MOTORUN TASARIMI, ÜRETİMİ VE YAPAY SİNİR AĞ TABANLI
SKALER HIZ DENETİMİ**

**başlıklı bu çalışma, Enstitü Yönetim Kurulunun 13 / 01 / 2015 gün ve 1585 sayılı kararıyla
oluşturulan jüri tarafından yapılan sınavda
DOKTORA TEZİ
olarak kabul edilmiştir.**

Jüri Üyeleri

Başkan : Prof. Dr. İbrahim ŞENOL

Üye : Prof. Dr. Adem Sefa AKPINAR

Üye : Doç. Dr. Halil İbrahim OKUMUŞ

Üye : Yrd. Doç. Dr. Hüseyin PEHLİVAN

Üye : Yrd. Doç. Dr. Hakan KAHVECİ

Prof. Dr. Sadettin KORKMAZ
Enstitü Müdürü

ÖNSÖZ

Bu çalışma, çift yanlı doğrusal hareketli sürekli mıknatıslı senkron motorun hız denetimini yapay sinir ağları ile gerçekleştirmeyi amaçlamaktadır.

Bu çalışmayı yürütürken beni yönlendiren Danışmanım Sayın Hocam Prof. Dr. A. Sefa AKPINAR'a gösterdiği ilgi, hoşgörü ve çabalarından ötürü teşekkürlerimi saygılarımla sunarım.

Tez izleme jürimde bulunan ve doktora eğitimimde eleştiri ve fikirleriyle bana her konuda yardımcı olan beni yönlendiren Sayın Hocam Doç. Dr. Halil İbrahim OKUMUŞ'a ve Sayın Hocam Yrd. Doç. Dr. Hüseyin PEHLİVAN'a teşekkür ederim.

Yapay sinir ağları konusunda çalışmalarımda hep yanımda olan ve desteğini esirgemeyen Öğr. Gör. Ramazan Özgür DOĞAN'a minnettarım.

Her türlü desteği sunmaktan geri kalmayan Gümüşhane MYO yöneticilerine, yardımlarından dolayı Öğr. Gör. Faruk DEMİR'e, Öğr. Gör. Nihat ÖZDEMİR'e ve tüm Gümüşhane MYO personeline destekleri için teşekkür ederim.

Bu çalışmayı yürütürken desteklerini esirgemeyen ve onları ihmal etmeme katlanan eşime, kızıma ve oğluma minnettarım.

Özcan OTKUN

Trabzon 2015

TEZ BEYANNAMESİ

Doktora Tezi olarak sunduđum “ ÇİFT YANLI DOĐRUSAL HAREKETLİ SÜREKLİ MIKNATISLI SENKRON MOTORUN TASARIMI, ÜRETİMİ VE YAPAY SİNİR AĐ TABANLI SKALER HIZ DENETİMİ ” başlıklı bu çalışmayı baştan sona kadar danışmanım Prof. Dr. Adem Sefa AKPINAR‘ın sorumluluđunda tamamladıđımı, verileri kendim topladıđımı, deneyleri ilgili laboratuvarlarda yaptıđımı, başka kaynaklardan aldıđım bilgileri metinde ve kaynakçada eksiksiz olarak gösterdiđimi, çalışma sürecinde bilimsel araştırma ve etik kurallara uygun olarak davrandıđımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiđimi beyan ederim. 18/02/2015

(İmza)

Özcan OTKUN

İÇİNDEKİLER

	<u>Sayfa no</u>
ÖNSÖZ.....	III
TEZ BEYANNAMESİ.....	IV
İÇİNDEKİLER.....	V
ÖZET	VIII
SUMMARY	IX
ŞEKİLLER DİZİNİ	X
TABLolar DİZİNİ.....	XIV
SİMGELER DİZİNİ.....	XV
1. GENELBİLGİLER.....	1
1.1. Giriş	1
1.1.1. Doğrusal Hareketli Motorlar	2
1.1.2. Doğrusal Hareketli Motor Türleri.....	3
1.1.3. Doğrusal Hareketli Motorların Uygulama Alanları.....	5
1.2. Konuyla İlgili Yapılmış Çalışmalar.....	10
1.3. Tezin Amacı	14
2. ÇİFT YANLI DOĞRUSAL HAREKETLİ SÜREKLİ MIKNATISLI SENKRON MOTORLAR (ÇYDHSMSM)	15
2.1. Uzay Vektör Denklemleri, Eksen Sistemleri ve Dönüşümleri	15
2.2. Motorun Matematiksel Modeli.....	18
2.3. Doğrusal Hareketli Senkron Motorlarda Çalışmayı Etkileyen Etmenler	20
2.3.1. Uç Etkileri	20
2.3.2. Hava Aralığı Etkileri	23
2.3.3. Sürtünme Etkileri.....	25
2.4. Motorun Sarım Teknikleri.....	26
2.5. Motorda Kullanılan Sürekli Miknatıslar	27

2.5.1.	Sürekli Mıknatısların Özellikleri	29
2.5.2.	Sürekli Mıknatısların Yerleşimleri	30
2.6.	Motor Denetim Yöntemleri	31
2.6.1.	Skaler (V/f) Denetim Yöntemi	32
2.6.2.	Vektörel Denetim Yöntemi	32
2.6.3.	P+I+D Denetim Yöntemi	33
2.6.4.	Akıllı Denetim Yöntemleri.....	34
3.	YAPAY SİNİR AĞLARI İLE DENETİM SİSTEMLERİ.....	35
3.1.	Yapay Sinir Ağları.....	35
3.2.	Biyolojik Beyin Hücresi ve Yapay Sinir Ağlarının Yapısı	36
3.2.1.	Girişler.....	37
3.2.2.	Ağırlıklar	37
3.2.3.	Toplam Fonksiyonu.....	37
3.2.4.	Aktivasyon Fonksiyonu.....	38
3.2.5.	Ölçekleme ve Sınırlama.....	40
3.2.6.	Hücrenin Çıkışı.....	40
3.3.	Yapay Sinir Ağlarının Sınıflandırılması.....	41
3.3.1.	Yapılarına Göre Sınıflandırma	41
3.3.2.	Öğrenme Yaklaşımlarına Göre Sınıflandırma.....	42
3.3.3.	Öğrenme Kurallarına Göre Sınıflandırma	44
3.4.	Öğrenme Yöntemleri	45
3.4.1.	Geri Yayılımlı Öğrenme Yöntemi.....	45
3.4.2.	Geri Yayılımlı Öğrenme Yönteminin Uygulama Adımları.....	48
3.5.	Yapay Sinir Ağlarının Üstünlükleri ve Sakıncaları.....	50
4.	YAPILAN ÇALIŞMALAR VE BULGULAR	51
4.1.	İncelenen Motorun Yapısal Ayrıntıları.....	51
4.1.1.	Birincil Yan Yapım Aşamaları	52

4.1.2.	İkincil Yan Yapım Aşamaları.....	56
4.2.	Kullanılan Değişken Frekanslı Sürücüsü	59
4.3.	Motor Parametrelerinin Belirlenmesi	62
4.3.1.	Birincil Yan Sargı Direncinin Bulunması	62
4.3.2.	Birincil Yan İndüktansının Bulunması	63
4.3.3.	Sürekli Mıknatıs Akısının Hesaplanması	64
4.4.	Çift Yanlı Doğrusal Hareketli Sürekli Mıknatıslı Senkron Motorun Skaler (V/f) Yöntem ile Hız Denetimi	65
4.5.	Çift Yanlı Doğrusal Hareketli Sürekli Mıknatıslı Senkron Motorun İkincil Yan Düzeninin İtme Kuvvetine Etkisinin İncelenmesi.....	67
4.5.1.	Mıknatısların Dikey Yerleştirilmesi	67
4.5.2.	Mıknatısların Yatay Yerleştirilmesi	68
4.5.3.	DeneySEL Sonuçlar	68
4.6.	Tasarlanan Motorun Matlab/Simulink Yazılım Programı ile Benzetimi	71
4.6.1.	Motorun Benzetim Modelinin V/f Denetim Yöntemi ile Durum İncelemesi	73
4.6.1.1.	Motorun Açık ve Kapalı Çevrim Benzetim Modelleri.....	74
4.6.1.2.	Benzetim Sonuçları ve Değerlendirme.....	77
4.7.	Çift Yanlı Doğrusal Hareketli Sürekli Mıknatıslı Senkron Motorun Yapay Sinir Ağları ile Hız Denetimi	79
4.7.1.	Çift Yanlı Doğrusal Hareketli Sürekli Mıknatıslı Senkron Motorun Yapay Sinir Ağları ile Yol-Zaman Denetimi	79
4.7.1.1.	Deneylerden Elde Edilen Sonuçlar.....	83
4.7.2.	Çift Yanlı Doğrusal Hareketli Sürekli Mıknatıslı Senkron Motorun Yapay Sinir Ağları ile Hız-Zaman Denetimi.....	87
4.7.2.1.	Deneylerden Elde Edilen Sonuçlar.....	87
5.	SONUÇLAR.....	98
6.	ÖNERİLER	100
7.	KAYNAKLAR.....	101
8.	EKLER	110

ÖZGEÇMİŞ

Doktora Tezi

ÖZET

ÇİFT YANLI DOĞRUSAL HAREKETLİ SÜREKLİ MIKNATISLI SENKRON MOTORUN TASARIMI, ÜRETİMİ VE YAPAY SINIR AĞ TABANLI SKALER HIZ DENETİMİ

Özcan OTKUN

Karadeniz Teknik Üniversitesi
Fen Bilimleri Enstitüsü
Elektrik Elektronik Mühendisliği Anabilim Dalı
Danışman: Prof. Dr. A. Sefa AKPINAR
2015, 109 Sayfa, 13 Sayfa Ek

Bu çalışmada, Çift Yanlı Doğrusal Hareketli Sürekli Mıknatıslı Senkron Motorun (ÇYDHSMSM) Yapay Sinir Ağları (YSA) ile hız denetimi deneysel olarak araştırıldı. Bu denetim için değişken frekanslı bir motor sürücüsü kullanıldı. YSA ile sürücü frekans değeri değiştirilerek motorun hız denetimi gerçekleştirildi.

Tezde öncelikle Doğrusal Hareketli Motorlar (DHM) hakkında genel bilgiler verildi. Bu motorların matematiksel ifadelerine, sarım tekniklerine, kullanılan sürekli mıknatıs özelliklerine, günümüze kadar kullanılan denetim yöntemlerine yer verildi.

Deneysel çalışmalar için incelenecek ÇYDHSMSM'nin yapısal ayrıntıları açıklandı. Motora ait Birincil Yan ve İkincil Yan yapım aşamaları üzerinde duruldu. C# yazılımıyla denetim programı ve arayüzü hazırlandı. Değişken frekanslı sürücü ile motorun hız denetim çalışmaları yapıldı. Daha sonra motorun İkincil Yan kısmında bulunan sürekli mıknatısların denetim başarısına etkisi araştırıldı. Bu çalışmalar için deneysel ve benzetim çalışma sonuçları verildi.

Son olarak, motorun YSA ile hız denetim çalışmaları yapıldı. YSA yapısında, çok katmanlı ileri beslemeli yapay sinir ağı, geri yayımlı öğrenme algoritması ve danışmanlı öğrenme metodu kullanıldı. Bu yöntem için iki farklı eğitim seti hazırlandı. Birinci eğitim seti yol ve zaman bilgilerine göre, ikinci eğitim seti hız ve zaman bilgilerine göre rastgele oluşturuldu. Burada, iki eğitim setinin arka planında istenen hıza ulaşmak için gereken frekans değeri hesaplanmaktadır ve motor sürücüsünden alınan anlık frekans bilgisine göre YSA frekans değerini arttırıp azaltarak motorun hız denetimini gerçekleştirmektedir. Çalışmanın sonunda farklı hızlar için yapılan deney sonuçları tartışılmıştır.

Anahtar Kelimeler: Doğrusal hareketli motor, Doğrusal hareketli sürekli mıknatıslı senkron motor, V/f denetim, Yapay sinir ağı, Hız denetimi

PhD. Thesis

SUMMARY

DESIGN, PRODUCT AND NEURAL NETWORK BASED SCALAR SPEED CONTROL OF A
DOUBLE SIDED LINEAR PERMANENT MAGNET SYNCHRONOUS MOTOR

Özcan OTKUN

Karadeniz Technical University
Institute of Science
Department of Electrical and Electronics Engineering
Supervisor: Prof. Dr. A. Sefa AKPINAR
2015, 109 Pages, 13 Pages Appendix

In this study, speed control of Double Sided Linear Permanent Magnet Synchronous Motor (DSLPMMSM) was investigated experimentally with Artificial Neural Networks (ANN). A variable frequency drive was used for this control. Motor speed control was performed by changing the drive frequency with an ANN.

First in this thesis, general information about Linear Motors (LM) were given. The mathematical expressions, winding techniques, the used permanent magnet properties and the control methods used until today were given.

For the experimental work, structural details of designed DSLPMMSM were described. It was focused on the construction of the primary and secondary side of the motor. The control program and the interface were prepared with C# software. Speed control study of motor was performed with variable frequency drives. After that, the impact of permanent magnet in the secondary side of motor on control achievements was investigated. For these studies, experimental and simulation results were carried out.

Finally, speed control studies of motor were conducted with ANN. In the ANN structure, multi-layer, feed-forward neural network, back-propagation learning algorithm and supervised learning method were used. Two different training sets were prepared for this method. The first training set according to road and time information, the second training set according to speed and time information was generated randomly. Here, the frequency needed to reach the desired speed is calculated in the background of two training sets and according to the knowledge of the instantaneous frequency obtained from the motor drive. ANN carries out the motor speed control by increasing or decreasing the frequency. At the end of the study, the experimental results for different speeds were discussed.

Key Words : Linear motor, Linear permanent magnet synchronous motor, V/f control, Artificial neural network, Speed control

ŞEKİLLER DİZİNİ

	<u>Sayfa no</u>
Şekil 1. Doğrusal hareketli motorun yapısı	2
Şekil 2. Doğrusal hareketli motorlarda kuvvet oluşumu	3
Şekil 3. Doğrusal hareketli motor türleri	3
Şekil 4. Doğrusal hareketli motorların yapısal sınıflandırılması	4
Şekil 5. İncelenen motor modeli	5
Şekil 6. Taşıma sistemlerinde kullanılan doğrusal hareketli motor	5
Şekil 7. Otomasyon sistemlerinde kullanılan doğrusal hareketli motor	6
Şekil 8. Tıp alanında kullanılan manyetik rezonans cihazı	6
Şekil 9. Robotik sanayisinde X-Y ekseninde hareket	7
Şekil 10. İmalat sanayisinde kullanılan CNC cihazı	7
Şekil 11. Doğrusal hareketli yürüyen merdiven	8
Şekil 12. Doğrusal hareketli lazer tarayıcı	8
Şekil 13. Elektronik elemanları baskı devresine yerleştiren doğrusal hareketli motor	9
Şekil 14. Doğrusal hareketli hızlı tren uygulaması	9
Şekil 15. 1960 yılından günümüze doğrusal hareketli motor çalışmaları	10
Şekil 16. Üç fazlı eşdeğer devre modeli	15
Şekil 17. Doğrusal hareketli sürekli mıknatıslı senkron motorun elektriksel modeli	16
Şekil 18. $ABC-\alpha\beta$ referans düzlemi	17
Şekil 19. $\alpha\beta -dq$ referans düzlemi	17
Şekil 20. Doğrusal hareketli motorun $d-q$ modeli	18
Şekil 21. Boyuna uç etkisi grafiği	21
Şekil 22. Enine uç etkisi grafiği	21
Şekil 23. Hava aralığının etkileri	24
Şekil 24. Sürtünme modeli grafiği (Stribeck eğrisi)	25
Şekil 25. Sargı düzeni	26
Şekil 26. Bir faza ait oluk sayısı, bir olan; tam adımlı tek tabakalı sargı düzeni	26
Şekil 27. Bir faza ait oluk sayısı, bir olan; kısa adımlı çift tabakalı sargı düzeni	27
Şekil 28. Sürekli mıknatısların demanyetizasyon eğrileri	30
Şekil 29. İkincil yan sürekli mıknatıs yerleşimi	30
Şekil 30. Motor denetim devresinin genel blok yapısı	31
Şekil 31. Skaler denetim blok şeması	32

Şekil 32. Biyolojik sinir hücresi ve bileşenleri.....	36
Şekil 33. Yapay sinir hücresinin yapısı (Perseptron model)	37
Şekil 34. Doğrusal aktivasyon fonksiyonu grafiği	38
Şekil 35. Tanjant hiperbolik aktivasyon fonksiyonu grafiği	39
Şekil 36. Sigmoid aktivasyon fonksiyonu grafiği	39
Şekil 37. Signum aktivasyon fonksiyonu grafiği	40
Şekil 38. İleri beslemeli ağın blok diyagramı.....	41
Şekil 39. Geri beslemeli ağın blok diyagramı	42
Şekil 40. Denetimli öğrenme modeli blok diyagramı	43
Şekil 41. Denetimsiz öğrenme blok diyagramı	43
Şekil 42. Takviyeli öğrenme blok diyagramı	44
Şekil 43. Yapay sinir ağı katman grafiği.....	46
Şekil 44. Geri yayılım ağ yapısı	47
Şekil 45. İleri beslemeli bir ağ için geri yayılım öğrenme yöntemi akış şeması.....	49
Şekil 46. Birincil yan görünümü	51
Şekil 47. İkincil yan görünümü	51
Şekil 48. Birincil yan kesim görünümü.....	52
Şekil 49. Birincil yanın yapım aşamaları.....	53
Şekil 50. Birincil yan sarımı	54
Şekil 51. Birincil yan son durum görünümü	55
Şekil 52. İkincil yan ilk yapım aşamaları	56
Şekil 53. Kullanılan sürekli mıknatıslar	57
Şekil 54. İkincil yan sürekli mıknatıs yerleşimi	57
Şekil 55. Laboratuvar çalışma ortamı.....	58
Şekil 56. RS485'i RS232'ye çeviren konvertör	59
Şekil 57. LG-SV022iG5A-4 sürücü	60
Şekil 58. LG-SV022iG5A-4 sürücüsünün bağlantı terminalleri	60
Şekil 59. DFS ile V/f denetimin blok diyagramı.....	61
Şekil 60. Bilgisayar ile sürücü haberleşme bağlantısı	61
Şekil 61. Doğru akım deneyi bağlantı şeması	62
Şekil 62. Skalere denetim blok diyagramı.....	65
Şekil 63. V/f denetimde frekans ve hız grafikleri.....	66
Şekil 64. V/f denetimde elde edilen akım ve itme kuvveti grafikleri.....	66

Şekil 65. Sürekli mıknatısların dikey yerleştirilmesi.....	67
Şekil 66. Sürekli mıknatısların yatay yerleştirilmesi.....	68
Şekil 67. Mıknatısların dikey ve yatay konumdaki hız grafikleri	69
Şekil 68. Mıknatısların dikey ve yatay konumdaki frekans grafikleri	69
Şekil 69. Mıknatısların dikey ve yatay konumdaki çıkış gücü grafikleri.....	70
Şekil 70. Mıknatısların dikey ve yatay konumdaki itme kuvveti grafikleri	70
Şekil 71. Çift yanlı doğrusal hareketli sürekli mıknatıslı senkron motorun elektriksel benzetim modeli.....	71
Şekil 72. Çift yanlı doğrusal hareketli sürekli mıknatıslı senkron motorun mekaniksel benzetim modeli.....	72
Şekil 73. dq - ABC eksen sistemi dönüşümünün benzetim modeli	72
Şekil 74. ABC - dq eksen sistemi dönüşümünün benzetim modeli	72
Şekil 75. Doğrusal hareketli sürekli mıknatıslı senkron motorun benzetim modeli	73
Şekil 76. Skaler denetimin benzetim modeli	74
Şekil 77. Skaler denetimin açık çevrim benzetim modeli	75
Şekil 78. Skaler denetimin kapalı çevrim benzetim modeli	76
Şekil 79. 7 Hz frekans değeri için elde edilen benzetim sonuçları.....	77
Şekil 80. 4 Hz frekans değeri için elde edilen benzetim sonuçları.....	78
Şekil 81. Deneysel çalışmanın yapay sinir ağı yapısı.....	81
Şekil 82. Deneysel çalışmanın blok devre şeması.....	83
Şekil 83. Manyetik lineer enkoder.....	84
Şekil 84. MLE'den alınan verileri okuyan devre	84
Şekil 85. Manyetik lineer enkoderden alınan 1. test sonuçları.....	84
Şekil 86. Manyetik lineer enkoderden alınan 2. test sonuçları.....	85
Şekil 87. Manyetik lineer enkoderden alınan 7. test sonuçları.....	85
Şekil 88. YSA için ağ oluşturma ekranı	88
Şekil 89. YSA eğitim seti oluşturma ekranı	88
Şekil 90. YSA eğitim hata grafiği	89
Şekil 91. Ana pencere ekranı.....	90
Şekil 92. Motorun yüksüz durumu	90
Şekil 93. Motorun yüklü durumu	91
Şekil 94. Birinci deney sonuçları.....	92
Şekil 95. İkinci deney sonuçları	93
Şekil 96. Üçüncü deney sonuçları	94

Şekil 97. Dördüncü deney sonuçları.....	95
Şekil 98. Beşinci deney sonuçları.....	96

TABLULAR DİZİNİ

	<u>Sayfa no</u>
Tablo 1. Sürekli mıknatısların yıllara göre değişimi	28
Tablo 2. Mıknatıslara ait özellikler.....	29
Tablo 3. Biyolojik sinir hücresinin YSA'daki karşılığı	36
Tablo 4. Paket halindeki saçların kesim ölçüleri.....	53
Tablo 5. Kullanılan neodyum mıknatıs ölçüleri	56
Tablo 6. DA deneyinden elde edilen sonuçlar.....	63
Tablo 7. Kilitli Birincil Yan deneyi sonuçları	63
Tablo 8. Birincil Yan direnç ve indüktans değerleri	64
Tablo 9. İncelenen motorun parametreleri	65
Tablo 10. Örnek YSA eğitim seti (Yol-Zaman).....	80
Tablo 11. Yol ve zaman denetimi sonuçları	86
Tablo 12. Örnek YSA eğitim seti (Hız).....	87
Tablo 13. Motorun yüksüz durumu için elde edilen yol bilgileri.....	91

SİMGELER DİZİNİ

v	Doğrusal hareketli motorun hızı (Lineer hız)
S	Vektörel olarak birincil yan akım, gerilim, manyetik akı
R_a, R_b, R_c	A, B, C faz dirençleri
L_a, L_b, L_c	A, B, C faz indüktansları
α	Uzay vektör operatörü
k	Transformasyon sabiti
i	Birincil yan akımı
ψ	Manyetik akı
i_α, i_β	α ve β eksenlerinde akım
i_d, i_q	d ve q eksenlerinde akım
R	Birincil yan sargı direnci
w	Açısal hız
τ	Kutup adımını
τ_{PM}	Sürekli mıknatıs adımını
ψ_d, ψ_q	d ve q eksenlerinde manyetik akı
ψ_{sm}	Sürekli mıknatıs akısı
P	Birincil yan anlık güç girişi
P_{el}	Elektromanyetik güç
F_{itme}	İtme kuvveti
p	Kutup sayısı
$2p$	Çift kutup sayısı
F_h	Harici (dış) kuvvet
B	Sürtünme kuvveti
M	Doğrusal hareketli motorun ağırlığı
$F_{uç_et}$	Enine ve boyuna uç etkisi
C_{srt}	Coulomb katsayısı
V_{srt}	Viscous katsayısı
S_{srt}	Stribeck katsayısı
B_r	Sürekli mıknatısın akı yoğunluğu

BH_{max}	Maksimum enerji çarpımı değeri
H_cB	Zorlayıcı kuvvet
f	Motorun çalışma frekansı
X_1, X_2, \dots, X_n	Yapay sinir ağı girişleri
w_1, w_2, \dots, w_n	Yapay sinir ağı ağırlıkları
$f_{(net)}$	Aktivasyon fonksiyonu
O_k	Yapay sinir ağı çıkış katmanı
O_j	Yapay sinir ağı gizli katmanı
x	Birincil yan oluk sayısı
m	Faz sayısı
C	Bir kutup altında bir faza düşen renk sayısı
y_x	Oyuk adımı
N	Bir faza ait bir oluktaki sarım sayısı
k_{w1}	Sargı faktörü

1.GENEL BİLGİLER

1.1.Giriş

Teknolojik gelişmelere bağlı olarak doğrusal hareket önemini giderek arttırmaktadır. Doğrusal hareketi elde etmenin en kolay yolu doğrusal hareketli elektrik motorlarıdır. Çevremize baktığımızda doğrusal hareket elde etmek için döner motorların kullanıldığı çok sayıda örneği görebiliriz. Bunların yerine doğrusal hareketli motorlar kullanılarak daha az malzeme ve maliyetle aynı iş gerçekleştirilebilir. Birçok sistemde doğrusal hareket, döner hareketli elektrik motoru, kayış veya dişli sistemlerinin bileşimi ile gerçekleştirilmektedir. Döner hareketi doğrusal harekete dönüştürmek için dişliler, vidalar, krank milleri vb. bütün bu ek sistemler gereklidir. Bu ek sistemlerden dolayı sürtünme kayıplarında, eylemsizlik momentinde ve sistemin toplam kütleinde artış olur. Fakat kayıpların mümkün olduğunca azaltılması, sistemin daha az hareketli parçaya sahip olması ve daha az bakıma gerek duyulması, bu tür sistemlerden beklenen özelliklerdir [1].

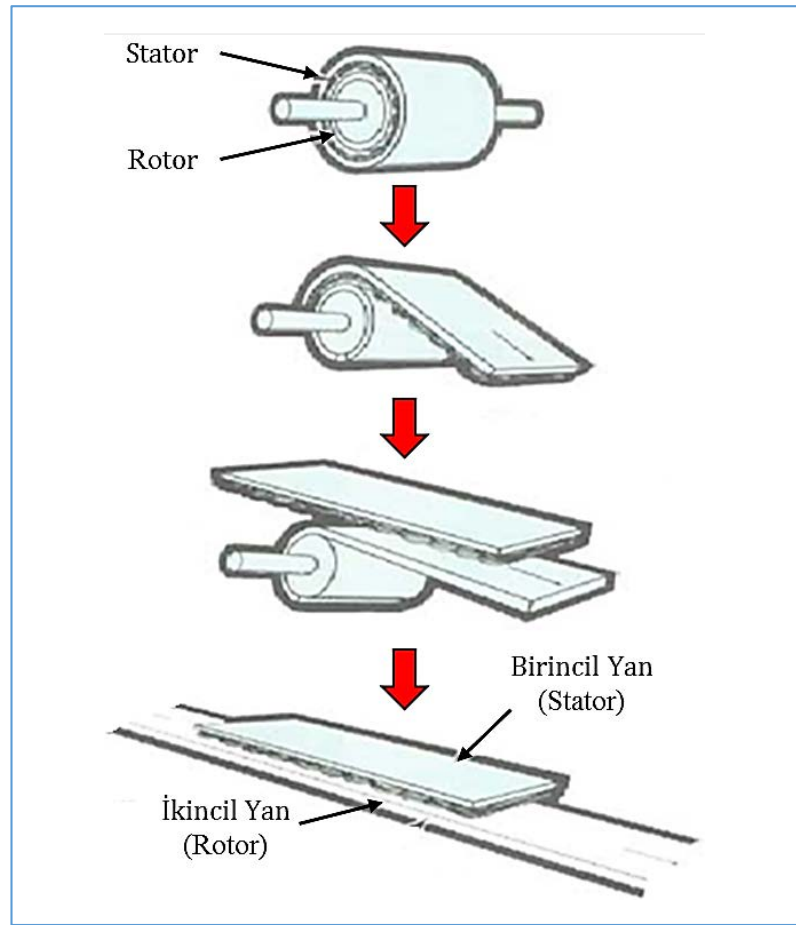
Doğrusal hareketi döner tip motorlardan ziyade doğrusal hareketli motor ile elde etme hem verimlilik hem de maliyet bakımından tercih edilmektedir. Bunun yanı sıra basit, güçlü, güvenilir olmaları, mekanik dönüştürücülere ihtiyaç duymamaları doğrusal hareketli motorları tercih edilir kılmıştır. Bu nedenle doğrusal hareketi ara düzeneklere ihtiyaç duymadan sağlayan Doğrusal Hareketli Motorlar (DHM) günümüzde yaygın olarak kullanılmaya başlanmıştır [1]- [3].

Araştırmacıların çalışmalarını DHM'ler üzerinde yoğunlaştırmasının iki ana nedeni bulunmaktadır. Bunlardan birincisi manyetik temelli malzemelerin daha kolay işlenebilmesi ve maliyetlerinin eskiye göre çok daha makul durumda olmasıdır. İkincisi de sayısal kontrol ve güç elektroniği teknolojisindeki gelişmelerin, yüksek performanslı aygıtların yüksek hassasiyetlerde ölçülebilir ve kontrol edilebilir hale gelmesidir [4].

Teknolojinin yeni sistemlere getirdiği çözümler, eski sistemlerin yerini ne şekilde ve nasıl dolduracağı araştırmacıların ve tasarımcıların bu konudaki çalışmalarıyla belirlenecektir. Doğrusal hareketin bu çalışmaların içerisinde yerinin ne olacağını, bu teknolojilere yatırım yapan firmaların sayısının artmasıyla ilgili olduğu söylenebilir. Hâlihazırda bulunan çözümlerin yerini şimdilerde doğrusal hareketli sistemlerin aldığı da görülmektedir [3].

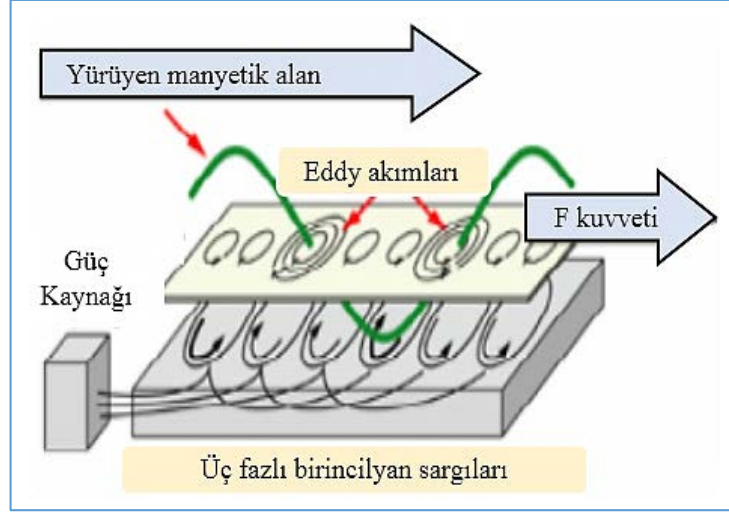
1.1.1. Doğrusal Hareketli Motorlar

DHM'lerin yapısı klasik döner hareketli motor yapısının kesilip açılmasıyla oluşturulduğu düşünülebilir. Döner Hareketli Motoru (DÖHM) oluşturan stator (dıştaki sabit kısım) ve rotor (içteki dönen kısım) dairesel şekilde iç içe oldukları için statordaki sargılar yardımıyla oluşturulan döner alan ile dönme hareketi gerçekleştirir. DHM'lerde ise statora karşılık gelen Birincil Yan, rotora karşılık gelen İkincil Yan, üst üste oldukları için Birincil Yanda bulunan sargılar tarafından meydana gelen “Yürüyen Alan” yardımıyla doğrusal bir hareket gerçekleşmiş olur [5]. Şekil 1’de DÖHM’den elde edilen DHM gösterilmektedir [6].



Şekil 1. Doğrusal hareketli motorun yapısı

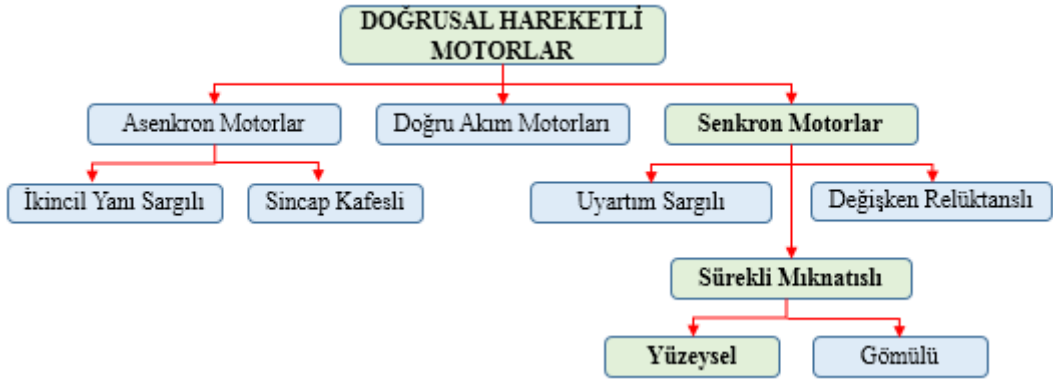
Bu yapıya sahip motorlarda; yürüyen alanın oluşumu, Eddy akımlarının meydana geldiği noktalar ve itme kuvvetinin yönü önem arz etmektedir. DHM'lerde iki kuvvet meydana gelmektedir. Bunlardan biri olan “Eksenel Kuvvet” doğrusal hareketi sağlarken, “Normal Kuvvet” olarak adlandırılan diğer kuvvet kaldırma işini yapmaktadır [7]. DHM’de meydana gelen bu kuvvet oluşumu Şekil 2’de gösterilmiştir [8].



Şekil 2. Doğrusal hareketli motorlarda kuvvet oluşumu

1.1.2. Doğrusal Hareketli Motor Türleri

Günümüzde DÖHM'lerin bütün türlerine (Senkron, Asenkron, Relüktans vb.) benzer DHM'ler bulunabilmektedir. Bu bakımdan DHM'leri çok sayıda gruba ayırmak mümkündür [9, 10]. DHM'lerin genel sınıflandırılması Şekil 3'de görülmektedir.

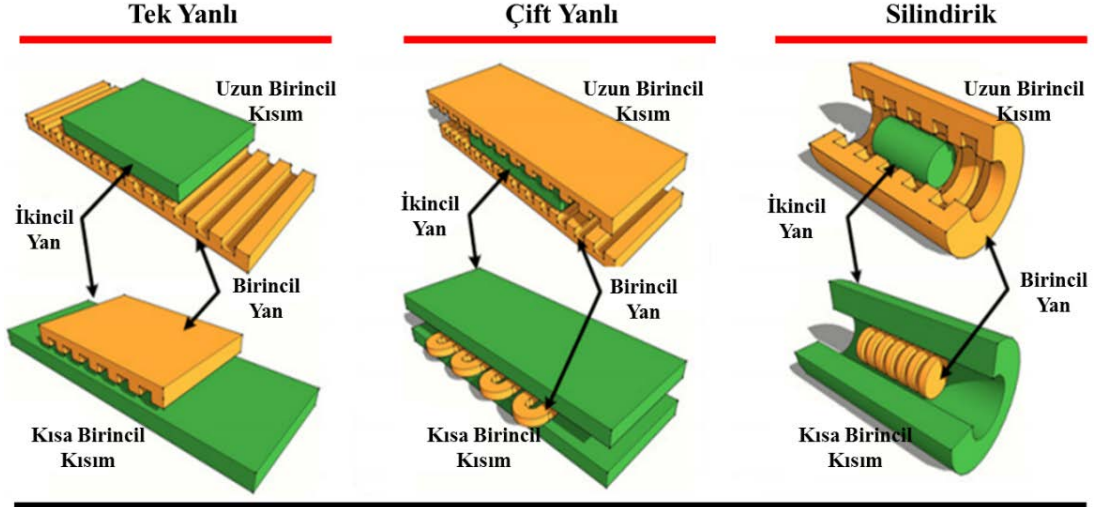


Şekil 3. Doğrusal hareketli motor türleri

DHM'ler en çok kullanılan türleri Doğrusal Hareketli Senkron Motor (DHSM) ve Doğrusal Hareketli Asenkron Motorlardır (DHAM) [11].

DHM'lerin yapısal farklılık gösteren türleride bulunmaktadır. Bunlar tek yanlı, çift yanlı ve silindirik modeller olarak sınıflandırılabilir. Şekil 4'de gösterilen bu modeller, Birincil Yan ile İkincil Yan arasındaki uzunluk farkına bağlı olarak, kısa Birincil Yanlı DHM'ler veya kısa İkincil Yanlı DHM'ler olarak sınıflandırılabilir.

DHM'lerin Yapısal Sınıflandırılması



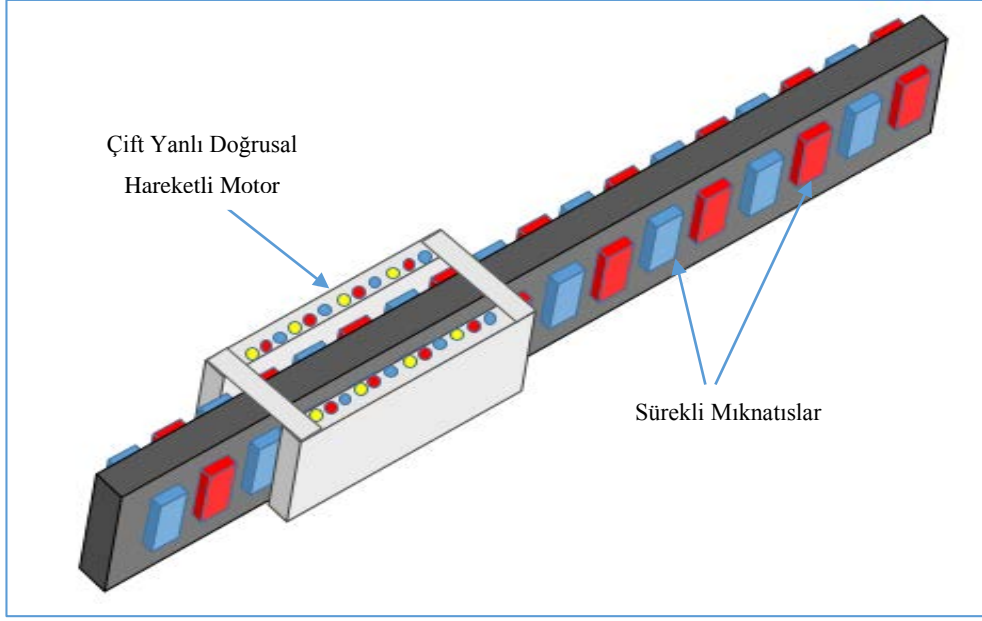
Şekil 4. Doğrusal hareketli motorların yapısal sınıflandırılması[10]

Şekil 4’de DHM’lerin yaygın olarak bilinen yapısal sınıflandırması görülmektedir. DHM’ler silindirik ve düz yapılı olmak üzere iki tür olarak yapılırlar. Düz yapılı DHM’nin Birincil Yanı, hem tek yanlı hem de çift yanlı olarak yapılmaktadır ve sargılar Birincil Yan üzerinde bulunmaktadır. İkincil Yan’da motor çeşidine göre; asenkron motorlarda alüminyum tabaka, senkron motorlarda yumuşak demir üzerine sıralanmış sürekli mıknatıslar bulunmaktadır [10].

Günümüzde, yüksek verimlilik ve güç yoğunluğuna sahip, hızlı tepki veren DHSMSM’ler tercih edilmektedir [11]. Bu motorların demir çekirdekli ve demir çekirdeksiz olmak üzere iki tasarım türü bulunmaktadır. Demir çekirdeksiz olan türlerde sargıların bulunduğu bir çekirdek yoktur. Bu durumda motorda olumsuz uç etkileri oluşmaz. Böylece motorun hız denetimi kolaylaşır ancak bu motorların çıkış gücü düşüktür. Demir çekirdekli olanlarda ise olumsuz uç etkileri oluşmasına rağmen sargıların oluşturduğu manyetik alan motorun itme kuvvetini ve çıkış gücünü artırmaktadır [4], [11].

Bu tezde, yüksek hızlanma kabiliyeti, yüksek itme kuvveti ve motor gücü [12], [13] diğer türlerine göre daha fazla olan, aynı zamanda hassas pozisyonlama yapılabilen [10], [11], Çift Yanlı Doğrusal Hareketli Sürekli Mıknatıslı Senkron Motor (ÇYDHSMSM) incelenecektir.

Bu motora ait benzer bir model, taşıma ve otomasyon sistemleri için geliştirilmiş ve Şekil 5’de gösterilmiştir.



Şekil 5. İncelenen motor modeli

1.1.3. Doğrusal Hareketli Motorların Uygulama Alanları

DHM'lerin uygulama alanları teknolojiye bağlı olarak giderek artmaktadır. Bu motorlar özellikle yüksek hız, yüksek ivme ve yüksek hassasiyete ihtiyaç duyulan tüm sistemlerde önemli ölçüde kullanılmaya başlanmıştır. Bunun yanı sıra DHM'ler aşağıda belirtilen birçok uygulama alanında yer almaktadır [14]:

- Taşıma ve otomasyon sistemlerinde kullanılmaktadır. Şekil 6 ve 7'de bu sistemlere ilişkin örnekler görülmektedir.



Şekil 6. Taşıma sistemlerinde kullanılan doğrusal hareketli motor



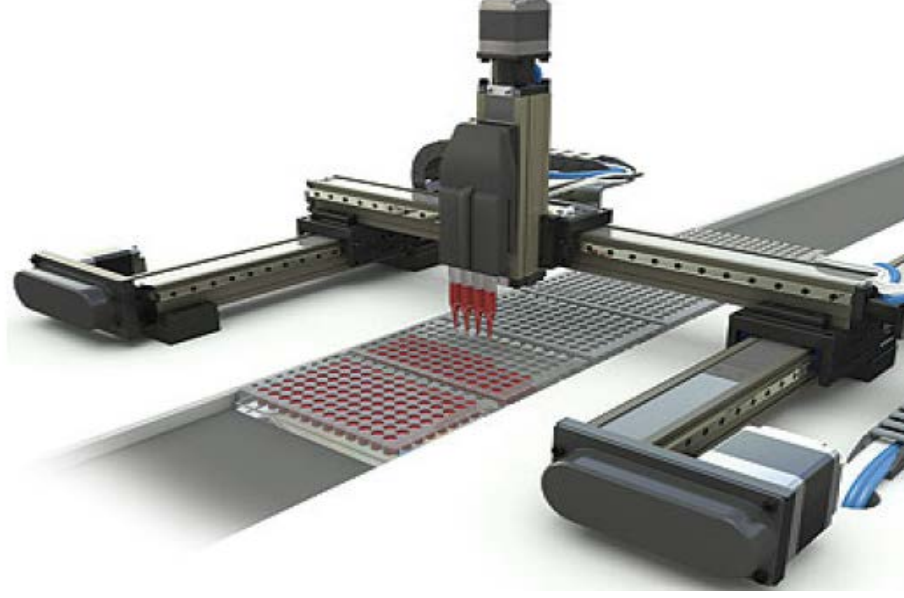
Şekil 7. Otomasyon sistemlerinde kullanılan doğrusal hareketli motor

- Tıp alanında kullanılmaktadır. Yüksek tekrarlılık, mikron altı çözünürlük ve büyük ivmelerin kullanıldığı uygulamalarda tercih edilmektedir. Şekil 8’de tıp alanında kullanılan bir Manyetik Rezonans (MR) cihazı görülmektedir.



