

ROBOT MANİPÜLATÖR DENETİMİ

Hikmet KOCA

**YÜKSEK LİSANS TEZİ
ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ**

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**EKİM 2007
ANKARA**

Hikmet KOCA tarafından hazırlanan ROBOT MANİPÜLATÖR DENETİMİ adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Prof.Dr.M.Cengiz TAPLAMACIOĞLU
Elektrik-Elektronik Müh. Anabilim Dalı

Bu çalışma, jürimiz tarafından oy birliği ile Elektrik-Elektronik Mühendisliği Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiştir.

Yrd.Doç.Dr. Mustafa DOĞAN
Elektrik-Elektronik Müh., Başkent Üni.
Prof.Dr.M. Cengiz TAPLAMACIOĞLU
Elektrik-Elektronik Müh., Gazi Üni.
Yrd.Doç.Dr. Nursel Akçam
Elektrik-Elektronik Müh., Gazi Üni.

Tarih: 04/10/2007

Bu tez ile G.Ü. Fen Bilimleri Enstitüsü Yönetim Kurulu Yüksek Lisans derecesini onamıştır.

Prof. Dr. Nermin ERTAN
Fen Bilimleri Enstitüsü Müdürü

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

Hikmet KOCA

ROBOT MANİPÜLATÖR DENETİMİ
(Yüksek Lisans Tezi)

Hikmet KOCA

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
Ekim 2007

ÖZET

Robot manipülatörün eksen sayısının artmasıyla birlikte, robotun hareket yeteneği de artmaktadır, fakat bununla beraber robot manipülatörün denetimi de zorlaşmakta ve denetim algoritması daha karmaşık hale gelmektedir. Bu çalışmada endüstride kullanılan eksen-ölgül ve kartezyen-ölgül robot manipülatör denetim yöntemleri uygulanmıştır. Yapılan eksen-ölgül çalışmada altı eksenli robot manipülatörün bir manevra kolu ile basit ve ekonomik olarak denetimi amaçlanmaktadır. Robot manipülatörün denetimi için tasarlanan sayısal devre, genel amaçlı FPGA (Alan programlanabilir kapı dizileri) tabanlı bir geliştirme kartı üzerinde Verilog donanım tanımlama dili kullanılarak gerçekleştirilmiştir. Geliştirilen sistem manevra kolundan alınan giriş sinyallerini yorumlayarak denetlenmek istenilen manipülatör eksenine ait motoru sürmek için gerekli PWM (Darbe genişlik modülasyonu) sinyalini üretir.

Kartezyen-özgöl çalışmada DSP (Sayısal işaret işleme) uygulamaları için geliştirilmiş bir FPGA kullanılarak üç eksenli robotun kartezyen denetimi gerçekleştirilmiştir. Geliştirilen sistem manevra kolundan gelen sinyalleri yorumlar ve robotun hangi yönde hareket edeceğini, yeni x, y değerlerini bulur ve çözümlenmesi için kinematik bloğuna gönderir. Kinematik bloğunun içerisinde, robotun ucunun konumlanması istenilen x, y değerleri için gerekli olan eklem açıları robotun ters kinematik denklemlerinin çözülmesi ile elde edilir. Bulunan eklem açıları robotun düz kinematik denklemlerinde yerine koyularak elde edilen sonuçların robotun çalışma alanı içinde olup olmadığı denetlenir. Eğer elde edilen sonuçlar robotun çalışma alanı içinde ise robotun eklemlerinin istenilen koordinatlarda konumlanması için gerekli açılar bulunmuş olur. Bulunan açılar, açı-sinyal genişliği bloğuna gönderilir. Açı-sinyal genişliği bloğu girişindeki açı değerlerini okur ve robotun eklemlerinde yer alan servo motorların bu açılarda konumlanması için gerekli olan darbe genişliklerini hesaplar ve bu genişlik değerlerini ilgili eksen denetleyicilerine gönderir. Eksen denetleyicileri girişlerindeki sinyal genişlik değerlerini okurlar ve bu genişlik değerlerine sahip kare dalgaları çıkışlarında üretirler. Bu sinyaller robotun eklemlerinde yer alan motorların kontrol uçlarına gönderilir, robotun istenilen koordinatlarda konumlanması sağlanır ve sistemin çözümü doğru ve hızlı bir şekilde elde edilmiş olur.

Bilim Kodu : 905.1.096
Anahtar Kelimeler : Robot manipülatör, FPGA, manevra kolu, denetim
Sayfa Adedi : 108
Tez Yöneticisi : Prof. Dr. M. Cengiz TAPLAMACIOĞLU

ROBOT MANIPULATOR CONTROL**(M.Sc. Thesis)****Hikmet KOCA****GAZİ UNIVERSITY****INSTITUTE OF SCIENCE AND TECHNOLOGY****October 2007****ABSTRACT**

As the number of robot joints increases in parallel with the robot dexterity. But, control of the robot manipulator also gets difficult and complex. In this study, axis-specific and cartesian-specific robot manipulator control methods are implemented which are common well-known methods in industry. In the axis-specific part of the research, the control of a six-axes robot manipulator with a joystick is achieved with simplicity and efficiency. The designed digital circuit for controlling the robot manipulator is implemented on a general purpose FPGA board with Verilog hardware description language. The developed system comments the input signals which are taken from the joystick then generate the necessary PWM signal to drive the related axis motor. In the cartesian-specific part of the research, cartesian-specific control of a three axes robot manipulator is implemented on a FPGA board which is developed for DSP applications. The developed system evaluates signals which are taken from the joystick, and finds the direction of motion of the manipulator, and finds the new coordinate x , y values, then sends them to the kinematic block. In the kinematic block, joint angles, which are necessary for the desired coordinates (x and y)

values, are obtained by solving inverse kinematic equations of the robot. These obtained joint angles are put into direct kinematics equations of the robot to check if the final gripper position is in the work space of the manipulator. If it is in the work space of the robot manipulator then the required joint angles for achieving the desired motion are kept correctly and rapidly. Thus, obtained joint angles are sent to the angle-pulse width block. The angle-pulse width block reads the input values of the angles, calculates the necessary pulse widths for allocating the servo motors at the desired angles, and then sends the obtained pulse widths to the axis controllers. The axis controllers reads the input values of the pulse widths, and generates necessary PWM signals to drive the corresponding axis motor.

Science Code : 905.1.096

Key Words : Robot manipulator, FPGA, joystick, control

Page Number : 108

Adviser : Prof. Dr. M. Cengiz TAPLAMACIOĞLU

TEŐEKKÜR

Çalıőmalarım boyunca önemli yardım ve katkılarıyla beni yönlendiren Hocam Prof. Dr. M. Cengiz TAPLAMACIOĐLU'na, kıymetli tecrübeleriyle bana yol gösteren Yrd. Doç. Dr. Mustafa DOĐAN Hocama, deđerli yardımlarıyla bana destek veren Hocam Yrd. Doç Dr. Hasan Őakir Bilge'ye ve manevi destekleriyle beni hiçbir zaman yalnız bırakmayan çok deđerli aileme en içten dileklerimle teőekkürü bir borç bilirim.

