

T.C.

GEBZE YÜKSEK TEKNOLOJİ ENSTİTÜSÜ
MÜHENDİSLİK VE FEN BİLİMLERİ ENSTİTÜSÜ

BULANIK MANTIKLA PROSES
KONTROLÜ İÇİN GENEL AMAÇLI
BİR ARABİRİM VE ARAYÜZ TASARIMI

Serkan Çağlar ÜNAL
YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ

ANABİLİM DALI

T.C. YÜKSEKÖĞRETİM KURULU
DOKÜMANTASYON MERKEZİ

TEZ DANIŞMANI

Doç.Dr. Emin Murat ESİN

GEBZE

Serkan Çağlar ÜNAL'ın tez çalışması G.Y.T.E. Mühendislik ve Fen Bilimleri/ Sosyal Bilimler Enstitüsü Yönetim Kurulu'nun 20/8/2001 tarih ve 2001-22 sayılı kararıyla oluşturulan jüri tarafından Bilgisayar Anabilim Dalında YÜKSEK LİSANS Tezi olarak kabul edilmiştir.

JÜRİ

Üye
(Tez Danışmanı)

Doç.Dr. E. Murat ESİN

Murat Esin

Üye

Yrd.Doç.Dr. Oğuz KUCUR

Oğuz Kucur

Üye

Yrd.Doç.Dr. Arif ERGİN

Arif Ergin

ONAY

GYTE Mühendislik ve Fen Bilimleri Enstitü Yönetim Kurulu'nun 17.09.2001 tarih ve 2001/24. sayılı kararı.



ÖZET

Gelişen teknoloji ile birlikte, sistemler büyümüş ve karmaşık bir hale gelmiştir. Bunun sonucu olarak sistemlerin matematik modelini oluşturmak güçleşmiştir. Bulanık kontrol özellikle çok sayıda giriş ve çıkışa sahip karmaşık sistemlerde rahatlıkla uygulanabilmektedir. Bunun nedeni bulanık kontrolün bir matematik modele ihtiyaç duymamasıdır.

Bu çalışmada bulanık kontrolün kullanımını kolaylaştırmak için bir arabirim kartı ve Delphi'de bir yazılım geliştirilmiştir. Arabirim kartında içerisinde ADC bulunan PIC16F877 kullanılmıştır. Bu arabirim kartı yazılım ile birlikte çalışmaktadır. Yazılım sayesinde kullanıcı, kontrol sistemine ait parametreleri kolaylıkla bilgisayara girmektedir. Arabirim kartı bilgisayar ile haberleşmek için RS485 protokolünü kullanmaktadır. Bunun avantajı elektronik gürültüye karşı dirençli olmasıdır. Bu sayede endüstriyel ortamlarda kullanılmaktadır. Kartın 16 analog girişi; 16 sayısal çıkışı veya 4 analog çıkışı bulunmaktadır. Kart 0V ile 5V arasında aldığı analog işaretleri sayısal işarete dönüştürmektedir. Yazılım tarafından bu sayısal işaret işlenmektedir. Daha sonra yazılım tarafından çıkış bilgileri oluşturulmaktadır. Kullanıcı çıkış verisinin analog veya sayısal olmasını seçebilmektedir. Bu kart ve yazılımın avantajı başka herhangi bir ek donanım kullanmadan bulanık kontrol ile bir sistemi kontrol edebilmesidir.

SUMMARY

Upon advances in technology, systems have become bigger and more complicated. As a result, mathematical models of such systems have become challenging. Fuzzy control can be applied easily to the complicated systems that have so many inputs and outputs. Therefore, it is not necessary to use a mathematical model in fuzzy control.

In this study, we manufactured an interface card and developed a software to use fuzzy logic more easily. The software developed by Delphi. PIC16F877 that includes ADC, is used in the interface card. This interface card works with software. With the help of the software, user can enter all control parameter to computer easily. The interface card uses RS485 protocol for communication with the software. Advantage of RS485 is that it is resistant to electronic noise. For that reason, it can be used in industrial applications. This card has got 16 analog inputs and 16 digital outputs or 4 analog outputs. It takes samples from 0V to 5V and converts to digital data. After that, the interface card sends this digital data to the computer. This digital data is processed by the software. Then, the software produces some output data. User can select to use analog or digital data. Advantage of this software and the interface card is that user does not need to use another hardware to control systems.

TEŞEKKÜR

Bu tezin gerçekleşmesi için, tez boyunca yardımını ve teşvikini esirgemeyen değerli hocam Doç.Dr. E.Murat ESİN'e şükranlarımı sunarım.

Çalışmalarında bana destek olan GYTE'de görevli yakın arkadaşlarıma ve TUBITAK-UEKAE'deki mesai arkadaşlarıma teşekkürü bir borç bilirim.

Yaşamım boyunca maddi ve manevi desteğini esirgemeyen, her zaman yanımda olan aileme teşekkür ederim.



İÇİNDEKİLER

SAYFA

ÖZET	iv
SUMMARY	v
TEŞEKKÜR	vi
İÇİNDEKİLER DİZİNİ	vii
KISALTMALAR	ix
ŞEKİLLER DİZİNİ	x
TABLolar DİZİNİ	xiii
1. GİRİŞ	1
2. BULANIK MANTIK TEORİSİ	4
2.1. Bulanık Mantık'ın Tarihsel Gelişimi	4
2.2. Bulanık Mantık Küme Teorisi	5
2.3. Boolean Küme Teorisine Karşı Bulanık Küme Teorisi	6
2.4. Üyelik Derecesi Kavramı	7
2.5. Üyelik Fonksiyonları	9
2.6. Matematiksel Prensipler İçin Terimlerin Tanımı	10
2.7. Boolean Matematiğine Karşı Bulanık Matematik	12
2.8. Bulanık İlişkileri	17
2.9. Tablo Oluşturulması	19
2.10. Üyelik Fonksiyonlarının Belirlenmesi	20
3. BULANIK KONTROL	23
3.1. Kontrol sistemleri	23
3.1.1. Açık Döngü Kontrol Sistemleri	23
3.1.2. Kapalı Döngü Kontrol Sistemi	24
3.2. Kontrolde Bulanık Mantık	25
3.3. Genel Bir FMC'nin Çalışması	27
3.3.1. Bulanıklaştırma (Fuzzifier)	28

3.3.2. Kontrolör (Controller)	29
3.3.3. Durulaştırma (Defuzzifier)	30
3.4. Bulanık Kontrolör	31
3.5. Bulanık Mantık'ın Temelleri	34
3.6. Bulanık Mantık'a Uymak	34
3.7. Bulanık Modeli	36
4. BULANIK MANTIK İLE PROBLEM ÇÖZÜMÜNE BİR ÖRNEK	38
4.1. Kural dizini	39
5. UYGULAMA	44
5.1. Arabirim Kartı	46
5.1.1. Arabirim Kartının Yapısı	47
5.1.2. Arabirim Kartının Testi ve Ayarı	65
5.2. Yazılım Paketinin Tasarlanması	68
5.2.1. Program Yapısı	69
5.2.2. Başlangıç Programı	73
5.2.3. Kural Modülü	74
5.2.4. Kontrol Modülü	79
6. ÖRNEK BİR SİSTEMİN KONTROLÜ	84
6.1. Prosesin Kontrol Elemanları	84
6.1.1. Girişler	84
6.1.2. Çıkışlar	88
6.2. Sistemi Tanımlayan Bulanık Mantık Üyelik Fonksiyonlarının Tanımlanması	88
6.3. Sistemin Çıkış Verileri	98
7. SONUÇ	103
KAYNAKLAR	105
ÖZGEÇMİŞ	106

KISALTMALAR

Simgeler

Açıklama

MCF

Üyelik karakteristik fonksiyonu
(Membership Characteristic Function)

μ

Üyelik derecesi
(Membership Degree)

DAC

Sayısal analog çevirici
(Digital to Analog Converter)

ADC

Analog sayısal çevirici
(Analog to Digital Converter)

USART

Dahili bir evrensel senkron asenkron
alıcı verici
(Universal Synchronous-
Asynchronous Receiver)

FOCS

Osilatör frekansı

RO

TTL çıkış
(Receiver Output)

RE

Veri alma modu
(Receiver Enable)

DI

TTL giriş
(Driver Input)

DE

Veri gönder modu
(Driver Enable)

ŞEKİLLER DİZİNİ

<u>Sekil</u>	<u>Sayfa</u>
2.1. Crips Mantık ile Bulanık Mantık'ın Karşılaştırılması	8
2.2. Üyelik Derecesi	8
2.3. Üçgen Tipi Üyelik Fonksiyonu	9
2.4. Trapez Tipi Üyelik Fonksiyonu	9
2.5. Gauss Çan Eğrisi Tipi Üyelik Fonksiyonu	10
2.6. A ve B kümeleri	14
2.7. A ve B Kümesinin Kesişimi	14
2.8. A ve B Kümesinin Birleşimi	15
2.9. A Kümesinin Terslenmişi	15
2.10. Üyelik Matrisi	18
2.11. Üyelik Tablosu	18
2.12. Saggital Diyagramı	18
2.13. Üyelik Grafiği	18
3.1. Açık Döngü Kontrol Sistemi	22
3.2. Kapalı Döngü Kontrol Sistemi	24
3.3(a). Matematiksel Model Kullanılarak Gerçekleştirilen Kontrol	26
3.3(b). Arama tablosu İle Gerçekleştirilen Kontrol	26
3.3(c). Bulanık Mantıkla Gerçekleştirilen Kontrol	26
3.4. Bir FMC Kartının Birimleri	27
3.5. Üyelik Fonksiyonları	28
3.6. PID Kontrol Şeması	37
3.7. Bulanık Kontrol Şeması	37
4.1. Su Miktarı Üyelik Dereceleri	40
4.2. Suyun Başlangıç Sıcaklığının Üyelik Dereceleri	41
4.3. Isıtıcı Gücü Üyelik Dereceleri	41
4.4. Fonksiyonun Çıkış Değerinin Bulunması	43
5.1 Fiziksel Ortamın ve Kontrol Ortamının İlişkisi	45
5.2 Arabirim Kartı Blok Diyagramı	47

5.3. PIC16F877'nin Bacak Bağlantıları	50
5.4. 4051'in Bacak Bağlantısı	56
5.5. MAX485'in Bacak Bağlantısı	60
5.6. Ara Birim Kartı Açık Şeması	62
5.7. Kontrol Kartı	63
5.8. Haberleşme Modülü	63
5.9. Mikrodenetleyici Programının Akış Diyagramı	64
5.10. Test Programı Giriş Ekranı	67
5.11. Program Giriş Ekranı	73
5.12. Kural Modülü Giriş Ekranı	74
5.13. Kural Modülü Menü Seçenekleri	75
5.14. Proje Tanımlama Ekranı	75
5.15. Değişken Giriş Ekranı	76
5.16. Giriş Tanımlama Ekranı	76
5.17. Giriş Grafiği Çizme Ekranı	77
5.18. Tanımlanmış Girişlerin Görüntüsü	78
5.19. Kural Tablosu	79
5.20. Dil Seçim Menüsü	79
5.21. Seri İskele Ayar Ekranı	80
5.22. Giriş Çıkış Değerlerinin Görüntüsü	80
5.23. Kontrol Parametre Ayarları	81
5.24. Giriş Çıkış Değerlerinin Görüntülenmesi	82
5.25. Renk Seçim Ekranı	83
5.26. Üyelik Derecelerinin Görüntülenmesi	83
6.1. Homojen Dağılımlı Sıcaklık Kontrol Prosesi	84
6.2. J Tipi Termokulp Gerilim Sıcaklık Grafiği	85
6.3. J Tipi Termokupl	85
6.4. Kazanç Devresi	86
6.5. Yükselteç Çıkışındaki Sıcaklık-Gerilim Grafiği	86
6.6. Filtre Devresi	87
6.7. Kazanç ve Filtre Devresi	87
6.8. Karıştırıcı Ve Isıtıcının Çalışma Süreleri	88
6.9. Fuzzy Engine Programının Giriş Ekranı	89
6.10. Kural Modülü Giriş Ekranı	89

6.11. Sisteme Ait Genel Bilgilerin Girildiği Ekran	90
6.12. Giriş Parametrelerin Girildi Ekran Görüntüsü	90
6.13. Girişe Ait Özelliklerin Girildiği Ekran	91
6.14. Girişlere Ait Kümelerin Tanımlanması	92
6.15. Girişlere Ait Tüm Kümelerin Tanımlanmış Hali	93
6.16. Isıtıcıya Ait Küme Tanımlamaları	93
6.17. Karıştırıcıya Ait Küme Tanımlamaları	94
6.18. Kural Tablosu	94
6.19. Kontrol Modülü Giriş Ekranı	95
6.20. Kontrol Edilecek Sisteme Ait Parametrelerin Olduğu Ekran	96
6.21. Seri İskele Seçim Ekranı	96
6.22. Kontrol Programı Ayarları	97
6.23. Kontrol Modülü Giriş Çıkış Ekranı	98
6.24. Çıkışlar Pasif İken Sistem Davranışı	98
6.25. Sadece Karıştırıcı Kullanıldığında Sistem Davranışı	99
6.26. Sadece Isıtıcı Kullanıldığında Sistem Davranışı	100
6.27. Isıtıcı ve Karıştırıcı Kullanıldığında Sistem Davranışı	100
6.28. Giriş ve Çıkış Çok Farklı Olduğunda Sistem Davranışı	101

TABLÖLAR DİZİNİ

Tablo

Sayfa

2.1. Fonksiyon Sadeleştirme	20
5.1. PIC16F877 Hafıza Paylaşımı	51
5.2. A/D İskele Ayarları	53
5.3. Başlık Bilgileri	65
5.4. Kural Modülünden Beklentiler	68
5.5. Kontrol Modülünden Beklentiler	68



1. GİRİŞ

Mühendislikte ve diğer bilim dallarında sistemler, kesin matematiksel işlemleri kullanmak suretiyle modellenir. Günlük hayatta karşılaşılan problemlerin büyük çoğunluğu bir kesin olmama durumu veya bir tanımlanmama durumu içerir. Bu problemlerin daha etkin çözülebilmesi için bir yol bulunması gerekmektedir. Günümüzde sistemler genel olarak kararlı, lineer ve zamana bağımlı değişmeyen bir yapıya sahiptir. Bu özelliklerin dışına çıkıldığı zaman sistem kontrol edilemez hale gelecektir. Eğer bu sistem insan gibi hareket edebilen bir yapıyla kontrol edilebilirse, kararlı veya kararsız, lineer olan ya da olmayan, zamanla değişen veya değişmeyen bir özellikte olsa dahi uygun bir şekilde çalıştırılabilir. İşte bu durumlar bilim adamlarını insan gibi algılayabilen, karar verebilen ve hareket edebilen sistemler yapmaya yöneltmiştir.

Bulanık mantık (fuzzy logic) klasik mantığı özel bir hal olarak içine alan, iki mantık seviyesi yerine çok sayıda mantık seviyesini benimseyen bir mantıktır. Bulanık mantık ilk defa 1965 yılında Lotfi A. Zadeh tarafından ortaya atılmıştır. Bulanık mantık ile ilk varılmak istenen temel hedef insan gibi düşünen, karar verebilen, insiyatif kullanabilen, duruma göre seçim yapabilen kontrol sistemleri oluşturmaktır. Bulanık mantık, klasik mantıktaki gibi kesin hatlarıyla birbirinden ayrılmış bölgeler yerine bizim tarafımızdan tanımlanan fonksiyonlarla birbirinin içine geçmiş çok sayıdaki bölgeyi kullanır [Bezdek,1993].

Dolayısıyla günlük hayatta karşılaşılan bir önermeye tamamen yanlış ya da tamamen doğru demek hatalı olur. Klasik mantık kullanılacak olunursa bu önerme ya yanlış, ya da doğrudur arası yoktur. Bu durum küme kavramına uygulanacak olunursa, klasik küme teorisinde söz konusu evrenin elemanları bir A kümesine göre o kümeye ait olanlar ve üye olmayanlar diye ikiye ayrılır. Üye olanların üyelik derecesi "1" ve üye olmayanların üyelik derecesi "0" dır. Bulanık mantıkta kesin doğru ya da kesin yanlış diye bir yaklaşım yoktur. Bu esneklikten yararlanılarak, girdilerin kesin olarak doğru ya da yanlış olduğuna bakılmaksızın sonuçlar üretilebilir. Yapılan iş, girdilerin doğru ya da yanlış kümesi içerisindeki üyelik derecesini hesaplamak ve belirlenen kurallara göre çıktı üretmektedir. Bu noktadan

görülebileceği gibi üyelik derecesi hesaplanırken ihtimal ve grup teorileri kullanılmaktadır.

Zaman domeninde çözülmesi güç problemler, frekans domeninde çözülüp tekrar zaman domenine aktarılmasına benzer olarak normal alanda kontrol edilmesi güç; hatta hiç mümkün olmayan sistemleri kontrol etmek için de bulanık alanına girip tekrar geriye sonuçları normal olana aktarmak mümkündür. Bulanık mantıkta bu özelliğin kullanılması ile birçok kontrol problemi sorun olmaktan çıkmıştır.

İlkönce sistem iyice incelendikten sonra, neden bulanık kontrol seçildiğine dair kararın verilmesi gerekmektedir. İkinci aşamada ise sistemin ne tip bir bulanık modelinin çıkarılacağına ve neden böyle olduğuna da karar verilmesi önemlidir.

Bütün bu açıklamalar ışığında bulanık mantık, girdileri tanımlanan kümeler üzerindeki üyelik derecesini hesaplayan ve yine tanımlanan kurallarla bu üyelik derecesini karşılaştırarak, her bir kural için farklı sonuçlar üretebilen bir mantıktır. Sonuç olarak bulanık mantığı çok seviyeli mantık olarak ifade etmek mümkündür.

Gelişen teknoloji bulanık mantığın öne çıkmasına neden olmuştur. Sistemler daha karmaşık yapılara kavuşmuş, sistemden beklentiler artmıştır. Günümüzde bulanık mantık üzerine olan çalışmalar gün geçtikçe artmaktadır. Yapılan çalışmaları iki kısımda inceleyebiliriz. Bunlardan birincisi yazılım ağırlıklı çalışmalar ki bunlar genelde sistem davranışlarını inceleyen programlardır. İkinci kısım ise kontrol üzerine yapılan donanım çalışmalarıdır.

Kontrol programları genelde sisteme ait matematik modelleri kullanarak sistem davranışları hakkında bilgi vermektedir. Bu sayede bazen çok masraflı veya çok zaman alacak problemlerin çözümlerine ulaşmak bu tür programlar sayesinde kolaylaşmaktadır. Bulanık kontrol üzerine de bir çok simülasyon programı bulunmaktadır. Bu programlar sistem davranışları hakkında bize bilgi vermektedirler. Fakat bu programlardan büyük bir oranı, donanım altyapıları bulunmadığı için kontrol edilecek olan sisteme bağlanamazlar. Dolayısıyla sistem kontrolünü gerçekleştirememektedirler. Bu programların en büyük faydası sistem

kontrolüne geçmeden sistem davranışlarını inceleyebilmek ve buradan çıkacak sonuçları donanıma taşıyıp sistem kontrolünü gerçekleştirmektir.

Bulanık kontrolün hızla yaygınlaşması kontrolör üreten firmalar üzerinde özel amaçlı kontrolör üretimi için bir baskı oluşturmuştur. Normal bir kontrolörü bulanık mantığın uygulandığı bir kontrol sisteminde kullanmak mümkündür fakat bulanık mantığa özgü birçok işlemin kullanıcı tarafından yazılması gerekmektedir. Bulanık mantık için geliştirilen kontrolörler de ise karşılaştırma işlemleri kontrolör içerisinde bulunmaktadır. Bunun dezavantajı ise kontrolörlerin kapasiteleriyle sınırlı kalınmasıdır. Kontrolörlerle yapılan donanım kartlarının bir diğer uygulama zorluğu, kontrolörü programlamanın belli bir birikim gerektirmesidir. Bununla beraber kontrol edilecek her yeni sistem için veya üzerinde yapılacak değişiklikler için kontrolörün tekrar programlanması, hatta bazen donanım değişikliklerinin yapılması gerekmektedir.

Bu çalışmada yapılması amaçlanan, hiç donanım tecrübesi olmayan bir kişinin dahi rahatlıkla kullanabileceği bilgisayar tabanlı bir kontrolör gerçekleştirmektir. Gerçekleştirilen arabirim kartı sayesinde bilgisayarda bulunan yazılımın kontrol edilecek olan sistemi kontrol etmesi sağlanmıştır. Yapılan çalışmayı yazılım ve donanım olarak iki kısımda inceleyebiliriz.

Yazılım, kural ve kontrol modülü olmak üzere iki kısımdan oluşmaktadır. Kullanıcı bulanık mantık ile kontrol edeceği sisteme ait verileri kural modülünde oluşturduktan sonra, kontrol modülü arabirim kartı sayesinde sistemi kontrol edecektir. Arabirim kartı 16 analog giriş, 16 sayısal çıkış veya 4 analog çıkıştan oluşmaktadır. Bilgisayar ile haberleşmesi ise RS485 protokolü ile gerçekleşmektedir, bu sayede özellikle endüstriyel ortamlarda kullanıma elverişlidir.

İkinci bölümden dördüncü bölüme kadar olan kısımda bulanık kontrolün tarihsel gelişimi ile teorisinden bahsedilmiştir. Beşinci bölümde ise yapılan yazılım ve donanım anlatılmıştır. Altıncı bölümde gerçek bir prosesin bu sistemle kontrolü ile ilgili çalışma verileri ve uygulaması anlatılmaktadır.

2. BULANIK MANTIK TEORİSİ

2.1. Bulanık Mantık'ın Tarihsel Gelişimi

Bulanık mantık temel olarak insanın düşünüş şeklini örnek almaktadır. Dolayısıyla bu mantığın küme yapısında kullanılan elemanlar insan mantığına çok yakın ya da aynı olan elemanlardır. Bulanık matematiğin en önemli özelliği, klasik matematiğin sağladığından daha fazla ifade edilebilir durum yaratması ve doğal dildeki ifadelerle daha uygun temsiller sağlamasıdır. Bulanık mantık teorisi ilk defa 1965'te Lotfi A. Zadeh tarafından üç temel küme yaklaşımına alternatif olmak üzere ortaya atılan bir teoridir. Bu üç küme yaklaşımı şunlardır [Patyra and Mlynek,1996]:

Klasik mantık: Tüm evrensel elemanların bir küme elemanı olup olmamasına dayanan bir teoridir. Buna göre elemanlar ya bir ya da sıfır diye tanımlanabilir.

Keskin sınırlar Teorisi: Sınırları kesin bir şekilde belirlenmiş alan teorisidir. Bir TTL(Transistor-Transistor Logic) seviyede 0-0,8 arası düşük seviye kümesine, 0,8-2,4V arası belirsiz kümesine, 2,4-5V arası da yüksek seviye kümesine aittir. Bu sınırlarla belirlenmiş olan düşük seviye, belirsiz, yüksek seviye kümeleri kesin sınırlarla alan teorisini örnekleyecektir.

İhtimal ve grup teorisi: Elemanların bir kümede bulunma veya bulunmama olasılıkları ile ilgilidir.

1980'li yıllardan sonra bulanık mantık hızla gelişmeye başlamıştır. Bu şekilde hızlı gelişmenin nedenleri şu şekilde sıralanabilir:

- i. Teorinin matematiksel olgunlaşması pratiğe geçişte büyük rol oynamıştır.

ii. The North American Fuzzy Information Processing Society, Japon Society Fuzzy System and Theory gibi uluslararası bilim kuruluşlarının destekleri de gelişimde önemli rol oynamıştır.

iii. Teorinin birçok üniversitede ders olarak okutulması ile gelişmeler daha da anlam kazanmıştır.

iv. Bütün bunların yanısıra, teorinin endüstriyel sahalara uygulanmaya başlamasıyla elde edilen başarılı sonuçlar, bulanık mantığın yaygın bir şekilde kullanılmasına neden olmuştur [KAUFMANN and GUPTA, 1991].

Günümüzde bulanık mantık ile ilgili bir çok çalışma yapılmakta ve bu alandaki çalışmalar uygulama alanı bulunmaktadır. Yakın gelecekte ise bulanık mantık sistemlerin kontrolünde daha fazla kullanılacak ve klasik mantığın yerine geçecektir. Özellikle lineer olmayan ve çok giriş çıkışa sahip karmaşık sistemlerin kontrolünde bulanık mantık büyük kolaylıklar sağlamaktadır.

2.2. Bulanık Mantık Küme Teorisi

Bulanık mantık, küme teorisine dayanır. Bu teori aslında daha genel bir matematiksel küme yaklaşımıdır. Klasik mantık yaklaşımından farkı, çok daha fazla kümenin olmasıdır; ayrıca bir eleman birden fazla küme elemanı olabilmektedir. Bu yaklaşımla çözülmesi çok güç olan problemler basitleştirilerek genel bir yapıya kavuşturulur ve bu sayede daha kolay bir sonuca gidilir.

Bulanık mantıkta da Boolean mantığında olduğu gibi çeşitli girişler insan tarafından tanımlanan fonksiyonlara etki ettirilerek çıkış ya da çıkışlar elde edilir. Aralarındaki en önemli fark ise, bir bulanık girdisinin üyesi olduğu kümeye 0 ile 1 arasında değişen bir üyelik derecesinin bulunmasıdır. Bu üyelik derecesi (Membership Characteristic Function, MCF) adlı bir fonksiyonla kullanıcı tarafından belirlenir [BANDEMAR and GOTTWOLD, 1996]. Bulanık küme teorisi bilinen

klasik küme teorisinin genel bir halidir. Yani tam tersi de klasik küme teorisi, bulanık küme teorisinin özel bir halidir şeklinde ifade edilebilir.

Bulanık mantık insanın düşünme yeteneği model alınarak, insanların bilgiyi değerlendirerek ve insan beynindeki verilerden bilgiyi veya sonucu çıkarabilme özelliğine dayanır. Bulanık mantıkta günlük hayatta kullandığımız bir takım sözel ifadeler kullanılarak olaylar tanımlanır.

Bulanık küme teorisi kısmi üyeliğe izin veren bir teoridir. Yani, bir kümenin tam üyeliği ile o kümenin üyesi olmama durumları arasında, kademe kademe geçişe izin verir. Çünkü, bulanık küme teorisi hem tam üyeliğe hem de hiç üye olmamaya izin verir. İşte bundan dolayı klasik küme teorisinin genelleştirilmiş hali denilebilir. Klasik küme anlayışının bir diğer adı da CRISP'tir [KLIR and YUAN, 1995].

Bulanık küme teorisindeki bulanıklık ile elemanlarının üyelik derecelerini veren üyelik fonksiyonları geliştirilerek kesin sınırlar ortadan kaldırılmıştır. Böylece üyeler ve üye olmayanlar arasında yumuşak bir geçiş sağlanmıştır.

2.3. Boolean Küme Teorisine Karşı Bulanık Küme Teorisi

Boolean cebri, kesin tanımlı kümelerin elemanları olan değişkenlere dayanır. Buna göre sonuç, ya doğrudur ya da yanlıştır, yani ya vardır ya da yoktur. Sonucun bu iki değerden başka bir değer alması söz konusu değildir. Klasik tabanlı kümelerde bir nesne ya da değişken kümenin ya tam üyesidir ya da üyesi değildir [ZADEH and KACPRZYK, 1992]. Örneğin;

$$\text{Sıcaklık}(t) = \begin{cases} x=0 \text{ (Sıcak değil)} & \text{Eğer } t \leq 40 \\ x=1 \text{ (Sıcak)} & \text{Eğer } t > 40 \end{cases}$$

Boolean deęiřkeni x , sıcaklık 40°C 'den yüksek sıcaklıklar kümesinin üyesi ise $x=1$ ve üyesi deęil ise $x=0$ 'dır. X deęiřkeni sadece 0 ve 1 deęerlerini alabilir ve $\{0,1\}$ kümesi x 'e baęlı kesikli üyelik kümesi denebilir. “BÜYÜKLÜK” ($>$) operatörü kesin tanımlı bir işlemi gösterir. Dięer yandan eęer x , 40°C 'den yüksek sıcaklıklar kümesinin bir elemanı olarak düşünülürse, açıktır ki küme tanımlayan fonksiyonunda kesin tanımlamalar yoktur. Bulanık üyelik karakteristik fonksiyonu (MCF) ve alfa altındaki ya da üstündeki bütün elemanlarının tamamen Boolean mantığındaki kümenin elemanları olduęu noktayı gösteren parametresi genellikle kullanıcı tarafından tanımlanır. [TERANO, ASIA and SUGENO, 1992]. MCF ve alfa parametresi bulanık alt kümesindeki bütün elemanlar arasındaki ilişkiyi belirler. Bundan dolayı ařaęıdaki sonuçlar çıkarılabilir:

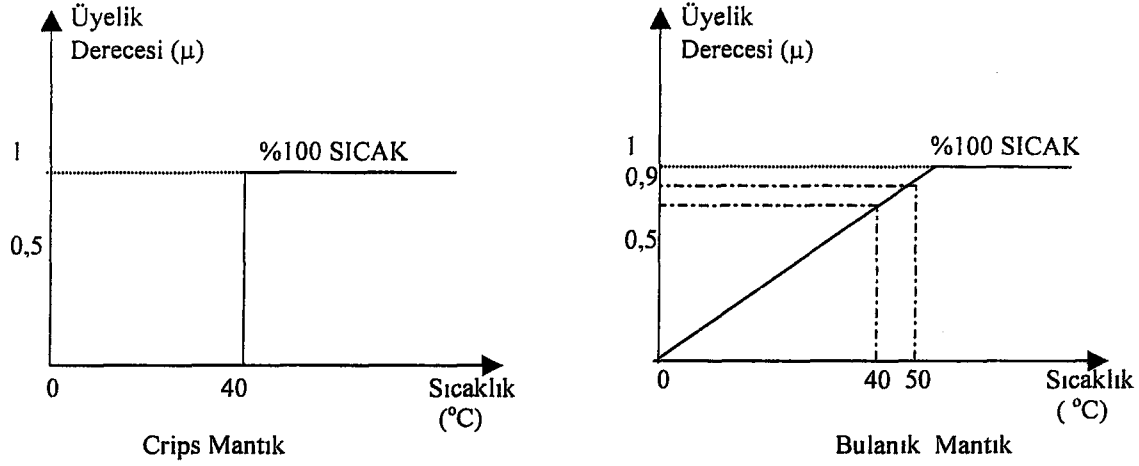
MCF sonsuz sayıda řekil alabilir (probleme baęlı olarak doęru, parabol v.b.) . Dięer taraftan Boolean üyelik karakteristik fonksiyonu (MCF) kesin olarak basamak fonksiyonudur.

Bulanık üyelik karakteristik fonksiyonu “yüksek” belirleyicisi, “BÜYÜK EŐİTLİK” ($\geq$) belirleyicisi ile deęiřtirildięinde Boolean üyelik karakteristik fonksiyonuna dönüşür. Böylece Bulanık kümelerinin, Boolean kümelerinin genel hali olduęu söylenebilir.

2.4. Üyelik Derecesi Kavramı

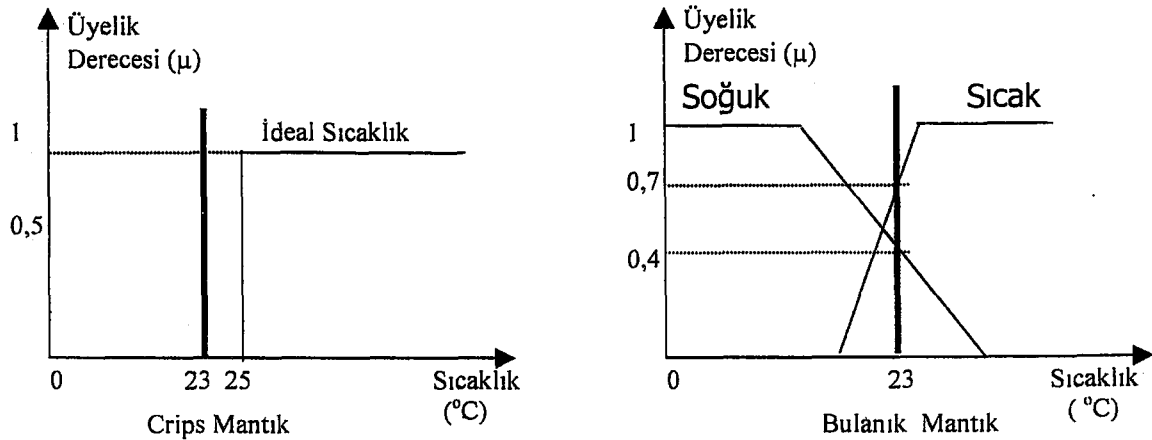
Bulanık mantık ile geręekleřtirilen çoęu sistemlerde bir kümenin kısmi üyelięe geęiř bölgesi oluşur. Klasik mantık için ya yanlış ya da doęru olacaktır. Fakat bulanık mantık, hem tam üyelięi hem de olmamayı kapsayan bir mantıktır, yani verilen durumlar kısmi üyelik de olabilir. řekil 2.1'de görüldüęü gibi 40°C 'den yüksek sıcaklıkların tartıřıldıęı kabul edilsin. Klasik küme anlayıřına göre 30°C sıcak kümesi içinde deęildir, hatta 39°C sıcak kümesi içinde bulunmamaktadır. Bulanık mantıęa göre 39°C 'deki sıcaklıęa soęuk denilemez; çünkü kısmen de olsa sıcak kümesi içindedir. Hatta bulanık kümesi olan sıcak kümesi, tam üyelikten hiç üye olmamaya kadar uzanan bölgede kısmi üyelięe izin verdięi için 25°C sıcaklık

bile sıcak bölgesinde değildir denilemez. 50 °C sıcaklığın klasik kümede üyelik derecesi sorulacak olunur ise, grafikten alınacak cevap “ %100 SICAK kümesindedir “ olmasına rağmen, bulanık kümesinin cevabı %90 sıcak kümesindedir olacaktır.



