

**ESNEK ROBOT KOLLARI İÇİN
KONTROL YÖNTEMLERİ
UYGULAMALARI**

İSA KÜÇÜKYILMAZ
Yüksek Lisans Tezi

**Anadolu Üniversitesi Fen Bilimleri Enstitüsü
Elektrik-Elektronik Mühendisliği
Anabilim Dalı
Ekim 2000**

JÜRİ VE ENSTİTÜ ONAYI

İsa Küçükylmaz'ın "Esnek Robot Kolları İçin Kontrol Yöntemleri Uygulamaları" başlıklı Elektrik-Elektronik Mühendisliği Anabilim Dalındaki, Yüksek Lisans Tezi 28.09.2000 tarihinde, aşağıdaki jüri tarafından Anadolu Üniversitesi Eğitim - Öğretim ve Sınav Yönetmeliğinin ilgili maddeleri uyarınca değerlendirilerek kabul edilmiştir.

Adı Soyadı

İmza

Üye (Tez Danışmanı) : Prof. Dr. Altuğ İFTAR

Üye : Yrd. Doç. Dr. Osman PARLAKTUNA

Üye : Yrd. Doç. Dr. D. Gökhan ECE

Anadolu Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun
.04.10.2000.. tarih ve26/1..... sayılı kararı ile onaylanmıştır.

Enstitü Müdürü
Prof. Dr. Orhan ÖZER
Fen Bilimleri Enstitüsü
MÜDÜRÜ

ÖZET

Yüksek Lisans Tezi

ESNEK ROBOT KOLLARI İÇİN KONTROL YÖNTEMLERİ UYGULAMALARI

İSA KÜÇÜKYILMAZ

Anadolu Üniversitesi

Fen Bilimleri Enstitüsü

Elektrik-Elektronik Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Altuğ İFTAR

2000

Bu tezde ilk olarak tek eklemli esnek robot kolu kontrolü için deneysel düzenek hazırlanmış, hazırlanan bu düzeneğe PD, PID, doğrusal olmayan PD benzeri ve doğrusal olmayan PID benzeri kontrol yöntemleri uygulanarak verdikleri tepkiler incelenmiştir. Titreşim bilgisi ise esnek kolun üzerine yerleştirilen piezoelektrik seramik sensörler (PZT) aracılığıyla elde edilmiş ve sisteme geri beslenerek bu bilgiyi de kullanan yeni kontrol yöntemleri uygulanmıştır. Daha sonraki aşamada ise iki eklemli esnek robot kolu düzeneği kurulmuş ve uygulanan kontrol yöntemlerine sistemin verdiği tepki incelenmiştir. Elde edilen sonuçlar ortaya konularak etkinlikleri tartışılmıştır. Sistemin kontrolü için C programlama dilinde kontrol programları geliştirilip uygulanmıştır.

Anahtar Kelimeler: Esnek Robot Kolları, P(I)D Kontrol, Doğrusal Olmayan P(I)D Benzeri Kontrol, Piezoelektrik Seramik Sensörler

ABSTRACT**Master of Science Thesis****APPLICATIONS OF CONTROL STRATEGIES FOR
FLEXIBLE ROBOT MANIPULATORS****İSA KÜÇÜKYILMAZ****Anadolu University****Graduate School of Natural and Applied Sciences****Electrical and Electronics Engineering Program****Supervisor: Prof. Altuğ İFTAR****2000**

In this thesis, firstly, an experimental setup is developed to control a single link flexible robot arm. Response of the manipulator to PD, PID, non-linear PD and PID like control strategies is investigated. The vibration of the manipulator is sensed by using piezoelectric ceramic sensors mounted on the manipulator and new control algorithms are applied by feeding back the vibration data to the system. Then, an experimental setup for a two-link flexible robot arm system is developed and the system response to the applied control strategies is investigated. The obtained results are presented and their effectiveness are discussed. Control programs are developed in C programming language to control the system.

Keywords: Flexible Robot Arms, P(I)D Control, Non-Linear P(I)D Like Control, Piezoelectric Ceramic Sensors

TEŞEKKÜR

Çalışmalarında beni yönlendirerek benden yardımlarını esirgemeyen hocam sayın Prof. Dr. Altuğ İftar'a, Yrd. Doç. Dr. Osman Parlaktuna'ya, Yrd. Doç. Dr. Gökhan Ece'ye ve Yrd. Doç. Dr. Aydın Doğan'a teşekkür ederim. Ayrıca çalışmalarında beni destekleyen nişanlım Aysin Şenel'e, arkadaşlarım Serhat Aydın'a, Hakkı Ulaş Ünal'a ve Osman Afşin Güngör'e teşekkür ederim.

Bu çalışma "Esnek Robotik Sistemler İçin Görüntü Geri Beslemeli Gürbüz Kontrol Tasarımı Yaklaşımlarının Uygulanması" başlıklı araştırma projesi çerçevesinde Anadolu Üniversitesi Araştırma Fonu ve Devlet Planlama Teşkilatı tarafından desteklenmiştir.

İÇİNDEKİLER

	<u>Sayfa</u>
ÖZET	i
ABSTRACT	ii
TEŞEKKÜR	iii
İÇİNDEKİLER	iv
ŞEKİLLER DİZİNİ	vi
ÇİZELGELER DİZİNİ	viii
1. GİRİŞ	1
1.1 Geçmişte Yapılmış Çalışmalar	1
1.2 Tezde Yapılan Çalışmalar	4
2. DENEYSEL DÜZENEGİN HAZIRLANMASI	5
2.1 Tek Eklemlili Esnek Robot Kolu Düzeneginin Hazırlanması	5
2.1.1 Düzenekte kullanılan malzemeler	5
2.2 İki Eklemlili Esnek Robot Kolu Düzenegi	10
2.3 Piezoelektrik Seramik Sensörler (PZT)	13
2.3.1 Piezoelektrik seramik sensörler için yük kuvvetlendiricileri	13
2.3.2 Piezoelektrik seramik sensörün (PZT) modellenmesi	14
2.4 Programlama	16
3. KONTROLÖR TASARIMI	21
3.1 Tek Eklemlili Esnek Robot Kolları İçin Uygulanan Kontrol Yöntemleri	21
3.1.1 PD kontrolör tasarımı	21
3.1.2 PID kontrolör tasarımı	22
3.1.3 Doğrusal olmayan PD benzeri kontrolör tasarımı	22
3.1.4 Doğrusal olmayan PID benzeri kontrolör tasarımı	23
3.1.5 Titreşim bilgisinin kontrolöre ilave edilmesi	23
3.2 İki Eklemlili Esnek Robot Kolu İçin Uygulanan Kontrol Yöntemleri	24
3.2.1 PID kontrolör tasarımı	24
3.2.2 Doğrusal olmayan PID kontrol yöntemi	24
4 DENEYSEL SONUÇLAR.....	26
4.1 Tek Eklemlili Esnek Robot Kolları İçin Alınan Deneysel Sonuçlar	26

İÇİNDEKİLER (Devamı)

	<u>Sayfa</u>
4.1.1 A kolu için elde edilen deneysel sonuçlar	26
4.1.2 B kolu için elde edilen deneysel sonuçlar	41
4.1.3 C kolu için elde edilen deneysel sonuçlar	44
4.2 İki Eklemlı Esnek Robot Kolu İin Alınan Deneysel Sonular.....	47
5. SONU.....	51
KAYNAKLAR	53
EK 1 Tek Eklemlı Esnek Robot Kolu İin Geliştirilen PID Kontrol Programı	55
EK 2 Tek Eklemlı Esnek Robot Kolu İin Uygulanan Doğrusal Olmayan PID Benzeri Kontrol Yöntemi İin Geliştirilen Program	62
EK 3 İki Eklemlı Esnek Robot Kolu İin Uygulanan Doğrusal Olmayan PID Benzeri Kontrol Yöntemi İin Geliştirilen Program	69

ŞEKİLLER DİZİNİ

	<u>Sayfa</u>
2.1 Tek eklemlı esnek robot kolu düzeneđi	7
2.2 Tek eklemlı esnek robot kolu düzeneđinin şematik gösterimi	8
2.3 Fark alıcı sinyal işleme devresi	9
2.4 İki eklemlı esnek robot kolu düzeneđi	11
2.5 İki eklemlı esnek robot kolunun şematik gösterimi	12
2.6 PZT için kullanılan sinyal işleme devresi	14
2.7 PZT sensörün yapısı	14
2.8 Hesaplamalarda kullanılacak parametreler	15
2.9 Tek eklemlı esnek robot kolu için geliştirilen program akış şeması	19
2.10 İki eklemlı esnek robot kolu için geliştirilen program akış şeması	20
4.1 PD kontrolör sonuçları (A kolu ve 10 derece referansı için)	27
4.2 PD kontrolör sonuçları (A kolu ve 20 derece referansı için).....	27
4.3 Doğrusal olmayan PD benzeri kontrolör sonuçları (A kolu ve 10 derece için).....	28
4.4 Doğrusal olmayan PD benzeri kontrolör sonuçları (A kolu ve 20 derece için).....	29
4.5 PD kontrolör sonuçları (A kolu ve 50 derece referansı için)	30
4.6 PD kontrolör sonuçları (A kolu ve 80 derece referansı için)	31
4.7 Doğrusal olmayan PD benzeri kontrolör sonuçları (A kolu ve 50 derece referansı için)	32
4.8 Doğrusal olmayan PD benzeri kontrolör sonuçları (A kolu ve 80 derece referansı için)	32
4.9 PID kontrolör sonuçları (A kolu ve 10 derece referansı için).....	33
4.10 PID kontrolör sonuçları (A kolu ve 20 derece referans için)	33
4.11 PID kontrolör sonuçları (A kolu ve 50 derece referans için)	34
4.12 PID kontrolör sonuçları (A kolu ve 80 derece referans için)	35
4.13 Doğrusal olmayan PID benzeri kontrolör sonuçları (A kolu ve 10 derece referansı için)	36

	<u>Sayfa</u>
4.14 Doğrusal olmayan PID benzeri kontrolör sonuçları (A kolu ve 20 derece referansı için)	36
4.15 Doğrusal olmayan PID benzeri kontrolör sonuçları (A kolu ve 50 derece referansı için)	37
4.16 Doğrusal olmayan PID benzeri kontrolör sonuçları (A kolu ve 80 derece referansı için)	37
4.17 PZT ilave edilmiş PID benzeri kontrolör sonuçları (A kolu ve 10 derece referansı için)	38
4.18 PZT ilave edilmiş PID benzeri kontrolör sonuçları (A kolu ve 20 derece referansı için)	39
4.19 PZT ilave edilmiş PID benzeri kontrolör sonuçları (A kolu ve 50 derece referansı için)	40
4.20 PZT ilave edilmiş PID benzeri kontrolör sonuçları (A kolu ve 80 derece referansı için)	40
4.21 PID kontrolör sonuçları (B kolu ve 50 derece referans için)	41
4.22 PID kontrolör sonuçları (B kolu ve 80 derece referans için)	42
4.23 Doğrusal olmayan PID benzeri kontrolör sonuçları (B kolu ve 50 derece referansı için)	43
4.24 Doğrusal olmayan PID benzeri kontrolör sonuçları (B kolu ve 80 derece referansı için)	43
4.25 PID kontrolör sonuçları (C kolu ve 50 derece referansı için).....	44
4.26 PID kontrolör sonuçları (C kolu ve 80 derece referansı için).....	45
4.27 Doğrusal olmayan PID benzeri kontrolör sonuçları (C kolu ve 50 derece referansı için)	46
4.28 Doğrusal olmayan PID benzeri kontrolör sonuçları (C kolu ve 80 derece referansı için)	46
4.29 İki eklemlili kol için alınan PID kontrolör sonuçları	48
4.30 İki eklemlili kol için alınan doğrusal olmayan PID benzeri kontrolör sonuçları (Birinci parça 10 derece 2. parça 20 derece referansı için)	49
4.31 İki eklemlili kol için alınan doğrusal olmayan PID benzeri kontrolör sonuçları (Birinci parça 10 derece 2. parça 72 derece referansı için)	50

ÇİZELGELER DİZİNİ

	<u>Sayfa</u>
1. Deneysel çalışmada kullanılan kolların özellikleri	26

1.GİRİŞ

Hafiflik, verim, yüksek hız gibi birtakım üstün özellikleri nedeniyle esnek robot kollarının modellenmesi ve kontrolü konusunda birçok çalışma yapılmıştır. Hafif robotların yüksek hızlarda hareket ettirilmesi nedeniyle yapısal titreşimler meydana gelmektedir. Bu titreşimler birçok uygulamada verimi ve hassasiyeti etkilemektedir. Özellikle lazer teknolojisi, yarı iletken malzeme üretimi gibi yüksek hassasiyetli kontrol gerektiren çeşitli uygulamalarda bu konu giderek artan bir öneme sahiptir. Bu durum, ihmal edilen birtakım dinamik etkilerin daha çok dikkate alınmasını gerekli kılmaktadır.

Hafif bir robot kolu ağır bir yük taşıdığında kol, katı bir kol olarak görülemez. Bu durumda katı hal durumlarıyla birlikte malzemenin yapısından kaynaklanan esnek hal durumlarının da sisteme ilave edilmesi gerekmektedir.

Esnek bir robot kolunun istenilen bir referans değerine götürülmesi için gerekli kontrol uygulanırken en kısa zamanda yatışkın durum pozisyon hatasının sönümlenmesi istenir. Esnemedi kaynaklanan titreşim miktarını ölçebilmek amacıyla kolun yüzeyine yerleştirilen titreşim ölçen sensörlerden faydalanılmaktadır. Bu iş için kullanılan piezo-elektrik seramik kristaller (PZT) sayesinde titreşim bilgisi kontrolöre girdi olarak geri beslenmektedir.

1.1 Geçmişte Yapılmış Çalışmalar

Özellikle son yıllarda sanayide az enerji harcayan, yüksek hızlı robot kollarına duyulan ihtiyaç nedeniyle bu alanda yapılan çalışmalar hız kazanmış ve başarılı laboratuvar düzenekleriyle deneysel olarak da bu alandaki çalışmaların etkinliği ortaya konmuştur. Yapılan çalışmaların çoğu tek eklemlili esnek robot kolları üzerinde yoğunlaşmıştır.

Katı yapılarla ilgili ilk teoriyi Galileo ortaya koymuş [1], katı yapıların darbe-momentum ilişkisini incelemiş ve enerji dönüşümleri ile ilgili bağıntıları ortaya koymuştur. Daha sonra Newton esneklik derecesini ortaya koyan ve enerji

kayıplarını içeren çalışmalar yapmıştır [1]. İki esnek yapı arasındaki etkiyi ve gerilme analizini ilk olarak Hertz tatmin edici bir şekilde ortaya koymuştur [1]. Hertz, yapıların elastik özellikleri ve geometrilerinin bir fonksiyonu olarak bağlantı noktalarındaki gerilmeyi tanımlayarak potansiyel formda ifade etmiştir. Hertz yasası bağlantı bölgesindeki elastik etkileri içeren doğrusal olmayan integral denklemlerden oluşmaktadır. Bu model birçok yapıya geniş bir şekilde uygulanmış ve başarılı birçok deneysel sonuç elde edilmiştir. Doğrusal olmayan bu denklemlerin çözümünde nümerik yöntemler kullanılmıştır.

Timoshenko, nümerik çözüm için küçük artım yöntemini (*small increment method*) geliştirmiş ve başka yaklaşım yöntemlerinin geliştirilmesine öncülük etmiştir [1]. Bunlardan bir tanesi Zener ve Feshbach tarafından enerji yönteminin geliştirilmesi ve Lee tarafından bunun uygulanmasıdır [1]. Geliştirilen bir başka yöntem de açık çevrim kontrol yöntemidir. Salmatjidis ve Chapnik birbirlerinden bağımsız olarak tek eklemlerli esnek bir robot kolunun kontrolü problemi konusunda çalışmışlardır [1].

Sakawa vd. [2], esnek bir robot kolu için ilk olarak kısmî diferansiyel denklemler türetmiş ve titreşim ile ilgili gerekli sınır koşullarını incelemişlerdir. Daha sonra ise sensör çıkışlarını kullanarak dinamik bir kompanzator içeren geri besleme kontrol sistemi tasarlamışlardır. Esnek kol için geliştirilen kontrol stratejisini göstermek için deneysel bir düzenek kurmuşlar ve titreşim sensörü olarak gerilim ölçerler kullanmışlardır.

Yurkovich ve Pacheco [3], ucunda bilinmeyen ve değişken bir yük bulunan tek eklemlerli esnek bir robot kolunun kontrolü ve sistem tanımlamasını deneysel sonuçlarla göstermişlerdir. Rovner ve Cannon [4] kendinden ayarlı regülatör temeline dayalı bir adaptif kontrol algoritmasını oldukça esnek tek eklemlerli bir robot koluna uygulayarak deneysel olarak incelemişlerdir.

Luo ve Guo [5], esnek kolun titreşimini tatmin edici biçimde sönümlendirebilecek doğrudan-gerilme geri beslemesini açıkça ortaya koyabilecek bir kontrol problemini incelemişlerdir. Burada uygun sınır koşullarında doğrusal kısmi diferansiyel denklemlerle tek eklemlerli esnek robot kolunun dinamik modelleri üzerinde durulmuştur.

Doğan [6], ilk parçası katı, ikincisi ise esnek bir yapı olan ve uç kısmında kütlesi bilinmeyen ancak belirli bir aralıkta değiştiği varsayılan bir yüke sahip iki parçalı bir robot kolunun modellenmesi ve kontrolünü ele almış, sistemin dinamik denklemlerini Hamilton prensibini kullanarak türetmiş ve sistemin dinamiğini diferansiyel özdeğer problemi yaklaşımı kullanarak sonsuz boyutlu bir modelle göstermiştir. Robot kolunun eklemlerinin her bir eklemler için seçilen sabit bir referans sinyali veya bir yörüngeyi takip etmesi ve esnek parçanın uç sapmasının sönümlenmesi için PD kontrolörler ve PD benzeri kontrolörler ile tekil pertürbasyon metodunu temel alan iki zaman ölçekli bir kontrolör tasarlamıştır.

Wei [1], esnek yapılar üzerine darbe etkisinin dinamik modelini incelemiş, Hertz yasası kullanılarak elde edilen doğrusal olmayan integral denklemler için nümerik yöntemler ortaya koymuş ve elde ettiği sonuçları diğer yöntemlerle karşılaştırmıştır.

Kantor [7], piezo-elektrik seramik kristaller (PZT) kullanarak esnek bir koldaki aktif titreşim sönümlenme problemini incelemiştir. Bu amaçla kapalı döngü kontrol uygulanıp değişik laboratuvarlarda deneysel sonuçlar elde edilmiştir. Esnek bir kol ile üzerine yerleştirilen bir küre arasındaki darbe etkisi incelenmiştir.

Bai vd. [8], iki eklemlerli esnek bir robot kolu için adaptif durum geri besleme kontrolü yaklaşımı uygulamışlardır. Adaptif bir kompanzator ile kararlı durum doğrusal kuadratik regülatör tekniğini temel alan bir kontrolör tasarlanmıştır. Geliştirilen kontrol yöntemi deneysel olarak uygulanmış ve kontrolörün oldukça tatmin edici olduğu gösterilmiştir.

Salmatjidis [9] yaptığı çalışmada esnek robot kollarının kuvvet kontrolü problemini incelemiş, Newmark integrasyon metodu kullanarak nümerik olarak kolun serbest ve sınırlı hareketini tahmin etmiş, bulduğu nümerik sonuçları deneysel sonuçlarla karşılaştırmıştır.

Hastings ve Book [10] tek eklemlerli esnek bir robot kolu için doğrusal durum uzayı modelini geliştirmiş ve simülasyon sonuçlarını incelemiştir. Feliu vd. [11] ise tek eklemlerli hafif ve esnek bir robot kolu için bir adaptif kontrol metodu geliştirmiştir.

1.2 Tezde Yapılan Çalışmalar

Bu çalışmada öncelikle tek eklemlili esnek bir robot kolunun kontrolü laboratuvarında deneysel olarak incelenmiştir. Sistemin kontrolü, doğrusal olmayan PD ve PID benzeri kontrol yöntemleri, PD ve PID kontrol yöntemleriyle karşılaştırılmış, elde edilen sonuçlar tartışılmıştır. Kolun titreşim bilgisi, üzerine monte edilen piezo-elektrik sensörlerle, pozisyon bilgisi optik kodlayıcılarla ve hız bilgisi de motor miline bağlı takometre ile ölçülüp uygulanan kontrol yöntemiyle kolun en kısa zamanda istenilen pozisyon değerine gitmesi sağlanmaya çalışılmıştır. Farklı malzemelerle farklı ağırlıklardaki kolların tepkileri incelenmiştir.

Çalışmanın sonraki aşamasında kola ikinci bir parça ilave edilip iki eklemlili esnek robot kolunun kontrolü incelenmiştir. Bu amaçla birinci parça olarak çelik levha ikinci parça olarak alüminyum levha kullanılmıştır. Kontrolör olarak PID ve doğrusal olmayan PID benzeri kontrolör kullanılmış, farklı değerler için elde edilen sonuçlar karşılaştırılmıştır.

2. DENEYSEL DÜZENEGİN HAZIRLANMASI

2.1 Tek Eklemlı Esnek Robot Kolu Düzenegının Hazırlanması

Bu bölümde öncelikle tek eklemlı esnek bir robot kolu için deneysel düzenegın hazırlanması incelenmiştir. Esnek kollu robot sistemi, bir ucundan motorun miline sabitlenmiş hafif ve esnek levhadan oluşur. Bu kol tabana yerleştirilmiş bir DC motor yardımıyla hareket ettirilir. DC motorun kontrolü için ise bir servo yükseltici kullanılmaktadır. Bu yükselticinin kontrolü bilgisayar tarafından PCL 1800 kartı yardımıyla [-5V,+5V] aralığında yapılmaktadır. Her örnekleme anında kolun açısıl pozisyonunu ölçmek için DC motorun miline bir optik kodlayıcı yerleştirilmiştir. Optik kodlayıcının ürettiğı 12 bitlik pozisyon bilgisi PCL 1800 veri toplama kartı tarafından alınmaktadır. Ayrıca motor üzerine yerleştirilmiş bulunan takojeneratör tarafından üretilen hız bilgisi kart üzerindeki analog girişler vasıtasıyla okunmaktadır. Kolun esnemesinden kaynaklanan titreşimleri ölçmek amacıyla kolun ucuna piezo-elektrik seramik kristaller (PZT) yerleştirilmiştir. PZT çıkışı kart tarafından okunmadan önce kuvvetlendirilip alçak geçiren filtreden geçirilmiştir. Tüm bu sensörlerden elde edilen bilgiler toplanıp kontrolöre geri beslendikten sonra uygun kontrol sinyali yine aynı kart vasıtasıyla sürücüye gönderilmektedir. Sürücü olarak SSA-8/100 kartı kullanılmış, kol olarak alüminyum ve çelik levhalar kullanılmıştır. Tek eklemlı esnek robot kolu için hazırlanan düzenek Şekil 2.1’de gösterildiğı gibidir. Tüm sistemin blok şeması ise Şekil 2.2’de verilmiştir.

