

**YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**PID YÖNTEMİNİN PLC'DE YAZILARAK
GERÇEKLENMESİ: ÇİFT CİDARLI REAKTÖR TANK
SICAKLIK KONTROLÜ ÖRNEĞİ**

Elektrik Müh. Mehmet Aslan YERSEL

**FBE Elektrik Mühendisliği Anabilim Dalı Kontrol ve Otomasyon Programında
Hazırlanan**

YÜKSEK LİSANS TEZİ

Tez Danışmanı: Yrd. Doç. Dr. Şeref Naci ENGİN

İSTANBUL, 2007

İÇİNDEKİLER

	Sayfa
SİMGE LİSTESİ	iv
KISALTMA LİSTESİ	v
ŞEKİL LİSTESİ	vi
ÇİZELGE LİSTESİ	viii
ÖNSÖZ	ix
ÖZET	x
ABSTRACT	xi
1. GİRİŞ	1
2. PID KONTROL	3
2.1 PID Denetleyicinin Tanımı	3
2.2 PID Denetleyici Tasarımı	4
2.3 Denetleyici Tipleri	5
2.3.1 Oransal (P) Denetleyici	6
2.3.2 Oransal - Türevsel (PD) Denetleyici	8
2.3.3 Oransal - İntegral (PI) Denetleyici	12
2.3.4 Oransal - İntegral - Türevsel (PID) Denetleyici	16
2.4 Genel PID Denetleyici Tasarımı İçin İzlenecek Yol	19
2.5 Ziegler - Nichols Metodu	19
2.6 Cohen - Coon Metodu	21
2.7 Haalman Metodu	22
2.8 İntegral Yığılması (Anti - Windup)	22
3. DİJİTAL PID DENETLEYİCİ	24
3.1 Dijital PID Denetleyicinin Oluşturulması	24
3.2 Artımsal PID	25
4. ÇİFT CİDARLI (CEKETLİ) REAKTÖR TANK ISITMA SİSTEMİNİN MATEMATİKSEL MODELİ	26
4.1 Giriş	26
4.2 Matematiksel Model	27
4.3 Doğrusallaştırma	28
4.3.1 Bir Durum Değişkeni İçin Doğrusallaştırma	28
4.3.2 Tek Durum Tek Değişkenli Sistem İçin Doğrusallaştırma	29
4.3.3 Çıkış Değişkeni İçin Doğrusallaştırma	29
4.3.4 Genel Doğrusallaştırma	30

4.4	Çift Cidarlı Isıtma Sisteminin Doğrusallaştırılmış Durum Denklemleri.....	30
4.5	Uygun PID Katsayılarının Seçimi	35
4.5.1	Ziegler – Nichols Yöntemiyle Katsayıların Belirlenmesi	36
4.5.2	Cohen - Coon Yöntemiyle Katsayıların Belirlenmesi	39
5.	PLC 'NİN YAPISI VE PROGRAMLANMASI	42
5.1	PLC' nin Yapısı	42
5.1.1	CPU (Central Processing Unit).....	42
5.1.2	Bellek (Memory)	43
5.1.3	Giriş/Çıkış Bölümü (I / O).....	43
5.2	PLC' nin Programlanması	43
6.	ÇİFT CİDARLI REAKTÖR TANK ISITMA SİSTEMİNİN PLC İLE KONTROLÜ	50
6.1	Akış Diyagramı.....	51
6.2	PLC Programı	52
7.	DENEYSEL SONUÇLAR	58
8.	SONUÇLAR VE TARTIŞMA	60
	KAYNAKLAR.....	61
	ÖZGEÇMİŞ.....	62

SİMGE LİSTESİ

K_p	Analog Oransal Kazanç
K_i	Analog İntegral Kazanç
K_d	Analog Türevsel Kazanç
K_P	Dijital Oransal Kazanç
K_I	Dijital İntegral Kazanç
K_D	Dijital Türevsel Kazanç
ω_n	Doğal Frekans
ζ	Sönüm oranı
A	Isı transferi için yüzey alanı
T	Reaktördeki ürün sıcaklığı
T_{jin}	Cekete giren suyun sıcaklığı
F	Debi
V	Reaktör hacmi
t_s	Yerleşme Zamanı
t_r	Yükselme Zamanı
e_{ss}	Kalıcı Hal Hatası
c	Sıcaklık kapasitesi
Q	Isı transfer oranı
ρ	Yoğunluk
U	Isı transfer katsayısı
τ_d	Zaman Gecikmesi
τ	Zaman Sabiti

KISALTMA LİSTESİ

P	Proportional (Oransal Kontrolör)
I	Integral (İntegral Kontrolör)
D	Derivative (Türevsel Kontrolör)
PI	Proportional Integral
PD	Proportional Derivative
PID	Proportional Integral Derivative
PLC	Programmable Logic Controller (Programlanabilir Lojik Kontrolör)

ŞEKİL LİSTESİ

	Sayfa
Şekil 2.1 PID denetleyicili sistem	3
Şekil 2.2 Geri beslemeli sistem	4
Şekil 2.3 Örnek sistemin birim basamak cevabı	5
Şekil 2.4 Oransal denetleyicili sistem	6
Şekil 2.5 Oransal kontrollü sistemin birim basamak cevabı	7
Şekil 2.6 Tek opampli P denetleyici devresi	8
Şekil 2.7 İki işlemsel kuvvetlendiricili P denetleyici devresi.....	8
Şekil 2.8 PD kontrollü sistem.....	9
Şekil 2.9 PD kontrollü sistemin birim basamak cevabı.....	10
Şekil 2.10 Tek opampli PD denetleyici.....	11
Şekil 2.11 Kuvvetlendiricili PD denetleyici.....	11
Şekil 2.12 Üç işlemsel kuvvetlendiricili PD denetleyicinin devre şeması	12
Şekil 2.13 PI denetleyicili sistem	12
Şekil 2.14 PI denetleyicili sistemin birim basamak cevabı	13
Şekil 2.15 Tek opampli PI denetleyici	14
Şekil 2.16 Kuvvetlendiricili PI denetleyici	15
Şekil 2.17 Üç işlemsel kuvvetlendiricili PI denetleyicinin devre şeması.....	15
Şekil 2.18 PID kontrollü sistem.....	16
Şekil 2.19 PID kontrollü sistemin birim basamak cevabı	17
Şekil 2.21 İki işlemsel kuvvetlendiricili PID devre şeması.....	18
Şekil 2.22 Üç işlemsel kuvvetlendiricili PID devre şeması	19
Şekil 2.23 Ziegler – Nichols reaksiyon eğrisi.....	21
Şekil 2.24 Anti - windup	23
Şekil 4.1 Çift cidarlı (ceketli) karıştırılmalı tank ısıtıcı	26
Şekil 4.2 Tank ısıtma sisteminin PID denetleyicili blok diagramı	36
Şekil 4.3 Tank ısıtma sisteminin açık çevrim birim basamak cevabı.....	37
Şekil 4.4 Ziegler - Nichols tablosuyla belirlenen PID katsayılarının sistem cevabı	38
Şekil 4.5 Cohen - Coon tablosuyla belirlenen PID katsayılarının sisteme etkisi	40
Şekil 4.6 Uygun PID katsayılarıyla sistemin birim basamak fonksiyonuna cevabı.....	41
Şekil 5.1 Load komutu	43
Şekil 5.2 Load not komutu	44
Şekil 5.3 Out komutu.....	44

Şekil 5.4 Or komutu	44
Şekil 5.5 Timer komutu	45
Şekil 5.6 Counter komutu	45
Şekil 5.7 Move komutu	46
Şekil 5.8 Greater than komutu	46
Şekil 5.9 Greater than or equal komutu	46
Şekil 5.10 Less than komutu	47
Şekil 5.11 Set komutu	47
Şekil 5.12 Reset komutu	47
Şekil 5.13 Sub komutu	48
Şekil 5.14 Cpt komutu	48
Şekil 5.15 PID komutu	49
Şekil 5.16 PID bloğu ayar penceresi	49
Şekil 5.17 End komutu	49
Şekil 6.1 Kontrol edilecek sistemin blok diyagramı	50
Şekil 6.2 PLC programının akış diyagramı	51
Şekil 6.3 Program Satırı 1	52
Şekil 6.4 Program satırı 2	53
Şekil 6.5 Program satırı 3	53
Şekil 6.6 Program satırı 4	54
Şekil 6.7 Program satırı 5	54
Şekil 6.8 Program satırı 6	55
Şekil 6.9 Program satırı 7	55
Şekil 6.10 Program satırı 8 ve 9	56
Şekil 6.11 Program satırı 10 ve 11	56
Şekil 6.12 Program satırı 12	57
Şekil 6.13 Veri monitörü aracılığıyla PID değerlerinin atanması	57
Şekil 7.1 PID katsayılarının PLC 'ye aktarımı	58
Şekil 7.2 Uygun PID katsayılarıyla sistemin cevabı	58

ÇİZELGE LİSTESİ

	Sayfa
Çizelge 2.1 Denetleyicilerin etkileri.....	5
Çizelge 2.2 Oransal denetleyicinin sisteme etkileri	7
Çizelge 2.3 PD denetleyicinin sisteme etkileri.....	10
Çizelge 2.4 PI denetleyicinin sisteme etkileri	14
Çizelge 2.5 PID denetleyicinin sisteme etkileri	17
Çizelge 2.6 Ziegler - Nichols osilasyon metodu katsayıları.....	20
Çizelge 2.7 Ziegler – Nichols reaksiyon eğrisine göre PID parametreleri.....	21
Çizelge 2.8 Cohen – Coon metodu katsayıları	22
Çizelge 2.9 Haalman metodu katsayıları.....	22
Çizelge 4.1 Sistem parametreleri.....	27
Çizelge 4.2 Ziegler – Nichols ile belirlenen katsayılara göre sistem karakteristikleri.....	38
Çizelge 4.3 Cohen - Coon ile belirlenen katsayılara göre sistem karakteristikleri.....	40
Çizelge 4.4 Uygun PID katsayılarına göre sistem karakteristikleri.....	41
Çizelge 6.1 PLC programında kullanılan parametrelerin adresleri	52

ÖNSÖZ

Tez çalışmalarımın her adımını titizlikle takip ederek bilgilerini, tecrübelerini ve yardımlarını esirgemeyen değerli hocam ve danışmanım Sayın Yrd. Doç. Dr. Şeref Naci ENGİN'e sonsuz teşekkürlerimi ve saygılarımı sunarım.

Ayrıca çalışmalarım sırasında bana her yönden destek olan arkadaşlarım ile maddi ve manevi katkılarını hiçbir zaman benden esirgemeyen değerli aileme en içten teşekkürlerimi bir borç bilirim.

ÖZET

Son yıllarda endüstriyel uygulamalarda sıkça kullanılmaya başlanılan PID denetleyicilerin yapısı, tasarımı ve uygulamasının ele alındığı bu çalışmada bir çift cidarlı (ceketli) reaktör tank ısıtma sisteminin PLC tabanında PID yöntemiyle kontrolü yapılmıştır.

Bu çalışmada öncelikle temel kontrol sistemleri hakkında bilgi verilmiştir. Kullanılan çeşitli denetleyicilerin yapıları tanıtılmış, bunların sistem karakteristiklerine etkileri anlatılmıştır. Bir çift cidarlı reaktör tank sisteminin doğrusallaştırılmış transfer fonksiyonu elde edilerek PID ile kontrol edilmiştir. PID katsayılarının belirlenmesinde Cohen - Coon, Ziegler - Nichols gibi ayar yöntemleri kullanılmıştır.

MATLAB ortamında incelenen kontrol sistemi endüstride en çok kullanılan denetleyici olan PLC' ye aktarılmış, PLC algoritması oluşturulan PID denetleyicisiyle elde edilen deneysel sonuçlar sunulmuş ve yorumlanmıştır. PLC 'de yazılan program, kullanım rahatlığı olan ve endüstriyel uygulamalara uygun bir PID denetleyicisi sağlamıştır.

Anahtar Kelimeler: PLC, PID, Sıcaklık Kontrolü, Çift Cidarlı (Ceketli) Reaktör

ABSTRACT

PID controllers have been in use immensely in industrial applications since mid 1970s. In this thesis, PID controllers' architecture, design and applications have been studied. Also, jacketed reactor tank heating system has been controlled with a PID control method on a PLC.

First, basic control systems are introduced in this study. Then, different controllers' architectures and their effects to system characteristics are introduced. Next, transfer function of jacketed reactor tank heating system, which has been linearized, is controlled with a PID method. PID coefficients are determined with different PID tuning methods like Cohen - Coon and Ziegler - Nichols.

The control system, which is initially analyzed on MATLAB environment, is transferred to PLC. PID controller has been programmed in the PLC and the obtained results are commented. The program coded in PLC provided on easy to use and industrially applicable PID software tool.

Keywords: PLC, PID, Temperature Control, Jacketed Reactor

1. GİRİŞ

Otomatik kontrol sistemleri, ait olduğu sistemi insan müdahalesi gerektirmeksizin arzu edilen değerlerde tutmayı amaçlayan sistemlerdir. İyi bir kontrol sisteminde giriş büyüklükleri değiştikçe çıkış büyüklükleri de mümkün olduğu kadar kısa bir sürede ayar değerlerine ulaşır ve bozucu büyüklükler tarafından meydana gelen değişimler sistem tarafından kendi kendine giderilir. Bu özellikleri içermeyen kontrol sistemine istenen özellikleri sağlayacak bir denetleyici eklenerek istenilen koşulların sağlanmasına çalışılır.

Endüstride kullanılan en önemli denetleyicilerden PLC “Programlanabilir Lojik Kontrolör” İngilizce kelimelerinin baş harflerinin kısaltılması ile oluşur. Endüstriyel uygulamaların her dalında yapılan genel amaçlı kumanda ve otomasyon çalışmalarının bir sonucu olan PLC tekniği 25 yıl önce sanayi uygulamalarında kullanılmaya başlanmıştır. Son 10 yıldır bir çok firmanın tabanı ve programlama mantığı birbirine çok yakın fakat kendi aralarında değişik üstünlükler ile ayrılan PLC sistemlerini geliştirmeleri ile otomatik kontrol sistemlerinde hız, kontrol, güvenlik, ürün kalitesi artmıştır. Ayrıca sistemin değiştirilmesi için kumanda devrelerinin yeniden oluşturulması yerine sadece PLC programlama ile giderilmesi çok büyük bir avantaj sağlamıştır. Getirdiği bu kolaylıklar PLC’ nin endüstride vazgeçilmez bir denetleyici olarak kullanılmasını sağlamış, aynı zamanda her geçen gün yeni özellikler ile güncelleştirilmesi gereğini doğurmuştur.

PID (Proportional Integral Derivative) ise günümüzde en çok kullanılan kontrol yöntemlerinden biridir. P oransal denetleyiciyi, I integral denetleyicisini, D ise türevsel denetleyiciyi temsil etmektedir. Hatasız bir kontrol elde etmek için, kontrol edilen işaret geri beslenir ve referans işaretle karşılaştırılır. Giriş ve çıkış işaretleri arasındaki fark ile orantılı bir sürücü işaret PID denetleyici tarafından hesaplanarak hatayı gidermek üzere sisteme uygulanır. Bu şekilde hata değeri sıfıra getirilmeye çalışılır. Ancak hata sıfırlanırken, sistemin aşım ve oturma zamanı değerlerinede dikkat edilmelidir.

PID denetleyicisinin katsayılarını bulmak için bir çok farklı metod vardır. Sıcaklık kontrolü gibi yavaş ve gecikmeli sistemlerde Ziegler – Nichols kullanımı kolay olduğu, ayrıca uygun PID katsayılarına ulaşmada güzel bir başlangıç noktası verdiği için tercih edilmelidir.(Dreinhofer, 1988)

PLC endüstri uygulamalarında kullanıldığı gibi akademik çalışmalarda da faydanılan bir denetleyicidir. Robert Ziemba yaptığı çalışmada (1988) bir plastik enjeksiyon sisteminin sıcaklık, hız, basınç ve pozisyon kontrolünü PLC kullanarak gerçekleştirmiştir.

Pan ve Zheng ise 2004 yılında yaptıkları iki AC motorun hız senkronizesi çalışmasında PLC kullanmışlardır. Bu çalışma sırasında PLC içerisinde bulunan PID bloğunu kullanmalarına rağmen kullanıcıların kendi sistemlerine özel PID algoritmasını yazmalarının daha esnek olacağını belirtmişlerdir.

Bu PID blokları çok fazla parametre içermekte, aynı zamanda sistem kurulu olmadığı sürece çalışmamaktadır. Bir başka deyişle kullanıcıya simulasyon imkanı vermemektedir. Bu çalışmada amaç PLC programı üzerinde bir PID algoritması geliştirerek, matematiksel modeli bilinen bir sistemi PLC' ye aktarıp kontrolünü gerçekleştirebilmektir.

Burada örnek sistem olarak kimya, gıda ve çevre mühendisliği uygulamalarında yaygınlıkla kullanılmakta olan çift cidarlı reaktör tank ele alınmıştır. Sistemin doğrusallaştırılmış matematiksel modeli PLC' ye aktararak en önemli işletme parametrelerinden sıcaklığının kontrolü ele alınmıştır.

Bu çalışmada ikinci bölümde ilk olarak PID denetleyicinin yapısı, sisteme etkileri ve tasarımı üzerinde durulmuştur. Daha sonra üçüncü bölümde dijital PID denetleyici anlatılmış, PID denetleyicinin PLC' ye aktarılacağı form olan artımsal(hız) PID ele alınmıştır. Dördüncü bölümde çift cidarlı reaktör tankın doğrusallaştırılmış matematiksel modeli elde edilerek sistem PLC' ye aktarılmaya hazır hale getirilmiştir.

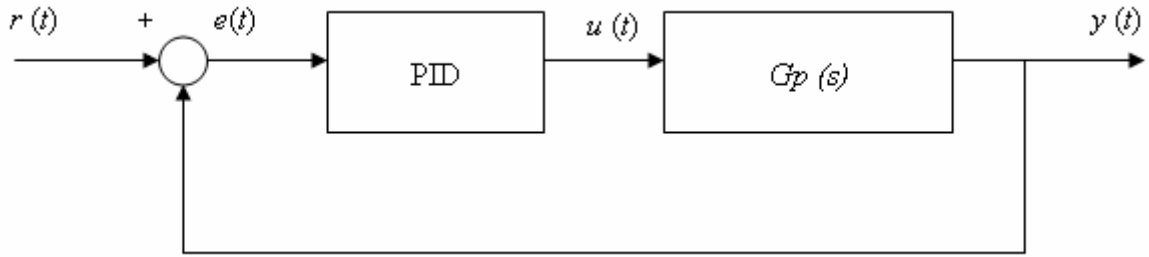
Beşinci bölümde program yazımında kullanılan PLC komutları tek tek anlatılmıştır. Altıncı bölümde PLC üzerinde PID denetleyicisi programı yazılarak örnek sistem olarak kullanılan çift cidarlı reaktör tank sıcaklık sistemi PLC' ye aktarılmıştır. Yedinci bölümde ise program çalıştırılarak farklı PID katsayılarında nasıl sonuçlar verdiği incelenmiş ve deneysel sonuçlar karşılaştırılmıştır.

2. PID KONTROL

Bu bölümde uygulamada en yaygın ve en çok bilinen denetleyici olan PID'nin tanımı, tasarımı, denetleyicilerin sistem cevabına etkileri ve uygun PID denetleyici katsayılarının belirlenmesinde kullanılan Ziegler - Nichols, Cohen – Coon ve Haalman yöntemleri ele alınacaktır.

2.1 PID Denetleyicinin Tanımı

PID (Proportional-Integral-Derivative) günümüzde en çok kullanılan kontrol yöntemidir. P oransal denetleyiciyi, I integral denetleyiciyi, D ise türevsel denetleyiciyi temsil etmektedir. Endüstrideki uygulamaların %95'inde kullanılmaktadır. Şekil 2.1'de PID denetleyicili bir sistem görülmektedir.



Şekil 2.1 PID denetleyicili sistem

Çok geniş bir uygulama alanı olmasına rağmen PID denetleyici için standart bir tanımlama yoktur. Ancak en yaygın PID algoritması aşağıdaki gibidir(Astrom,1995).

$$u(t) = K \left[e(t) + \frac{1}{T_i} \int_0^t e(t) d(t) + T_d \frac{de(t)}{dt} \right] \quad (2.1)$$

Hata işareti olan $e(t)$, referans işareti ile çıkış işareti arasındaki farka eşittir.