Şekil 8. Tıp alanında kullanılan manyetik rezonans cihazı

- Robotik sanayinde (Şekil 9), Taşıma bantlarının sürülmesinde, Gemi geliştirmede modelin hareketinde, Uçak ivmelendirme, Kapı açma- kapamada, Tekstil sektöründe mekiklerde kullanılmaktadır.



Şekil 9. Robotik sanayisinde X-Y ekseninde hareket

- Metrolojide çeşitli ölçüm araçlarının imalatında ve ölçümlerde kullanılmaktadır.
- İmalat makinelerinde özellikle CNC (Computer Numerical Control- Bilgisayarlı Nümerik Denetim) tezgâhlarında, lazerli kesme tezgâhlarında ve takım tezgâhlarında tablanın veya kesicinin hareketinde kullanılmaktadır. Şekil 10’da bir CNC cihaz görülmektedir.



Şekil 10. İmalat sanayisinde kullanılan CNC cihazı

- Uzay arařtırmalarında kullanılmaktadır.
- Yürüyen merdiven köprülerin doğrusal hareketinde kullanılmaktadır. Şekil 11’de bir yürüyen merdiven görölmektedir.



Şekil 11. Doğrusal hareketli yürüyen merdiven

- Baskı teknolojisinde büyük ebatlı, yüksek çözünürlüklü yazıcılarda, doğrusal hareketli lazer tarayıcılarda, kullanılmaktadır. Şekil 12’de bir tarayıcı görölmektedir.



Şekil 12. Doğrusal hareketli lazer tarayıcı

- Elektronik sanayisinde, elektronik kartlarda devre elemanlarının yerlerine yerleştirilmesi, gibi işler DHM’ler ile seri bir şekilde yapılabilmektedir. Şekil 13’de baskı devresi elektronik devre elemanlarını yerleştiren DHM görölmektedir.



Şekil 13. Elektronik elemanları baskı devresine yerleştiren doğrusal hareketli motor

- Ulaşımında, hızlı tren diye bildiğimiz MAGLEV (Manyetik Levitasyon) uygulamalarında kullanılmaktadır. Şekil 14’de bir hızlı tren uygulaması görülmektedir.



Şekil 14. Doğrusal hareketli hızlı tren uygulaması

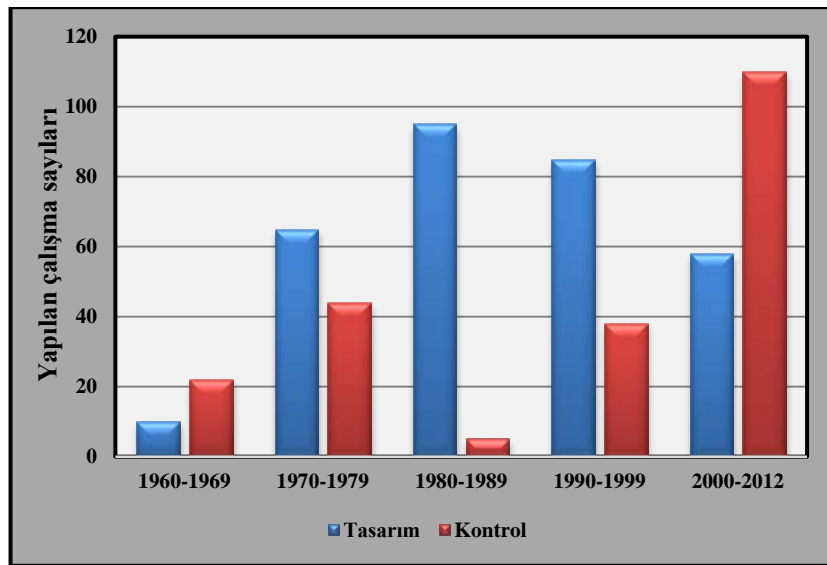
1.2. Konuyla İlgili Yapılmış Çalışmalar

DHM'lerin tarihi 19. yy sonlarına kadar dayanmaktadır. Bu makineler daha sonraki 30–40 yıl boyunca pratikte kullanılamamış ancak DHM konusunda çeşitli araştırmalar yapılmıştır [15], [16].

İlk DHM, Charles Wheatstone tarafından 1845 de yapılmış olan bir relüktans makinesidir ve hemen ardından benzer makine Henry Fox Talbert tarafından yapılmıştır. Şimdiye kadar yapılan çalışmalarda – bir kısım araştırmacılar konu ile ilgili farklı tarihler vermesine rağmen- Doğrusal Hareketli Asenkron Motor (DHAM) fikri Tesla, Dolivo-Dobrovolsky ve Ferrari'nin 1885 yılında döner alanlı makineyi icat etmesiyle hemen hemen aynı zamana denk gelmektedir. 1889 da Amerikalı Schuyler S. Wheeler and Charles S. Bradley demiryolu sistemlerinde Doğrusal Hareketli Senkron ve Asenkron Motor uygulamaları yapmış ve patent başvurusunda bulunmuştur [17].

1935 ve 1945 yılları arasında Hermann Kemper tarafından DHM'li Maglev (Manyetik Levitasyon) trenler hakkında bir dizi patent çalışmaları yayınlanmıştır. 1940'ların sonlarında, Londra'daki Imperial College'dan Profesör Eric Laithwaite ilk tam boyutlu bir çalışma modeli geliştirmiştir [14].

1946 yılında Westinghouse şirketi tarafından uçak fırlatıcı sistemlerde ilk kez çok amaçlı DHAM kullanılmıştır. 1954'lü yıllarda uçak geliştirme çalışmalarında rüzgâr tüneline alternatif olarak kullanılmıştır [16]. Şekil 15'deki grafikler [18] DHM'lerin geçmişten günümüze çalışma alanlarını göstermektedir.



Şekil 15. 1960 yılından günümüze doğrusal hareketli motor çalışmaları

1960'lı yıllardan sonra DHM'lerin uygulamaları hızla gelişmiş ve çok hızlı trenlerin sürülmesinden, manyetik yastıklara kadar pek çok sayıda uygulama alanı bulmuştur [16]. Şekil 15'e bakıldığında 1960-2000 yılları arasında genellikle tasarım çalışmalarının yapıldığı, 2000'li yıllardan sonra ise denetim çalışmalarında önemsenecek bir artışın olduğu görülmektedir.

Ancak, DHM'lerin hız ve konum denetimleri zor ve karmaşıktır. Denetim sistemlerinden beklenen; sürekli ve geçici durum hız ve konum davranışlarının oldukça düzgün olması, değişken hızlara iyi cevap vermesi, sistemin performansının harici kuvvet (yük torku) gibi bozucu giriş ve parametre değişimlerinden etkilenmemesidir. Bu nedenle günümüze kadar elektrik motorları için çeşitli hız denetim yöntemleri geliştirilmiş ve halen bu çalışmalara devam edildiği görülmektedir [19], [20]. DHM sürekli durum modeli ile skaler denetim, dinamik modeli ile vektör denetim yapılmaktadır. Skaler denetim düşük performanslı pek çok endüstriyel sistemler için yeterli olurken, yüksek performanslı uygulamalarda vektörel denetim yöntemleri tercih edilmektedir [21], [22].

DHM'lerin hız ve konum denetimini gerçekleştirebilmek için doğru konum ve hız sinyallerine ihtiyaç vardır. Uzun birincil yanlı DHM'lerde motor konumunun bilinmesi gerekir. Kısa birincil yanlı motorlarda konum bilgisi en uygun itme kuvvetinin bulunmasına yardımcı olmaktadır. Hız ve konum bilgileri için sensörler, manyetik lineer enkoderler kullanılır. Ancak bu sensörlerin, motor ve denetim ünitesi arasındaki bağlantıların artması, sensörlerin çevresel faktörlerden etkilenmesi (ısı, nem vb.), sistem maliyetini arttırması vb. gibi bazı sakıncaları bulunmaktadır. Bu sakıncaları ortadan kaldırmak için sensörsüz denetim yöntemleri giderek önemini arttırmaktadır [23].

Itoh [24] vd. sürekli mıknatıslı senkron motor hız denetimi için v/f denetim ve konum sensörsüz vektör denetim yöntemleri üzerinde çalışmışlar. Düşük maliyetli ve basit olan v/f denetimin yüksek hızlarda vektör denetimden daha iyi sonuçlar verdiğini ancak vektör denetimin sifıra yakın düşük hızlarda v/f denetime göre daha iyi sonuçlar verdiğini belirtmişlerdir.

Perera [25] vd. sürekli mıknatıslı senkron motorlarda sensörsüz kararlı v/f denetim üzerine çalışmışlar. Yaptıkları çalışmada motor manyetik akısını sabit tutmak için stator gerilimini sabit tutmayı önermişlerdir. Sonuç olarak, nominal hızın %5 ile %100 arasında daha iyi sonuçlar alındığını belirtmişlerdir.

Zhao [26] vd. yüksek hızlı sürekli mıknatıslı senkron motorlar için en uygun skaler denetim (v/f) tasarımı üzerinde çalışmışlar. Burada motorun sargı direncinin azaltılmasıyla skaler denetimde pozitif sonuçlar alındığını belirtmişlerdir.

Kavuran [27] ve Orhan, yüksek hızlı sürekli mıknatıslı senkron motorlar için frekans denetimde matris çevirici kullanarak, başka bir denetleyiciye ihtiyaç duymadan olumlu sayılabilecek sonuçlar aldıklarını belirtmişlerdir.

Jafari [28] vd. sürekli mıknatıslı senkron motorlarda skaler denetim performansını arttırmak için motor kayıplarını azaltmayı amaçlamışlardır. Burada motor parametrelerine ihtiyaç duymadan sadece motor giriş sinyalleri ile reaktif gücü azaltarak i_d akımını normal v/f denetimden elde edilen değerden daha minimize edilmiş bir değere indiğini tespit etmişler ve v/f denetim için bu yöntemi önermişlerdir.

Bilinen denetim yöntemlerinde [22] matematiksel modellerin doğru bir şekilde elde edilmesi oldukça önemlidir. Geleneksel denetleyiciler endüstriyel alanda çok fazla kullanılmasına rağmen, sistemde modellenemeyen parametrelerden ötürü istenilen performansı gösteremezler [29]. Bu anlamda standart denetim yöntemleri, transfer fonksiyonlar ya da durum uzay denklemleri ve sabit parametrelere sahiptir. Ancak elektrik motorlarının parametreleri değişkendir. Yani sargı dirençleri sıcaklığa, sargı indüktansları manyetik doyuma bağlıdır. Bu durumun denetim sisteminin dinamik davranışının bozulmasına ve kararsızlığa yol açtığı görülmektedir [30]- [33]. Bu nedenle yapay sinir ağları (YSA), bulanık mantık denetleyiciler vb. gibi matematiksel model gerektirmeyen akıllı denetim sistemlerinin tercih edildiği görülmektedir [34]- [40].

YSA, karmaşık sistemlerin modellenmesinde yaygın olarak kullanılan bir yapay zekâ tekniğidir ve insan beynindeki biyolojik sinir hücrelerinin basitleştirilmiş olarak modellenmesine dayanmaktadır. YSA, belirlenen sonucu elde etmek için hücreler arasındaki bağlantı ağırlıklarını çeşitli öğrenme yöntemleri ile değiştirebilen ve en uygun ağırlıkları bulmak için çalışan paralel bir işlemcidir [41].

Doğrusal olmayan problemlerin çözümü için YSA en uygun yöntem olarak görülmektedir [35], [36]. 1990 yılından sonra kullanımı hızla artan YSA uygulamaları, son yıllarda güç elektroniği tekniklerine göre geliştirilmiş ve AC motorların hız, konum denetimleri için önerildiği görülmektedir [38], [39], [42]. Sürekli mıknatıslı senkron motorun hız denetim çalışmasında geleneksel yöntemlerin dışında, YSA ile yapılan denetimin, sistemde meydana gelen durum değişimine diğer denetim yöntemlerinden daha hızlı cevap verdiği anlaşılmaktadır [43], [44].

YSA'nın doğrusal olmayan yapıları, uyarılma ve genelleme yeteneklerine sahip olmasından dolayı asenkron motorların hız, moment ve akı denetimlerine ilişkin çeşitli çalışmalar yapılmıştır [45, 46, 47, 48].

Wei [49] vd. doğrusal hareketli sürekli mıknatıslı senkron motor denetiminde PI ve YSA+PI olmak üzere iki yöntem üzerinde çalışmalar. Motorun YSA+PI denetimde durum incelemesi yapılmış ve bu yöntemin diğerine göre daha hızlı cevap verdiğini tespit etmişlerdir.

Hamidia [50] vd. Kalman filtreli DHSMSM için, tork ve hız denetimini PI ve YSA ile benzetim modelleri ile gerçekleştirmişler. Benzetim sonuçlarına göre geleneksel PI denetim yerine YSA ile denetimi önermişlerdir.

Gaur [51] vd. Sürekli mıknatıslı senkron motorlarda PI ve YSA ile hız denetimi çalışmalarında sistemin davranışına, ara denetleyicinin olamadığı YSA ile yapılan denetimin, PI denetimden daha iyi cevap verdiğini tespit etmişlerdir.

Kumar [36] vd. YSA'nın yapısı motorun matematiksel modelinden elde etmişler ve YSA'da geri yayımlı öğrenme yöntemini kullanarak sürekli mıknatıslı senkron motorun hız denetimini başarılı bir şekilde gerçekleştirmişlerdir. YSA motorun zamanla değişen yük ve parametrelerini başarıyla denetlediğini belirtmişlerdir.

Öte yandan birçok uygulamada, DHM'lerin hız kontrolü için tork kontrol esas alındığı görülmektedir. Özellikle elektrikli araçlarda tork kontrol sıklıkla kullanılmaktadır. Yine tork denetim için bazı araştırmacılar vektör denetimi kullanmışlardır [52].

Krishnan [52] vd. YSA ile yaptıkları hız denetim çalışmasında, akı ve tork denetimi üzerinde çalıştıkları görülmektedir. Akı ve tork denetiminin YSA ile gerçekleştirilmesinde başarılı sonuçlar elde ettiklerini belirtmişlerdir.

Chaoui [53] vd. sürekli mıknatıslı senkron motorlar için YSA ile hız gözlemcisi denetimi üzerine simülasyon olarak çalıştıkları görülmektedir. Yapılan çalışmada uzay vektör PWM denetime göre daha iyi sonuçlar elde ettikleri görülmektedir. Yapılan teorik çalışmada yüksek performanslı sürücülerde YSA'yı motorların hız denetimi için önermektedirler.

Wang [54] vd. sürekli mıknatıslı senkron motor denetimi için, PID ile vektör denetim ve YSA ile vektör denetim çalışmalarını simülasyon olarak gerçekleştirmişler. YSA tabanlı vektör denetim çalışması beklenen sonuçları verdiğini ve bu denetim çalışmasının önerildiğini belirtmişlerdir.

1.3. Tezin Amacı

Bu çalışma, taşıma ve otomasyon sistemlerinde kullanılan, teknolojideki gelişmelere bağlı olarak önemini giderek arttıran Doğrusal Hareketli Sürekli Mıknatıslı Senkron Motorun (DHSMSM) hız denetimini, maliyeti düşük, makine ve yük parametrelerinden bağımsız, insan beyni gibi düşünebilen bir akıllı denetim sistemiyle gerçekleştirmek ve denetim alanındaki çalışmalara katkıda bulunmak amacıyla gerçekleştirilmiştir.

Bu çalışmada, incelenen motorun hız denetimi Yapay Sinir Ağları (YSA) ile gerçekleştirilmiştir. YSA ile motor arasındaki bağlantıda, değişken frekanslı denetim yapabilen ve modeli LG-SV022İG5A-4 olan sürücü kullanılmıştır. DHSMSM'nin değişken frekanslı hız denetimi, çok katmanlı ileri beslemeli sinir ağı (Multilayer Feedforward Network) ile gerçekleştirilmiştir. Ağın eğitiminde hata geri yayılım algoritması (Error Back-Propagation Training) kullanılmıştır. YSA, programa girilen verilere göre (Yol veya Hız) arka planda frekans değerini hesaplar ve anlık frekans değeriyle karşılaştırır. Anlık frekans istenen hız değerine ulaşmak için yeterli değilse YSA frekansı arttırır ve motorun istenen hız değerine ulaşması sağlanır. Ters durumda YSA frekansı azaltarak motorun istenen hıza ulaşmasını sağlar. Yapılan döngüde programa girilen zaman değeri sıfır olduğunda motorun durması sağlanmıştır.

Bu amaçla hazırlanan bu tezin;

Birinci Bölümünde; DHM'lar hakkında genel bilgilere yer verilmiş ardından tezin amacına yönelik şimdiye kadar yapılmış denetim çalışmaları aktarılmıştır.

İkinci Bölümünde; DHSMSM yapısal özellikleri ve çalışma ilkeleri izah edilmiştir. Diğer yandan, hız denetiminde çok önemli olduğu bu sistemlerde, kullanılan geleneksel denetim yöntemleri, bu denetim yöntemlerinin üstünlükleri ve sakıncaları üzerinde durulmuştur.

Üçüncü Bölümünde; Akıllı denetim yöntemi olan YSA'nın yapısı, öğrenme yöntemleri üstünlükleri hakkında bilgiler verilmiştir.

Dördüncü Bölümünde; Hız denetimi yapılacak motorun imalatı, motorun hız denetiminin değişken frekanslı sürücüyle gerçekleştirilmesi, YSA ile yapılan hız denetim çalışmaları izah edilmiş ve çalışmalardan elde edilen bulgular verilmiştir.

Beşinci Bölümünde; Çalışma sonucunda elde edilen bulguların değerlendirmesi yapılmıştır.

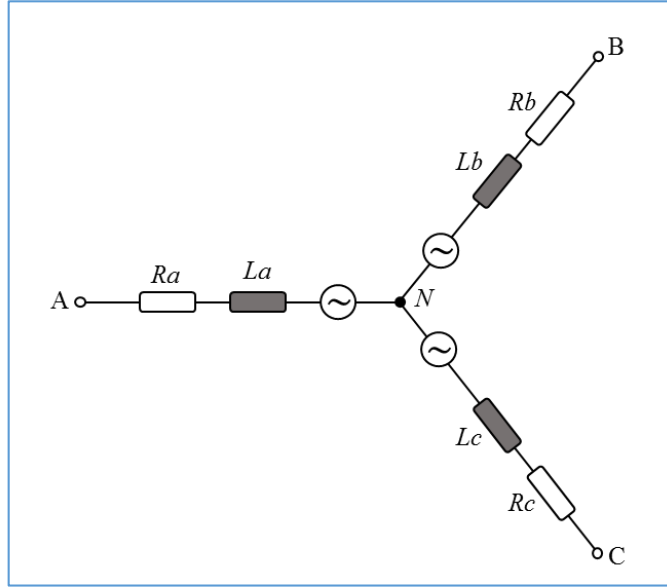
Altıncı Bölümde; Önerilere yer verilmiştir.

2. ÇİFT YANLI DOĞRUSAL HAREKETLİ SÜREKLİ MIKNATISLI SENKRON MOTORLAR (ÇYDHSMSM)

Bu bölümde, motora ait eksen sistemi dönüşümleri, motorun matematiksel denklemleri, kullanılan sürekli mıknatıs özellikleri, motor için kullanılan sarım teknikleri ve bu motorlarda kullanılan denetim yöntemlerinden bahsedilecektir.

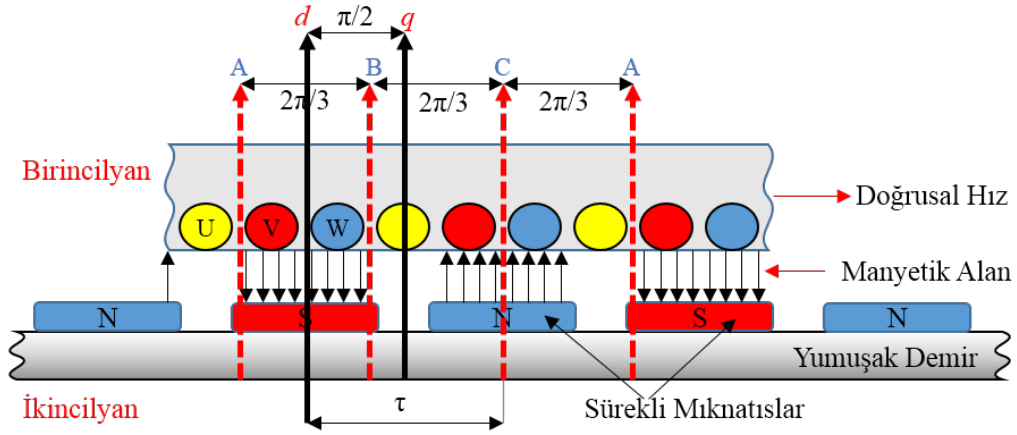
2.1. Uzay Vektör Denklemleri, Eksen Sistemleri ve Dönüşümleri

ÇYDHSMSM'nin ABC modeli oluşturulurken Şekil 16'da gösterilen eşdeğer devre kullanılmıştır. Birincil Yan sargıları 120° faz farkı ile yerleştirilmiştir. Her bir faz sargısının eşit ve her bir faza ait R sargı direncinin de eşit olduğu kabul edilmiştir [11].



Şekil 16. Üç fazlı eşdeğer devre modeli

Tasarlanan model için makinenin 3 fazlı yıldız bağlı Birincil Yan sargıları ve sürekli mıknatıslı bir İkincil Yan'a sahip olduğu Şekil 17'de verilen elektriksel devrede görülmektedir.



Şekil 17. Doğrusal hareketli sürekli mıknatıslı senkron motorun elektriksel modeli [55]

Motorun davranışını geçici ve kararlı rejimde temsil eden matematiksel model, hesaplama kolaylığı açısından uzay vektörleri kullanılarak tanımlanmıştır.

Sistemin parametrelerini anlık dengelenmiş olarak kabul edelim. Bu durumda [11]:

$$\vec{S}_A + \vec{S}_B + \vec{S}_C = 0 \quad (1)$$

Burada S vektörel olarak Birincil yan akım, gerilim ve manyetik akı'yı ifade etmektedir. Bu parametrelerin uzay vektörleri aşağıdaki gibi tanımlanabilir.

$$\vec{S} = k (\alpha^0 S_A + \alpha^1 S_B + \alpha^2 S_C) \quad (2)$$

Burada,

α : uzay vektör operatörü

k : transformasyon sabitidir.

$$\alpha = e^{j\frac{2\pi}{3}} \quad (3)$$

$k = 2/3$, olarak alınmıştır [56].

Sistem parametrelerinin anlık bir değerinde, birincil yan akımı (i), gerilimi (u) ve manyetik akı (ψ) olarak adlandırılan alan vektörlerinin oluşturulması için kullanılabilir.

$$\vec{i} = \frac{2}{3} [\alpha^0 i_A + \alpha^1 i_B + \alpha^2 i_C] \quad (4)$$

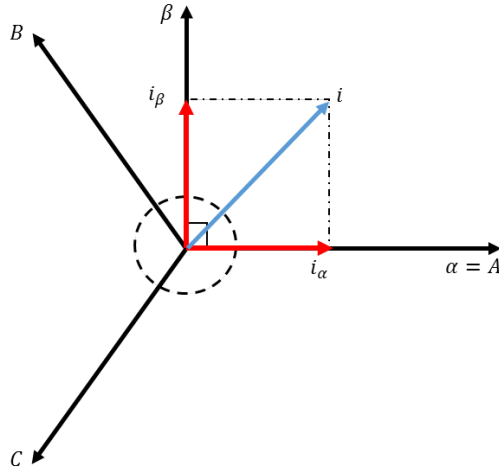
$$\vec{u} = \frac{2}{3} [\alpha^0 u_A + \alpha^1 u_B + \alpha^2 u_C] \quad (5)$$

$$\vec{\psi} = \frac{2}{3} [\alpha^0 \psi_A + \alpha^1 \psi_B + \alpha^2 \psi_C] \quad (6)$$

Üç fazlı alternatif akım motorlarında yüksek performanslı sürücü geliştirmek için faz düzlemleri arasında dönüşüm gerçekleştirilir. Faz dönüşümleri kullanılarak motorun dinamik eşitliklerinin çözümü daha hızlı ve kolay olmaktadır. Faz dönüşüm işlemleri [11];

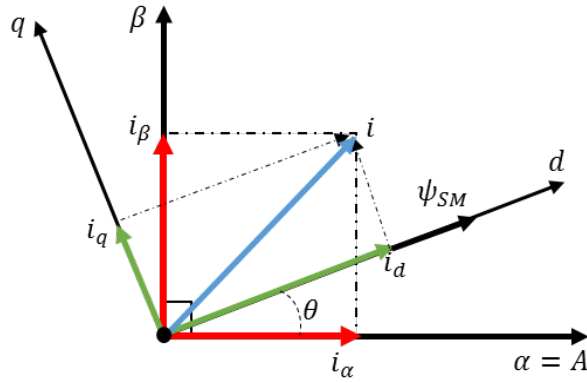
- $ABC \rightarrow Clark \rightarrow \alpha\beta \rightarrow Park \rightarrow dq$
- $dq \rightarrow TersPark \rightarrow \alpha\beta \rightarrow TersClark \rightarrow ABC$

şeklinde gerçekleştirilir. Şekil 18’de $ABC \rightarrow \alpha\beta$ eksen sistemine ait referans düzlem gösterilmiştir.



Şekil 18. $ABC-\alpha\beta$ referans düzlemi [11]

Şekil 19’da $\alpha\beta \rightarrow dq$ eksen sistemine ait referans düzlem gösterilmiştir.



Şekil 19. $\alpha\beta-dq$ referans düzlemi [11]

Verilen referans düzlemlere göre $ABC \rightarrow dq$ eksen sisteminin matematiksel ifadesi aşağıdaki denklemde görülmektedir.

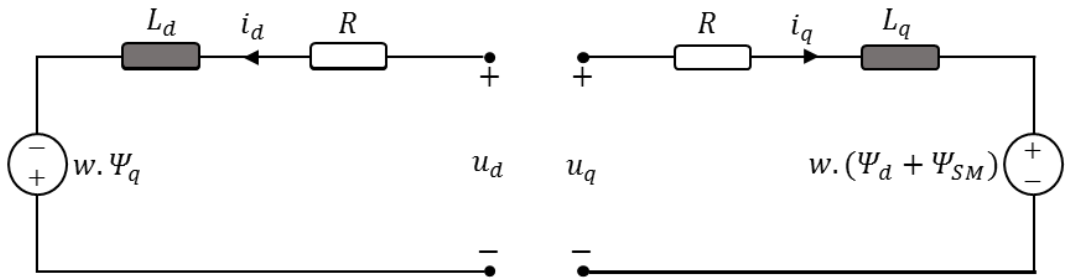
$$\begin{bmatrix} S_d \\ S_q \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \cos(\theta) & \cos\left(\theta - \frac{2\pi}{3}\right) & \cos\left(\theta + \frac{2\pi}{3}\right) \\ \sin(\theta) & \sin\left(\theta - \frac{2\pi}{3}\right) & \sin\left(\theta + \frac{2\pi}{3}\right) \end{bmatrix} \begin{bmatrix} S_A \\ S_B \\ S_C \end{bmatrix} \quad (7)$$

Aynı şekilde $dq \rightarrow ABC$ eksen sisteminin matematiksel ifadesi de aşağıdaki denklemde gösterilmektedir.

$$\begin{bmatrix} S_A \\ S_B \\ S_C \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ \cos\left(\theta - \frac{2\pi}{3}\right) & \sin\left(\theta - \frac{2\pi}{3}\right) \\ \cos\left(\theta + \frac{2\pi}{3}\right) & \sin\left(\theta + \frac{2\pi}{3}\right) \end{bmatrix} \begin{bmatrix} S_d \\ S_q \end{bmatrix} \quad (8)$$

2.2. Motorun Matematiksel Modeli

Çift Yanlı Doğrusal Hareketli Sürekli Miknatıslı Senkron Motorun (ÇYDHSMSM) matematiksel modeli oluşturulurken, manyetik devresinin doğrusal olduğu varsayılmıştır [57]. Bu anlamda Döner Hareketli Senkron Motorun (DÖHSM) iki eksen takımlı (d-q) matematiksel modeli ile benzer kabul edilerek yaygın olarak kullanılan matematiksel ifadeler aşağıda verilmiştir [58]. DHSMSM'nin bilinen d - q eşdeğer devresi Şekil 20'de verilmiştir.



Şekil 20. Doğrusal hareketli motorun d - q modeli

Üç fazlı senkron motorların denetim algoritmaları oluşturulurken genellikle d - q eksen sistemi modeli kullanılır.

d - q eksen sisteminde gerilim denklemleri:

$$u_d = Ri_d + \frac{d\psi_d}{dt} - w\psi_q \quad (9)$$

$$u_q = Ri_q + \frac{d\psi_q}{dt} + w\psi_d \quad (10)$$

Burada, u_d ve u_q Birincil Yan gerilimlerini, i_d ve i_q Birincil Yan akımlarını, R Birincil Yan sargı direncini göstermektedir. w açısal hızı göstermektedir.

$$w = 2\pi f = v \frac{\pi}{\tau} \quad (11)$$

Burada, τ kutup adımı, v doğrusal hızı göstermektedir. Doğrusal hız:

$$v = 2f\tau \quad (12)$$

d - q eksenlerindeki akı denklemleri:

$$\psi_d = L_d i_d + \psi_{SM} \quad (13)$$

$$\psi_q = L_q i_q \quad (14)$$

Burada, L_d ve L_q Birincil Yan sargı indüktanslarını ve ψ_{SM} sürekli mıknatıs akısını göstermektedir. Üç fazlı Birincil Yan sargılarının anlık güç girişi [11]:

$$P = u_A i_A + u_B i_B + u_C i_C = \frac{3}{2} (u_d i_d + u_q i_q) \quad (15)$$

Motora ait üç fazlı elektromanyetik güç:

$$P_{el} = \frac{3}{2} w (\psi_d i_q - \psi_q i_d) = \frac{3}{2} w [\psi_{SM} + (L_d - L_q) i_d] i_q \quad (16)$$

Buradan denklem 17'de DHSMSM'nin elektromanyetik itme kuvveti elde edilir.

$$F_{itme} = \frac{3}{2} p \frac{\pi}{\tau} (\psi_d i_q - \psi_q i_d) = \frac{3}{2} p \frac{\pi}{\tau} [\psi_{SM} + (L_d - L_q) i_d] i_q \quad (17)$$

DHSMSM'nin mekanik itme denklemi:

$$F_{itme} = F_h + Bv + M \frac{dv}{dt} \quad (18)$$

Burada, F_h harici (dış) kuvvet, B sürtünme kuvveti, M motorun ağırlığıdır.

2.3. Doğrusal Hareketli Senkron Motorlarda Çalışmayı Etkileyen Etmenler

Döner tip makinelerde oluşmayan sadece DHM'lerde ortaya çıkan bazı bozucu etkiler bulunmaktadır. Bunlar enine (kenar), boyuna uç etkisi ve hava aralığı etkisidir [5], [59]. Bu uç etkileri motor boyunca manyetik akı dağılımını bozmaktadır.

2.3.1. Uç Etkileri

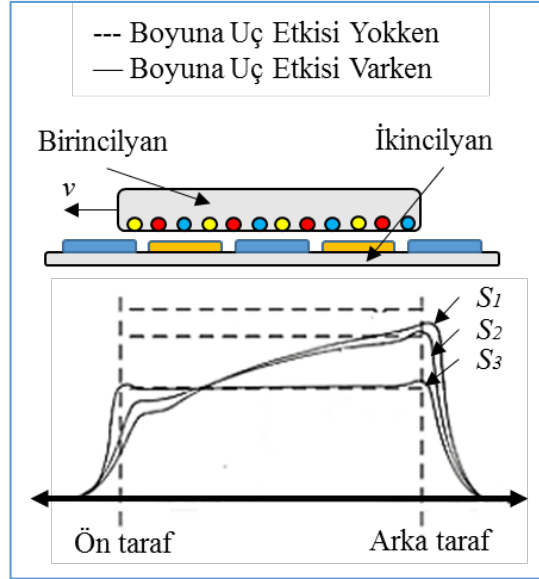
Genelde uç etkilerini matematiksel olarak ifade etmek oldukça zordur. Bu anlamda motorun manyetik alan analizi çok iyi yapılmalıdır. Araştırmacılar uç etkisini genellikle $k_{uç}$ katsayısı ile ifade etmişler ve deneysel yöntemle en iyi sonuçların elde edildiğini belirtmişlerdir. Aynı zamanda bu katsayının 0.01 olduğunu ileri sürmektedirler. Denklem 19 uç etkisi verilmiştir [1], [11], [60].

$$F_{uç_et} = k_{uç} \cdot F_{itme} \quad (19)$$

Burada $F_{uç_et}$ enine ve boyuna uç etkilerini ifade etmektedir.

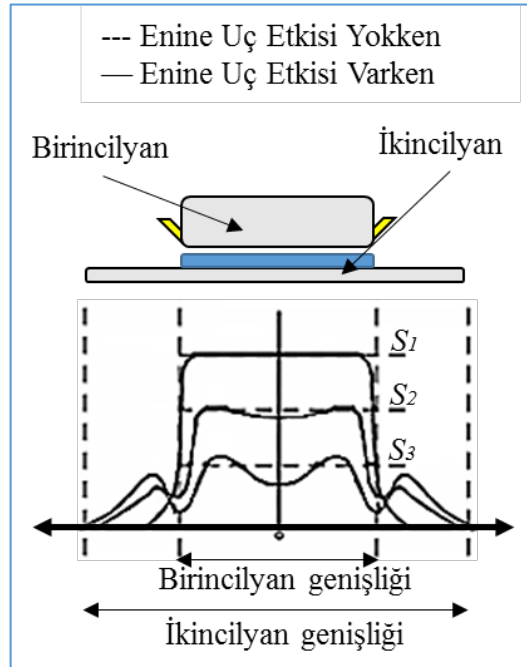
Değişen alan içinde kalan iletkenlerde gerilim indüklendiği bilinmektedir. Bu indüklenen gerilim etkisiyle ikincil kısımdan akım akar. Bu akımın meydana getirdiği alanın birincil kısma etkisi, motorun hareketini zorlaştırır, motordan alınan gücü, dolayısıyla momenti düşürür. Birincil kısımda oluşan uç etkisi, hareket doğrultusunda ve her iki uçta meydana gelir. Birincil kısım itme etkisi ile sürekli bir doğrusal hareket halindedir. Bu hareket sırasında, İkincil kısmın bir kısmı birincil kısmın altına girerken, bir kısmında Birincil kısım etkisinden uzaklaşmaktadır. Böylece birincil kısmın uç kısımlarındaki ikincil kısım bölgelerinde bir akı değişimi oluşur [59].

DHM'nin giriş kısmındaki uç etkileri yüksektir [5]. Çıkış kısmındaki uç etkileri giriş kısmına göre daha düşük seviyededir. Şekil 21'de boyuna uç etkisi gösterilmiştir.



Şekil 21. Boyuna uç etkisi grafiği

Birincil kısımda oluşan kısa devre akımlarının iki bileşeni vardır. Bu bileşenlerden hareket yönündeki bileşen, enine uç etkisini oluşturur. Şekil 22’de enine uç etkilerine ait grafikler gösterilmiştir. Burada enine uç etkisi ikincil yan direncini artırır ve mıknatıslanma reaktansının azalmasına yol açar. Aynı zamanda motordan alınan gücü azaltır [5].



Şekil 22. Enine uç etkisi grafiği

DHM’lerde uç ve kenar etkileri üzerine şimdiye kadar yapılan çalışmalarını aşağıdaki gibi özetleyebiliriz.

Gireas (1986) ve arkadaşları, DHM performans hesabı yaptılar ve boyuna uç etkilerini hesaba kattılar. Boyuna uç etkisinin hesabı için ‘T türü eşdeğer devrenin ortak empedans karşı düşen elektro motor kuvvetinin düzenlenmesi yönteminin’ en iyi sonucu verdiğini, analitik ve hesap olarak en basiti olduğunu öngördüler [61].

Nonaka (1987) ve Higuchi, DHM’nin birincil kısım çekirdeğinin, tasarımında uç etkilerini azaltmak ve ikincil kısım tepki rayının maliyetini azaltmak için, ince dar ve uzun tasarladılar. Uç etkisinin düşük hız uygulamalarında ciddi olmadığını belirttiler. Bu anlamda uç etkisini azaltmak için motor uzunluğunun mümkün olduğunca uzun yapılmasını vurguladılar [62].

Mori (1999) ve arkadaşları, DHM performansının, uç etkisinden dolayı azaldığını belirttiler. DHM uç etkilerinin analizinde Dalgacık dönüşüm tekniği ve Fourier serisi kullandılar. DHM’nin birincil kısmının sonlu uzunlukta olması uç etkisinin ortaya çıkmasına sebep olduğunu öngördüler [63].

Faiz (2000) ve Jafari, DHM tasarımını ve analizini zorlaştıran boyuna uç etkisi ve kenar etkisinin bu tür makinelerin iki önemli parametresi olduğunu, boyuna uç etkisinin motor performansında etki sahibi olduğu vurguladılar. Değişik analiz yöntemlerinin tasarlanmış olduğunu ve bunlardan ‘Fourier katman metodunun’ daha iyi sonuçlar verdiğini belirttiler [64].

Fujii (2002) ve arkadaşları, DHM için bir uç etki kompensatörü önerdiler ve DHM’nin çalışma kayması ile girdap akımı adımının ayarlanmadan sadece DHM’nin önünde DHM frekanslı girdap akım senkronizasyonu sağlanmasıyla uç etkisinin kompanze edilmesini amaçladılar. DHM’ye ait uç etkili ve uç etkisi kompanze edilmiş itme, verim ve akı dağılımlarını ortaya koydular [65].

Kang (2003) ve arkadaşları, DHM uç etkilerinin hesaba katılmadığı, dolayısıyla DHM matematiksel modelinin DÖHM modeli ile aynı olduğunu belirttiler. 18 Hz’den düşük frekanslarda uç etkisinin DHM’ler için ihmal edilebileceğini belirttiler [66].

Jamali (2003), DHM’lerde uç etkisini inceledi. Hareket yönünde demir çekirdeğin (stator, duran bölge ya da birincil kısım) sınırlı uzunlukta olması DHM’de uç etkisinin meydana getirdiği, ikincil kısım demir çekirdekten ayrılırken ikincil kısmın çıkış kenarında en büyük uç etkisini meydana geldiğini vurguladı [67].

DHM'lerin kenar etkileri birincil ve ikincil kısmın sonlu genişlikte olmasından kaynaklanır. Aynı zamanda kenar etkisi birincil kısmın akımının tepkimesi ve manyetik dirençteki ani artıştan dolayı, çekirdek kenarlarında y-ekseni boyunca hava aralığındaki manyetik akı yoğunluğunun düzensiz dağılımından kaynaklanır [5].

Son yıllarda kenar etkisiyle ilgili çalışmalara bakıldığında;

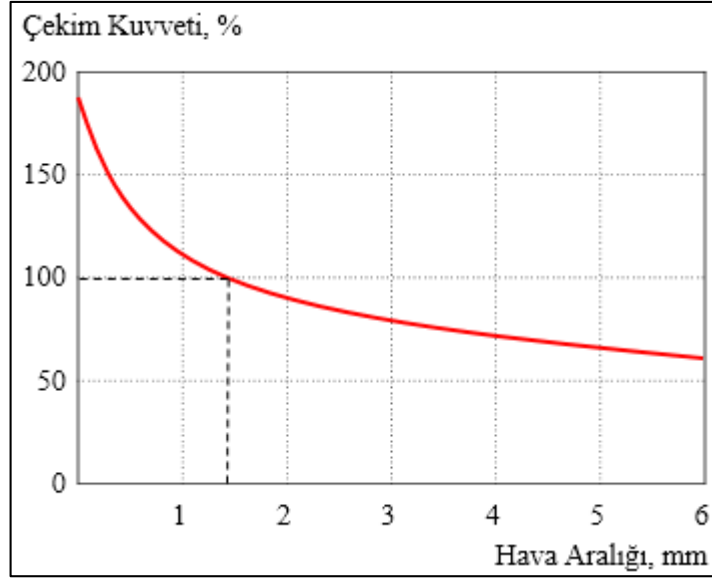
Faiz (2000) ve arkadaşları, DHM'ye ait olan manyetik alandaki süreksizliğin, döner tip motorlara göre kendine özel problemleri ortaya çıkardığını ve kenar etkisinin de problemlerden biri olduğunu belirttiler. Kenar etkisi hesaba katılarak tek ve iki boyutlu analizler yaptılar ve elde edilen sonuçların deneysel sonuçlar ile uyum içinde olduğunu gösterdiler [68].

Mirsalim (2002) ve arkadaşları, DHM'de meydana gelen akımların sekonder üzerinde izlediği özel bir yol olmadığını belirttiler. Kenar etkisinin eşdeğer sekonder direncini artırmakta kullanıldığını ve kenar etkisi gibi bazı etkilerin kayma ile DHM'nin parametrelerini değiştirdiğini öngördüler [69].