İÇİNDEKİLER

	Sayfa
ÖZET	iv
ABSTRACT	vi
TEŞEKKÜR.....	viii
İÇİNDEKİLER	ix
ÇİZELGELERİN LİSTESİ.....	xiii
ŞEKİLLERİN LİSTESİ.....	xiv
RESİMLERİN LİSTESİ.....	xvii
SİMGELER VE KISALTMALAR	xviii
1. GİRİŞ	1
2. ENDÜSTRİYEL ROBOTLAR	2
2.1. Endüstriyel Robotun Tanımı	2
2.2. Endüstriyel Robotu Meydana Getiren Kısımlar	2
2.3. Endüstriyel Robotların Tanımlama Parametreleri.....	3
2.4. Manipülatör Yapıları	4
2.4.1. Kartezyen (Cartesian) manipülatör	4
2.4.2. Silindirik (Cylindrical) manipülatör	5
2.4.3. Küresel (Spherical) manipülatör	6
2.4.4. SCARA manipülatör.....	7
2.4.5. Mafsallı (Anthropomorphic) manipülatör	8
2.5. Robotların Kullanıldığı Endüstri Kolları.....	9
2.6. Dünya Robot Marketi İstatistikleri	10

	Sayfa
2.7. Robot Teknolojisinin Türkiye İçin Önemi	12
3. ROBOT MANİPÜLATÖRÜN EKSEN-ÖZGÜL DENETİMİ.....	15
3.1. Giriş.....	15
3.2. Kinematik	16
3.2.1. Katı nesnenin konumu ve yönü	17
3.2.2. Dönme matrisi.....	18
3.2.3. Temel dönmeler	19
3.2.4. Homojen koordinat dönüşümleri	19
3.3. Geliştirilen Sistemin Genel Blok Şeması	21
3.4. Lynx-6 Manipülatörünün Matematiksel Modeli	22
3.5. Lynx-6 Manipülatörünün Özellikleri	26
3.6. Servo Motorlar.....	27
3.7. Manevra Kolu	28
3.8. Denetim Kartı	28
3.9. Geliştirilen Sayısal Tasarım.....	29
3.10. Güç Kaynağı.....	33
4. ROBOT MANİPÜLATÖRÜN KARTEZYEN-ÖZGÜL DENETİMİ	34
4.1. Giriş.....	34
4.2. FPGA (Alan Programlanabilir Kapı Dizileri).....	35
4.2.1. FPGA mimarisi.....	36
4.2.2. Mikroişlemci ile FPGA'nın karşılaştırılması	37
4.2.3. ASIC (uygulamaya özgül tümdevre) ile FPGA'nın karşılaştırılması.....	38

	Sayfa
4.2.4. FPGA ile tasarım.....	38
4.2.5. FPGA ile tasarımda kullanılan araçlar.....	38
4.2.6. FPGA'nın uygulama örnekleri	39
4.3. Geliştirilen Sistemin Genel Blok Şeması	48
4.4. Manipülatörün Matematiksel Modeli.....	49
4.4.1. Ters kinematik.....	51
4.5. Denetim Kartı	51
4.6. Denetleyici Tasarımı.....	52
4.6.1. Giriş denetleyicisi donanımı	54
4.6.2. Kinematik donanımı	56
4.6.2.1. Ters kinematik donanımı.....	57
4.6.2.2. Düz kinematik donanımı.....	61
4.6.2.3. Limit kontrol donanımı.....	65
4.6.2.4. Hata kontrol donanımı.....	69
4.6.3. Açık-sinyal genişliği donanımı.....	80
4.6.4. Eksen denetleyicisi donanımı.....	81
5. SONUÇ VE ÖNERİLER.....	84
KAYNAKLAR	86
EKLER.....	90
EK-1 Eksen özgül denetleme modülünün kaynak kodu	91
EK-2 Giriş denetleyicisi modülünün kaynak kodu (eksen-özgül).....	92
EK-3 Eksen denetleyicisi modülünün kaynak kodu (eksen-özgül)	95
EK-4 Giriş denetleyicisi modülünün kaynak kodu (kartezyen-özgül).....	97
EK-5 Ters kinematik modülünün verilog kodu (kartezyen-özgül)	99
EK-6 Düz kinematik modülünün verilog kodu (kartezyen-özgül).....	100
EK-7 Limit kontrol modülünün verilog kodu (kartezyen-özgül)	101

Sayfa

EK-8 Hata kontrol modülünün verilog kodu (kartezyen-özügöl)	102
EK-9 Kinematik modülün verilog kodu (kartezyen-özügöl).....	104
EK-10 Açı-sinyal genişliđi modülünün verilog kodu (kartezyen-özügöl)	105
EK-11 Eksen denetleyicisi modülünün verilog kodu (kartezyen-özügöl)	106
EK-12 Kartezyen-özügöl denetleyicinin verilog kodu	107
ÖZGEÇMİŞ.....	108

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 2.1. Endüstriyel robotların kurulum ve mevcut kullanılma değerleri.....	14
Çizelge 3.1. Manevra kolu sinyalleri-robotun eksen hareketleri.	28
Çizelge 3.2. Denetleyici sinyalleri.	32

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1. Kartezyen manipülatör	5
Şekil 2.2. Silindirik manipülatör	6
Şekil 2.3. Küresel manipülatör	7
Şekil 2.4. SCARA manipülatör	8
Şekil 2.5. Mafsallı manipülatör	9
Şekil 2.6. Endüstriyel robotların yıllık kurulumu.....	11
Şekil 2.7. Robotların ana endüstri kollarındaki kurulumu (2004-2005).....	12
Şekil 3.1. Katı nesnenin konumu ve yönü.	17
Şekil 3.2. Sistemin blok şeması.	21
Şekil 3.3. Homojen koordinat sistemi atamaları.	22
Şekil 3.4. Lynx-6 robot manipülatörü.....	26
Şekil 3.5. Denetim sinyalinin dalga şekli.	27
Şekil 3.6. Denetim kartı.....	29
Şekil 3.7. Sayısal tasarımın blok şeması.	30
Şekil 3.8. Verilog kodu örneği.	31
Şekil 3.9. Verilog koda karşılık gelen donanım yapısı.....	31
Şekil 3.10. Sistemin giriş ve çıkışlarının dalga şekilleri.	33
Şekil 4.1. Mantık hücresi	36
Şekil 4.2. Şekillendirilebilir uygulama örneği.....	40
Şekil 4.3. Robot manipülatör denetim devresi	43
Şekil 4.4. Geliştirilen denetim sistemi.....	44