Şekil 2.1. Crips Mantık ile Bulanık Mantık'ın Karşılaştırılması

Klasik küme anlayışında bir değer ya kümenin elemanıdır ya da değildir. Zaten klasik küme anlayışında bir küme bulunmaktadır. Bulanık mantıkta ise küme sayısı birden fazla olabilmekte ve bir değer birden fazla kümenin elemanı olabilmektedir. Örnek olarak Şekil 2.2'de görüldüğü gibi ideal ev sıcaklığının tartışıldığı kabul edilsin. Klasik küme anlayışında 23 °C ideal ev sıcaklığı kabul edilmezken, bulanık mantıkta 23 °C % 40 soğuk kümesinde ve % 70 sıcak kümesinde yer almaktadır.

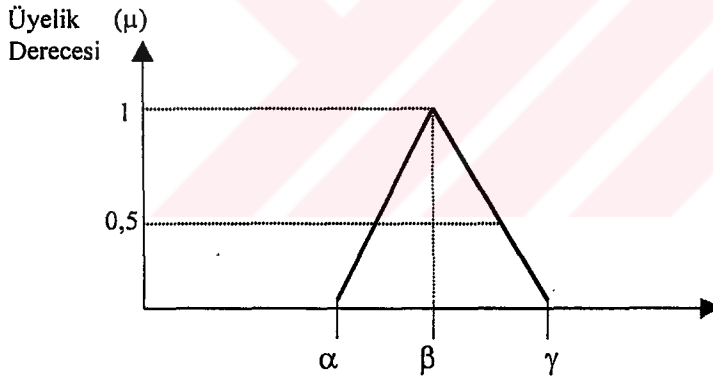


Şekil 2.2. Üyelik Derecesi

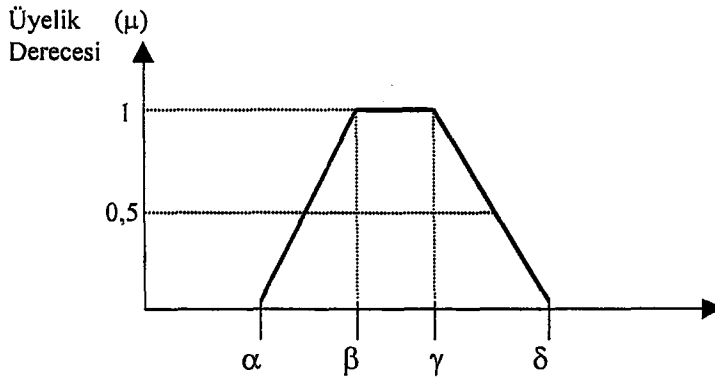
2.5. Üyelik Fonksiyonları

Bulanık kümeler üyelik fonksiyonu ile ifade edilmektedir. Bu üyelik fonksiyonları her nesnenin bir kümedeki ağırlık derecesini vermektedir. Bu ağırlık derecesi 0 ile 1 arasında değişmektedir. Üyelik fonksiyonu sözel terimlerle ifade edilir.

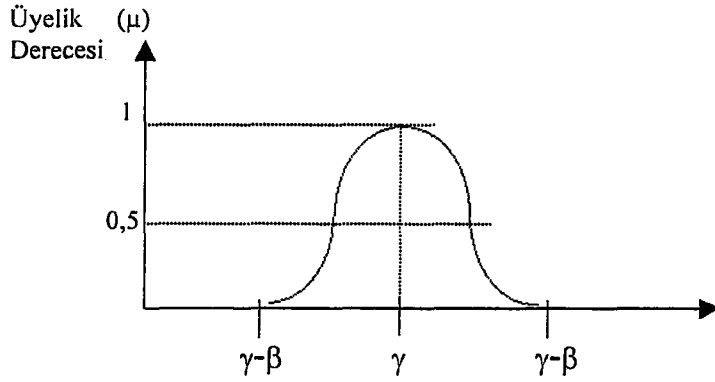
Literatürde, bulanık mantık için üyelik fonksiyonunun birçok farklı tipi önerildiği halde çoğu uygulamalarda standart üyelik fonksiyonu kullanılmaktadır. Doğrusal olmayan sistemlerde en çok kullanılan üyelik fonksiyonu tipleri üçgen (triangular), trapez (trapezoidal) ve Gauss çan eğrisi (Gauss bell-shaped) olarak sıralanabilir. Bu üç tip üyelik fonksiyonu sırasıyla Şekil 2.3, Şekil 2.4 ve Şekil 2.5'de görülmektedir [WANG, 1997].



Şekil 2.3. Üçgen Tipi Üyelik Fonksiyonu



Şekil 2.4. Trapez Tipi Üyelik Fonksiyonu



Şekil 2.5. Gauss Çan Eğrisi Tipi Üyelik Fonksiyonu

2.6. Matematiksel Prensipler İçin Terimlerin Tanımı

Klasik Küme: Klasik küme teorisindeki “0” ve “1” değerlerini içeren kümedir.

Bulanık Küme: Bulanık küme teorisinde kullanılan kümelerdir. Burada bulanık kümenin bir elemanının üyelik derecesini bulmak için üyelik fonksiyonları tanımlanmıştır. Çünkü bulanık küme teorisinde her şeyin bir doğruluk derecesi vardır.

Üyelik Fonksiyonu: Bulanık küme teorisinde bulanık kümeler kendi üyelik fonksiyonları ile tanımlıdır. Bu fonksiyonlar evrensel küme elemanlarının bulanık kümesindeki üyelik derecelerini tanımlamada kullanılır.

Destekleyen: X evrensel kümesindeki bir A bulanık kümesinin destekleyeni crips kümesi olup X 'in A bulanık kümesindeki 0'dan büyük üyelik derecesi olan elemanlarını kapsar.

$$\text{Dest. (A)} = \{ x \in X \mid \mu(x) > 0 \} \quad (2.1.)$$

Eğer X sonlu ve sayılabilir bir küme ise ve A bulanık kümesinin X 'de sonlu destekleyeni varsa, A aşağıdaki gibi yazılır:

$$A = (\mu_1/x_1) + (\mu_2/x_2) + \dots + (\mu_n/x_n) \quad (2.2.)$$

$$A = \sum_{i=1}^n (\mu_i/x_i) \quad (2.3.)$$

(μ_i, x_i 'in A bulanık kümesindeki üyelik derecesidir.)

Eğer X reel sayılar aralığında tanımlanmış bir evrensel küme ise A bulanık kümesi aşağıdaki notasyonla gösterilebilir:

$$A = \mu_A(x) \quad (2.4.)$$

Ağırlık: Bir bulanık kümenin ağırlığı evrensel küme elemanlarının bulanık kümesindeki en yüksek üyelik derecesidir.

Normalleştirilmiş Bulanık Küme: Eğer bulanık kümenin destekleyen elemanlarından en az biri en yüksek üyelik derecesinde ise buna normalleştirilmiş bulanık küme denir.

α - kesim: Bir A kümesinin α - kesimi, belirli bir α değerine eşit ya da daha büyük üyelik derecelerini içeren A kümesinin destekleyen elemanlarını taşıyan crisp kümedir. Eğer eşitlik yoksa yani bütün üyelik dereceleri α 'dan daha büyükse ona güçlü α - kesim formu denir [FOLGER, 1988].

$$A_\alpha = \{ x \in X \mid A(x) > \alpha, \alpha \in [0,1] \} \quad (2.5.)$$

Seviye Kümesi: A bulanık kümesinin seviye kümesi A'nın α - kesimlerini oluşturan bütün seviyelerin kümesidir.

Kardinalite: A bulanık kümesinin skalar kardinalitesi sonlu X evrensel kümesinde tanımlanır ve denklem 2.6'da görüldüğü gibi X'in elemanlarının A bulanık kümesindeki bütün üyelik derecelerinin toplamıdır.

$$|A| = \sum_{i=1}^n \mu_A(x), x \in X \quad (2.6.)$$

Alt Küme: A bulanık kümesindeki X evrensel kümesinin bütün elemanlarının üyelik dereceleri B bulanık kümesinin üyelik derecelerine eşit ya da küçük ise A, B'nin alt kümesidir.

Bulanık Elemanlar: Bulanık üyelik dereceleri, reel sayılar uzayında tanımlanmış özel tip bulanık kümeleridir.

Tanım: Bir A bulanık sayısı R (Reel sayılarda) tanımlanmış aşağıdaki özellikleri sağlayan bir bulanık kümedir:

- i. A normal bir bulanık kümesidir.
- ii. A'nın bütün α - kesimleri yakınsaktır.
 $(n \in [0,1] \cap (x,y \in R) A [nx+(1-n)y] \geq \min (A_x, A_y)$
- iii. A üst yarı sürekli veya bütün α - kesimleri R'nin kapalı aralıklarındadır.
- iv. A'nın sınırlı destekleyeni vardır.

A: Bulanık sayısı "0'a yakın", "3 civarında " v.b. yaklaşık tanımların gösteriminde uygun bir modeldir. Bunlar, üyelik fonksiyonlarının şekline bağlı olarak bazı özel ek özelliklere sahiptir.

2.7. Boolean Matematiğine Karşı Bulanık Matematik

Bulanık mantık farklı mantık yapılarından dolayı Boolean mantığından farklı matematiksel işlemlere sahiptir [KLIR and YUAN, 1995].

Bu işlemler aşağıda açıklanmıştır:

A ve B : Bulanık değişkenleri, yani tanımlanan kümeler ,

$\mu_A(x)$: x değişkeninin A kümesindeki üyelik derecesi ,

$\mu_B(x)$: x değişkeninin B kümesindeki üyelik derecesi ,

AND : Minimum üyelik derecesi ,

OR : Maksimum üyelik derecesi ,

NOT : $1 - \mu$ olmak üzere bu operatörler

$$\mu(A \text{ AND } B) = \min(\mu_A, \mu_B),$$

$$\mu(A \text{ OR } B) = \max(\mu_A, \mu_B),$$

$$\mu(\text{NOT } A) = 1 - \mu_A, \text{ şeklinde belirlenmiştir.}$$

Diğer taraftan

$\subset$: Kapsam

$\cap$: Kesişim

U: Birleşim

• :Bulanık toplamı

max : Maksimum

min : Minimum olmak üzere özellikler aşağıda verilmiştir.

1. Kapsama (Inclusion):

$$A \subset B, \mu_A(x) \leq \mu_B(x)$$

2. Eşitlik (Equality):

$$A=B, \mu_A(x) = \mu_B(x)$$

3. Tersleme (Complementation):

$$\bar{A}=B, \mu_A(x) = 1-\mu_B(x)$$

4. Kesişim (Intersection):

$$A \cap B = \min(\mu_A(x), \mu_B(x))$$

5. Birleşim (Union):

$$A \cup B = \max(\mu_A(x), \mu_B(x))$$

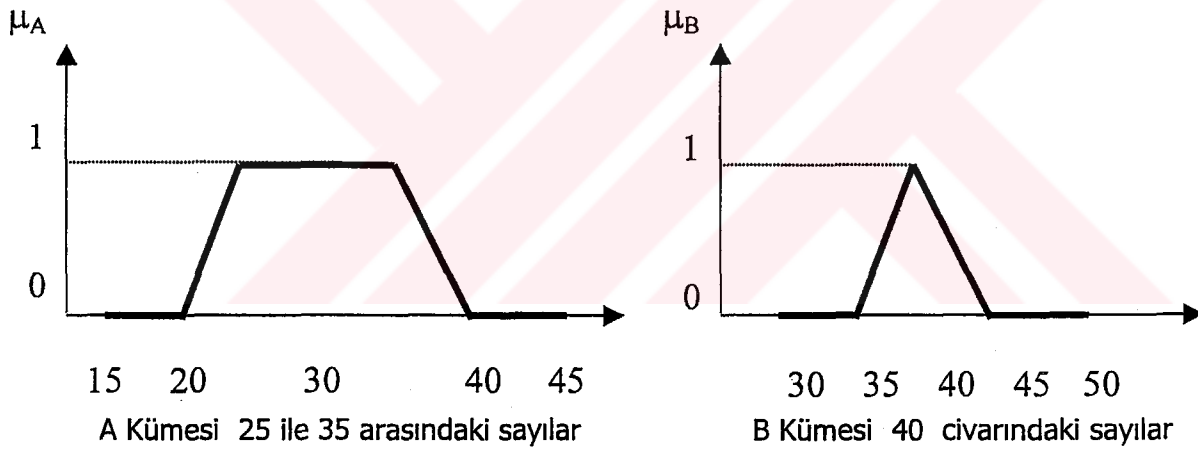
6. Toplam (Addition):

$$A \cdot B = (A \cup B)$$

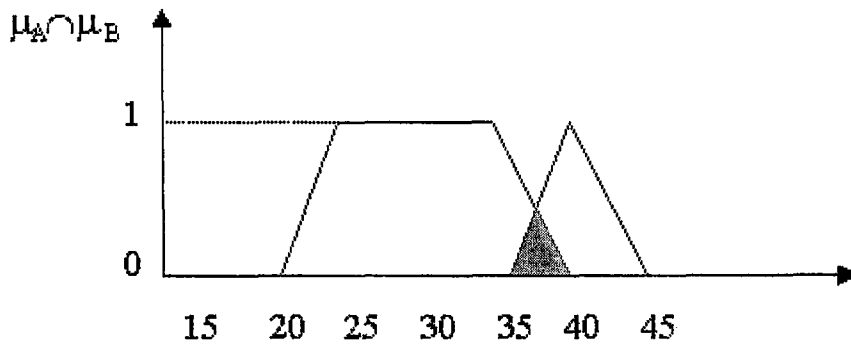
7. Çıkarma (Subtraction):

$$B - A = B \cap \bar{A}, \quad A - B = B - A \text{ ve } A - B = -(B - A)$$

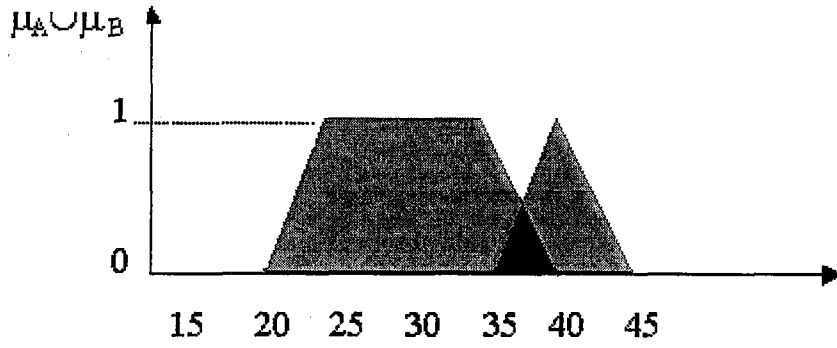
Kesişim (Bulanık AND), birleşim (Bulanık OR) ve tersleme işlemleri A ve B kümeleri için Şekil 2.6, Şekil 2.7, Şekil 2.8 ve Şekil 2.9'da gösterilmiştir.



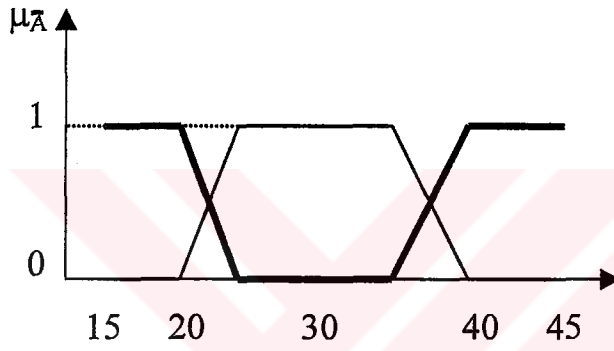
Şekil 2.6. A ve B kümeleri



Şekil 2.7. A ve B kümesinin kesişimi



Şekil 2.8. A ve B Kümesinin Birleşimi



Şekil 2.9. A Kümesinin Terslenmişi

8. Matematiksel İşlemler

Değişme Özelliği (Commutativity):

$$a \cap b = b \cap a, a \cup b = b \cup a$$

Birleşme Özelliği (Associativity):

$$(a \cup b) \cup c = a \cup (b \cup c)$$

$$(a \cap b) \cap c = a \cap (b \cap c)$$

Eşit Ağırlık (Idempotence):

$$a \cap a = a, a \cup a = a$$

Dağılma Özelliği (Distributivity):

$$(a \cup b) \cap c = (a \cap c) \cup (b \cap c)$$

$$(a \cap b) \cup c = (a \cup c) \cap (b \cup c)$$

Hakimiyet (Dominance)

$$a \cup 0 = a$$

$$a \cap 0 = 0$$

$$a \cup 1 = 1$$

$$a \cap 1 = a$$

Çift Değilleme (Double Negation):

$$\overline{\overline{a}} = a$$

De Morgan Teoremi

$$\overline{a \cap b} = \overline{a} \cup \overline{b}$$

$$\overline{a \cup b} = \overline{a} \cap \overline{b}$$

Tersleme (Complementation):

$$\overline{a} = 1 - a$$

Bulanık matematik verilen bütün özellikleri aşağıda gösterilen iki önemli fark dışında boolean cebrinin özellikleriyle aynıdır.

$$9. a \cap \overline{a} \neq 0$$

$$10. a \cup \overline{a} \neq 1$$

9. ve 10. özellikleri bulanık ile crips mantık arasındaki farkları ortaya çıkarır.

2.8. Bulanık İlişkiler

Crips kümeler iki ya da daha çok kümede belirli bir ilişkinin varlığını veya yokluğunu gösterir. Bulanık ilişkilerde crips kümeler, $x_1, x_2, \dots, x_n$ kartezyen çarpımında tanımlanmış bulanık kümeleridir. Bu yüzden üyelik fonksiyonu tanımlanan ilişkinin kuvvetini gösterir. Bulanık ilişkiler şekil tanıma, kontrol ve tıp gibi bir çok alanda uygulama imkanı bulmuştur.

I. Temel Prensipler: X ve Y kümeleri arasındaki bulanık ilişkisi X ve Y'nin direkt kartezyen çarpımı

$$X * Y = \{(x,y) \mid x \in X, y \in Y\} \quad (2.7.)$$

şeklinde tanımlanmış bir bulanık kümedir. Bulanık ilişkiler üyelik fonksiyonlarıyla karakterize edilir. Üyelik fonksiyonu μ_R ise aşağıdaki şekilde gösterilir:

$$\mu_R : X * Y \Rightarrow [0,1] \quad (2.8.)$$

Bulanık ilişkileri tablo şeklinde de gösterilebilir. Tabloda bu elemanların üyelik dereceleri yer alır. Bunun yanısıra üyelik matrisi de kullanılır.

Örneğin: $X=(a,b,c)$ ve $Y=(1,2)$ kümeleri arasında R ilişkisi aşağıdaki gibi ifade edilsin:

$$R=0,3/(a,1)+1/(a,2)+0,8/(b,1)+0,1/(c,2) \quad (2.9.)$$

Farklı gösterilimler Şekil 2.10, Şekil 2.11, Şekil 2.12 ve Şekil 2.13'de görülmektedir.

i. Üyelik Matrisi:

	1	2
a	0,3	1
b	0,8	0
c	0	0,1

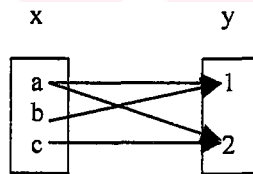
Şekil 2.10. Üyelik Matrisi

ii. Üyelik Tablosu:

x	y	$\mu_R(x,y)$
a	1	0,3
a	2	1
b	1	0,8
c	2	0,1

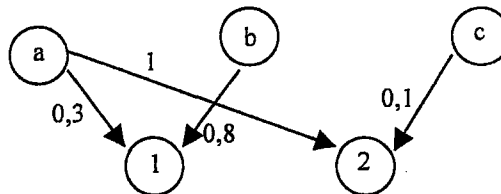
Şekil 2.11. Üyelik Tablosu

iii. Saggital Diyagramı:



Şekil 2.12. Saggital Diyagramı

iv. Üyelik Grafiği:



Şekil 2.13. Üyelik Grafiği

2.9. Tablo Oluşturulması

Klasik Mantık, karno haritası ve diğer metotları kullanarak fonksiyonları basitleştirir. Bu basitleştirmenin yapılabilmesinin temel sebebi

1. $a \cap \bar{a} = 0$
2. $a \cup \bar{a} = 1$ olmasıdır.

Zaten bu yapıdan dolayı çarpımların toplamı (sum of product) veya toplamların çarpımı (product of sum) şeklinde yazılmaktadır.

Bulanık küme teorisi yukarıdaki iki özellikleri, üyeliğin 0 ya da 1 olduğu durum hariç kabul etmeyen bir teoridir. Dolayısıyla klasik mantık gibi sadeleştirme yapılamaz. Ancak girdinin her durumu taranır ve ona göre çıktının ne olması gerektiği belirlenir. Bu sadeleştirmede kümelerin girdiler için üreteceği üyelik dereceleri karşılaştırılmaktadır.

Bir bulanık fonksiyonun sadeleştirilmesi:

$f = (a \cup \bar{a}) \cdot (\bar{a} \cup b \cup \bar{b})$ fonksiyonun sadeleştirilmesi Tablo 2.1' de gösterilmiştir.

Bulanık AND ve OR işlemlerinde giriş üyelik derecelerinin büyüklüklerinin birbirine göre olan durumları çıkışı belirlediği için girişlerin bütün farklı büyüklük sıralamaları gözönüne alınarak tablo oluşturulmuştur.

Dört adet giriş, büyüklüklerine göre sekiz farklı şekilde sıralanabileceklerinden bu fonksiyon sekiz farklı bölge için incelenmiştir.

Tablo 2.1. Fonksiyon Sadeleştirilmesi

$\geq \geq \geq$	$a \cup \bar{a}$	$\bar{a} \cup b \cup \bar{b}$	f
$a \ b \ \bar{b} \ \bar{a}$	a	b	b
$a \ \bar{b} \ b \ \bar{a}$	a	$\bar{b}$	$\bar{b}$
$\bar{a} \ b \ \bar{b} \ a$	$\bar{a}$	$\bar{a}$	$\bar{a}$
$\bar{a} \ \bar{b} \ b \ a$	$\bar{a}$	$\bar{a}$	$\bar{a}$
$b \ a \ \bar{a} \ \bar{b}$	a	b	a
$b \ \bar{a} \ a \ \bar{b}$	$\bar{a}$	b	$\bar{a}$
$\bar{b} \ a \ \bar{a} \ b$	a	$\bar{b}$	a
$\bar{b} \ \bar{a} \ a \ b$	$\bar{a}$	$\bar{b}$	$\bar{a}$

Bu noktadan bulanık mantığın, boolean mantık ile ayrıldığı bir başka özellik daha vardır. Kısaca özetlemek gerekirse, bir bulanık mantık fonksiyonunun çıkışını incelemek için Karnough haritası yerine, girişlerin birbirine göre büyük sıralamaları dikkate alınarak her durum için çıkışın ne olacağını gösteren bir tablo gerekmektedir.

2.10. Üyelik Fonksiyonlarının Belirlenmesi

Bulanık küme ile modellenen olaylar veya nesneler genel durumdan bağımsızdır. Uygulamada içerik anlamdan farklıdır. Örneğin bir endüstriyel kurumun sıcaklık kontrolünde “az pozitif”in anlamı bir evin oda sıcaklığının kontrolünden çok farklıdır. İlk durumda 100 °C’lik bir değer küçük pozitif bulanık kümesinde derecesi “0”dan büyük olabilir. Fakat ikinci örnekte bu sıcaklık değeri kabul edilebilir “AZ SICAK” kavramı içerisinde değildir. Bir problemi çözmek için uygun ölçekleme kullanılmalıdır. Sıkça kullanılan ölçekleme metodları Thole ve Zimmerman’ın mutlak ölçek, Saaty’nin oran ölçeği ve Gougen’in sıralı ölçeğidir. Uygun ölçekleme metodunu seçme, problem uygulamasına bağlıdır.

Mühendislik uygulamalarında üyelik fonksiyonlarının belirlenmesinde en çok kullanılan yöntemler aşağıda verilmiştir [KARWOWSKI, 1986].

1. Uygulama konusunda çalışan bir uzman şu soruyu sorar: “m bulanık kümesi A tarafından belirtilen içerikle uyumlu görüntüyor mu?” Her uzman bu soruya evet veya hayır cevabı verebilir. Böylece A bulanık kümesinde x’in üyelik derecesi pozitif cevapların toplam (evet: pozitif cevaptır) cevaba oranı ile belirlenir. Eğer n sayıda uzman soruyu sorar ve n(x) kadar pozitif cevap verirse A’da x’in üyelik derecesi aşağıda verildiği gibi bulunur:

$$\mu_A(x) = n(x) / n \quad (2.10.)$$

2. Saaty’nin Çiftli Karşılaştırma Metodu: Bu metod çiftli karşılaştırmada oran ölçeği ile üyelik fonksiyonlarının tanımlanmasında kullanılır. Üyelik değeri oran formunda verilir. İlkönce üyelik dereceleri oranları

$$\{(\mu_i / \mu_j), i, j = 1, 2, 3, \dots, n\} \quad (2.11.)$$

kare matris şeklinde düzenlenir. Daha sonra bu matris

$\mu = [\mu_1, \mu_2, \dots, \mu_n]$ vektörü ile çarpılır ve aşağıdaki formda denklemler elde edilir:

$$\begin{aligned} [\mu_i / \mu_j] \mu &= n \cdot \mu \\ ([\mu_i / \mu_j] - n) \mu &= 0 \end{aligned} \quad (2.12.)$$

Böylece hem en büyük özdeğer (kalan sıfır) ve hem de μ ’ye karşılık gelen değer bulunur. Evrenin elemanları arasında çiftli karşılaştırma yöntemi derecelendirme değerleri tahmin etmek için kullanılır.

Bir araştırmacı, i. elemanı , j. elemana göre değerlendirmeye çalışır. Matris kullanarak üyelik fonksiyonunu hesaplamak için bazı nümerik yöntemler vardır. Bunlardan biri kareler toplamının minimizasyonu yöntemidir.

$$\min_{\mu} \sum_{i=1}^n \sum_{j=1}^n (a_{ij} \cdot \mu_j - \mu_i)^2 \quad (2.13.)$$

3. Olasılık karakteristiği yöntemi: Bu metotla olasılık fonksiyonunun üyelik fonksiyonuna çevrilmesinde birebir karşılaştırma uygulanır. Sürekli olmayan $P_1, P_2, \dots, P_n$ olasılıkları azalan bir düzende sıralandığında $P_1 > P_2 > \dots > P_n$ üyelik fonksiyonu değerleri aşağıdaki formülle ifade edilir.

$$\mu = 1 - \sum_{j=1}^{i-1} (p_j - p_i); \quad i \geq 1 \quad (2.14.)$$

Normalleştirme şartlarından dolayı $\mu_1 = 1$ alınır. Böylece

$$\mu_1 = i \cdot p_1 + \sum_{j=i+1}^n p_j \quad (2.15.)$$

P_i 'li kısımda herhangi bir düzeltme farzedilmediği zaman

$$\mu_1 = \sum_{j=1}^n \min(p_i, p_j) \quad (2.16.)$$

olur.

Olasılık fonksiyonu, üyelik fonksiyonu kullanılarak ilk denklem P_i için yazılıp sonuç olarak

$$p_i = \sum_{j=1}^n (1/j) \cdot (\mu_j - \mu_{j+1}), i = 1, 2, 3, \dots, n \quad (2.17.)$$

$$\mu_{n+1} = 0$$

elde edilir.

3. BULANIK KONTROL

3.1. Kontrol sistemleri:

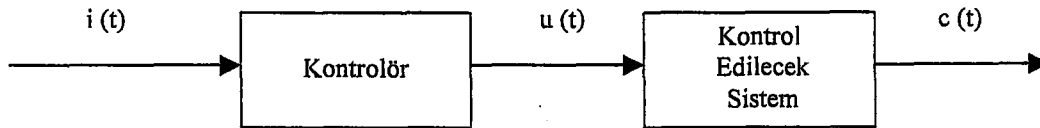
Bütün kontrol sistemlerinin asıl amacı, verilen zaman aralıklarında sistemin belirli durum ve değişkenlerini takip ederek sistemi istenilen durumda en az hata ile tutmak veya sistemi istenilen duruma getirmek için tanımlanmış kontrol fonksiyonlarını yerine getirmektir. Bir sistem, belirli bir görevin yerine getirilmesi için bir araya gelmiş çalışma ünitelerinden oluşmaktadır. Bir kontrol ünitesinden kontrol edilen sistem birkaç fiziksel elemandan meydana gelmiştir. Kontrol sisteminde girişler ve çıkışlar bulunmaktadır. Girişler denetleyicide işlenerek kontrol edilen sisteme gerekli çıkışları sağlamaktadır. Bu çıkış, sistemi kumanda etmek için kullanılan bir kontrol sinyali [SARIOĞLU, 1997].

İki tip kontrol sistemi vardır:

1. Açık döngü kontrol sistemi
2. Kapalı döngü kontrol sistemi

3.1.1. Açık Döngü Kontrol Sistemleri

Şekil 3.1'de verilen denetleyicinin çıkış değeri olan kontrol sinyali, kontrol edilen sistemin durumuna bağlı değildir.

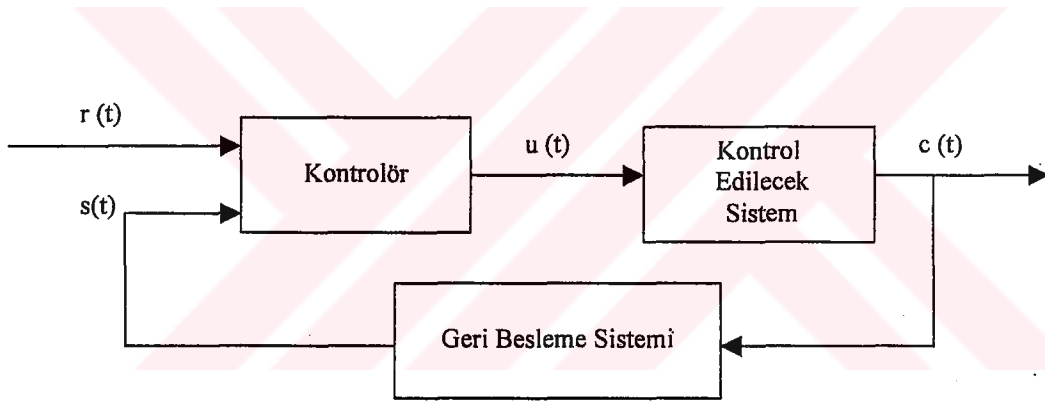


Şekil 3.1. Açık Döngü Kontrol Sistemi

Burada $i(t)$ denetleyiciye giriş ve $u(t)$ 'de denetleyiciden çıkıştır. $u(t)$ aynı zamanda kontrol edilen sistemin mevcut durumunu gösteren değerleri belirleyebilir. Görüleceği gibi kumanda sinyali $u(t)$, $c(t)$ 'den bağımsızdır. Bundan dolayı sistemi, istenilen duruma getirmek ve orada tutmak bu tip kontrol sisteminde mümkün değildir.

3.1.2. Kapalı Döngü Kontrol Sistemi

Şekil 3.2'de gösterildiği gibi bu tip kontrol sistemlerinde kumanda sinyali, kontrol edilen sistemin durumuna bağlıdır. Değişkenler veya durumlar örneklenerek bir geri besleme birimi üzerinden kontrol sistemine giriş olarak kullanılabilir. Başka bir giriş işareti de denetleyiciye referans sinyali olarak kullanılabilir.



Şekil 3.2. Kapalı Döngü Kontrol Sistemi

Burada kontrol ünitesinde bir karşılaştırmacı birim bulunmaktadır. Karşılaştırmacı devrenin iki girişi vardır. Bunların birisi referans sinyali $r(t)$. $r(t)$ sistemin taşıması gereken durumu ve değeri veya en az hatayla kalması istenen değeri verir. Karşılaştırmacı devrenin ikinci girişi $s(t)$ ise kontrol edilen sistemin çıkışından elde edilen geri besleme sinyalidir. Karşılaştırmacı devrenin çıkışı, denetleyicinin tek girişi olan hata sinyalini oluşturur.

$$e(t) = r(t) - s(t) \quad (3.1.)$$

Burada $e(t)$ hata sinyali, belirli bir zamanda kontrol sisteminin durumu ile istenilen durum arasındaki farkı verir ve hatayı azaltmak için denetleyici gerekli $u(t)$ çıkışını vererek sistemi istenilen duruma getirmeye çalışır. Geri besleme birimi denetleyici ünitesine $c(t)$ 'yi doğrudan transfer edebilir veya gerekli transformasyonları yaparak $c(t)$ sinyalinin türevini denetleyiciye iletir. Çeşitli tip kapalı döngü kontrol sistemleri vardır.

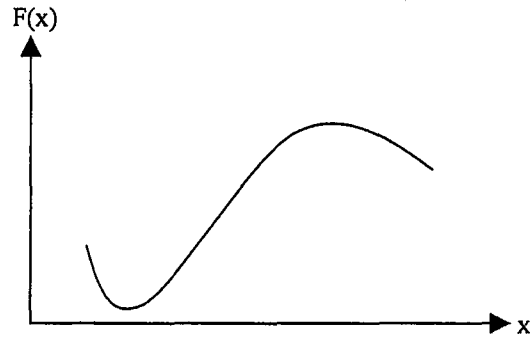
3.2. Kontrolde Bulanık Mantık

Kontrol sistemleri için en önemli nokta, giriş değerine karşı istenilen çıkış değerini üretebilmektir. Yapılan bu iş, giriş ve çıkış arasındaki fonksiyonları gerçekleştirmekten ibarettir. Günümüzde yapılan gerçeklemeler Şekil 3.3'de gösterilmiştir. Şekil 3.3(a)'da matematiksel fonksiyon olarak, Şekil 3.3(b)'de alışılmış program teknikleri kullanılarak veya arama tablosu oluşturularak ve Şekil 3.3(c)'de bulanık mantık küme ve terimleri kullanarak yapılmaktadır.