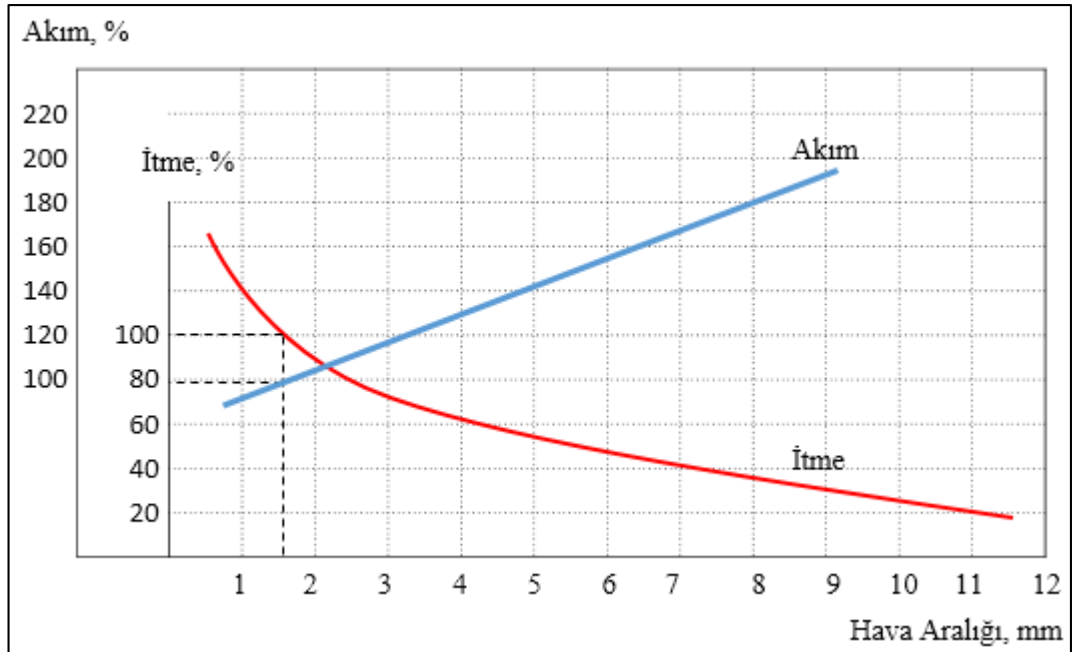
Amirkhani (2003) ve arkadaşı, Russell–Norsworthy düzeltme faktöründen yararlanılarak kenar etkisinin hesaba katıldığını ve bu faktörün sekonder direnci ve primer ile sekonder arasındaki ortak indüktansını düzeltilmesinde kullanıldığını belirttiler [70].

2.3.2. Hava Aralığı Etkileri

DÖHM çok küçük hava aralıklarına sahiptir. Bu durum hava aralığında yüksek akı yoğunluğu oluşturur. DHM'lerde ise hava aralıkları santimetrelere ulaşmaktadır. Geniş hava aralıklarında manyetik devre direnci yükselmektedir. Aynı zamanda mıknatıslanma akımı da yükselir. Bunun yanında güç katsayısını düşüren kaçak akımlar meydana gelmektedir. Şekil 23 (a)'da çekim kuvvetine karşı hava aralığı etkisi, Şekil 23 (b)'de hat akımı ve itmeye karşı hava aralığı etkisi gösterilmiştir [5], [71].



(a)



(b)

Şekil 23. Hava aralığının etkileri [5], a) Çekim kuvvetine karşı, b) Hat akımı ve itmeye karşı

Şekil 23 (a) ve (b)'den anlaşılacağı gibi hava aralığının artması durumunda itme kuvvetinde azalma, motor akımında ise yükselmeye neden olmaktadır.

Bu anlamda DHM'lerin boşluk etkisi ile ilgili yapılan çalışmaları aşağıdaki gibi özetleyebiliriz;

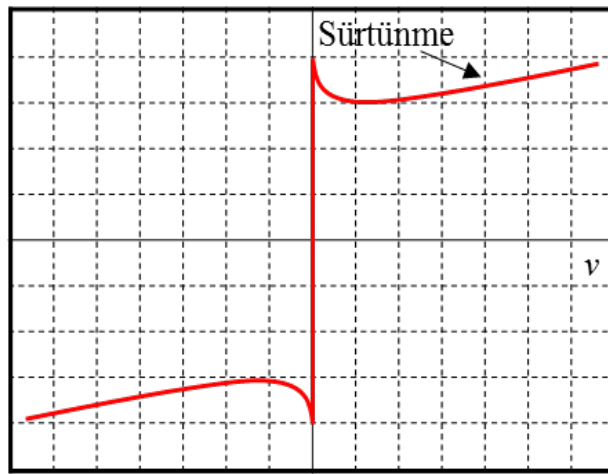
Levi (1973), Doğrusal Hareketli Senkron Motorun (DHSM), Doğrusal Hareketli Asenkron Motora (DHAM) göre üstünlüklerinden birinin de doğrusal senkron tahrikinin DHAM fazla olması olduğunu ve bununda cihaz ile ray kısmı arasındaki hava boşluğu mesafesinden kaynaklandığını belirtti. Bunun yanında manyetik akı yolu içinde hava aralığının iki katına çıkma sonucunda meydana geleceğini ve manyetik alan şiddetinin itme ile doğru orantılı olarak artma eğiliminde olduğunu öngördü [72].

Yoshida (1975) ve arkadaşı, akım denetimli DHM’de itme ve kaldırma kuvvetini hava aralığının değişimine göre incelediler. Hava aralığındaki artışın itme kuvvetini azalmakta olduğunu ve eğer geniş açık hava aralığı gerekli ise büyük değerlerde kayma değerinin seçilmesi gerektiğini öngördüler [73].

Kim (1992) ve arkadaşları, 8 adet eksenel akı tipi elektromıknatis ve 2 adet tek taraflı DHM’den oluşan KOMAG-01 adlı cihaz üzerinde akı dağılımını, kaldırma kuvvetini ve akım değişimini hava aralığına bağlı olarak incelediler. Akı dağılımında ve kaldırma kuvvetinde hava aralığının etkili olduğunu belirttiler [74].

2.3.3. Sürtünme Etkileri

Birincil Yan ve sürekli mıknatıslar arasında meydana gelen çekim kuvvetinin oluşturduğu sürtünme kuvveti DHM’ler için önemli bir olumsuz etkidir. Bu nedenle DHM tasarlanırken bu çekim kuvveti dikkate alınmalı ve oluşabilecek titreşim, gürültü, sürtünme hesaba katılmalıdır. Genel sürtünme modeli (Stribeck eğrisi) Şekil 24’deki gibidir [11], [75].



Şekil 24. Sürtünme modeli grafiği (Stribeck eğrisi) [11]

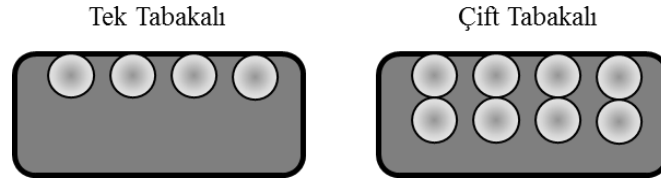
DHM'nin hızına bağlı olarak sürtünme kuvveti [11]:

$$F_{srt}(v) = C_{srt} \text{sign}(v) + V_{srt} v + S_{srt} e^{-kv} \text{sign}(v) \quad (20)$$

Burada, C_{srt} Coulomb katsayısı, V_{srt} Viscous katsayısı, S_{srt} Stribeck katsayısı, v doğrusal hız, k Stribeck hız faktörünü göstermektedir.

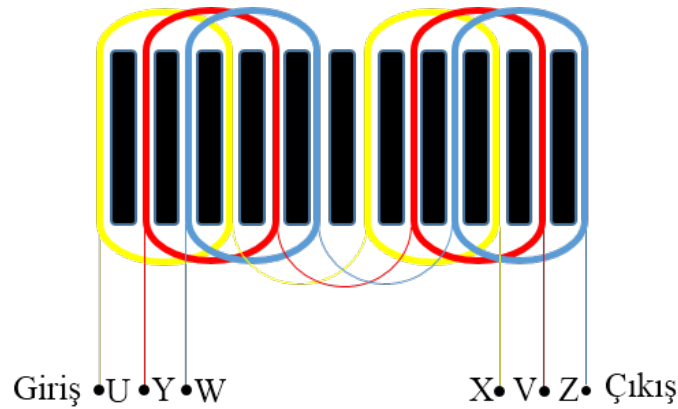
2.4. Motorun Sarım Teknikleri

Bilindiği gibi döner alan elde etmek için en az iki fazlı sargıya ihtiyaç vardır. Ancak yürüyen bir alan elde edebilmek için en az üç fazlı sargıya gerek duyulmaktadır. Üç fazlı sargıların yürüyen alan oluşturabilmesi için sargı akı eksenleri arasında $2\pi/3$ 'lük yer farkı ve sargı akımları arasında 120° 'lik faz farkı olmalıdır. Bu sargılar, tek ya da çift tabakalı olarak yapılabilir [1], [5], [76]. Şekil 25'de tek ve çift tabakalı sargı düzeni gösterilmiştir.

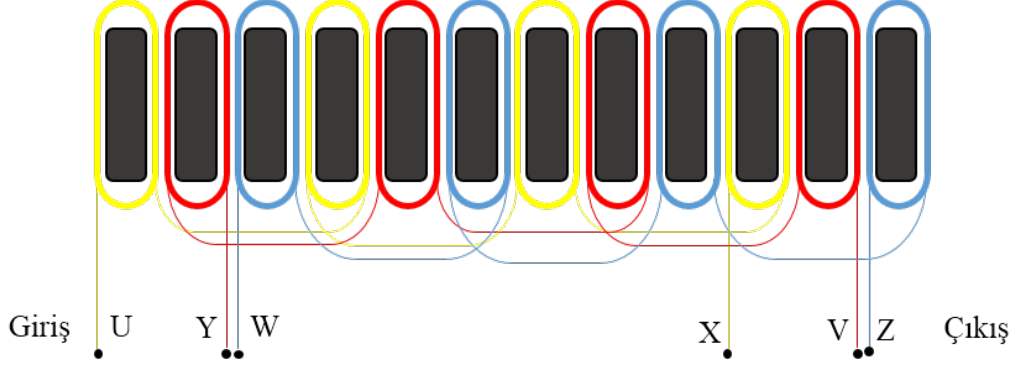


Şekil 25. Sargı düzeni

Üç fazlı motorlar için çok sayıda sarım şekli bulunmaktadır. DÖHM için kullanılan sarım şekilleri DHSM içinde önerilmiştir [30]. Burada pratikte ilgi gören modellerine yer verilmiştir. Şekil 26'da tek tabakalı şekil 27'de çift tabakalı sargı için önerilen modeller gösterilmiştir.



Şekil 26. Bir faza ait oluk sayısı, bir olan; tam adımlı tek tabakalı sargı düzeni



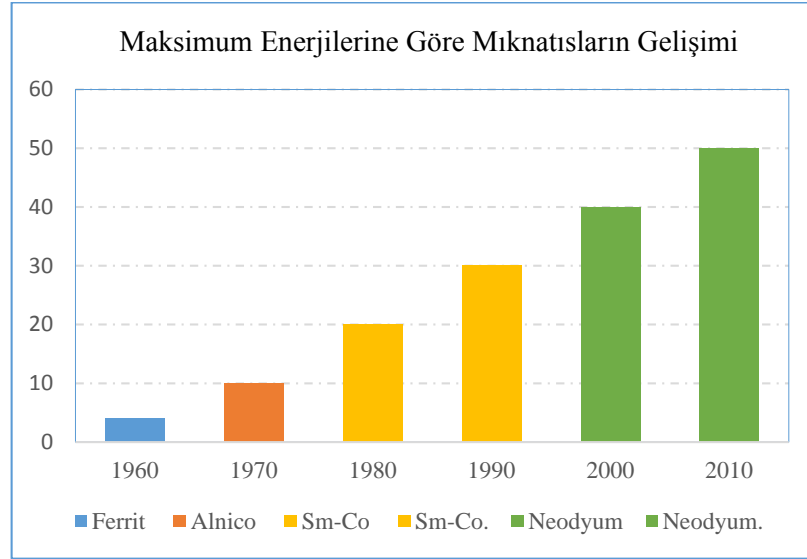
Şekil 27. Bir faza ait oluk sayısı, bir olan; kısa adımlı çift tabakalı sargı düzeni

Yukarıda verilen sargı düzenlerinin tamamında bir kutbun altında bir faza ait oluk sayısı birdir. Fakat daha büyük ve güçlü makineler oluşturulmak isteniyorsa bu sayı artırılabilir. Burada verilen sargı modellerinde, motordan yüksek güç elde etmek için Şekil 26, yüksek hız elde etmek için Şekil 27 tercih edilmektedir [77].

2.5. Motorda Kullanılan Sürekli Mıknatıslar

Sürekli mıknatıslar, üzerinde herhangi bir uyarım olmadan bir manyetik alan üretebilen malzemelere denir. Manyetik malzemeler yapısal olarak yumuşak ve sert malzemeler olmak üzere iki çeşittirler. Yumuşak olanlar, üzerinde bir uyarım manyetik kuvveti olduğunda mıknatıs özelliği gösteren, bu kuvvet ortadan kalktığında ise mıknatıs özelliğini hemen hemen kaybeden malzemelerdir. Sert malzemeler ise üzerindeki uyarım manyetik kuvveti kalkınca da mıknatıs özelliğini devam ettirebilen malzemelerdir. Sürekli mıknatıslar genellikle demir, nikel ve kobalt gibi elementlerin alaşımlarından meydana gelir. Son yıllarda malzeme alanındaki gelişmelerle birlikte sürekli mıknatıslar tercih nedeni olmuştur. Sürekli mıknatısların yıllara göre gelişimi ve kullanımı Tablo 1’de verilmiştir [2], [30], [39], [78].

Tablo 1. Sürekli mıknatısların yıllara göre değişimi



1930’lu yıllarda Al-Ni-Co alaşımlı sürekli mıknatısların bulunması, sürekli mıknatısla uyarma işlevi yapılabilen büyük makine tasarımı gerçekleştirilmiştir. 1950’li yıllarda Baryum, Stronsiyum veya kurşunun demir oksitle oluşturduğu Ferrit mıknatısların bulunması, bu mıknatısların da elektrik makinalarında kullanımını sağlamıştır [78].

1980’li yıllarda nadir toprak elementi mıknatısları Samaryum Cobalt (Sm-Co) ve Neodyum (Nd-Fe-B) mıknatısların geliştirilmesinden sonra elektrik makinalarında, mıknatısların “uyarma alanı sağlamak amacıyla” kullanılma fikri yaygınlık kazanmıştır [79]. Sürekli mıknatısların elektrik motorlarında kullanılmaya başlanması, mıknatıslar ile meydana gelen akının kayıp olmaksızın elde edilebilmesine olanak sağlamıştır. Özellikle, çok kutuplu olarak düşünülen senkron motor tasarımları, sargılı tiplere göre daha küçük boyutlarda yapılmış ve yüksek verimli olarak tasarlanmışlardır [80].

Motorun moment kapasitesinin yüksek olması isteniyorsa, sürekli mıknatısın ürettiği enerjinin yüksek olması gerekir. Aynı zamanda bu motorun boyutu da aynı güçteki başka bir motora göre, daha küçük olur [81]. Bu da ağırlık ve atalet momentinin azalması üstünlüğünü beraberinde getirmektedir. Motorlarda mıknatısların kullanılması aynı zamanda motor tasarımında üreticilere esneklik sağlamaktadır [78].

2.5.1. Sürekli Mıknatısların Özellikleri

Tablo 2’de günümüzde kullanılan sürekli mıknatıs malzemelere ilişkin değerler verilmiştir. Tabloda verilen değerler; kimyasal formül, üretim şekli, uygulanan ısı işlemler gibi etkenlere bağlı olarak değişim gösterebilir [1], [2], [30], [78].

Tablo 2. Mıknatıslara ait özellikler

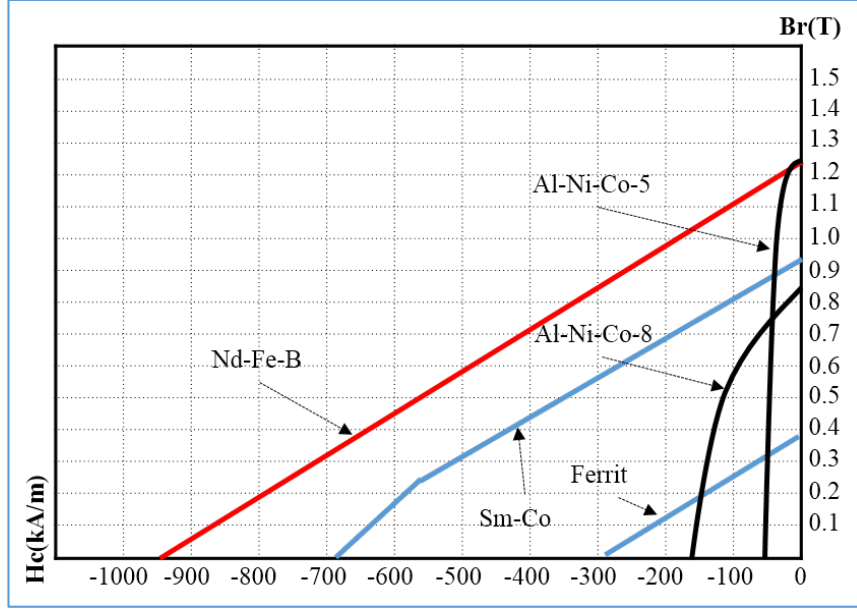
Mıknatıs Cinci	B_r (T)	H_cB (kA/m)	BH_{max} (kJ/m ³)
Al-Ni-Co	1.2	50	44
Ferit	0.4	250	30
Sa-Co	0.9	650	150
Nd-Fe-B	1.2	900	250

Tabloda verilen parametreler [2], [78], [81]:

B_r : Sürekli mıknatısın Akı Yoğunluğu ya da Artık Mıknatısiyet olarak bilinir. DHSMSM’nin torkunu ve performansını etkileyen en önemli parametredir. Teknolojik gelişmelere bağlı olarak akı yoğunluğu değeri 1.45 W/m² (T: Tesla)’ye ulaşmıştır. Ancak sıcaklığa bağlı olarak akı yoğunluğu azalmaktadır.

BH_{max} : Maksimum enerji çarpımı değeridir. Bu anlamda malzemenin ürettiği enerji ne kadar yüksek olursa, sürekli mıknatıs boyutları ve motor boyutları o oranda küçük olur. Akı yoğunluğu (B) ve Alan şiddeti (H)’nin çarpılması ile elde edilir.

H_cB : Zorlayıcı kuvvet olarak bilinir. Sürekli mıknatısın mıknatısiyet özelliğinin kaybetmesi için zıt yönde uygulanan mıknatısiyettir. İyi bir mıknatıstan beklenen, zorlayıcı mıknatısiyet değerinin yüksek olması ve dış etkilerden kolayca etkilenmemesidir. Şekil 28’de sürekli mıknatısların demanyetizasyon eğrileri görülmektedir.



Şekil 28. Sürekli mıknatısların demanyetizasyon eğrileri [78]

Burada verilen bilgilere göre, özellik olarak diğer türlerinden üstün olan ve son yıllarda kullanımı artan Neodyum mıknatıslar tercih edilmiştir. Neodyum mıknatıslar yüksek enerji üretimi, yüksek akı yoğunluğu ve yüksek performans gerektiren servo uygulamalarda kullanılmasını sağlamıştır. Maliyetinin düşük ancak korozyona karşı dayanıklılığı azdır. Bu nedenle bazı özel uygulamalarda mıknatısın yüzeyi epoxy reçine gibi manyetik olmayan madde ile kaplanır [2], [60], [78].

2.5.2. Sürekli Mıknatısların Yerleşimleri

DHSMMSM, İkincil Yan kısmında kullanılan mıknatısın özelliği ve mıknatısların yerleştirilme şekli motorun performansını pozitif yönde etkilediğinden, motor performansını artırmak için farklı tasarım şekilleri uygulanmaktadır. Bunlar içinde en çok rastlanan; mıknatısların İkincil Yan yüzeyine ve mıknatısların İkincil Yan içine (gömülü mıknatıslar) yerleştirilmesidir [1].

Şekil 29'da sürekli mıknatıs yerleşimleri gösterilmektedir.

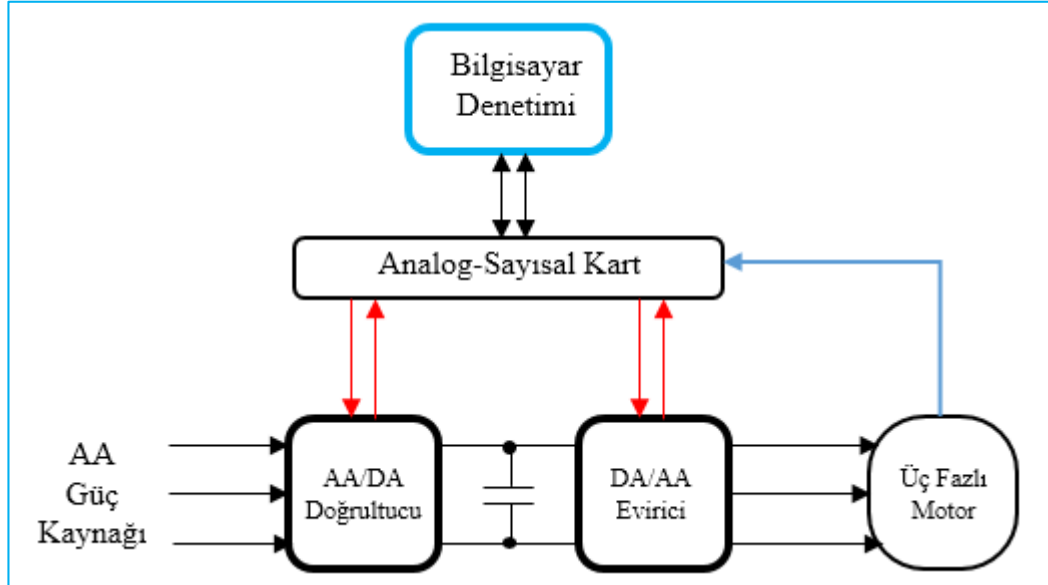


Şekil 29. İkincil yan sürekli mıknatıs yerleşimi

Mıknatısların yüzeyde bulunan modeli, basit yapılı olması nedeniyle en çok kullanılan türüdür. İkinciyen tasarımlarından biri olan bu şeklin en büyük sakıncası, mıknatısların oluşan çekim kuvvetlerine karşı mukavemetlerinin düşük olmasıdır. Bu nedenle; mıknatısların yüzeysel yerleşimi düşük hızlı uygulamalar, gömülü yerleşim ise yüksek hızlı uygulamalarda tercih edilmektedir [78], [82].

2.6. Motor Denetim Yöntemleri

DHSMMSM'ler yapı olarak değilse bile çalışma prensibi olarak döner hareketli motorlarla birbirine benzemektedir. Bu nedenle üç fazlı motorlar için kullanılan denetim yöntemleri DHM'ler içinde kullanılabilir [83]. AA motorlarının denetim yöntemleri genel olarak Skaler, Vektörel, PID ve Akıllı denetim yöntemleri olarak sınıflandırılabilir. Şekil 30'da üç fazlı motorlar için kullanılan genel blok devre şeması verilmiştir [22].

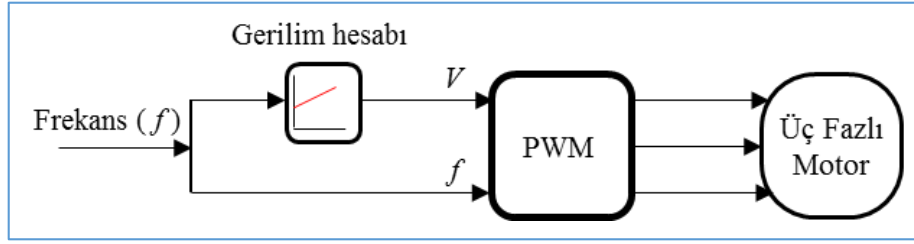


Şekil 30. Motor denetim devresinin genel blok yapısı

Burada adı geçen denetim yöntemleri Evirici devreler ve Denetleyici devreler ile beraber kullanılmaktadır. Bu eviriciler; PWM eviriciler, uzay vektör PWM eviriciler vb. 'dir [22].

2.6.1. Skaler (V/f) Denetim Yöntemi

Bu yöntem, çıkış frekansı ve çıkış gerilimi arasında bulunan oranı sabit tutar. Bu yöntem motor sargılarından geçen akımı da sabit tutar. Bu yöntem kullanılarak frekans; sıfırdan başlayarak, sabit momentle, aşırı güç harcamadan artırılarak motor hareket ettirilmektedir. Skaler denetim yöntemi, diğer denetim türlerinden farklı olarak, karmaşık işlemler gerektirmez [1], [22]. Pozisyon sensörü kullanmadan açık-çevrim denetim yapılabilmesi, düşük maliyet, geri beslemeye ihtiyaç duymaması en büyük üstünlüklerinden sayılabilir. Aynı zamanda motorun yük tarafından zorlanacağı uygulamalar için uygundur [78], [84]. Ancak motor konumunun tespit edilememesi, moment denetiminin olmaması, alan yönlendirilmesinin yapılmaması sakıncaları arasında sayılmaktadır [85]. Şekil 31’de V/f (gerilim/frekans) denetimin blok şeması verilmiştir [22], [85].



Şekil 31. Skaler denetim blok şeması

Bu yöntem, bu tür makinaların denetimi için basit bir yoldur. Yarı iletken teknolojisindeki gelişmelere paralel olarak V/f denetim metodu ile motoru denetlemek son yıllarda yaygınlaşmıştır [86], [87]. Bu çalışmada da V/f denetim yöntemi sürücü sistemi olarak kullanılmıştır.

2.6.2. Vektörel Denetim Yöntemi

Vektör denetiminin temel prensibi; moment ve akı üretimini üstlenen akımın bileşenlerine ayrılmasına ve bunların bağımsız olarak denetim edilmesine dayanır [78]. DHSMSM’lerde İkincil Yan’da uyarım mıknatıslarla sağlandığından, Birincil Yan akımları bileşenlerinden İkincil Yan manyetik alanını meydana getiren bileşen i_d ’ye ihtiyaç duyulmamaktadır. Bunun yerine Birincil Yan akımının tork üreten bileşenine i_q ’ya ihtiyaç vardır. Motorun denetimi i_q ’ya bağlı olarak gerçekleştirilir [22].

Vektör denetim uygulamada doğrudan ve dolaylı olmak üzere iki farklı şekilde yapılmaktadır. Bu iki denetim yapılan çalışmalarda genellikle farklı yöntemler olarak ele alınır. Ancak bazı çalışmalarda bu iki yöntemin teorik olarak aralarında bir fark olmadığı ve iki yöntem arasındaki temel farkın seçilen motorun akı kestiriminden kaynaklandığı belirtilmektedir [20].

Günümüzde başarıyı yüksek vektör denetimli motor sürücülerini birçok endüstriyel alanda kullanılmaktadır. Buna rağmen, düşük hızların gerekli olduğu yerlerde henüz istenilen başarıyı gösterememektedirler. Bu nedenle yüksek başarımlı motor sürücüler konusunda çalışmalar halen devam etmektedir [88].

2.6.3. P+I+D Denetim Yöntemi

Son yıllarda elektrik motorlarının konum denetiminde sıklıkla kullanılan denetleyici yöntemidir. P+I+D denetleyicilerin yapısı basit olduğundan kullanımı kolaydır. Ancak P+I+D denetimin parametre hassasiyeti oldukça zayıftır. Eğer sistem denetleyici katsayıları iyi ayarlanırsa sistem iyi bir şekilde denetlenebilir. Fakat sistem parametrelerindeki herhangi bir değişim, denetleyicinin eski performansını gösterememesine ve sistemin hassas bir şekilde çalışmamasına neden olabilir. Bu durumda P+I+D kontrol katsayılarının yeniden ayarlanması gerekir [78], [89].

Bilinen denetim yöntemlerinde matematiksel modellerin doğru bir şekilde elde edilmesi oldukça önemlidir. Geleneksel denetleyiciler sistemde modellenemeyen parametreler bulunması durumunda istenilen performansı gösteremedikleri açıktır. Bu anlamda standart denetim yöntemleri, transfer fonksiyonlar ya da durum uzay denklemleri ve sabit parametrelere sahiptir. Ancak elektrik motorlarının parametreleri değişkendir. Yani sargı dirençleri sıcaklığa, sargı indüktansları manyetik doyuma bağlıdır. Bu durumun denetim sisteminin dinamik davranışının bozulmasına ve kararsızlığa yol açtığı görülmektedir [30]. Bu nedenle, matematiksel model gerektirmeyen akıllı denetim sistemlerinin yaygın olarak kullanılmaya başlandığı görülmektedir [34]- [39].

2.6.4. Akıllı Denetim Yöntemleri

Üç fazlı motorların benzetiminde kullanılan yöntemler, genellikle bilinen matematiksel denklemlerle ifade edilirler. Bu denklemler içerisinde birçok parametre yer alır. Sistem çalıştırıldığında makine parametrelerinin de bu modellerde sabit kaldığı kabul edilmektedir. Hâlbuki motorların pek çok parametresi sistemin çalışma koşulları ve oluşabilecek dış etkilere göre değişmektedir. Üç fazlı motorların modelleri oluşturulurken parametrelerin sistem koşullarına göre değişim göstermesi istenir. Tüm bu parametreler matematiksel modellerde kullanıldığında, motor modelleri karmaşıklaşır ve kullanışlı olmazlar. Bu nedenle akıllı denetim yöntemleri geliştirilmiştir. Teknolojik gelişmelere bağlı olarak akıllı denetim sistemleri, günlük hayatın hemen hemen her alanında kullanılmaya başlanmıştır. Akıllı denetim yöntemleri; Bulanık Mantık (BM), Yapay Sinir Ağları (YSA), Genel Algoritmalar gibi sıralanabilir [90], [91].

Bulanık Mantık; bir uzman bilgisinden yararlanarak kullandığı ‘if’, ‘then’ bloklarıyla sonuca ulaşabilme yeteneğine sahiptir. Bu nedenle doğrusal olmayan sistemler ile belirsiz parametrelere sahip sistemlerin denetiminde kullanıldığı görülmektedir. Bulanık mantık denetleyicinin, fonksiyonlarının şekli, fonksiyon sayısı ve kural tablosunun doğru oluşturulması, sistemin denetim başarısını önemli ölçüde etkilemektedir.

Yapay Sinir Ağları; Öğrenme ve genelleme yeteneğine sahip olan YSA’lar, doğrusal olmayan yapılarından dolayı, doğrusal olmayan sistemleri, belirli bir eğitim sürecinden sonra denetim yapabilecek hale gelirler. DHSMSM denetiminde kullanılan YSA’lar bilinen denetleyicilerin aksine, kısa sürede cevap vermesi, basit basamaklardan oluşması, basit ve anlaşılır olması, değişen sistem koşullarına daha çabuk uyum sağlaması gibi birçok üstünlüğünden dolayı tercih edilmektedir. Sonraki bölümde YSA’nın sistem denetimi üzerinde durulacaktır.

3. YAPAY SİNİR AĞLARI İLE DENETİM SİSTEMLERİ

İnsan, doğuştan belli bir zekâ seviyesine sahiptir. Zekâ, deneyimle, eğitimle, becerilerle, geliştirilebilir. İlk kez karşılaşılan veya ani gelişen olaylar karşısında, düşünme, anlama, kavrama, analiz etme yeteneği zekâ ile gerçekleştirilmektedir. Zekâ bilgisayar yazılım programlarıyla taklit edilebilmektedir. Zekânın taklit edilebildiği bu duruma ‘Yapay Zekâ’ denir [92]. İnsan davranışlarını taklit etmek amacıyla geliştirilen Yapay zekânın bir alt dalı olan Yapay Sinir Ağları (YSA) özellikle son yıllarda teknolojik gelişime paralel olarak araştırma ve uygulama alanı bulmaktadır [20], [93].

YSA günümüzde; elektrik, bilgisayar, matematik vb. gibi birçok farklı bilim dalında kullanılır hale gelmiştir. Mühendislik alanında kullanılmasının en önemli nedeni, klasik tekniklerle çözümü zor olan problemleri çok hızlı bir şekilde tanımlaması ve çözüm yolu bulmasıdır [20].

3.1. Yapay Sinir Ağları

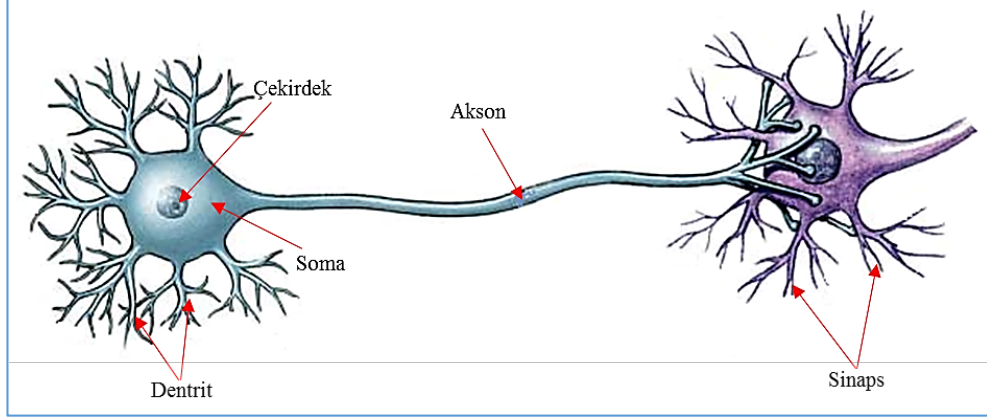
YSA, insanlar tarafından örnekleri gerçekleştirilen olayları kullanarak öğrenebilen, oluşabilecek ani olaylara karşı nasıl tepkiler üretilebileceğini belirleyen bilgisayar programlarıdır. YSA Öğrenme, İlişkilendirme, Sınıflandırma, Genelleme, Özellik belirleme gibi konularda örneklerden elde ettikleri sonuçlarla, benzer konulara benzer cevaplar verirler [93].

Teknik olarak bakıldığında, bir YSA'nın en temel görevi, yapması istenilen olayla ilgili örnekleri kendisine verilen veya öğretilen bilgilere karşılık gelebilecek bir çıktı seti belirlemektir [94], [95].

YSA günümüzde birçok probleme çözüm üretebilecek durumdadır. YSA'lar birbirine düzenli olarak bağlı ve paralel olarak çalışabilen yapay hücrelerden oluşmaktadır. Bu hücrelerin birbirine bağlantılı olduğu ve bu bağlantıların her birinin bir değeri olduğu kabul edilir. Aynı zamanda bu hücrelerin bir birleri ile bağlanmaları sonucu oluşan ağa, ‘Yapay Sinir Ağı’ denmektedir. Bu ağın oluşturulması için biyolojik sinir sisteminin bilinmesi gerekmektedir [34], [42], [93], [96].

3.2. Biyolojik Beyin Hücresi ve Yapay Sinir Ağlarının Yapısı

YSA, insan beyininde bulunan biyolojik sinir ağlarını taklit ederek geliştirilmiş bilgisayar programlarıdır. Şekil 32’de bir biyolojik sinir hücresi ve bileşenleri gösterilmiştir [20], [97].



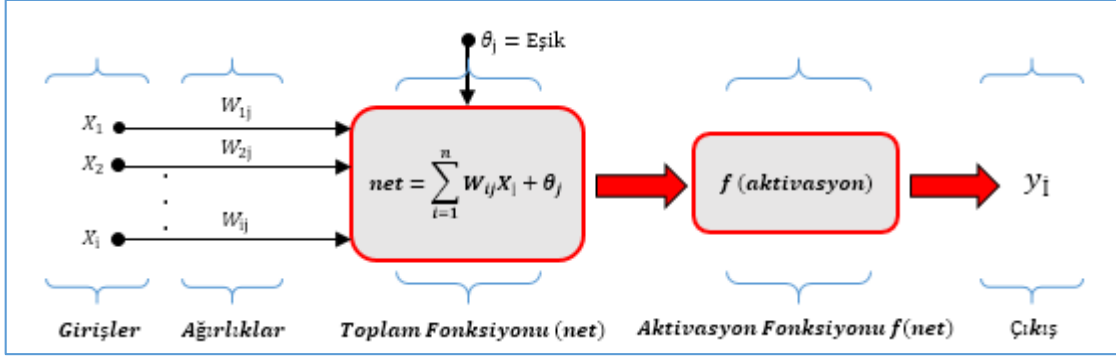
Şekil 32. Biyolojik sinir hücresi ve bileşenleri

Şekil 32’ye bakıldığında, Dendritler, gelen sinyalleri çekirdeğe iletir. Çekirdek dendritten gelen sinyalleri bir araya toplar ve aksona iletir. Toplanan bu sinyaller, akson tarafından işlenerek sinaplara gönderilir. Sinapslar da yeni üretilen sinyalleri diğer sinir hücrelerine iletir. Biyolojik sinir hücresinde bulunan bileşenlerin YSA’da ne anlama geldiği Tablo 3’de verilmiştir.

Tablo 3. Biyolojik sinir hücresinin YSA'daki karşılığı

Biyolojik Sinir Hücresi	Yapay Sinir Hücresi
Akson	Çıktı
Dentrit	Toplama Fonksiyonu
Çekirdek	Aktivasyon Fonksiyonu
Sinaps	Ağırlıklar

YSA’nın temel birimi, yapay bir sinirdir. Bu aynı zamanda işlem elemanı ya da düğüm olarak adlandırılır. Bir yapay sinir, bir biyolojik sinire göre yapısı daha basittir ve biyolojik sinirlerin dört temel görevini taklit ederek işlem yaparlar. Şekil 33’de yapay bir sinir hücresi gösterilmektedir [96].



Şekil 33. Yapay sinir hücresinin yapısı (Perseptron model)

YSA'nın tüm çeşitleri Şekil 33'den hareketle türetilmiştir. Yapısal farklılıklar YSA'ların farklı sınıflandırılmasını sağlar. YSA'nın öğrenme yeteneği, seçilen öğrenme yöntemine ve ağırlıklara bağlıdır [98].

3.2.1. Girişler

X ile belirtilen girişler, dışarıdan alınan bilgiler ya da diğer bir hücreden gelen bilgilerdir. Bir sinir genellikle dışarıdan birçok giriş alabilir [42], [95].

3.2.2. Ağırlıklar

W ile belirtilen ağırlıklar, hücreler arasındaki bağlantıların sayısal değerlerini ifade eder. Bir hücreye gelen bilginin değerini ve hücre üzerindeki etkisini gösterir. Yani Şekil 32'deki W_1 ağırlığı X_1 girişinin hücre üzerindeki etkisini gösterir [92].

3.2.3. Toplam Fonksiyonu

net olarak belirtilen fonksiyon, hücreye gelen girdileri ağırlıklarla çarpıp-toplayarak o hücrenin net girdisinin hesaplanmasını sağlar. Farklı fonksiyon çeşitleri bulunmaktadır. En yaygın olan ağırlıklı toplamı bulan fonksiyondur. net genel formülü [92];

$$net = \sum_{i=1}^n I_i W_i + \theta_j \quad (21)$$

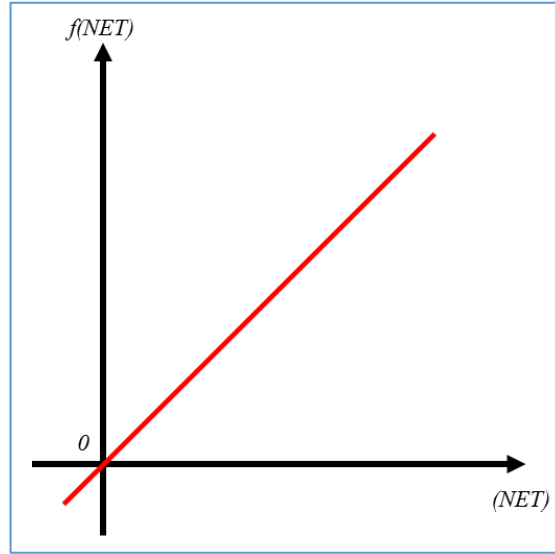
Burada, I girişleri (X_1, X_2, \dots, X_i), W ağırlıkları ($W_{1j}, W_{2j}, \dots, W_{ij}$), θ eşik değeri (Toplama fonksiyonu takılmasın diye, sıfırdan farklı küçük bir değer kullanılır), n bir hücreye gelen toplam girdi sayısıdır.

3.2.4. Aktivasyon Fonksiyonu

$f(net)$ olarak belirtilen fonksiyon, hücreye gelen net girdiyi işleyerek hücrenin bu girdiye karşılık üreteceği çıktının belirlenmesini sağlar. Çıktıyı hesaplamak için farklı aktivasyon fonksiyonları kullanılır.

Doğrusal Aktivasyon Fonksiyonu: Doğrusal bir problemi çözmek amacıyla kullanılır. Bu fonksiyon hücrenin net girdisini doğrudan hücre çıkışına verir. Doğrusal aktivasyon fonksiyonu Denklem 22'de, grafiği Şekil 34'deki gibidir [99].

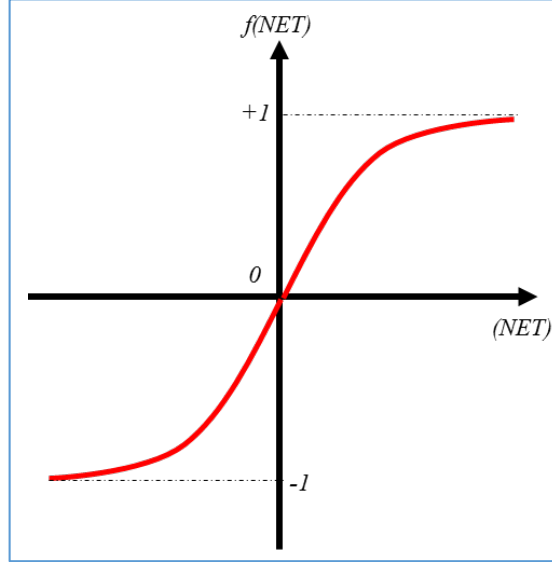
$$f(net) = net \quad (22)$$



Şekil 34. Doğrusal aktivasyon fonksiyonu grafiği

Tanjant Hiperbolik Aktivasyon Fonksiyonu: Doğrusal olmayan bir problemi çözmek amacıyla kullanılır. Bu fonksiyon hücrenin net girdisini tanh aktivasyon fonksiyonundan geçirerek hesaplanır. Tanjant hiperbolik aktivasyon fonksiyonu Denklem 23'de, grafiği Şekil 35'deki gibidir [99].

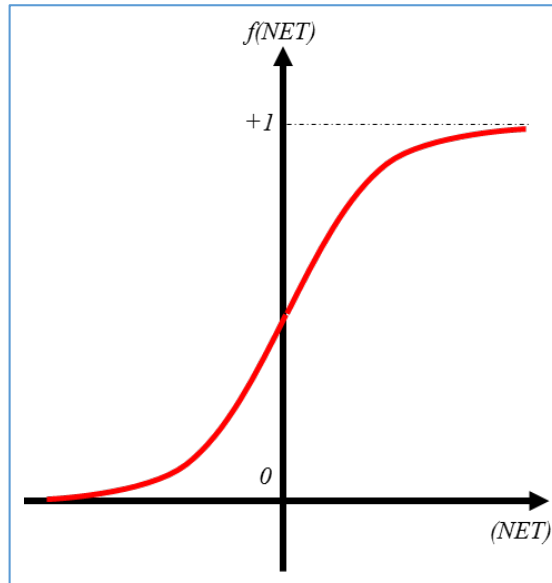
$$f(NE T) = \tanh(NE T) = \frac{e^{NE T} + e^{-NE T}}{e^{NE T} - e^{-NE T}} \quad (23)$$



Şekil 35. Tanjant hiperbolik aktivasyon fonksiyonu grafiği

Sigmoid Aktivasyon Fonksiyonu: Doğrusal olmayan bir problemi çözmek amacıyla kullanılır ve günümüzde en yaygın olarak kullanılan aktivasyon fonksiyonudur. Sigmoid aktivasyon fonksiyonu Denklem 24’de, grafiği Şekil 36’daki gibidir [99].