2.1.1 Düzenekte kullanılan malzemeler

Tek eklemlı esnek robot kolu için taban oluşturan motorun seçimi yapılırken kola ikinci bir eklemin daha ilave edileceğı dikkate alınarak torkunun yüksek olması istenmiştir. Bu amaçla Bautz firmasının üretmiş olduğı E726MGB model DC servomotor tercih edilmiştir. Motorun nominal değerleri şu şekildedir; gücü 550 watt, torku ise 2.1 Nm. boştaki hızı 2500 rpm. çektiğı akım 6 amper,

ağırlığı ise 6.5 kg.'dır. Motor miline eklenen takojeneratör voltaj sabiti $14,0 \text{ V}/1000 \text{ min}^{-1}$ dir. Kolun pozisyonunu ölçmek için kullanılan kodlayıcı 12 bit çözünürlükte gray kod üreten TTL çıkışlı bir kodlayıcıdır. Kodlayıcı motor miline, uygun kaplinler kullanılarak monte edilmiştir. Motor, takojeneratör ve kodlayıcı bloğu çelikten yapılmış bir sehpa üzerine yerleştirilmiştir. Esnek robot kolu, motorun miline uygun aparatlarla sabitlenmiş farklı uzunluk ve ağırlıktaki levhalardan oluşturulmuştur. Çelik ve alüminyum kollar için çeşitli kontrol yöntemleri uygulanarak kolun en kısa zamanda istenilen pozisyon değerine gitmesi sağlanmıştır.

Sistemde kontrolör olarak bilgisayar kullanılmıştır. Bilgisayara bağlı olarak PCL-1800 veri toplama kartı bulunmaktadır. Bu kartın çıkışında elde edilen kontrol sinyali motorun her iki hareket yönünde kontrolünü sağlamak için bir fark alıcı devreden geçirilmiştir. Devre çıkışı sürücüye uygulanarak motorun istenilen kontrol bilgisine göre hareketi sağlanmıştır. Motor üzerinde sabitlenmiş bulunan kodlayıcı ve takometreden pozisyon ve hız bilgileri alınmıştır.

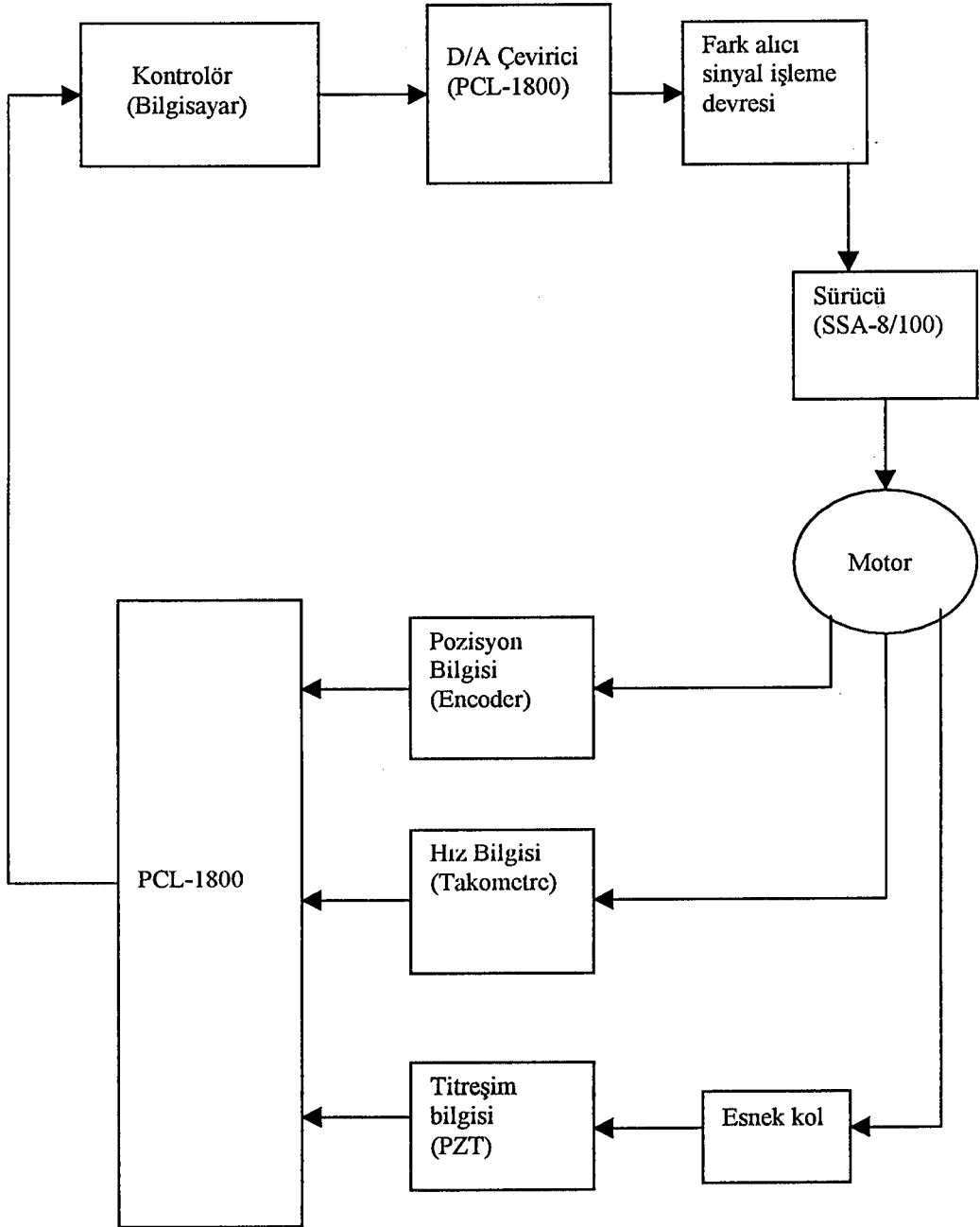
Esnek kolun esnemesi sonucu oluşan titreşim bilgisini sisteme geri besleyebilmek için piezoelektrik seramik kristaller kullanılmıştır. Kullanılan sensör *cymbal* olarak literatüre girmiştir. Kolun titreşimiyle birlikte titreşim şiddetiyle orantılı olarak gerilim üretmektedir. Kullanılan PZT sensör titreşimin en fazla hissedildiği uç noktasına yerleştirilmiştir. Elde edilen sinyal Bölüm 2.3'te incelenen sinyal işleme devresinden geçirilerek veri toplama kartları vasıtasıyla kontrolöre geri beslenmiştir.

Elde edilen pozisyon, hız ve titreşim bilgilerinin kontrolöre geri besleme sinyali olarak uygulanabilmesi için bilgisayara veri toplama kartları yerleştirilmiştir. Bu amaç için kullanılan veri toplama kartı PCL-1800 kartıdır. PCL-1800 veri toplama kartı gerekli olan analog-dijital ve dijital-analog çevirme işlemlerini yapmakta, gerekli geri besleme bilgilerini kontrolöre iletmekte ve kolun kontrolü için üretilen kontrol sinyalini motor sürücü düzeneklerine göndermektedir. Kartın üzerinde 16 bit TTL dijital girdi, 0-10 volt aralığında 16 analog girdi ve 0-10 volt aralığında analog çıkış pinleri vardır.

PCL-1800 veri toplama kartının analog çıkışı ancak 0-10 volt aralığında çalışmaya izin vermektedir. Bu durumda motoru iki yönde hareket ettirebilmek



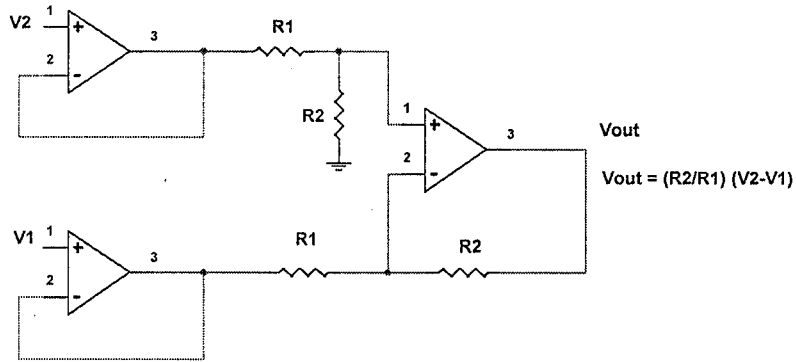
Şekil 2.1 Tek eklemlı esnek robot kolu dñzeneđi



Şekil 2.2 Tek Eklemlı Esnek Robot Kolu Düzenegının Şematik Gösterimi

mümkün olmamaktadır. Bu amaçla kartın çıkışına uygun sinyal işleme düzenekleri kurulmuş, kontrol programı bu özellik dikkate alınarak yazılmıştır. Kurulan devre Şekil 2.3'te gösterildiği gibidir. Burada karttan gelen 0-10 volt aralığındaki kontrol sinyali bir fark alıcı devreden geçirilerek +5V -5V aralığına çekilmiştir. Kurulan devrenin kazancı 1 olacak şekilde ayarlanmıştır.

Sistemin kontrolü için C programlama dili kullanılmış, kullanılan programlar eklerde gösterilmiştir. Programlarda kodlayıcının ürettiği gray kod gerçek açı değerine çevrilip elde edilen pozisyon bilgisi kontrolöre geri beslenmiş, veri toplama kartından okunan hız ve esneme bilgileriyle esnek kolun kontrolü için gerekli kontrol sinyali üretilip aynı kart üzerinden motorun sürülmesi sağlanmıştır.



Şekil 2.3 Fark alıcı sinyal işleme devresi.

Kullanılan devre elemanları:

$R_1 = R_2 = 10K \Omega$, Op-amp: LM741.

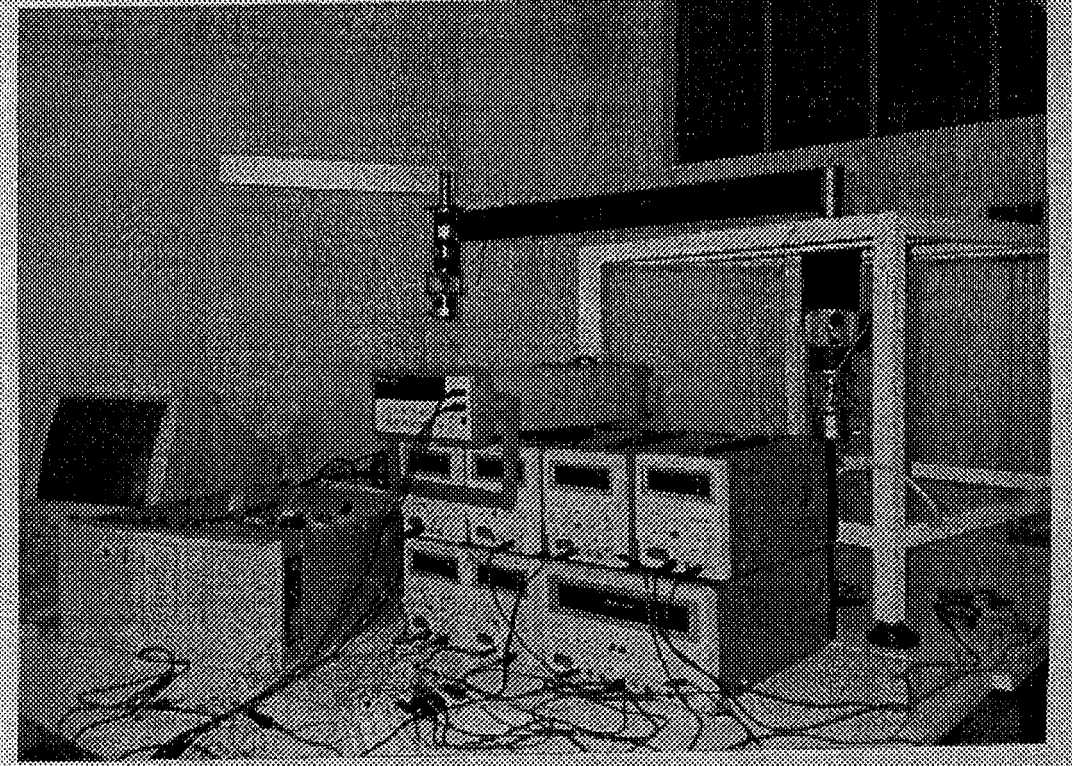
$V_1 = +5V$, $V_2 =$ Kontrol sinyali.

2.2 İki Eklemlili Esnek Robot Kolu Düzenegi

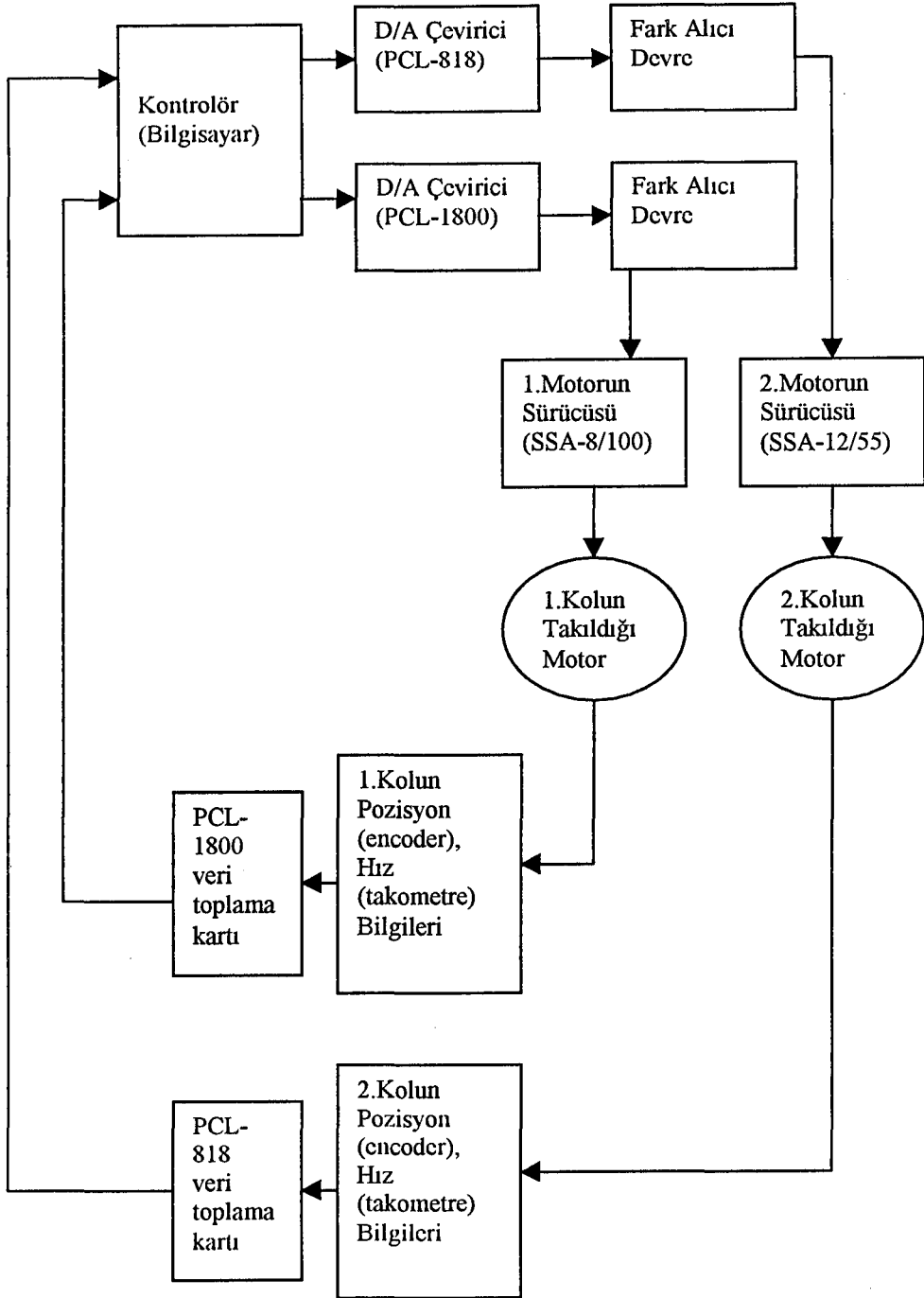
Tek eklemlili esnek robot kolu düzeneginin hazirlanmasinin ardından iki eklemlili esnek robot kolu için gerekli mekanik düzenek hazirlanmiş ve iki eklemlili esnek robot kolunun kontrolü için gerekli kontrolörler tasarlanip sisteme uygulanmıştır. Elde edilen sonuçlar ortaya konulmuş ve tartışılmıştır. Bu bölümde iki eklemlili esnek robot kolu için hazirlanılan deneysel düzenek anlatılacaktır.

İki eklemlili esnek robot kolu düzenegi Şekil 2.4'te gösterilmiştir. Ayrıca tüm düzenegin blok şeması da Şekil 2.5'te verilmiştir. Burada tek eklemlili esnek robot kolu düzeneginde olduğu gibi yine kontrolör olarak bilgisayar kullanılmıştır. İkinci eklemlili ilgili verilerin toplanıp gerekli kontrol sinyallerinin motora iletilebilmesi için ikinci bir veri toplama kartı sisteme ilave edilmiştir. Bu amaçla PCL-818 veri toplama kartı kullanılmıştır. Birinci eklem için ise PCL-1800 veri toplama kartı kullanılmıştır. Her iki kartın çıkışı da iki yönlü pozisyon kontrolü yapabilmek amacıyla Şekil 2.3'te gösterilen fark alıcı devreden geçirilmiştir. Elde edilen +5V -5V aralığındaki kontrol sinyalleri her iki motor için uygun motor sürücü devrelerine gönderilmiştir. Her iki kol için de pozisyon ve hız bilgileri yine aynı veri toplama kartları yardımıyla alınıp kontrolöre geri beslenmiştir.

İki eklemlili esnek robot kolu sistemi için hazirlanan düzenekte temel olarak tek eklemlili esnek robot kolu için hazirlanılan düzenek kullanılmıştır. Tabana yerleştirilen ve birinci eklemi hareket ettirecek olan motorun miline yerleştirilen çelik kolun ucuna, ikinci eklemi hareket ettirecek olan motor monte edilmiştir. Bu ikinci motorun da, birincisinde olduğu gibi miline bağlı bir optik kodlayıcı ve takojeneratör bulunmaktadır. Birinci motor Bölüm 2.1'de anlatılan tek eklemlili esnek robot kolu için kullanılan motorun aynısıdır. Motorun tüm özellikleri bu bölümde anlatılmıştır. İkinci motor da birinci motor gibi Bautz firmasından alınmıştır. Modeli E588A-MGC-8, gücü 140 watt torku 0.28 Nm, boştaki hızı 5000 rpm. çektiğı akım 2.8 amperdir. Motor miline takılı bulunan takojeneratörün voltaj sabiti 14.0 V/1000 min⁻¹ dir. İki eklemlili esnek robot kolu düzeneginde kullanılan her iki motorun da miline pozisyon değerini ölçmek amacıyla yerleştirilen optik kodlayıcılar Hohner firması tarafından üretilmiştir. Kullanılan MS633A3/4096 model nolu kodlayıcı 12 bit çözünürlükte gray kod üretmektedir.



Şekil 2.4 İki eklemli esnek robot kolu düzenneđi



Şekil 2.5 İki Eklemlili Esnek Robot Kolumun Şematik Gösterimi

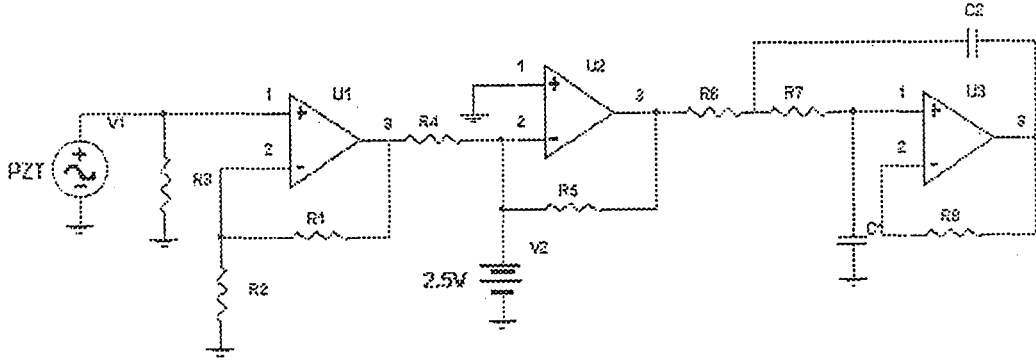
2.3 Piezoelektrik Seramik Sensörler (PZT)

Robotik uygulamaları için son yıllarda tasarlanan ve gerçekleştirilen yeni sensörler robot kollarının performansını geliştirmiş ve aynı zamanda endüstriyel süreçlerde yüksek hassasiyet göstermişlerdir. Bu sınıf sensörlerdeki en kritik bölüm analog süreç devresi ve mekanik hareketten dolayı meydana gelen küçük sinyalleri belirgin bir hale getiren elektriksel yük kuvvetlendiricileridir.

2.3.1 Piezoelektrik sensörler için geliştirilen yük kuvvetlendiricileri

Piezoelektrik aktif yüzeyi temeline dayanan sensörler, yüksek hassasiyetli elektriksel yük kuvvetlendirici devrelerine gereksinim duyarlar. Bu yolla, hareket ettirilen nesnenin mekanik hareketi sonucunda oluşan bilgilerin en iyi şekilde değerlendirilmesi için kompleks, gelişmiş analog ve ayrıca dijital sinyal süreç devreleri geliştirilmiştir. Algılayıcının algılama alanı seramik malzemeye yerleştirilen piezoelektrik film kullanılarak gerçekleştirilmiştir. Analog ve dijital sinyal işleme devreleri gürültüyü ve çevreden kaynaklanan parazitik etkileri azaltmak için aktif yüzeyin yakınlarına kurulmuştur. Şekil 2.6'da bu amaç için geliştirilmiş bir tasarım örneği gösterilmektedir.

