



T.C.

SELÇUK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

ELEKTROMANYETİK FİLTRELER İÇİN FPGA TABANLI ADAPTİF KONTROLÖR TASARIMI VE GERÇEKLEŞTİRİLMESİ

İlker Ali ÖZKAN

DOKTORA TEZİ

Elektrik-Elektronik Mühendisliği Anabilim Dalı

Haziran-2013
KONYA
Her Hakkı Saklıdır

ÖZET

DOKTORA TEZİ

ELEKTROMANYETİK FİLTRELER İÇİN FPGA TABANLI ADAPTİF KONTROLÖR TASARIMI VE GERÇEKLEŞTİRİLMESİ

İlker Ali ÖZKAN

Selçuk Üniversitesi Fen Bilimleri Enstitüsü
Elektrik-Elektronik Mühendisliği Anabilim Dalı

Danışman: Prof.Dr. Saadetdin HERDEM

Yıl, 2013, 168 Sayfa

Jüri
Danışmanın Prof.Dr. Saadetdin HERDEM
Doç.Dr. Salih GÜNEŞ
Yrd.Doç.Dr. Nihat YILMAZ
Yrd.Doç.Dr. Fuat KARAKAYA
Yrd.Doç.Dr. A. Afşin KULAKSIZ

Bu çalışmada, endüstriyel alanlardaki sıvıların içerdikleri mikron boyutlu manyetik özellikli parçacıklardan temizlenmesi işleminde kullanılan elektromanyetik filtre (EMF)'lerin yüksek performanslı çalışmalarını sağlayacak uygun bir kontrolör gerçekleştirilmiştir.

Literatürde; giren parçacık konsantrasyonu ve akış hızını kullanarak EMF kontrolü gerçekleştirilmişken, bu çalışma ile diğer teknolojik parametreleri ve EMF performansını da değerlendirebilen farklı kontrol yapıları geliştirilmiştir. Böylece EMF performansını dolaylı ya da direkt etkileyen tüm teknolojik parametrelerin göz önüne alınması sağlanmıştır. Ayrıca kontrol sisteminin endüstriyel sıvının türüne ve EMF performansını etkileyen teknolojik parametrelerin zamanla değişimine duyarlı olabilmesini sağlamak için, kontrolörler adaptif yapıda oluşturulmuştur.

Bu amaçla yapılan tez çalışmasında yapay zeka tekniklerinden bulanık kontrol, adaptif bulanık kontrol ve sinirsel bulanık kontrol yapıları kullanılarak beş farklı kontrolörler tasarlanmış ve gerçekleştirilmiş, bu kontrolörlerin avantaj ve dezavantajları değerlendirilerek EMF kontrolü için bir hibrid zeki kontrolör yapısı oluşturulmuştur. Bu kontrolör, teknolojik parametrelerin değişiminin EMF performansını etkilemesini önlemek amacıyla bir erken algılama sistemi ile güçlendirilmiştir.

Kontrolörlerin FPGA tabanlı olarak tasarlanması ile, çok fazla işlem yükünü istenilen sürede gerçekleştirebilmesi, bilgisayardan bağımsız olarak ve endüstriyel ortamlarda çalışabilmesi sağlanmıştır. Filtrasyon işleminin sürekliliği ve EMF matrislerinin tekrar tekrar kullanımı da kontrol edilmiştir.

Sonuç olarak bu çalışmada; EMF performansını etkileyen parametreleri değerlendiren ve bu parametrelerdeki değişimleri erken algılayan FPGA tabanlı hibrid zeki bir kontrolör tasarlanmış ve gerçekleştirilmiştir. Yapılan deneyler sonucunda gerçekleştirilen bu kontrol sisteminin güvenilir ve kararlı bir kontrol sağladığı görülmüştür.

Anahtar Kelimeler: Elektromanyetik Filtre, Elektromanyetik Filtre Kontrolü, Adaptif Kontrol, Hibrid Zeki Kontrolör, FPGA tabanlı kontrolör

ABSTRACT

Ph.D THESIS

FPGA BASED ADAPTIVE CONTROLLER DESIGN AND IMPLEMENTATION FOR ELECTROMAGNETIC FILTERS

Ilker Ali OZKAN

THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCE OF SELÇUK UNIVERSITY
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN ELECTRICAL-ELECTRONIC ENGINEERING

Advisor: Prof.Dr. Saadetdin HERDEM

Year, 2013, 168 Pages

Jury
Advisor Prof.Dr. Saadetdin HERDEM
Assoc.Prof.Dr. Salih GÜNEŞ
Asst.Prof.Dr. Nihat YILMAZ
Asst.Prof.Dr. Fuat KARAKAYA
Asst.Prof.Dr. A. Afşin KULAKSIZ

In this study, an appropriate controller for increasing the performance of electromagnetic filters (EMF) that are used to clean the micron size particles that have magnetic properties from the industrial liquids was implemented.

Apart from literature, considers input particle concentration and flow rate for EMF control, this study makes use of different controller structures that evaluate different technological parameters and performance of EMF. Thus, developed controller takes into account all technological parameters that affects performance of EMF either directly or indirectly. Moreover controller was implemented in adaptive structure to make it sensitive against industrial liquid types and time-varying technological parameters that affect the performance of EMF.

Five different controllers were designed and implemented by using artificial intelligence techniques namely fuzzy control, adaptive fuzzy control and neuro-fuzzy control. The advantages and disadvantages of these controllers were evaluated and a hybrid intelligent controller was implemented. This controller is strengthened with early sensing system to prevent the effects of technological parameter changes in to EMF's performance.

FPGA based implementation of the controller is provided following benefits; computer independent operation, operation in industrial environment and heavy processing capability in a required time. The continuity of filtration and reuse of EMF matrix were also controlled.

As a result FPGA based hybrid intelligent controller that evaluates the parameters that affect EMF performance and also perceives the changes in these parameters were designed and implemented. Experimental studies showed that developed controller system provides reliable and robust control.

Keywords: Electromagnetic filter, Electromagnetic filter control, Adaptive control, Hybrid intelligent controller, FPGA based controller

ÖNSÖZ

Tez çalışmamın yürütülmesi esnasında, çalışmalarıma yön veren, değerli bilgi ve yardımlarını esirgemeyen ve bana her türlü desteği sağlayan danışmanım Sayın Prof. Dr. Saadetdin HERDEM'e,

Bilgi ve tecrübeleri ile bana yol gösteren ve laboratuvar çalışmalarımda desteğini esirgemeyen Doç.Dr. İsmail SARITAŞ'a,

Her türlü fikir ve yardımlarıyla çalışmalarımda bana yardımcı olan tez izleme komitesi üyeleri Yrd.Doç.Dr. Fuat KARAKAYA ve Yrd.Doç.Dr. Nihat YILMAZ'a,

Proje kapsamında sağladığı maddi imkanlar nedeniyle TÜBİTAK (proje no : 109E037) ve Selçuk Üniversitesi Bilimsel Araştırmalar Koordinatörlüğü'ne (proje no: S.Ü. BAP 09101032) teşekkürlerimi sunar şükranlarımı arz ederim.

Ayrıca tez çalışmam süresince sabır ve desteklerinden dolayı sevgili eşime ve canım oğluma, moral ve manevi desteklerinden dolayı anne ve babama, kardeşlerime de teşekkürlerimi sunarım.

İlker Ali ÖZKAN
KONYA-2013

İÇİNDEKİLER

ÖZET ..... **iv**

ABSTRACT ..... **v**

ÖNSÖZ ..... **vi**

İÇİNDEKİLER ..... **vii**

SİMGELER VE KISALTMALAR ..... **x**

1. GİRİŞ **1**

2. KAYNAK ARAŞTIRMASI **4**

3. ELEKTROMANYETİK FİLTRELEME **11**

3.1. Elektromanyetik Filtre 11
3.2. Elektromanyetik Filtre Yapısı 12
3.3. Elektromanyetik Filtrelerde Performans 14
3.4. Bulgular 15

4. ELEKTROMANYETİK FİLTRE KİTİ **17**

4.1. Kullanılan Cihazlar 17
4.1.1. Akışmetre (Flowmeter) 17
4.1.2. Invertör 18
4.1.3. FPGA geliştirme kartı 18
4.1.4. Sirkülasyon pompası 20
4.1.5. Akuatörlü vana ve elektropnömatik valf 21
4.2. Kullanılan Yazılımsal Bileşenler 21
4.2.1. VHDL tasarım dili 21
4.2.1.1. VHDL tanımlama türleri 22
4.2.1.2. Simülasyon 24
4.2.1.3. Sentezleme 24
4.2.1.4. VHDL veri nesneleri 25
4.2.1.5. VHDL temel yapısal elemanları 26
4.2.2. Verilerin gösterimi 28
4.2.2.1. Kayan noktalı sayılar 28
4.2.2.2. Sabit Noktalı Sayı Gösterimi 29
4.3. Elektromanyetik Filtre 30
4.3.1. Elektromanyetik nüve 31
4.3.2. Filtre matrisi 31
4.3.3. Filtre bobini 32
4.4. Tasarımı Yapılan Elektronik Devre ve Bileşenler 33
4.4.1. Parçacık konsantrasyon sensör bobinleri 33
4.4.2. Yüksek frekanslı sinüs güç kaynağı 34
4.4.3. Parçacık konsantrasyon yükselteç devresi 35
4.4.4. ADC/DAC kartı 36

4.4.5. Röle kartları 37
4.4.6. PWM sürücü kartı 38
4.5. EMF Kiti Gövde Tasarımı 39
4.6. EMF kitinin genel çalışma yapısı 40

5. KULLANILAN YAPAY ZEKA TEKNİKLERİ 44

5.1. Bulanık Mantık 45
5.1.1. Bulanık kümeler 47
5.1.2. Bulanık çıkarım sistemleri 48
5.1.2.1. Bulanıklaştırıcı 50
5.1.2.2. Bulanık kurallar tabanı 51
5.1.2.3. Bulanık çıkarım mekanizması 52
5.1.2.4. Durulaştırıcı 54
5.1.3. Bulanık çıkarım sistemindeki sınırlılıklar 55
5.2. Yapay Sinir Ağları 55
5.2.1. Biyolojik nöron yapısı 56
5.2.2. Yapay sinir hücresi 57
5.2.3. Aktivasyon fonksiyonları 58
5.2.3.1. Sigmoid (*logsig(x)*) aktivasyon fonksiyonu 58
5.2.3.2. Lineer (*lin(x)*) aktivasyon fonksiyonu 59
5.2.3.3. Tanjant hiperbolik (*tansig(x)*) aktivasyon fonksiyonu 60
5.2.3.4. Doyum fonksiyonu (*sat(x)*) aktivasyon fonksiyonu 60
5.2.3.5. Keskin Sınırlayıcı (*signum(x)*) aktivasyon fonksiyonu 61
5.2.4. YSA'nın yapısı ve işlem elemanları 62
5.2.5. Çok katmanlı ileri beslemeli ağ yapısı 63
5.2.6. Hatanın geriye yayılması algoritması ve genelleştirilmiş delta kuralı 64
5.3. ANFIS 71
5.3.1. ANFIS mimarisi 71
5.3.2. ANFIS için geri yayılım algoritması 74

6. KULLANILAN DENETLEYİCİ YAPILARI 76

6.1. Bulanık Mantık Denetleyici 76
6.2. PI Tip Bulanık Mantık Denetleyici 78
6.3. Kendini Organize Edebilen Bulanık Mantık Kontrolör 79
6.4. Kendini Ayarlayabilen PI tip Bulanık Mantık Kontrolör 81
6.5. Adaptif ANFIS Kontrolör 83

7. EMF KONTROLÖR BİLEŞENLERİ 85

7.1. FPGA Yapısında Kullanılan Temel Bileşenler 85
7.1.1. Sistem saati bileşeni 85
7.1.2. Giriş arayüzü bileşeni 86
7.1.3. Çıkış arayüzü bileşeni 87
7.1.4. EMF performans bileşeni 88
7.1.5. RS232 verici bileşeni 89
7.1.6. RS232 alıcı bileşeni 90
7.1.7. Filtre dolum kontrol bileşeni 92
7.1.8. Filtre temizleme ve röle kontrol bileşeni 92
7.2. Bilgisayar Veri Toplama Arayüzü 94

8. DENEYSEL ÇALIŞMALAR 97

8.1. Sistemin Temiz Sıvı ile Çalıştırılması 98
8.2. GPK ve AH Değişimine Bağlı Bulanık Kontrol 99
8.2.1. GPK-AH bulanık kontrolörün yapısı 99
8.2.2. GPK-AH bulanık kontrolörün FPGA'da programlanması 101
8.2.3. GPK-AH bulanık kontrolör deneyi 105
8.3. PI tip Bulanık Kontrolör 108
8.3.1. PI tip bulanık kontrolörün yapısı 108
8.3.2. PI tip bulanık kontrolörün FPGA'da programlanması 110
8.3.3. PI tip bulanık kontrolör deneyi 111
8.4. Kendini Organize Edebilen Bulanık Kontrolör 112
8.4.1. Kendini organize edebilen bulanık kontrolörün yapısı 112
8.4.2. Kendini organize edebilen bulanık kontrolörün FPGA'da programlanması 115
8.4.3. Kendini organize edebilen bulanık kontrolör deneyi 117
8.5. Kendini Ayarlayabilen PI tip Bulanık Kontrolör 119
8.5.1. Kendini ayarlayabilen PI tip bulanık kontrolörün yapısı 119
8.5.2. Kendini ayarlayabilen PI tip bulanık kontrolörün programlanması 122
8.5.3. Kendini ayarlayabilen PI tip bulanık kontrolör deneyi 122
8.6. ANFIS Kontrolör 124
8.6.1. ANFIS kontrolörün yapısı 124
8.6.2. ANFIS kontrolörün kontrolörün FPGA'da programlanması 125
8.6.3. ANFIS kontrolör deneyi 129
8.7. Hibrid Zeki Kontrolör 130
8.7.1. Hibrid zeki kontrolörün yapısı 130
8.7.2. Hibrid zeki kontrolörün FPGA'da programlanması 133
8.7.3. Hibrid zeki kontrolör deneyi 133

9. SONUÇLAR ve ÖNERİLER 135

9.1. Sonuçlar 135
9.2. Öneriler 145

KAYNAKLAR 146

EKLER 152

ÖZGEÇMİŞ 166

SİMGELER VE KISALTMALAR

Simgeler

μ	:	Üyelik fonksiyonu
Ψ	:	Elektromanyetik filtre performansı
C_i	:	Giren parçacık konsantrasyonu
C_o	:	Çıkan parçacık kosnantrasyonu
Fe_3O_4	:	Manyetit
Fe_2O_3	:	Hematit
S_1	:	Giren parçacık sensörü
S_2	:	Çıkan parçacık sensörü
S_3	:	Akış hızı sensörü
ε	:	Öğrenme Oranı
f_g	:	Giriş frekansı
f_c	:	Çıkış frekansı

Kısaltmalar

EMF	:	Elektromanyetik filtre
KOEBK	:	Kendini organize edebilen bulanık kontrol
BK	:	Bulanık kontrol
KABK	:	Kendini ayarlayabilen PI tip bulanık kontrol
ANFIS	:	Adaptive Neuro- Fuzzy Inference System
FPGA	:	Field Programming Gate Array
YSA	:	Yapay sinir ağı
BM	:	Bulanık mantık
GPK	:	Giren parçacık konsantrasyonu
ÇPK	:	Çıkan parçacık konsantrasyonu
AH	:	Akış hızı
AF	:	Aktif filtre
TF	:	Temizlenecek filtre
YZ	:	Yapay zeka

1. GİRİŞ

Endüstriyel alanda teknolojik işlemlerin birçoğunda kullanılan temel veya yardımcı elemanlardan birisi de sıvı ve gazlardır. Bu endüstriyel sıvı ve gazların içerdikleri yabancı maddelerin büyük çoğunluğu manyetik özellikli parçacıklardan oluşmaktadır. Sıvı ve gazların içerisinde bulunan bu yabancı maddeler, çoğunlukla çok düşük konsantrasyonlarda ve mikron boyutlarında olmaktadır. Bu parçacıklar, teknolojik işlemlerin içerdikleri cihaz ve düzeneklerin arızalanmasına, ürünlerin bozulmasına hatta bazı durumlarda sistemin tümüyle durmasına sebep olabilmektedirler. Bu durum ürünün kalitesinin düşmesine, iş gücü, zaman ve dolayısıyla ekonomik kayıplara sebep olmaktadır. Bu nedenlerle teknolojik işlemlerde sıvı ve gazların içerdiği manyetik özellikli bu parçacıklardan temizlenmesi gerekmektedir. Endüstriyel sıvı ve gazları bu parçacıklardan temizleme işleminde ayrıştırıcı (seperator), yüksek etkinlikte parçacık yakalayıcı (hepa filter) gibi yöntemler kullanılmaktadır. Ancak normal filtreleme yöntemlerinin performanslarının düşük olması, yüksek sıcaklıklarda kullanılamaması ve hızlı temizleme gereksinimini karşılamaması gibi sebeplerle Elektromanyetik Filtre (EMF) teorisi geliştirilerek pek çok alanda uygulanmıştır.

Bir EMF, manyetik özellikli gövde, bu gövde üzerinde sarılı dış manyetik alan oluşturan bobin, içerisinden endüstriyel sıvı ve gazın geçtiği filtre matrisi elemanlarından oluşur. Filtre matrisi, dış homojen alanla kolayca mıknatıslanabilen manyetik malzemelerden oluşturulur. Matris elemanları dış alanın etkisi ile mıknatıslanarak temas noktaları etrafında yüksek gradyantlı (gradient) tutma bölgeleri oluştururlar. Temizlenecek olan teknolojik sıvı filtre matrisinden geçerken, içerdiği küçük boyutlu parçacıklar bu tutma bölgelerinde tutularak biriktirilir. EMF'lerin filtre matrisinin dayanıklı olması ve filtreleme işleminde hiçbir yardımcı kimyasal, biyolojik vb. ayıraçlar içermediğinden, radyoaktif özellikli ve yüksek sıcaklığa sahip endüstriyel ortamlarda da kullanılabilirler.

EMF'lerin bu parçacıkları tutabilme başarıları performanslarını belirlemektedir. EMF'lerin performanslarını; endüstriyel sıvının akış hızı, sıvı içerisindeki manyetik özellikli parçacık miktarı, bu parçacıkların büyüklükleri, manyetik geçirgenlikleri, filtre matrisinin çapı, uzunluğu, tipi, bobin tipi ve sarım sayısı gibi etkileyen pek çok faktör bulunmaktadır. EMF'lerin performanslarının yüksek tutulması endüstriyel sıvıların işlem süresince sürekli temiz olmalarını sağlamaktadır. Bu durum endüstriyel sıvının

içerdiği parçacıkların sebep oldukları olumsuzlukları ortadan kaldırmaktadır. Günümüzde manyetik filtreleme işleminde EMF performansının her koşulda yüksek tutulması, tekrar tekrar kullanılarak ekonomik kazanç sağlanması ve sürekli görevini sürdürmesi gibi konularda önemli problemler vardır. Ayrıca, kullanım alanları ve endüstriyel sıvılara bağlı olarak EMF'lerin teknolojik parametreleri değişim göstermektedir. Bu değişimin oluşturduğu belirsizlik yapılacak kontrol sistemi için bir problem teşkil etmektedir. Literatürde bu problemin giderilmesi amacıyla teknolojik parametrelerden endüstriyel sıvının akış hızı ve konsantrasyonunu dikkate alan bir bulanık kontrol sistemi gerçekleştirilmiştir (Sarıtaş, 2008). Literatürde EMF'nin kontrolü üzerine bu çalışmadan başka bir çalışmaya rastlanmamıştır.

Bu çalışmanın amacı; mikron boyutlu parçacıkların tutulmasını sağlayacak bir EMF'nin ve farklı çalışma şartları için kontrolünün bilgisayardan bağımsız bir şekilde modern kontrol yöntemleri kullanarak gerçek zamanda yapılmasıdır. İstenen optimum şartlar, sensörler yardımı ile alınan anlık değerlerin tasarlanan kontrolör tarafından değerlendirilmesi ve gerektiğinde filtre bobinine uygulanan akımın dolayısıyla manyetik akının kontrol edebilmesiyle oluşturulacaktır. Yapılan tez çalışmasıyla literatürden farklı olarak kontrol sistemi, verilen teknolojik parametreleri ve EMF performansını değerlendirilmiştir. Endüstriyel sıvının türü, değişimi ve filtre performansını etkileyen teknolojik parametrelerin zamanla değişimine sistemin duyarlı olabilmesini sağlamak için adaptif bir kontrolör kullanılmıştır. Bu adaptif kontrolü gerçekleştirmek için hibrid yapay zeka yöntemleri kullanılmıştır. Süreklilik ve tekrar tekrar kullanım için çift EMF'nin sırasıyla kendini temizleme ve/veya sıvıyı temizleme amaçlı olarak kontrolü sağlanacaktır. Böylece filtrenin matrislerinin temizlenmesi için üretimde kaybedilen zaman, iş gücü ve ekonomik kayıp ortadan kaldırılmıştır. Gerçekleştirilen kontrol sisteminde istenen en uygun şartlar, sensörler yardımı ile alınan anlık değerlerin tasarlanan Alan Programlanabilir Kapı Dizileri (*Field Programming Gate Array- FPGA*) tabanlı kontrolör tarafından değerlendirilmesi ve gerektiğinde filtre bobinine uygulanan akımın dolayısıyla manyetik akının kontrol edebilmesiyle oluşturulmuştur. Geliştirilen kontrol sisteminin FPGA tabanlı olarak gerçekleştirilmesi ile sistemin bilgisayardan bağımsız olarak çalışması, endüstriyel ortam şartlarında çalışabilmesi, paralel işlem yeteneği sayesinde fazla işlem yüküne rağmen kontrolün hızlı bir şekilde gerçekleştirilmesi sağlanmıştır.

Ayrıca bu tez çalışması, Tübitak-1001 Bilimsel ve Teknolojik Araştırma Projelerini Destekleme Programı tarafından desteklenen "Endüstriyel Sıvıların

Temizlenmesinde Kullanılan Elektromanyetik Filtrelerin Yapay Zeka Yöntemleriyle Adaptif Kontrolü ve Kit Tasarımı (109E037)" isimli projenin kontrolör tasarımı ve gerçekleştirilmesi kısmını da oluşturmaktadır. Bu sayede, yukarıdaki özelliklere göre oluşturulan adaptif EMF kontrol sistemi endüstride kullanılabilecek şekilde bir kit haline getirilmiştir.

2. KAYNAK ARAŞTIRMASI

Literatürde, manyetik filtre teorisi ve hibrid zeki kontrolörler üzerine ayrı ayrı yapılan çeşitli çalışmalar bulunmaktadır. Fakat literatür taramalarında, hibrid zeki sistemler kullanılarak yapılan EMF kontrolü ile ilgili çalışmalarla karşılaşılmamıştır.

Hibrid zeki sistemlerin kontrol alanında uygulamaları ile ilgili pek çok çalışmalar mevcuttur. Çalışma alanımızla farklılıklar içerse de, yapılacak çalışmalara ışık tutması açısından bu kaynaklar da incelenmiştir.

Mukhtiar ve ark. (2013) şebekeye bağlı yenilenebilir enerji kaynaklarının kontrolünün güç elektroniğinde önemli bir rolü olduğunu vurgulamışlardır. Yaptıkları çalışmalarında yenilenebilir invertör arayüzü için adaptif neuro-fuzzy kontrol uygulaması sunmaktadırlar. Doğrusal olmayan bu sistemin dinamiklerinin hızlı değişiminden dolayı geleneksel PI kontrolörün başarısız olduğu belirtilmiş ve gerçekleştirilen kontrolörün çift yönlü yük akışını ve doğrusal olamayan yük dengelenmesini eş zamanlı yaptığı vurgulanmıştır. Hedeflenen sistemin Matlab SimPowerSystem ile değişik çalışma şartlarında benzetimi yapılmıştır.

Arulmozhiyal (2012) yaptığı çalışmada endüstriyel uygulamalarda yaygın olarak kullanılan fırçasız DC motorlarda hız kontrolü için Bulanık PID kontrolör geliştirmiştir. Gerçekleştirilen bulanık PID kontrolör ile geleneksel PID kontrolör karşılaştırılması yapılmıştır. Modelleme, kontrol ve simülasyon işlemi Matlab/Simulink kullanılarak yapılmış ve Xilinx FPGA XC3S 400E üzerine uygulanmıştır. Deney sonuçlarında bulanık PID kontrolörün geleneksel PID kontrolörden daha iyi sonuç verdiği görülmüştür.

Gatt ve ark. (2012) yaptıkları çalışmada araba benzeri bir robotun otomatik olarak park etme problemine çözüm sunmuşlardır. Önerilen sistem geçerli bir park yeri için alanı taramakta ve gerekli manevraları yaparak bu alana robotu park etmektedir. Paralel ve dik park işlemi hızlı cevap verebilmesi ve donanımsal olarak az yer kaplamasından dolayı bulanık mantık tabanlı bir sistem ile oluşturulmuştur. En iyi verimlilik için sistem FPGA'ya uygulanarak, düşük maliyetli ve kararlı bir sistem elde edilmiştir.

Özkan ve ark. (2012) manyetik filtrelemenin endüstriyel sıvılardaki manyetik parçacıkları temizlemede en etkili metotlardan biri olduğunu ve yüksek sıcaklık, radyasyon gibi değişik çalışma şartlarında çalışabildiklerini belirtmişlerdir. Bu özelliklerinin yanında EMF performansını etkileyen bir çok faktör olduğu belirtilmiştir.

Yaptıkları çalışmada taşınabilir, kompakt bir EMF kitini endüstriyel kullanıma uygun olarak tasarlayıp gerçekleştirmişlerdir. EMF performansını etkileyen faktörleri dikkate alan bir FPGA tabanlı kontrol sistemi gerçekleştirmişlerdir. Bu kontrol sistemi ile EMF performansının maksimum seviyede tutulması ve EMF'nin kendi kendisini temizlemesi sağlanmaktadır. Yapılan etkili filtreleme, üretimin kalitesini, sürdürülebilirliğini, enerji kazancını sağlamaktadır.

Sarıtaş ve ark. (2012) yaptıkları çalışmada Yapay Sinir Ağı (YSA) kullanarak MF performansı ve zaman ilişkisi için bir model geliştirmişlerdir. Model tek gizli katmana sahiptir. Geliştirilen YSA modelinin giriş parametreleri EMF çıkış konsantrasyonu ve zaman, çıkış parametresi ise EMF performansıdır. Elde edilen YSA modelinin istatistiksel olarak başarılı bir şekilde modellediği gösterilmiştir. Geliştirilen modelin EMF performansını tahmin etmek için pratik bir teknik olarak kullanılabileceği belirtilmiştir.

Herdem ve ark. (2011) gerçekleştirdikleri projede teknolojik alanda manyetik özellikli parçacıkları endüstriyel sıvılar içerisinden temizleme yöntemi olan manyetik filtreleme çalışması yapmışlardır. Projede endüstriyel sıvılardaki mikron boyutlu manyetik özellikli parçacıkları temizleyecek bir manyetik filtre tasarlamış ve endüstriyel sıvının akış hızı, manyetik filtreye giren ve manyetik filtreden çıkan parçacık konsantrasyonu değerlerini dikkate alan FPGA tabanlı bir adaptif bulanık kontrol sistemi ile kontrol etmişlerdir. Bu kontrol ile filtre performansı maksimum seviyede tutulmakta ve filtrenin kendi kendini temizlemesi sağlanmaktadır. Yapılan çalışma ile hem filtre performansı optimize edilmiş hem de filtre matrislerinin kontrolü yapılmıştır. Böylece filtrenin matrislerinin temizlenmesi için üretimde kaybedilen zaman, iş gücü ve ekonomik kayıp ortadan kaldırılmıştır.

Loan ve ark. (2011) yaptıkları çalışmada yaygın olarak kullanılan bulanık mantığın çeşitli alanlarda bulanık işlemcilerin hızının düşük olması nedeniyle problem oluşturduğunu belirtmişlerdir. Bunun gidermek için çalışmalarında yamuk şekilli üyelik fonksiyonları arasındaki eşleştirme derecesini hesaplamak için bir maksimum minimum devre mimarisi önermişlerdir. Elde ettikleri mimarideki optimizasyonlar işlemi sayesinde %33 daha az yer kaplamaktadır ve mimariyi XILINX FPGA üzerine uygulamışlardır.

Özkan ve ark. (2011) yaptıkları çalışmada EMF performansını en yüksek seviyede tutacak bir bulanık mantık kontrolör tasarımı VHDL kullanılarak FPGA üzerinde gerçekleştirmişlerdir. Gerçekleştirilen modelin test sonuçları ile

MATLAB&Fuzzy Logic Toolbox'dan elde edilen test sonuçları karşılaştırılarak kontrol sisteminin doğrulaması yapılmıştır. Diğer sistemlere göre önerilen sistemin dikkate değer şekilde başarılı olduğu görülmüştür.

Sarıtaş ve ark. (2010) endüstriyel sıvıların içerisinde bulunan mikron boyutundaki manyetik parçacıkların temizlenmesi için bir manyetik filtre tasarlamışlardır. Tasarladıkları filtre, hem endüstriyel sıvının temizlenmesi hem de filtre matrisinin temizlenmesi için bulanık mantık yöntemiyle kontrol edilmiştir. Bu tasarım ile filtre matrisi içerisinden geçen manyetik özellikli parçacıkların miktarı arttıkça veya endüstriyel sıvının akış hızı yükseldikçe filtre bobinine uygulanan akım da artmaktadır. Böylece filtrenin performansı kontrol edilebilmektedir. Ayrıca filtrenin performansı düştüğünde filtre matrisinin temizlenmesi sağlanarak aynı filtre matrisinin tekrar kullanımına olanak sağlanmıştır ve kontrol sisteminin %68 enerji kazancı sağladığı belirtilmiştir.

Sarıtaş (2008) yaptığı çalışmada endüstriyel sıvılardaki mikron boyutlu manyetik özellikli parçacıkları temizlemek amacıyla mevcut teoriden elde edilen sonuçları kullanarak filtre gövdesi, filtre bobini ve filtre matrisini tasarlamış ve gerçekleştirmiştir. Geri besleme bilgilerini sağlamak amacıyla filtreye giren ve çıkan parçacık konsantrasyonu ve endüstriyel sıvının akış hızı verilerini DAQ kartı ile bilgisayara aktarmıştır. Giren parçacık konsantrasyonu ve endüstriyel sıvının akış hızı parametrelerini giriş ve bobin akımı değerini çıkış olarak alan bir bulanık kontrolörü Labview programı kullanarak tasarlamıştır. Manyetik akının kontrolü ile yüksek performans ve enerji kazancı sağlamıştır. Filtre matrislerinin iki adet kullanılması, filtre seçiminin ve akışın otomatik olarak kontrol edilmesi ile temizleme işleminin sürekliliğini sağlamıştır.

Pal ve ark. (2007) yaptıkları çalışmada ısıtma, havalandırma ve klimalarda hava basıncı kontrolünde kullanılan kendini ayarlayabilen bir bulanık kontrol sistemi geliştirmişlerdir. Gerçekleştirilen sistemde kontrol edilen işlemin o anki durumuna uygun çıkış ölçeklendirme faktörü bulanık kurallar ile belirlenmektedir. Çıkış ölçeklendirme faktörünü ayarlayan kural tabanı kontrol edilen değişkenin hata ve hata değişimine göre belirlenmektedir. Gerçekleştirilen sistem PID tip kontrol sistemi ile karşılaştırılmış daha iyi sonuç verdiği gösterilmiştir.

Abbasov (2007) yaptığı çalışmada farklı manyetik filtreleme karakteristiklerini ve bu karakteristiklerin manyetik filtreleme parametrelerine etkisini açıklamaktadır. Gözeneklerdeki manyetik alanlar ve bu alanın manyetik parçacıklar üzerindeki etkisi,

ferromanyetik küreciklerden oluşan paketlenmiş yatakların mıknatıslanma özelliği olarak belirtmiştir. Biriktirme profili, parçacıkları yakalama bölgeleri ve bu bölgelerde ki akış hızı değişiminin etkisi sunulmuştur. Manyetik filtre performansındaki rejim parametrelerinin etkisi incelenmiştir. Filtre elemanlarının geometrisi, filtreleme hızının logaritmik katsayısı, parçacıkların büyüklüğü ve diğer filtreleme parametreleri ile analitik bir ifade elde edilmiştir.

Özkan ve ark. (2007) deneysel ve teorik olarak ifade edilebilen manyetik filtre performansı-zaman ilişkisine yapay zeka tekniklerinden ANFIS kullanarak bir model geliştirmişlerdir. Geliştirdikleri bu modelin, manyetik sıvılardaki teknolojik sıvının temizleme işlemindeki filtre performansı-zaman değişken karakteristiğini tahmin etmeyi sağladığını ifade etmişlerdir. Tahmin edilen bu sonuçların manyetik filtrasyon işleminin optimizasyonu, kontrolü ve yeni filtre tasarımı için kullanılabilecek çözümler sunduğunu belirtmişlerdir.

Jacobo ve ark. (2006) gelecekteki enerji talebine göre otomobil uygulamalarında senkron fazlı DC-DC dönüştürücülerin tasarımının çok aktif bir alan olduğunu belirtmişlerdir. Bu dönüştürücülerin optimal davranışı için denetleyici tasarımı önemlidir. Çalışmalarında bir kontrol algoritması tasarlamış ve Matlab ile Simulink üzerinde gerçekleştirmişlerdir. Bulanık kontrolör Xilinx Systems Generator kullanılarak Spartan 3 FPGA'ye uygulanmış ve gerçek zamanlı bir kontrol elde edilmiştir.

Sarıkaya ve ark. (2006) yaptıkları çalışmalarında, manyetik filtreleme teorisini, manyetik filtreler ile endüstriyel sıvıların filtrasyonunu güçlendirmek için incelenmişlerdir. Filtre performansı üzerine filtreleme hızı ve harici manyetik alan yoğunluğun etkilerini araştırmışlar ve filtreleme hızı üzerinde logaritmik verim katsayısına bağımlılığını sorgulamışlardır. Harici manyetik alan tarafından sebep olan bu dispersiyon halindeki parçacıkların manyetik alınganlığındaki değişim ve filtrenin küçük açıklıkları boyunca sıvının akış oranı özelliklerindeki değişimlerin, çeşitli endüstriyel alanlarda manyetik filtreleme sistemlerinin tasarımı, geliştirilmesi ve modellenmesinde, dikkate alınacak en temel faktörler olduğunu göstermişlerdir.

Vasantha ve ark. (2005) artan sayıdaki bulanık uygulamanın paralellik ve yüksek hızlı bulanık işlem gereksinimine uygun olarak FPGA üzerinde bulanık kontrolörün donanımsal uygulamasını gerçekleştirmişlerdir. Tasarım Spartan FPGA çip üzerinde test edilmiştir. Farklı giriş ve çıkış sayıları için maksimum frekans ve kullanılan kaynaklar karşılaştırılmıştır.

Herdem (2005) yaptığı çalışmada, manyetik filtre ve seperatörler tarafından tutulan ve gerçek endüstriyel sıvılarda meydana gelen parçacıkların manyetik alınganlığının ortalama değerinin, filtreleme işleminin başlangıcında bilinemeyeceğini belirtmiştir. Tutulan parçacıkların manyetik alınganlığı manyetik alan yoğunluğuyla değiştiği için manyetik filtrasyon denkleminde manyetik filtre performansı ile manyetik alan yoğunluğu arasındaki ilişkinin tamamen doğru kabul edilemeyeceğini belirtmiştir. Bu problemin küçük aralıklarda manyetik alanın ortalama değerini kullanmak ve bu alandaki parçacıkların ortalama alınganlığını belirlemek suretiyle halledilebileceğini belirtmiştir. Filtre matrisinin gözenekleri içinde meydana gelen manyetik alanın ve filtre elemanlarının manyetik özelliklerinin dikkate alınarak filtrasyon sürecinin özelliklerinin yaklaşık olarak elde edilebileceğini ifade etmiştir.

Zezulka ve ark. (2004) yaptıkları çalışmada kalıcı mıknatıslar ile yüksek gradyanlı bir manyetik saperatör esasına dayanan manyetik filtre geliştirmişler ve yapısını sunmuşlardır. Bu manyetik filtrede NdFeB materyalinden manyetik akı kullanarak elektromıknatıs elde etmişlerdir. Manyetik blokların yüksekliği, genişliği ve bunlar arasındaki hava boşluklarının ölçülmesini amaçlamışlar ve bunlara bağlı manyetik akı büyüklüğünü hesaplamışlardır. Böylece yüksek değerde manyetik indüksiyon elde etmişler ve birçok uygulama yapmışlardır.

Köksal ve ark. (2003) yaptıkları çalışmada manyetik filtrede endüstriyel sıvıların temizlenme sürecinin karakteristiğinin belirlenmesi için bir matematiksel model önermişlerdir. Çalışmada filtre uzunluğunun manyetik filtre performansına etkisi de gösterilmiştir. Deneysel sonuçlarla karşılaştırarak yaptıkları modellemenin geçerliliğini istatistiksel olarak göstermişlerdir.

Abbasov (2001) yaptığı çalışmada, endüstriyel sıvı ve gazların saflığının mikron ve mikron altı boyutlara sahip parçacıklar tarafından etkilendiğini ve bu parçacıkların birçoğunun demir ya da paramanyetik maddelerden oluştuğunu belirmiştir. Bu parçacıkların düşük konsantrasyonlu olmasından dolayı klasik filtreler (mekanik, ayırıcı, vb) önemli bir filtrasyon işlemi yapmadığından dolgulu yataklı manyetize olan manyetik filtreler bu parçacıkların yakalanma kapasitesini daha fazla arttırdığını göstermiştir.

Herdem ve ark. (2001) yaptıkları çalışmalarında, teknolojik sıvı ve gazların temizlenme süreci mekanizmalarını tahmin ve kontrol etmek için genel bir model bulunmadığını ve bir teknolojik ortamın manyetik filtreleme işleminin elektrik devre teorisi kullanarak simüle edilmesinin mümkün olduğunu belirtmişlerdir. Çalışma ile

farklı rejim altındaki filtreleme işleminin karakteristiklerini elektriksel devrenin doğrusal olmayan parametrelerinin uygun değişiklikleri ile elde etmişlerdir. Ayrıca, elde edilen bu modelin manyetik olmayan filtreleme veya benzer süreçli işlemler için uygulanabileceği belirtilmiştir.

Rajani ve ark. (2000) çalışmalarında kararlı bir kendini ayarlayabilen bulanık PI kontrol sunmuşlardır. Kontrol edilen işlemin durumuna göre çıkış ölçeklendirme katsayısı online olarak güncelleme katsayısı ile değiştirilmektedir. Güncelleme katsayısı kontrol edilen değişkenin hata ve hata değişimi ile belirlenmektedir. Gerçekleştirilen kontrolör basit bir kural tablosu ve simetrik üçgensel üyelik fonksiyonu ile tasarlanmıştır. Gerçekleştirilen kontrolör geleneksel PI tip kontrolör ile karşılaştırılmış ve daha iyi sonuç verdiği görülmüştür.

Herdem ve ark. (2000) çalışmalarında, laminer akış şartlarında eksenel olarak dizilmiş manyetik filtredeki filtre empedansı ve filtreleme veriminin teorisini açıklamışlardır. Uzunlamasına dizilmiş teller boyunca eksenel olarak akan süspansiyonun akış profili Kuwabara-Happel hücre modeli ile belirlenmiştir. Hem filtre empedansı hem de filtre verimi için ifadeler elde etmişlerdir. Genelde bu ifadelerin ideal akış şartlarında daha önceki filtreleme teorileri ile belirlenenlerden farklı olduğunu göstererek türetilen teorik formülleri basitleştirmişler, böylece mühendislik uygulamalarında kolayca kullanılabilir olduğunu belirtmişlerdir. Sonuçları literatürde yer alan deneysel sonuçlar ile karşılaştırmışlar ve birbiriyle tutarlı olduklarını göstermişlerdir.

Herdem ve ark.'nın (1999) yaptıkları çalışmada, yakalanan parçacıkları biriktiren ve ferromanyetik kürelerden oluşan granüler matrise sahip olan bir manyetik filtrenin gözeneklerindeki sıvının akış rejimini dikkate alan bir filtreleme mekanizmasını incelemişlerdir. EMF verimliliğinin filtreleme sisteminin farklı parametrelerinin etkisi ile tanımlandığı belirtilmiş ve EMF verimliliği kolayca ölçülüp tespit edilebilen manyetik ve statik basınç oranları ile ifade edilmiştir. Elde edilen teorik sonuçlar deneysel sonuçlar ile karşılaştırılarak uyum içinde olduğu gösterilmiştir.

Abbasov ve ark. (1999) tarafından yapılan çalışmada, kürelerden oluşan tanecikli matrise sahip bir yüksek gradyantlı manyetik filtrenin performans karakteristikleri, sistem parametrelerinin geniş bir aralığı için boyutsuz parametrelere göre belirlenmiştir. Elde edilen sonuçların, manyetik filtrelerle ilgili literatürde görülen bazı çelişkileri gidermek bakımından açıklayıcı olduğu belirtilerek laboratuvar veya

endüstride kullanılan filtreler için literatürde verilen deneysel veriler ile iyi bir uyum içerisinde olduğunu ifade etmişlerdir.

Herdem ve ark. (1999) çalışmalarında, magnetize edilmiş manyetik kürelerden oluşturulan filtre matrisindeki parçacıkların biriktirilmesinin stokastik modelini göz önüne almışlar. Yakalama bölgesinde oluşan hem yakalama hem de kopma olayları ile karakterize edilen doyma yarıçapının zamana bağımlılığını veren dağılım fonksiyonunu belirlemişlerdir. Yakalama bölgesindeki parçacıkların hareket özelliklerini hesaba katarak, dağılımı Fokker-Planck denkleminin çözümünden değerlendirmişlerdir. Elde ettikleri sonuçları diğer metotların sonuçları ile ve literatürde rapor edilen bazı deneyler ile karşılaştırmışlar ve benzer olduklarını göstermişlerdir.

Abbasov ve ark. (1999) tarafından yapılan çalışmada, yüksek gradyantlı manyetik filtreler için genel bir model tanıtmışlardır. Filtre performansının fonksiyonel bağımlılığı sistemin fiziksel ve geometrik parametrelerinden elde edilmektedir. Türetilen teorik ifade ile literatürde görünüşte ters gibi duran iki deney sonucunun her ikisi de açıklanabilmiştir. Yeni kalite faktörü süspansiyon viskozitezi yerine süspansiyon yoğunluğunu içermektedir.

3. ELEKTROMANYETİK FİLTRELEME

Sanayide üretim kalitesini etkileyen faktörlerden biri de teknolojik ortamda yer alan endüstriyel sıvı ve gazların kalitesidir. Bu endüstriyel sıvıların kalitesinin düşük olması teknolojik işlemin bozulmasına, teknolojik işlemlerin içerdiği set ve cihazların kullanım sürelerinin azalmasına ve sanayinin genel olarak veriminin düşük olmasına neden olmaktadır.

Endüstriyel sıvı ve gazların kalitesinin yüksekliği, bunların teknolojik işlemlerde kullanıldığı süre içerisinde fiziksel ve kimyasal özelliklerinin sabit kalması ve teknolojik işlemler sonucu içerdikleri çeşitli karışımlar bakımından temiz olmaları ile belirlenir (Abbasov, 2002). Endüstriyel sıvı ve gazların kalitesini yüksek tutmak için sürekli olarak karışımlardan temizlenmesi gerekmektedir. Bu temizleme işlemi teknolojik işlemin başlangıcında, sonunda veya herhangi bir aşamasında gerçekleştirilebilir (Herdem, 1999).

Yapılan çok sayıda deney sonucunda, endüstriyel sıvı ve gazların içerdikleri karışımların fiziksel ve kimyasal özelliklerinin geniş spektruma sahip olduğu belirlenmiştir. Fakat bütün durumlarda yabancı maddelerin çoğunluğunun demir bileşenine sahip olduğu bildirilmektedir (Watson, 1973; Abbasov, 2007).

Bu demir bileşenlerinden endüstriyel sıvıların temizlenmesi sanayi dallarının önemli problemlerinden biridir. Endüstriyel sıvılar bu karışımlardan temizlenmediğinde, çalışma süresi kısalmakta ve aşınmalara bağlı olarak yapılan işlemin ve servis maliyetlerinin artmasına, set ve cihazlarda aşınmalara, üretilen malzeme üzerinde paslanma bölgeleri oluşturarak maliyetin düşmesine ve sonuç olarak kullanılan sistemin veriminin düşmesine veya tamamen yok olmasına yol açarlar.

Endüstriyel sıvı ve gazların içerdikleri manyetik özellikli parçacıkların boyutları mikron seviyesinde olup 0.1-1 μm civarındadır (Heitmann, 1969; Liberman, 1984; Ebner, 2007). Demir bileşenli bu oksitlerin kolay kolay erimemesi nedeniyle bunlar teknolojik sıvıların içerisinde parçacık şeklinde bulunmaktadırlar (Abbasov, 2002).

3.1. Elektromanyetik Filtre

Çeşitli endüstriyel alanlarda kullanılan sıvı ve gazların içerdikleri çoğu durumda manyetik özellik sergilemektedirler. Endüstriyel sıvı ve gazların küçük boyutlu manyetik özellikli parçacıklardan temizlenmeleri sistem verimliliği açısından büyük

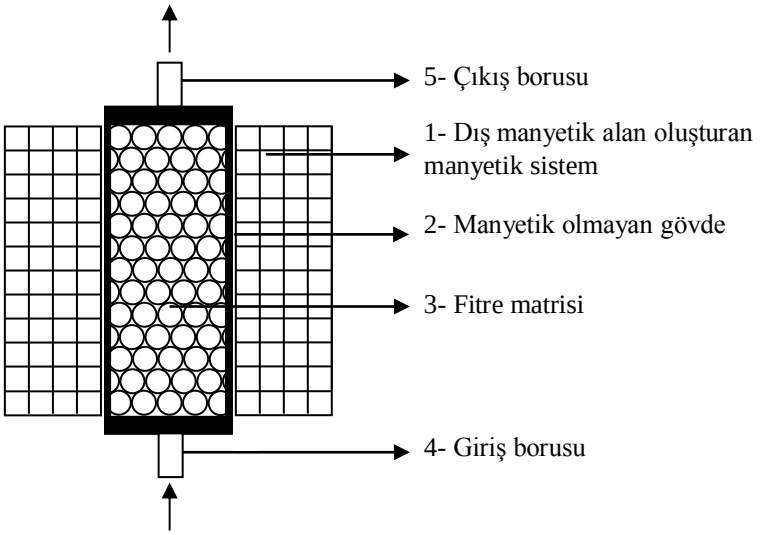
önem arz etmektedir. Bu temizleme işlemi, fizyolojik veya kimyasal yöntemlerle gerçekleştirilmektedir. Fakat bu bileşenlerin parçacık şekilde olması ve temel olarak manyetik özellik sergilemelerinden dolayı elektromanyetik yöntemlerin kullanılması daha avantajlıdır. Parçacıkların manyetik özelliğinden faydalanıp, farklı yapı ve tiplerde filtre matrisleri kullanılarak temizleme işlemi yapan filtrelere elektromanyetik filtre (EMF) denilmektedir (Abbasov, 2002).

EMF'ler günümüzde işlem süreçlerinden manyetik duyarlı bileşenleri çıkarmak için kullanılan basit ve diğer filtreleme yöntemlerine göre uygun maliyetli bir filtreleme çözümüdür (Milhous, 2013). EMF temel olarak manyetitleri (Fe_3O_4) endüstriyel sıvılardan temizlemeyi amaçlamış ise de; hematit (Fe_2O_3), bakır, kobalt, nikel gibi elementlerin manyetit ile ferritlerinden oluşan zayıf manyetik özellikli parçacıkları da temizleyebilmektedir. Ayrıca temizlenmesi istenilen parçacıkların manyetikliği arttırılarak da EMF ile filtreleme işlemi yapılabilmektedir. Örneğin Okada ve ark. (2004) yaptıkları çalışmada jeotermal sular içerisindeki arseniği filtreleme işlemine tabii tutmak için manyetikliğini demir hidroksit ($Fe(OH)_3$) kullanarak arttırmış ve sonra EMF ile filtreleme işlemini gerçekleştirmişlerdir.

EMF'nin diğer bir özelliği ise oldukça verimli olarak çalışabilmesidir. Günümüzde endüstriyel olarak kullanılan EMF'lerin çalışma şartlarına bağlı olarak %90-95 oranında verim ile çalıştığı belirtilmektedir (Milhous, 2013). EMF'ler çalışma şartları açısından da esnek bir yapıya sahiptirler. Endüstriyel ve sanayi alanlarında yüksek sıcaklık, radyasyon vb. etkilerin bulunduğu ortamlarda etkili bir biçimde çalışabilmektedirler. Ayrıca Endüstriyel alandaki sisteme kolayca adapte edilebilmekte ve temizlediği endüstriyel sıvının kimyasal yapısını değişikliğe uğratmamaktadırlar.

3.2. Elektromanyetik Filtre Yapısı

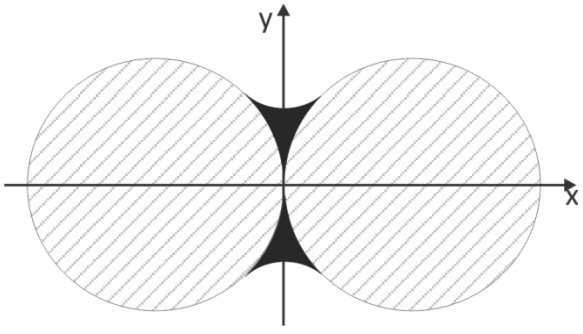
EMF'lerin klasik filtrelerden farklı olarak süzgeç elemanlarını ferromanyetik malzemeler oluşturmaktadır. Temel bir EMF'nin prensip şeması Şekil 3.1'de verilmiştir (Sarıtaş, 2010).



Şekil 3.1. Manyetik filtrenin prensip şeması

EMF'de filtrelenecek olan endüstriyel sıvı, giriş borusunu (4) geçerek filtreleme işleminden geçirilir ve çıkış borusundan (5) sisteme geri verilir. Filtre matrisi elemanları arasında manyetik alan oluşturmak için dış manyetik sistem (1) elektromıknatıs veya mıknatıslardan oluşturulur. Gövde (2) manyetik olmayan paslanmaz malzemeden oluşturulmaktadır. Filtre matrisi elemanları, manyetik akı yoğunluğu $B = 0{,}05–1{,}5\ T$ olan dış homojen alanın etkisiyle kolayca mıknatıslanırlar ve kendi etraflarında yüksek gradyanta sahip olan çoklu alanlar oluştururlar. Bu alanların manyetik alan şiddetleri uygulanan dış manyetik alanın şiddetinden çok çok daha büyük olurlar. Bu alanların oluştuğu bölgeler "aktif bölge" veya "tutma bölgesi" adını alırlar (Alizade ve ark. 1997, Herdem ve ark. 1999).

Teknolojik sıvılar filtre matrisinin gözeneklerinden geçerken içerdikleri manyetik özellikli parçacıklar, tutma bölgelerinde kuvvetli dış manyetik alan etkisi ile toplanırlar (Şekil 3.2). Burada manyetik akı x ekseni doğrultusundadır ve sıvı akışı z ekseni doğrultusundadır.



Şekil 3.2. Manyetik tutma bölgesinin profili

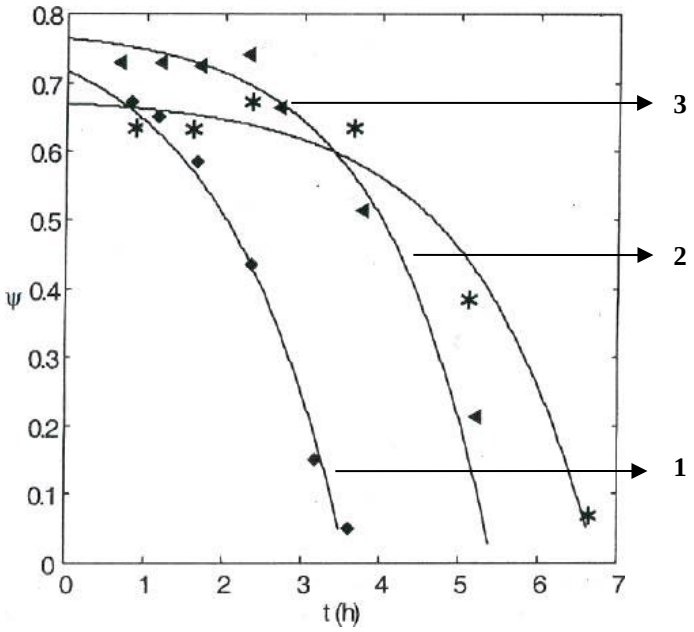
Mekanik ve hidrolik kuvvetlerin etkisi ile karakterize edilen klasik filtrelerden farklı olarak, EMF'de manyetik özellikli parçacıklar daha büyük bir kuvvetin etkisinde kalırlar (Abbasov, 2002).

Günümüzde kullanılan EMF'lerin pek çok yapı tipi bulunmaktadır. Bu tez çalışmasında Tübitak projesi kapsamında gerçekleştirilen selenoit tip manyetik filtre kullanılmıştır (Herdem, 2011).

3.3. Elektromanyetik Filtrelerde Performans

Elektromanyetik filtrelerin zamanla performansları düşmekte, çok uzun süre çalıştırılmalarında performansı sıfıra kadar düşebilmektedir. Doğal olarak bu durum pek çok parametreye bağlı olarak değişim göstermektedir. Filtrelerin teknolojik parametreleri olarak ifade edilen bu parametreler, filtre uzunluğu, filtre elemanlarının büyüklüğü ve tipi, sıvının akış hızı, filtreye uygulanan manyetik akı vb. ile belirlenmektedir.

Şekil 3.3'de termik santral kondensatı ile yapılan çalışma sonucunda filtre performansının filtreleme zamanına göre değişimi grafiği gösterilmiştir (Abbasov, 1999; Herdem, 1999; Herdem, 2005).



Şekil 3.3. Filtre performansının (ψ) filtreleme zamanına (t) göre değişimi (1,2,3-Termik santral kondensatı)

Literatürde farklı pek çok ifadesi olmakla birlikte filtre performansı genel olarak Denklem 3.3'de belirtilen ifade ile hesaplanır.

$$\psi = 1 - \frac{C_o}{C_i} \tag{3.1}$$

Burada C_i giren parçacık konsantrasyonu, C_o çıkan parçacık konsantrasyonu'dur.

3.4. Bulgular

Manyetik filtreleme işlemi yapılan sistemlerde Abbasov (2002) tarafından belirtilen genel bulgular aşağıdaki gibidir.

- Her hangi teknolojik sıvıların veya gazların temizlenmesi için kullanılacak EMF'nin yapı tipinin seçilmesi temizlenen ortamın tipine, içerdiği karışımın konsantrasyonuna, bu karışımların manyetik özelliklerine ve boyutlarına bağımlıdır.
- EMF elemanları paslanmayan elemanlardan yapılmalıdır.
- Filtre matrisi elemanları birbiriyle teğetlenen mıknatıslanmış ferromanyetik malzemelerden oluşturulmalıdır.
- Daha iyi sonuçlar elde etmek için filtreleme işleminden önce karışımlar manyetik alanda işleme tabi tutulabilir.
- EMF ile filtreleme işleminin genel bir teorisi yoktur. Bu konudaki teorik çalışmalar esasen mıknatıslanmış filtre elemanı etrafında oluşan yüksek gradyantlı manyetik alanda parçacığın hareketini karakterize eden diferansiyel denkleminin belli durumlar için çözümüdür.
- EMF uygulamalarında karşılaşılan en önemli problemlerden biri de mikron seviyeli parçacıklardan temizlenmesinde etkili olan performans, endüstriyel sıvının akış hızı, basınç vb. teknolojik parametrelerin önceden belirlenmesidir.
- EMF'nin çalışma rejiminin belirlenmesi için içerdiği manyetik karışım konsantrasyonunun belirlenmesi büyük önem arz eder.
- EMF'yi etkileyen tüm parametrelerin değişimini dikkate alan kontrol sistemleri EMF performansını yüksek seviyede tutmak için gereklidir.

- EMF'nin endüstride bütün çalışma süresi boyunca belli bir performansı sağlaması önemlidir.
- EMF'nin temizleme sürecinde paralel bir EMF ile filtreleme işleminin sürekliliği sağlanabilir (rejenerasyon). Rejenerasyon işlemi basit ve etkin olmalıdır.

4. ELEKTROMANYETİK FİLTRE KİTİ

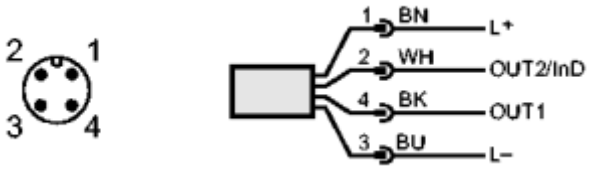
Bu tez çalışmasında Tübitak-1001 Bilimsel ve Teknolojik Araştırma Projelerini Destekleme Programı tarafından desteklenen "Endüstriyel Sıvıların Temizlenmesinde Kullanılan Elektromanyetik Filtrelerin Yapay Zeka Yöntemleriyle Adaptif Kontrolü ve Kit Tasarımı (109E037)" isimli proje ile gerçekleştirilen EMF kiti kullanılmıştır (Herdem, 2011). Bu bölümde kullanılan EMF kitinin genel yapısı ve adaptif kontrolü için kullanılan cihaz, sensör ve devreler hakkında genel bilgi verilecektir. EMF kiti hakkındaki ayrıntılı bilgiye 109E037 nolu Tübitak-1001 projesinden ulaşılabilir (Herdem, 2011).

4.1. Kullanılan Cihazlar

4.1.1. Akışmetre (Flowmeter)

Akışmetre doğrusal ya da doğrusal olmayan, kitle veya sıvı veya gazların hacimsel akış hızını ölçmek için kullanılan bir araçtır. Bu tez çalışmasında filtreleme işlemine tabi tutulacak endüstriyel sıvının akış hızını ölçülmesinde kullanılmıştır. Elektromanyetik, ultrasonik, kütlesel ve türbinmetre tipinde akışmetreler bulunmaktadır. Yapılan çalışmada akışmetre sensörünün manyetik alandan etkilenmemesi için ultrasonik çalışma prensibiyle ölçüm yapan "IFM Elector Inc." firmasına ait SU-8001 modeli akışmetre kullanılmıştır (SU8001, 2013).

Kullanılan ultrasonik akışmetre *200 l/dk* hızına kadar ölçme kapasitesine sahiptir. Ayrıca ölçme işleminde sıvıya temas etmeden hassas ölçüm yapabilmektedir. Akışmetre veri aktarmak için *M12* soket tipinde veri çıkış portuna sahiptir. Sistem üzerinde akışmetre *OUT2* modunda hacimsel akış analog çıkışı aktif edilerek kullanılmıştır (Şekil 4.1).



Şekil 4.1. Akışmetre M12 soket bağlantı uçları

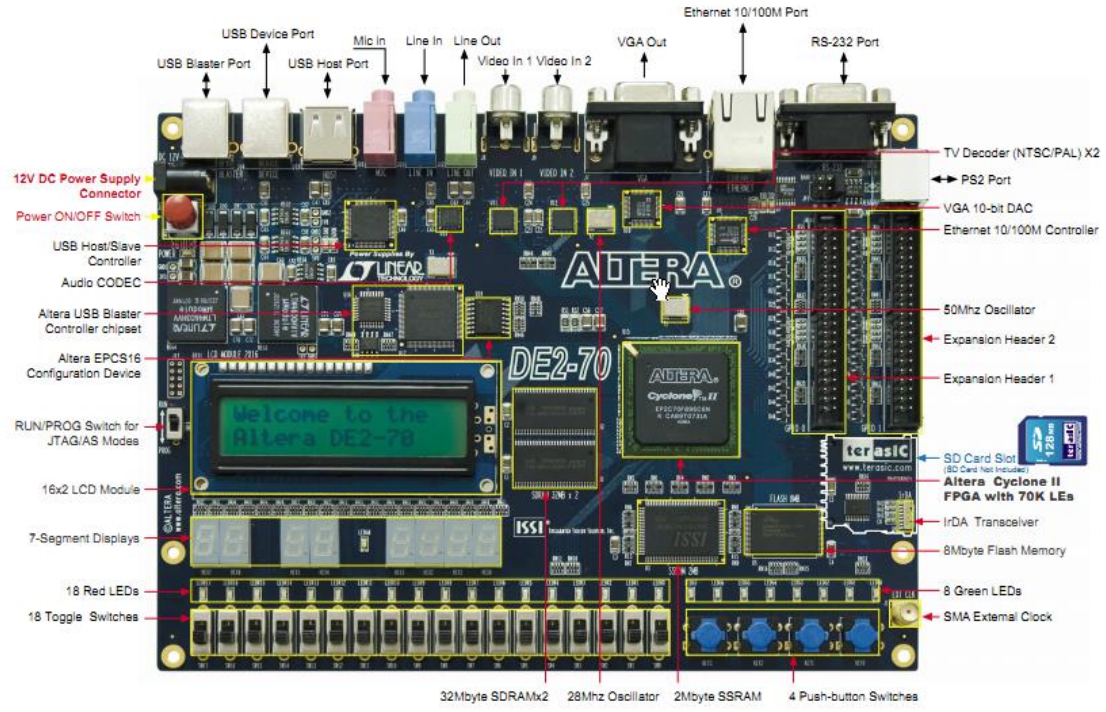
4.1.2. Invertör

Filtrenin matrisinin temizlenmesi işleminde düşük frekanslı AC gerilimin filtre bobinine uygulanmasında kullanılmak üzere invertör kullanılmıştır.