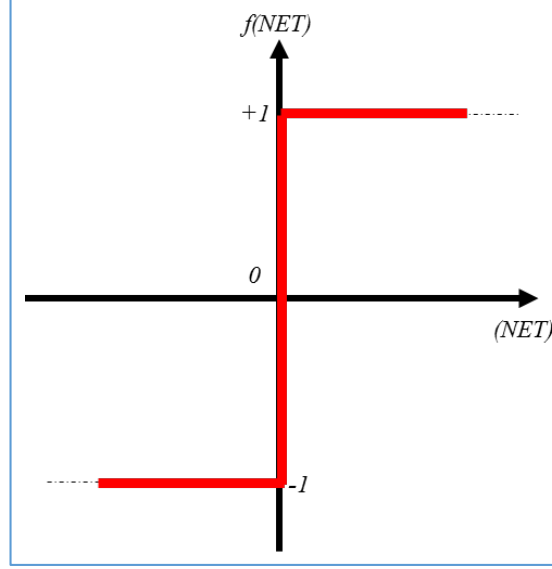
$$f(NE T) = \frac{1}{1 + e^{-NE T}} \quad (24)$$



Şekil 36. Sigmoid aktivasyon fonksiyonu grafiği

Signum Aktivasyon Fonksiyonu: Doğrusal olmayan bir problemi çözmek amacıyla kullanılır ve günümüzde en yaygın olarak kullanılan aktivasyon fonksiyonudur. Sigmoid aktivasyon fonksiyonu Denklem 25’de, grafiği Şekil 37’deki gibidir [99].

$$f(NE T) = \text{sgn}(NE T) = \begin{cases} +1, NE T > 0 \\ -1, NE T < 0 \end{cases} \quad (25)$$



Şekil 37. Signum aktivasyon fonksiyonu grafiği

3.2.5. Ölçekleme ve Sınırlama

Yapay hücrelerde, aktivasyon fonksiyonundan elde edilen sonuçlar bir ölçek veya sınır işlemlerinden geçebilir. Örneğin sigmoid fonksiyonu 0 ve +1 arasında sonuçlar verirken, fonksiyon sonucunda 1 çıkardığımızda +1 ve -1 arasında yeni bir ölçeklendirme yapılmış olur. Sınırlandırma ise, ölçeklenmiş sonuçların en küçük ve en büyük sınırlarını aşmamasını sağlamaktadır [92].

3.2.6. Hücrenin Çıkışı

Aktivasyon fonksiyonu tarafından belirlenen ve y_i ile gösterilen değerdir. Üretilen değer dışarıya veya başka bir hücreye gönderilebilir. Aynı zamanda hücre kendi çıkışını kendisine giriş olarak da gönderebilir. Bu durum YSA'nın yapısına bağlı olarak değişmektedir [92], [94].

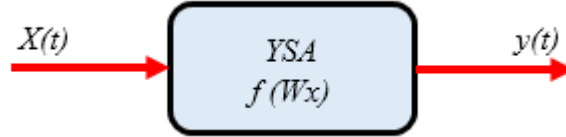
3.3. Yapay Sinir Ağlarının Sınıflandırılması

YSA modelleri, bir ağına girişte ve çıkışta öğrenmesi gereken bilgilere ve sinirler arası ilişkilerle ifade edilen mimarilerine göre farklılıklar göstermektedirler. Farklı öğrenme türleri YSA modellerini belirlemektedir. YSA'lar yapılarına ve öğrenme algoritmalarına göre sınıflandırılırlar [19].

3.3.1. Yapılarına Göre Sınıflandırma

YSA'lar yapılarına göre [19], ileri beslemeli (feedforward) ve geri beslemeli (feedback) ağlar olarak sınıflandırılırlar.

İleri Beslemeli Ağlar: İleri beslemeli ağlarda işlem elemanları çoğunlukla katmanlara ayrılmışlardır. Alınan işaretler, giriş katmanından çıkış katmanına doğru tek yönlü bağlantılarla iletilir. İşlem elemanları aynı katmanda bağlantı kuramaz ancak farklı katmanlar arasında kurabilir. Şekil 38'de ileri beslemeli ağ için blok diyagram gösterilmiştir. İleri beslemeli ağlara örnek olarak çok katmanlı perseptron (Multi Layer Perceptron-MLP) ağları verilebilir [19].



Şekil 38. İleri beslemeli ağın blok diyagramı [93]

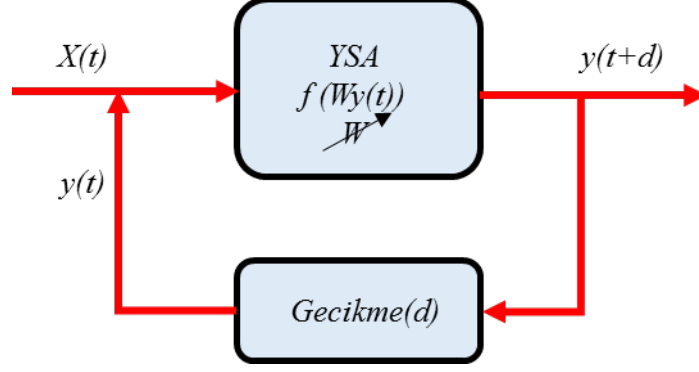
Burada ağın giriş vektörü;

$$X = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_i \end{bmatrix} \quad (26)$$

Ağırlık matrisi;

$$W = \begin{bmatrix} W_{11} & W_{12} & \cdots & W_{1j} \\ W_{21} & W_{22} & \cdots & W_{2j} \\ \vdots & \vdots & \cdots & \vdots \\ W_{i1} & W_{i2} & \cdots & W_{ij} \end{bmatrix} \quad (27)$$

Geri Beslemeli Ağlar: Bu modelde iki şekilde geri besleme yapılabilir. Birincisi çıkış ve ara katmandaki çıkışların girişe, ikincisi çıkış ve ara katmandaki çıkışların önceki ara katmanlara verildiği bir ağ yapısıdır. Şekil 39’da geri beslemeli ağ için blok diyagram gösterilmiştir. Bu ağlara örnek olarak Hopfield, Elman ve Jordan ağları verilebilir [19].



Şekil 39. Geri beslemeli ağın blok diyagramı [93]

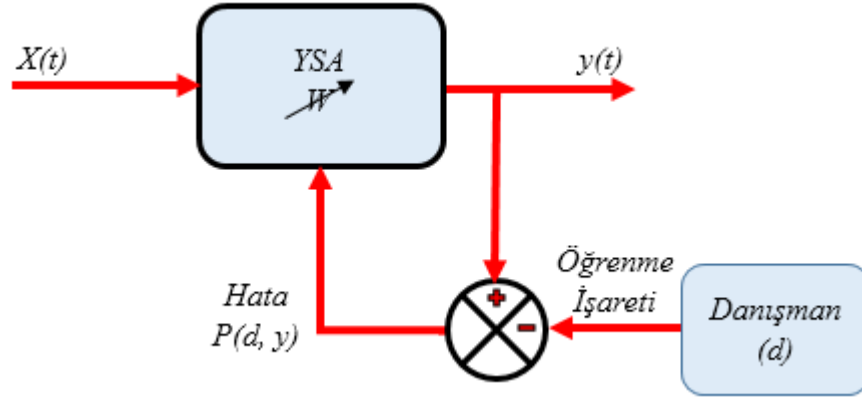
3.3.2. Öğrenme Yaklaşımlarına Göre Sınıflandırma

YSA’larda işlem elemanlarının bağlantılarının W değerlerinin belirlenmesi işlemine “ağın eğitilmesi” denir. Başlangıçta bu değerleri rastgele belirlenir. YSA kendisine örnekler gösterildikçe bu ağırlık değerlerini değiştirir. Burada amaç ağa gösterilen örnekler için doğru çıktıları sonuçlarını üretecek ağırlık değerlerini bulmaktır. Bu nedenle örnekler ağa defalarca gösterilerek en doğru ağırlık değerleri bulunmaya çalışılır. Ağın doğru ağırlık değerlerini tespit etmesi, örneklerin temsil ettiği olay hakkında genelleme yapabilme yeteneğinin sağlanması demektir. Bu işleme “ağın öğrenmesi” denir. Ağırlık değerlerinin değişmesi belirli bir kurala göre yürütülmektedir. Bu kurallara da “öğrenme kuralları” denir. Kullanılan öğrenme yaklaşımına göre farklı öğrenme kuralları geliştirilmiştir [94].

Ağın eğitimi bittikten sonra performansını (öğrenip öğrenmediğini) ölçmek için yapılan denemelere “ağın test edilmesi” denilmektedir. Sistemi test etmek için ağın öğrenme sırasında görmediği örnekler kullanılır. Bu sırada ağın ağırlık değerleri değiştirilmez. Test edilmek istenen örnekler ağa gösterilir. Ağ eğitim sırasında belirlenen bağlantı ağırlıklarını kullanarak daha önce görmediği bu örnekleri için genelleme yaparak çıktılar üretir. Bu çıktıların doğruluk değerleri ağın öğrenmesi hakkında bilgiler verir. Sonuçlar ne kadar iyi olursa ağın eğitimi o kadar iyi demektir. YSA’nın eğitiminde kullanılan örnek sete “eğitim seti”, test için kullanılan sete ise “test seti” adı verilmektedir. Yapay sinir ağlarının bu şekilde bilinen örneklerden belirli bilgileri çıkartarak bilinmeyen örnekler hakkında genelleme

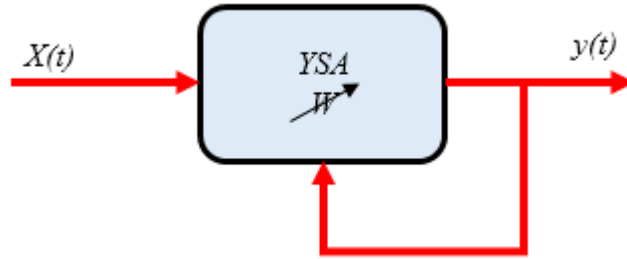
yapabilme yeteneğine “Adaptif öğrenme” denilmektedir [19], [94]. Üç farklı türde öğrenme modeli bulunmaktadır [92].

Denetimli Öğrenme: Bu tür öğrenmede, ağa örnek olarak bir adet doğru çıkış verilir. İstenilen çıkış ve gerçek çıkış arasındaki farka (hataya) göre, işlem elemanları arasındaki bağlantıların ağırlıkları, istenilen çıkışa en uygun çıkışı elde etmek için sonradan düzenlenir. Bu sebeple denetimli öğrenme modeli bir “öğreticiye” veya “danışmana” ihtiyacı duyar [92]. Şekil 40’da denetimli öğrenme blok diyagramının yapısı gösterilmiştir.



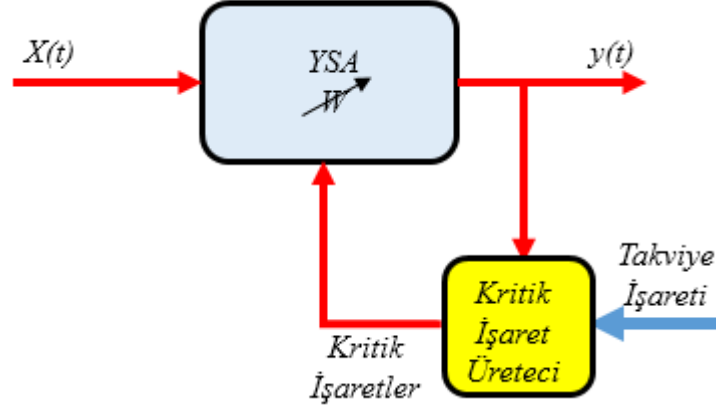
Şekil 40. Denetimli öğrenme modeli blok diyagramı [93]

Denetimsiz Öğrenme: Bu tür öğrenme sisteminde, ağın öğrenmesi için herhangi bir öğretici veya danışman bulunmamaktadır. Dolayısıyla girişe verilen örnekten elde edilen çıkış bilgisine göre ağ sınıflandırma kurallarını kendi geliştirmektedir [92]. Şekil 41’de denetimsiz öğrenme yapısı gösterilmiştir.



Şekil 41. Denetimsiz öğrenme blok diyagramı [93]

Takviyeli Öğrenme: Bu öğrenme modelinde öğrenen sisteme bir öğretici vardır. Ancak öğretici her girdi seti için olması gereken (üretilmesi gereken) çıktı setini sisteme göstermez. Bunun yerine sistemin kendisine gösterilen girdilere karşılık, çıktısını üretmesini bekler ve üretilen çıktının doğru veya yanlış olmasına göre bir sinyal üretir [92]. Şekil 42’de takviyeli öğrenme yapısı gösterilmiştir.



Şekil 42. Takviyeli öğrenme blok diyagramı [93]

3.3.3. Öğrenme Kurallarına Göre Sınıflandırma

Öğrenme sistemlerinde farklı öğrenme kuralları bulunmaktadır. YSA'larda bu öğrenme kuralları genellikle Hebb kuralına dayanmaktadır. Öğrenme algoritmalarının gelişimi ve çeşitleri Hebb, Delta, Kohenen ve Hopfield olmak üzere dört farklı öğrenme kuralına dayanmaktadır [100].

Hebb Kuralı: Öğrenme kuralları içindeki en eski öğrenme kuralıdır. Diğer öğrenme kuralları bu kuraldan hareketle oluşturulmuştur. Bu öğrenme kuralının çalışma ilkesi; iki işlem elemanı aktif iken (matematiksel olarak aynı işareti taşıdıklarında), bir işlem elemanı (hücre) diğer bir işlem elemanından bilgi aldığı anda, her iki hücrenin arasındaki bağlantı kuvvetlendirilmelidir. Örneğin; bir hücre kendisi aktif olduğunda bağlı olduğu diğer hücreyi aktif yapmaya, pasif olduğunda ise pasif yapmaya çalışmaktadır. Hebb öğrenme kuralı öğreticili (danışmanlı) öğrenme kurallarından sayılmaktadır [100].

Hopfield Kuralı: Bu öğrenme kuralında YSA elemanları arasındaki bağların ne kadar zayıflatılması veya ne kadar kuvvetlendirilmesi gerektiği belirlenmektedir. Öğrenme katsayısı yardımı ile ağırlıkların kuvvetlendirilmesi veya zayıflatılması gerçekleştirilmektedir. Öğrenme katsayısı, kullanıcı tarafından belirlenir ve genel olarak 0 – 1 arasında atanan sabit ve pozitif bir değerdir [92], [100].

Delta Kuralı: Bu kuralın en temel özelliği beklenen çıktı ile gerçekleşen çıktı arasındaki farklılığı azaltmaktır. Hebb kuralının geliştirilmiş şeklidir. Matematiksel olarak anlamı, ağırlık ürettiği çıktı ile beklenen çıktı arasındaki hatanın karelerinin ortalamasını en aza indirmektir [19], [92].

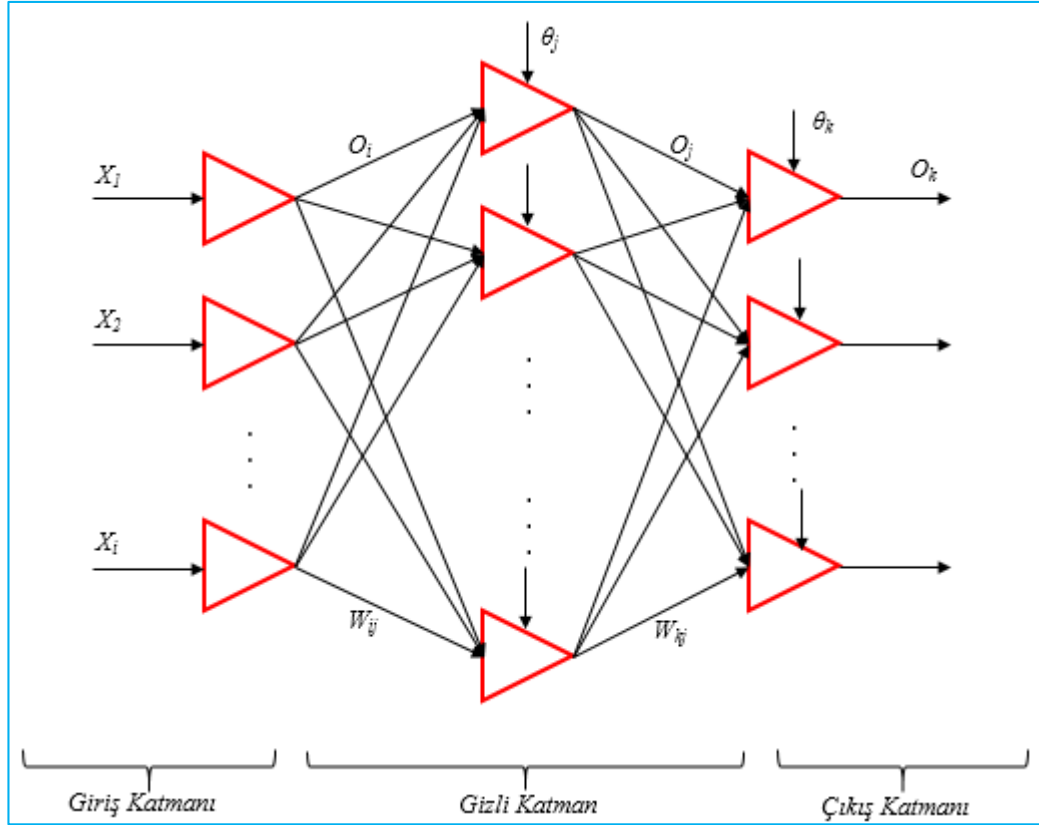
Kohenen Kuralı: Kohenen tarafından geliştirilen bu yöntem biyolojik sistemlerdeki öğrenmeden esinlenmiştir. Bu kurala göre ağırlık elemanları (hücreleri) ağırlıklarını değiştirmek için birbirleri ile yarışır. En büyük çıktıyı üreten hücre kazanan hücre olmakta ve bağlantı ağırlıkları değiştirilmektedir. Bu, o hücrenin yakınındaki hücrelere karşı daha kuvvetli hale gelmesi demektir. Hem kazanan elemanların hem de komşuları sayılan elemanların (hücrelerin) ağırlıklarını değiştirmesine izin verilmektedir [92].

3.4. Öğrenme Yöntemleri

Yapılan araştırmalara bakıldığında pek çok öğrenme algoritmasının kullanıldığı görülmektedir. Bunlar; Geri Yayılım Yöntemi, Esnek Yayılım Yöntemi, Delta Bar Delta Yöntemi, Geliştirilmiş Delta Bar Delta Yöntemi, Hızlı Yayılım Yöntemi, Hızlı Yayılım Yöntemi, Levenberg-Marquardt Yöntemi, Eşleştirmeli Eğitim Yöntemi, Genetik Algoritma vb. Ancak burada tez çalışmasında kullanılan Geri yayılım öğrenme yöntemi açıklanmıştır [92], [100].

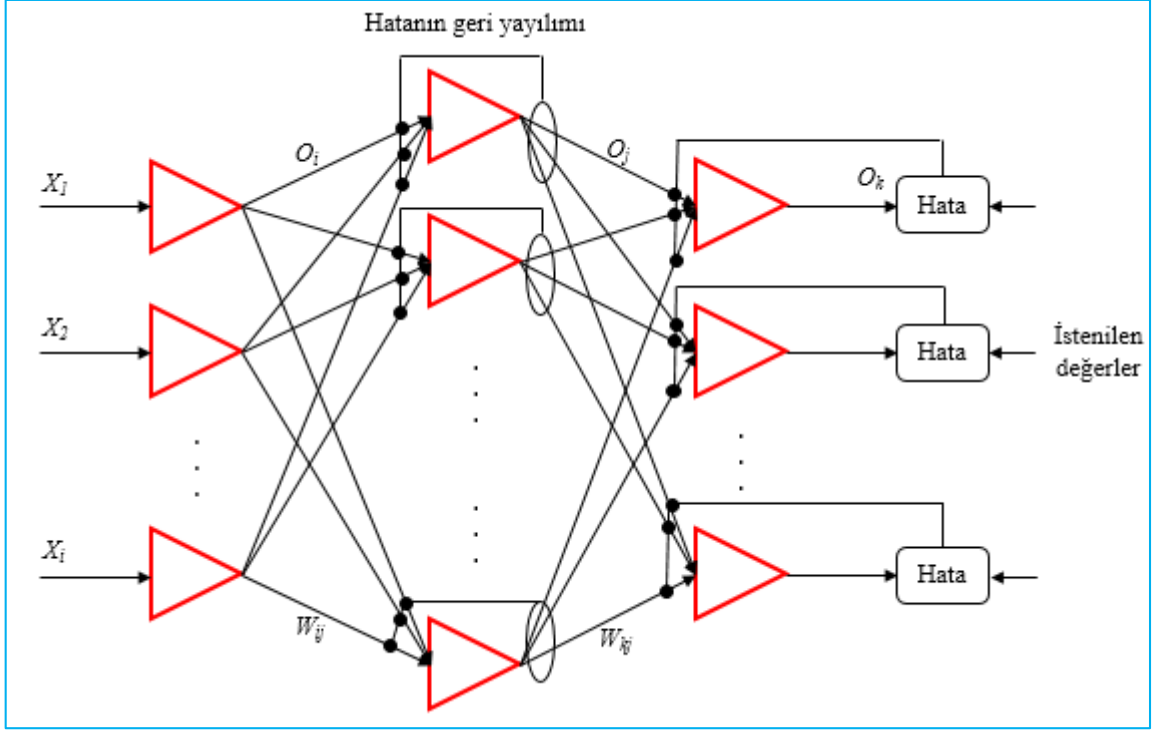
3.4.1. Geri Yayılımlı Öğrenme Yöntemi

Geri Yayılımlı Öğrenme Yöntemi (GYÖY); Giriş, Çıkış ve en az bir adet Gizli katman olmak üzere üç katmandan meydana gelir. Şekil 43'de üç katmanlı YSA grafiği verilmiştir [96].



Şekil 43. Yapay sinir ağı katman grafiği

Gizli katman sayısı arttıkça dört katmanlı, beş katmanlı olarak adlandırılır. Gizli katman ve her gizli katmanda bulunan düğüm sayısı değiştirilebilir. Burada düğüm sayısının artması ağıın hatırlama yeteneğini artırmaktadır. Ancak öğrenme işleminin süresini uzatmaktadır. Giriş ve Gizli katmandaki birbirine bağlıdır. Bir katmana ait düğümler kendi içindeki hiçbir düğüme bağlı değildir. Her katmanın çıkışında elde edilen değerler bir sonraki katman için giriş değerleridir. Bu şekilde YSA'nın giriş değerlerinin ağıın girişinden başlayarak çıkışa doğru ilerlemesine ileri besleme denir. Şekil 44'de bir geri yayılım ağı örneği görülmektedir [92].



Şekil 44. Geri yayılım ağı yapısı

Geri yayılım ağında oluşan hatalar, ileri besleme mekanizması içinde kullanılan aynı bağlantılar aracılığıyla, ileri beslemedeki aktarım fonksiyonu türevi tarafından, geriye doğru yayılmaktadır. Öğrenme işlemi, çift yönlü hafıza birleştirmeye dayanmaktadır [19].

Öğrenme sırasında, giriş örnekleri belli bir sıra ile ağı verilir. Her bir çalışma örneği katmanlar boyunca sıra ile ilerleyerek çıkış örneği hesaplanana kadar ileri yayılır. Çıkışta elde edilen değer beklenen değer ile karşılaştırılıp fark yani hata bulunur. Hatalar, her katmanda geri besleme bağlantılarından ‘girişler’ olarak tekrar kullanılır. Geriye doğru bağlantılar sadece ‘öğrenme’ için kullanılır. İleri doğru olan bağlantılar ise hem öğrenme hem de işlemsel amaçlı kullanılır [92].

Şekil 43’deki üç katmanlı YSA’yı örnek alalım. Bu durumda çıkış katmanı formülü;

$$O_k = f(net_k) \quad (28)$$

$$net_k = \sum_j W_{kj} O_j + \theta_k \quad (29)$$

Gizli katmana ait formül;

$$O_j = f(net_j) \quad (30)$$

$$net_j = \sum_i W_{ji} O_i + \theta_j \quad (31)$$

Aktivasyon fonksiyonu Denklem 24 (Sigmoid) olduğunu kabul edelim. Bu fonksiyonun türevi [96];

$$f'(net) = \frac{1}{2}(1 - net^2) \quad (32)$$

3.4.2. Geri Yayımlı Öğrenme Yönteminin Uygulama Adımları

GYÖY'ü kullanırken aşağıda verilen adımlar takip edilmesi önerilmektedir [96].

1.ADİM: Başlangıçta W_{kj} , W_{ji} , θ_k , θ_j ve öğrenme hızı η ayarlanır.

2.ADİM: Ağ girişlerini üret ve O_j , O_k hesapla.

3.ADİM: Denklem 33 ile δ_k hesapla.

$$\delta_k = O_k (1 - O_k) (t_k - O_k) \quad (33)$$

4.ADİM: Ağırlıkların ve eşik değer değişimini hesaplayın.

$$\Delta W_{kj} = \eta \delta_k O_j \quad (34)$$

$$\Delta \theta_k = \eta \delta_k$$

5.ADİM: Denklem 35 ile δ_j hesapla.

$$\delta_j = O_j (1 - O_j) \sum_k \delta_k W_{kj} \quad (35)$$

6.ADİM: Ağırlıkların ve eşik değer değişimini hesaplayın.

$$\Delta W_{ji} = \eta \delta_j O_i \quad (36)$$

$$\Delta \theta_j = \eta \delta_j$$

7.ADİM: Yeni ağırlıkların ve eşik değer değişimini hesaplayın.

$$W_{ji}(t+1) = W_{ji}(t) + \Delta W_{ji} \quad (37)$$

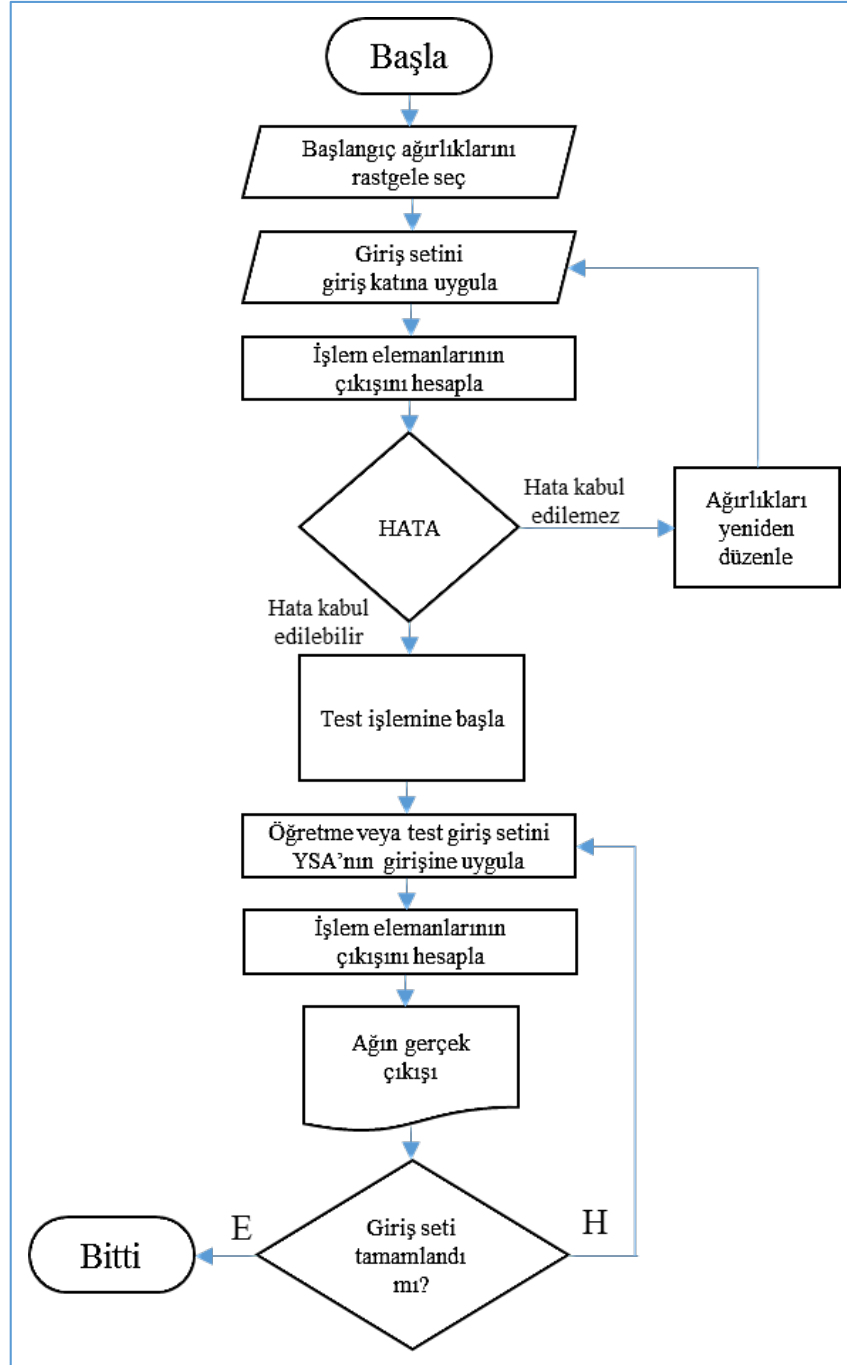
$$\theta_j(t+1) = \theta_j(t) + \Delta \theta_j$$

$$W_{kj}(t+1) = W_{kj}(t) + \Delta W_{kj}$$

$$\theta_k(t+1) = \theta_k(t) + \Delta \theta_k$$

8.ADİM: $t \leftarrow t+1$ ayarlayın ve adım 2 ye gidin.

Bu işlemler, çalışma süresi içinde 'toplam hata' değeri en aza indirilinceye kadar tekrarlanır. Şekil 45'de ileri beslemeli bir ağ için geri yayılım öğrenme yöntemi akış şeması gösterilmektedir [19], [92].



Şekil 45. İleri beslemeli bir ağ için geri yayılım öğrenme yöntemi akış şeması

3.5. Yapay Sinir Ağlarının Üstünlükleri ve Sakıncaları

Aşağıda verilen ifadeler YSA'nın üstünlükleri arasında yer almaktadır [92], [100].

- ✓ Doğrusal olmayan çok boyutlu, gürültülü, eksik bilgili ve özellikle problemin çözümünde kesin bir matematiksel modelin veya algoritmanın bulunmadığı durumlarda başarılıdır.
- ✓ İstisnai ve anormal veri sayısı çok olan konularda iyi sonuçlar verir.
- ✓ Adaptasyon yeteneği vardır.
- ✓ Bilgiler ağın tamamında saklanır.
- ✓ Daha önce görülmemiş örnekler hakkında bilgi üretebilir.
- ✓ Kendi kendini organize etme ve öğrenebilme yetenekleri vardır.
- ✓ Eksik bilgi ile çalışabilmektedirler.
- ✓ Hata toleransına sahiptirler.
- ✓ Belirsiz ve tam olmayan bilgileri işleyebilmektedirler.
- ✓ YSA, ani bozulma göstermezler.
- ✓ Sadece nümerik bilgiler ile çalışabilmektedirler.
- ✓ YSA, normal yollarla çözülmesi zor olan problemleri çözmek için tasarlanmışlardır.

Aşağıda verilen ifadeler YSA'nın sakıncaları arasında yer almaktadır [92], [100].

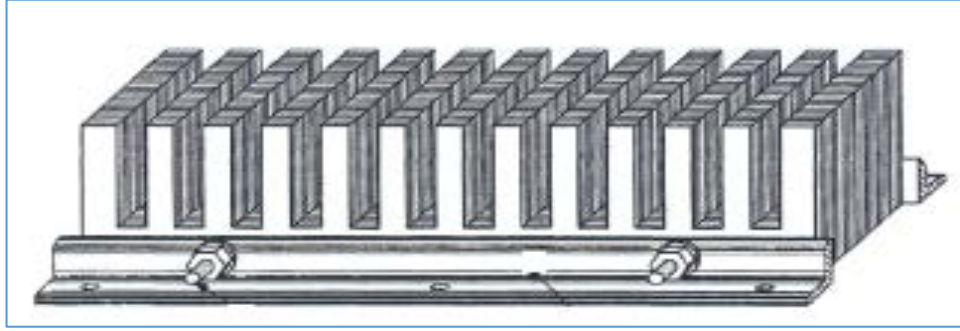
- ✓ 'Kara Kutu', verdiği sonucun açıklamasını yapamaz.
- ✓ Uygun ağ yapısının belirlenmesinde belli bir kural yoktur.
- ✓ Ağın parametre değerlerinin belirlenmesinde belli bir kural yoktur.
- ✓ Eğitim örnekleri seçiminde genel bir kural yoktur.
- ✓ Öğrenilecek problemin ağa gösterimi önemli bir problemdir.
- ✓ Ağın eğitiminin ne zaman bitirilmesi gerektiğine ilişkin belli bir yöntem yoktur.

4. YAPILAN ÇALIŞMALAR VE BULGULAR

Bu bölümde Çift Yanlı Doğrusal Hareketli Sürekli Mıknatıslı Senkron Motor (ÇYDHSMSM) ile ilgili yapılan çalışmalar anlatılmaktadır. Öncelikle yapısal ayrıntılar ardından gerçekleştirilen çalışmalar anlatılacaktır.

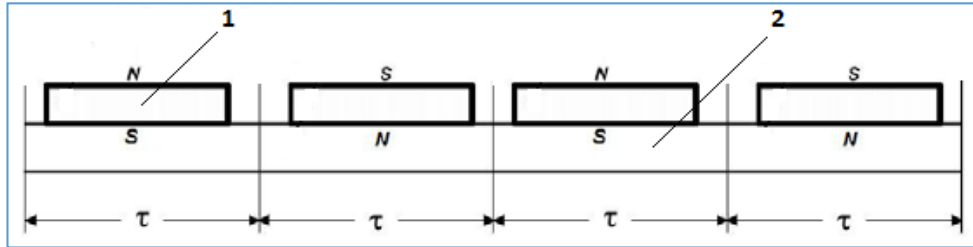
4.1. İncelenen Motorun Yapısal Ayrıntıları

Doğrusal Hareketli Sürekli Mıknatıslı Senkron Motorlarda (DHSMSM) çekirdeği oluşturan saçlar, transformatörlerdeki saçlara benzerler. Birincil yan sargıları, saçların paketlenmesiyle ortaya çıkan oluklara yerleştirilir. Şekil 46'da Birincil yan görüntüsü gösterilmiştir [30].



Şekil 46. Birincil yan görünümü [30]

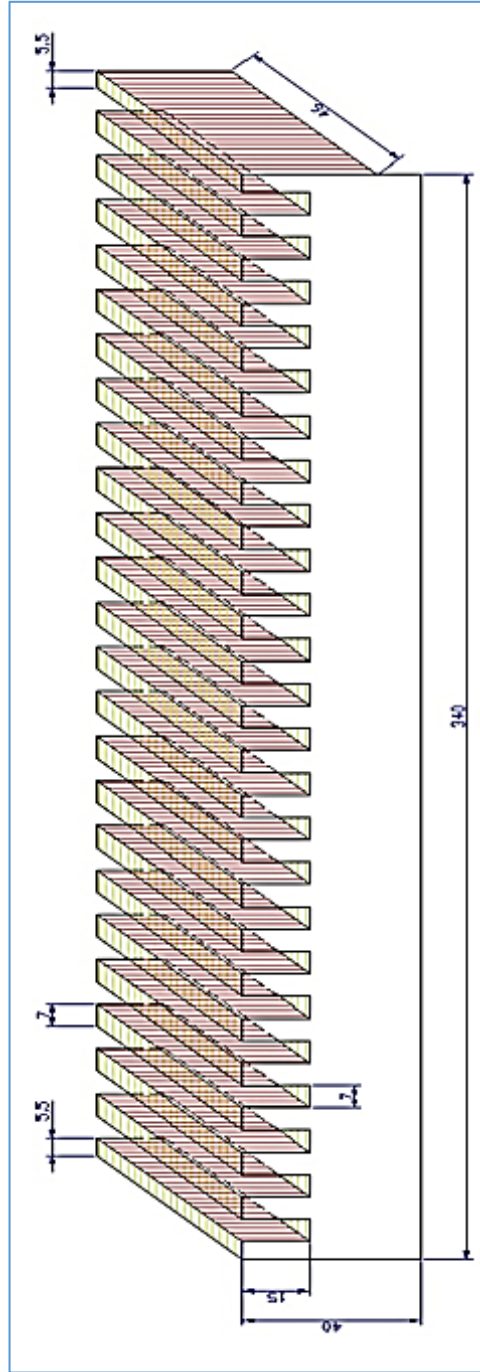
İkincil Yan ise yumuşak demir üzerine yüzeysel olarak yerleştirilmiş sürekli mıknatıslardan meydana gelmiştir. Şekil 47'de bu durum gösterilmiştir [30].



Şekil 47. İkincil yan görünümü, 1-Sürekli mıknatıslar, 2-Yumuşak demir ya da sabit kısım

4.1.1. Birincil Yan Yapım Aşamaları

Birincil yan için 0.35 mm kalınlığında saclar paket haline getirilmiş ve Şekil 46'daki gibi kesim işlemi yapılmıştır. Kesim görünümü Şekil 48'de verilmiştir. Ölçüler mm olarak verilmiştir.



Şekil 48. Birincil yan kesim görünümü

Birincil yan için kullanılan kesim ölçüleri Tablo 4’de verilmiştir.

Tablo 4. Paket halindeki sacların kesim ölçüleri

Birimler	Ölçüler
Sac paket boyu	340 mm
Sac paketin eni	40 mm
Sac paket kalınlığı	46 mm
Oluk derinliği	15 mm
Oluk eni	7 mm
Diş eni	7 mm
Baş kısımlardaki diş eni	5.5 mm

Bu duruma göre sac paketimizden elde edilen Birincil Yan Şekil 49’da gösterilmektedir.



Şekil 49. Birincil yanın yapım aşamaları

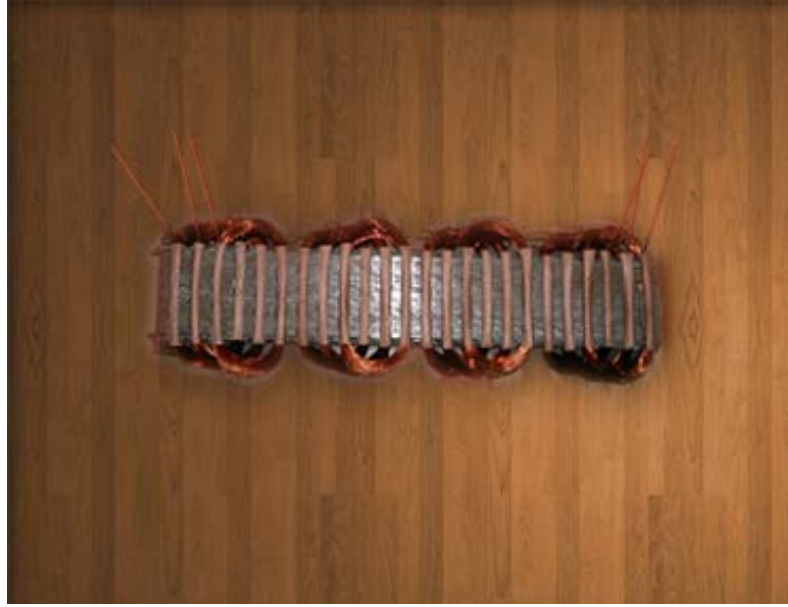
Motor sarımı için aşağıdaki denklemler kullanılmıştır.

$$C = \frac{x}{2.P.m} \quad (38)$$

$$\alpha = \frac{180.2.P}{x} \quad (39)$$

$$y_x = \frac{x}{2.P} \quad (40)$$

Burada, x Birincil Yan oluk sayısı, $2P$ çift kutup sayısı, P tek kutup sayısı, m faz sayısı, α iki oluk arasındaki elektriksel açı, C bir kutup altında bir faza düşen renk sayısı, y_x oyuk adımını göstermektedir. Verilenlere göre yapılan sarım Şekil 50'de gösterilmiştir.



Şekil 50. Birincil yan sarımı

Yapılan bu işlemlerden sonra elde edilen Birincil Yan görünümü Şekil 51’de gösterilmektedir.



Şekil 51. Birincil yan son durum görünümü

4.1.2. İkincil Yan Yapım Aşamaları

ÇYDHSMSM hareketine göre tasarlanan İkincil Yan yaklaşık 5 m olarak düşünülmüştür. Burada İkincil Yan'da mıknatıslar yüzeye yerleştirilecek biçimde tasarlanmıştır. Şekil 52'de İkincil Yan yapımının ilk aşamaları görülmektedir.



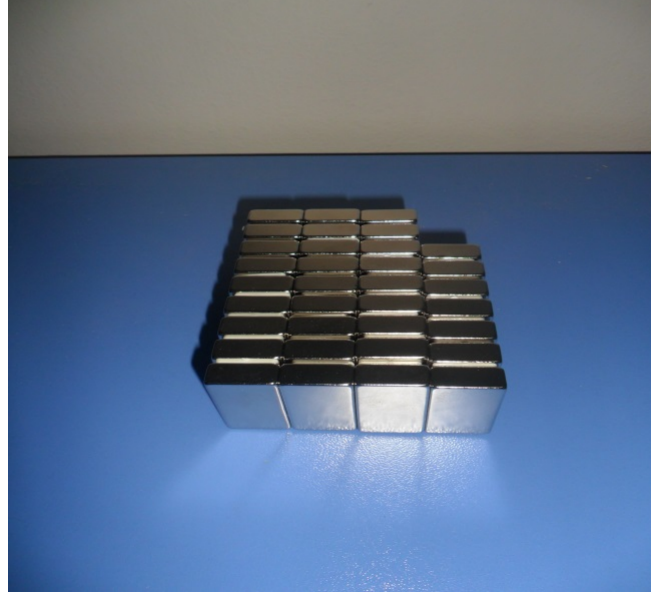
Şekil 52. İkincil yan ilk yapım aşamaları

İkincil Yan, yumuşak demir çekirdek üzerine mıknatıslar yüzeyde olacak şekilde yerleştirilmiştir. Bu çalışmada manyetik akı yoğunluğu diğer çeşitlerine göre daha fazla olan Neodyum (Nd-Fe-B) mıknatıslar kullanılmıştır. Burada kullanılan Neodyum mıknatıslara ait ebatlar Tablo 5'de verilmiştir.