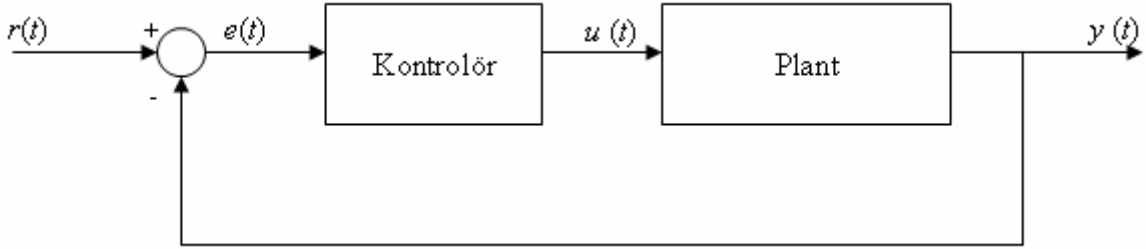
$$e(t) = r(t) - y(t) \quad (2.2)$$

Burada $e(t)$ hata işareti K oran sabiti, T_i integral veya reset zamanı (sn) ve T_d türev veya hız zamanı ile integral ve türev işlemine tabi tutularak $u(t)$ kontrol veya sürme işareti elde edilir. Yukarıdaki formülü biraz daha basite indirgenirse genel olarak $u(t)$ denetleyici çıkışı (2.3) olarak tanımlanabilir.

$$u(t) = P(t) + I(t) + D(t) \quad (2.3)$$

2.2 PID Denetleyici Tasarımı

Bu kısımda oransal (P), integral (I) ve türevsel (D) denetleyicilerinin her birinin karakteristiklerini ve istenilen tepkiyi bulmakta nasıl kullanıldıklarını incelenmiştir. İşlemlerde Şekil 2.2'de gösterilen geri beslemeli sistem kullanılmıştır.



Şekil 2.2 Geri beslemeli sistem

PID denetleyicinin transfer fonksiyonu aşağıdaki gibidir:

$$G_C(s) = \frac{U(s)}{E(s)} = K_p + \frac{K_i}{s} + K_d \cdot s$$

$$T_i = K_p / K_i$$

$$T_d = K_d / K_p$$
(2.4)

K_p = Oransal kazanç

K_i = İntegral kazancı

K_d = Türevsel kazanç

Hata işareti (e), istenilen giriş değeri (r) ile gerçek çıkış değeri (y) arasındaki farkı gösterir. Bu hata sinyali PID denetleyiciye gönderilir ve denetleyici bu hata sinyalinin hem türevini hem de integralini hesaplar. Denetleyici çıkışında elde edilen (u) sinyali, oransal kazanç K_p ile hata değerinin çarpım değeri, integral kazancı K_i ile hatanın integralinin çarpım değeri ve türevsel kazanç K_d ile hatanın türevi çarpım değerinin toplamına eşittir.

Bu (u) sinyali denetlenen sisteme gönderilir ve yeni çıkış (y) elde edilmiş olur. (y) çıkış sinyali algılayıcıya geri gönderilerek yeni hata sinyali (e) bulunur. Denetleyici yeni hata sinyaline aynı işlemleri uygular ve bu işlem böyle devam eder.

P, I ve D Denetleyicilerin Karakteristikleri: Oransal denetleyicinin, yükselme zamanını kısaltma etkisi vardır ama asla yok etmez (sürekli hal hatası). İntegral denetleyicinin ise

kararlı hal hatasını yok eder ancak bu geçici tepkinin daha kötü olmasına sebep olur. Türevsel denetleyici ise sistemin kararlılığının yükseltir, aşımı azaltır ve geçici cevabı iyileştirir. Kapalı çevrimli sistemlerdeki her bir denetleyicinin etkisi Çizelge 2.1 'de gösterilmiştir.

Çizelge 2.1 Denetleyicilerin etkileri

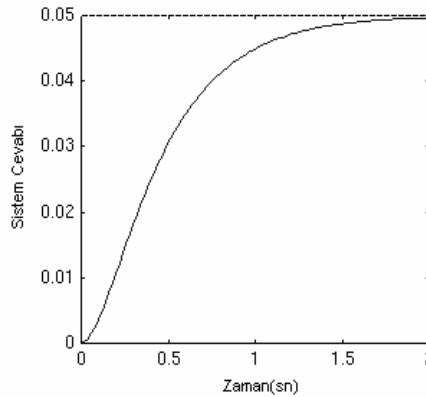
	YÜKSELME ZAMANI	AŞIM	YERLEŞME ZAMANI	KALICI HAL HATASI
K_p	Azalar	Artar	Az Değişir	Azalar
K_i	Azalar	Artar	Artar	Yok olur
K_d	Az Değişir	Azalar	Azalar	Az Değişir

Unutmamalıdır ki bu çizelge tam olarak geçerli değildir. Çünkü K_p , K_i ve K_d birbirlerine bağımlıdır. Yani değişkenlerden birinin değişimi diğer ikisinin etkisini değiştirebilir. Bu yüzden Çizelge 2.1 K_i , K_p ve K_d değerlerinin belirlenmesinde sadece bir referanstır.

2.3 Denetleyici Tipleri

P, PD, PI ve PID denetleyicilerinin sistem çıkışına etkilerini görebilmek için transfer fonksiyonu (2.5) olan sistem örnek olarak ele alınmıştır. Bu sistem örneğin motor + redüktör + yük sistemini temsil etsin. Sistemin doğrudan (açık çevrim) birim basamak fonksiyonuna olan cevabı Şekil 2.3' te dir.

$$G(s) = \frac{Y(s)}{U(s)} = \frac{1}{s^2 + 10s + 20} \quad (2.5)$$

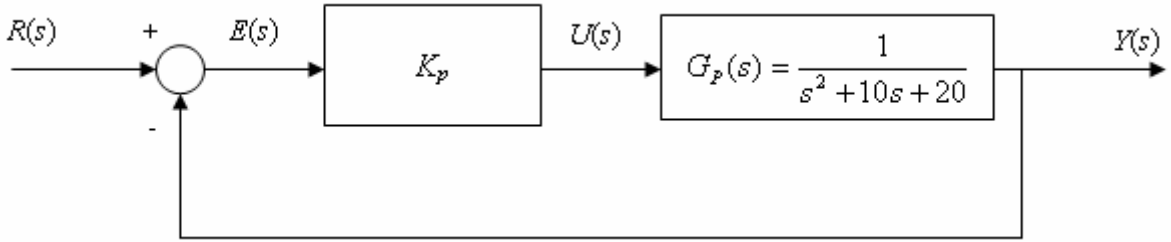


Şekil 2.3 Örnek sistemin birim basamak cevabı

Sistemin doğal frekansı $\omega_n = \sqrt{20}$, sönüm oranı $\zeta = 1.12$ olduğundan sistem aşırı sönümlüdür. Yükselme zamanı 0.88 sn, yerleşme zamanı 1.6 sn olan bir cevap vermektedir.

2.3.1 Oransal (P) Denetleyici

Çizelge 2.1'de oransal denetleyicinin (K_p) yükselme zamanını düşürdüğünü, aşmayı arttırdığını ve kararlı hal hatasını azalttığını görülmektedir. (2.5)'te verdiğimiz $G(s)$ sistemin oransal denetleyicisiyle birlikte kapalı çevrim blok diyagramı Şekil 2.4'de gösterilmiştir.



Şekil 2.4 Oransal denetleyicili sistem

Bu sistemin kapalı çevrim transfer fonksiyonu (2.6)'daki gibi olur.

$$\frac{C(s)}{R(s)} = \frac{K_p}{s^2 + 10s + (20 + K_p)} \quad (2.6)$$

Oransal kazanç (K_p) 100, 200, 300 gibi 3 farklı değer vererek sisteme etkilerini incelemek için Matlab programında aşağıdaki yazılım kullanılabilir.

```
Kp=100;
```

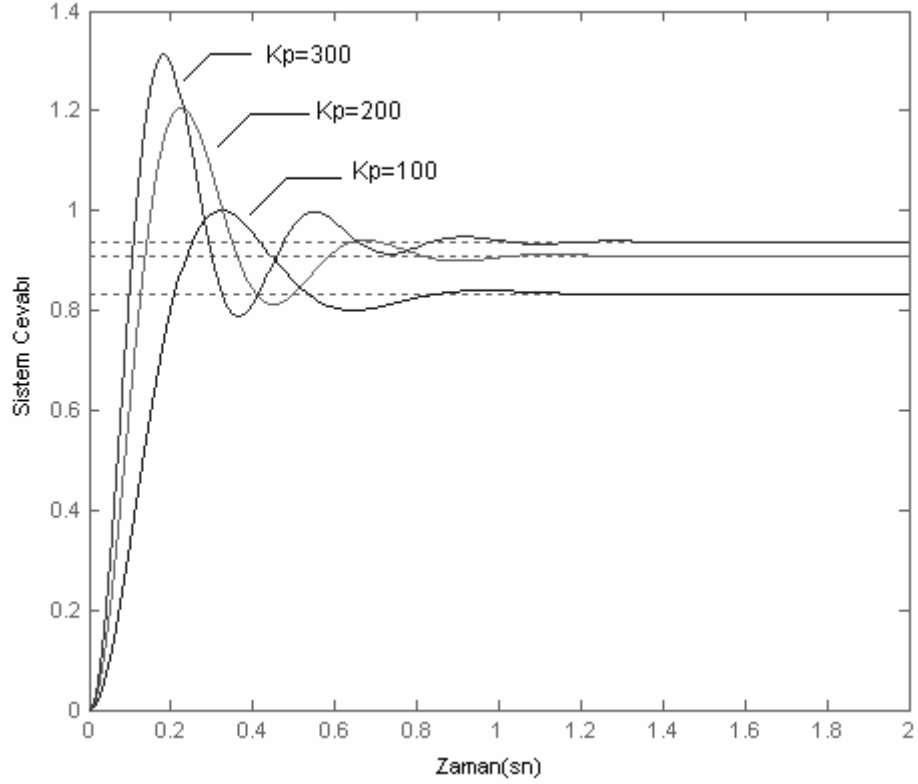
```
num=[Kp];
```

```
den=[1 10 20+Kp];
```

```
t=0:0.01:2;
```

```
step(num,den,t)
```

Çizilen grafiğin hold on komutuyla silinmesi önlenerek $K_p=200$ ve $K_p=300$ değerleri içinde aynı programı çalıştırarak grafikler aynı şekil üzerinde görülebilir (Şekil 2.5).



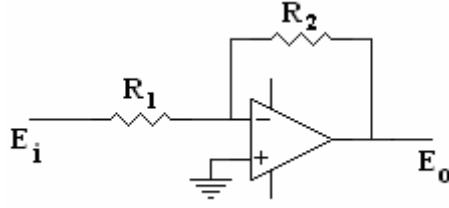
Şekil 2.5 Oransal kontrollü sistemin birim basamak cevabı

Şekil 2.5' te elde edilen grafikte görüldüğü gibi oransal denetleyici yükselme zamanını ve kararlı hal hatasını düşürmekte, aşımı arttırmakta ve yerleşme zamanını az bir miktarda düşürmektedir. Bu değerler Çizelge 2.2'de verilmiştir.

Çizelge 2.2 Oransal denetleyicinin sisteme etkileri

K_p	t_s (sn)	t_r (sn)	e_{ss}	Yüzde Üst Aşım
100	0.76	0.14	0.17	20
200	0.75	0.09	0.10	32.4
300	0.74	0.07	0.07	40

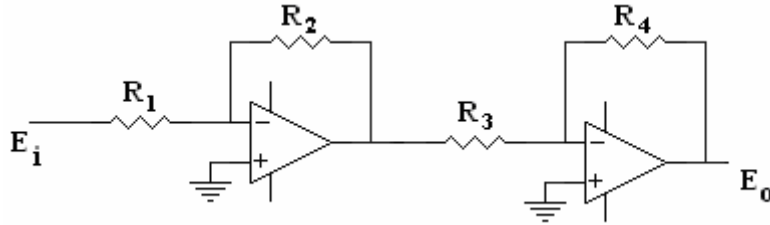
P Denetleyici Gerçeklenmesi : P denetleyiciyi gerçekleyecek olursak istenilen kazancı elde etmek için Şekil 2.6' daki basit eviren opampli devre kullanılabilir. Örneğin $K_p=300$ için $R_1 = 1 \text{ k}\Omega$ ve $R_2 = 300 \text{ k}\Omega$ seçilebilir.



Şekil 2.6 Tek opamp'lı P denetleyici devresi

$$\frac{E_o(s)}{E_i(s)} = -\frac{R_2}{R_1} \quad (2.7)$$

Evirme istenmiyorsa ya opamp girişi negatif beslenir ya da ikinci bir eviren opamp katı kullanılır. Bu kararlılığında arttırmaya yarar. R_3 ve R_4 dirençlerinin değerleri birbirine eşittir.



Şekil 2.7 İki işlemsel kuvvetlendiricili P denetleyici devresi

Bu devrenin transfer fonksiyonu (2.8)' dir.

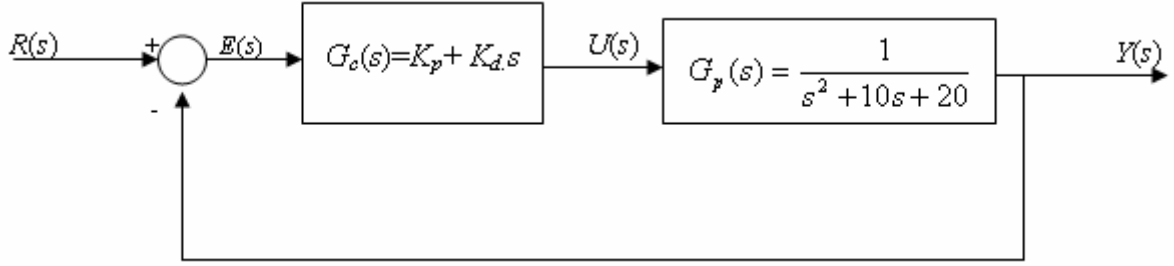
$$\frac{E_o(s)}{E_i(s)} = -\frac{R_2}{R_1} \left(-\frac{R_4}{R_3} \right) = \frac{R_2}{R_1}$$

2.3.2 Oransal - Türevsel (PD) Denetleyici

Çizelge 2.1'e göre türevsel denetleyici (K_d), yerleşme zamanını ve aşmayı azaltır. (2.5)' te transfer fonksiyonu verilen $G_p(s)$ sisteminin PD denetleyiciyle birlikte oluşturduğu blok diyagram Şekil 2.8'de gösterilmiştir. PD denetleyicinin transfer fonksiyonu ve türev zamanı T_d (2.9)'de ifade edilmiştir.

$$G(s) = K_p + K_d \cdot s = K_d \left(s + \frac{K_p}{K_d} \right) \quad (2.9)$$

$$T_d = \frac{K_p}{K_d}$$



Şekil 2.8 PD kontrollü sistem

Bu sistemin kapalı çevrim transfer fonksiyonu (2.10)'daki gibi olur.

$$\frac{Y(s)}{R(s)} = \frac{K_d s + K_p}{s^2 + (10 + K_d)s + (20 + K_p)} \quad (2.10)$$

K_p 300 olarak sabit tutulurak, K_d 'ye ise 10, 20 ve 30 şeklinde üç farklı değer verilerek Matlab ortamında yazılan program vasıtasıyla sistemin tepkileri ve türevsel denetleyicinin sisteme etkileri incelenebilir.

$K_p=300$;

$K_d=10$;

$num=[K_d \ K_p]$;

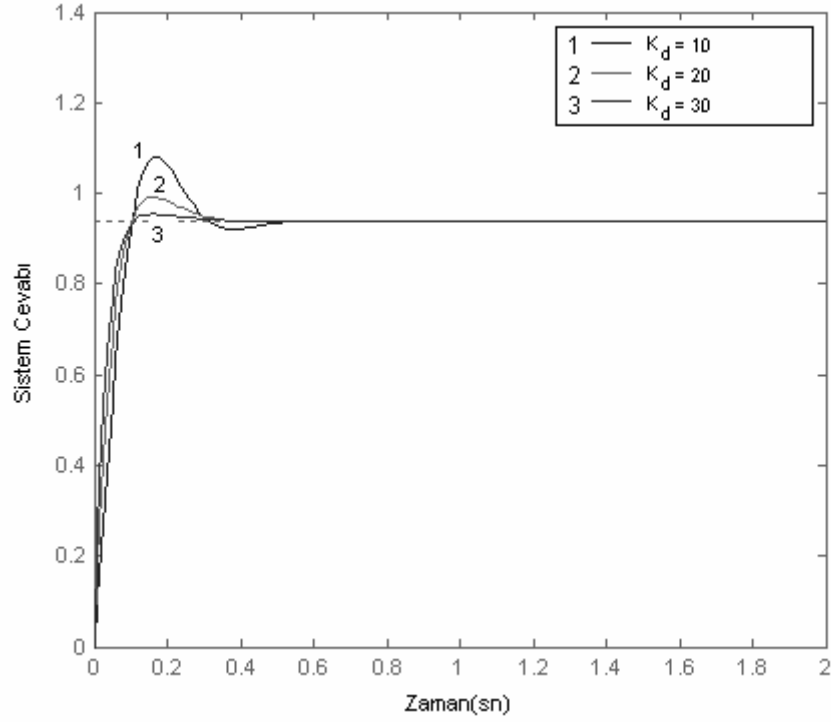
$den=[1 \ 10+K_d \ 20+K_p]$;

$t=0:0.01:2$;

$step(num,den,t)$

Çizilen grafiğin hold on komutuyla silinmesini önleyerek, $K_d=20$ ve $K_d=30$ değerleri içinde aynı programı çalıştırarak grafikler aynı şekil üzerinde görülebilir (Şekil 2.9).

Çizelge 2.3 'te ise çizdirelen bu grafiğin, yüzde üst aşım, kararlı hal hatası, yerleşme zamanı değerleri belirtilmiştir.



Şekil 2.9 PD kontrollü sistemin birim basamak cevabı

Çizelge 2.3 PD denetleyicinin sisteme etkileri

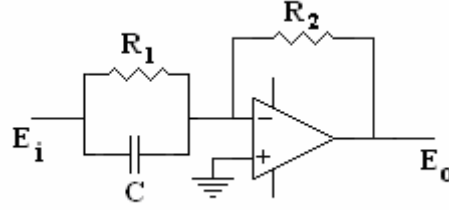
K_d	t_s (sn)	t_r (sn)	e_{ss}	Yüzde Üst Aşım
10	0.29	0.07	0.07	15.3
20	0.27	0.06	0.07	5.79
30	0.09	0.06	0.07	1.77

Şekil 2.9 'a göre oluşturulan Çizelge 2.3' te görüldüğü gibi PD denetleyicide bulunan K_d türev katsayısındaki artış sistem cevabındaki aşım ve yerleşme zamanını azaltmakta, yükselme zamanı ve kararlı hal hatasını ise çok az etkilemektedir.

PD denetleyicinin transfer fonksiyonu (2.11) olur.

$$G_c(s) = K_p + K_d s \quad (2.11)$$

PD Denetleyici Gerçeklenmesi : PD denetleyici tek opamplı devreyle gerçekleştirilecek olursa devre Şekil 2.10 'daki gibi olur.



Şekil 2.10 Tek opamplı PD denetleyici

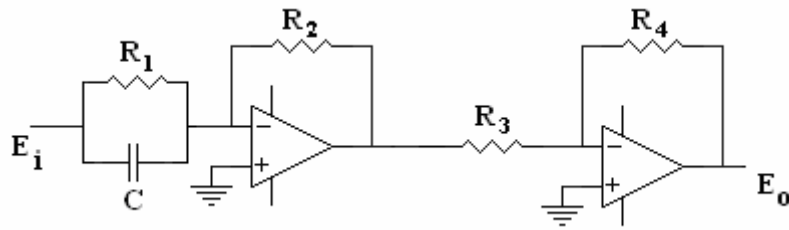
Bu devrenin transfer fonksiyonu (2.12) 'dir.

$$G_C(s) = \frac{E_o(s)}{E_i(s)} = -\left(\frac{R_2}{R_1} + R_2 C s\right) \quad (2.12)$$

$$K_p = R_2 / R_1 \quad (2.13)$$

$$K_d = R_2 \cdot C \quad (2.14)$$

Evirme etkisinden kurtulmak için Şekil 2.11'deki gibi ikinci bir kuvvetlendirme katı eklenebilir. Burada R_3 direnci R_4 direncine eşittir.