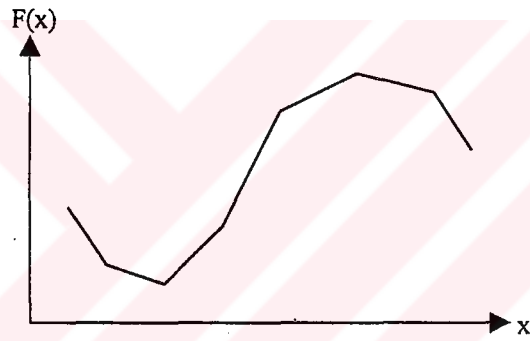
Bazen $y=f(x)$ fonksiyonu tam bir matematiksel model olarak ifade edilemez.

Bazen de sistemlerin kontrolü sırasında arama (look up) tabloları oluşturulur, bir fonksiyonun arama tablolarını oluşturulurken periyodu dar seçilirse giriş değerlerine karşı hafıza çıkış değerlerinin hazır olması gerekeceğinden çok büyük hafıza gereksinimi ortaya çıkacaktır. Tablonun büyük olması hem pahalıya neden olabilir, hem de kontrolün yavaş olmasına neden olur. Periyodun geniş aralıklarla seçilmesinde ise ara değerlerin olma olasılığı artacaktır. Ara değerlerde girilen giriş işaretleri sistemi kararsız yapar.

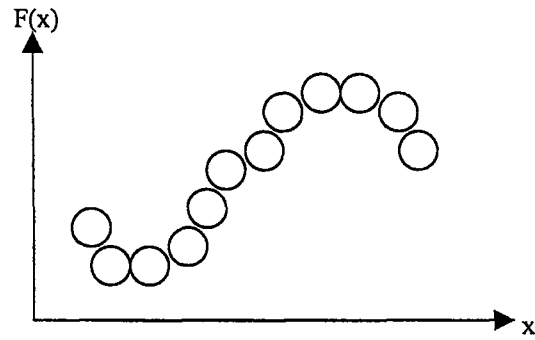
Bulanık mantıkta bölgesel olarak kümeler tanımlanır. Tanımlanan bu kümeler, girdinin bütün durumlarını kapsayacak yapıdadır. Kümeler daireler şeklinde oluşturulur ve maksimum üyelik derecesi merkezde bulunur. Girdi, bu merkezden hangisine daha yakın ise o kümede aksiyon üretilir. Bütün bu kümeler eşit büyüklükte ise bazı tekniklerle seçim yapılır.



Şekil 3.3(a). Matematiksel Model Kullanılarak Gerçekleştirilen Kontrol



Şekil 3.3(b). Arama Tablosu İle Gerçekleştirilen Kontrol

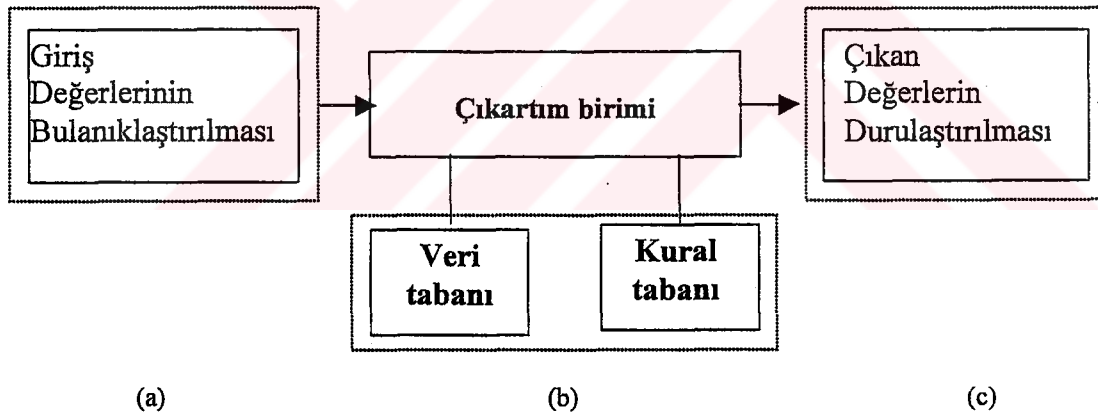


Şekil 3.3(c). Bulanık Mantıkla Gerçekleştirilen Kontrol

Bulanık yaklaşımlarında kompleks ve lineer olmayan ilişkilerin tanınmasını basitleştiren orantılama (built in interpolation) vardır. Bulanık mikrodenetleyici (Fuzzy Micro Controller, FMC) oluşabilecek her türlü giriş değeri için ayrı ayrı hesap yapmak yerine, girdi uzayını üyelik kümelerine böler ve girişlerin hangi kümede olup olmadığına bakar. Ayrıca bulanık kartları paralel mantık kullandığı için hem hızlıdır hem de programlama mantığına göre daha basittir [ITOH, 1993].

3.3. Genel Bir FMC'nin Çalışması:

Şekil 3.4'de gösterildiği gibi bir bulanık kart temel olarak üç kısımdan meydana gelir. Bunlar Bulanıklaştırma (Fuzzifier), Kontrolör (Controller) ve Durulaştırma (Defuzzifier) olarak isimlendirilmektedir [KRUSE, GEBHARDT and KLAUWONN, 1994].



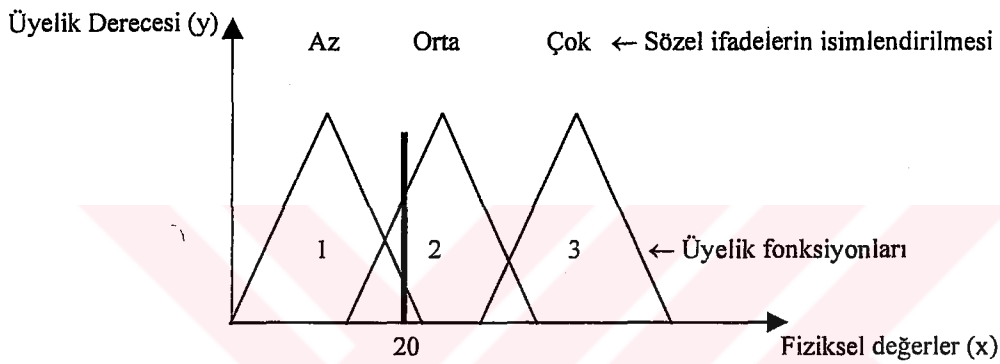
Şekil 3.4. Bir FMC Kartının Birimleri

3.3.1. Bulanıklaştırma (Fuzzifier)

Kontrol edilen sistem çıktılarından sayısal bilgilerin, sözel değişkenlere çevrildiği birimdir. Bu değişkenler bulanık denetleyicide bulanık kümeler tarafından tanımlanmaktadır. Bir başka deyişle bir büyüklüğü bulanık hale dönüştürmekle, sözel olarak ifade edilmesi sağlanır. Bulanıklaştırma, elde edilen crisp girdileri ile bulanık

kümelerini karşılaştırarak üyelik dereceleri üretir. Burada dikkat edilirse tanımlanan her küme, üyelik derecesinin oluşmasını sağlar. Üyelik fonksiyon sayısı kullanılacak FMC tarafından sınırlıdır.

Üyelik fonksiyonları, sistem değişkenlerini tanımlar. Üyelik fonksiyonlarının sayısına ve şekline ait bir kısıtlama yoktur. Bütünüyle tasarımcının istek ve tecrübesine bağlıdır. Şekil 3.5’de üç adet üyelik fonksiyonuna sahip değişken görülmektedir.



Şekil 3.5. Üyelik Fonksiyonları

Burada; x eksenini sistemden okunan ve kontrol edilecek olan sayısal değişkenleri ifade etmektedir. Bulanıklaştırma işlemi için 3 adet üyelik fonksiyonu kullanılmıştır. Bunların her birisi az, orta ve çok gibi sözel değişkenlerle ifade edilmiştir.

Sistemden okunan fiziksel değerler, x ekseninde bir noktaya karşılık düşmektedir. Bu noktaya karşılık gelen bir veya birkaç sözel ifade ismi bulunmaktadır ve bu sözel ifade ismine karşılık gelen üyelik derecesini oluşturmaktadır.

3.3.2. Kontrolör (Controller)

Bulanık kontrol sistemlerinde bilgi tabanına ilişkin kavramlar, bulanık mantık kontrolöründeki bulanık kontrol kurallarını ve bulanık verileri karakterize etmekte kullanılır. Bu kavramlar, tamamen uzmanların tecrübesine ve yargısına bağlıdır. Bulanık kümelerdeki üyelik fonksiyonlarının seçiminde ve uygulanmasında, başarılı bir sonuç çıkmasında önemli rol oynarlar.

Bilgi tabanı, veri tabanı ve kural tabanı olarak iki kısma ayrılır. Veri tabanı bulanık mantık kontrolöründeki verileri ve sözel kontrol kurallarına bağlı bilgilerin üyelik fonksiyonlarını tanımlar. Kural tabanı ise; uzman kişinin bilgileri ve tecrübelerinden yararlanarak oluşturulmuş bulanık kontrolörün davranışlarını belirleyen kontrol kurallarını belirler.

Veri tabanı hazırlanırken, sistemi etkileyen değişkenlerin tespiti ve bunların her biri için ayrı ayrı bulanık küme sayılarının belirlenmesi ve üyelik fonksiyonlarının belirlenmesi, tasarlanması gerekmektedir. Bu aşamada yine uzman tecrübeleri ve sayısal değerler kullanılır.

Bulanık mantıkta kontrolörün tasarlanmasında denetlenecek sistemin matematik modeline ihtiyaç duyulmamaktadır.

Bunun yerine o sistemi çalıştıran operatörün sistem davranışı konusunda sahip olduğu bilgiler ve tecrübesi ön plana çıkar. Tasarım sırasında genelde bu tür bilgilerden faydalanılmaktadır. Böyle bir yaklaşım uzun yıllar boyunca kazanılan tecrübelerin bulanık mantık denetleyicisi içerisinde yorumlanmış halde kolaylıkla yerleştirilmesine imkan vermektedir. Ancak kontrolör tasarımında belli kriterlerin olmaması tasarımı belirli bir otomasyona bağlamaya imkan vermektedir.

Bununla birlikte, bulanık kontrolörün en önemli kısmını oluşturan bulanık kontrol kurallarının üretilmesi için aşağıda verilen dört adet prensip yöntem uygulanmaktadır.

1. Bir uzmanın tecrübesi veya kontrol mühendisliği bilgileri,
2. Operatörün kontrol hareketlerinin modellenmesi,
3. Kontrol edilecek sistemin bulanık modeli,
4. Kuralların kontrolör tarafından öğrenilmesi.

3.3.3. Durulaştırma (Defuzzifier)

Son kısım elde edilen bulanık kontrol davranışlarını, bulanık olmayan sayısal kontrol işaretine dönüştürme işlemidir. Kontrolör, kural tablosunu dikkate alarak tanımlanan her bir kümedeki üyelik derecelerine göre yine üyelik dereceleri cinsinden bir çok aksiyonu tetikleyebilir. Bir vananın hız kontrolünde kullanıldığı varsayılın, “vanayı aç”, “vanayı kapa” veya “hiçbirşey yapma” olmak üzere üç aksiyonun da tetikleneceği kabul edilsin. Kontrolör 0,75 vanayı aç, 0,45 vanayı kapa, 0,2 hiçbirşey yapma olmak üzere 3 eylemi harekete geçirsin. Böyle bir durumda hangi aksiyonun gerçekleşmesi gerekir? Bu sorunu çözmek için bazı teknikler kullanmak gereklidir. Bu durum da aşağıda verilen yöntemler kullanılır [PATYRA, and MLYNEK, 1996]:

Maksimumların Merkezi Yöntemi:

Kural tablosu ile karşılaştırma yapıldıktan sonra, bir aksiyon üyelik derecelerine göre tetiklenecektir. Bu yöntem, üyelik derecesi en büyük aksiyonu seçer ve o aksiyon gereği elde edilen çıkış değerini de crips alanına aktarır. Eğer iki üyelik derecesi birbirine eşit çıkarsa hangi aksiyonu gerçekleştireceğini bilecektir. Bunun için kullanıcı tarafından tanımlanan kümelerde aksiyonların üyelik dereceleri eşit olmalıdır.

Küme tanımları, problemi tamamen çözecek yapıda olmalıdır. Tanımlanan kümeler vasıtasıyla girdi uzayının bütün durumları kapsanmalıdır. Eğer bu işlem gerçekleştirilemiyorsa donanım olarak girdinin tanımlanan kümelerde sıkıştırılması gerekecektir. Bu işlem de yapılamıyorsa elde bulunan FMC kartı problemi çözecek

yapıda değildir. En çok kullanılan yöntemlerden biri maksimumların merkezi metodudur.

Maksimumların Ortalaması Yöntemi:

Maksimumların ortalaması metodu; aktif olan kurallardan elde edilen maksimum kontrol işaretinin aritmetik ortalamasını kontrol işareti olarak almaktır.

Merkezi Ağırlık Yöntemi:

Burada yerel kontrol kural çıkışlarının ağırlık merkezi aşağıdaki formülle hesaplanır.

$$Z = \frac{\sum_{j=1}^n \mu_j(x_j) \cdot x_j}{\sum_{i=1}^n \mu_i(x_i)} \quad (3.2.)$$

Denklem 3.2’de Z: crips kontrol hareketini, n: seviyelerin sayısını, x_i : bulanık kümesinin destekleyenini, $\mu_i(x_i)$: bulanık kümesinde x_i ‘nin üyelik derecesini göstermektedir.

Merkezi ağırlık metodu, diğerlerine göre daha iyi ve güvenilir sonuçlar vermektedir.

3.4. Bulanık Kontrolör

Bulanık kontrolör, bulanık kontrol algoritması içeren bir birimdir. Tüm karar verme işlemleri bu birimde yapılır. Giriş değerleri alınarak bulanıklaştırılır buradan kural tabanına ve veri tabanına ulaşır. Kural tabanındaki kurallar değişkenlere bulanık kontrol tarafından uygulanır. Kontrolörün uygun kontrol hareketine karar vermesi ve bunu kontrol edilecek sisteme uygulamasının birkaç yöntemi vardır.

Yöntem 1) Bu metodu anlatmak için belirli bir kontrol kuralının uygulanmasına bakmak uygun olacaktır. Aşağıdaki formda bir grup kontrol kuralları olduğunu farz edelim.

Eğer $x \in A_i$ ise, $y \in B_i$ ise, $z \in C_i$ 'dir. $i = 1,2,3,\dots,n$

Burada n : kontrol kurallarının sayısı,

i : ise kural numarası,

A, B, C : Bulanık kümeleri,

x ve y : Giriş değişkenleri,

z : Kontrol parametresidir.

Böylece sistemin kontrol edilecek sistemden gelen iki girişi ve kontrol edilecek sinyali olan bir çıkışı vardır. Bu değerlerin daha önceki kurallara uygunluğu hesaplanmalıdır. Bu uygunluk ilgili bulanık kümesindeki üyelik derecesi ile gösterilir. Her kuralın uygunluğu aşağıdaki terimlerle gösterilebilir.

$$W_i = A_i(X_0) * B_i(Y_0) \quad (3.3.)$$

Denklem 3.3'de n : kural tabanındaki kuralların sayısını, W_i : çıkış değişkenini, $x=x_0$, $y=y_0$: giriş değişkenlerini, i : uzunluğu hesaplanacak olan kuralın numarasını verir.

Bu kural için aşağıdaki sonuç çıkarılır:

$$z \in W_i C_i \text{ 'dir ama } W_i C_i(z) = W_i * C_i(z) \quad (3.4.)$$

$$B = W_i C_i \cup \dots \cup W_n C_n \quad (3.5.)$$

Burada n : kontrol kuralı sayısıdır. Buradan crips kontrol hareketi belirlenebilir.

Yöntem 2) Tekdüze üyelik fonksiyonlu bulanık küme isimleri olan bulanık değişkenler için iyi bir metottur. Bu metodu açıklamak için iki kontrol kuralı örnek verilmiştir:

EĞER x N ise, y P ise, z N' dir.

EĞER x P ise, y N ise, z P' dir.

x ve y girdileri sırasıyla x_0 ve y_0 olsun. Önceliklerin uygunluğu ilk yöntemdeki gibi hesaplanır. W_1 ve W_2 olarak isimlendirilir. Bulanık olmayan Z_1 ve Z_2 sonuçları aşağıdaki gibi hesaplanır:

$$W_1 = N(Z_1), W_2 = P(Z_2)$$

Tüm sonuç ise:

$$Z = \frac{W_1 * Z_1 + W_2 * Z_2}{W_1 + W_2} \quad (3.6.)$$

şeklinde hesaplanır.

Yöntem 3) Kullanılan bulanık değişkenlerinin üyelik fonksiyonları trapezoid şeklindedir. Bu metodu açıklamak için her iki kuralda aşağıdaki şekilde tanımlanabilir.

EĞER x A_1 ise, y B_1 ise, $z = f(x,y)$

EĞER x A_2 ise, y B_2 ise, $z = g(x,y)$

EĞER $x = x_0$, $y = y_0$ için çıkış durumları W_1 ve W_2 olursa,

$$Z = \frac{W_1 * f(x_0, y_0) + W_2 * g(x_0, y_0)}{W_1 + W_2} \quad (3.7.)$$

şeklinde hesaplanır .

Bulanık kontrolü yukarıda açıklanan sonuç metotlarından biri kullanılarak kontrol hareketinden sonuca varılır.

3.5. Bulanık Mantık'ın Temelleri

Karmaşık kontrol sistemlerinin geliştirilmesinde bulanık mantık hızla günümüzün en başarılı teknolojilerinden biri olmaktadır. Bulanık mantık yardımıyla kompleks tasarım istekleri son derecede basit, kolay ve ucuz kontrolörlerle gerçekleştirilebilir. Aynı bulanık teknolojisi yaklaşık sebeplendirme şeklinde, bilgi teknolojisinde de yer almaktadır. Burada, karar destekleme ve minimum kurallarla çok güçlü sebeplendirme kapasitesi sağlayan gelişmiş uzman sistemler sağlamaktadır.

Tabii ki, bulanık mantık her kontrol problemi için en iyi yaklaşım değildir. Tasarımı yapan kişiler onun gücüne ve anlatım kabiliyetine baktıklarında onu nereye uygulayacaklarına karar vermelidirler. Aynı zamanda bu yeni teknoloji ile yapılan yazılım (software) projelerini nasıl yöneteceklerine karar vermelidirler. Proje yöneticileri, sistem mimarları ve tasarım mühendisleri aşağıdaki sorulara cevap aranmaktadır :

- Bulanık mantık kullanımından hangi tip projeler yararlanabilir ?
- Bu teknoloji ile projeleri nasıl kontrol edebilirim ?
- Bulanık mantık ve ilgili teknolojiler ile sistemleri tasarlama, geliştirme ve test etme konularında en iyi yol nedir ?

3.6. Bulanık Mantık'a Uymak

Bulanık mantık, analog süreçleri, sayısal bilgisayarda kolayca gerçekleştirmenin bir metodudur. Bu süreçler, sürekli olan ve kolayca ayrık (discrete) parçalara ayrılmayan ve matematiksel modellenmesi çok zor olan, bazen de imkansız olanlardır. Bir otomobilin ABS sistemi buna örnek olabilir. Bu tipteki bir problem için kontrol kuralları (denklemleri) şöyle değişkenler içerebilir:

Araba hızı, fren basıncı, fren sıcaklığı, fren uygulama aralığı, arabanın gecikmeli hareketinin doğrusal hareketine olan açısı gibi değişkenlerin hepsi sürekli ve değerlerinin aralıkları sistem tasarımcısının çevirisine bağlıdır.

Sıcaklık değişkenlerinin, değişik durum aralıkları olabilir : soğuk, serin, orta, ılık, sıcak, çok sıcak. Bu durumda bir aralıktan diğerine geçiş kesin olarak tanımlanmamıştır. Hiç bir noktada 1/10'luk bir derece artışı aralıklar arası geçişi sağlayamaz. Yani neyin soğuk, neyin ılık, neyin sıcak olduğunu değişik uzmanların değişik noktalardaki değişik yorumlarına bağlıdır.

Bu durum, sürekli sistem modellenmesinin ana temelidir ve bulanık mantığın güç ve esnekliğinin kalbidir. Bulanık mantık ile kontrol komutları bu kesin olmayan ve değişkenin durumunu tam olarak belirlemeyen cümleler şeklinde yazılır. Örneğin, bir ABS sisteminde bir bulanık kuralı şöyle olabilir :

Eğer fren sıcaklığı ILIK ve hız ÇOK HIZLI DEĞİL ise fren basıncını BİRAZ AZALT.

Oysa bir PID (Proportional Integrative Derivative) kontrolör burada çok spesifik olmalıdır.

- Eğer fren sıcaklığı > 280 ve hız < 45 ise fren basıncını 190 yap.

Bulanık mantık kuralı sadece daha tabii ve teknik olarak çok şey anlatıyor olmakla kalmayıp, daha geniş fren sıcaklığı ve araba hızlarında çalışacaktır. Değişkenlerin hangi derecelerde ILIK veya ÇOK HIZLI DEĞİL olarak kabul edildiği fren basıncının azaltılma derecesi olarak kabul edilecektir. Böylece bir bulanık mantık kuralı birçok alışılmış kuralın yerini tutabilir.

Bulanık mantık, bulanık kümeleri ve kuralları ile bir kontrol yüzeyi oluşturduğu için, tasarımcılara daha tam olarak sistemin matematiksel işlevini

anlamadan kontrolörler oluşturma imkanı sağlayacaktır. Neticede, ne zaman bulanık mantık kullanılması uygun olur sorusuna verilecek cevaplar aşağıdadır:

- Bir veya birden çok değişken sürekli olduğu zaman,
- Sürecin matematiksel modeli olmadığı veya matematiksel modeli olup da, modele göre kodlama yapmak imkansız olduğu veya eş zamanlı işlemlerde kullanılmayacak kadar kompleks bir ifade olduğu veya chip yapısında çok fazla hafıza gerektirdiği zamanlarda,
- Çok gürültülü ortamlarda ve düşük hassasiyetli mikrokontrolörlerin kullanılması zorunluluğunda,
- Sistem davranışının altındaki ilkeleri belirleyebilen ve her bir değişkenin karakteristiğini belirten bulanık küme çıkarabilecek bir uzman olduğu zaman bulanık mantık başarıyla uygulanabilir.

3.7. Bulanık Modeli

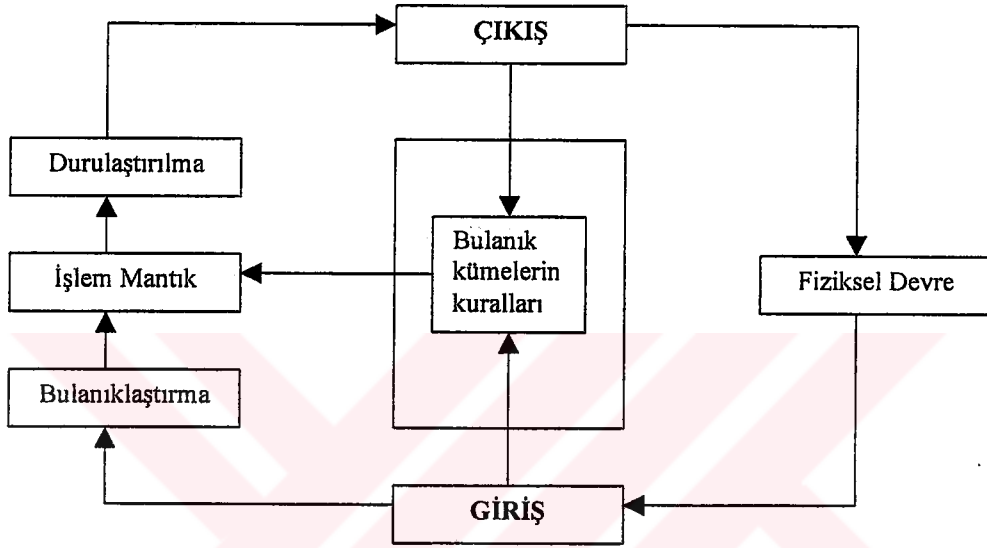
Geleneksel ve bulanık mantık sistemlerinin bileşenleri birbirine benzer. Aradaki fark bulanık sistemlerinin bulanıklaştırma içermesidir ki, bunlar girişleri bulanık alanına çevirirler. Durulaştırma çıkışları bulanık alanından crisp alanına çevirirler (nümerik değerlere). Bir bulanık sisteminde, bulanık girişe bütün kurallar uygulanır ve ona göre bir çıktı oluşturulur. Bu işlem yeni bir bulanık çıkışı veya çözüm değişkeni oluşturur.

Fiziksel sistemler için, çıkışın değeri sık sık gerçekleyicinin değerini düzeltmek için kullanılır ki böylece fiziksel sistemin seviyesi ayarlanmış olur. Algılayıcılar vasıtasıyla tepe değerleri değiştirilir ve bütün işlemler yeniden başlar.

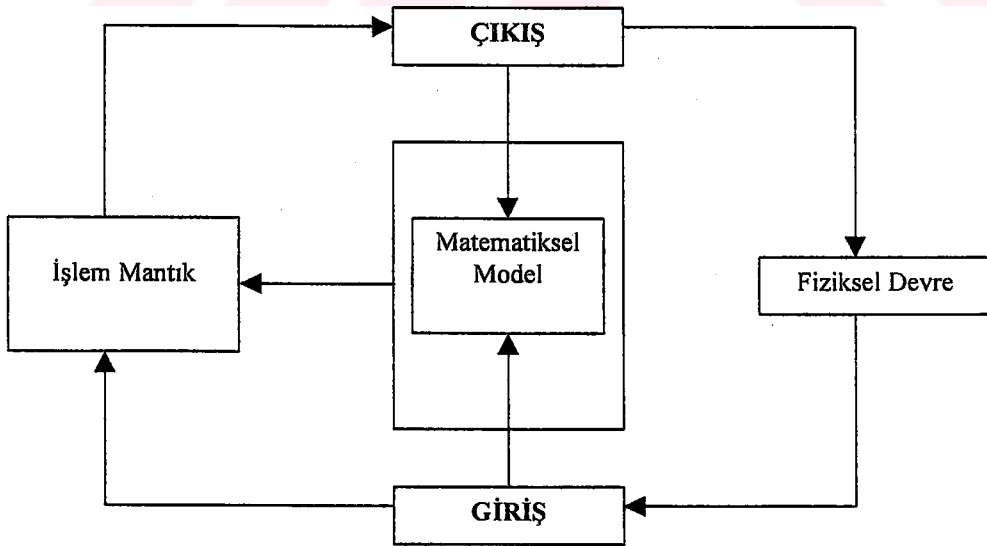
PID temelli bazı doğrusal işlemlerin matematiksel modeli Şekil 3.6'da gösterilmiştir. PID kontrolörler algılayıcı değerini okur, matematiksel modele

uygular ve matematiksel algoritmaya göre bir özel çıkış üretir. PID model bu sebepten dolayı uygulanıyor gibi görünse de asıl sebep daha çok ekonomiktir.

Bulanık kontrolörler daha kısa zamanda daha yüksek hassasiyet sağlarlar, Şekil 3.7’de bulanık kontrol şeması gösterilmiştir.



Şekil 3.6. PID Kontrol Şeması



Şekil 3.7. Bulanık Kontrol Şeması

4. BULANIK MANTIK İLE PROBLEM ÇÖZÜMÜNE BİR ÖRNEK

Eğer bir kap içerisindeki suyun kaynaması için gerekli süre hesaplanmak istenirse, burada öncelikli olarak bunu doğrudan etkileyen etkenleri ortaya çıkarmak gerekmektedir. Bunları, suyun miktarı, ilk sıcaklığı ve ısıtıcının gücü şeklinde sıralayabiliriz. Daha sonra bu değişkenlere ait kümelerin çıkarılması ve daha sonra kural dizinin tespiti gerekmektedir.

- Su miktarı için kümeler az, normal ve çok şeklinde ifade edilebilir.
- Suyun ilk sıcaklığı için düşük ve yüksek.
- Ocağın gücü için ise zayıf ve güçlü şeklinde tanımlanabilir.

Burada küme sayılarının fazla olması sistemin daha iyi incelenmesini sağlar, fakat bu yapılan hesaplamaların artmasına neden olacaktır. Küme sayısını tespit ederken ağırlıklı değişkenin veya süreksiz olan değişkenin küme sayısının fazla olmasına dikkat edilmelidir.

Aynı zamanda çıkış değişkenin de kümelerinin tespiti gerekmektedir.

Kaynama süresinin kümeleri;

çok kısa

kısa

orta

uzun

çok uzun

şeklinde tanımlanabilir.

4.1. Kural dizini

Değişkenlerin küme sayılarının çarpımı bize kural tablosunun büyüklüğünü verecektir.

Kural 1) Eğer kap içindeki su miktarı az, suyun ilk sıcaklığı düşük ve ocağın gücü zayıf ise suyun kaynama süresi **orta** olacaktır.

Kural 2) Eğer kap içindeki su miktarı az, suyun ilk sıcaklığı düşük ve ocağın gücü güçlü ise suyun kaynama süresi **kısa** olacaktır.

Kural 3) Eğer kap içindeki su miktarı az, suyun ilk sıcaklığı yüksek ve ocağın gücü zayıf ise suyun kaynama süresi **kısa** olacaktır.

Kural 4) Eğer kap içindeki su miktarı az, suyun ilk sıcaklığı yüksek ve ocağın gücü güçlü ise suyun kaynama süresi **çok kısa** olacaktır.

Kural 5) Eğer kap içindeki su miktarı normal, suyun ilk sıcaklığı düşük ve ocağın gücü zayıf ise suyun kaynama süresi **uzun** olacaktır.

Kural 6) Eğer kap içindeki su miktarı normal, suyun ilk sıcaklığı düşük ve ocağın gücü güçlü ise suyun kaynama süresi **orta** olacaktır.

Kural 7) Eğer kap içindeki su miktarı normal, suyun ilk sıcaklığı yüksek ve ocağın gücü zayıf ise suyun kaynama süresi **orta** olacaktır.

Kural 8) Eğer kap içindeki su miktarı normal, suyun ilk sıcaklığı yüksek ve ocağın gücü güçlü ise suyun kaynama süresi **kısa** olacaktır.

Kural 9) Eğer kap içindeki su miktarı çok, suyun ilk sıcaklığı düşük ve ocağın gücü zayıf ise suyun kaynama süresi **çok uzun** olacaktır.

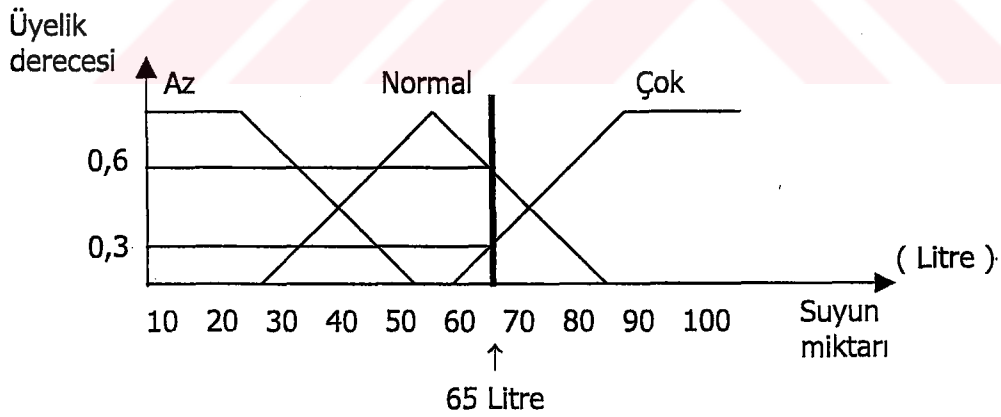
Kural 10) Eğer kap içindeki su miktarı çok, suyun ilk sıcaklığı düşük ve ocağın gücü güçlü ise suyun kaynama süresi **uzun** olacaktır.

Kural 11) Eğer kap içindeki su miktarı çok, suyun ilk sıcaklığı yüksek ve ocağın gücü zayıf ise suyun kaynama süresi **uzun** olacaktır.

Kural 12) Eğer kap içindeki su miktarı çok, suyun ilk sıcaklığı yüksek ve ocağın gücü güçlü ise suyun kaynama süresi **orta** olacaktır.

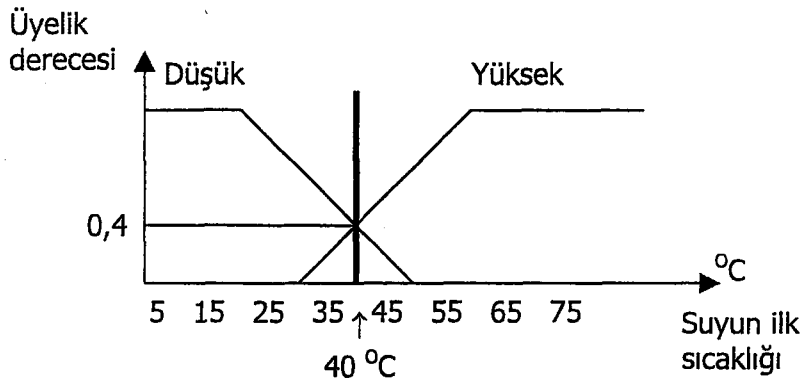
Daha sonra giriş değerlerinin daha önceden belirlenen kümelerde hangi üyelik derecelerine karşı geldiği bulunur.