Tablo 5. Kullanılan neodyum mıknatıs ölçüleri

Birimler	Ölçüler
Neodyum mıknatıs boyu	40 mm
Neodyum mıknatıs eni	20 mm
Neodyum mıknatıs kalınlığı	10 mm

Tasarlanan motorda kullanacağımız sürekli mıknatıs görüntüleri toplu halde Şekil 53'de gösterilmektedir.



Şekil 53. Kullanılan sürekli mıknatıslar

Mıknatıs ölçüleri Birincil Yan için kullanılan kutup adımı dikkate alınarak ayarlanmıştır. Yetkili firma ile görüşülerek çift yanda olacak şekilde 300 e yakın neodyum mıknatıs kullanılmıştır. Mıknatıs yerleşimi Şekil 54'deki gibi gerçekleştirilmiştir.



Şekil 54. İkincil yan sürekli mıknatıs yerleşimi

Bu işlemlerden sonra tasarlanan motorun son hali ve laboratuvar çalışma ortamı Şekil 55'deki gibi gerçekleştirilmiştir.



Şekil 55. Laboratuvar çalışma ortamı

4.2. Kullanılan Değişken Frekanslı Sürücüsü

ÇYDHSMSM'nin hız denetimini sağlamak için modeli LG-SV022iG5A-4 olan Değişken Frekanslı Sürücü (DFS) kullanılmıştır [101]. Bu cihaz 2.2 kw'lık trifaze bir sürücüdür. Kullanılan bu sürücü V/f ve sensörsüz vektör denetim yapabilmekte, 0 Hz'den maksimum 400 Hz'e kadar frekans çıkışı sağlamakta ve DFS dışarıdan programlanabilir giriş çıkış terminaline ve dâhili RS485 seri porta sahiptir. Bilgisayar çıkışlarımız genellikle RS232 (RS232 ve RS485 bilgisayarlar ve diğer elektronik cihazlarda kullanılan seri haberleşme metotlarıdır.) olduğundan, RS232'yi RS485'e dönüştüren konvertör kullanılarak DFS ile haberleşme sağlanmıştır. Bu konvertör Şekil 56'da gösterilmiştir.



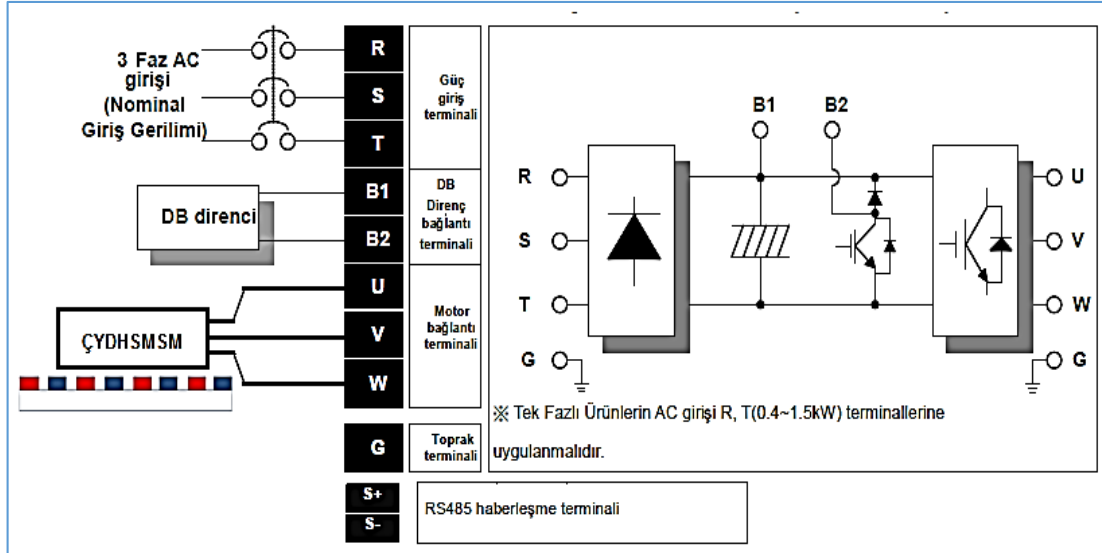
Şekil 56. RS485'i RS232'ye çeviren konvertör

Ayrıca bu sürücüye ait parametre grupları kullanılarak incelemek istediğimiz motorun parametreleri elde edilebilmektedir. Şekil 57'de bu sürücü gösterilmektedir.



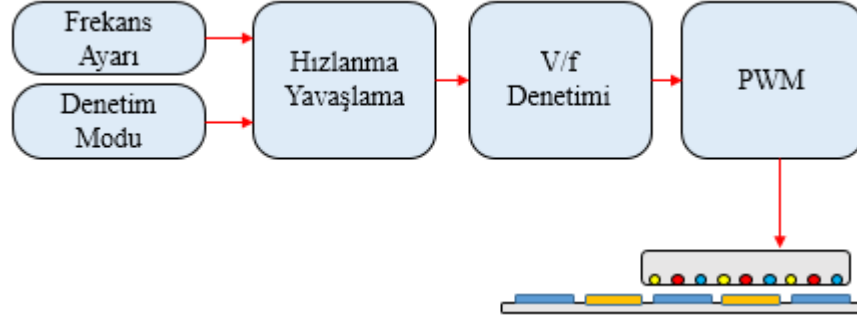
Şekil 57. LG-SV022iG5A-4 sürücü

Sürücünün bağlantı terminalleri Şekil 58'deki gibi gösterilmiştir. Burada R, S, T üç fazın giriş uçlarını, U, V, W motor giriş uçlarını, G toprak terminalini, S+ ve S- uçları bilgisayar haberleşme terminalini göstermektedir.



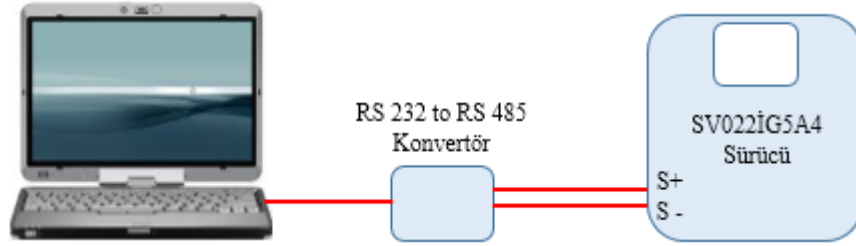
Şekil 58. LG-SV022iG5A-4 sürücüsünün bağlantı terminalleri

Yapılan bağlantı çalışmasıyla sürücünün V/f denetim gerçekleştirilmiştir. Bu duruma ait blok devre şeması Şekil 59'da gösterilmiştir.



Şekil 59. DFS ile V/f denetimin blok diyagramı

Burada sırasıyla, V/f denetim yapan sürücünün bilgisayar ile haberleşmesi sağlanmıştır. Sürücüye ait haberleşme protokolü olan **Modbus RTU** ile **C#** programlama dili ile yazmış olduğumuz program uyumlu olarak çalıştırılmıştır. Sürücünün register'a (kayıt altına) aldığı bilgiler alınarak hazırladığımız programa kaydedildi. Bu verilere göre hazırlanan ara yüz ile ÇYDHSMMSM 'İleri', 'Geri', 'Durdur' butonları ile denetim sağlanmıştır. Şekil 60'da bilgisayar ile haberleşme bağlantısı gösterilmiştir.



Şekil 60. Bilgisayar ile sürücü haberleşme bağlantısı

Bu işlemler için öncelikle RS-485 haberleşme hattını, sürücünün denetim terminallerine Şekil 59'daki gibi (S+), (S-) bağlanmıştır. Bağlantı denetlendi ve sürücü AÇIK hale getirildi. Haberleşme hattı doğru olarak bağlandıktan sonra haberleşme ile ilgili parametreler aşağıdaki gibi ayarlanmıştır.

- DRV-03 [Çalıştırma Modu]: **3** (RS485)
- DRV-04 [Frekans Modu]: **7** (RS485)
- I/O-60 [Sürücü Adresi]: 1~250 (1 'den fazla sürücü bağlıysa, her sürücü için farklı adresler kullandığımızdan emin olun)
- I/O-61 [Haberleşme hızı]: **3** (Fabrika ayarı olarak 9,600 bps)
- I/O-62 [Kayıp Modu]: **0** - Eylem yok (Fabrika ayarı)
- I/O-59 [Haberleşme Protokolü]: **0** - Modbus-RTU, 1 – LS BUS

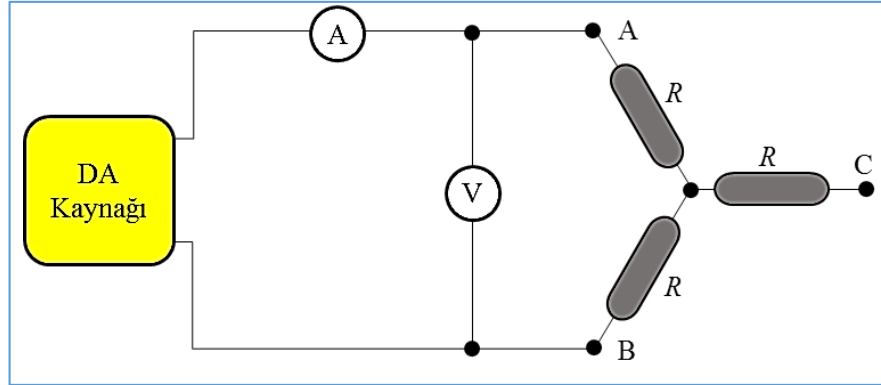
4.3. Motor Parametrelerinin Belirlenmesi

DHSMMSM'lerin tasarımında ve denetim sistemlerinde makine modelinin ve parametrelerinin kesin olarak bilinmesi, makineden yüksek verim ve düşük maliyet elde edilmesini sağlar. Bu parametrelerin belirlenmesinde motor boşta ve yükte iken deneyler yapılmaktadır [102] [103].

İncelenen motorun parametrelerinin (R ve X_L değerlerinin) bulunabilmesi için üç farklı yöntem kullanılmıştır. Birinci yöntem deneysel yöntemdir ve sonraki bölümde detaylı olarak açıklanmaktadır. İkinci yöntemde ise bu değerler hassas bir LCR (İndüktans-Kapasitans-Direnç) metre ile ölçülmektedir. Son yöntemde ürün kodu LG-SV022iG5A-4 olan motor sürücüsüne parametre bulma deneyi yaptırılmıştır.

4.3.1. Birincil Yan Sargı Direncinin Bulunması

Deneysel Yöntem: Doğru Akım (DA) deneyi için temel devre şeması Şekil 61'de gösterildiği gibidir. Bu şekil, Birincil Yanı çift yanlı ve sargıları yıldız bağlı senkron motorun giriş uçlarından ikisine bağlanan bir DA güç kaynağını göstermektedir. Bağlanan ölçü aletlerinden okunan değerler ile Birincil Yan direnci elde edilmektedir.



Şekil 61. Doğru akım deneyi bağlantı şeması

Güç kaynağından uygulanan, ayarlanabilir DA ile ölçü aletlerinden okunan değerler 11 numaralı bağıntıda kullanılarak Birincil Yan direnci elde edilmiştir.

$$R = \frac{V_{DA}}{2I_{DA}} \quad (41)$$

Uygulanan gerilim değerleri ile elde edilen akım değerlerine karşılık gelen direnç değerleri Tablo 6’da verilmiştir. Bu tablodaki verilere dayanarak ortalama Birincil Yan R değeri **11.58 Ω** olarak hesaplanmıştır.

Tablo 6. DA deneyinden elde edilen sonuçlar

$V_{AB}(V)$	$I_{AB}(A)$	$R_{AB}(\Omega)$	$V_{AC}(V)$	$I_{AC}(A)$	$R_{AC}(\Omega)$	$V_{BC}(V)$	$I_{BC}(A)$	$R_{BC}(\Omega)$
19	0.81	11.73	19	0.81	11.73	19	0.82	11.56
22	0.95	11.58	22	0.96	11.46	22	0.96	11.46
26	1.12	11.61	26	1.13	11.51	26	1.12	11.61

Hassas LCR Metre ile Yapılan Ölçüm: Hassas bir LCR metre ile yapılan ölçüme, $R_{AB}=23.5 \Omega$, $R_{AC}=23.3 \Omega$, $R_{BC}=23.7 \Omega$ olarak bulunmuştur. Bu ölçme sonuçlarından hareketle R değeri **11,75 Ω** olarak hesaplanmaktadır.

LG-SV022iG5A-4 Sürücüsü ile Yapılan Parametre Bulma Deneyi: LG-SV022iG5A-4 sürücüsüne parametre bulma deneyi yaptırılabilirdiği üretici firma tarafından belirtilmektedir. Bu parametre H42 adresinden okunmaktadır.

LG-SV022iG5A-4 sürücüsüne Birincil Yan direncini bulma deneyi yaptırılarak R değeri **12.78 Ω** bulunmuştur.

4.3.2. Birincil Yan İndüktansının Bulunması

Deneyel Yöntem: Bu parametrenin belirlenebilmesi için 5 Hz ve 10 Hz frekanslarında kilitli ya da tutulu Birincil Yan deneyi yapılmıştır [102], [103]. Bu deneylerde elde edilen I_K (Kilitli Birincil Yan akımı) ve V_K (Kilitli Birincil Yan gerilimi) değerlerinden, incelenen motora ait Z_K (Birincil Yan empedansı) ve X_L hesaplanmıştır. Yapılan bu işlemlerde elde edilen sonuçlar Tablo 7’de gösterilmektedir.

Tablo 7. Kilitli Birincil Yan deneyi sonuçları

$f_K (Hz)$	$V_K (V)$	$I_K (A)$	$Z_K (\Omega)$	$X_L (\Omega)$
5	78.8	1.6	28.43	25.97
10	160.3	3.3	28.05	25.55

Yüzeysel mıknatıslı senkron motorlarda relüktans moment sıfır olduğu için d ve q eksenlerindeki indüktans değerleri birbirlerine eşit ve hesaplanan L değerinin $2/3$ ’ü kadar olduğu bilinmektedir [76]. Bu verilerden yola çıkılarak L değeri 41.99 mH olarak tespit edilmiştir.

Hassas LCR Metre ile Yapılan Ölçüm: Hassas bir LCR metre ile yapılan ölçüme, $L_{AB}=91.1$ mH, $L_{AC}=80.2$ mH, $L_{BC}=79.8$ mH olarak bulunmuştur. Bu ölçme sonuçlarından hareketle L değeri **41.81** mH hesaplanmıştır.

LG-SV022iG5A-4 Sürücüsü ile Yapılan Parametre Bulma Deneyi: Sürücü parametrelerinden biri olan H44 adresinden okunan değer Birincil Yan indüktansına ait olduğu üretici firma tarafından belirtilmektedir. Sürücüye Birincil Yan indüktansını bulma deneyi yaptırılarak L değeri **43.48mH** olarak bulunmuştur. Deneylerden elde edilen sonuçlar Tablo 8’de gösterilmektedir.

Tablo 8. Birincil Yan direnç ve indüktans değerleri

Parametreler	Deney Sonuçları	LCR Metre	SV022İG5A
Direnç (R)	11.58Ω	11.75Ω	12.78Ω
İndüktans (L)	41.99mH	41.81mH	43.48mH

Yapılan bu çalışmada incelenen motorun parametreleri üç farklı yöntemle bulunmuştur. Bu yöntemlerden elde edilen sonuçlar arasındaki küçük farklılıkların kullanılan ölçü aletlerinin hassasiyetlerinden kaynaklandığı düşünülmektedir.

4.3.3. Sürekli Mıknatıs Akısının Hesaplanması

Sürekli mıknatısın manyetik akı değerinin hesaplanmasında 42 numaralı denklem kullanılarak parametre değeri hesaplanmıştır [2].

Bu denklem ile yapılan hesaplama, $\psi_f = 0.189$ Wb olarak elde edilmiştir.

$$\psi_f = \frac{2}{\pi} \tau N B_f l_1 k_{w1} \quad (42)$$

Burada N bir faza ait bir oluktaki sarım sayısı, l_1 Birincil Yan sac paket yığınının boyu, $B_f = 0.7$ T manyetik akı yoğunluğunu, $k_{w1}=1$ sargı faktörünü göstermektedir. Burada B_f değeri manyetik malzemelerin 0-400 Hz’ deki B-H histerisiz eğrisinden düşük frekanslar için tercih edilen değeri alınmıştır [2].

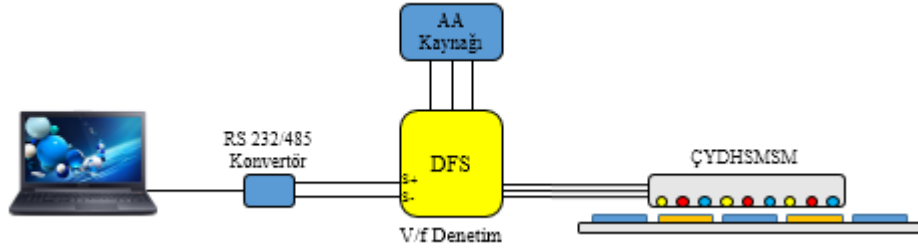
Bu çalışmada ÇYDHSMSM’nin tasarım aşamalarından bahsedilmiş ve parametrelerinin bulunması için de üç farklı yöntem kullanılmıştır. Burada deneysel yol ile bulunan parametreler denetim yöntemleri için kullanılacaktır. Tablo 9’da motora ait parametreler topluca verilmiştir.

Tablo 9. İncelenen motorun parametreleri

Parametreler	Simge	Değer
Kutup sayısı	p	8
Birincil Yan direnci	R	11.58 Ω
d -ekseni indüktansı	L_d	28mH
q -ekseni indüktansı	L_q	28mH
Mıknatıs akısı	ψ_f	0.189Wb
Faz voltajı	V	160.3V
Faz akımı	I	2.9A
Kutup adımı	τ	0.042m
Motor ağırlığı	M	15.1kg
Çalışma frekansı	f	15Hz

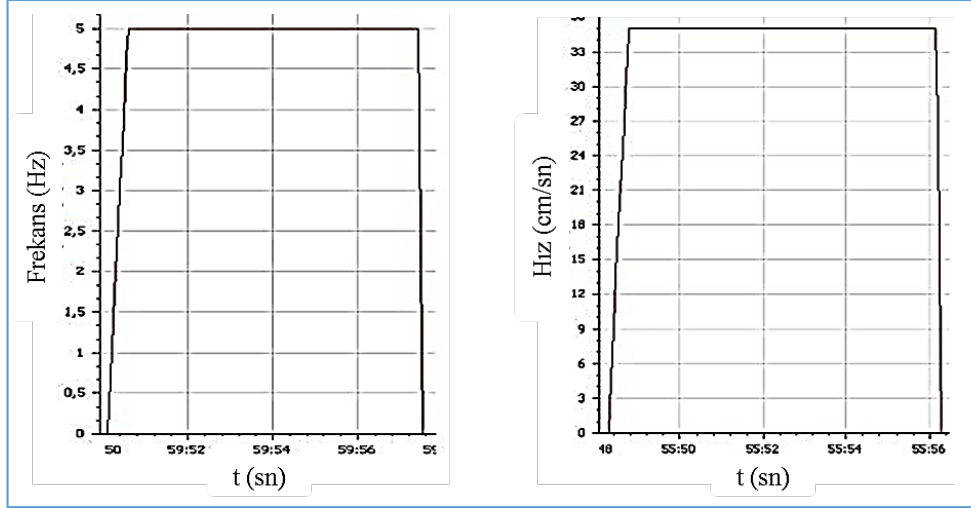
4.4. Çift Yanlı Doğrusal Hareketli Sürekli Mıknatıslı Senkron Motorun Skaler (V/f) Yöntem ile Hız Denetimi

Sürücü v/f denetim moduna ayarlandıktan sonra Şekil 62’de verilen blok devre şeması uygulanmıştır. Bilgisayar programında hazırlanan arayüz ile ÇYDHSMMSM hareket ettirilmiştir. Arayüze girilen frekans değerine karşılık akım, hız ve itme kuvveti grafikleri deneysel yol ile elde edilmiştir.

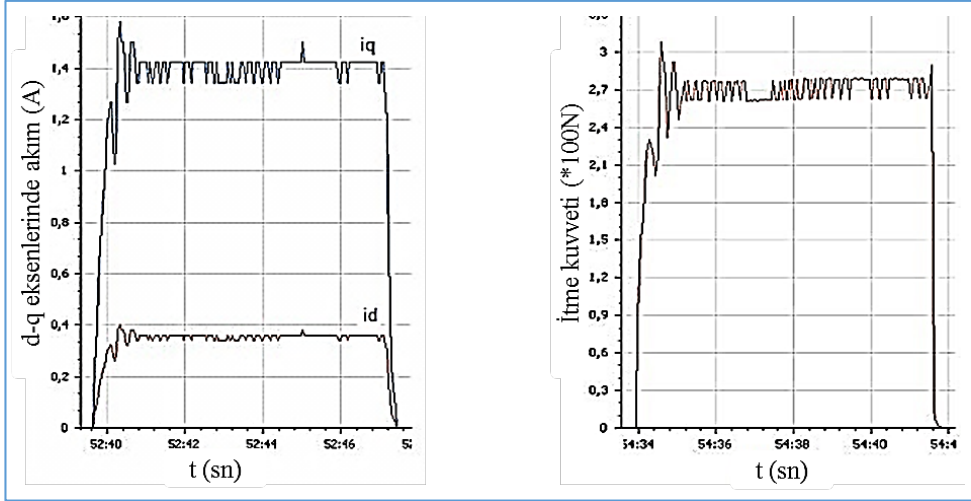


Şekil 62. Skaler denetim blok diyagramı

Bilgisayara girilen 5 Hz’lik frekanstan, motorun matematiksel denklemleri kullanılarak Şekil 63 ve 64’deki grafikler elde edilmiştir. Burada akım grafikleri için 9 ve 10 numaralı denklem, itme grafiği için 18 numaralı denklem, hız için 11 numaralı denklem kullanılmıştır.



Şekil 63. V/f denetimde frekans ve hız grafikleri



Şekil 64. V/f denetimde elde edilen akım ve itme kuvveti grafikleri

Deney düzeneği yaklaşık 500 cm olduğundan motor test edilirken sürücü frekansı maksimum 15 Hz' e kadar çıkarılabilmektedir. Burada test sonuçları elde edilirken 5 Hz' de motor çalıştırılmıştır. Dolayısıyla 5 Hz için maksimum hız 35 cm/s'dir. Test edilen bu motor için, sürücünün 400 Hz çıkış frekansı verdiği ve uygun bir hareket mesafesi olduğu düşünüldüğünde teorik olarak saatte yaklaşık 100 Km yol alabilmektedir.

11 numaralı denklem göz önüne alındığında, bu tür motorların hız değişiminde iki parametrenin etkili olduğu görülmektedir. Bunlar Frekans ve Kutup adımı'dır. Burada frekansın veya kutup adımının artması motorun saatteki hızının artması anlamına gelmektedir.

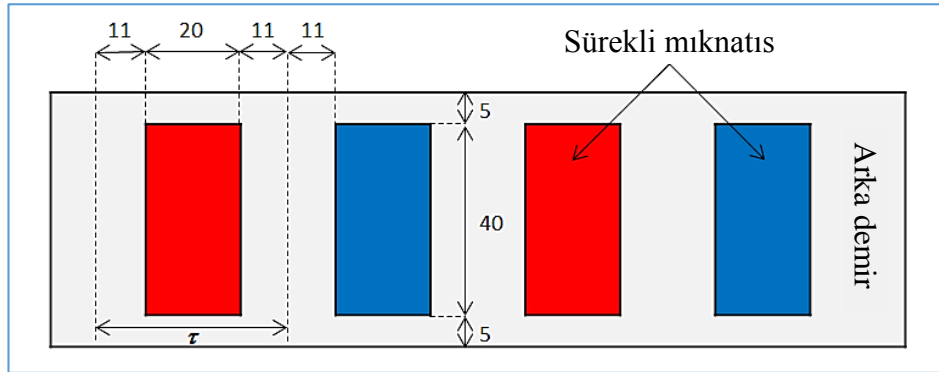
4.5. Çift Yanlı Doğrusal Hareketli Sürekli Mıknatıslı Senkron Motorun İkincil Yan Düzeninin İtme Kuvvetine Etkisinin İncelenmesi

Bu çalışmada, incelenen ÇYDHSMSM'nin İkincil Yan'ında bulunan sürekli mıknatıslarının yerleşiminin motorda meydana getirdiği itme kuvvetine etkisi deneysel olarak incelenecektir. Bu deney için Şekil 61'deki blok devre şeması kullanılmıştır.

İkincilyan'da neodyum mıknatıslar kullandığımızı daha önceki bölümlerde belirtmiştik. Yumuşak demir levha (arka demir) üzerindeki bu mıknatıslar Tablo 5'de verilen boyutlardadır. İtme kuvvetindeki değişimlerin tespiti için iki yöntem deneysel olarak gerçekleştirilmiştir. Bu iki yöntemden birincisi mıknatısların kutup adımı boyunca dikey olarak yerleştirilmesi, ikincisi ise yine mıknatısların kutup adımı boyunca yatay yerleştirilmesidir.

4.5.1. Mıknatısların Dikey Yerleştirilmesi

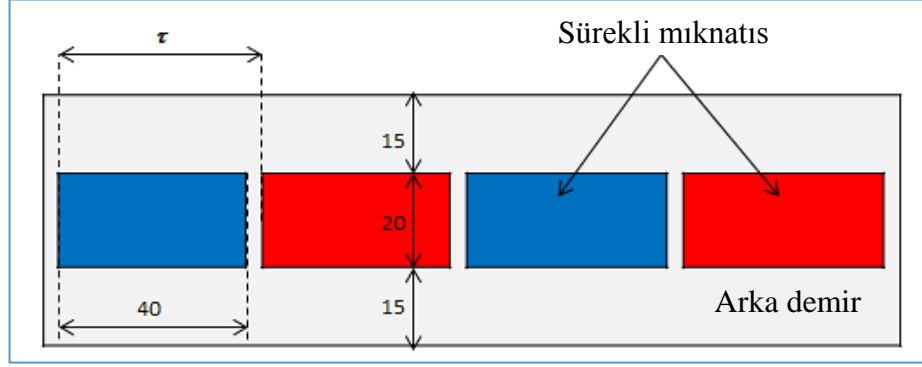
İtme kuvvetindeki değişimi görmek amacıyla İkincil Yan'da test edilen ilk yöntemdir. Birincil Yan'da bulunan sargıların kutup adımına göre sabit kısımda yani İkincil Yan'da mıknatısların yerleştirme işlemi yapılmıştır. İkincilyan'da mıknatısların dikey yerleşim modeli Şekil 65'deki gibidir. Model üzerindeki ölçülerin birimleri mm'dir.



Şekil 65. Sürekli mıknatısların dikey yerleştirilmesi

4.5.2. Mıknatısların Yatay Yerleştirilmesi

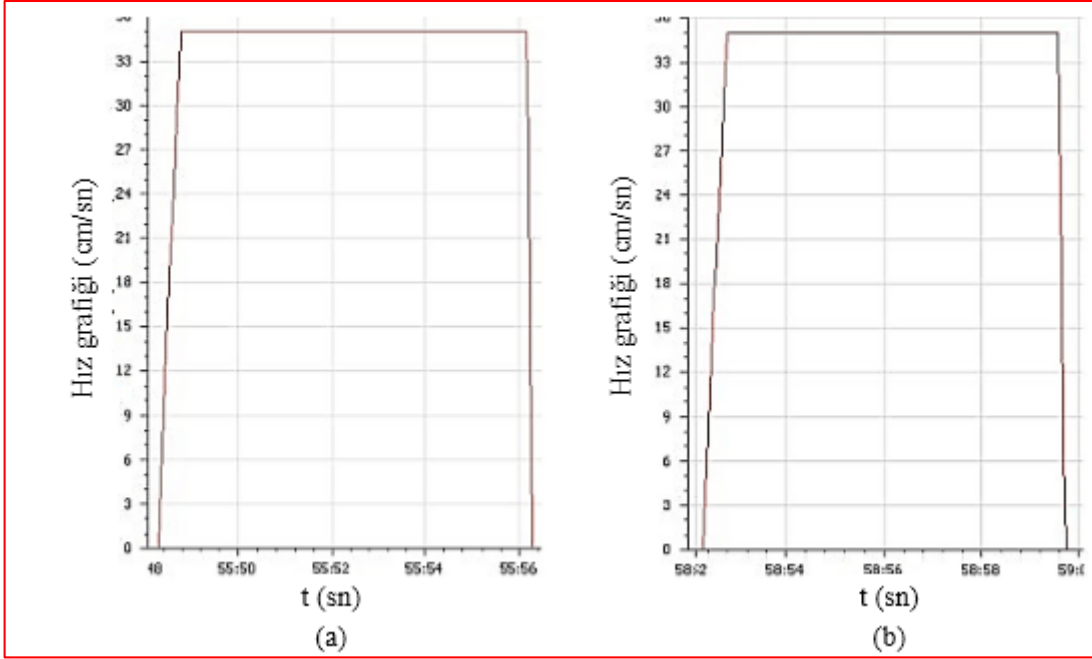
İtme kuvvetindeki değişimi görmek amacıyla İkincil Yan'da test edilen ikinci yöntemdir. Burada da kutup adımı göz önünde bulundurularak sabit kısımda mıknatısların yerleştirme işlemi yapılmıştır. Yapılan yatay yerleşim modeli Şekil 66'daki gibidir. Yine model üzerinde verilen ölçülerin birimleri mm'dir.



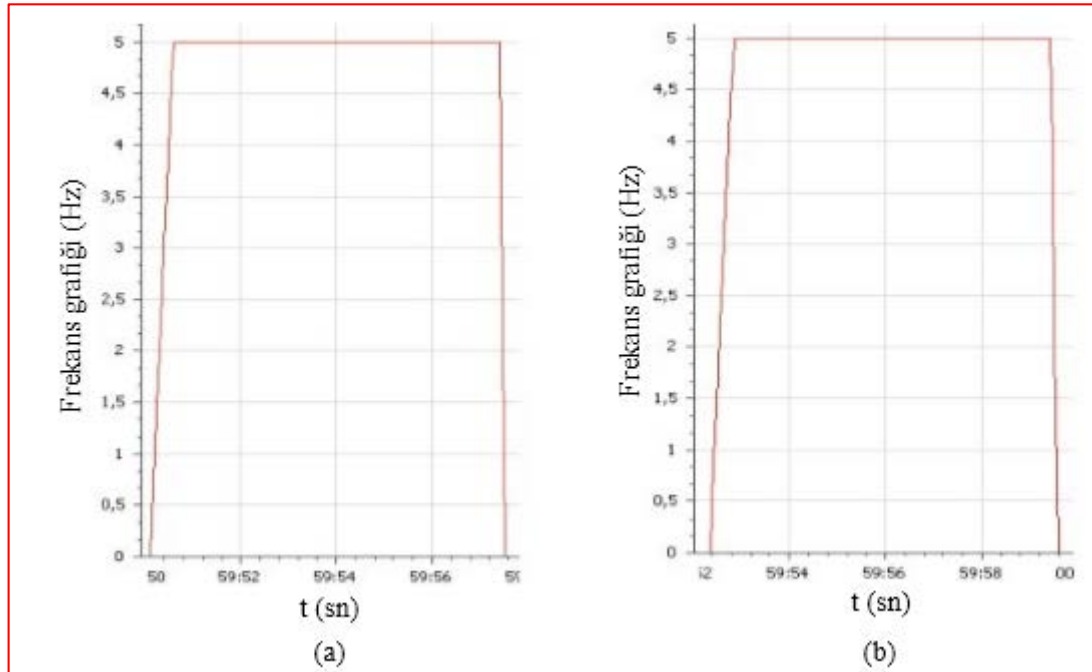
Şekil 66. Sürekli mıknatısların yatay yerleştirilmesi

4.5.3. Deneysel Sonuçlar

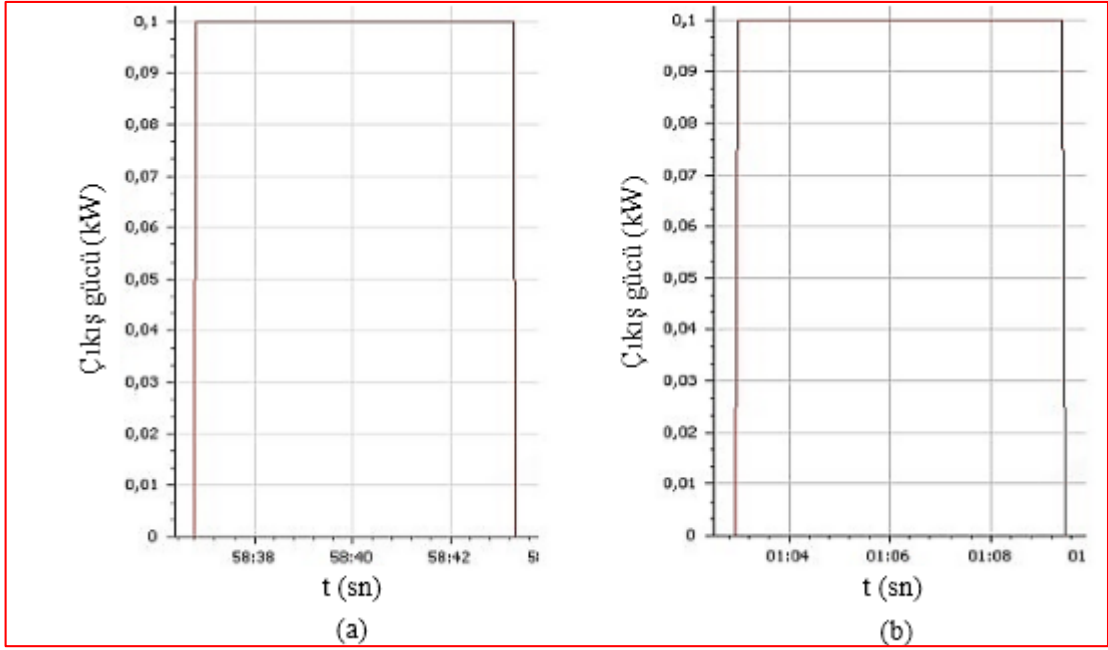
Aşağıda mıknatısların konumlarına göre ÇYDHSMSM'nin test sonuçlarından elde edilen grafikler verilmiştir. Şekil 67, 68, 69 ve 70'de verilen grafiklerin, sol sütununda olanlar mıknatısların dikey yerleşiminden, sağ sütununda olanlar ise mıknatısların yatay yerleşiminden elde edilmiştir. Karşılaştırma yapılabilmesi için mıknatısların dikey ve yatay yerleşiminden elde edilen aynı türden grafikler yan yana verilmiştir. Aradaki farklılıkların anlaşılır olabilmesi için ÇYDHSMSM'de yapılan deneyler Denklem 11'e göre aynı hızın elde edildiği 5 Hz'lik sabit frekansta gerçekleştirilmiştir.



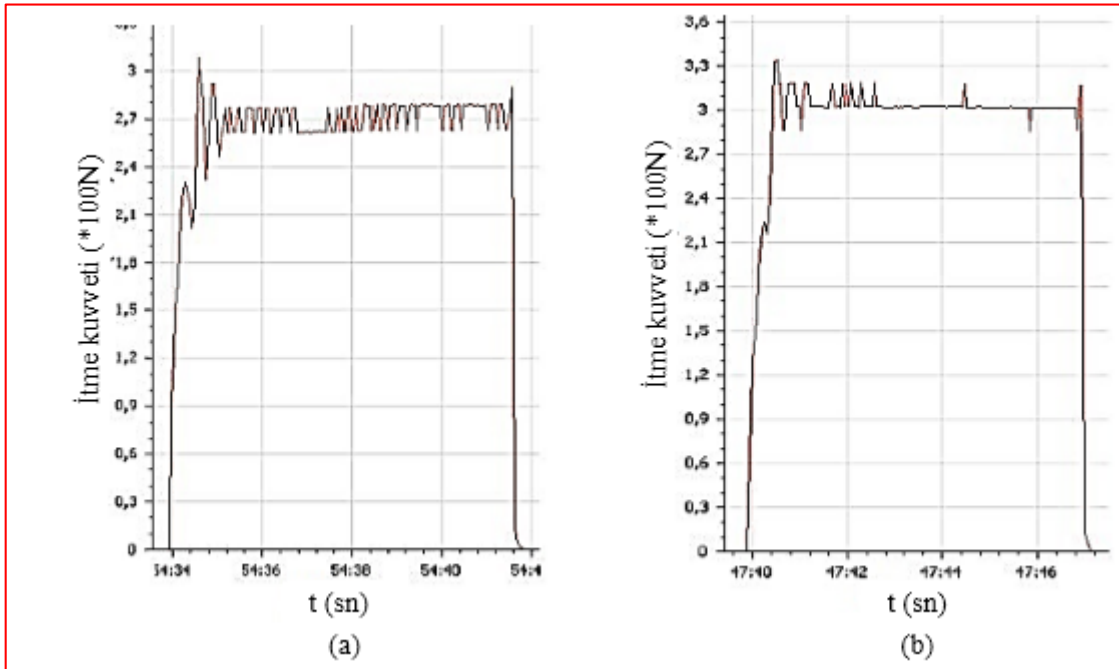
Şekil 67. Mıknatısların dikey ve yatay konumdaki hız grafikleri
 (a) Dikey konum
 (b) Yatay konum



Şekil 68. Mıknatısların dikey ve yatay konumdaki frekans grafikleri
 (a) Dikey konum
 (b) Yatay konum



Şekil 69. Mıknatısların dikey ve yatay konumdaki çıkış gücü grafikleri
 (a) Dikey konum
 (b) Yatay konum



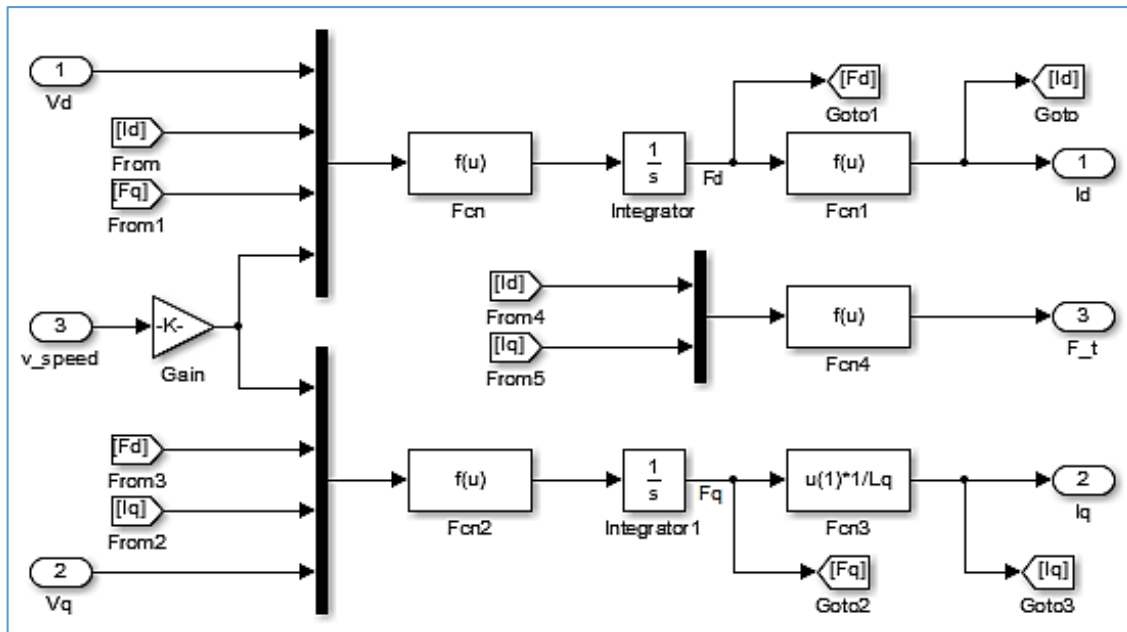
Şekil 70. Mıknatısların dikey ve yatay konumdaki itme kuvveti grafikleri
 (a) Dikey konum
 (b) Yatay konum

Verilen grafiklerde; mıknatısların dikey ve yatay konumu için, motorun hızında ve çıkış gücünde bir değişikliğin olmadığı Şekil 67 ve 69'dan anlaşılmaktadır. Buna rağmen Şekil 70'deki itme kuvveti grafikleri incelendiğinde, mıknatısların dikey konumu için itme kuvveti değeri 270 N iken mıknatısların yatay konumu için itme kuvveti değeri 300 N olarak tespit edilmiştir. Mıknatıslar yatay konumda iken itme kuvvetinde 30 N'luk bir artış elde edilmiştir. Başka bir ifadeyle mıknatıslar yatay yerleştirildiğinde itme kuvvetinde %11'lik bir artış olduğu görülmektedir.

Elde edilen sonuçlara göre; sürekli mıknatısların, kutup adımı boyunca yatay olarak yerleştirildiğinde itme kuvvetine olumlu yönde katkı sağladığı görülmektedir. İtme kuvvetindeki bu artış motorun kalkış esnasındaki verimini olumlu yönde etkilemektedir. Bu çalışmanın yapılması planlanan tasarım çalışmalarına olumlu bir katkı sağlayacağı düşünülmektedir.

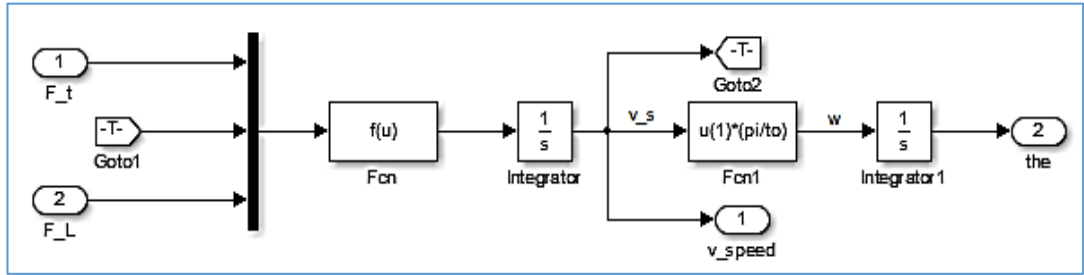
4.6. Tasarlanan Motorun Matlab/Simulink Yazılım Programı ile Benzetimi

Performans artırıcı analiz çalışmalarına imkân tanınması [57] amacıyla benzetim çalışmaları Matlab/Simulink yazılımı ile gerçekleştirilmiştir. ÇYDHSMSM modeli oluşturulurken 9-14 numaralı denklemler ve 17-18 numaralı denklemler kullanılmıştır. Şekil 71'de motora ait elektriksel simulink model gösterilmiştir.