Tez çalışmasında Schneider firmasına ait Altiver 312 frekans invertörü kullanılmıştır. Bu invertör; *220-240 V* arasında *5.8 A* akıma kadar kullanılabilmektedir. 0,55 *kW* güce sahiptir. *0.5 Hz - 500 Hz* arasında frekans ayarlaması yapılabilmektedir (Altiver, 2013).

4.1.3. FPGA geliştirme kartı

EMF kitinin kontrolünde kullanılmak üzere Şekil 4.2'de verilen Terasic firması tarafından üretilen FPGA tabanlı Altera DE-2 70 kartı kullanılmıştır. DE2-70 kartı güçlü bir Cyclone II yongası içermektedir. Kart üzerindeki tüm bileşenler bu yonganın pinlerine bağlı olması ile kullanıcı amacına uygun bileşenleri istediği şekilde yapılandırabilmektedir.



Şekil 4.2. Altera DE-2 70 FPGA geliştirme kartı (Altera, 2013)

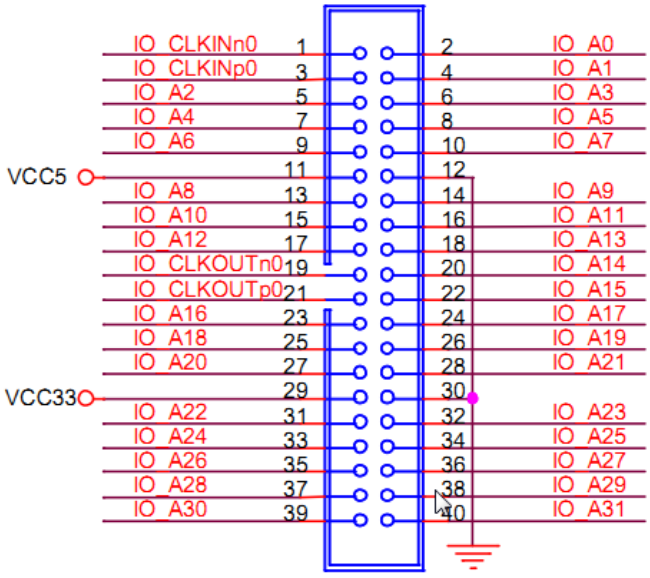
DE-2 70 kartı basit devrelerden geniş multimedya projelerine kadar geniş bir uygulama kapasitesine sahiptir. Kart Çizelge 4.1'de verilen donanım özelliklerine sahiptir (Altera, 2013).

Altera DE2-70 kartı üzerinde iki adet 40 pinlik genişleme yuvası barındırmaktadır. Her bir genişleme yuvasındaki 36 pin Cyclone II FPGA'ya direkt bağladır. Diğer 4 adet pin ise toprak ve güç iletim hatları için kullanılmaktadır.

Çizelge 4.1. Altera DE-2 geliştirme kartı özellikleri

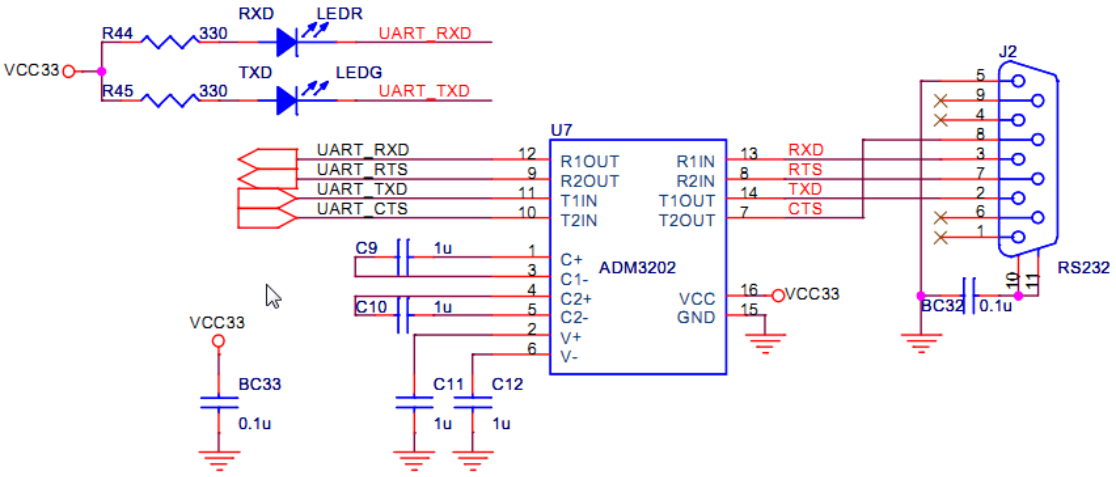
Özellik	Açıklama
FPGA	Cyclone II EP2C70F896C6 FPGA
	EPCS64 64-Mbit seri yapılandırma cihazı
G/Ç Arayüzleri	FPGA yapılandırması için dahili USB-Blaster
	Line In/Out, Mikrofon (24-bit Audio CODEC)
	Video Çıkış (VGA 10-bit DAC)
	Video Giriş (NTSC/PAL/Multi-format)
	RS232
	Infrared port
	PS/2 fare veya klavye portu
	10/100 Ethernet
	USB 2.0
	İki adet 40 pinlik genişleme yuvası
Hafıza	64 MB SDRAM, 2 MB SSRAM, 8 MB Flash
	SD Bellek için kart slotu
Görüntü	Sekiz adet 7-segment display
	16 x 2 LCD
Anahtar ve ledler	18 adet geçiş anahtarı
	18 adet kırmızı LED
	9 adet yeşil LED
	4 adet pushbutton anahtar
Saat hızı	50 MHz saat
	27 MHz saat
	Dışsal SMA saat girişi

Şekil 4.3'de genişleme yuvalarının şematik gösterimi verilmiştir. Genişleme yuvaları IDE kablo seti ile çalışmaya uygun yapıdadır. Sensör devrelerinden gelen analog veriler ADC ile dijital hale çevrilerek IDE kablosu ile FPGA'ye iletilmektedir. Bu sebepten dolayı ADC kart tasarımı yapılırken genişleme yuvalarının çıkışı DE2-70 kartı pin tablosuna uygun olarak tasarlanmıştır.



Şekil 4.3. Genişleme yuvalarının şematik gösterimi (Altera, 2013)

Tez çalışmasında elde edilen verilerin DE2-70 FPGA kartından bilgisayara aktarılması için RS232 seri portu kullanılmıştır. DE2-70 kartı RS-232 iletişimi için 9 pinli D-SUB bağlantı ucu ve ADM3202 iletişim yongasını kullanmaktadır. RS-232 seri portunun yonga bağlantı şeması Şekil 4.4'da verilmiştir.



Şekil 4.4. RS-232 yonga bağlantı şeması (Altera, 2013)

4.1.4. Sirkülasyon pompası

Endüstriyel sıvı veya süspansiyonların kapalı çevriminde kullanılabilmesi için deney düzeneğinde sirkülasyon pompası kullanılmıştır. Grunfos firmasına ait olan sirkülasyon pompası kademeli devir ayarı yapılabilen özelliğe sahiptir. Sirkülasyon

pompasının bu özelliği çalışmada deney esnasında endüstriyel sıvının akış hızını değiştirmede kullanılmıştır.

4.1.5. Akuatörlü vana ve elektropnömatik valf

Manyetik filtrelere sıvı giriş ve çıkış yönlerini kontrol etmek için pnömatik olarak çalışan akuatörlü vanalar, aktüatörlü vanaları elektriksel olarak kontrol etmek için elektropnömatik valfler kullanılmıştır. Bu valflerin kontrolör tarafından kumanda edilmesi ile sistem üzerinde çalışma rejimleri arasında geçiş sağlanabilmektedir.

4.2. Kullanılan Yazılımsal Bileşenler

FPGA'lar kadar etkili olan başka bir kavram ise HDL'dir (*Hardware Description Language - Donanım tanımlama dili*). Donanım tanımlama dillerinin kullanılması ile *System-on-Chip* (*SoC*) teknolojisini oluşturmuştur. SoC teknolojisinde çoğunlukla donanım tanımlama dilini kullanılmaktadır. Sistemin tanımlama aşamasından sonra derleme ve davranışsal benzetim adımları gerçekleştirilir. Sistemden beklenen cevap elde edildiğinde zamansal benzetim aşamasına gelinir. Bütün birimler sentezleme ve yerleştirme işlemi sonunda tekrar programlanabilir olan devreye aktarılır.

EMF kitinin kontrol edilmesinde Bölüm 4.1.3'de verilen Altera DE2-70 FPGA geliştirme kartı kullanılmıştır. Bu geliştirme kartı üzerindeki tüm programlar VHDL donanımsal tanımlama dili kullanılarak yazılmıştır. VHDL dili ve programlanmasın da kullanılan sayı gösterim formatları bu bölümde verilmektir.

4.2.1. VHDL tasarım dili

VHDL bir donanım tanımlama dilidir. Açılımı VHSIC Hardware Description Language terimlerinin baş harflerinden oluşmaktadır. VHSIC ise *Very High Speed Integrated Circuits* teriminin kısaltılmışıdır. Bir elektronik devre veya sistemi davranışını tanımlamada kullanılmaktadır.

VHDL dili dokümantasyon, doğrulama ve büyük dijital sistemlerin sentezlenilmesi gibi farklı amaçlar içinde kullanılmaktadır. Teorik olarak aynı VHDL kodu kullanılarak her üç amaç gerçekleştirilmektedir. Bu VHDL dilinin temel özelliklerinden biridir ve zamandan ve emekten tasarruf sağlar.

Ayrıca VHDL dili devre simülasyonu ve devre sentezinde kullanılarak donanım katmanı tanımlanabilmektedir. Yaygın olarak kullanımını sağlayan unsurlardan biride bir teknoloji veya satıcıdan bağımsız standart bir dil olmasıdır. Bu yüzden dolayı taşınabilir ve tekrar kullanılabilir.

VHDL ile donanımı tanımlamak için kullanılan yaklaşımlar yapısal, veri akışı ve davranışsal metot olarak sıralanabilir. Çoğu zaman dizayn sırasında bu üç yaklaşımın karışımı kullanımı görülür.

VHDL, dijital sistemlerdeki fiziksel donanımların modellenmesinde kullanılır. Tasarımcılar için VHDL'i önemli kılan özellikleri şunlardır;

a) Çok yaygın bir HDL dili olması

b) Sistemlerin farklı seviyelerde yapısal ve/veya davranışsal olarak belirlenebilmesi

c) VHDL tabanlı simülasyon araçlarının makul fiyatlarda bulunabilmesi

d) Birçok EDA tool sağlayıcısının VHDL'i desteklemesi

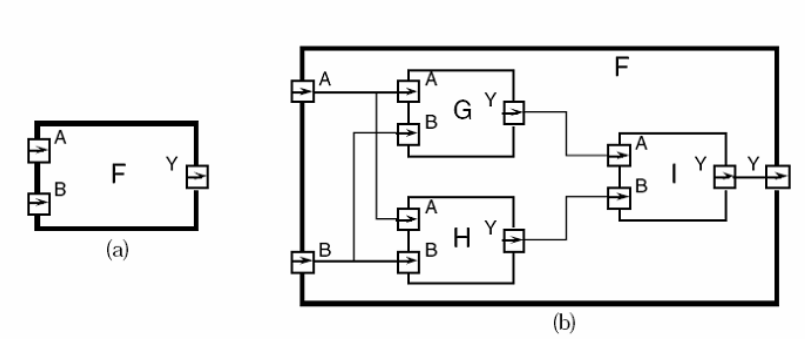
e) VHDL sadece elektronik modelleme ile kısıtlanmamıştır. Evrensel bir modelleme dili olup, elektromanyetik, kimyasal, hidrodinamik ve diğer sistemlerin modellemesin de kullanılabilir.

4.2.1.1. VHDL tanımlama türleri

VHDL dijital bir sistemi açıklayan metin dosyaları olarak görülebilir. Dijital sistem bir davranışsal tanımlama veya yapısal tanımlama gibi farklı formlarda temsil edilebilir. En yaygın soyutlama seviyeleri olarak bilinen bu tanımlamalar, karmaşık sistemleri gerçekleştirmede tasarımcılara yardımcı olmaktadırlar.

4.2.1.1.1. Yapısal tanımlama

Yapısal tanımlama istenen işlemleri gerçekleştirmek için kapılar veya bileşen bloklarının bağlantılarını tanımlar. Yapısal tanımlama öncelikle dijital sistemin grafiksel bir temsilidir ve bu yüzden sitemin fiziksel sunumuna daha yakındır. Şekil 4.5'de bir yapısal tanımlama örneği verilmektedir.



Şekil 4.5. Bir yapı tanımlama örneği

Bir sayısal elektronik sistemi giriş ve/veya çıkışlarıyla beraber bir modül olarak tanımlanabilir. Çıkışın elektriksel değerleri girişin elektriksel değerlerinin bir ve birkaç fonksiyonu olarak karşımıza çıkarlar. Şekil 4.5.a'da bir sayısal sistemin modül halinde gösterimi mevcuttur. Görüldüğü üzere F modülünün A ve B olmak üzere iki girişi ve bir Y çıkışı vardır. VHDL terminolojisi kullanılarak F bir varlık (entity) olarak düşünülürse girişler ve çıkışlar port diye nitelendirilebilir.

Modüle işlevi kazandırmanın bir yolu modülü alt modüllere ayırmaktır. Bu modüllerin portlarını oluşturduktan *signal* tipi yapılarla birbirleriyle haberleştirmektir. Şekil 4.5.b'de bir bütün yapının daha küçük yapılara nasıl ayrıldığı gösterilmiştir. Burada F modülü G, H ve I alt modüllerine ayrıştırılıp bunların istenen fonksiyonu gerçekleştirecek konfigürasyonunu yapmak gerekecektir. Bu tür tanımlamaya yapısal tanımlama denir. F bir yapısal tanımlama olduğu gibi G, H, ve I alt modülleri de yapısal tanımlama olabilir.

4.2.1.1.2. Davranışsal tanımlama

Birçok durumda bir işlevi yapısal olarak tanımlama uygun olmayabilir. Bazen bir modül diğer yapısal tanımlanmış bir hiyerarşinin en alt tabanında olabilir. Bazen sadece satıcıdan alınan tüm devrelerle sistem tasarlanmak istenebilir. Belki de içsel bir yapıya ihtiyaç duyulmayan fonksiyonlar gerçekleştirilmek istenebilir. Bu tür tanımlamaları yapmak için fonksiyonel veya davranışsal tanılama adı verilen bir tanımlama tipi ile tanımlanırlar. Yani bu tip tanımlamalarda tasarımın yapısını tanımlamadan davranışını tanımlamak yeterlidir. Bir tasarımda gerektiği yerde yapısal tanımlama gerektiği yerde ise davranışsal tanımlama kullanılabilir.

4.2.1.2. Simülasyon

VHDL'de yazılan bir tasarım tanımı, bir VHDL simülasyon programından faydalanılarak test edilir. Tanımlamanın simülasyonunu yapabilmek için simülasyon programına bir takım test girişleri verilir, program daha önceden belirlenen aralıklarla bu test girişlerini tasarımın modeline uygular ve çıkışlar üretir. Bu sonuçlar tasarımcı tarafından gözlenerek modelin istenildiği gibi çalışıp çalışmadığına karar verilir.

Simülatör tasarımın her aşamasında kullanılabilir. Tasarımın yüksek seviyelerinde yapılan simülasyonda sadece tasarımın fonksiyonel davranışı hakkında bilgi elde edilir. Bu aşamada simülasyon çok hızlıdır, fakat simülasyon sonuçlarından tasarımın gerçek devre elemanlar ile çalışması ve zamanlaması konusunda çok fazla elde edilemez. Daha düşük tasarım aşamalarına gidildikçe simülasyon daha fazla zaman alacaktır, fakat simülasyon sonuçlarından tasarımı çalışması ve zamanlaması konusunda daha fazla bilgi elde edilebilir.

4.2.1.3. Sentezleme

Sentezleme, tasarım tanımlamasının bir seviyeden daha düşük seviyedeki başka bir seviyeye çevrilmesi olarak tanımlanabilir. Bu çevirme işlemi, C programla dilinde, yüksek seviyeli dilin makine diline çevrilmesine benzetilebilir. Sentezleme programına HDL tanımlama verilir, sentezlenmiş devrede bulunması gereken gecikme ve alan özellikleri programa girilir. Verilen özelliklere sahip devre, sentezleme program ile, tasarmcı tarafından belirtilen teknoloji kütüphanesindeki elemanlar kullanılarak elde edilir. Sentezleme sonuçları, optimal devre şeması, sentezleme sonucunda elde edilen devreden beklenen performans ve sentezlenmiş devrenin kapladığı alandır. Aşağıda davranışsal sentezleme, RTL sentezleme ve lojik sentezleme anlatılmıştır.

• Davranışsal sentezleme, C programına benzer, algoritmik olarak yazılmış tanımlamanın RTL tanımlamaya çevrilmesidir. RTL'deki tanımlama veri yollar, bellek elemanlar, kontrol birimleri içerir.

• RTL sentezleme, saklayıcı iletişim fonksiyonlarından ardışıl bir devre elde etmek üzere devre şemasının oluşturulmasıdır.

• Lojik sentezleme, Boole fonksiyonlarının kombinezonsal devre elemanlarıyla gerçekleştirilmesidir.

4.2.1.4. VHDL veri nesneleri

Her bir veri nesnesi belirli bir tipe ait olan değerleri tutmaktadır. Kullanılan nesneler donanımda değerler alırlar ve donanım seklini belirlerler.

Sinyal (Signal) : Fiziksel olarak devre içinde yer alan donanım içindeki ara değişkenlerdir. Güncel değeri ve belirlenmiş bir sonraki değerleri tutarlar. Sinyal tanımlama *architecture* alanı içinde en basta *begin*'den önce olmalıdır.

Sinyal tanımlaması şu şekildedir:

SIGNAL sinyalin_adı : sinyalin_tipi [:=başlangıç değeri];

SIGNAL tanımlamasının ardından sinyal ismi gelir. Sinyal isimleri ihtiyaç kadar belirtilir. Belirttiğimiz isim sayısınca sinyal üretilir. Tanımlamada da belirtildiği gibi sinyal nesnelerine ilk değer atanabilir. Son olarak sinyaller *entity, architecture,* ve *package*(*paket*) yapılarının içinde tanımlanabilir. Paketler her yerde kullanılabileceği için burada yapılan tanımlamalar global özellik taşır.

Degişken (Variable) : Değişkenler yerel olarak sadece *Process*'ler veya altprogramlar içinde geçici değerleri tutmak için kullanılırlar. Sinyal tanımlamanın tersine değişken tanımlamada, değişkenin tanımlandığı yerde değer atanamayabilirler. Değer atamaları daha sonraki satırlarda olabilir. Değişkenler ortaya çok hızlı çıkıp işini yapıp ortadan kaybolurlar yani hafızada yer kaplamazlar. *Function, procedure* ve *process* içinde değişken kullanılabilir. Bir değişkenin tanımlanması genel hatlarıyla aşağıdaki gibi olur;

VARIABLE değişkenin_adı : değişkenin_tipi [:=başlangıç_değeri] ;

Sabit (Constant) : Sabitler bir tasarımda çok özel verileri tutmak için kullanılırlar. Sabitler tasarımcıya çok iyi dokümantasyona sahip bir model ortaya koyma olanağı sağlar. Eğer tasarımı yapılan modelde değişmez değerlere ihtiyaç varsa sabitler kullanılır. Tasarımcı isterse sabitlerin değerini elle değiştirip programı tekrar derleme ve optimum değeri bulma olanağına kavuşur. Genel sabit bildirimi aşağıdaki gibi olur;

CONSTANT sabitin_adı : veri_tipi [:= başlangıç_değeri];

Tip tanımı (type): Her hangi bir veri tipi tanımlamak için kullanılan VHDL yapısıdır. Genelde birden fazla boyutu olan diziler veya katar yapıları oluşturmak için kullanılırlar. Subtype ise bir veri tipinin kısıtlanmış halini yeni bir veri tipi olarak tanımlamak için kullanılır. Örnek tip tanımları;

TYPE agirliklar_h IS ARRAY (1 to 3) OF std_logic_vector(16 downto 0);

SUBTYPE toplam IS std_logic_vector(15 downto 0);

4.2.1.4.1. Ön tanımlamalı veri tipleri

1. Standart Lojik Tip : STD_LOGIC(0 ve 1) , STD_LOGIC_VECTOR(bit dizisi)
2. Integer (tamsayi tip): INTEGER
3. Floating Point (kayan noktali sayi): REAL
4. Physical (fiziksel) Tip : TIME
5. Enumeration (Liste) Tip : BOOLEAN ,CHARACTER

VHDL kütüphane dosyalarına dayanan bir yapıya sahiptir. Her bir kütüphane dosyası aslında birer paket dosyadırlar ve işlem, değişken tipi ve fonksiyon kullanmak için program öncesinde çağırılmalı ve projeye eklenmelidirler.

4.2.1.5. VHDL temel yapısal elemanları

VHDL kodu temel olarak en az üç bölümden oluşmalıdır. Bu bölümler Kütüphane (Library), varlık (Entity) ve Mimari (Architecture)'dir.

4.2.1.5.1. Kütüphane (Library) yapısı

Her *entity* başlangıcında bu devre içinde kullanılacak olan programlar ve tip tanımlamaları paket programlar ile *entity* içinden çağırılmalıdır. Paketler genelde kütüphane içinde tanımlanırlar. Kütüphane tanımlamak için en az iki kod satırına ihtiyaç duyulur. İlk kod satırı kütüphanenin ismini içerirken ikinci kod satırı paket ismini içerir. Kütüphane tanımının genel yapısı aşağıdaki gibidir.

LIBRARY kütüphane_adı;

USE kütüphane_adı.paket_ismi.paket_bolumu;

Bir tasarımda genellikle *ieee*, *std* ve *work* kütüphaneleri kullanılır. *ieee* ve *std* kütüphaneleri vhdl diline ait olan kütüphanelerdir ve bir kısmı vhdl yüklenmesi ile beraber gelirken bir kısmı ise internetten indirilebilmektedir. *ieee* kütüphanesinde matematiksel işlemler, numerik işlemler, text işlemleri paketleri yer alırken *std* kütüphanesinde dosya okuma fonksiyonları ve standart değişkenlerin bulunduğu paketler yer alır. *work* kütüphanesi ise kullanıcı tarafından yazılmış olan paketleri içerir.

4.2.1.5.2. Varlık (Entity) tanımlanması

Entity alanı bir devrenin giriş ve çekişlerinin tanımlandığı bir alandır. Entity alanında yapılan port tanımlamasının dört değişik durumu olabilir.

- IN : Devreye giriş sağlayan (okunabilen) port bilesen biçimidir.
- OUT : Devreden çıkış sağlayan (yazılabilen) port bilesen biçimidir.
- INOUT: Hem giriş hem de çıkış olabilen port bilesen biçimidir.
- BUFFER: Hem giriş hem de çıkış olarak kullanılabilen port bilesen biçimidir.

Varlık tanımının genel yapısı aşağıdaki gibidir.

```
ENTITY varlık_adı IS
        PORT (
                Port_ismi : sinyal_modu sinyal_türü;
Port_ismi : sinyal_modu sinyal_türü;
....);
END varlık_adı;
```

4.2.1.5.3. Mimari (Architecture) tanımlanması

Devrenin nasıl davranması gerektiğinin belirlendiği alan *architecture* alanıdır. Bir devreye istenirse birden fazla *architecture* yazılıp bunlardan bir tanesinin seçilmesi durumu söz konusu olabilmektedir. Fakat genelde her bir devrenin bir adet davranış bicimi olur. Genel formu aşağıdaki gibidir.

```
ARCHITECTURE mimari _isim OF varlık_ismi IS
   Sinyal Tanımlanması ;
   Sabit Tanımlanması ;
   Tip Tanımlanması ;
   Parça Tanımlanması ;
BEGIN
   Rastlantısal Görev ifadeleri ;
   İşlem ifadeleri ;
END mimari_isim ;
```

4.2.1.5.4. Alt Programlar (Fonksiyon ve Prosedür) tanımlanması

Altprogramlar VHDL' nin önemli yapılarındandır. Program içerisinde tekrar tekrar çağrılabilirler. VHDL dili '*procedure*' ve '*function*' olmak üzere iki tür alt program yapısını destekler. Birinci tür altprogramlar dönüş değeri almazlar. Altprogram tanımlamaları sadece 'interface' bilgisini içerir. Altprogram yapısı ise '*interface*' bilgisi, yerel tanımlamalar ve ifadeleri içerir. Altprogram tanımlaması ile altprogram yapısı arasındaki fark, '*entity*' bildirimi ve '*architecture*' yapısı arasındaki fark gibidir. Altprogram bildirimleri aşağıdaki gibi yapılır. Örnek:

PROCEDURE kimlik interface listesi

FUNCTION kimlik interface listesi RETURN dönüş türü

Bu bildirimlerde, 'interface' listelerinin olma zorunluluğu bulunmamaktadır. Yani altprogramlar parametresiz olabilir. Altprogram tanımlamaları aşağıda ifade edildiği gibi yapılabilir. Her iki altprogram türü için aynı yapı geçerlidir. Genel yapıları aşağıdaki gibidir.

Altprogram özelliği IS

Bildirimler

BEGIN

İfadeler;

END kimlik;

4.2.2. Verilerin gösterimi

FPGA giriş olarak sadece dijital girişleri kabul etmektedir. Yapılan bu tez çalışmasında ilk olarak analog verilerin ADC devresi tasarlanarak dijital hale dönüştürülmüştür. Bu dijital veriler üzerinde FPGA içinde kullanılabilen iki çeşit sayı formatı verdir. Bunlar kayan noktalı sayı format ve sabit noktalı sayı formatıdır. FPGA içinde yapılan uygulamalar bu iki sayı tipine göre yapılmaktadır.

4.2.2.1. Kayan noktalı sayılar

Kayan noktalı sayı aritmetiği gösteriminde sayılar dört bölümden oluşur: işaret, üs e, üs tabanı b ve anlamlı kısım t. Bu bölümler bir sayıyı Denklem 4.1 eşitliğinde olduğu gibi temsil eder.

$$x = \mp s \times b^e \tag{4.1}$$

Burada dikkat edilmesi gereken kayan noktalı gösterimde iki tane işaretin olduğudur. Bunlardan biri sayının işareti, diğeri ise üssün işaretidir. Üssün işareti ile uğraşmamak için genelde yanlı (biased) gösterim kullanılır. Böylece üs belirli bir sayıyla toplanarak, üssün negatif olması engellenir. Şekil 4.6'da kayan noktalı gösterimin en genel hali görülmektedir.



Şekil 4.6. Kayan noktalı sayı gösterimi

Kayan nokta gösteriminde kullanılan standart "ANSI/IEEE Std 754-1985" standardıdır. Bu standartta üs yanlı biçimde ifade edilirken, anlamlı kısım sabit bir '1' ile başlar ve bu bite saklı bit denir. Anlamlı kısım *s, 1.f* şeklinde gösterilir ve f'nin bir diğer ismi de mantistir. Tek duyarlı gösterim biçimi 32 bit uzunluğunda, çift duyarlı gösterim biçimi ise 64 bit uzunluğundadır. Bu gösterimdeki kayan noktalı sayıların on tabanında temsil ettiği değer Denklem 4.2'de verilmiştir.

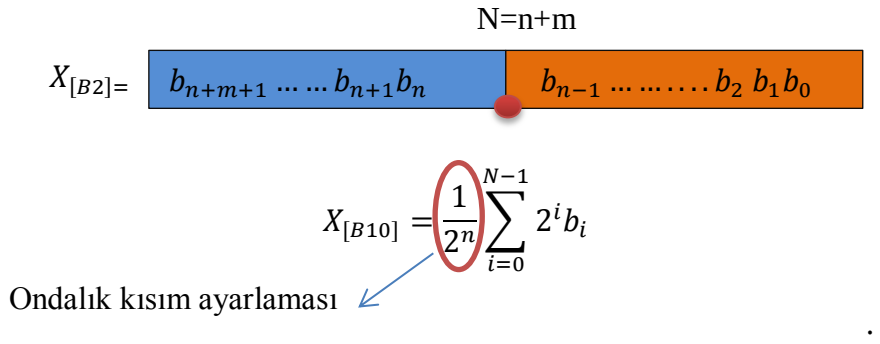
$$x = (-1)^{isaret} \times 2^{e-yanlılık} \times 1.f \tag{4.2}$$

Kayan noktalı sayılar hassasiyetleri sebebiyle tercih edilirler. Fakat donanım kaynaklarını sabit noktalı sayılara göre daha fazla tüketirler.

4.2.2.2. Sabit Noktalı Sayı Gösterimi

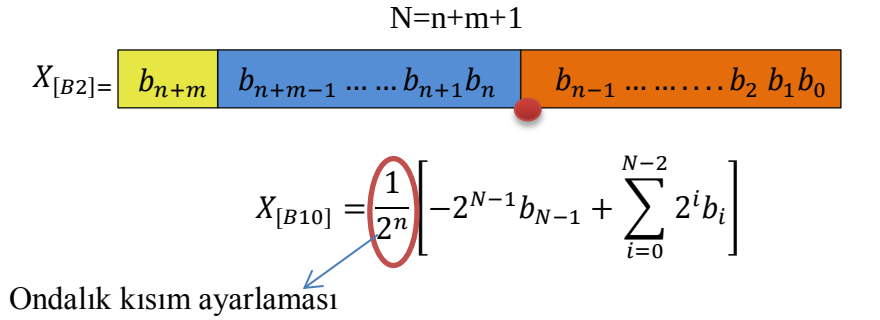
Sabit noktalı sayı formatı sayının tam kısmı ve ondalık kısmı ile ifade edilir. Bu sayı formatında tasarlanacak sisteme göre, istenilen sayının maksimum büyüklüğü ve hassasiyet göz önünde bulundurularak bir uzunlukları belirlenir. Sabit noktalı sayıların sunumumu için farklı paketler bulunmaktadır. Yapılan çalışmada *fixed_pkg.vhdl* kullanılmıştır. *Ufixed* işaretsiz sayı, *Sfixed* ise pozitif ve negatif veri tipinin ikili gösterimidir. m tamsayı kısmı be n ondalık kısım uzunluğu olmak üzere Qm.n

formatında işaretsiz bir sabit noktalı ve onluk karşılığının gösterimi Şekil 4.7'da verilmiştir.



Şekil 4.7. İşaretsiz sabit noktalı sayı formatı

m tamsayı kısmı be n ondalık kısım uzunluğu olmak üzere Qm.n formatında işaretli bir sabit noktalı ve onluk karşılığının gösterimi Şekil 4.8'de verilmiştir.



Şekil 4.8. İşaretli sabit noktalı sayı formatı

Yapılan tez çalışmasında gerçekleştirilecek kontrol sistemi için yeterli hassasiyeti sağlaması ve daha az FPGA alanı gerektirmesinden dolayı sabit noktalı sayı formatı kullanılmıştır.

4.3. Elektromanyetik Filtre

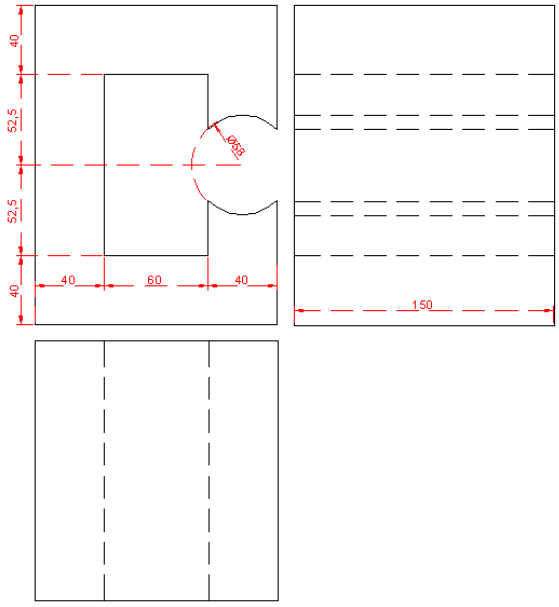
Bu tez çalışmasında selenoit tip bir EMF kullanılmıştır. EMF temel olarak

1. Manyetik nüve,
2. Filtre matrisi,
3. Filtre bobini

olmak üzere üç ana bölümden oluşmaktadır.

4.3.1. Elektromanyetik nüve

Elektromanyetik nüvenin gövdesi DC akım kullanılacağından dolayı *μr=1045* olan yekpare demir ile oluşturulmuştur. Abbasov (2002) tarafından verilen EMF tasarım denklemleri kullanılarak uygun elektromanyetik nüve oluşturulmuştur. Elektromanyetik nüvenin hesaplanması ile ilgili ayrıntılı bilgi Tübitak tarafından desteklenen 109E037 nolu projede verilmiştir (Herdem, 2011) . Elektromanyetik nüvede *H=20kA/m, l=150 mm, NI= 1,8 kA sarım, N=450 sarım, I=4 A, R=5 Ω* ve *D=1,6 mmø* değerleri kullanılmıştır (Herdem, 2011). Elektromanyetik nüvenin teknik resmi Şekil 4.9'de verilmiştir. Elde edilen elektromanyetik nüve bilgisayarla kontrol çalışmalarında kullanılmış, EMF kitine uygun hale getirmek için 1/3 oranında küçültülerek yeni bir manyetik nüve elde edilmiştir.



Şekil 4.9. Manyetik nüve teknik resmi (üç görünüş)

4.3.2. Filtre matrisi

Filtre matrisi olarak 14 *mm* çapında demir (*St1040*) küreler kullanılmıştır. İki elektromanyetik nüve için ayrı ayrı filtre matrisleri tasarlanmıştır. Birinci filtre matrisi için 57 *mm* çapında 150 *mm* uzunluğunda ve 2 *mm* kalınlığında mika boru kullanılmıştır. İkinci filtre matrisi için 40 *mm* çapında 100 *mm* uzunluğunda 2 *mm* kalınlığında mika boru kullanılmıştır. Filtre matrisinin her iki tarafında 2 *mm*

kalınlığında ortası delinerek küreleri tutan süzgeçler sabitlenmiş, giriş ve çıkışlarına sarı pirinçten ½" parmak musluk ucu bulunmaktadır. Kullanılan filtre matrisi Şekil 4.10'da verilmiştir.

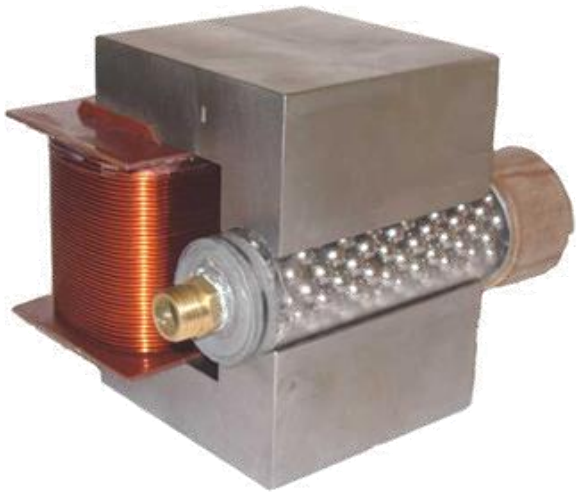


Şekil 4.10. Filtre matrisi

4.3.3. Filtre bobini

Filtre bobini için Abbasov (2002) literatüründe verilen hesaplamalara uygun olacak bir şekilde filtre bobini tasarımı yapılmıştır. Filtre bobini hesaplanması ile ilgili ayrıntılı bilgi Tübitak tarafından desteklenen 109E037 nolu projede verilmiştir (Herdem, 2011). Bobin teli çapı 2.30 mm, sarım sayısı 450 sarım, omik direnci 1.3 Ω ve endüktansı 28 *mH* olan bir filtre bobini EMF kitinde kullanılmıştır.

EMF kitinde kullanılan elektromanyetik filtrenin nüve, bobin ve matrisinin görünümü Şekil 4.11'de verilmiştir.



Şekil 4.11. EMF kitinden kullanılan manyetik filtrenin nüve, bobin ve matrisi

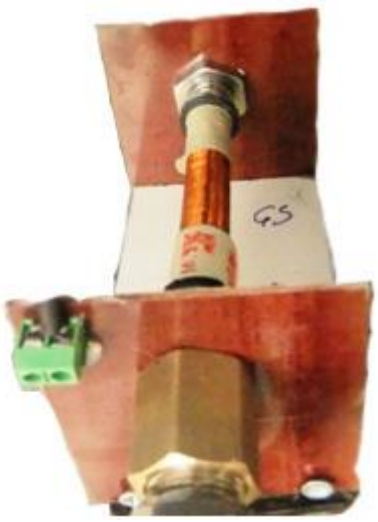
4.4. Tasarımı Yapılan Elektronik Devre ve Bileşenler

Bu tez çalışmasında, Endüstriyel sıvı içerinde bulunan parçacık konsantrasyonunu belirlemek için sensörler, sistem içerisindeki sensörleri beslemek ve enstrümantasyon yükseltecinden geçirmek için sensör devreleri, sensör devrelerinden alınan bilgileri dijital hale çevirmek ve FPGA üzerinden alınan dijital verileri analog hale çevirmek için ADC/DAC arayüz kartı tasarlanmış ve gerçekleştirilmiştir. Bu bölümde tasarımı yapılan elektronik devreler verilmiştir.

4.4.1. Parçacık konsantrasyon sensör bobinleri

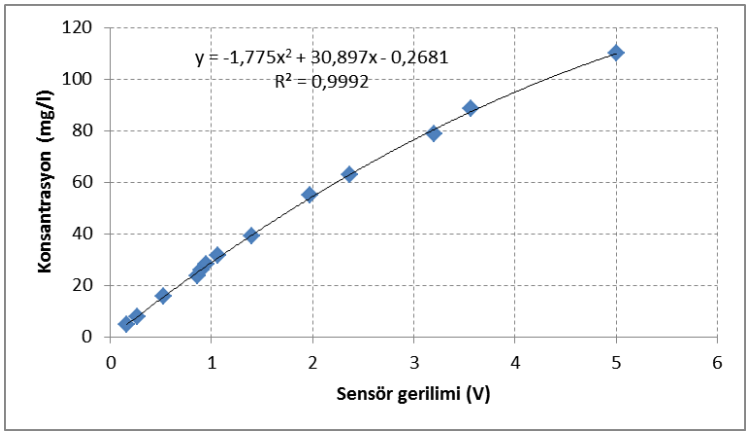
Filtre giriş ve çıkışındaki parçacık konsantrasyonlarını ölçmek için tasarlanmıştır. Parçacık konsantrasyon sensörü, içinden sıvı ve süspansiyon geçtiğinde sıvının içerisindeki manyetik parçacık miktarını ölçer ve bu ölçülen değerleri kontrolöre gönderir. Bu amaçla endüktif tip bir parçacık sensörü kullanılmıştır.

Sensör için, manyetik olmayan 5 *mm* çaplı, 12 *cm* uzunluğundaki PVC bir boru üzerine 10.9 *mH* ve 58 *Ω*'luk bir dirence sahip olan 0,10 *mm* çaplı 4000 sarımlık emaye yalıtımlı telden bobinler oluşmaktadır. EMF'nin farklı endüstriyel alanlarda çalışılabilmesi için yüksek basınçlarda *(<16 bar*) çalışabilen, uçlarına 3/8" pnömatik malzeme ile bağlantıları yapılmıştır. (Şekil 4.12). Parçacık konsantrasyon sensörü filtre girişi ve çıkışında olmak üzere iki adet kullanılmıştır.



Şekil 4.12. Prototip için geliştirilen parçacık konsantrasyon sensör bobini

Hassas terazi ile yapılan demir tozu ölçümleri ile 10 µg-1 mg arasında demir tozları ince ve yalıtkan PVC içerisine yine yalıtkan bir yapıştırıcı yardımıyla yapıştırılarak kalibrasyon için hazırlanmıştır. Bu demir tozları ile tasarlanan parçacık sensörü deneyleri yapılmıştır. Buna göre 0-1,5 V arasında gerilim çıkışı elde edilmiştir. Parçacık sensörünün konsantrasyon miktarına göre ürettiği gerilim farkı grafiği Şekil 4.13'da gösterilmiştir.



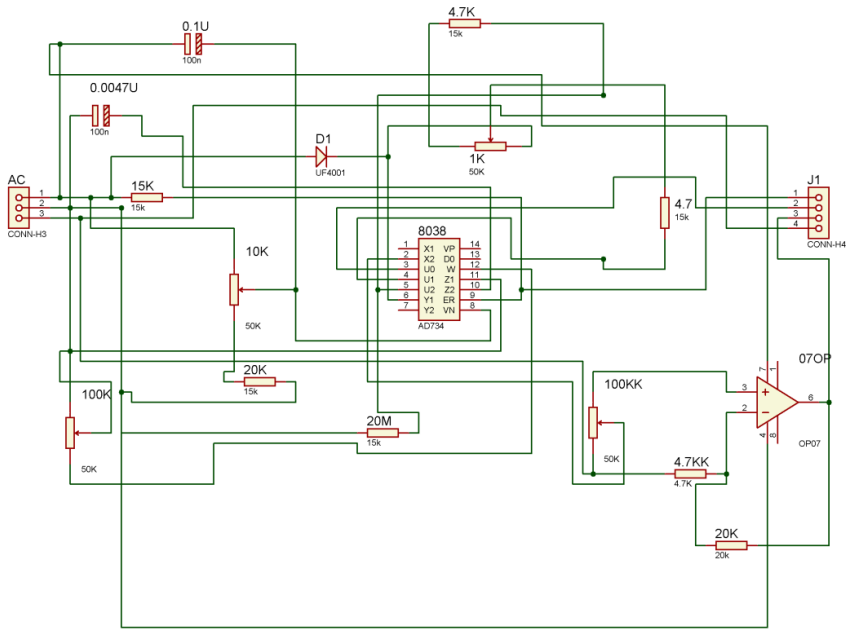
Şekil 4.13. Parçacık sensörü çalışma grafiği

Kalibre edilen sensör için grafikten bir eğri uydurulmuş ve Denklem 4.3 ampirik formülü elde edilmiştir.

$$y = -1{,}775x^2 + 30{,}897x - 0{,}2681 \qquad (4.3)$$

4.4.2. Yüksek frekanslı sinüs güç kaynağı

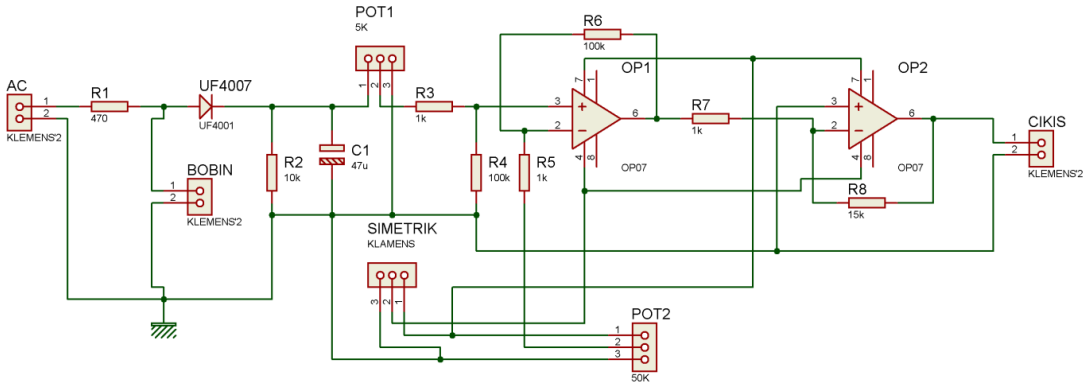
Sensör bobinlerine yüksek frekanslı sinüs gerilimi uygulanması için sinüs üreteç kullanılmıştır. Bunun için ICL8038 entegresi kullanılmıştır. 8038 entegresi ile yüksek frekanslı bir AC üreteci için Şekil 4.14'teki 20 Hz ile 20 kHz'lik osilatör devresi kullanılmıştır.



Şekil 4.14. Sinüs sinyal üretmek için kullanılan devre şeması

4.4.3. Parçacık konsantrasyon yükselteç devresi

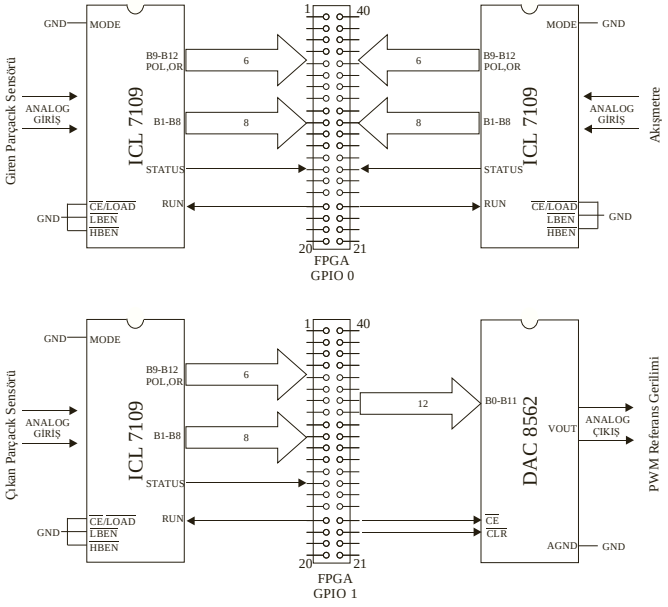
Giren ve çıkan parçacık konsantrasyon sensörü olarak gerçekleştirilen bobinler için sinyal işleme ve yükseltme devresi ve 9 *KHz*, 2 *V* AC giriş ile beslenmektedir. Parçacık konsantrasyon sensöründen geçen manyetik parçacıkların oluşturduğu değişimleri algılayabilmek için opamp fark yükselteci olarak kullanılmaktadır. Devre dengede iken parçacık konsanstrasyon sensör bobinlerinden geçen manyetik parçacıklar dengeyi bozmakta ve oluşan fark yükseltilerek çıkışa iletilmektedir. Parçacık konsantrasyonu yükselteç devresi Şekil 4.15'de verilmiştir.



Şekil 4.15. Parçacık konsantrasyonu yükselteç devresinin şeması

4.4.4. ADC/DAC kartı

FPGA ile kontrol devresi arasındaki analog sayısal ve sayısal analog dönüştürme birimidir. Sensör devrelerinden gelen sinyallerin sayısala dönüştürülmesi amacıyla üzerinde üç adet ICL7109 entegresi bulunduran ve FPGA'den alınan sayısal veriyi analog hale dönüştürmek için DAC8562 entegresini barındıran ADC/DAC kartının tasarımı yapılmıştır. ICL7109 entegresi ile kullanılan EMF kitinin çalışma yapısı ve hızı için yeterli bir süre olan saniyede 30 çevirime kadar dönüşüm yapılabilmektedir ve entegre tam zamanlı paralel çevirici olarak çalıştırılmıştır. Sensörlerden gelen analog veriler 12 bitlik sayısal verilere dönüştürülerek FPGA'ye aktarılmaktadır. Ayrıca sayısal verinin durum, taşma ve işaret bitleri de FPGA'ye iletilmektedir. Güç ünitesine gidecek olan PWM sinyal referansı FPGA'den sayısal olarak kart üzerinde analog veriye dönüştürülerek PWM kontrolü sağlanmaktadır. FPGA ile kart arasındaki bağlantı iki adet 40 pinlik IDE konnektör ile sağlanmıştır. Şekil 4.16'da ADC/DAC kartın devre şeması verilmiştir.

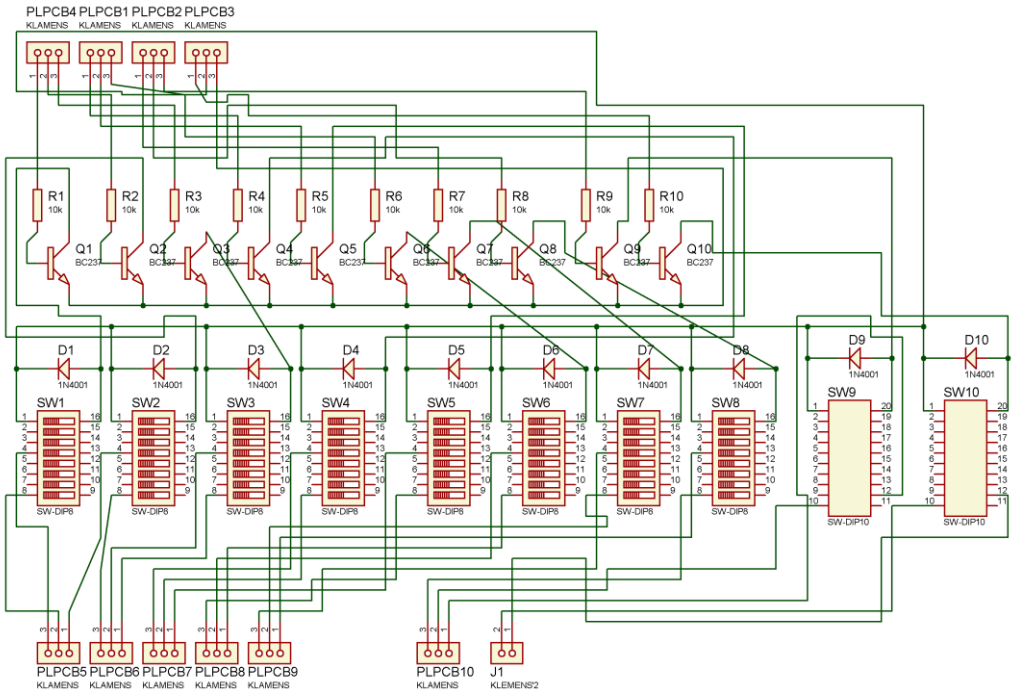


Şekil 4.16. ADC/DAC kartı devre şeması

A/D dönüşümünde referans 5 *Volt* uygulanmıştır. 12 Bit bir A/D çevirici kullandığından dolayı hassasiyet 1/4096 birim olur. A/D dönüşümünde 5000/4096 = 1.22 *mV*'luk bir çözünürlük elde edilmiştir.

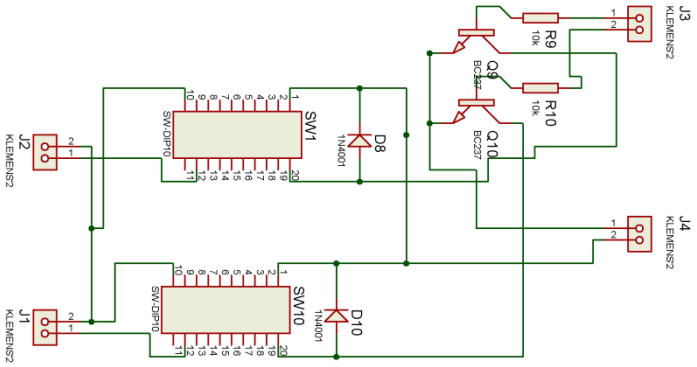
4.4.5. Röle kartları

Namur valfleri, invertörü ve EMF bobinlerini FPGA tabanlı kontrolörlerden gelen sinyallerle kontrol etmek için bir röle kartı kullanılmaktadır. Namur valflerin selenoid valfleri 24 V DC ile çalışmaktadır. Sekiz adet elektropnömatik valfi kontrol etmek için HK19F-DC röle kullanılmıştır. EMF bobinlerini 200 V'luk kontaktörlerle kontrol edildiğinden iki adet EMF bobini için EKE H8-V röleler kullanılmıştır. Röleleri sürmek için BC237 transistorler kullanılmıştır ve röle bobinlerinin beslemesi için ise 12 V DC kullanılmıştır. Röle kartı baskı devre şeması Şekil 4.17'de verilmiştir.



Şekil 4.17. Röle kartı devre şeması

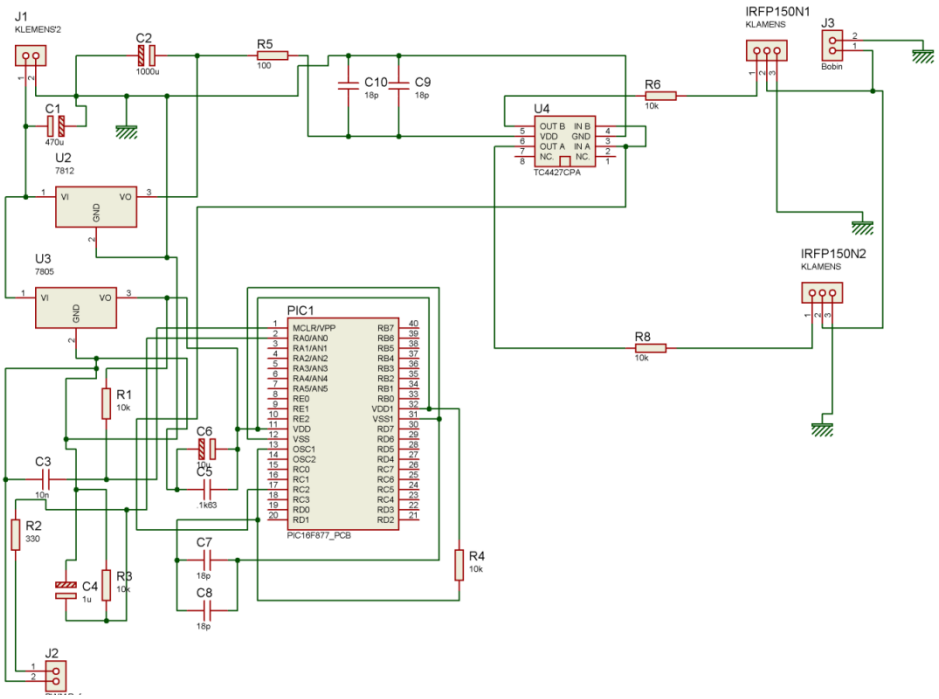
Filtrenin temizlenmesi işleminde artık mıknatısiyetin sıfıra indiği anlarda kopma işlemini gerçekleştirmek amacıyla filtre bobine düşük frekanslı AC gerilim uygulamak amacıyla tasarlanmıştır. Sadece filtre temizleme işlemi sırasında invertörü kontrol etmek amacıyla röle kartı devresi tasarlanmıştır (Şekil 4.18).



Şekil 4.18. İnvertör için röle kartı devre şeması

4.4.6. PWM sürücü kartı

PWM kontrol ile filtre bobinine uygulanacak olan gerilimin kontrolü dolayısıyla bobinden geçen akımın kontrolü sağlanmıştır. Akımın kontrolü ile filtre matrisinden geçen manyetik akı kontrol edilmiştir. PWM için PIC 16F877 ile gerçekleştirilen PWM ile mosfet sürücü devre kullanılmıştır. PWM sürücü kartının devre şeması Şekil 4.19'de verilmiştir.



Şekil 4.19. PWM devre şeması

4.5. EMF Kiti Gövde Tasarımı

Elektromanyetik filtre deney düzeneğinin endüstriyel alanda kullanımına uygun duruma getirilebilmesi için deney düzeneğinde bulunan manyetik filtreler, pnömatik bileşenler ve elektromekanik aksamları kapsayacak şekilde ve uygun ebatlarda bir prototip gövde kullanılmaktadır. EMF kiti gövdesi Elektro Manyetik Filtre sisteminin etkilememesi amacıyla gövde manyetik özelliği olmayan malzemeden (*INOX*) oluşmaktadır.

EMF kiti gövdesi üç bölümden oluşmaktadır. Üst kısımdaki bölme elektronik kumanda elemanlarının yerleşimi için, orta bölme filtreleme elemanları ve namur valflerin yerleşimi için, alt bölme ise atık deposu, akışmetre, güç ünitesi ve giriş-çıkışlar için ayrılmıştır.

Orta bölümünde iki adet manyetik filtre bobini ve matrisi ile birlikte giriş ve çıkış olmak üzere prototip için geliştirilen parçacık konsantrasyon sensörü ve sekiz adet namur valf bulunmaktadır. Bu elemanların alt ve üst bölümler ile elektriksel, pnömatik ve akışkan bağlantıları vardır.

Üst bölümünde FPGA, kontrol için gerçekleştirilen ADC/DAC kartı, akışmetre devresi, röle kontrol kartları, sensörlerin veri kartları, sigorta, paket şalter, bir adet AC bir adet DC voltmetre, bir adet DC ampermetre, sinyal lambaları, 24V DC ve 12 V DC besleme üniteleri, kontaktör ve tüm elektriksel bağlantılar için klemens grubu bulunmaktadır. Tüm bu elemanların diğer bölümler ile bağlantıları vardır.

EMF kitinin tüm bölümlerinin görünüşü Şekil 4.20'de verilmiştir.