Giriş değerlerimizin 40°C ilk sıcaklık, su miktarının 65 litre ve ocağın gücünün 1350W olduğunu kabul edelim.



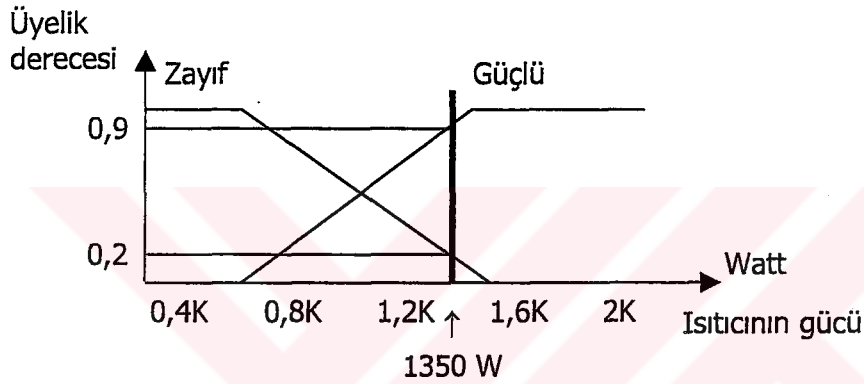
Şekil 4.1. Su Miktarı Üyelik Dereceleri

Su miktarı 0,3 oranında çok 0,6 oranında normaldir.



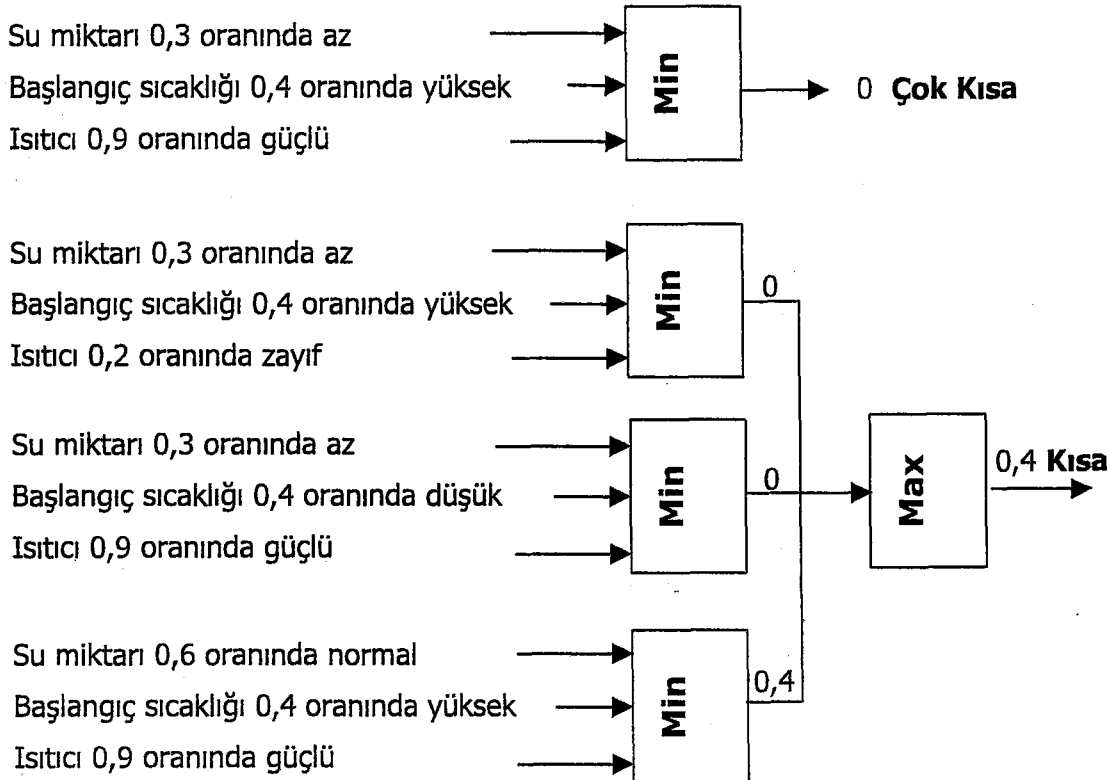
Şekil 4.2. Suyun Başlangıç Sıcaklığının Üyelik Dereceleri

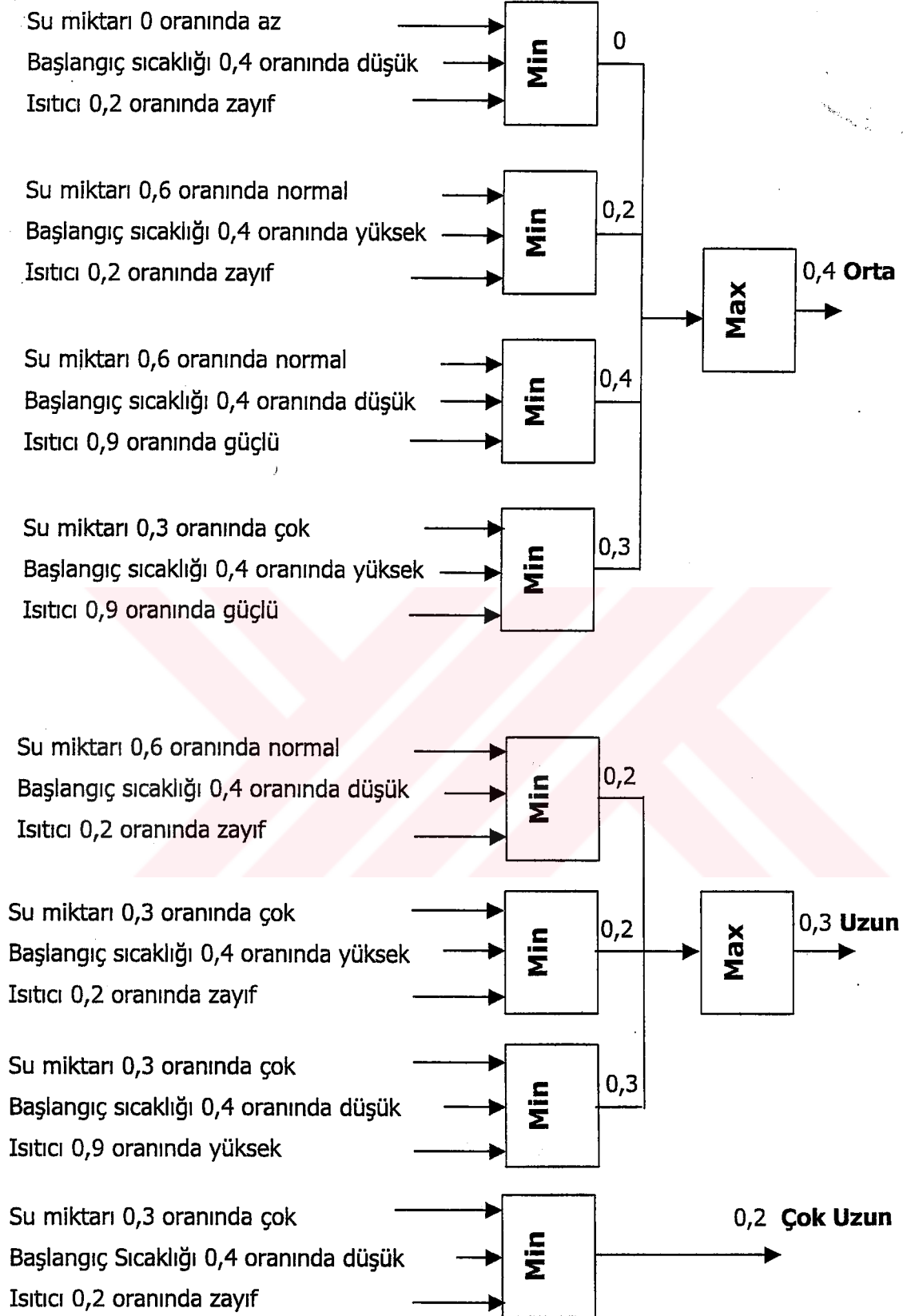
Suyun ilk sıcaklığı 0,4 düşük ve yine 0,4 oranında yüksektir.

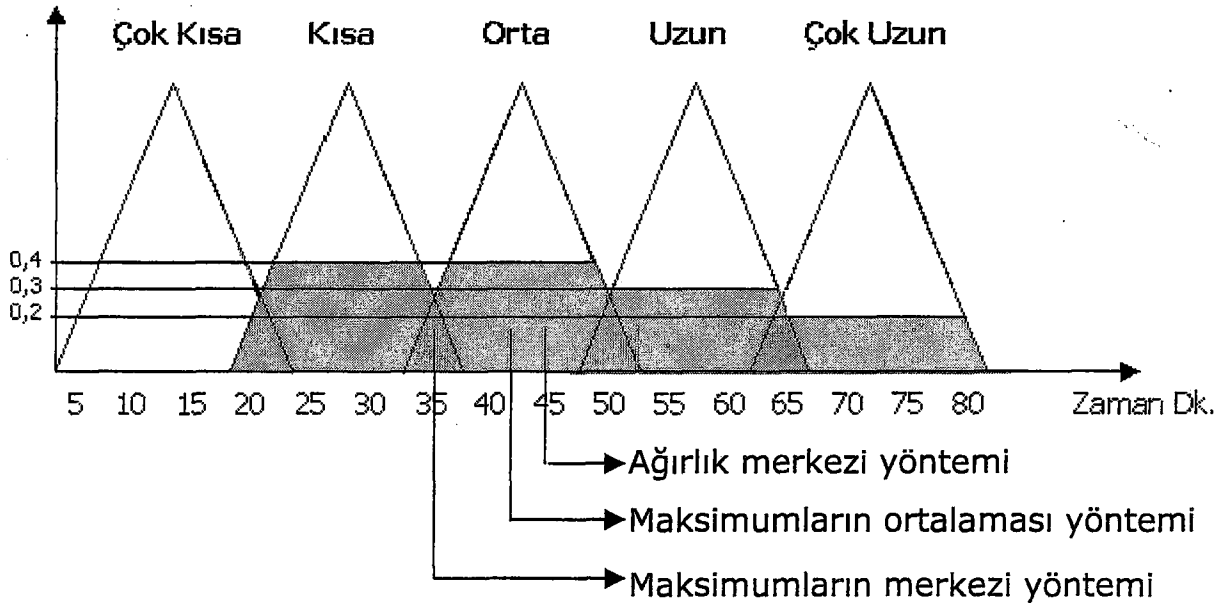


Şekil 4.3. Isıtıcı Gücü Üyelik Dereceleri

Isıtıcının gücü 0,2 oranında zayıf ve 0,9 oranında güçlüdür.







Şekil 4.4. Fonksiyonun Çıkış Değerinin Bulunması

5. UYGULAMA

Gelişen teknoloji ile birlikte insanların hayatlarını kolaylaştıran sistemler de artmıştır. Bu sistemlerin artmasının en büyük nedeni sistemlerin kontrolöründe alınan yoldur. İlk başlarda katı mantık (hard logic) kontrollü sistemler oluşturulmuştur; daha sonraları bu sistemler yerlerini esnek mantık (soft logic) kontrolüne bırakmıştır. Bunun başlıca nedeni önceden öngörülemeyen hataları tolere edebilmek ve daha esnek bir kontrol sağlayabilmektir [ESİN, 2000].

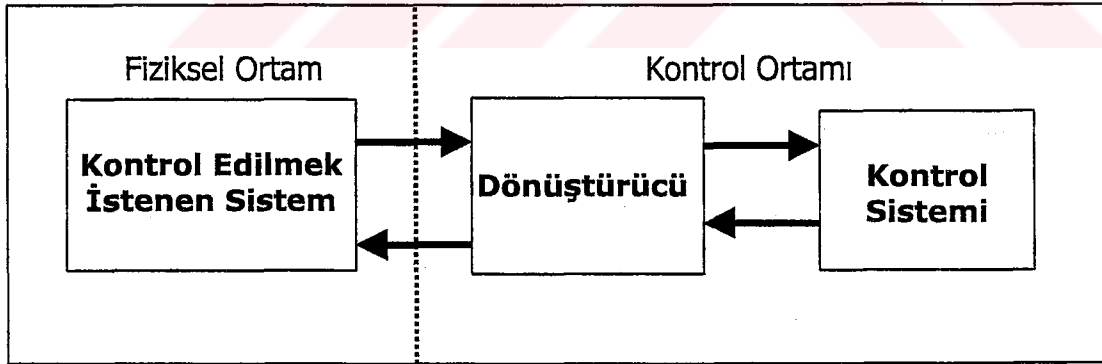
Özellikle kontrolörün esnek bir yapıya geçişi sistem kontrolüne zenginlik katmıştır. Kontrolörler bu şekilde hızlı, daha az hata yapan, kolay kullanılan ve daha fazla giriş çıkışa sahip bununla birlikte; daha karmaşık kontroller yapabilmeye başlamıştır.

Cihazların kontrolünde önceleri özel amaçlı elektronik kartlar kullanılmaktaydı; bu kontrol sistemini hızlandırmış ayrıca kontrol sistemine olan güveni de arttırmıştı. Bunun dezavantajı ise her kontrol sistemi için ayrı bir kart tasarlama zorunluluğu ve sistem üzerinde yapılacak değişikliklere izin vermemesiydi. Daha esnek yapıda kontrol sistemleri gerekmektedir. Yarı iletken teknolojisinin de gelişmesiyle mikrodenetleyicilerin kullanıldığı fonksiyonel devreler ortaya çıkmıştır. Bunlara programlanabilir mantıksal kontrolörler (Programmable Logic Controller, PLC) adı verilmiştir. Çok yaygın bir şekilde özellikle endüstriyel ortamlarda PLC'ler kullanılmaya başlanmıştır. PLC'lerin kendinden beklenildiği gibi hem hızlı hem de farklı sistemlere kolaylıkla adapte edilebilen bir yapısı vardı, fakat kullanıcı arayüzünü kullanmak zordu. Gelişen teknoloji ile birlikte bilgisayar fiyatlarının düşmesi kontrol alanında bilgisayarların kullanımı arttırmıştır. Bilgisayara takılan arabirim kartları sayesinde dışarıdan veriler okunabilmekte; ardından bu bilgisayarda işlendikten sonra yine arabirim kartı vasıtası ile kontrol işareti olarak dışarıya verilebilmektedir. PC'lerin avantajları, yüksek hızlardaki işlemcileri sayesinde yüksek seviyeli dillerle programlama olanağı, bir CRT ekrana sahip olduğu için operatöre kullanım kolaylığı sağlaması ve geniş RAM ve sabit disk gibi kayıt ortamlarının varlığı olarak sıralanabilir. RS-232, RS-485, CAN Bus, PCI Bus, ISA

Bus, SCSI, PCMCIA gibi dahili ve harici haberleşme standartlarına yönelik ek donanımlar ve bunlara ait yazılımlar mevcuttur. Tüm bu özellikleri nedeniyle bilgisayarlı kontrol sistemi çok daha hızlı, üzerinde değişiklik yapmasına rahatlıkla izin veren, çevre birim cihazları ve sistemleri ile birlikte kolaylıkla kullanılabilen bir kontrol sistemi haline gelmiştir.

Bu çalışmada hedeflenen bilgisayara ek donanım gerektirmeden bulanık mantık ile genel amaçlı bir kontrolör yapmaktır. Kapalı çevrim kontrol sistemleri iki ana bloktan meydana gelmişlerdir; bunlardan birisi sistemin kendisi, diğeri de sistemden gelen bilgileri değerlendirip sisteme geri besleme olarak veren kontrol kısmıdır.

Kontrol edilecek sistem üzerindeki değişimleri gözleyebilmek için algılayıcılar (sensör) kullanılır. Bu algılayıcılardan alınan bilgiler kontrol sistemine gelmeden önce kontrol sisteminin işlemesine uygun hale getirilirler. Sistemden gelen bilgiler kontrol kurallarına göre işlendikten sonra kontrol bilgisi eylemciye (actuator) uygun hale dönüştürülür ve eylemci hareketi sağlanır.



Şekil 5.1. Fiziksel Ortamın ve Kontrol Ortamının İlişkisi

Kontrol ortamında görüldüğü gibi iki basamak bulunmaktadır. Bunlardan birincisi sistemden gelen bilgileri işleyen ve kontrol edilecek sisteme bilgi çıkışı yapan dönüştürücü kısmı, ikinci ise kontrol kurallarının bulunduğu ve kontrol işaretlerini üreten kontrol sistemidir. Bu çalışmada dönüştürücü kısım için bir arabirim kartı ve kontrol için Windows işletim sistemi üzerinde çalışan bir yazılım yapıldı.

5.1. Arabirim Kartı

Bu kartın yapılmasının amacı fiziksel ortamdan alınan analog sinyallerin bir mikrodenetleyici aracılığıyla bilgisayar ortamına aktarılması, bilgisayarda bulunan yazılımla donanımdaki dijital çıkışlar yardımıyla kontrolün yapılmasıdır. Kontrolün bilgisayar ortamında yapılması görsel sunuş ve kullanım kolaylığı açısından büyük bir avantaj sağlayacaktır. Kontrol parametrelerinin ve algoritmalarının değiştirmesinde büyük esneklik sağlayacaktır.

Bir kontrol sisteminin etkinliğini belirleyen faktörlerden birisi de kontrol parametrelerinin değişen şartlara göre çok hızlı bir şekilde değiştirilebilmesidir. Analog kontrol devrelerinde şartların değişmesi, doğrudan devredeki elemanların değerlerinin, hatta devre topolojisinin değişmesini gerektirir.

Mikro-işlemci ve mikrobilgisayar teknolojisinin gelişmesi, üretim adetlerinin artması ve buna bağlı olarak fiyatlarının düşmesi sayesinde endüstriyel uygulamalarda dijital kontrol sistemlerinin kullanımı hızla yaygınlaşmaktadır.

Mikrodenetleyiciler, işlem hızı, komut seti zenginliği, dahili yan birimler açısından çok geniş bir yelpazeye sahiptir. Bu, yapılacak uygulamaya en uygun, optimum performans/fiyat oranına sahip bir dijital kontrol sistemi tasarlanmasını sağlayacaktır. Ancak mikrodenetleyiciler için yazılım geliştirmek biraz daha zor bir mühendislik çalışması gerektirir.

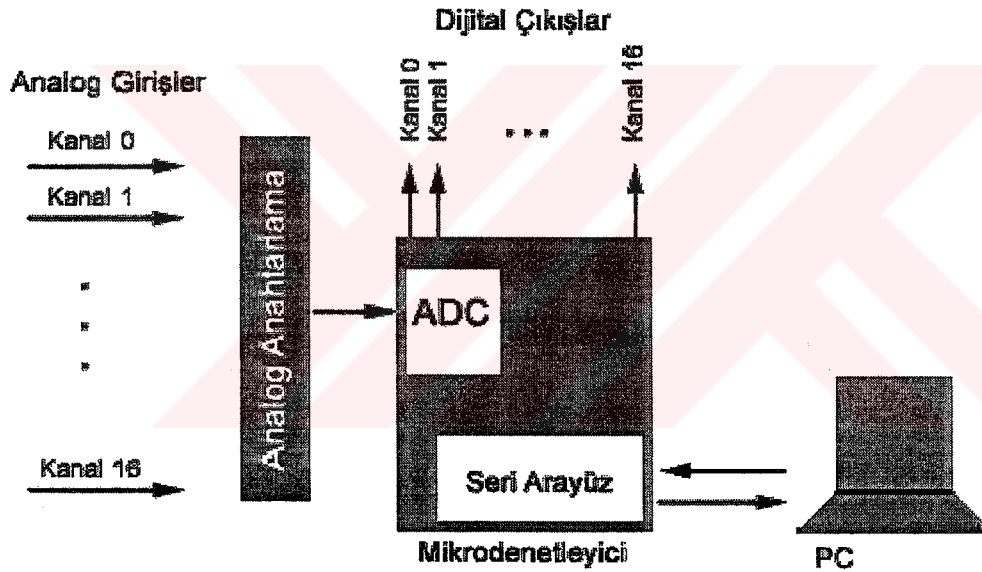
Arabirim kartının yapılma amacı, bir donanım vasıtasıyla 16 analog sinyali sayısallaştırıp bilgisayar ortamına iletmek ve bilgisayardaki yazılımla bu verileri işlemek, aynı donanım üzerinden dijital çıkışlarla kontrol gerçekleştirmektir. Sözü edilen 16 analog sinyal termokupl, termistör, potansiyometre, nem ölçer, yük hücresi, fotodiyot gibi sensörlerden alınan işaretler olabilir. Tabi ki bu sensörlerin herbirinin çıkış sinyallerinin genlikleri oldukça düşük ve değişik karakteristiklere sahiptir. Bir yükseltici ve analog sinyal işleme tekniği ile çıkış genliğinin minimum ve maksimum

değerleri uygun bir aralığa kaydırılmalıdır. Endüstriyel uygulamalarda en çok 4-20mA ve 0-10V standartları kullanılır.

Donanımın sahip olduğu dijital çıkışlar ısıtıcı, motor, fan gibi cihazları kontrol etmekte kullanılabilir. Bu çıkışların kullanılması için de güç transistörü veya röle gibi harici sürücü düzenekler kullanılması gerekecektir.

5.1.1. Arabirim Kartının Yapısı

Sistemin blok diyagramı Şekil 5.2’de gösterilmiştir.



Şekil 5.2. Arabirim Kartı Blok Diyagramı

Blok diyagramın belirlenmesinden sonra ilk olarak kullanılacak mikrodnetleyici seçilmiştir. Bu projede mikrodnetleyici olarak Microchip firmasının PIC16Fxxx serisinden PIC16F877 mikrodnetleyicisi seçilmiştir. Bu seriden bir mikrodnetleyicinin seçilmesindeki etkenler şunlardır:

- Bu mikrodnetleyicinin piyasada bulunmasının kolay olması.

- PIC16Fxxx serisi mikrodnetleyicilerin FLASH program belleğine sahip olması sayesinde defalarca ve hızlı bir şekilde programlanabilmesi.

Bu serinin içinden özellikle PIC16F877 mikrodnetleyicisinin seçilmesinin sebepleri ise şunlardır:

- Dahili 10bit'lik bir analogdan sayısala çeviriciye (Digital to Analog Converter, DAC) sahip olması.
- Bir dahili evrensel senkron asenkron alıcı vericiye (Universal Synchronous - Asynchronous Receiver Transmitter, USART) sahip olması.
- I/O iskele (Port) sayısının fazla olması.(Böylece 16 çıkış flip-flop kullanmadan doğrudan verilebilmiştir.)

PIC16F877 RISC (Reduced Instruction Set Computer) tabanlı 8 bit bir çekirdeğe sahiptir. 35 adet komuta sahiptir ve bu komutların herbiri eşit sürede işlenir. Maksimum çalışma frekansı olan 20MHz'de bir komutun işlenme süresi 200ns'dir. Koşullu dallanma komutlarında komutla birlikte kullanılan koşul TRUE ise bu komutun işlenmesi ile atlamanın yapılması 400ns sürer.

Program belleği FLASH teknolojisi ile üretilmiştir ve 8Kx14word genişliğindedir. Program belleğine programlama esnasında programlayıcı cihaz yardımıyla erişilebileceği gibi programın kendisiyle, çalışma esnasında da okuma-yazma yapmak mümkündür. Böylece programdan arta kalan bölge veri kaydı okunması için kullanılabilir.

Özel amaçlı ve genel amaçlı saklayıcıların yer aldığı 368x8 byte'lık yaz/oku beleğe (RAM) sahiptir.

Bunların yanında 256x8 byte'lı EEPROM'u vardır. Bu belleğe erişim yavaş olsa da mikrodenetleyicinin içinde kullanıldığı cihazın kapatıldığı sırada kaybolmaması istenen verilerin kaydının tutulması için kullanılabilir.

Program belleğinde RESET vektörü 00H adresidir. Bu adreste yer alan komut RESET olayı gerçekleştikten sonra ilk işlenecek komuttur. Buraya yazılan komut asıl programın başladığı adrese dallanma komutu (goto) olmalıdır.

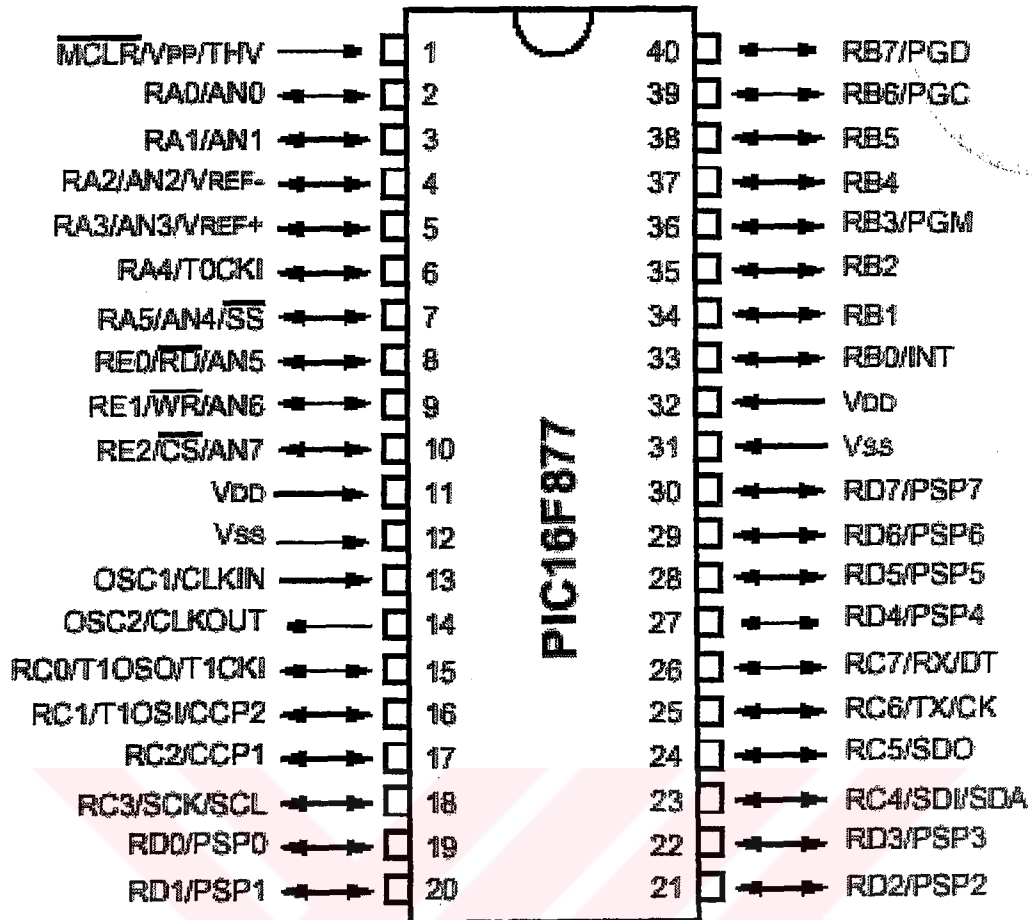
Kesme vektörü 04H adresini gösterir. 04H adresinde yer alan komut INTERRUPT gerçekleştiği zaman ilk işlenecek komuttur. Bu adresteki komut da kesme servis alt programına dallanma komutu olmalıdır. 14 ayrı kesme kaynağı vardır.

Doğrudan (Direct), dolaylı (Indirect) ve bağıl (Relative) olmak üzere üç çeşit adresleme modu vardır. Doğrudan adreslemede komutun opcode'unda işlenen olarak yer alan 7 bit'lik veri doğrudan 0-127 arasında bir adresi gösterir. Dolaylı adreslemede bir adres indeks saklayıcı (FSR) ve bu indeksin gösterdiği adresin içeriği olan data saklayıcı (INDF) olarak iki saklayıcı kullanılır. Bağıl adreslemeyi dallanma komutları kullanır.

PIC16F877'nin 33 adet I/O iskelesi vardır. Bunlar 6 adet iskele A'da, 8 adet iskele B'de, 8 adet iskele C'de, 8 adet iskele D'de ve 3 adet de iskele E'de olmak üzere ayrılmıştır. Her bir iskele ayrı ayrı giriş veya çıkış olarak yazılımla programlanabilir. TRISA, TRISB, TRISC, TRISD, TRISE saklayıcıları ilgili iskelerin giriş veya çıkış olarak programlanmasını sağlar.

Bu çalışmada kullanılan mikrodenetleyici devrenin fiziksel olarak gerçekleştirilmesinde kolaylık sağlaması için DIP kılıf içinde seçilmiştir.

PIC16F877'nin DIP kılıf içindeki görüntüsü Şekil 5.3'de gösterilmiştir.



Şekil 5.3. PIC16F877'nin Bacak Bağlantıları

PIC16F877'nin RAM bölgesi 4 ayrı BANK'a ayrılmıştır ve bu BANK'ların seçimi STATUS saklayıcısının RPO ve RP1 bitleri ile yapılmaktadır. Bir saklayıcı kullanılacaksa o saklayıcının içinde olduğu BANK'a geçilmelidir.

Tablo 5.1. PIC16F877 Hafıza Paylaşımı

Indirect addr. ^(*)	00h	Indirect addr. ^(*)	80h	Indirect addr. ^(*)	100h	Indirect addr. ^(*)	180h
TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h
PCL	02h	PCL	82h	PCL	102h	PCL	182h
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h
FSR	04h	FSR	84h	FSR	104h	FSR	184h
PORTA	05h	TRISA	85h		105h		185h
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h
PORTC	07h	TRISC	87h		107h		187h
PORTD ^(*)	08h	TRISD ^(*)	88h		108h		188h
PORTE ^(*)	09h	TRISE ^(*)	89h		109h		189h
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh
PIR1	0Ch	PIE1	8Ch	EEDATA	10Ch	EECON1	18Ch
PIR2	0Dh	PIE2	8Dh	EEADR	10Dh	EECON2	18Dh
TMR1L	0Eh	PCON	8Eh	EEDATH	10Eh	Reserved ⁽²⁾	18Eh
TMR1H	0Fh		8Fh	EEADRH	10Fh	Reserved ⁽²⁾	18Fh
T1CON	10h		90h		110h		190h
TMR2	11h	SSPCON2	91h		111h		191h
T2CON	12h	PR2	92h		112h		192h
SSPBUF	13h	SSPADDD	93h		113h		193h
SSPCON	14h	SSPSTAT	94h		114h		194h
CCPR1L	15h		95h		115h		195h
CCPR1H	16h		96h		116h		196h
CCP1CON	17h		97h	General Purpose Register 16 Bytes	117h	General Purpose Register 16 Bytes	197h
RCSTA	18h	TXSTA	98h		118h		198h
TXREG	19h	SPBRG	99h		119h		199h
RCREG	1Ah		9Ah		11Ah		19Ah
CCPR2L	1Bh		9Bh		11Bh		19Bh
CCPR2H	1Ch		9Ch		11Ch		19Ch
CCP2CON	1Dh		9Dh		11Dh		19Dh
ADRESH	1Eh	ADRESL	9Eh		11Eh		19Eh
ADCON0	1Fh	ADCON1	9Fh		11Fh		19Fh
	20h		A0h		120h		1A0h
General Purpose Register 96 Bytes		General Purpose Register 80 Bytes		General Purpose Register 80 Bytes		General Purpose Register 80 Bytes	
		accesses 70h-7Fh		accesses 70h-7Fh		accesses 70h-7Fh	
Bank 0	7Fh	Bank 1	FFh	Bank 2	17Fh	Bank 3	1FFh

STATUS saklayıcısının içeriği:

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	TO	PD	Z	DC	C
bit7							bit0

bit 7: IRP: Dolaylı adresleme BANK seçme biti

1 = Bank 2, 3 (100h - 1FFh)

0 = Bank 0, 1 (00h - FFh)

bit 6-5: RP1:RP0: Doğrudan adresleme saklayıcı BANK seçme bitleri

11 = Bank 3 (180h - 1FFh)

10 = Bank 2 (100h - 17Fh)

01 = Bank 1 (80h - FFh)

00 = Bank 0 (00h - 7Fh)

Her BANK 128 byte'tır.

bit 4: TO: Time-out biti

bit 3: PD: Power-down biti

bit 2: Z: Zero biti

1 = aritmetik veya lojik işlem sonucu sıfır

0 = aritmetik veya lojik işlem sonucu sıfır değil

bit 1: DC: Dijit Elde/Ödünç Biti (Nibble'lar için)

bit 0: C: Carry/borrow biti

PIC16F877'nin 10 bit'lik dahili bir analogdan sayısala çeviriciye sahiptir. Analog girişler iskele A ve iskele E'deki iskelelerde yer alır ve yazılımla bu iskelelerin analog giriş veya sayısal giriş-çıkış olarak programlanabilir. Bu programlama ADCON1 saklayıcısı ile yapılır.

ADCON1 saklayıcısının bitlerinin anlamları şu şekildedir:

U-0	U-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	—	—	—	PCFG3	PCFG2	PCFG1	PCFG0
bit7							bit0

Bit7: ADFM :A/D Result Format)

1: ADC Sonucu ADRESL ve ADRESH saklayıcılarında sağa yanaşık olarak yer alır.

0: ADC Sonucu ADRESL ve ADRESH saklayıcılarında sola yanaşık olarak yer alır.

Bit6-4: Kullanılmıyor

Bit3-0: PCFG3-PCFG0: A/D Port Configuration

16 ayrı iskele konfigürasyonu yapılabilir. Hangi iskelelerin dijital giriş-çıkış olacağı, hangilerinin ADC için analog giriş veya +/- referans olarak kullanılacağı bu bitlerle belirlenir.