Şekil 2.11 Kuvvetlendiricili PD denetleyici

$$G_C(s) = \frac{E_o(s)}{E_i(s)} = -\left(\frac{R_2}{R_1} + R_2 C s\right) \left(-\frac{R_4}{R_3}\right) = \left(\frac{R_2}{R_1} + R_2 C s\right)$$

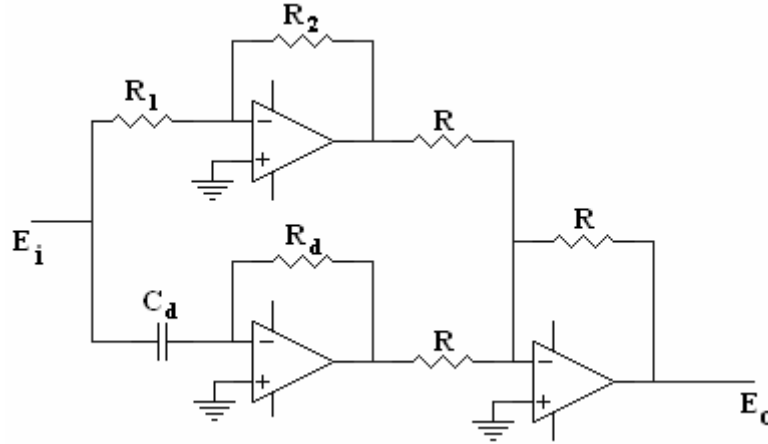
PD denetleyici üç işlemsel kuvvetlendiricili elektronik bir devre ile gerçekleştirilmek istenirse ise devrenin transfer fonksiyonu (2.15) 'de gösterildiği şekilde elde edilir.

$$G_c(s) = \frac{E_o(s)}{E_i(s)} = \frac{R_2}{R_1} + R_d C_d s \quad (2.15)$$

$$K_p = R_2 / R_1 \quad (2.16)$$

$$K_d = R_d \cdot C_d \quad (2.17)$$

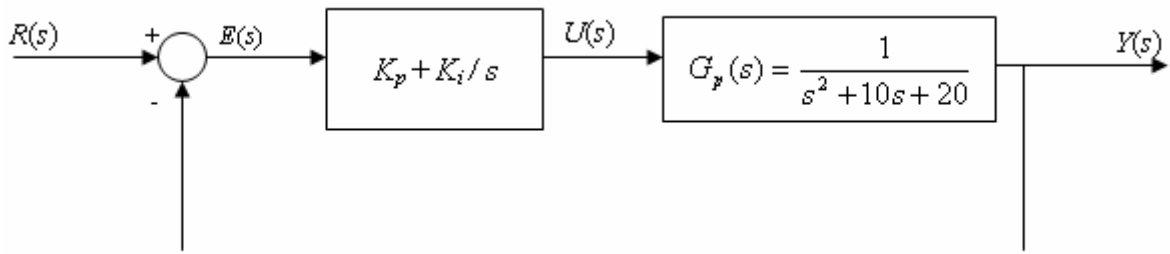
Devrede (2.16) ve (2.17)' de gösterilen K_p ve K_d katsayıları bağımsız olarak ayarlanabilir.



Şekil 2.12 Üç işlemsel kuvvetlendiricili PD denetleyicinin devre şeması

2.3.3 Oransal - İntegral (PI) Denetleyici

PID denetleyiciyi incelemeden önce son olarak PI denetleyici ele alınacak olunursa Çizelge 2.1'e göre integral denetleyici (K_i) yükselme zamanını azaltmakta, aşımı ve yerleşme zamanını arttırmakta, kararlı hal hatasını ise yok etmektedir. Transfer fonksiyonu (2.5) olan $G(s)$ sisteminin PI denetleyiciyle birlikte oluşturduğu blok diyagram Şekil 2.13'de gösterilmiştir.



Şekil 2.13 PI denetleyicili sistem

Bu sistemin kapalı çevrim transfer fonksiyonu (2.18)' dir.

$$\frac{Y(s)}{R(s)} = \frac{K_p s + K_i}{s^3 + 10s^2 + (20 + K_i)s + K_i} \quad (2.18)$$

K_p ' yi 30 olarak sabit kabul edilir, K_i ' ye ise 50, 80 ve 120 olarak üç farklı değer vererek MATLAB ortamında yazılan program ile elde edilen grafikte, integral katsayısının sisteme etkileri görülebilir (Şekil 2.14).

$K_p=30$;

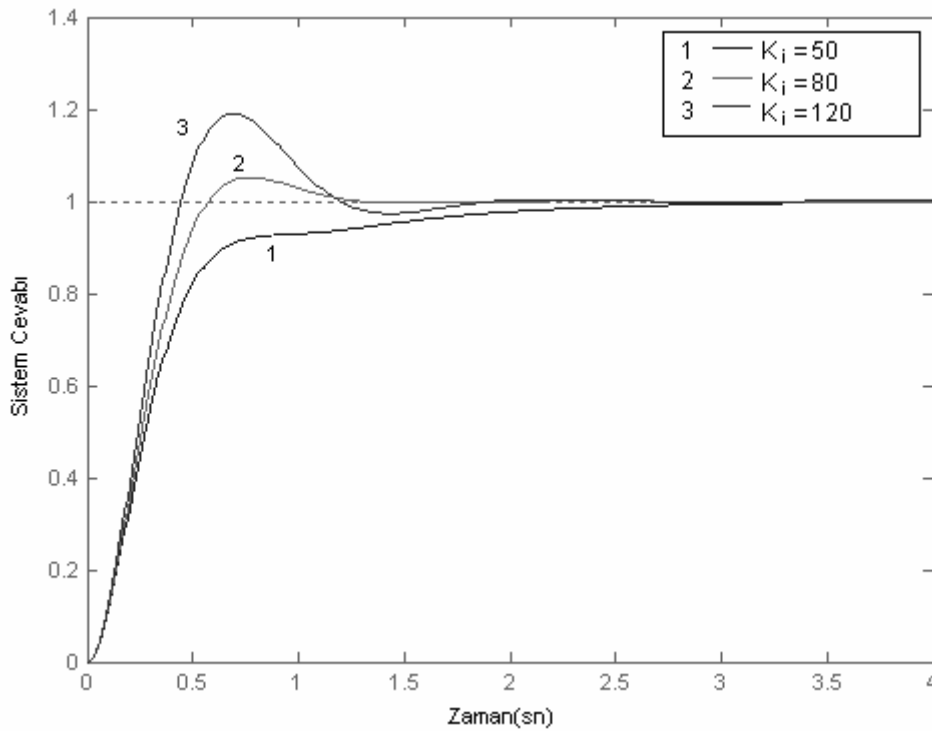
$K_i=120$;

num=[K_p K_i];

den=[1 10 20+ K_p K_i];

t=0:0.01:4;

step(num,den,t)



Şekil 2.14 PI denetleyicili sistemin birim basamak cevabı

Grafikte görüldüğü gibi hem oransal hem de integral denetleyici yükselme zamanını azalttığı için çift etki oluşur. İntegral denetleyici kararlı hal hatasını yok eder ancak aşımında arttırır. Bu değerler Çizelge 2.4 'te verilmiştir.

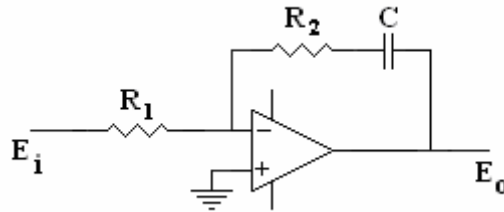
Çizelge 2.4 PI denetleyicinin sisteme etkileri

K_i	t_s (sn)	t_r (sn)	e_{ss}	Yüzde Üst Aşım
50	2.10	0.56	0	0
80	1.06	0.37	0	5.2
120	1.60	0.30	0	19

PI denetleyicinin transfer fonksiyonu ise (2.19) olur.

$$G_C(s) = K_p + \frac{K_i}{s} \quad (2.19)$$

PI Denetleyicinin Gerçeklenmesi : PI denetleyici basit tek opamplı devreyle gerçekleştirilecek olursa devre Şekil 2.15' teki gibi olur.



Şekil 2.15 Tek opamplı PI denetleyici

Bu devrenin transfer fonksiyonu (2.19)' da gösterilmiştir

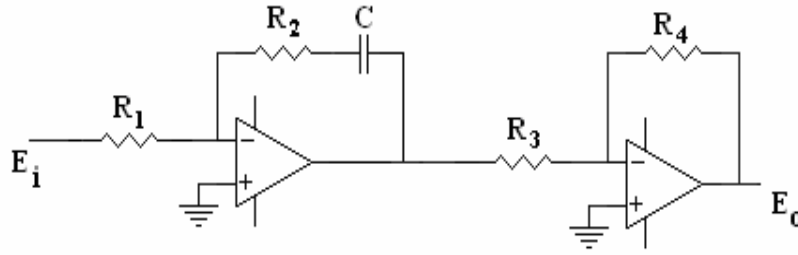
$$G_C(s) = \frac{E_o(s)}{E_i(s)} = - \left(\frac{R_2}{R_1} + \frac{R_2}{R_1 \cdot C \cdot s} \right) \quad (2.20)$$

$$K_p = R_2 / R_1 \quad (2.21)$$

$$K_d = R_2 / (R_1 \cdot C) \quad (2.22)$$

Evirme etkisinden kurtulmak için Şekil 2.16' daki gibi ikinci bir kuvvetlendirme katı eklenebilir. Burada R_3 direnci R_4 direncine eşittir. Transfer fonksiyonu ise (2.23) olur.

$$G_c(s) = \frac{E_o(s)}{E_i(s)} = - \left(\frac{R_2}{R_1} + \frac{R_2}{R_1 \cdot C \cdot s} \right) \left(- \frac{R_4}{R_3} \right) = \left(\frac{R_2}{R_1} + \frac{R_2}{R_1 \cdot C \cdot s} \right) \quad (2.23)$$



Şekil 2.16 Kuvvetlendiricili PI denetleyici

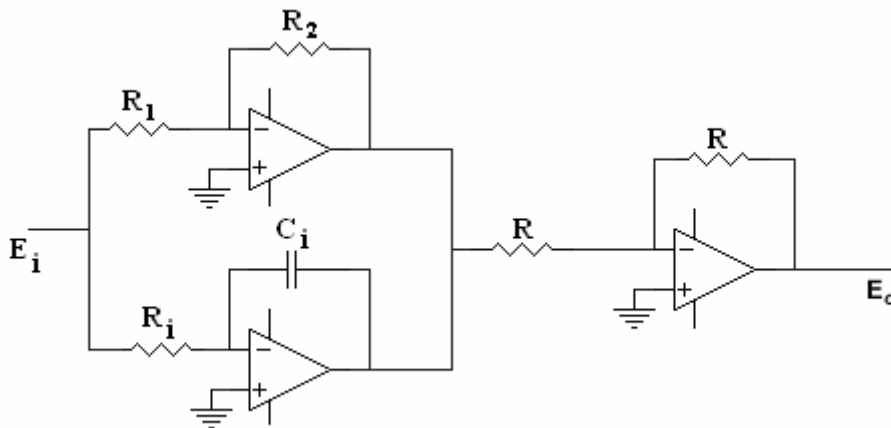
Üç işlemsel kuvvetlendiricili PI denetleyici devresinin transfer fonksiyonu ise (2.24)' tür

$$G_c(s) = \frac{E_i(s)}{E_o(s)} = \frac{R_2}{R_1} + \frac{R_2}{R_1 C_i s} \quad (2.24)$$

$$K_p = R_2 / R_1 \quad (2.25)$$

$$K_i = 1 / R_1 C_i \quad (2.26)$$

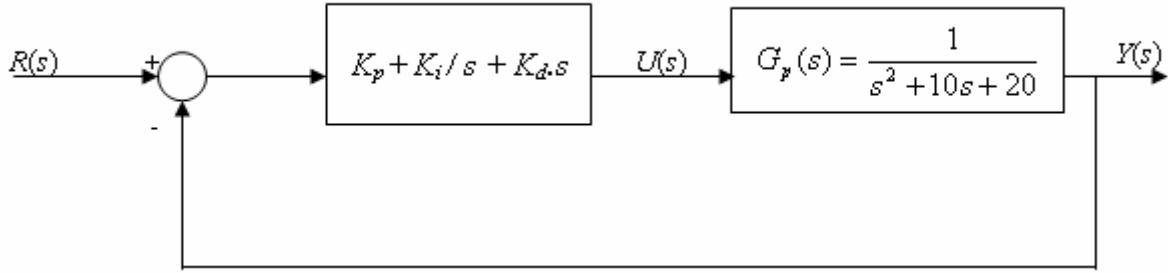
Devrede K_p ve K_i katsayıları bağımsız olarak ayarlanabilir. Ancak K_i katsayısı kapasite değeriyle ters orantılıdır. Etkin bir PI kontrol tasarımında küçük K_i değerine ihtiyaç duyulmaktadır. Buda gerçek dışı kapasite değerleriyle karşı karşıya gelinmesi anlamı taşımaktadır.



Şekil 2.17 Üç işlemsel kuvvetlendiricili PI denetleyicinin devre şeması

2.3.4 Oransal - İntegral - Türevsel (PID) Denetleyici

PI ve PD denetleyicilerin iyi yönlerinden birlikte yararlanmak için üç denetleyici birlikte uygun katsayılarla beraber kullanmak gerekir. PID denetleyicinin sistem cevabına ait karakteristik değerlerinin nasıl olduğunu incelemek için öncelikle (2.5)' te transfer fonksiyonu verilen $G_p(s)$ sistemiyle, PID denetleyicinin birlikte yer aldığı blok diyagramı oluşturulmuştur (Şekil 2.18).



Şekil 2.18 PID kontrollü sistem

Bu sistemin kapalı çevrim transfer fonksiyonu (2.27) olur.

$$\frac{Y(s)}{U(s)} = \frac{K_d s + K_p s + K_i}{s^3 + (10 + K_d)s^2 + (20 + K_p)s + K_i} \quad (2.27)$$

PID denetleyicinin katsayıları birkaç denemeyle $K_p=350$, $K_i=300$, $K_d=50$ olarak ele alınır ve MATLAB ortamında yazılan programla sistemin birim basamak fonksiyonuna verdiği cevap çizdirilir (Şekil 2.19). Bu katsayıları elde etmede Ziegler – Nichols ve Cohen – Coon gibi yöntemlerde kullanılabilir.

$K_p=350$;

$K_i=300$;

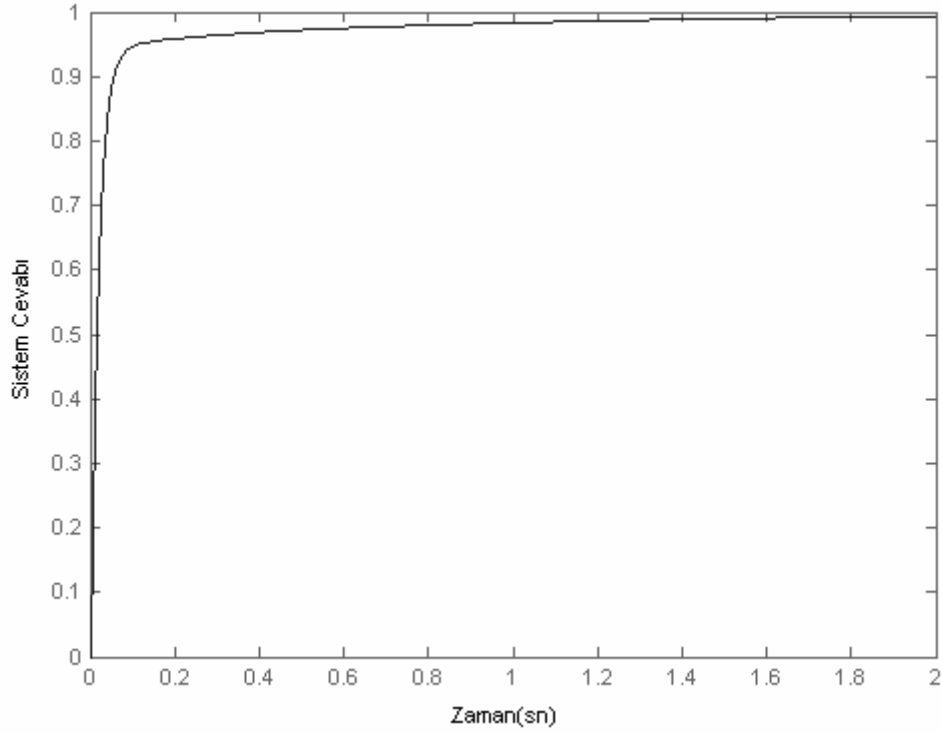
$K_d=50$;

num=[Kd Kp Ki];

den=[1 10+Kd 20+Kp Ki];

t=0:0.01:4;

step(num,den,t)



Şekil 2.19 PID kontrollü sistemin birim basamak cevabı

Bu durumda aşım yapmayan, kararlı hal hatası olmayan bununla birlikte hızlı yükselme zamanına sahip bir sistem elde edilir. Çizelge 2.5'te $K_p=350$, $K_i=300$, $K_d=50$ katsayılarıyla elde edilen sistem karakteristikleri görülmektedir.

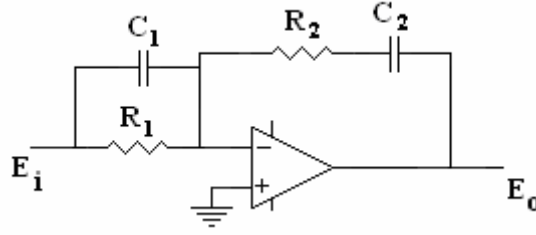
Çizelge 2.5 PID denetleyicinin sisteme etkileri

t_s (sn)	t_r (sn)	e_{ss}	Yüzde Üst Aşım
0.83	0.05	0	0

PID denetleyicinin transfer fonksiyonu (2.25)'de gösterilmiştir.

$$G_c(s) = K_p + \frac{K_i}{s} + K_d \quad (2.28)$$

PID Denetleyicinin Gerçeklenmesi : PID denetleyici basit tek opamplı devreyle gerçekleştirilecek olunursa devre Şekil 2.20'deki gibi olur.

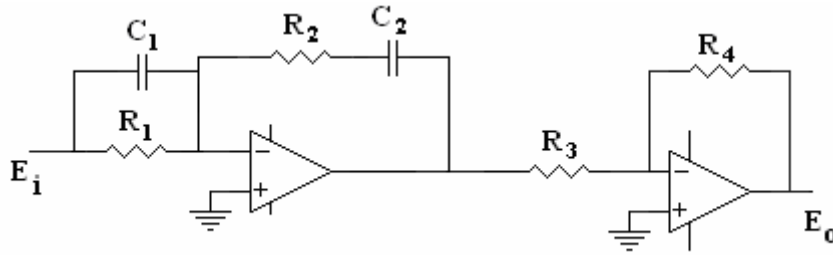


Şekil 2.20 Tek opamp'lı PID devre şeması

Bu devrenin transfer fonksiyonu ise (2.26)' dır.

$$G_c(s) = \frac{C_1(s + 1/R_1C_1)}{C_2(s + 1/R_2C_2)} \quad (2.29)$$

Evirme etkisini yok etmek ve kararlılığı arttırmak için ikinci bir opamp katı kullanılabilir. İki işlemsel kuvvetlendiricili PID kontrol devre şeması Şekil 2.21'deki gibi olacaktır. Devrede R_3 direnci R_4 direncine eşittir.

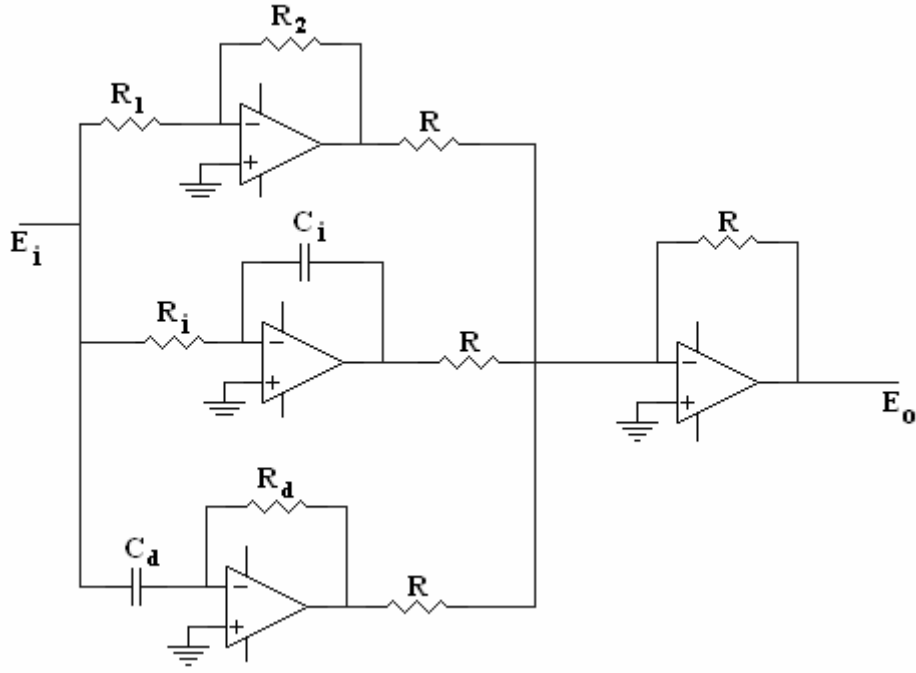


Şekil 2.21 İki işlemsel kuvvetlendiricili PID devre şeması

Şekil 2.22' de üç işlemsel kuvvetlendirici ile yapılan bir PID denetleyici görülmektedir. Devrenin transfer fonksiyonu (2.27)' dir.

$$G(s) = \frac{E_o(s)}{E_i(s)} = \frac{R_2}{R_1} + \frac{1}{R_i C_i s} + R_d C_d s \quad (2.27)$$

Tasarım istenen K_p , K_i , K_d değerlerini sağlayan, işlemsel kuvvetlendirici direnç ve kapasite değerlerinin seçimiyle tamamlanır.