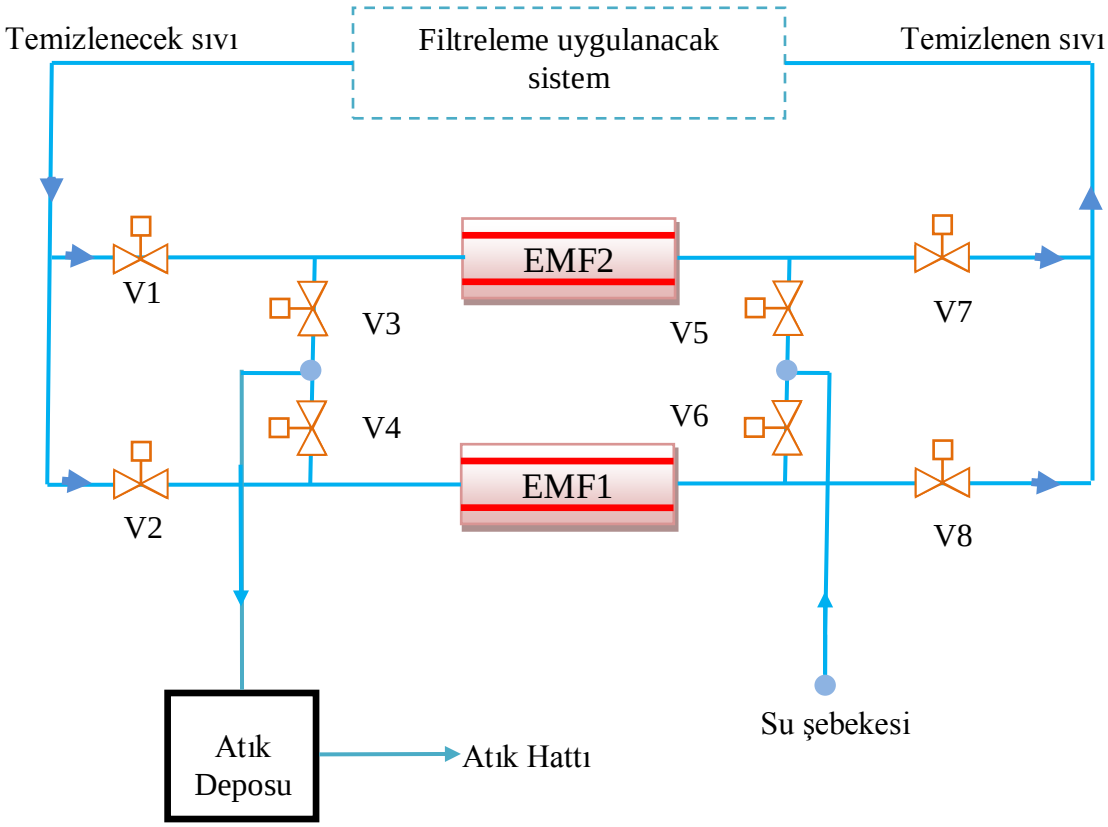


Şekil 4.20. EMF kiti

4.6. EMF kitinin genel çalışma yapısı

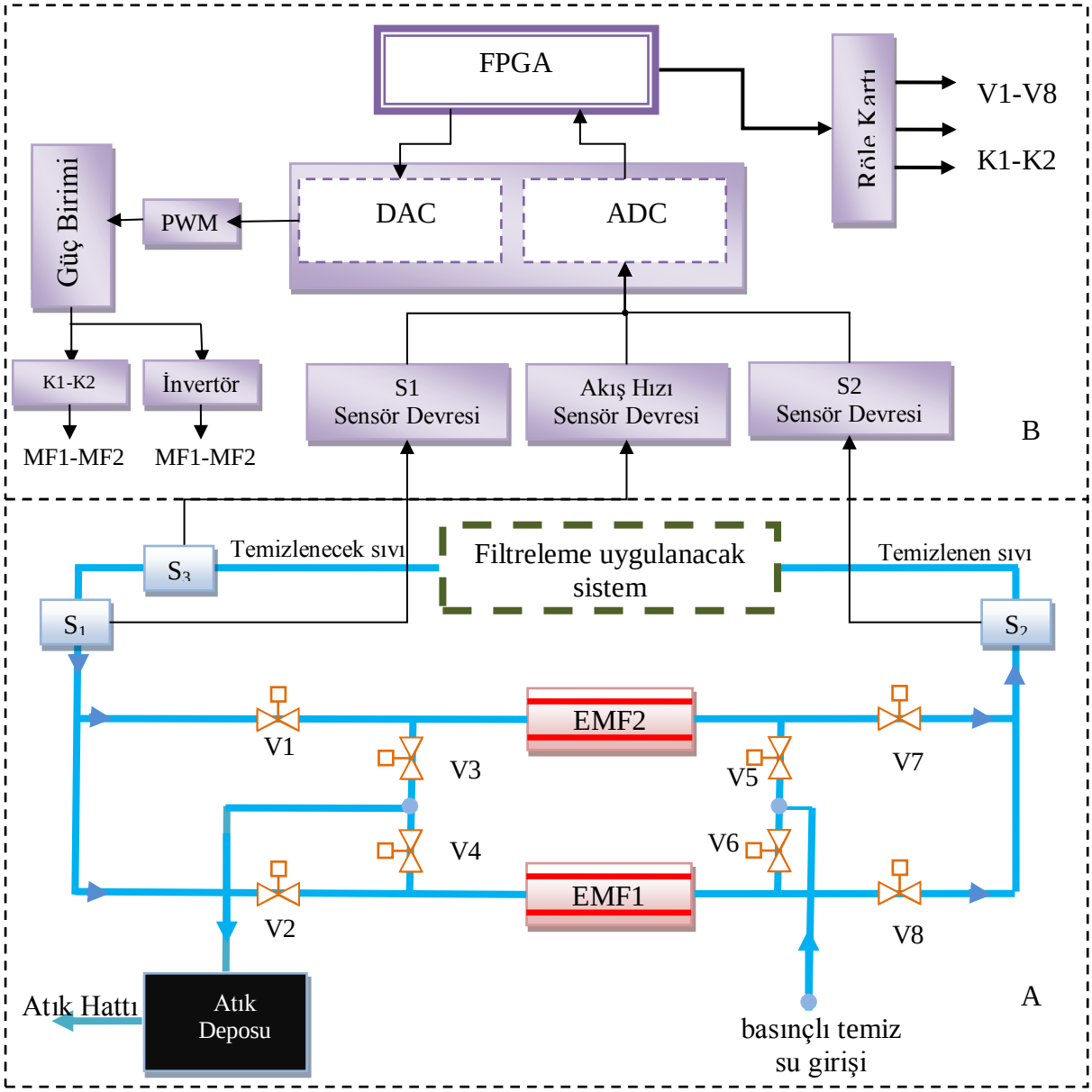
EMF kitinin güç kaynağı, gövde, bobin, manyetik filtre elemanı, süspansiyon karıştırıcı, pompa, kontrol duyargaları, taşıyıcı borular, vanalar ve depolayıcılar bileşenlerinden oluşturulan filtrenin akış diyagramı Şekil 4.21'da verilmiştir.

Temizlenen sıvı filtreleme uygulanan sisteme geri verilmektedir. EMF1 aktif iken V2-V8 açık, diğer elektropnömatik valfler kapalı konumda, EMF2 aktif iken V1-V7 açık, diğer namur valfler kapalı konumdadır. Aktif EMF'nin filtreleme performansı düştüğünde pasif konuma, pasif olan diğer EMF ise aktif konuma geçmektedir.



Şekil 4.21. Oluşturulan filtrenin akış şeması

EMF kitinin kontrolü için FPGA ve çevre birimlerinin de etkisi Şekil 4.22'de gösterilmiştir.



Şekil 4.22. Manyetik filtre kontrol sisteminin çalışma prensibi

Şekil 4.22'de A kısmında görülen sistem üzerindeki fiziksel yapı filtreleme uygulanacak sistemden gelen endüstriyel sıvının EMF'de temizlendikten sonra çıkışa aktarılması ile özetlenebilir. Temizlenecek sıvı, sırası ile akış hızı (AH) sensörü (S3), giren parçacık konsantrasyonu (GPK) sensörü (S1), aktif EMF ve çıkan parçacık konsantrasyon (ÇPK) sensöründen (S2) geçerek filtrasyon işlemine tabi tutulmaktadır. Temizlenen sıvı filtreleme uygulanan sisteme geri verilmektedir. Aktif EMF'nin filtreleme performansı düştüğünde pasif konuma, pasif olan diğer EMF ise aktif konuma geçmektedir.

Pasif konuma geçen filtrenin tekrar kullanıma hazırlanabilmesi için filtre matrisini oluşturan elemanlardaki artık mıknatısiyetin ortadan kaldırılması ve tutma bölgelerinde biriken manyetik özellikli parçacıkların matristen uzaklaştırılması gerekmektedir. Bunun için pasif konuma geçen filtre bobinine invertör aracılığı ile düşük frekanslı AC gerilim uygulanırken aynı zamanda uygun elektropnömatik valfler (MF1 için V4-V6, MF2 için V3-V5) ile EMF'nin matrisine basınçlı su uygulanmaktadır. Bu uygulama ile filtre matrisini oluşturan küreciklerin tutma bölgelerinde biriken manyetik özellikli parçacıklar koparılarak atık deposuna aktarılmaktadır. Filtre matrisinin temizleme işlemi bittikten sonra kontrolör tarafından bu işlemler sonlandırılmakta ve pasif olan filtre tekrar kullanım için bekleme konumuna alınmaktadır.

Şekil 4.22'de B kısmında filtreleme sisteminin kontrol yapısı görülmektedir. S1,S2 ve S3 sensörlerinden elde edilen sinyaller, alınan verileri enstrümantasyon yükselteci işlemine tabii tutmak için tasarlanan sensör devrelerine aktarılmaktadır. Sensör devrelerinden alınan analog bilgiler paralel çıkışlı ICL7109 entegresi kullanılarak gerçekleştirilen ADC arayüz kartı ile dijital hale çevrilerek FPGA'ye iletilmektedir. FPGA üzerinde çalışan kontrol sistemi çıkışı ile manyetik akının kontrolü yapılmaktadır. Bu amaçla FPGA üzerinden alınan dijital verileri analog hale çevirmek için DAC8562 entegresi ile tasarlanıp gerçekleştirilen DAC arayüz kartı kullanılmaktadır. FPGA ile oluşturulan kontrolörün çıktısı olan PWM referans gerilimi yine DAC kartı ile PWM devresine analog olarak iletilmektedir. Referans gerilimine bağlı olarak PWM ile filtre bobinine uygulanacak olan gerilim kontrol edilmiştir. Bobindeki gerilimin değişimi ile EMF'nin manyetik akısının kontrolü sağlanmaktadır.

5. KULLANILAN YAPAY ZEKA TEKNİKLERİ

Yapay zeka (YZ), bir bilgisayarın yada bilgisayar denetimli bir makinenin, genellikle insana özgü nitelikler olduğu varsayılan akıl yürütme, anlam çıkartma, genelleme ve geçmiş deneyimlerden öğrenme gibi zihinsel süreçlere ilişkin süreçleri yerine getirme yeteneği olarak tanımlanmaktadır (Nabiyev, 2005).

YZ, insan tarafından yapıldığında zeka olarak adlandırılan davranışların (akıllı davranışların) makine tarafından da yapılmasıdır. Yapay zekanın insan aklının nasıl çalıştığını gösteren bir kuram olduğu da söylenebilir (Russell, 1995; Haykin, 1998; Munakata, 1998; Nilssons, 1998).

Bilgisayar teknolojisindeki gelişime paralel olarak özellikle son yirmi yılda hemen hemen her alanda karşımıza çıkan ve sezgisel olarak çözülebilen ya da çözülmesi matematik teknikler ile mümkün olmayan gerçek hayat problemlerini çözmeye yönelik teknikler YZ teknikleri olarak adlandırılır. Bunların başlıcaları şunlardır:

a-) Bilgi tabanlı uzman sistem yaklaşımı

b-) Yapay sinir ağları yaklaşımı

c-) Bulanık mantık yaklaşımı

d-) Geleneksel olmayan optimizasyon teknikleri

i-) Genetik algoritma

ii-) Benzetilmiş tavlama (*Simulated annealing*)

iii-) Hibrit algoritmalar

e-) Esnek programlama (*Soft computing*)

f-) Yapay bağışıklık sistemi yaklaşımı

Belirtilen yapay zeka tekniklerinin çoğunun kökeni insanların veya hayvanların davranışsal olaylarına veya organlarının basit halleri olarak karşımıza çıkmaktadır. Hibrit zeki sistemler ise iki, üç yada daha fazla yapay zeka tekniğinin avantajlı sonuçlar üretecek şekilde bir araya gelmesi ile oluşmaktadır. Bu tez çalışmasında yapay sinir ağları, bulanık mantık ve hibrit yapıları kullanılmıştır.

Yapay sinir ağları (YSA) ve bulanık çıkarım sistemleri belirsiz durumlarda çalışabilen iki ayrı yöntemi temsil etmektedirler ve sahip oldukları avantajlardan dolayı değişik bilimsel ve mühendislik uygulamalarında giderek daha cazip bir hale gelmişlerdir. Bunun temel sebebi, geleneksel çözüm yollarına kıyasla belirli bir kesinlik toleransına sahip olmaları ve parçalı doğrular ile çalışabilme yetenekleridir (Mendi, 2002).

Bulanık Mantık (BM), denetim ve bilgi süreçlerinin birçoğu için güçlü bir problem çözme yöntemidir ve kesin olmayan bulanık bilgiden dikkate alınacak kadar basit bir şekilde kesin sonuçlar elde edilmesine olanak sağlar (Zadeh, 1965; Wang 1993).

YSA'nın en önemli özelliği öğrenebilme yeteneğidir. YSA kullanan sistemlerin başlıca belirgin özellikleri algoritmasız, tamamıyla paralel, adaptif, öğrenebilen ve paralel dağıtılmış bir hafızaya sahip olmalarıdır. BM ve YSA'nın birbirlerini tamamlayıcı özelliklerinden faydalanarak birçok uygulama gerçekleştirilmiştir (Wang, 1997; Chen, 1994, Buja, 1994).

5.1. Bulanık Mantık

Bulanık kümelere dayalı olan Bulanık Mantık genelde insan düşüncesine özdeş işlemlerin gerçekleşmesini sağlamak, gerçek dünyada sık sık meydana gelen belirsiz ve kesin olmayan verileri modellemede yardımcı olmaktadır Klasik mantıkta bir önerme "doğru" veya yanlıştır". Fakat gerçek dünyadaki olayların ne derecede iyi veya yanlış olmasının belirlenmesi gerekmektedir. Örneğin 100 ^{o}C suyun sıcaklığı "sıcak" olarak ifade edilirse 95 ^{o}C, 80 ^{o}C'lerdeki su için "sıcak değildir" ifadesi bu anlamda doğru olmadığı gibi yanlış da değildir. Bu nedenle önermelerin doğru ve yanlış değerleri arasındaki değerler (az sıcak, ılık, az soğuk, vs.) kullanılarak bulanık küme kavramı ortaya atılmıştır Bulanık küme teorisi az, sık, orta, düşük çok, birçok gibi dilbilimsel yapılan kullanarak dereceli veri modellemesini gerçekleştirmektedir. Böylece olayların modellenmesinde daha gerçekçi ve doğala yakın sonuçların elde edilmesini sağlar (Nabiyev, 2005).

Literatürde bulanık çıkarım sistem teorisi için iki tür gerekçelendirme vardır (Wang, 1997). Bunlar;

- Gerçek dünyada kesin tanımlamaları elde etmek oldukça karmaşıktır, bu nedenle uygun bir model elde etmek amacıyla üretilmiştir.
- Günümüz bilgi çağında, insan bilgisi daha önemli hale gelmektedir. Sistematik bir şekilde bu insan bilgisinin, sensör ölçümleri ve matematiksel ölçümleri ile birlikte mühendislik sistemine aktarmada bir teori gerekmektedir.

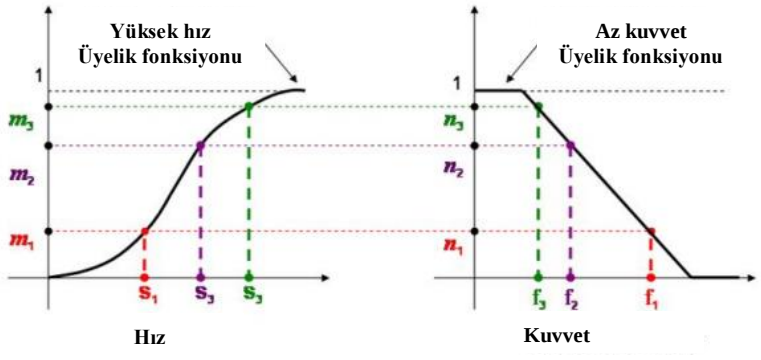
Yukarıda verilen iki gerekçe bulanık sistem teorisinin benzersiz özelliklerini karakterize etmekte ve mühendislik alanlarındaki bağımsız bir dal olan bulanık sistemlerin varlığını haklı çıkarmaktadır.

Birçok pratik sistemde önemli bilgilerin iki türlü kaynağı vardır: bu kaynaklardan ilki, uzman kişiler tarafından sistemin doğal dile tanımlanan bilgisidir. İkinci kaynak ise, fizik kanunlarına göre elde edilen sensör ölçümleri veya matematiksel modellerdir. Karşılaşılan önemli problemlerden birisi de sistem tasarımında bu iki bilgi türünü birleştirebilmektir. Bu birleşimi sağlamada anahtar soru insan bilgisinin nasıl formüle edileceğidir. Temel olarak bulanık sistemler bu dönüşümü gerçekleştirmektedirler.

Bulanık sayılar, bulanık kümeler ve bulanık EĞER-O HALDE kurallarına dayanan bulanık sistemlerin insan bilgi ve çıkarımını yakalayabilme özelliği nedeniyle araştırmalar için en aktif ve verimli alanlardan birini oluşturmaktadırlar. Ayrıca bulanık sistemlere kural tabanlı sistemler adı da verilmektedir. Bir bulanık EĞER- O HALDE kuralı bazı kelimeleri sürekli üyelik fonksiyonları ile karakterize edilen EĞER-O HALDE ifadeleridir. Örnek olarak bir EĞER-O HALDE kuralı aşağıda verilmiştir (Denklem 5.1).

EĞER arabanın hızı yüksek O HALDE gaza daha az güç uygula. (5.1)

Denklem 5.1'de "*yüksek*" ve "*daha az*" kelimeleri Şekil 5.1'de verilen iki ayrı üyelik fonksiyonu ile ifade edilir. Şekil 5.1'de m_i ve n_i ($i=1,2,3$) üyelik fonksiyonu değerlerini, s_i ve f_i ($i=1,2,3$) araba hızı ve gaz için gücü ifade etmektedir. Farklı kurallar bulanık ifadeleri bulanık olmayan dilsel ifadelere dönüştüren farklı şekillerdeki üyelik fonksiyonlarına sahip olabilirler.



Şekil 5.1. Bulanık *EĞER- O HALDE* kuralların üyelik fonksiyonları

Bulanık kurallar sebep sonuç ilişkisi belirten *EĞER-O HALDE* deyimlerinin dilsel ilişkilerini açıklayan biçimde olabilmektedirler. Genel olarak, bulanık kurallarının elde edilmesinin 4 yöntemi vardır. Bunlardan birincisi uzman deneyim ve bilgisinden, ikincisi uygulayıcının davranışından, üçüncüsü bir sürecin bulanık modelinden, dördüncüsü öğrenme yoluyla elde edilmesidir. Ve bir bulanık sistem tüm bulanık *EĞER-O HALDE* kurallarının birleştirilmesi ile oluşturulmaktadır.

5.1.1. Bulanık kümeler

Bir klasik kümenin kesin sınırları belli olan bir kümedir. 100'den büyük gerçel sayıların klasik kümesi *N* Denklem 5.2 gibi tanımlansın;

$$N = \{x | \ x \ > 100 \}, x \ \in \mathcal{R} \tag{5.2}$$

Burada açıkça kesin sınır 100'dür. Eğer *x* bu değerden büyük ise *N* kümesine aittir. Diğer durumlarda bu kümeye dahil değildir. Klasik kümeler matematik ve bilgisayar bilimleri için önemli bir araç olup farklı uygulamalar için uygun olmasına rağmen, soyut ve belirsiz olma eğilimindeki insan kavram ve düşüncelerinin doğasını tam ve doğru bir şekilde yansıtmayabilirler.

Bulanık kümeler, klasik kümelerin tersine, adından da anlaşılacağı üzere kesin olmayan sınırlara sahip olana kümelerdir. Bu ise, "*dahildir*" veya "*dahil değildir*" türündeki aşamalı bir geçiş yerine, "*su sıcak*" veya "*sıcaklık yüksek*" gibi yaygın olarak kullanılan dilsel ifadelerin modellenmesinde bulanık kümelere esneklik sağlayan üyelik fonksiyonları ile karakterize edilen yumuşak bir geçiş sağlamaktadır.

Eğer *X*, *x* tarafından ifade edilen bir nesneler topluluğu ise, *X*'de verilen bir A bulanık kümesi sıralı çiftler şeklinde aşağıdaki gibi tanımlanır;

$$A = \{(x, \mu_A(x)) | \ x \ \in X\} \tag{5.3}$$

burada $\mu_A(x)$ *A* bulanık kümesinin üyelik fonksiyonu olarak isimlendirilir. Bu üyelik fonksiyonu her bir *X* öğesine 0 ile 1 arasında bir üyelik değeri atamaktadır. $\mu_A(x)$ karakteristik fonksiyonu ifade etmektedir. Bulanık küme, karakteristik fonksiyonu 0 ile

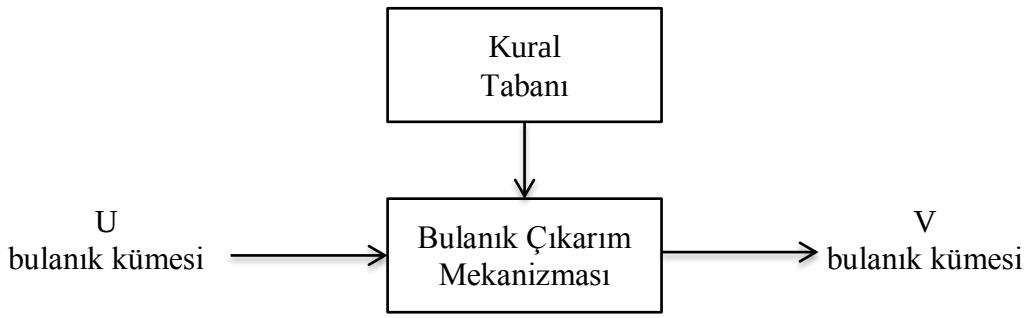
1 arasında herhangi bir değer ile ifade edilebilen klasik kümenin basit bir uzantısı olarak tanımlanır. Genellikle X evreni ifade eder, ayrık nesnelerin veya sürekli alanların birleşimi ile oluşmaktadır.

Öğeler bulanık kümeye kısmi derecede aittir. Klasik kümelerdeki karakteristik fonksiyon, μ_A: $E \rightarrow \{0,1\}$, bulanık kümelerde yerini, μ_A: $E \rightarrow [0,1]$ olarak gösterilen üyelik fonksiyonuna bırakır.

5.1.2. Bulanık çıkarım sistemleri

Pratikte yaygın olarak kullanılan üç adet bulanık çıkarım sistemi bulunmaktadır. Bunlar salt bulanık sistemler, Takagi-Sugeno-Kang (TSK) bulanık sistemler ve bulanıklaştırıcı ve durulaştırıcı içeren bulanık sistemlerdir.

Salt bulanık sistemin temel yapısı Şekil 5.2'de verilmiştir. Bulanık kural tabanı bulanık *EĞER- O HALDE* kurallarının toplanmasıyla oluşur. Bulanık çıkarım mekanizması bu *EĞER-O HALDE* kurallarının birleşimini, U giriş uzayındaki bulanık kümelerden V çıkış uzayındaki bulanık kümelere doğru eşleştirmektedir. Mühendislik sistemlerindeki giriş çıkış değişkenlerinin gerçek değerli olmasına karşın, bu bulanık sistemin giriş ve çıkışlarının bulanık küme olması problem oluşturmaktadır.

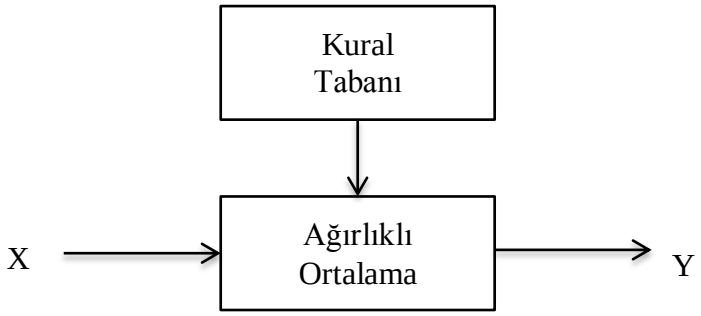


Şekil 5.2. Salt bulanık sistemlerin temel yapısı

Takagi ve Sugeno (1995) ve Sugeno ve Kant (1994) Takagi-Sugeno-Kant olarak isimlendirilen giriş ve çıkışlarının gerçek değerli değişkenler olan bir diğer bulanık sistemi tasarladılar. TSK bulanık sistemde Denklem 5.1'de verilen *EĞER-O HALDE* bulanık yapısı aşağıda belirtildiği şekilde değişmiştir.

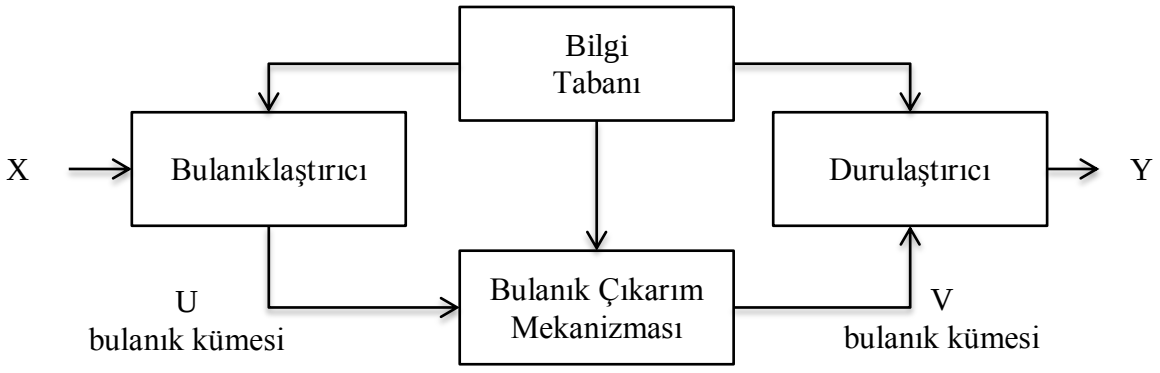
EĞER arabanın hızı yüksek O HALDE gaza uygulanacak güç $y=c.x$ (5.4)

Denklem 5.4 “yüksek” kelimesi Denklem 5.1 ile aynı şekilde kullanılmaktadır ve *c* ise sabittir. Denklem 5.1 ile Denklem 5.3 karşılaştırıldığında kuralın *O HALDE* kısmında doğal dil ile kullanılan açıklamanın matematiksel formüle dönüştüğü görülür. Bu değişiklik ile kuralları birleştirmek daha basitleşmektedir. Aslında, TSK bulanık sistemi Şekil 5.3’de görüldüğü gibi, *O HALDE* kısmında değerlerin ağırlıklı ortalamasıdır. TSK bulanık sistemin iki problemi vardır. Bunlardan birincisi *O HALDE* kısmının matematiksel ifade olmasından dolayı insan bilgisini sunmak için doğal bir çerçeve oluşturmaz, ikincisi ise bulanık mantığın farklı ilkeleri uygulanamamaktadır.



Şekil 5.3. TSK bulanık sistemin temel yapısı

TSK bulanık sistemin problemlerin çözümünde uygulanabilmesi için, bulanıklaştırıcı ve durulaştırıcı bileşenlerin sisteme eklenmesi gerekmektedir. Şekil 5.4’de görüldüğü gibi bulanıklaştırıcı kesin değerli giriş değerini giriş bulanık kümesine, durulaştırıcı ise bulanık küme değerini kesin değerli çıkışa dönüştürmektedir. Bu dönüşümler sayesinde kural tabanlı bulanık sistemler kontrol, sinyal işleme, modelleme, vb. birçok mühendislik uygulamalarında yaygın olarak kullanılabilmektedir.

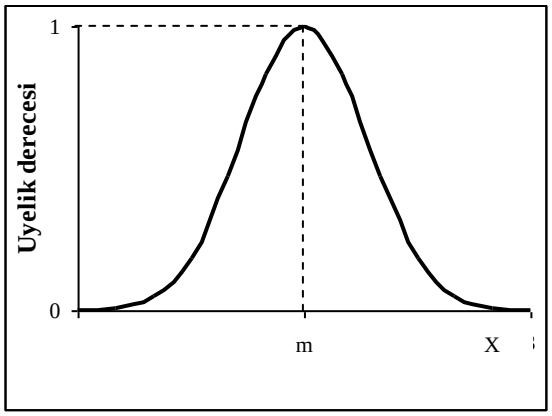


Şekil 5.4. Bulanıklaştırıcı ve durulaştırıcı içeren bulanık sistemin genel yapısı

Sonuç olarak belirtilen sistemlerin kombinasyonu, bulanık çıkarım sistemi, bulanık kural tabanlı sistem veya kontrolör olarak kullanıldığında bulanık kontrolör olarak isimlendirilmektedir. Temel olarak bir bulanık çıkarım sistemi bulanıklaştırıcı, bilgi tabanı, bulanık çıkarım mekanizması ve durulaştırıcı olmak üzere dört fonksiyonel bloktan oluşmaktadır.

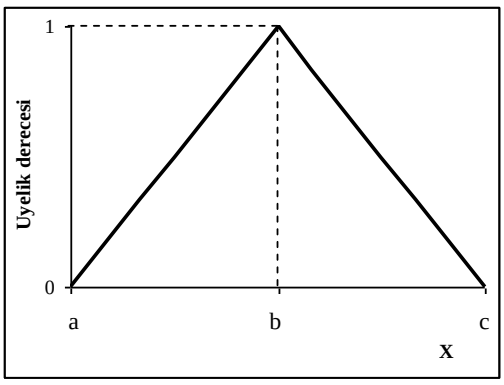
5.1.2.1. Bulanıklaştırıcı

Kesin değerlere sahip giriş ve çıkış parametreleri bulanık değerlere üyelik fonksiyonu olarak dönüştürülür. Bu üyelik fonksiyonları çözülecek probleme göre farklılık arz edebilir. Genel olarak çan eğrisi, üçgen ve yamuk (sırasıyla Şekil 5.5,Şekil 5.6 ve Şekil 5.7) üyelik fonksiyonları bulanıklaştırma metodu olarak kullanılmaktadır.



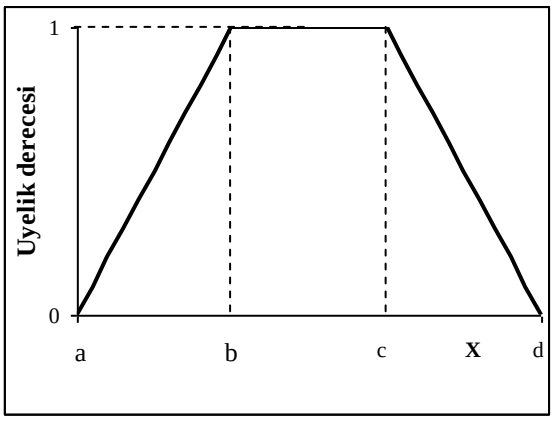
Şekil 5.5. Çan eğrisi üyelik fonksiyonu

$$\mu_A(X) = e^{-a(x-m)^2} \qquad a > 0,\ m \in R \tag{5.5}$$



Şekil 5.6. Üçgen üyelik fonksiyonu

$$\mu_A(x) = \mu_A(x,a,b,c) = \begin{cases} (x-a)/(x-b) & eğer \quad a \le x < b \\ (c-x)/(c-b) & eğer \quad b \le x \le c \\ 0 & eğer \quad x > c \ veya \ x < a \end{cases} \tag{5.6}$$



Şekil 5.7. Yamuk üyelik fonksiyonu

$$\mu_A(x) = \mu_A(x,a,b,c,d) = \begin{cases} (x-a)/(b-a) & eğer \quad a \le x < b \\ 1 & eğer \quad b \le x < c \\ (d-x)/(d-c) & eğer \quad c < x \le b \\ 0 & eğer \quad x > d \ veya \ x < a \end{cases} \tag{5.7}$$

5.1.2.2. Bulanık kurallar tabanı

Bir bulanık kural, bulanık bir "Eğer- O halde" kuralıdır.

Örneğin, "*Eğer oda sıcaklığı T düşük ise, O halde kazan kalorifer sıcaklığını artır*" kuralı bulanık bir kuraldır. Görüldüğü gibi düşük ve artır değerleri bulanıktır ve uygun üyelik dereceleri ile belirlenirler. Aynı süreci temsil eden bulanık kurallar topluluğu bulanık kurallar tabanını (*Fuzzy rules base*) oluşturur. Bir bulanık kurallar tabanında birbirine bağlı olan kurallar mevcuttur ve böyle bağlılık sistemin girişine verilen gerçeklerden yola çıkarak bir sonuca varmaya olanak vermektedir (Wang, 1997).

M tane bulanık kural olduğunu, $A_i^l \subset U_i$, $B^l \subset V$, $x = (x_1, x_2, ..., x_n)^T \in U$ ve $y \in V$ olduğunu varsayarak bir bulanık kuralın genel yapısını aşağıdaki gibi belirleyebiliriz:

Ku^l Eğer x_1 A_1^l *dir* *ve....ve* x_n A_n^l *dir* ise O halde y B^l'dir. (5.8)

Burada l=1,2,.....,M ifadesi birkaç değişik tip bulanık kuralı içerdiğinden dolayı yasal (*kanonik*) bulanık Eğer- O halde kuralı olarak adlanır.

5.1.2.3. Bulanık çıkarım mekanizması

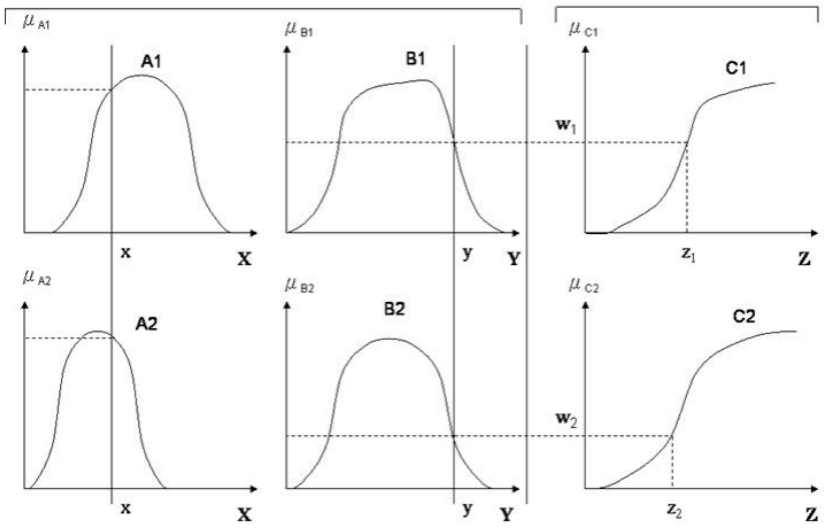
Çıkarım mekanizması yani kural yorumlayıcısı, işçi bellek adı verilen veri tabanındaki verileri ve bilgi tabanında bulunan kuralları kontrol ederek, işçi belleğe yeni verilerin eklenmesi görevini yerine getirip bunların bir dizi içerisinde sunulması ve bu kurallar içerisinden hangisinin olacağına karar verilmesini sağlayan sistemdir. Yani bir nevi yönetim görevini de üstlenmiştir.

Çıkarım mekanizması bulanık EĞER-O HALDE kuralları ve bilinen olgulardan bir dizi sonuçlar lede etme işlemidir. (Allahverdi, 2003; Saritas, 2003; Donescu ve ark., 1996). Bulanık çıkarım mekanizması tarafından gerçekleştirilen bulanık çıkarım dört adımdan oluşmaktadır.

1. Her bir dilsel ifadenin üyelik değerlerini elde etmek için öncül kısımda bulunan üyelik fonksiyonları ile giriş değişkenlerini karşılaştırır.
2. Belirlenen bir T-norm operatörü ile, genellikle minimum veya çarpım olmak üzere, herbir kuralın ateşleme seviyesini bulmak için öncül kısımda bulunan üyelik değerlerini birleştirir.
3. Ateşleme seviyesine bağlı olarak her bir kural için nitelikli sonuçlar oluşturur.
4. Bir çıkış üretmek için nitelikli alt bileşenleri toplar.

Bugüne kadar bulanık kural çıkarımının farklı tipleri ileri sürülmüştür ve bu bulanık çıkarım sistemleri bulanık çıkarım türü ve bulanık EĞER-O HALDE kuralların uygulanmasına göre üç tip olarak sınıflandırılabilir.

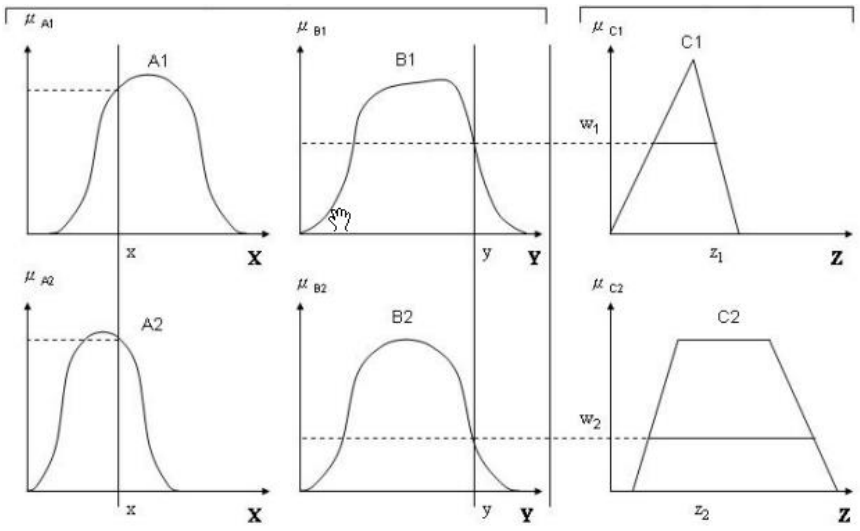
- Tip-I: Sonuç çıktısı her bir kuralın ateşleme gücü ve çıkış üyelik fonksiyonları tarafından oluşturulan kesin çıktı değerlerinin ağırlıklı ortalamasıdır. Ancak bu tipin kullanıldığı çıkış üyelik fonksiyonları monoton azalmayan olmalıdır (Şekil 5.8).



Şekil 5.8. Tip- I Bulanık çıkarım mekanizması

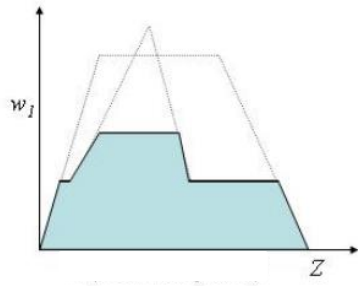
$$\because \mu_{A1}(x) > \mu_{B1}(y) \Rightarrow w_1 = \mu_{B1}(y) \\ \because \mu_{A2}(x) > \mu_{B2}(y) \Rightarrow w_2 = \mu_{B2}(y) \rightarrow z = \frac{w_1.z_1 + w_2.z_2}{w_1 + w_2} \qquad (5.9)$$

- Tip-II: Sonuç çıktısı her bir kuralın ateşleme gücü ve çıkış üyelik fonksiyonlarının minimumu ile elde edilen bulanık çıkışlara maksimum işleminin uygulanması ile elde edilir. Bulanık çıkış üzerinden sonuç çıktısını elde etmek için ağırlık merkezi, maksimumların ortalaması gibi çeşitli yöntemler önerilmiştir (Şekil 5.9, Şekil 5.10).



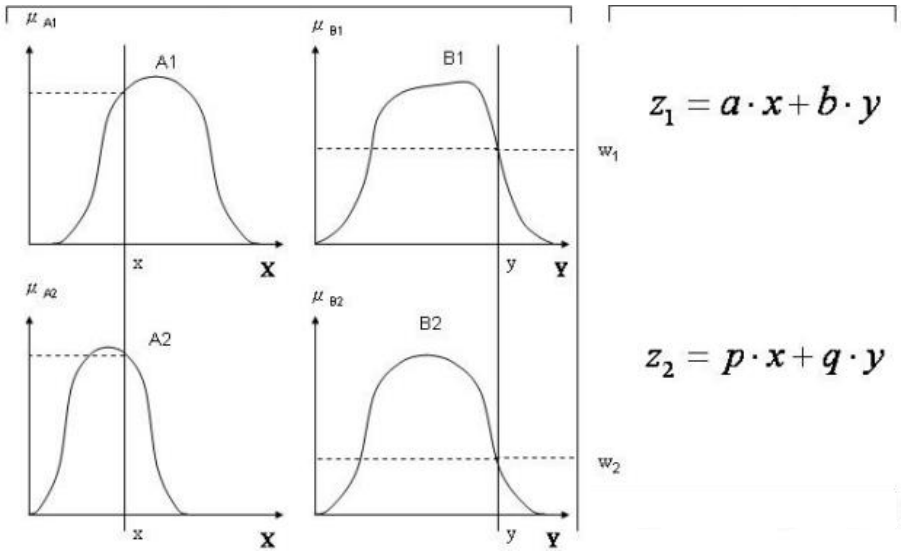
Şekil 5.9. Tip- II Bulanık çıkarım mekanizması (minimum)

$$\begin{matrix} \because \mu_{A1}(x) > \mu_{B1}(y) \Rightarrow w_1 = \mu_{B1}(y) \\ \because \mu_{A2}(x) > \mu_{B2}(y) \Rightarrow w_2 = \mu_{B2}(y) \end{matrix} \rightarrow z = Alan\ merkezi \quad (5.10)$$



Şekil 5.10. Tip- II Bulanık çıkarım mekanizması (maksimum)

- Tip-III: Her bir kuralın çıkışı girişin doğrusal kombinasyonuna sabit bir terim eklenmesiyle oluşur ve sonuç çıktısı her bir kuralın çıkışının ağırlıklı ortalamasıdır (Şekil 5.11).



Şekil 5.11. Tip-III Bulanık çıkarım mekanizması

$$\begin{matrix} \because \mu_{A1}(x) > \mu_{B1}(y) \Rightarrow w_1 = \mu_{B1}(y) \\ \because \mu_{A2}(x) > \mu_{B2}(y) \Rightarrow w_2 = \mu_{B2}(y) \end{matrix} \rightarrow z = \frac{w_1.z_1 + w_2.z_2}{w_1 + w_2} \quad (5.11)$$

5.1.2.4. Durulaştırıcı

Bulanık değerlerin kesin değerlere dönüştürülmesi yani bulanık $B^{'}CV$ kümesinden kesin bir $y^{\forall} \in V$ noktasına gömülme süreci "*durulaştırıcı*" olarak

adlandırılır. Burada $V \subset R$ bulanık çıkarım mekanizmasının çıkışıdır. B' özel yollarla elde edildiğinden dolayı ona en iyi denk gelen noktayı seçmek için birkaç yöntem vardır.

Durulaştırıcı için de aşağıdaki üç kriter istenmektedir.

Akla yakınlık (*plausibility*). y^* noktasının B'nü temsil etmesi sezgisel (*intuitive*) yolla belirlenir. Örneğin bu nokta yaklaşık olarak B' desteğinin ortasında yerleşebilir ve ya B' üyeliğinin en yüksek derecesini alabilir.

Hesaplama basitliği. Bu kriter gerçek zaman ölçeğinde çalışan kontrol sistemleri için çok önemlidir.

Devamlılık (*continuity*) B'deki küçük bir değişiklik y^* de büyük değişikliğe neden olmamalıdır.

5.1.3. Bulanık çıkarım sistemindeki sınırlılıklar

Elmas (2003) bulanık mantık yaklaşımının klasik mantığa göre sınırlılıklarını aşağıdaki gibi belirmiştir.

- Bulanık mantık uygulamalarında kuralların uzman deneyimlerine göre oluşturulması gereklidir. Üyelik işlevlerini ve bulanık kuralları tanımlamaz her problem için kolay olamayabilir.
- Üyelik işlev değişkenlerinin belirlenmesinde kesin olarak ortaya konmuş bir yöntem yoktur. En uygun yöntem deneme yanılma metodu olup buda uzun zaman alabilmektedir.
- Sistemlerin kararlılık, gözlemlenebilirlik ve denetlenebilirlik analizi yapılması için kesin bir yöntemin olmayışı bulanık mantığın problemlerinden biridir.
- Belirlenen üyelik işlevleri o sisteme özel olup, başka sistemlere uyarlanamaz. Ayrıca öğrenme yeteneğinin olmaması da bir dezavantajıdır.

5.2. Yapay Sinir Ağları

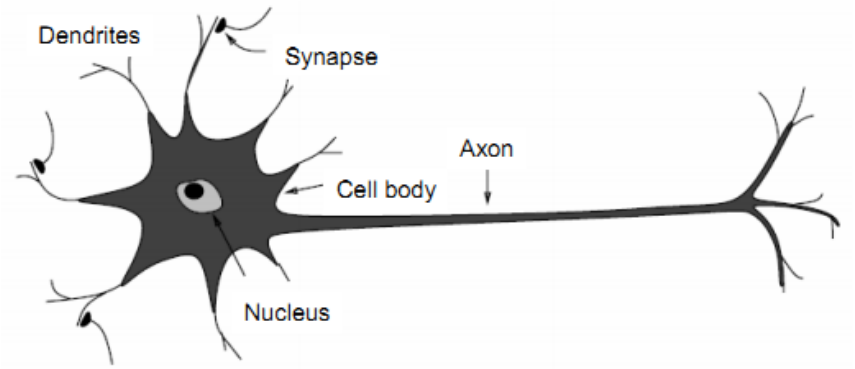
Yapay sinir ağları (YSA), insan beyninin özelliklerinden olan öğrenme yolu ile yeni bilgiler türetebilme, yeni bilgiler oluşturabilme ve keşfedebilme gibi yetenekleri, herhangi bir yardım almadan otomatik olarak gerçekleştirebilmek amacı ile geliştirilen bilgisayar sistemleridir (Öztemel, 2003).

YSA ile ilgili çalışmalar insan beynini modellemek ve nasıl çalıştığını anlamak amacıyla başlamıştır. İnsan beynini çalışma fonksiyonuna benzetme isteği YSA modellerine esin kaynağı olmuştur. Beyin, birbirine değişik etki seviyeleri ile bağlantılı binlerce nörondan oluşmaktadır. YSA modellerinin de temelini birbirine ağırlık bağlantıları ile bağlanan nöron modelleri oluşturmaktadır. YSA sonuçta bir bilgi işleme modeli olarak karşımıza çıkar (Haykin, 1998).

YSA sistemlerine ilginin bu denli yoğun olmasının ve mühendislik uygulamalarında geniş bir yer bulmasının sebebi, klasik yöntemlerle çözülemeyen veya çözümü çok karmaşık olan problemlere başarıyla uygulanmış olması ve klasik metotlara bir alternatif oluşturmasıdır (Ansett ve Kreider, 1993).

5.2.1. Biyolojik nöron yapısı

İnsan beyni, 10^{12} kadar nörondan oluşur ve her bir nöron yaklaşık 10000 farklı nöronla bağlantılıdır. Nöron, bir çekirdekten kendisine bağlı *dentrit* ve aksonlardan oluşur. Şekil 5.12'de biyolojik bir nöronun yapısı verilmiştir. Her nöron, *dentritleri* aracılığıyla diğer nöronlardan elektrokimyasal girişler alır. Bu girişlerin elektriksel toplamı belli bir eşik değerini aşarsa, nöron aktive olur ve aksonlarından çıkış sinyalleri üretir. Böylece aksonlara aktarılan elektrokimyasal sinyaller, bağlantı halinde olan diğer nöronların *dentritlerine* giriş olarak aktarılmış olur. Bütün beyin buna benzer birbiri ile bağlantılı nöronlardan oluşur ve bu nöronlar elektrokimyasal sinyaller yardımıyla iletişim halindedir. Biyolojik nöron basit bir işlem yapıyor gibi görünse de beyin, bu işlemler sayesinde oldukça karmaşık görünen problemleri kolaylıkla çözebilmektedir (Yalçıntaş ve Akkurt, 2005).



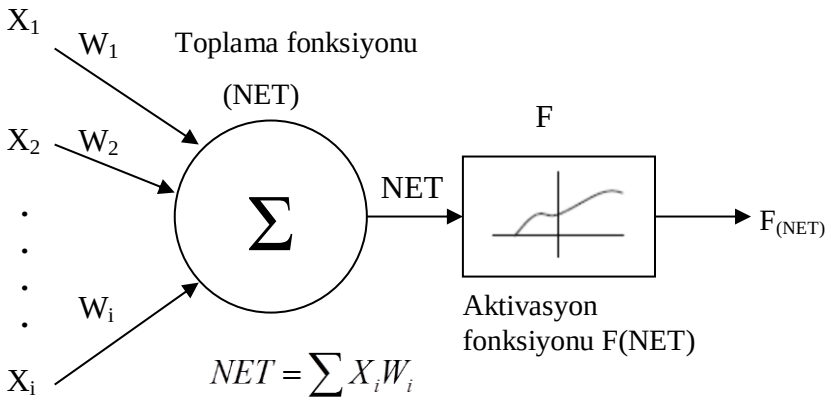
Şekil 5.12. Biyolojik nöronun yapısı

Biyolojik nöronun çalışma prensibi, yapay sinir ağları çalışma modeline temel oluşturmuştur. Fakat yapay sinir ağları modelinde yapılan sürekli araştırma ve geliştirmeler, beynin oluşturduğu karmaşık yapıyı modellemekten henüz çok uzaktır. Buna rağmen YSA'lardan, görüntü tanıma, geçmiş bilgilere dayanarak ileriye yönelik tahmin gibi problemlerde oldukça başarılı sonuçlar alınmıştır. Bu tür işlevler, insan beyninin günlük, sıradan ve sürekli çözdüğü fakat klasik bilgisayar mantığı çerçevesinde çözümü oldukça zor problemlerdir (Allahverdi, 2002; Breekweg ve ark., 2000).

YSA'lar, herhangi bir problemin girdi ve çıktılar arasındaki ilişkiler lineer olsun veya olmasın öğrenerek çözümler üretebilme özelliğine sahiptirler.

5.2.2. Yapay sinir hücresi

YSA'ları oluşturan en küçük bilgi işlem birimi, yapay sinir hücresidir. Şekil 3.10'da *i* adet girişi ve tek çıkışı olan bir yapay sinir hücresi verilmiştir. Şekil 5.13'deki yapay sinir hücresinde X_i sinir girişlerini, w_i girişler ile sinir hücresi arasındaki ağırlık katsayılarını ve *F* aktivasyon fonksiyonunu ifade etmektedir. Aktivasyon fonksiyonu, sinir hücresinin toplamdan sonraki çıkışını bir işlemden geçirdikten sonra hücre çıkışını *F(NET)* oluşturur. Genellikle aktivasyon fonksiyonu nonlineer bir fonksiyondan oluşur (Karna ve ark., 1989)



Şekil 5.13. Yapay sinir hücresinin yapısı

Sinir girişlerinin toplamını ifade eden NET fonksiyonu;

$$NET = \sum_{i=1}^{n} X_i w_i - \theta_i \tag{5.12}$$

şeklinde hesaplanır. Burada θ_i eşik değeridir. Sinir çıkışını oluşturan $F_{(NET)}$ fonksiyonu ise,

$$F_{(NET)} = F\left(\sum_{i=1}^{n} X_i w_i - \theta_i \right) \tag{5.13}$$

denklemi ile hesaplanabilir.

Biyolojik sinir sistemi ile yapay sinir sistemi arasında bazı benzerlikler vardır. Bu benzerlikler Çizelge 5.1'de verilmiştir.

Çizelge 5.1. Biyolojik sinir sistemi ile yapay sinir sisteminin benzerlikleri

Biyolojik sinir sistemi	**Yapay sinir sistemi**
Nöron	İşlem elemanı
Dendrit	Toplama fonksiyonu
Hücre gövdesi	Transfer fonksiyonu
Aksonlar	Eleman çıkışı
Sinapslar	Ağırlıklar

YSA'nın çizelgede bahsedilen özellikleri, karmaşık problemleri çözebilme yeteneğini göstermektedir.

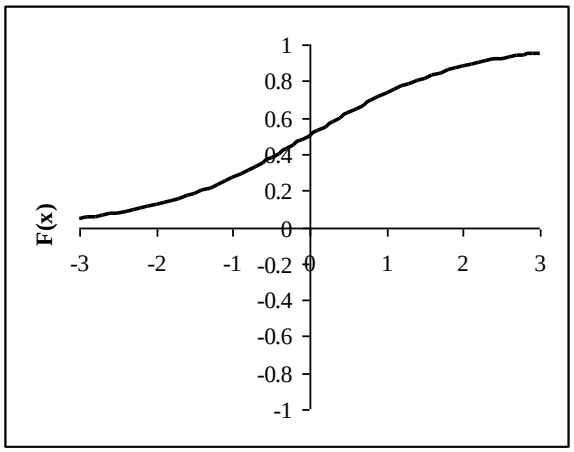
5.2.3. Aktivasyon fonksiyonları

YSA, *nonlineer* aktivasyon fonksiyonları kullanmadan kompleks ve *nonlineer* problemleri çözemez. Aktivasyon fonksiyonları, girişlerin veya problemde istenilen çıkışların özelliklerine göre seçilebilir. Hangi problemlerde hangi fonksiyonun kullanılacağı, YSA'nın çözmeye çalışacağı problemin şartlarına bağlıdır ve etkin bir şekilde deneme yanılma yoluyla tespit edilebilir (Hagan, 1997).

5.2.3.1. Sigmoid (*logsig(x)*) aktivasyon fonksiyonu

Sigmoid fonksiyonu, sadece pozitif değerler üretir. Matematiksel ifadesi Denklem 5.14'de, üyelik fonksiyonu Şekil 5.14'de verilmiştir.

$$F(x) = \frac{1}{1+e^{-x}} \tag{5.14}$$

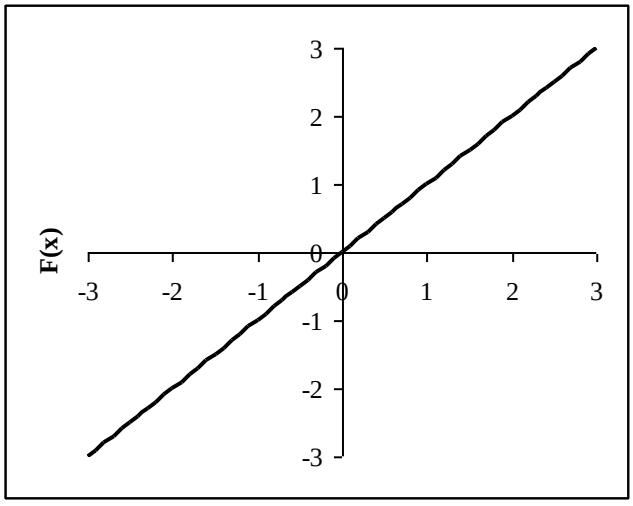


Şekil 5.14. Sigmoid aktivasyon fonksiyonu grafiksel yapısı

5.2.3.2. Lineer (*lin(x)*) aktivasyon fonksiyonu

Eğer YSA, lineer aktivasyon fonksiyonu kullanıyorsa ağ, lineer işlem özellikleri gösterir ve lineer problemleri çözebilir. Matematiksel ifadesi Denklem 5.15'te, üyelik fonksiyonu Şekil 5.15'te verilmiştir

$$F(x) = x \tag{5.15}$$

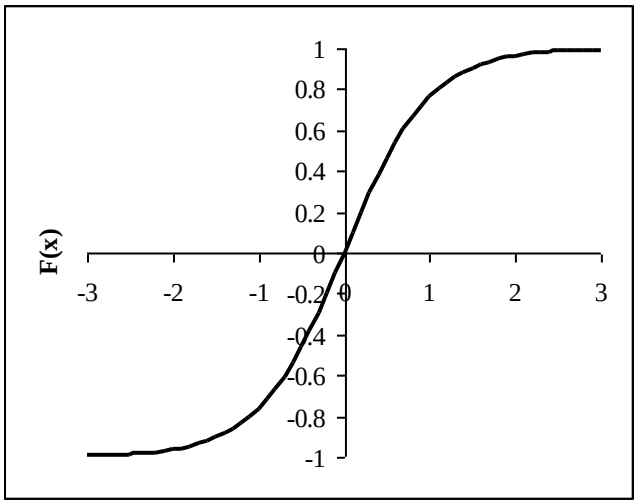


Şekil 5.15. Lineer aktivasyon fonksiyonu grafiksel yapısı

5.2.3.3. Tanjant hiperbolik (*tansig(x)*) aktivasyon fonksiyonu

Tanjant hiperbolik fonksiyon ise hem pozitif hem de negatif değerler üretir. Genellikle çok katmanlı YSA'larda logaritmik *sigmoid* veya tanjant hiperbolik fonksiyonları kullanılmaktadır. Matematiksel ifadesi Denklem 5.16'da, üyelik fonksiyonu Şekil 5.16'da verilmiştir

$$F(x) = \frac{2}{1+e^{-2x}} - 1 \tag{5.16}$$

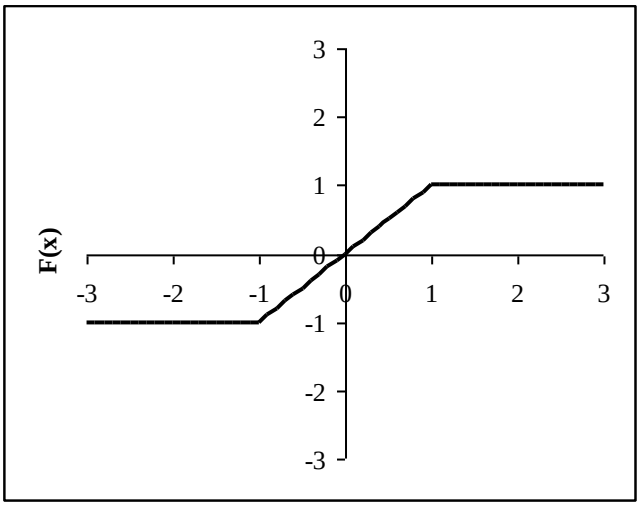


Şekil 5.16. Tanjant hiperbolik aktivasyon fonksiyonu grafiksel yapısı

5.2.3.4. Doyum fonksiyonu (*sat(x)*) aktivasyon fonksiyonu

Doyum fonksiyonu, mantıksal işlemler veya mantıksal sınıflandırma gerektiren problemlerde kullanılmaktadır. Matematiksel ifadesi Denklem 5.17'de, üyelik fonksiyonu Şekil 5.17'de verilmiştir.

$$F(x) = \begin{cases} +1 & x \geq 1 \\ x & -1 \leq x \leq +1 \\ -1 & x \leq -1 \end{cases} \tag{5.17}$$

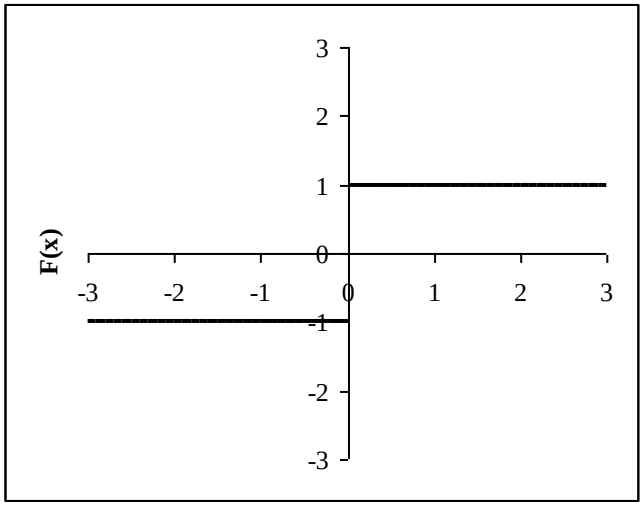


Şekil 5.17. Doyum aktivasyon fonksiyonu grafiksel yapısı

5.2.3.5. Keskin Sınırlayıcı (*signum(x)*) aktivasyon fonksiyonu

Keskin sınırlayıcı fonksiyonu da doyum fonksiyonu gibi mantıksal işlemler veya mantıksal sınıflandırma gerektiren problemlerde kullanılmaktadır. Matematiksel ifadesi Denklem 5.18'de, üyelik fonksiyonu Şekil 5.18'de verilmiştir.

$$F(x) = \begin{cases} +1 & x \geq 0 \\ -1 & x \leq 0 \end{cases} \tag{5.18}$$



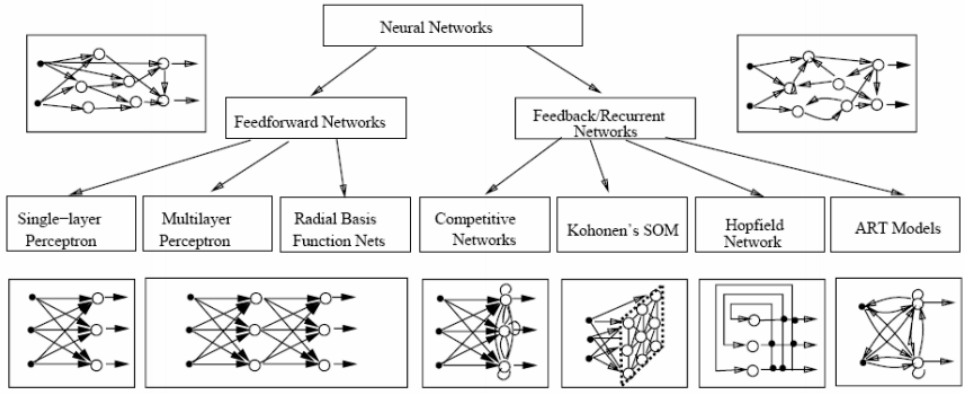
Şekil 5.18. Signum aktivasyon fonksiyonu grafiksel yapısı

5.2.4. YSA'nın yapısı ve işlem elemanları

Yapay sinir hücrelerinin bir çatı altında birleşiminden oluşan ve beynimizin sinir yapısının basit bir yapısı olarak düşünülen yapıya yapay sinir ağı adı verilmektedir. YSA'lar insan beyninde olduğu gibi bazı özelliklere sahiptirler:

- Yüksek hızlı paralel işlem yeteneğine sahip olan son derece birbirine bağlı düğümlerden oluşur.
- Kararlı ve hata toleranslıdır.
- Esnektir, öğrenme yoluyla yeni ortamlara adapte olabilir.
- Belirsiz, gürültülü ve ya tutarsız veriler ile başa çıkabilir.
- Bilgiyi bağlantıları aracılığı ile depolar

YSA'lar her ne kadar nörobilimden esinlenmiş olsa da, gerçekte nispeten basit matematiksel yapılara göre işlem yapan alternatif bir hesaplama paradigmasıdır. YSA düğüm noktalarını nöronlara ve bazı nöronların çıkışından bazı nöronların girişine ağırlıklandırılmış bağlantılara sahip bir çizge (graph) olarak görülebilir. Şekil 5.19'da görüldüğü gibi bağlantı yapısına göre YSA'lar iki ana kategoride toplanabilir. Bunlardan ilki ağ yapısında geri döngü bulunmayan ileri beslemeli ağlar, ikincisi ağ yapısında geribildirim bağlantısı sağlayan döngü barındıran geri dönüşümlü ağlardır.