İlk bölümdeki kuvvetlendirici yardımıyla öncelikle sensörden elde edilen düşük değerdeki bilginin belli bir büyüklüğe ulaşması sağlanmış ardından ofset ayarı için gerekli ikinci blok yerleştirilmiş ve son olarak da oluşan gürültüyü ve parazitik etkileri ortadan kaldırmak amacıyla alçak geçiren filtre ilave edilmiştir. Böylece esnek robot kolunun titreşimini ölçmek için elde edilen sinyal bilgisayarın veri toplama kartı vasıtasıyla okunabilecek hale getirilmiştir.



Şekil 2.6 PZT için kullanılan sinyal işleme devresi.

Kullanılan devre elemanları:

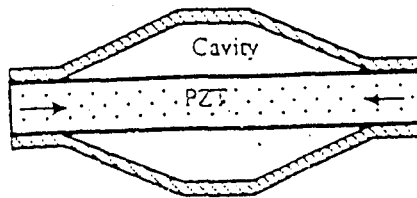
$R1=100K \Omega$ $R2=100 \Omega$ $R3=4.7K \Omega$ $R4=10K \Omega$

$R5=10K \Omega$ $R6=10K \Omega$ $R7=10K \Omega$ $R8=22K \Omega$

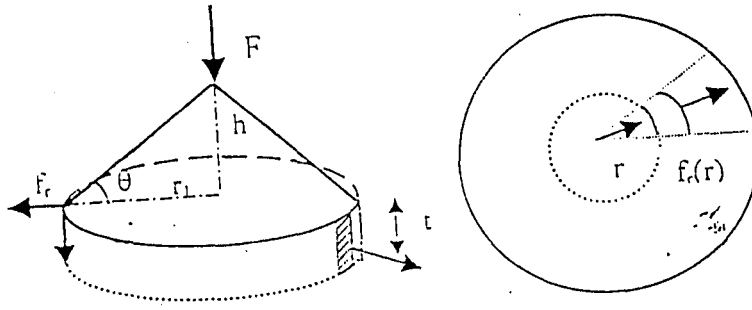
$C1=2.2UF$ $C2=20nF$

2.3.2 Piezoelektrik seramik sensörün (PZT) modellenmesi

Cymbal tipi sensörler iç yüzeyinin çukuru fazla derin olmayan iki metal levha arasına sıkıştırılmış elektro-aktif seramik sürücü elemanından oluşmaktadır. Sensörün yapısı Şekil 2.7’de gösterilmektedir. *Cymbal* sensörün dış kaplamasında çukurlar koni şeklinde kısaltılmış haldedir. Bu çukurlukların amacı metal yüzeylerin anlık eksenler yönündeki gerilmeleri zıt işaretli teğetsel ve açısallı gerilmelere çevirmek ve yükseltmektir. Ters etkiye gelince, seramik sürücü elemanın açısallı hareketi metal dış kaplamalar tarafından eksenler yönünde çevrilerek yükseltilirler. Böylece malzeme sensör olarak kullanılabilir [12].



Şekil 2.7 PZT sensörün yapısı



Şekil 2.8 Hesaplamalarda kullanılacak parametreler

PZT malzemeyi oluşturan kristaller malzeme içerisinde dağınık bir şekilde kutuplanmışlardır. Bu durumda üretilen elektriksel yük malzeme içerisinde dağılarak belli bir gerilime dönüşmeyecektir. Bunu önlemek amacıyla PZT kristallere belli bir gerilim uygulanarak istenilen yönde kutuplanması sağlanmaktadır. Bunun sonucunda PZT malzemeye uygulanan kuvvete bağlı olarak kristaller aynı yönde gerilim üretirler.

PZT üzerindeki metal kılıfa uygulanan etki sonucu yaratılan elektriksel yük şu şekilde ifade edilebilir [12];

$$Q = d \frac{r_1}{t} \frac{F}{\tan \theta_p}$$

Burada F Şekil 2.8'de gösterildiği gibi PZT malzemeye etkiyen kuvvet, d PZT sabiti, t PZT malzemenin kalınlığı, r_1 PZT malzemenin yarı çapı, ve θ_p ise PZT malzeme ile konik kılıf arasındaki açıdır. PZT malzemenin kapasite değeri ise

$$C = \frac{\pi r_1^2}{t} \epsilon$$

şeklindedir. Bu durumda PZT sensörden elde edilecek gerilim değeri

$$V_p = \frac{Q}{C} = d \frac{1}{\epsilon \pi r_1} \frac{F}{\tan \theta_p}$$

şeklinde bulunur. Burada d PZT sabiti, Q üretilen elektriksel yük, C kapasitans ve ε yalıtkanlık sabitidir. Kolun uç noktasına yerleştirilen sensöre etkiyen kuvvet,

$$F = m_p a$$

$$a = L \ddot{\theta} + Z_u(L, t)$$

$$F = L m_p \ddot{\theta} + m_p Z_u(L, t)$$

şeklinde elde edilir. Burada m_p PZT kılıfın kütlesi, $\ddot{\theta}$ eklemin açısal ivmesini, $Z(L, t)$ esnek kolun uç noktasının katı hal pozisyonundan olan dik sapmasını, $Z_u(L, t)$ ise $Z(L, t)$ 'nin zamana göre ikinci türevini (ivmeyi) vermektedir. Bu durumda PZT tarafından üretilen gerilim ifadesi,

$$V_p = d \frac{1}{\varepsilon \pi r_1} \frac{(L m_p \ddot{\theta} + m_p Z_u(L, t))}{\tan \theta_p}$$

şeklinde bulunur. Bu ifade PZT tarafından üretilen gerilim ile kolun ivmesi arasındaki ilişkiyi vermektedir.

2.4 Programlama

Hazırlanan tek ve iki eklemlili esnek robot kolu düzeneklerinin kontrolü için C programlama dili kullanılmıştır. Bu amaçla hazırlanan programlar eklerde sunulmuştur. Kullanılan programların işleyişi şu şekildedir:

Ek 1'de verilen tek eklemlili esnek robot kolu için geliştirilen PID kontrol programının Şekil 2.9'daki akış şemasına göre işleyişi şu şekildedir. Program çalıştırıldığında ekrana kullanıcının referans değerini girmesini isteyen satır gelir. İstenilen referans değeri yazılıp giriş tuşuna basıldıktan sonra öncelikle kodlayıcıdan okunacak olan gray kodlu pozisyon bilgisini ondalıklı sayıya çevirmek için gerekli olan "init()" isimli alt programın yüklenmesi sağlanır. Ardından PCL-1800 veri toplama kartının bilgisayara tanıtılması amacıyla "Init 1800" alt programı çalıştırılır. PCL-1800 veri toplama kartının bilgisayara tanıtılmasından sonra pozisyon, pozisyon hatası ve kontrol sinyali verilerinin

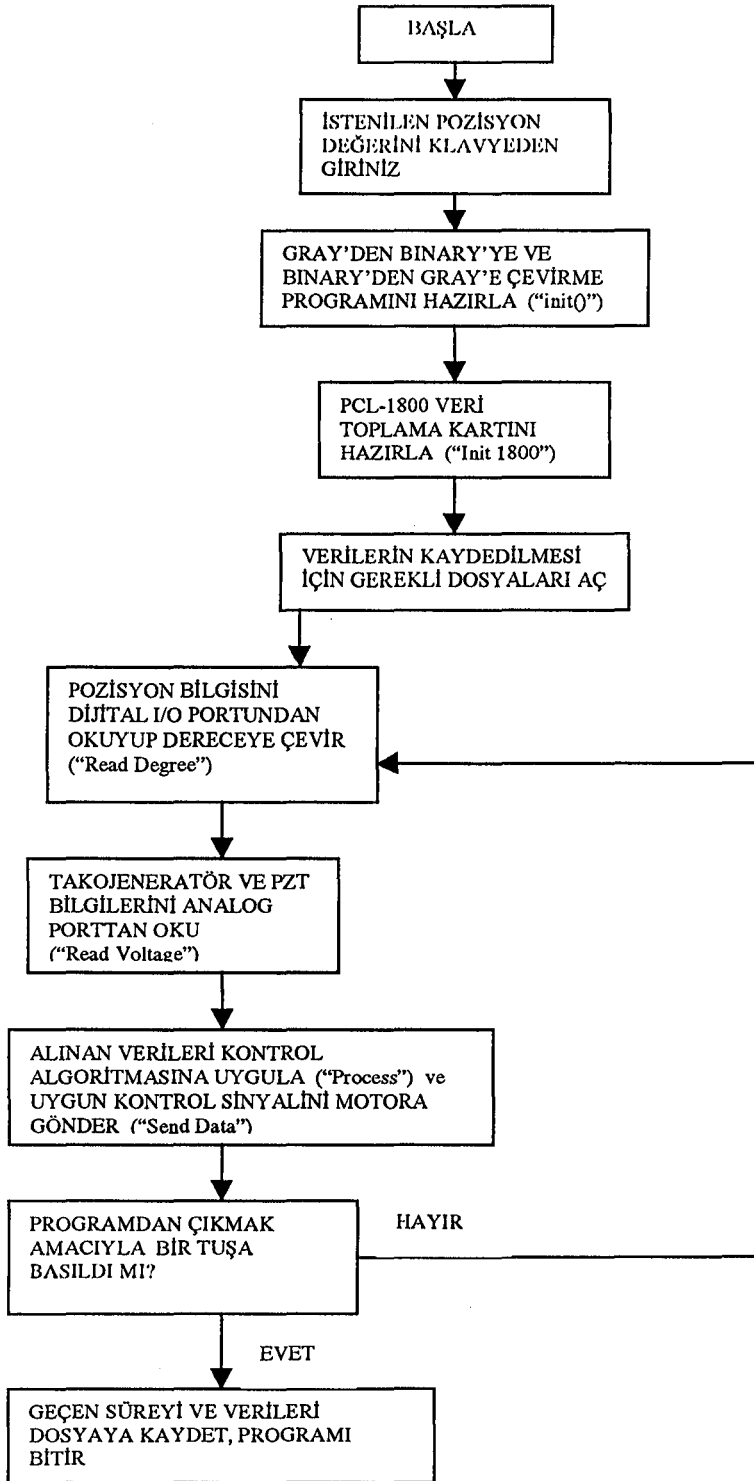
kaydedilmesi için “.DAT” uzantılı dosyaların açılması sağlanır. Daha sonra pozisyon, hız ve PZT bilgilerinin okunması işlemine geçilir. Bu amaçla öncelikle eski verilerin silinmesi için başlangıç değerleri sıfırlanır. Ardından pozisyon bilgisini okumak için “Read Degree” alt programı çalıştırılarak dijital I/O portundan okunan gray kodlu pozisyon bilgisi “Gray to Dec” fonksiyonu ile ondalıklı sayıya daha sonra da dereceye çevirilir. Ardından da takojeneratör ve PZT’den alınan analog veriler analog I/O portundan okunur. Alınan veriler “Process” alt programında uygun kontrol algoritmasına uygulanarak elde edilen kontrol bilgisi “Send Data” alt programı kullanılarak veri toplama kartı vasıtasıyla sürücüye, oradan da motora iletilir. Bu aşamada programdan çıkılıp çıkılmadığını anlamak amacıyla bir tuşa basılıp basılmadığı kontrol edilir. Eğer programdan çıkmak amacıyla bir tuşa basılmışsa verilerin kaydedilme işlemi tamamlanarak geçen süre hesaplatılır ve programdan çıkılır. Eğer bir tuşa basılmamışsa veriler okunarak kontrolöre gönderilmeye devam edilir. Bu işleme, bir tuşa basılarak programdan çıkılana kadar devam edilir.

Ek 2’de verilen tek eklemlili esnek robot kolu için uygulanan doğrusal olmayan PID kontrol için geliştirilen programda ise yukarıdakinden farklı olarak kontrolör kısmı değiştirilmiştir. Kontrolör kısmında “Process” alt programında değişiklik yapılmıştır. Geri kalan kısım ilk programla aynıdır.

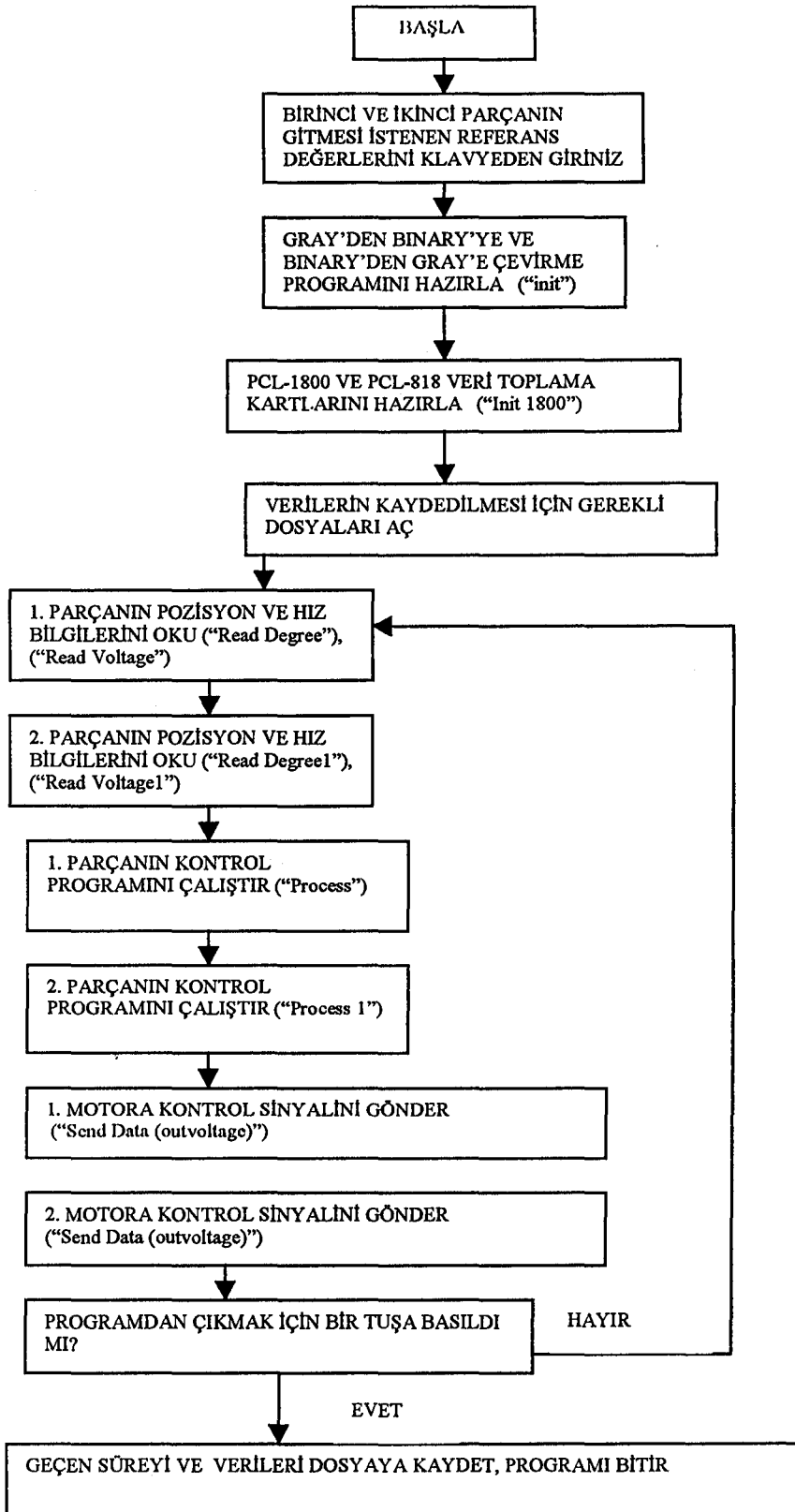
Ek 3’teki iki eklemlili esnek robot kolu kontrolü için geliştirilen kontrol programının Şekil 2.10’da verilen akış şemasına göre işleyişi şu şekildedir. Programın çalıştırılmasıyla birlikte klavyeden, birinci ve ikinci parça için istenilen referans değerleri girilir. Referans değerleri girilip giriş tuşuna basıldıktan sonra kodlayıcılardan gelecek olan gray kodlu pozisyon bilgilerini ondalıklı sayıya çevirmek amacıyla kullanılacak olan “init()” isimli alt programın çalışması sağlanır. Ardından “Init 1800” alt programında birinci parça için kullanılacak olan PCL-1800 veri toplama kartının ve ikinci parça için kullanılacak olan PCL-818 veri toplama kartlarının bilgisayara tanıtılması sağlanmaktadır. Ardından verilerin kaydedilmesi için gerekli dosyalar açılır. Verilerin başlangıç değerleri sıfırlandıktan sonra ise “Read Degree” ve “Read Degree1” alt programları vasıtasıyla her iki parça pozisyon bilgileri okunur ve okunan bilgiler gray koddan dereceye çevirilir. Ardından da takojeneratör bilgileri okunarak alınan veriler

“Process” ve “Process1” alt programları vasıtasıyla uygun kontrol algoritmalarında kullanılır. Elde edilen kontrol sinyalleri ise “Send Data(outvoltage)” ve “Send Data(outvoltage1)” alt programları aracılığıyla önce her iki motorun sürücülerine oradan da motorlara iletilir.

Daha sonra ise programı sonlandırmak amacıyla herhangi bir tuşa basılıp basılmadığı kontrol edilir, eğer basılmışsa elde edilen verilerin kaydedildiği dosyalar kapatılır, geçen süre hesaplanır ve programdan çıkılır. Eğer herhangi bir tuşa basılmamışsa veriler okunup kontrolöre iletmeye devam edilir. Bu işlem herhangi bir tuşa basılıp programdan çıkılıncaya kadar devam eder.



Şekil 2.9 Tek Eklemli Esnek Robot Kolu İçin Geliştirilen Program Akış Şeması



Şekil 2.10 İki Eklemlı Esnek Robot Kolu İçin Geliştirilen Program Akış Şeması

3. KONTROLÖR TASARIMI

Bu bölümde öncelikle tek eklemlili esnek robot kolu için kullanılan ve kolun uç noktası sapmasını en kısa zamanda sönümlendirecek şekilde tasarlanan kontrol yöntemleri ele alınacaktır.

3.1 Tek Eklemlili Esnek Robot Kolları İçin Uygulanan Kontrol Yöntemleri

Tek eklemlili esnek robot kolu için öncelikle PD kontrol yöntemi denenmiş, daha sonra PID, doğrusal olmayan PD benzeri ve doğrusal olmayan PID benzeri kontrol yaklaşımlarından elde edilen sonuçlar karşılaştırılmıştır. Kolun titreşimlerini ölçmek için kullanılan piezoelektrik sensörden alınan veriler de geri besleme sinyali olarak kontrolöre ilave edilmiş ve elde edilen sonuçlar değerlendirilmiştir.

3.1.1 PD kontrolör tasarımı

Tek eklemlili esnek robot kolu için öncelikle PD kontrol yöntemi denenmiştir. Sürekli zaman sistemleri için bir PD kontrolör şu şekilde ifade edilebilir:

$$u(t) = K_p e(t) + K_v \dot{e}(t) \quad (1)$$

Burada u sürücüye uygulanan gerilim bilgisini, K_p ve K_v değerleri oransal ve türevsel kazanç sabitlerini ifade etmektedir; $e(t) = \hat{\theta}(t) - \theta(t)$, kolun takip etmesi gereken sabit bir referans sinyali veya bir yörünge sinyali $\hat{\theta}$ ile kolun açısı θ arasındaki farktır. Eğer kol sabit bir referans değerine gidiyorsa

$$\dot{\hat{\theta}}(t) = 0, \quad \dot{e}(t) = -\dot{\theta}(t)$$

olacaktır. K_p ve K_v değerleri esnek kolun uç sapmasını en kısa zamanda sönümleyip sistemi istenilen referans değerine götürecektir. Deneme yanılma yöntemiyle belirlenmiştir. K_p 'nin birimi volt/derece, K_v 'nin birimi volt.sn/radyan'dır.

3.1.2 PID kontrolör tasarımı

İkinci aşamada PD kontrolöre integratör ilave edilerek PID kontrolör uygulanmıştır. Sürekli zamanda PID kontrolör:

$$u(t) = K_p e(t) + K_v \dot{e}(t) + K_i \int e(t) dt \quad (2)$$

şeklinde ifade edilebilir. Burada K_i integratör sabitidir ve birimi volt/derece.sn cinsindedir.

3.1.3 Doğrusal olmayan PD benzeri kontrolör tasarımı

PD ve PID kontrolörlerde hata teriminin mutlak değeri başlangıçta en büyük değerdedir. Hata sinyali yüksek kazanç değerleriyle geri beslendiğinde başlangıç anında motora çok yüksek bir kalkınma torku uygulanacağından hem harcanan enerji çok yüksek olacak, hem de yüksek bir ivmeyle harekete başlayan esnek kolun başlangıç anındaki uç sapması çok yüksek değerlere ulaşacaktır. Başlangıç anında motora çok yüksek torklar uygulamamak için kontrolörün kazanç değerleri hata teriminin mutlak değeriyle orantılı olarak değiştirilmektedir. Bu amaçla, e hata terimi başlangıçta daha küçük bir kazanç değeriyle geri beslenirken, ani hızlanmayı önlemek için $\dot{\theta}$ terimi daha yüksek bir kazanç değeriyle geri beslenmektedir. Hata terimi küçüldükçe referans değerine daha hızlı yaklaşabilmek amacıyla e hata terimi daha yüksek bir katsayıyla çarpılmakta, $\dot{\theta}$ terimi ise sifıra hızlı yaklaşımı sağlamak amacıyla daha düşük bir kazanç sabitiyle çarpılmaktadır. Bu kontrolör şu şekilde ifade edilebilir [6]:

$$u(t) = (K_{p1} - K_{p2} s(e(t))) e(t) - (K_{v1} + K_{v2} s(e(t))) \dot{\theta}(t) \quad (3)$$

Burada,

$$s(e(t)) = \begin{cases} |e(t)|, & |e(t)| \leq a \\ a, & |e(t)| > a \end{cases} \quad (4)$$

şeklinde tanımlanmıştır. K_{p1} , K_{p2} , K_{v1} , K_{v2} kazanç değerleri deneme yanılma yöntemiyle bulunmuştur. a , $K_{p1} > K_{p2}$ $s(e(t))$ koşulunu sağlayacak biçimde seçilmelidir.