Şekil 2.22 Üç işlemsel kuvvetlendiricili PID devre şeması

2.4 Genel PID Denetleyici Tasarımı İçin İzlenecek Yol

PID denetleyici tasarımında istenilen tepkiyi elde etmek için aşağıdaki adımlar izlenir:

- 1) Açık döngü tepkisi bulunur ve ihtiyaçlar belirlenir (yükselme zamanı, yerleşme zamanı, maksimum aşım vb.).
- 2) Yükselme zamanını kısaltmak için oransal denetleyici eklenir.
- 3) Kararlı hal hatasını yok etmek için integral denetleyici eklenir.
- 4) Aşımı azaltmak için türevsel denetleyici eklenir.
- 5) İstenilen tepki elde edilene kadar K_p , K_i ve K_d ayarlanır. Hangi denetleyicinin hangi karakteristiği kontrol ettiğini Çizelge 2.1'den yararlanılarak bulabiliriz.
- 6) Denetleyici tasarımında mümkün olduğu kadar basit tasarıma gidilmelidir. Eğer PI denetleyici ile istenilen tepki sağlanıyorsa, sisteme türevsel denetleyici eklenerek sistem karmaşılaştırılmamalıdır.

2.5 Ziegler - Nichols Metodu

Sistemin istenilen şekilde çalışabilmesi için PID parametrelerinin ayarlanması gerekmektedir.

Bunun için çeşitli yöntemler uygulanmaktadır. Bunlardan en yaygınlarından birisi de Ziegler - Nichols metodudur. İki farklı Ziegler - Nichols metodu vardır.

İlki osilasyon metodudur. Kontrol edilen sistem tam salınıma girebiliyorsa uygulanır. Tam salınıma sokma işlemi aşağıdaki gibi yapılır.

- 1) K_p ' si olabildiğince küçük olan bir P denetleyici ile başlanır,
- 2) Sistem osilasyona girinceye kadar K_p küçük adımlar ile artırılır,

Osilasyondaki sistemin çıkışını gözleyerek şu parametreleri not edilir;

- K_u : Osilasyondaki sistemin K_p katsayısı
- P_c : Osilasyondaki sistemin periyodu

K_u ve P_c parametreleri ve Çizelge 2.6 kullanılarak PID parametreleri hesaplanır.

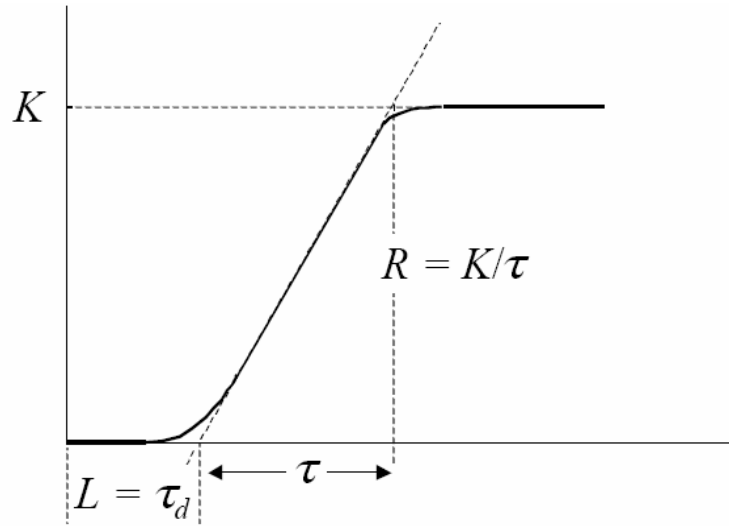
Çizelge 2.6 Ziegler - Nichols osilasyon metodu katsayıları

	K_p	T_i	T_d
P	$0.5.K_u$	-	-
PI	$0.45. K_u$	$0.833.P_c$	-
PID	$0.6. K_u$	$0.5.P_c$	$0.125.P_c$

Günümüzde bu yöntem pek kullanılmamaktadır. Sisteme osilasyon yaptırmak veya sistemi osilasyona götürebilmek için sisteme sürekli artan bir giriş vermek sisteme zarar verebilecek unsurlardır.

İkinci Ziegler - Nichols metodu reaksiyon eğrisi metodu olarak adlandırılır. Bu metotta ise PID katsayıları reaksiyon eğrisi yardımıyla bulunur. Sistemin açık çevrim fonksiyonun, birim basamak fonksiyonuna verdiği cevap incelenerek Şekil 2.23' teki grafik üzerinde gösterilen değerler bulunur. Bu cevap (2.28)' deki gibi birinci dereceden zaman gecikmeli bir fonksiyonun, birim basamak fonksiyonuna verdiği cevaba denktir.

$$G(s) = \frac{K.e^{-\tau_d s}}{\tau.s + 1} \quad (2.28)$$



Şekil 2.23 Ziegler – Nichols reaksiyon eğrisi

Bulunan bu değerler Çizelge 2.7’deki formüller kullanılarak ideal PID katsayıları elde edilir. Burada τ zaman sabiti (sn), τ_d ise zaman gecikmesidir (sn).

Çizelge 2.7 Ziegler – Nichols reaksiyon eğrisine göre PID parametreleri

	K_p	T_i	T_d
P	$\frac{\tau}{K \cdot \tau_d}$	∞	0
PI	$0.9 \frac{\tau}{K \cdot \tau_d}$	$3 \cdot \tau_d$	0
PID	$1.2 \frac{\tau}{K \cdot \tau_d}$	$2 \cdot \tau_d$	$0.5 \tau_d$

2.6 Cohen - Coon Metodu

Bu metod Ziegler – Nichols reaksiyon eğrisine çok benzemektedir. Şekil 2.23’teki grafik elde edildikten sonra PID parametrelerinin hesaplamalarında farklılık göstermektedir. Bu hesaplamalarda Çizelge 2.8 kullanılır.

Çizelge 2.8 Cohen – Coon metodu katsayıları

	K_p	T_i	T_d
P	$\frac{\tau}{K \cdot \tau_d} \left(1 + \frac{\tau_d}{3\tau} \right)$	∞	0
PI	$\frac{\tau}{K \cdot \tau_d} \left(0.9 + \frac{\tau_d}{3\tau} \right)$	$\frac{\tau_d [30\tau + 3\tau_d]}{[9\tau + 20\tau_d]}$	0
PID	$\frac{\tau}{K \cdot \tau_d} \left(\frac{4}{3} + \frac{\tau_d}{4\tau} \right)$	$\frac{\tau_d [32\tau + 6\tau_d]}{[13\tau + 8\tau_d]}$	$\frac{4\tau_d \cdot \tau}{11\tau + 2\tau_d}$

2.7 Haalman Metodu

Bu metotta da Şekil 2.23'teki grafikte elde ettiğimiz parametreler kullanılmaktadır. Elde edilen parametreler kullanılarak sisteme uygun bir PI denetleyici belirlenmektedir. Bu katsayılar Çizelge 2.9'da verilmiştir.

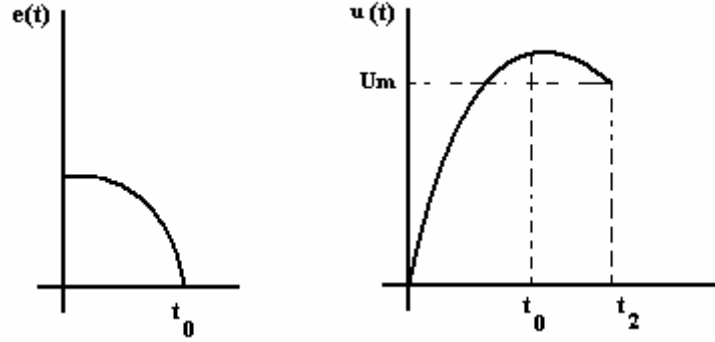
Çizelge 2.9 Haalman metodu katsayıları

	K_p	T_i
PI	$\frac{2\tau}{3 \cdot K \cdot \tau_d}$	τ

2.8 İntegral Yığılması (Anti - Windup)

İntegral denetleyici hata işaretinin integralini alır. Ancak bir çok sistem ve elemanda doyum bulunmaktadır. İdealde t_0 anında hata pozitiften negatife geçtiğinde aynı etkinin yine t_0 anında $u(t)$ ' de de belirmesi gerekir. Fakat integrasyon nedeniyle $u(t)$, t_0 anında u_m değerinin üstündedir ve u_m değerine ulaşması t_2 anında olur. Bu nedenle, hatanın $u(t)$ 'deki etkisi ancak $t_2 - t_0$ kadar bir gecikmeden sonra t_2 anında belirir. Sonuç olarak integrasyondan dolayı $u(t)$ işareti doyma seviyesinin üzerine çıkar ve $u(t)$ ' ye etki

etmesi içinde bu seviyenin altına inmesi gereklidir. Bu olaya integrasyon yığılması adı verilir. $e(t)$ hatası işaret deęiřtirmedięi sũrece $u(t)$ artmaya devam ettięinden $c(t)$ kontrol edilen bũyũklũęũn cevap eęrisinde bũyũk ařımlar oluřur.



řekil 2.24 Anti - windup

Bunu ۆnlemenin yolu $u(t)$ doyma seviyesine ulařtıęı an integratۆrũ devreden ıkarmaktır. Bu tezde kullanılacak olan ve Bۆlũm 3' te detaylı olarak inceleyeceęimiz artımsal PID denetleyici formu integral yığılmasını kendisi ۆnlemektedir (Bequette,2003).

3. DİJİTAL PID DENETLEYİCİ

Bu bölümde analog PID denetleyiciden dijital PID denetleyiciye geçiş, dijital PID denetleyicinin tasarımı, dijital PID denetleyicinin katsayıları olan K_P , K_I , K_D ile analog PID denetleyici katsayıları olan K_p , T_i , T_d arasındaki ilişki ve PLC programında kullanacağımız artımsal PID denetleyici ele alınacaktır.

3.1 Dijital PID Denetleyicinin Oluşturulması

Dijital kontrol sistem tasarımı sürekli kontrol sistem tasarımı ile prensipte benzerdir. Kontrol edilen sistem genelde aynıdır sadece denetleyici örneklenmiş prosese ya da dijital bilgiye göre tasarlanır. Günümüzde sadece yazılım ile oluşturulan dijital denetleyicilerin kullanımı yaygın hale gelmiştir. Dijital denetleyici dinamiği z düzleminde ifade edilen transfer fonksiyonu ya da fark denklemleri ile tanımlanabilir.

Dijital PID denetleyicinin z tabanındaki ifadesini oluşturalım. Analog PID ifadesi (3.1) 'dir.

$$u(t) = K \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right] \quad (3.1)$$

$u(t)$ denetleyici çıkışı, $e(t)$ ise yine hata sinyalidir. Buradan yamuk integrasyon yöntemiyle integral ve türev ifadeleri ayrık karşılıklarıyla ifade edilirse:

$$u(kT) = K_c \left[e_k(kT) + \frac{T}{T_I} \sum_{h=1}^k \frac{e((h-1)T) + e(hT)}{2} + \frac{T_D}{T} [e(kT) - e((k-1)T)] \right] \quad (3.2)$$

Bu ifadedeki terimleri z düzlemine taşıyacak olursak dijital PID denetleyici ifademiz (3.4) 'ü bulmuş oluruz.

$$U(z) = K \left[1 - \frac{T}{2T_i} + \frac{T}{T_i} \frac{1}{1-z^{-1}} + \frac{T_d}{T} (1-z^{-1}) \right] E(z) \quad (3.3)$$

$$\frac{U(z)}{E(z)} = \frac{(K_P + K_I + K_D)z^2 - (K_P + 2K_D)z + K_D}{z(z-1)} \quad (3.4)$$

K_P , K_I , K_D dijital PID katsayıları 3.5, 3.6 ve 3.7 denklemlerinde gösterildiği gibidir (Ogata,1995).

$$K_p = K - \frac{K.T}{2T_i} \quad (3.5)$$

$$K_I = \frac{K.T}{T_i} \quad (3.6)$$

$$K_D = \frac{K.T_D}{T} \quad (3.7)$$

Anlaşılaçağı üzere K , T_i ve T_d analog PID denetleyici katsayıları, K_p , K_I , K_D dijital (sayısal) PID katsayılarıdır. T ise saniye cinsinden örnekleme periyodudur.

3.2 Artımsal PID

Dijital PID kontrolde temel olarak iki tür algoritma vardır. Bunlardan ilki mutlak, diğeri ise artımsal PID kontrolüdür. Mutlak PID, konum PID'si olarak da bilinir. Çıkışın, referans girişine göre kontrolü söz konusudur. Artımsal PID kontrolde (hız PID kontrolü) ise denetleyici çıkışı ve bu çıkışın bir önceki periyottaki durumu arasındaki farka göre hesaplama yapılır.

Tezimizde kullanacağımız PID formu bu olacaktır. Bu form ayırık zamanlı olarak bulduğumuz PID formunun fark denklem haline getirilmesiyle bulunmaktadır.

$$\frac{U(z)}{E(z)} = \frac{(K_p + K_I + K_D) - (K_p + 2K_D)z^{-1} + K_Dz^{-2}}{1 - z^{-1}} \quad (3.8)$$

Denklem (3.8)'i fark denkleme haline getirecek olursak (3.9) ifadesini elde ederiz.

$$u(k) - u(k-1) = (K_p + K_I + K_D)e(k) - (K_I + 2K_D)e(k-1) + K_De(k-2) \quad (3.9)$$

$$u(k) = u(k-1) + (K_p + K_I + K_D)e(k) - (K_I + 2K_D)e(k-1) + K_De(k-2)$$

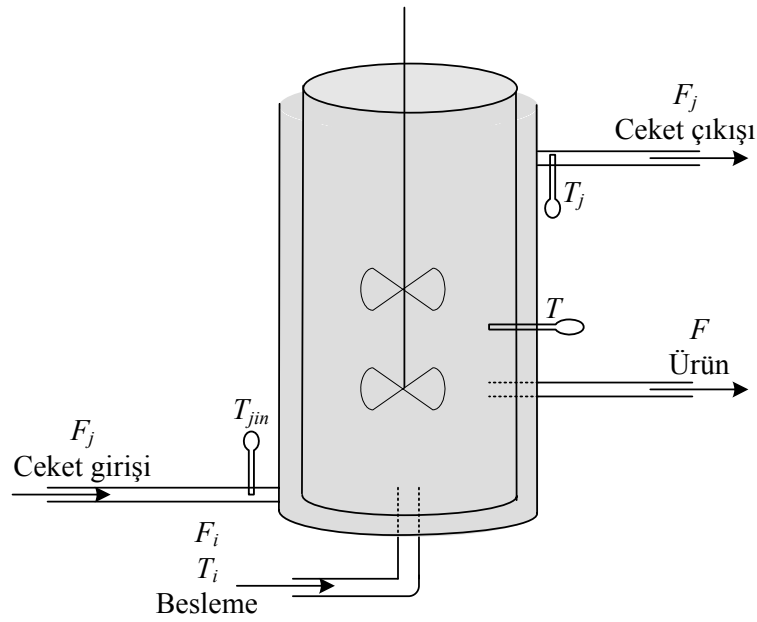
Bu ifadede bulunan parametreleri daha sonra PLC için yazdığımız programa girerek PID denetleyicimizi oluşturacağız.

4. ÇİFT CİDARLI (CEKETLİ) REAKTÖR TANK ISITMA SİSTEMİNİN MATEMATİKSEL MODELİ

Bu bölümde bir çift cidarlı (ceketli) reaktör tank ısıtma sisteminin matematiksel modeli doğrusallaştırılarak oluşturulacak, elde edilen bu matematiksel modele uygun PID denetleyici Ziegler - Nichols ve Cohen - Coon yöntemleriyle tasarlanacaktır.

4.1 Giriş

Sistemin çalışma prensibi bir ısıtıcı tankta ısıtılan sıcak suyun, ceketin içinden geçirilerek, ceketle dolaşan su ile reaktör içerisindeki sıvı arasında ısı alışverişini meydana gelmesiyle reaktörün ısıtılması şeklindedir. Ceketten çıkan su tekrar ısıtıcı tanka gitmekte, böylece sistem devir daim yapmaya devam etmektedir. Şekil 4.1 de bir karıştırılmalı çift cidarlı reaktör tank ısıtma sistemi görülmektedir (Bequette, 2003).



Şekil 4.1 Çift cidarlı (ceketli) karıştırılmalı tank ısıtıcı

Şekil 4.1' de de görüldüğü gibi F_j cekete giren sıcak suyun debisi, T_{jin} ise bu sıvının sıcaklığıdır. F_i reaktöre giren ürünün debisi, F ürünün çıkış debisi, T_i ürünün girişteki sıcaklığı, T ise ürünün reaktörden çıktığı andaki sıcaklığıdır. Sistemde kontrol edeceğimiz transfer fonksiyonunda giriş T_{jin} parametresi, çıkış ise T parametresidir.

Sistemde kullanılacak olan parametreler, açıklamaları Çizelge 4.1' de gösterilmiştir.

Çizelge 4.1 Sistem parametreleri

Parametre	Açıklama	Birim
A	Isı transferi alanı	m^2
c	Sıcaklık kapasitesi	$kcal/ g.^{\circ}C$
F	Debi	m^3 / dk
ρ	Yoğunluk	g / m^3
T	Sıcaklık	$^{\circ}C$
Q	Isı transfer oranı	$kcal / dk$
U	Isı transfer katsayısı	$kcal / ^{\circ}C.dk.m^2$
V	Hacim	m^3

4.2 Matematiksel Model

Sistemin matematiksel modelinde tank çevresindeki malzeme denge kanuna göre yoğunluk sabit kabul edilerek (4.1) ifadesi bulunur.

$$\frac{dV\rho}{dt} = F_i\rho - F\rho \quad (4.1)$$

Burada hacim (V) sabit olduğundan $F_i = F$ eşitliğine ulaşılır. Cidarlar arasındaki malzeme denge kanuna göre yine yoğunluk ve hacim sabit kabul edilerek (4.2) ifadeleri bulunur.

$$\begin{aligned} \frac{dV_j\rho_j}{dt} &= F_{ji}\rho_j - F_j\rho_j \\ F_{ji} &= F_j \end{aligned} \quad (4.2)$$

Bu da cekte giriş ve çıkış debilerinin eşit olduğunu belirtir. Tankın, enerji dengesi kanuna göre bir matematiksel modelini çıkarttığımızda ise (4.3) 'ü elde ederiz.

$$V \cdot \rho \cdot c_p \frac{dT}{dt} = F_i \cdot \rho \cdot c_p (T_i - T) + Q \quad (4.3)$$

dT/dt ifadesini yalnız bıraktığımızda (4.4) eşitliği elde edilir.

$$\frac{dT}{dt} = \frac{F_i}{V} \cdot (T_i - T) + \frac{Q}{V \cdot \rho \cdot c_p} \quad (4.4)$$

Benzer şekilde enerji dengesi kanunu ceketin matematiksel modeline uygulandığında (4.5) denkleminde ulaşılır.

$$\frac{dT_j}{dt} = \frac{F_j}{V_j} \cdot (T_{ji} - T_j) + \frac{Q}{V_j \cdot \rho_j \cdot c_{p_j}} \quad (4.5)$$

Ceketle tank arasındaki ısı transfer denklemi (4.6)'de ifade edilmiştir.

$$Q = UA(T_j - T) \quad (4.6)$$

(4.6) ifadesini (4.4) ve (4.5) denklemlerinde yerine yazdığımız takdirde ise bu sistemin aşağıdaki iki durum değişkenli matematiksel modelini elde etmiş oluruz (Bequette, 2003).

$$\frac{dT}{dt} = \frac{F_i}{V} (T_i - T) + \frac{UA(T_j - T)}{V \cdot \rho \cdot c_p} \quad (4.7)$$

$$\frac{dT_j}{dt} = \frac{F_j}{V_j} (T_{ji} - T_j) - \frac{UA(T_j - T)}{V_j \cdot \rho_j \cdot c_{p_j}} \quad (4.8)$$

4.3 Doğrusallaştırma

Kimyasal proseslerin malzeme ve enerji dengesi modelleri genellikle doğrusal olmayan denklemlerdir. Kontrol sistemlerinin tasarımı ve analizi için bu denklemlerin doğrusallaştırılması gereklidir. Burada kullanacağımız doğrusallaştırılma yönteminde doğrusal olmayan modeli Taylor serisine açarak veya denge noktası kalıcı hal noktası civarında çakışılacağı kabulüyle doğrusal olmayan sisteme yakın bir doğrusal denklem elde etmeye çalışacağız.