Şekil 5.19. YSA yapılarının sınıflandırılması

YSA yapısında her bir düğüm, hücre denilen n. dereceden lineer olmayan bir devredir. Düğümler işlem elemanı olarak tanımlanır. Düğümler arasında bağlantılar vardır. Her bağlantı tek yönlü işaret iletim yolu olarak görev yapar. Her işlem elemanı istenildiği sayıda giriş bağlantısı ve tek bir çıkış bağlantısı alabilir. Fakat bu bağlantı kopya edilebilir. Yani bu tek çıkış, bir çok hücreyi besleyebilir. Ağdaki tek gecikme,

çıkışları ileten bağlantı yollarındaki iletim gecikmeleridir. İşlem elemanının çıkışı isteğe bağlı olarak sürekli/reel veya iki tabanlı bir değeri olan herhangi bir matematiksel tipte olabilir.

Bu tez çalışmasında çok katmanlı ileri beslemeli ağ yapısı kullanılmıştır. Bu ağ yapısı Bölüm 5.2.5'de özetlenmiştir.

5.2.5. Çok katmanlı ileri beslemeli ağ yapısı

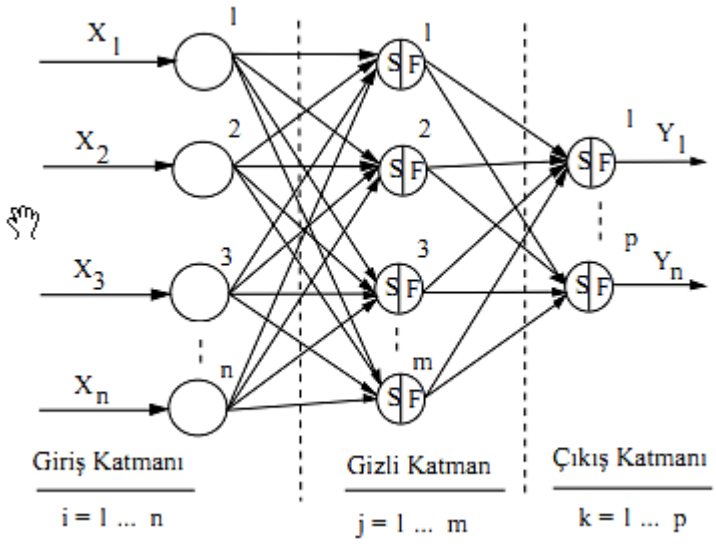
Rosebblat (1962)'ın ilk tek katmanlı algılayıcıyı tanıtmasından itibaren, birçok YSA modeli önerilmiş ve incelenmiştir. Bu modeller danışmalı ve danışmansız öğrenme algoritmaları, ikili veya sürekli düğüm değerleri, ileri beslemeli veya geri dönüşümlü, ayarlanabilir veya sabit parametreli gibi birçok kritere göre sınıflandırılabilmektedir.

Tek bir nöron (düğüm) veya tek katmanlı nöronlar sadece doğrusal çözümleri olan problemleri çözebilir. Ama, eğer problem sınırlı ve çok daha karmaşık ise doğrusal olmayan bir bölümleme gerektirmektedir. Bir YSA yapısı üç veya daha fazla katmandan oluşmaktadır. İlk katman veya giriş katmanı çevreden gelen bilgileri veya dış sinyalleri alır. Çıkış katmanı girişe karşılık oluşan ağın tepkisini çevreye iletir. Giriş ve çıkış katmanları arasında bir ya da daha fazla gizili katman bulunabilmektedir.

Bu tez çalışmasında, istenilen çıkış değerinin oluşturulmasına odaklanarak bir model oluşturulmaya çalışılmıştır, bu yüzden YSA'nın ayarlanabilir parametreleri danışmalı öğrenme kuralı tarafından güncellenmektedir.

Nöronlar arası bağlantılar ağırlıklar ile sağlanmaktadır. Bu sınıf ağ yapılarında düğümler katmanlar arasında bağlantıyı sağlarlar. Öğrenme ağın ilk rastgele oluşturulmuş durumundan dengeye doğru değişen ağırlıkların tekrarlı ayarlanması yoluyla gerçekleşir. Danışmalı öğrenme bilinen bir giriş-çıkış çift gerektirir. Eğitim boyunca, ağın çıkışı istenilen veya hedeflenen çıkış ile karşılaştırılır. Ağırlıklar ağın çıkışı ile istenilen çıkış arasındaki farkı yani hatayı azaltmak için ayarlanır. tüm öğrenme sürecinin amacı hatayı en aza indirmektir. Ağ bir kez eğitildikten sonra, girişe uygulanacak değerler ile ileri besleme adımında nöron ve ağırlıklar çıkış sonucunu vermektedir. Yapılan çalışmalar yeterli sayıda gizli nöron bulunduğunda, herhangidir sürekli fonksiyonun çok katmanlı ileri beslemeli ağ yapısı tarafından tahmin edebildiği görülmüştür (Cybenko, 1989;.Funahashi, 1989; Hornik ve ark., 1989)

Şekil 5.?'de aynı katmandaki nöronların ve katmanların arasında geri besleme bağlantısının bulunmadığı tipik bir ileri beslemeli ağ yapısı verilmiştir. Ağ yapıları her katmandaki nöron sayısına göre isimlendirilir, örneğin Şekil 5.20 için bu [n-m-p] olup, F aktivasyon fonksiyonunu, S nöron *giriş x ağırlıklardan* oluşan nöron NET değerini temsil etmektedir.



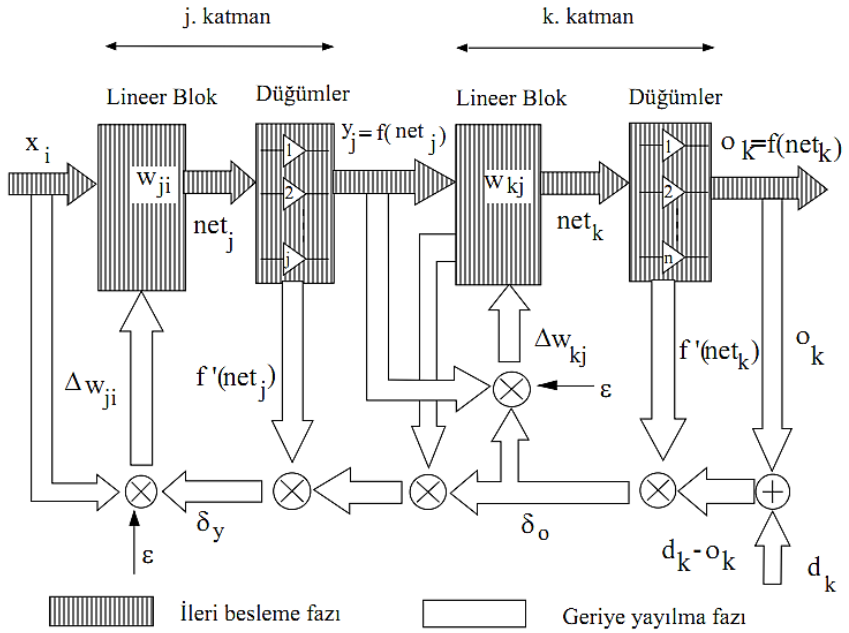
Şekil 5.20. [n-m-p] mimarisine sahip çok katmanlı YSA

Çok sayıda gizli katmana sahip sistemlerde, her sistemin hata işaretleri, bir önceki katmanın düzeltilmiş işlemlerinden çıkartılarak işlem tekrarlanır. Sonuç olarak, ağırlık düzeltme işlemi çıkış seviyesine bağlı ağırlıklardan başlar ve işlem ters yönde, giriş seviyesine varana kadar devam eder. Sonuçta sistem hatalar yapar, ama bu hatalardan bir şeyler öğrenip isteneni bulana kadar işleme devam eder. Bu yönteme "*hatanın geriye yayılması algoritması*" (back-propagation algorithm) denilir.

5.2.6. Hatanın geriye yayılması algoritması ve genelleştirilmiş delta kuralı

Hatanın geriye yayılması eğitme algoritması, çok katmanlı, ileri yayılımlı bir perceptrondan elde edilen çıkışlar ile eldeki hedef çıkışlar arasındaki hataların karesinin ortalamasını minimum yapmak için geliştirilmiş yinemeli bir gradyan algoritmadır ve eğitme işlemi için genelleştirilmiş delta kuralını (Generalized Delta Rule) kullanır. Bu öğrenme algoritması ilk olarak Werbos (1974) tarafından istatistik alanındaki tezinde

"dinamik geribesleme" ismi ile sunuldu, Rumelhard ve ark. (1986) tarafından yayınlandıktan sonra popüler hale gelmiştir.



Şekil 5.21. Hatanın geriye yayılması algoritmasının blok diyagramı

Şekil 5.21'de görüldüğü gibi, bu algoritma ileri besleme fazı ve geri yayılma fazı olmak üzere iki safhadan oluşmaktadır. İleri besleme fazında, giriş sinyalleri giriş katmanından çıkış katmanına doğru yayılmaktadır. Geri yayılma fazında ise ağırlıkları ayarlamak amacıyla hata sinyali ağda geriye doğru yayılmaktadır. Sonuç çıktısı ise ileri yönde katman katman olarak hesaplanmaktır. Bir katmanın çıkışı bir sonraki katmanın girişi olmaktadır. Geri yayılma fazında, çıkış katmanının ağırlıkları ilk önce ayarlanmaktadır. Sonra gizli katmanda bulunan sırasıyla ayarlanmaktadır. Geri yayılım algoritması gradient descent metodu kullanarak hataların karesi fonksiyonunu minimum etmektedir. Geri yayılım algoritmasının adımları aşağıdaki gibi özetlenebilir.

1. Rasgele ağırlık değerleri verilir.
2. Eğitim setinden bir giriş ağa uygulanır.
3. Ağ çıkışı hesaplanır.
4. Ağ çıkışı ile istenen çıkış arasındaki hata hesaplanır.
5. Her bir bias ve ağırlığın değişimi hesaplanır.
6. Adım 5'de verilen değişim ile hatayı en aza indirmek için ağırlıklar ayarlanır.
7. Eğitim setinden yeni bir giriş ağa uygulanır.

8. Tüm sistem için hata kabul edilebilir bir duruma gelinceye kadar 2-7 arasındaki adımlar tekrarlanır.

Ağa giriş sunulur iken, m-boyutlu giriş örüntüleri girildiğinde; $x_i = [x_1, x_2, ... , x_m]^T$ dir. Benzer şekilde istenilen n-boyutlu çıkış örüntülerini ise; $d_k = [d_1, d_2, ... , d_n]^T$ belirtir. x_i değerleri *i* katmanındaki nöronların çıkış değerleri ise, *j* katmanındaki bir nörona gelecek olan toplam giriş;

$$net_j = \sum_{i=1}^{m} w_{ji}.x_i \tag{5.19}$$

olur. Gizli katmandaki j nöronunun çıkışı (transfer fonksiyonu çıkışı);

$$y_j = f_j(net_j), j= 1,2,....,J \tag{5.20}$$

dir ve burada f_j transfer fonksiyonudur. Çıkış katmanındaki *k* nöronuna gelecek olan toplam giriş ise,

$$net_k = \sum_{j=1}^{J} w_{kj}.x_j \tag{5.21}$$

dir. Yine çıkış katmanındaki bir k nöronun lineer olmayan çıkışı;

$$o_k = f_k(net_k), \text{ k= 1,2,….,n} \tag{5.22}$$

Ağdan elde edilen çıkış, asıl çıkış ile karşılaştırılır ve e_k hatası aşağıdaki gibi hesaplanır;

$$e_k = (d_k - o_k) \tag{5.23}$$

Burada d_k ve o_k sırasıyla çıkış katmanındaki herhangi bir *k* nöronunun hedef ve ağdan elde edilen çıkışlardır. Dikkat edilmesi gereken husus, böyle bir karşılaştırmanın sadece ağın çıkış katmanı için mümkün olmasıdır. Böylece çıkış katmanıyla olan bağlar için ağırlık ayarlaması öncelikle göz önüne alınır. Her bir örüntü için toplam karesel hata;

$$E = \frac{1}{2}\sum_k (d_k - o_k)^2 \tag{5.24}$$

olur. Gizli katman olduğu zaman; hata düzeyi sadece bir minimumdan oluşmaz, çeşitli minimumlar oluşur. Bu minimumlar lokal ve global olarak sınıflandırılırlar. Lokal minimumlar birden fazla olabilir. Global minimum, hatanın en küçük olduğu minimumdur. Öğrenme sırasında global minimuma ulaşmak amaçlanır. Ağırlıkların değişimi;

$$\Delta w_{kj} = -\varepsilon \frac{\partial E}{\partial w_{kj}} \tag{5.25}$$

Burada ε, öğrenme oranı katsayısı adı verilen küçük değerde pozitif bir sayıdır (çoğunlukla *0.01* ile *0.9* arasında seçilir). Bu öğrenme oranı, bir sayısal optimizasyon algoritmasındaki adım boyutu parametresine benzemektedir. Denklem 5.25'de eşitliğin sağ tarafı açılırsa (Rumelhart ve ark.1986), şu sonuç elde edilir:

$$\frac{\partial E}{\partial w_{kj}} = \frac{\partial E}{\partial net_k}\frac{\partial net_k}{\partial w_{kj}} \tag{5.26}$$

$$\frac{\partial net_k}{\partial w_{kj}} = \frac{\partial}{\partial w_{kj}}\sum_j w_{kj}.y_j = \sum_j \frac{\partial w_{kj}.y_j}{\partial w_{kj}} = y_j \tag{5.27}$$

Amaç, uygun w seçimiyle, $E = \sum_p E_p$ toplam hatayı yeterince küçük yapmaktır. Bu amacı gerçekleştirmek için, bir $p \in P$ örüntüsü ard arda ve rasgele biçimde seçilir. *k.* nöronda oluşan ve "*delta*" adı verilen hata işareti,

$$\delta_o = -\frac{\partial E}{\partial net_k} \tag{5.28}$$

Denklem 5.27 ve 5.28'yi Denklem 5.26'de yerine konulursa,

$$\frac{\partial E}{\partial w_{kj}} = \delta_o y_j \tag{5.29}$$

ve, Denklem 5.26 ifadesi Denklem 5.25'de yerine konulursa,

$$\Delta w_{kj} = \varepsilon \delta_o y_j \tag{5.30}$$

elde edilir. E_p değerini düşürmek demek, ağırlığı $\delta_o y_j$'ye bağlı olarak değiştirmek demektir. Buna "*Delta kuralı*" denir. Önceki gizli katmandaki tüm nöronlar ile çıkış katmanındaki k. nöron arasındaki bağ mukavemetleri, Denklem 5.30'da ki Δw_{kj} miktarı ile ayarlanır. Denklem 5.28 kısmi türevlerine ayrılırsa,

$$\delta_o = -\frac{\partial E}{\partial net_k} = -\frac{\partial E}{\partial o_k}\frac{\partial o_k}{\partial net_k} \tag{5.31}$$

$$\frac{\partial E}{\partial o_k} = -(d_k - o_k) \tag{5.32}$$

olup; *k.* nöron çıkışının lokal hatasını verir.

$$\frac{\partial o_k}{\partial net_k} = f_k'(net_k) \tag{5.33}$$

Son iki formül Denklem 5.31'de yerine konulursa,

$$\delta_o = (d_k - o_k) f_k'(net_k) \tag{5.34}$$

bulunur. Bu son terim Denklem 5.30'da yerine konulduğunda *k* nöronu için;

$$\Delta w_{kj} = \varepsilon (d_k - o_k) f_k'(net_k) y_j \tag{5.35}$$

olur. Eğer ağırlıklar çıkış nöronlarını doğrudan etkilemiyorsa (arada gizli katman varsa), Denklem 5.30'a benzer biçimde delta kuralı bu gizli katman için de şu şekilde uygulanır.

$$\Delta w_{ji} = -\varepsilon \frac{\partial E}{\partial w_{ji}}$$

$$= -\varepsilon \frac{\partial E}{\partial net_j}\frac{\partial net_j}{\partial w_{ji}} = -\varepsilon \frac{\partial E}{\partial net_j} x_i \tag{5.36}$$

$$= -\varepsilon \left(-\frac{\partial E}{\partial y_j}\frac{\partial y_j}{\partial net_j}\right) x_i = \varepsilon \left(\frac{\partial E}{\partial y_j}\right) f_j'(net_j) x_i$$

$$\Delta w_{kj} = \varepsilon \delta_y x_i \tag{5.37}$$

Bununla birlikte, $\partial E/\partial y_j$ faktörü doğrudan geliştirilemez. Özel olarak çıkış katmanına etkisi,

$$-\frac{\partial E}{\partial y_j} = -\sum_j \frac{\partial E}{\partial net_k}\frac{\partial net_k}{\partial y_j} = \sum_j \left(-\frac{\partial E}{\partial net_k}\right)\frac{\partial}{\partial y_j}\sum_m w_{km}\, y_j = \sum_k \delta_o\, w_{kj} \qquad (5.38)$$

şeklinde bulunur. Bu durumda;

$$\delta_y = f_j'\left(net_j\right)\sum_k \delta_o\, w_{kj} \qquad (5.39)$$

olur. Kısaca özetlenecek olursa; Denklem 5.39 formülü şayet *j*. nöron, çıkış katmanı nöronuysa Denklem 5.34'e benzer biçimde,

$$\delta_y = (d_y - o_y) f_j'(net_j) \qquad (5.40)$$

olur. Eğer *j* nöronları gizli katmana ait nöronlar ise, o zaman Denklem 5.29 kullanılır. Aktivasyon fonksiyonu olarak sigmoid fonksiyonu kullanıldığında,

$$f\left(net_j\right) = y_j = \frac{1}{1+e^{-net_j}} \qquad (5.41)$$

olur. Bu ifadenin türevi alınıp, gerekli sadeleştirme yapıldığında,

$$f'\left(net_j\right) = \frac{1}{1+e^{-net_j}}\frac{1+e^{-net_j}-1}{1+e^{-net_j}} \qquad (5.42)$$

$$\frac{\partial y_j}{\partial net_j} = y_j(1-y_j) \qquad (5.43)$$

bulunur. Benzer işlem *k* katmanı için de yapılırsa;

$$\frac{\partial y_k}{\partial net_k} = f_k'(net_k) = o_k(1-o_k) \qquad (5.44)$$

elde edilir. Denklem 5.42'yi Denklem 5.40'da, Denklem 5.44'de Denklem 5.34'de yerine konursa, Denklem 5.34 ve Denklem 5.44'deki delta ifadeleri aşağıdaki gibi olur. (δ_o çıkış katmanı, δ_y: gizli katman elemanları içindir);

$$\delta_o = (d_k - o_k)o_k(1 - o_k) \tag{5.45}$$

$$\delta_y = y_j(1 - y_j)\sum_k \delta_o w_{kj} \tag{5.46}$$

w_{ji}'yi öğrenme durumunda her eğitme örüntüsünün seti için Δw_{ji}'yi hesaplamak gerekir. Öğrenme oranı ε, hızlı öğrenmeyi sağlar; fakat dalgalanmalara sebep olabilir. Rumelhart ve McClelland (1986), Denklem 5.25 ve Denklem 5.37 ifadelerine bir tür momentum terimi dahil ederek bir değişim öngörmüşlerdir. Bu durumda;

$$\Delta w_{ji}(n+1) = \varepsilon\delta_y x_i + \propto \Delta w_{ji}(n) \tag{5.47}$$

$$\Delta w_{kj}(n+1) = \varepsilon\delta_o y_j + \propto \Delta w_{kj}(n) \tag{5.48}$$

olarak yazılabilir. Burada n, öğrenme döngülerinin (iterasyon) sayısını gösterir. Momentum terimi olan α, küçük değerde pozitif bir sayıdır. YSA ile ilgili bir başka sorun da, düzgün bir öğrenme katsayısının *(ε)* ayarlanmasıdır. Ağırlıkları çok yüksek tutmak davranışın bozulmasına sebep olabilir.

Öğrenme katsayısını böyle bir davranışı önlemek için küçük tutmak gereklidir. Öğrenme katsayısı, $0.01 < \varepsilon < 10$ aralığında seçilen sabit bir sayıdır. Öte yandan, çok küçük bir öğrenme oranında, öğrenme işleminin yavaşlamasına yol açar. Momentum (α) fikri bu noktadan hareketle ortaya atılmıştır. Momentum mevcut delta ağırlığı üzerinden önceki delta ağırlığının belirli bir kısmını besler. Böylece daha düşük öğrenme katsayısı ile daha hızlı öğrenme elde edilir. Momentum katsayısı genellikle $0<\alpha<1$ aralığında değişen bir sayıdır. Hatanın geriye yayılması algoritması genellikle iyi bir performans göstermesine rağmen; bu algoritma, bir gradyent arama tekniği olduğu için, global minimum yerine, en küçük kareler (Least Mean Square) fonksiyonunun bir lokal minimumunu bulabilmektedir. Performansı artırmak ve lokal minimum problemini azaltmak için; gizli düğümler ilave edilmesi, ağırlıkların ayarlanması için kullanılan öğrenme katsayısının azaltılması ve farklı rastgele ağırlık setlerinden başlayarak birçok defa eğitme işleminin tekrarlanması öneriler arasındadır (Lippmann 1987).

5.3. ANFIS

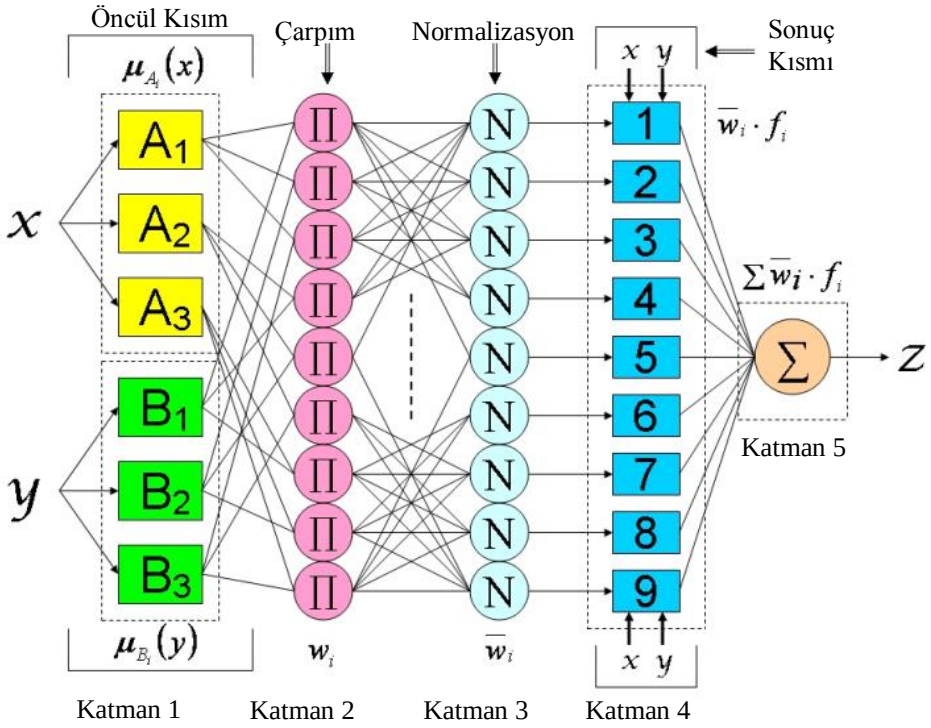
ANFIS ismi Uyarlamalı Bulanık Sinirsel Çıkarım Sistemi anlamına gelen "*Adaptive Network Fuzzy Inference Systems*" adının baş harflerinden oluşmaktadır. ANFIS YSA'nın öğrenme kapasitesine odaklanan ve bir bulanık çıkarım sistemine fonksiyonel olarak eşit olan ileri beslemeli bir ağ sınıfıdır. Yani, ANFIS bulanık mantık sistemini kullanırken adaptif durumda ayarlamak için YSA kullanılmaktadır. Bu sayede YSA için uygulanan tüm tasarım yöntemleri bulanık çıkarım sistemlerine uygulanabilmektedir. Öğrenme kuralı geri yayılım algoritması ve öğrenme sürecini hızlandıran Kalman filtresi birlikte kullanılması ile oluşturulmaktadır. YSA ve Bulanık Mantığın avantajlarını birleştiren ANFIS ile YSA eğitiminden giriş-çıkış verileri ile birlikte insan bilgisine dayalı doğrusal olmayan süreç modelleri oluşturulabilmektedir.

ANFIS giriş uzayını bölümlenmesine göre ızgara ve dağılım olmak üzere iki tipte sınıflandırılabilir. Izgara tipi bölümlendirme tüm giriş parametrelerinin üyelik fonksiyonlarının olası kombinasyonlarının hepsi için kurallar oluşturulmasına dayanır. Bu tip bölümlendirmede giriş sayısındaki artış üstel bir artışa yol açar. Her biri 3 üyelik fonksiyonuna sahip 7 girişli bir ANFIS yapısında ızgara bölümlendirme 3^7(=2187) kural oluşturur. Bu yapının tersine, dağılım bölümlendirmede her bir giriş için üyelik fonksiyonları ile aynı sayıda kurallar üretir.

ANFIS'in adaptif ileri beslemeli bir ağ olarak birbirine bağlı olan düğümlerin bir araya gelmesiyle oluşmaktadır. Her bir düğüm bir işlem birimini temsil etmektedir ve bu düğümler arasındaki bağlantı nedensel bir ilişkiyi belirtmektedir. Düğümlerin tamamı veya bir kısmı adaptif olabilmektedir. YSA'dan farklı olarak ayarlanabilir parametreler düğümlerde bulunmaktadırlar. Genellikle bir düğüm fonksiyonu değişebilen parametreleri bulunan bir fonksiyon şeklindedir. Bu parametreleri değiştirerek düğüm fonksiyonunu ve bunu yanında ağın davranışı değiştirebiliriz.

5.3.1. ANFIS mimarisi

Şekil 5.22'de iki girişli ve üç kurallı Tip-III bulanık çıkarım sistemine eşdeğer olan ANFIS mimarisi görülmektedir. Bu tip çıkarım sisteminde kurallar Takagi-Sugeno tip bulanık çıkarım sistemine göre oluşturulmaktadır. Bulanık kümeler sadece öncül kısımda mevcut olup, sonuç kısmı girişlerin bir fonksiyonu şeklindedir.



Şekil 5.22. 9 kurala sahip ANFIS mimarisi

Bu tip çıkarım sisteminde kurallar Takagi-Sugeno tip bulanık çıkarım sistemine göre oluşturulmaktadır. Bulanık kümeler sadece öncül kısımda mevcut olup, sonuç kısmı girişlerin bir fonksiyonu şeklindedir. Bulanık kural yapısı Denklem 5.49' da ifade edildiği şekildedir.

$$Eğer\ x\ A_i\ ve\ y\ B_i\ \ O\ HALDE\ f_i = p_i x + q_i y + r_i \tag{5.49}$$

ANFIS mimarisi içerisindeki her katma ait düğüm işlevleri ve dolayısıyla katmanların işlevleri aşağıda sırasıyla verilmiştir.

Katman 1 : Bu katmandaki her bir düğüm A_i ve B_i gibi bir bulanık kümeyi ifade etmektedir Bu katmandaki düğümlerin çıkışı, giriş örneklerine ve üyelik fonksiyonuna bağlı olan üyelik dereceleridir. Bu katmanlardan elde edilen üyelik dereceleri veya düğüm çıkışları Denklem 5.50'de verilmiştir.

$$o_i^1 = \mu_{A_i}(x)\ ,\ o_{i+3}^1 = \mu_{B_i}(x), i = 1,2,3 \tag{5.50}$$

x ve y gibi iki farklı girişe sahip olduğundan iki farklı düğüm çıkışı yazılmaktadır. Bu katmanda her bir düğüm için üyelik fonksiyonu olarak çan eğrisi kullanılırsa, sonuç işlevi Denklem 5.51'deki gibi olur.

$$\mu_{A_i}(x) = \frac{1}{1+\left[\left(\frac{x-c_i}{a_i}\right)^2\right]^{b_i}} \tag{5.51}$$

Burada $[a_i, b_i, c_i]$ öncül parametre kümesidir. Bu parametrelerin değerleri değiştirildiğinde, çan eğrisi fonksiyonu değişecektir.

Katman 2 : Bu katmadaki her düğüm Π ile etiketlenmiştir ve giren tüm işaretlerin çarpımını göstermektedir. Düğüm çıkışı Denklem 5.52'deki gibi ifade edilir.

$$o_i^2 = \mu_i = \mu_{A_i}(x)\mu_{B_i}(y) \tag{5.52}$$

Bu katmandaki her düğümün çıkışı bir kuralın ateşleme seviyesini temsil eder.

Katman 3 : Bu katmandaki her bir düğüm *N* ile etiketlenmiştir ve bir kuralın normalleştirilmiş ateşleme seviyesini ifade eder. Düğüm çıkışı Denklem 5.53'deki gibi ifade edilir.

$$o_i^3 = \overline{\mu_i} = \frac{\mu_i}{\sum_i \mu_i} \tag{5.53}$$

Katman 4 : Bu katmandaki her i düğümü adaptif bir düğümdür. Her düğüm sonuç ağırlık değerlerini hesaplamaktadır. Düğüm çıkışı Denklem 5.54'deki gibi ifade edilir.

$$o_i^4 = \overline{\mu_i} f_i = \overline{\mu_i}(p_i x + q_i y + r_i) \tag{5.54}$$

Burada $[p_i, q_i, r_i]$ sonuç parametre kümesidir ve eğitim boyunca belirlenir. Eğer p_i, q_i parametreleri sıfıra eşit ise sıfırıncı dereceden sugeno model olarak ifade edilir. Bu tez çalışmasında sıfırıncı dereceden sugeno model kullanılmıştır.

Katman 5 : Bu katmanda sadece bir düğüm vardır ve Σ ile etiketlenmiştir. Elde edilen toplam sistemin çıkışını verir.

$$o^5 = \sum_i \overline{\mu_i} f_i = \frac{\sum_i \mu_i f_i}{\sum_i \mu_i} \quad (5.55)$$

5.3.2. ANFIS için geri yayılım algoritması

Geri yayılım algoritmasının ANFIS mimarisine uygulanması ile ağın çıkışından elde edilen hata geriye doğru yayılarak öncül ve sonuç parametreleri ayarlanmaktadır. Amaç bütün çıkışlar için hata değerini sıfıra indirmektir. Ağın çıkışındaki hata istenen değer *d*, ile gerçek çıkış *f* arasındaki fark olarak ifade edilir (Denklem 5.56).

$$e = d - f \quad (5.56)$$

Bu hata değerine en küçük kareler yöntemi uygulanarak hata ölçütü E hesaplanmaktadır.

$$E = \frac{1}{2} e^2 \quad (5.57)$$

Hata ölçütü ile hata geriye doğru katman katman aşağıdaki gibi yansıtılır,

Katman 5 : Bu katmanda çıkıştaki hatanın katman çıkışına yansıması hesaplanır.

$$\delta^5 = -\frac{\partial \mathrm{E}}{\partial \mathrm{f}} = -\left[\frac{\partial \mathrm{E}}{\partial \mathrm{e}} \frac{\partial \mathrm{e}}{\partial \mathrm{f}}\right] = e \quad (5.58)$$

Katman 4 : Bu katmanda $[p_i, q_i, r_i]$ sonuç parametre kümesinin ayarlanması gerçekleştirilir.

$$\Delta p_i = -\frac{\partial \mathrm{E}}{\partial p_i} = \left[-\frac{\partial \mathrm{E}}{\partial \mathrm{f}}\right]\left[\frac{\partial \mathrm{f}}{\partial p_i}\right] \quad (5.59)$$

$$f = \sum_i \overline{\mu_i}(p_i x + q_i y + r_i) \quad (5.60)$$

$$\frac{\partial \mathrm{f}}{\partial p_i} = \overline{\mu_i} x \quad (5.61)$$

buradan

$$\Delta p_i = \delta^5 \overline{\mu_i} x \quad (5.62)$$

Aynı işlem diğer sonuç değişkenlerinde uygulanır. Elde edilen bu ifadelere örenme oranı eklenerek delta kuralı uygulanır. n={p,q,r} olarak alınırsa bu ifadeler aşağıdaki gibi belirtilir.

$$n_i(k+1) = n_i(k) + \eta_n \Delta n_i(k+1) \tag{5.63}$$

Katman 3 : Bu katmanda çıkış katmanından yansıyan hata değeri hesaplanır.

$$\delta^3 = -\frac{\partial \mathrm{E}}{\partial \overline{\mu_i}} = \left[-\frac{\partial \mathrm{E}}{\partial \mathrm{f}}\right]\left[\frac{\partial \mathrm{f}}{\partial \overline{\mu_i}}\right] \tag{5.64}$$

$$\frac{\partial \mathrm{f}}{\partial \overline{\mu_i}} = f_i \tag{5.65}$$

$$\delta^3 = \delta^5 f_i \tag{5.66}$$

Katman 2 : Bu katmanda çıkış katmanından yansıyan hata değeri hesaplanır, ağırlık hesaplaması yoktur.

$$\delta^2 = -\frac{\partial \mathrm{E}}{\partial \mu_i} = \left[-\frac{\partial \mathrm{E}}{\partial \mathrm{f}}\frac{\partial \mathrm{f}}{\partial \overline{\mu_i}}\right]\left[\frac{\partial \overline{\mu_i}}{\partial \mu_i}\right] \tag{5.67}$$

$$\frac{\partial \overline{\mu_i}}{\partial \mu_i} = \left[\frac{\sum_i \mu_i - \mu_i}{\sum_i {\mu_i}^2}\right] \tag{5.68}$$

$$\delta^2 = \delta^3 \left[\frac{\sum_i \mu_i - \mu_i}{\sum_i {\mu_i}^2}\right] \tag{5.69}$$

Katman 1 : Bu katmanda hem hata değeri hemde giriş üyelik işlevlerine ait değişkenlerin ayarlanması gerçekleştirilir.

$$\delta^1 = -\frac{\partial \mathrm{E}}{\partial \mu_{Ai}} = \left[-\frac{\partial \mathrm{E}}{\partial \mathrm{f}}\frac{\partial \mathrm{f}}{\partial \overline{\mu_i}}\frac{\partial \overline{\mu_i}}{\partial \mu_i}\right]\left[\frac{\partial \mu_i}{\partial \mu_{Ai}}\right] \tag{5.70}$$

$$\frac{\partial \mu_i}{\partial \mu_{Ai}} = \mu_{Bi} \tag{5.71}$$

$$\delta^1 = \delta^2 \mu_{Bi} \tag{5.72}$$

Bu katmandaki üyelik değerlerini değişimi hatanın üyelik parametrelerine yayılması ile elde edilmektedir. Elde edilen değişim üyelik işlevine ait parametrelerin ayarlanması Denklem 5.63'deki gibi gerçekleştirilir.

6. KULLANILAN DENETLEYİCİ YAPILARI

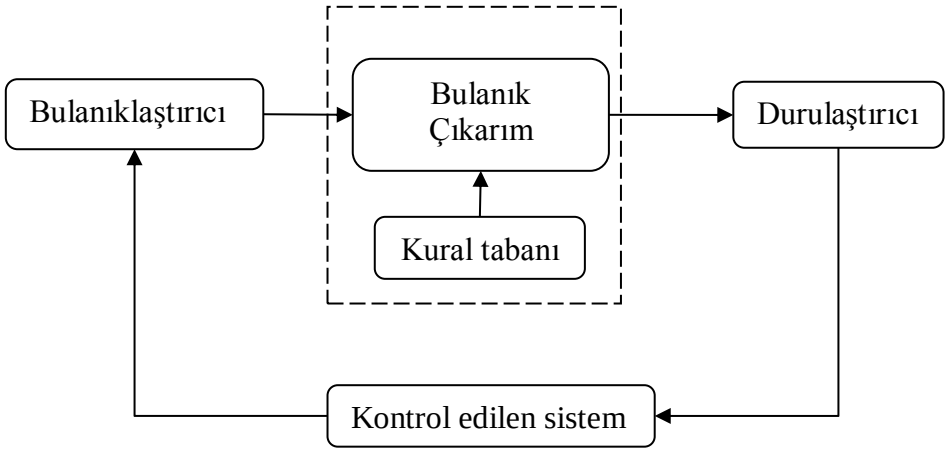
EMF sistemin akışkanlar dinamiği, manyetik alan, güç elektroniği gibi bir çok alanı kapsaması, sistem davranışının doğrusal olmaması, sistem üzerinden deneyler ile elde edilen verilerin gürültü gibi sebeplerden dolayı mükemmel olmaması, sistemin zamanla değişiminden dolayı zamandan bağımsız bir modelinin olmaması gibi nedenler ile sistemin matematiksel modelinin elde edilmesi oldukça zordur. Sistemin matematiksel modelinin çıkarmanın güç olduğu bu durumda en önemli bilgi kaynağı sensörlerden gelen veriler ve uzman kişilerdir. Bu sebepten dolayı yapılan tez çalışmasında EMF'nin kontrolünde farklı yapay zeka tabanlı denetleyiciler ve bunların hibrit yapıları kullanılarak sonuçları karşılaştırılmıştır. Bu bölümde kullanılan denetleyici yapıları hakkında bilgi verilecektir.

6.1. Bulanık Mantık Denetleyici

Klasik kontrol sistem tasarımındaki ilk adım, kontrol edilecek sistemin transfer fonksiyonunun tam olarak elde edilmesidir. Başka bir deyişle matematiksel modelinin oluşturulmasıdır. Ayrıca uygun ve etkin bir kontrol sağlamak için sistem parametrelerinin zamanla değişmemesi istenir. Fakat uygulamadaki sistemlerin pek çoğu bilinmeyen parametrelere veya kompleks ve lineer olmayan karakteristiklere sahiptirler. Bu tip uygulamalarda, uygulama alanındaki uzman kişilerin bilgisi klasik kontrol sistemlerinden daha faydalı ve etkili olabilmektedir (Erkaymaz, 2010).

Bu yüzden insan düşünme yeteneğini ve bilgisini kontrol sisteminin içine sokabilecek bir kontrol yöntemi olan Bulanık Mantık Denetleyicisi (BMD) (*Fuzzy Logic Controller, FLC*) iyi bir çözüm olabilmektedir. Bulanık mantık yaklaşımı, makinelere insanların özel verilerini işleyebilme ve onların deneyimlerinden ve önsezilerinden yararlanarak çalışabilme yeteneği verir. Bu yeteneği kazandırırken sayısal ifadeler yerine sembolik ifadeler kullanılır. İşte bu sembolik ifadelerin makinelere aktarılması matematiksel bir temele dayanır. Bulanık mantık, klasik mantığın aksine iki seviyeli değil, çok seviyeli işlemleri kullanmaktadır (Elmas, 2003).

Özetle, BMD bilgi tabanlı kontrol metodudur ve genellikle yeterli kesinlikte bilgiye ulaşılamayan komplex sistemlerin denetimi ve sistem hakkında kesin ölçüm bilgilerine ulaşmanın pratik olmadığı durumlarda kullanılırlar.



Şekil 6.1. Temel bulanık mantık denetleyici yapısı

Temel olarak BMD sistemleri Şekil 6.1'de görüldüğü gibi dört temel arabirimden oluşur. Burada bilgi tabanı; uygulanacak olan sistemin değişkenlerini ve bu değişkenler hakkındaki bilgileri içerir. Bulanıklaştırma arabirimi, gelen bilgilerin dilsel niteleyicilere dönüştürüldüğü bölümdür. Sonuç çıkarım mekanizması, giriş ve çıkış bulanık değerlerinin incelenerek bulanık kontrol kurallarının değerlendirildiği bölümdür. Durulaştırma arabirimi ise bulanık değerlerin sayısal değerlere dönüştüğü bölümdür (Passino, 1998; Yen, 1995, Özkan, 2011).

Bulanık mantık kontrolün avantajları kısaca aşağıdaki gibi açıklanabilir (Kıyak, 2003).

1. İnsanın kontrol stratejisini taklit ettiğinden, anlaşılması kolaydır.
2. Gerçekleştirilmesi basittir, çok sayıda paralel işleme izin verir.
3. Diğer sistemlere göre yazılımları basit olduğundan, uygulamalar ucuza mal olmakta ve yapılan işler basitleşmektedir. Bu yüzden bulanık kontrol yüksek performans/maliyet oranına sahip bir yaklaşımdır.

BMD ile kontrol sistemi tasarımı yapılırken sırasıyla aşağıdaki adımlar izlenmelidir (Elmas, 2003)

a. Kontrol girişlerinin tanımlanması
b. Herbir giriş için bulanık kümelerin üyelik fonksiyonlarının tanımlanması
c. Bulanık kuralların tanımlanması
d. Kontrol çıkışlarının tanımlanması
e. Herbir çıkış için bulanık kümelerin üyelik fonksiyonlarının tanımlanması
f. Bulanık çıkarımın belirlenmesi
g. Durulaştırma işleminin belirlenmesi

Kontrol edilmek istenen sistemden bulanık kuralların çıkarılması için dört değişik metod kullanılır:

1. *Bir uzmanın tecrübesi ve/veya kontrol mühendisliği bilgisi*: Bunun için uzmana sistem hakkında sorular sorulur ya da sistem hakkında uzmanın tüm bilgisini ortaya koyması istenir. Günlük hayatımızda bizim kararlarımızı ilgilendiren bilgilerin çoğu doğal olarak sayısaldan çok dilbilimselliğe dayanmaktadır. Bu çerçevede bulanık kontrol kuralları insan davranışını ve karar analizini karakterize etmek için doğal bir yapı sağlamaktadır. Çoğu uzmanlar bulanık kontrol kurallarının onların bilgi alanlarını ortaya çıkarmak için uygun bir yol sağladığını bulmuşlardır.
2. *Operatörün kontrol hareketlerinin modellenmesi.* Bu yöntemde güvenirliği sağlayan tek unsur operatörün süreç üzerindeki deneyimi ve bilgisidir.Genellikle sistemin matematiksel modelinin elde edilmesinin mümkün olmadığı durumlarda kullanılır.
3. *Sürecin bulanık modeli.* Sistemin dinamik karakteristiğinin dilsel anlatımı bulanık modeli meydana getirir.
4. *Öğrenen algoritmalar kullanılır:* Kontrolör kendi kural tabanını oluşturur. İnsanın öğrenme yeteneği simüle edilir. Yapay sinir ağları, kendi kendine ayarlama (self tuning) ve kendi kendine yapılanmalar (self organization) böyle sistemleri oluşturmak için kullanılmaktadır.

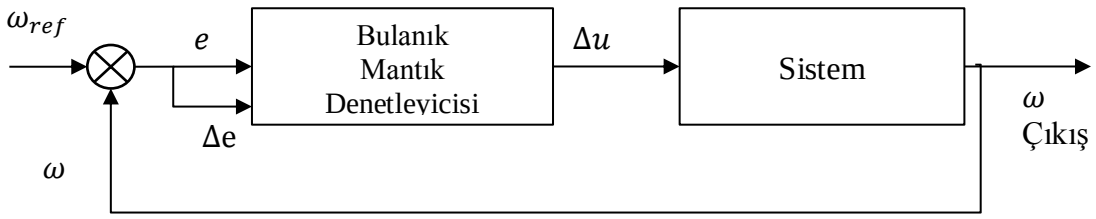
Yapılan tez çalışmasında EMF sistemin literatürdeki deney bilgileri kullanılarak, BMD yapısı ile EMF'ye giren parçacık konsantrasyonu ve endüstriyel sıvının akış hızı giriş parametrelerine göre EMF bobinine uygulanan akımı kontrol eden bir denetleyici yapısı oluşturulmuştur.

6.2. PI Tip Bulanık Mantık Denetleyici

Şimdiye kadar ki BMD'lerin yapısının iki ana türü bulunmaktadır. Bunlardan ilki pozisyon tip bulanık kontrolör olup, bu kontrolör tipinde kontrol girişi (u), hata (e) ve hata değişimine (Δe) göre belirlenmektedir. İkincisi hız türü kontrolör olup kontrol girişi değişimi (Δu), hata ve hata değişimine göre belirlenmektedir. Pozisyon tip BMD'ler PD tip BMD, hız tipi BMD'ler PI tip BMD olarak isimlendirilmektedirler. PI tip BMD'nin uygulanması PD tip BMD'lere göre daha pratiktir. PD tip BMD ile çoğu

sistem sınıflarında kalıcı durum hatasını kaldırmak mümkün olamamaktadır (Lee, 1993; Gopala, 2010).

Bir PI tip BMD tipik olarak Şekil 6.2' de görüldüğü gibi kapalı döngü kontrol sistemi şeklindedir. Sistem değişkenleri denetlenen sistemden ölçülen giriş değişkenleri hata (e) ve hatadaki değişim (Δe) ve süreci kontrol etmek için Bulanık Mantık Denetleyicisi tarafından kullanılan çıkış değişkenleri (Δu) olmak üzere iki ana çeşittir. Kuralların ifadesinde kullanılan her bir sistem değişkeni için izin verilen değerler uygun evrensel kümede bulanık değerler olarak tanımlanmaktadırlar. Bu kümelerin tanımlanmaları tasarım işleminde en kritik adımlardan biridir ve sistem performansını doğrudan etkilemektedir (Akçayol, 2000).



Şekil 6.2. PI Tip BMD genel yapısı

Bu tez çalışmasında EMF performansının istenen performans değerinden farkı olan performans hatası ve performans hata değişimine göre EMF sistemini kontrol edebilen bir denetleyici yapısı oluşturulmuştur.

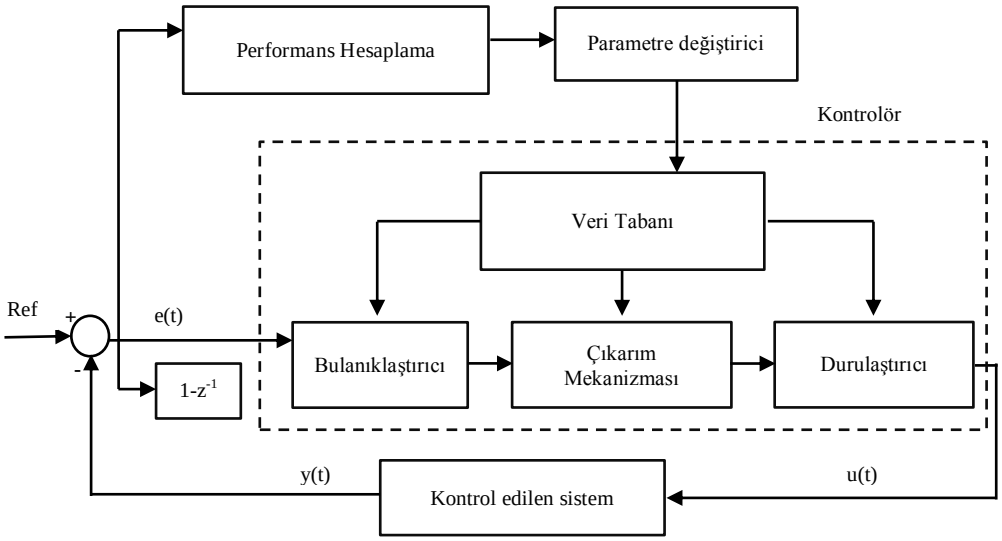
6.3. Kendini Organize Edebilen Bulanık Mantık Kontrolör

Adaptif olmayan bulanık kontrol ile uygulamalarda değerini kanıtlamış olsa da, bazı sistemler için kural tabanı belirlemek zor olabilir yada sistem değişiklerinde kural tabanı parametrelerinin ayarlanması gerekebilir. Bu zorluklar bulanık kontrolörün online olarak sentez ve ayar edilmesine odaklanan bir adaptif BMD gerçekleştirilmesi için gerekli motivasyonu sağlamaktadır.

Adaptif BMD'in geleneksel adaptif kontrol yapıları karşısındaki en büyük avantajı insan operatörleri tarafından oluşturulan dilsel bulanık bilgileri içerme yeteneğine sahip olmasıdır (Layne, 1996) . Bu durum, kimyasal süreçler veya uçak kontrolü gibi teorik açıdan bakıldığında zor olmasına rağmen insan operatörleri

tarafından başarıyla kontrol edilebilen yüksek derecede belirsizlik içeren sistemler için özellikle önemlidir (Alçı ve ark., 2003). İlk Adaptif bulanık kontrolör Procyk ve Mamdani tarafından (1979) kendini organize edebilen (self-organizing controller -SOC) olarak tanıtıldı. Literatürde bu yöntemin uygulamalarının başarılı sonuçlar verdiği gösterilmiştir (Kwong, 1995;. Lennon 1995; Chang, 2006).

Kendi kendini organize edebilen BMD Şekilde görüldüğü gibi, alt seviyede bir kural tabanlı kontrolör ve üst seviyede bir ayarlama mekanizmasından oluşur.



Şekil 6.3. Kendi kendini organize edebilen BMD yapısı

Alt seviyede hata (*e*) ve hata değişimi (Δ*e*) gibi iki girişe sahip bir kontrolördür. Çıkış kazancı ile çarpılarak oluşturulan *u* kontrolün çıkışıdır (Jantzen, 1998). Kendi kendini organize eden BMD'yi oluşturan fikir kontrolörün mevcut performansına göre kontrolörün kural tabanının bir ayarlama mekanizması ile güncellemektir. Teme olarak, eğer performans kötü ise sorumlu kural değiştirilecektir, böylece kural sonraki çağırımda daha iyi kontrol sinyali üretecektir.

Üst seviyede hata (*e*) ve hata değişimi (Δ*e*) izlenir ve gerektiğinde kural tabanı bir modifiye algoritması ile değiştirilir. Her bir değişikliğin büyüklüğüne karar vermek için bir performans ölçüsü kullanılır. Performans ölçüler P tablosunda sayısal olarak tutulur ve geçici yanıta göre istenen değere ulaşılıp ulaşılmadığı kontrol eder. P tablosu dilsel kurallar ile belirlenir fakat sıklıkla deneme yanılma yoluyla oluşturulmaktadır. Süreci hakkında bir bilgi mevcut değilse aynı performans tablosu farklı işlemlerde kullanılabilir.

P performans tablosu mevcut durumu değerlendirir ve $P(i_n,j_n)$ performans ölçüsü geri döndürmektedir. Bu performans ölçüsünde i_n e'ye karşılık geçen indeksi, j_n Δe'ye karşılık gelen indeksi göstermektedir. Çizelge 6.1'de bir performans tablo örneği verilmiştir.

Çizelge 6.1. Örnek performans tablosu (Procyk, 1979)

e/Δe	**-6**	**-5**	**-4**	**-3**	**-2**	**-1**	**0**	**1**	**2**	**3**	**4**	**5**	**6**
-6	-6	-6	-6	-6	-6	-6	-6	0	0	0	0	0	0
-5	-6	-6	-6	-6	-6	-6	-6	-3	-2	-2	0	0	0
-4	-6	-6	-6	-6	-6	-6	-6	-5	-4	-2	0	0	0
-3	-6	-5	-5	-4	-4	-4	-4	-3	-2	0	0	0	0
-2	-6	-5	-4	-3	-2	-2	-2	0	0	0	0	0	0
-1	-5	-4	-3	-2	-1	-1	-1	0	0	0	0	0	0
0	-4	-3	-2	-1	0	0	0	0	0	1	2	3	4
1	0	0	0	0	0	0	1	1	1	2	3	4	5
2	0	0	0	0	0	0	2	2	2	3	4	5	6
3	0	0	0		2	3	4	4	4	4	5	5	6
4	0	0	0	2	4	5	6	6	6	6	6	6	6
5	0	0	0	2	2	3	6	6	6	6	6	6	6
6	0	0	0	0	0	0	6	6	6	6	6	6	6

Eğer performans ölçü değeri sıfır ise durumun tatmin deici olduğunu gösterir. diğer durumlarda ise istenmeyen durumlar derecelendirilmiştir ve kontrol sinyalinin değişmesi gerektiğini belirtir. İlgili kuralın değiştirilmesi için Denklem 6.1 kullanılır.

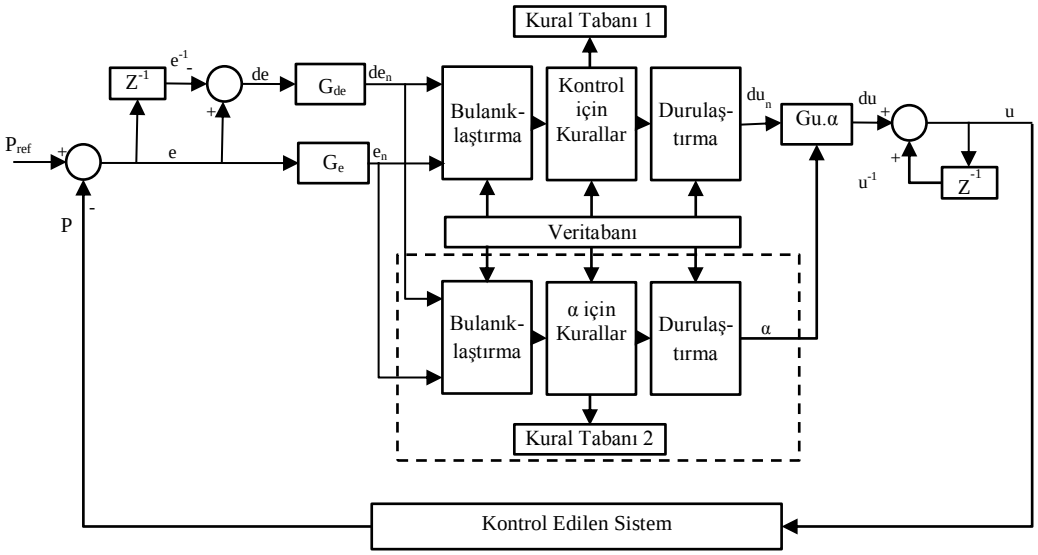
$$u_n = u_{n-d} + p_n \tag{6.1}$$

Burada n zaman indisi olup geçerli örneği gösterir. Performans tablosundan gelen ceza miktarının eklenmesiyle kuralın yeni değeri belirlenmektedir.

6.4. Kendini Ayarlayabilen PI tip Bulanık Mantık Kontrolör

Bulanık kontrolörler arasında PI tip bulanık kontrolörler en yaygın ve pratik olanıdır. PI tip BMD performansının birinci dereceden sistemler için oldukça tatmin edici olduğu bilinmektedir. Geleneksel PI kontrolörlerde olduğu gibi PI-tip BMD, uzun ölü zamana sahip ve doğrusal olmayan yüksek dereceli sistemler için büyük aşma ve yüksek salınımları nedeniyle kötüdür.

Mudi ve Pal (2000) çalışmalarında PI tip bulanık kontrolörler için kendini ayarlama mekanizması önermişlerdir. Kendini ayarlayabilen bulanık Mantık kontrolör (KABK) (*Self organizing PI type fuzzy controller*) adı verilen bu sistem yetenekli bir insan operatörünün hata ve hata değişimine karşılık sistem sürecinin en iyileme çalışmasına dayanmaktadır. İnsan operatör yapılan bu işlem oldukça karmaşıktır ve matematiksel olarak modellemesi çok zordur. Önerilen kontrolör sitem üzerindeki performans ve karalılığa etkisi nedeniyle ölçeklendirme katsayısını değiştirmeye yöneliktir. Bu kontrolör ile her bir örnekleme durumunda ölçeklendirme katsayısını kazanç güncelleme ile değiştirmektedir. Şekil 6.4'de KABK yapısı verilmiştir (Mudi ve Pal, 2000).



Şekil 6.4. KABK yapısı

Sistemin kontrolü için PI-tip BK ile değişim (e) ve bu değişimin türevi (de) giriş değişkenleri olarak kullanılırken, kontrol çıkışındaki değişim miktarı (du) çıkış olarak kullanılmaktadır. Örnekte görüldüğü gibi giriş ölçeklendirme katsayıları sabit tutulurken çıkış ölçeklendirme katsayısı değişime bağlı olarak kazanç güncelleme katsayısı (α) ile sürekli olarak ayarlanmaktadır. KABK'nın çıkış ölçeklendirme katsayısı Denklem 6.2'de belirtildiği gibi her örnekleme periyodunda α gibi bir kazanç güncelleme katsayısının çarpımı ile elde edilmektedir.

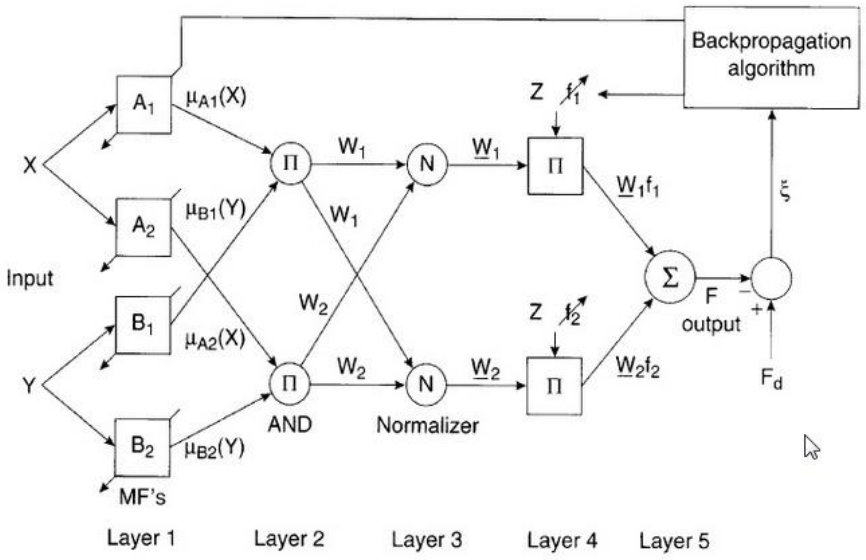
$$du = (\alpha . G_u) . du_n \tag{6.2}$$

BK'nın çıkış ölçeklendirme katsayısını ayarlayan kazanç ayarlama sistemi kesikli çizgiler ile Şekil 6.4'de belirtilmiştir.

6.5. Adaptif ANFIS Kontrolör

Hibrid zeki kontrolörler en az iki adet yapay zekanın alt disiplin ve yöntemlerini bir arada kullanan bir sistemi ifade etmektedir. Örneğin bulanık mantık ile yapay sinir ağının bir arada kullanılması ile oluşan bulanık sinir ağı (ANFIS) bir hibrit zeki kontrolörü ifade etmektedir.

ANFIS optimal kontrol hedeflerine ulaşmak için kendi kendine öğrenme yeteneği ile bulanık kontrolü geliştirmektedir. Şekil 6.5'de ki adaptif ağ yapısında her düğüm gelen sinyalleri işleyerek çok katmalı ileri beslemeli bir yapı oluşturmaktadır. Farklı adaptif yetenekleri yansıtmak için düğümlerde kare ve daire sembolleri kullanılmıştır. Bir daire düğüm sabit kısmı, kare düğüm adaptif kısımları belirtmektedir. Adaptif düğümlerin parametreleri ağın parametre kümesini oluşturmaktadır. ANFIS için en yaygın tasarım teknikleri YSA'dan doğrudan elde edilmektedir. Hedeflenen çıkış durumuna göre ağ öğrenme algoritması ile güncellenmektedir (Mar, 2001).