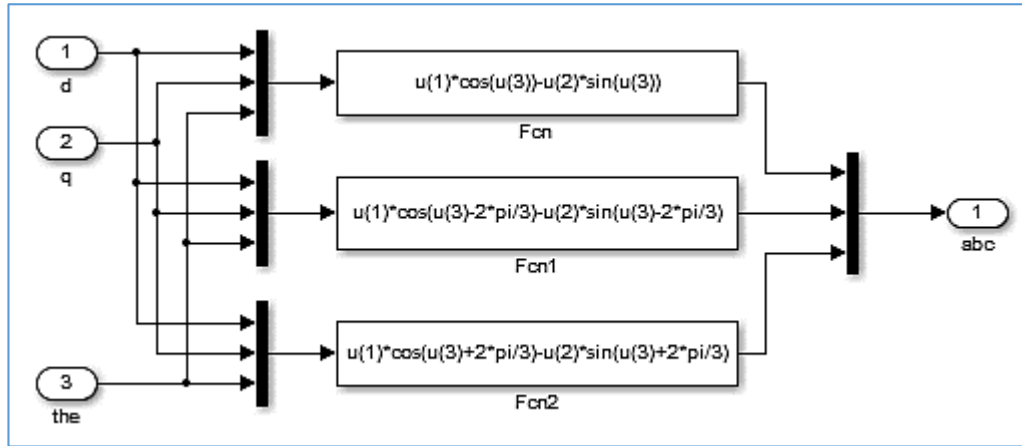
Şekil 71. Çift yanlı doğrusal hareketli sürekli mıknatıslı senkron motorun elektriksel benzetim modeli

Benzer olarak oluşturulan ÇYDHSMMSM'nin mekanik modeli de Şekil 72'de gösterilmiştir.



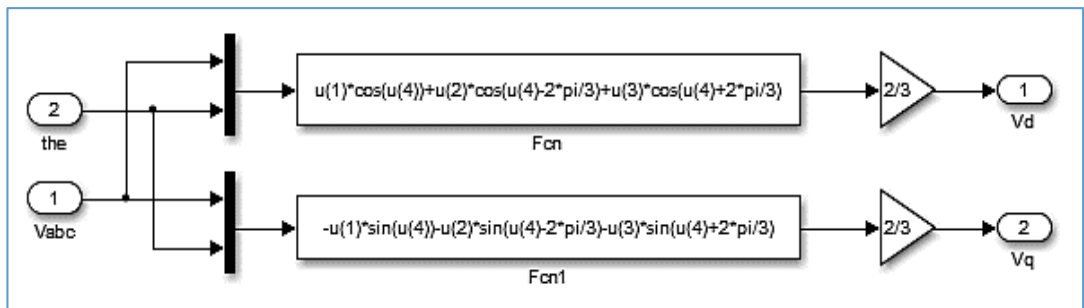
Şekil 72. Çift yanlı doğrusal hareketli sürekli mıknatıslı senkron motorun mekaniksel benzetim modeli

8 numaralı denklemden dq - ABC eksen sistemine ait dönüşümün simulink modeli de Şekil 73'de gösterilmiştir.



Şekil 73. dq - ABC eksen sistemi dönüşümünün benzetim modeli

7 numaralı denklem yardımıyla elde edilen ABC - dq eksen sistemine ait dönüşümün simulink modeli Şekil 74'de gösterilmektedir.

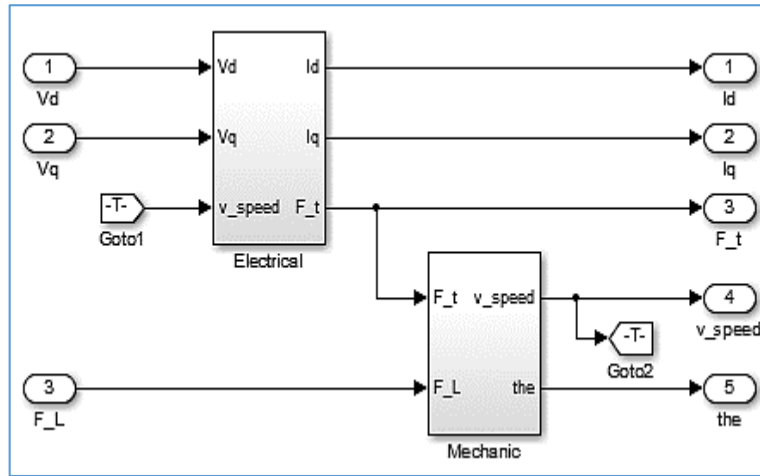


Şekil 74. ABC - dq eksen sistemi dönüşümünün benzetim modeli

4.6.1. Motorun Benzetim Modelinin V/f Denetim Yöntemi ile Durum İncelemesi

Bu çalışmada, ÇYDHSMSM ve Skaler Denetim Yöntemi (SDY, V/f)'nin, Matlab/Simulink yazılım programında benzetim modeli oluşturulmuştur. Elde edilen modellerle açık ve kapalı çevrim hız denetimi gerçekleştirilmiştir. Benzetim çalışmalarına göre sistemin performans analizi yapılmıştır.

Verilen denklemlere göre ÇYDHSMSM'nin benzetim modeli Şekil 75'deki gibidir.

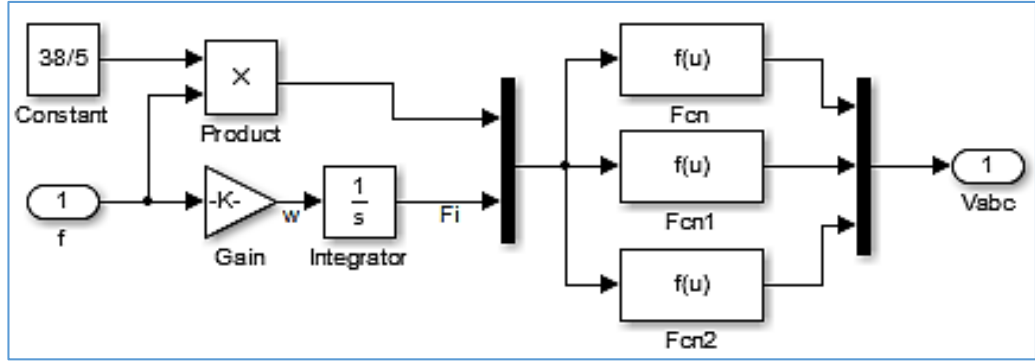


Şekil 75. Doğrusal hareketli sürekli mıknatıslı senkron motorun benzetim modeli

SDY'nin benzetimi için frekans 0-10Hz alınmıştır. SDY oranı (sabiti) 7.6 olarak tespit edilmiştir. Bunun yanında, SDY'nin sinüs üretici için kullanılan denklemler [104], [105] aşağıda verilmiştir.

$$\begin{aligned}
 S_a &= \sqrt{2} \cdot S_{etkin} \cdot \sin \theta \\
 S_b &= \sqrt{2} \cdot S_{etkin} \cdot \sin \left(\theta - \frac{2\pi}{3} \right) \\
 S_c &= \sqrt{2} \cdot S_{etkin} \cdot \sin \left(\theta + \frac{2\pi}{3} \right)
 \end{aligned} \tag{43}$$

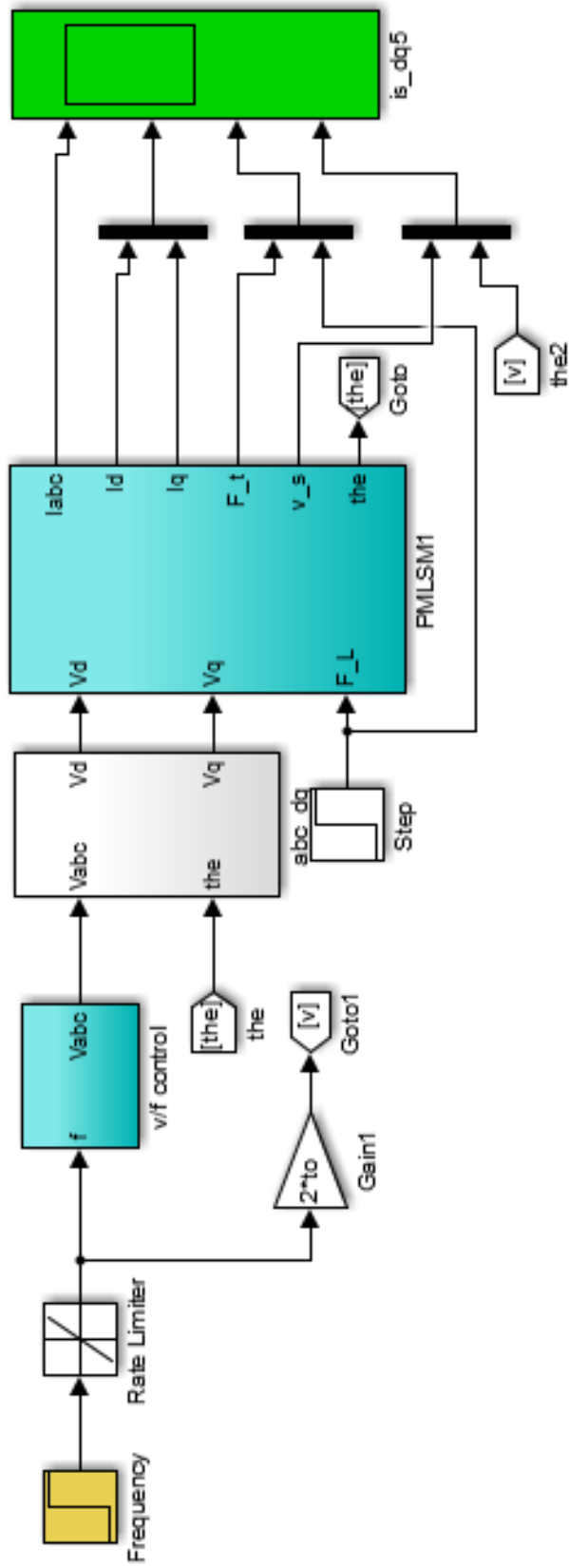
Burada S parametresi akım ve gerilimi ifade etmektedir. SDY için simulink model Şekil 76'da gösterildiği gibidir.



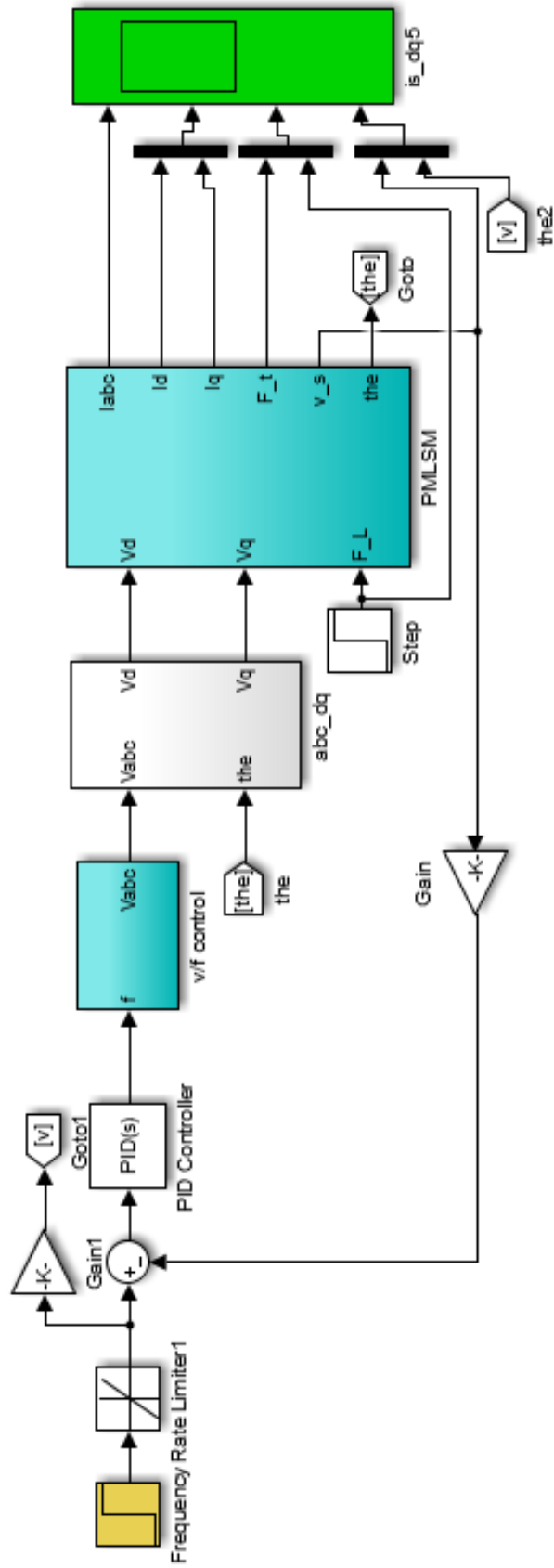
Şekil 76. Skaler denetimin benzetim modeli

4.6.1.1. Motorun Açık ve Kapalı Çevrim Benzetim Modelleri

ÇYDHSMSM performans analizi için iki yöntem kullanılmıştır. İki yöntemde de sisteme giriş olarak referans frekans uygulanmıştır. Girişe uygulanan frekans ile çıkışta elde edilen akım, itme kuvveti ve hız değişimleri analiz edilmiştir. SDY ile yapılan açık çevrim denetim benzetim modeli Şekil 77’de, kapalı çevrim denetim benzetim modeli Şekil 78’de gösterilmiştir.



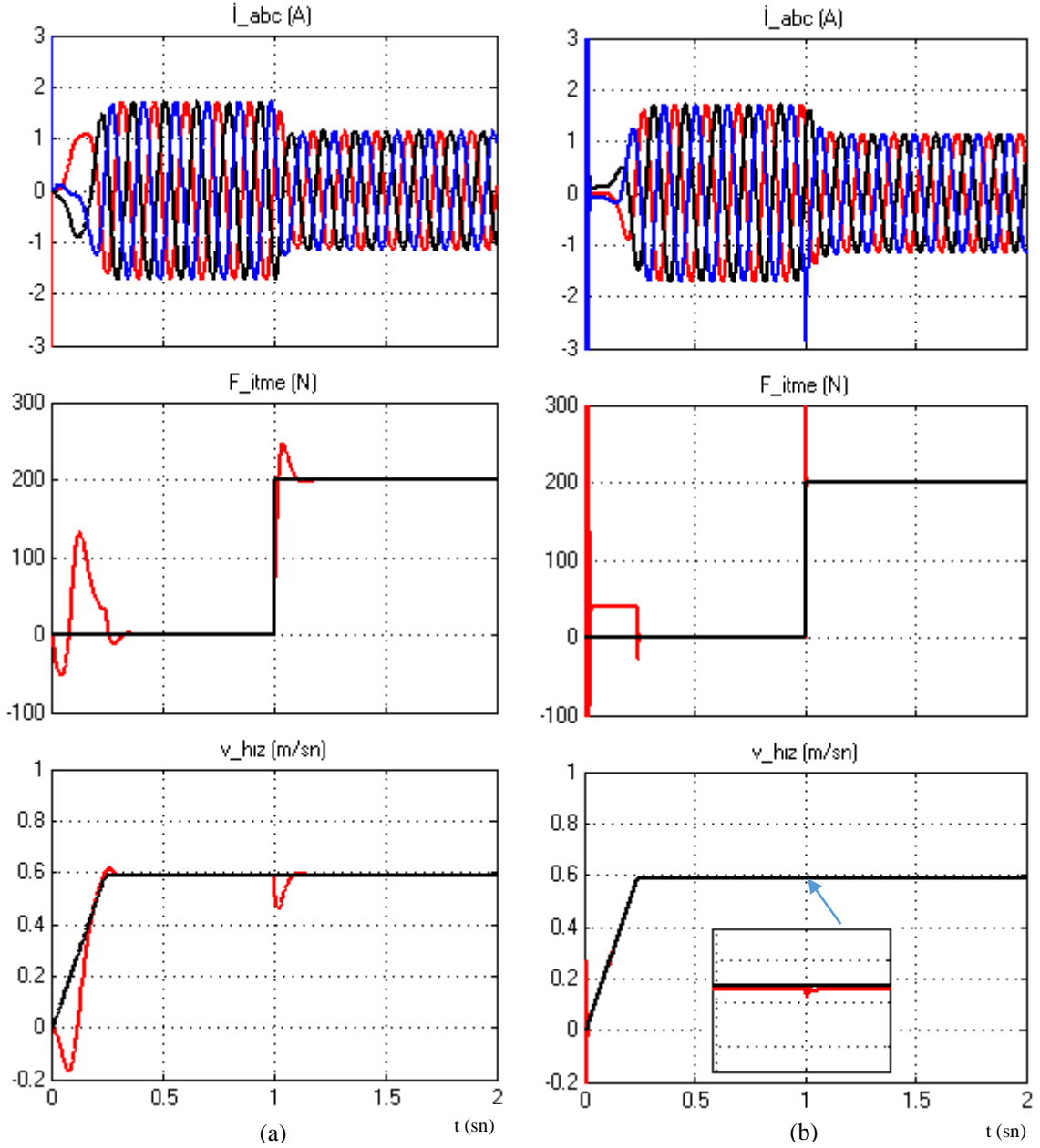
Şekil 77. Skaler denetimin açık çevrim benzetim modeli



Şekil 78. Skaler denetimin kapalı çevrim benzetim modeli

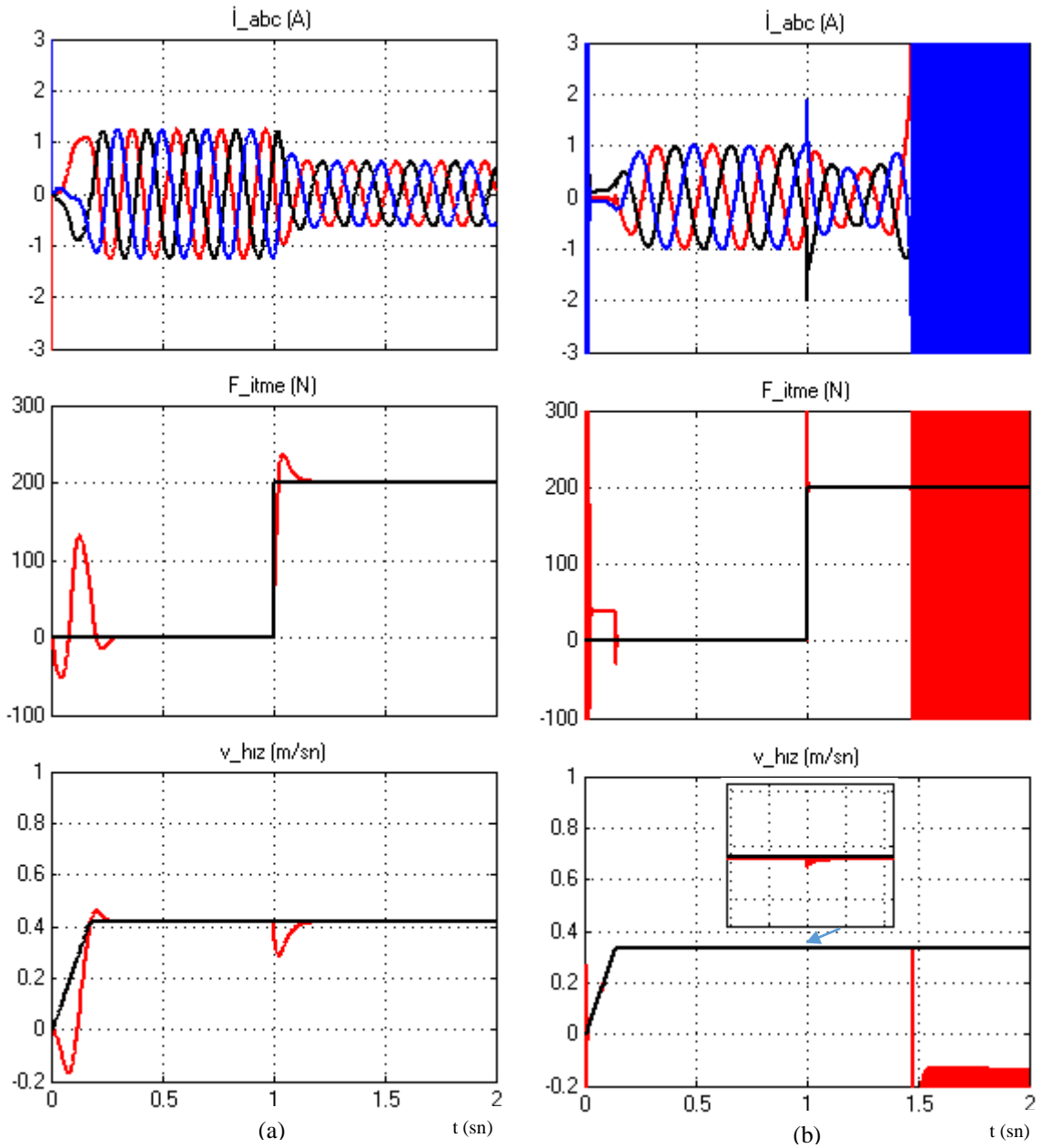
4.6.1.2. Benzetim Sonuçları ve Değerlendirme

ÇYDHSMSM'nin 7 ve 4 Hz frekans değerleri için açık ve kapalı çevrim benzetim modeli grafikleri sırasıyla Şekil 79 ve 80'de gösterilmektedir.



Şekil 79. 7 Hz frekans değeri için elde edilen benzetim sonuçları

- (a) Açık çevrim sonuçları
- (b) Kapalı çevrim sonuçları



Şekil 80. 4 Hz frekans değeri için elde edilen benzetim sonuçları

(a) Açık çevrim sonuçları

(b) Kapalı çevrim sonuçları

Şekil 79'da (7 Hz frekans); i_{abc} grafiklerinde önemli bir değişim gözlenmemektedir. Ancak itme kuvveti ve hız grafiklerinde özellikle kalkış ve yüklenme anında değişimler gözlenmektedir. Açık çevrimde itme, kalkış anında düşük seviyededir. Buna karşılık kapalı çevrimde yüksek seviyede olduğu görülmektedir. ÇYDHSMSM yüke bindiği sırada açık çevrim itme kuvveti referansını kapalı çevrime göre daha uzun zamanda yakaladığı görülmektedir. Açık ve kapalı çevrimde yaklaşık 0.10 s fark bulunmaktadır. Hız grafikleri incelendiğinde motor yüklendiği anda kapalı çevrim denetiminde hızda bir değişimin

olmadığı görülmektedir. Buna karşılık açık çevrim denetimde hızın yaklaşık 0.10 s bir gecikme ile referans hızı yakaladığı görülmektedir.

Şekil 80'de (4 Hz frekans); açık çevrim benzetim sonuçları 5 Hz'lik frekans değerlerinin altında kapalı çevrime göre daha iyi sonuçlar üretmektedir. Kapalı çevrimde hem istenen hıza ulaşılamamakta hem de motor yüke bindikten sonra hız değeri bozulma göstermektedir.

Bu sonuçlara göre SDY, PID ile yapılan kapalı çevrim denetim modelinde, açık çevrim modeline göre daha iyi sonuçlar verdiği görülmektedir. Ancak düşük frekanslarda açık çevrim benzetimi kapalı çevrime göre daha iyi sonuçlar vermektedir.

4.7. Çift Yanlı Doğrusal Hareketli Sürekli Mıknatıslı Senkron Motorun Yapay Sinir Ağları ile Hız Denetimi

4.7.1. Çift Yanlı Doğrusal Hareketli Sürekli Mıknatıslı Senkron Motorun Yapay Sinir Ağları ile Yol-Zaman Denetimi

Yapay sinir ağları (YSA); Denetim, Haberleşme, Radar Sinyallerinin işlenmesi, Örüntü tanıma, Non-linear sistemler, vb. alanlarda başarıyla çalışmaktadır. Özellikle denetim alanında birçok çalışma gerçekleştirilmiştir [37, 106]. Bu bağlamda bu çalışmada doğrusal hareketli motorun denetimi çok katmanlı ileri beslemeli sinir ağı (Multilayer Feedforward Network) ile gerçekleştirilmiştir. Ağın eğitimi hata geri yayılım algoritması (Error Back-Propagation Training) kullanılarak yapılmıştır. Çok katmanlı ileri beslemeli sinir ağları non-linear problemlerin çözümünde başarılı sonuçlar üretmektedir [35, 107].

Bu çalışmada, YSA ile motor sürücüsünün frekansı değiştirilerek Doğrusal Hareketli Sürekli Mıknatıslı Senkron Motorun (DHSMSM) istenen hıza ulaşması sağlanmıştır.

Burada YSA motorun gitmesi istenen yol ve bu yolu gitmesi istenen zamana göre (referans yol ve zaman) ortalama hız değerini hesaplar. Aynı zamanda bu hıza karşılık frekans değeri hesaplanır. Motor yol aldıkça kalan yol ve kalan zaman hesabı yapılarak bu işlemler tekrarlanır. YSA bu durumun denetimini yapar. Kalan yol ve zamana göre frekansın arttırılıp azaltılacağına karar verir.

Sistemin çalışma aşamaları aşağıdaki gibi izah edilebilir:

1. Aşama: Bilgisayar hazırlanan programın arayüzüne motorun gitmesi istenen yol ve bu yolu gitmesi gereken zaman bilgileri girilir.

$$x = vt \quad (44)$$

Sabit hızlarda geçerli olan bu denklemde x yol, t ise motorun bu yolu gitmesi istenen zamandır. Bu denklemden ortalama hız hesaplanır. Bu hıza karşılık frekans değeri denklem 45 den elde edilir.

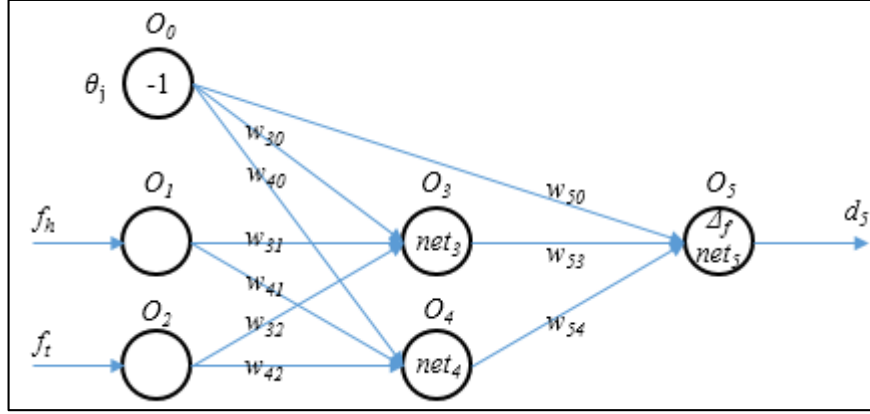
$$f = \frac{v}{2\tau} \quad (45)$$

2. Aşama: Girilen yol ve zaman bilgilerine göre YSA eğitim setini oluşturur. YSA girilen değerlerin haricinde random yol ve zaman değerleri üreterek eğitim setini gerçekleştirir. Örnek eğitim seti Tablo 10'da gösterilmiştir.

Tablo 10. Örnek YSA eğitim seti (Yol-Zaman)

$x(\text{cm})$	$t(\text{s})$	$v(\text{cm/s})$	$f_h(\text{hz})$	$f_t(\text{hz})$	$\Delta f(\text{çıkış})$
350	10	35,0	4,17	3	+1
200	5	40,0	4,76	4	+1
250	7	35,7	4,25	3	+1
300	7	42,9	5,10	5,5	-1
150	5	30,0	3,57	1	+1
.

Burada, f motor frekansı (Hz), τ kutup adımı'dır. Verilen x ve t sabit değerlere karşılık frekans değeri hesaplanır. YSA değişebilen $x-t$ değerleri için rasgele değerler üretilerek bu değerlere karşılık gelecek f değerlerini hesaplar. Burada hesaplanan her bir değere f_h ve doğrusal hareketli motorun anlık frekansına f_t dersek YSA blok yapısı şekil 81 deki gibi kurulmuş olacaktır. Burada YSA'nın ürettiği çıktı Δf değeridir ve bu değer doğrusal hareketli motorun bir sonraki frekansı f_{t+1} in hesaplanmasında değişim miktarı olarak kullanılmaktadır. Denklem 46'da f_{t+1} gösterilmiştir.



Şekil 81. Deneysel çalışmanın yapay sinir ağı yapısı

$$f_{t+1} = f_t + \Delta f \quad (46)$$

Burada, Δf değeri anlık artış ya da azalış miktarı olarak kullanılacağı için YSA da kullanılan aktivasyon fonksiyonu +1 ile -1 aralığında değerler üretilecek şekilde gerçekleştirilmelidir. f_h YSA programı arayüzüne girilen yol ve zamana göre atanmaktadır. Bulunan değer sürücüdeki değerden küçük ise YSA çıkışı "+1" yani frekansı arttır demektir. Eğer büyük ise YSA çıkışı "-1" yani frekansı azalt demektir. Bu sırada gidilen yol ve geçen zaman toplam yol ve zamandan düşülür. Bu işlem her 0.1 saniyede tekrarlanır. Yol ve zaman sıfır olduğunda ise motor durur.

Aktivasyon fonksiyonu olarak +1 ve -1 aralığında değer üreten bipolar sigmoid fonksiyonu denklemi kullanılmıştır [108].

$$f(x) = \frac{2}{1 + e^x} - 1 \quad (47)$$

YSA da Δf değeri f_{t+1} anındaki değişim miktarını ifade edeceği için +1 ve -1 aralığında çıkış üreten bipolar sigmoid fonksiyonu:

$$f(net) = \frac{2}{1 + e^{-net}} - 1 \quad (48)$$

Bu fonksiyonun türevi;

$$f'(net) = \frac{1}{2}(1 - net^2) \quad (49)$$

YSA'nın bir adım ileri yönde hesaplanması aşağıdaki şekilde yapılmaktadır.

$$net_3 = O_0 \cdot w_{30} + O_1 \cdot w_{31} + O_2 \cdot w_{32} \Rightarrow O_3 = f(net_3) = \frac{1}{2} \cdot (1 - net_3^2) \quad (14)$$

$$net_4 = O_0 \cdot w_{40} + O_1 \cdot w_{41} + O_2 \cdot w_{42} \Rightarrow O_4 = f(net_4) = \frac{1}{2} \cdot (1 - net_4^2) \quad (15)$$

$$net_5 = O_0 \cdot w_{50} + O_3 \cdot w_{53} + O_4 \cdot w_{54} \Rightarrow O_5 = f(net_5) = \frac{1}{2} \cdot (1 - net_5^2) \quad (16)$$

İleri yönde hesaplama gerçekleştirildikten sonra elde edilen O_5 çıkış değeri Δf miktarına karşılık gelen değerdir ($O_5 = \Delta f$). Eğitim setindeki f_t ve f_h giriş değerleri için beklenen çıkış değerini d_5 olarak adlandırırsak ağırlık bir iterasyondaki hata hesaplaması aşağıdaki gibi gerçekleştirilir.

$$E = \frac{1}{2} \cdot (d_5 - O_5)^2 \quad (50)$$

E hata değeri beklenen hata değerinden büyük olduğu sürece eğitim iterasyonları aşağıdaki gibi tekrarlanır.

$$\delta_{O_5} = (d_5 - O_5) \cdot f'(net) = (d_5 - O_5) \cdot \frac{1}{2} \cdot (1 - O_5^2) = \frac{1}{2} \cdot (d_5 - O_5) \cdot (1 - O_5^2) \quad (51)$$

$$\delta_{O_3} = (\delta_{O_5} \cdot w_{53}) \cdot \frac{1}{2} \cdot (1 - O_3^2) = \frac{1}{2} \cdot (\delta_{O_5} \cdot w_{53}) \cdot (1 - O_3^2) \quad (52)$$

$$\delta_{O_4} = (\delta_{O_5} \cdot w_{54}) \cdot \frac{1}{2} \cdot (1 - O_4^2) = \frac{1}{2} \cdot (\delta_{O_5} \cdot w_{54}) \cdot (1 - O_4^2) \quad (53)$$

Bulunan bu değerler momentum katsayısı (η) ve çıkış değerleri ile birlikte kullanılarak w ağırlıkları aşağıdaki gibi güncellenir. Herhangi bir t zamanda güncellenen ağırlıklar;

$$w_{53}^t = w_{53}^{(t-1)} + \eta \cdot \delta_{O_3} \cdot O_3 \quad (54)$$

$$w_{54}^t = w_{54}^{(t-1)} + \eta \cdot \delta_{O_4} \cdot O_4 \quad (55)$$

$$w_{42}^t = w_{42}^{(t-1)} + \eta \cdot \delta_{O_4} \cdot O_4 \quad (56)$$

$$w_{30}^t = w_{30}^{(t-1)} + \eta \cdot \delta_{O_3} \cdot O_3 \quad (57)$$

$$w_{40}^t = w_{40}^{(t-1)} + \eta \cdot \delta_{O_4} \cdot O_4 \quad (58)$$

$$w_{31}^t = w_{31}^{(t-1)} + \eta \cdot \delta_{O_3} \cdot O_3 \quad (59)$$

$$w_{41}^t = w_{41}^{(t-1)} + \eta \cdot \delta_{O_4} \cdot O_4 \quad (60)$$

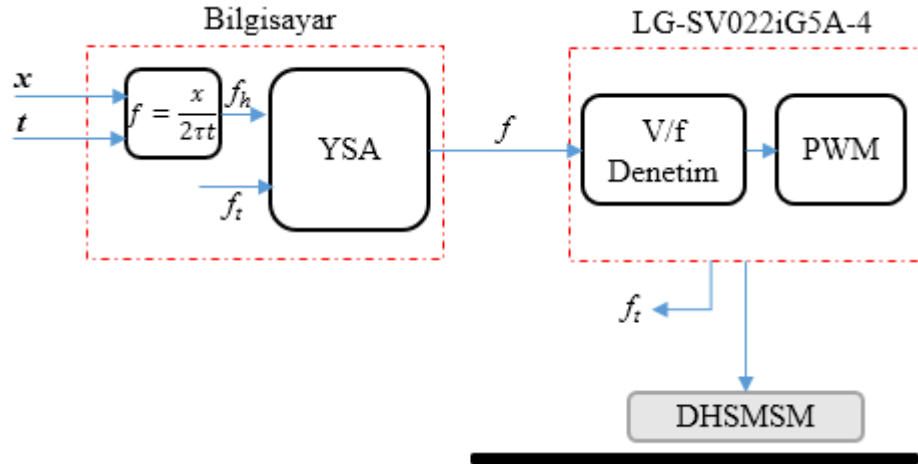
$$w_{32}^t = w_{32}^{(t-1)} + \eta \cdot \delta_{O_3} \cdot O_3 \quad (61)$$

$$w_{42}^t = w_{42}^{(t-1)} + \eta \cdot \delta_{O_4} \cdot O_4 \quad (62)$$

Böylece ağırlıklar güncellenmiş olur. Beklenen E , hata değerine ulaşana kadar bu işlemler tekrarlanır.

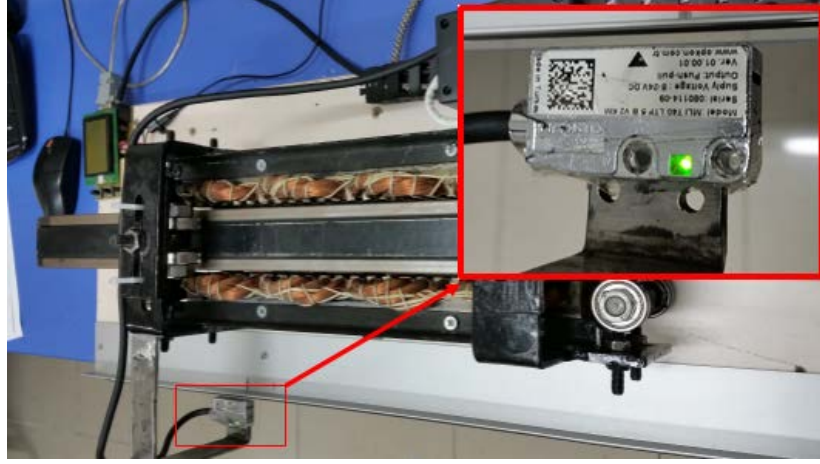
4.7.1.1. Deneylerden Elde Edilen Sonuçlar

Deneysel çalışmanın blok devre şeması Şekil 82'deki gibi gerçekleştirilmiştir.



Şekil 82. Deneysel çalışmanın blok devre şeması

YSA eğitimi, yol için 300-450 cm ve zaman için 5-10 sn. aralıklarında gerçekleştirilmiştir. YSA her deney öncesi belirtilen aralıklarda eğitilir. Yol ve zaman bilgileri bu aralıklarda seçilir. Motorun bu bilgilere göre hareketini tamamlaması beklenir. Sisteme girilen yol ve zaman bilgilerinin denetimi (teyit) motor üzerinde bulunan Manyetik Lineer Enkoderden (MLE) alınan bilgilerle karşılaştırılır. MLE Şekil 83'de, bu enkoderden alınan bilgileri almak için kullanılan pic devresi Şekil 84'de gösterilmiştir.



Şekil 83. Manyetik lineer enkoder



Şekil 84. MLE'den alınan verileri okuyan devre

Aşağıda yapılan testler ve sonuçları verilmiştir.

1. Test: Yapılan deneyde motorun 300 cm yi 5 sn'de alması istenmiştir. Verilen bilgilere göre motorun ortalama hızı (denklem 10) 60 cm/sn. olması gerekmektedir. Bu bilgilere göre YSA eğitimi gerçekleştirilir ve sistem test edilir. Motor üzerindeki MLE'den alınan bilgilerin ekran görüntüsü Şekil 85'de verilmiştir.

<< RESULTS >>	<< SONUCLAR >>
Motion Time (sec)	H. S. = 00:05:07
Moving Distance (path)	FARK = -292.712 cm
	MESAFE = 342.712 cm
Speed (cm/sec)	HIZ = 58.425 cm/Sn

Şekil 85. Manyetik lineer enkoderden alınan 1. test sonuçları

2. Test: Yapılan deneyde motorun 300 cm yi 6 sn'de alması istenmiştir. Verilen bilgilere göre motorun ortalama hızı (denklem 10) 50 cm/sn. olması gerekmektedir. Bu bilgilere göre YSA eğitimi gerçekleştirilir ve sistem test edilir. Motor üzerindeki MLE'den alınan bilgilerin ekran görüntüsü Şekil 86'da verilmiştir.

<< RESULTS >>	<< SONUCLAR >>
Motion Time (sec)	H. S. = 00:06:06
Moving Distance (path)	FARK = -289.848 cm MESAFE= 343.848 cm
Speed (cm/sec)	HIZ = 48.308 cm/Sn

Şekil 86. Manyetik lineer enkoderden alınan 2. test sonuçları

Dikkat edilirse her iki testte de motorun 300 cm yol alması istenmiştir. Motor birinci test de yaklaşık 7 cm, ikinci test de yaklaşık 10 cm hata yaptığı görülmektedir. Bu durumun motorun sürtünme kayıplarından meydana geldiği söylenebilir. Bu durumdan emin olmak için deneylere devam edilmiş ve 3, 4, 5 ve 6. testler gerçekleştirilmiştir. Ardından uç etkileri ve sürtünme kayıpları dikkate alınıp 7. Test gerçekleştirilmiştir. Yani sisteme sürtünme kaybı eklenerek yeni bir test yapılmıştır.

7. Test: Yapılan deneyde motorun 350 cm yi 7 sn'de alması istenmiştir. Verilen bilgilere göre motorun ortalama hızı (denklem 10) 50 cm/sn. olması gerekmektedir. Bu bilgilere göre YSA eğitimi gerçekleştirilir. Motor üzerindeki MLE'den alınan bilgilerin ekran görüntüsü Şekil 87'de verilmiştir.

<< RESULTS >>	<< SONUCLAR >>
Motion Time (sec)	H. S. = 00:07:26
Moving Distance (path)	FARK = -350.584 cm MESAFE= 402.584 cm
Speed (cm/sec)	HIZ = 48.692 cm/Sn

Şekil 87. Manyetik lineer enkoderden alınan 7. test sonuçları

Şekil 83'deki bilgiler denetim edildiğinde motor konum bilgisinde 0.584 cm fark olduğu ve motorun yaklaşık % 0.17 hata yaptığı görülmüştür. Hareket zamanı yaklaşık 0.26 sn. farkla % 3.7 hata yaptığı tespit edilmiştir. Motor hızı ortalama % 0.27 hata ile 48.692 cm/ sn olarak bulunmuştur.

Benzer şekilde farklı deneyler gerçekleştirilmiş ve elde edilen yol, hız verileri Tablo 11’de toplu olarak verilmiştir.

Tablo 11. Yol ve zaman denetimi sonuçları

Testler	İstenen Yol (cm)	Elde Edilen Yol (cm)	İstenen Hız (cm/sn)	Elde Edilen Hız (cm/sn)
1.Test	300	292.7	60	57.79
2.Test	300	289.8	60	59.15
3.Test	300	290	50	47.85
4.Test	300	291	50	49.19
5.Test	300	287	43	39.63
6.Test	300	290	37.5	34.81
7.Test	350	350.5	50	49.57
8.Test	350	352	50	49.44
9.Test	350	353	50	49.24

Grafiklerde görüldüğü gibi yol ve hız grafikleri arasında küçük farklar meydana gelmiştir. Bu durumun iki nedenden kaynaklandığı düşünülmektedir. Birincisi motorda meydana gelen uç etkileri ve sürtünme kayıplarının sistem parametrelerine sağlıklı olarak eklenmemesidir. İkincisi MLE’den alınan bilgilerin sistemde işlerken yavaşlamasıdır.

4.7.2. Çift Yanlı Doğrusal Hareketli Sürekli Mıknatıslı Senkron Motorun Yapay Sinir Ağları ile Hız-Zaman Denetimi

Bu çalışmada YSA ya giriş olarak istenen hız ve anlık hız bilgileri verilmiştir. Eğitim seti bu bilgilere göre düzenlenerek tekrar oluşturulmuştur. YSA modeli diğer çalışmayla aynıdır. Yani çok katmanlı ileri beslemeli YSA modeli, öğrenme modeli hata geri yayılım algoritması ve danışmanlı öğrenme kullanılmıştır. Bu denetimde de yine sürücü frekansı YSA ile değiştirilerek hız denetimi gerçekleştirilmiştir. YSA için örnek eğitim seti Tablo 12’de olduğu gibidir. YSA rastgele 50 örnek belirleyerek eğitimi gerçekleştirir.

Tablo 12. Örnek YSA eğitim seti (Hız)

$v_{istenen}$ (cm/sn)	$v_{anlık}$ (cm/s)	$\Delta f(\text{çıkış})$
30	25	+1
20	15	+1
45	50	- 1
35	42	- 1
25	20	+1
.	.	.
.	.	.