4.3.1 Bir Durum Değişkeni İçin Doğrusallaştırma

Tek değişkenli bir fonksiyon ele alalım. Fonksiyonun değerine Taylor serisine açılarak (4.9) denklemindeki gibi ulaşılabilir(Bequette,2003).

$$\frac{dx}{dt} = f(x) \quad (4.9)$$

$$f(x) \approx f(x_s) + \left. \frac{\partial f(x)}{\partial x} \right|_{x_s} (x - x_s) + \frac{1}{2} \left. \frac{\partial^2 f(x)}{\partial x^2} \right|_{x_s} (x - x_s)^2$$

Burada x_s çalışma noktasıdır. İki ve daha yüksek dereceli terimleri ihmal edersek denklem (4.10) elde edilir.

$$\frac{dx}{dt} = f(x) \approx f(x_s) + \left. \frac{\partial f(x)}{\partial x} \right|_{x_s} (x - x_s) \quad (4.10)$$

Bu durumda fonksiyonun doğrusallaştırılmış son hali (4.11) olacaktır.

$$\frac{d(x - x_s)}{dt} = \left. \frac{\partial f(x)}{\partial x} \right|_{x_s} (x - x_s) \quad (4.11)$$

4.3.2 Tek Durum Tek Değişkenli Sistem İçin Doğrusallaştırma

Tek durum tek değişkenli (4.12) fonksiyonunu ele alalım.

$$\frac{dx}{dt} = f(x, u) \quad (4.12)$$

Taylor serisi kullanılarak fonksiyon (4.13)' e eşit kabul edilebilir.

$$f(x, u) \approx f(x_s, u_s) + \left. \frac{\partial f(x, u)}{\partial x} \right|_{x_s, u_s} (x - x_s) + \left. \frac{\partial f(x, u)}{\partial u} \right|_{x_s, u_s} (u - u_s) \quad (4.13)$$

x_s değeri sabit olduğu için, fonksiyonun doğrusallaştırılmış son hali (4.14) olacaktır.

$$\frac{d(x - x_s)}{dt} = \left. \frac{\partial f(x, u)}{\partial x} \right|_{x_s, u_s} (x - x_s) + \left. \frac{\partial f(x, u)}{\partial u} \right|_{x_s, u_s} (u - u_s) \quad (4.14)$$

4.3.3 Çıkış Değişkeni İçin Doğrusallaştırma

Şimdide bir çıkış değişkenini doğrusallaştıralım. Bunun için (4.15) ifadesini ele alacağız.

$$y = g(x, u) \quad (4.15)$$

Taylor serisine açarak yüksek dereceli ifadeleri ihmal ettiğimiz zaman (4.16) denklemini buluruz.

$$g(x, u) \approx g(x_s, u_s) + \left. \frac{\partial g(x, u)}{\partial x} \right|_{x_s, u_s} (x - x_s) + \left. \frac{\partial g(x, u)}{\partial u} \right|_{x_s, u_s} (u - u_s) \quad (4.16)$$

x_s ifadesini yerine yazıldığında, doğrusallaştırılmış çıkış değişkeni fonksiyonu (4.17) olacaktır.

$$y_s - y = \left. \frac{\partial g(x, u)}{\partial x} \right|_{x_s, u_s} (x - x_s) + \left. \frac{\partial g(x, u)}{\partial u} \right|_{x_s, u_s} (u - u_s) \quad (4.17)$$

4.3.4 Genel Doğrusallaştırma

Özetle x durumlarını , y çıkışlarını , u girişlerini ve p parametrelerini içeren bir doğrusal olmayan sistem ele aldığımızda doğrusallaştırılmış matrisel ifadelerini (4.19)' daki gibi yazabiliriz.

$$\dot{x}_1 = f_1(x_1, \dots, x_{nx}, u_1, \dots, u_{nu}, p_1, \dots, p_{np}) \quad (4.18)$$

$$y = g_1(x_1, \dots, x_{nx}, u_1, \dots, u_{nu}, p_1, \dots, p_{np})$$

$$\begin{aligned} A_{ij} &= \left. \frac{\partial f_i}{\partial x_j} \right|_{x_s, u_s} & B_{ij} &= \left. \frac{\partial f_i}{\partial u_j} \right|_{x_s, u_s} \\ C_{ij} &= \left. \frac{\partial g_i}{\partial x_j} \right|_{x_s, u_s} & D_{ij} &= \left. \frac{\partial g_i}{\partial u_j} \right|_{x_s, u_s} \end{aligned} \quad (4.19)$$

Örneğin B matrisinin B_{ij} nci elemanı oluşturulurken j ' nci giriş ve i ' nci durum kullanılır. Bütün terimler oluşturularak doğrusal sistem matrisel olarak ifade etmiş olunur.

4.4 Çift Cidarlı Isıtma Sisteminin Doğrusallaştırılmış Durum Denklemleri

Şekil 4.1' deki çift cidarlı tank reaktör ısıtma sisteminde tekrar inceleyelim. Sistemde bulunan tank ve çift cidarlı ısıtıcının (4.7) ve (4.8) gösterilen matematiksel modellerini giriş ve çıkış parametrelerine göre düzenleyelim.

$$f_1(T, T_j, F_j, F, T_i, T_{ji}) = \frac{dT}{dt} = \frac{F_i}{V} (T_i - T) + \frac{UA(T_j - T)}{V \cdot \rho \cdot c_p} \quad (4.20)$$

$$f_2(T, T_j, F_j, F, T_i, T_{ji}) = \frac{dT_j}{dt} = \frac{F_j}{V_j} (T_{ji} - T_j) - \frac{UA(T_j - T)}{V_j \cdot \rho_j \cdot c_{p j}} \quad (4.21)$$

Burada çıkışlar tank ve ceketin sıcaklıklarıdır. Bunlar aynı zamanda durum değişkenleridir. Girişler ise ceket içindeki sıvının debisi, ürünün debisi, ürünün sıcaklığı ve ceketin iç sıcaklığıdır. Bu ifadeler matrisel olarak (4.22) ve (4.23)' de gösterildiği gibidir.

$$y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} T - T_s \\ T_j - T_{js} \end{bmatrix} \quad (4.23)$$

$$u = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} F_j - F_{js} \\ F - F_s \\ T_i - T_{is} \\ T_{ji} - T_{jis} \end{bmatrix} \quad (4.24)$$

Bu matrisel ifadeyi doğrusallaştırdığımızda A matrisinin terimlerini aşağıdaki gibi buluruz.

$$\begin{aligned} A_{11} &= \frac{\partial f_1}{\partial x_1} = \frac{\partial f_1}{\partial (T - T_s)} = \frac{\partial f_1}{\partial T} = -\frac{F_s}{V} - \frac{UA}{V \cdot \rho \cdot c_p} \\ A_{12} &= \frac{\partial f_1}{\partial x_2} = \frac{\partial f_1}{\partial (T_j - T_{js})} = \frac{\partial f_1}{\partial T_j} = \frac{UA}{V \cdot \rho \cdot c_p} \\ A_{21} &= \frac{\partial f_2}{\partial x_1} = \frac{\partial f_2}{\partial (T - T_s)} = \frac{\partial f_2}{\partial T} = \frac{UA}{V_j \cdot \rho_j \cdot c_{p j}} \\ A_{22} &= \frac{\partial f_2}{\partial x_2} = \frac{\partial f_2}{\partial (T_j - T_{js})} = \frac{\partial f_2}{\partial T} = -\frac{F_{js}}{V_j} - \frac{UA}{V_j \cdot \rho_j \cdot c_{p j}} \end{aligned} \quad (4.25)$$

$$A = \begin{bmatrix} -\frac{F_s}{V} - \frac{UA}{V \cdot \rho \cdot c_p} & \frac{UA}{V \cdot \rho \cdot c_p} \\ \frac{UA}{V_j \cdot \rho_j \cdot c_{p j}} & -\frac{F_{js}}{V_j} - \frac{UA}{V_j \cdot \rho_j \cdot c_{p j}} \end{bmatrix} \quad (4.26)$$

B, C, D matrislerinin de A matrisine benzer şekilde bulalım.

$$B = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} & \frac{\partial f_1}{\partial u_3} & \frac{\partial f_1}{\partial u_4} \\ \frac{\partial f_2}{\partial u_1} & \frac{\partial f_2}{\partial u_2} & \frac{\partial f_2}{\partial u_3} & \frac{\partial f_2}{\partial u_4} \end{bmatrix} = \begin{bmatrix} 0 & \frac{T_i - T_{is}}{V} & \frac{F_s}{V} & 0 \\ \frac{T_{jis} - T_{is}}{V_j} & 0 & 0 & \frac{F_{js}}{V_j} \end{bmatrix} \quad (4.27)$$

$$C = \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \frac{\partial g_1}{\partial x_2} \\ \frac{\partial g_2}{\partial x_1} & \frac{\partial g_2}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (4.28)$$

$$D = \begin{bmatrix} \frac{\partial g_1}{\partial u_1} & \frac{\partial g_1}{\partial u_2} & \frac{\partial g_1}{\partial u_3} & \frac{\partial g_1}{\partial u_4} \\ \frac{\partial g_2}{\partial u_1} & \frac{\partial g_2}{\partial u_2} & \frac{\partial g_2}{\partial u_3} & \frac{\partial g_2}{\partial u_4} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (4.29)$$

Bizim ele alacağımız sistemdeki parametre değerleri aşağıda verilmiştir.

$$F_s = 0.5 \text{ m}^3 / \text{dk} \quad T_s = 95 \text{ }^\circ\text{C} \quad T_{is} = 50 \text{ }^\circ\text{C}$$

$$T_{js} = 100 \text{ }^\circ\text{C} \quad T_{jis} = 110 \text{ }^\circ\text{C} \quad V = 2.366 \text{ m}^3$$

$$V_j = 0.353 \text{ m}^3 \quad \rho_j c_{pj} = 61.3 \text{ kcal / }^\circ\text{C.m}^3 \quad \rho c_p = 61.3 \text{ kcal / }^\circ\text{C.m}^3$$

Yukarıda verilen değerler kullanılarak kalıcı haldeki iki dinamik denklemin çözülmesiyle UA ve F_{js} değerleri bulunarak denklem tekrar düzenlenir. İlk denklemin çözülmesiyle UA = 275.85 kcal / °C.dk, ikinci denklemin çözümüyle ise F_{js} = 2.25 m³/dk bulunur.

Burada, sıfır başlangıç şartları kabulüyle durum ve çıkış denklemlerinin Laplace dönüşümleri alınarak sistemin istenen çıkış ve giriş değişkenleri arasında transfer fonksiyonları kurulacaktır.

Bizim kullanacağımız transfer fonksiyonunda çıkış tank sıcaklığı (ürün sıcaklığı), giriş ise ceket giriş sıcaklığıdır.

$$\frac{Y(s)}{U(s)} = G(s) = C(sI - A)^{-1} B = \frac{T - T_s}{T_{ji} - T_{jis}} \quad (4.30)$$

Kontrol edeceğimiz sistemin Laplace formundaki matematiksel modelini A, B, C, D matrislerini oluşturarak Matlab programındaki ss2tf komutu yardımıyla oluşturacağız.

$$F_s=0.5;$$

$$F_{js}=2.25;$$

$$T_s=95;$$

Tis=50;

Tjs=100;

Tjis=110;

V=2.366;

Vj=0.353;

rhoc=61.3;

rhocj=61.3;

UA=275.85;

A=[-(Fs/V)-((UA)/(V*rhoc)) (UA)/(V*rhoc);(UA)/(Vj*rhocj) -(Fjs/Vj)-((UA)/(Vj*rhocj))]

B=[0 (Tis-Ts)/V Fs/V 0;(Tjis-Tjs)/Vj 0 0 Fjs/Vj]

C=[1 0; 0 1]

D=[0 0 0 0;0 0 0 0]

G=ss(A,B,C,D);

[num1,den]=ss2tf(A,B,C,D,1)

[num2,den]=ss2tf(A,B,C,D,2)

[num3,den]=ss2tf(A,B,C,D,3)

[num4,den]=ss2tf(A,B,C,D,4)

Bu programdan alınacak sonuçlar şu şekilde olacaktır.

>>A =

-2.1133 1.9019

12.7479 -19.1218

B =

0 -19.0194 0.2113 0

28.3286 0 0 6.3739

C =

1 0

0 1

D =

0 0 0 0

0 0 0 0

num1 =

0 0.0000 53.8794

0 28.3286 59.8660

num2 =

0 -19.0194 -363.6862

0 0 -242.4575

num3 =

0 0.2113 4.0410

0 0 2.6940

num4 =

0 0 12.1229

0 6.3739 13.4699

den =

1.0000 21.2351 16.1638

Burada elde edilen sonuçlarla 8 adet transfer fonksiyonu elde edilmiş oluruz(4.31).

$$\begin{aligned}
G_{11}(s) &= \frac{53.8794}{s^2 + 21.2351s + 16.1638}, \quad T \rightarrow F_j \\
G_{21}(s) &= \frac{28.3286s + 59.8660}{s^2 + 21.2351s + 16.1638}, \quad T_j \rightarrow F_j \\
G_{12}(s) &= \frac{-19.0194s - 363.6862}{s^2 + 21.2351s + 16.1638}, \quad T \rightarrow F \\
G_{22}(s) &= \frac{-242.4575}{s^2 + 21.2351s + 16.1638}, \quad T_j \rightarrow F \\
G_{13}(s) &= \frac{0.2113s + 4.0410}{s^2 + 21.2351s + 16.1638}, \quad T \rightarrow T_i \\
G_{23}(s) &= \frac{2.6940}{s^2 + 21.2351s + 16.1638}, \quad T_j \rightarrow T_i \\
G_{14}(s) &= \frac{12.1229}{s^2 + 21.2351s + 16.1638}, \quad T \rightarrow T_{jin} \\
G_{24}(s) &= \frac{6.3739s + 13.4699}{s^2 + 21.2351s + 16.1638}, \quad T_j \rightarrow T_{jin}
\end{aligned} \tag{4.31}$$

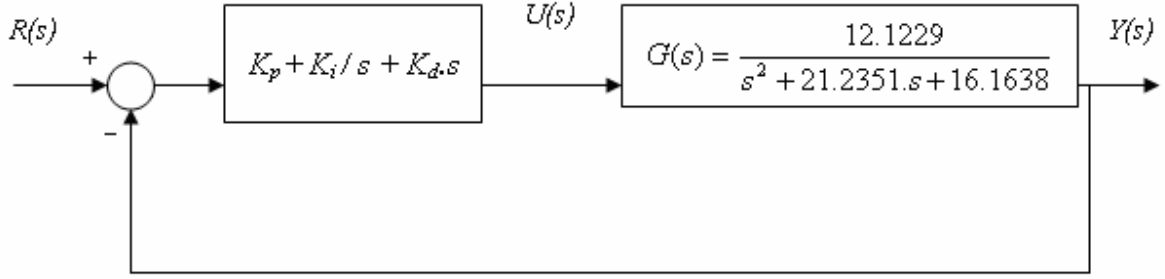
(4.30) denkleminde görüleceği gibi bizim kullanacağımız transfer fonksiyonu 1. çıkış olan y_1 ile 4. giriş olan u_4 arasındaki G_{14} denklemidir. Özetle kullanacağımız ifade (4.32)' dir.

$$G(s) = \frac{12.1229}{s^2 + 21.2351s + 16.1638} \tag{4.32}$$

Ancak bu ifade sadece seçilen çalışma noktası için oluşturulmuştur. Yapılan doğrusallaştırma işlemi bu çalışma noktası etrafındaki yerel bir çözümdür. Kullanılan parametre değerlerinde ve çalışma noktasındaki değişiklikler transfer fonksiyonunu doğrudan etkileyecektir. Transfer fonksiyonundaki değişiklik, kullanılan PID katsayılarının da değişmesi anlamına gelmektedir. Bu çalışmada çift cidarlı reaktör tank sıcaklığının doğrusallaştırılmış transfer fonksiyonu sadece örnek olarak kullanılmıştır.

4.5 Uygun PID Katsayılarının Seçimi

Kontrol edilecek sistemin s düzlemindeki ifadesi bulunduktan sonra sıra bu sisteme en uygun P, I, D katsayılarının seçimindedir. Şekil 4.2 'de ısıtma sisteminin blok diagramı görülmektedir.



Şekil 4.2 Tank ısıtma sisteminin PID denetleyicili blok diagramı

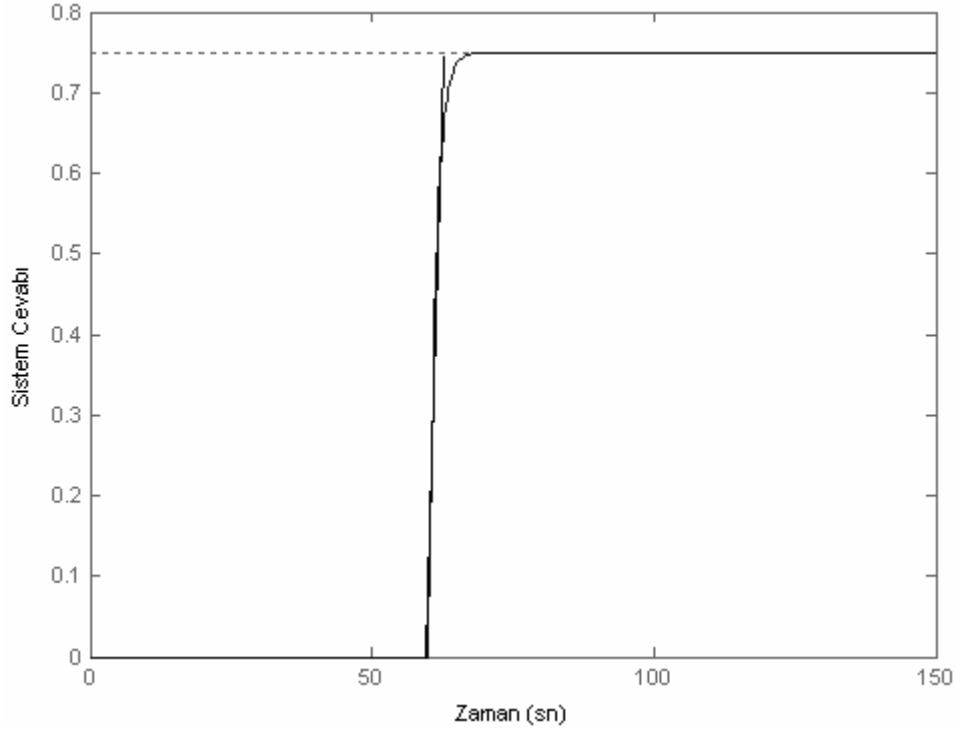
Böyle bir sisteme göre uygun P, I, D katsayılarının nasıl seçildiğini Bölüm 2 'de incelenmişti. Ziegler - Nichols ve Cohen - Coon yöntemleriyle çift cidarlı tank reaktör ısıtma sisteminin uygun P, I, D katsayıları bulunur. Ancak burada dikkat edilmesi gereken bir nokta vardır. Bulunacak bu katsayılar s domeni için geçerli olacaktır. Normal çalışmada ise sistem PLC ile yani dijital olarak kontrol edilecektir. Bu nedenle bulunan katsayılar dijital denetleyiciye uygun biçimde (3.4), (3.5) ve (3.6) denklemleriyle dönüştürülmelidir. Ayrıca sisteme bir dakikalık gecikme zamanı koyulacaktır.

4.5.1 Ziegler – Nichols Yöntemiyle Katsayıların Belirlenmesi

Bu yöntem Bölüm 2'de incelenmiş ve iki farklı Ziegler – Nichols yöntemi olduğunu görülmüştü. Bunlardan ilki olan osilasyon metodu kullanılamayacaktır. Bunun nedeni transfer fonksiyonumuzun sönüm oranı $\zeta = 2.5 > 1$ olduğundan sistemin aşırı sönümlü oluşu ve osilasyon yapmamasıdır.