Şekil 6.5. ANFIS adaptif ağ yapısı (Bose, 2001)

Sistemi eğitmek amacıyla sistem hatası olan ξ ile geri yayılım algoritması kullanılarak eğitim yapılmaktadır. ANFIS yapısını online olarak ayarlamak için

genellikle hatanın karesi (Denklem 6.3) şeklinde verilen bir fonksiyon kullanılarak gradyan azaltma tekniği ile hata azaltılır, bu işlem esnasında hata çıkış katmanından giriş katmanına doğru yayılmaktadır. Bu işlem geri yayılım olarak adlandırılır. ANFIS eğitim algoritması öncül parametre ayarlanması ve sonuç parametrelerinin ayarlanması olmak üzere iki safhada tamamlanır.

$$\xi = (F_d - F_o)^2 \tag{6.3}$$

Önşart parametrelerinin ayarlanması için Denklem 6.4'de verilen geri yayılım ifadesi kullanılır. Burada α önşart parametresini, i üyelik fonksiyonu sayısını, η öğrenme oranını temsil etmektedir.

$$\Delta\alpha_{A_i} = -\eta \frac{\partial \xi^2}{\partial \alpha_{A_i}} \tag{6.4}$$

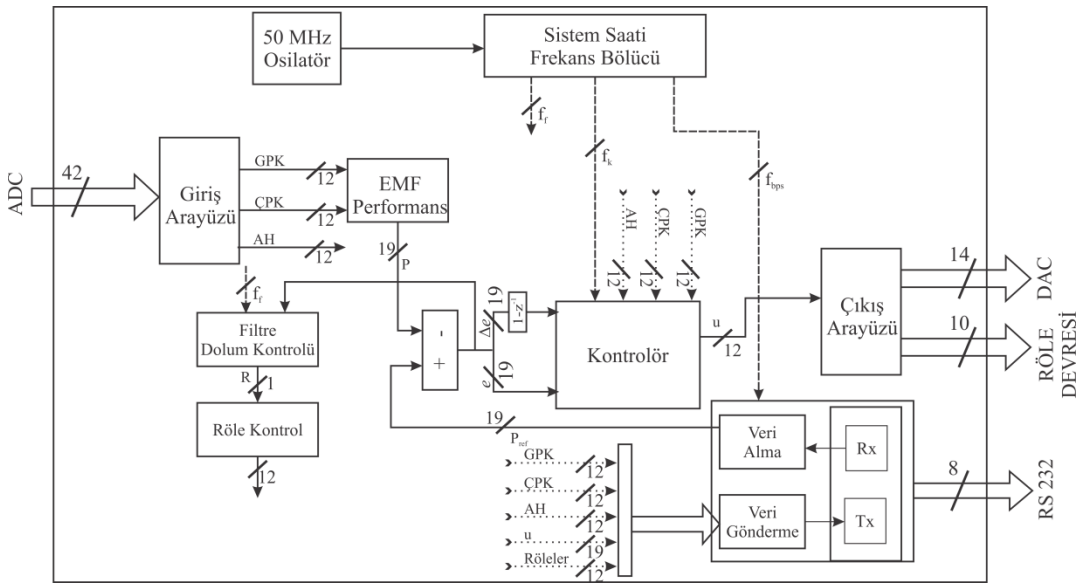
Sonuç parametrelerinin ayarlanması için Denklem 6.5'de verilen geri yayılım ifadesi kullanılır. Burada a sonuç parametresini, η öğrenme oranını temsil etmektedir.

$$\Delta a_{O_i} = -\eta \frac{\partial \xi^2}{\partial a_{O_i}} \tag{6.5}$$

7. EMF KONTROLÖR BİLEŞENLERİ

7.1. FPGA Yapısında Kullanılan Temel Bileşenler

EMF kontrolörü ALTERA DE2-70 FPGA geliştirme kartı kullanılarak programlanmıştır. Tasarlanan tüm EMF kontrolör yapılarında sistem saatini sağlayan frekans bölücü, giriş arayüzü, çıkış arayüzü, RS232 giriş ve çıkış arabirimleri, EMF performans modülü, filtre dolum kontrol modülü ve röle kontrol modülü ortak olarak kullanılmıştır. Şekil 7.1'de kontrol yapısını oluşturan temel bileşenler ve bağlantı durumları verilmiştir.



Şekil 7.1. EMF kontrolör yapısı bileşenleri

7.1.1. Sistem saati bileşeni

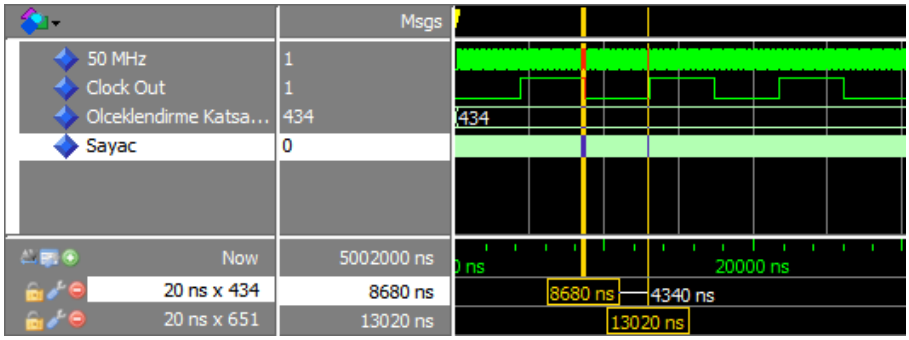
FPGA DE2-70 geliştirme kartı 50 MHZ'lik bir osilatöre sahiptir. Filtre dolum kontrol ve RS232 seri iletişim bileşenlerinin daha küçük saat frekansına sahip olması gerekmektedir. Bu bileşenlerde kullanmak üzere sistem saat frekansını istenilen saat frekanslarına dönüştürmek amacıyla frekans dönüşümünü sağlayan sistem saati modülü oluşturulmuştur. Bu modül bir ölçeklendirme faktörü ve sayaç ile oluşturulurmuştur. Burada ölçeklendirme katsayısı Denklem 7.1'de verilen giriş frekansı ile çıkış frekansı arasındaki ilişkidir.

$$Ölçeklendirme\ katsayısı = \frac{f_g}{f_c} \quad (7.1)$$

Örneğin 115200 bps RS232 biriminde kullanmak için ölçeklendirme katsayısı 434 olarak bulunur (Denklem 7.2).

$$Ölçeklendirme\ katsayısı = \frac{50\ MHz}{115.2\ KHz} = 434 \quad (7.2)$$

Şekil 7.2'de 115.2 KHz'lik frekans bölücü için modelsim simülasyon çıktısı verilmiştir.



Şekil 7.2. Frekans bölücü simülasyon görüntüsü

Tez çalışmasında kullanılan bu bileşen ile RS232 ile işlem sağlayan birime, kontrolöre ve filtre dolum kontrol birimine saat frekansları sağlanmaktadır.

7.1.2. Giriş arayüzü bileşeni

Giriş arayüzü bileşeni ile ADC/DAC kartında bulunan ICL7109 entegresi ile paralel olarak sayısal hale çevrilen Giren Parçacık Konsantrasyonu (GPK), Çıkan Parçacık Konsanstrasyonu (ÇPK) ve Endüstriyel Sıvının Akış Hızı (AH) verileri DE2-70 FPGA kartı üzerinde bulanan genişleme yuvaları aracılığı ile sisteme sinyaller şeklinde aktarılmaktadır.

DE2-70 FPGA geliştirme kartı üzerinde bulunan genişleme yuva gösterimi Şekil 4.3'de verilmiştir. Bölüm 4.4.4'de verilen ADC/DAC kartı üzerindeki IDE yuvaları aracılığı ile aktarılan GPK, ÇPK ve AH sensör verilerinin FPGA karta aktarılmasında kullanılan pin atamaları Çizelge 7.1'de verilmiştir. Çizelge 7.1'de verilen GPK ve ÇPK sensörleri "0" nolu genişleme yuvasına, AH sensörü ise "1" nolu genişleme yuvasına

bağlı olup, pin atamaları en yüksek değerli bitten en düşük değerli bite doğru verilmiştir.

Çizelge 7.1. Giriş arayüz bileşeni için pin atamaları

GPK		ÇPK		AH	
Bit	Pin	Bit	Pin	Bit	Pin
11	GPIO_0(18)	11	GPIO_0(5)	11	GPIO_1(5)
10	GPIO_0(16)	10	GPIO_0(7)	10	GPIO_1(7)
9	GPIO_0(24)	9	GPIO_0(9)	9	GPIO_1(9)
8	GPIO_0(22)	8	GPIO_0(11)	8	GPIO_1(11)
7	GPIO_0(12)	7	GPIO_0(13)	7	GPIO_1(13)
6	GPIO_0(10)	6	GPIO_0(14)	6	GPIO_1(14)
5	GPIO_0(8)	5	GPIO_0(15)	5	GPIO_1(15)
4	GPIO_0(6)	4	GPIO_0(17)	4	GPIO_1(17)
3	GPIO_0(4)	3	GPIO_0(19)	3	GPIO_1(19)
2	GPIO_0(2)	2	GPIO_0(21)	2	GPIO_1(21)
1	GPIO_0(1)	1	GPIO_0(23)	1	GPIO_1(23)
0	GPIO_0(0)	0	GPIO_0(25)	0	GPIO_1(25)

7.1.3. Çıkış arayüzü bileşeni

Çıkış arayüzü bileşeni aracılığı ile EMF'nin kontrolör çıkışı olan PWM referans gerilimi, elektropnömatik valfleri, EMF bobinleri ve invertörün kontrolünde kullanılan bağlantıları sağlanmaktadır. PWM referans gerilimi DE2-70 geliştirme kartı üzerinde bulunan "1" nolu geliştirme yuvası aracılığı ile Bölüm 4.4.4'de verilen ADC/DAC kart üzerinde bulunan DAC8562 entegresine aktarılmaktadır. EMF kiti üzerinde bulunan 8 adet elektropnömatik valfi ve aktif EMF'yi belirlemek amacıyla kullanılan kontrol bitleri geliştirme yuvaları aracılığı ile Bölüm 4.4.5'de verilen röle kartına aktarılmaktadır. Çıkış arayüzü modülü için genişleme yuvasında kullanılan pin atamaları Çizelge 7.2'de verilmiştir.

Çizelge 7.2. Çıkış arayüzü için pin atamaları

PWM		Röle kartı	
Bit	Pin	Kontrol Edilen	Pin
11	GPIO_1(4)	Valf 1	GPIO_0(27)
10	GPIO_1(6)	Valf 2	GPIO_0(29)
9	GPIO_1(8)	Valf 3	GPIO_0(31)
8	GPIO_1(10)	Valf 4	GPIO_0(28)
7	GPIO_1(12)	Valf 5	GPIO_0(30)
6	GPIO_1(0)	Valf 6	GPIO_1(27)
5	GPIO_1(1)	Valf 7	GPIO_1(31)
4	GPIO_1(16)	Valf 8	GPIO_1(29)
3	GPIO_1(18)	EMF 1	GPIO_1(22)
2	GPIO_1(26)	EMF 2	GPIO_1(24)
1	GPIO_1(28)	İnvertör	GPIO_1(20)
0	GPIO_1(30)		

7.1.4. EMF performans bileşeni

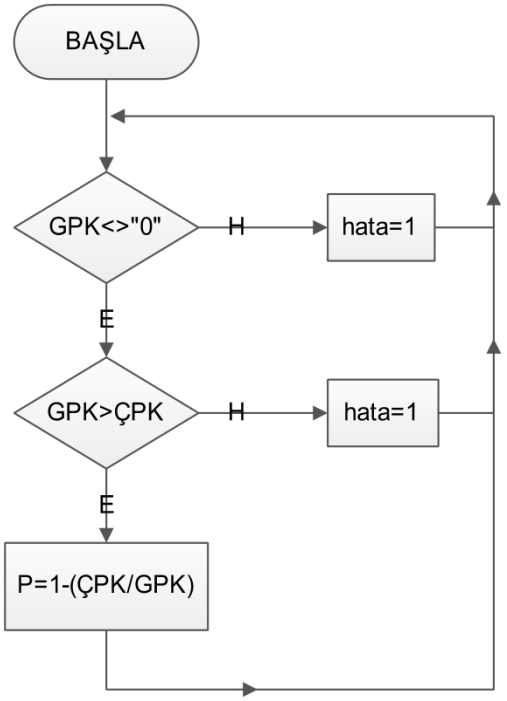
EMF performansı; Denklem 7.3'de belirtildiği gibi elektromanyetik filtreden çıkan parçacık konsantrasyonunun elektromanyetik filtreye giren parçacık konsantrasyonuna oranı kullanılarak hesaplanmaktadır.

$$\psi = 1 - \frac{ÇPK}{GPK} \tag{7.3}$$

EMF performans bileşeninde giriş arayüzünden alınan GPK ve ÇPK verilerine göre EMF performansının hesaplanması yapılmaktadır. Hesaplanan EMF performansı kontrol için denetleyiciye ve filtrenin dolup dolmadığının tespiti için filtre dolum kontrol bileşenine aktarılmaktadır.

EMF performans bileşeninde GPK ve ÇPK değerleri sabit noktalı sayı formatına dönüştürülerek işlem yapılmaktadır. Modülün çıkışında oluşan performans değeri Q7.7 sabit noktalı sayı formatındadır.

EMF performans bileşeninde GPK sıfır olması durumunda oluşacak olan sıfıra bölme hatası ve ÇPK'nın GPK'dan büyük olması gibi sistemin hata durumları kontrol ettirilmektedir. Ayrıca hata sinyali filtre dolum bileşenine ve bilgilendirme amaçlı olarak DE2-70 FPGA geliştirme kartı üzerindeki lede aktarılmaktadır. Modülün akış diyagramı Şekil 7.3'de verilmiştir.

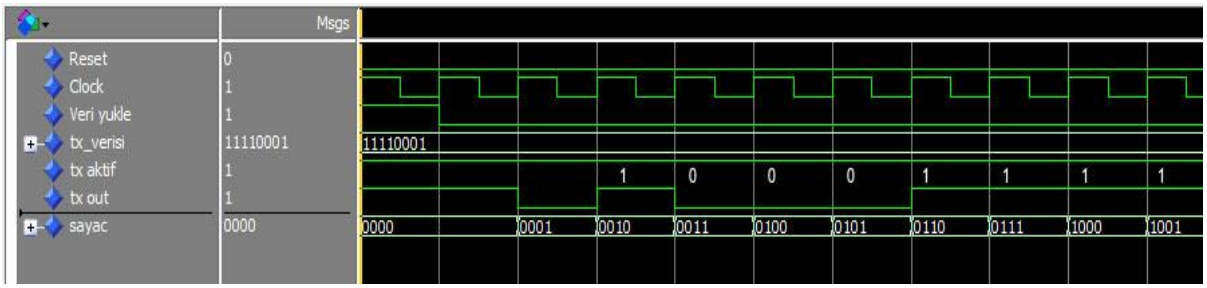


Şekil 7.3. EMF performans bileşeni akış diyagramı

7.1.5. RS232 verici bileşeni

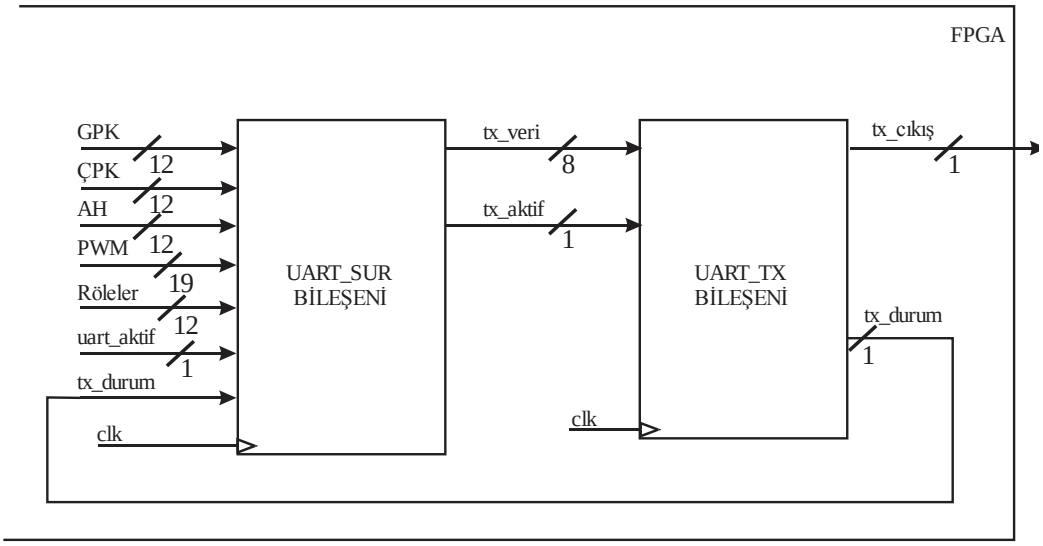
EMF kontrol sistemi verilerinin bilgisayara aktarılması için seri RS232 portu kullanılmıştır. Senkron seri iletişimde, iletişim boyunca aynı faz ve frekansa sabit bir saat bulunmalıdır. Ama seri iletişime dahil olan cihazların iletişimi boyunca aynı saat frekansı ve fazını koruması çok zordur. Bu sorunu çözmek için yapılan çalışmada Evrensel Asenkron Alıcı Verici seri iletişim kullanılmıştır.

DE2-70 kartından alınan verilerin RS232 portundan gönderilmesi için UART_TX isimli bileşen, Deepak (2013) verilen UART yapısı geliştirilerek tasarlanıştır. Bu bileşene giriş olarak verilen gönderim verisi seri hale çevrilmektedir. Bileşenin modelsim simülasyonu Şekil 7.4'de verilmiştir. Şekilde görüldüğü gibi "11110001" verisi tx_out çıkışı üzerinden seri olarak aktarılmaktadır.



Şekil 7.4. UART_TX bileşeni simülasyon görüntüsü

GPK, ÇPK, AH, valflerin durumları ve kontrolör çıkışlarını sırasıyla bilgisayara aktarmak için *UART_SUR* isimli bileşen kullanılmıştır. *UART_SUR* bileşeni kendisine giriş olarak verilen sinyalleri 8 bitlik parçalara ayırarak sırasıyla *UART_TX* bileşenine aktarmaktadır. Bileşen tüm veriler aktarılmadan önce ve aktarımı bittikten sonra başlama ve bitişi simgeleyen 8 bitlik özel veriler eklemektedir. Bu işlem ile bilgisayara aktarılan veriler başlangıç ve bitişinde verilen özel veriler aracılığı ile her bir veri kümesi belirlenip ayrıştırılmaktadır. RS232 verici bileşeninin genel yapısı Şekil 7.5'de verilmiştir. UART bileşeninin VHDL kod yapısı EK-1'de verilmiştir.

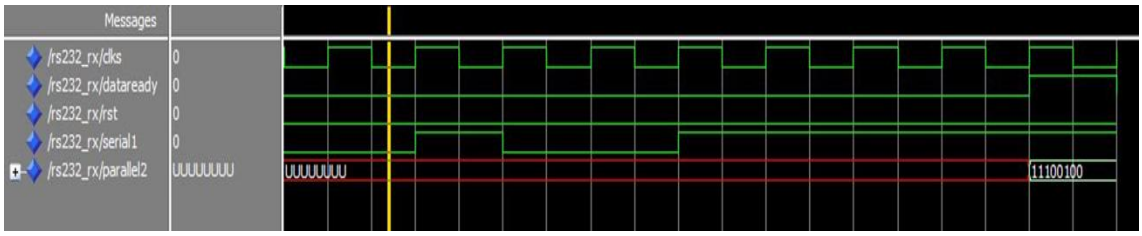


Şekil 7.5. RS232 verici bileşenlerinin genel yapısı

7.1.6. RS232 alıcı bileşeni

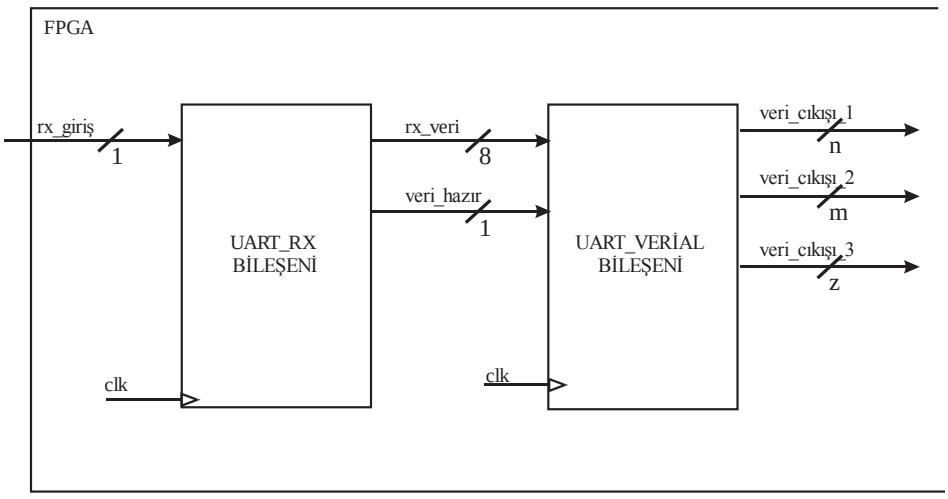
Bu bileşen EMF kiti üzerinde yapılan deneylerde bilgisayar üzerinde oluşturulan program aracılığı ile FPGA tabanlı kontrolör üzerindeki değişkenleri online olarak değiştirebilmek amacıyla tasarlanmıştır. RS232 alıcı bileşeni *UART_RX* ve

UART_VERİAL olmak üzere iki ayrı bileşenden oluşmaktadır. *UART_RX* bileşeni Anonim (2013) tarafından verilen kod yapısı kullanılarak hazırlanmıştır ve bu bileşende RS232 RX veri hattından gelen tek bitlik verileri birer baytlık veriler halinde toplamaktadır. *UART_RX* Bileşenin modelsim simülasyonu Şekil 7.6'da verilmiştir. Şekil 7.6'da görüldüğü gibi seri olarak aktarılan veriler paralel olarak "11100100" şeklinde 8 bitlik veri olarak aktarılmaktadır.



Şekil 7.6. UART_RX bileşeni simülasyon görüntüsü

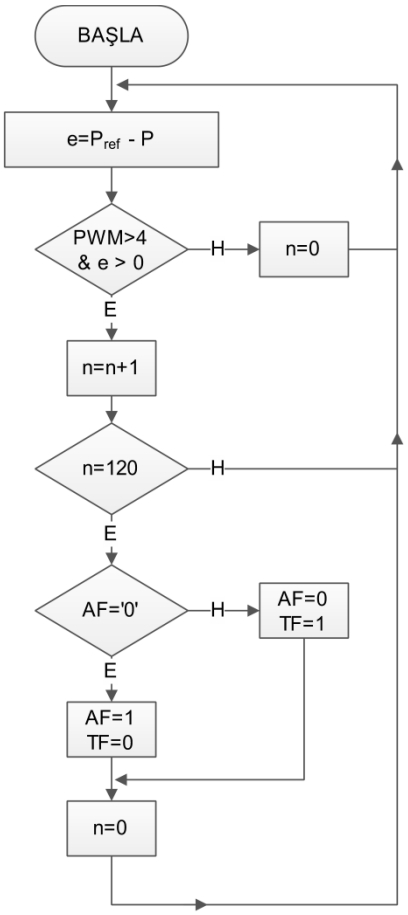
UART_RX bileşeninden alınan bir baytlık veriler *UART_VERİAL* bileşeni tarafından birleştirilerek kontrolör içerisinde kullanılan formata uygun hale getirilmektedir. Örneğin sabit noktalı sayı sisteminde Q12.5 olarak bilgisayar üzerinden FPGA kontrolörüne aktarılacak değişken için 3 baytlık verinin birleştirilmesi ile oluşturulmaktadır. Bilgisayar üzerinden gönderilen veri kümesinin başlangıç ve bitişinin belirlenmesi için 8 bitlik bir veri başlangıç ve bitişe eklenmekte ve bu veriler bu bileşende kontrol edilmektedir. *RS232* alıcı bileşeninin genel yapısı Şekil 7.7'de verilmiştir. Tasarlanan kontrolördeki ihtiyaca göre *UART_VERİAL* bileşeni *n, m* veya *z* bitlik veri çıkışı barındırmaktadır.



Şekil 7.7. RS232 alıcı bileşenlerinin genel yapısı

7.1.7. Filtre dolum kontrol bileşeni

Şekil 3.3'de görüldüğü gibi EMF performansı filtre matrisinin tutma bölgelerinin dolmasına bağlı olarak zamanla düşmektedir. EMF performansı istenilen performans değerinin altına düştüğünde temizlenecek olan sıvının çalışmakta olan sistemin kesintiye uğramaması için diğer manyetik filtreye yönlendirilmesi gerekmektedir. Kullanılan EMF kitinin iki adet manyetik filtreye sahip olması, manyetik filtrenin biri dolduğunda diğer manyetik filtreye geçişine ve dolan manyetik filtrenin de temizlenmesine olanak sağlamaktadır. Bu bileşen, maksimum PWM referans gerilimi verilmesine rağmen EMF performansı istenilen EMF performansına 2 dakika boyunca ulaşamadığında çalışan aktif filtreyi (*AF*) değiştirmekte ve dolan filtreyi (*TF*) de temizlemeye aktarmaktadır. Modülün akış diyagramı Şekil 7.8'de verilmiştir.

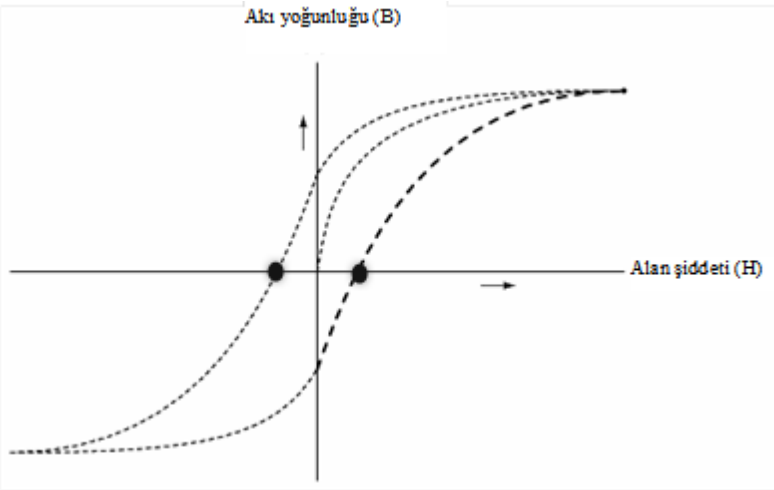


Şekil 7.8. Filtre dolum bileşeninin akış diyagramı

7.1.8. Filtre temizleme ve röle kontrol bileşeni

Bu bileşen filtre dolum bileşeninden gelen aktif filtre (AF) ve temizlenecek filtre (TF) verilerine göre elektropnömatik valflerin ve invertörün kontrolünü sağlamaktadır. AF değeri sıfıra eşit olduğunda Şekil 4.22'de görülen EMF1, bire eşit olduğunda EMF2 çalıştırılmaktadır. AF değerinde değişiklik olduğu durumlarda, EMF1 kullanılırken 4 ve 6 nolu vanalar, EMF2 kullanılıyor iken 3 ve 5 nolu vanalar açılmaktadır.

Temizlenecek olan filtre matrisinin gövdesinde artık mıknatısiyet kalacağından Bölüm 4.1.2'de verilen invertör ile 12 V ve 15 Hz'lik bir AC gerilim elde edilerek temizlenecek filtre bobinine uygulanmaktadır. Böylece artık mıknatısiyetin etkisinin sıfır olduğu anlar elde edilerek filtre içerisindeki manyetik parçacıkların temizlenmesi sağlanmaktadır (Şekil 7.9)



Şekil 7.9. Histerisiz eğrisi

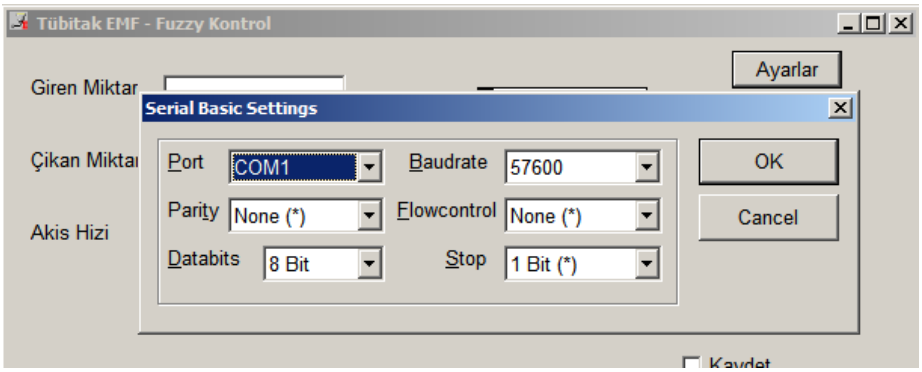
TF değeri sıfıra eşit olduğunda EMF1, bire eşit olduğunda EMF2 temizleme işlemine tabi tutulmaktadır. TF değerinde değişiklik olduğu durumlarda ilgili EMF bobinine invertör aracılığı ile düşük frekanslı AC gerilim uygulanmakta ve aynı zamanda uygun elektropnömatik valfler ile EMF'nin matrisine basınçlı su uygulanmaktadır. Bu uygulama ile filtre matrisini oluşturan küreciklerin tutma bölgelerinde biriken manyetik özellikli parçacıklar koparılarak atık deposuna aktarılmaktadır. EMF durumuna göre hangi vanaların kullanılacağı Çizelge 7.3'de verilmiştir.

Çizelge 7.3. EMF durumuna göre vana kontrolü

EMF Durumu	Filtre	Aktif Vanalar
Filtreleme	EMF-1 (AF=0)	V2, V8
	EMF-2 (AF=1)	V1, V7
Temizleme	EMF-1 (TF=0)	V4, V6
	EMF-2 (TF=1)	V3, V5

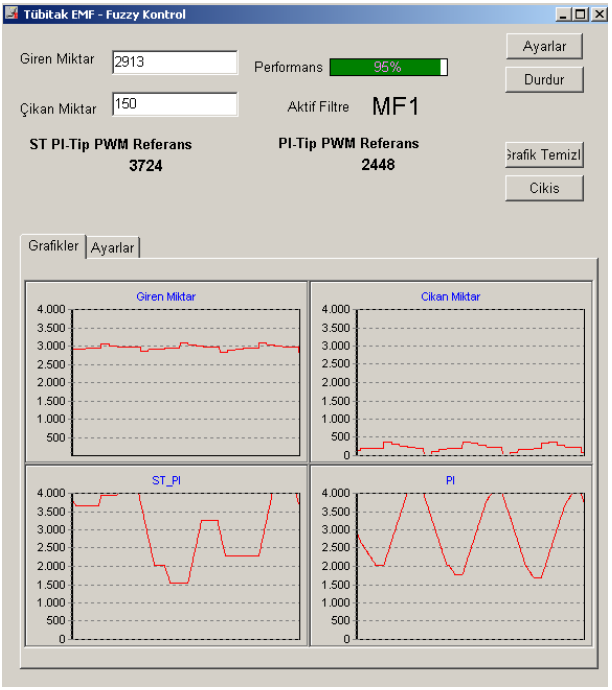
7.2. Bilgisayar Veri Toplama Arayüzü

EMF kiti ile yapılan kontrol denelerinde, deney süresince elde edilen GPK, ÇPK, performans değeri, kontrolör çıkışı ve röle durumlarının değerleri RS232 portu kullanılarak bilgisayara aktarılmıştır. Gelen verileri değerlendirmek ve kaydedebilmek için Delphi programlama dili ile bir yazılım oluşturulmuştur. Oluşturulan bilgisayar yazılımında RS232 portundan gelen verileri programa aktarabilmek için SerialNG isimli bileşen kullanılmıştır (SerialNg, 2013). Bu bileşen aracılığı ile seri iletişim için gerekli olan port, baudrate vb. ayarlamaları yapılmıştır (Şekil 7.10).



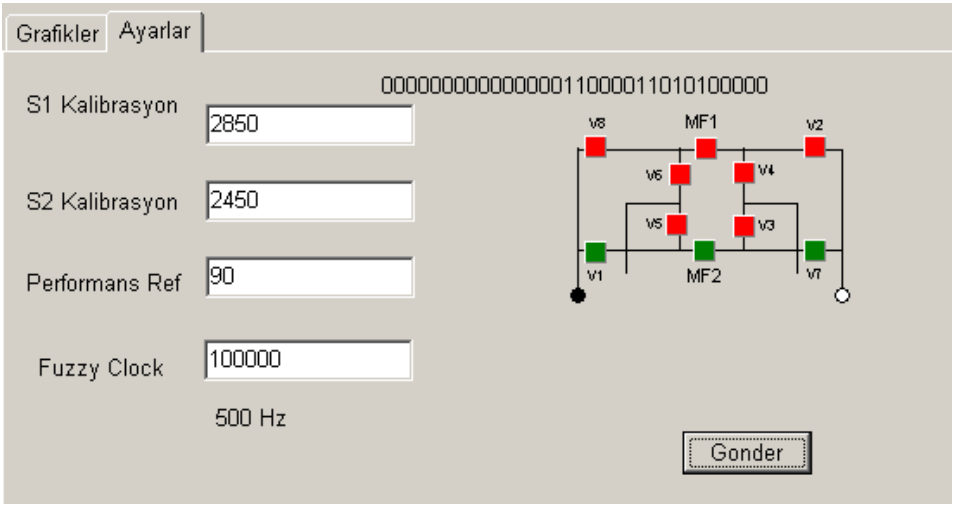
Şekil 7.10. Seri port ayar formu görüntüsü

Oluşturulan yazılım, RS232 üzerinden gelen verilerde öncelikle veri grubunun başlangıcını belirten özel 8 bitlik veriyi kontrol etmektedir. Bu veriyi aldıktan sonra FPGA tabanlı kontrolörden gönderilen n bitlik verinin tamamını bir değişkene aktarılmaktadır. Elde edilen paketlenmiş veri grubu FPGA üzerinde gönderildiği sırada tekrar çözümleme işlemine tabii tutulmaktadır. Yazılan fonksiyonlar ile gelen ikili tabandaki veriler gönderildiği sayı formatına çevrilmektedir. Elde edilen tüm kontrol veriler ekranda değerleriyle ve grafiksel olarak gösterilmektedir (Şekil 7.11).



Şekil 7.11. EMF veri toplama yazılımının ekran görüntüsü.

Elde edilen veriler daha sonra değerlendirilebilmek amacıyla *csv* formatında bilgisayara kayıt edilmektedir. Ayrıca gerçekleştirilen kontrolörde sensör kalibrasyonları, EMF referans performans değeri, kontrolör saat hızı gibi ayarlanabilir parametrelerin değişimi yazılım aracılığı ile yapılmaktadır. Şekil 7.12'de görülen form ile değişiklik yapılacak veriler bilgisayardan FPGA üzerinde bulunan RS232 alıcı bileşenine iletilmektedir. Ayrıca bu kısımda rölelerin durumları da grafiksel olarak izlenebilmektedir.



Şekil 7.12. Veri gönderim formu görüntüsü.

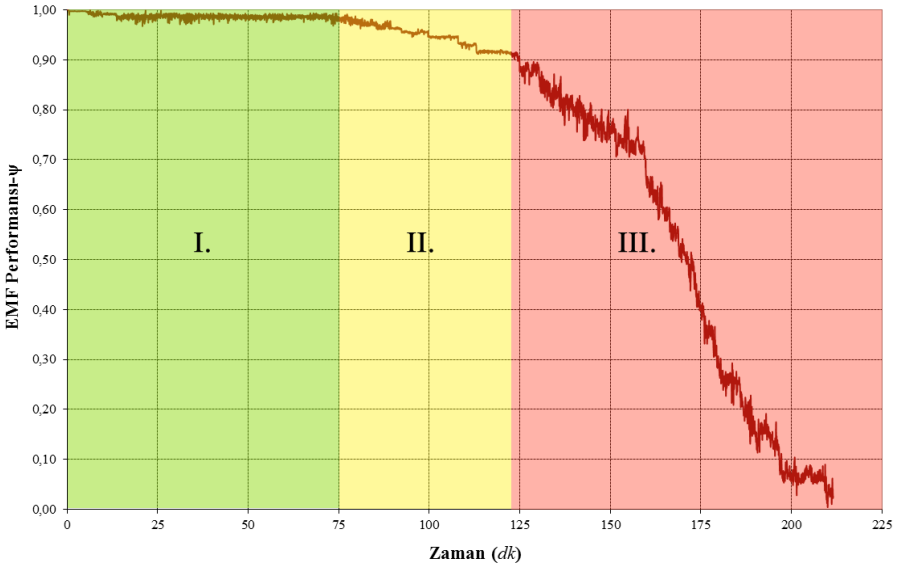
Gerçekleştirilen denetleyicilerden gelen verilerin her biri için değişiklik göstermesi ve ayarlanabilir parametrelerinin de farklı olmasından dolayı kullanılan denetleyiciye göre yazılım tekrar düzenlenerek kullanılmıştır.

8. DENEYSEL ÇALIŞMALAR

Deneysel çalışmalarda Bölüm 4'de verilen TÜBİTAK 1001 proje kapsamında gerçekleştirilen EMF kiti kullanılarak deney düzeneği hazırlanmış, verileri kayıt edebilmek için bilgisayar ile bağlantıları yapacak arabirim oluşturulmuştur. EMF kitini kontrol edebilmek için uygun kontrolör yapıları FPGA üzerinde programlanmıştır. EMF kiti üzerinde FPGA tabanlı kontrolör ile 14 mm çaplı filtre matrisi elemanları kullanarak kontrol deneyleri yapılmıştır.

Bu bölümde EMF kontrolü için gerçekleştirilen denetleyici yapıları ve bu denetleyiciler ile yapılan deney sonuçları verilmektedir. Deneysel çalışmada ilk önce sabit akı altında sistemin temiz sıvı ile çalıştırılması ile ilgili sonuçlar daha sonra EMF kontrolü için gerçekleştirilen denetleyici yapıları ve yapılan deneyler sırasıyla verilmiştir.

EMF'lerin sabit akı ile çalışmasında zamana bağlı olarak performans değişim grafiği Şekil 8.1'de görüldüğü gibidir. EMF performans değişimi üç bölgeye ayrılabilir. Bölge I, EMF performansının yüksek olduğu ve dolayısıyla manyetik akının yeterli olduğu alandır. Bölge II, EMF matrisi tutma bölgelerinde manyetik parçacıkların tutulamayıp ÇPK'nın yükselmeye başladığı diğer bir ifade ile manyetik akının yetersiz kaldığı bölgedir. Bölge III, istenmeyen durum olup tutma bölgelerinde artık gelen parçacıklar tutulamayarak çıkışa yansıdığı bölgedir. Bu durum filtre matrisinin dolduğu anlamını taşımaktadır. Filtre matrisinin dolması gerçek anlamda fiziksel olarak filtre matrisi elemanlarının arasındaki boşlukların dolması anlamına gelmeyip giren parçacıkların büyük oranda filtre matrisi çıkışına yansıması anlamındadır. Bu çalışmada EMF'nin dolması bu anlamda kullanılmıştır (Saritas, 2007; Saritas, 2008; Saritas, 2010; Ozkan, 2012).



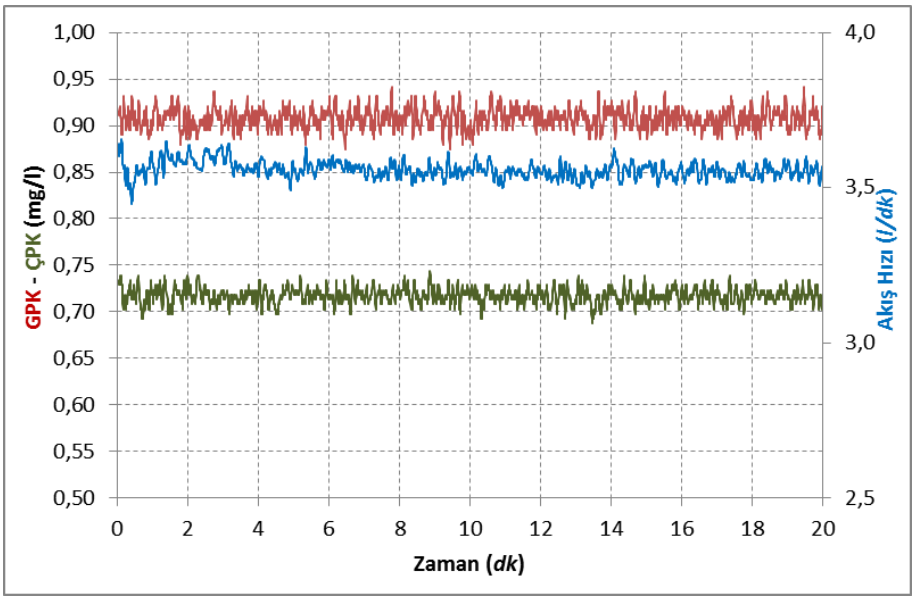
Şekil 8.1. EMF performansı değişimi (Saritas, 2008).

Bölge II'de EMF'nin performansının düşmeye başlaması EMF'nin çıkan parçacık konsantrasyonunun artması anlamına gelmektedir. Bu durum manyetik akı kontrolü ile giderilebilmektedir. Bu çalışmadaki deneyler Bölge II çalışma ortamında gerçekleştirilmiştir.

EMF'nin manyetik özellikli gövdesindeki manyetik akı yoğunluğu; EMF bobinine uygulanan gerilim ve dolayısıyla bobinin akımı ile orantılıdır. EMF'nin bobin gerilimi ise kontrolör PWM sinyali ile belirlenmektedir. Yapılan deneylerin değerlendirilmesinde bu durum göz önüne alınarak deney süresince bilgisayar ortamına aktarılan PWM referans gerilimi dikkate alınmıştır.

8.1. Sistemin Temiz Sıvı ile Çalıştırılması

Gerçekleştirilen bu deneysel çalışmada EMF şebeke suyu ile bobinlere enerji uygulanmadan çalıştırılmıştır. 14 mm çaplı filtre elemanları kullanılarak yapılan ölçümler Şekil 7.1'de verilen FPGA tabanlı denetleyici yapısı ile parçacık konsantrasyon verileri alınmıştır. Şebeke suyu ile çalışan filtrede 20 dk süren deney yapılmıştır. Deneyde zamana bağlı olarak GPK, ÇPK ve AH'nın değişimi Şekil 8.2'de verilmiştir.



Şekil 8.2. Şebeke suyu ile çalışan filtrede GPK, ÇPK ve AH değişimi

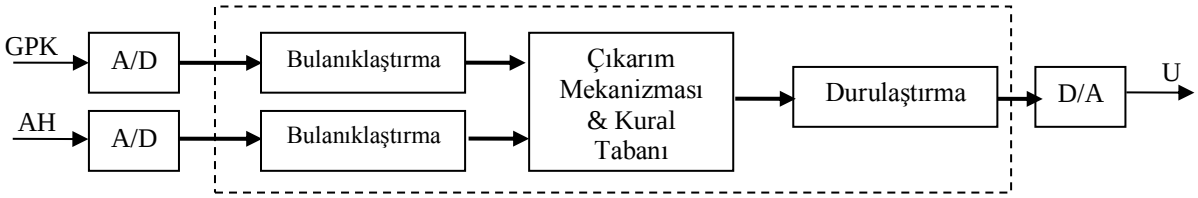
Şebeke suyu ile yapılan deney grafiğinden görüldüğü gibi giren parçacık sensörü ortalama 0,90 mg/l ve çıkan parçacık sensörü ortalama 0,71 mg/l değerlerini ölçmüşlerdir. Deneysel çalışmalar için bu değerlere göre sensörler ofset yapılarak kullanılmıştır.

8.2. GPK ve AH Değişimine Bağlı Bulanık Kontrol

Bu tez çalışmasında kontrolör yapısı geliştirilirken öncelikle literatürde (Sarıtaş, 2008) manyetik filtre kontrolü için geliştirilen bulanık kontrol yapısı kullanılmış ve bu kontrolörün EMF kiti üzerindeki sonuçları alınmıştır. Bu kontrolör temel olarak GPK ve AH'nın değişimlerinden dolayı EMF performansında oluşan etkileri yok etmeyi amaçlamaktadır. Bu bölümde GPK ve AH'ye bağlı olarak manyetik filtre akımını oluşturan bulanık kontrolörün yapısı, programlanması ve yapılan deney sonuçları verilmiştir.

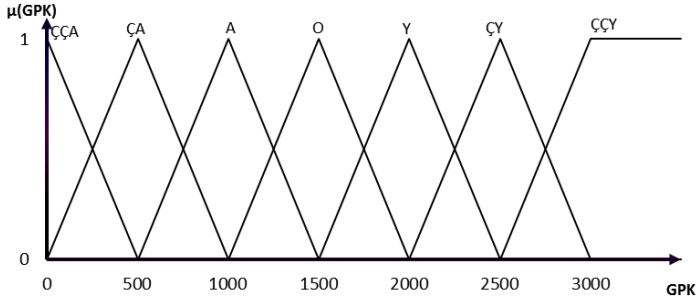
8.2.1. GPK-AH bulanık kontrolörün yapısı

EMF için tasarlanan bulanık mantık kontrolörün giriş parametreleri GPK ve endüstriyel sıvının AH'dır. Çıkış parametresi filtre bobinine uygulanan akımı kontrol eden PWM referans gerilimi (*U*)'dur. Tasarlanan kontrol sisteminin yapısı Şekil 8.3'de verilmiştir.

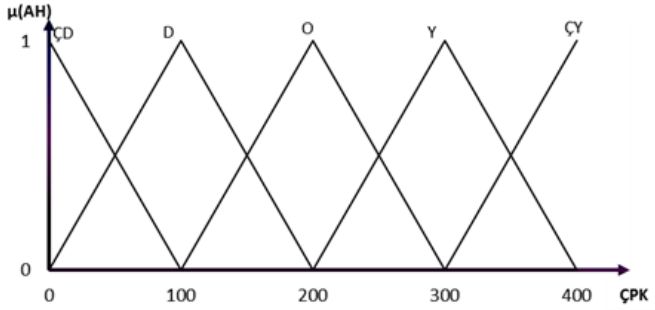


Şekil 8.3. Tasarlanan GPK-AH Bulanık kontrol yapısı

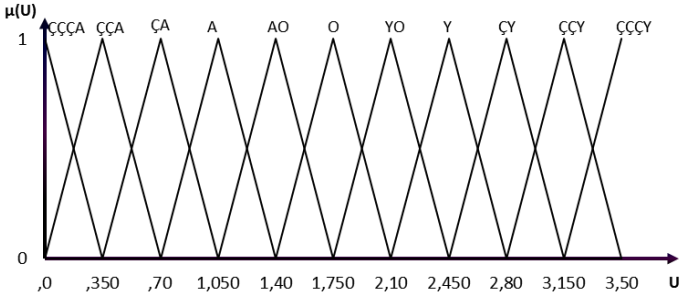
Bulanık mantık kontrolörün ilk bileşeni bulanıklaştırıcı, kesin giriş değerlerini ilgili bulanık kümenin [0,1] aralığındaki üyelik değerlerine dönüştürmektedir. GPK giriş değişkeni {çok çok az (ÇÇA), çok az (ÇA), az (A) , orta (O), yüksek (Y), çok yüksek (ÇY), çok çok yüksek (ÇÇY)} olmak üzere 7 dilsel ifadeye ayrılmıştır. AH giriş değişkeni ise {çok düşük (ÇD), düşük (D), orta (O), yüksek (Y), çok yüksek (ÇY)} olmak üzere 5 dilsel ifadeye ayrılmıştır. U çıkış değişkeni ise {çok çok çok az (ÇÇÇA), çok çok az (ÇÇA), çok az (ÇA), az (A), az orta (AO), orta (O), yüksek orta (YO), yüksek (Y), çok yüksek (ÇY), çok çok yüksek (ÇÇY), çok çok çok yüksek (ÇÇÇY)} olmak üzere 11 dilsel ifadeye ayrılmıştır. Giriş değişkenleri GPK ve AH için üyelik fonksiyonlarının gösterimi Şekil 8.4 ve Şekil 8.5'de, çıkış değişkeni PWM referans gerilimi üyelik fonksiyonlarının gösterimi Şekil 8.6'da verilmiştir.



Şekil 8.4. GPK üyelik fonksiyonları



Şekil 8.5. AH üyelik fonksiyonları



Şekil 8.6. PWM referans gerilimi üyelik fonksiyonları

Giriş ve çıkış değişkenleri için bulanık kural tablosu Çizelge 8.1'de verilmiştir. Bulanık kural tablosu GPK ve AH değişimlerine bağlı olarak PWM referans gerilimini değiştirecek şekilde düzenlenmiştir.

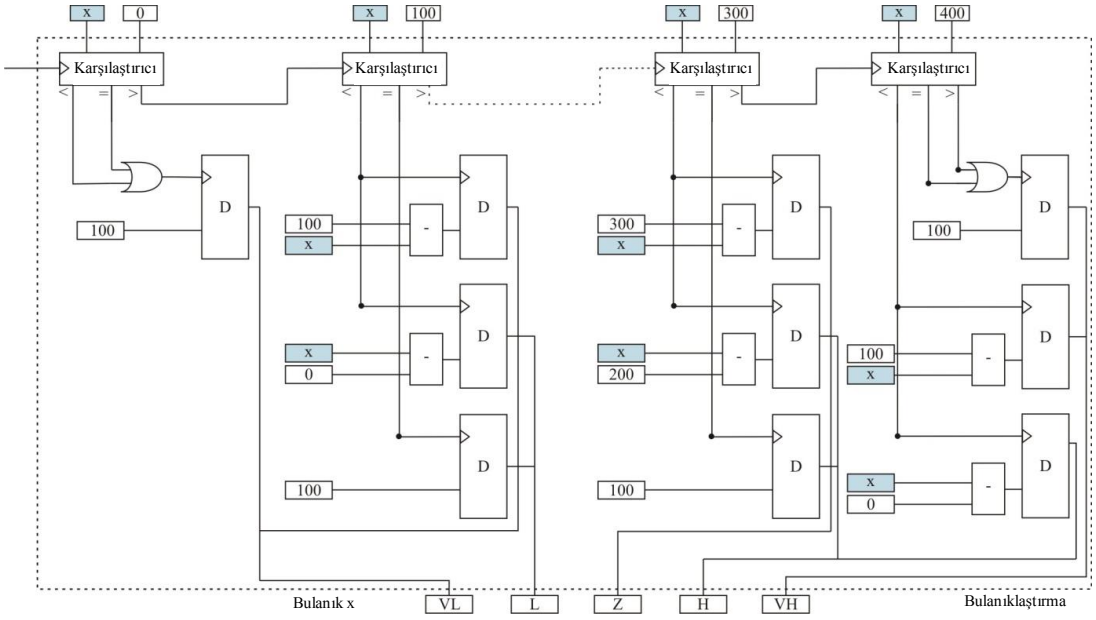
Çizelge 8.1. GPK-AH BK için kural tabanı

		GPK						
		ÇÇA	**ÇA**	**A**	**O**	**Y**	**ÇY**	**ÇÇY**
AH	**ÇD**	ÇÇÇA	ÇÇA	ÇA	A	AO	O	YO
	D	ÇÇA	ÇA	A	AO	O	YO	Y
	O	ÇA	A	AO	O	YO	Y	ÇY
	Y	A	AO	O	YO	Y	ÇY	ÇÇY
	ÇY	AO	O	YO	Y	ÇY	ÇÇY	ÇÇÇY

8.2.2. GPK-AH bulanık kontrolörün FPGA'da programlanması

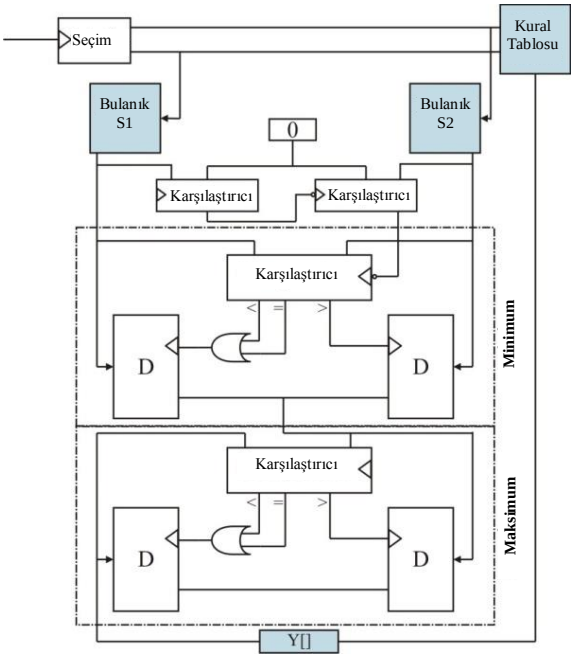
FPGA üzerinde bulanık kontrolör, Şekil 8.3'da verilen yapıda bileşenler halinde tasarlanmıştır. Tasarım VHDL dili kullanılarak Altera Quartus programı ile yapılmıştır. Kontrolörde verilerin gösterimi için ADC karttan gelen veriler 12 bit olması ve gerçekleştirilen kontrolörün MATLAB ile yapılan tamsayı sonuçları ile karşılaştırıldığında EMF kontrolü için uygun hassasiyette sonuç vermesinden dolayı, 12 biti tam sayı kısmı ve 7 biti ondalıklı sayı kısmından oluşan Q12.7 sabit noktalı sayı gösterimi kullanılmıştır.

Bulanıklaştırma bileşeninin mimarisi Şekil 8.7'de verilmiştir. Grafiksel gösterim kolaylığı sağlaması açısından üyelik fonksiyon değerleri [0,100] aralığında alınmıştır. Bu bileşende karşılaştırıcılar aracılığı ile uygun bulanık kümelere ait bulanıklaştırma işlemi gerçekleştirilmektedir.



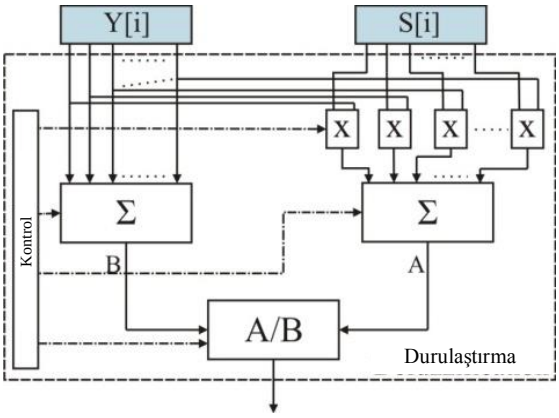
Şekil 8.7. Bulanıklaştırma bileşeninin mimarisi

Çıkarım bileşeninin mimarisi Şekil 8.8'de verilmiştir. Çıkarım mekanizmasında min-max çıkarım metodu kullanılmıştır.



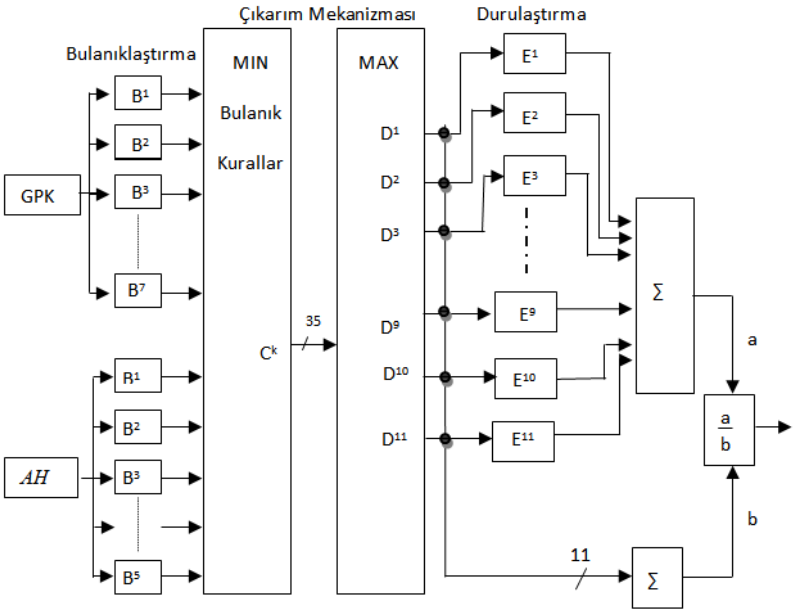
Şekil 8.8. Çıkarım bileşeninin mimarisi

Durulaştırıcı bileşeninin mimarisi Şekil 8.9'de verilmiştir. Bu işlem için ağırlıklı ortalama durulaştırma metodu kullanılmaktadır. Üç kısımdan oluşan bu bölümde A ile bölünen, B ile bölen belirlenerek bölme işlemine tabii tutulmaktadır.



Şekil 8.9. Durulaştırıcı bileşeninin mimarisi

Şekil 8.10'da GPK-AH BK'nın genel blok diyagramı verilmiştir.



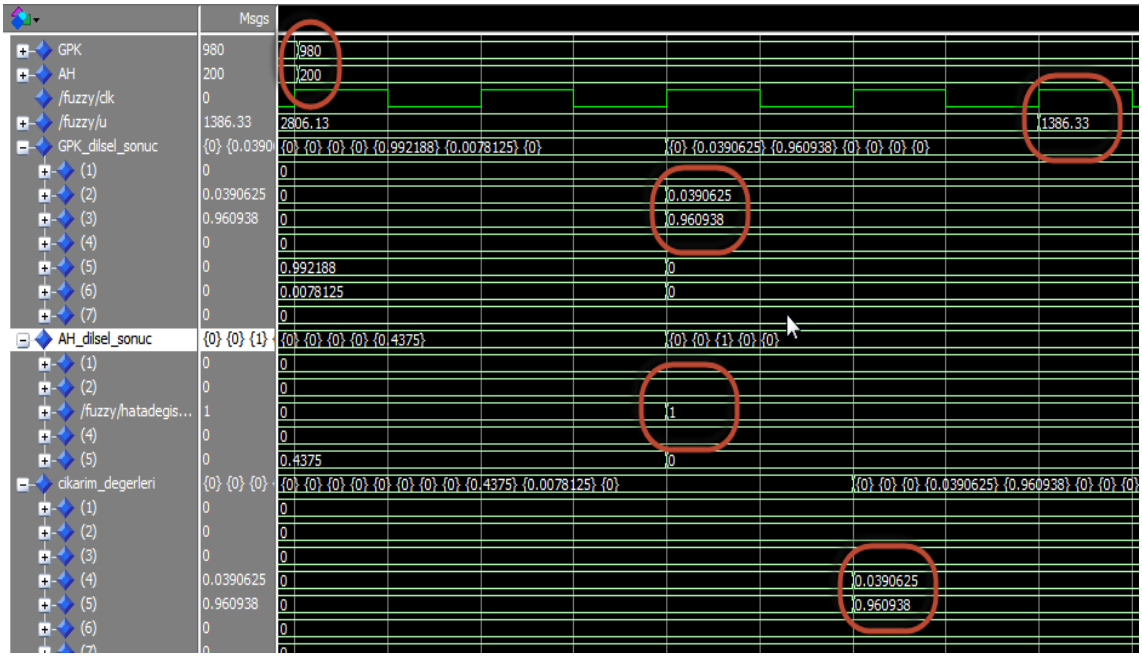
Şekil 8.10. GPK-AH BK'nın genel blok diyagramı

VHDL ile kodlaması yapılan bulanık kontrolörün Modelsim programı ile simülasyon sonuçları alınmıştır. Bulanık kontrolörün lojik sentezleme, yerleştirme ve route işlemleri Altera Quartus programı ile yapılmıştır. Bulanık kontrol yapısının kodlamasında kullanın fonksiyonlar paketi EK-2'de verilmiştir. Cyclone II EP2C70F896 FPGA lojik elementinin %60'ı kullanılarak sentezlenmiştir (Çizelge 8.2).

Çizelge 8.2. Bulanık Kontrol için FPGA kaynak kullanımı

Kaynak	**Kullanım oranı**
Kombinasyonel fonksiyonlar	37355/68416 (%53)
Ayrılmış yazmaçlar	36530/68416 (%19)
Toplam Lojik Element	1859/68416 (%60)
Pin	186/622 (%30)
Çarpım Blokları	75/300 (%20)
Maksimum frekans	7.72 Mhz

Hazırlanan tasarım Modelsim simülasyon programı ile derlenmiştir ve simülasyon sonuçları Şekil 8.11'de verilmiştir. Şekilde GPK, AH, saat, PWM referans gerilimi, aktifleşen GPK dilsel ifade çıkışları, aktifleşen AH dilsel ifade sonuçları ve aktifleşen çıkarım değerleri değişkenleri bulunmaktadır. GPK'nın 980 değeri için Kural 2 ve 3'ün ve AH'nin 200 değeri için Kural 3'ün aktifleştiği görülmektedir. Çıkış değeri ise 1386 olarak oluşmaktadır (Şekil 8.11).



Şekil 8.11. Modelsim programında elde edilen simülasyon sonuçları

GPK-AH bulanık kontrolör bileşeni 5 çevrim süresinde hesaplama yapmaktadır. 7.72 MHz ile çalıştığında toplam işlem zamanı 0,645 µs'dir. Kullanılan EMF kiti üzerinde bulanan ADC kartı 34ms'de bir çevirim yaptığı göz önüne alındığında gerçekleştirilen FPGA tabanlı BK mimarisinin EMF kitini kontrol etmek için yeterli sürede sonuç verdiği görülmektedir.

Giriş, çıkış değişkenleri ve bulanık kural tablosu verilen bulanık kontrol sistemi "Matlab&Fuzzy Logic Toolbox" yazılımı kullanılarak oluşturulmuştur. GPK ve AH'nin

farklı değerleri için Matlab&Fuzzy Logic Toolbox ve Modelsim simülasyonundan elde edilen sonuç değerleri Çizelge 8.3'de karşılaştırmalı olarak verilmiştir. Sonuç değerleri birbirleri ile karşılaştırılarak birbirleri arasındaki hata yüzdesi hesaplanmıştır.