Şekil	Sayfa
Şekil 4.5. Kullanılan denetim kartı	44
Şekil 4.6. İnsansı robot	46
Şekil 4.7. Geliştirilen sistem	47
Şekil 4.8. El komutları	48
Şekil 4.9. Sistemin genel blok şeması	49
Şekil 4.10. Robot manipülatörün geometrisi	50
Şekil 4.11. FPGA içindeki sayısal tasarımın blok şeması	54
Şekil 4.12. Giriş denetleyicisi donanımı	55
Şekil 4.13. Giriş denetleyicisi benzetim sonucu 1	55
Şekil 4.14. Giriş denetleyicisi benzetim sonucu 2	56
Şekil 4.15. Kinematik donanımının blok şeması	56
Şekil 4.16. Ters kinematik donanımı	57
Şekil 4.17. Ters kinematik donanımının detaylı yapısı	58
Şekil 4.18. Ters kinematik benzetim sonucu 1	60
Şekil 4.19. Ters kinematik benzetim sonucu 2	60
Şekil 4.20. Düz kinematik donanımı	61
Şekil 4.21. Düz kinematik donanımının detaylı yapısı	62
Şekil 4.22. Düz kinematik benzetim sonucu 1	64
Şekil 4.23. Düz kinematik benzetim sonucu 2	65
Şekil 4.24. Limit kontrol donanımı	65
Şekil 4.25. Limit kontrol donanımının detaylı yapısı	66
Şekil 4.26. Limit kontrol benzetim sonucu 1	68
Şekil 4.27. Limit kontrol benzetim sonucu 2	68

Şekil	Sayfa
Şekil 4.28. Hata kontrol donanımı.....	69
Şekil 4.29. Hata kontrol donanımının detaylı yapısı	70
Şekil 4.30. Hata kontrol benzetim sonucu 1.....	72
Şekil 4.31. Hata kontrol benzetim sonucu 2.....	72
Şekil 4.32. Kinematik donanımı.....	73
Şekil 4.33. Kinematik donanımının detaylı yapısı	75
Şekil 4.34. Kinematik donanımının benzetim sonucu 1	77
Şekil 4.35. Kinematik donanımının benzetim sonucu 2.....	78
Şekil 4.36. Kinematik donanımının benzetim sonucu 3	78
Şekil 4.37. Kinematik donanımının benzetim sonucu 4	79
Şekil 4.38. Kinematik donanımının benzetim sonucu 5.....	80
Şekil 4.39. Açık-sinyal genişliği donanımı	80
Şekil 4.40. Açık-sinyal genişliği donanımı benzetimi	81
Şekil 4.41. Robot denetleyicisi donanımı.....	82

RESİMLERİN LİSTESİ

Resim	Sayfa
Resim 4.1. Mars Pathfinder ve Mars Surveyor.....	41
Resim 4.2. Solucansı robot.....	45
Resim 4.3. Kullanılan denetim kartı.....	46
Resim 4.4. Görme sistemi.....	47
Resim 4.5. Denetim kartı.....	52

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış bazı simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	Açıklama
A	Homojen dönüşüm matrisi
S_g	Sinyal genişliği, mili saniye
θ	Eklem açısı, derece
Kısaltmalar	Açıklama
ASIC	Uygulamaya özel tümleşik devre
CAD	Bilgisayar destekli tasarım
DA	Doğru akım
DSP	Sayısal işaret işleme
FPGA	Alan programlanabilir kapı dizileri
IFR	Uluslararası Robot Federasyonu
NRE	Tasarım, test, prototip ve mühendislik masrafları
PD	Orantılı türev
PID	Orantılı integral türev
PWM	Darbe genişlik modülasyonu

1. GİRİŞ

Endüstriyel robot, deęişken görevleri yerine getirmek için programlanabilen, çok fonksiyonlu, malzeme, para ve eşyaları taşıyabilmek için tasarlanmış bir manipölatördür. Son yıllarda robotik manipölatörler çeşitli endüstrilerde ve deęişik uygulamalarda kullanılmaktadır.

Bu alıřmanın ikinci bölümünde endüstriyel robotların tanımı, kısımları, tanımlama parametreleri, manipölatör yapıları, kullanıldıkları endüstri dalları, market istatistikleri ve Türkiye için öneminden bahsedilmiştir.

Üçüncü bölümde, robotikte kullanılan matematiksel yöntemler hakkında kısa bilgi verildikten sonra alıřmada kullanılan robot manipölatörün eksen-öztöl (axis-specific) denetlemesinden bahsedilmiştir.

alıřmanın dördüncü bölümünde FPGA hakkında genel bilgi ve robotikteki uygulamaları hakkında bilgi verilmiştir. Kullanılan robot manipölatörün kartezyen öztöl (cartesian-specific) denetlemesinden bahsedilmiştir.

2. ENDÜSTRİYEL ROBOTLAR

2.1. Endüstriyel Robotun Tanımı

ISO 8373:1994 tarafından verilen tanıma göre endüstriyel robot: Üç veya daha fazla programlanabilir eksenli olan, otomatik kontrollü, tekrar programlanabilir, çok amaçlı manipülatördür [1].

ANSI/RIA R15.06-1999 Robot Güvenlik Standartına göre endüstriyel robot: Üç veya daha fazla programlanabilir eksenli olan, otomatik kontrollü, tekrar programlanabilir, bir yerde sabit duran veya tekerlekleri olan endüstriyel uygulamalarda kullanılan çok amaçlı manipülatördür [2].

Yukarıdaki tanımlara dayanarak endüstriyel robotik pratik bir şekilde, robot sistemlerinin üretim-imalat için tasarımı ve kullanımı olarak tanımlanabilir.

2.2. Endüstriyel Robotu Meydana Getiren Kısımlar

- Bir mekanik yapı ya da eklemlerle birbirine bağlanmış sıralı katı cisimlerden (uzuvlardan) oluşan manipülatör; serbestliği sağlayan bir koldan (arm), el becerisi sağlayan bir bilekten (wrist) ve robotun yapması gereken görevi tamamlayan uç elemanından (end-effector) oluşmaktadır.
- Eklemlerin hareketlenmesiyle manipülatörün hareketini sağlayan hareketlendiriciler için genellikle elektrikli motorlar, hidrolik ve pnömatik devindiriciler (actuator) kullanılır.
- Manipülatörün durumunu gözleyen algılayıcılar ve gerekirse çevrenin durumunu gözleyen algılayıcılar (sensors).
- Manipülatör hareketini denetleyen ve yöneten bir denetim sistemi (computer) [3].