3.1.4 Doğrusal olmayan PID benzeri kontrolör tasarımı

Bu bölümde doğrusal olmayan PD benzeri kontrolöre integratör ilave edilmiş ve sistemin tepkisi incelenmiştir. Uygulanan kontrolör şu şekilde ifade edilebilir.

$$u(t) = (K_{p1} - K_{p2} s(e(t))) e(t) - (K_{v1} + K_{v2} s(e(t))) \dot{\theta}(t) + K_i \int e(t) dt \quad (5)$$

Burada,

$$s(e(t)) = \begin{cases} |e(t)|, & |e(t)| \leq a \\ a, & |e(t)| > a \end{cases}$$

şeklinde tanımlanmıştır.

3.1.5 Titreşim bilgisinin kontrolöre ilave edilmesi

Şimdiye kadar uygulanan kontrol yöntemlerinde kolun esnemesiyle ilgili bilgi doğrudan ölçülüp geri beslenmemekte idi. Bu bölümde piezoelektrik sensörlerden alınan titreşim bilgisi kontrolöre geri beslenerek esnemeyi de içeren

yeni bir kontrol yaklaşımı denenmektedir. Titreşim sensöründen gelen bilgi doğrusal olmayan PID kontrolöre oransal olarak ilave edilmekte ve esnemeyi de içeren daha gerçekçi bir kontrol yöntemi uygulanmaktadır. Kontrolörün ifadesi şu şekildedir:

$$u(t) = (K_{p1} - K_{p2} s(e(t))) e(t) - (K_{v1} + K_{v2} s(e(t))) \dot{\theta}(t) + K_i \int e(t) dt + K_d V_p \quad (6)$$

Burada V_p PZT tarafından üretilen gerilimi göstermektedir. K_d ise birimsiz kazanç sabitidir.

3.2 İki Eklemlı Esnek Robot Kolu İçin Uygulanan Kontrol Yöntemleri

Bu bölümde birinci parçası 110 cm. uzunluğunda çelik bir koldan, ikinci parçası ise 57 cm. uzunluğunda alüminyum levhadan oluşan iki eklemlı esnek robot kolu düzeneğinin çeşitli yöntemlerle kontrolü incelenmiştir. Bu amaçla hazırlanan düzenekte (Bölüm 2.2'de anlatıldığı gibi) birinci kol için 550 watt gücünde, ikinci kol içinse 140 watt gücünde DC servo motor kullanılmıştır.

3.2.1 PID kontrolör tasarımı

İki eklemlı esnek robot kolu düzeneğine öncelikle PID kontrol yöntemi uygulanmış elde edilen sonuçlar dördüncü bölümde gösterilmiştir. Sürekli zamanda PID kontrolör:

$$u_j(t) = K_{pj} e_j(t) + K_{vj} \dot{e}_j(t) + K_{ij} \int e_j(t) dt \quad j=1,2 \quad (7)$$

şeklinde ifade edilebilir.

3.2.2 Doğrusal olmayan PID kontrolör tasarımı

İki eklemlilik esnek robot kolu kontrolü için her iki ekleme de doğrusal olmayan PID kontrol algoritması uygulanmış elde edilen sonuçlar dördüncü bölümde gösterilmiştir.

Uygulanan kontrolün sürekli zamanda ifadesi

$$u_j(t) = (K_{p1j} - K_{p2j} s(e_j(t))) e_j(t) - (K_{v1j} + K_{v2j} s(e_j(t))) \dot{\theta}_j(t) + K_{ij} \int e(t) dt \quad j=1,2 \quad (8)$$

şeklindedir. Burada,

$$s(e_j(t)) = \begin{cases} |e(t)|, & |e(t)| \leq a \\ a, & |e(t)| > a \end{cases}$$

şeklindedir.

4. DENEYSEL SONUÇLAR

4.1 Tek Eklemli Esnek Robot Kolları İçin Alınan Deneysel Sonuçlar

Tek eklemli esnek robot kolu kontrolü için yapılan çalışmada üç farklı kol kullanılmış ve herbiri için elde edilen deneysel sonuçlar incelenmiştir. Kullanılan birinci kol A kolu olarak tanımlanmıştır. A kolu 105 cm. uzunluğunda 1400 gram ağırlığında, kalınlığı 3 mm, eni ise 6 cm. olan çelik bir levhadan oluşmaktadır. Kullanılan ikinci kol B kolu olup kolun ucunda 3800 gram ağırlığında yük bulunmaktadır. Kolun ağırlığı ise 1600 gramdır. Yük ile birlikte toplam ağırlık 5400 gramdır. Kolun uzunluğu 110 cm., kalınlığı 3 mm., eni 8 cm. olup çelik bir levhadan oluşmaktadır. Üçüncü kol olan C kolu ise esnekliği daha fazla olan alüminyum levhadan yapılmıştır. Alüminyum kolun uzunluğu 90 cm., eni 7 cm., kalınlığı ise 1 mm.'dir. Alüminyum levhadan oluşan kolun ağırlığı ise 280 gramdır. Kolların özellikleri Çizelge 1'de gösterilmiştir. Yapılan çalışmada her üç kol için de değişik kontrol yöntemleri için deneysel sonuçlar alınmış ve elde edilen veriler değerlendirilmiştir.

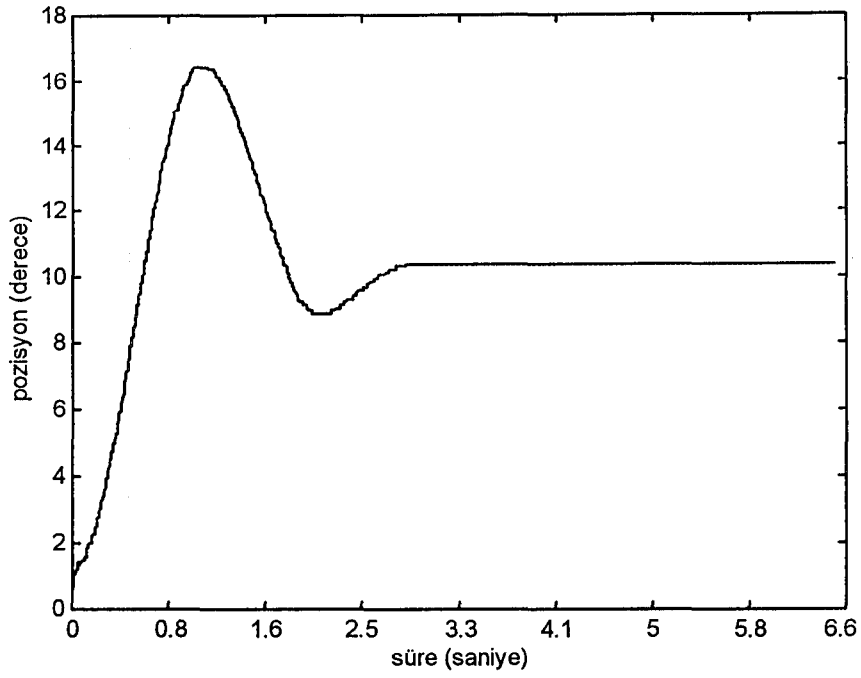
4.1.1 A Kolu İçin Elde Edilen Deneysel Sonuçlar

105 cm. uzunluğunda 1400 gram ağırlığındaki A koluna (1)'de verilen PD kontrol uygulanmıştır. Uygulanan kontrolörün kazanç değerleri $K_p = 0.035$, $K_v = 0.030$ olarak seçilmiştir. Kolun gitmesi istenen referans değeri ise 10 derecedir. Elde edilen deneysel sonuç Şekil 4.1'de gösterilmiştir. Burada K_p volt/derece, K_v volt.sn/radyan cinsindedir.

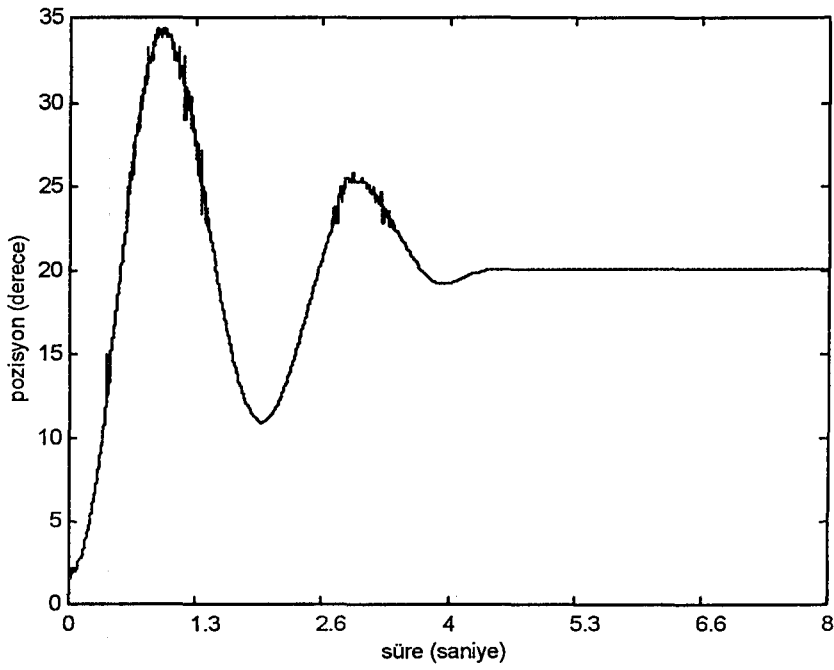
Şekil 4.2'de ise aynı kola aynı şartlarda referans değeri 20 derece olacak şekilde (1)'de verilen PD kontrol uygulanmış, kararlı duruma ulaşana dek kolun daha fazla salınım yaptığı ve daha geç sönümlendiği görülmüştür. Kontrolörün kazanç değerleri $K_p = 0.035$, $K_v = 0.030$ olarak seçilmiştir. Yatışkın durum pozisyon hatası yaklaşık 0.12 derece olarak ölçülmüştür.

Kolun adı	Yapıldığı malzeme	Uzunluğu	Eni	Kalınlığı	Ağırlığı	Yük
A kolu	Çelik	105 cm.	6 cm.	3 mm.	1400 gram	--
B kolu	Çelik	110 cm	8 cm.	3 mm.	1600 gram	3800 gram
C kolu	Alüminyum	90 cm.	7 cm.	1 mm.	280 gram	--

Çizelge 1. Deneysel çalışmada kullanılan kolların özellikleri



Şekil 4.1 PD kontrolör sonuçları (A kolu ve 10 derece referansı için)

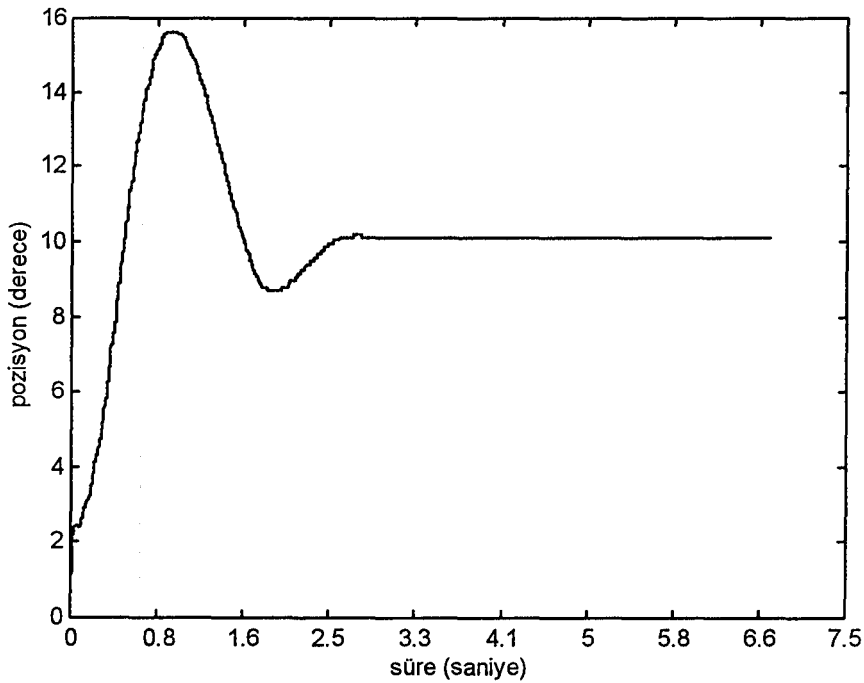


Şekil 4.2 PD kontrolör sonuçları (A kolu ve 20 derece referansı için)

(3) 'te verilen doğrusal olmayan PD kontrolörle kazanç değerleri $K_{p1} = 0.034$, $K_{p2} = 0.00035$, $K_{v1} = 0.037$ ve $K_{v2} = 0.00035$ alınarak yapılan çalışma sonucunda Şekil 4.3'teki tepki elde edilmiştir. Burada,

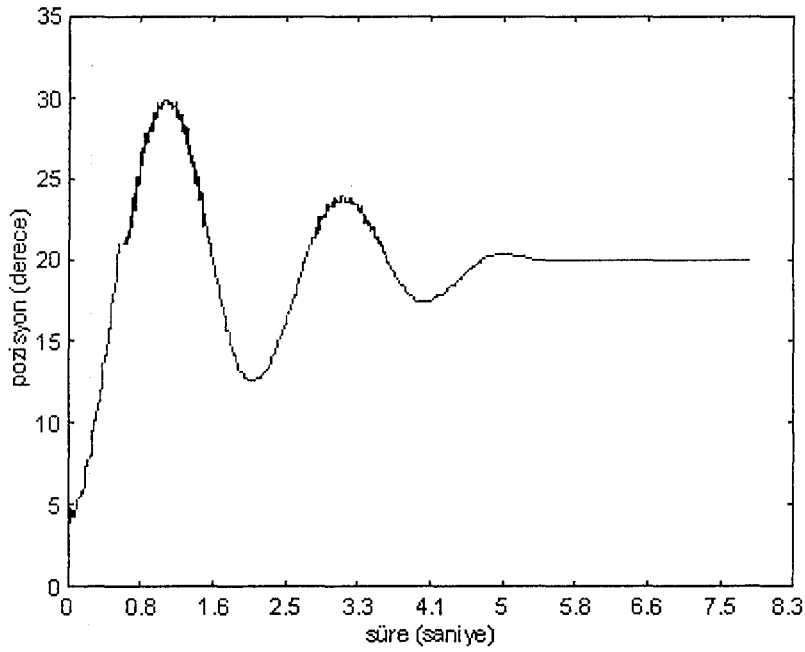
$$s(e(t)) = \begin{cases} |e(t)|, & |e(t)| \leq 10 \text{ derece} \\ 10, & |e(t)| > 10 \text{ derece} \end{cases}$$

olarak alınmıştır. Elde edilen sonuçlarda pozisyon hatası 0.1 derece olarak bulunmuştur. Burada referans değeri olarak 10 derece alınmıştır.



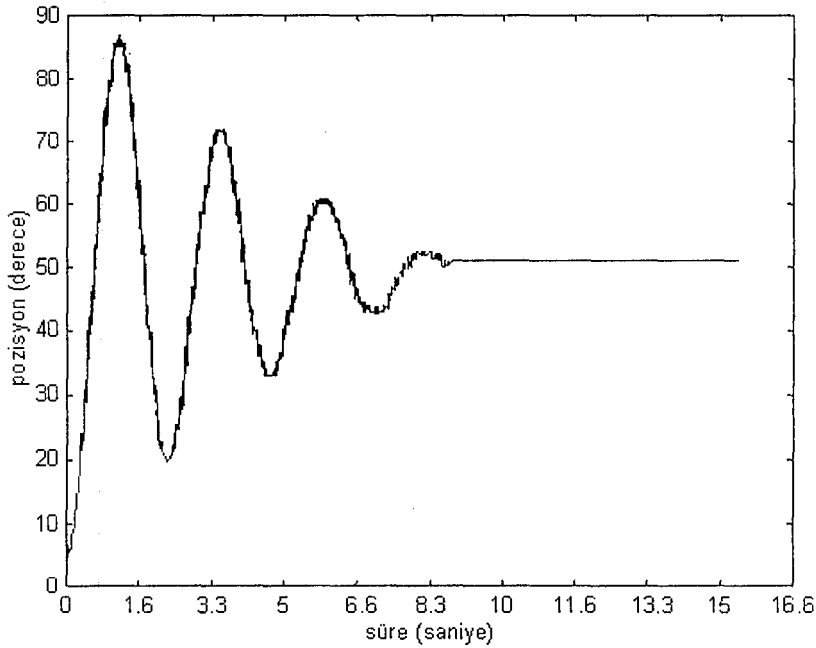
Şekil 4.3 Doğrusal olmayan PD benzeri kontrolör sonuçları
(A kolu ve 10 derece için)

Bir başka çalışmada ise kola yine aynı kazanç değerlerinde doğrusal olmayan PD benzeri kontrolör uygulanmış ve sonuçlar Şekil 4.4'te gösterilmiştir. Burada kolun gitmesi istenen referans değeri 20 derece olarak seçilmiştir. Hata değeri 0.1 dereceden düşüktür. Elde edilen sonuçta kolun daha düşük genlikli salınımlar yaptığı görülmektedir.



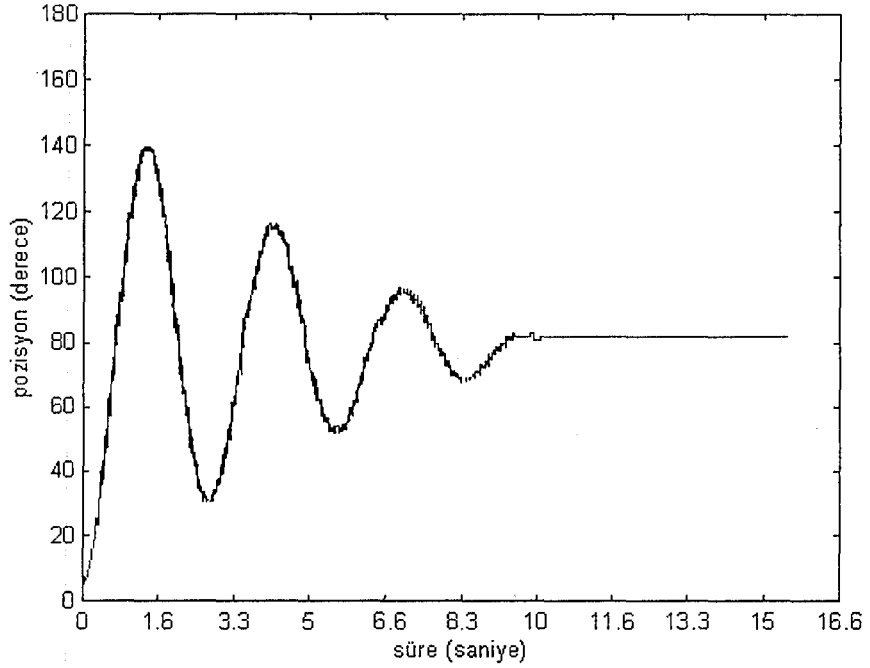
Şekil 4.4 Doğrusal olmayan PD benzeri kontrolör sonuçları
(A kolu ve 20 derece için)

Bir başka uygulamada yine A koluna 50 derece referansı için PD kontrolör (1) uygulanmış, elde edilen veriler Şekil 4.5'te gösterilmiştir. Uygulanan kontrolörün kazanç değerleri Şekil 1 ve Şekil 2'dekilerle aynı alınmıştır.



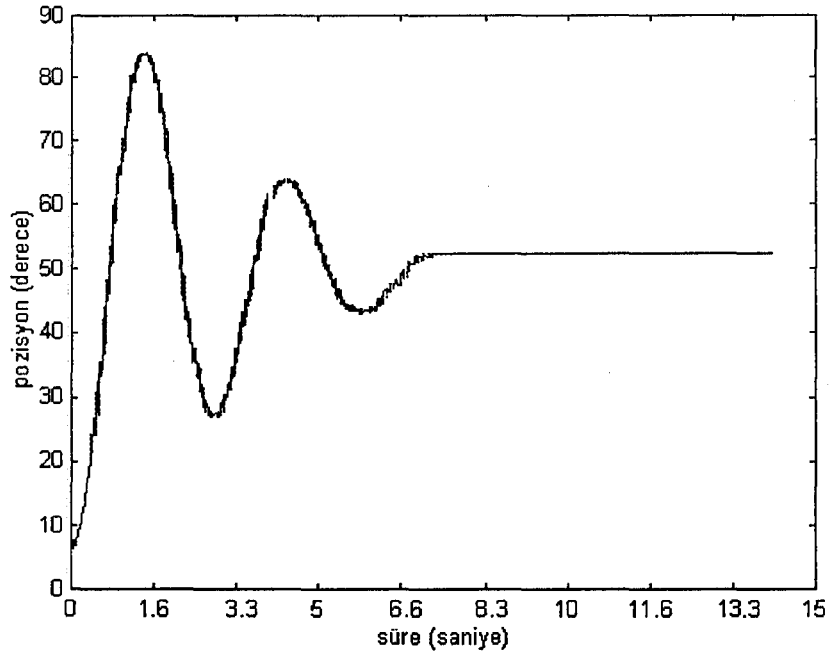
Şekil 4.5 PD kontrolör sonuçları (A kolu ve 50 derece referansı için)

Yine aynı kola aynı kontrolör ve kontrolör kazanç değerleri uygulandığında 80 derece referansı için alınan sonuçlar ise Şekil 4.6'da gösterilmiştir. Bir başka uygulamada ise 50 ve 80 derece referans değerleri için (3) eşitliğinde verilen doğrusal olmayan PD benzeri kontrol uygulanmıştır. Uygulanan kontrolörün kazanç değerleri Şekil 4.3 ve Şekil 4.4'tekilerle aynıdır. $K_{p1} = 0.034$, $K_{p2} = 0.00035$, $K_{v1} = 0.037$ ve $K_{v2} = 0.00035$ şeklindedir. 50 derece referans değeri için alınan değerler Şekil 4.7'de 80 derece referansı için alınan veriler ise Şekil 4.8'de gösterilmiştir.