Çizelge 8.3. Bulanık Kontrol için Matlab ve Modelsim sonuçlarının karşılaştırılması

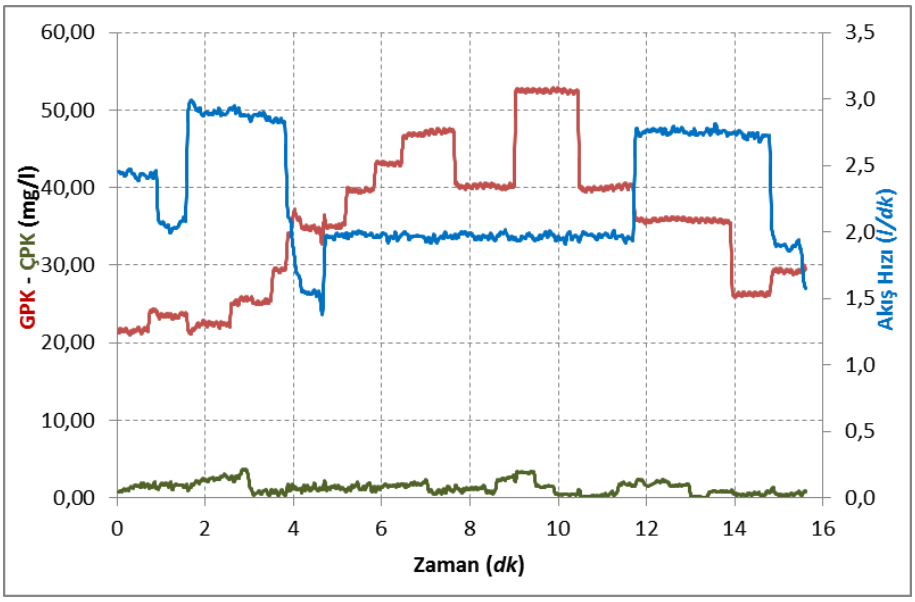
Giren Parçacık Konsantrasyonu	Akış Hızı	Matlab Sonucu	FPGA Sonucu	Hata (%)
400	80	560	553	0.07
700	300	1540	1539	0.01
1200	200	1540	1539	0.01
2500	45	1910	1909	0.01
3000	350	3330	3328	0.02
1000	335	1870	1873	0.03
2020	70	1660	1663	0.03
1800	160	1810	1801	0.09
100	250	954	964	0.1
20	30	119	126	0.07

Matlab&Fuzzy Logic Toolbox ve Modelsim test sonuçları birbirleri ile karşılaştırıldığında %0.044'lük bir hata ortalaması ile modelin başarılı olduğu görülmektedir. Sonuçlar arasındaki fark donanımsal olarak tanımlanan sistemin ondalıklı kısmının hassasiyetinden kaynaklanmaktadır.

8.2.3. GPK-AH bulanık kontrolör deneyi

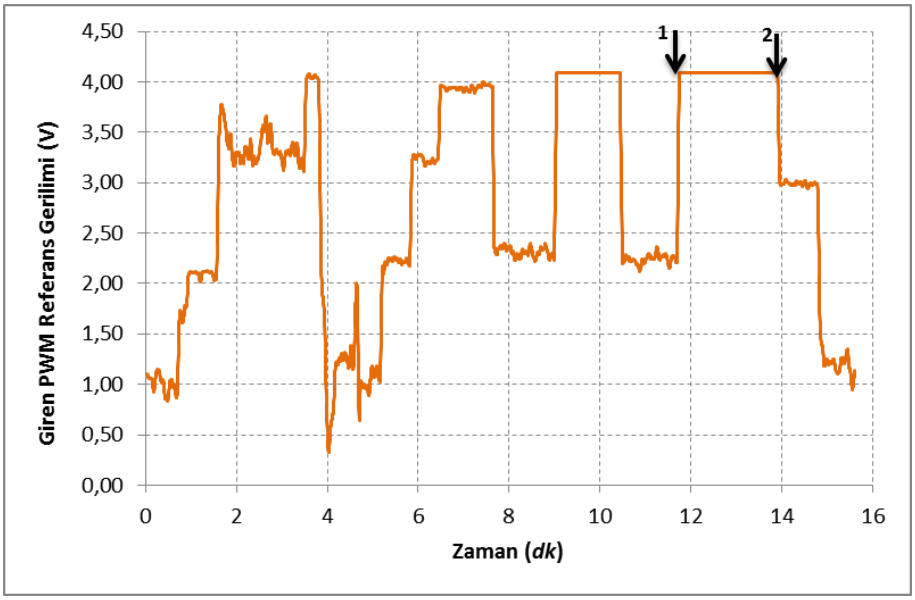
Gerçekleştirilen bulanık kontrolör (BK) ile 14 mm çaplı manyetik özellikli kürecikler ile oluşturulan EMF matrisi ile deneyler yapılmıştır. Deneylerde kullanılan endüstriyel sıvılar bir depo içerisindeki 200 lt suya, 4-10x10^{-4} mm büyüklüğündeki demir tozu ilave edilerek hazırlanmıştır. Deney, EMF'nin Bölge II çalışma şartlarında gerçekleştirilmiştir. Deney süresince AH pompa aracılığı ile, GPK demir tozu ilave edilerek değiştirilmiştir. İstenilen performans değeri %95 olarak belirlenmiştir. Gerçekleştirilen kontrolörün değişen durumlarına göre davranışlarının tutarlılığını görebilmek için bu deney iki tekerrürlü olarak yapılmıştır. İlk tekerrürde deney süresi 16 dk ikinci tekerrürde 15,5 dk sürmüştür.

İlk tekerrür deney süresince GPK, ÇPK ve AH değişimleri Şekil 8.12'de görüldüğü gibi elde edilmiştir.



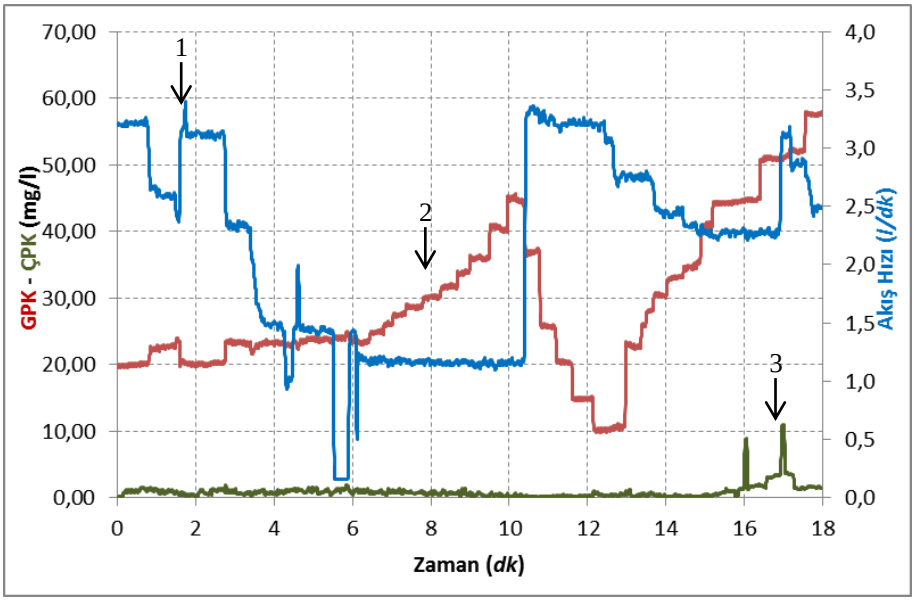
Şekil 8.12. GPK-AH BK ile çalışan filtrenin GPK, ÇPK ve AH değişimi

Deney süresince GPK-AH BK ile elde edilen PWM referans geriliminin değişimi Şekil 8.13'de görüldüğü gibi elde edilmiştir. Şekilden de açıkça görüldüğü gibi PWM referans gerilimi GPK ve AH'ya bağlı olarak değişmektedir. Örneğin 1 nolu noktada AH değerinin yükselmesine karşılık PWM referans gerilimi artmış ve devam eden süreçte AH değeri değişmemesine rağmen 2 nolu noktada GPK'deki düşmeye bağlı olarak PWM referans gerilimi düşmüştür.



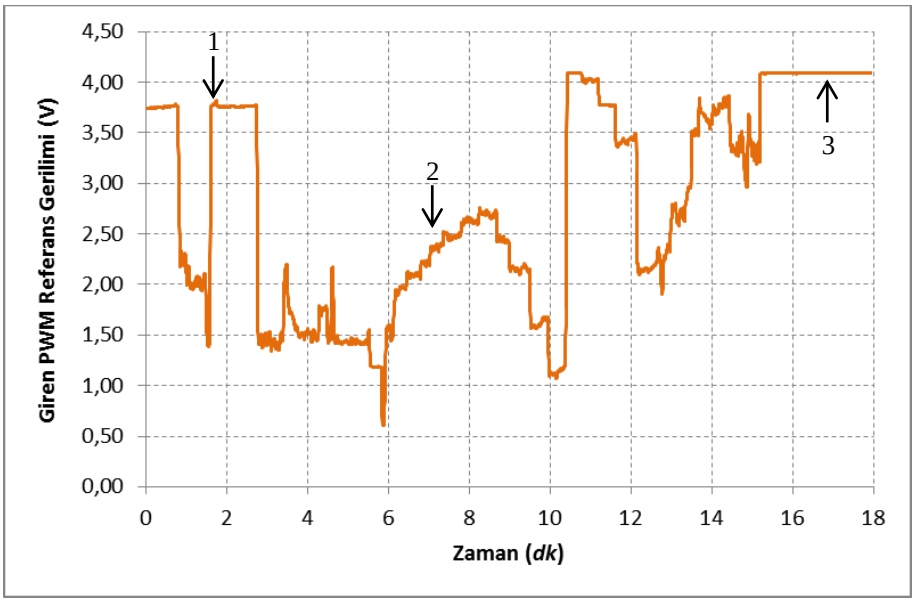
Şekil 8.13. GPK-AH BK ile çalışan filtrenin PWM Referans gerilimi değişimi

İkinci tekerrür deneyi 18 dk sürmüştür ve deney süresince GPK, ÇPK ve AH değişimleri Şekil 8.14'de görüldüğü gibi elde edilmiştir.



Şekil 8.14. GPK-AH BK ile çalışan filtrenin GPK, ÇPK ve AH değişimi

İkinci tekerrür deney süresince GPK-AH BK ile elde edilen PWM referans geriliminin değişimi Şekil 8.15'de görüldüğü gibi elde edilmiştir.



Şekil 8.15. GPK-AH BK ile çalışan filtrenin PWM referans gerilimi değişimi

Şekil 8.14 ve Şekil 8.15 karşılıklı olarak incelendiğinde 1 nolu noktada AH'nın değişimine karşılık PWM referans geriliminin arttığı ve 2 nolu bölümdeki AH sabit

oluduğu ve GPK'nın arttığı durumda PWM referans geriliminin arttığı görülürken, 3 nolu bölümde ÇPK'daki yükselmeye tepki vermediği görülmektedir.

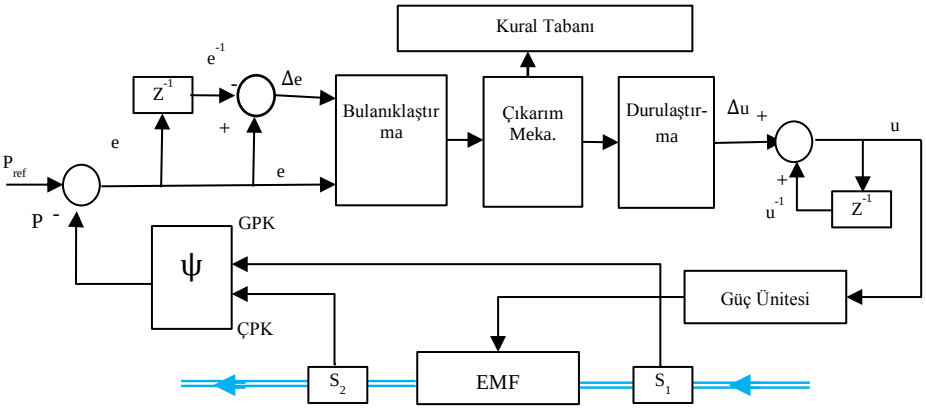
İki tekerrürde de görüldüğü gibi GPK ve AH'nın farklı durumlarına bağlı olarak bulanık kontrolör tutarlı olarak davranmaktadır. Daha sonraki kontrolör deneyleri tek tekerrürlü olarak gerçekleştirilmiştir.

8.3. PI tip Bulanık Kontrolör

Bu bölümdeki çalışmada, EMF performansını etkileyen tüm faktörleri dikkate alabilen bir PI tip Bulanık Kontrolör (PI tip BK) tasarımı yapılmış ve EMF kitinde bulanan FPGA üzerinde programlanmıştır. Bu kontrolör temel olarak EMF performansının istenilen (referans) performansından farkı olan performans hatasına bağlı olarak manyetik filtre akımını oluşturan PWM referans gerilimini değiştirilmesinden oluşmaktadır. Bu bölümde PI tip bulanık kontrolörün yapısı, programlanması ve yapılan deney sonuçları verilmiştir.

8.3.1. PI tip bulanık kontrolörün yapısı

EMF'ye uygulanmak üzere tasarlanan PI tip bir bulanık kontrolörün yapısı Şekil 8.16'da verilmiştir.



Şekil 8.16. PI tip kontrol sisteminin blok diyagramı

Şekil 8.16'de görüldüğü gibi EMF performans hatası ve hata değişimi bulanık kontrolörün giriş değişkenleridir. Hata ve hata değişimi ifadeleri sırasıyla Denklem 8.1 ve 8.2'de verilmiştir.

$$e(k)=P_{ref}(k)-P(k) \quad (8.1)$$

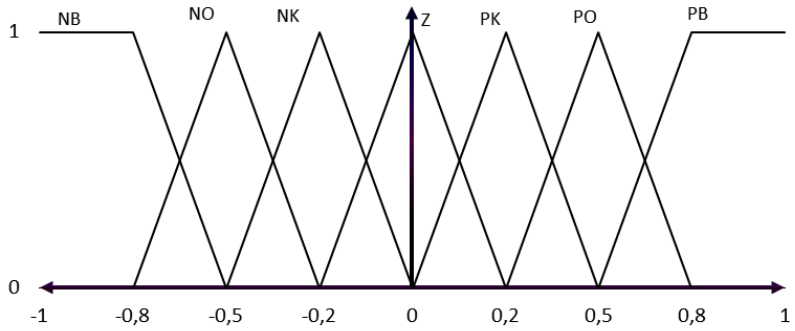
$$\Delta e(k)=e(k)-e(k-1) \quad (8.2)$$

Burada k örnekleme zamanı, P_{ref} referans performans değeri, P EMF'nin performans değeridir. Kontrolörün çıkışı ise Denklem 8.3 ile ifade edildiği gibidir.

$$u(k)=u(k-1)+\Delta u(k) \quad (8.3)$$

Burada $\Delta u(k)$ PI-tip bulanık kontrolörün çıkışı olup kontrol değerinin değişimini ifade etmektedir.

Giriş ve çıkış değişkenlerinin bulanıklaştırılmasında kullanılan üyelik fonksiyonları Şekil 8.17'da verilmiştir. Kullanılan üyelik fonksiyonları {negatif büyük (NB), negatif orta (NO), negatif küçük (NK), sıfır (S), pozitif küçük (PK), pozitif orta (PO), pozitif büyük (PB)} dilsel ifadeleri ile tanımlanmıştır.



Şekil 8.17. e, de, Δu için üyelik fonksiyonları.

Tüm üyelik fonksiyon değerleri [-1,1] aralığında tanımlanmış ve giriş ve çıkış değişkenlerinde ölçeklendirme katsayıları kullanılmıştır. Ölçeklendirme katsayılarının uygun değerlerinin seçimi EMF kontrolü hakkındaki uzman bilgi ve deneme-yanılma metodu kullanılarak oluşturulmuştur. PI-tip bulanık kontrolör için kullanılan bulanık kural tabanı Çizelge 8.4'de verilmiştir.

Çizelge 8.4. PI tip BK için bulanık kural tabanı

de \ e	NB	NO	NK	S	PK	PO	PB
NB	NB	NB	NB	NO	NK	NK	S
NO	NB	NO	NO	NO	NK	S	PK
NK	NB	NO	NK	NK	S	PK	PO
S	NB	NO	NK	S	PK	PO	PB
PK	NO	NK	S	PK	PK	PO	PB
PO	NK	S	PK	PO	PO	PO	PB
PB	S	PK	PK	PO	PB	PB	PB

8.3.2. PI tip bulanık kontrolörün FPGA'da programlanması

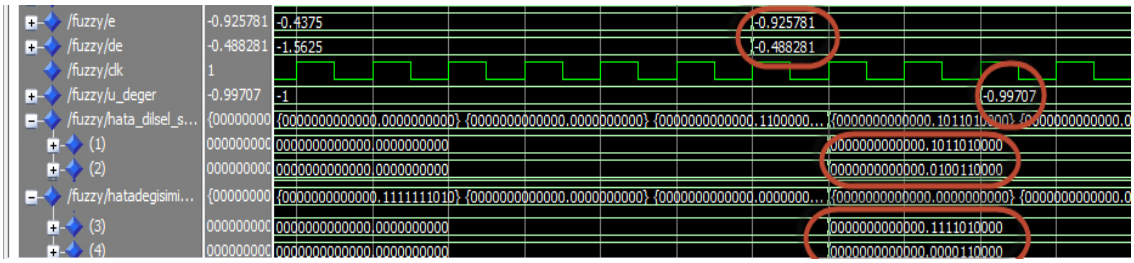
PI tip BK'nın FPGA üzerinde programlanması işlemi Bölüm 8.2.2'de verilen FPGA tabanlı bulanık kontrolör mimarisi kullanılarak yapılmıştır. Bu mimari PI tip BK giriş, çıkış, üyelik fonksiyonları ve kural tabanına göre tekrar düzenlenmiştir.

PI tip bulanık kontrolörün lojik sentezleme, yerleştirme ve route işlemleri Altera Quartus programı ile yapılmıştır. Cyclone II EP2C70F896 FPGA lojik elementin %35'i kullanılarak sentezlenmiştir (Çizelge 8.5).

Çizelge 8.5. PI tip BK için FPGA kaynak kullanımı

Kaynak	**Kullanım oranı**
Kombinasyonel fonksiyonlar	24021/68416 (%35)
Ayrılmış yazmaçlar	954/68416 (%1)
Toplam Lojik Element	24109/68416 (%35)
Pin	186/622 (%30)
Çarpım Blokları	32/300 (%10)
Maksimum frekans	7.1 Mhz

PI tip bulanık kontrolör bileşeni 5 çevrim süresinde hesaplama yapmaktadır. 7.1 MHz ile çalıştığında toplam işlem zamanı 0,7 µs'dir. Hazırlanan tasarım Modelsim simülasyon programı ile derlenmiştir ve simülasyon sonuçları Şekil 8.18'de verilmiştir. Performans hatasının -0,925781 değeri için Kural 1 ve 2'nin ve performans hata değişiminin -0,488281 değeri için Kural 3 ve 4'ün aktifleştiği görülmektedir. Çıkış değeri ise -0,99707 olarak oluşmaktadır (Şekil 8.18).

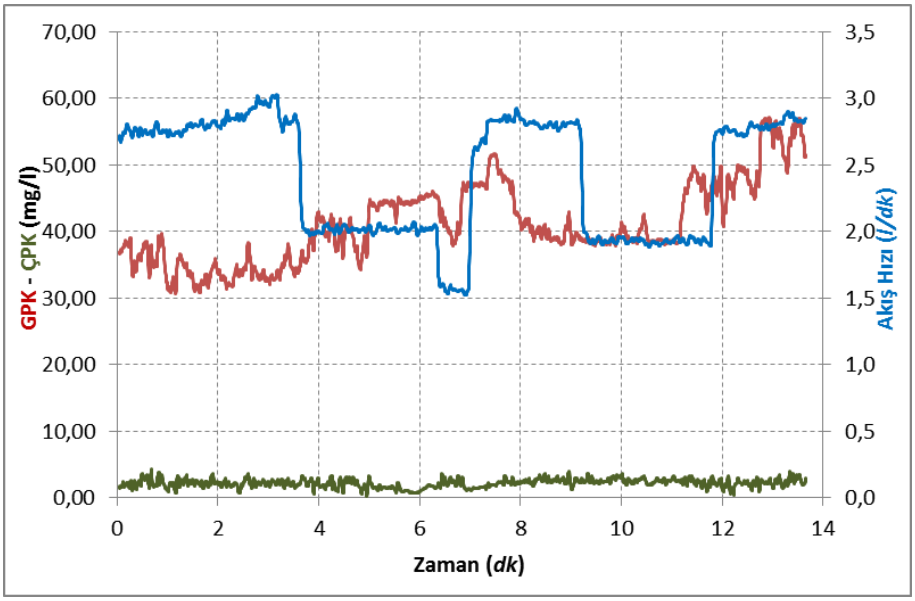


Şekil 8.18. PI-tip BK simülasyon sonuçları

8.3.3. PI tip bulanık kontrolör deneyi

Gerçekleştirilen PI tip bulanık kontrolör ile 14 mm çaplı manyetik özellikli kürecikler ile oluşturulan EMF matrisi ile deneyler yapılmıştır. Deneylerde kullanılan endüstriyel sıvılar bir depo içerisindeki 200 lt suya, 4-10x10^{-4} mm büyüklüğündeki demir tozu ilave edilerek hazırlanmıştır. İstenilen performans değeri %95 olarak belirlenmiş ve deney 13,5 dk sürmüştür.

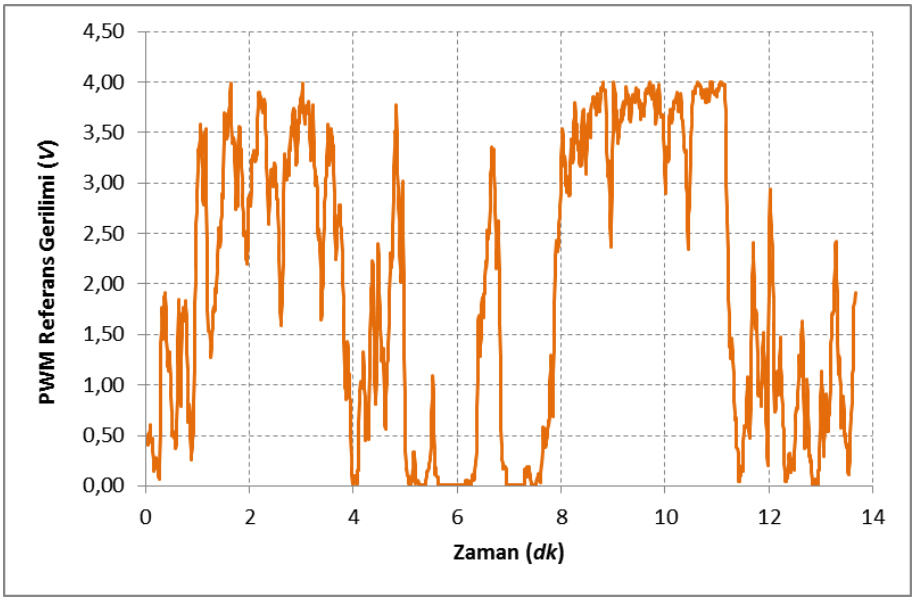
Deney süresince GPK, ÇPK ve AH değişimleri Şekil 8.19'de görüldüğü gibi elde edilmiştir.



Şekil 8.19. PI tip BK ile çalışan filtrenin GPK, ÇPK ve AH değişimi

Deney süresince PI tip BK ile elde edilen PWM referans geriliminin değişimi Şekil 8.20'de görüldüğü gibi elde edilmiştir. Şekilden de görüldüğü gibi PWM referans gerilimi ÇPK'nın arttığı yani EMF performansının düştüğü durumlarda yükselmiştir.

AH'de meydana gelen değişimler EMF performansını etkilemediği sürece direkt olarak PWM referans gerilimi değişimine etkisi yoktur.



Şekil 8.20. PI tip BK ile çalışan filtrenin PWM Referans gerilimi değişimi

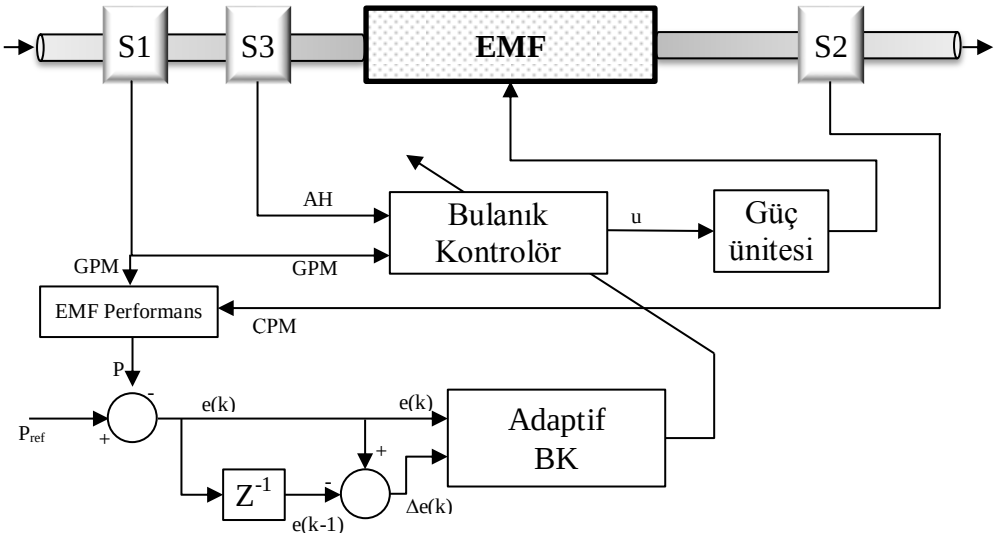
8.4. Kendini Organize Edebilen Bulanık Kontrolör

Kendini organize edebilen bulanık kontrolörde (KOEBK) iki katmandan oluşan bir BK yapısı kullanılmıştır. İlk katman BK, GPK ve AH değişimine bağlı olarak PWM referans gerilimini kontrol ederken, ikinci katman BK, ilk katman BK'nın çıkış üyelik fonksiyonunu düzenlemektedir. Bu bölümde KOEBK'nın yapısı, programlanması ve yapılan deney sonuçları verilmiştir.

8.4.1. Kendini organize edebilen bulanık kontrolörün yapısı

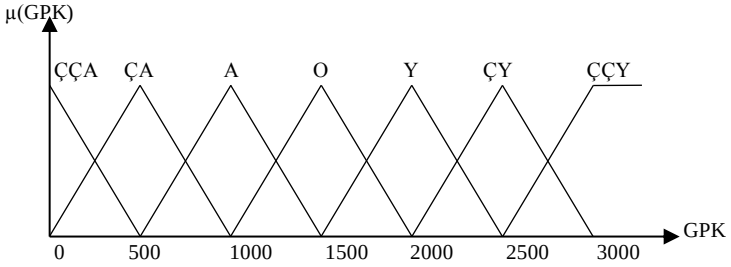
KOEBK'in yapısı GPK-AH BK yapısına ek olarak bir öğrenme algoritmasına sahiptir. Sistemin dinamikleri konusunda önceden bir bilgisi olmaksızın, adaptasyon sinyali ile ilk katman çıkış üyelik fonksiyonu parametreleri online olarak değiştirmektedir.

Şekil 8.21'de iki katmanlı KOEBK kontrolör yapısı verilmiştir. İlk katmandaki bulanık kontrolör GPK ve AH olmak üzere iki giriş, manyetik akı değişimini sağlayan PWM referans gerilimi olmak üzere bir çıkışa sahiptir.

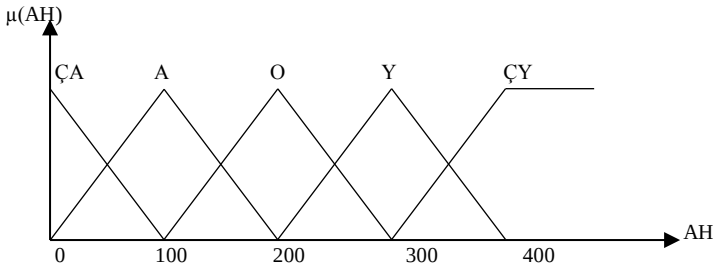


Şekil 8.21. KOEBK sisteminin blok diyagramı.

İlk katmanda Mamdani bulanıklaştırıcı ile giriş değerleri Şekil 8.22 ve Şekil 8.23'de verilen üçgensel üyelik fonksiyonları ile dilsel ifadelere dönüştürülmektedir. Şekillerde üçgensel üyelik fonksiyonları {çok çok az (ÇÇA),çok az (ÇA), az (A), orta (O), yüksek (Y), çok yüksek (ÇY), çok çok yüksek (ÇÇY)} dilsel ifadeleri ile tanımlanmıştır.



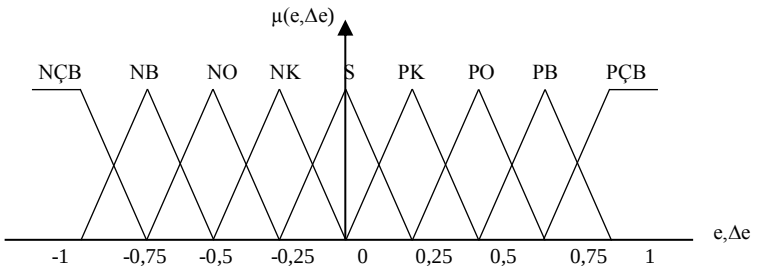
Şekil 8.22. GPK üyelik fonksiyonu.



Şekil 8.23. Akış hızı üyelik fonksiyonu

EMF'nin performansı (P), GPK ve ÇPK'ya bağlı olarak Hesaplama Modülünde Denklem 7.3'e göre hesaplanmaktadır. İkinci katmanda bulunan KOEBK ise performansın istenilen performanstan farkı olan e(k) (Denklem 8.1) ve hatanın değişimi Δe(k) (Denklem 8.2) girişlerine sahiptir.

İkinci katmanda Mamdani tip bulanıklaştırıcı ile giriş değerleri Şekil 8.24'de verilen üçgensel üyelik fonksiyonları ile dilsel ifadelere dönüştürülmektedir. Şekilde üçgensel üyelik fonksiyonları {negatif çok büyük (NÇB), negatif büyük (NB), negatif orta (NO), negatif küçük (NK), sıfır (S), pozitif küçük (PK), pozitif orta (PO), pozitif büyük (PB), pozitif çok büyük (PÇB)} dilsel ifadeleri ile belirlenmiştir.



Şekil 8.24. Hata ve hata değişimi üyelik fonksiyonları

Adaptif katman tarafından online ayarlanan birinci katmandaki çıkış üyelik fonksiyonlarının merkezi Çizelge 8.6'de verilmiştir. KOEBK üyelik fonksiyonlarının merkezi sabittir ve Çizelge 8.7'da verilmiştir.

Çizelge 8.6. Birinci katman için bulanık kural tablosu

		Giren parçacık miktarı						
		0	500	1000	1500	2000	2500	3000
Akış hızı	0	100	350	700	1050	1400	1750	2100
	60	350	700	1050	1400	1750	2100	2450
	120	700	1050	1400	1750	2100	2450	2800
	180	1050	1400	1750	2100	2450	2800	3150
	240	1400	1750	2100	2450	2800	3150	3500

Çizelge 8.7. İkinci katman için bulanık kural tablosu

		Hata değişimi								
		-1	-0,75	-0,5	-0,25	0	0,25	0,5	0,75	1
Hata	-1	1	1	1	1	0,75	0	0	0	0
	-0,75	1	1	1	0,75	0,5	0	0	0	0
	-0,5	1	1	0,75	0, 5	0,25	0	0	0	0
	-0,25	1	0,75	0, 5	0,25	0	0	0	0	0
	0	0,75	0,5	0,25	0	0	0	-0,25	-0,5	-0,75
	0,25	0	0	0	0	0	-0,25	-0,5	-0,75	-1
	0,5	0	0	0	0	-0,25	-0,5	-0,75	-1	-1
	0,75	0	0	0	0	-0,5	-0,75	-1	-1	-1
	1	0	0	0	0	-0,75	-1	-1	-1	-1

8.4.2. Kendini organize edebilen bulanık kontrolörün FPGA'da programlanması

KOEBK yapısında ilk katman ve ikinci katman BK olmak üzere iki adet BK bulunmaktadır. Bu BK'lar Bölüm 8.2.2'de verilen FPGA tabanlı bulanık kontrolör mimarisi kullanılarak gerçekleştirilmiştir. Bu mimari KOEBK giriş, çıkış, üyelik fonksiyonları ve kural tabanına göre tekrar düzenlenmiştir.

KOEBK'nın lojik sentezleme, yerleştirme ve route işlemleri Altera Quartus programı ile yapılmıştır. Cyclone II EP2C70F896 FPGA lojik elementinin %98'i kullanılarak sentezlenmiştir (Çizelge 8.8).

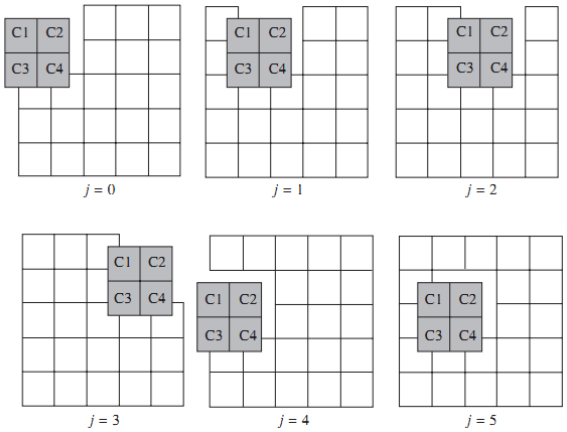
Çizelge 8.8. KOEBK için FPGA kaynak kullanımı

Kaynak	**Kullanım oranı**
Kombinasyonel fonksiyonlar	66176/68416 (%97)
Ayrılmış yazmaçlar	2577/68416 (%4)
Toplam Lojik Element	66786/68416 (%98)
Pin	186/622 (%30)
Çarpım Blokları	233/300 (%78)
Maksimum frekans	7.11 Mhz

KOEBK için FPGA kaynak kullanımı incelendiğinde toplam lojik elementlerin %98'nin kullanıldığı görülmektedir. Sistem kaynaklarının kullanımını azaltmak amacıyla kullanılan BK yapısında optimizasyon işlemi uygulanmıştır.

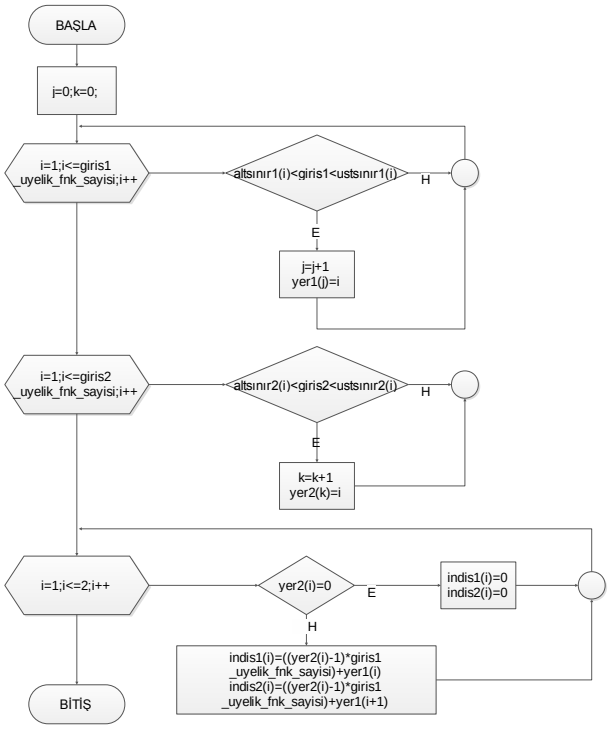
Çizelge 8.7'da görüldüğü gibi bulanık kural tablosunda 81 adet bulanık kural çıkışı bulunmaktadır ve çıkarım mekanizmasının her bir işlemi için 81 adet kuralın hepsi hesaplamaya tabii tutulmaktadır.

BK'nın optimizasyonu için hesaplama esnasında sadece ilgili kurallara odaklanılmış ve diğerleri göz ardı edilmiştir. Şekil 8.25'de görüldüğü gibi BK'nın giriş değerlerine göre maksimum 4 adet kural çıkışı aktif olmaktadır.



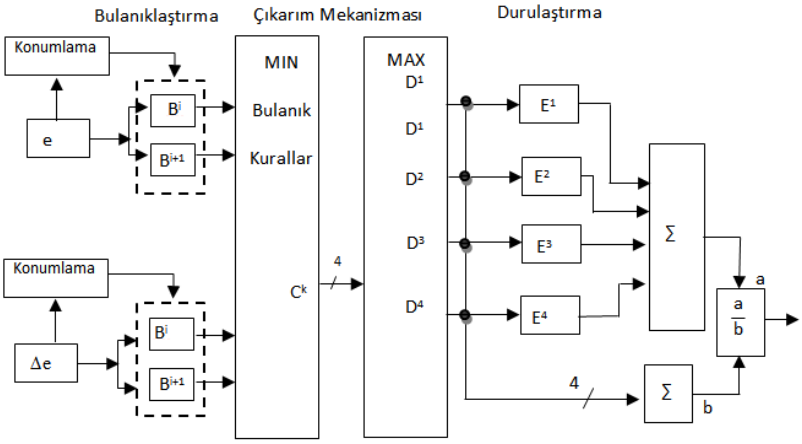
Şekil 8.25. BK optimizasyonu için mini kural tablosu

Gerçekleştirilen optimizasyon işleminde girişlerin öncelikle hangi bölgede olduğu tespit edilerek o bölge üzerinde bulanıklaştırma, çıkarım mekanizması ve durulaştırma işlemi uygulanmıştır. Konumlama işlemi için akış diyagramı Şekil 8.26'da verilmiştir.



Şekil 8.26. Konumlama fonksiyonu akış diyagramı

Konumlama işleminde bulunduğu bölge tespit edildikten sonra bulanık kural tablosundaki yeri indis1 ve indis2 dizi değişkelerinde saklanmaktadır. Optimizasyon işlemi sonrası elde edilen BK genel yapısı Şekil 8.27'de verilmiştir.



Şekil 8.27. Optimize edilen BK genel yapısı

Şekil 8.27'de görüldüğü gibi bulanıklaştırma, durulaştırma ve çıkarım işlemleri sadece aktif bulunan üyelik fonksiyonları üzerinden gerçekleştirilmektedir. KOEBK'nın optimizasyonu sonrasında Cyclone II EP2C70F896 FPGA lojik elementinin %50'si kullanılarak sentezlenmiştir (Çizelge 8.9).

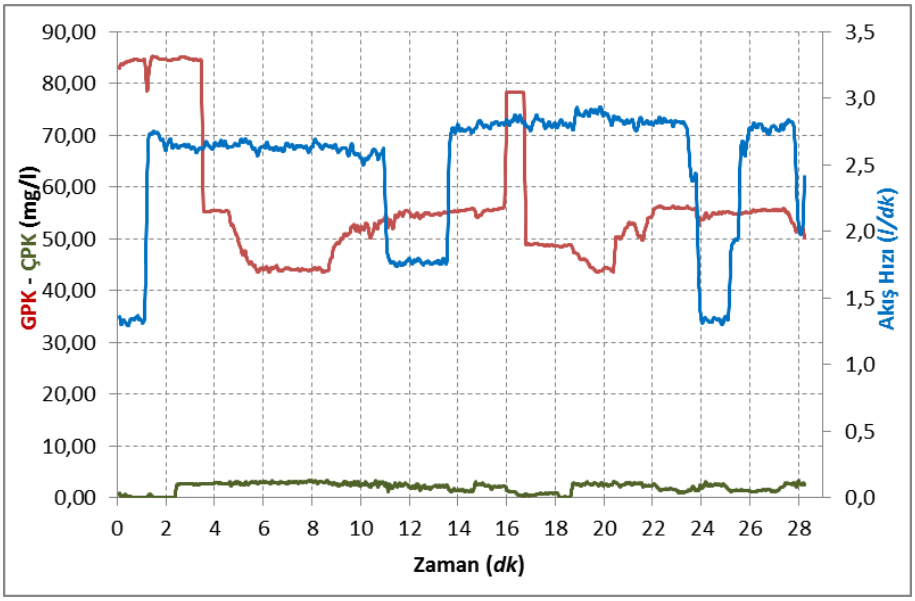
Çizelge 8.9. Optimize edilen KOEBK için FPGA kaynak kullanımı

Kaynak	**Kullanım oranı**
Kombinasyonel fonksiyonlar	31961/68416 (%46)
Ayrılmış yazmaçlar	2823/68416 (%4)
Toplam Lojik Element	34784/68416 (%50)
Pin	186/622 (%30)
Çarpım Blokları	124/300 (%41)
Maksimum frekans	5.32 Mhz

8.4.3. Kendini organize edebilen bulanık kontrolör deneyi

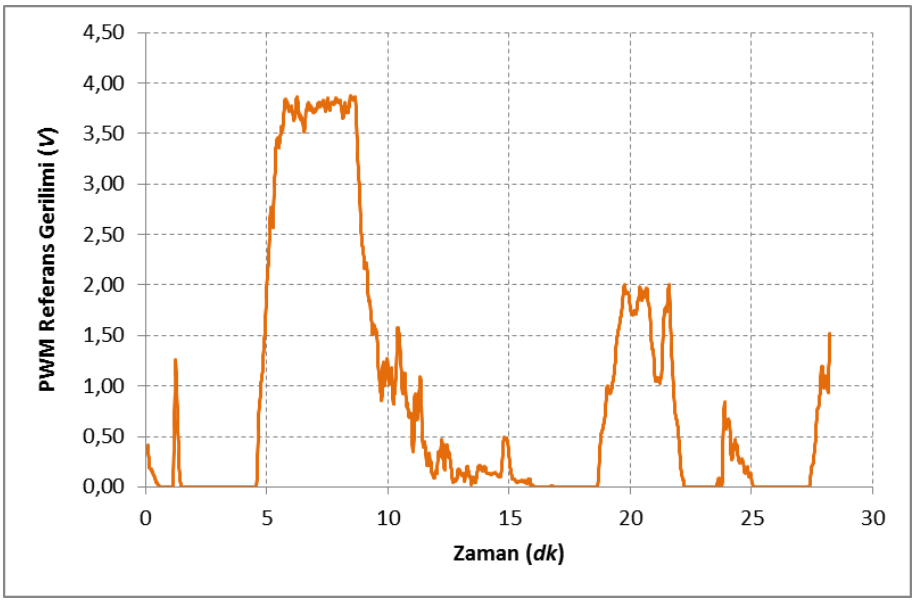
Gerçekleştirilen KOEBK ile 14 mm çaplı manyetik özellikli kürecikler ile oluşturulan EMF matrisi ile deneyler yapılmıştır. Deneylerde kullanılan endüstriyel sıvılar bir depo içerisindeki 200 lt suya, 4-10x10^{-4} mm büyüklüğündeki demir tozu ilave edilerek hazırlanmıştır. İstenilen performans değeri %95 olarak belirlenmiş ve deney 28 dk sürmüştür.

Deney süresince GPK, ÇPK ve AH değişimleri Şekil 8.28'de görüldüğü gibi elde edilmiştir.



Şekil 8.28. KOEBK ile çalışan filtrenin GPK, ÇPK ve AH değişimi

Deney süresince KOEBK ile elde edilen PWM referans geriliminin değişimi Şekil 8.29'de görüldüğü gibi elde edilmiştir. Şekil 8.27 ve Şekil 8.28 beraber incelendiğinde PWM referans gerilimi ÇPK'nın arttığı yani EMF performansının düştüğü durumlarda ve aynı zamanda GPK ve AH değişimlerinde de tepki vermektedir.



Şekil 8.29. KOEBK ile çalışan filtrenin PWM Referans gerilimi değişimi

8.5. Kendini Ayarlayabilen PI tip Bulanık Kontrolör

Bu kontrolör çalışmasında daha önce yapısı verilen PI tip BK'nın EMF performansı değişimine tepkisini geliştirmek amacıyla kendini ayarlayabilen PI tip BK (KABK) tasarlanarak gerçekleştirilmiştir. Bu bölümde KABK'nın yapısı, programlanması ve yapılan deney sonuçları verilmiştir.

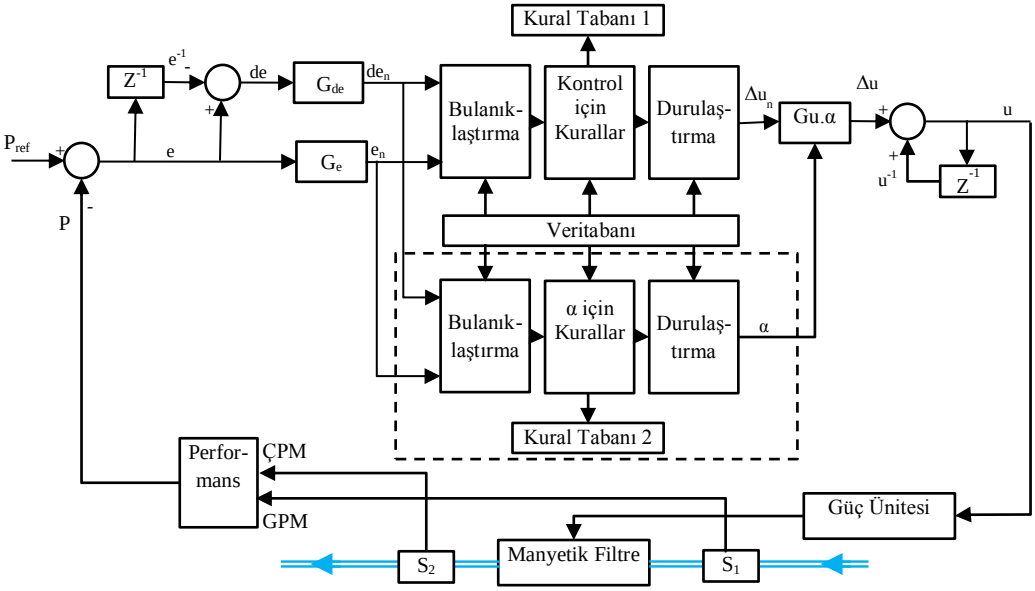
8.5.1. Kendini ayarlayabilen PI tip bulanık kontrolörün yapısı

EMF kontrolü için PI-tip BK ile EMF performansındaki değişim (Denklem 8.1) ve bu değişimin türevi (Denklem 8.1) giriş değişkenleri olarak kullanılırken, kontrol çıkışındaki değişim miktarı Δu(k) çıkış olarak kullanılmaktadır.

Giriş ölçeklendirme katsayılarının (*Ge*, *Gde* ve *Gu*) uygun değerlerinin seçimi EMF kontrolü hakkındaki uzman bilgi ve deneme-yanılma metodu kullanılarak oluşturulmuştur. KABK'un tasarımında aynı kazanç ilkesi kullanılmıştır. Bu çalışmada giriş ölçeklendirme katsayıları sabit tutulurken çıkış ölçeklendirme katsayısı EMF performansındaki değişime bağlı olarak kazanç güncelleme katsayısı (α) ile sürekli olarak ayarlanmaktadır. KABK'nın çıkış ölçeklendirme katsayısı Denklem 8.4'de belirtildiği gibi her örnekleme periyodunda α gibi bir kazanç güncelleme katsayısının çarpımı ile elde edilmektedir.

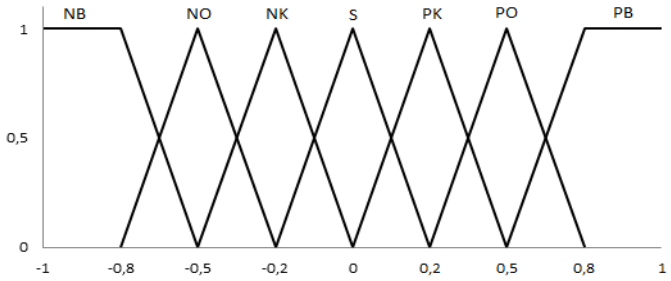
$$\Delta u = (\alpha . G_u) . du_n \tag{8.4}$$

KABK'nın genel yapısı çıkış ölçeklendirme katsayısını ayarlayan kazanç ayarlama sistemi kesikli çizgiler belirtilerek Şekil 8.30'de verilmiştir.

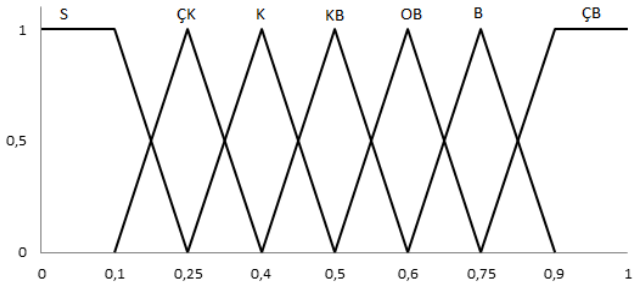


Şekil 8.30. KABK genel yapısı

Kontrol girişlerinin tüm üyelik fonksiyonları (e, de) ve kontrol çıkışındaki değişim miktarı (Δu) [-1,1] normalize aralığında {negatif büyük (NB), negatif orta (NO), negatif küçük (NK), sıfır (S), pozitif küçük (PK), pozitif orta (PO), pozitif büyük (PB)} dilsel ifadeleri ile tanımlanmıştır (Şekil 8.31). Kazanç güncelleme katsayısını (α) hesaplamak için kullanılan üyelik fonksiyonları [0,1] normalize aralığında {sıfır (S), çok küçük (ÇK), küçük (K), küçük büyük (KB), orta büyük (OB), büyük (B), çok büyük ÇB) ile tanımlanmıştır (Şekil 8.32).



Şekil 8.31. e, de, Δu için üyelik fonksiyonları.



Şekil 8.32. α için üyelik fonksiyonları.

Δu değerini hesaplamakta kullanılan kural tabanı Çizelge 8.10'de verilmiştir. α değeri ise Çizelge 8.11'de verilen kural tablosu ile hesaplanmaktadır.

Çizelge 8.10. Δ*u* için kural tabanı

de \ *e*	**NB**	**NO**	**NK**	**S**	**PK**	**PO**	**PB**
NB	NB	NB	NB	NO	NK	NK	S
NO	NB	NO	NO	NO	NK	S	PK
NK	NB	NO	NK	NK	S	PK	PO
S	NB	NO	NK	S	PK	PO	PB
PK	NO	NK	S	PK	PK	PO	PB
PO	NK	S	PK	PO	PO	PO	PB
PB	S	PK	PK	PO	PB	PB	PB

Çizelge 8.11. *α* için kural tabanı

de \ *e*	**NB**	**NO**	**NK**	**S**	**PK**	**PO**	**PB**
NB	ÇB	ÇB	ÇB	B	KB	K	S
NO	ÇB	ÇB	B	B	OB	K	ÇK
NK	ÇB	OB	B	ÇB	ÇK	K	ÇK
S	K	KB	OB	S	OB	KB	K
PK	ÇK	K	ÇK	ÇB	B	OB	ÇB
PO	ÇK	K	OB	B	B	ÇB	ÇB
PB	S	K	KB	B	ÇB	ÇB	ÇB

8.5.2. Kendini ayarlayabilen PI tip bulanık kontrolörün programlanması

KABK yapısında bulunan BK'lar Bölüm 8.2.2'de verilen FPGA tabanlı bulanık kontrolör mimarisi kullanılarak gerçekleştirilmiştir. Bu mimari üzerinde KABK giriş, çıkış, üyelik fonksiyonları ve kural tabanına göre tekrar düzenlenme yapılmıştır.

KABK'nın lojik sentezleme, yerleştirme ve route işlemleri Altera Quartus programı ile yapılmıştır. Cyclone II EP2C70F896 FPGA üzerinde 39391 lojik element kullanılarak sentezlenmiştir (Çizelge 8.12).

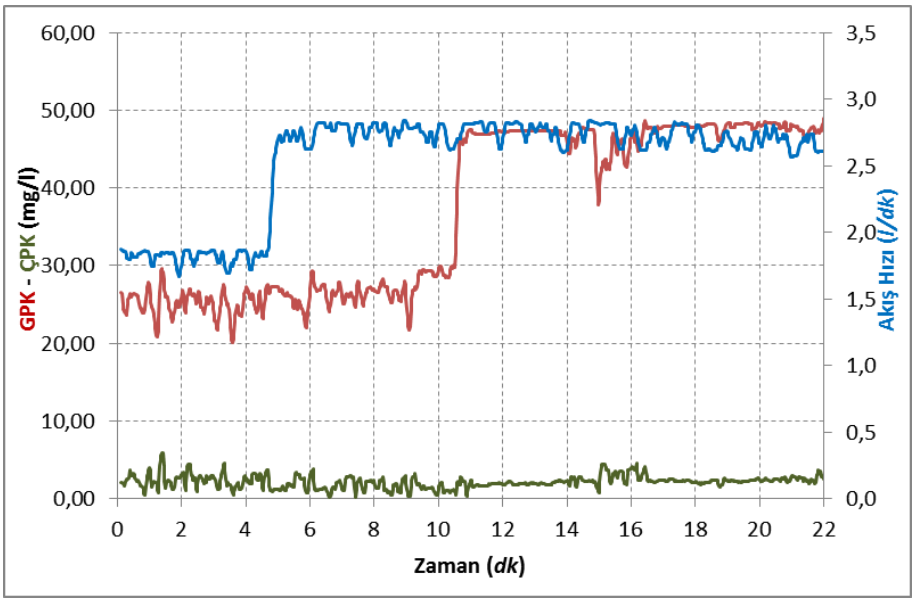
Çizelge 8.12. KABK için FPGA kaynak kullanımı

Kaynak	**Kullanım oranı**
Kombinasyonel fonksiyonlar	39286/68416 (%57)
Ayrılmış yazmaçlar	1281/68416 (%2)
Toplam Lojik Element	39391/68416 (%58)
Pin	186/622 (%30)
Çarpım Blokları	46/300 (%15)
Maksimum frekans	7.13 Mhz

8.5.3. Kendini ayarlayabilen PI tip bulanık kontrolör deneyi

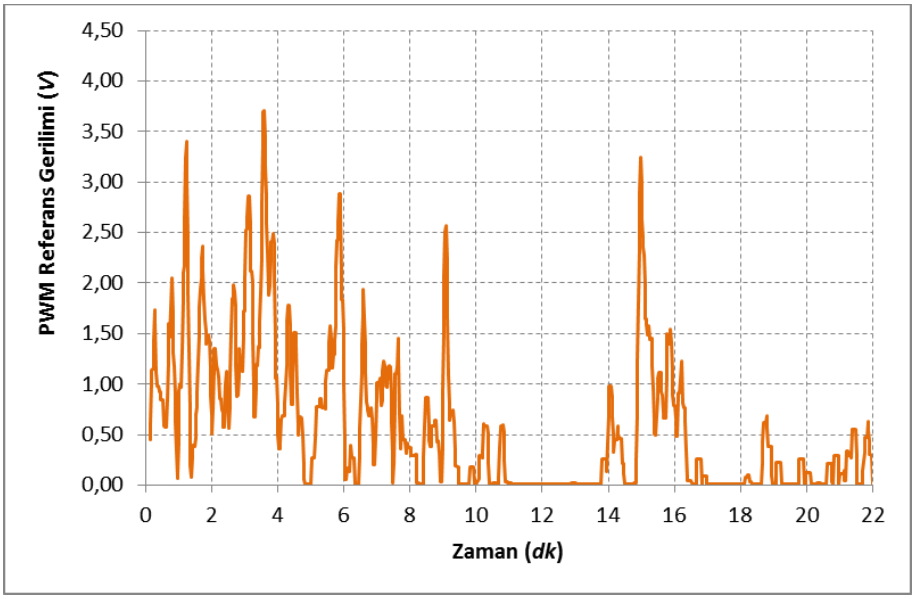
Gerçekleştirilen Kendini Ayarlayabilen PI tip Bulanık Kontrolör (KABK) ile 14 mm çaplı manyetik özellikli kürecikler ile oluşturulan EMF matrisi ile deneyler yapılmıştır. Deneylerde kullanılan endüstriyel sıvılar bir depo içerisindeki 200 lt suya, 4-10x10^{-4} mm büyüklüğündeki demir tozu ilave edilerek hazırlanmıştır. İstenilen performans değeri %95 olarak belirlenmiş ve deney 22 dk sürmüştür.

Deney süresince GPK, ÇPK ve AH değişimleri Şekil 8.33'de görüldüğü gibi elde edilmiştir.



Şekil 8.33. KABK ile çalışan filtrenin girişindeki parçacık konsantrasyonu

Deney süresince KABK ile elde edilen PWM referans geriliminin değişimi Şekil 8.34'da görüldüğü gibi elde edilmiştir. Şekil 8.32 ve Şekil 8.33 incelendiğinde PWM referans gerilimi ÇPK'nın arttığı yani EMF performansının düştüğü durumlarda ve GPK ve AH değişimlerinde tepki vermektedir.



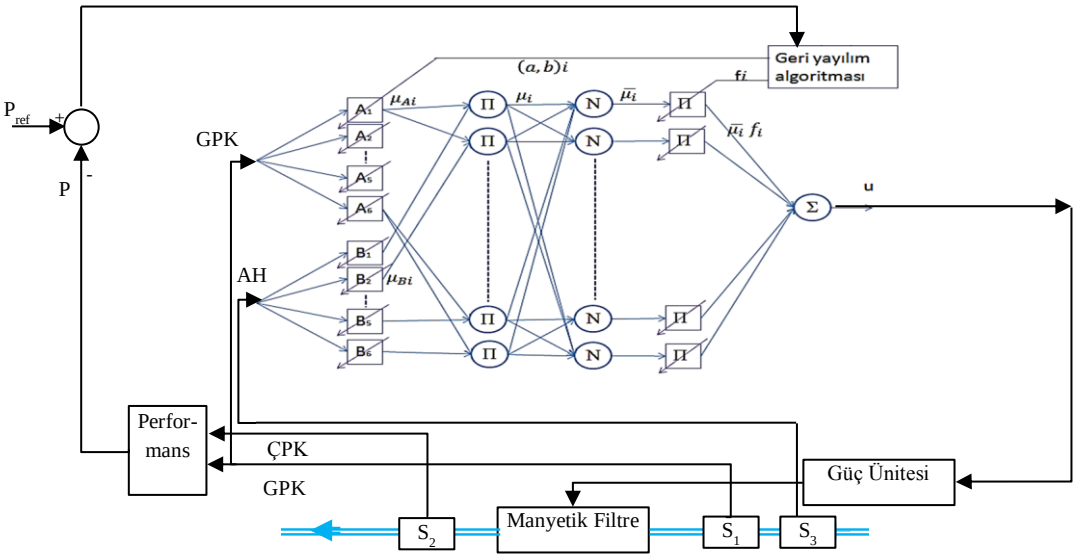
Şekil 8.34. KABK ile çalışan filtrenin PWM Referans gerilimi değişimi

8.6. ANFIS Kontrolör

Bu çalışmada, GPK ve AH giriş ve PWM referans gerilimi çıkışına sahip ANFIS kontrolör kullanılarak EMF kontrolü gerçekleştirilmiştir. Bu bölümde ANFIS kontrolörün yapısı, programlanması ve yapılan deney sonuçları verilmiştir.

8.6.1. ANFIS kontrolörün yapısı

EMF kontrolü için tasarlanan ANFIS kontrolörün genel yapısı Şekil 8.35'da verilmiştir.



Şekil 8.35. Tasarlanan ANFIS kontrolörün genel yapısı

Tasarlanan Takagi-Sugeno-Kang model tabanlı ANFIS kontrolör GPK ve AH olmak üzere iki giriş ve PWM referans gerilimi olmak üzere tek çıkışa sahiptir. GPK ve AH giriş değerlerinin bulanıklaştırmasında GPK-AH BK'da verilen dilsel ifadeler ve Şekil 8.4 ve Şekil 8.5'de verilen üyelik fonksiyonları kullanılmıştır. Referans performans değeri ile performans arasındaki fark ile ANFIS öncül ve sonuç parametreleri ayarlanmıştır. ANFIS kontrolörün eğitimi için Bölüm 5.3.2'de verilen geriye yayılım algoritması kullanılmıştır.

8.6.2. ANFIS kontrolörün kontrolörün FPGA'da programlanması

ANFIS kontrolörün tasarımı Şekil 8.35'da verilen yapıya uygun olarak tasarlanmıştır. ANFIS ile kontrol için kontrolör ve kontrolör eğitimi olmak üzere iki katmanlı bir yapı oluşturulmuştur. Kontrolör sürekli olarak çalışırken parametreleri eğitim bileşeni tarafından online olarak güncellenmektedir. ANFIS kontrolör yapısında sabit noktalı sayı formatı kullanılmıştır.