2.3. Endüstriyel Robotların Tanımlama Parametreleri

- Eksen Sayısı (Number of Axes): Bir düzlemde herhangi bir noktaya ulaşmak için iki eksen gereklidir; uzayda herhangi bir noktaya ulaşmak için üç eksen gereklidir. Robotun uç elemanının yön kontrolü için üç ya da daha fazla eksene ihtiyaç vardır.
- Serbestlik Derecesi (Degrees of Freedom): Genellikle eksenler ile aynı sayıdadır. Robotun ucunun uzayda istenilen pozisyonda konumlanması için üç, istenilen yönde konumlanması için ise üç yani toplam altı serbestlik derecesine ihtiyaç vardır.
- Çalışma Alanı (Workspace): Robotun ulaşabildiği bölge.
- Kinematik (Kinematics): Robotun yapabileceği hareketleri belirleyen, robot üzerindeki katı cisimlerin ve eklemlerin düzenlemesi, sıralaması.
- Taşıma Kapasitesi (Payload): Robotun taşıyabileceği maksimum yük.
- Hız (Speed): Robotun ucunu konumlandığı maksimum hızdır. Bu hız robotun her bir ekleminin ya da ucunun açısal veya doğrusal hızı olabilir.
- İvme (Acceleration): Robotun bir ekleminin maksimum ivmesidir. Bu bir sınırlayıcı etken olduğundan dolayı robot küçük mesafelerde ya da yönünün sıklıkla değiştiği hareketlerde maksimum hızına ulaşamayabilir.
- Doğruluk (Accuracy): Robotun konumlanması istenilen noktaya ne kadar yaklaşabildiğini gösterir. Doğruluk robotun hızı, çalışma alanındaki yeri ve yükü ile değişebilir. Robotun kalibre edilmesi ile iyileştirilebilir.
- Tekrarlama Kabiliyeti (Repeatability): Robotun daha önce ulaştığı bir noktaya dönebilmesi ile ilgili ölçüdür [4].

2.4. Manipülâtör Yapıları

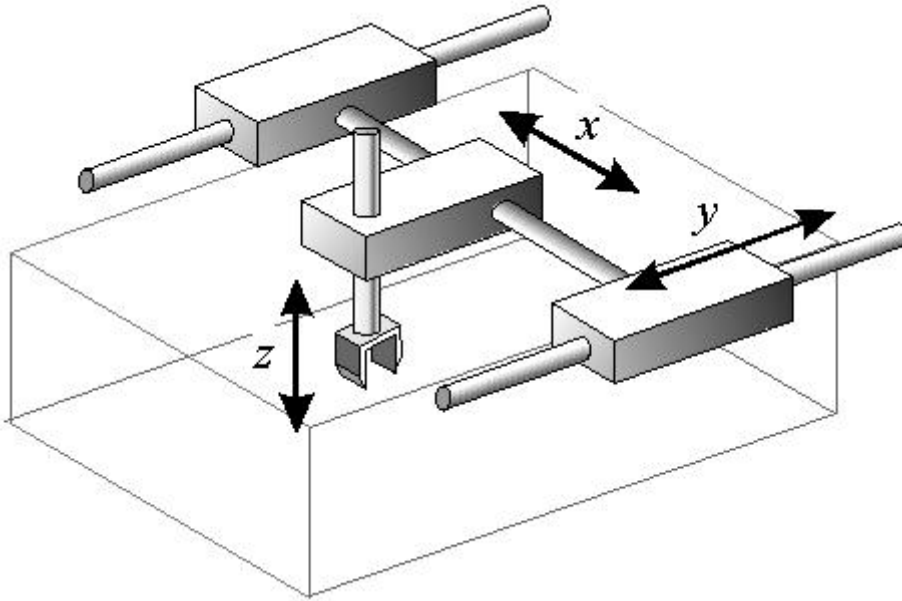
Bir manipülâtörün temel yapısı açık kinematik zincirdir (open kinematic chain). Bir manipülâtörün iki ucu sadece bir dizi uzuvla birbirine bağlanıyorsa açık kinematik olarak adlandırılır. Buna alternatif olarak, manipülâtörün uzuvları kapalı bir döngü oluşturuyorsa bu manipülâtör kapalı kinematik zincir yapısındadır.

Manipülâtörün hareket kabiliyeti eklemler ile sağlanır. Temel (base) eklemden başlayarak eklemlerin tipleri ve sıralanışları manipülâtörlerin: kartezyen (cartesian), silindirik (cylindrical), küresel (spherical), SCARA ve mafsallı (anthropomorphic) olarak sınıflandırılmasına imkân verir.

2.4.1. Kartezyen (Cartesian) manipülâtör

Kartezyen geometri birbirine dik olan üç tane prizmatik (prismatic) eklem kullanılarak elde edilir (Şekil 2.1). Basit geometrisinden dolayı her bir eksen kartezyen uzayda birer serbestlik derecesine karşılık gelmektedir ve böylelikle uzayda istenilen hareketler yerine getirilebilir. Kartezyen geometri çok iyi derecede mekanik sertliğe sahiptir. Bileğin pozisyon doğruluğu çalışma alanının her yerinde sabittir. Çalışma alanı ise paralelyüz tarafından çevrilmiş bölgedir [3].

Bu yapı büyük hacimde çalışma alanına, büyük boyut ve kütledeki malzemelerin manipülasyonuna imkân verir. Yük taşıma kapasitesi diğer robot türlerine göre daha büyüktür. İnsan gücünün taşıma kapasitesini aşan yüklerin taşınmasında kullanılır. Bu nedenle genellikle yük araçlarına, yükleme ve boşaltma işlerinde, fabrikalar da ağır yükleri taşımak amacı ile fabrikaların tavanlarına monte edilerek kullanımı yaygındır.

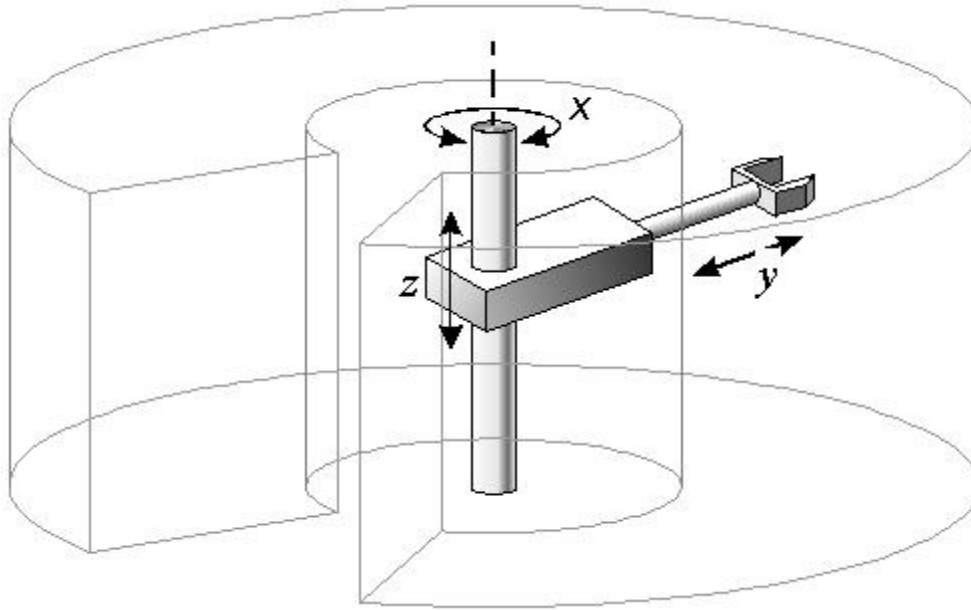


Şekil 2.1. Kartezyen manipölatör

2.4.2. Silindirik (Cylindrical) manipölatör

Silindirik geometri bir tane döner (revolute) ve iki tane prizmatik eklem kullanılarak elde edilir (Şekil 2.2). Eğer robotun yerine getireceği görev silindirik koordinatlarda tanımlanırsa her bir eklem birer serbestlik derecesine karşılık gelmektedir. Manipölatörün silindirik yapısı iyi derecede mekanik sertliğe sahiptir. Bileğin pozisyon doğruluğu yatay harekete bağılı olarak değişir [3].