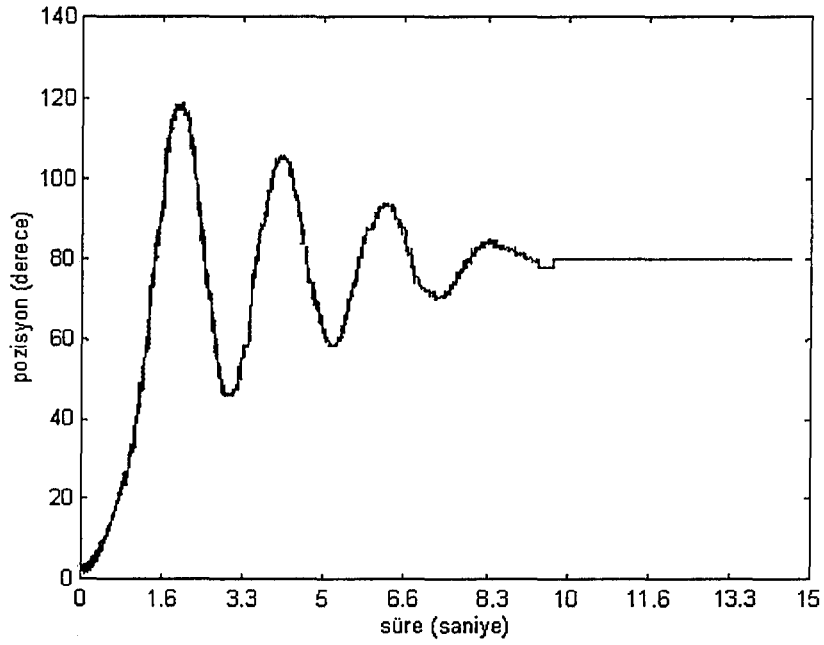


Şekil 4.6 PD kontrolör sonuçları (A kolu ve 80 derece referansı için).

Elde edilen verilere göre doğrusal olmayan PD benzeri kontrolör kola uygulanması gereken kontrol bilgisini belli aralıkta sınırlandırdığı için başlangıçta daha düşük bir torkla kol hareket etmekte ve bunun sonucunda kolun ulaşması istenen referans değerine daha geç gittiği gözlenmektedir. Doğrusal olmayan PD benzeri kontrolör kola başlangıç anında daha düşük tork uyguladığı için motorun ani torklar karşısında zorlanması önlenmiş ve ani torklardan kaynaklanan esnemeler en aza indirilmiş olur. Şekil 4.1 ve Şekil 4.3'te 10 derece referansı için Şekil 4.2 ve Şekil 4.4'te 20 derece referansı için, Şekil 4.5 ve Şekil 4.7'de 50 derece referansı için Şekil 4.6 ve Şekil 4.8'de ise 80 derece referansı için bu sonuçlar görülebilir.

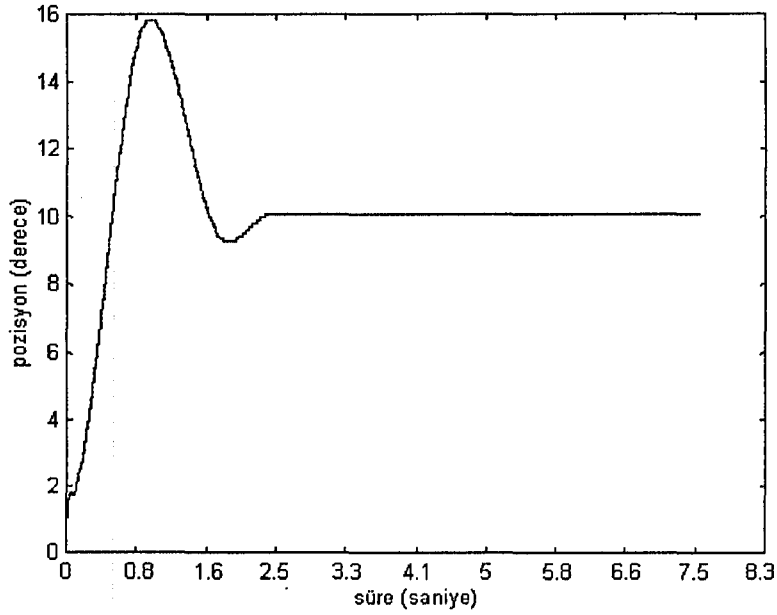


Şekil 4.7 Doğrusal olmayan PD benzeri kontrolör sonuçları
(A kolu ve 50 derece referansı için)

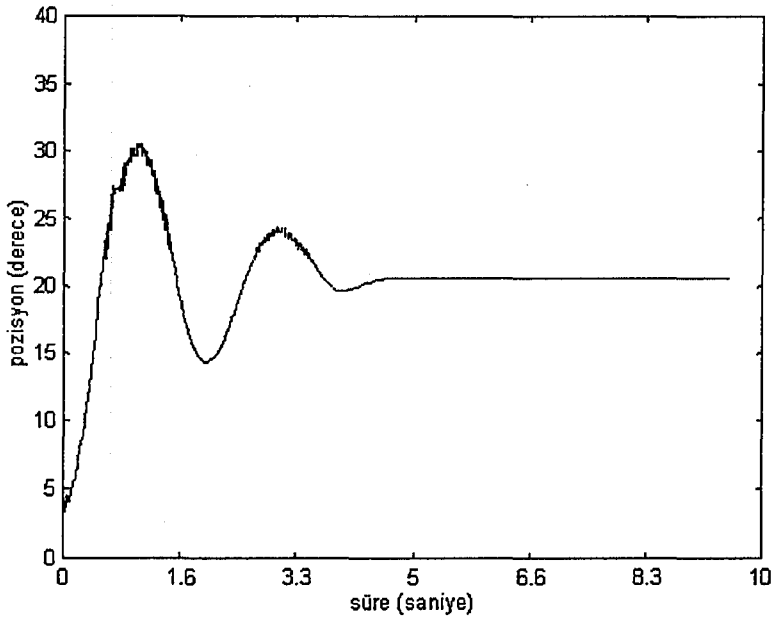


Şekil 4.8 Doğrusal olmayan PD benzeri kontrolör sonuçları
(A kolu ve 80 derece referansı için)

Daha sonra yapılan çalışmada ise (1) eşitliği ile verilen kontrolöre integratör eklenerek elde edilen PID kontrol (2) uygulanmıştır. Referans değeri olarak 10 ve 20 derece seçilen çalışma sonucu elde edilen veriler Şekil 4.9 ve Şekil 4.10'da gösterildiği gibidir. Kullanılan kontrolör kazanç değerleri $K_p=0.002$, $K_i=0.015$ ve $K_v=0.0125$ olarak seçilmiştir.

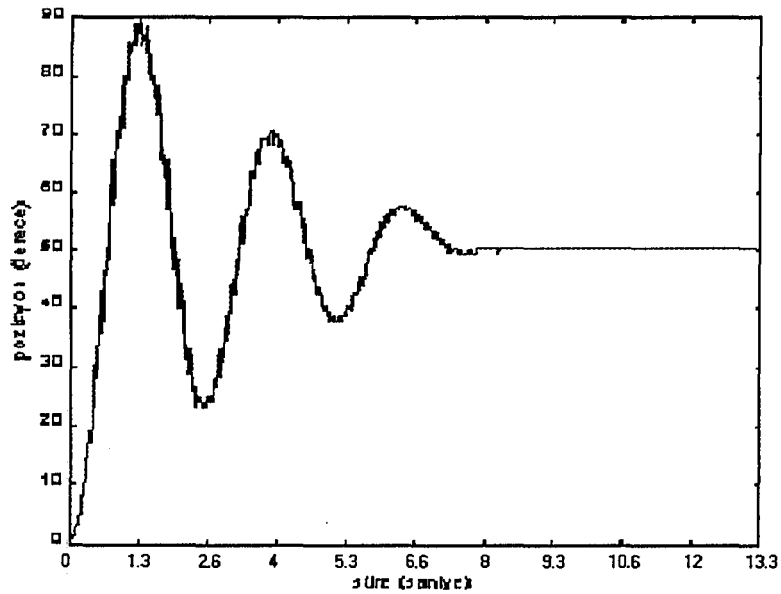


Şekil 4.9 PID kontrolör sonuçları (A kolu ve 10 derece için)

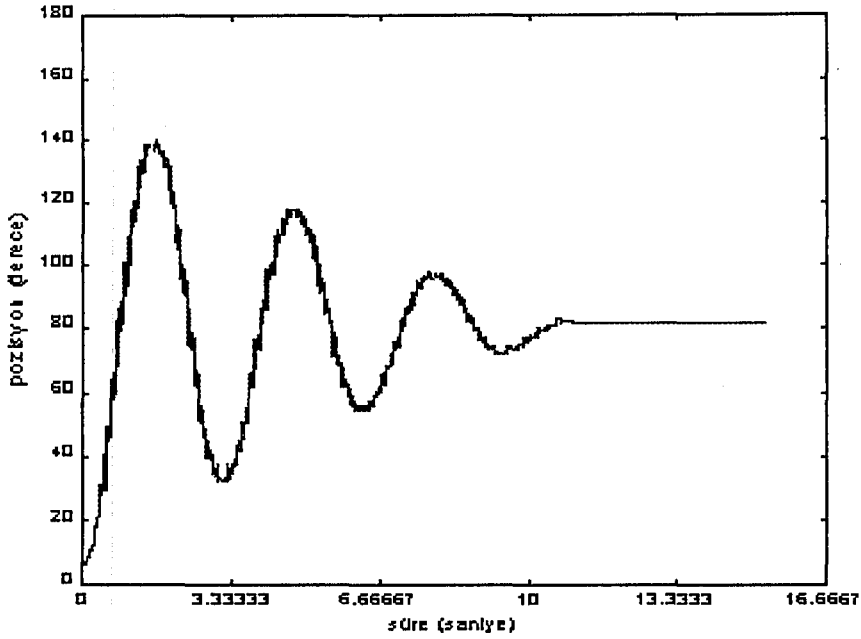


Şekil 4.10 PID kontrolör sonuçları (A kolu ve 20 derece için)

Daha sonra yapılan çalışmada ise yine aynı kontrolör (2) ve kontrolör kazanç değerleri kullanılarak 50 ve 80 derece referansı için deneysel sonuçlar alınmıştır. Referans değeri olarak 50 derece seçilen çalışma sonucu elde edilen veriler Şekil 4.11’de, 80 derece seçilen çalışma sonucu elde edilen veriler Şekil 4.12’de gösterildiği gibidir. Kullanılan kontrolör kazanç değerleri $K_p=0.002$, $K_i=0.015$ ve $K_v=0.0125$ olarak seçilmiştir.

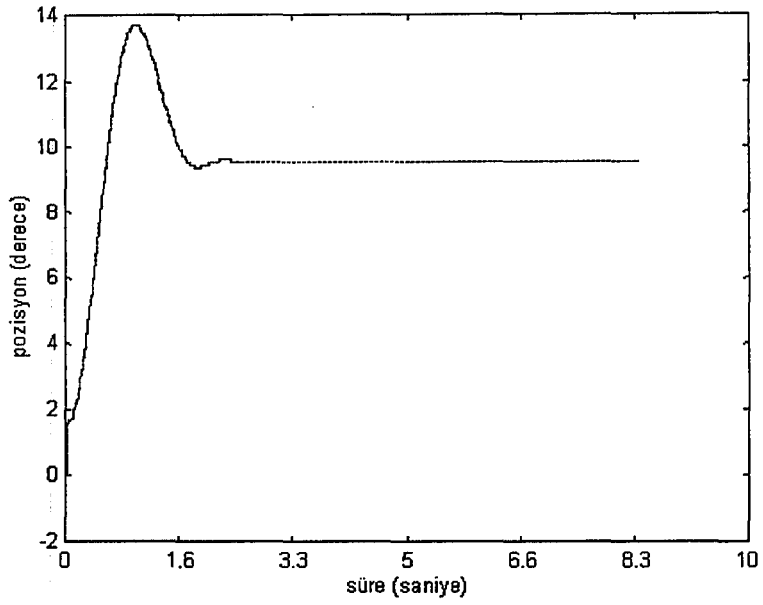


Şekil 4.11 PID kontrolör sonuçları (A kolu ve 50 derece referansı için)

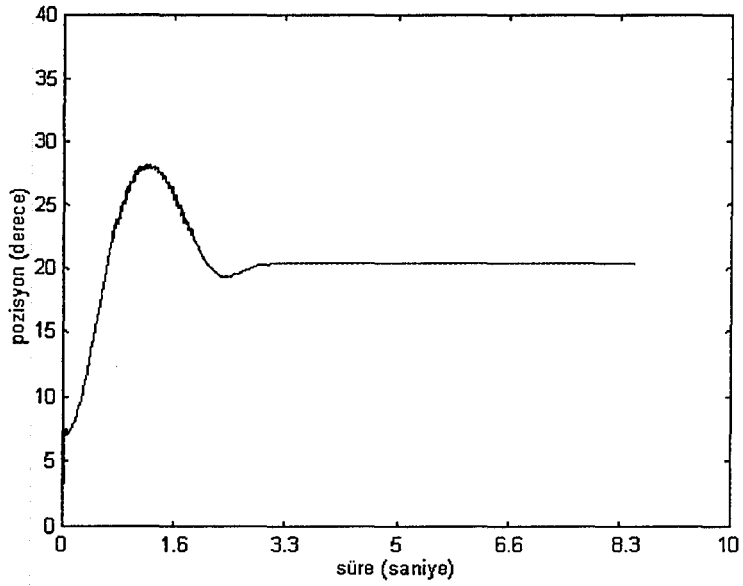


Şekil 4.12 PID kontrolör sonuçları (A kolu ve 80 derece referans için)

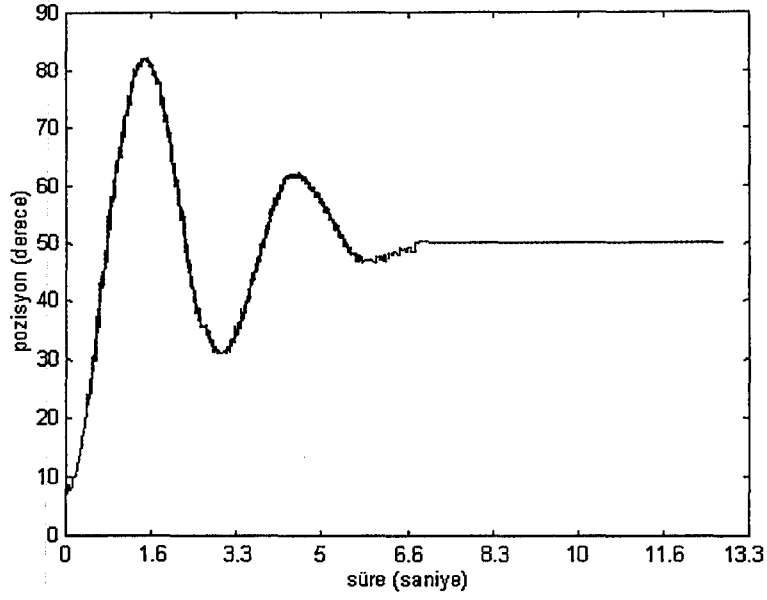
Kola uygulanan doğrusal olmayan PID benzeri kontrol (5) yöntemiyle referans değeri için 10 ve 20 derece alınarak yapılan deneysel çalışmalar sonucunda Şekil 4.13 ve Şekil 4.14'teki sonuçlar alınmıştır. Uygulanan kontrolörün kazanç değerleri şu şekildedir $K_{p1} = 0.035$, $K_{p2} = 0.00035$, $K_{v1} = 0.037$, $K_{v2} = 0.00037$ ve $K_i = 0.025$. Benzer şekilde aynı kontrol kazanç değerleri için 50 ve 80 derece referans alınarak yapılan deneysel çalışmalarda alınan sonuçlar ise Şekil 4.15 ve Şekil 4.16'da gösterilmiştir.



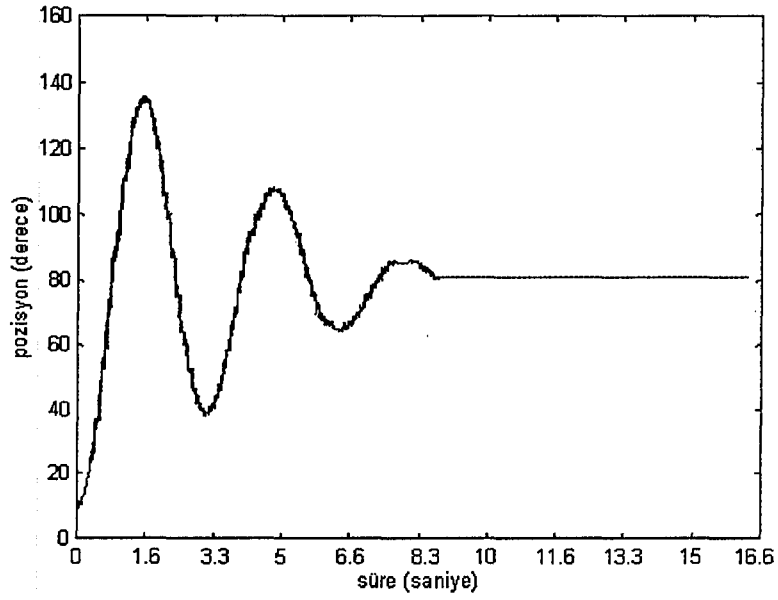
Şekil 4.13 Doğrusal olmayan PID benzeri kontrolör sonuçları
(A kolu ve 10 derece referansı için)



Şekil 4.14 Doğrusal olmayan PID benzeri kontrolör sonuçları
(A kolu ve 20 derece referansı için)



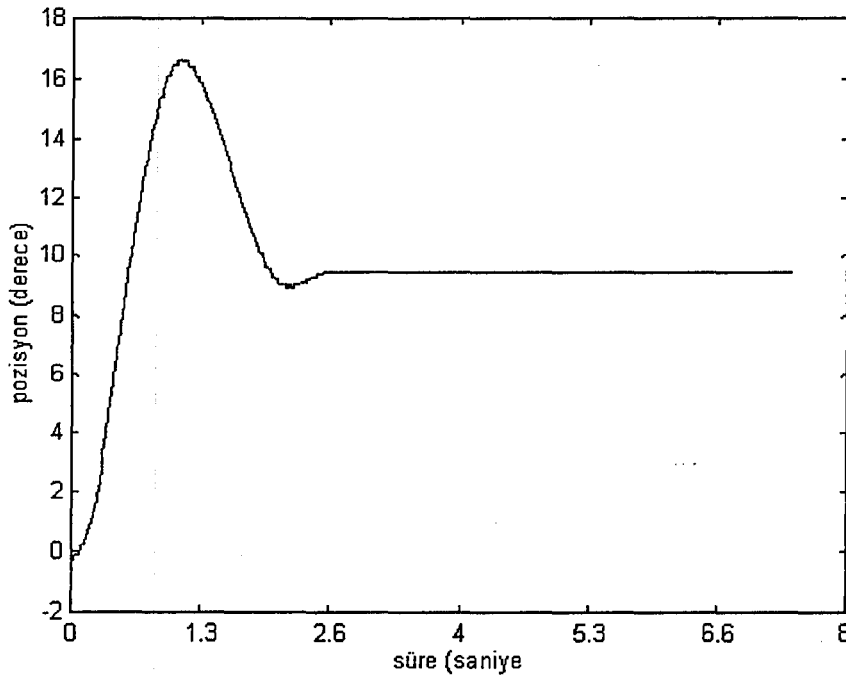
Şekil 4.15 Doğrusal olmayan PID benzeri kontrolör sonuçları
(A kolu ve 50 derece referansı için)



Şekil 4.16 Doğrusal olmayan PID benzeri kontrolör sonuçları
(A kolu ve 80 derece referansı için)

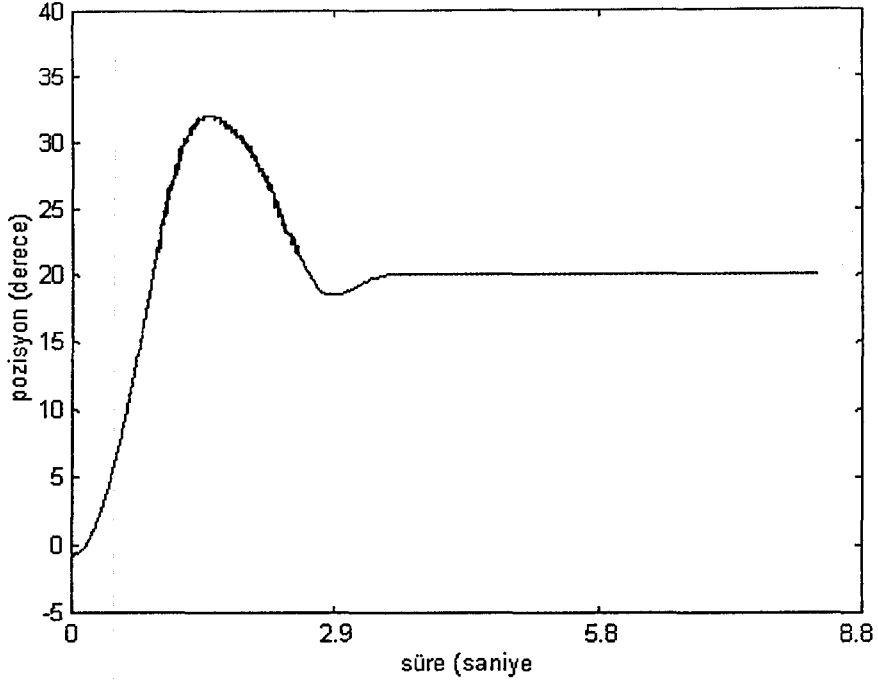
A koluna uygulanan PID ve doğrusal olmayan PID benzeri kontrolör sonuçlarına göre doğrusal olmayan PID benzeri kontrolör kola uygulanan başlangıç torkunu sınırladığı için kolun başlangıçta yaptığı salınımlar azalmaktadır.

A kolunun üzerine PZT sensörler yerleştirilerek elde edilen titreşim bilgisinin kontrolöre geri beslenmesi ile elde edilen veriler Şekil 4.17'de gösterilmiştir. Kullanılan kazanç değerleri $K_{p1} = 0.035$, $K_{p2} = 0.00037$, $K_{v1} = 0.035$, $K_{v2} = 0.00037$ ve $K_i = 0.025$, $K_d = 0.001$ olarak alınmıştır. Referans değeri ise 10 derecedir. Kontrolör (6) nolu eşitlikte verildiği gibidir.