ANFIS kontrolör bulanıklaştırma işleminde Denklem 8.5'de verilen üçgensel üyelik fonksiyonu kullanılmıştır. Üçgensel üyelik fonksiyonunun a üçgenin orta noktası ve b üçgenin taban uzunluğu önşart parametreleri olup ANFIS eğitiminde adaptif düğümü oluşturmaktadırlar.

$$\mu_{Ai} = \begin{cases} 0 & ; \quad x < a - \frac{b}{2} \\ 1 - 2\frac{|x-a|}{b} & ; a - \frac{b}{2} < x < a + \frac{b}{2} \\ 0 & ; \quad x > a + \frac{b}{2} \end{cases} \tag{8.5}$$

Bu çalışmada sıfırıncı dereceden sugeno modeli kullanılmış olup, amaç parametrelerinin güncellenmesi Denklem 8.6'da verilen fonksiyon kullanılmıştır.

$$r_i(n+1) = r_i(n) + \eta e \frac{\mu_i}{\sum \mu_i} \tag{8.6}$$

ANFIS kontrolörün öncelikle MATLAB programı üzerinde kodlaması yapılarak uygun giriş verilerine karşılık sonuç parametrelerinin değişimi incelenmiştir. Örneğin GPK'nın 1048, AH'nın 40 ve performans hatasının 1 değerine karşılık sistem 448 çıkışını vermekte iken eğitim sonrasında 453 çıkışını vermektedir.

Sugeno tip bulanık kontrolörün çıkışı hesaplanırken ANFIS katmanları halinde hesaplamaya tabii tutularak sonuç elde edilmiştir. ANFIS bulanık kontrolörün basitleştirilmiş kaba kod yapısı aşağıda verilmiştir.

```
for i= 1 to GPK üyelik fonksiyon sayısı
{
        GPK_dilsel_sonuc(i)=bulanıklastirma(GPK_ucgensel_uyelik_orta_nokta(i),
        GPK_ucgensel_uyelik_taban_uzunlugu(i), GPK)
}
```

```
for i= 1 to AH üyelik fonksiyon sayısı
{
        AH_dilsel_sonuc(i)=bulanıklastirma(AH_ucgensel_uyelik_orta_nokta(i),
        AH_ucgensel_uyelik_taban_uzunlugu(i), AH)
}
for i= 1 to GPK üyelik fonksiyon sayısı
{
        for j= 1 to AH üyelik fonksiyon sayısı
        {
        katman3((i-1)*GPK üyelik fonksiyon sayısı + j) = AH_dilsel_sonuc(j)*GPK_dilsel_sonuc(i)
        }
}
katman4toplam=0;
for i= 1 to GPK üyelik fonksiyon sayısı * AH üyelik fonksiyon sayısı
{
        katman4toplam=katman4toplam+katman3(i)
}
for i= 1 to GPK üyelik fonksiyon sayısı * AH üyelik fonksiyon sayısı
{
        katman4(i)=katman3(i)/katman4toplam
}
for i= 1 to GPK üyelik fonksiyon sayısı * AH üyelik fonksiyon sayısı
{
        katman5(i)=katman4(i)/sugeno cikis (i)
}
cikis=0;
for i= 1 to GPK üyelik fonksiyon sayısı * AH üyelik fonksiyon sayısı
{
        cikis=cikis+katman5(i)
}
PWM referans değer = cikis;
```

Eğitim bileşeninde hesaplamada harcanan çevrim süresini azaltmak amacıyla GPK ve AH giriş değerlerinin aktif olduğu bölge *a_yeri* dizi değişkeninde belirlenmiş ve sadece bu bölgede işlem yapılmıştır. ANFIS eğitim bileşeninde Bölüm 5.3.2'de verilen ANFIS için geri yayılım algoritması kullanılarak, giriş değerlerinin üyelik fonksiyonları üçgensel tipte seçildiğinden ön şart parametrelerinin güncellenmesi işleminde Denklem 8.7 ve 8.8'de verilen fonksiyonlar elde edilmiştir.

$$a_i(n+1) = a_i(n) + \eta . e . f_i . \frac{\sum \mu_i - \mu_i}{\sum {\mu_i}^2} . \mu_{Bi} . -\frac{2}{b} . \text{sgn(a–GPK)} \tag{8.7}$$

$$b_i(n+1) = b_i(n) + \eta . e . f_i . \frac{\sum \mu_i - \mu_i}{\sum {\mu_i}^2} . \mu_{Bi} . 2 . \frac{|a-x|}{b^2} \tag{8.8}$$

ANFIS geri yayılım algoritması ile yapılan eğitim ile bulanık kontrolün çıkışında kullanılan amaç parametreleri ve üyelik fonksiyonlarının parametreleri olan ön şart parametreleri uyarlanmaktadır. ANFIS bulanık kontrol eğitim bileşeninin amaç parametrelerinin ve GPK ön şart parametrelerinin uyarlanmasını ifade eden basitleştirilmiş kaba kod yapısı aşağıda verilmiştir.

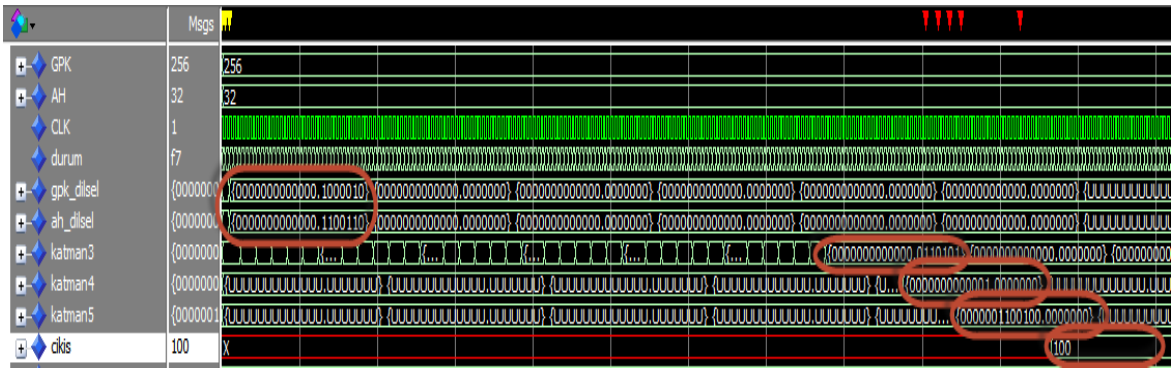
```
for i= 1 to GPK üyelik fonksiyon sayısı
{
        GPK_dilsel_sonuc_a(i)=bulanıklastirma(GPK_ucgensel_uyelik_orta_nokta(i),
                                GPK_ucgensel_uyelik_taban_uzunlugu(i), GPK)
}
for i= 1 to AH üyelik fonksiyon sayısı
{
        AH_dilsel_sonuc_a(i)=bulanıklastirma(AH_ucgensel_uyelik_orta_nokta(i),
                                AH_ucgensel_uyelik_taban_uzunlugu(i), AH)
}
for i= 1 to GPK üyelik fonksiyon sayısı
{
        for j= 1 to AH üyelik fonksiyon sayısı
        {
        a_katman3((i-1)*GPK üyelik fonksiyon sayısı + j) =
                                AH_dilsel_sonuc_a(j)*GPK_dilsel_sonuc_a(i)

        if (a_katman3(((i-1)*AH üyelik fonksiyon sayısı)+j)>0)
                {
                a_yeri(j)=0;
                }
                else
                {
                a_yeri(j)=(((i-1)*AH üyelik fonksiyon sayısı)+j)
                }

        }
}
a_katman4toplam=0;
for i= 1 to 4
{
        a_katman4toplam=a_katman4toplam+a_katman3(a_yeri(i))
}
for i= 1 to 4
{
    a_yeni_cikis(i)=a_katman4toplam*a_katman3(a_yeri(i))*ogrenme oranı* performans hatası
    sugeno_cikis(i)=sugeno_cikis(i)+a_yeni_cikis(i)
}
Oncul parametre carpani = hata*ogrenme/(a_katman4toplam*a_katman4toplam);
for i= 1 to 4
{
        indis1(i)=a_yeri(i) mod GPK uyelik fonksiyon sayısı;
        indis2(i)=((a_yeri(i)-indis1) / AH uyelik fonksiyon sayısı)+1
        oncul_parametre_geri yayilim ortak carpim = oncul parametre carpani
*sugeno_cikis(a_yeri(i))*(a_katman4toplam-a_katman3(a_yeri(i)))*GPK_dilsel_sonuc_a(i)
        if (( GPK_ucgensel_uyelik_orta_nokta(indis1(i))-GPK)>0)
        {
                GPK_orta_nokta_geri yayilim=oncul_parametre_geri yayilim ortak carpim*-2 /
GPK_ucgensel_uyelik_taban_uzunlugu(indis1(i))
        }
        else
        {
                GPK_orta_nokta_geri yayilim=oncul_parametre_geri yayilim ortak carpim*-2 /
GPK_ucgensel_uyelik_taban_uzunlugu(indis1(i))
        }
        if (( GPK_ucgensel_uyelik_orta_nokta(indis1(i))-GPK)>0)
        {
```

```
                GPK_uzunluk_geri yayilim=oncul_parametre_geri yayilim ortak carpim*2*
(GPK_ucgensel_uyelik_orta_nokta(indis1(i))-GPK)/
GPK_ucgensel_uyelik_taban_uzunlugu(indis1(i))*GPK_ucgensel_uyelik_taban_uzunlugu(indis1(i));
        }
        else
        {
                GPK_uzunluk_geri yayilim=oncul_parametre_geri yayilim ortak carpim*-2*
(GPK_ucgensel_uyelik_orta_nokta(indis1(i))-GPK)/
GPK_ucgensel_uyelik_taban_uzunlugu(indis1(i))*GPK_ucgensel_uyelik_taban_uzunlugu(indis1(i));
        }
        GPK_ucgensel_uyelik_orta_nokta(indis1(i))=
        GPK_ucgensel_uyelik_orta_nokta(indis1(i))+GPK_orta_nokta_geri yayilim
        GPK_ucgensel_uyelik_taban_uzunlugu(indis1(i))
=GPK_ucgensel_uyelik_taban_uzunlugu(indis1(i))+GPK_uzunluk_geri yayilim
}
```

ANFIS kontrolör bileşenleri EMF kontrolü için yeterli sürenin bulunması ve hibrid yapılarda kullanılması esnasında FPGA alanını optimize edebilmek için sonlu durum makinesi kullanılarak gerçekleştirilmiştir. Şekil 8.36'de kontrolör katmanının modelsim simülasyon sonucu verilmiştir. İşlemler ardışıl sırada ve ANFIS katman yapısında programlanmıştır. ANFIS kontrol ve eğitim bileşenin VHDL kod yapısı EK-3'de verilmştir. ANFIS kontrolör bileşeni 214, kontrolör eğitim bileşeni ise 270 çevrim süresinde hesaplama yapmaktadır.



Şekil 8.36. Kontrolör katmanı Modelsim simülasyon sonuçları

VHDL ile kodlaması yapılan ANFIS kontrolörün lojik sentezleme, yerleştirme ve route işlemleri Altera Quartus programı ile yapılmıştır. Cyclone II EP2C70F896 FPGA lojik elementinin %60'ı kullanılarak sentezlenmiştir (Çizelge 8.13).

Çizelge 8.13. ANFIS kontrolör için FPGA kaynak kullanımı

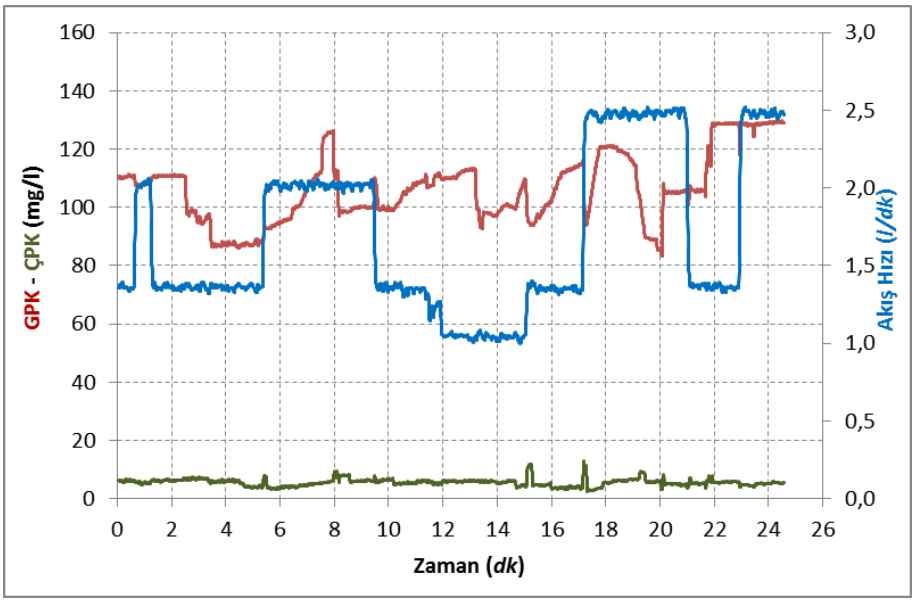
Kaynak	**Kullanım oranı**
Kombinasyonel fonksiyonlar	36332/68416 (%53)
Ayrılmış yazmaçlar	12772/68416 (%9)
Toplam Lojik Element	40918/68416 (%60)
Pin	186/622 (%30)
Çarpım Blokları	70/300 (%23)
Maksimum frekans	68.8 Mhz

50 MHz saat frekansında ANFIS kontrolör çalıştığında toplam işlem zamanı kontrolör bileşeni için 4.28 µs, kontrolör eğitim bileşeni için 5.4 µs'dir.

8.6.3. ANFIS kontrolör deneyi

Gerçekleştirilen ANFIS kontrolör ile 14 mm çaplı manyetik özellikli kürecikler ile oluşturulan EMF matrisi ile deneyler yapılmıştır. Deneylerde kullanılan endüstriyel sıvılar bir depo içerisindeki 200 lt suya, 4-10x10^{-4} mm büyüklüğündeki demir tozu ilave edilerek hazırlanmıştır. İstenilen performans değeri %95 olarak belirlenmiş ve deney yaklaşık 19 dk sürmüştür.

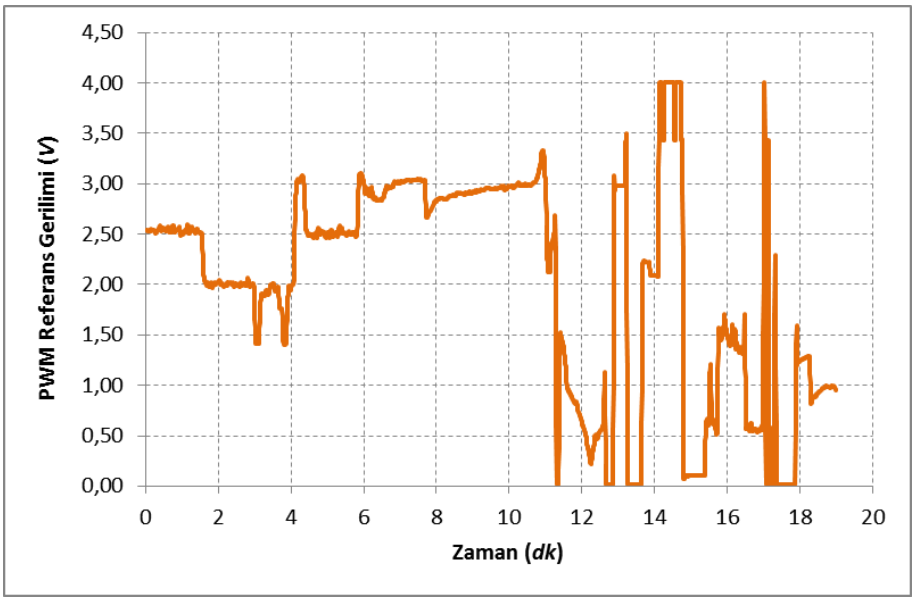
Deney süresince GPK, ÇPK ve AH değişimleri Şekil 8.37'de görüldüğü gibi elde edilmiştir.



Şekil 8.37. ANFIS kontrolör ile çalışan filtrenin GPK, ÇPK ve AH değişimi

Deney süresince ANFIS kontrolör ile elde edilen PWM referans geriliminin değişimi Şekil 8.38'de görüldüğü gibi elde edilmiştir. Şekil 8.36 ve Şekil 8.37 beraber

incelendiğinde ANFIS kontrolörün GPK ve AH girişlerindeki değişimlere göre tepki verdiği ve ANFIS kontrol sistemin EMF performans hatasına göre öncül ve sonuç parametrelerinin ayarlaması neticesinde EMF performansına göre de PWM referans gerilimini değiştirdiği görülmektedir.



Şekil 8.38. ANFIS kontrolör ile çalışan filtrenin PWM Referans gerilimi değişimi

8.7. Hibrid Zeki Kontrolör

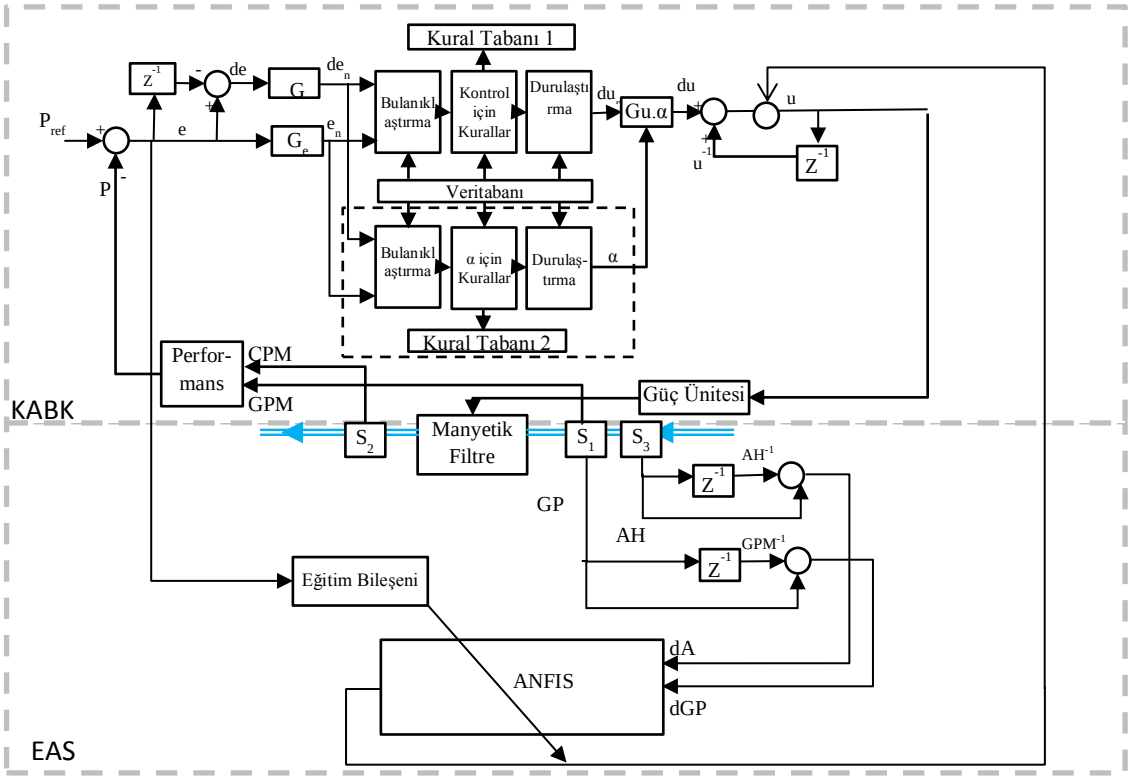
Bu çalışmada performans hatasına göre çalışan KABK yapısı ile ANFIS kontrolör yapısı birleştirilerek bir hibrid zeki kontrolör oluşturulmuştur. Bu bölümde Hibrid zeki kontrolörün yapısı, programlanması ve yapılan deney sonuçları verilmiştir.

8.7.1. Hibrid zeki kontrolörün yapısı

EMF performansının değişimine göre kontrol işlemini gerçekleştiren kontrolörler EMF performansını kontrol altında tutabilmektedirler. Bu kontrolörler ile EMF performansını etkileyen tüm teknolojik parametrelere duyarlı bir kontrol yapısı elde edilmektedir. Ayrıca GPK ve AH teknolojik parametrelerindeki ani değişimlerin EMF performansını etkilediği yapılan deneylerde görülmektedir. Bu çalışmada GPK ve AH ani değişimlerine tepki verebilecek bir kontrolör yapısı geliştirilerek diğer teknolojik parametrelere duyarlı olmasını da sağlamak için EMF performans hatasına göre çalışan kontrolör ile hibrid olarak kullanılmıştır. Böylece EMF kiti üzerinde

bulunan sensörler ile ölçülebilen GPK ve AH değişimlerine EMF performansına etki etmeden tepki verebilen ve ölçülemeyen tüm teknolojik parametreleri de performans hatası değişimine göre kontrol edebilen bir kontrolör oluşturulmuştur.

Tasarlanan hibrid zeki kontrolör (HZK) genel olarak iki kontrolör yapısının birleştirilmesinden oluşturulmuştur. Bunlardan ilki KABK, ikincisi ise ANFIS kontrolör yapısındaki Erken Algılama Sistemi (EAS)'dir. Hibrid zeki kontrolörün genel yapısı Şekil 8.39'de verilmiştir.



Şekil 8.39. Tasarlanan hibrid zeki kontrolörün genel yapısı

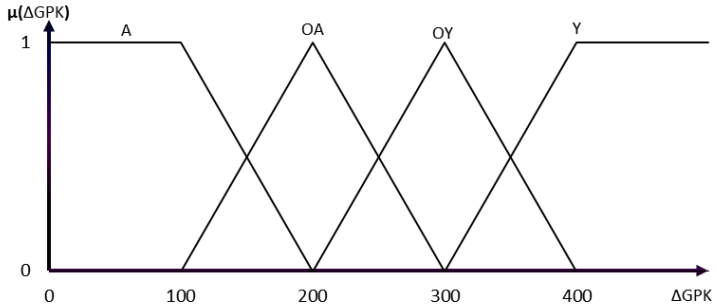
EMF performans değişimine göre kontrol işlemini geçekleştiren KABK Bölüm 8.5.1'de verildiği yapıda kullanılmıştır.

ANFIS yapısındaki EAS ise GPK değişimi ve AH değişimi giriş değerlerine ve EAS PWM referans gerilimi çıkış değerine sahiptir. GPK değişimi ve AH değişimi girişleri Denklem 8.9 ve 8.10'da verilmiştir.

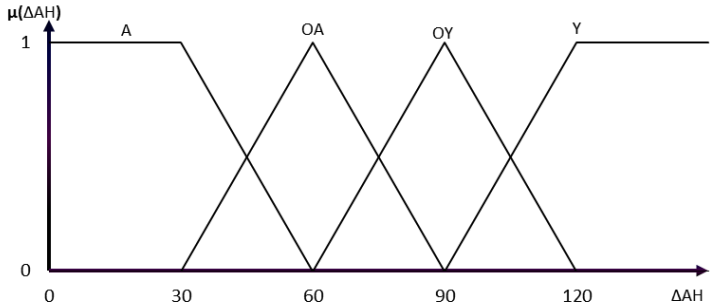
$$\Delta GPK(k)=GPK(k)-GPK(k-1) \quad (8.9)$$

$$\Delta AH(k)=AH(k)-AH(k-1) \quad (8.10)$$

GPK değişimi ve AH değişimi giriş değişkenleri {az (A), orta az (OA), orta yüksek (OY) , yüksek (Y)} olmak üzere 4 dilsel ifadeye ayrılmışlardır. Giriş değişkenleri GPK değişimi ve AH değişimi için üyelik fonksiyonlarının gösterimi Şekil 8.40 ve Şekil 8.41'de verilmiştir.



Şekil 8.40. ΔGPK üyelik fonksiyonları



Şekil 8.41. ΔAH üyelik fonksiyonları

Giriş ve çıkış değişkenleri için bulanık kural tablosu Çizelge 8.14'da verilmiştir.

Çizelge 8.14. EAS PWM referans gerilimi için kural tabanı

		ΔGPK			
		A	**OA**	**OY**	**Y**
ΔAH	**A**	100	200	300	400
	OA	200	300	400	500
	OY	300	400	500	600
	Y	400	500	600	700

Hibrid zeki kontrolörün çıkışı ise Denklem 8.11 ile ifade edildiği gibidir

$$u(k)=u(k-1)+\Delta u(k)+u_{EAS} \quad (8.11)$$

8.7.2. Hibrid zeki kontrolörün FPGA'da programlanması

Şekil 8.39'de görüldüğü gibi Hibrid Zeki Kontrolör KABK ile ANFIS kontrolör yapılarının birleştirilmesi ile oluşmaktadır. Bu kontrolörün FPGA üzerinde programlanmasında Bölüm 8.5.2'de verilen KABK ve Bölüm 8.6.2'de verilen ANFIS kontrolör yapıları kullanılmıştır. Bu kontrol yapılarının giriş, çıkış, üyelik fonksiyonu gibi parametreleri hibrid zeki kontrolör için tekrar düzenlenmiştir.

Hibrid zeki kontrolörün lojik sentezleme, yerleştirme ve route işlemleri Altera Quartus programı ile yapılmıştır. Cyclone II EP2C70F896 FPGA lojik elementinin %91'i kullanılarak sentezlenmiştir (Çizelge 8.15).

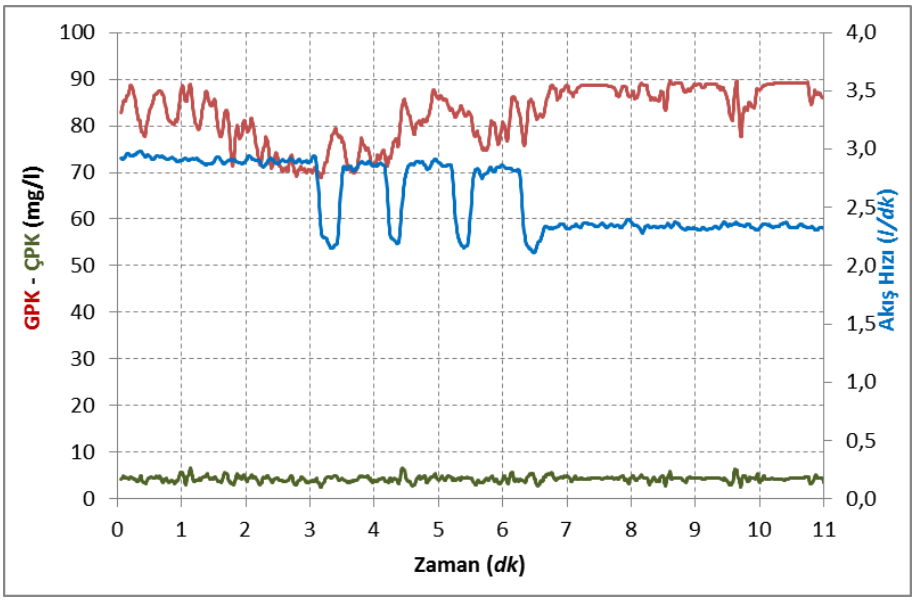
Çizelge 8.15. Hibrid Zeki Kontrolör için FPGA kaynak kullanımı

Kaynak	**Kullanım oranı**
Kombinasyonel fonksiyonlar	54750/68416 (%80)
Ayrılmış yazmaçlar	8069/68416 (%12)
Toplam Lojik Element	62819/68416 (%91)
Pin	186/622 (%30)
Çarpım Blokları	104/300 (%35)
Maksimum frekans	7.1 Mhz

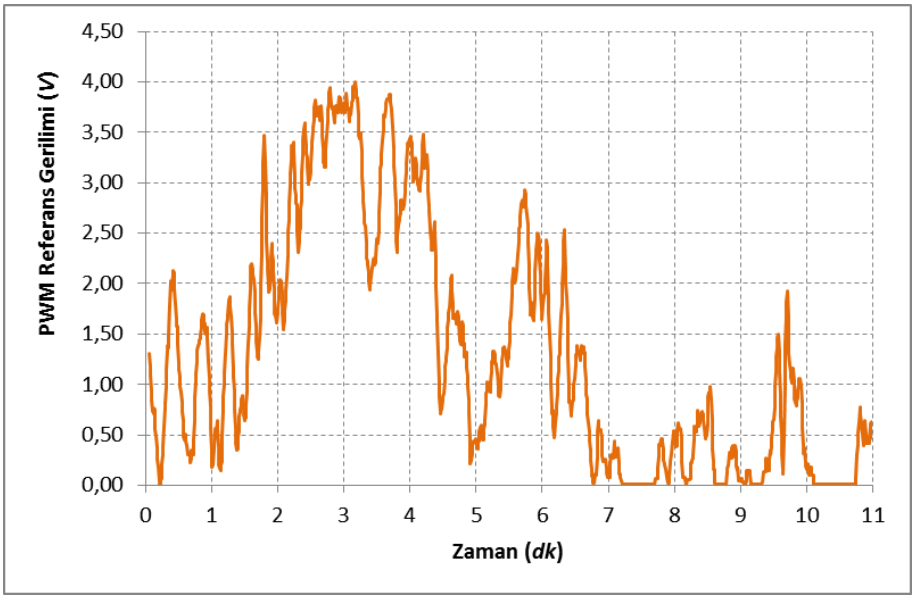
8.7.3. Hibrid zeki kontrolör deneyi

Gerçekleştirilen HZK ile 14 mm çaplı manyetik özellikli kürecikler ile oluşturulan EMF matrisi ile deneyler yapılmıştır. Deneylerde kullanılan endüstriyel sıvılar bir depo içerisindeki 200 lt suya, 4-10x10^{-4} mm büyüklüğündeki demir tozu ilave edilerek hazırlanmıştır. İstenilen performans değeri %95 olarak belirlenmiş ve deney 11 dk sürmüştür.

Deney süresince GPK, ÇPK ve AH değişimleri Şekil 8.42'de ve HZK ile elde edilen PWM referans geriliminin değişimi Şekil 8.43'de görüldüğü gibi elde edilmiştir.



Şekil 8.42. HZK ile çalışan filtrenin GPK, ÇPK ve AH değişimi



Şekil 8.43. HZK ile çalışan filtrenin PWM Referans gerilimi değişimi

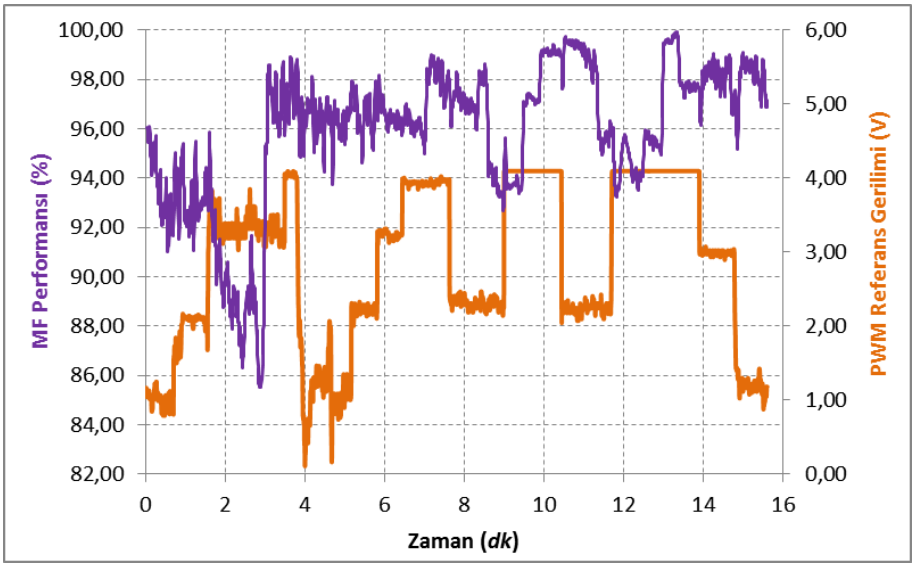
Şekil 8.41 ve Şekil 8.42 incelendiğinde HZK sistemin EMF performans hatasına PWM referans gerilimini değiştirmesinden dolayı, GPK'nın düşük ÇPK'nın yüksek olduğu durumlara artış yönünde tersi durumlarda ise düşüş yönünde tepkiler oluşmaktadır. Ayrıca HZK, GPK ve AH'da artış meydana geldiği durumlarda da PWM referans gerilimini arttırmaktadır.

9. SONUÇLAR ve ÖNERİLER

9.1. Sonuçlar

Yapılan çalışmada EMF kontrolü için farklı denetleyici yapıları geliştirilmiş ve bu denetleyiciler kullanılarak deneyler yapılmıştır. Yapılan bu kontrolör deneylerinde aşağıdaki sonuçlar elde edilmiştir.

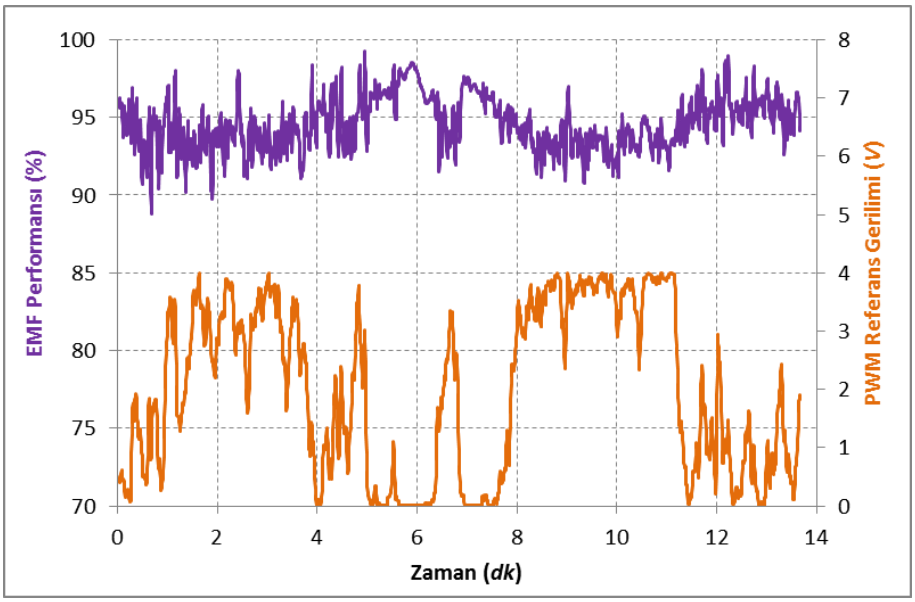
GPK ve AH değişimine göre EMF performansının kontrol edilmesi (Bölüm 8.2) deneyinde EMF performansının ve PWM referans geriliminin değişimi Şekil 9.1'de verilmiştir.



Şekil 9.1. GPK-AH BK deneyi: EMF performans ve PWM referans gerilimi

Kontrolör GPK ve AH'ye göre sistemi kontrol ettiğinden dolayı ÇPK'yı dolayısıyla EMF performansı değişimlerini dikkate almamaktadır. Şekil 9.1'de görüldüğü gibi EMF performansının değişim gösterdiği bölümlerde PWM referans gerilimi bundan bağımsız olarak değişim göstermektedir. Bu durum yapılan kontrolörün EMF performansını istenilen seviyede kontrol edemediğini göstermektedir.

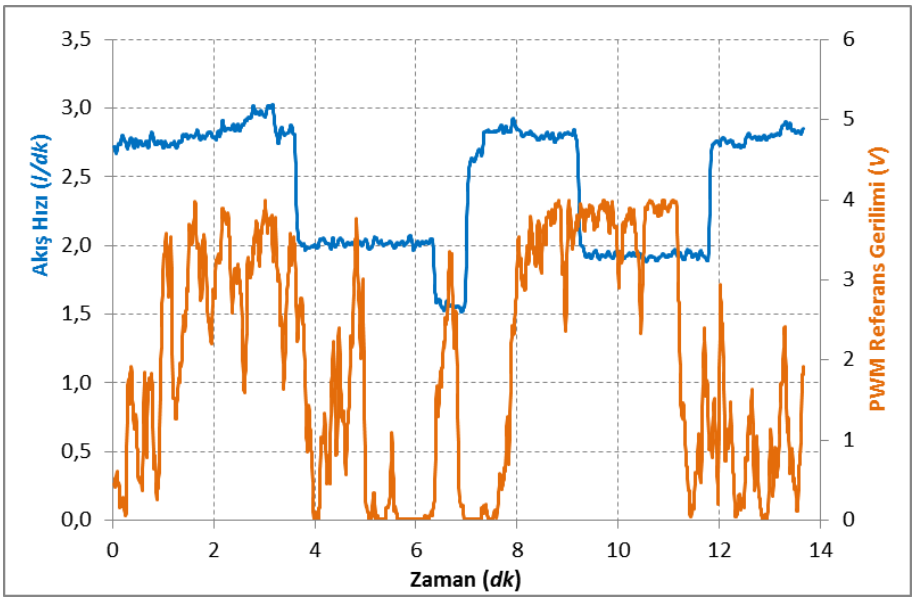
Bu eksikliğin giderilebilmesi için EMF performansını dikkate alan PI tip BK tasarlanmış ve gerçekleştirilmiştir (Bölüm 8.3). Gerçekleştirilen PI tip BK deneyinin EMF performansının ve PWM referans geriliminin değişimi Şekil 9.2'de verilmiştir.



Şekil 9.2. PI tip BK deneyi: EMF performans ve PWM referans gerilimi

Şekil 9.2 incelendiğinde EMF performansı değişimlerine PWM referans geriliminin hemen tepki verdiği ve EMF performans değerinin %95 referans performans değeri üzerine çıktığı durumlarda PWM referans gerilimi değerini düşürdüğü altına düştüğü durumlarda da PWM referans gerilimi değerini yükselttiği görülmektedir.

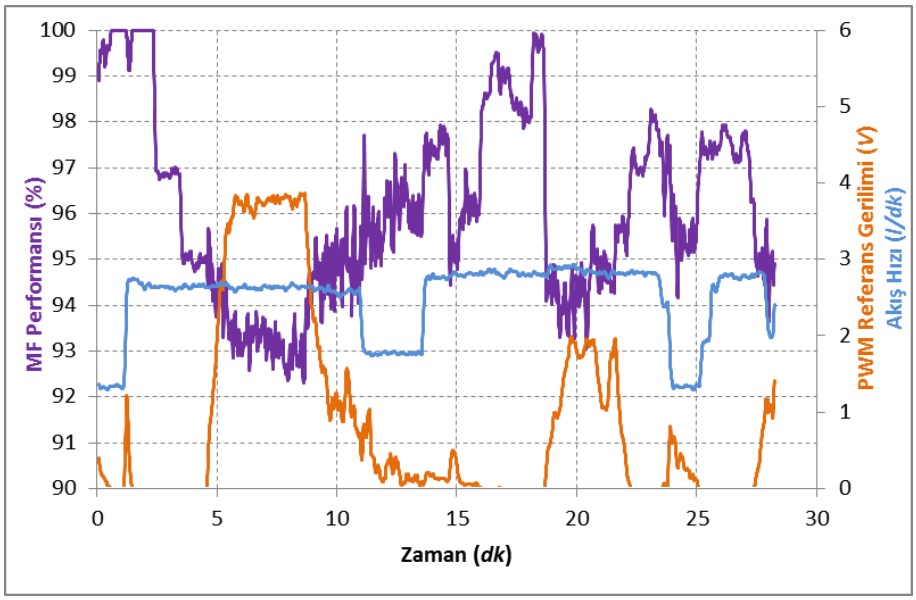
Şekil 9.3'de görüldüğü gibi PWM referans gerilimi AH'dan bağımsız olarak tepki vermektedir. Her ne kadar AH'yı dikkate almıyorsa da AH'daki değişimler ÇPK'yı manyetik akı yoğunluğunun yetersiz kaldığı durumlarda değiştirmektedir.



Şekil 9.3. PI tip BK deneyi: AH ve PWM referans gerilimi

Şekil 9.2 ve 9.3 incelendiğinde PI tip BK, EMF performansını dikkate almakta akış hızını ise dikkate almamaktadır. Yine grafiklerden görüldüğü gibi PI tip BK performans değişimini algıladıktan sonra tepki vermektedir. Bu durum EMF performansını iyileştirme de bir dezavantajdır. Ayrıca gerçekleştirilen kontrolörün EMF performansını istenilen seviyede tutamadığı durumlara kendini adapte edip düzenleyememesi de bir dezavantaj oluşturmaktadır.

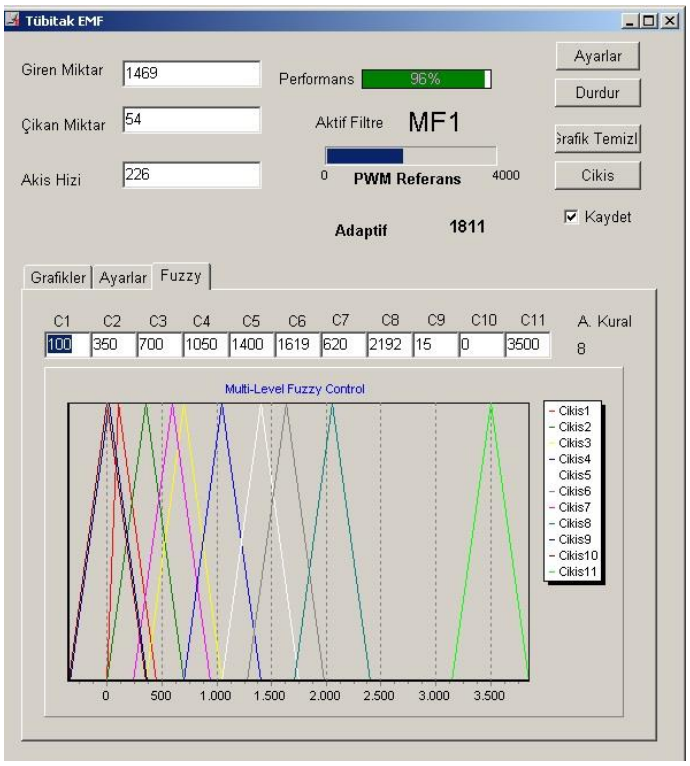
Bahsedilen dezavantajları ortadan kaldırabilmek için KOEBK tasarlanıp gerçekleştirilmiştir (Bölüm 8.4.1). Gerçekleştirilen KOEBK deneyinin EMF performansı, AH ve PWM referans geriliminin değişimi Şekil 9.4'de verilmiştir.



Şekil 9.4. KOEBK deneyi: EMF performans, AH ve PWM referans gerilimi

Şekil 9.4 incelendiğinde kontrolörün EMF performansı ve AH değişimlerini dikkate aldığı görülmektedir. EMF performansının istenilen %95 referans performans değerinin üzerinde olduğu durumlarda AH değişimlerini dikkate alarak kendini adapte etmekte ve tepki vermektedir. Bu deney süresince GPK kontrolörün tepki vermesini sağlayacak değişim göstermediğinden grafikte karşılaştırılmalı olarak verilmemiştir.

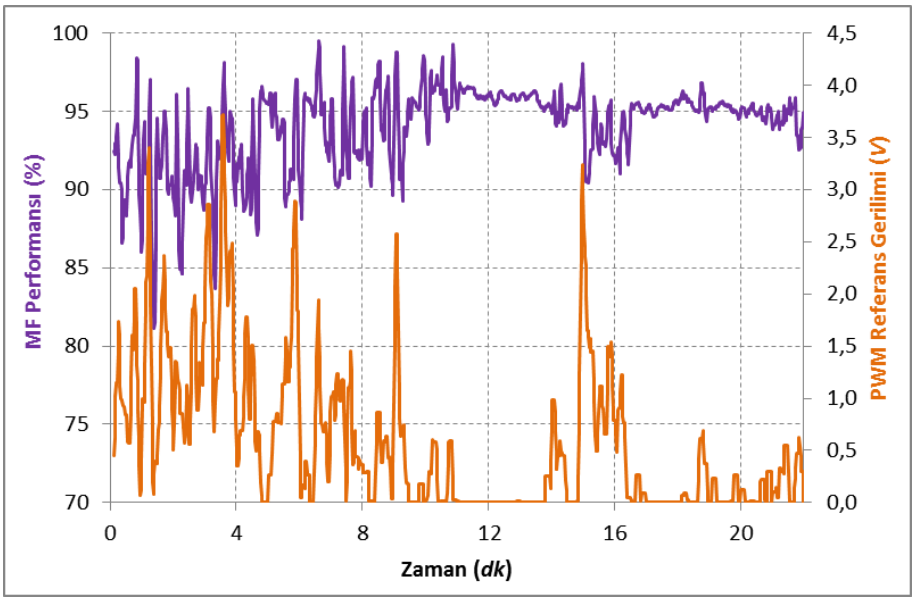
Kontrolör yapısı incelendiğinde ilk katman BK, GPK ve AH değişimlerine göre PWM referans gerilimi vermekte iken ikinci katman BK ilk katman kontrolü sonucunda oluşan EMF performans hatasını göz önüne alarak ilk katman BK'nın çıkış üyelik değerlerini değiştirerek düzenlemektedir. Bu durum deney sürecinde herhangi bir anda alınan Şekil 9.5'deki bilgisayar arayüzü görüntüsünden görülmektedir.



Şekil 9.5. KOEBK için çıkış üyelik değerlerinin değişimi

KOEBK'da ikinci katman ilk katmanın çıkış üyelik değerlerini değiştirme işleminde, GPK ve AH'nın ani tepkilerinde tekrar değiştirme işlemi gerçekleştirilirken performans değişiminden sonra ilk katmanda adapte işlemi uygulanmaktadır. Bu durum ise bir dezavantaj oluşturmaktadır.

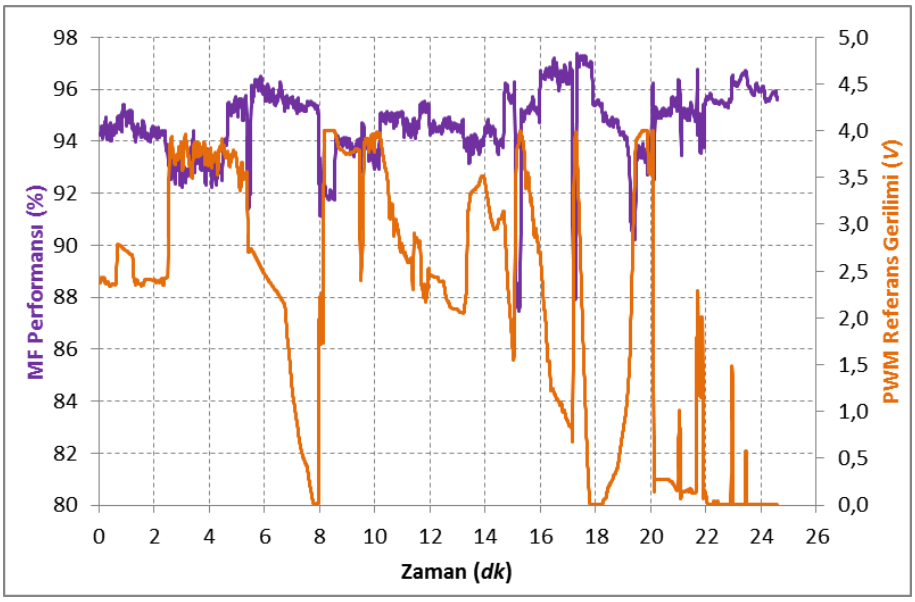
Ayrıca PI tip BK'nın EMF performansının ani değişimlerindeki tepkisini geliştirebilmek amacıyla KABK tasarlanarak gerçekleştirilmiştir (Bölüm 8.5.1). KABK deneyindeki EMF performansının ve PWM referans geriliminin değişimi Şekil 9.6'da verilmiştir.



Şekil 9.6. KABK deneyi: EMF performansı ve PWM referans gerilimi

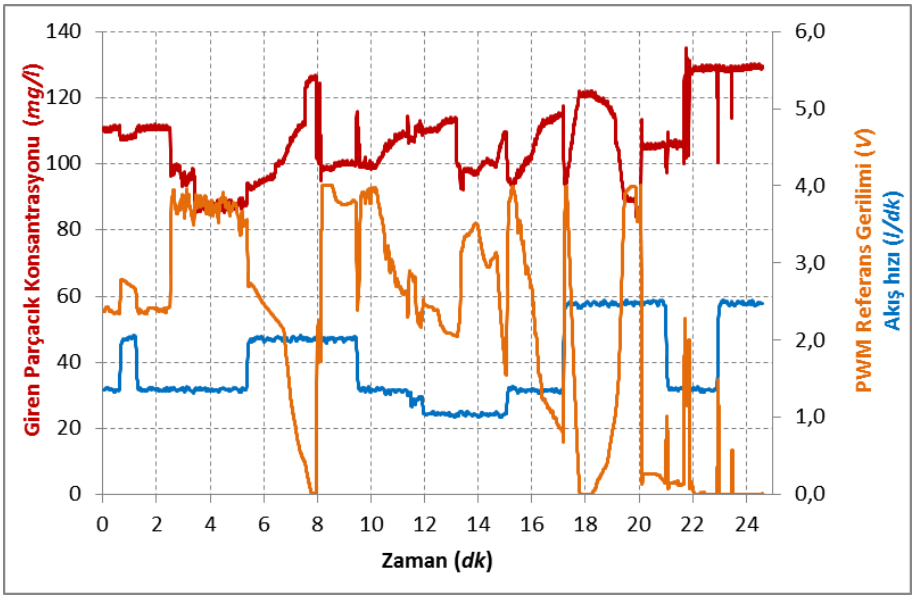
Şekil 9.2'de verilen PI tip BK grafiğinde görüldüğü gibi EMF, referans performans değerinin altında çalıştığı süre içerisinde maksimum PWM gerilimi verilmemektedir. Ayrıca, KABK deneyinin Şekil 8.33'de verilen ÇPK grafiğinde de verildiği gibi Bölge II çalışma karakteristik özelliğine uygun olarak tutma bölgelerinde ani kopmalar meydana gelmek ve bu durum EMF performansını anlık olarak düşürebilmektedir. KABK ile ilk katman PI tip BK çıkışı olan artırım miktarının ölçeklendirme katsayısını ayarlamak suretiyle, Şekil 9.6'da görüldüğü gibi istenilen EMF performansını elde edebilmek için EMF performansındaki ani değişimlere daha uygun tepkiler verilmektedir.

Kullanım alanlarına ve endüstriyel sıvılara bağlı olarak EMF performansını etkileyen teknolojik faktörlerin değişimi, bir uzman görüşüne dayalı olarak gerçekleştirilen BK, EMF performansını kontrol etmede zamanla yetersiz olabilmektedir. Bu durumu ortadan kaldırarak sistemi etkileyen faktörlerin zamanla değişimine bağlı olarak kontrolörün kendini adapte edebilmesi için BK'ya öğrenme yeteneği kazandırılması amaçlanmıştır. Bu amaçla elde edilen BK'yı kullanabilen ve bunu EMF performans hatasına göre eğiten ANFIS kontrolör yapısı kullanılmıştır (Bölüm 8.6.1). ANFIS deneyindeki EMF performansının ve PWM referans geriliminin değişimi Şekil 9.7'de; GPK, AH ve PWM referans gerilimi değişimi Şekil 9.8'de verilmiştir.



Şekil 9.7. ANFIS kontrolör deneyi: EMF performansı ve PWM referans gerilimi

Şekil 9.7 incelendiğinde EMF performansı değişimlerine referans performans değerinden büyük yada küçük olmasına bağlı olarak PWM referans geriliminin değiştiği, ANFIS kontrolörün tepki verdiği görülmektedir. Ancak, yalnızca EMF performans değişimine değil aynı anda Şekil 9.8'de görüldüğü gibi GPK ve AH değişimlerine karşı da tepki vermektedir. Böylece ANFIS kontrolör; GPK, AH ve EMF performansına bağlı olarak EMF performansını istenilen %95 referans performans değerine hızlı bir şekilde ulaştırmak için PWM referans gerilimini değiştirmektedir.



Şekil 9.8. ANFIS kontrolör deneyi: GPK, AH ve PWM referans gerilimi

ANFIS kontrolörü, deney süresince Geri Yayılım Algoritması ile online eğitimi gerçekleştirerek giriş ve çıkış üyelik fonksiyonlarının parametrelerini sisteme adapte etmektedir. Bu deney süresi sonunda çıkış üyelik fonksiyonlarının değişimi Çizelge 9.1'de verilmiştir.

Çizelge 9.1. ANFIS çıkış üyelik fonksiyon değişimi

de* \ *e	NB	NO	NK	S	**PK**	**PO**
NB	100	350	700	1050	1400	1750
NO	350	700	1050	1400	**1763**	**2281**
NK	700	1050	1400	1750	**2147**	**2562**
S	1050	1450	1750	2100	**2463**	**2885**
PK	1450	1750	2100	2450	**2906**	**3364**
PO	1750	2100	2450	2800	3150	3500

ANFIS kontrolör, sistem çalışırken GPK ve AH değişimlerine göre PWM referans gerilimini üretmekte ve oluşan EMF performans hatasına göre kendini eğitmektedir. Bu durum, GPK ve AH üyelik fonksiyonu parametrelerinin performans hatasına göre düzenlenmesi şeklindedir. Elde edilen sonuç PI tip BK gibi performans hatasına göre çalışan kontrolör yapılarıyla benzerlik göstermektedir.

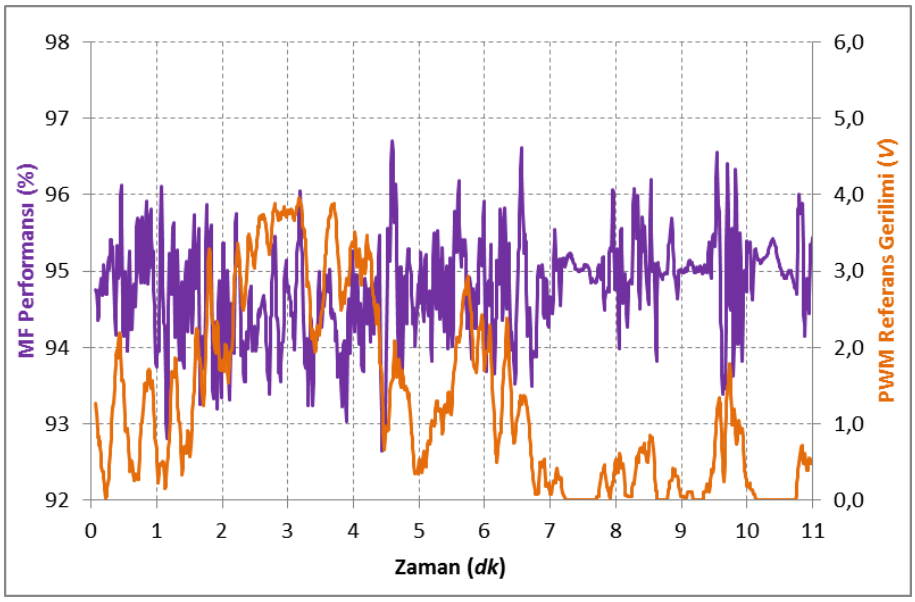
Buraya kadar yapılan kontrolör çalışmaları EMF performansını dikkate alan bir biçimde ilerlemiştir. Bu kontrolör yapılarında elde edilen sonuçlar genel anlamda değerlendirildiğinde GPK ve AH değişimlerini erken algılayarak tepki verememektedirler. Yani filtrasyon işleminde ÇPK ve dolayısıyla EMF performansı değiştikten sonra tepki vermektedirler. BK yapısında uzman tarafından önceden oluşturulan GPK ve AH üyelik fonksiyon parametrelerini EMF'nin kullanılacağı endüstriyel alana göre belirleme zorunluluğunu da beraberinde getirmektedir.

Bir filtrasyon işleminde;

- Endüstriyel sıvının AH sürekli değişim gösterebilmektedir ve t_1 zamanında AH sensörü tarafından AH algılanmaktadır.
- Giren parçacık miktarı sürekli değişim gösterebilmektedir ve t_1 zamanında Giren parçacık sensörü tarafından GPK algılanmaktadır.
- Parçacıklar EMF matrisine ulaşmakta ve filtrasyon işlemine tabii tutulmaktadır.

- Endüstriyel sıvı EMF matrisinden filtrasyon işlemine tabii tutularak çıkış sensörüne ulaşmakta ve $t_1+\Delta t$ zamanında ÇPK algılanmaktadır.

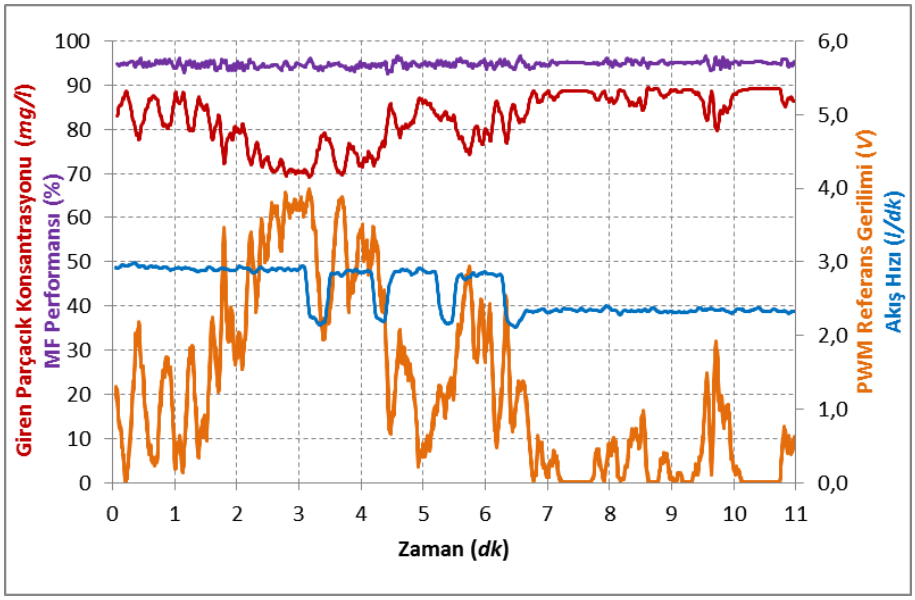
Tüm bunlar göz önüne alındığında, yukarıda bahsedilen dezavantajları ortadan kaldırmak ve erken algılamayı sağlamak için ΔGPK ve ΔAH'ya göre EMF sistemini kontrol eden bir Erken Algılama Sistemi (EAS) tasarlanmıştır. Kullanılan endüstriyel sıvının türü ve çalışma şartlarına göre EAS için verilen BK üyelik fonksiyonu parametre ve çıkış değerlerinin değişebileceğinden ve BK parametrelerini doğrulayacak bir uzman bilgisi bulunmadığından dolayı EAS ANFIS yapısında tasarlanmıştır. EAS'nin ANFIS yapısında olması ile EMF performans hatasına göre öncül ve sonuç parametreleri sisteme uygun olarak güncellenmektedir. Bu sayede sistem ΔGPK ve ΔAH için en uygun tepkiyi verebilmektedir. EMF performans hatasına göre sistemi kontrol edebilen KABK ile ANFIS yapısında ki EAS birleştirilerek bir hibrid zeki kontrolör gerçekleştirilmiştir (Bölüm 8.7.1). Hibrid zeki kontrolör deneyinin EMF performansı ve PWM referans geriliminin değişimi Şekil 9.9'de; GPK, AH, EMF Performansı ve PWM referans gerilimi değişimi Şekil 9.10'da; GPK, AH ve EAS PWM referans gerilimi Şekil 9.11'de verilmiştir.