Robotun kullanım alanı ve yük taşıma kapasitesine göre hidrolik, pnömatik veya elektrik tahrikli olarak kullanılmaktadır. Silindirik robot kollar nemli, rutubetli ve tozlu ortamlarda, deniz altı, uzay gözlem araçlarında ve nokta kaynağı işlerinde yaygın olarak kullanılmaktadır.

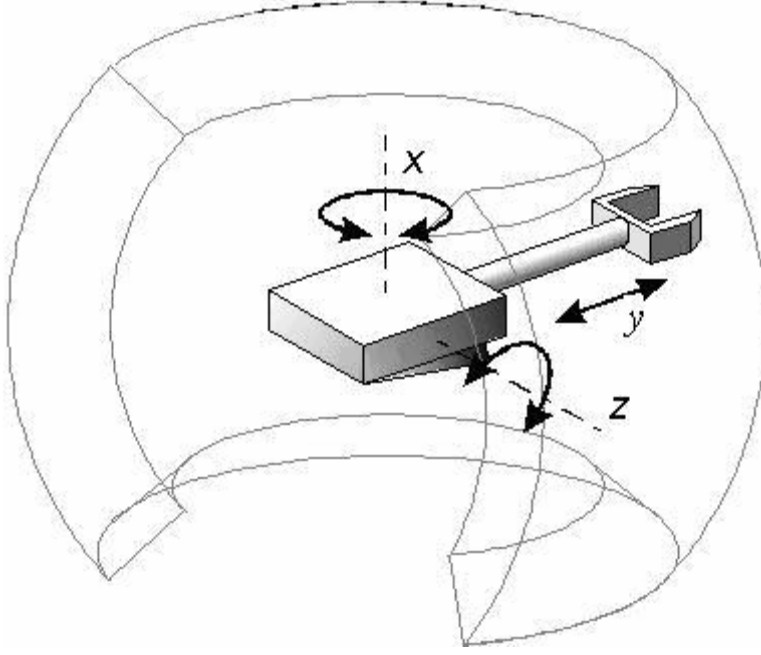


Şekil 2.2. Silindirik manipulatör

2.4.3. Küresel (Spherical) manipulatör

Küresel geometri iki tane döner ve bir tane prizmatik eklem kullanılarak elde edilir (Şekil 2.3). Robotun görevi küresel koordinatlarda tanımlandığında her eklem birer serbestlik derecesine karşılık gelmektedir. Bu yapılarıdaki manipulatörlerin mekanik sertliği yukarıda bahsedilen her iki manipulatörden de daha düşüktür. Bileğin pozisyon doğruluğu radyal harekete bağlı olarak değişir [3].

Hidrolik tahrik sistemine sahip olan küresel robot kollar eğme, bükme işlerinde, kameralı izleme işlerinde kullanılmaktadır. Ayrıca sarkaç robot olarak da küçük bir moment ile hareketlerini devam ettiren bu robotlar kaynak ve zamlama işlemlerinde kullanılırlar.

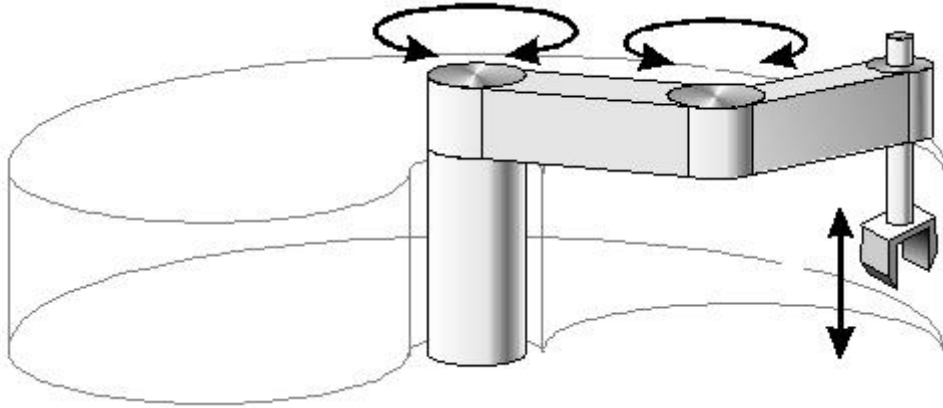


Şekil 2.3. Küresel manipülâtör

2.4.4. SCARA manipülâtör

SCARA (Selective compliance assembly robot arm) geometrisi iki tane döner ve bir tane prizmatik eklem kullanılarak elde edilir (Şekil 2.4). Bu yapıda bütün eklemler birbirlerine paralel şekilde hareket ederler. Bileğin pozisyon doğruluğu bilek birinci eklemden uzaklaştıkça azalır [3].

Hız ve konum performansı çok iyi olduğundan dolayı bu robot kol en çok elektronik sanayinde, elektronik kartlara malzemelerin montajını gerçekleştirmek için kullanılmaktadır. Tutma ve taşıma işlerinde maliyetinin ucuz olmasından ve programlanmasının kolay olmasından dolayı çok kullanılmaktadır.