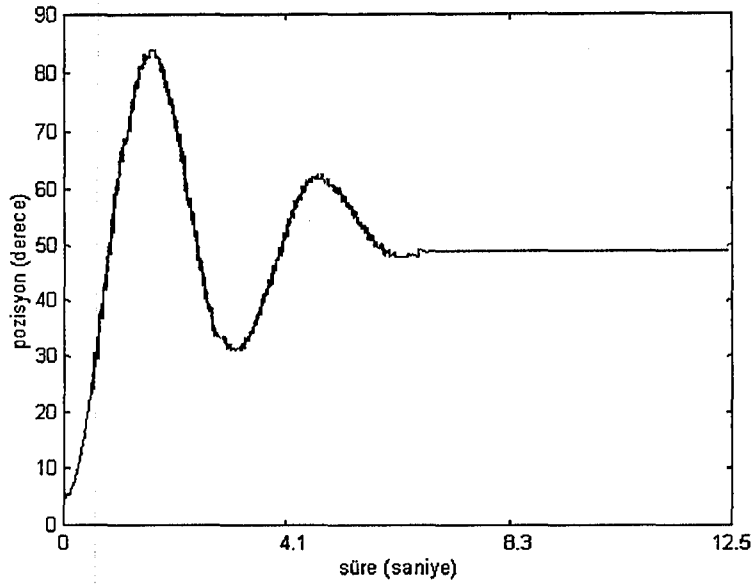
Şekil 4.17 PZT ilave edilmiş PID benzeri kontrolör sonuçları
(A kolu ve 10 derece referansı için)

Aynı kola aynı kontrolör uygulanarak 20 derece referansı için alınan sonuçlar ise Şekil 4.18'de gösterilmiştir.

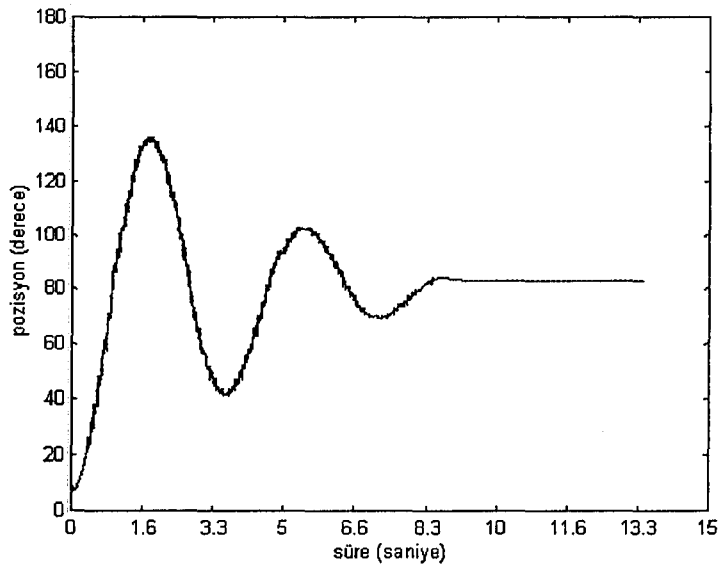


Şekil 4.18 PZT ilave edilmiş PID benzeri kontrolör sonuçları
(A kolu ve 20 derece referansı için)

Aynı kola aynı kontrolör uygulanarak 50 ve 80 derece referansı için alınan sonuçlar ise Şekil 4.19 ve Şekil 4.20'de gösterilmiştir. A koluna uygulanan doğrusal olmayan PID benzeri kontrolöre PZT bilgisinin de ilave edilmesi ile alınan sonuçlara göre kol istenilen referans değerine doğrusal olmayan PID benzeri kontrolörden daha kısa sürede gitmektedir.



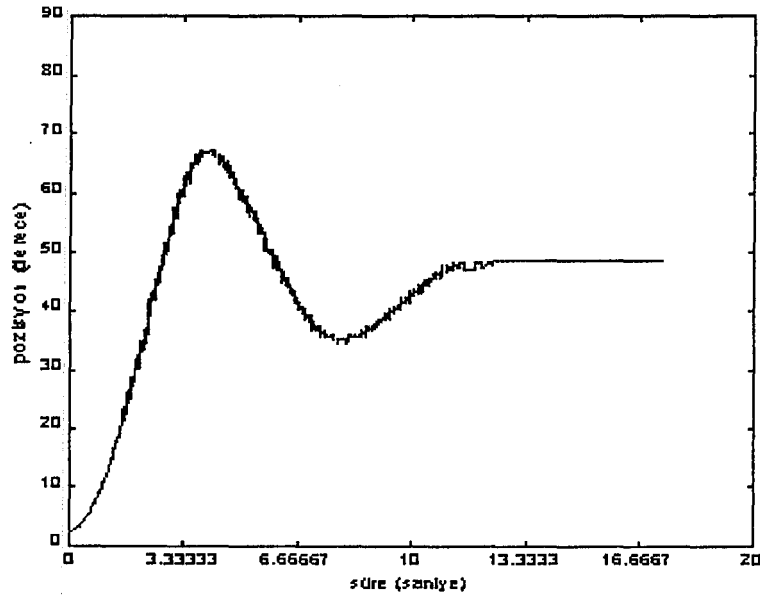
Şekil 4.19 PZT ilave edilmiş doğrusal olmayan PID benzeri kontrolör
(A kolu ve 50 derece için)



Şekil 4.20 PZT ilave edilmiş doğrusal olmayan PID benzeri kontrolör
(A kolu ve 80 derece için)

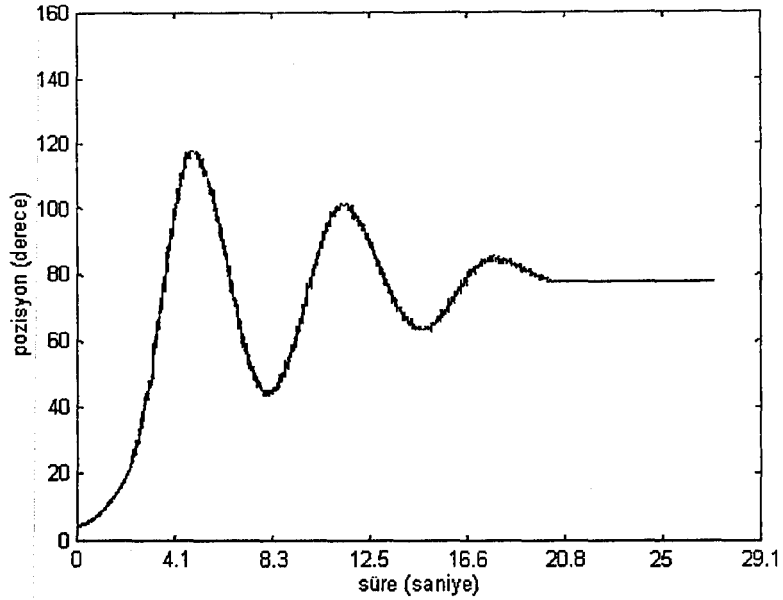
4.1.2 B Kolu İçin Elde Edilen Deneysel Sonuçlar

B kolu için öncelikle PID kontrol yöntemi (2) uygulanmış elde edilen sonuçlar incelenmiştir. Kullanılan kontrolör kazanç değerleri $K_p=0.002$, $K_i=0.015$ ve $K_v=0.0125$ olarak seçilmiştir. Buna göre 50 derecelik referans değeri için alınan sonuçlar Şekil 4.21'de gösterilmiştir.



Şekil 4.21 PID kontrolör sonuçları (B kolu ve 50 derece referansı için)

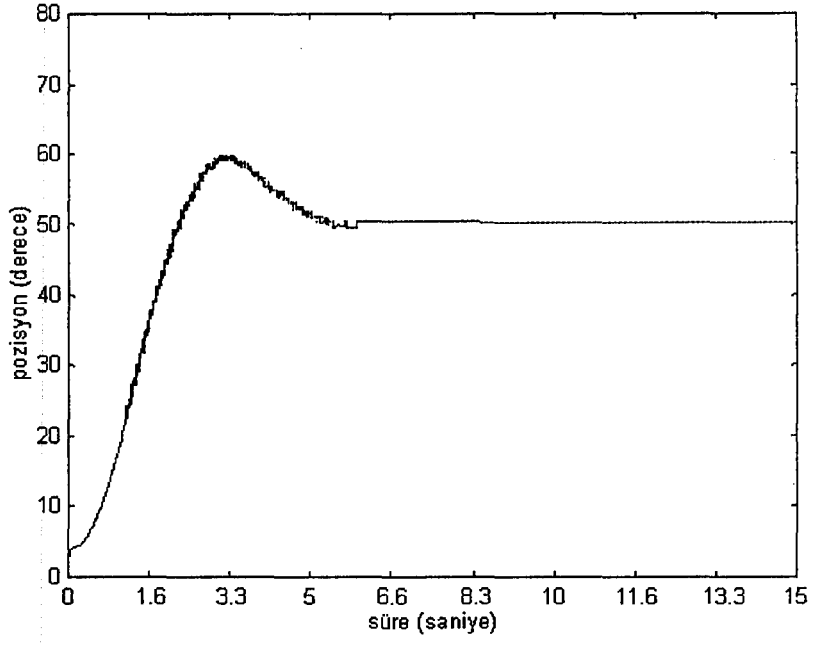
Daha sonra aynı kontrolör kazanç değerleri için 80 derecelik referans değerinde kolun verdiği tepki incelenmiştir. Uygulanan kontrol sonucu alınan veriler de Şekil 4.22'de gösterilmiştir.



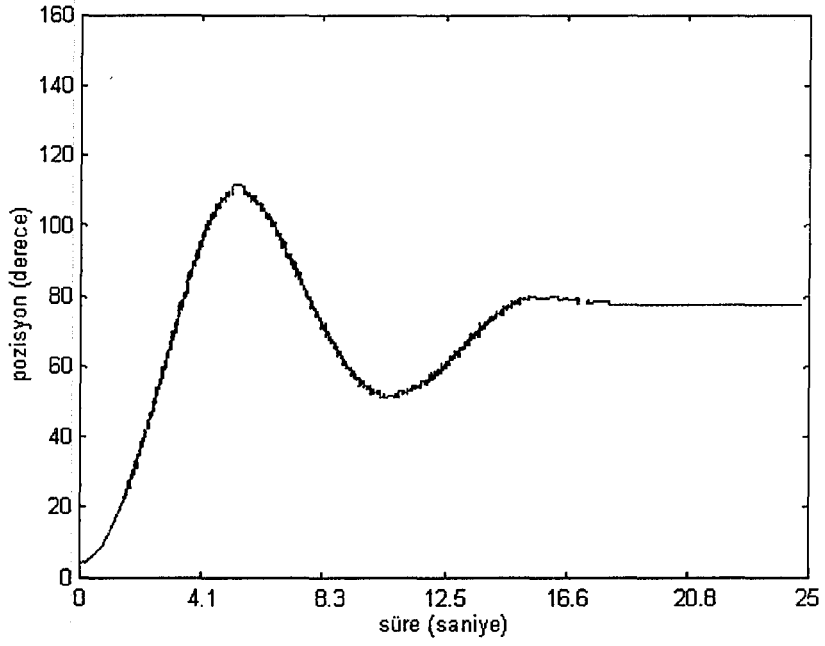
Şekil 4.22 PID kontrolör sonuçları (B kolu ve 80 derece referansı için)

Aynı kol için 50 derece referans değerlerine sahip doğrusal olmayan PID benzeri kontrol (5) uygulanmış ve alınan sonuçlar Şekil 4.23'te gösterilmiştir. Uygulanan kontrolörün kazanç değerleri şu şekildedir $K_{p1} = 0.035$, $K_{p2} = 0.00035$, $K_{v1} = 0.037$, $K_{v2} = 0.00037$ ve $K_3 = 0.025$. Benzer şekilde aynı kontrol kazanç değerleri için 80 derece referans alınarak yapılan deneysel çalışmada alınan sonuçlar Şekil 4.24'te gösterilmiştir.

Elde edilen verilere göre başlangıç anında kola uygulanan tork sınırlandırıldığı için doğrusal olmayan PID benzeri kontrolör PID kontrolöre oranla daha az salınım yapmaktadır. Ayrıca doğrusal olmayan PID benzeri kontrolör ile elde edilen sonuçlarda istenilen referans değerine daha kısa sürede ulaşıldığı görülmektedir.



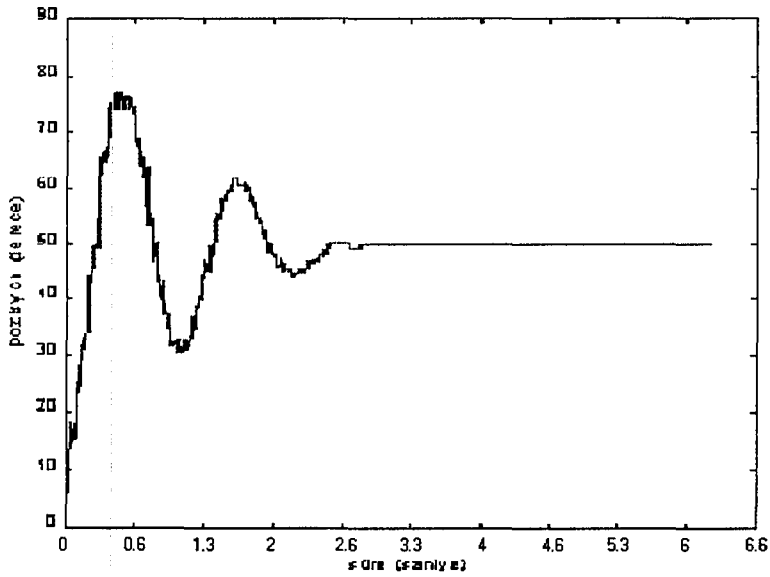
Şekil 4.23 Doğrusal olmayan PID benzeri kontrolör sonuçları
(B kolu ve 50 derece referansı için)



Şekil 4.24 Doğrusal olmayan PID benzeri kontrolör sonuçları
(B kolu ve 80 derece referansı için)

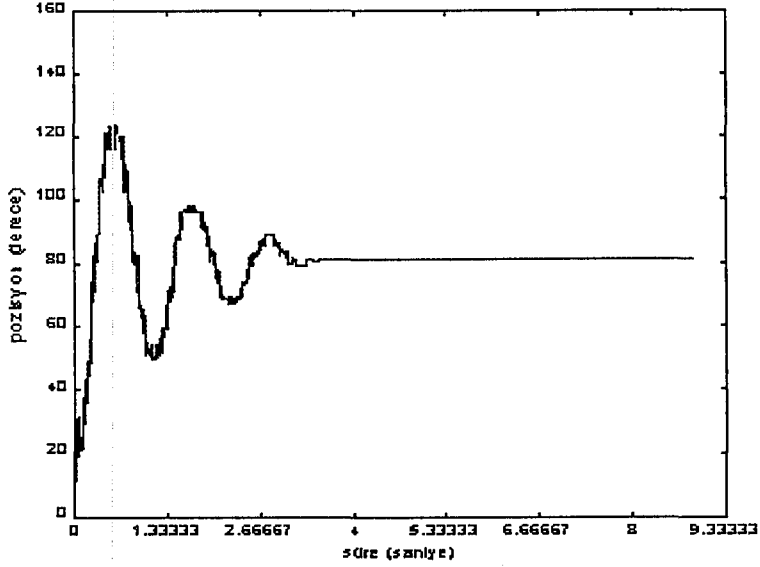
4.1.3 C Kolu İçin Elde Edilen Deneysel Sonuçlar

C kolu için yapılan deneysel çalışmada kola öncelikle PID kontrol yöntemi (2) uygulanmıştır. Kullanılan kontrolör kazanç değerleri $K_p=0.002$, $K_i=0.015$ ve $K_v=0.0125$ olarak seçilmiştir. Buna göre 50 derecelik referans değeri için alınan sonuçlar Şekil 4.25'te gösterilmiştir..



Şekil 4.25 PID kontrolör sonuçları (C kolu ve 50 derece referansı için)

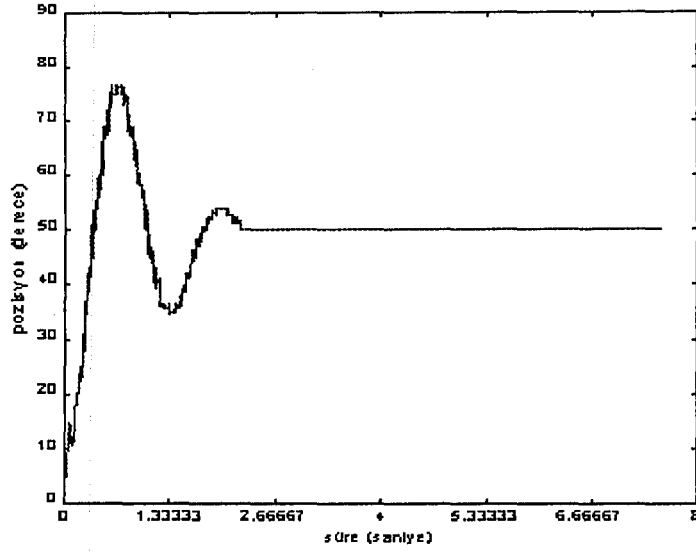
Daha sonra aynı kontrolör kazanç değerleri için 80 derecelik referans değerinde kolun verdiği tepki incelenmiştir. Uygulanan kontrol sonucu alınan değerler Şekil 4.26'da gösterilmiştir.



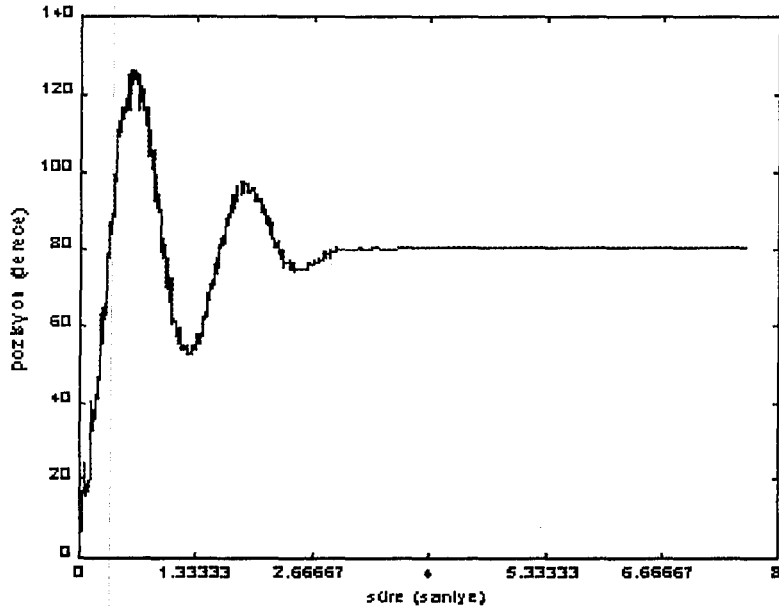
Şekil 4.26 PID kontrol sonuçları (C kolu ve 80 derece referansı için)

Daha sonra kola doğrusal olmayan PID benzeri kontrol (5) uygulanmış ve sonuçları aşağıda incelenmiştir. Uygulanan kontrolörün kazanç değerleri şu şekildedir $K_{p1} = 0.035$, $K_{p2} = 0.00035$, $K_{v1} = 0.037$, $K_{v2} = 0.00037$ ve $K_i = 0.025$. Referans değeri 50 derece alınarak yapılan deneysel çalışma sonucunda Şekil 4.27'deki sonuçlar alınmıştır. Şekil 4.28'de yine aynı kontrolör kazanç değerleri için 80 derece referansında alınan deneysel sonuçlar gösterilmiştir.

Elde edilen verilere göre doğrusal olmayan PID kontrol ile alınan sonuçlar PID kontrolöre göre daha kısa sürede istenilen referans değerine ulaşmakta ve salınımların genliği daha düşük seviyede kalmaktadır.



Şekil 4.27 Doğrusal olmayan PID benzeri kontrolör sonuçları
(C kolu ve 50 derece referansı için)



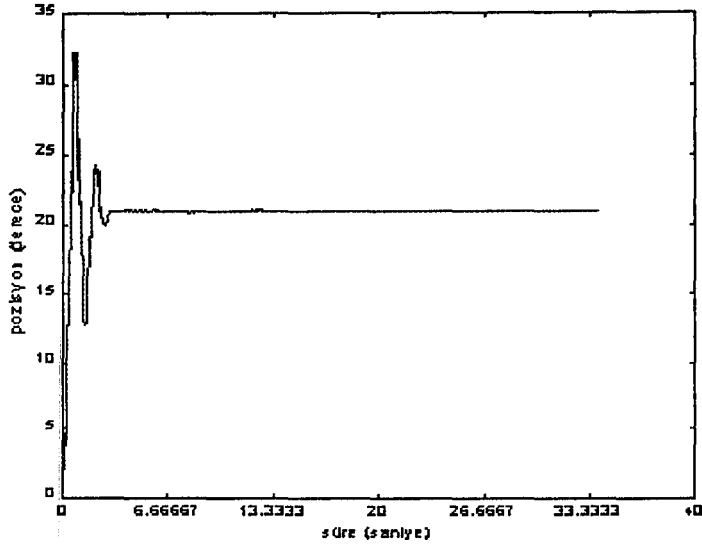
Şekil 4.28 Doğrusal olmayan PID benzeri kontrolör sonuçları
(C kolu ve 80 derece referansı için)

4.2 İki Eklemlı Esnek Robot Kolu İin Alınan Deneysel Sonular

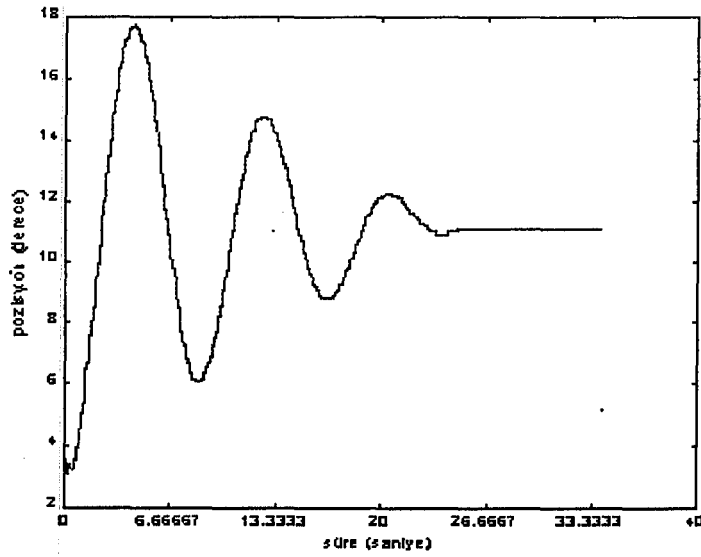
Bu blmde birinci parası 110 cm. uzunluęunda elik bir levhadan, ikinci parası ise 57 cm. uzunluęunda alminyum levhadan oluřan iki eklemlı esnek robot kolu dzeneęinin eřitli yntemlerle kontrol incelenmiřtir. Kolun birinci parasını oluřturan elik levhanın eni 8 cm., kalınlıęı 3 mm. ve aęırlıęı ise 1600 gramdır. Kolun ikinci parasını oluřturan alminyum levha ise 7 cm. eninde, 1 mm. kalınlıęında ve 220 gram aęırlıęındadır. İki eklemlı esnek robot kolu dzeneęi hakkında daha ayrıntılı bilgiler Blm 2.2'de anlatılmıřtır.