Tablo 12’den anlaşılacağı gibi YSA ya giriş olarak bir hız değeri girilir. Bu hız değerine karşılık olması gereken frekans değeri hesaplanır. Program çalışırken motor sürücüsünden anlık frekans değerleri alarak, bu frekansa karşılık gelen hız değeri hesaplanır ve YSA ya verilir. YSA, hesaplanan hızın olması gerekenden düşük olması durumunda frekansı azaltacak değilse frekansı arttıracaktır.

4.7.2.1. Deneylerden Elde Edilen Sonuçlar

Bu çalışmayla ilgili yapılan testler motorun yüklü ve yüksüz durumları için gerçekleştirilmiştir. Elde edilen sonuçlar grafiklerle gösterilmiştir. Çalışma aşamaları aşağıda sırasıyla verilmiştir.

1. Aşama: Şekil 88’de verilen Ağ oluşturma ekranda YSA’nın Ara Katman Sayısı, Maksimum Hatası, Alfa ve Momentum Katsayıları belirlenir.

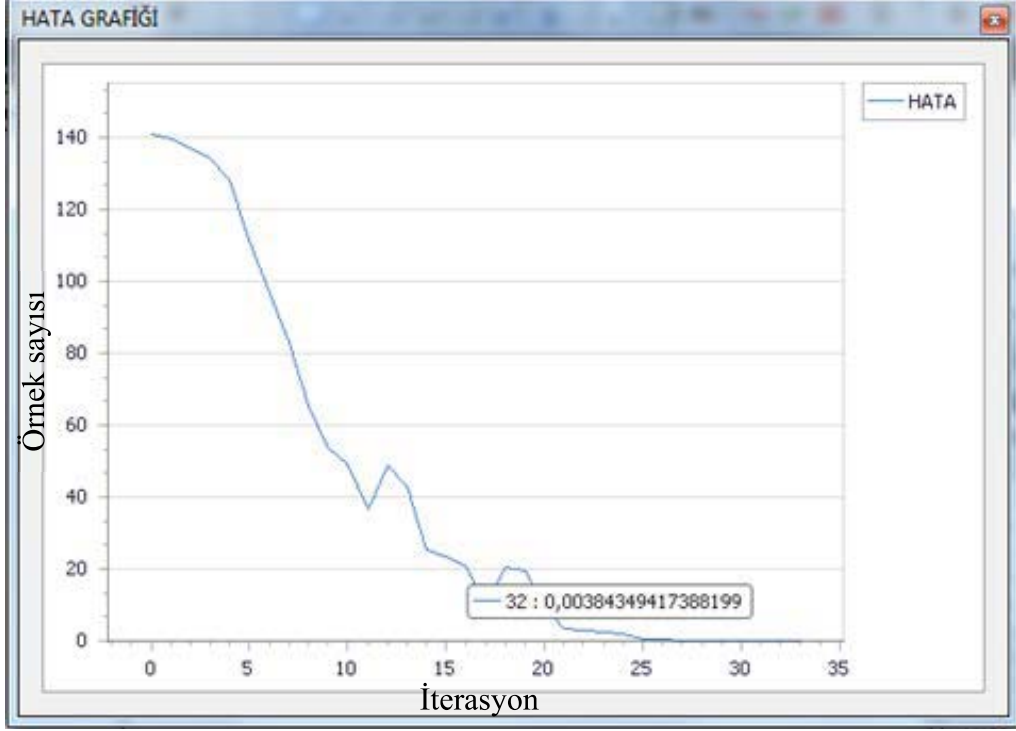
Şekil 88. YSA için ağ oluşturma ekranı

2. Aşama: Eğitim setinin oluşturulması için Şekil 88'deki 'OLUŞTUR' butonuna basılır ve Şekil 89'daki ekranda eğitim seti oluşturulur.

V girilen (cm/sn)	V makine (cm/sn)	YSA Çıkaş
45	64	-1
85	23	1
46	46	-1
71	13	1
35	42	-1
62	14	1
66	66	-1
78	16	1
15	65	-1
71	11	1
30	79	-1
75	25	1
54	74	-1
63	51	1
24	47	-1
54	36	1

Şekil 89. YSA eğitim seti oluşturma ekranı

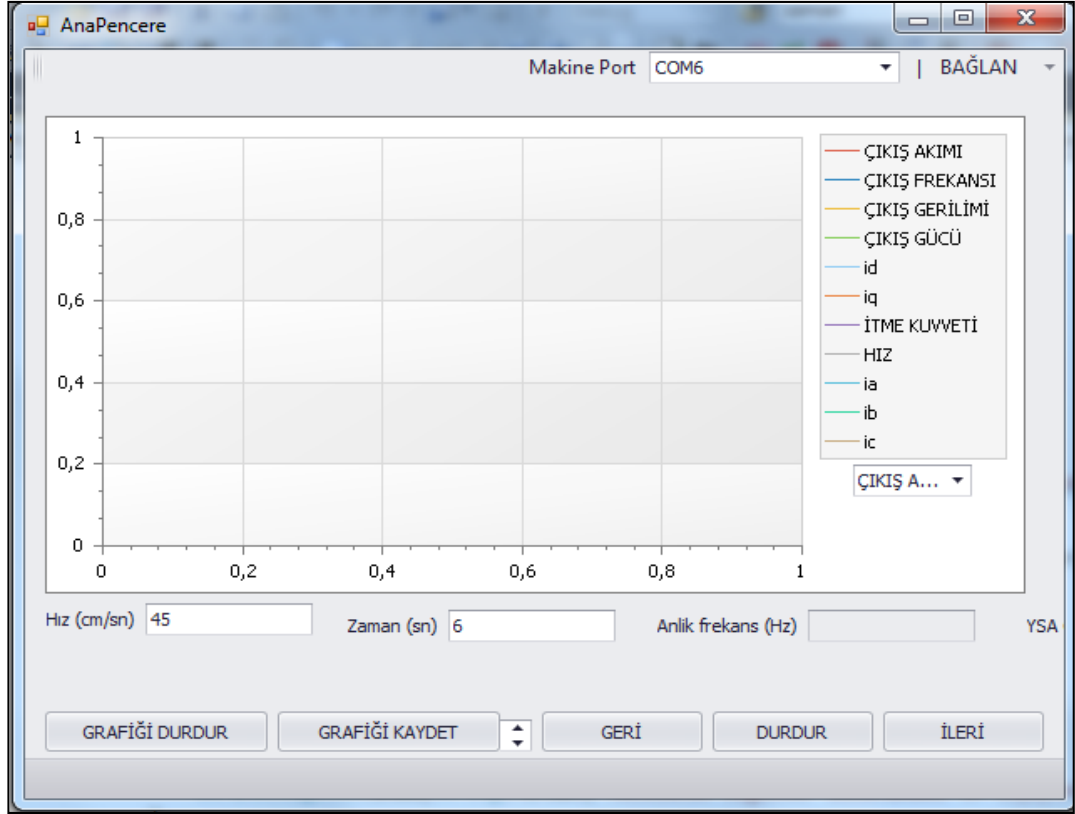
3. Aşama: Eğitimdeki hata grafiği Şekil 90'daki gibi oluşur. Burada her eğitimde farklı hata grafikleri oluşmaktadır. Hata grafiği YSA'nın ne kadar hata yapacağı ile ilgilidir ve Şekil 88'de maksimum hata 0.005 olarak belirlenmiştir. YSA'nın eğitimi bu hata değerinin altında tamamlaması gerekir.



Şekil 90. YSA eğitim hata grafiği

Şekil 90'daki ekran görüntüsünde YSA 32. iterasyonda maksimum hatanın altına düşerek eğitimi tamamlamıştır.

4. Aşama: Şekil 89'daki "TEST" butonuna basıldığında Şekil 91'deki ana pencere ekranı açılır ve testler gerçekleştirilir.



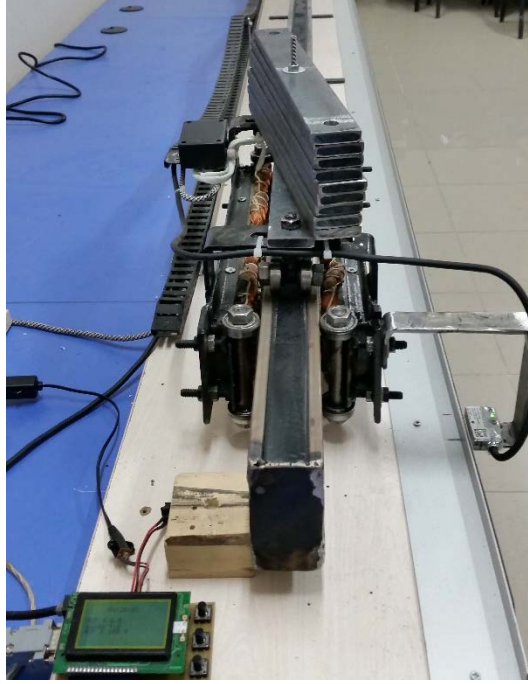
Şekil 91. Ana pencere ekranı

Testler: ÇYDHSMSM'nin yüklü ve yüksüz durumda yapılan deney sonuçlarını içermektedir. Motorun yüksüz ağırlığı 15.5 kg'dır. Motorun yüksüz durumu Şekil 92'deki gibidir.



Şekil 92. Motorun yüksüz durumu

Motorun yüklü hali Şekil de görülmektedir. Motora kendi ağırlığını haricinde 10 kg harici yük konulmuştur. Motorun yüklü durumu Şekil 93'deki gibidir.



Şekil 93. Motorun yüklü durumu

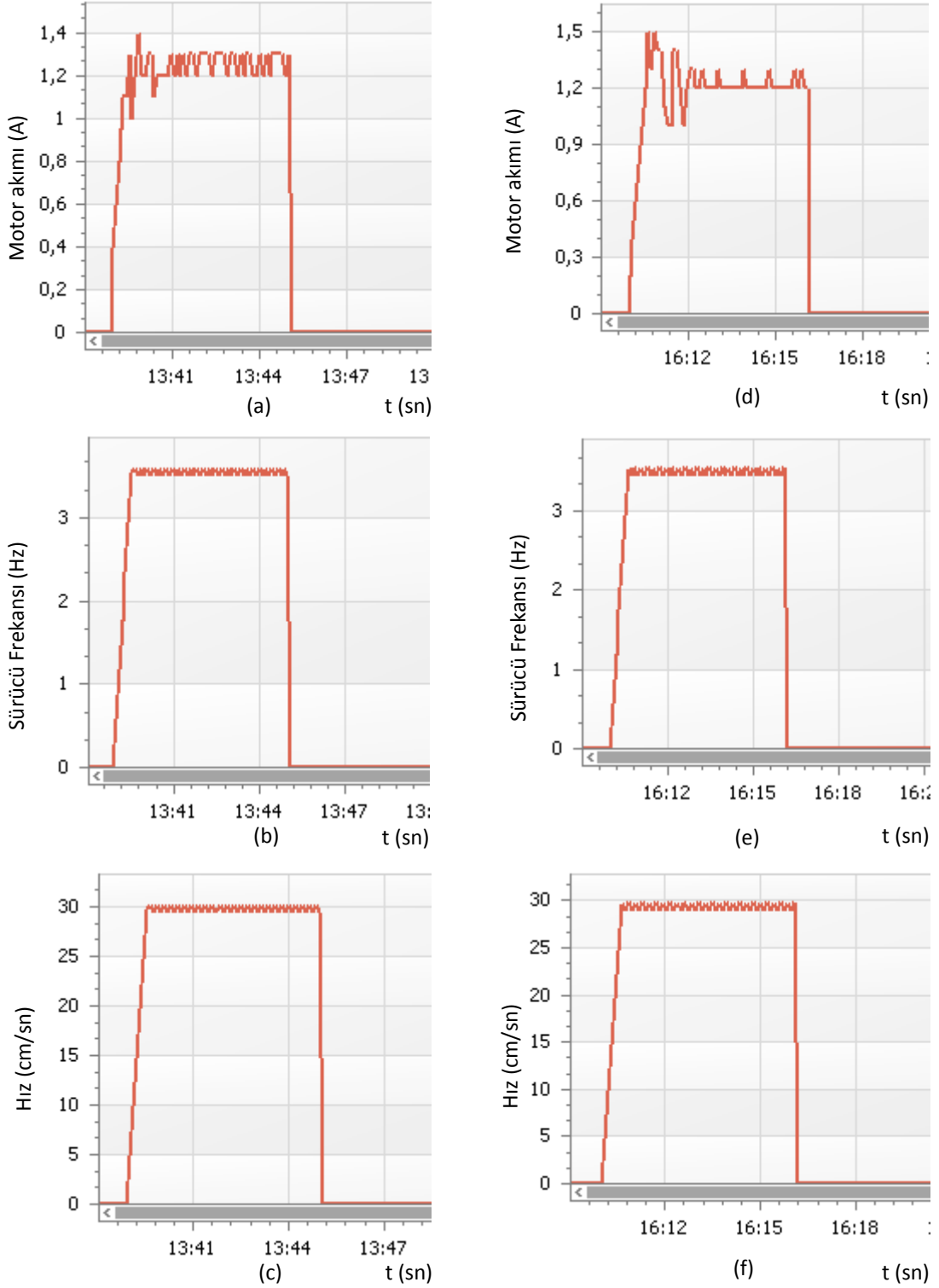
Deney sonuçlarında elde edilen mesafeler toplu olarak Tablo 13'de verilmiştir.

Tablo 13. Motorun yüksüz durumu için elde edilen yol bilgileri

<i>İstenen Hız (cm/sn)</i>	<i>İstenen Zaman (sn)</i>	<i>Beklenen Yol (cm)</i>	<i>Yüksüz Durumda Alınan Yol (cm)</i>	<i>Yüklü Durumda Alınan Yol (cm)</i>
30	6	180	180.6	176.6
30	8	240	238.7	236.2
30	10	300	298.2	296.7
40	8	320	319.2	317.2
50	6	300	298.2	296.4

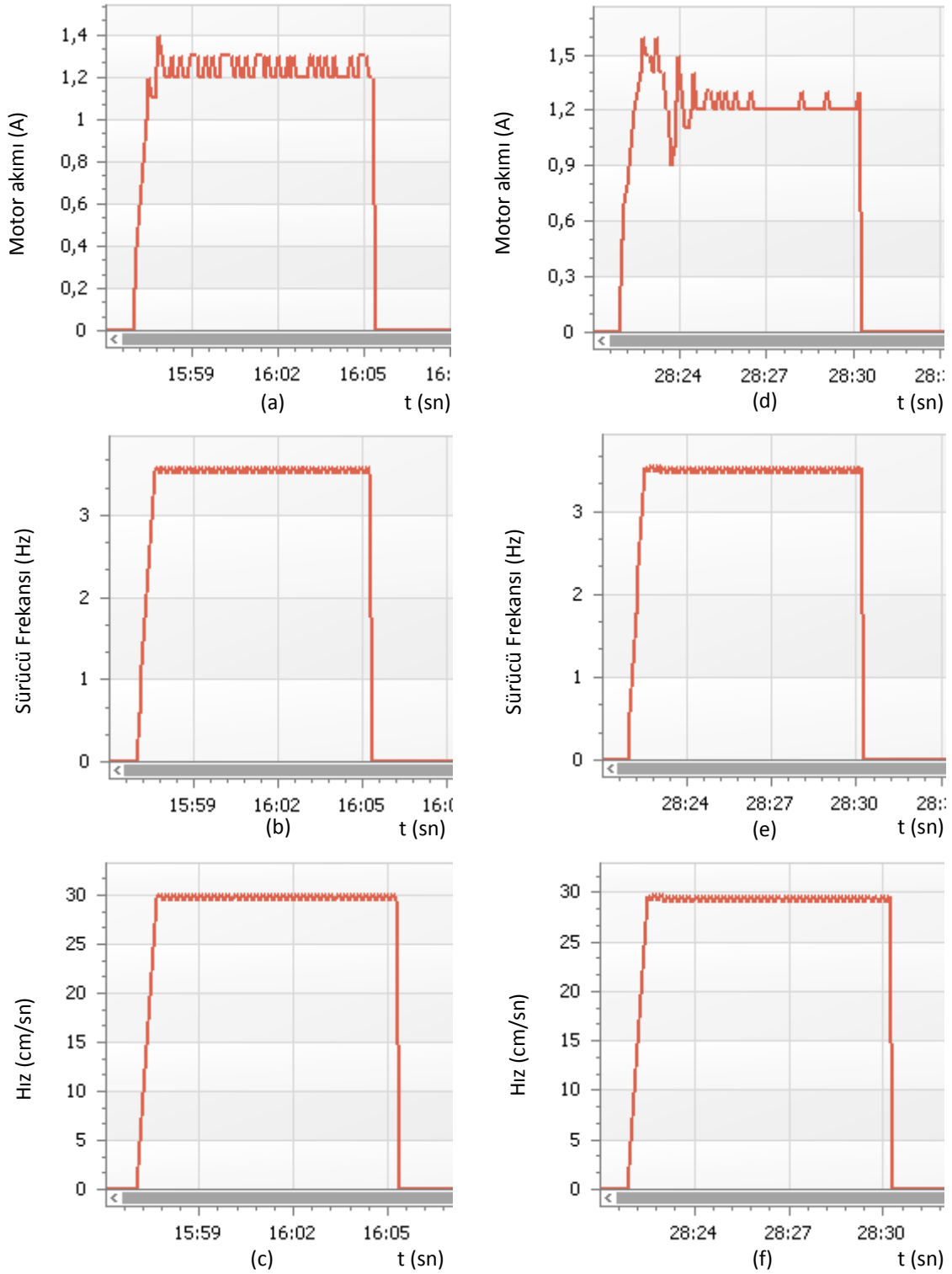
Bu çalışma toplam 5 deneyden oluşmaktadır. Yapılan her deney öncesi YSA eğitimi gerçekleştirilir. Deney sonunda elde edilen motor akımı, sürücü frekansı ve hız grafikleri verilmiştir. Grafiklerin kolay analizi için yüklü ve yüksüz durum grafikleri yan yana verilmiştir.

Birinci Deney: Motor yüklü ve yüksüz durumda iken 30 cm/sn hızla 6 sn boyunca gitmesi istenmiştir.



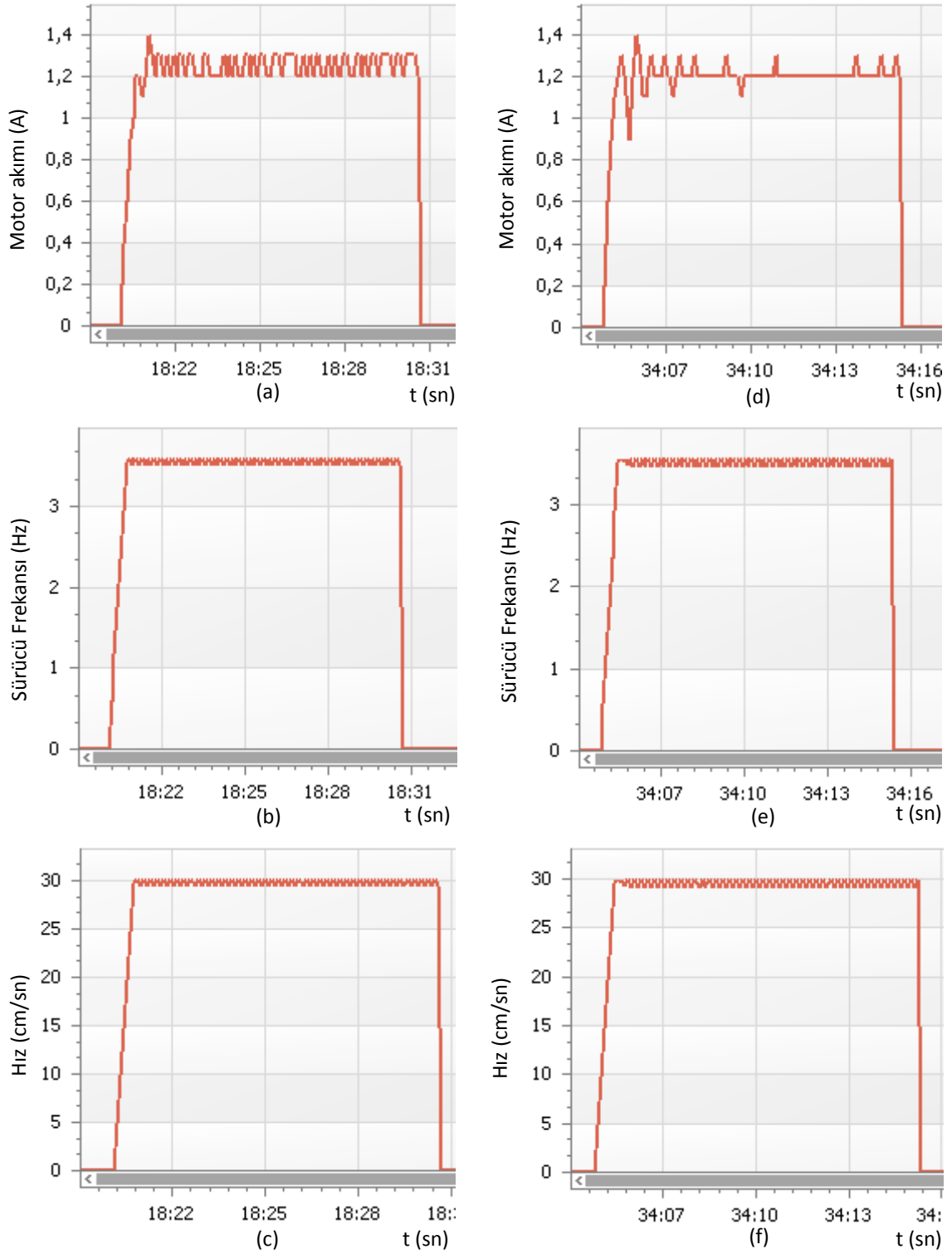
Şekil 94. Birinci deney sonuçları, (a), (b), (c) Motor yüksüz, (d), (e), (f) Motor yüklü

İkinci Deney: Motor yüklü ve yüksüz durumda iken 30 cm/sn hızla 8 sn boyunca gitmesi istenmiştir.



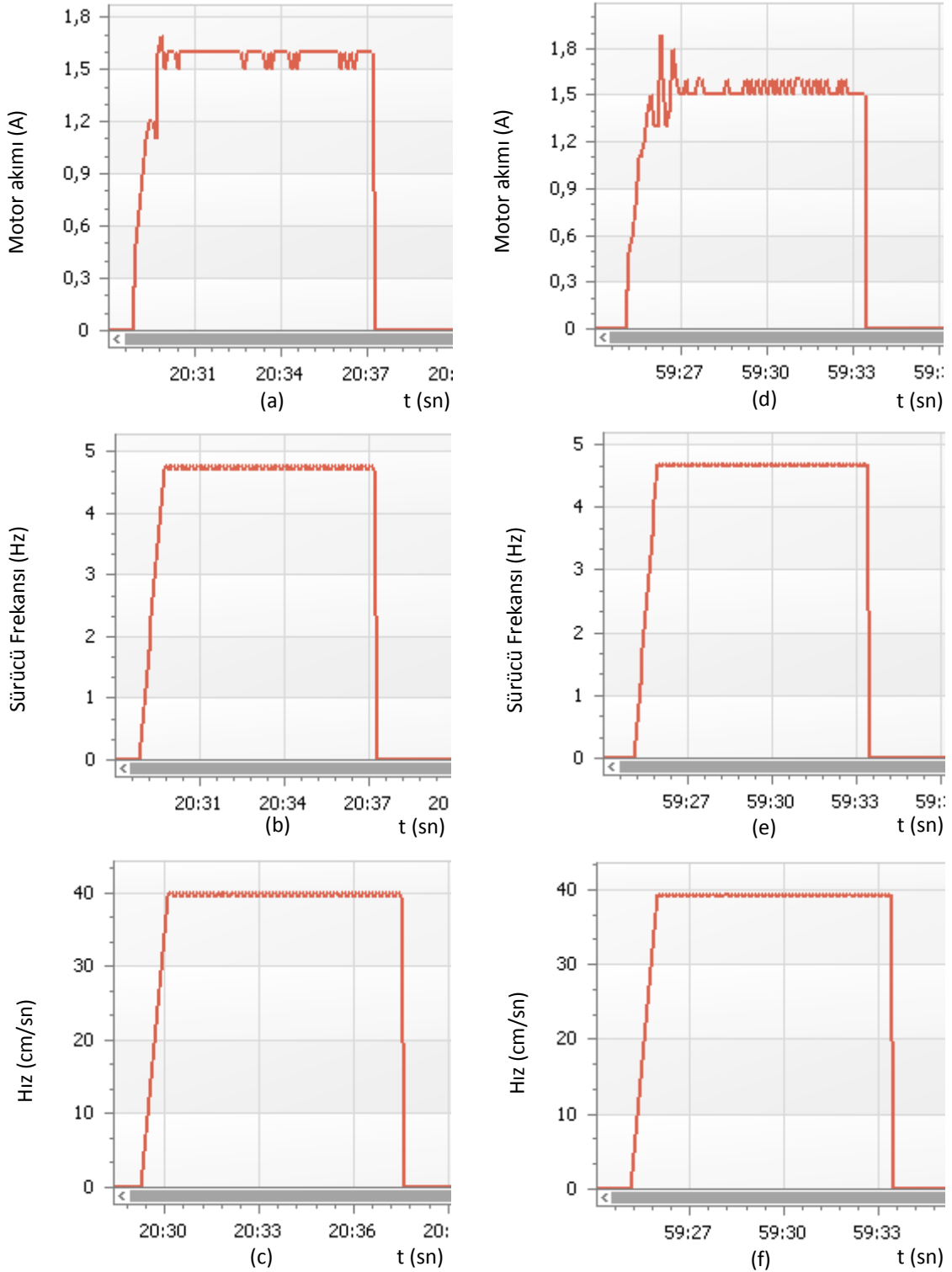
Şekil 95. İkinci deney sonuçları, (a), (b), (c) Motor yüksüz, (d), (e), (f) Motor yüklü

Üçüncü Deney: Motor yüklü ve yüksüz durumda iken 30 cm/sn hızla 10 sn boyunca gitmesi istenmiştir.



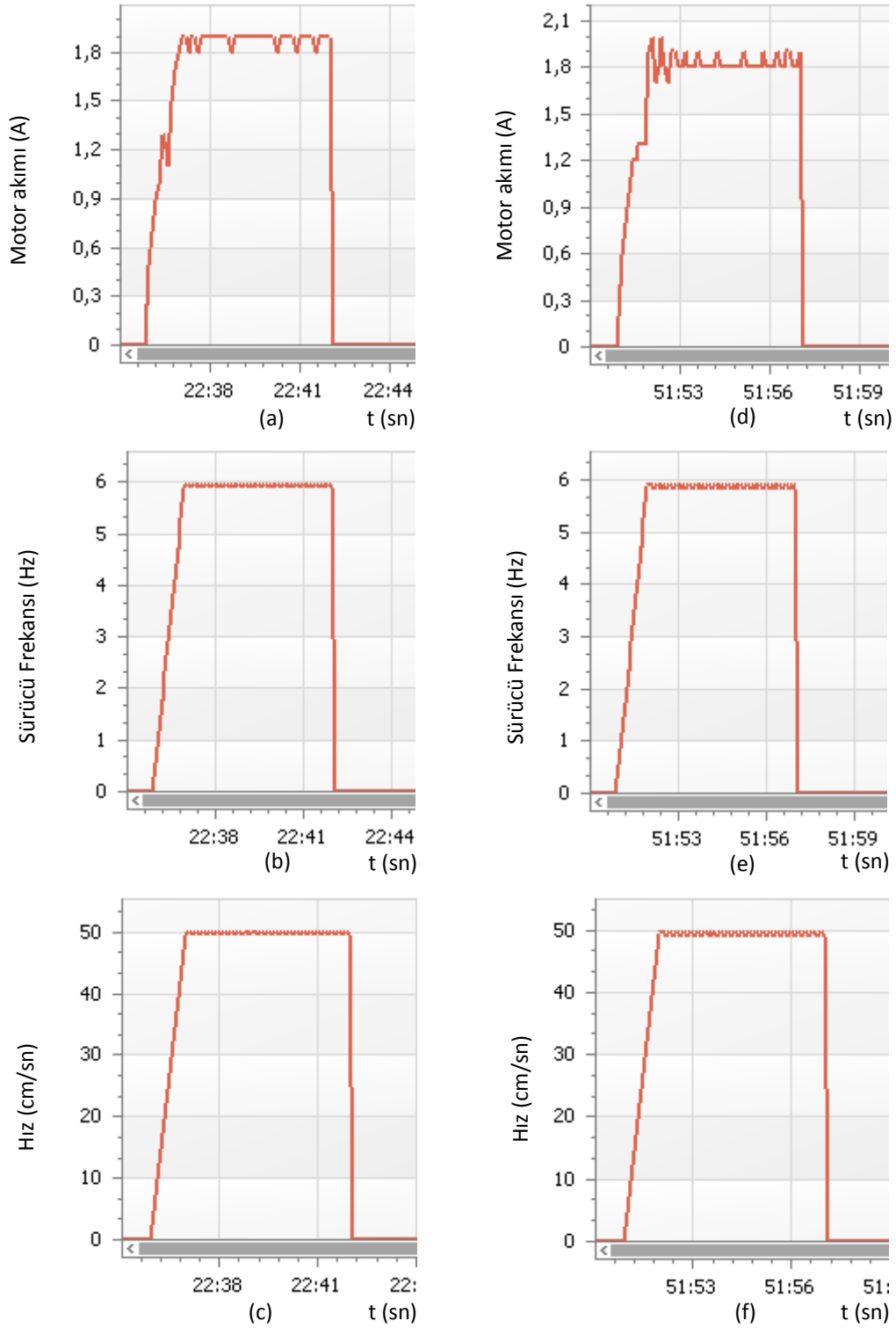
Şekil 96. Üçüncü deney sonuçları, (a), (b), (c) Motor yüksüz, (d), (e), (f) Motor yüklü

Dördüncü Deney: Motor yüklü ve yüksüz durumda iken 40 cm/sn hızla 8 sn boyunca gitmesi istenmiştir.



Şekil 97. Dördüncü deney sonuçları, (a), (b), (c) Motor yüksüz, (d), (e), (f) Motor yüklü

Beşinci Deney: Motor yüklü ve yüksüz durumda iken 50 cm/sn hızla 6 sn boyunca gitmesi istenmiştir.



Şekil 98. Beşinci deney sonuçları, (a), (b), (c) Motor yüksüz, (d), (e), (f) Motor yüklü

Verilen grafiklerde, program arayüzüne girilen bilgiler ve elde edilen grafikleri bulunmaktadır. Yapılan deneyler ray boyu dikkate alınarak gerçekleştirilmiştir. Örneğin 45 cm/sn hızla maksimum 10 sn yol alabilir. Çünkü ray boyu 480 cm dir.

Şekil 94'e bakıldığında, sırasıyla motorun yüksüz ve yüklü durumları için elde edilen grafikler görülmektedir. (a), (b), (c) grafikleri motor yüksüz iken elde edilmiştir. (d), (e), (f) grafikleri motor yüklü iken elde edilmiştir. Burada motorun 30 cm/sn hızla 6 sn süresince gitmesi istenmiştir. Yapılan deneylerin çıkış akımı grafikleri incelendiğinde motorun yüksüz durumda ortalama 1.2 A çektiği görülmektedir. Motor yüklü durumda iken çıkış akımının kalkınma sırasında 1.5 A'e kadar çıktığı ve motor hareketi sağlandıktan sonra tekrar 1.2 A'e düştüğü görülmektedir. Motorun yüklü ve yüksüz durumları için frekans grafikleri incelendiğinde, her iki durumda da frekansların yaklaşık eşit olduğu görülmektedir. Motorun hız grafikleri incelendiğinde, motor hızının yüksüz durumda 30.24 cm/sn ve 29.48 cm/sn aralığında olduğu ve istenen hıza ulaştığı görülmektedir. Motor yüklü durumda iken hız 29.82 cm/sn ve 29.06 cm/sn aralığında olduğu görülmektedir.

Bu durumda motor yüksüz iken ortalama hızı 29.86 cm/sn, yüklü iken 29.44 cm/sn'dir. Böylece ÇYDHSMSM istenen hıza yaklaşık % 0.5 hata ile ulaştığı anlaşılmaktadır.

Sırasıyla Şekil 95, 96, 97, 98 incelendiğinde benzer sonuçlara ulaşıldığı görülmektedir.

5. SONUÇLAR

Bu çalışmada, Çift Yanlı Doğrusal Hareketli Sürekli Mıknatıslı Senkron Motorun (ÇYDHSMSM), Yapay Sinir Ağları (YSA) ile hız denetimi incelendi. Deneysel çalışmalar için ÇYDHSMSM modeli hazırlandı. Ayrıca bu motorlara ait bilinen matematiksel ifadeler kullanılarak motor model benzetimi yapıldı. Skaler denetim yöntemiyle motor hız denetimi hem deneysel hem de benzetim modeliyle gerçekleştirildi. Bu motorlarda kullanılan sürekli mıknatıs ölçülerinin denetimdeki etkisi araştırıldı. YSA ile hız denetiminde değişken frekans denetimi yapabilen hazır bir sürücü kullanıldı. İki farklı hız denetimi yapıldı ve YSA için iki ayrı eğitim seti oluşturuldu. Deneylelerden elde edilen sonuçları aşağıdaki gibi sıralayabiliriz.

- Çalışmada modeli LG-SV022İG5A-4 olan motor sürücüsü kullanıldı. Bu sürücünün bilgisayarda hazırladığımız denetim programı ile haberleşmesi için sürücü haberleşme protokolü MODBUS-RTU kullanıldı. Hazırlanan program kullanılacak sürücülerin haberleşme protokolüne uyumlu hale getirildi. Böylece farklı model bir sürücü, programda yapılacak basit değişikliklerle hız denetiminde kullanılabilir hale getirilmiştir.
- Deneysel ve benzetim olarak yapılan çalışmada, bu motorlarda kullanılan sürekli mıknatıs adımının (τ_{pm}) kutup adımına (τ) yakın seçilmesi denetime olumlu katkılar sağlamıştır.
- Programa yol ve zaman girilerek yapılan YSA ile hız denetiminde, istenen ve elde edilen sonuçlar arasındaki farklılıkların motorun uç-kenar ve sürtünme etkilerinden kaynaklanmaktadır. Yapılan deneylerde bu parametrelerin sağlıklı olarak sisteme eklenmesi sonuçları olumlu yönde etkilemiştir.
- Programa hız ve zaman girilerek yapılan YSA hız denetiminde, sabit hızlarda kullanılan denklem kullanıldığı için istenen hız ile elde edilen hız arasında çok az farklılıklar olduğu görülmektedir. Yüklü ve yüksüz durumdaki hız farkı ise yüklü durumdaki sürtünme etkisinin artmasından kaynaklanmaktadır.
- Deneylerde frekans 1.5 hz e ulaştıktan sonra motorun hareket ettiği gözlenmiştir. Yani senkron frekans 1.5 hz dir. Motor 15 cm/sn lik hız limitinin altında güçlükle hareket etmiştir. Düşük hızlarda sürekli mıknatıslar denetimi zorlaştırmıştır.

- Bu motorlarda kullanılan hız denetim yönteminde, frekans değerindeki artma azalma miktarı uygun olmadığı durumlarda, motorun senkron frekansı yakalayamadığı için hareket etmediği ve olduğu yerde titreşim gösterdiği görülmüştür.
- Yapılan deneylerde motorun aldığı mesafenin beklenenden biraz farklı olduğu görülmüştür. Bunun en önemli nedeni, motorun durma esnasında bir mıknatıs boyu kadar ileride ya da geride kalmasıdır. Yani bu durum sürekli mıknatıs adımı ve motor hızı ile ilgilidir. Motor bir mıknatıs ileride veya geride durduğunda yaklaşık 4 cm fark oluşmuştur.
- Önerilen yöntemin, son yıllarda teknolojinin gelişimiyle birlikte kullanımı artan DHM'lerin hız denetimi için uygun bir yöntem olduğu düşünülmektedir.

6. ÖNERİLER

Bu çalışmada Çift Yanlı Doğrusal Hareketli Sürekli Mıknatıslı Senkron Motorun (ÇYDHSMSM) Yapay Sinir Ağları (YSA) ile hız denetimi gerçekleştirilmeye çalışılmıştır.

Bu anlamda;

Motor sürücüsü olarak piyasada hâlihazırda bulunabilen skaler ve vektör denetim yapabilen bir sürücü kullanılmıştır. Bu sürücü yerine hassasiyeti daha yüksek olan bir sürücünün kullanılması bu çalışmanın denetim başarısını arttıracığı söylenebilir.

YSA eğitim seti gerçek değerlerden oluşturulursa bu durumun motor denetim başarısını daha da arttıracığı söylenebilir.

Bu motorların tasarımı yapılırken kullanılan sürekli mıknatıs adımının kutup adımına yakın seçilmesi motor denetim başarımını arttırmaktadır. Bu nedenle mıknatıs parametrelerinin, denetim başarımını arttırmak için belirtilen değerlere uygun seçilmesi gerektiği düşünülmektedir.

Bu çalışmada sensörsüz denetim yapılmıştır. Çalışmada sensör kullanılması bu çalışmanın denetim başarımını arttırabilir.

YSA ile daha geniş aralıklı hız denetimi gerçekleştirilebilir ve bu motorların kullanım alanlarının genişletilebileceği söylenebilir.

7. KAYNAKLAR

1. Gireas, J. F. ve Piech, Z. J., Linear Synchronous Motors:Transportation and Automation Systems, Connecticut: CRC Press, 1999.
2. Boldea, I., Linear Electric Machines, Drives, and MAGLEVs, London: CRC Press, 2013.
3. Karaçar, E., Hidrolik ve Pnömatiğe Alternatif Çözüm:Doğrusal Motorlar, II. Hidrolik Pnömatik Kongresi Bildiriler Kitabı, İzmir, 2001.
4. Küçükali, M., Akyazı, Ö., Akpınar, A. S., Doğrusal Hareketli Asenkron Motor Sürücülü Mekanik Osilatör ve Evirici ile Kontrolü, Yüksek Lisans Tezi, Trabzon, Karadeniz Teknik Üniversitesi, Fen Bilimleri Enstitüsü,1-2, 2008.
5. Özkop, E., Doğrusal Hareketli Asenkron Motorun Bulanık Mantıkla Kontrolü, Yüksek Lisans Tezi, Trabzon, Karadeniz Teknik Üniversitesi, Fen Bilimleri Enstitüsü, 1-2, 2006.
6. <http://www.crbasic.info/high-tech-trains-automated-railways-metro-lines-light-railways.html>, RAİLring, 15 Haziran 2013. [Çevrimiçi]. [Erişildi: 10 10 2013].
7. Altun, H., Lineer asenkron motorun vektör kontrolü, Doktora Tezi, Elazığ: Fırat Üniversitesi, Fen Bilimleri Enstitüsü, 2001.
8. El-Halim, A.F.A., Ashraf, M.A., Al-Arabawy, I.F., Design Two Degree of Linear Motion X-Y Plane Machine Using Linear Induction Motor, Power System Conference, Aswan, 2008.
9. Budig, P. K., The Application of Linear Motors, IEEE Power Electronics and Motion Control Conference Proceedings, no. 3, pp. 1336-1341, 2000.
10. Fernandes, T. R., Short Primary Linear Drive Designed for Synchronous and Induction Operation Mode with On-Board Energy Storage, Darmstadt: Technischen Universität, 2012.
11. Ponomarev, P., Control of Permanent Magnet Linear Synchronous Motor in Motion Control Applications, Lappeenranta : Department of Electrical Engineering, Faculty of Technology, Lappeenranta University of Technology , Master's thesis, 2009.
12. Röhrig, C., Jochheim, A., Motion Control of Linear Synchronous Motors With Force Ripple Compensation Using Current Shaping, 15th Triennial World Congress, Barcelona, Spain, 2002.
13. Boldea, I., ve Nasar, S. A., Linear Electric Actuators and Generators, London: Cambridge University Press (U.K.), 1997.

14. Yapıcı, K. O. ve Cengiz, C., <http://oytunyapici.net/LM/kullanim.html>, İstanbul Teknik Üniversitesi, Mart 2006. [Çevrimiçi]. [Erişildi: 10 Ekim 2013].
15. Gürdal, O., Elektrik Makinalarının Tasarımı, İstanbul: Atlas Yayınları, 2001.
16. Akpınar, A. S., Doğrusal Hareketli Elektrik Makinaları, Trabzon: KTÜ, Yüksek Lisans Ders Notları, 2008.
17. Mnich, P., Hellinger, R., Linear Motor-Powered Transportation:History, Present Status, and Future Outlook, Proceedings of the IEEE, p. 11, November 2009.
18. Salman, M., Analysis, Design and Control Aspects of Linear Machines Using Co-simulation, Master Thesis, Stockholm, Department of Electrical Energy Conversion, School of Electrical Engineering, Royal Institute of Technology (KTH), syf.18-19, 2012.
19. Bose, B. K., Modern Power Electronics and AC Drives, Upper Saddle: Prentice-Hall, 2002.
20. Şahin, K., Asenkron Motorların Dinamik Yapay Sinir Ağları İle Hız Kontrolü, Elazığ: Fırat Üniversitesi, Fen Bilimleri Enstitüsü, 2006.
21. Garcia, G.O., Stephan, R. M. ve Watanabe, E. H., Comparing the indirect field-oriented control with a scalar method, IEEE Trans. Ind. Appl., cilt 2, no. 41, pp. 201-207, 1994.
22. Haitham, A., Atif, I., Jaroslaw, G., High Performance Control of AC Drives With Matlab/Simulink Models, Noida, India: Wiley, 2012.
23. Lu, H., Li, M., Xu, Y., Sensorless position and speed estimation with improved integrator for PMLSM drive, Information and Automation, ICIA '09. International Conference on , Ohio, 2009.
24. Itoh, J.-I., Nomura, N., Ohsawa, H., A comparison between V/f control and position-sensorless vector control for the permanent magnet synchronous motor, Power Conversion Conference, PCC-Osaka, 2002.
25. Perera, P., Blaabjerg, F., Pedersen, J. ve Thogersen, P., A sensorless, stable V/f control method for permanent-magnet synchronous motor drives, Industry Applications, IEEE Transactions on , cilt 39, no. 3, pp. 783-791, 2003.
26. Zhao, L., Ham, C.H.; Han, Q., Wu, Thomas X., Zheng, L., Sundaram, K.B., Kapat, J., Chow, L., Design of an optimal V/f control for a super high speed permanent magnet synchronous motor, Industrial Electronics Society, IECON 2004. 30th Annual Conference of IEEE , Busan, Korea, 2004.
27. Kavuran, G. ve Orhan, A., Open-loop speed control of permanent magnet synchronous motor fed by matrix converter, Electrical, Electronics and Computer Engineering (ELECO) , Bursa, 2010.