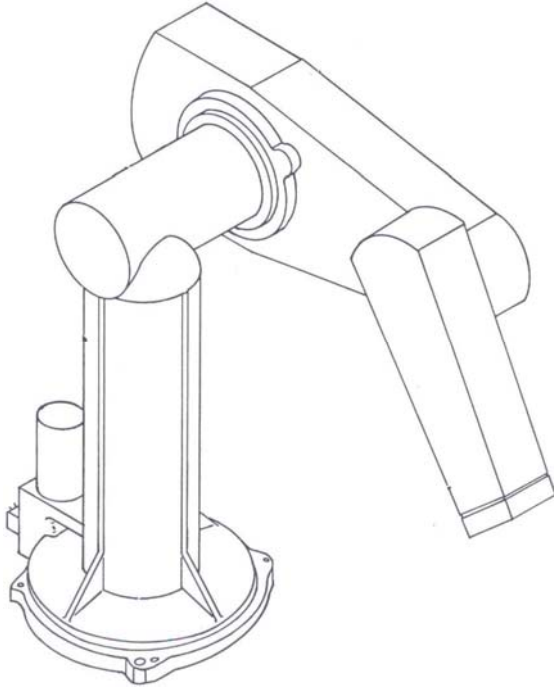


Şekil 2.4. SCARA manipölatör

2.4.5. Mafsallı (Anthropomorphic) manipölatör

Mafsallı geometri üç tane döner eklem kullanarak elde edilir (Şekil 2.5). Birinci eklem diğer iki ekleme dik olacak şekilde tasarlanmıştır, ikinci ve üçüncü eklem ise birbirlerine paraleldirler. İnsan koluna benzerliğinden dolayı ikinci eklem omuz (shoulder), üçüncü eklem ise dirsek (elbow) olarak adlandırılır. Bu geometrideki robotların üç eklemi de döner yapıda olduğundan yukarıda bahsedilen robotlardan daha çok hareket kabiliyetine sahiptirler [3].

Yapılacak uygulamanın niteliğine göre robot kolun eksen sayısı tercihi yapılmalıdır. Daha basit işlemlerin uygulanmasında 3 eksenli robot kol yeterli gelmekte iken daha karmaşık ve çok fonksiyonlu bir uygulama işleminde 3 eksenli robot kol yeterli olmamaktadır. Uygulanan işlemler karmaşıklaştıkça mafsallı sayısının artması gerekmektedir. Mafsallı sayısının artması robotun hareket serbestliğini arttırmaktadır.



Şekil 2.5. Mafsallı manipölatör

2.5. Robotların Kullanıldığı Endüstri Kolları

Robotların kullanım alanları ve kullanım alanlarındaki uygulamaları IFR (Uluslararası Robot Federasyonu) tarafından ISIC (International Standart Industrial Classification of all Economic Activities)'e göre sınıflandırılmıştır. Buna göre, robotların kullanıldığı endüstri kolları [5];

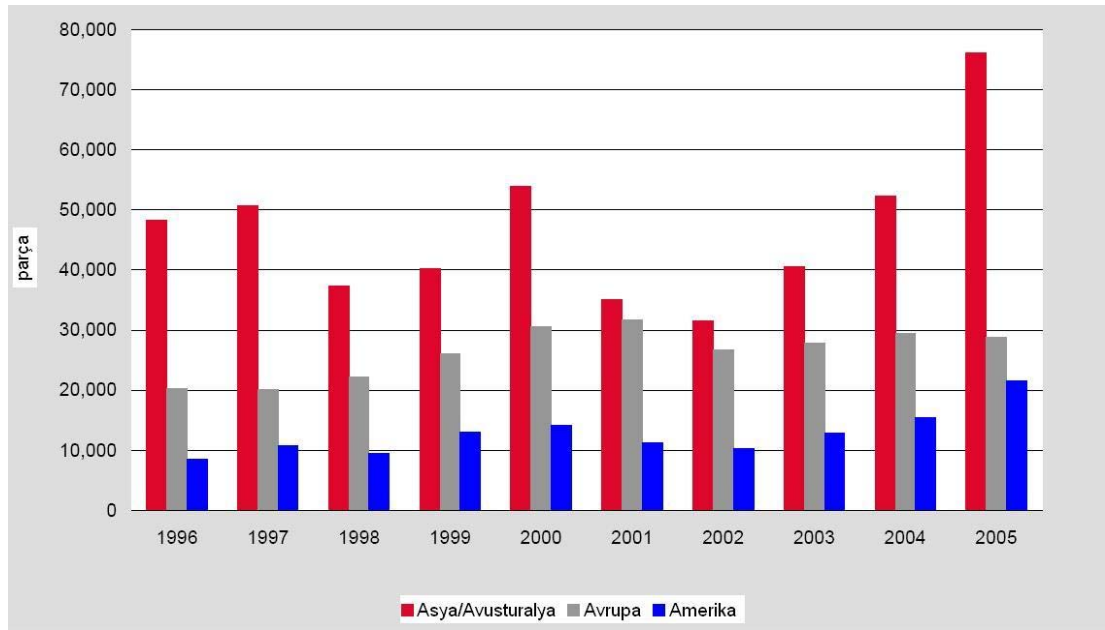
- Tarım, ormancılık, hayvancılık
- Maden işleme
 - o Gıda
 - o Tekstil, deri ve konfeksiyon
 - o Ağaç işleme, mobilyacılık
 - o Kâğıt, kâğıt ürünleri ve matbaa, basın sektörü
 - o Kimyasallar, petrol ve plastik ürünler imalatı

- Seramik ve metal olmayan mineral ürünler
- Temel metaller
 - Metal ürün imalatı (makine imalatı hariç)
 - Makine imalatı (elektrikli ürünler hariç)
 - Elektrikli ürün imalatı
 - Taşımacılık sektörü
 - Motorlu imalat ve otomobil imalatı
 - Uçak ve hava taşıtları imalatı
 - Hassas ve optik aletler imalatı
- Diğer imalat sektörleri
- Elektrik, gaz ve su dağıtım sektörü
- Yapı ve inşaat sektörü
- Eğitim
- Temiz oda
- Tıp
- Araştırma ve geliştirme

2.6. Dünya Robot Marketi İstatistikleri

Dünya Robot Marketi (World Robotics 2006) raporuna göre yaklaşık 126,700 yeni endüstriyel robotun kurulmasıyla beraber 2005 senesinde robot satışları en yüksek değerine ulaşmıştır. Bu değer 2004 senesinde yapılan satışlardan %30 fazladır. 2005 senesi satış değerleri şimdiye kadar kayıt edilmiş en büyük yıllık satış değeridir. Bununla birlikte robot endüstrisindeki gelişmeler üç büyük endüstri bölgesinde Avrupa, Amerika ve Asya'da farklılıklar göstermektedir.