Bu projede bu bitler 0100 olarak ayarlanmıştır.

Tablo 5.2. A/D İskele Ayarları

PCFG3: PCFG0	AN7 ⁽¹⁾ RE2	AN6 ⁽¹⁾ RE1	AN5 ⁽¹⁾ RE0	AN4 RA5	AN3 RA3	AN2 RA2	AN1 RA1	AN0 RA0	VREF+	VREF-	CHAN / Refs ⁽²⁾
0000	A	A	A	A	A	A	A	A	VDD	VSS	8/0
0001	A	A	A	A	VREF+	A	A	A	RA3	VSS	7/1
0010	D	D	D	A	A	A	A	A	VDD	VSS	5/0
0011	D	D	D	A	VREF+	A	A	A	RA3	VSS	4/1
0100	D	D	D	D	A	D	A	A	VDD	VSS	3/0
0101	D	D	D	D	VREF+	D	A	A	RA3	VSS	2/1
011x	D	D	D	D	D	D	D	D	VDD	VSS	0/0
1000	A	A	A	A	VREF+	VREF-	A	A	RA3	RA2	6/2
1001	D	D	A	A	A	A	A	A	VDD	VSS	6/0
1010	D	D	A	A	VREF+	A	A	A	RA3	VSS	5/1
1011	D	D	A	A	VREF+	VREF-	A	A	RA3	RA2	4/2
1100	D	D	D	A	VREF+	VREF-	A	A	RA3	RA2	3/2
1101	D	D	D	D	VREF+	VREF-	A	A	RA3	RA2	2/2
1110	D	D	D	D	D	D	D	A	VDD	VSS	1/0
1111	D	D	D	D	VREF+	VREF-	D	A	RA3	RA2	1/2

ADC ile ilgili diğer bir saklayıcı da ADCON0 saklayıcısıdır. Bu saklayıcının içeriği şöyledir:

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/DONE	—	ADON
bit7							bit0

bit 7-6: ADCS1:ADCS0: A/D Conversion Clock Select bits

ADC'nin clock frekansının seçimi içindir. Mikrodenetleyicinin kristal osilatörünün frekansının belirli oranlarda bölünerek ADC'ye verilmesini sağlar.

00 = FOSC/2

01 = FOSC/8

10 = FOSC/32

11 = FRC (saat bir RC osilasyonundan sağlanır)

bit 5-3: CHS2:CHS0: Analog Channel Select bits

ADC'nin mümkün olan 8 girişten hangisini kullanacağını seçmek içindir.

000 = Kanal 0, (RA0/AN0)

001 = Kanal 1, (RA1/AN1)

010 = Kanal 2, (RA2/AN2)

011 = Kanal 3, (RA3/AN3)

100 = Kanal 4, (RA5/AN4)

101 = Kanal 5, (RE0/AN5) (1)

110 = Kanal 6, (RE1/AN6) (1)

111 = Kanal 7, (RE2/AN7) (1)

bit 2: GO/DONE: A/D Conversion Status bit

Bu bit yazılımla 1 yapılarak A/D çevrimi başlatılır. Çevrim bittiğinde mikrodenetleyici bu biti tekrar 0 yapar. PIC16F877'de 20MHz kristal osilatörüyle elde edilebilecek minimum örnekleme periyodu 17µs'dir ve bu komut işleme süresi olan 200ns'ye göre

oldukça uzun bir zamandır. Bu yüzden A/D çevrimi bu bit 1 yapılarak başlatılmalı, sonuç okunmadan önce bu bitin tekrar 0 olması beklenmelidir.

bit 1: Kullanılmıyor.

bit 0: ADON: A/D On bit

1 = A/D çeviriciyi açar.

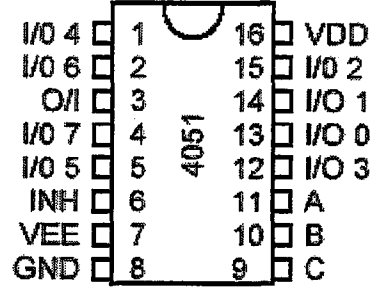
0 = A/D çeviriciyi kapatır ve güç tasarrufu sağlar.

ADRESL ve ADRESH saklayıcıları A/D çevriminin sonucunun saklandığı saklayıcılardır. ADCON1 saklayıcısı ile A/D sonucunun sağa yanaşık olması seçildiğinde 10 bit'lik çevrim sonucunun alt 8 bit'i ADRESL saklayıcısında, diğer 2 bit'i ADRESH saklayıcısının en düşük anlamlı iki bitinde yer alır.

Bu çalışmada toplam 16 analog girişten veriler alınması öngörülmüştür. Ancak PIC16F877 yazılımla seçilebilen 8 analog girişe sahiptir. Bu yüzden harici bir analog çoklayıcı kullanmak gerekmiştir. Bu işlem için 4051 entegresi kullanılmıştır. 4051 8 analog girişe ve 1 analog çıkışa sahiptir. A,B,C adlı lojik kontrol uçlarıyla bu 8 girişten istenilen bir tanesi çıkışa aktarılır. 16 analog giriş için iki adet 4051 kullanılmış ve bunların çıkışları PIC16F877'nin iki analog girişine (AN0 ve AN1) verilmiştir. İki 4051'in de kontrol uçları ortak bağlıdır ve PIC16F877'nin E iskelesiyle kontrol edilmektedir. Sistem çalışırken, analog veri alınmak istendiğinde önce AN0 veya AN1 uçlarından birisi yazılımla ADC giriş kanalı olarak seçilmekte, daha sonra E iskelesine yazılan veri ile 4051'lerin istenilen kanalı seçilmektedir.

Inhibit girişi normal çalışma modunda toprağa bağlı olmalıdır. VEE analog sinyalin toprağıdır ve yüksek hassasiyet gerektirmeyen uygulamalarda uygun şekilde sayısal toprak ile birleştirilebilir.

4051 entegresinin DIP kılıf içindeki görüntüsü Şekil 5.3'de verilmiştir.



Şekil 5.4. 4051'in Bacak Bağlantısı

PIC16F877 dahili bir USART'a sahiptir. Senkron ve asenkron seri iletişimi donanım ile destekler. Bu projede bilgisayarla mikrodnetleyici arasında asenkron, tam duplex (aynı anda hem veri alınabilir, hem de veri gönderilebilir) seri iletişimi sağlanacaktır. PIC16F877'nin C iskelesinde yer alan RC7 ve RC6 uçları seri haberleşmede veri alma(RX) ve veri gönderme(TX) fonksiyonları için kullanılırlar. Bu çıkışlar TTL seviyesindedir.

USART'ın haberleşme hızı (baud rate) yazılımla ayarlanabilir. Mikrodnetleyicinin içindeki BRG (baud rate generator) devresi doğrudan SPBRG saklayıcısı ile programlanarak istenilen haberleşme hızı elde edilir. Düşük hız ve yüksek hız olarak iki mod vardır ve TXSTA saklayıcısında yer alan BRGH biti ile bu modlardan biri seçilir. Yüksek hız moduyla asenkron haberleşmede SPBRG saklayıcısında istenilen bir haberleşme hızını elde etmek için yazılması gereken değer şu şekilde hesaplanır:

$$\text{Baud Rate} = \text{FOSC} / (16 (X + 1)) \quad (5.1.)$$

Baud Rate : Elde etmek istediğimiz haberleşme hızı

FOSC : Mikrodnetleyicinin kristal osilatörünün çalışmakta olduğu frekans

X: SPBRG saklayıcısına yazılması gereken sayı.(0-255 arası)

Bu projede seri haberleşme hızı 115200 baud seçilmiştir. Bunun için SPBRG saklayıcısına yazılması gereken değer:

$$SPBRG = (FOSC / (\text{Haberleşme Hızı} \times 16)) - 1 \quad (5.2.)$$

$$FOSC = 20 \text{ MHz}$$

$$\text{Haberleşme Hızı} = 115200$$

formülüyle hesaplanırsa SPBRG saklayıcısına 10 sayısı yazılmasının gerektiği bulunur.

Veri göndermeyle ilgili USART ayarları TXSTA saklayıcısında yer alır:

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R-1	R/W-0
CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D
bit7							bit0

- bit 7: C: Clock Source Select biti
Asenkron modda kullanılmıyor.
- bit 6: TX9: 9-bit Transmit Enable biti
1 = 9 bit veri gönderme modu
0 = 8 bit veri gönderme modu. Bu projede 8 bitlik mod kullanılmıştır.
- bit 5: TXEN: Transmit Enable biti
1 = Veri gönderme aktif
0 = Veri gönderme inaktif
- bit 4: SYNC: USART Mode Select biti
1 = Senkron mod
0 = Asenkron mod
- bit 3: Kullanılmıyor
- bit 2: BRGH: High Baud Rate Select bit
Asenkron mod:
1 = High speed

0 = Low speed

Senkron mod:

Kullanılmıyor

bit 1: TRMT: Transmit Shift Register Status biti

1 = TSR boş

0 = TSR dolu

bit 0: TX9D: 9 bit veri gönderme seçildiğinde gidecek verinin 9. bitidir.

Veri almayla ilgili USART ayarları RCSTA saklayıcısında yer alır:

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-0	R-0	R-x
SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
bit7							bit0

bit 7: SPEN: Serial Port Enable biti

1 = Seri iskeleyi açar

0 = Seri iskeleyi kapatır

bit 6: RX9: 9-bit Receive Enable biti

1 = 9 bit veri alma modu

0 = 8 bit veri alma modu

bit 5: SREN: Single Receive Enable biti

Asenkron modda kullanılmıyor

bit 4: CREN: Continuous Receive Enable biti

Asenkron mod

1 = Ard arda veri almayı aktifleştirir.

0 = Ard arda veri almayı devre dışı bırakır.

bit 3: ADDEN: Address Detect Enable biti

bit 2: FERR: Framing Error bit
1 = Çerçeveleme hatası gerçekleşti
0 = Çerçeveleme hatası yok

bit 1: OERR: Overrun Error bit
1 = Overrun hatası gerçekleşti
0 = Overrun hatası yok

bit 0: RX9D: 9 bit alma modunda alınan verinin 9.biti bu bite yerleşir.

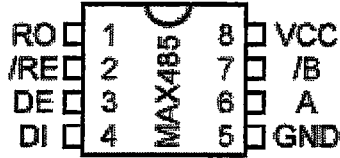
TXREG ve RCREG adlı saklayıcılar doğrudan gönderilecek verinin yazılacağı ve gelen verinin yer alacağı saklayıcılardır. TXREG saklayıcısı yazılmadan önce USART'ın çıkış tamponunun boş olup olmadığı PIR1 saklayıcısının TXIF bitine bakılarak anlaşılır.

Endüstriyel uygulamalar için RS232C seri haberleşme standardı uygun değildir. Bu standart hem gürültüye karşı fazla bağışıklı değildir, uzun mesafelerde haberleşme sağlayamaz hem de aynı kablo üzerinde birden fazla cihazın, hattı bir veri yolu gibi kullanması zordur. Bu tezde hazırlanan donanımın endüstriyel koşullarda uzun mesafelerde haberleşebilmesini sağlamak için özellikle fabrikalarda yaygın olarak kullanılan RS485 haberleşme standardı kullanılmıştır. RS485'te seri bilgi, bir akım döngüsü sinyali olarak burulu çift (twisted pair) kablo ile iletilir. Sinyalin "differential" sinyal olması gürültüye karşı oldukça dayanıklı olmasını sağlar. RS485 ile uygun kablo kullanılarak 1200 metreye 500kbps hızda veri iletimi öngörülür.

Bu tezde üzerinde mikrodenetleyicinin yer aldığı devre kartına bir çift RS485 hat sürücü entegre eklenerek RS485 haberleşme yapılması sağlanmıştır. RS485 hat sürücü olarak MAX485 (veya eşleniği 75176) entegreleri kullanılmış olup bunların bir tanesi veri almak için, diğeri veri göndermek içindir. Yani tam duplex bir RS485 haberleşmesi sağlanmıştır. Devre kartından iki çift haberleşme kablosu çıkar. Hattın

diğer ucunda yer alan bir RS485-RS232C çevirici devresi ile sinyaller bilgisayara iletilir.

MAX485 entegresinin DIP kılıf içindeki görüntüsü Şekil 5.4’de verilmiştir.



Şekil 5.5. MAX485’in Bacak Bağlantıları

RO : Receiver Output (/B ve A uçlarından alınmış differential RS485 Verinin TTL’e dönüştürülmüş halde çıkışı)

/RE : Receiver Enable (Bu uç LOW’a çekilerek entegre veri alma moduna geçirilir.)

DI : Driver Input (RS485’e Çevrilecek Verinin TTL Girişi)

DE : Driver Enable (Bu uç HIGH’a çekilerek entegre veri gönderme moduna geçirilir.)

Bilgisayar ve mikrodnetleyici arasındaki iletişim emir-rica şeklinde olur. Bilgisayar mikrodnetleyiciye bir byte’ı gönderir. Mikrodnetleyici de bu byte’a karşılık gelen işi yerine getirir. Bilgisayarın gönderdiği byte’ın içinde komut ve parametreleri kodlanmıştır. Bu yapıya sanal makine mimarisi (virtual machine architecture) denilmektedir çünkü burada mikrodnetleyiciye yazılımla tanımlanmış sanal komutlarla iş yaptırılmaktadır.

Bilgisayarın mikrodnetleyiciye gönderdiği emir byte’ının içeriği şöyledir:

Bit 7,6,5:	Komut	Bitler	Açıklama
	DATARQ	001	AD Çevrim Sonucu Gönder

OUTCTR	010	Dijital Çıkış Kontrol
LEDCTR	110	Aktivite LED'i Aç / Kapat komutu

Bit 4: DATARQ: Kullanılmıyor

OUTCTR: 0: Çıkış Low
1: Çıkış High

LEDCTR: Kullanılmıyor

Bit 3: DATARQ: Analog Multiplexer Entegresi Seçimi

0: Chip (Kanal) A
1: Chip (Kanal) B

OUTCTR: Çıkış İskelesi Seçimi
0: Dijital İskele A
1: Dijital İskele B

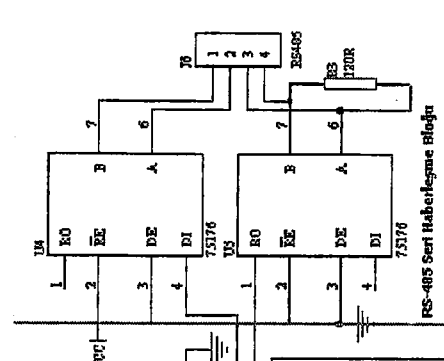
LEDCTR: LED Durumu:
0: LED'i Söndür
1: LED'i Yak

Bit2,1,0: DATARQ: Seçilen Multiplexer'den 8'e 1 Kanal Seçme

OUTCTR: Seçilen Çıkış İskelesinden 8'e 1 Kanal Seçme

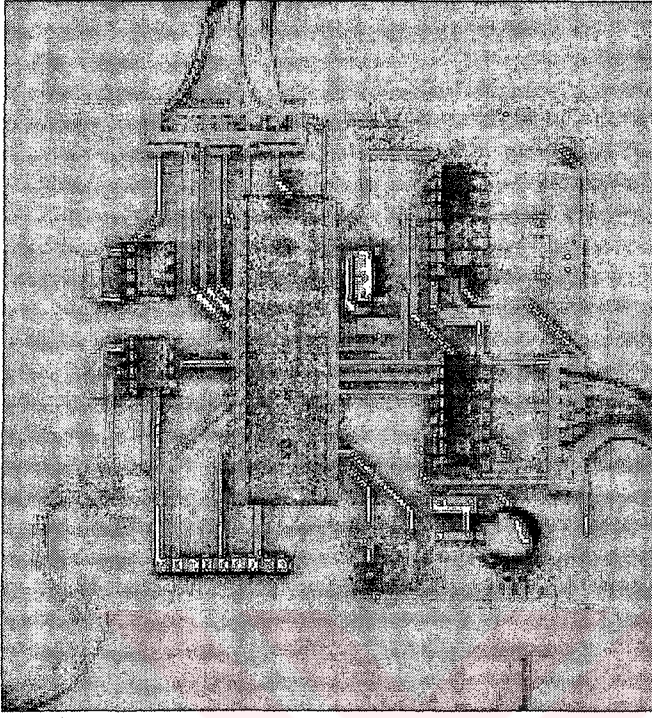
LEDCTR: Kullanılmıyor

Donanımın açık şeması Şekil 5.6'da gösterilmiştir.



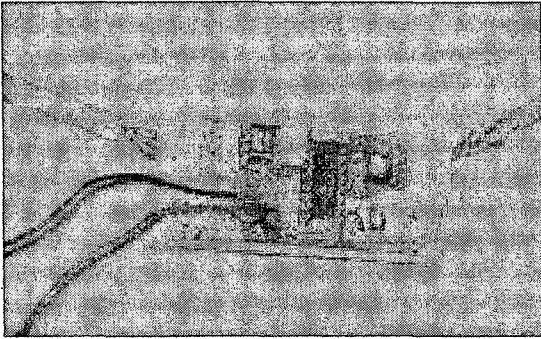
1

Yapılmış olan kartın fiziksel görüntüsü Şekil 5.7’de görülmektedir.



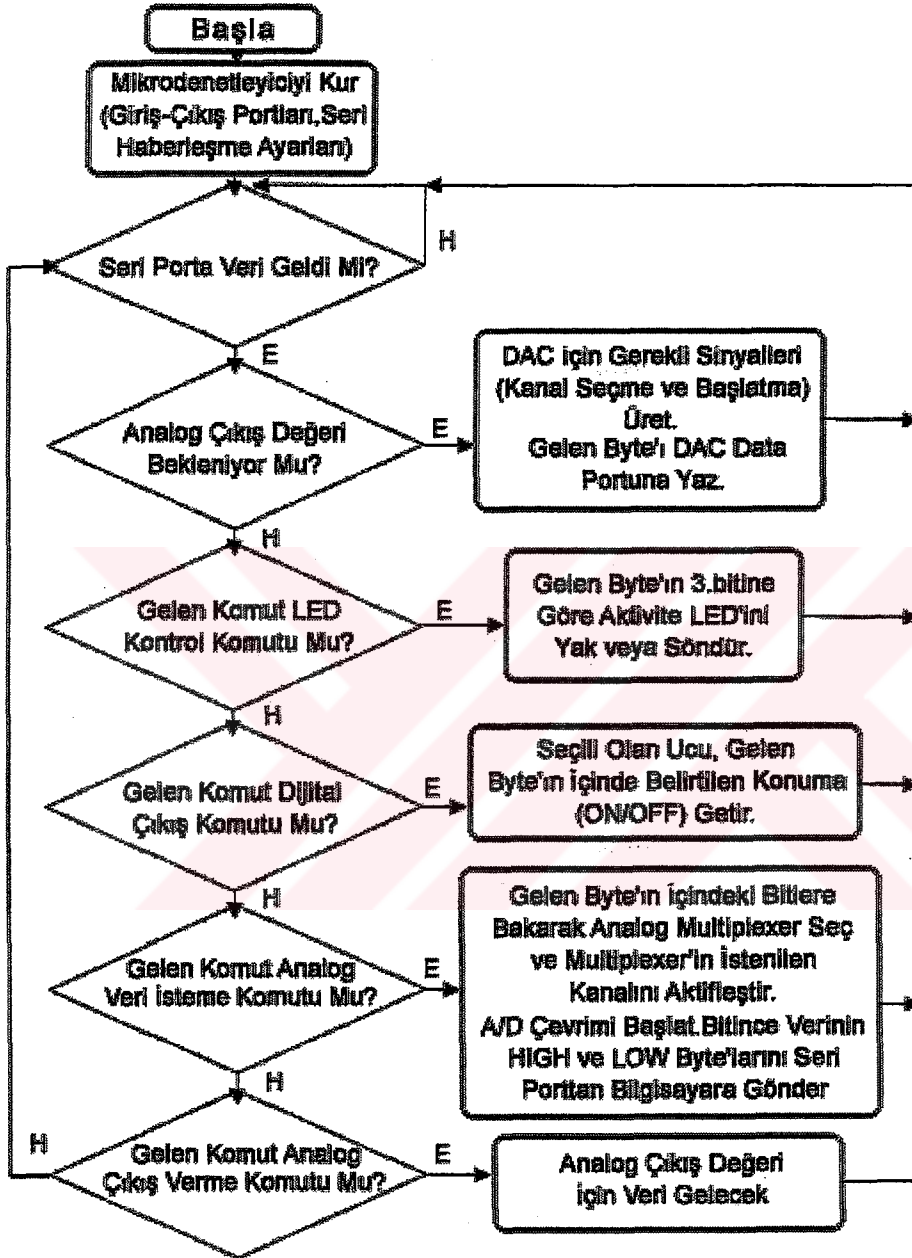
Şekil 5.7. Kontrol Kartı

Gerçekleştirilen kontrol kartı bilgisayarla haberleşmede RS485 protokolünü kullanmaktadır. Kontrol kartı üzerinde bulunan iki adet 75176 entegresi bulunmaktadır. Tam dubleks haberleşme yapıldığı için 75176 entegresinden iki adet kullanılmıştır. Bilgisayar tarafında yine RS485’den RS232’ye çevirici bir modül yapılmıştır. Şekil 5.8’de haberleşme modülü görülmektedir.



Şekil 5.8. Haberleşme Modülü

Mikrodenetleyici içerisinde bulunan programın akış diyagramı şekil 5.9'da görülmektedir.



Şekil 5.9. Mikrodenetleyici Programının Akış Diyagramı



5.1.2. Arabirim Kartının Testi ve Ayarı

Program arabirim kartının testi için hazırlanmıştır. Önceki bölümlerde bahsedildiği gibi arabirim kartı analog sinyalleri 10 bitlik sayısal bilgiye dönüştürmekte ayrıca kontrol modülünden gelen bilgilere göre sayısal çıkışları kontrol etmektedir.

Test programı her bir sayısal çıkışı teker teker açıp kapama işlemini yapabilmektedir. Ayrıca istenilen kanaldaki analog sinyalin seviyesini göstermektedir bu sayede arabirim kartının işlevleri kontrol edilmiş olur.

Mikrodenetleyiciye 8 bitlik veri yollar bu veriye göre mikrodenetleyici ya istenilen kanaldaki analog verinin sayısal karşılığını yollar, ya da istenilen sayısal çıkışı açar veya kapar. Ayrıca uyarı ledinin kontrolünde program yapmaktadır. Bilgisayar tarafından yollanan 8 bitlik veri mikrodenetleyici tarafından şu şekilde denetlenir:

Mikrodenetleyici öncelikle yollanan verinin 7, 6 ve 5. bitlerine bakar, bu bitlerin anlamları Tablo 5.3'de gösterilmiştir.

Tablo 5.3. Başlık Bilgileri

Bitler	Anlamı
001	Programın analog sinyal değeri istediği anlamına gelir
010	Sayısal çıkışları kontrol etmek istediğini gösterir
110	Kontrol ledini kontrol edileceği anlamına gelir

Eğer analog sinyal değeri isteniyor ise mikrodenetleyici 3. bite bakar eğer bu bit 0 ise Kanal A'dan, eğer 1 ise Kanal B'den veri istenileceği anlamına gelir. Daha sonra 2, 1 ve 0. bitlere bakar bu üç bit kanalların hangi girişini istendiği söyler.

Örneğin gelen veri 2C yani 00101100 ise bu kanal B nin 4. bitindeki analog sinyalin sayısal karşılığı isteniyor demektir.

Eğer sayısal çıkışlar kontrol edilmek isteniyor ise 4. bite bakar eğer 4. bit 0 ise kontrol edilmek istenilen çıkışın düşük seviyeye, eğer 1 ise bu çıkışın yüksek seviyeye çekileceği anlamına gelir. Daha sonra mikrodnetleyici 3. bite bakar 3. bit 0 ise A iskelesinin, eğer 1 ise B iskelesinin çıkışlarından birisinin kontrol edileceğini gösterir. Son üç bit 2, 1 ve 0. bitler ise iskelelerden hangi çıkışın kontrol edileceğini tanımlar. Örneğin gelen veri 5F yani 01011111 ise bu B iskelesinin 7. bitinin açılacağı anlamına gelir.

Eğer uyarı ledinin kontrolü isteniyor ise sadece 3. bite bakılır. Eğer 3. bit 0 ise led söndürülür eğer 1. ise led yakılır.

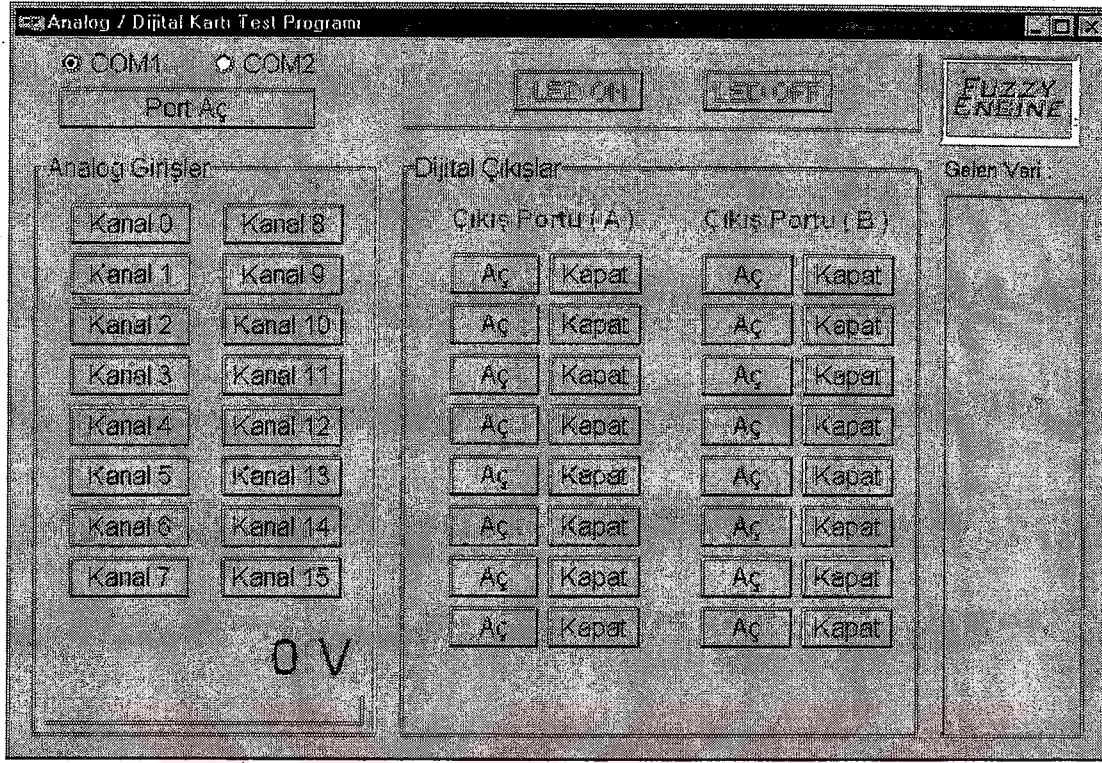
Gelen veri 110X0XXX ise uyarı ledi söndürülür, eğer gelen veri 110X1XXX ise ledi yakar.

Program tarafından bir analog sinyalin seviyesi isteniyorsa mikrodnetleyici cevap olarak üç bytelik bir veri paketi yollar. Bu yollanan bilginin açılmış hali şu şekildedir.

1. Byte istenilen analog sinyalin cevabı olduğunun belirtisi için konmuştur, bir tür başlık bilgisi diyebiliriz.
2. Byte 10 bitlik ADC çıkışının low bytedir.
3. Byte ise en yüksek dereceli ADC çıkışları 9 ve 10. bitleridir.

Sayısal çıkışların kontrolünde ve uyarı ledi kontrolünde mikrodnetleyici bir cevap yollamaz.

Şekil 5.10'da arabirim kartının çalışmasını test etmek için hazırlanan program görülmektedir.



Şekil 5.10. Test Programı Giriş Ekranı

Programda öncelikle hangi iskelele kullanılacağı seçilir. Daha sonra Port Aç butonuna basılır bu sayede iskele kullanıma açılır. Eğer analog veri isteniyor ise analog giriş kısmında bulunan Kanal 0'dan Kanal 15'e kadar olan butonlardan birine basılır istenilen analog sinyalin karşılığı ekranda görülür. Ayrıca ekranın en sağında bulunan kısımda ise gelen 3 byte'lık verin yazılır. Eğer sayısal çıkış kontrolü yapılmak isteniyorsa iskele A ve iskele B çıkışlarından hangisinin çıkışı kontrol edilmek isteniyor ise dijital çıkışlar kısmında bulunan butonlara basılarak çıkış istendiğinde açılır istendiğinde ise kapanır. Uyarı ledi kontrol edilmek istendiğinde üst tarafta bulunan LED ON ve LED OFF butonlarına basılarak uyarı ledinin sönmesi veya yakılması sağlanır.

Yapılan bu program sayesinde arabirim kartının en temel işlevleri kontrol edilmiş olur.

5.2. Yazılım Paketinin Tasarımı

Yazılım paketi iki ana kısımdan oluşmaktadır; bunlardan birincisi kural modülü diğeri kontrol modülüdür her iki modül içerisinde birbiriyle bağlantılı görsel ve mantıksal işlemler vardır.

- **Kural Modülü**

Tablo 5.4. Kural Modülünden Beklentiler

	Görsel	Mantıksal
Değişkenler	Değişken giriş	Değişkenler için alt kümelerin oluşturulması
Kural dizini	Kural dizinin oluşturulması	Kural dizinin kodlanarak kaydedilmesi
Çizim ekranı	Değişkenlere ait üyelik fonksiyonlarının çizilmesi	Üyelik fonksiyonunu oluşturan kümelerin konumlarının tespiti ve kaydı

- **Kontrol Modülü**

Tablo 5.5. Kontrol Modülünden Beklentiler

	Görsel	Mantıksal
Kontrol	Kontrol için bir dosya seçilmesi	Dosya içindeki değişkenlerin kural dizinine aktarılması
	Kontrol ayarlarının yapılması	Çıkış verilerinin sayısal veya analog olarak hazırlanması
Veri alma ve aktarma	Kontrolün başlatılması	Kontrol kartından verilerin istenmesi, hesaplamaların yapılması ve çıkış değerlerinin gönderilmesi
Çizim ekranı	Giriş ve çıkışların çizilmesi	Verilerin bir dosyaya atılarak saklanması

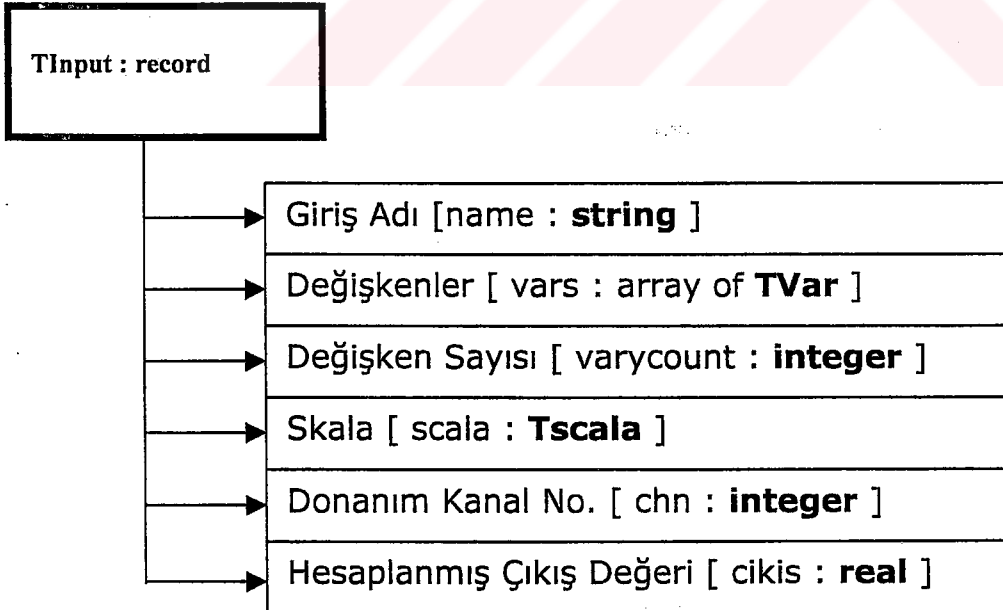
Sistemden alınan bilgilerin alınıp kontrol kurallarına göre işlendiği ve kontrol işaretlerinin oluşturulduğu yazılım kısmı iki ana modülden meydana gelmektedir. Bu iki ayrı modül bir başlangıç programı ile çağrılmaktadır.