Bu nedenle ikinci yöntem olan reaksiyon eğrisi metodu kullanılmalıdır. Bunun için öncelikle bir dakika kadar geciktirilmiş olan kontrol edilecek olan sistemin birim basamak fonksiyonuna cevabı Matlab programında yazdığımız algoritma yardımıyla bulunur.

```
s=tf('s');
Gps=tf([12.1229],[1.0000 21.2351 16.1638], 'inputdelay',60);
figure(1)
step(Gps)
```



Şekil 4.3 Tank ısıtma sisteminin açık çevrim birim basamak cevabı

Şekil 4.3 'teki grafikten Bölüm 2' de incelediğimiz Ziegler - Nichols reaksiyon eğrisi metodu yardımıyla P, I, D katsayılarını bulmak için gerekli olan 3 parametre bulunur.

$$\tau_d = 60$$

$$\tau = 2$$

$$K = 0.75$$

Bu parametreler Çizelge 2.7 kullanılarak sürekli zamana uygun olan şu değerler bulunur.

$$K_p = \frac{1.2 \cdot \tau}{K \cdot \tau_d} = 0.053$$

$$T_i = 2 \cdot \tau_d = 120 \quad (4.33)$$

$$T_d = 0.5 \tau_d = 30$$

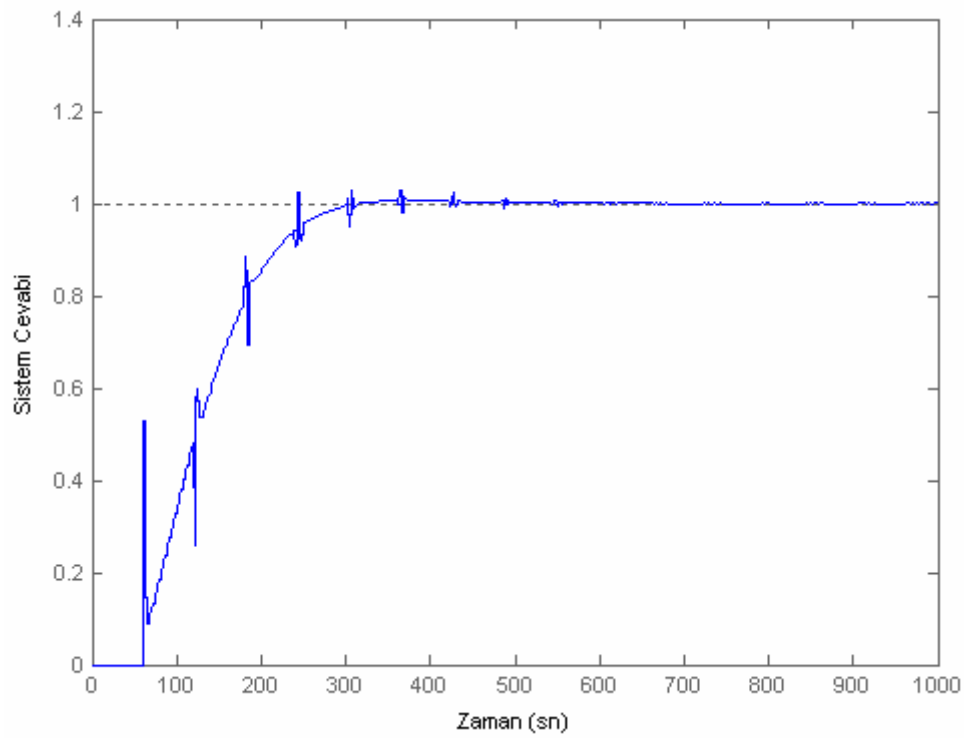
Bu değerlerin ayrık zamanlı sisteme uygun hale getirilmesi gerekir. Örnekleme zamanımız(T) 1 saniyedir. Buna göre (4.34) 'teki değerler elde edilir.

$$K_p = K - \frac{K.T}{2.T_i} = 0.053 - \frac{(0.053).1}{2.(120)} = 0.052$$

$$K_I = \frac{K.T}{T_i} = \frac{(0.053).1}{60} = 0.01 \quad (4.34)$$

$$K_D = \frac{K.T_D}{T} = \frac{(0.053).(30)}{1} = 1.59$$

MATLAB ortamında bir program yazarak kapalı çevrim kontrol sisteminin cevabı birim basamak fonksiyonuna vereceği yanıt Şekil 4.4' teki gibi olacaktır.



Şekil 4.4 Ziegler - Nichols tablosuyla belirlenen PID katsayılarının sistem cevabı

Çizelge 4.2 Ziegler – Nichols ile belirlenen katsayılara göre sistem karakteristikleri

t_s (dk)	t_r (dk)	e_{ss}	Yüzde Üst Aşım
8	2.6	0	%3

4.5.2 Cohen - Coon Yöntemiyle Katsayıların Belirlenmesi

Cohen - Coon metodunun Ziegler - Nichols yönteminden farkı sadece P, I, D katsayılarını bulurken kullanılan tablodur. Şekil 4.3'ten aşağıdaki parametre değerleri elde edilmiştir.

$$\tau_d = 60$$

$$\tau = 2$$

$$K = 0.75$$

Bu parametrelere göre Cohen - Coon tablosu kullanılarak sürekli zaman için geçerli olan (4.35)' teki değerler bulunur.

$$K_p = \frac{\tau}{K \cdot \tau_d} \left(\frac{4}{3} + \frac{\tau_d}{4 \cdot \tau} \right) = 0.392$$

$$T_I = \frac{\tau_d [32 \cdot \tau + 6 \tau_d]}{[13 \cdot \tau + 8 \cdot \tau_d]} = 50.2 \quad (4.35)$$

$$T_D = \frac{4 \cdot \tau_d \cdot \tau}{[11 \cdot \tau + 2 \cdot \tau_d]} = 3.38$$

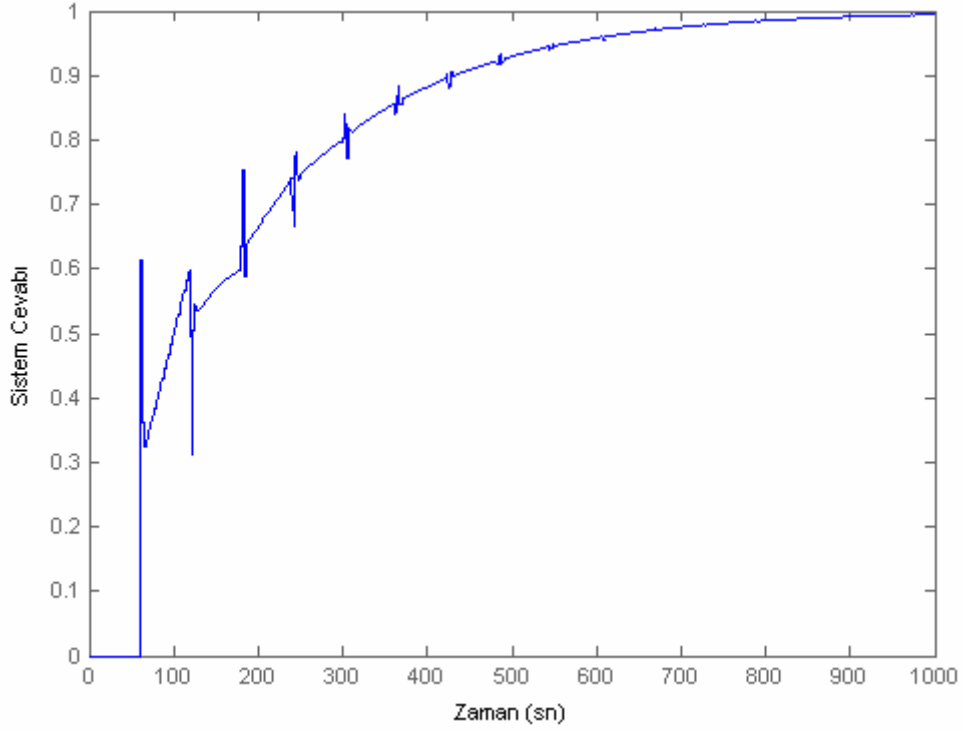
Yine bu değerlerin ayrık zamanlı sisteme uygun hale getirilmesi gerekir. Buna göre, denklem (4.36)' daki değerler elde edilir.

$$K_p = K - \frac{K \cdot T}{2 \cdot T_I} = 0.392 - \frac{(0.392) \cdot 1}{2 \cdot (50.2)} = 0.388$$

$$K_I = \frac{K \cdot T}{T_I} = \frac{(0.392) \cdot 1}{50.2} = 0.007 \quad (4.36)$$

$$K_D = \frac{K \cdot T_D}{T} = \frac{(0.392) \cdot (3.38)}{1} = 1.32$$

Bu değerler kullanıldığında kapalı çevrim kontrol sisteminin MATLAB' de yazılan program yardımıyla çizdirilen birim basamak cevabı Şekil 4.5 'te gösterilmiştir.



Şekil 4.5 Cohen - Coon tablosuyla belirlenen PID katsayılarının sisteme etkisi

Çizelge 4.3 Cohen - Coon ile belirlenen katsayılara göre sistem karakteristikleri

t_s (dk)	t_r (dk)	e_{ss}	Yüzde Üst Aşım
12.4	6	0	%0

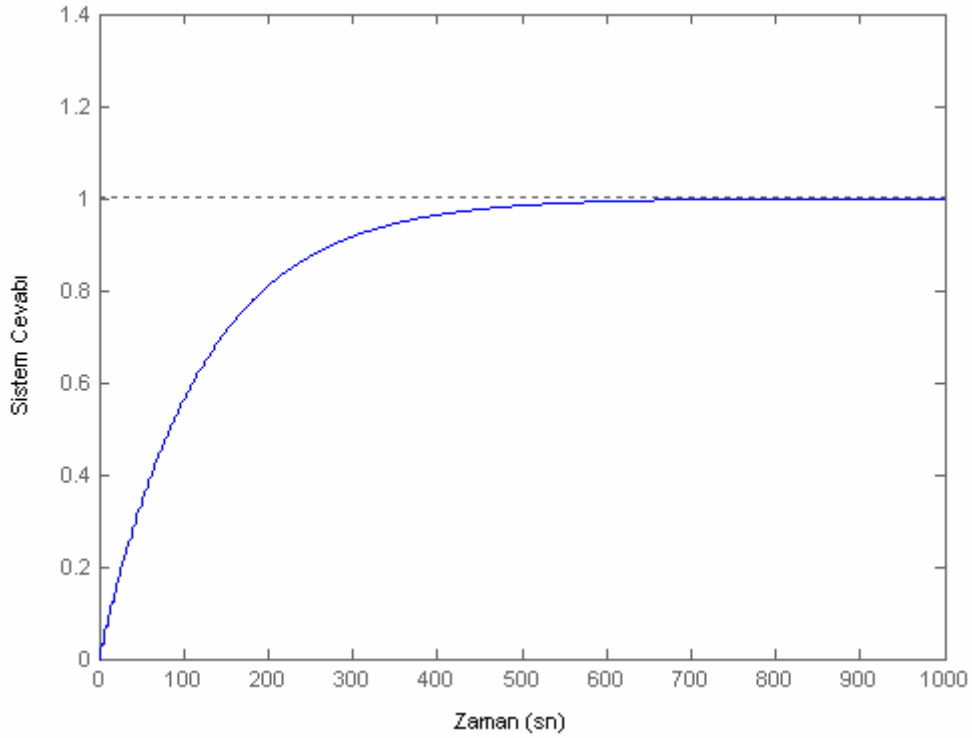
Çizelge 4.2 ve Çizelge 4.3 'te görüldüğü gibi bu iki yöntemle elde edilen PID katsayılarında aşım ve kalıcı hal hatası bakımından güzel sonuçlar elde edilmiş ancak K_D türev katsayısının yüksek oluşu nedeniyle sistem cevabında bazı aksaklıklar (ani sıçramalar) göze çarpmıştır. Bu sorunu gidermek için PID katsayılarında (özellikle K_D 'de) bazı değişiklikler yaparak daha iyi bir cevaba ulaşılabilir. Türev alma işlemi, sinyaldeki istenmeyen yüksek frekanslı bileşenleri (yani gürültüyü) yüksek genlik olarak esas işarete yansıtacağından K_D 'nin azaltılması yoluna gidilmiştir. Örneğin,

$$K_P = 0.017;$$

$$K_I = 0.011;$$

$$K_D = 0.006;$$

değerlerine göre MATLAB programında grafiği çizdirelim(Şekil 4.6).



Şekil 4.6 Uygun PID katsayılarıyla sistemin birim basamak fonksiyonuna cevabı

Çizelge 4.4 Uygun PID katsayılarına göre sistem karakteristikleri

t_s (dk)	t_r (dk)	e_{ss}	Yüzde Üst Aşım
7.8	4.4	0	%0

Görüldüğü gibi üst aşımsız ve düzgün bir çıkış yanıtı elde edilmiştir. Buradaki oturma zamanı değeri bir sıcaklık prosesi için oldukça mantıklıdır.

Bölüm 6' da PLC ortamında hazırladığımız PID denetleyicisine sistemin matematiksel modelini aktararak aynı PID katsayılarıyla sistem çıkışının aynı cevabı verip vermediğini karşılaştıracacağız.

5. PLC 'NİN YAPISI VE PROGRAMLANMASI

PLC “Programlanabilir Lojik Kontrolör” ingilizce kelimelerinin baş harflerinin alınarak kısaltılması ile oluşur. Genel olarak PLC, endüstri alanında kullanılmak üzere tasarlanmış, dijital prensiplere göre yazılan fonksiyonu gerçekleyen, bir sistemi yada sistem gruplarını, giriş çıkış kartları ile denetleyen, içinde barındırdığı zamanlama, sayma, saklama ve aritmetik işlem fonksiyonları ile genel kontrol sağlayan elektronik bir cihazdır.

Otomatik kontrol sistemlerinde, hız, kontrol, güvenlik ve ürün kalitesinin yanı sıra, yeni bir ürün imali için kumanda devrelerinin yeniden oluşturulması montajı ve bağlantıları yerine sadece PLC programlama ile giderilmesi çok büyük bir avantaj sağlamıştır. Buda PLC tabanlı kontrol sistemlerinin endüstriyel otomasyon, devrelerinden vazgeçilmez bir sistem olarak kullanılmasını getirmiştir.

Bir PLC ile kontrol sistemlerinin oluşturulması:

- a) Kontrol probleminin ifade edilmesiyle sorunun kağıda dökülmesi ve senaryonun yazılması,
- b) Sorunun çözümü için gerekli program veya fonksiyonların belirlenmesi,
- d) Programın diagrama aktarılması (LADDER STL, SCL, FBD)
- e) Programın yazılması

olarak sıralanabilir. En yaygın programlama dili olarak merdiven (LADDER) kullanılır. Fakat kompleks uygulamalarda ve yoğun matematiksel ve sisteme ilişkin blok yazılımı gerektiren programlarda STL daha ön plana çıkmaktadır.

5.1 PLC' nin Yapısı

PLC nin iç yapısı genel olarak 3 ana bölüme ayrılmıştır. Bunlar yazılımın saklandığı bellek, bu yazılım içindeki komutları çalıştırıp giriş çıkış birimleri ile haberleşen CPU ve PLC 'nin çevresel cihazlar ile iletişimini sağlayan giriş çıkış birimidir.

5.1.1 CPU (Central Processing Unit)

Merkezi işlem birimleri PLC sisteminin beyni olarak düşünülebilir. Mikroişlemciden oluşan bir entegre devredir. Bellek içine depolanmış komutları yürüterek, hafıza, giriş çıkış birimleri ile haberleşir, yönlendirir, aritmetik hesaplamaları yapar.

5.1.2 Bellek (Memory)

Bellek, denetleyicideki kontrol planı veya programını saklamak için kullanılır. Memoryde saklanan bilgi, hangi girişe göre hangi çıkış işaretinin saklanacağı ile ilgilidir ve gerekli hafıza miktarını programın yapısı belirler.

5.1.3 Giriş/Çıkış Bölümü (I / O)

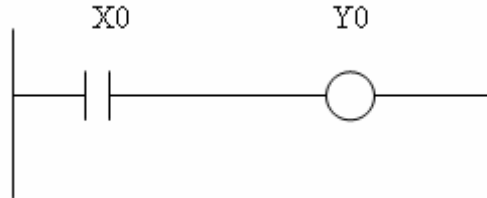
Bu bölüm sahadaki cihazlarla PLC nin haberleşmesini sağlar. Onlardan sinyal alıp, onlara sinyal gönderme amaçlı kullanılır.Giriş cihazlarına örnek sensörler ve butonlardır. Çıkış cihazları ise motorlar, gösterge ışıkları ve valflerdir.

5.2 PLC' nin Programlanması

Bu kısımda PLC de kullanılan temel programlama komutlarının herbiri kısaca ele alınmıştır. Başlıkta ele alınan komutlar tezde de kullanılan Allen Bradley PLC programından verilen örneklerle somutlaştırılmıştır.

LOAD :

Ladder diyagramında yatay çizginin sol tarafına yerleştirilen normalde açık kontak komutudur.

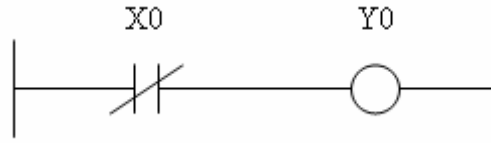


Şekil 5.1 Load komutu

Şekil 5.1' deki program satırında X0 giriş kontağı ON konumuna gelmesi durumunda Y0 çıkışının kontağı ON konumuna gelir.

LOAD NOT :

Ladder diyagramında yatay çizginin sol tarafına yerleştirilen normalde kapalı kontak komutudur.



Şekil 5.2 Load not komutu

Şekil 5.2 'deki örnekte X0 giriş kontağı off konuma geldiğinde Y0 çıkışının kontağı ON konumuna gelir.

OUT :

Ladder akım diyagramında sebep şartlarının oluşması durumunda sonuç şartlarına çıkış veren komuttur.

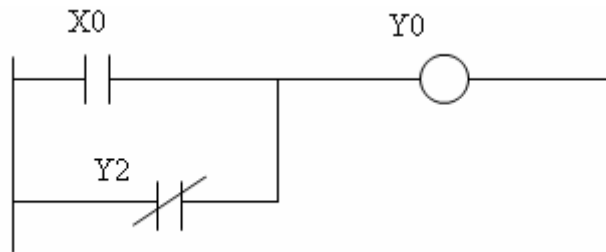


Şekil 5.3 Out komutu

Şekil 5.3' teki örnekte X0 giriş kontağı ON konumuna geldiğinde Y0 çıkış kontağı ON konumuna gelir.

OR :

Normalde açık olan giriş kontaklarını olarak birbirine paralel olarak birbirine bağlamaya yarayan program komutudur.

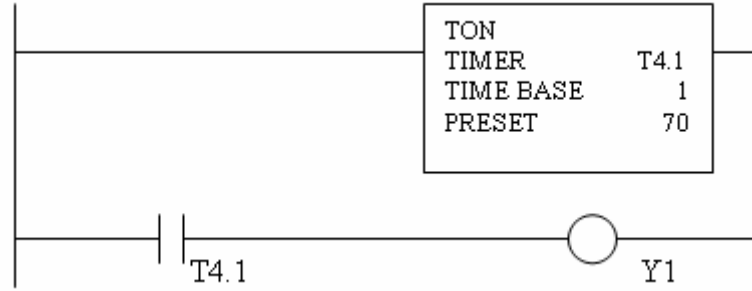


Şekil 5.4 Or komutu

Örnekte X0 kontağı ON konumuna geldiğinde veya Y2 çıkış kontağı ON konumuna geldiğinde Y0 çıkış kontağı ON konumuna gelir.

TIMER :

Timer belli bir zaman sonra açılması veya kapanması istenen kontakların kontrolünde kullanılır. Genelde TON (Timer On) tipi zamanlayıcılar kullanılır. Örnekte bir timer komutunda neler olduğu gösterilmiştir. Timer'ın hangi kontağa çıkış vereceği, birim zamanı, ne kadar süre sonra çıkış vereceği girilir. Bu örnekte enerji verildikten 70 sn sonra T4.1 e enerji gelir ve Y1 çıkışında enerjilenmiş olur.