İki Eklemlı esnek robot kolu dzeneęine ncelikle PID kontrol yntemi (7) uygulanmıř elde edilen sonular Őekil 4.29'da gsterilmiřtir. Burada kazanç deęerleri her iki para iin de $K_p = 0.05$, $K_v = 0.015$ ve $K_i = 0.0325$ olarak alınmıřtır. Referans deęerleri olarak birinci para iin 10 derece, ikinci para iinse 20 derece seilmiřtir.

Daha sonra iki eklemlı esnek robot kolu kontrol iin her iki ekleme de doęrusal olmayan PID benzeri kontrol algoritması (8) uygulanmıř elde edilen sonular Őekil 4.30 ve Őekil 4.31'de gsterilmiřtir. İlk olarak birinci para iin $K_{p1} = 0.0437$, $K_{p2} = 0.0037$, $K_{v1} = 0.0437$, $K_{v2} = 0.0037$ ve $K_3 = 0.0324$ ikinci kol iin $K_{p1} = 0.01037$, $K_{p2} = 0.00037$, $K_{v1} = 0.01037$, $K_{v2} = 0.00037$ ve $K_3 = 0.012$ alınmıř ve referans deęerleri olarak birinci para iin 10 derece ikinci para iinse 20 derece seilmiřtir. α deęeri 10 derece olarak alınmıř elde edilen sonular Őekil 4.30'da gsterilmiřtir. Őekil 4.31'de gsterilen sonularda ise yine aynı kazanç deęerleri uygulanmıř fakat farklı referans deęerleri kullanılmıřtır. Birinci para iin bařlangıta 20 derece olan pozisyon deęeri 10 derece referansına ikinci para iin ise 72 derece referansına gnderilmiřtir.

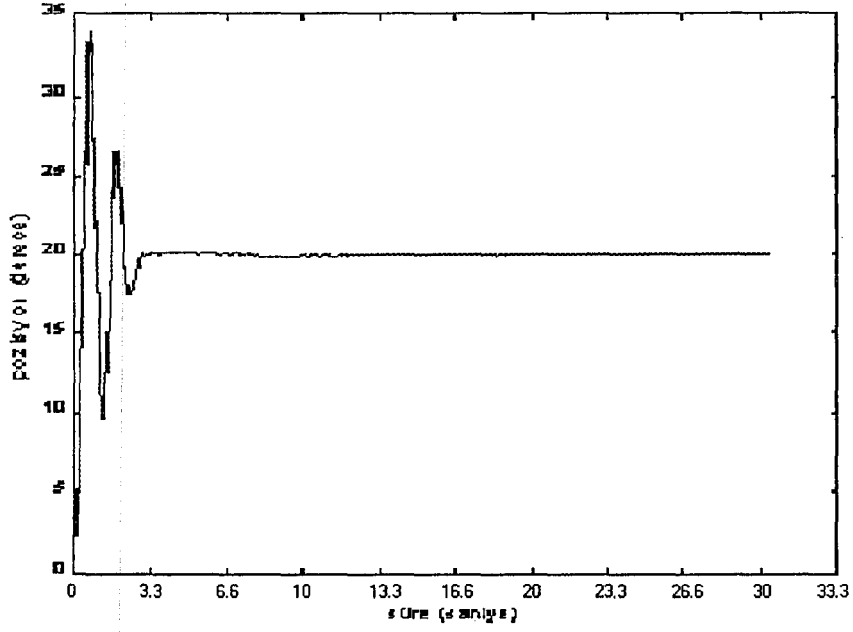


(a)

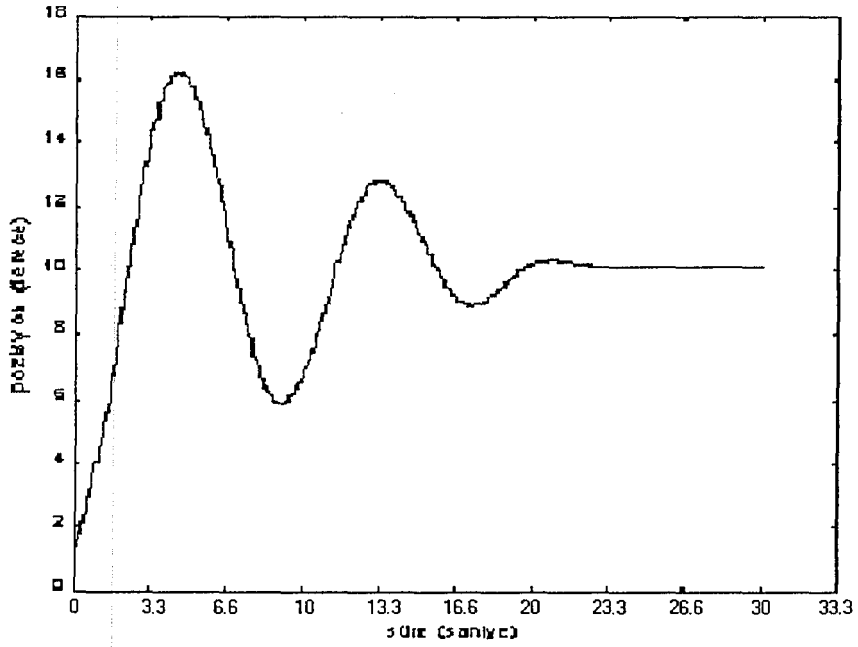


(b)

Şekil 4.29 İki eklemlı kol için alınan PID kontrolör sonuçları
 (1. parça 20 derece, 2. Parça 10 derece)
 (a) İkinci parça için alınan pozisyon bilgisi
 (b) Birinci parça için alınan pozisyon bilgisi

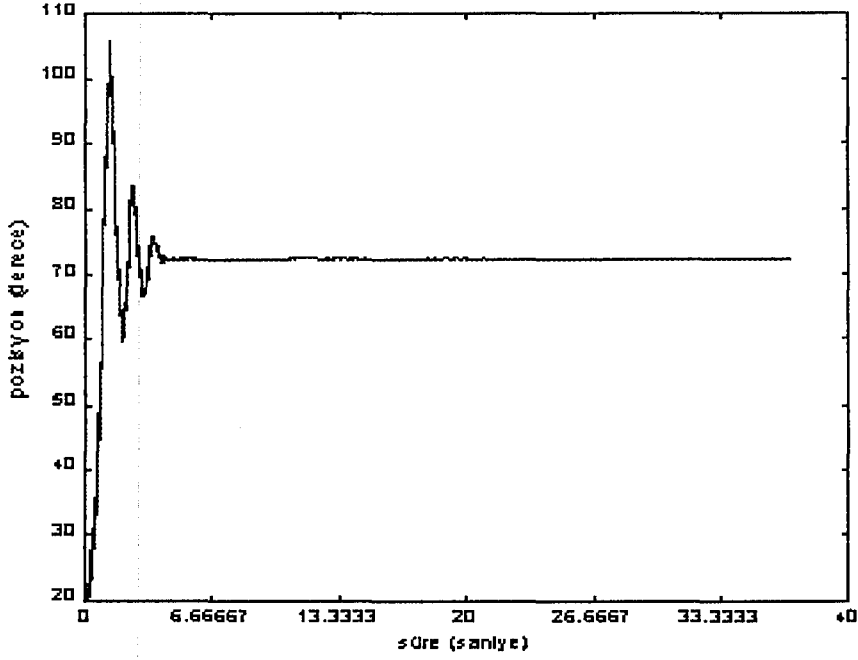


(a)

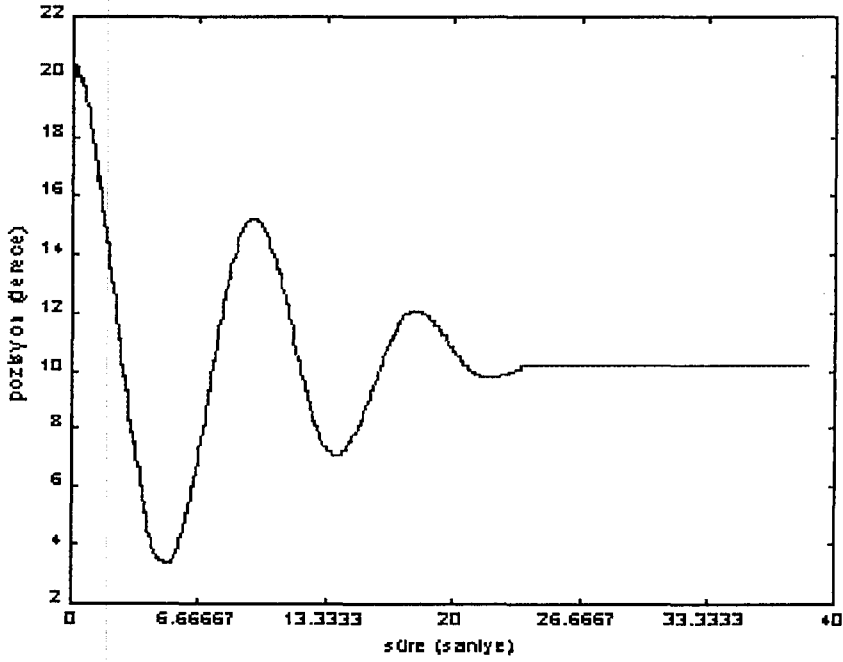


(b)

Şekil 4.30 İki eklemlı kol için alınan doğrusal olmayan PID benzeri kontrolör sonuçları (1. parça 10 derece 2. parça 20 derece)
 (a) İkinci parça için alınan pozisyon bilgisi
 (b) Birinci parça için alınan pozisyon bilgisi



(a)



(b)

Şekil 4.31 İki eklemlı kol için alınan doğrusal olmayan PID benzeri kontrolör sonuçları (1. parça 10 derece 2. parça 72 derece)
 (a) İkinci parça için alınan pozisyon bilgisi
 (b) Birinci parça için alınan pozisyon bilgisi

5. SONUÇ

Bu çalışmada esnek robot kollarının kontrolü için geliştirilen düzeneklerle tek ve iki eklemlili esnek robot kollarının değişik kontrol yöntemlerine verdikleri tepkiler incelenmiştir. Tek eklemlili esnek robot kolu için farklı özelliklerdeki kollarla deneysel çalışmalar yapılmış, esneklik özellikleri farklı çelik ve alüminyum kolların performansları karşılaştırılmıştır.

Tek eklemlili esnek robot kolları üzerine geçmişte farklı kontrol yöntemleriyle çeşitli çalışmalar yapılmıştır. Bu çalışmada ise PD, PID, doğrusal olmayan PD ve PID kontrol yöntemleri uygulanıp farklı durumlardaki performanslar incelenmiştir. PID ve doğrusal olmayan PID benzeri kontrolörler tasarlanırken kontrolör kazanç değerleri yatışkın durum pozisyon hatasını en kısa sürede söndümlendirecek şekilde deneme yanılma yöntemiyle bulunmuştur. Bu amaçla alınan farklı değerlerdeki kontrolör kazançlarına karşı sistemin verdiği tepki incelenmiştir.

Doğrusal olmayan PID benzeri kontrol uygulanmasının nedeni PID kontrolörden farklı olarak kola uygulanan kalkınma torkunun sınırlandırılmış olmasıdır. Bu sayede motorun fazla enerji tüketimi engellenmiş ve aynı zamanda esnek kolun uç sapmasının ani torktan dolayı çok yüksek değerlere çıkması önlenmiş, esneme miktarı azaltılmış olur. Hata değeri azaldığında ise kolun istenilen referans değerine daha çabuk ulaşması için kontrolör kazançları artırılır. Bu durum esnek robot kollarının kontrolü için PID kontrolörden daha iyi bir performans sağlar. Ancak doğrusal olmayan PID kontrolör kullanılırken kola uygulanması gereken tork sınırlandırıldığı için kolun referans değerini yakalama süresi uzamaktadır.

Uygulanan kontrol yöntemlerinin kolun esnekliğini de içerecek şekilde yeniden tasarlanması amacıyla esnemedi kaynaklanan titreşim miktarını ölçen piezoelektrik seramik sensörler kullanılmış, elde edilen veriler uygun sinyal işleme devrelerinden geçirilerek sisteme geri beslenmiştir. Kolun uç noktası sapmasında meydana gelen hatalar genellikle sürücünün kazanç değerlerinden kaynaklanmaktadır. Sürücünün kazanç değerleri belli bir aralıktaki referans

değerlerine tepki vermemektedir. Bu da kolun uç noktasında meydana gelen pozisyon hatasının artmasına neden olmaktadır. Sürücü kazanç değerleri artırılıp tepki vermediği aralık daraltıldığında ise yüksek kazanç değeri nedeniyle salınımlar artmakta ve uç noktası geç sönümlenmektedir. Bu amaçla hem salınımları azaltıp hem de uç noktası sapmasını en aza indirecek şekilde sürücü kazanç değerleri deneme yanılma yöntemiyle bulunmuştur.

Farklı malzemelerden yapılmış çeşitli uzunluk ve ağırlıktaki kollar kullanılarak yapılan çalışmalarda aynı kontrol ve kontrolör kazanç değerleri uygulandığında farklı özelliklerdeki kolların farklı tepkiler verdikleri görülmektedir. Belirlenen kontrolör kazanç değerleri bir kol için en uygun performansı sağlarken bir diğeri için aynı tepkiyi vermeyebilmektedir. Bu nedenle kontrolör kazanç değerleri belirlenirken her özellikteki kol için kolun fiziksel yapısı dikkate alınmalı ve deneme yanılma metoduyla en uygun kazanç değerleri seçilmelidir. Deneysel çalışma sırasında bir kolla aynı kontrolör ve aynı referans değeri için veriler alındığında elde edilen sonuçlar her zaman birbirinin aynı olmayabilmektedir. Bunun en önemli nedeni kurulan deneysel düzeneğin mekanik kısmının yeterince hassas monte edilememesi ve kolun başlangıçtaki pozisyon değerinin manuel olarak ayarlanması gösterilebilir.

Bu tezde yapılan çalışma ile farklı fiziksel yapıdaki tek ve iki eklemli esnek robot kolları için geliştirilen kontrol yöntemlerinin uygulaması incelenmiştir. Esnek kolun titreşimini ölçen piezoelektrik sensörlerle daha hassas ölçüm yapıp elde edilen veriler kontrolöre girdi olarak geri beslenmiştir. Bu bilgi kontrolöre oransal olarak ilave edilmiştir. Yapılan çalışma sonucu elde edilen verilere göre, kontrolöre ilave edilen PZT bilgisi ile sistemin kontrolü daha kısa sürede sağlanmıştır.

Yapılan çalışma ileride yapılacak olan deneysel çalışmalar için bir temel oluşturmaktadır. İlerideki çalışmalarda farklı kontrol yöntemleri uygulanarak çok daha farklı yapısal özellikteki esnek robot kollarının kontrolü incelenebilecektir. Ayrıca ilerideki çalışmalarda esnek kolun esnemesinden kaynaklanan titreşim bilgisi piezoelektrik seramik sensörler yerine gerilme ölçerler kullanılarak da elde edilebilir. Elde edilen sonuçlar PZT ile elde edilen sonuçlarla karşılaştırılabilir.

KAYNAKLAR

1. Q. Wei, *Modeling and Control of Dynamical Effects due to Impact on Flexible Structures*, Ph.D Thesis, Faculty of Graduate School of the University of Maryland, 1994.
2. Y. Sakawa, F. Matsuno ve S. Fukushima, "Modeling and Feedback Control of a Flexible Arm," *J. Robotic Syst.*, vol. 2, no. 4, pp. 453-472, 1985.
3. S. Yurkovich ve F. E. Pacheco, "On Control Tuning for a Flexible-Link Manipulator with Varying Payload," *J. Robotic Syst.*, vol. 6, no. 3, pp. 233-254, 1989.
4. D. M. Rovner ve R. H. Cannon, "Experiments Toward On-Line Identification and Control of a Very Flexible One-Link Manipulator," *Int. Journal of Robotics*, vol. 6, no. 4, 1987.
5. Z. Luo ve B. Guo, "Shear Force Feedback Control of a Single-Link Flexible Robot with a Revolute Joint," *IEEE Trans. Automatic Cont.*, vol. 42, no. 1, pp. 53-64, 1997.
6. A. Doğan, *İki Eklemlı Esnek Bir Robot Kolumun Modellenmesi ve Kontrolü*, Yüksek Lisans Tezi, Anadolu Üniversitesi Fen Bilimleri Enstitüsü, Eskişehir, 1997.
7. G.A. Kantor, *Linear Control Theory as Applied to Smart Structures*, Ph.D Thesis, Faculty of Graduate School of the University of Maryland, 1995.

8. M. Bai, D. H. Zhou ve H. Schwarz, "Adaptive Augmented State Feedback Control for an Experimental Planar Two-Link Flexible Manipulator," *IEEE Trans. Robotics and Auto.*, vol. 14, no. 6, 1998.
9. I. Salmatjidis, "Impact and Force Control of Flexible Manipulators," Ms.Thesis , Faculty of Graduate School of the University of Maryland, 1991.
10. G. G. Hastings ve W. J. Book, "A Linear Dynamic Model for Flexible Robotic Manipulators," *IEEE Robotics and Auto. Conf*, California, 1986.
11. V. Feliu, K. S. Rattan ve H. B. Brown, "Adaptive Control of a Single-Link Flexible Manipulator," *IEEE Conf. Rob. Auto.* Arizona, 1989.
12. A. Doğan, K. Uchino ve R. E. Newnham, "Flextensional Composite Transducers: Designing, Fabrication and Application," *Technology and Applications*, pp. 357-374, Netherland, 2000.

EK 1

Tek eklemli esnek robot kolu için geliştirilen PID kontrol programı

```

/*****
/*****Tek eklemlı esnek robot kolu için geliştirilen *****/
/*****PID kontrol programı*****/
/*****

```

```

#include<stdio.h>
#include<dos.h>
#include<conio.h>
#include<ctype.h>
#include<stdlib.h>
#include <alloc.h>
#include <time.h>

```

```

#if defined(_MSC_VER) | defined(_QC)
#include <graph.h>

```

```

#define outportb(port,data) outp(port,data)
#define inportb(port) inp(port)
#define inport(port) inpw(port)
#define gotoxy( col, row) _settextposition(row,col)
#define clrscr() _clearscreen( _GCLEARSCREEN)
#define cursor_off() _displaycursor( _GCURSOROFF)
#define cursor_on() _displaycursor( _GCURSORON)
#else
#define cursor_off() _setcursortype( _NOCURSOR)
#define cursor_on() _setcursortype( _NORMALCURSOR)
#endif

```

```

unsigned int Base = 0x300;

```

```

int far *DecToGray;
int far *GrayToDec;

```

```

char st[4096][12];

```

```

int Bintodec (char *s)
{
int res, i, e;

res=0; e=1;
for (i=0;i<12;i++) {
if(s[i]=='1') res+=e;

```

```

    e*=2;
}
return res;
}

void init()
{
    int digit, index, i, k, e;
    char arr[4]={'0','1','1','0'};

    DecToGray=(int far *) farmalloc(4096*sizeof(int));
    GrayToDec=(int far *) farmalloc(4096*sizeof(int));

    e=1;
    for (digit=0;digit<12;digit++) {
        for (i=0;i<4096/(e*4);i++)
            for (index=0;index<4;index++)
                for (k=0;k<e;k++)
                    st[i*e*4+index*e+k][digit]=arr[index];
        e*=2;
    }

    for (i=0;i<4096;i++)
        *(DecToGray+i)=Bintodec(st[i]);

    for (i=0; i<4096; i++) {
        k=0;
        while (i!=*(DecToGray+k) && k<4096) k++;
        *(GrayToDec+i)=k;
    }
}

void Init1800()
{
    unsigned char cId1, cId2 ;

    cId1 = inportb(Base+30);    /* get the pcl-1800 ID code    */
    cId2 = inportb(Base+30);
    if(((cId1 != 0) || (cId2 != 0x18)) && ((cId1 != 0x18) || (cId2 != 0))) {
        printf("PCL-1800 initial failure !\n");
        printf("Please set base address to %Xh and retry ...",Base);
        exit(1);
    }
}

float ReadVoltage(){
    unsigned int wChannel, wGainCode, wData;
    unsigned char cId1, cId2 ;

```

```

float fFullScale, fVoltage ;
char* sBound ;

    wChannel = 0;
    outportb(Base+2, (wChannel << 4 ) | wChannel); /* set start/stop channel*/

wGainCode = 0 ;

    wGainCode = 0x08 ;
    fFullScale = 20 ;
    sBound= "ñ 10 V " ;

gotoxy(1,9);
    printf("Gain Code = %s",sBound);
gotoxy(1,14);
outportb(Base+9,0xa4); /* set to software trigger */
outportb(Base+21,0); /* set to normal mode */
outportb(Base+22,0); /* not to generate any interrupt */
outportb(Base+23,0); /* disable dual DMA */
outportb(Base+24,0); /* base frequency to 1 mHz */
outportb(Base+30,0); /* reset FIFO */
outportb(Base+1,wGainCode); /* setting the gain */

gotoxy(1,15);
printf("Please press any key to stop ....");
cursor_off() ;

    outportb(Base,0); /* Software trigger */
    do{ } while ((inportb(Base+8) & 0x80) != 0) ; /* check A/D EOC */
    wData = inport(Base); /* get A/D data */
    fVoltage = fFullScale / 0xffff * ((int)(wData >> 4) - 0x800) ;
    gotoxy(1,11);
    printf("Reading from A/D converter data = %4X (HEX)\n",wData);
    printf("The Voltage on channel %d is = %2.4f V ",
        wData & 15 ,fVoltage) ;
    return(fFullScale / 0xffff * ((int)(wData >> 4) - 0x800));
}

void SendData(float Data)
{
    float Vout;
    int MSB, LSB;

    Vout=Data*409.6;
    MSB=(Vout/16);
    LSB=(Vout-16*MSB);
    outportb(Base+4,LSB);
}