Şekil 9.9. Hibrid zeki kontrolör deneyi: EMF performansı ve PWM referans gerilimi

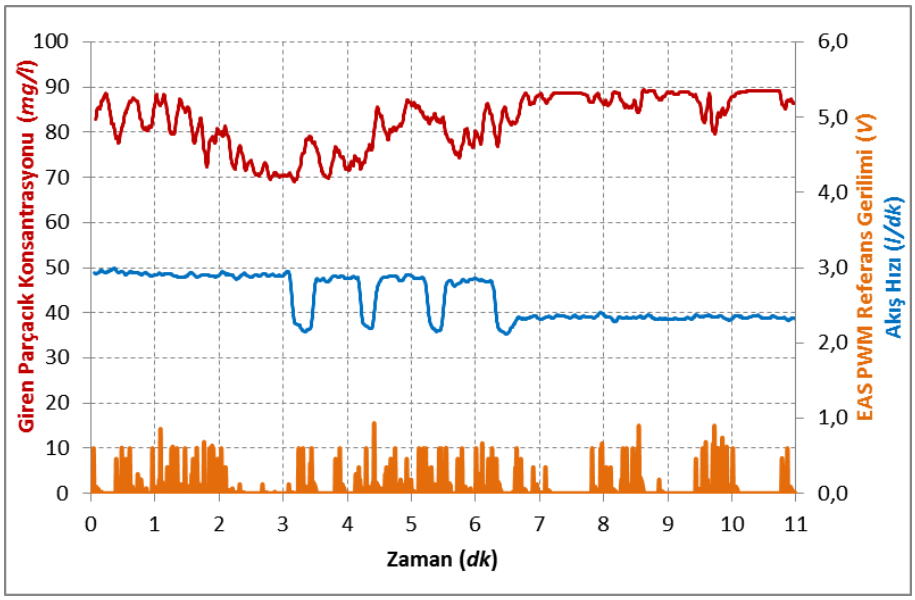
Şekil 9.9 incelendiğinde EMF performansının %95 referans performans değerinden değişimlerine göre kontrolörün tepki verdiği ve EMF performansını istenilen performans değerine getirmeye çalıştığı görülmektedir. Burada yalnızca EMF

performansına göre değil, aynı zamanda GPK ve AH değişimi değerlerine göre de kontrolör tepki vermektedir. Şekil 9.10'da bu durum AH'nın değişimi durumlarında PWM referans gerilimi değişimi ile görülmektedir.



Şekil 9.10. Hibrid zeki kontrolör deneyi: GPK, AH, EMF Performansı ve PWM referans gerilimi

Şekil 9.11 incelendiğinde EAS PWM referans geriliminin GPK ve AH değişimi değerlerine göre verdiği tepki miktarı görülmektedir. GPK ve AH değerlerinde değişim olmadığı durumlarda EAS PWM referans gerilimi değeri sıfır olmaktadır. EAS PWM gerilimi EMF bobin akımını kontrol eden PWM referans gerilimini arttırmak suretiyle oluşabilecek EMF performans değişimlerine karşı önceden tepki vermektedir.



Şekil 9.11. Hibrid zeki kontrolör deneyi: GPK, AH ve EAS PWM referans gerilimi

Hibrid zeki kontrolör PWM referans gerilimini EAS ile önceden ayarlayarak istenen EMF performans değerinden uzaklaşılmasını engellemektedir. Bu durum, filtrasyon işleminde Bölge II çalışma şartları için bile EMF performansının istenilen seviyede tutulduğunu göstermektedir.

Hibrid zeki kontrolör deneyi, diğer deneylerde görülen ani GPK ve AH değişimine bağlı olarak gerçekleşen EMF matrisi elemanları tarafından tutulan manyetik özellikli parçacıkların kopmalarının veya tutulamamalarının önlendiğini göstermektedir.

Ayrıca, tasarlanıp gerçekleştirilen filtre matrisi dolan EMF'yi temizleme sürecini başlatan ve diğer EMF'yi endüstriyel sıvıyı temizleme işlemine sokan kontrolör sayesinde endüstriyel üretim sürecindeki kesintilerin, iş kaybının, zaman kaybının ve ekonomik kayıpların da önüne geçilmiş olunmaktadır.

Giriş bölümünde belirtildiği gibi, filtrasyon işleminde tutulamayan veya kopan manyetik özellikli parçacıkların endüstriyel üretimde kaliteyi düşürdüğü, ekonomik ve zaman kayıplarına yol açtığı dezavantajlı durumların, gerçekleştirilen sistem tarafından başarılı bir şekilde ortadan kaldırdığı görülmektedir.

Geliştirilen kontrol sisteminin FPGA tabanlı olarak gerçekleştirilmesi ile sistemin bilgisayardan bağımsız olarak çalışması, endüstriyel ortam şartlarında çalışabilmesi, paralel işlem yeteneğiyle kontrolün EMF kiti için kabul edilebilir bir sürede gerçekleştirilmesine olanak sağlanmıştır.

Gerçekleştirilen hibrid zeki kontrol ile EMF performansını etkileyen faktörlerin değişimine duyarlı ve EAS'ye sahip olması ile farklı alanlardaki endüstriyel sıvılarla yapılacak uygulamalarda başarılı olacağı düşünülen bir sistem elde edilmiştir.

9.2. Öneriler

EAS'ye, EMF performansını etkileyen GPK ve AH haricindeki diğer teknolojik parametrelerin ilave edilmesi ile hibrid zeki kontrolör daha da geliştirilebilir.

Bu çalışmada daha önceden tasarımı yapılan ve gerçekleştirilen Selenoit tip nüveye, küresel elemanlardan oluşan filtre matrisine sahip EMF kiti kullanılarak kontrolör deneyleri gerçekleştirilmiştir. Bu bağlamda tasarlanıp gerçekleştirilecek olan :

a) Selenoit tip haricinde gerçekleştirilecek diğer tip nüveye sahip EMF'ler üzerinde denemeler yapılarak elde edilen sonuçlara göre kontrolörde iyileştirmeler yapılabilir.
b) Küresel tip haricinde manyetik özellikli talaş ve çubuk şeklinde matris elemanları kullanılan EMF'ler üzerinde denemeler yapılarak elde edilecek sonuçlar doğrultusunda kontrolör daha da geliştirilebilir.
c) Farklı çaplarda filtre matrisi ve küreciklerden oluşan EMF yapıları ile denemeler yapılarak elde edilecek sonuçlar doğrultusunda kontrolör geliştirilebilir.

KAYNAKLAR

Abbasov, T., Herdem, S., and Köksal, M., 1999. Performance of High Gradient Magnetic Filters with Granular Matrix, *Separation Science and Technology*, 34(2), pp. 263-276.

Abbasov, T., Köksal, M. and Herdem, S., 1999, Theory of High-Gradient Magnetic Filter Performance, *IEEE Trans. on Magnetics*, 35(4), pp. 2128-2132.

Abbasov T., 2001, Theoretical interpretation of the filtration process in magnetized packed beds, *Elseiver Powder Technology*, vol. 115, 215–220.

Abbasov, T., 2002, Elektromanyetik Filtreleme İşlemleri (Teori, uygulama ve konstrüksiyon), *Seçkin Yayıncılık*, Ankara.

Abbasov, T., 2007, Magnetic filtration with magnetized granular beds: Basic principles and filter performance, *China Particuology*, 5, 71–83.

Abdul-wahab, O., 2006, Self Organızıng Fuzzy Controller For A Non-Lınear Tıme Varyıng System, *Journal of Engineering*, 3(13), 777-785.

Akçayol, B., 2000, Bulanık mantık [Online], *ceng.gazi.edu.tr/~akcayol/files/FSL5FLC.pdf*, [Ziyaret Tarihi: 17 Nisan 2013].

Alci, M., Karabiyik, A., and Karatepe, E., 2003, Adaptive fuzzy control of nonlinear systems and comparative analyses, *IJCI Proc. of international Conf. Signal Processing*, vol. 1, no. 2.

Alizade P., Abbasov T., Nayie A., 1997, Akışkan Ortamların Elektromanyetik Filtrelerle Temizlenmesi, *Çev-Kor*, Cilt 7, Sayı 25, Sayfa 17-18.

Altera, 2013, DE2-70 Development and Education Board [Online], Altera Terasic, *http://www.altera.com/education/univ/materials/boards/de2-70/unv-de2-70-board.html* [Ziyaret Tarihi: 15 Nisan 2013].

Altiver, 2013, Altivar 312 frequency inverter [Online], Schneider, *http://www.cia-productguides.org/catalogue/do/printProduct/id/copg_schneider_altivar_312/copg_schneider_altivar_312.pdf* [Ziyaret Tarihi: 15 Nisan 2013].

Ansett, M., Kreider, J.F., 1993, Application of Neural Networking Models to Predict Energy use, *ASHRAE Transactions: Research*, 99(1), pp. 505–517.

Anonim, 2013, RS-232 receiver [Online] , *http://www.fpga4fun.com/SerialInterface4.html*, [Ziyaret Tarihi: 15 Nisan 2013].

Arulmozhiyal R., 2012, Design and Implementation of Fuzzy PID controller for BLDC motor using FPGA, *Power Electronics, Drives and Energy Systems (PEDES), 2012 IEEE International Conference*, pp. 1-6.

Bose, B.K., 2001, Modern Power Electronics and AC Drives, *Prentice Hall*.

Buja, G.S., Todesco, F., 1994, Neural Network Implementation of A Fuzzy Logic Controller, *IEEE Transactions on Industrial Electronics*, 41, 6,663-667.

Chang, M., Yen, P., Yuan, T., 2006, Angle Control of a one-Dimension Pneumatic Muscle Armusing Self-Organizing Fuzzy Control, *IEEE International Conference on systems*, Taiwan.

Chen, C., Chen W., 1994, Fuzzy Controller Design by Using Neural Network Techniques, *IEEE Transactions on Fuzzy Systems*, 2, 3, 235-244.

Cybenko, G., Approximation by superpositions of a sigmoidal dunction. *Mathematics of Control, Signals, and Systems*, 1989(2): p. 303-314.

Deepak, K.T., 2013, Simple UART VHDL [Online] , Asic-World, http://www.asic-world.com/examples/vhdl/uart.html, [Ziyaret Tarihi: 17 Nisan 2013].

Ebner, N.A., 2007, Filter Capacity Predictions for the Capture of Magnetic Microparticles by High-Gradient Magnetic Separation, *IEEE Transactions on Magnetics*, 43(5), 1945-1949.

Elmas, Ç., 2003, Bulanık Mantık Denetleyiciler (Kuram, Uygulama, Sinirsel Bulanık Mantık), *Seçkin Yayınevi*, Ankara.

Erkaymaz, H., Çayıroğlu, İ, 2010, Bulanık mantık ve PIC kullanılarak bir klima sisteminin kontrolü, *Mehmet Akif Ersoy Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 2, 167-180.

Funahashi, K., On the Approximate Realization of Continuous Mappings by Neural Networks. *Neural Networks*, 1989(2): p. 183-192.

Gopala, K.A., Reddy, B.A., Phavani, B.D., 2010, Fuzzy PI and Integrating Type Fuzzy PID Controllers of Linear, Nonlinear and Time Delay Systems, *International Journal of Computer Applications*, 1(6), 43-49.

Gatt, E., Casha, O., Grech ve ark. 2012, FPGA-based autonomous parking of a car-like robot using Fuzzy Logic Control, *Electronics, Circuits and Systems (ICECS), 2012 19th IEEE International Conference*, pp 229-232.

Hagan MT, Demuth HB, Beale MH., 1997, Neural Network Design, *PWS Publishing*.

Haykın S., 1998, Neural Networks, A Comprehensive Foundation, Second Edition, *Pearson Prentice Hall*, Canada.

Heitmann, H.G., 1969, Iron oxides in boiler water removed magnetically, *Industrial water engineering*, 12, 31-33.

Herdem, S., Abbasov, T., and Köksal, M., 1999, Filtration Model of High Gradient Magnetic Filters with Granular Matrix, *Powder Technology*, 106(3), pp. 176-182.

Herdem, S., Abbasov, T., and Köksal, M., 1999, Modelling of Buildup Process of Particles in Pores of High Gradient Magnetic Filters by Using Fokker-Planck Equation, *J. Phys. D: Appl. Phys.*, 32(24), pp. 3146-3150.

Herdem, S., Köksal, M., and Abbasov, T., 2000, Model for Predicting Filtration Efficiency and Pressure Drop in Axial Magnetic Filters, *Separation Science and Technology*, 35(7), pp. 941-950.

Herdem, S., Köksal, M. and Abbasov, T., 2001, Simulation of Magnetic Filtration Process of Technological Liquids and Gases by a Nonlinear Electrical Circuit, *J. Phys. D: Appl. Phys.*, 34(2), pp. 238-242.

Herdem, S., 2005, Influence of the magnetic susceptibility of the particles on the magnetic filter performance, *Int. J. of Appl. Electromagnetics and Mechanics*, vol. 22, no. 3-4, pp. 189-197.

Herdem, S., Sarıtaş, İ., Özkan, İ.A., 2011, Endüstriyel sıvıların temizlenmesinde kullanılan elektromanyetik filtrelerin yapay zeka yöntemleriyle adaptif kontrolü ve kit tasarımı, *TÜBİTAK EEEAG Proje 109E037*, Ankara, 1-197.

Hornik, K., M. Stinchcombe, and H. White, Mulltilayer feedforward networks are universal approximators. *Neural Networks*, 1989(2): p. 359-366.

Jacobo, A., Alfanso, L., Andres, N., Carlos, P., Jorge, M., Jesus, D., Oscar, L., 2006, FPGA Implementation of a Fuzzy Controller for Automobile DC-DC Converters, *Field Programmable Technology (FPT 2006)*, Bangkok, pp. 237-240.

Jantzen, J., 1998, The self-organizing fuzzy controller, *Tech. Report no. 98*, H 869.

Karna K. N., Breen D. M., 1989, An Artificial Neural Networks Tutorial: Basics, *Neural Networks*, Vol.1, No:1, pp: 4 -23.

Kıyak, E., Kahvecioğlu, A., 2003, Bulanık Mantık ve Uçuş kontrol Problemine Uygulanması, *Havacılık ve Uzay Teknolojileri Dergisi*, 1(2), 63-72.

Köksal, M., Abbasov, T. and Herdem, S., 2003, Mathematical Modeling of the Magnetic Filtration Processes, *Int. J. of Appl. Electromagnetics and Mechanics*, vol. 18(4), 227-234.

Kwong, W. A., Passsino, K.M. , Lauknonen, E. G. and Yurkovich, S., 1995, Expert supervision of fuzzy learning systems for fault tolerant aircraft control, *Proc. IEEE Spec.. Issue Fuzzy Logic Eng. Applicat.*, vol. 83, pp. 466-483.

Layne, J., Passino, K and Yurkovich, S., 1992, Fuzzy learning control for antiskid braking systems, *IEEE Conf. Decision Contr.*, Tucson, AZ, pp. 2523-2528.

Lee, J., 1993, On Methods for Improving Performance of PI-Type Fuzzy Logic Controllers, *IEEE Transactions on Fuzzy Systems*, 1(4), 298-301.

Lennon, W. and Passino, K., 1995, Intelligent control for brake systems, *IEEE Int. Symp. Intell. Contr.*, pp. 499-504.

Liberman, S.J., Kelland, D.R., 1984, Magnetic filtration of aqueous suspensions of submicron reactor corrosion products, *IEEE Transactions on Magnetics*, 20(5), 1195-1197.

Loan, S.A., Murshid, A.M., 2011, A novel fuzzy inference processor using trapezoidal-shaped membership function, *Open Systems (ICOS), 2011 IEEE Conference*, pp 220-224.

Mar, J. And Lin, F., 2001, An ANFIS Controller for the Car-Following Collision Prevention System, *IEEE Transactions On Vehicular Technology*, 50(1), 1106-1113.

Mendi F., Boran K., Kulekci M.K., 2002, Fuzzy Controlled Central Heating System,*Int. J. Energy Res.*, Vol. 26: pp. 1313–1322.

Milhous, 2013, Learn about the EMF Process [Online], Milhous Company, http://www.milhous.com/electromagnetic-filtration/ [Ziyaret Tarihi: 14 Nisan 2013].

Mudi, R.K., Pal, N.R. , 2000, A self-tuning fuzzy PI controller, *Fuzzy Sets and Systems*, 115, 327-338.

Mukhtiar, S., Ambrish, C., 2013, Real-Time Implementation of ANFIS Control for Renewable Interfacing Inverter in 3P4W Distribution Network, *IEEE Transactions on Industrial Electronics*, 60(1), pp. 121-128

Munakata T. 1998, Fundamentals of The New Artifical Intelligence: Beyond Traditional Paradigms, Springer-Verlag, New York, USA.

Nabiyev, V.V., 2005, Yapay Zeka Problemler – Yöntemler – Algoritma, *Seçkin Yayınevi*, Ankara.

Nilsson, N. J., 1998, Artificial Intelligence :A New Synthesis, *Morgan Kaufmann Publishers*.

Okada, H., Kudo, Y., Nakazawa, H., Chiba, A., Mitsuhashi, K., Ohara, T., Wada, H., 2004, Removal System of Arsenic From Geothermal Waterby High Gradient Magnetic Separation-HGMS Reciprocal Filter, *IEEE Transactions on Applied Superconductivity*, 14(2), 1576-1579.

Özkan, İ.A., Sarıtaş, İ., Herdem, S., 2011, Manyetik Filtreler için FPGA Tabanlı Bulanık Kontrolör Tasarımı, *Teknik-Online Dergisi*, 10(3), 271-284.

Ozkan, I.A., Saritas, I., Herdem, S., 2012, The control of magnetic filters by FPGA based fuzzy controller, *Energy Education Science and Technology Part A*, 29(2), pp. 1093-1102.

Öztemel, E., 2003, Yapay Sinir Ağları, *Papatya Yayıncılık*, İstanbul.

Pal, A.K., Mudi, R.K., 2008, Self-Tuning Fuzzy PI Controller and its Application to HVAC Systems, *International Journal of Computational Cognition*, 6(1), pp. 25-30.

Passino , K.M., Yurkovich , S., 1998, *Fuzzy Control*, Addison-Wesley.

Procyk, T. and Mamdani, E., 1979, A liguistic self-organizing process controller, *Automatica*, vol. 15, no. 1, pp. 15-30, 1979.

Rajani, K.M., Nikhil, R.P., 2000, A self-tuning fuzzy PI controller, *Fuzzy Sets and Systems*, 115(2), pp. 327-338.

Rosenblatt, R., 1962, Principles of Neurodynamics, *Spartan Books*, New York, U.S.A.

Russell, S.J., and Norvig, P., 1995, Artificial Intelligence : A Modern Approach, Prentice-Hall.

Sarıkaya, M., Abbasov, T. and Erdemoglu, M., 2006, Some Aspects of Magnetic Filtration Theory for Removal of Fine Particles from Aqueous, *Journal of Dispersion Science and Technology*, 27, 193–198.

Sarıtaş, İ., 2008, Elektromanyetik Filtre Tasarımı ve Yapay Zeka Yöntemleriyle Adaptif Kontrolü, Doktora Tezi, *Selçuk Üniversitesi Fen Bilimleri Enstitüsü*, Konya.

Sarıtaş, İ., Ozkan, I.A. and Herdem, S., 2010, The effects of fuzzy control of magnetic flux on magnetic filter performance and energy consumption, *Expert Systems with Applications*, Vol.37(12), 7542-7551.

Sarıtaş, İ, Ozkan, I.A., Herdem, S., 2012, An Artificial Neural Network Model for Magnetic Filter Performance, *1st Taibah University International Conference on Computing and Information Technology (ICCIT 2012)*, Al-Madinah, pp. 50-55.

SerialNg, 2013, Fully Serial Communication with Delphi and Windows [Online], DomIS Integrated Services, *http://www.domis.de/cms/index.php?module=ContentExpress&func=display&bid=24&btitle=Content_Menu&mid=3&ceid=7* [Ziyaret Tarihi: 27 Nisan 2013].

SU8001, 2013, SU8001 Flow Sensors [Online], IFM Elector Inc., *http://www.ifm.com/products/us/ds/SU8001.htm* [Ziyaret Tarihi: 15 Nisan 2013].

Vasantha, S.P., Kanagasabapathy, P., Satlhish, A., 2005, Digital Fuzzy Logic Controller using VHDL, *IEEE Indicon 2005 Conference*, Chennai, pp. 463-466.

Wang, L.X., 1993, Stable Adaptive Fuzzy Control of Nonlinear Systems, *IEEE Transactions on Fuzzy Systems*, 1(2), 46-155.

Wang, L.X., 1997, A Course in Fuzzy Systems and Control, *Pearson Education POD*, ISBN: 0135408822.

Watson, J.H.P., 1973, Magnetic filtration, *J. Appl. Phys.*, 44(9), 4209-4213.

Yalcintas M., Akkurt S., 2005, Artificial Neural Networks Applications in Building Energy Predictions and A Case Study for Tropical Climates, *Int. J. Energy Res.*, Vol. 29, pp. 891–901.

Yen, J., Langari, R., Zadeh, L.A., 1995, Industrial Applications of Fuzzy Logic and Intelligent Systems, *IEEE Press*, New York.

*Z*adeh, L.A., 1965. Fuzzy Sets, *Elsevier Information and Control*, 8.3: 338-353.

Zezulka V., Straka P., and Mucha P., 2004, A Magnetic Filter With Permanent Magnets on The Basis of Rare Earths, *Journal of Magnetism and Magnetic Materials*, vol. 268, pp. 219–226.

EKLER

EK-1 Basitleştirilmiş UART VHDL yapısı

```
library ieee;
use ieee.std_logic_1164.all;

entity uart_sur is
   port(
      clk      : in   std_logic;
      data_in  : in   std_logic;
      veri1 : in std_logic_vector(11 downto 0);
      veri2 : in std_logic_vector(11 downto 0);
           …..
      veri8 : in std_logic_vector(19 downto 0);
      data_out     : out  std_logic_vector(7 downto 0)
   );

end entity;

architecture rtl of uart_sur is
   type state_type is (s0,s1,s2,s3,s4,s5,s6,s7,s8,s9,s10,s11,s12,s13,s14,s15,s16);
   signal state   : state_type;
begin
   process (clk)
   begin
      if (rising_edge(clk)) then
         case state is
            when s0=>
               if data_in = '1' then
                  state <= s1;
               else
                  state <= s0;
               end if;
            when s1=>
               if data_in = '1' then
                  state <= s2;
               else
                  state <= s1;
               end if;
                     ….
            when s16 =>
               if data_in = '1' then
                  state <= s0;
               else
                  state <= s16;
               end if;
         end case;
      end if;
   end process;
   process (state,veri1,veri2,veri3,veri4,veri5)
   begin
      case state is
         when s0 =>
            data_out <= "01011010";  --baslangic verisi
         when s1 =>
            data_out <= "1" & veri1(11 downto 5);  --bilgisayara aktarimi yapılacak veriler
           …..
         when s16 =>
```

```
            data_out <= "1" & veri8(0)&"111010" ;    --bitis verisi
        end case;
    end process;
end rtl;
-------------------------------------------------------

library ieee;
    use ieee.std_logic_1164.all;
    use ieee.std_logic_unsigned.all;
entity uart_tx is
    port (
        reset       :in  std_logic;
        txclk       :in  std_logic;
        ld_tx_data  :in  std_logic;
        tx_data     :in  std_logic_vector (7 downto 0);
        tx_enable   :in  std_logic;
        tx_out      :out std_logic;
        tx_empty    :out std_logic
        );
end entity;
architecture rtl of uart_tx is
    signal tx_reg        :std_logic_vector (7 downto 0);
    signal tx_over_run   :std_logic;
    signal tx_cnt        :std_logic_vector (3 downto 0);
    signal tx_is_empty   :std_logic;
begin
    process (txclk, reset) begin
        if (reset = '1') then
            tx_reg       <= (others=>'0');
            tx_is_empty  <= '1';
            tx_over_run  <= '0';
            tx_out       <= '1';
            tx_cnt       <= (others=>'0');
        elsif (rising_edge(txclk)) then
            if (ld_tx_data = '1') then
                if (tx_is_empty = '0') then
                    tx_over_run <= '0';
                else
                    tx_reg   <= tx_data;
                    tx_is_empty <= '0';
                end if;
            end if;
            if (tx_enable = '1' and tx_is_empty = '0') then
                tx_cnt <= tx_cnt + 1;
                if (tx_cnt = 0) then
                    tx_out <= '0';
                end if;
                if (tx_cnt > 0 and tx_cnt < 9) then
                    tx_out <= tx_reg(conv_integer(tx_cnt) -1);
                end if;
                if (tx_cnt = 9) then
                    tx_out <= '1';
                    tx_cnt <= X"0";
                    tx_is_empty <= '1';
                end if;
            end if;
            if (tx_enable = '0') then
                tx_cnt <= X"0";
            end if;
        end if;
```

```
   end process;
   tx_empty <= tx_is_empty;
end architecture;
-----------------------------------------------
library ieee;
use ieee.std_logic_1164.all;

entity uart_verial is
   port(
      clk        : in    std_logic;
      rx_d       : in    std_logic;
      veri_rx    : in    std_logic_vector(7 downto 0);
      veri_out1  : out   std_logic_vector(11 downto 0);
      veri_out2  : out   std_logic_vector(11 downto 0);
      veri_out3  : out   std_logic_vector(11 downto 0);
      veri_out4  : out   std_logic_vector(23 downto 0)
   );
end entity;

architecture rtl of verial is
   type state_type is (s0,s1,s2,s3,s4,s5,s6,s7,s8,s9,s10);
   signal state   : state_type;
   signal ilkatama : std_logic:='0';
begin
   process (clk,rx_d,veri_rx)
   begin
      if (rising_edge(clk)) then
         case state is
            when s0=>
               if (rx_d='1' AND veri_rx = "01100001") then   --baslangic verisini kontrol et
                     state <= s1;
                     ilkatama<='1';
                  else
                     state <= s0;
                  end if;
             when s1=>
               if (rx_d='1' AND veri_rx(7) = '1') then
                  state <= s2;
               else
                  state <= s1;
               end if;
            ……..
             when s10=>
               if (rx_d='1' AND veri_rx = "00100000") then  --bitis verisini kontrol et
                  state <= s0;
               else
                  state <= s10;
               end if;
         end case;
      end if;
   end process;
process (state)
   variable ara_sonuc1 : std_logic_vector(11 downto 0);
   variable ara_sonuc2 : std_logic_vector(11 downto 0);
   variable ara_sonuc3 : std_logic_vector(11 downto 0);
   variable ara_sonuc4 : std_logic_vector(23 downto 0);
   begin
      case state is
         when s1 =>
            ara_sonuc1(11 downto 5) := veri_rx(6 downto 0); --bilgisayar üzerinden verilerin alimi
```

```
        when s2 =>
          ara_sonuc1(4 downto 0) := veri_rx(6 downto 2);
          ara_sonuc2(11 downto 10) := veri_rx(1 downto 0);
         ……
        when s10 =>
          if(ilkatama='1') then
            veri_out1<= ara_sonuc1;
            veri_out2<= ara_sonuc2;
            veri_out3<= ara_sonuc3;
            veri_out4<= ara_sonuc4;
          else
            veri_out1<= "111110100000"; --sistemin ön tanımlı verileri
            veri_out2<= "111110100000";
            veri_out3<= "000001100000";
            veri_out4<= "000000000100111000100000";
          end if;
      end case;
    end process;
end rtl;
```

EK-2 Basitleştirilmiş Bulanık Fonksiyon Paketi

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;
use ieee.std_logic_signed.all;
use ieee.fixed_float_types.all;
use ieee.fixed_pkg.all;

package fonksiyonlar is
-- giris degerleri icin
 constant tam :integer:=12; -- tam sayili kismin uzunlugu
 constant noktali :integer:=-7; -- noktali sayi kisminin uzunlugu
 -- dilsel degerler icin
 constant tam1 :integer:=7; -- tam sayili kismin uzunlugu
 constant noktali1 :integer:=-2; -- noktali sayi kisminin uzunlugu
  -- giris degerleri icin
 subtype fixsayi is sfixed (12 downto -7);
 subtype bfixsayi is sfixed (14 downto -7);
 subtype fixsayi3 is sfixed (14 downto -1);
 subtype fixsayi4 is ufixed (14 downto -1);
  -- dilsel ifadeler icin
 subtype fixsayi2 is sfixed (7 downto -2);
 constant triangular :integer:=4; -- dilsel ifade ucgen veya yamuk
  type t_node is array ( 1 to triangular) of fixsayi2;
 constant e_dilsel_sayi :integer:=7;  -- hata dilsel ifade sayisi
 constant ed_dilsel_sayi :integer:=7;  -- hata degisimi dilsel ifade sayisi
type e_dilsel is array ( 1 to e_dilsel_sayi) of t_node; -- hata icin dizi
type ed_dilsel is array ( 1 to ed_dilsel_sayi) of t_node; -- hata degisimi icin dizi
constant umin: sfixed:=to_sfixed(0,tam,noktali);
constant umax: sfixed:=to_sfixed(1,tam,noktali);
constant yuz : fixsayi := to_sfixed(100,tam,noktali);
constant bir : fixsayi := to_sfixed(1,tam,noktali);
constant bos : fixsayi := to_sfixed(0,tam,noktali);
-- dilsel sonuclar hata ve hata degisimi icin
type e_d_sonuc is array ( 1 to e_dilsel_sayi) of fixsayi;
type ed_d_sonuc is array ( 1 to ed_dilsel_sayi) of fixsayi;
type cikarim_d_sonuc is array ( 1 to 7) of fixsayi;
type son_d_sonuc is array ( 1 to e_dilsel_sayi*ed_dilsel_sayi) of fixsayi;
function durulastirma (a1:cikarim_d_sonuc) return fixsayi;
function carpma_h(a: fixsayi; b:fixsayi)return fixsayi;
function topla_h (a:fixsayi; b:fixsayi) return fixsayi;
function cikar_h(a: fixsayi; b:fixsayi)return fixsayi;
function bolme_h(a: fixsayi; b:fixsayi)return fixsayi;
function cikarim( a1:ed_d_sonuc; b1 : e_d_sonuc) return cikarim_d_sonuc;
function max_islem (a1:fixsayi;b1:fixsayi;c1:fixsayi;d1:fixsayi;e1:fixsayi;f1:fixsayi;g1:fixsayi) return
fixsayi ;
 function bulaniklastirma( a1: fixsayi2;b1 : fixsayi2;c1 : fixsayi2;d1 : fixsayi2;giris : fixsayi) return
fixsayi;
function performans_h(a: fixsayi; b:fixsayi)return fixsayi;
end fonksiyonlar;
package body fonksiyonlar is
   function carpma_h(a: fixsayi; b:fixsayi)return fixsayi  is
      variable pdt:   fixsayi;
   begin
      pdt:=resize(a*b,tam,noktali);
      return  pdt ;
   end carpma_h;
function topla_h (a:fixsayi;b:fixsayi) return fixsayi is
      variable pdt : fixsayi;
```

```
   begin
      pdt:=resize(a+b,tam,noktali);
      return  pdt ;
   end topla_h;
function cikar_h(a: fixsayi; b:fixsayi)return fixsayi  is
      variable pdt:   fixsayi;
   begin
      pdt:=resize(a-b,tam,noktali);
      return  pdt ;
   end cikar_h;
function bolme_h(a: fixsayi; b:fixsayi)return fixsayi  is
      variable pdt:   fixsayi;
   begin
      pdt:=resize(a/b,tam,noktali);
      return  pdt ;
   end bolme_h;
function max_islem (a1:fixsayi;b1:fixsayi;c1:fixsayi;d1:fixsayi;e1:fixsayi;f1:fixsayi;g1:fixsayi) return
fixsayi is
     variable max_sonuc: fixsayi;
   begin
      if (a1>=b1 AND a1>=c1 AND a1>=d1 AND  a1>=e1 AND a1>=f1 AND a1>=g1) then
max_sonuc:=a1;
      elsif (b1>=a1 AND b1>=c1 AND b1>=d1 AND  b1>=e1 AND b1>=f1 AND b1>=g1) then
max_sonuc:=b1;
      elsif (c1>=a1 AND c1>=b1 AND c1>=d1 AND  c1>=e1 AND c1>=f1 AND c1>=g1) then
max_sonuc:=c1;
      elsif (d1>=a1 AND d1>=c1 AND d1>=b1 AND  d1>=e1 AND d1>=f1 AND d1>=g1) then
max_sonuc:=d1;
      elsif (e1>=a1 AND e1>=c1 AND e1>=d1 AND  e1>=b1 AND e1>=f1 AND e1>=g1) then
max_sonuc:=e1;
      elsif (f1>=a1 AND f1>=c1 AND f1>=d1 AND  f1>=e1 AND f1>=b1 AND f1>=g1) then
max_sonuc:=f1;
      else max_sonuc:=g1;
      end if;
      return max_sonuc;
end max_islem;
function performans_h(a: fixsayi; b:fixsayi)return fixsayi  is   --EMF performans hesabı
      variable d1,pdt,pdt_o:   fixsayi;
      begin
          if b=bos then pdt:=(others=>'0');
            else
         d1:=resize(a/b,tam,noktali);
         pdt_o:=cikar_h(bir,d1);
         pdt:=carpma_h(yuz,pdt_o);
      end if;
   return  pdt ;
   end performans_h;
function bulaniklastirma( a1: fixsayi2;b1 : fixsayi2;c1 : fixsayi2;d1 : fixsayi2;giris : fixsayi) return
fixsayi  is
      variable g_c1,g_c2,aa1,ab1,ac1,ad1 : fixsayi;
      variable sonuc_b :  fixsayi;
   begin
      aa1:=resize(a1,tam,noktali);
      ab1:=resize(b1,tam,noktali);
      ac1:=resize(c1,tam,noktali);
      ad1:=resize(d1,tam,noktali);
      if (giris<=a1) then
         sonuc_b:=umin;
      elsif (giris>a1 and giris<b1) then
         g_c1:=resize(cikar_h(giris,aa1),tam,noktali);
```

```
            g_c2:=resize(cikar_h(ab1,aa1),tam,noktali);
            sonuc_b:=resize(bolme_h(g_c1,g_c2),tam,noktali);
        elsif (giris>=b1 and giris<=c1) then
            sonuc_b:=umax;
        elsif (giris>c1 and giris<d1) then
            g_c1:=resize(cikar_h(ad1,giris),tam,noktali);
            g_c2:=resize(cikar_h(ad1,ac1),tam,noktali);
            sonuc_b:=resize(bolme_h(g_c1,g_c2),tam,noktali);
        else
            sonuc_b:=umin;
        end if;
        return sonuc_b;
    end bulaniklastirma;
function cikarim( a1:ed_d_sonuc;b1 : e_d_sonuc) return cikarim_d_sonuc  is
        variable kucuk: son_d_sonuc;
        variable maxsonuc: cikarim_d_sonuc;
        variable i,j,gecici: integer;
    begin
        gecici:=0;
        for i in 1 to ed_dilsel_sayi loop
            for j in 1 to e_dilsel_sayi loop
                gecici:=gecici+1;
                kucuk(gecici) :=minimum (a1(i),b1(j));
            end loop;
        end loop;
       -- kural tablosuna gore maxsonuc ifadeleri
        maxsonuc(1) := max_islem(kucuk(1),kucuk(2),…,kucuk(22),bos);
        maxsonuc(2) := max_islem(kucuk(4),kucuk(9),kucuk(10),kucuk(11),…,kucuk(29));
        maxsonuc(3) := max_islem(kucuk(5),…..,kucuk(36),bos,bos,bos,bos,bos));
        maxsonuc(4) := max_islem(kucuk(7),…..,kucuk(37),kucuk(43));
        maxsonuc(5) := max_islem(kucuk(14),….kucuk(44),kucuk(45),bos,bos,bos,bos,bos));
        maxsonuc(6) := max_islem(kucuk(46),kucuk(41),…..,kucuk(27),kucuk(21));
        maxsonuc(7) := max_islem(kucuk(28),….,kucuk(47),bos);
        return maxsonuc;
    end cikarim;
function durulastirma (a1:cikarim_d_sonuc) return fixsayi is
        variable duru_sonuc: fixsayi;
        variable bolen,bolunen: bfixsayi;
    begin
        bolunen:=resize((carpma_h(a1(1),e_sonuc(1))
            +carpma_h(a1(2),e_sonuc(2))+
            carpma_h(a1(3),e_sonuc(3))+
            carpma_h(a1(4),e_sonuc(5))+
            carpma_h(a1(6),e_sonuc(6))+
            carpma_h(a1(7),e_sonuc(7))),14,-7);
        bolen:=resize(a1(1)+a1(2)+a1(3)+a1(4)+a1(5)+a1(6)+a1(7),14,-7);
        if bolen=bos then duru_sonuc:=(others=>'0');
        else duru_sonuc:=resize(bolunen*reciprocal(bolen),tam,noktali);
        end if;
        return duru_sonuc;
    end durulastirma;
end package body fonksiyonlar;
```

EK-3 Basitleştirilmiş ANFIS Kod Örneği

```
library ieee;
library ieee_proposed;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_signed.all;
use ieee.fixed_pkg.all;
use work.fonksiyonlar.all;

entity anfis is
  port(
    gpk : in std_logic_vector ( 11 downto 0); --giren parçacık konsantrasyonu
    cpk : in std_logic_vector ( 11 downto 0); --çıkan parçacık konsantrasyonu
    akh : in std_logic_vector ( 11 downto 0); --akış hızı
    ref_p : in std_logic_vector( 11 downto 0); --referans performans
    clk : in std_logic;
    clk2 : in std_logic;
    clk3 : in std_logic;
    adap_enable : in std_logic;
    u: out fixsayi;
     );
end anfis;

architecture davranis of anfis is
type int_array is array(0 to 36) of integer ;
type states is (
init,bulaniklastirma_baslangic,bulaniklastirma_dongu,gpk_dilsel_ifade_hesaplama,ah_dilsel_ifade_hesaplama,bulaniklastirma_dongu_bitis,katman3_on_deger,katman3_baslangic,katman3_ilk_dongu,katman3_ikinci_dongu,katman3_hesaplama,aktif_uyeleri_belirle,aktif_uyeleri_belirle_dizi,aktif_uyeleri_belirle_dizi_atama,katman3_ikinci_dongu_bitis,katman3_ilk_dongu_bitis,katman4_toplam_baslangic,katman4_toplam,katman4_toplam_dongu_bitis,amac_fonksiyon_uyarlama_baslangic,amac_fonksiyon_uyarlama_dongu,amac_fonksiyon_uyarlama_bir,amac_fonksiyon_uyarlama_iki,amac_fonksiyon_uyarlama_uc,amac_fonksiyon_uyarlama,amac_fonksiyon_uyarlama_dongu_bitis,onsart_ortak_carpim,onsart_ortak_carpim_2,onsart_ortak_carpim_3,on_sart_uyarlama_baslangic,on_sart_uyarlama_dongu,bk_tablosu_indis1,bk_tablosu_indis2,on_sart_uyarlama_amacfnk_carpim,on_sart_uyarlama_pay,on_sart_uyarlama_arahesaplama1,on_sart_uyarlama_arahesaplama2,on_sart_uyarlama_arahesaplama3,on_sart_uyarlama_isaretfonksiyonu,on_sart_uyarlama_yeni_a_parametresi,on_sart_uyarlama_dongu_bitis);
type states2 is (init,
bulaniklastirma_dongu_baslangic,bulaniklastirma_dongu,gpk_dilsel_ifade_hesaplama,ah_dilsel_ifade_hesaplama,bulaniklastirma_dongu_bitis,katman3_hesaplama_baslangic,katman3_ilk_dongu,katman3_ikinci_dongu,katman3_hesaplama,katman3_ikinci_dongu_bitis,katman3_ilk_dongu_bitis,katman4toplam_baslangic,katman4toplam_dongu_baslangic,katman4toplam_dongu,katman4toplam_hesap,katman4toplam_dongu_bitis,katman4_baslangic,katman4_dongu,katman4_hesaplama,katman4_dongu_bitis,katman5_baslangic,katman5_hesap,katman5_dongu_bitis,cikis_hesaplama_baslangic,cikis_hesaplama_dongu,cikis_hesaplama,cikis_hesaplama_dongu_bitis,sonuc);
signal durum : states2 := init;
signal state : states := init;
signal error,a_katman4toplam : fixsayi;
signal ck3,e,de,u_deger,on_carpim,on_carpim1,on_carpim2: fixsayi;
signal gpk_dilsel_sonuc,a_gpk_dilsel_sonuc : gpk_d_sonuc ;
signal ah_dilsel_sonuc,a_ah_dilsel_sonuc : ed_d_sonuc ;
signal a_katman3,a_katman4,katman3,katman4,katman5 : k3;
signal a_yeni,a_gpm,a_ah,g1_f,g2_f,g3_f,g4_f,g5_f,g6_f,g4_f1,g5_f1,g6_f1,g7_f,g7_f1,g8_f,g8_f1
,g9_f,g9_f1 ,g10_f,g10_f1
,g5_fa,g5_fb,g5_fc,g5_f1a,g5_f1b,g5_f1c,g6_fa,g6_f1a,g7_fa,g7_f1a,g8_fa,g8_f1a,g9_fa,g9_f1a,g6_fu,g6_f1u,g8_f1u,g8_fu :adaptif_deger;
signal x,y,k,z,j,i:integer;
signal katman4toplam,cikis: fixsayi:=(others => '0');
```

```
signal a_yer,gec: integer;
signal a_yeri,u_deger2,u_deger3: int_array;
signal guncelle,istek,fuzzy_enable: std_logic;
signal gpk_dilsel_deger: gpk_dilsel:=(  -- gpk dilseldeger orta nokta
                (to_sfixed(-60,tam,noktali), to_sfixed(30,tam,noktali), to_sfixed(60,tam,noktali),
to_sfixed(90,tam,noktali),to_sfixed(120,tam,noktali),umin));
signal ah_dilsel_deger: ah_dilsel:=( -- ah dilseldeger orta nokta
                (to_sfixed(-400,tam,noktali), to_sfixed(200,tam,noktali),
to_sfixed(400,tam,noktali),to_sfixed(600,tam,noktali), to_sfixed(800,tam,noktali),umin));
signal gpk_dilsel_deger_u: gpk_dilsel:=( -- gpk dilseldeger taban uzunluk
            (to_sfixed(120,tam,noktali), to_sfixed(60,tam,noktali),to_sfixed(60,tam,noktali),
to_sfixed(60,tam,noktali),to_sfixed(60,tam,noktali),umin));
signal ah_dilsel_deger_u: ah_dilsel:=( -- ah dilseldeger taban uzunluk
                (to_sfixed(800,tam,noktali), to_sfixed(400,tam,noktali), to_sfixed(400,tam,noktali),
to_sfixed(400,tam,noktali),to_sfixed(400,tam,noktali),umin));
signal o_b: k3:=(-- amac fonksiyon cikisi
            (to_sfixed(1,tam,noktali), to_sfixed(1,tam,noktali), to_sfixed(1,tam,noktali),
to_sfixed(1,tam1,noktali), to_sfixed(1,tam1,noktali),
             to_sfixed(1,tam,noktali), to_sfixed(100,tam,noktali), to_sfixed(200,tam,noktali),
to_sfixed(300,tam,noktali),to_sfixed(400,tam1,noktali),
             to_sfixed(1,tam,noktali), to_sfixed(200,tam,noktali), to_sfixed(300,tam,noktali),
to_sfixed(400,tam,noktali),to_sfixed(500,tam1,noktali),
             to_sfixed(1,tam,noktali), to_sfixed(300,tam,noktali), to_sfixed(400,tam,noktali),
to_sfixed(500,tam,noktali),to_sfixed(600,tam1,noktali),
             to_sfixed(1,tam,noktali), to_sfixed(400,tam,noktali), to_sfixed(500,tam,noktali),
to_sfixed(600,tam,noktali), to_sfixed(700,tam1,noktali),umin));
begin
-----------------------------------
Process(clk)
-- anfis  fuzzy katman --
begin
   if clk'event and clk='1'  then
      case durum is
         when init =>
            -- sinyaller ilk kullanima hazirlaniyor
            a_gpm<=(umax,umax,umax,umax);
            a_ah<=(umax,umax,umax,umax);
            katman3(17)<=umin;
            katman4(17)<=umin;
            katman5(17)<=umin;
            durum <= bulaniklastirma_dongu_baslangic;
         when bulaniklastirma_dongu_baslangic =>
            x<=0;
            durum <= bulaniklastirma_dongu;
         when bulaniklastirma_dongu =>
            x<=x+1;
            durum <= gpk_dilsel_ifade_hesaplama;
         when gpk_dilsel_ifade_hesaplama =>
               gpk_dilsel_sonuc (x)<=bulaniklastirma(gpk_dilsel_deger(x),gpk_dilsel_deger_u(x),gpk);
               durum <= ah_dilsel_ifade_hesaplama;
         when ah_dilsel_ifade_hesaplama =>
               ah_dilsel_sonuc (x)<=bulaniklastirma(ah_dilsel_deger(x),ah_dilsel_deger_u(x),ah);
               durum <= bulaniklastirma_dongu_bitis;
         when bulaniklastirma_dongu_bitis =>
            if x=5 then
               x<=0;
               durum<=katman3_hesaplama_baslangic;
            else
               durum<=bulaniklastirma_dongu;
            end if;
```

```
            when katman3_hesaplama_baslangic =>
               x<=0;
               y<=0;
               durum <= katman3_ilk_dongu;
            when katman3_ilk_dongu =>
               x<=x+1;
               durum<=katman3_ikinci_dongu;
            when katman3_ikinci_dongu =>
               y<=y+1;
               durum<=katman3_hesaplama;
            when katman3_hesaplama =>
               katman3((x-1) *gpk_dilsel_sayi+y) <=
resize(carpma_h(gpk_dilsel_sonuc(y),ah_dilsel_sonuc(x)),tam,noktali);
               durum<=katman3_ikinci_dongu_bitis;
            when katman3_ikinci_dongu_bitis =>
              if y=5 then
                  y<=0;
                  durum<=katman3_ilk_dongu_bitis;
            else
                  durum<=katman3_ikinci_dongu;
            end if;
            when katman3_ilk_dongu_bitis =>
               if x=5 then
                  x<=0;
                  durum<=katman4toplam_baslangic;
               else
                  durum<=katman3_ilk_dongu;
               end if;
            when katman4toplam_baslangic =>
               katman4toplam<=umin;
               durum<=katman4toplam_dongu_baslangic;
            when katman4toplam_dongu_baslangic =>
               x<=0;
               durum<=katman4toplam_dongu;
            when katman4toplam_dongu =>
               x<=x+1;
               durum<=katman4toplam_hesap;
            when katman4toplam_hesap =>
               katman4toplam<=topla_h(katman4toplam,katman3(x));
               durum<=katman4toplam_dongu_bitis;
            when katman4toplam_dongu_bitis =>
               if x>=25 then
                  durum<=katman4_baslangic;
               else
                  durum<=katman4toplam_dongu;
               end if;
            when katman4_baslangic =>
               x<=0;
               durum<=katman4_dongu;
            when katman4_dongu =>
               x<=x+1;
               durum<=katman4_hesaplama;
            when katman4_hesaplama =>
               katman4(x)<=bolme_h(katman3(x),katman4toplam);
               durum<=katman4_dongu_bitis;
            when katman4_dongu_bitis =>
               if x=25 then
                  x<=0;
                  durum<=katman5_baslangic;
               else
```

```
                durum<=katman4_dongu;
            end if;
        when katman5_baslangic =>
            x<=0;
            durum<=katman5_dongu;
        when katman5_dongu =>
            x<=x+1;
            durum<=katman5_hesap;
        when katman5_hesap =>
            katman5(x)<=carpma_h(katman4(x),o_b(x));
            durum<=katman5_dongu_bitis;
        when katman5_dongu_bitis =>
            if x=25 then
                x<=0;
                durum<=cikis_hesaplama_baslangic;
            else
                durum<=katman5_dongu;
            end if;
        when cikis_hesaplama_baslangic =>
            cikis<=umin;
            x<=0;
            durum<=cikis_hesaplama_dongu;
        when cikis_hesaplama_dongu =>
            x<=x+1;
            durum<=cikis_hesaplama;
        when cikis_hesaplama =>
            cikis<=topla_h(cikis,katman5(x));
            durum<=cikis_hesaplama_dongu_bitis;
        when cikis_hesaplama_dongu_bitis =>
            if x=25 then
                x<=0;
                durum<=sonuc;
            else
                durum<=cikis_hesaplama_dongu;
            end if;
        when sonuc =>
            u_deger<=cikis;
            durum<=bulaniklastirma_dongu_baslangic;
        when others=>
            durum <= init;
        end case;
end if;
end process;

Process(clk2)
-- geri yayılım algoritması ile eğitimi -----
begin
    if clk2'event and clk2='1' and adap_enable='1' then
        case state is
            when init =>
                a_yeni<=(umin,umin,umin,umin);
                a_katman3(17)<=umin;
                a_katman4(17)<=umin;
                state <= bulaniklastirma_baslangic;
            when bulaniklastirma_baslangic =>
                j<=0;
                state <= bulaniklastirma_dongu;
            when bulaniklastirma_dongu =>
                j<=j+1;
                state <= gpk_dilsel_ifade_hesaplama;
```

```
            when gpk_dilsel_ifade_hesaplama =>
               a_gpk_dilsel_sonuc (j)<=bulaniklastirma(gpk_dilsel_deger(j),gpk_dilsel_deger_u(j),gpk);
               state <= ah_dilsel_ifade_hesaplama;
            when ah_dilsel_ifade_hesaplama =>
               a_ah_dilsel_sonuc (j)<=bulaniklastirma(ah_dilsel_deger(j),ah_dilsel_deger_u(j),ah);
               state <= bulaniklastirma_dongu_bitis;
            when bulaniklastirma_dongu_bitis =>
               if j=4 then
                  j<=0;
                  state<=katman3_on_deger;
               else
                  state<=bulaniklastirma_dongu;
               end if;
            when katman3_on_deger =>
               a_yer<=0;
               a_hata_dilsel_sonuc (5)<=umin;
               a_hatadegisimi_dilsel_sonuc (5)<=umin;
               state <= katman3_baslangic;
            when katman3_baslangic =>
               j<=0;
               k<=0;
               state<=katman3_ilk_dongu;
            when katman3_ilk_dongu =>
               j<=j+1;
               state<=katman3_ikinci_dongu;
            when katman3_ikinci_dongu =>
               k<=k+1;
               state<=katman3_hesaplama;
            when katman3_hesaplama =>
               a_katman3((j-
1)*4+k)<=resize(carpma_h(a_gpk_dilsel_sonuc(k),a_ah_dilsel_sonuc(j)),tam,noktali);
               state<=aktif_uyeleri_belirle;
            when aktif_uyeleri_belirle =>
               gec<=karsi(a_katman3(((j-1)*4)+k),(((j-1)*4)+k));
               state<=aktif_uyeleri_belirle_dizi;
            when aktif_uyeleri_belirle_dizi =>
               if (gec>0) then
                  a_yer<=a_yer+1;
                  state<=aktif_uyeleri_belirle_dizi_atama;
               else
                  state<=katman3_ikinci_dongu_bitis;
               end if;
            when aktif_uyeleri_belirle_dizi_atama =>
               a_yeri(a_yer)<=gec;
               state<=katman3_ikinci_dongu_bitis;
            when katman3_ikinci_dongu_bitis =>
               if k=4 then
                  k<=0;
                  state<=katman3_ilk_dongu_bitis;
               else
                  state<=katman3_ikinci_dongu;
               end if;
            when katman3_ilk_dongu_bitis =>
               if j=4 then
                  j<=0;
                  state<=katman4_toplam_baslangic;
               else
                  state<=katman3_ilk_dongu;
               end if;
            when katman4_toplam_baslangic =>
```

```
            a_katman4toplam<=umin;
            i<=0;
            state <= katman4_toplam_dongu;
         when katman4_toplam_dongu =>
            i<=i+1;
            state <= katman4_toplam;
         when katman4_toplam =>
            a_katman4toplam<=topla_h(a_katman4toplam,a_katman3(a_yeri(i)));
            state <= katman4_toplam_dongu_bitis;
         when katman4_toplam_dongu_bitis =>
            if i=4 then
               i<=0;
               state<=amac_fonksiyon_uyarlama_baslangic;
            else
               state<=katman4_toplam_dongu;
            end if;
         when amac_fonksiyon_uyarlama_baslangic =>
            i<=0;
            state <= amac_fonksiyon_uyarlama_dongu;
         when amac_fonksiyon_uyarlama_dongu =>
            i<=i+1;
            state <= amac_fonksiyon_uyarlama_bir;
         when amac_fonksiyon_uyarlama_bir =>
            a_yeni(i)<=bolme_h(a_katman3(a_yeri(i)),a_katman4toplam);
            state <= amac_fonksiyon_uyarlama_iki;
         when amac_fonksiyon_uyarlama_iki =>
            a_yeni(i)<=carpma_h(a_yeni(i),error);
            state <= amac_fonksiyon_uyarlama_uc;
         when amac_fonksiyon_uyarlama_uc =>
            a_yeni(i)<=carpma_h(a_yeni(i),ogrenme);
            state <= amac_fonksiyon_uyarlama;
         when amac_fonksiyon_uyarlama =>
            a_o_b(a_yeri(i))<=topla_h(o_b(a_yeri(i)),a_yeni(i));
            state <= amac_fonksiyon_uyarlama_dongu_bitis;
         when amac_fonksiyon_uyarlama_dongu_bitis =>
            if i=4 then
               i<=0;
               state<=onsart_ortak_carpim;
            else
               state<=amac_fonksiyon_uyarlama_dongu;
            end if;
         when onsart_ortak_carpim =>
            on_carpim1<=carpma_h(a_katman4toplam,a_katman4toplam); --hata*ogrenme/(k4toplam^2)
            state <= onsart_ortak_carpim_2;
         when onsart_ortak_carpim_2 =>
            on_carpim2<=carpma_h(ogrenme2,error);
            state <= onsart_ortak_carpim_3;
         when onsart_ortak_carpim_3 =>
            on_carpim<=bolme_h(on_carpim2,on_carpim1); --hata*ogrenme/(k4toplam^2)
            state<=on_sart_uyarlama_baslangic;
         when on_sart_uyarlama_baslangic =>
            -- bu bolumde sadece gpk amac parametrelerinden ucgensel uyelik fonksiyonu orta noktasının
uyarlanması verilmistir.
            i<=0;
            state <= on_sart_uyarlama_dongu;
         when on_sart_uyarlama_dongu =>
            i<=i+1;
            state <= bk_tablosu_indis1
         when bk_tablosu_indis1 =>
            indis1(i)<=a_yeri(i) mod 4;
```

```
            state <= bk_tablosu_indis2;
         when bk_tablosu_indis2 =>
            indis2(i)<=((a_yeri(i)-indis1(i)) / 4)+1;
            state <= on_sart_uyarlama_amacfnk_carpim;
         when on_sart_uyarlama_amacfnk_carpim =>
            g1_f(i)<=carpma_h(on_carpim,o_b(a_yeri(i)));
            state <= on_sart_uyarlama_pay;
         when on_sart_uyarlama_pay =>
            g2_f(i)<=cikar_h(a_katman4toplam,a_katman3(a_yeri(i)));
            state <= on_sart_uyarlama_arahesaplama1;
         when on_sart_uyarlama_arahesaplama1 =>
            g3_f(i)<=carpma_h(g2_f(i),g1_f(i));
            state <= on_sart_uyarlama_arahesaplama2;
         when on_sart_uyarlama_arahesaplama2 =>
            g4_f(i)<=carpma_h(g3_f(i),a_ah_dilsel_sonuc(indis2(i)));
            state <= on_sart_uyarlama_arahesaplama3;
         when on_sart_uyarlama_arahesaplama3 =>
            g5_f(i)<=bolme_h(g4_f(i),gpk_dilsel_deger_u(indis1(i)));
            state <= on_sart_uyarlama_isaretfonksiyonu;
         when on_sart_uyarlama_isaretfonksiyonu =>
            if resize(gpk_dilsel_deger(indis1(i))-gpk,tam,noktali)>=umin then
               g6_f(i)<=carpma_h(g5_f(i),to_sfixed(-2,tam,noktali));
            else
               g6_f(i)<=carpma_h(g5_f(i),to_sfixed(2,tam,noktali));
            end if;
            state <= on_sart_uyarlama_yeni_a_parametresi;
         when on_sart_uyarlama_yeni_a_parametresi =>
            a_gpk_dilsel_deger(indis1(i))<=topla_h(gpk_dilsel_deger(indis1(i)),g6_f(i));
            state <= on_sart_uyarlama_dongu_bitis;
         when on_sart_uyarlama_dongu_bitis =>
            if i=4 then
               i<=0;
               state<=bulaniklastirma_baslangic;
            else
               state<=on_sart_uyarlama_dongu;
            end if;
         when others=>
            state <= init;
      end case;
      end if;
end process;
end davranis;
```

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : İlker Ali ÖZKAN
Uyruğu : T.C.
Doğum Yeri ve Tarihi : Karaman - 1978
Telefon : 505 7370719
Faks : -
e-mail : ilkerozkan@yahoo.com

EĞİTİM

Derece		Adı, İlçe, İl	Bitirme Yılı
Lise	:	Cumhuriyet Lisesi, Çumra, KONYA	1995
Üniversite	:	S.Ü. Elektronik ve Bilgisayar Eğitimi, KONYA	2003
Yüksek Lisans	:	S.Ü. Fen Bilimleri Enstitüsü, KONYA	2006
Doktora	:	S.Ü. Fen Bilimleri Enstitüsü, KONYA	2013

İŞ DENEYİMLERİ

Yıl	Kurum	Görevi
2005	S.Ü. Fen Bilimleri Enstitüsü	Araştırma Görevlisi
2011	S.Ü. Teknoloji Fakültesi	Araştırma Görevlisi

YABANCI DİLLER

İngilizce

YAYINLAR

I.A. Ozkan, I. Saritas, S. Herdem, The control of magnetic filters by FPGA based fuzzy controller, Energy Education Science and Technology Part A, 29(2), pp. 1093-102, 2012.

İ.A. Özkan ve S. Herdem, FPGA Tabanlı Kendini Ayarlayabilen Bulanık Kontrolör ile Manyetik Filtrenin Kontrolü, Otomatik Kontrol Ulusal Toplantısı TOK 2012, 2012.

İlker Ali Özkan, İsmail Sarıtaş, Saadetdin Herdem, "Manyetik filtreler için FPGA Tabanlı Bulanık Kontrolör Tasarımı", Teknik-Online Dergi, 10 (3), 2011.

Ismail Saritas, Ilker Ali Ozkan, Ibrahim Unal Sert, "Prognosis of prostate cancer by artificial neural networks", Expert System with Applications, Volume 37, Issue 9, Pages 6646-6650, 2010.

Ismail Saritas, Ilker Ali Ozkan, Saadetdin Herdem, "The effects of fuzzy control of magnetic flux on magnetic filter performance and energy consumption ", Expert System with Applications, 2010.

I. Saritas, I. A. Ozkan, N. Allahverdi and M. Argindogan, "Determination of the drug dose by fuzzy ezpert system in treatment of chronic intestine inflammation ", Journal of Intelligient and manufacturing, Volume 20, Number 2, Pages 169-176, 2009.

Ilker Ali Ozkan, Ismail Saritas, Suleyman Yaldiz, " A Comparative Study of ANN and FES for Predicting of Cutting Forces and Tool Tip Temperature in Turning ", International Conference on Computer Systems and Technologies - CompSysTech'10, 2010, Bulgaria.

Saritas I, Ozkan I.A., Herdem S., "The control of magnetic filter by neural network", 5. Uluslararası İleri teknolojiler Sempozyumu - IATS'09, 13-15 Mayıs 2009, Karabük.

Ilker Ali Ozkan, Ismail Saritas, Suleyman Yaldiz, "Prediction of cutting forces and tool tip temperature in turning using artifitical neural network" 5. Uluslararası İleri teknolojiler Sempozyumu - IATS'09, 13-15 Mayıs 2009, Karabük.

Saritas I., Ozkan I.A. ,Herdem S., "Design of a fuzzy expert system for Magnetic Filter According to magnetic Field ", 5. International conference on electrical and Electronics Engineering - ELECO'2007, pp.401-405, December 2007, Bursa , Turkey.

Ilker Ali Ozkan, Ismail Saritas, Saadetdin Herdem, "Manyetik filtreleme işleminin ANFIS ile Modellenmesi", Elektrik Elektronik Bilgisayar Biyomedikal Mühendisliği 12. Ulusal Kongre ve Sergisi- EMO'07, pp 415-418, 2007.

Ismail Saritas, Ilker Ali Ozkan, Saadetdin Herdem, "Manyetik filtreleme işleminin bulanık mantık yöntemiyle kontrolü ", Elektrik Elektronik Bilgisayar Biyomedikal Mühendisliği 12. Ulusal Kongre ve Sergisi- EMO'07, pp 415-418, 2007.

Ismail Saritas, Ilker Ali Ozkan, Mustafa Argindogan, Novruz Allahverdi, "Determination of the dose of drug by fuzzy expert system in treatment of chronic intestine inflammation", 5th International symposium on intelligient Manufacturing Systems IMS'2006, pp. 497-505, 2006, Turkey.

Novruz Allahverdi, Ismail Saritas, Ilker Ali Ozkan, Mustafa Argindogan, "An example of determination of medicine dose in the treatment by fuzzy method ", International conference on computer systems and technologies -CompSysTech'06, pp. III.A1.1.-III.A.1.10,2006, Bulgaria.

Ismail Saritas, Ilker Ali Ozkan, Suleyman Sulak, Novruz Allahverdi, "The Role of Internet in Computer-Aided Education", International conference on computer systems and technologies -CompSysTech'06, pp. IV.18.1.-IV.18.5,2006, Bulgaria.