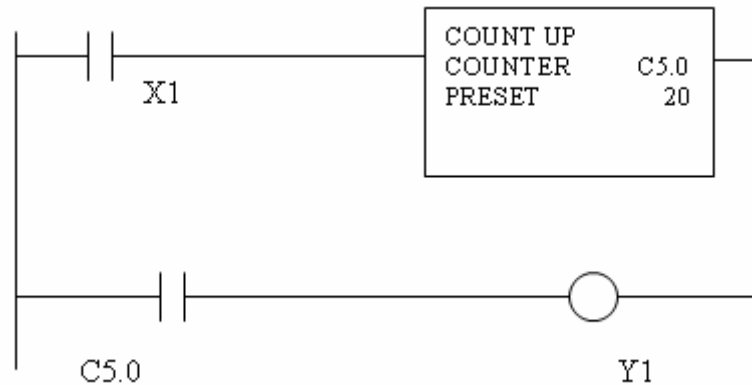


Şekil 5.5 Timer komutu

Örnekte bir timer komutunda neler olduğu gösterilmiştir. Timer'ın hangi kontağa çıkış vereceği, birim zamanı, ne kadar süre sonra çıkış vereceği girilir. Bu örnekte enerji verildikten 70 sn sonra T4.1 e enerji gelir ve Y1 çıkışında enerjilenmiş olur.

COUNTER :

Sayıcı girişine verilen "1" sinyalinin belirli sayısından sonra çıkışını 1 yapan komuttur. Yukarı veya aşağı yönde olabilirler.



Şekil 5.6 Counter komutu

Örnekte X1 sinyali 20 kere çıkış verdiğinde C5.0 kontağı çıkış verecektir ve Y1 enerjilenir.

MOVE :

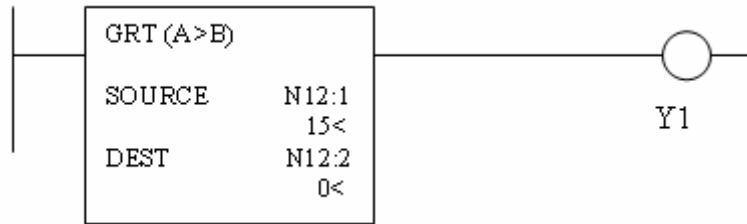
Move girişine verilen bilgiyi çıkışına aktaran ve girişin bu aktarmadan etkilenmediği komuttur. Bu örnekte X1 kontağı enerjilendiği anda N12:1 adresindeki bilgi N12:2 adresine kopyalanacaktır.



Şekil 5.7 Move komutu

GREATER THAN :

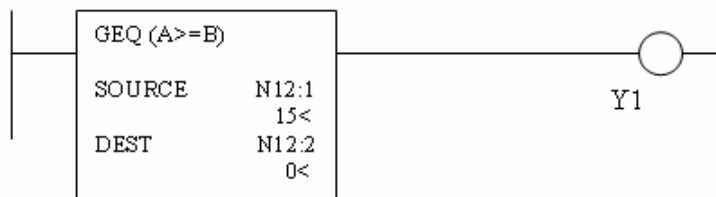
GRT büyük karşılaştırma komutudur. Örnekte N12:1 adresindeki değer N12:2 adresinden okunan değerden büyük olduğu sürece Y1 kontağı çıkış verecektir.



Şekil 5.8 Greater than komutu

GREATER THAN OR EQUAL :

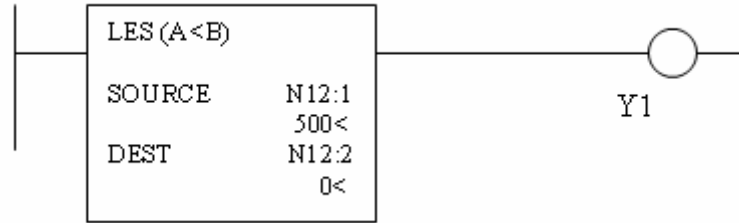
GEQ büyük yada eşit karşılaştırma komutudur. Şekil 5.9' daki örnekte N12:1 adresindeki değer N12:2 adresinden okunan değerden büyük yada bu değere eşit olduğu sürece Y1 kontağı çıkış verecektir.



Şekil 5.9 Greater than or equal komutu

LESS THAN :

LES küçük karşılaştırma komutudur. Bu örnekte N12:1 adresindeki değer N12:2 adresinden okunan değerden küçük olduğu sürece Y1 kontağı çıkış verecektir.



Şekil 5.10 Less than komutu

SET :

Sürekli çalışması istenilen bir çıkış veya yardımcı kontağı ON konumuna getirir. Aynı kontak RESET komutuyla durdurulmadığı sürece çalışmasını sürdürecektir.



Şekil 5.11 Set komutu

Örnekte X1 kontağı enerjilendiğinde B3:0/11 adresindeki bit setlenecek ve sürekli '1' sinyali verecektir.

RESET :

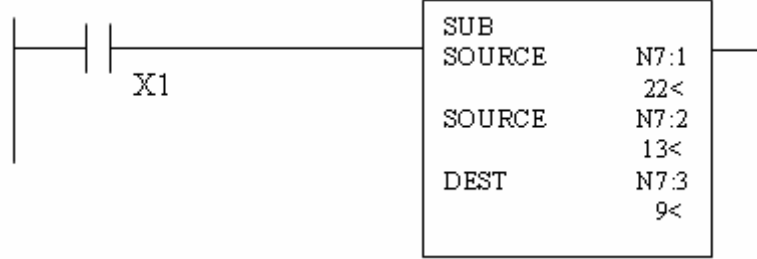
Set edilen veya çalışan bir röleyi durdurmak yada zamanlayıcı ve sayıcıları sıfırlamak için kullanılır. Örnekte X1 kontağı enerjilendiğinde C5:0 counter resetlenecektir.



Şekil 5.12 Reset komutu

SUB :

Sub komutu iki farklı adreste saklı olan değerleri birbirinden çıkarır ve sonucu çıkış adresine yazar.

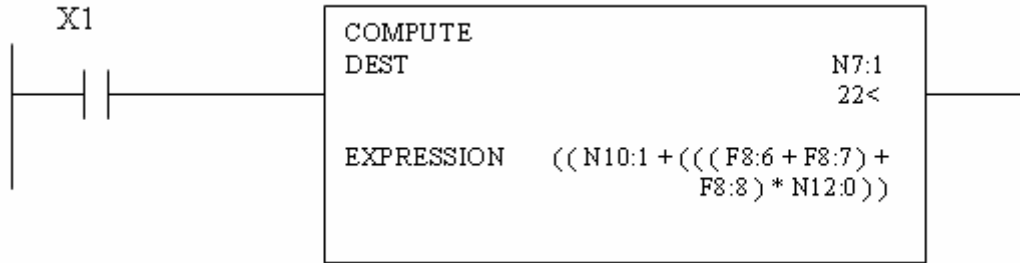


Şekil 5.13 Sub komutu

Bu örnekte X1 kontağı enerjilendiği sürece N7:1 adresindeki bilgiden N7:2 adresindeki bilgi çıkarılarak N7:3 adresine yazılacaktır.

CPT :

Compute komutu değişik matematiksel formüllerin programa eklenmesi için kullanılır.



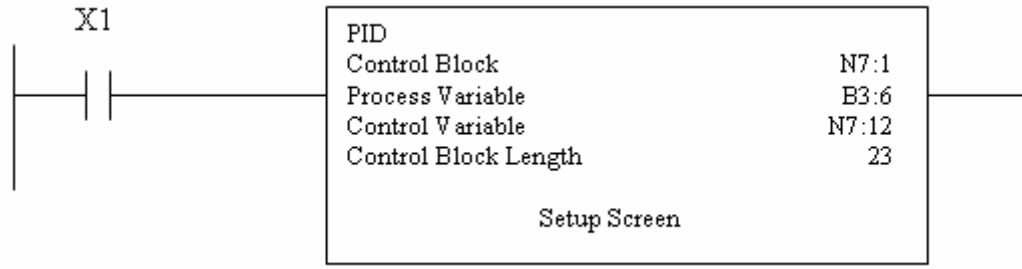
Şekil 5.14 Cpt komutu

Şekil 5.14' teki komut satırında hesaplanması istenilen denklemin sonucu N7:1 adresine yazılmaktadır. F adresleri ondalık sayılar için kullanılmaktadır. Sistem çalışırken denklem içerisindeki adreslerde olabilecek değişiklikler compute komutunun sonucunda hemen etkilemektedir.

PID :

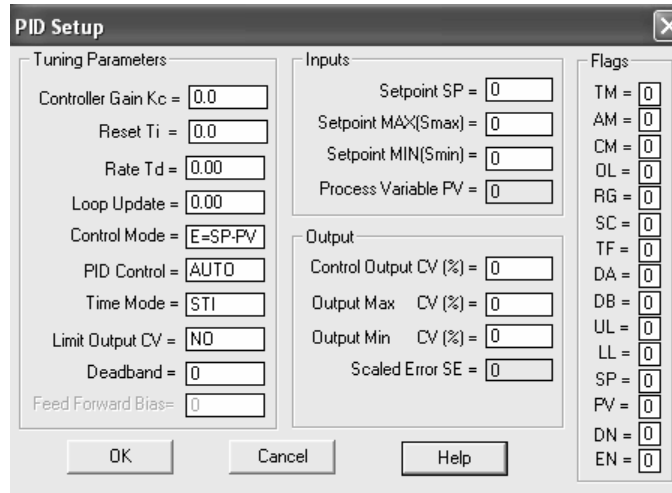
PLC programlarında komutlar dışında bir çok komut kullanılarak yapılacak ancak çok sık kullanılan bazı fonksiyonlar blok olarak bulunabilmektedir. Bunlardan biriside endüstriyel

uygulamalarda sıkça rastlanılan PID fonksiyonu için oluşturulan PID bloğudur.



Şekil 5.15 PID komutu

Programda Şekil 5.16’ da görülen ayar penceresi açılarak P, I, D değerleri istenilen şekilde ayarlanır. Sistemin oturması istenen referans değerinde (SP) bu pencereden ayarlanmaktadır.



Şekil 5.16 PID bloğu ayar penceresi

END :

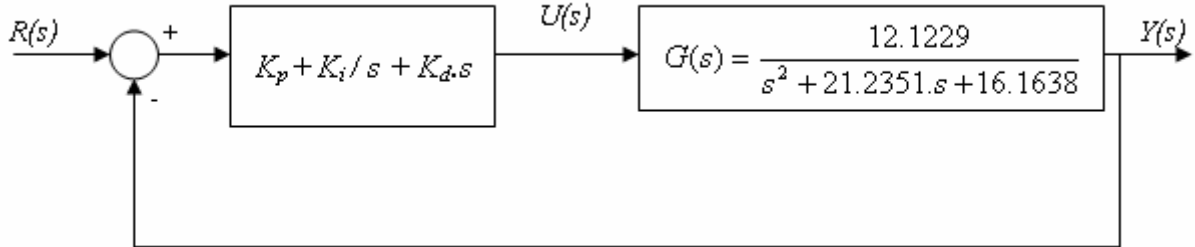
Bu komut programı sonlandırmak için yazılımın son komut satırına konur. Kullanılması mecburidir.



Şekil 5.17 End komutu

6. ÇİFT CİDARLI REAKTÖR TANK ISITMA SİSTEMİNİN PLC İLE KONTROLÜ

Bu bölümde Bölüm 4' te matematiksel modelini çıkardığımız çift cidarlı reaktör tank ısıtma sistemini PLC yardımıyla PID yöntemiyle kontrolünü sağlayacağız. Sistemin simülasyonunu yapabilmek için çıkardığımız bu matematiksel modeli PLC' ye tanıtacağız.



Şekil 6.1 Kontrol edilecek sistemin blok diyagramı

$$G(s) = \frac{Y(s)}{U(s)} = \frac{12.1229}{s^2 + 21.2351.s + 16.1638} \quad (6.1)$$

Daha öncede belirttiğimiz gibi sistemi MATLAB yardımıyla ayrıklaştıralım.

`s = tf('s');`

`Gs = tf([12.1229],[1.0000 21.2351 16.1638]);`

`z = tf('z');`

`Gz = c2d(Gs, 1, 'tustin');`

Programın bize vereceği z tabanındaki model (6.2) 'deki gibidir.

$$G(z) = \frac{Y(z)}{U(z)} = \frac{0.1936 + 0.3871z^{-1} + 0.1936z^{-2}}{1 + 0.3834z^{-1} - 0.3561z^{-2}} \quad (6.2)$$

Ayrıklaştırılmış olan bu sistemi fark denklem haline getirelim(6.3). Böylece sistemin matematiksel modelini PLC ye aktarabileceğimiz forma getirmiş oluruz.

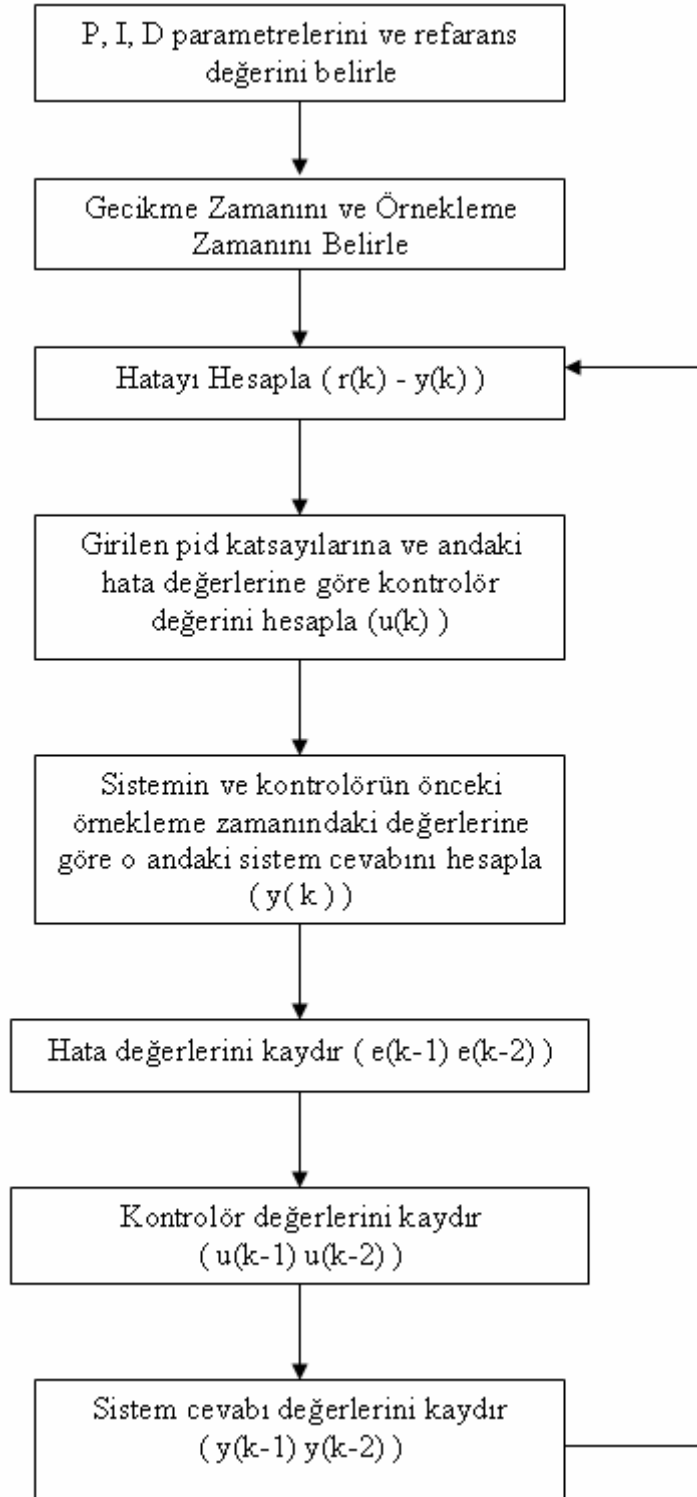
$$y(k) = -0.3834.y(k-1) + 0.3561.y(k-2) + 0.1936.u(k) + 0.3871.u(k-1) + 0.1936.u(k-2) \quad (6.3)$$

PID fonksiyonun fark denklem halindeki formu (6.4)' tedir.

$$u(k) = u(k-1) + (K_p + K_i + K_d).e(k) - (K_p + 2.K_d).e(k-1) + K_d.e(k-2) \quad (6.4)$$

6.1 Akış Diyagramı

Programı yazmaya başlamadan önce akış diyagramımızı oluşturalım. Program sürecinin başlama anından bitiş anına kadar işlemlerin hangi sırayla yapılacağını gösteren bu diyagram, program yazımını oldukça kolaylaştırmaktadır.



Şekil 6.2 PLC programının akış diyagramı

6.2 PLC Programı

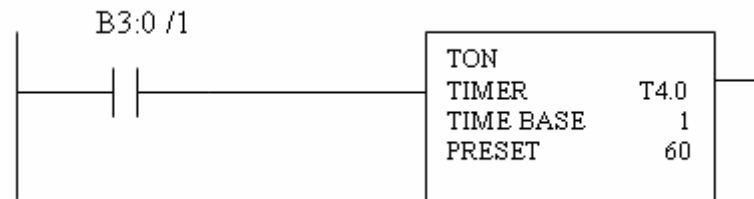
Akış diyagramı oluşturulduktan sonra program kolay bir biçimde yazılabilir. Burada ladder programın her satırını tek tek ele alınarak anlatımı yapılacaktır. Öncelikle kullanacağımız parametrelerin her birine bir adres ataması yapılır. Bu adresler Çizelge 6.1’de gösterilmiştir.

Çizelge 6.1 PLC programında kullanılan parametrelerin adresleri

Adı	Açıklaması	Adres
T_1	Gecikme Zamanı	T4:0
T_2	Örnekleme zamanı	T4:1
$r(k)$	Sistem girişi (Referans değeri)	F8:0
$e(k)$	Hata değeri	F8:4
$e(k-1)$	Bir önceki örnekleme zamanındaki hata değeri	F8:5
$e(k-2)$	İki önceki örnekleme zamanındaki hata değeri	F8:6
$u(k)$	PID kontrolör değeri	F8:7
$u(k-1)$	Bir önceki örnekleme zamanındaki PID kontrolör değeri	F8:8
$u(k-2)$	İki önceki örnekleme zamanındaki PID kontrolör değeri	F8:9
P	Oransal kontrolör katsayısı	F8:10
I	İntegral kontrolör katsayısı	F8:11
D	Türevsel kontrolör katsayısı	F8:12
$y(k)$	Sistem çıkış değeri	F8:1
$y(k-1)$	Bir önceki örnekleme zamanındaki Sistem çıkış değeri	F8:2
$y(k-2)$	İki önceki örnekleme zamanındaki Sistem çıkış değeri	F8:3

Satır 1:

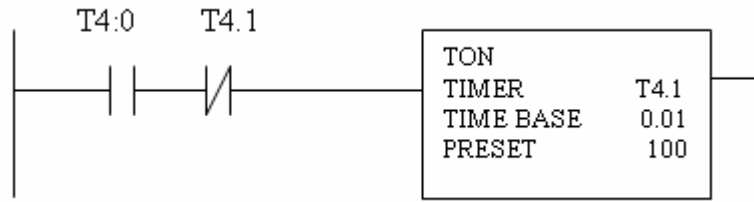
Programın ilk satırına gecikme zamanını istediğimiz değere ayarlayabileceğimiz bir zamanlayıcı komutu yerleştirilmiştir. Bu komut sayesinde simulasyon gerçek bir sistemde olduğu gibi belli bir zaman gecikmesiyle çalışacaktır.



Şekil 6.3 Program Satırı 1

Satır 2:

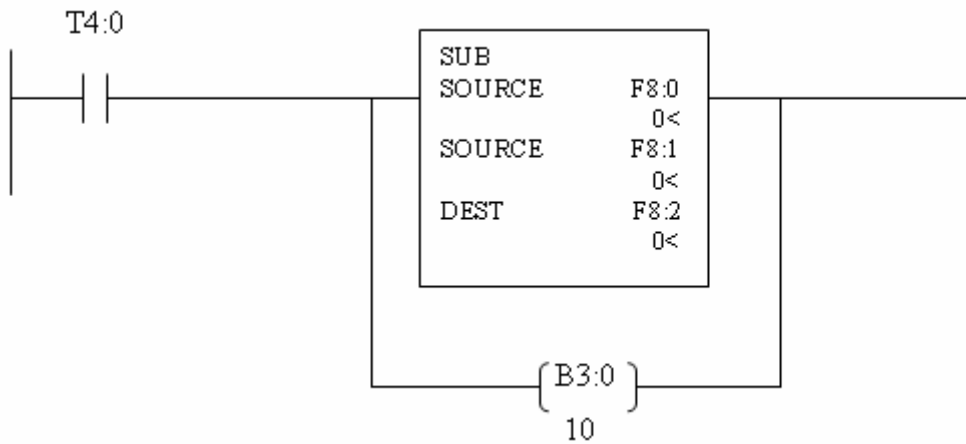
Akış diyagramında da (Şekil 6.2) görüldüğü gibi öncelikle bir örnekleme zamanı belirlenmelidir. Program her örnekleme aralığında bir çıkış değeri alıp bunu sisteme yansıtarak referans değere daha yakınlaştıracaktır. Bunun için bir timer kullanılır. Timer'ın başına kendi kapalı kontağını koyarak geri kalan programın her örnekleme zamanı aralığında çıkış vermesi sağlanır ve bu şekilde bir döngü oluşturulur. Bu sistem için örnekleme aralığı 1 saniye olarak seçilmiştir.