```

```

    outportb(Base+5,MSB);
}

void ReadnConvert()
{
    float fFullScale, fVoltage, Vref, Vout ;
    int fir;
    char* sBound ;
}

/* save a to the file */
float Process(float derece, float voltage, float derece1, float refe)
{
    float a;

    gotoxy(1,7); printf("process output = %f V",voltage);

    gotoxy(1,5); printf("process derece = %f °",derece); /* < */

    derece1=(refe-derece)+derece1;

    return (((((refe-derece)*0.06510520)+derece1*0.18)-(voltage)*0.01 +5)); /*
example process */
                                /* return(5);*/
}

float ReadDegree()
{
    int base0,base1;
    int a,b;

    base0=(inport(Base+3));
    base1=(inport(Base+11)&0x0F)*0xFF;
    b=base0+base1;
    return GrayToDec[b]*0.087890625;
}

void main()
{
    float degree, voltage, outvoltage,dereceski,derece1;
    float refe;
    int FINISH=0;
    char tus;
    FILE *out1,*out2,*out3;

```

```

clock_t bir,iki,fark;
clrscr();
printf("reference derecesini giriniz: ");
scanf("%f",&refe);

init(); /* initialize the GrayToDec & DecToGrey arrays */
Init1800(); /* initialize the DAC */
if ((out1=fopen("DEGREE1.DAT","w+"))==NULL)
printf("Dosya olusturmada hata olustu" );
if ((out2=fopen("VOLTAGE.DAT","w+"))==NULL)
printf("Dosya olusturmada hata olustu" );
if ((out3=fopen("OUTVOLTAGE.DAT","w+"))==NULL)
printf("Dosya olusturmada hata olustu" );
degree=0;
voltage=0;
/* dereceski=0;*/
outvoltage=0;
clrscr();
bir=clock();
do {
    dereceski=degree;
    degree=ReadDegree();
    if(degree>=359)
    degree=dereceski;
    derece1=0;
    voltage=ReadVoltage();
    outvoltage=Process(degree, voltage,derece1,refe);
    fprintf (out1, "%3.9f \n",degree);
    fprintf (out2, "%3.9f \n",voltage);
    fprintf (out3, "%3.9f \n",outvoltage-5);
    SendData(outvoltage);
gotoxy(1,1); printf("outvolt bilgisi = %f V",outvoltage-5);
gotoxy(1,3); printf("analog voltage bilgisi = %f V",voltage);

    if (kbhit()) {
        tus=getch();
        switch(toupper(tus)) {
            case 27 : FINISH=1;
        }
    }
} while(!FINISH);
fclose(out1);
fclose(out2);
fclose(out3);
iki=clock();
fark=(iki-bir)/CLK_TCK;
gotoxy(1,10);
printf("Sure=%ld saniye \n",fark);

```

```
farfree(DecToGray);  
farfree(GrayToDec);  
} //end of the program
```

EK 2

Tek eklemlil esnek robot kolu için uygulanan doğrusal olmayan PID benzeri kontrol yöntemi için geliştirilen program


```

/*****
/*Tek eklemli esnek robot kolu için uygulanan doğrusal olmayan PID kontrol */
/*****yöntemi için geliştirilen program*****/
/*****/

```

```

#include<stdio.h>
#include<dos.h>
#include<conio.h>
#include<ctype.h>
#include<stdlib.h>
#include <alloc.h>
#include <time.h>

```

```

#if defined(_MSC_VER) | defined(_QC)
#include <graph.h>

```

```

#define outportb(port,data) outp(port,data)
#define inportb(port) inp(port)
#define inport(port) inpw(port)
#define gotoxy( col, row) _settextposition(row,col)
#define clrscr() _clearscreen(_GCLEARSCREEN)
#define cursor_off() _displaycursor(_G_CURSOROFF)
#define cursor_on() _displaycursor(_G_CURSORON)

```

```

#else
#define cursor_off() _setcursortype(_NOCURSOR)
#define cursor_on() _setcursortype(_NORMALCURSOR)
#define one 0
#define two 1

```

```

#endif

```

```

unsigned int Base = 0x300;
float first,second;

```

```

int far *DecToGray;
int far *GrayToDec;

```

```

char st[4096][13];

```

```

int Bintodec (char *s)

```

```

{
int res, i, e;

```

```

res=0; e=1;

```

```

for (i=0;i<12;i++) {
    if(s[i]=='1') res+=e;
    e*=2;
}
return res;
}

void init()
{
    int digit, index, i, k, e;
    char arr[4]={'0','1','1','0'};

    DecToGray=(int far *) farmalloc(4096*sizeof(int));
    GrayToDec=(int far *) farmalloc(4096*sizeof(int));
    for (i=0; i<4096; i++) st[i][12]=0;
    e=1;
    for (digit=0;digit<12;digit++) {
        for (i=0;i<4096/(e*4);i++)
            for (index=0;index<4;index++)
                for (k=0;k<e;k++)
                    st[i*e*4+index*e+k][digit]=arr[index];
        e*=2;
    }

    for (i=0; i<2048; i++) st[i][11]='0';
    for (i=2048; i<4096; i++) st[i][11]='1';

    for (i=0;i<4096;i++)
        *(DecToGray+i)=Bintodec(st[i]);

    for (i=0; i<4096; i++) {
        k=0;
        while (i!=*(DecToGray+k) && k<4096) k++;
        *(GrayToDec+i)=k;
    }
}

void Init1800()
{
    unsigned char cId1, cId2 ;

    cId1 = inportb(Base+30);    /* get the pcl-1800 ID code    */
    cId2 = inportb(Base+30);
    if(((cId1 != 0) || (cId2 != 0x18)) && ((cId1 != 0x18) || (cId2 != 0))) {
        printf("PCL-1800 initial failure !\n");
        printf("Please set base address to %Xh and retry ...",Base);
    }
}

```

```

    exit(1);
}
}

```

```

float ReadVoltage(int Channel){
    unsigned int wChannel, wGainCode, wData;
    unsigned char cId1, cId2 ;
    float fFullScale, fVoltage ;

```

```

    wGainCode = 0 ;
    wGainCode = 0x08 ;
    fFullScale = 20 ;

```

```

    outportb(Base+9,0xa4);    /* set to software trigger    */
    outportb(Base+21,0);     /* set to normal mode        */
    outportb(Base+22,0);     /* not to generate any interrupt */
    outportb(Base+23,0);     /* disable dual DMA          */
    outportb(Base+24,0);     /* base frequency to 1 mHz    */
    outportb(Base+30,0);     /* reset FIFO                 */
    outportb(Base,0);        /* Software trigger */

```

```

switch(Channel){
case one:{

```

```

    outportb(Base+1,wGainCode); /* setting the gain */
    outportb(Base+2, 0x00); /* set start/stop channel*/
    break;}

```

```

case two:{

```

```

    outportb(Base+1,wGainCode); /* setting the gain    */
    outportb(Base+2, 0x11); /* set start/stop channel*/
    break;}

```

```

}

```

```

    do{ } while ((inportb(Base+8) & 0x80) != 0) ; /* check A/D EOC    */
    wData = inport(Base); /* get A/D data    */
    fVoltage = fFullScale / 0xffff * ((int)(wData >> 4) - 0x800) ;

```

```

switch(Channel){

```

```

case one: return (fFullScale / 0xffff * ((int)(wData >> 4) - 0x800));
case two: return (fFullScale / 0xffff * ((int)(wData >> 4) - 0x800));
}

```

```

return(fFullScale / 0xffff * ((int)(wData >> 4) - 0x800));
}

```

```

float tacho(){

```

```

    float first;

```

```

    first=ReadVoltage(one);

```

```

gotoxy(1,5);
printf("The Voltage of the tacho is = %2.4f V ", first) ;

return first;
}

float strain(){
float second;
second=ReadVoltage(two);
gotoxy(1,7);
printf("The Voltage of the strain is = %2.4f V ", second) ;
return second;
}
void SendData(float Data)
{
float Vout;
int MSB, LSB;

Vout=Data*409.6;
MSB=(Vout/16);
LSB=(Vout-16*MSB);
outportb(Base+4,LSB);
outportb(Base+5,MSB);
}

float Process(float derece, float voltage, float derece1, float refe, float first, float
second, int encol)
{
float a;
float k1,k2;

gotoxy(1,3); printf("process derece = %f",derece); /* ◇ */
gotoxy(1,1); printf("process outvoltage = %f",((((refe-
derece)*0.06510520)+derece1*0.18)-(first)*0.01 ));

gotoxy(1,12); printf("refe-derece = %f",refe-derece); /* ◇ */

/* ◇ */
encol=abs(refe-derece);
if(encol<=10){
k1=0.035;
k2=0.0003495;
encol=encol;
/*gotoxy(1,23);printf("encol : %d",abs(refe-derece));*/
}
else {

```

```

enco1=10;
k1=0.037;
k2=0.00037;
}

gotoxy(1,15); printf("enco1=%d",enco1);

derece1=(refe-derece)+derece1; /* integrator */

/*return (((0.0132095-0.00102598*enco1)*(refe-derece))-
((0.01085+0.000598*enco1)*first)+derece1*0.01425 +5);*/

return (((k1-enco1*k2)*(refe-derece)+derece1*0.025-
(first*(k1+k2*enco1))+((second-1.17)*0.0) +5)); /* example process */

/*return (((((refe-derece)*0.034950)+derece1*0.035)-(first)*0.025+((second-
1.17)*0.0) +5));*/ /* example process */

}

float ReadDegree()
{
int base0,base1;
int a,b;

base0=(inport(Base+3));
base1=(inport(Base+11)&0x0F)*0xFF;
b=base0+base1;
return GrayToDec[b]*0.087890625;
}

void main()
{
float degree, voltage, outvoltage,dereceski,derece1;
float refe,first,second,enco;
int FINISH=0;
char tus;
FILE *out1,*out2,*out3;
clock_t bir,iki,fark;
clrscr();
printf("reference derecesini giriniz: ");
scanf("%f",&refe);

init(); /* initialize the GrayToDec & DecToGrey arrays */
Init1800(); /* initialize the DAC */
if ((out1=fopen("DEGREE1.DAT","w+"))==NULL)
printf("Dosya olusturmada hata olustu" );

```

```

if ((out2=fopen("VOLTAGE.DAT","w+"))==NULL)
printf("Dosya olusturmada hata olustu" );
if ((out3=fopen("OUTVOLTAGE.DAT","w+"))==NULL)
printf("Dosya olusturmada hata olustu" );
degree=0;
voltage=0;
/* dereceski=0;*/
outvoltage=0;
clrscr();
bir=clock();
do {
    dereceski=degree;
    degree=ReadDegree();
    if(degree==239.062500)
        degree=dereceski;
    derecel=0;

first=tacho();
second=strain();
outvoltage=Process(degree, voltage,derecel,refe,second,first,enco);
    fprintf (out1,"%3.9f\n",degree);
    fprintf (out2,"%3.9f\n",refe-degree);
    fprintf (out3,"%3.9f\n",first);
    SendData(outvoltage);

    if (kbhit() {
        tus=getch();
        switch(toupper(tus)) {
            case 27 : FINISH=1;
        }
    }
} while(!FINISH);
fclose(out1);
fclose(out2);
fclose(out3);
iki=clock();
fark=(iki-bir)/CLK_TCK;
gotoxy(1,10);
printf("Sure=%ld saniye \n",fark);
farfree(DecToGray);
farfree(GrayToDec);
} //end of the program

```

EK 3

İki eklemlı esnek robot kolu için uygulanan doğrusal olmayan PID benzeri kontrol yöntemi için geliştirilen program

```

/*****
/*İki eklemli esnek robot kolu için uygulanan doğrusal olmayan PID benzeri */
/*****kontrol yöntemi için geliştirilen program*****/
/*****/

```

```

#include<stdio.h>
#include<dos.h>
#include<conio.h>
#include<ctype.h>
#include<stdlib.h>
#include <alloc.h>
#include <time.h>

```

```

#if defined(_MSC_VER) | defined(_QC)
#include <graph.h>

```

```

#define outportb(port,data) outp(port,data)
#define inportb(port) inp(port)
#define inport(port) inpw(port)
#define gotoxy( col, row) _settextposition(row,col)
#define clrscr() _clearscreen(_GCLEARSCREEN)
#define cursor_off() _displaycursor(_G_CURSOROFF)
#define cursor_on() _displaycursor(_G_CURSORON)

```

```

#else

```

```

#define cursor_off() _setcursortype(_NOCURSORS)
#define cursor_on() _setcursortype(_NORMALCURSOR)
#define one 0
#define two 1

```

```

#endif

```

```

unsigned int Base = 0x300;
unsigned int Base1 = 0x200;
float first,second;

```

```

int far *DecToGray;
int far *GrayToDec;

```

```

char st[4096][13];

```

```

int Bintodec (char *s)
{
int res, i, e;

```



```

res=0; e=1;
for (i=0;i<12;i++) {
    if(s[i]=='1') res+=e;
    e*=2;
}
return res;
}

void init()
{
    int digit, index, i, k, e;
    char arr[4]={'0','1','1','0'};

    DecToGray=(int far *) farmalloc(4096*sizeof(int));
    GrayToDec=(int far *) farmalloc(4096*sizeof(int));
    for (i=0; i<4096; i++) st[i][12]=0;
    e=1;
    for (digit=0;digit<12;digit++) {
        for (i=0;i<4096/(e*4);i++)
            for (index=0;index<4;index++)
                for (k=0;k<e;k++)
                    st[i*e*4+index*e+k][digit]=arr[index];
        e*=2;
    }

    for (i=0; i<2048; i++) st[i][11]='0';
    for (i=2048; i<4096; i++) st[i][11]='1';

    for (i=0;i<4096;i++)
        *(DecToGray+i)=Bintodec(st[i]);

    for (i=0; i<4096; i++) {
        k=0;
        while (i!=*(DecToGray+k) && k<4096) k++;
        *(GrayToDec+i)=k;
    }
}

void Init1800()
{
    unsigned char cId1, cId2 ;

    cId1 = inportb(Base+30);    /* get the pcl-1800 ID code */
    cId2 = inportb(Base+30);
    if(((cId1 != 0) || (cId2 != 0x18)) && ((cId1 != 0x18) || (cId2 != 0))) {

```

```

printf("PCL-1800 initial failure !\n");
printf("Please set base address to %Xh and retry ...",Base);
exit(1);
}
}

float ReadVoltage(int Channel){
    unsigned int wChannel, wGainCode, wData;
    unsigned char cId1, cId2 ;
    float fFullScale, fVoltage ;

    wGainCode = 0 ;
    wGainCode = 0x08 ;
    fFullScale = 20 ;

    outportb(Base+9,0xa4);    /* set to software trigger */
    outportb(Base+21,0);     /* set to normal mode */
    outportb(Base+22,0);     /* not to generate any interrupt */
    outportb(Base+23,0);     /* disable dual DMA */
    outportb(Base+24,0);     /* base frequency to 1 mHz */
    outportb(Base+30,0);     /* reset FIFO */
    outportb(Base,0);        /* Software trigger */

    switch(Channel){
    case one:{

        outportb(Base+1,wGainCode); /* setting the gain */
        outportb(Base+2, 0x00); /* set start/stop channel*/
        break;}
    case two:{
        outportb(Base+1,wGainCode); /* setting the gain */
        outportb(Base+2, 0x11); /* set start/stop channel*/
        break;}

    }

    do{ } while ((inportb(Base+8) & 0x80) != 0) ; /* check A/D EOC */
    wData = inport(Base); /* get A/D data */
    fVoltage = fFullScale / 0xffff * ((int)(wData >> 4) - 0x800) ;

    switch(Channel){
    case one: return (fFullScale / 0xffff * ((int)(wData >> 4) - 0x800));
    case two: return (fFullScale / 0xffff * ((int)(wData >> 4) - 0x800));
    }

    return(fFullScale / 0xffff * ((int)(wData >> 4) - 0x800));
}

float tacho(){

```

```

float first;
first=ReadVoltage(one);
/* gotoxy(1,9);
printf("The Voltage of the 1.tacho is = %2.4f V ", first);*/

return first;
}

float strain(){
float second;
second=ReadVoltage(two);
/* gotoxy(1,7);
printf("The Voltage of the strain is = %2.4f V ", second);*/
return second;
}

void SendData(float Data)
{
float Vout;
int MSB, LSB;

Vout=Data*409.6;
MSB=(Vout/16);
LSB=(Vout-16*MSB);
outportb(Base+4,LSB);
outportb(Base+5,MSB);
}

void SendData1(float Data)
{

float Vout;
int MSB, LSB;

Vout=Data*409.6;
MSB=(Vout/16);
LSB=(Vout-16*MSB);
outportb(Base1+4,LSB);
outportb(Base1+5,MSB);
}

float Process(float derece, float voltage,float derece1,float refe,float first,float
second,float enco1)
{
float a;
float k1,k2;

```

```

gotoxy(1,3); printf("1.process derece = %f \0",derece+1); /* < */
/*gotoxy(1,4); printf("process outvoltage = %f",(((refe-
derece)*0.06510520)+derece1*0.18)-(first)*0.01 ));*/

/*gotoxy(1,5); printf("refe-derece = %f \0",refe-derece); */

/* < */
enco1=abs(refe-derece);
if(enco1<=10){
k1=0.0535;
k2=0.003495;
enco1=enco1;
/*gotoxy(1,23);printf("enco1 : %d",abs(refe-derece));*/
}
else {

enco1=10;
k1=0.0437;
k2=0.0037;
}

/*gotoxy(1,6); printf("enco1=%d",enco1);*/

derece1=(refe-derece)+derece1; /* integrator */
return (((k1-enco1*k2)*(refe-derece)+derece1*0.0325-
(second*(k1+k2*enco1))+((second-1.17)*0.0) +5)); /* example process */

}

float Process1(float derece, float voltage,float derece1,float refe,float enco1)
{
float a;
float k1,k2;

gotoxy(1,13); printf("2.process derece = %f \0",derece-1); /* < */
/*gotoxy(1,14); printf("process outvoltage = %f",(((refe-
derece)*0.06510520)+derece1*0.18));*/
/*gotoxy(1,15); printf("refe-derece = %f \0",refe-derece); */

/* < */

enco1=abs(refe-derece);
if(enco1<=10){
k1=0.01035;
k2=0.0003495;
enco1=enco1;

```

```

}
else {
enco1=10;
k1=0.01037;
k2=0.00037;
}

/*gotoxy(1,16); printf("enco1=%d",enco1);*/

derece1=(refe-derece)+derece1; /* integrator */
return (((k1-enco1*k2)*(refe-derece)+derece1*0.0120-
(first*0.0*(k1+k2*enco1))+((second-1.17)*0.0)+5)); /* example process */

}

float ReadDegree()
{
int base0,base1;
int a,b;

base0=(inport(Base+3));
base1=(inport(Base+11)&0x0F)*0xFF;
b=base0+base1;
return GrayToDec[b]*0.087890625;
}

float ReadDegree1()
{
int base0,base1;
int a,b;

base0=(inport(Base1+3));
base1=(inport(Base1+11)&0x0F)*0xFF;
b=base0+base1;
return GrayToDec[b]*0.087890625;
}

void main()
{
float degree, voltage, outvoltage,dereceski,derece1;
float dereceski1,voltage1,degree1,derece11;
float outvoltage1;
float refe,first,second,enco1,refe1,enco;
int FINISH=0;
char tus;
FILE *out1,*out2,*out3,*out4;

```

```

clock_t bir,iki,fark;
clrscr();

printf("1.kol icin reference derecesini giriniz: ");
scanf("%f",&refe);
printf("2.kol icin reference derecesini giriniz: ");
scanf("%f",&refe1);

init(); /* initialize the GrayToDec & DecToGrey arrays */
Init1800(); /* initialize the DAC */
if ((out1=fopen("DEGREE1.DAT","w+"))==NULL)
printf("Dosya olusturmada hata olustu" );
if ((out4=fopen("DEGR1.DAT","w+"))==NULL)
printf("Dosya olusturmada hata olustu" );
if ((out1=fopen("DEGREE1.DAT","w+"))==NULL)
printf("Dosya olusturmada hata olustu" );

if ((out2=fopen("VOLTAGE.DAT","w+"))==NULL)

printf("Dosya olusturmada hata olustu" );
if ((out3=fopen("OUTVOLTAGE.DAT","w+"))==NULL)
printf("Dosya olusturmada hata olustu" );
degree=0;
voltage1=0;
voltage=0;
/* dereceski=0;*/
outvoltage=0;
outvoltage1=0;
clrscr();
bir=clock();
do {
    dereceski=degree;
    degree=ReadDegree();
    if(degree==239.062500)
    degree=dereceski;
    derece1=0;

    dereceski1=degree1;
    degree1=ReadDegree1();
    if(degree1==239.062500)
    degree1=dereceski1;
    derece11=0;

first=tacho();
second=strain();
outvoltage=Process(degree, voltage,derece1,refe,second,first,enco);
outvoltage1=Process1(degree1,voltage,derece11,refe1,enco1);

```

```
fprintf (out1, "%3.9f\n", degree+1);
    fprintf (out4, "%3.9f\n", degree1-1);
fprintf (out2, "%3.9f\n", refe-degree);
fprintf (out3, "%3.9f\n", outvoltage-5);
SendData(outvoltage);
SendData1(outvoltage1);
if (kbhit()) {
    tus=getch();
    switch(toupper(tus)) {
        case 27 : FINISH=1;
    }
}
} while(!FINISH);
fclose(out1);
fclose(out2);
fclose(out3);
iki=clock();
fark=(iki-bir)/CLK_TCK;
gotoxy(1,10);
printf("Sure=%ld saniye \n",fark);
farfree(DecToGray);
farfree(GrayToDec);
} //end of the program
```