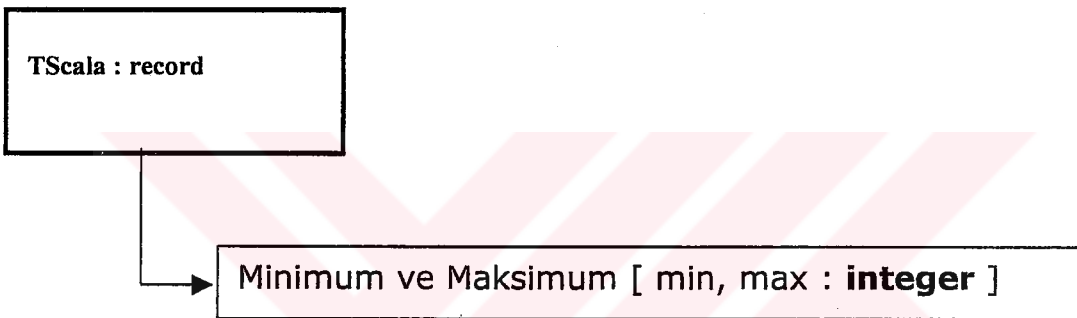
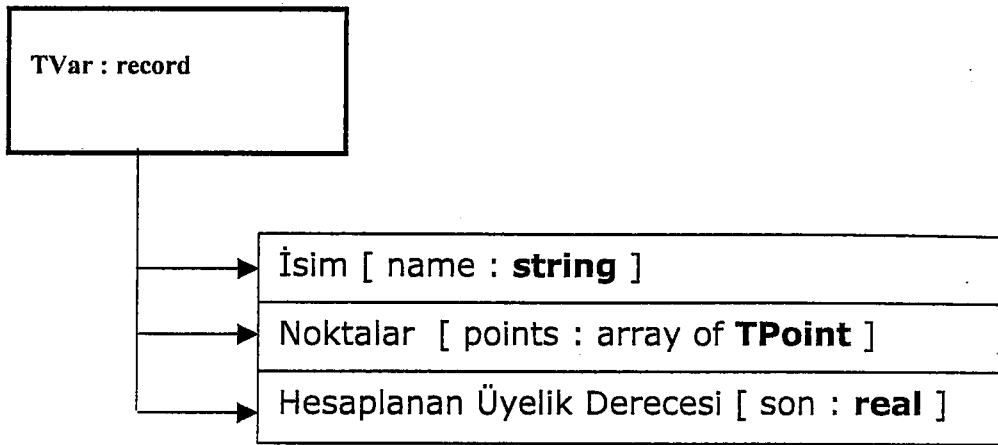
5.2.1. Program Yapısı

Programda yapısal programlama tekniklerinden faydalanılmıştır. Giriş ve çıkışlar birer yapı (Pascal programlama dilinde 'type' , C'de 'structure') olarak tanımlanmıştır. Aşağıda tanımlanan ana TInput yapısı ve üyeleri gösterilmiştir. Ana yapının üyelerinden TVar, Tscala da tanımlanan başka yapılardır.

Giriş ve çıkış yapıları aynı olduğu için ikisi için de Tinput yapısı kullanılmıştır:

Inputs : array of Tinput
Outputs : array of Tinput





'Kontrol' programında, 'Kural' programı ile hazırlanan bulanık proje dosyası yüklendikten sonra şu işlemler yapılır:

- Kullanıcıdan İşlem Parametrelerinin Alınması :
 - İşlem Periyodu
 - Grafik Tazeleme Periyodu
 - Çıkışların Sayısal veya Analog Seçilmesi
- İlk İşlem Periyodu Geçildiğinde :
 - i : 1'den 'Giriş Sayısı'na Kadar Tekrarla:
 - {
 - Giriş[i] için karttan analog veri iste.
 - Gelen veriyi Scala ile tanımlanan aralığa ölçekle.
 - }

Tüm giriş değerleri alındığında Durulaştırma prosedürünü çalıştır.

- Durulaştırma prosedüründe şu işlemler yapılmaktadır :

i : 1'den 'Giriş Sayısı'na Kadar Tekrarla:

```
{
    j : 1'den 'Giriş[i]'nin Değişken Sayısı'na Kadar Tekrarla
    {
        Alınan analog değer, Giriş[i].Değişken[j]'de parçalı grafikte
        aldığı 0-1 arasındaki değeri bul.
        Hesaplanan sonucu Giriş[i].Değişken[j].son değerinde sakla.
    }
}
```

Tüm girişlerin üyelik dereceleri hesaplandıktan sonra 'Kural' programında tanımlanan ve bulanık proje dosyasına kaydedilen kural tablosunda bu değerler yerleştirilir.

i : 1'den 'Kural Tablosu Satır Sayısı'na Kadar Tekrarla :

```
{
    Tablo.Satır[i]'nin minimum değerini bul.
    Bulunan değeri Sonuc[i]'de sakla
}
```

i : 1'den 'Çıkış Sayısı'na Kadar Tekrarla :

```
{
    j : 1'den Çıkış[i].Değişken[j]'ye Kadar Tekrarla :
    {
        Çıkış[i].Değişken[j]'nin Sonuc[x] dizisinde
        Almış olduğu maksimum değeri bul.
        Maksimum değeri Çıkış[i].Değişken[j]'nin
        Parçalı grafiğinde yerleştir ve aldığı değeri bul.
    }
}
```

Çıkış[i]'nin tüm değişkenlerinin aldığı değerler bulunduktan sonra, değişkenlerin ağırlıklı ortalamaları alınarak Çıkış[i].Cikis değeri hesaplanır.

}

Böylece Min-Max teoremine göre çıkışlar hesaplanmış olur. Bundan sonra:

Çıkışlar Analog Seçilmişse :

i : 1'den Çıkış Sayısı'na Kadar Tekrarla:

{

Karta DAC ile analog çıkış verme komutu gönder.

Gönderilen ilk byte'ta çıkış komutu ve kanal bilgisi vardır.

İkinci byte'ta sayısal veri (ölçeklenmiş Çıkış[i].Cikis) bulunur.

}

Çıkışlar Dijital Seçilmişse :

i : 1'den Çıkış Sayısı'na Kadar Tekrarla:

{

Zamanlayıcı[i]'nin periyodunu ölçeklenmiş Çıkış[i].Cikis ve

işlem periyodu ile hesaplanan değere kur. Karta dijital çıkış[i]'yi HIGH yapma komutu gönder.

Zamanlayıcı[i]'yi başlat.

}

- Her zamanlayıcı[x] periyodu dolduğunda:

Zamanlayıcı[x]'i durdur.

Karta dijital çıkış[x]'i LOW yapma komutu gönder.

- Her grafik tazeleme periyodu dolduğunda:

i : 1'den Giriş Sayısı'na Kadar Tekrarla :

{

Giriş[i]'ye ait renkle Giriş[i]'nin değerine grafikte nokta koy.

}

Girişler grafiğini kaydır.

i : 1'den Çıkış Sayısı'na Kadar Tekrarla :

{

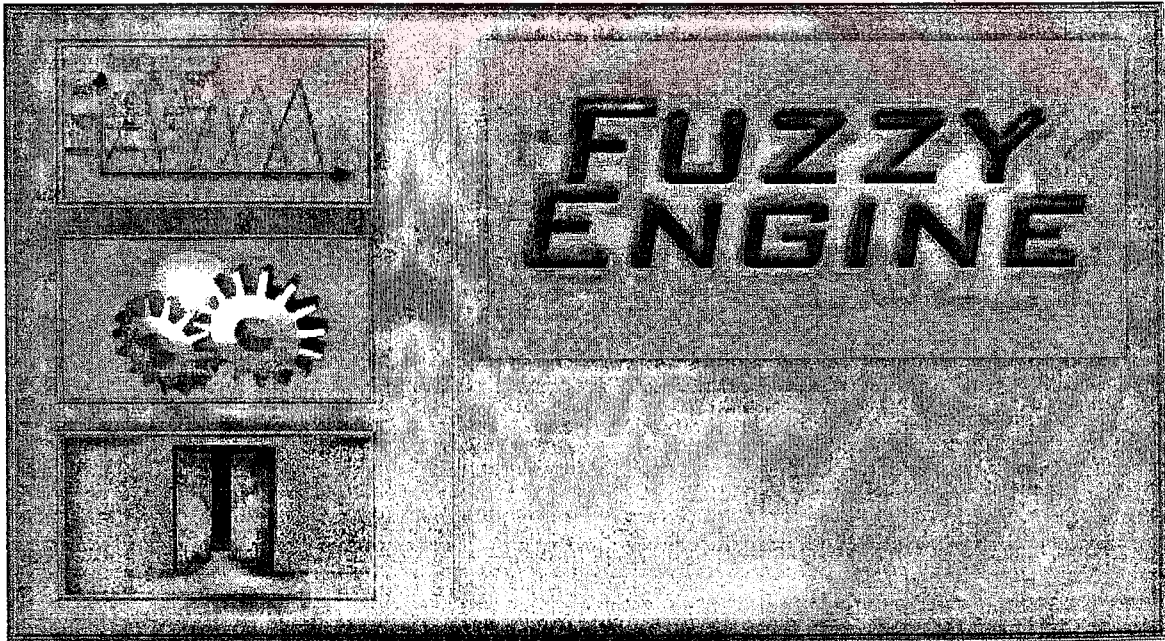
Çıkış[i]'ye ait renkle Çıkış[i]'nin değerine grafikte nokta koy.

}

Çıkışlar grafiğini kaydır.

5.2.2. Başlangıç Programı

İki ana modülün çağrılmasında kullanılır. Modüllerin herhangi birinden çıkılması durumunda yine başlangıç programı çalışmaktadır bu sayede programlar arasında geçiş sağlanmaktadır. Programda üç ayrı ikondan meydana gelmektedir. Şekil 5.11'de başlangıç programı giriş ekranı görülmektedir.



Şekil 5.11. Program Giriş Ekranı

Üstteki ikon giriş ve çıkış grafiklerinin ve kural tablosunun çıkarıldığı kural modülünü çalıştır. Ortadaki ikon ise bilgilerin alınıp kontrol kurallarına göre işlendikten sonra kontrol işaretlerinin oluşturduğu kontrol modülünü çalıştırır. Üçüncü ikon ise programdan çıkışta kullanılır.

Kullanılan bu program sayesinde aslında kural ve kontrol olarak iki ayrı programdan oluşan kısımlar bir arada kullanılarak tek bir programın iki ayrı modülü şeklinde kullanılmaktadır.

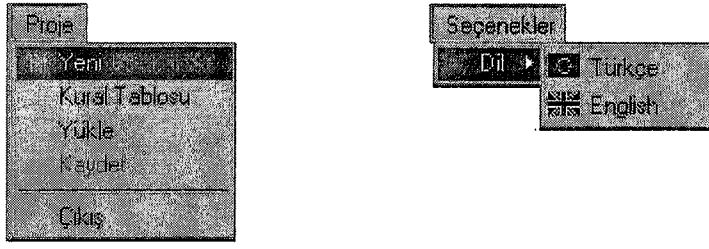
5.2.3. Kural Modülü

Kural modülü giriş ve çıkışların tanımlandığı ve kural tablosunun oluşturulduğu kısımdır. Başlangıç programında üstteki ikona tıklandığında kural modülünün girişi gelir. Şekil 5.12’de kural modülü giriş ekranı görülmektedir.



Şekil 5.12. Kural Modülü Giriş Ekranı

Kural modülüne ait iki ayrı menü seçeneği vardır. Şekil 5.13’de kural modülüne ait menü seçenekleri görülmektedir.



Şekil 5.13. Kural Modülü Menü Seçenekleri

Proje menüsünde bulunan yeni seçeneği tıpkı kural tablosu, yükle, kaydet ve çıkış seçeneğinde olduğu gibi kural modülünün girişinde bulunan butonlarla da ilişkilendirilmiştir.

Seenekler menüsünde dil seçimi yapılabilmektedir.

Yeni bir kontrol projesi oluşturmak için yeni butonuna tıklanır. Burada kontrol edilmek istenen sisteme ait giriş çıkış sayıları girilmektedir. Ayrıca sistemi tanımlayacak bir ad verilmektedir. Şekil 5.14’de girişlerin tanımlandığı ekran görülmektedir. Bu bilgiler girildikten sonra ileri butonuna tıklanır.

Şekil 5.14. Proje Tanımlama Ekranı

Sisteme ait girişlerin tanımlandığı ve bunlara ait grafiklerin çizildiği ana ekran görüntüsü ekrana gelir. Şekil 5.14’de görüleceği gibi daha önceden girilen giriş sayısına göre değişkenler sınırlandırılmıştır.

Girişlerin üzerine tıklanarak girişlere ait parametre girişleri yapılabilmektedir. Giriş ekran görüntüsü Şekil 5.15’de görülmektedir.

Girişler	Kanal	Min	Max	Değişken 1	Değişken 2	Değişken 3	Değişken 4	Değişken 5	Değişken 6
Input1	1	0	5	None	None	None	None	None	None
Input2	2	0	5	None	None	None	None	None	None
Input3	3	0	5	None	None	None	None	None	None

Şekil 5.15. Değişken Giriş Ekranı

Şekil 5.16’da giriş tanımlama ekranı görülmektedir. Tüm girişler öncelikle kaç kümesi olduğu hangi kanala bağlı olduğu ve kullanılacak 0V ile 5V arasındaki analog girişin hangi aralığının kullanılacağı belirtilir. Bilindiği gibi her zaman analog girişlerimiz 0V ile 5V arasında olamayabilir. Kullanılacak olan aralığın girilmesiyle program girilen aralığı alt seviyesini 0V üst seviyesini ise 5V olarak işlemektedir, bu sayede uygulama ile program arasında uyum sağlanmıştır.

Giriş Değişkeni

Giriş Adı: Yol durumu

Giriş Değişkeni: 2

Kanal Numarası: 1

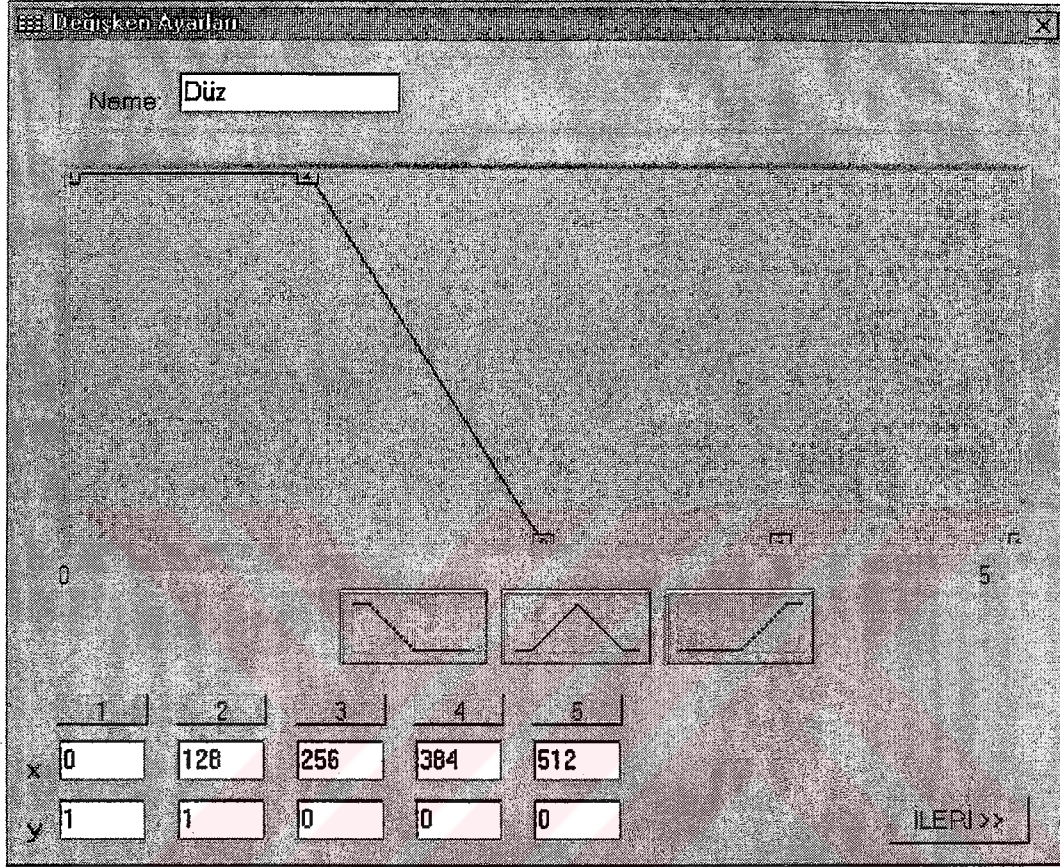
En Az: 0

En Çok: 5

Tamam

Şekil 5.16. Giriş Tanımlama Ekranı

Giriş e ait küme bilgileri ve donanım bilgileri girildikten sonra girişlerin fonksiyon kümelerinin sınırları girilir. Şekil 5.17'de girişlerin üyelik fonksiyonlarının tanımlandığı kısım görülmektedir.



Şekil 5.17. Giriş Grafiği Çizme Ekranı

Küme sınırları fare ile kırmızı karelere gelinerek bunların sürüklenmesiyle oluşturulabileceği gibi direkt olarak konum girilerek de yapılmaktadır. Çizim aşamasının kolaylaştırılması için pencere ortasına üç adet buton koyulmuştur. Konum değerleri sabit olan bu butonlara basıldığında grafiğin şekli değişmektedir. Tüm girişlere ait kümeler belirlendikten sonra ileri butonuna tıklanarak geçilir. Üyelik fonksiyonlarının oluşturulmasında beş adet doğru parçasından yararlanılmaktadır.

Tüm girişlerin girilmesinin ardından girişlere ait değişken isimleri bunlara ait fonksiyonların isimleri ile analog sinyalin kullanım aralığı ve hangi kanaldan

kullanılacağına dair bilgilerin bulunduğu kısım ekrana gelir. Şekil 5.18'de tüm girişlere ait üyelik fonksiyonlarının toplu halde bulunması görülmektedir.

Girişler	Kanal	Min	Max	Değişken-1	Değişken-2	Değişken-3	Değişken-4	Değişken-5	Değişken-6
Yol durumu	1	0	5	Düz	Engel	None	None	None	None
Hava	2	2	4	Yağışlı	Kuru	None	None	None	None
Arabanın Hızı	3	0	5	Hızlı	Normal	Yavaş	None	None	None

Şekil 5.18. Tanımlanmış Girişlerin Görüntüsü

Çıkış değişkenleri ve bunların kümeleri girişlerde olduğu gibi girildikten sonra kuralların girildiği kural tablosu çağrılır.

Kural tablosunda her giriş ve bu girişlere ait kümelerin tüm olasılıkları bulunmaktadır. Mavi kısımda giriş değişkenleri pembe kısımda ise çıkışlar bulunmaktadır. Her çıkışa ait küme adları çıkış tarafında bulunmaktadır, hangi olasılıkta çıkış kümesi ne olacağı burada belirlenir. Şekil 5.19'da kural tablosunun giriş ekranı görülmektedir.

Tüm olasılıklara karşılık gelen çıkış kümeleri belirlendikten sonra bu dosya kaydet butonuna basılarak kaydedilir. Oluşan dosya text formatında açılabilir.

Kaydedilen dosya içerisinde projenin adı, projenin oluşturulma tarihi, giriş, çıkış sayıları ve bunlara ait parametreler bulunmaktadır. Bu parametreler içerisinde giriş ve çıkışa ait kümeleri oluşturan beş adet doğru parçasının konumları ve hem giriş hem de çıkışa ait matris kodlanmıştır.

Bulanık Kontrol Kural Tablosu

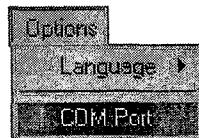
Yol Durumu	Hava	Araç Hızı	Stres	Fren Mesafesi
Düz	Yağışlı	Hızlı	Yavaşla	Çok
Düz	Yağışlı	Normal	Normal	Orta
Düz	Yağışlı	Yavaş	Hızlan	Az
Düz	kuru	Hızlı	Normal	Orta
Düz	kuru	Normal	Hızlan	Az
Düz	kuru	Yavaş	Çok hızlan	Az
Engeli	Yağışlı	Hızlı	Çok yavaşla	Çok
Engeli	Yağışlı	Normal	Yavaşla	Orta
Engeli	Yağışlı	Yavaş	Normal	Az
Engeli	kuru	Hızlı	Yavaşla	Orta
Engeli	kuru	Normal	Normal	Az
Engeli	kuru	Yavaş	Hızlan	Az

Çok hızlan
Hızlan
Normal
Yavaşla
Çok yavaşla

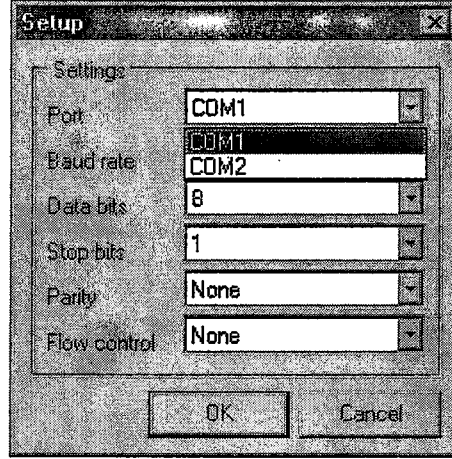
Şekil 5.19. Kural Tablosu

5.2.4. Kontrol Modülü

Kural modülü tarafından oluşturulan dosyayı işleyerek kontrolü gerçekleştiren kısımdır. Kontrol modülü iki menüden oluşmaktadır, bunlardan Seçenekler(Options) seçeneği seçildiğinde dil ve seri iletişim seçenekleri gelmektedir. Seri iletişim seçeneği seçildiğinde seri iletişime ait özellikler ekrana gelecektir buradan kartın bağlı olduğu seri iletişim çıkışı seçilir. Şekil 5.20’de dil seçimi, Şekil 5.21’de seri iskele seçimine ait ekran görüntüleri görülmektedir.

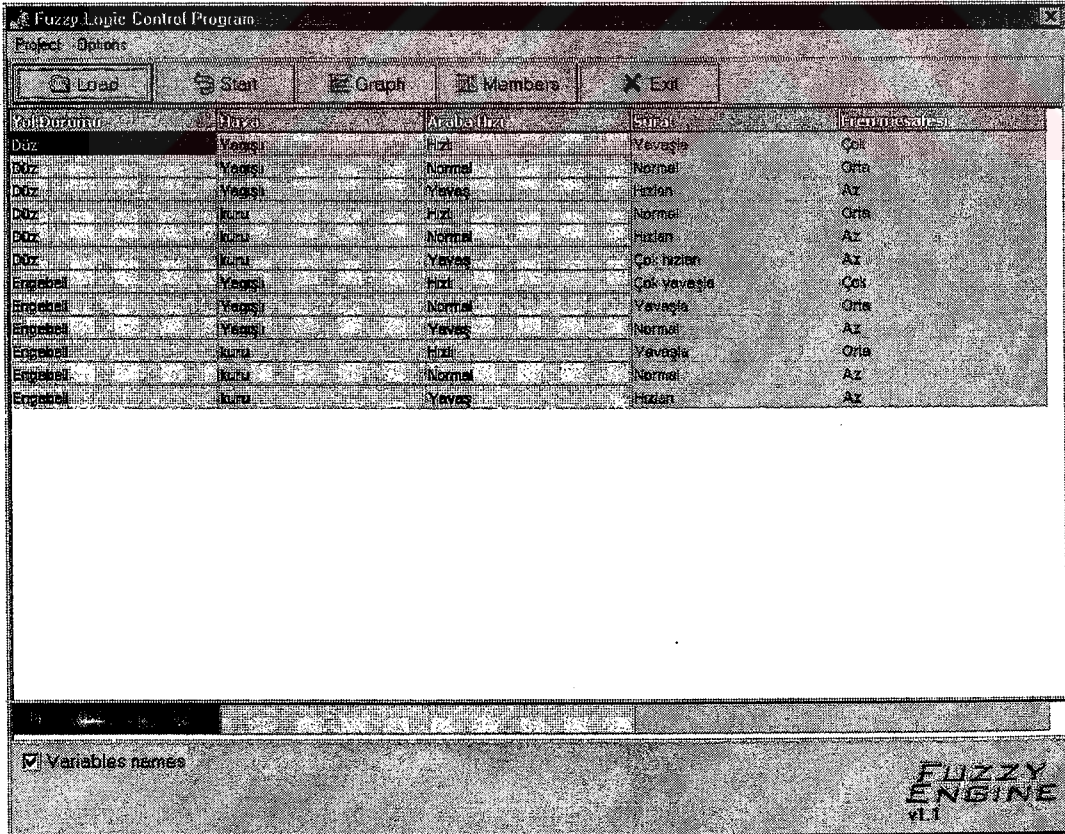


Şekil 5.20. Dil Seçim Menüü



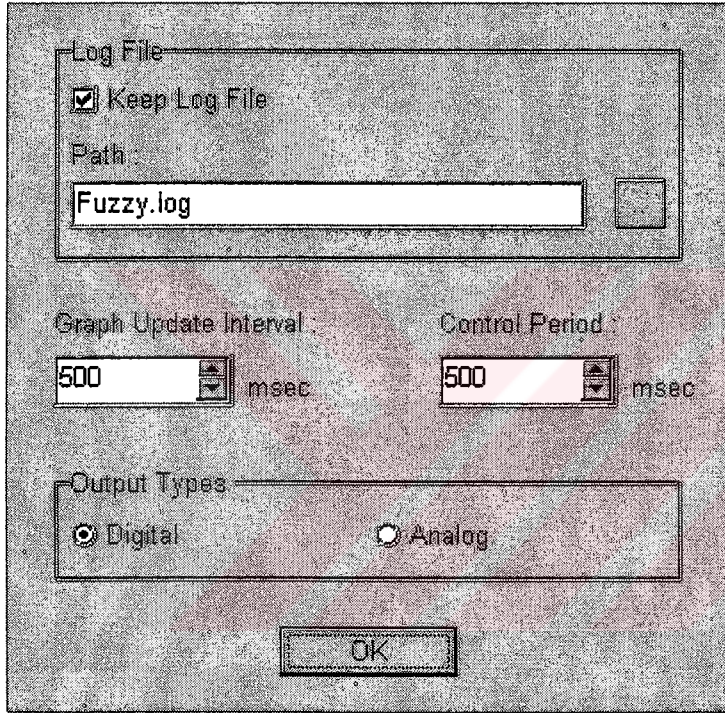
Şekil 5.21. Seri İskele Ayar Ekranı

Seri iletişime ait ayarlamalar yapıldıktan sonra kontrol edilecek projeye ait dosya yükle kısmı seçilerek yüklenir. Şekil 5.22’de kontrol modülü giriş ekranı görülmektedir.



Şekil 5.22. Giriş Çıkış Değerlerinin Görüntüsü

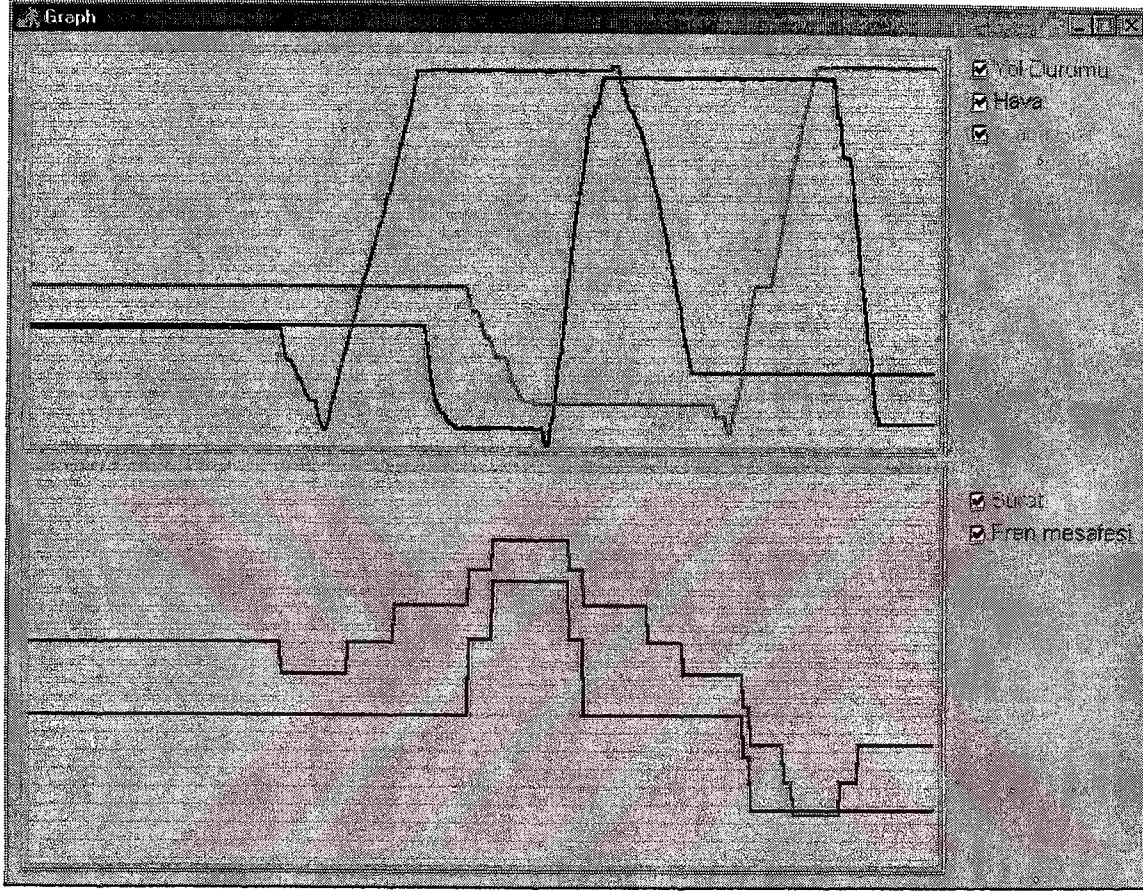
Burada daha önceden hazırlanmış dosya içindeki verilere göre giriş ve çıkışlar yerleştirilir. Başlat menüsü seçildiğinde kontrol edilecek sisteme ait verilerin hangi aralıklarla işlenmesi istendiği, bu bilgilerin grafik olarak çizilme periyodu, çıkış bilgilerinin sayısal mı yoksa analog mu olacağına ve giriş çıkış bilgilerinin tutulduğu dosyanın kaydedileceği yere ait bilgilerin girildiği pencere ekrana gelir. Burada girilebilecek kontrol periyodu en az 50ms dir. Şekil 5.23'de kontrol parametre ayarların yapıldığı ekran görülmektedir.



Şekil 5.23. Kontrol Parametre Ayarları

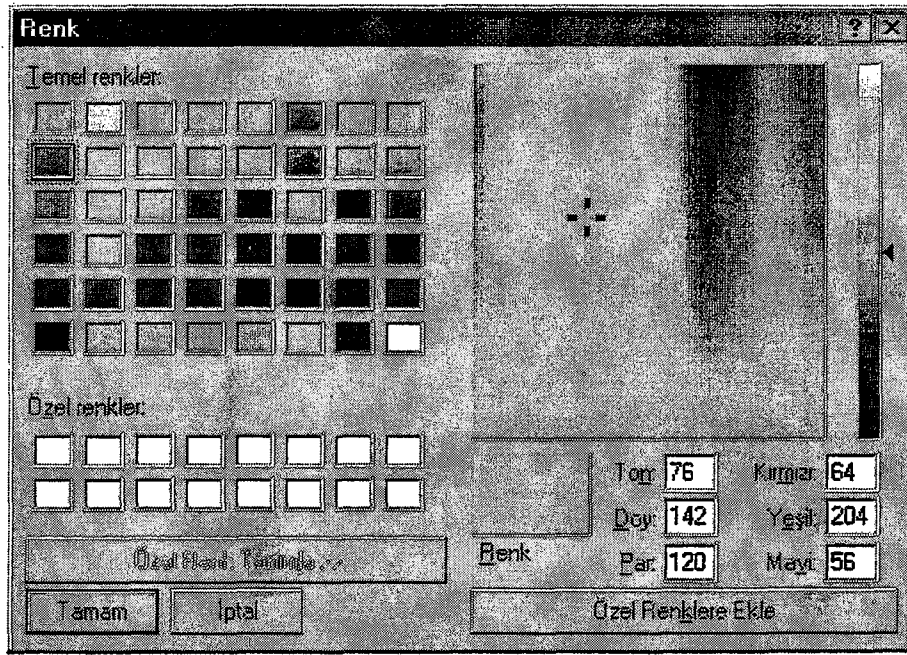
Bu ayarlar yapıldıktan sonra program kullanılan girişlere ait bilgileri karttan ister. Bu bilgiler geldikten sonra bulanık kontrol kurallarına göre giriş değişkenleri işlenir ve çıkışa verilirler. İki ayrı çıkış şekli vardır. Bunlardan ilki sayısal, diğeri analog çıkışlardır. Sayısal çıkışların aktif olduğu süre ölçme periyodunun çıkışla çarpılmasıyla bulunur. Eğer çıkışımız 0,65 değerinde ise ve bizim ölçme periyodumuz 5000ms ise çıkışımızın aktif kaldığı süre 3250ms olacaktır. Eğer kontrol edilecek olan sistemde ölçme periyodu çok kısa ise ve zaman çok kritik ise burada giriş-çıkış bilgilerinin istenmemesi tavsiye edilir çünkü kontrol periyodunda alınan bilgilerin işlenip çıkış üretilmesinden sonra tutulan kayıt bilgisi kontrol işleminde en uzun zamanı almaktadır.

Grafik seçeneği seçildiğinde giriş çıkış değerlerinin grafiksel olarak görüldüğü pencere açılır. Şekil 5.24'de grafiklerin çizildiği kısım görülmektedir.