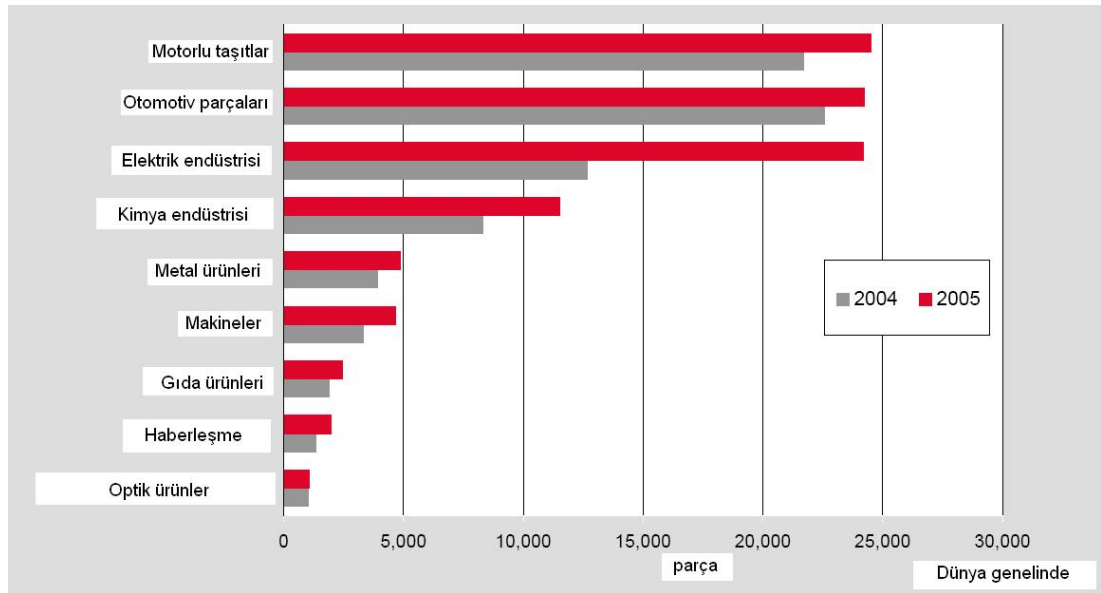
Asya ve Amerika'da robotik yatırımlarında hızlı bir artış yaşanırken aynı durum Avrupa için geçerli değildir. Otomotiv sektörü bu üç sanayi bölgesindeki sonuçları etkilemektedir. Asya'da otomotiv sektörüne ek olarak elektronik parça endüstrisi, haberleşme teçhizatı endüstrisi ve bilgisayar endüstrisi 2004 senesinde görülen ilerlemeyi takviye etmektedir. Endüstriyel robotların yıllık kurulumu için tahmini değerler Şekil 2.6'da verilmiştir [6].



Şekil 2.6. Endüstriyel robotların yıllık kurulumu

Otomotiv sektörünün haricinde diğer sektörlerde endüstriyel robotlara olan talep sürekli artmaktadır: Plastik ve kauçuk, gıda ve paketlenme, ev eşyaları, ahşap ve mobilya, cam ve seramik üretimlerindeki yayılma devam etmektedir. Endüstriyel robotların 2004–2005 yıllarındaki ana endüstri kollarındaki kurulumu için tahmini değerler Şekil 2.7'de verilmiştir.

Dünya genelindeki endüstride hâlihazırda kullanılan endüstriyel robotların 2005 yılı sonundaki tahmini sayısı minimum 923,000 parça ve maksimum ise 1,120,000 parçadır.



Şekil 2.7. Robotların ana endüstri kollarındaki kuruluşu (2004–2005)

Endüstriyel robotların 2004–2005 yılı için kurulum ve mevcut kullanıma sayıları ve 2006–2009 yılları için tahmin değerleri Çizelge 2.1’de verilmiştir.

2.7. Robot Teknolojisinin Türkiye İçin Önemi

Dünyada toplumlar 10 yıl öncesine kadar sanayileşmiş, geri kalmış; ya da az gelişmiş, gelişmekte olan ve gelişmiş ülkeler olarak sınıflandırılırken, bundan sonra “bilgi toplumu” olmuş ve “bilgi toplumu” olmamış ülkeler sınıflaması yapılacaktır. 2020’li yıllarda, çalışanların büyük bir kısmının bugün icat edilmemiş mesleklerde çalışacakları veya başka mesleklere geçecekleri bilinmektedir.

Kiři bařına üretkenlik son 130 yılda yaklaşık 25 kat artmıřtır. Bu üretkenlik artışının yarısı yani 13 kat kadarı fiziki ürün artışından, diđer yarısı da insanların çalışma sürelerinin yaklaşık yarıya düşmesinden kaynaklanmıřtır. Fiziki ürün artışı ancak, robotik ve otomasyon teknolojileri ile anında ve esnek üretim ile gerçekleştirilebilmektedir.

Bugün yarı yarıya çalışıp 13 kat daha yüksek bir refah seviyesinde yaşamak, sadece sanayi devriminin getirdiđi makineleşme ile deđil, robot ve otomasyon teknolojilerinin günden güne artan kullanımı ve bilgi teknolojisine hâkimiyet sayesinde gerçekleşmiştir.

İç ve dış pazarda rekabet günden güne artmakta, ürün ve hizmetlerden yüksek kalite ve standartlara uyum beklenmektedir. Gerek Avrupa Birliđi düzenlemelerinin gerekse ISO–9000 ve diđer kalite yönetim sistemlerinin, kaliteli ve standart ürünlerin esnek üretim hatlarından çıkması konusunda ısrarcı olması, ulusal endüstrimizin robot teknolojisi ile kurulan karmaşık, çok yönlü ve esnek üretim yapıları bütünleşik montaj tesisleri kullanımını zorunlu hale getirmiştir.

Türkiye, Otomotiv endüstrisinin artan ihracat potansiyeli ile imalat endüstrisinde yakalanan “Avrupa’nın üretim üssü Türkiye” tarihi fırsatını, gelecek 20 yılda ancak, ülkemiz için son derece önem taşıyacak olan robotik ve mekatronik teknolojisine ulusal yatırım vizyonu ile yakalayabilir.

Ulusal değerlerimizin dünya pazarlarında yarışabilmeleri için stratejimiz, kısa vadede ihtiyaç duyulacak, robotik-mekatronik ve ileri teknolojileri geliřtirmek; uzun vadede ise, bu ileri teknolojileri üreterek ülkeyi bilgi toplumu haline dönüřtürecektir alt yapıyı kurmak olmalıdır [5].

Çizelge 2.1. Endüstriyel robotların kurulum ve mevcut kullanıma değerleri

Ülke	Yıllık Kurulum				Yıl Sonunda Mevcut Kullanılan			
	2004	2005	2006	2009	2004	2005	2006	2009
Amerika	15400	21555	17200	20100	126961	143203	153500	182500
Brezilya	208	320			2352	2672		
Kuzey Amerika	15170	21136	16500	19100	123663	139553	149400	176000
Amerika (Diğer)	22	99			946	978		
Asya/Avustralya	52311	76047	65000	76000	443193	481664	502000	583000
Çin	3493	4461			7096	11557		
Hindistan	369	450			619	1069		
Endonezya	74	193			121	314		
Japonya	37086	50501	40000	46000	356483	373481	372000	388500
Malezya	250	243			1452	1695		
Filipinler	65	80			93	173		
Kore Cumhuriyeti	5457	13005			51302	61576		
Singapur	244	424			5443	5463		
Tayvan	3680	4096			11881	15464		
Tayland	757	1458			1014	2472		
Vietnam	14	99			14	113		
Asya (Diğer)	170	124			3505	3349		
Avustralya/Yeni Zelanda	652	913			4170	4938		
Avrupa	29409	28863	28200	33800	279019	297374	307700	345400
Avusturya	545	485			3907	4148		
Belçika/Hollanda/Lüksemburg	536	1097			8749	9362		
Danimarka	296	354			2342	2661		
Finlandiya	401	556			3712	4159		
Fransa	3009	3275	3000	3700	28133	30434	32200	37900
Almanya	13401	10506	10700	13000	120544	126725	132300	142700
İtalya	5679	5425	5100	6200	53244	56198	58900	66400
Norveç	61	115			724	811		
Portekiz	211	144			1488	1542		
İspanya	2826	2649			21893	24081		
İsveç	833	939			7341	8028		
İsviçre	310	442			3539	3732		
Türkiye	24	207			196	403		
İngiltere	785	1363	800	1200	14176	14948	14700	14300
Merkezi/Batı Avrupa	419	1149			8372	9337		
Avrupa (Diğer)	73	157			659	805		
Afrika	87	204	220	250	430	634	900	1600
Toplam	97207	126669	110620	130150	849603	922875	964100	1112500