Şekil 6.4 Program satırı 2

Satır 3:

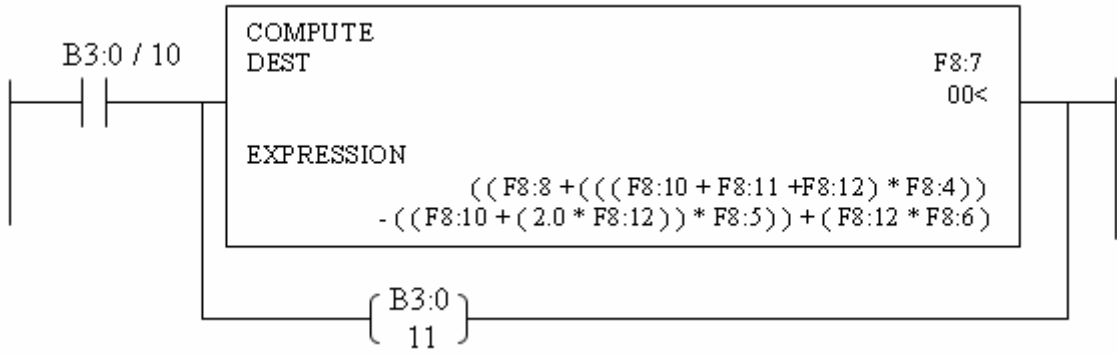
Örnekleme zamanı belirlendikten sonra şimdi PID fonksiyonunda kullanılacak olan hata değerini bulunur. Bu değer ulaşmak istenilen referans değeriyle, son örnekleme zamanında alınan çıkış değeri arasındaki farktır. Bu hesaplama Sub komutuyla yapılır. Satırın başına timer çıkış kontağı koyularak komut çalıştırılır ve aynı satıra birde çıkış kontağı ekleyerek bir alttaki satıra sırayla geçilmesi sağlanır.



Şekil 6.5 Program satırı 3

Satır 4:

Hata değeri belirlendikten sonra PID denetleyici oluşturulur. Bunun için CPT komutunu kullanılır. Bu komut yardımıyla (6.4) 'te fark denklem haline getirilen PID denetleyici PLC 'ye aktarılır. Burada bulunan tüm parametrelere Çizelge 6.1 'de birer adres ataması yapılmıştı. Buna göre komut satırımız Şekil 6.5 'teki gibi olacaktır.

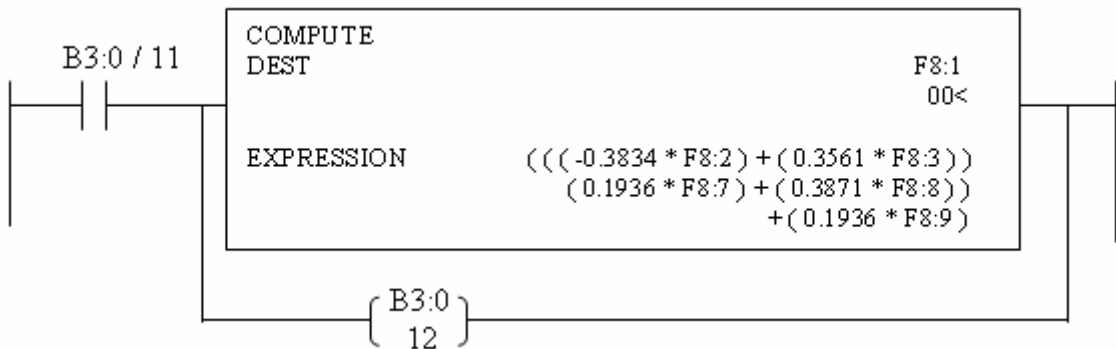


Şekil 6.6 Program satırı 4

Böylece denetleyicinin çıkış değeri $u(k)$ hesaplanmış oldu. Aynı anda B3:0/11 çıkışı enerjilendirilerek bir alt satıra geçilir.

Satır 5:

Şimdi kontrol edilen sistemi PLC 'ye tanıtılacaktır. Bunun için (6.3) 'te sistemin matematiksel modelini fark denklem haline getirmiştik. Bu fark denklem yine CPT komutu yardımıyla PLC 'ye aktarılır.



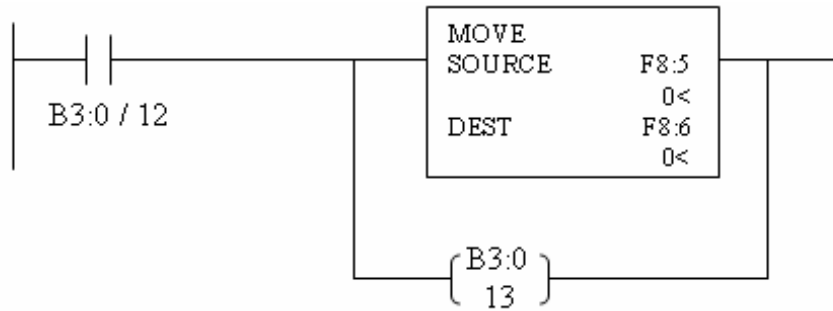
Şekil 6.7 Program satırı 5

Bu hesaplamanın sonucunda ilk örnekleme zamanı için bir çıkış değeri elde edilmiş oldu.

Bundan sonraki aşamada bir sonraki örnekleme zamanındaki çıkış değerini elde etmekte kullanılacak olan $e(k-1)$, $e(k-2)$, $u(k-1)$, $u(k-2)$, $y(k-1)$ ve $y(k-2)$ değerleri elde edilir. Bunun için yapılması gereken bu değerlerin MOV komutuyla kaydırılarak parametrelerin hafızada tutulmasını sağlamaktır.

Satır 6:

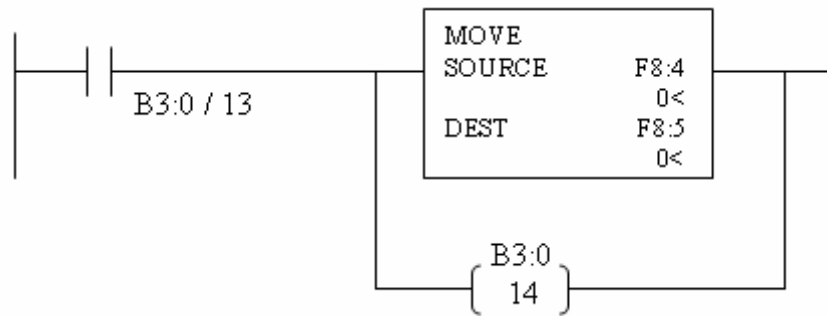
İlk olarak (k-1) anındaki hata değerimiz $e(k-1)$ ' i $e(k-2)$ ' ye taşınmalıdır.



Şekil 6.8 Program satırı 6

Satır 7:

$e(k-2)$ adresinin değerini hafızaya almamızın ardından yine $e(k-1)$ değeri belirlenmelidir. Bunun için içinde bulunduğumuz k döngüsünün hata değeri olan $e(k)$ ' yı $e(k-1)$ adresine kopyalanmalıdır.

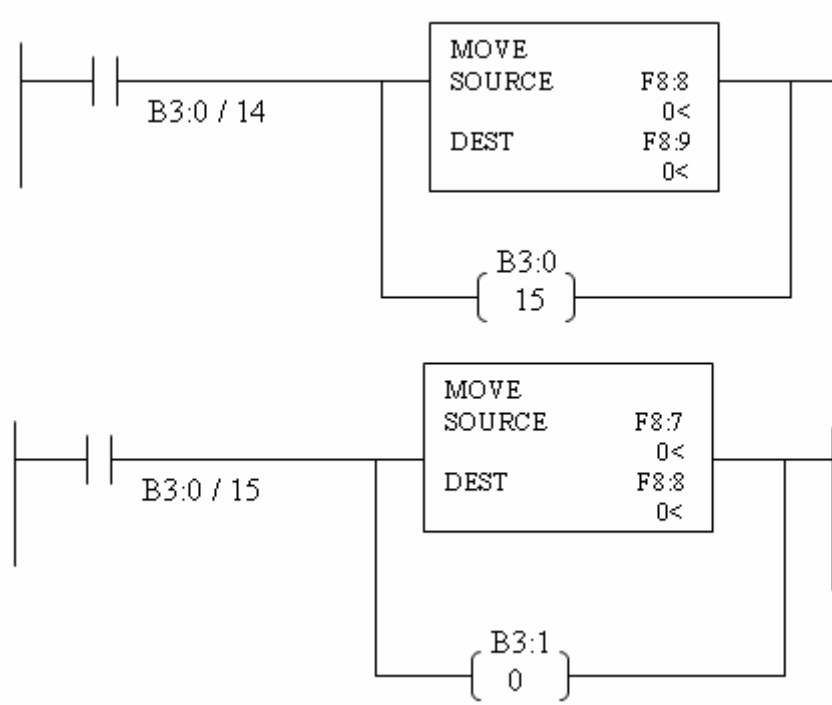


Şekil 6.9 Program satırı 7

Satır 8 ve Satır 9 :

Control value (CV) olarak adlandırılan kontrol değeri $u(k)$ bir sonraki döngülerin çıkışını etkilemektedir. Bu nedenle hata değerlerine yaptığımız gibi sırasıyla $u(k-1)$ değerini $u(k-2)$

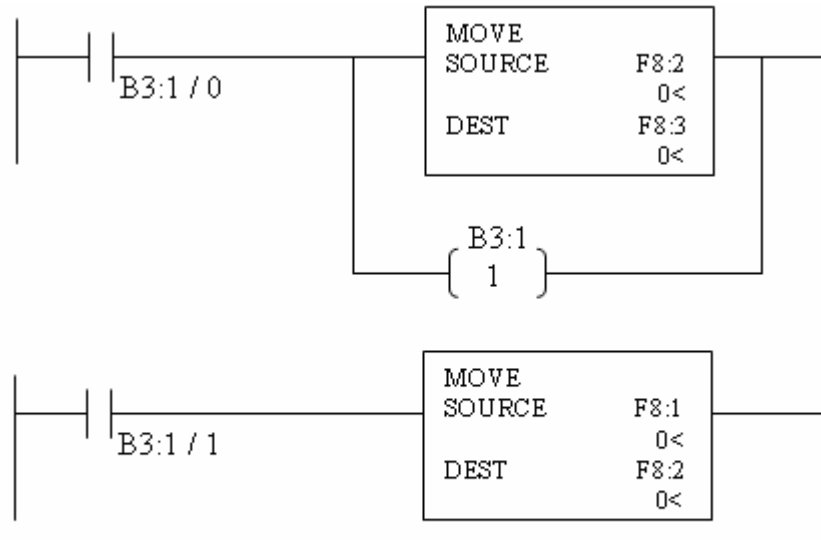
adresine, $u(k)$ değerini de $u(k-1)$ adresine kopyalanmalıdır.



Şekil 6.10 Program satırı 8 ve 9

Satır 10 ve satır 11:

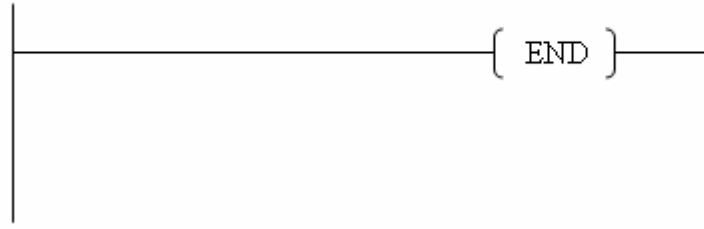
$y(k)$ değerimiz ise process value (PV) olarak adlandırılır. Bu değerinde k ve $k-1$ anındaki değerleri sırayla kaydırılmalıdır.



Şekil 6.11 Program satırı 10 ve 11

Satır 12:

Son olarak END komutuyla program bitirilir.



Şekil 6.12 Program satırı 12

Programın yazımı tamamlandıktan sonra Allen – Bradley PLC programı olan RSLogix in özelliği olan Şekil 6.12’ de görülen Custom Data Monitör yardımıyla K_P , K_D , K_I ve referans giriş değeri olan set point (SP) değerlerinin atanması gerekmektedir. Diğer PLC programlarında bu değerler MOV komutuyla verilebilir.

The screenshot shows a window titled "CDM 0 - Untitled". It contains a table with two columns: "Address" and "Value". The table has four rows of data.

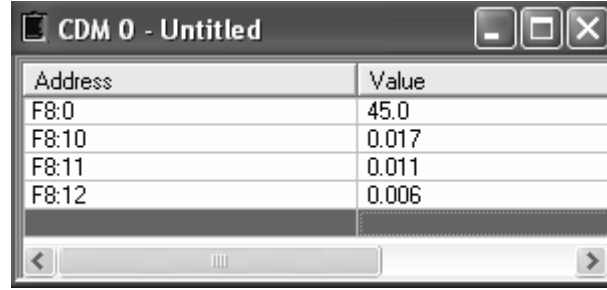
Address	Value
F8:0	25.0
F8:10	1.8
F8:11	1.3
F8:12	0.7

Şekil 6.13 Veri monitörü aracılığıyla PID değerlerinin atanması

7. DENEYSEL SONUÇLAR

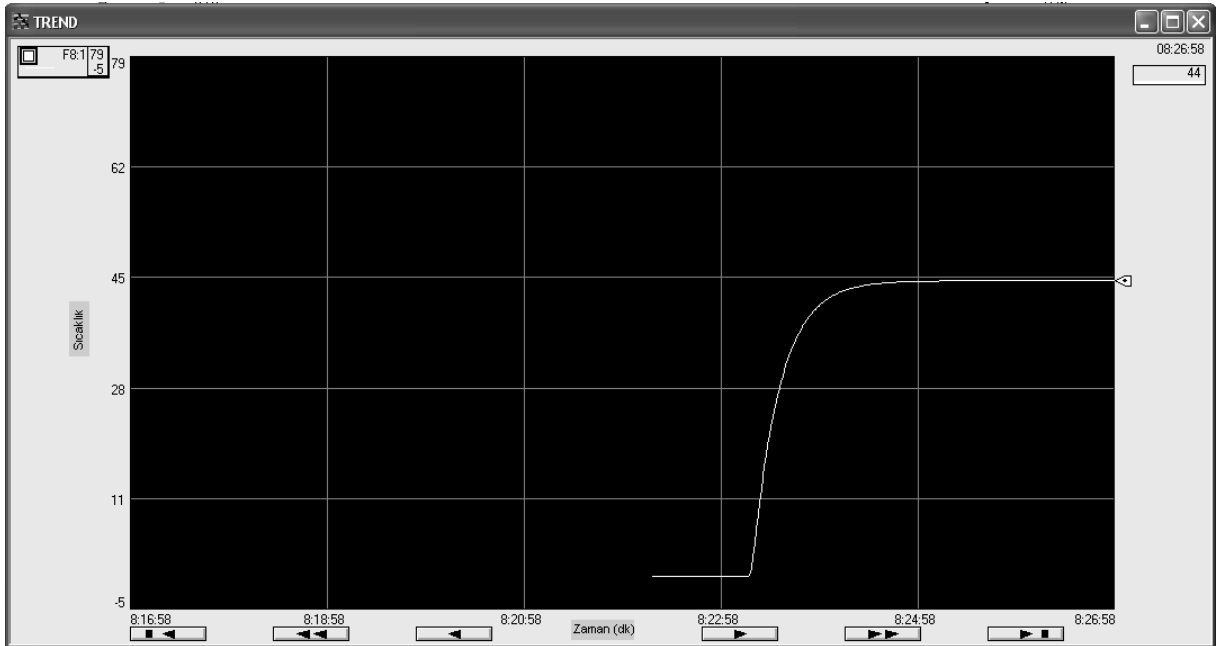
Bu bölümde PLC için hazırlanan PID denetleyicisinin verdiği sonuçlar ele alınmıştır. Bölüm 4' te çeşitli denemeler yaparak bulmaya çalıştığımız sistem için en uygun PID katsayılarının PLC programında çalıştıracak olursak Şekil 7.6' daki grafik elde edilmiştir.

Bulunan bu K_P , K_D , K_I değerlerini yine veri monitörü aracılığıyla PLC 'ye girilmiştir. (Şekil 7.5).



Address	Value
F8:0	45.0
F8:10	0.017
F8:11	0.011
F8:12	0.006

Şekil 7.1 PID katsayılarının PLC 'ye aktarımı



Şekil 7.2 Uygun PID katsayılarıyla sistemin cevabı

Şekil 7.2 'de görüleceği gibi Bölüm 4 'de MATLAB yardımıyla bulduğumuz dijital kontrol cevaplarıyla sistemi PLC 'ye fark denklem haline getirip yükleyerek artımsal PID formunda oluşturduğumuz PID algoritmasıyla kontrolünde aynı sonuçlar elde edilmiştir.

Diđer yandan çift cidarlı reaktör tank sıcaklık sisteminin PID ile etkin kontrolü sağlanmış, PID katsayılarında gerekli ayarlar yapılarak sıcaklık kontrolünde makul oturma zamanı ve aşımı olmayan bir cevap elde edilmiştir.

8. SONUÇLAR VE TARTIŞMA

Bu çalışmada günümüzde endüstriyel proseslerin bir çoğunda kullanılan PID kontrol, programlanabilir lojik kontrolörde (PLC) yazılan bir algoritma ile uygulanmıştır. Kontrol edilen sistem örneği olarak çift cidarlı reaktör tank ısıtma sistemi kullanılmış, algortima yazılırken PLC'ye sistemin doğrusallaştırılmış transfer fonksiyonu aktarılmış, PID kontrolü için ise artımsal PID formu kullanılmıştır.

PID katsayıları bulunurken Ziegler-Nichols ve Cohen-Coon yöntemleri kullanılmış ancak bu yöntemlerle bulunan katsayılar kullanıldığında iyi sonuçlar elde edilememiştir. PID katsayılarında yapılan çeşitli denemeler sonucu sistem aşımının olmadığı ve aşımsız kararlı bir cevap elde edilmiştir.

Sistem PLC'ye bağlı durumda iken, PID katsayılarında çeşitli oynamalar yapmak sistemi kararsızlığa götürebilir. Bu tezde yazılan algoritma ile matematiksel modeli çıkarılan sistem PLC'ye aktarılabilen, PID katsayılarında çeşitli denemeler yaparak sistemin tepkisi birebir PLC programı üzerinde görülebilmektedir. Sistemin devreye alınma aşamasında daha önce bulunan ve denenmiş olan katsayılar girilerek etkin bir gerçek zamanlı kontrol sağlanabilir.

KAYNAKLAR

- Aström K. J and Hagglund T., (1995), PID Controllers, 2nd ed. ISA, New York
- Bequette B. W., (2003), Process Control, Modeling, Design and Simulation, Prentice Hall, New Jersey.
- Bequette B. W., (1998), Process Dynamics, Modeling, Analysis and Simulation, Prentice Hall, New Jersey.
- Kuo B., (1999), Automatic Control Systems, P. Hall, New Jersey.
- Kurtalan S, (2001), PLC ile Endüstriyel Otomasyon, Birsen Yayınevi, İstanbul.
- Ogata K., (2002), Modern Control Engineering, 4th ed., P. Hall
- Ogata K., (2003), System Dynamics, 4th ed., P. Hall
- Ogata K., (1995), Discrete Time Control Systems, 2nd ed., P. Hall
- Pan S., Zheng P. (2003) “A Study for a Optimized AC Motor Velocity Synchronous Control System by Using PLC and Multi-Frequency Converter”, Electrical Machines and Systems, ICEMS 2003. Sixth International Conference
- Qiang S., Gaol X. Z., and Zhuang X., (2002), “PLC-based Control Systems for Industrial Production of Fuel Alcohol”, IEEE ICIT’02, Bangkok, THAILAND
- Uzunoğlu M. , (2002), MATLAB 6.0 - 6.5, Türkmen Kitabevi, İstanbul.
- Ziemba R., (1988), “Use of a Programmable Logic Controller (PLC) for Injection Molding Machinery Temperature, Position, Velocity and Pressure Control”, Industrial Applications Society Annual Meeting.

ÖZGEÇMİŞ

Doğum tarihi	04.01.1983	
Doğum yeri	İstanbul	
Lise	1997 - 2001	Kenan Evren Süper Lisesi
Lisans	2001 - 2005	Yıldız Üniversitesi Elektrik-Elektronik Fak. Elektrik Mühendisliği Bölümü
Yüksek Lisans	2005 -	Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Elektrik Müh. Anabilim Dalı Kontrol ve Otomasyon Programı