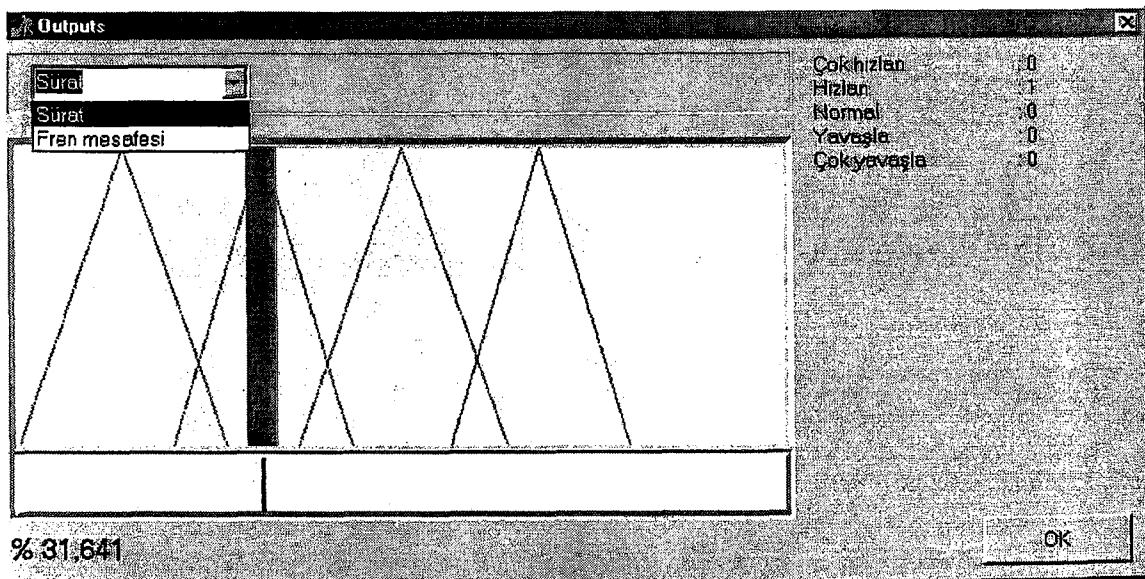
Şekil 5.24. Giriş Çıkış Değerlerinin Görüntülenmesi

Girişler ve çıkışlar anı anda bu pencereden izlenebilmektedir. Tüm giriş ve çıkışlar ayrı birer renk ile gösterilmiştir, istendiğinde seçeneklerin üzerine iki defa tıklandığında karşımıza renk seçim penceresi gelecektir. Şekil 5.25'de istenilen renk seçilir.



Şekil 5.25. Renk Seçim Ekranı

Çıkış değerlerinin üyelik dereceleri üyelik butonuna basıldığında açılan pencereden görülebilir. Her çıkışa kümelerin üyelik dereceleri ve hesaplanan çıkış değerleri bu pencereden izlenmektedir. Her kontrol periyodunda pencere güncellenmektedir. Şekil 5.26'da çıkışa ait üyelik fonksiyonlarının üyelik dereceleri görülmektedir.

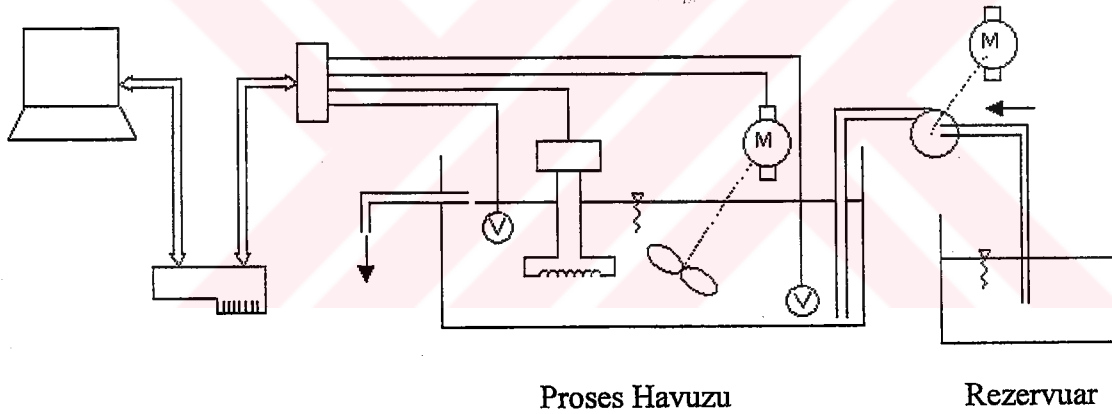


Şekil 5.26. Üyelik Derecelerinin Görüntülenmesi

6. ÖRNEK BİR SİSTEMİN KONTROLÜ

Fuzzy Engine sisteminin gerçek ortamda uygulanması ve kontrol yeteneğini göstermek amacıyla örnek bir proses düzenlenmiştir.

Bu proses birçok biyokimyasal prosesin temel özelliklerini taşıyacak şekilde; sıvı dolu bir havuz ile bu havuzu besleyen bir pompa ve fazla suyu tahliye eden bir taşma çıkışından oluşmuştur. Havuz 30 litre su almaktadır. Giriş ve çıkış uçlarında sıcaklık birer termokupl aracılığıyla ölçülmekte, çıkışta belli bir sıcaklık sağlayacak şekilde ısıtma imkanı bulunmaktadır. Havuzdaki sıcaklık dağılımını homojenleştirmek için ise gerektiğinde çalışacak bir karıştırıcı (mikser) kullanılmaktadır. Şekil 6.1'de homojen dağılımlı sıcaklık kontrol prosesi görülmektedir.

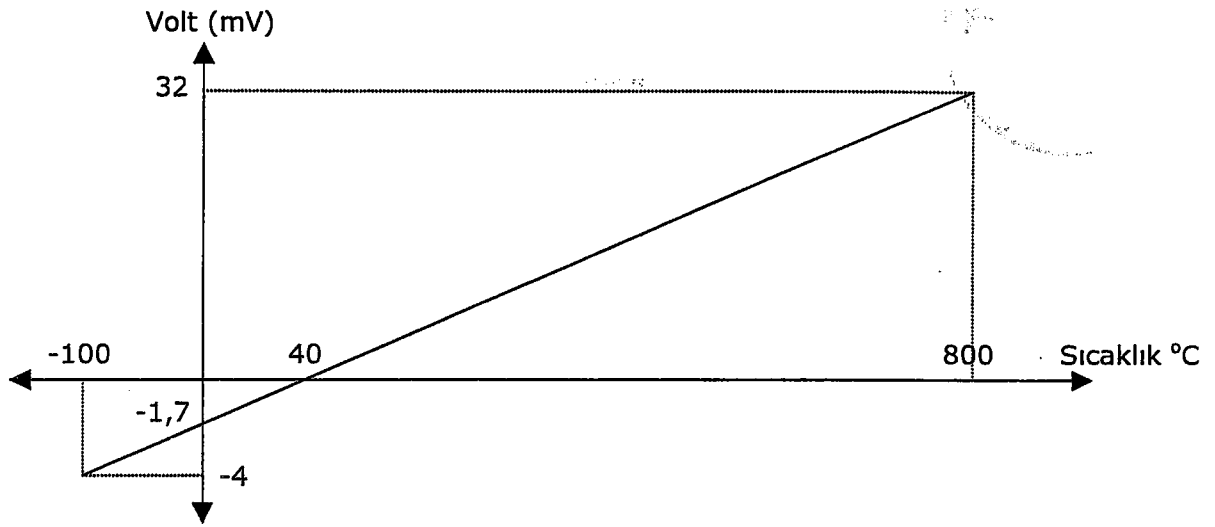


Şekil 6.1. Homojen Dağılımlı Sıcaklık Kontrol Prosesi

6.1. Prosesin Kontrol Elemanları

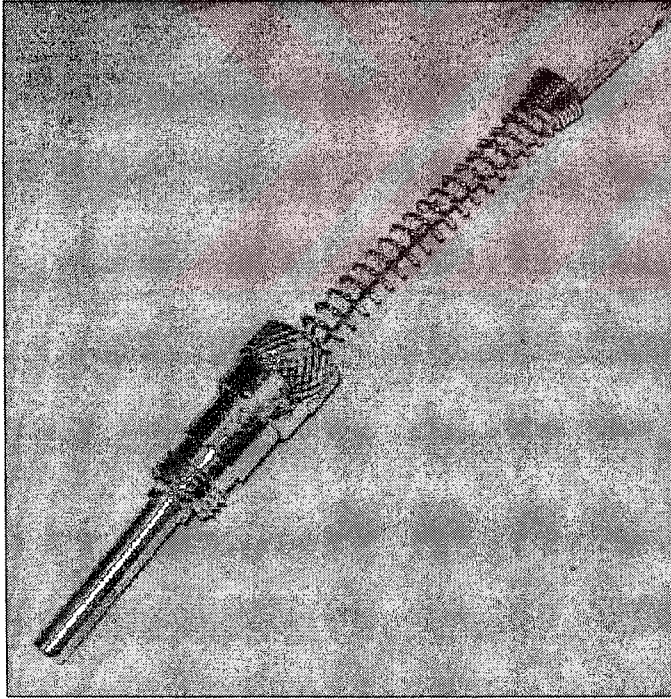
6.1.1. Girişler

Sıcaklık ölçümleri için iki adet J tipi termokupl kullanılmıştır. J Tipi termokupl -100 ile $+800$ °C arasında ölçüm yapabilmektedir. J tipi termokupl için gerilim sıcaklık grafiği Şekil 6.2'de gösterilmiştir.



Şekil 6.2. J Tipi Termokupl Gerilim Sıcaklık Grafiği

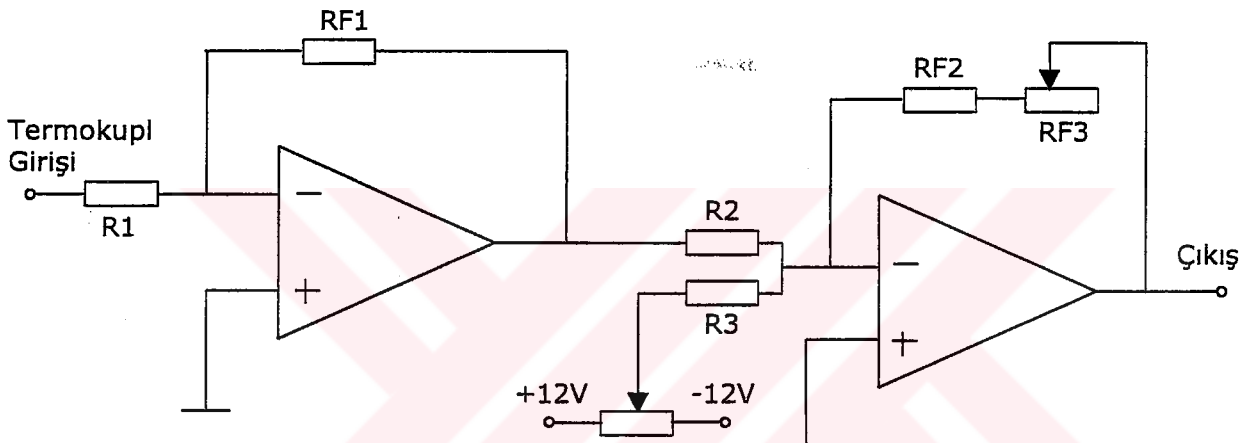
Şekil 6.3'de J tipi termokupl görüntüsü görülmektedir.



Şekil 6.3. J Tipi Termokupl

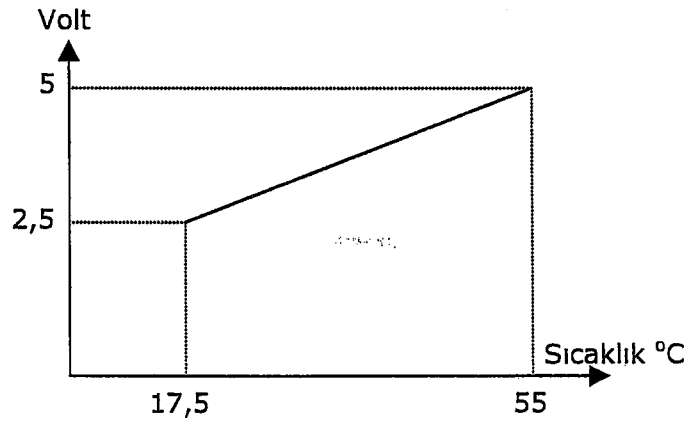
J Tipi termokupl yaklaşık -4 ile 32 mV arasında gerilim üretmektedir. Özellikle çalışma aralığının düşük olduğunu düşünürsek termokupldan alınan gerilimin işlenebilir değerlere getirilebilmesi için çalışma sıcaklık aralığı tespit edildikten sonra yükseltilmesi gerekmektedir.

Sıcaklık ölçüm modülü bir kazanç katı ve filtreden oluşmaktadır. Kazanç katı iki katlı bir yükselteç devresinden oluşmaktadır. Her iki katta da kullanılan yükselteç devresi evirendir. İlk katta kullanılan yükselteç devresinin kazancı 30'dur. İkinci kattaki yükselteç devresinin kazancı 20 ile 70 arasında ayarlanabilmektedir. Bu sayede toplam kazanç yaklaşık olarak 600 ile 2100 arasında değişmektedir. Sistemin çalışma aralığı 0V ile 5V arasında olduğu için termokupldan gelen negatif gerilimin pozitifleştirilmesi gerekmektedir. Bu nedenle ikinci kattaki yükselteç devresinde birinci kattan gelen gerilim ile sabit bir gerilim toplanmaktadır. Şekil 6.4'de yapılan kazanç devresi görülmektedir.



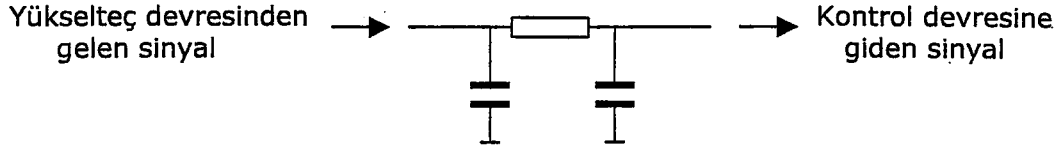
Şekil 6.4. Kazanç Devresi

Yapılan kazanç devresi sayesinde termokupl girişleri işlenebilir değerlere yükseltilmiştir. Yükseltilen değer ve sıcaklık grafiği Şekil 6.5'de görülmektedir.



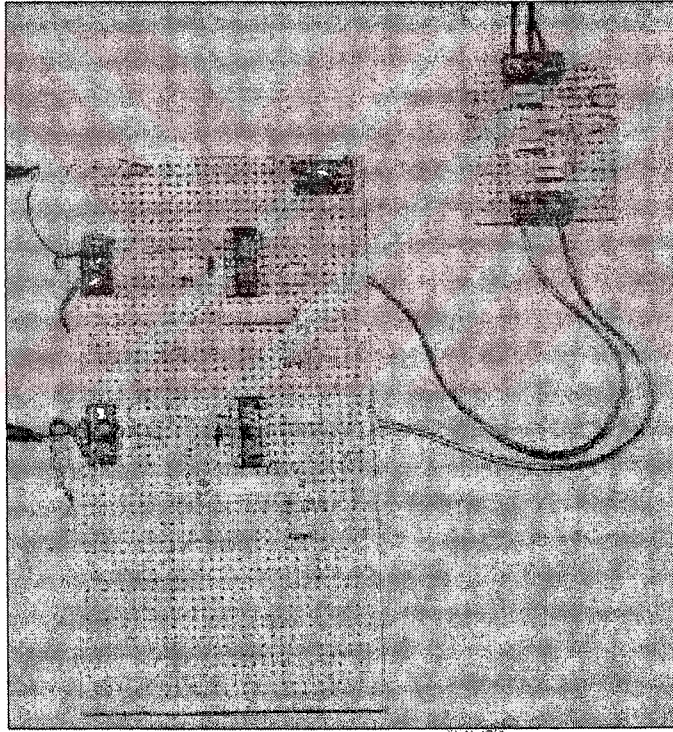
Şekil 6.5. Yükselteç Çıkışındaki Sıcaklık-Gerilim Grafiği

Yükselteç devresinin kazancının çok fazla olması ve termokupllardan meydana gelen gürültünün engellenmesi için bir filtre kullanılmıştır. Kullanılan bu filtrenin yapısı Şekil 6.6'da görülmektedir.



Şekil 6.6. Filtre Devresi

Şekil 6.7'de Termokupllardan gelen sinyali güçlendiren kazanç devresi ve bu sinyali filtre eden filtre katı görülmektedir.



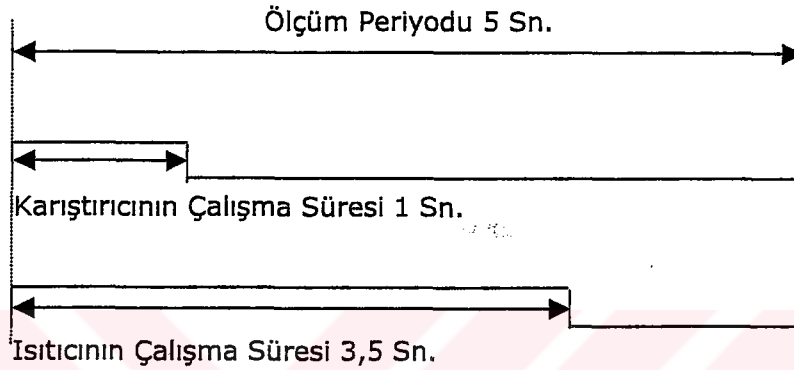
Şekil 6.7. Kazanç ve Filtre Devresi

6.1.2. Çıkışlar

Çıkışlarda bir adet karıştırıcı ve bir adet ısıtıcı bulunmaktadır. Bunlar kontrol kartından kumanda edilen röleler tarafından anahtarlanmaktadır.

Röleler ON konumunda iken karıştırıcı ve ısıtıcıda devreye girmektedir.

Isıtıcıların güç kontrolü bir sonraki ölçüme kadarki periyodun yüzdesi ile ifade edilebilecek ısıtma süresi ile yapılır. Örneğin çıkış değerleri ısıtıcı için %70, karıştırıcı için %20 ve kontrol periyodu 5sn ise karıştırıcı ve ısıtıcı rölelerinin açık kalma süreleri Şekil 6.8’de görüldüğü gibidir.

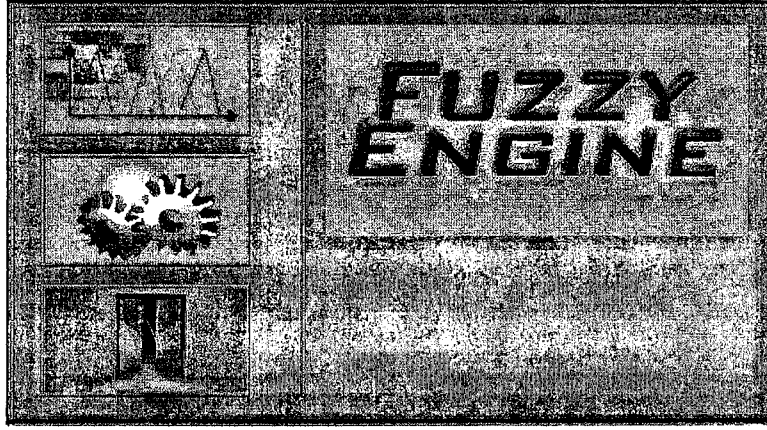


Şekil 6.8. Karıştırıcı Ve Isıtıcının Çalışma Süreleri

6.2. Sistemi Tanımlayan Bulanık Mantık Üyelik Fonksiyonlarının Tanımlanması

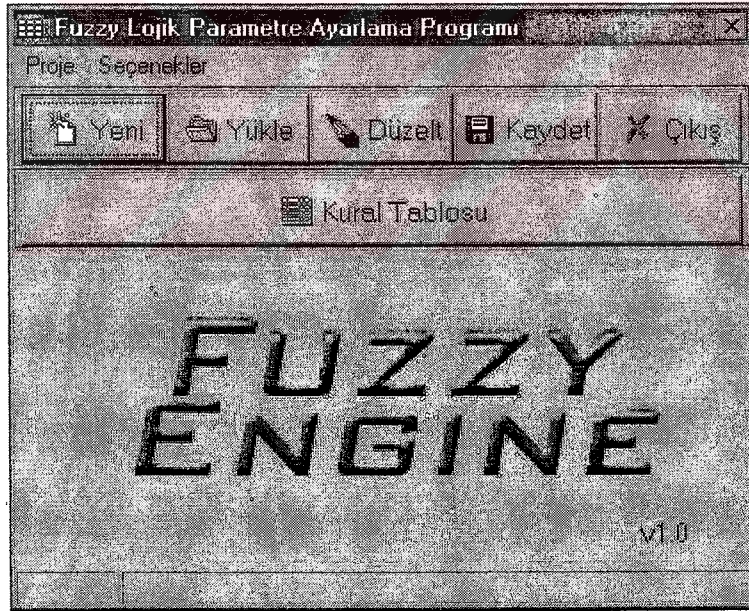
Sistemin kontrolüne geçmeden önce kontrol edilecek sisteme ait giriş ve çıkışların tespit edilmesi gerekmektedir. Daha sonra girişlerin çıkışlara olan etkileri incelenmelidir.

Sistemin kontrol edilebilmesi için öncelikle sisteme ait verilerin girildiği kural modülünün seçilmesi gerekmektedir. Bunun için Şekil 6.9’da görülen Fuzzy Engine programında bulunan ikonlardan üsttekine basılır.



Şekil 6.9. Fuzzy Engine Programının Giriş Ekranı

Kural modülünde kontrol edilecek olan sisteme ait parametreler girilmektedir. Kontrol edilecek sistem üzerinde daha önce herhangi bir çalışma yapılmadığı için yeni butonuna tıklanır. Şekil 6.10'da Kural modülünün girişi görülmektedir.



Şekil 6.10. Kural Modülü Giriş Ekranı

Sistemin genel hatlarıyla tanımlandığı kısım ekrana gelir, Şekil 6.11'de kontrol edilecek sisteme verilen ad ile giriş ve çıkış sayıların tanımlandığı ekran görülmektedir.

Şekil 6.11. Sisteme Ait Genel Bilgilerin Girildiği Ekran

Giriş sayısı 2 olarak tespit edildiği için Şekil 6.12’de görülen girişe ait parametrelerin girileceği ekranda input1 ve input2 diye tanımlanan iki satır bulunmaktadır.

Parametre	Kanal	Min	Max	Değişken 1	Değişken 2	Değişken 3	Değişken 4	Değişken 5	Değişken 6
input1	2	0	5	None	None	None	None	None	None
input2	2	0	5	None	None	None	None	None	None

Şekil 6.12. Giriş Parametrelerin Girildi Ekran Görüntüsü

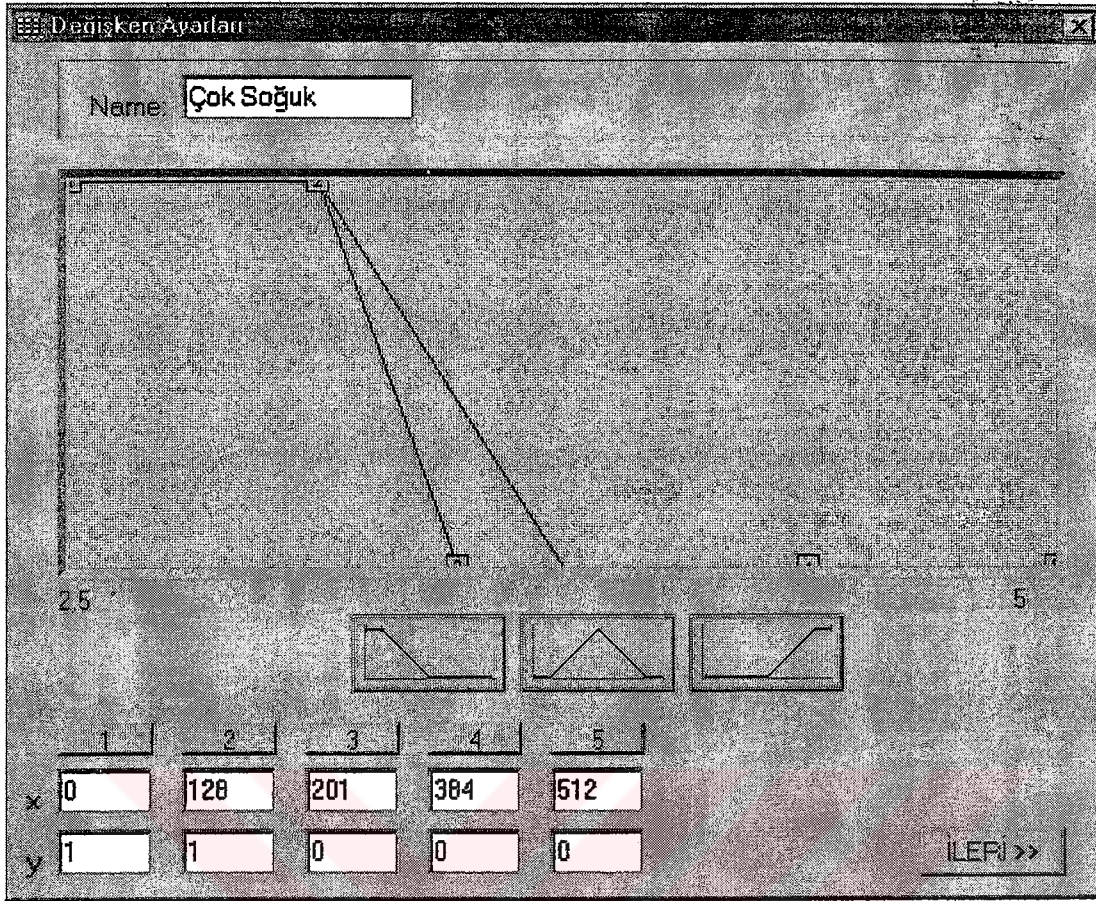
Girişe ait parametrelerin girileceği ekrana geçiş yapmak için input1 butonuna tıklanır. Şekil 6.13’de girişlerden birine ait parametrelerin girileceği ekran görüntüsü görülmektedir.

Şekil 6.13. Girişe Ait Özelliklerin Girildiği Ekran

Burada girişin adı, bu girişe ait küme sayısı, kontrol kartında hangi kanalın kullanılacağı ve girişe bağlı algılayıcıdan gelen genliğin aralığı girilmektedir.

Bu kontrol sisteminde 4 adet küme olacağına kara verilmiştir; algılayıcıdan gelen genliğin aralığı ise 2,5 ile 5 arasında olacaktır. Bunun sebebi kullanılacak olan havuzdaki suyun sıcaklığının yaklaşık olarak 20 °C ile 50 °C arasında olacağı düşünülmektedir; bu aralıkta ise algılayıcıdan gelen genlik yaklaşık 2,5V ile 5V arasında olacaktır.

Girişe ait girilen temel bilgilerden sonra girişe ait küme bilgilerin girilmesi gerekmektedir. Dört adet küme tanımlanacağı için giriş parametre ayarlarının yapıldığı ekranda dört değişken gözükür. Bunlardan Değişken-1'e tıklanarak birinci girişin birinci kümesine ait parametrelerin girileceği ekran Şekil 6.14'de görülmektedir.

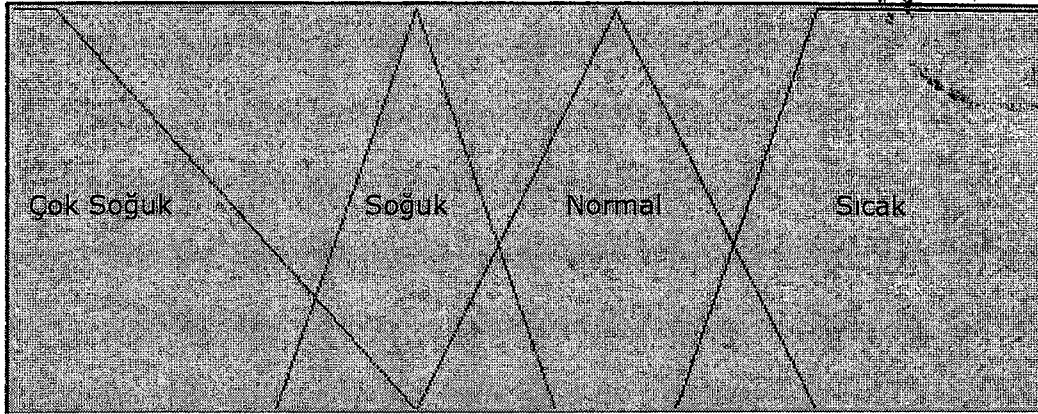


Şekil 6.14. Girişlere Ait Kümelerin Tanımlanması

Küme tanımlanması için 5 adet doğru parçası kullanılmaktadır, bunlar sayesinde bulanık kontrolde kullanılan belli başlı şekiller oluşturulabilmektedir. Giriş ekranının hemen altında bulunan üç adet buton ile daha kolay çizim yapılabilir. Doğru parçalarının uçlarında bulunan kırmızı çerçeveli kısma gelindiğinde fare imleci ok halinden el haline dönüşecek ve fare sayesinde bu noktanın konumu değiştirilebilecektir.

Çizim ekranının altın 2,5 ile 5 yazdığı görülmektedir bunun nedeni girişlerin tanımlandığı giriş ekranında algılayıcıdan gelen genliğin 2,5V ile 5V sınırlaması yapıldığı içindir. Gelen sinyal eğer 2,5V'dan düşük olursa bir başka değişle su sıcaklığı 17,5 °C'in altına düşerse kontrolör bunu 2,5V'a karşılık gelen sıcaklığı 17,5 °C olarak kabul edecektir.

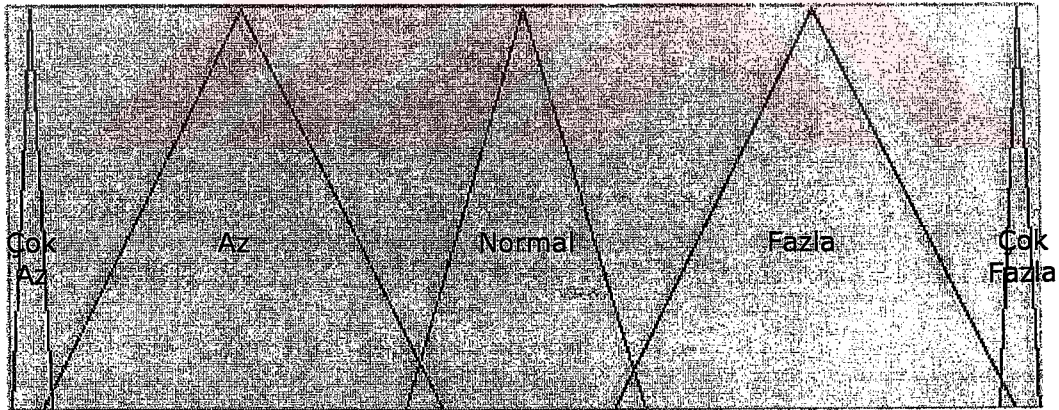
Tüm giriş ve bunlara ait kümeler bu şekilde tanımlanır. Şekil 6.15'de girişlere ait kümelerin hepsinin tanımlandığı ekran görülmektedir.



Şekil 6.15. Girişlere Ait Tüm Kümelerin Tanımlanmış Hali

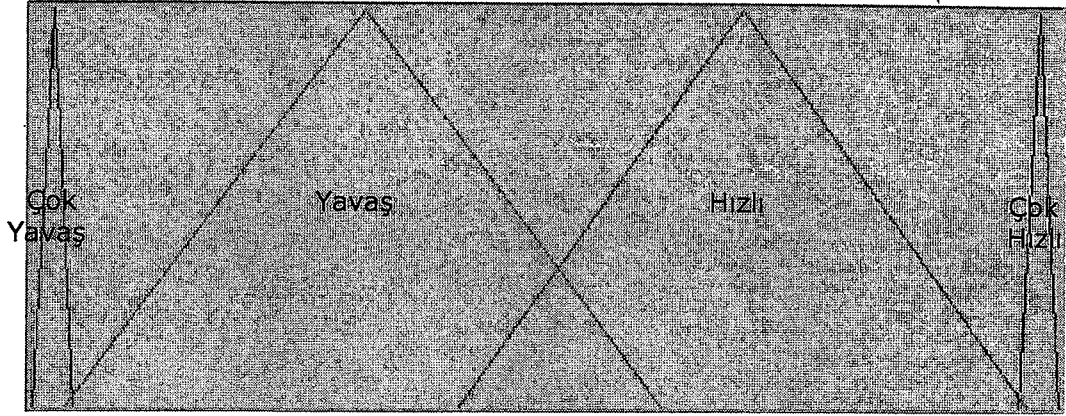
Çıkışlarda ise ısıtıcı ve karıştırıcıya ait kümelerin tanımlanması gerekmektedir. Isıtıcı çıkış değişkenine ait beş adet küme tanımlanmıştır.

Bunlar çok az, az, normal, fazla, çok fazla şeklindedir. Isıtıcı çıkış değişkenine ait küme tanımlamaları Şekil 6.16'da gösterilmektedir.



Şekil 6.16. Isıtıcıya Ait Küme Tanımlamaları

Karıştırıcı çıkış değişkene ait dört adet küme tanımlanmıştır. Bunlar çok yavaş, yavaş, hızlı, çok hızlı şeklindedir. Karıştırıcı çıkış değişkenine ait küme tanımlamaları Şekil 6.17'de gösterilmektedir.