28. Jafari, S., Corzine K. ve Huang, J., Efficiency optimization of a sensorless V/f control method for PMSM, Sensorless Control for Electrical Drives (SLED), 2012 IEEE Symposium on, Milwaukee, WI., 2012.
29. Zhang, Y., Song, J., Song, S., Yan, M., Adaptive PID Speed Controller Based on RBF for Permanent Magnet Synchronous Motor System, International Conference on Intelligent Computation Technology and Automation, Changsha, China, 2010.
30. Gireas, J. F., Piech, Z. J. ve Tomzucuk, B. Z., Linear Synchronous Motors:Transportation and Automation Systems, Florida: CRC Press, 2013.
31. Junyou, Y., Jiefan, C., Guofeng, H., Direct Thrust Control of PMLSM Using SVM and Sliding Mode Variable Structure, J. Transactions of China Electrotechnical Society, cilt 22, no. 6, pp. 24-29, 2007.
32. Huacai, L., Yuetong, X., Weimin, Y., Zichen, C., Fuzzy PID Controller Design for a Permanent Magnet Linear Synchronous Motor Feeding System, J. Transactions of China Electrotechnical Society, cilt 22, no. 4, pp. 59-63, 2007.
33. Srinivasu, B., Prasad, P.V.N., Ramana Rao, M.V., Adaptive Controller Design for Permanent Magnet Linear Synchronous Motor Control System, New Delhi, India, Power Electronics, Drives and Energy Systems, PEDES '06, International Conference on , 2006.
34. Zare, J., Vector control of permanent magnet synchronous motor with surface magnet using artificial neural networks, 43rd International Universities Power Engineering Conference, UPEC , Padova, Italy, 2008.
35. Kumar, R., Gupta, R. ve Bansal, A. K., Novel topologies for identification and control of PMSM using artificial neural network, IET-UK International Conference on Information and Communication Technology in Electrical Sciences (ICTES), Chennai, India., 2007.
36. Kumar, R., Gupta, R.A., Bansal, Ajay Kr., Identification and Control of PMSM Using Artificial Neural Network, IEEE International Symposium on Industrial Electronics (ISIE), Vigo, Spain, 2007.
37. Shiek, S., Begam, B., Devi, S., Intlligent Control Of Permanent Magnet Linear Synchronous Motor, International Journal of Communications and Engineering, cilt 3, no. 4, pp. 66-73, 2012.
38. Guoqi, X. ve Dagui, H., A Neural Network Based Space Vector PWM for Permanent Magnet Synchronous Motor Drive, International Conference on Mechatronics and Automation (ICMA) , Harbin, Heilongjiang, China, 2007.
39. Maleknia, A.R., Rahimi, K., Zarchi, H.A., Soltani, J., Robust backstepping control of permanent magnet linear synchronous motor in extended region using Artificial Neural Network, IEEE International Conference on Industrial Technology (ICIT), Sichuan University, Chengdu, China, 2008.

40. Panah, P.G.; Shafiei, A.; Sharifian, R., An improved variable structure design for velocity control of a permanent magnet linear synchronous motor, Power Electronics Electrical Drives Automation and Motion (SPEEDAM), 2010 International Symposium on, Naples, Italy, 2010.
41. Arias, A., Romeral, L. ve Joyne, M.G., Fuzzy Logic Direct Torque Control, IEEE Int. Symp. Ind, cilt E, pp. 253-258, 2000.
42. Fukuda, T. ve Shibata, T., Theory and applications of neural networks for industrial control systems, IEEE Trans, Ind. Electro., pp. 472-491, 1992.
43. Elwer, A.S., Wahsh, S.A., Saleh, Kh., Badr, M.A., Analysis of permanent magnet synchronous motor using artificial neural network for electric vehicles, Information and Communication Technologies: From Theory to Applications, International Conference on, Omayyad Palace, Damascus, Syria, 2004.
44. Saleh, Kh., Badr, M.A., Elwer, A.S., Wahsh, S., Analysis of controlled permanent magnet synchronous motor using artificial neural network, Electrical Machines and Systems, ICEMS 2001. Proceedings of the Fifth International Conference on, Shenyang, China, 2001.
45. Zergaoui, A., Bennis, A., Identification And Control of an Asynchronous Machine Using Neural Networks, Proceedings of ICECS '99. The 6th IEEE International Conference on vol:2, 1999.
46. Von Zuben, F.J., Netto, M.L.A., Bim, E., Szajner, J., Adaptive vector control of a three-phase induction motor using neural networks, IEEE World Congress on Computational Intelligence, IEEE International Conference, cilt 6, pp. 3750-3755, 1994.
47. Heredia, J.R., Perez Hidalgo, F., Duran Paz, J.L., Control of Induction Motors By Artificial Neural Networks, Industrial Electronics, Proceedings of the IEEE International Symposium, cilt 2, pp. 792-794, 1999.
48. Miloudi, A., Miloud, Y., Draou, A., A neural network based speed control design strategy of an indirect vector controlled induction machine drive, Power Tech Conference Proceedings, IEEE Bologna, cilt 2, p. 4, 2003.
49. Wei, W., YuHua, W., ShiRong, W., A Speed Control System of Permanent Magnet Linear Synchronous Motor Using Neuron Adaptive Controller, Second International Conference on Intelligent Computation Technology and Automation, Washington, DC, USA, 2009.
50. Hamidia, F., Boucherit, M.S., Larabi, A., Bouhedda, M., Direct torque control with ANN speed controller based on Kalman filter for PMSM, Systems, Signals and Devices (SSD), 9th International Multi-Conference on, Chemnitz, Germany, 2012 .
51. Gaur, P., Singh, B., Mittal, A.P., Artificial Neural Network based Controller and Speed Estimation of Permanent Magnet Synchronous Motor, Power System Technology and IEEE Power India Conference, POWERCON 2008. Joint International Conference on, New Delhi, India, 2008.

52. Krishnan, R., Monajemy R. ve Tripathi, N., Neural control of high performance drives: an application to the PM synchronous motor drive, *Industrial Electronics, Control, and Instrumentation, Proceedings of the 1995 IEEE IECON 21st International Conference on*, Japan, 1995.
53. Chaoui, H., Gueaieb, W. ve Yagoub, M., Neural network based speed observer for interior permanent magnet synchronous motor drives, *Electrical Power & Energy Conference (EPEC)*, Canada, 2009.
54. Wang, J., Wang, H., Zhang, X., Wang, J., Neural network vector control of a permanent magnet synchronous motor drive, *Industrial Electronics and Applications, ICIEA 2008. 3rd IEEE Conference on*, Singapore, 2008.
55. Alicemary, K., Arundhati, B. ve Maridi, P., Modelling, Simulation and Nonlinear Control of Permanent Magnet Linear Synchronous Motor, *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, cilt 1, no. 6, pp. 555-562, 2012.
56. Balazovic, P., 3-Phase PM Synchronous Motor Torque Vector Control, *Czech Republic: Motorola, DRM018, Motorola Czech System Laboratories Roznov pod Radhostem*, 2003.
57. Akyazı Ö., Küçükali M., Akpınar, A.S., Sürekli Mıknatıslı Senkron Motorun Hız Denetiminin Farklı Bulanık Üyelik Fonksiyonları Kullanılarak Gerçekleştirilmesi, *Trabzon: Karadeniz Teknik Üniversitesi, Elektronik ve Bilgisayar Sempozyumu*, 2011.
58. Zheng, L. H., Jin, J. X., Investigation of HTS bulk magnet linear synchronous motors, *Chengdu: Proceeding of IEEE on Applied Superconductivity and Electromagnetic Devices*, pp. 17-21, 2009.
59. Gieras, J. F., *Linear Induction Drives*, Oxford: Clarendon Pres, 1994.
60. Jiefan, C., Chengyuan, W., Junyou, Y., Lifeng, L., Analysis of Direct Thrust Force Control for Permanent Magnet Linear Synchronous Motor., *Hangzhou: IEEE, Proceeding of the 5th World Congress on Intelligent Control and Automation*, June 15-19, 2004..
61. Gieras, J.F., Dawson G.E. ve Eastham A.R., Performance Calculation for Single Sided Linear Induction Motors with a Double-Layer Reaction Rail Under Constant Current Excitation, *IEEE Transactions on Magnetics*, cilt 22, no. 1, pp. 54-62, 1986.
62. Nonaka, S. ve Higuchi, T., Elements of Linear Induction Motor Design for Urban Transit, *IEEE Transactions on Magnetics*, cilt 23, no. 5, pp. 3002-3004, 1987.
63. Mori, Y., Torii, S. ve Ebihara, D., End Effect Analysis of Linear Induction Motor Based on the Wavelet Transform Technique, *IEEE Transactions on Magnetics*, cilt 35, no. 5, pp. 3739-3741, 1999.

64. Faiz J. ve Jafari, H., Accurate Modeling of Single-Sided Linear Induction Motor Considers End Effect and Equivalent Thickness, *IEEE Transactions on Magnetics*, cilt 36, no. 5, pp. 3785-3790, 2000.
65. Fujii, N., Kayasuga, T. ve Hoshi, T., Simple End Effect Compensator for Linear Induction Motor, *IEEE Transactions on Magnetics*, cilt 38, no. 5, pp. 3270-3272, 2002.
66. Kang, G., Kim, J. ve Nam, K., Parameter Estimation Scheme for Low-Speed Linear Induction Motors Having Different Leakage Inductances, *IEEE Transactions on Industrial Electronics*, cilt 50, no. 4, pp. 708-716, 2003.
67. Jamali, J., End Effect in Linear Induction and Rotating Electrical Machines, *IEEE Transactions on Energy Conversion*, cilt 18, no. 3, pp. 440-447, 2003.
68. Faiz, J. ve Jafari, H., Accurate Modeling of Single-Sided Linear Induction Motor Considers End Effect and Equivalent Thickness, *IEEE Transactions on Magnetics*, cilt 36, no. 5, pp. 3785-3790, 2000.
69. Mirsalim, M., Doroudi, A. ve Moghani, J.S., Obtaining the Operating Characteristics of Linear Induction Motors: A New Approach, *IEEE Transactions on Magnetics*, cilt 38, no. 2, pp. 1365-1370, 2002.
70. Amirkhani, H. ve Shulaie, A., Online Control of Thrust and Flux in Linear Induction Motors, *IEE Proc.-Electr. Power Appl.*, cilt 150, no. 5, pp. 515-520, 2003.
71. Nasar, S. A. ve Boldea, I., *LinearMotors: Theory, Design and Practical Applications*, New Jersey: Prentice-Hall,Inc., 1987.
72. Eastham, J.F. ve Williamson, S., Experiments On The Lateral Stabilisation And Levitation Of Linear Induction Motors, *IEEE*, pp. 470-473, 1973.
73. Yoshida, K. ve Nonaka S., Levitation Forces In Single-Sided Linear Induction Motors For High-Speed Ground Transport, *IEEE Transactions On Magnetics*, cilt 11, no. 6, pp. 1717-1719, 1975.
74. Kim, Y., J., Shin, P., S., Kang, D., H. ve Cho, Y., H., Design and Analysis of Electromagnetic System In A magnetically Levitated Vehicle, KOMAG-01, *IEEE Transactions On Magnetics*, cilt 28, no. 5, pp. 3321-3323, 1992.
75. Hirvonen, M., On the Analysis and Control of a Linear Synchronous Servomotor with a Flexible Load. Dissertation., Lappeenranta, Finland: Lappeenranta University of Technology, ISBN 952-214-320-0., 2006.
76. Demir, K., Doğrusal Hareketli Asenkron Motor Tasarımı ve Uygulaması, Seminer Notları, Trabzon, Karadeniz Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Mayıs 2012.
77. Richard, D. T., *Linear Synchronous Motor Design*, Acton: MagneMotion Inc. 20 Sudbury Road, 2005.

78. Özçira, S., Sürekli mıknatıslı senkron motorun kontrol yöntemleri ve endüstriyel uygulamaları, İstanbul: Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, 2007.
79. Rahman, M. A., ve Slemon, G. R., Promising Applications of NdFeB Magnets in Electrical Machines, IEEE, Cilt 1, Vol 2, MAG-26, no. 5, pp. 1712-1717, 1985.
80. Duru, H., Demiröz, R., Asansör Makinalarında Mıknatıs Uyarmalı Senkron Motor ve Doğrudan Tahrik Sisteminin Kullanımı, 1. EVK Sempozyumu, Kocaeli, 2005.
81. Trout, S. R., Understanding Permanent Magnets Meterials, an Attempt at Universal Magnetic Literacy, Germany: Magnequench International Inc., pp. 1-15, 2001.
82. Bal, G., Özel Elektrik Makinaları, Ankara: Seçkin Yayıncılık A.Ş., 2004.
83. Bizot, C., Brottes, J., Lungeanu, M., Poulsen, B., Séra, D. ve Sørensen, M. B., Sensorless Control for PMSM, Denmark: Power Electronics and Drives, Institute of Energy Technology, Aalborg University, 2003.
84. Gireas, J. F., Linear Induction Drives, Oxford: Clarendon Pres, 1994.
85. Giri, F., AC electric motor control, New Delhi, India: Wiley, 2013.
86. Sala-Perez, P., Galceran-Arellano, S., A sensorless stable V/f control method for a five-phase PMSM, Power Electronics and Applications (EPE), 15th European Conference on , 2013.
87. Agarlita, S.-C., Coman, C.-E., Andreescu, G.-D., Boldea, I., Stable V/f control system with controlled power factor angle for permanent magnet synchronous motor drives, Electric Power Applications, IET , cilt 7, no. 4, pp. 278-286, 2013.
88. Taşkafa, Ş., Asenkron Motorların Yapay Sinir Ağları ile Vektör Kontrolü, Elazığ: Fırat Üniversitesi, Fen Bilimleri Enstitüsü, 2006.
89. Rubaai, A. ve Kotaru, R., Online identification and control of a DC motor using learning adaptation of neural networks, IEEE Transactions on Industry Applications, cilt 36, no. 3, pp. 935-942, 2000.
90. Choi, H. S., Park, Y. H., Cho, Y. ve Lee, M., Global sliding-mode control: improved design for a brushless DC motor, IEEE Control Systems Magazine, cilt 21, no. 3, pp. 27-35, 2001.
91. Shieh, N. C., Tung, P. C. ve Lin, C. L., Robust output tracking control of a linear brushless DC motor with time-varying disturbances, IEE Proceedings - Electric Power Applications, cilt 149, no. 1, pp. 39-45, 2002.
92. Elmas, Ç., Yapay Sinir Ağları, Ankara: Seçkin yayıncılık, 2003.

93. Noğay, H. S., Üç kafesli asenkron motorlarda toplam harmonik distorsiyonunun yapay sinir ağları yöntemi kullanılarak saptanması, Doktora Tezi, İstanbul: Marmara Üniversitesi, Fen Bilimleri Enstitüsü, 2008.
94. Öztemel, E., Yapay Sinir Ağları, İstanbul: Papatya Yayıncılık, 2003.
95. Fausett, L., Fundamentals of Neural Networks : Architectures, USA: Algorithms and Applications, 1995.
96. Siddique, N., Adeli, H., Computational intelligence : synergies of fuzzy logic, neural networks, and evolutionary computing, New Delhi, India: Wiley, 2013.
97. Merkezi sinir sistemi yapisi ve isleyisi, 29 6 2011. [Çevrimiçi]. Available: <http://www.estanbul.com/merkezi-sinir-sistemi-yapisi-ve-isleyisi-188454.html> #. U1gvjfl_sT8. [Erişildi: 24 4 2014].
98. Freeman, A. J., Skapura, M. D., Neural Networks: Algorithms, Applications, and Programming Techniques, California, USA: Addison – Wesley Publishing Company, 1991.
99. Ham, M., Kostanic, I., Principles of Neurocomputing for Science and Engineering, New York, USA: Mcgraw – Hill Companies, 2001.
100. Abdi, H., Valentin, D., Edelman, B., Neural Networks Series :Quantitative Applications in the Social Sciences, London: Sage Publications, 1999.
101. <http://ls.inverterdrive.com/file/LS-Starvert-iG5A-Manual>, 28 04 2014. [Çevrimiçi]. [Erişildi: 28 4 2014].
102. Erdal, H., Doğan, B., Taşkın, S., DC Motor Parametrelerinin Bilgisayar Tabanlı Belirlenmesi, Simülasyonu ve Parametre İyileştirilmesi, 5. Uluslararası İleri Teknolojiler Sempozyumu (IATS'09), Karabük, Türkiye, 13-15 Mayıs 2009.
103. Plantic, Ž., Štumberger, G., Determining parameters of a three-phase permanent magnet synchronous machine using controlled single-phase voltage source, Przeglad Electrotechniczny, pp. 137-140, 2011.
104. Kavuran, G., Orhan, A., Open-loop speed control of permanent magnet synchronous motor fed by matrix converter, Electrical, Electronics and Computer Engineering (ELECO), National Conference on, pp. 334 - 338, 2010.
105. Bian, C., Ren, S., Ma, L., Study on direct torque control of super high-speed PMSM, Proceedings of the IEEE international conference on automation and logistice, Jian,China, 2007.
106. Tze-Fun, C. ve Keli, S., Applied Intelligent Control of Induction Motor Drives, Noida, India: IEEE Press, 2001.
107. Jacek, N. Z., Introduction To Artificial Neural Systems, St. Paul: West Publishing Company, 1992.

108. Jacek, N. Z., Introduction To Artificial Neural Systems, St. Paul: West Publishing Company, 1992.

8. EKLER

Ana Pencere Programı

```
using System;
using System.Collections.Generic;
using System.IO.Ports;
using DevExpress.XtraEditors.Repository;
using Modbus.Device;
using DevExpress.XtraEditors;
using System.Windows.Forms;
using System.Globalization;
using DevExpress.XtraCharts;
using System.Drawing;
using System.Drawing.Imaging;
using System.IO;
using AForge.Neuro;

namespace DHM
{
    public partial class AnaPencere : XtraForm
    {
        SerialPort sp = new SerialPort();

        IModbusSerialMaster master;
        const byte slaveId = 1;
        Series[] degerler = new Series[11];
        SwiftPlotDiagram diagram;
        ActivationNetwork ag;

        bool hareketbasladi = false;

        double Vgirilen = 0;
        double Vmakine = 0;
        double Veski = 0;
        double yol = 0;
        double zaman = 0;
        double Mmakine = 15.5;

        double tzaman = 0;
        double tinterval = 0.1;
        double mesafe = 0;

        public AnaPencere(ActivationNetwork ag)
        {
            InitializeComponent();
            Form.CheckForIllegalCrossThreadCalls = false;

            Vgirilen = Convert.ToDouble(textEdit1.Text);
            zaman = Convert.ToDouble(textEdit2.Text);

            this.ag = ag;

            for (int i = 0; i < degerler.Length; i++)
            {
                degerler[i] = new Series();
            }
            degerler[0].LegendText = "ÇIKIŞ AKIMI";
            degerler[1].LegendText = "ÇIKIŞ FREKANSI";
        }
    }
}
```

```

degerler[2].LegendText = "ÇIKIŞ GERİLİMİ";
degerler[3].LegendText = "ÇIKIŞ GÜCÜ";
degerler[4].LegendText = "id";
degerler[5].LegendText = "iq";
degerler[6].LegendText = "İTME KUVVETİ";
degerler[7].LegendText = "HIZ";
degerler[8].LegendText = "ia";
degerler[9].LegendText = "ib";
degerler[10].LegendText = "ic";

for (int i = 0; i < degerler.Length; i++)
{
    checkedComboBoxEdit1.Properties.Items.Add(degerler[i].LegendText);
    degerler[i].ChangeView(ViewType.SwiftPlot);
    checkedComboBoxEdit1.Properties.Items[i].CheckState =
CheckState.Checked;
    chartControl1.Series.Add(degerler[i]);

    ((SwiftPlotSeriesView)chartControl1.Series[i].View).LineStyle.Thickness = 2;
}

timer1.Interval = (int)(tinterval * 1000);
}

private void AnaPencere_Load(object sender, EventArgs e)
{
    string[] portList = SerialPort.GetPortNames();
    if (portList.Length != 0)
    {
        for (int i = 0; i < portList.Length; i++)
        {
            repositoryItemComboBox1.Items.Add(portList[i]);
            repositoryItemComboBox2.Items.Add(portList[i]);

            if(portList.Length>=0)
                barEditItem1.EditValue = portList[0];

            if(portList.Length>=0)
                barEditItem2.EditValue = portList[1];
        }
    }

    diagram = chartControl1.Diagram as SwiftPlotDiagram;
    diagram.EnableAxisXZooming = true;
    diagram.EnableAxisXScrolling = true;
    diagram.AxisX.CrosshairLabelVisibility =
DevExpress.Utils.DefaultBoolean.Default;
    diagram.AxisX.DateTimeGridAlignment =
DevExpress.XtraCharts.DateTimeMeasurementUnit.Millisecond;
    diagram.AxisX.DateTimeMeasureUnit =
DevExpress.XtraCharts.DateTimeMeasurementUnit.Millisecond;
    diagram.AxisX.DateTimeOptions.Format =
DevExpress.XtraCharts.DateTimeFormat.Custom;
    diagram.AxisX.DateTimeOptions.FormatString = "mm:ss";
    diagram.AxisX.GridLines.Visible = true;
    diagram.AxisX.Range.ScrollingRange.SideMarginsEnabled = true;
    diagram.AxisX.Range.SideMarginsEnabled = true;
    diagram.AxisX.Title.Text = "Time";
    diagram.AxisX.VisibleInPanelsSerializable = "-1";
}

```



```

        diagram.AxisY.CrosshairLabelVisibility =
DevExpress.Utils.DefaultBoolean.Default;
        diagram.AxisY.Interlaced = true;
        diagram.AxisY.Range.AlwaysShowZeroLevel = false;
        diagram.AxisY.Range.ScrollingRange.SideMarginsEnabled = true;
        diagram.AxisY.Range.SideMarginsEnabled = true;
        diagram.AxisY.Title.Text = "Values";
        diagram.AxisY.VisibleInPanesSerializable = "-1";
    }

    private void timer1_Tick(object sender, EventArgs e)
    {
        DateTime date = DateTime.Now;
        DateTime minDate = date.AddSeconds(-30);

        ushort startAddress = Convert.ToUInt16(int.Parse("0008",
NumberStyles.HexNumber));
        ushort numRegisters = 5;
        ushort[] registers = new ushort[numRegisters];

        //read registers
        try
        {
            registers = master.ReadHoldingRegisters(slaveId, startAddress,
numRegisters);

            double i = Convert.ToDouble(registers[0]) / 10.0;
            double f = Convert.ToDouble(registers[1]) / 100.0;
            double rpm = Convert.ToDouble(registers[2]) / 10.0;
            double p = Convert.ToDouble(registers[4]) / 10.0;
            double pole = 4.0;
            double Ld = 0.042;
            double Lq = 0.042;
            double R = 22.5;
            double Fd = 0;

            double W = 2.0 * Math.PI * f;

            double aci = W * tzaman;

            double im = i * Math.Sqrt(2.0) / 2.0;

            double ia = im * Math.Sin(aci);
            double ib = im * Math.Sin(aci - 2.0 * Math.PI / 3.0);
            double ic = im * Math.Sin(aci + 2.0 * Math.PI / 3.0);

            double id = 2.0 / 3.0 * (ia * Math.Cos(aci) + ib * Math.Cos(aci -
2.0 * Math.PI / 3.0) + ic * Math.Cos(aci + 2.0 * Math.PI / 3.0));
            double iq = 2.0 / 3.0 * (ia * Math.Sin(aci) + ib * Math.Sin(aci -
2.0 * Math.PI / 3.0) + ic * Math.Sin(aci + 2.0 * Math.PI / 3.0));

            double to = 4.2;

            double maki = 0.485;
            double fiakid = Ld * id + maki;

            double Fi = ((3.0 * Math.PI * maki * iq) / (2.0 * to)) * 100;

            double v = 2 * f * to;

            if (hareketbasladi && (zaman - tinterval > 0))

```

```

{
    zaman = zaman - tinterval;

    textEdit2.Text = zaman.ToString();
    textEdit3.Text = f.ToString();

    double[] giris = { Vgirilen, v };
    double cikis = ag.Compute(giris)[0];

    frekansartirazalt(cikis, f);

    textEdit4.Text = cikis+"";

    degerler[0].Points.Add(new SeriesPoint(date, i));
    degerler[1].Points.Add(new SeriesPoint(date, f));
    degerler[2].Points.Add(new SeriesPoint(date, rpm));
    degerler[3].Points.Add(new SeriesPoint(date, p));
    degerler[4].Points.Add(new SeriesPoint(date, id));
    degerler[5].Points.Add(new SeriesPoint(date, iq));
    degerler[6].Points.Add(new SeriesPoint(date, Fi));
    degerler[7].Points.Add(new SeriesPoint(date, v));
    degerler[8].Points.Add(new SeriesPoint(date, ia));
    degerler[9].Points.Add(new SeriesPoint(date, ib));
    degerler[10].Points.Add(new SeriesPoint(date, ic));

    if (zaman < 0.1)
        textEdit2.Text = "0";
}
else
{
    if (hareketbasladi)
        textEdit6.Text = mesafe + "";
    hareketbasladi = false;
    frekanssetle(0);

    degerler[0].Points.Add(new SeriesPoint(date, 0));
    degerler[1].Points.Add(new SeriesPoint(date, 0));
    degerler[2].Points.Add(new SeriesPoint(date, 0));
    degerler[3].Points.Add(new SeriesPoint(date, 0));
    degerler[4].Points.Add(new SeriesPoint(date, 0));
    degerler[5].Points.Add(new SeriesPoint(date, 0));
    degerler[6].Points.Add(new SeriesPoint(date, 0));
    degerler[7].Points.Add(new SeriesPoint(date, 0));
    degerler[8].Points.Add(new SeriesPoint(date, 0));
    degerler[9].Points.Add(new SeriesPoint(date, 0));
    degerler[10].Points.Add(new SeriesPoint(date, 0));
}
}
catch
{
    degerler[0].Points.Add(new SeriesPoint(date, 0));
    degerler[1].Points.Add(new SeriesPoint(date, 0));
    degerler[2].Points.Add(new SeriesPoint(date, 0));
    degerler[3].Points.Add(new SeriesPoint(date, 0));
    degerler[4].Points.Add(new SeriesPoint(date, 0));
    degerler[5].Points.Add(new SeriesPoint(date, 0));
    degerler[6].Points.Add(new SeriesPoint(date, 0));
    degerler[7].Points.Add(new SeriesPoint(date, 0));
}

```

```

degerler[8].Points.Add(new SeriesPoint(date, 0));
degerler[9].Points.Add(new SeriesPoint(date, 0));
degerler[10].Points.Add(new SeriesPoint(date, 0));
}

foreach (Series deger in degerler)
{
    if (deger.Points[0].DateTimeArgument < minDate)
        deger.Points.RemoveAt(0);
}

if (diagram != null && diagram.AxisX.DateTimeMeasureUnit ==
DateTimeMeasurementUnit.Millisecond)
    diagram.AxisX.Range.SetMinMaxValues(minDate, date);

tzaman+=0.1;
}

private void barButtonItem1_ItemClick(object sender,
DevExpress.XtraBars.ItemClickEventArgs e)
{
    if (barButtonItem1.Caption == "BAĞLAN")
    {
        sp.PortName = barEditItem1.EditValue.ToString();
        sp.BaudRate = 9600;
        sp.DataBits = 8;
        sp.Parity = Parity.None;
        sp.StopBits = StopBits.One;
        sp.Open();

        spsensor.PortName = barEditItem2.EditValue.ToString();
        spsensor.BaudRate = 2400;
        spsensor.DataBits = 8;
        spsensor.Parity = Parity.None;
        spsensor.StopBits = StopBits.One;
        spsensor.Open();

        master = ModbusSerialMaster.CreateRtu(sp);
        barButtonItem1.Caption = "DURDUR";
        timer1.Enabled = true;

        ushort startAddress = Convert.ToUInt16(int.Parse("0004",
NumberStyles.HexNumber));
        ushort numRegisters = 1;
        ushort[] registers = new ushort[numRegisters];

        // read registers
        try
        {
            registers = master.ReadHoldingRegisters(slaveId, startAddress,
numRegisters);

            for (int i = 0; i < numRegisters; i++)
                spinEdit1.EditValue = Convert.ToDouble(registers[i]/100);
        }
        catch
        {

```

```

        XtraMessageBox.Show("Okunacak Register Bulunamadı.", "HATA",
        MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}
else
{
    master.Dispose();
    sp.Close();
    barButtonItem1.Caption = "BAĞLAN";
    timer1.Enabled = false;
}
}

private void simpleButton1_Click(object sender, EventArgs e)
{
    hareketbasladi = true;
    Vgirilen = Convert.ToDouble(textEdit1.Text);
    zaman = Convert.ToDouble(textEdit2.Text);
    zaman -= 0.24;
    yol = Vgirilen * zaman;
    if (yol > 400)
    {
        zaman = 400 / Vgirilen;
        zaman = Math.Round(zaman, 1);
        MessageBox.Show("Girdiğiniz hız için "+yol+"cm mesafe gerekli fakat
        yolun uzunluğu 450cm olduğu için zaman değeri "+zaman+"sn olarak değiştirildi.");
    }
    //geri yön
    ushort startAddress = Convert.ToUInt16(int.Parse("0005",
    NumberStyles.HexNumber));

    // write register
    try
    {
        master.WriteSingleRegister(slaveId, startAddress, 4);
    }
    catch (Modbus.SlaveException se)
    {
        XtraMessageBox.Show(String.Format("Yazılacak Register Bulunamadı \n
        \n{0}\n\r", se), "HATA", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}

private void simpleButton2_Click(object sender, EventArgs e)
{
    hareketbasladi = false;
    //durdur
    ushort startAddress = Convert.ToUInt16(int.Parse("0005",
    NumberStyles.HexNumber));

    // write register
    try
    {
        master.WriteSingleRegister(slaveId, startAddress, 1);
    }
    catch (Modbus.SlaveException se)
    {
        XtraMessageBox.Show(String.Format("Yazılacak Register Bulunamadı \n
        \n{0}\n\r", se), "HATA", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}
}

```

```

private void simpleButton3_Click(object sender, EventArgs e)
{
    hareketbasladi = true;
    //spsensor.WriteLine("1");
    Vgirilen = Convert.ToDouble(textEdit1.Text);
    zaman = Convert.ToDouble(textEdit2.Text);
    zaman -= 0.24;
    yol = Vgirilen * zaman;
    if (yol > 400)
    {
        zaman = 400 / Vgirilen;
        zaman = Math.Round(zaman, 1);
        MessageBox.Show("Girdiğiniz hız için " + yol + "cm mesafe gerekli
fakat yolun uzunluğu 450cm olduğu için zaman değeri " + zaman + "sn olarak
değiştirildi.");
    }
    //ileri yön
    ushort startAddress = Convert.ToUInt16(int.Parse("0005",
NumberStyles.HexNumber));

    // write register
    try
    {
        master.WriteSingleRegister(slaveId, startAddress, 2);
    }
    catch (Modbus.SlaveException se)
    {
        XtraMessageBox.Show(String.Format("Yazılacak Register Bulunamadı \n
\n{0}\n\n", se), "HATA", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}

private void simpleButton4_Click(object sender, EventArgs e)
{
    if (timer1.Enabled)
    {
        timer1.Enabled = false;
        simpleButton4.Text = "GRAFİĞİ BAŞLAT";
    }
    else
    {
        timer1.Enabled = true;
        simpleButton4.Text = "GRAFİĞİ DURDUR";
    }
}

private void spinEdit1_EditValueChanged(object sender, EventArgs e)
{
    int miktar = Convert.ToInt32(spinEdit1.EditValue);
    ushort startAddress = Convert.ToUInt16(int.Parse("0004",
NumberStyles.HexNumber));
    try
    {
        master.WriteSingleRegister(slaveId, startAddress,
(ushort)(miktar*100));
    }
    catch (Modbus.SlaveException se)
    {
        XtraMessageBox.Show(String.Format("Yazılacak Register Bulunamadı \n
\n{0}\n\n", se), "HATA", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}

```

```

    }
}

private Image GetChartImage(ChartControl chart, ImageFormat format)
{
    // Create an image.
    Image image = null;

    // Create an image of the chart.
    using (MemoryStream s = new MemoryStream())
    {
        chart.ExportToImage(s, format);
        image = Image.FromStream(s);
    }

    // Return the image.
    return image;
}

private void simpleButton5_Click(object sender, EventArgs e)
{
    SaveFileDialog save = new SaveFileDialog();
    save.Filter = "Jpeg Image|*.jpg";
    save.Title = "Resmi kaydet";
    if (save.ShowDialog() == DialogResult.OK)
    {
        Image resim = GetChartImage(chartControl1, ImageFormat.Png);
        resim.Save(save.FileName, ImageFormat.Png);
    }
}

private void frekansartirazalt(double deger, double frekans)
{
    double miktar = (deger + frekans) * 100;
    ushort startAddress = Convert.ToUInt16(int.Parse("0004",
NumberStyles.HexNumber));
    try
    {
        master.WriteSingleRegister(slaveId, startAddress, (ushort)miktar);
    }
    catch (Modbus.SlaveException se)
    {
        XtraMessageBox.Show(String.Format("Yazılacak Register Bulunamadı \n
\n{0}\n\r", se), "HATA", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}

private void frekanssettle(double frekans)
{
    double miktar = frekans * 100;
    ushort startAddress = Convert.ToUInt16(int.Parse("0004",
NumberStyles.HexNumber));
    try
    {
        master.WriteSingleRegister(slaveId, startAddress, (ushort)miktar);
    }
    catch (Modbus.SlaveException se)
    {
        XtraMessageBox.Show(String.Format("Yazılacak Register Bulunamadı \n
\n{0}\n\r", se), "HATA", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}

```

```

    }
}

private string inBuffer = String.Empty;
private void spsensor_DataReceived(object sender,
SerialDataReceivedEventArgs e)
{
    try
    {
        string gelen = spsensor.ReadLine();
        string[] split = gelen.Split('=');
        mesafe = Convert.ToDouble(split[1].Replace('.', ','));

        textEdit5.Text = mesafe + "";
    }
    catch { }
}

private void checkedComboBoxEdit1_EditValueChanged(object sender, EventArgs
e)
{
    try
    {
        for (int i = 0; i < checkedComboBoxEdit1.Properties.Items.Count;
i++)
        {
            if (checkedComboBoxEdit1.Properties.Items[i].CheckState ==
CheckState.Unchecked)
                chartControl1.Series[i].Visible = false;
            else
                chartControl1.Series[i].Visible = true;
        }
    }
    catch { }
}
}
}

```

YSA Programı

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace DHM
{
    public class network
    {
        public double[] girisler; // network genel girişler --> girdi katmanı
        private int akatman; // arakatmanın neuron sayısı
        private neuron[] arakatman; // Ara Katman
        private double[] arkatmancikislar; // arakatmanın üreteceği çıkışlar -->
        çıkış katman girişleri
        private double[] arakatmanesikler; // arakatmanın eşik değerleri
        private double[] cikisesikler; // çıkış eşik değerleri
        private int ckatman;
        private neuron[] cikiskatman; // Çıkış Katmanı
        public double[] cikislar; // network genel çıkış
        public double[] beklenen; //çıkıştan çıkmasını beklediğimiz değer
        private double[,] giriskatsayilar; // [akatman,girisler.Length]
        private double[,] akatmankatsayilar;// [ckatman,akatman]
        private double[] cikissigma;//cikis sigma degerleri
        private double[] araktsigma;//arakatman sigma degerleri
        private double[,] Dgiriskatsayilar; // [akatman,girisler.Length] delta
        private double[,] Dakatmankatsayilar;// [ckatman,akatman] delta
        private double[] Dcikisesikler; // çıkış eşik değerleri delta
        private double[] Darakatmanesikler; // arakatmanın eşik değerleri
        public double genelhata = 10000;
        public double alfa = 0.5;
        public double momentum = 0.8;

        public network(int girissayisi, int akatman, int ckatman)
        {
            this.akatman = akatman;
            this.ckatman = ckatman;

            Dgiriskatsayilar = new double[akatman, girissayisi];
            Dakatmankatsayilar = new double[ckatman, akatman];
            Darakatmanesikler = new double[akatman];
            Dcikisesikler = new double[ckatman];

            Random random = new Random();
            girisler = new double[girissayisi];

            arakatman = new neuron[akatman];
            arkatmancikislar = new double[akatman];
            for (int i = 0; i < akatman; i++)
            {
                arakatman[i] = new neuron(girisler);
            }

            for (int i = 0; i < akatman; i++)
            {
                for (int j = 0; j < girissayisi; j++)
                {
                    arakatman[i].katsayilar[j] = random.NextDouble() * 2 - 1;
                }
                arakatman[i].esik = random.NextDouble() * 2 - 1;
            }
        }
    }
}

```



```

        arkatmancikislar[i] = arakatman[i].getCikis();
    }

    cikiskatman = new neuron[ckatman];
    cikislar = new double[ckatman];
    for (int i = 0; i < ckatman; i++)
    {
        cikiskatman[i] = new neuron(arkatmancikislar);
    }

    for (int i = 0; i < ckatman; i++)
    {
        for (int j = 0; j < akatman; j++)
        {
            cikiskatman[i].katsayilar[j] = random.NextDouble() * 2 - 1;
        }
        cikiskatman[i].esik = random.NextDouble() * 2 - 1;
        cikislar[i] = cikiskatman[i].getCikis();
    }
}

public void setGiris(double[] girisler)
{
    for (int i = 0; i < this.girisler.Length; i++)
    {
        this.girisler[i] = girisler[i];
    }

    for (int i = 0; i < akatman; i++)
    {
        for (int j = 0; j < girisler.Length; j++)
            arakatman[i].girisler[j] = girisler[j];

        arkatmancikislar[i] = arakatman[i].getCikis();
    }

    for (int i = 0; i < ckatman; i++)
    {
        for (int j = 0; j < akatman; j++)
            cikiskatman[i].girisler[j] = arkatmancikislar[j];

        cikislar[i] = cikiskatman[i].getCikis();
    }
}

public double[] getCikislar()
{
    return cikislar;
}

public void agiEgit(double[] beklenen)
{
    this.beklenen = beklenen;

    sigmaOlustur();
    agirlikDegistir();

    for (int i = 0; i < akatman; i++)
    {
        arkatmancikislar[i] = arakatman[i].getCikis();
    }
}

```

```

    }

    for (int i = 0; i < ckatman; i++)
    {
        for (int j = 0; j < akatman; j++)
            cikiskatman[i].girisler[j] = arkatmancikislar[j];

        cikislar[i] = cikiskatman[i].getCikis();
    }
}

public void sigmaOlustur()
{
    giriskatsayilar = new double[akatman, girisler.Length];
    akatmankatsayilar = new double[ckatman, akatman];
    arakatmanesikler = new double[akatman];
    cikisesikler = new double[ckatman];
    cikissigma = new double[ckatman];
    araktsigma = new double[akatman];
    genelhata = 0.0;
    for (int i = 0; i < ckatman; i++)// çıkış katman sigma degerleri
        bulunuyor
        {
            cikisesikler[i] = cikiskatman[i].esik;
            cikissigma[i] = cikislar[i] * (1 - cikislar[i]) * (beklenen[i] -
cikislar[i]);
            for (int j = 0; j < akatman; j++)
            {
                akatmankatsayilar[i, j] = cikiskatman[i].katsayilar[j];
            }
            genelhata += Math.Pow((beklenen[i] - cikislar[i]),(double)2);
        }
        genelhata = Math.Sqrt(genelhata);

    for (int i = 0; i < akatman; i++)//arakatman sigma degerleri bulunuyor
    {
        arakatmanesikler[i] = arakatman[i].esik;
        double sigmatoplam = 0.0;
        for (int k = 0; k < ckatman; k++)
        {
            sigmatoplam += cikissigma[k] * akatmankatsayilar[k, i];
        }

        araktsigma[i] = arkatmancikislar[i] * (1 - arkatmancikislar[i]) *
sigmatoplam;

        for (int j = 0; j < girisler.Length; j++)
        {
            giriskatsayilar[i, j] = arakatman[i].katsayilar[j];
        }
    }
}

public void agirlikDegistir()
{
    //cikis - arakatman arası ağırlıkların değiştirilmesi ve çıkış
eşiklerinin değiştirilmesi
    for (int i = 0; i < ckatman; i++)
    {
        for (int j = 0; j < akatman; j++)

```

```

        {
            Dakatmankatsayilar[i, j] = alfa * cikissigma[i] *
arkatmancikislar[j] + momentum * Dakatmankatsayilar[i, j];
        }
        Dcikisesikler[i] = alfa * cikissigma[i] + momentum *
Dcikisesikler[i];
    }

    for (int i = 0; i < ckatman; i++)
    {
        for (int j = 0; j < akatman; j++)
        {
            akatmankatsayilar[i, j] += Dakatmankatsayilar[i, j];
            cikiskatman[i].katsayilar[j] = akatmankatsayilar[i, j];
        }
        cikisesikler[i] += Dcikisesikler[i];
        cikiskatman[i].esik = cikisesikler[i];
    }

    //giriş-arakatman arası ağırlıkların değiştirilmesi ve arakatman
eşiklerinin değiştirilmesi
    for (int i = 0; i < akatman; i++)
    {
        for (int j = 0; j < girisler.Length; j++)
        {
            Dgiriskatsayilar[i, j] = alfa * araktsigma[i] * girisler[j] +
momentum * Dgiriskatsayilar[i, j];
        }
        Darakatmanesikler[i] = alfa * araktsigma[i] + momentum *
Darakatmanesikler[i];
    }

    for (int i = 0; i < akatman; i++)
    {
        for (int j = 0; j < girisler.Length; j++)
        {
            giriskatsayilar[i, j] += Dgiriskatsayilar[i, j];
            arakatman[i].katsayilar[j] = giriskatsayilar[i, j];
        }
        arakatmanesikler[i] += Darakatmanesikler[i];
        arakatman[i].esik = arakatmanesikler[i];
    }
}
}
}

```

ÖZGEÇMİŞ

Özcan OTKUN, 1975 yılında Trabzon Merkez'e bağlı Gözalan köyünde doğdu. İlk, Orta ve Lise öğrenimini babasının memur olması nedeniyle farklı il ve okullarda tamamladı. 2001 yılında Marmara Üniversitesi, Teknik Eğitim Fakültesi'nden Elektrik Öğretmeni olarak mezun oldu. 2007 yılında Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik-Elektronik Mühendisliği Anabilim dalında Yüksek Lisans eğitimini tamamladı ve aynı yıl Karadeniz Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik-Elektronik Mühendisliği Anabilim dalında Doktora çalışmalarına başladı. 2009 yılında Gümüşhane Üniversitesi, Gümüşhane Meslek Yüksekokulu'na Öğretim Görevlisi olarak atandı. Halen burada görev yapmakta olup evli ve iki çocuk babasıdır.