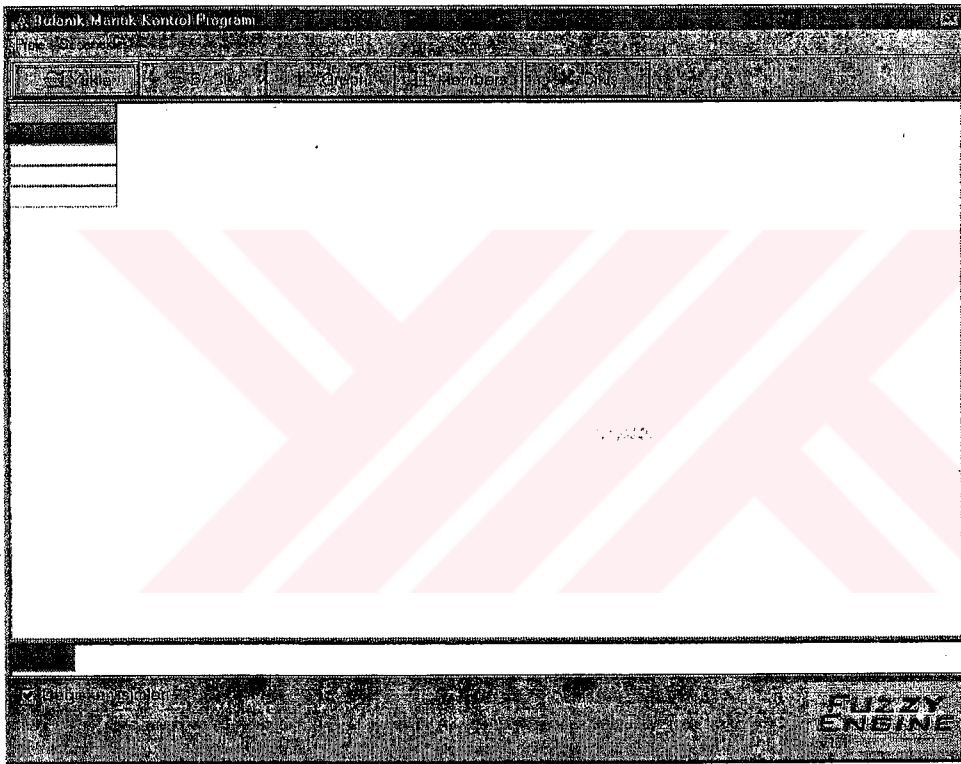
Şekil 6.17. Karıştırıcı Ait Küme Tanımlamaları

Giriş ve çıkışlara ait değişkenlerin küme tanımlamaları yapıldıktan sonra kural tablosunun çıkarılması gerekmektedir. Şekil 6.18’de bu sistem için oluşturulmuş kural tablosu görülmektedir.

Bulanık Kontrol Kural Tablosu			
İst.1	İst.2	Çıktı	Çıkış
Çok Soğuk	Çok Soğuk	Çok Az	Çok Yavaş
Çok Soğuk	Soğuk	Fazla	Yavaş
Çok Soğuk	Normal	Normal	Hızlı
Çok Soğuk	Sıcak	Az	Çok Hızlı
Soğuk	Çok Soğuk	Fazla	Yavaş
Soğuk	Soğuk	Normal	Çok Yavaş
Soğuk	Normal	Az	Yavaş
Soğuk	Sıcak	Çok Az	Hızlı
Normal	Çok Soğuk	Normal	Hızlı
Normal	Soğuk	Az	Yavaş
Normal	Normal	Çok Az	Çok Yavaş
Normal	Sıcak	Çok Az	Yavaş
Sıcak	Çok Soğuk	Az	Çok Hızlı
Sıcak	Soğuk	Çok Az	Hızlı
Sıcak	Normal	Çok Az	Yavaş
Sıcak	Sıcak	Çok Az	Çok Yavaş

Şekil 6.18. Kural Tablosu

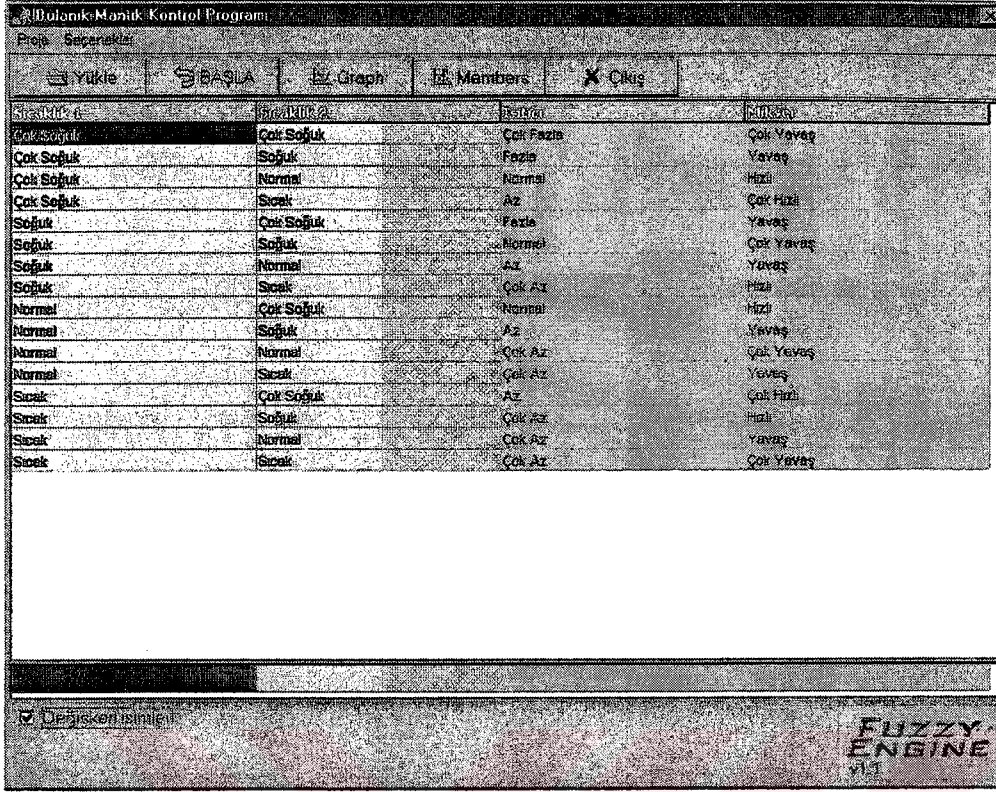
Kural tablolarının oluşturulmasının ardından bu sisteme ait kontrol projesini özetleyen dosyanın kaydedilmesi gerekmektedir, bunun için kural modülü girişine gelinir ve kaydet butonuna tıklanır. Belirlenen yere dosya oluşturulur. Bu dosyanın içinde giriş ve çıkışa ait bilgiler ile kural tablosuna ait bilgiler bulunmaktadır. Kural modülü kapatılmasından sonra Fuzzy Engine programının giriş ekranı gelir burada kontrol modülünü seçmek için ortadaki butona tıklanır. Bu sayede kontrol modülü çağrılmış olunur. Şekil 6.19’da kontrol modülü görülmektedir.



Şekil 6.19. Kontrol Modülü Giriş Ekranı

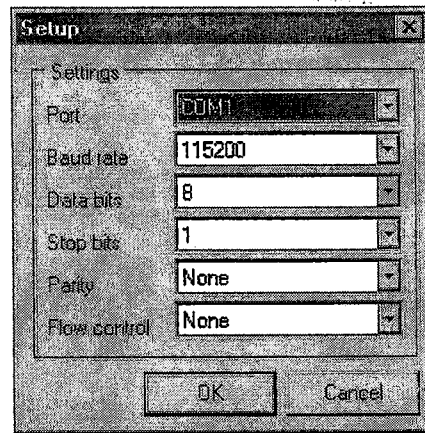
Yükle butonuna tıklanarak kontrol edilecek sisteme ait kural modülünde önceden hazırlanmış dosya çağrılır. Gelen ekranda kontrol edilecek sisteme ait girişler, çıkışlar ve bunlara ait kümeler görülmektedir.

Şekil 6.20’de kontrol modülüne kontrol edilecek sisteme ait bilgilerin bulunduğu kontrol modülü ekranı görülmektedir.



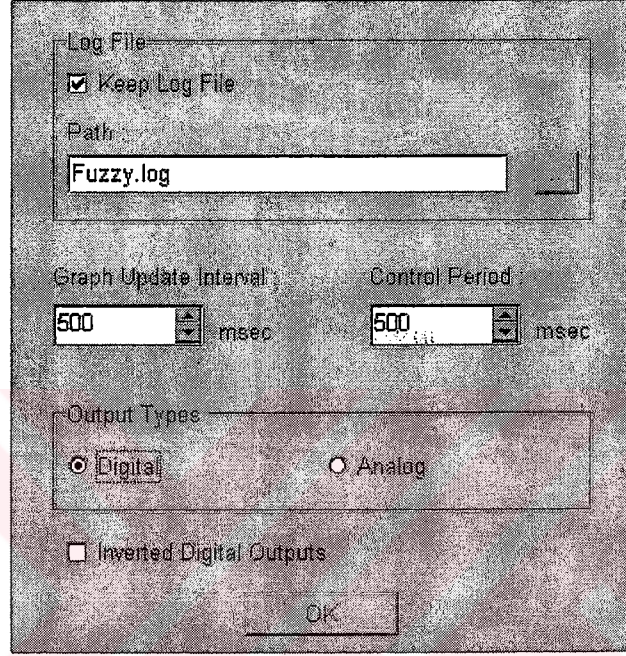
Şekil 6.20. Kontrol Edilecek Sisteme Ait Parametrelerin Olduğu Ekran

Burada kontrol kartının bağlı olduğu seri iskelenin seçilmesi gerekmektedir. Bunun için kontrol modülündeki seçeneklere tıklanır, buradan COM Port seçeneği tıklanır gelen ekranda kontrol kartının bağlı olduğu seri iskele seçilir. Şekil 6.21’de COM Port seçim ekranı görülmektedir.



Şekil 6.21. Seri İskele Seçim Ekranı

Seri iskele seçimi yapıldıktan sonra kontrol modülünde bulunan BAŞLAT butonuna tıklanır. Sistem kontrole başlamadan önce kontrolörün çalışma parametrelerinin girileceği ekran gelir. Şekil 6.22’de kontrol programına ait çalışma parametrelerin girildiği ekran görülmektedir.

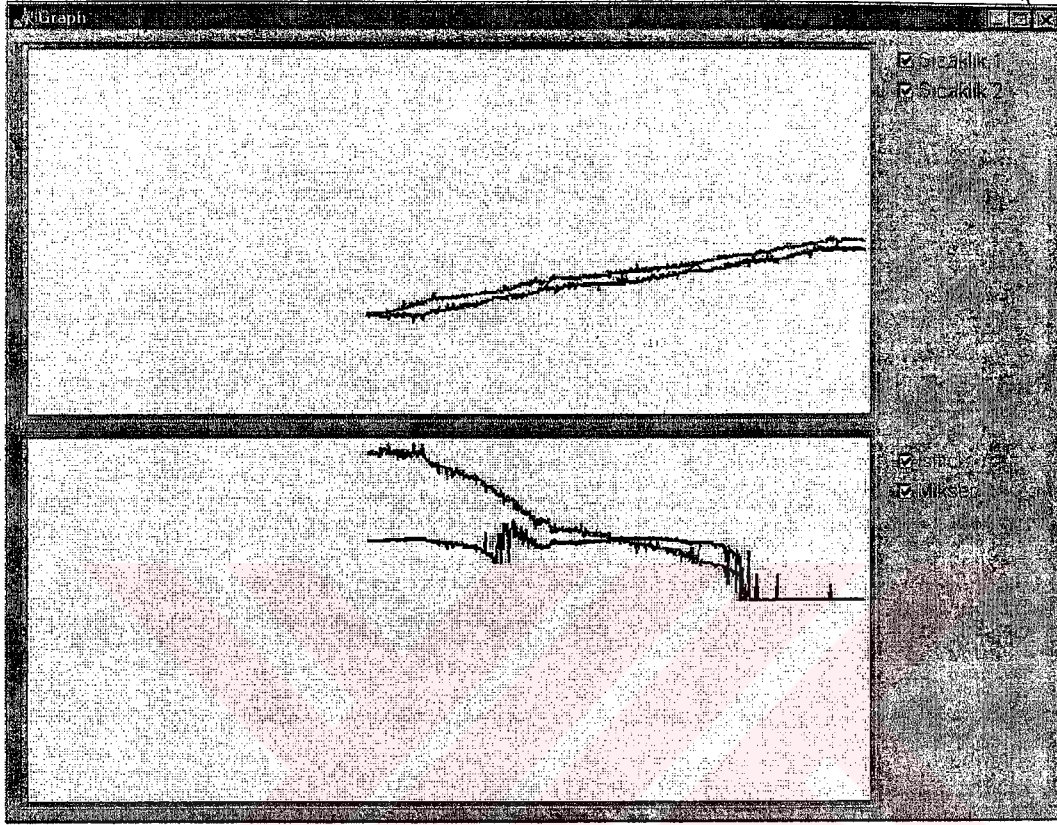


Şekil 6.22. Kontrol Programı Ayarları

Kontrol edilecek olan sisteme ait çalışma bilgilerinin tutulacağı log dosyası tutulmak isteniyorsa bunun yeri ve log dosyasının ismini girişi bu kısımda girilmektedir. Eğer konum belirtilmez ise log dosyası programın çalıştığı yere kaydedilecektir.

Sistemin kontrol periyodu ve grafik yenileme periyodu da burada girilmektedir. Bu periyod en az 50ms’dir. Kontrol edilecek olan sistemin çıkışlarına bağlı olarak çıkışlar analog veya dijital seçilir. TAMAM butonuna tıklandığında kontrol programı çalışmaya başlar.

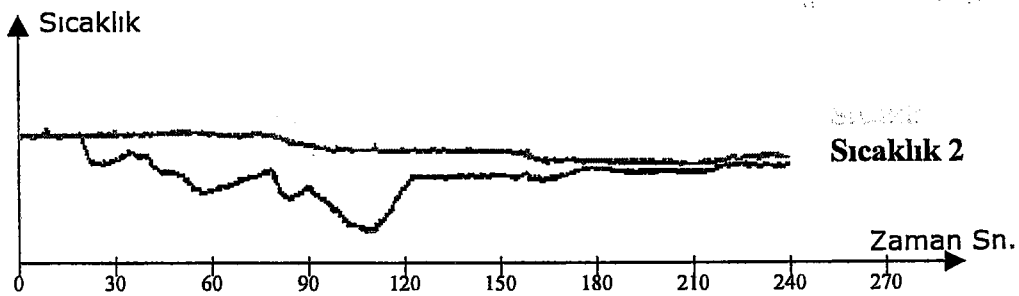
Şekil 6.23’de kontrol programının çalışmasıyla birlikte oluşan giriş ve çıkış değerleri görülmektedir.



Şekil 6.23. Kontrol Modülü Giriş Çıkış Ekranı

6.3. Sistemin Çıkış Verileri

Sisteme ait tüm tanımlamalar yapıldıktan sonra sistem çalıştırılmıştır. Öncelikle sisteme ait çıkışlar kullanılmadan sistemin davranışı gözlenmiştir. Şekil 6.24’de karıştırıcı ve ısıtıcı devre dışı iken elde edilen grafik görülmektedir.

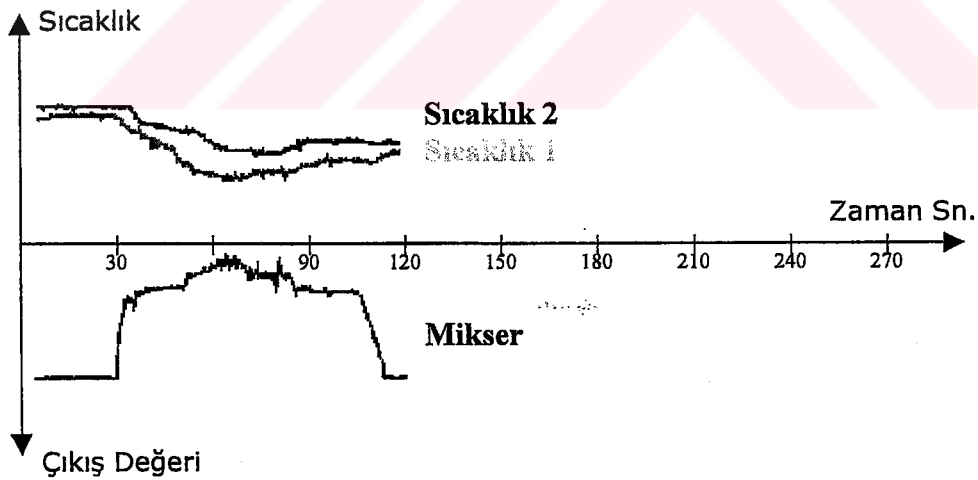


Şekil 6.24. Çıkışlar Pasif İken Sistem Davranışı

Sistem dengede iken sıcaklık ölçüm noktalarından birinin olduğu tarafa bir kalıp buz atılıp sistemin tekrar dengeye gelmesi beklenmiştir. Şekil 6.24'de buz kalıbının atılmasıyla geçen süre içerisinde iki ölçüm noktasındaki ölçüm değeri görülmektedir. Buz kalıbı Sıcaklık 2 şeklinde tanımlanan sıcaklık ölçüm noktasına atılmıştır. Sıcaklık 2 ölçüm noktasındaki sıcaklık hızla düşmüştür, belli bir süre sonra Sıcaklık 1 ölçüm noktasındaki sıcaklığın da düştüğü gözlenmiştir. Sıcaklık 2 ölçüm noktasındaki sıcaklık düşüşünün zamanla azaldığı, daha sonra sıcaklığın tekrar yükseldiği gözlenmiştir. Bu süre içerisinde Sıcaklık 1 ölçüm noktasındaki sıcaklık düşerek Sıcaklık 2 ölçüm noktasındaki sıcaklık değerine yaklaştığı ve sistemin dengeye geldiği görülmektedir.

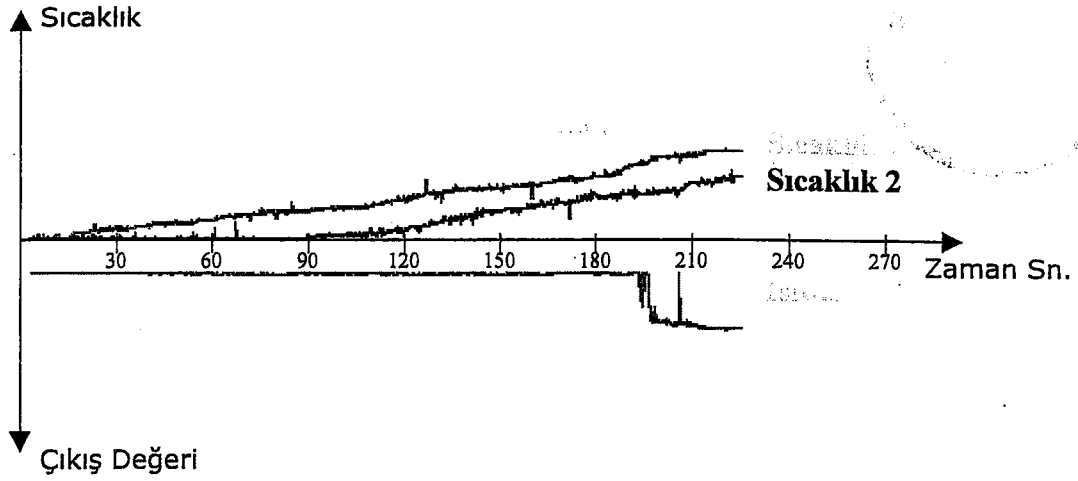
Sistem dengede iken bir buz kalıbı atılıp çıkışlardan karıştırıcı devrede iken çıkış verileri gözlenmiştir. Karıştırıcı kullanıldığında sistemin daha hızlı dengeye ulaştığı görülmüştür.

Şekil 6.25'de Karıştırıcı devredeyken sistemin davranışı görülmektedir.



Şekil 6.25. Sadece Karıştırıcı Kullanıldığında Sistem Davranışı

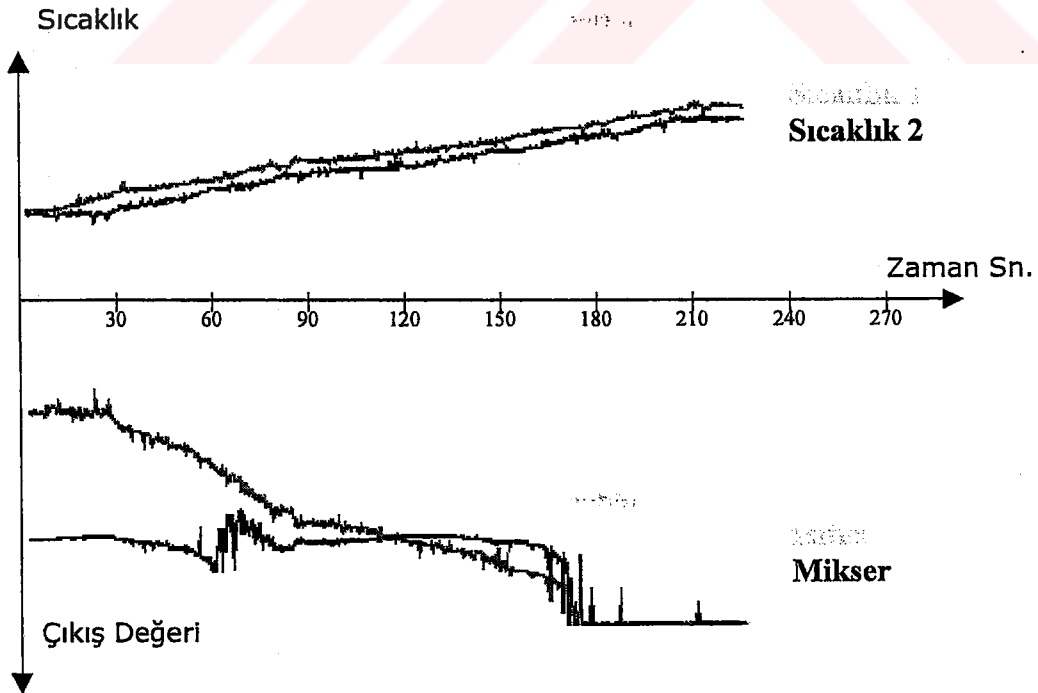
Sistem çıkışında sadece ısıtıcı varken sistem davranışı şekil 6.26'da görülmektedir. Kontrol edilecek sistemdeki suyun sıcaklığı yaklaşık 15 °C'da dengede olduğu bir sırada kontrolör çalıştırılmaya başlanmıştır.



Şekil 6.26. Sadece Isıtıcı Kullanıldığında Sistem Davranışı

Her iki ölçüm noktasındaki sıcaklık değeri de düşük olduğu için ısıtıcı en yüksek oranda çalışmaktadır. Isıtıcı Sıcaklık 1 ölçüm noktasına daha yakın olduğu için Sıcaklık 2 ölçüm noktasındaki sıcaklık artışı daha geç olmaktadır. Zamanla suyun sıcaklığı artmış, istenen değerlere ulaşmaya başladıkça ısıtıcının çalışma oranı da düşmüştür.

Bir sonraki deneyde hem ısıtıcı hem de karıştırıcı aynı anda kullanılmıştır. Şekil 6.27’de her iki çıkışında aktif olduğu deneyin çıkışı görülmektedir.

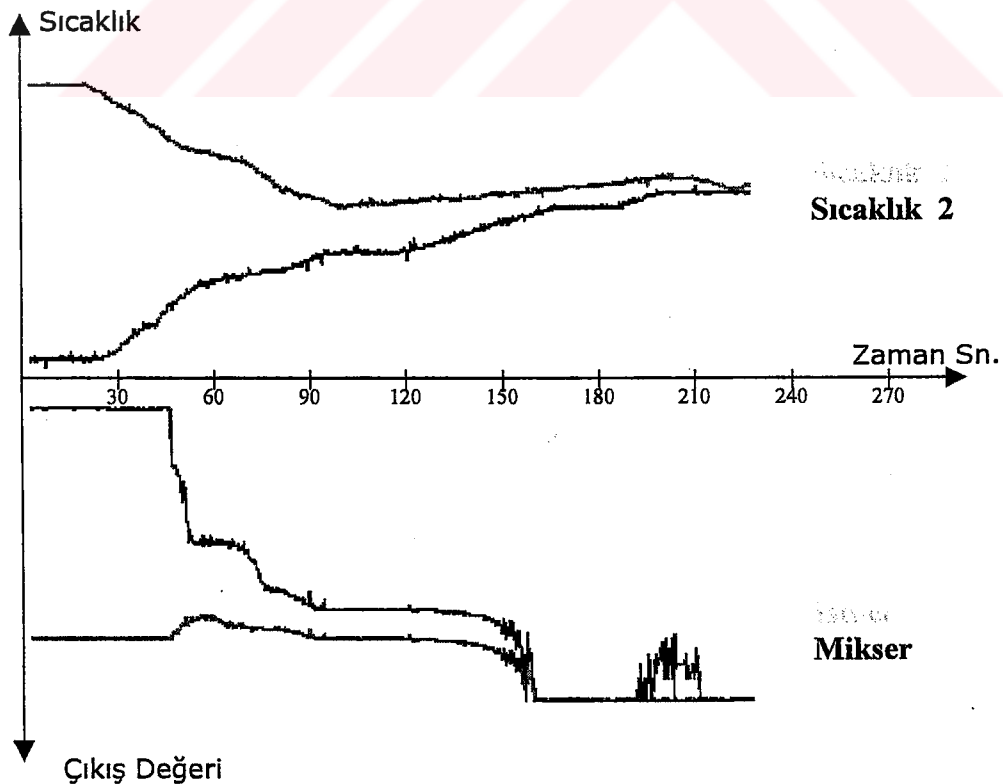


Şekil 6.27. Isıtıcı ve Karıştırıcı Kullanıldığında Sistem Davranışı

Sistem düşük bir sıcaklıkta dengede iken kontrolör çalıştırıldığında ısıtıcı en yüksek oranda çalışmaya başlıyor, zaman ilerledikçe suyun sıcaklığı artıyor ve ısıtıcının çalışma süresi de azalıyor. Sistemden ölçülen iki ayrı sıcaklık değeri yakında olsa karıştırıcı çalışmaya devam ediyor. Bunun nedeni, sistemden ölçülen iki sıcaklık değeri birbirinin aynı olsa dahi, eğer sıcaklık düşükse ve ısıtıcı çalışıyor ise karıştırıcıyı çalışmaktadır. Sistemde bulunan sıcaklık ölçüm noktalarından alınan değerler yükseldikçe ısıtıcının çalışma süresi düşmekte; ölçülen değerler birbirine yakın olduğu için de karıştırıcının çalışma oranı düşmektedir.

Sistemde her iki çıkış yine aktifken bu sefer iki ölçüm noktasındaki sıcaklık değerleri birbirinden olabildiğince farklı yapıyor. Bunun için iki ölçüm noktası arasına bir yalıtkan konarak bir taraftaki su, ısıtıcı ile ısıtılmakta diğer taraftaki ölçüm noktasının olduğu taraftaki suya ise buz eklenmemiştir.

Sistem kontrol edilmeden önce iki ölçüm noktasındaki sıcaklık farkı yaklaşık olarak 40 °C yapılmıştır. Sistemin davranışları şekil 6.28'de görülmektedir.



Şekil 6.28. Giriş ve Çıkış Çok Farklı Olduğunda Sistem Davranışı

Sistem kontrole başlandığında iki ölçüm noktası arasındaki sıcaklık farkı çok fazla olduğu için karıştırıcı çalışma oranı çok fazla olacaktır. İki ölçüm noktasından alınan değerlerin birisi çok düşük olmasına karşın diğeri çok yüksek olduğu için ısıtıcı çalışma oranı %30'larda kalmaktadır. Her iki ölçüm noktasındaki sıcaklık farkı azaldıkça karıştırıcının çalışma oranının düştüğü görülmektedir.



7. SONUÇ

Bu çalışmada amaçlanan, bulanık kontrolün uygulanabilirliğini arttırmak için daha az donanımlı ve daha kısa bir süre içerisinde herhangi bir sistemin kontrolünü gerçekleştirebilecek bilgisayar tabanlı bir kontrol sistemi oluşturmaktır.

Sistem, kontrol arabirimi ve bunu süren yazılım paketi olmak üzere iki kısımdan oluşmuştur.

Yazılım programında hedeflenen, kullanıcı tarafından girilecek olan bulanık kontrol parametrelerinin rahatça girilebilmesini sağlamak, zaman içerisinde bunlar üzerinde değişiklik yapabilmek ve hızlı bir şekilde kontrolü gerçekleştirip kontrol sonuçlarının grafik ortamına taşınması ve dosya olarak saklanabilmesiydi.

Bu amaç ile iki ayrı modülden oluşan bir yazılım paketi yapıldı. Bunlardan birincisi kural modülü ikincisi ise kontrol modülüdür. Kural modülünde bulanık kontrole ait parametreler girilebileceği, üyelik fonksiyonların rahatça çizilebileceği bir giriş ekranı oluşturulmuştur. Kural modülü, yapılan tüm girişleri bir dosyaya kaydetmekte ve daha sonra bu dosya çağrılarak üzerinde değişiklik yapmaya olanak sağlamaktadır. Kontrol modülü ise kontrol edilecek olan sistem ile veri alış verişi yapmaktadır. Kural modülünde oluşturulan dosya, kontrol modülü tarafından okunmakta, kullanıcı tarafından belirlenen kontrol periyot içerisinde girişlerden bilgileri alıp işleyip çıkan sonuçları arabirim kartına yollamaktadır. Ayrıca gelen giriş bilgilerinin ve gönderilen çıkış bilgilerinin bir grafik ortamında çizilmesini sağlamaktadır. İstendiği takdirde her ölçüm periyodunda kullanıcı tarafından belirtilen yer ve isimde çalışma bilgilerinin tutulduğu bir dosya oluşturulmaktadır. Bu dosya içerisinde kontrolün yapıldığı zaman ve her ölçüm anındaki giriş ve çıkış verileri bulunmaktadır. Bu sayede daha önceden yapılan bir kontrole ait veriler incelenebilmektedir.

Arabirim kartında hedeflenen ise çok sayıda giriş çıkışa sahip harici donanımlara gerek kalmadan kontrol sistemine bağlanabilen ve çok fazla ayar gerektirmeden kullanılacak bir donanım gerçekleştirmektir.

Gerçekleştirilen arabirim kartından 16 analog giriş, 16 sayısal çıkış veya 4 analog çıkış bulunmaktadır. 0V ile 5V arasında girişlerin yapıldığı analog girişler 10 bitlik sayısal bilgiye dönüştürülerek bilgisayara yollanmaktadır. Arabirim kartının bilgisayar ile haberleşmesi RS485 protokolü ile yapılmakta; bu sayede bilgisayar ile arabirim kartı arasındaki mesafeler uzak olabilmektedir. RS485'in bir avantajı ise özellikle endüstriyel ortamlarda oluşan gürültüye karşı dirençli olmasıdır. Bu sayede bilginin bozulma riski daha azdır. Kontrolör 20 MHz'de çalışmakta; bu da kontrol işlemlerinin hızlı yapılmasına olanak sağlamaktadır.

Gerçekleştirilen bu sistem ile gerçek bir sistem kontrol edilmiştir. Buna ait veriler ve çalışmalar Bölüm 6'da ayrıntılı verilmiştir. Bölüm 6'da görüldüğü gibi kontrol sistemi kendinden beklenen performansı göstermiştir. Çok kısa zaman içerisinde çok az hata payları ile sistemi dengede tutabilmiştir.

Bu sistem ile bulanık mantıkla ifade edilebilecek her türlü prosesi yeterli performansla kontrol edebileceğimiz anlaşılmıştır.

Ayrıca, bu çalışma ile edinilen donanım ve yazılım geliştirme tecrübesinin, bilimsel bir bakış açısıyla birçok mühendislik uygulamasına aktarılması mümkün olacaktır.

KAYNAKLAR

- [1] BEZDEK, J.C., Introduction to Fuzzy Logic, IEEE transactions on fuzzy systems Vol.1 No.1 (1-6), 1993.
- [2] PATYRA, M.J. and MLYNEK, D.M., Fuzzy Logic, John Wiley & Sons Ltd., England, (117-138), 1996.
- [3] KAUFMANN, A., and GUPTA, M.M., Introduction to Fuzzy Arithmetic, Theory and Applications, Van Nostrand Reinhold, New York, (11-22), 1991.
- [4] BANDEMAR, H. and GOTTWOLD, S., Fuzzy Sets and Fuzzy Logic, Fuzzy Methods With Applications, Biddles Ltd., Great Britain, (5-14), 1996.
- [5] KLIR, G.J., and YUAN, B., Fuzzy Theory and Applications, Prentice Hall PTR, New Jersey, (5-30), 1995.
- [6] ZADEH, L.A. and KACPRZYK, J., Fuzzy Logic for the Management of Uncertainty, Courier Companies, New York, (213-214), 1992.
- [7] TERANO, T., ASIA, K. and SUGENO M., Fuzzy Systems Theory, Academic Press, Inc., San Diego, (19-37), 1992.
- [8] WANG, L., A Course In Fuzzy Systems and Control, Prentice Hall PTR, New Jersey, (20-29), 1997.
- [9] FOLGER, T.A., Fuzzy Sets, Prentice Hall, New York, (32-36), 1988.
- [10] KAEWOWSKI, W., Applications of Fuzzy Set Theory in Human Factors, Elsevier Science Publishers, New York, (40-65), 1986.
- [11] SARIOĞLU M.K., Otomatik Kontrol 1, Birsen Yayınevi Ltd. Şti., İstanbul, (1-26), 1997.
- [12] ITOH, O., Industrial Application of Fuzzy Technology, Springer-Verlag, Tokyo, (135-158), 1993.
- [13] KRUSE, R., GEBHARDT, J. and KLAWONN, F., Foundations of Fuzzy Systems, , John Wiley & Sons Ltd., England, (160-181), 1994.
- [14] ESİN, E.M., PC Tabanlı Proses Kontrol İçin Genel Amaçlı Bir Türkçe Scada Ortamı, (1-7), GYTE Araştırma Fonu Projesi No:96-B-03-04-01, 2000.

ÖZGEÇMİŞ



1975 yılında Eskişehir’de doğan Serkan Çağlar ÜNAL ilk ve orta öğretimini Mersin’de tamamladı. 1992 yılında Marmara Üniversitesi Teknik Eğitim Fakültesi Elektronik ve Bilgisayar Bölümü’nde başladığı lisans eğitimini 1997 yılında başarıyla tamamladı. Gebze Yüksek Teknoloji Enstitüsü Bilgisayar Mühendisliği’nde 1998 yılında Y. Lisans eğitimine başladı. Şu anda TUBİTAK-UEKAE’de araştırmacı olarak çalışmaktadır.