3. ROBOT MANİPÜLATÖRÜN EKSEN-ÖZGÜL DENETİMİ

3.1. Giriş

Önceki bölümlerde bahsedildiği gibi, endüstriyel robot, değişken görevleri yerine getirmek için programlanabilen, çok fonksiyonlu, malzeme, parça ve eşyaları taşıyabilmek için tasarlanmış bir manipülatördür. Son yıllarda robotik manipülatörler çeşitli endüstrilerde ve değişik uygulamalarda kullanılmaktadır.

Literatürde robot manipülatörlerin denetimi amaçlı yapılmış çok sayıda çalışma yer almaktadır. Lynx robot manipülatörü kullanılarak gerçekleştirilen bir çalışmada, robot manipülatörün denetimi genel amaçlı bir mikro denetleyici tarafından gerçekleştirilmiştir [7].

Son zamanlarda FPGA (Field Programmable Gate Array) kullanımı birçok alanda hızla artmaktadır ve robot denetimi uygulamalarında gerçekleştirilmiş çeşitli çalışmalar bulunmaktadır [8–9]. Bazı çalışmalarda FPGA yüksek hızda seri haberleşme ve bilgi iletimini yerine getirmek amacıyla kullanılmıştır [10]. Başka bir çalışmada tasarlanan bulanık mantık algoritması FPGA üzerinde gerçekleştirilmiştir [11]. Yine yapılan bir çalışmada robot kontrolü için tasarlanan algoritma FPGA üzerinde gerçekleştirilmiştir [12]. Yüksek performanslı 3 boyutlu bir görüntüleme sistemi, FPGA'nın paralel işlem yeteneğinden faydalanılarak gerçekleştirilmiştir [13]. Robotların manevra kolu ile denetiminin gerçekleştirildiği çalışmalar da bulunmaktadır [14].

Bu çalışmada genel amaçlı bir FPGA kullanılarak altı eksenli robotun manevra kolu (joystick) ile denetimi gerçekleştirilmiştir. Tasarlanan denetim sistemi sayesinde robot istenilen hareketi manevra kolu ile yerine getirmektedir. Geliştirilen sayısal tasarım manevra kolundan alınan giriş sinyallerini yorumlayarak gerekli denetim sinyalini üretir ve ilgili eksenleri denetler.

Bundan sonra gelen bölümlerde sırasıyla, robotikte kullanılan matematiksel yöntemler hakkında kısa bilgi, geliştirilen sistemin genel blok şeması, kullanılan robot manipülatörün matematiksel modeli, deneysel düzenek, denetim kartı ve geliştirilen sayısal tasarımdan bahsedilmektedir.

3.2. Kinematik

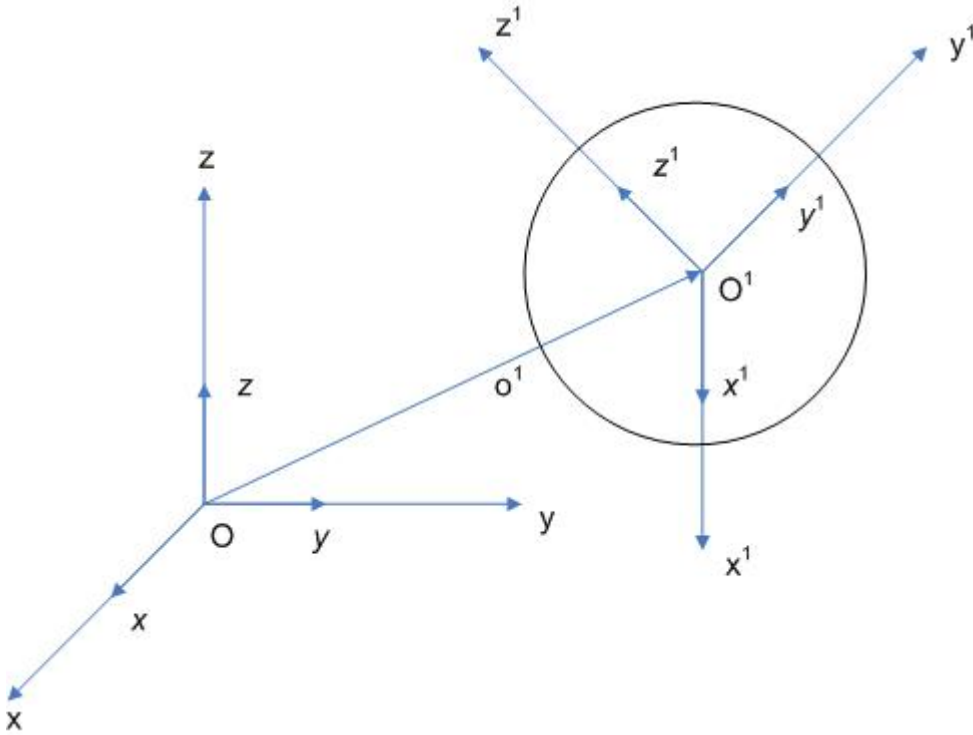
Bir manipülatör mekaniksel bakış açısıyla şematik olarak katı uzuvların döner ya da prizmatik eklemlerle birbirine bağlanmış hali olarak gösterilebilir. Mekanik zincirin bir ucuna robotun uç elemanı bağlanırken diğer ucu ise bir temele (base) sabitlenir. Bu mekanik yapının sonuç hareketi her bir uzvun kendinden önceki uzuv'a göre tanımlanmış hareketlerinin bileşiminden elde edilir. Bu nedenle bir nesneyi uzayda manipüle edebilmek için, robotun uç elemanının konum ve yön bilgisini tanımlamak gerekir.

Düz kinematik (direct kinematics) sayesinde robotun uç elemanının konumu ve yönü eklem değişkenleri cinsinden bir referans koordinat sistemine göre tanımlanabilir.

Ters kinematik (inverse kinematics) sayesinde robotun uç elemanının yönü ve koordinatları bilindiği takdirde eklem değişkenlerinin değerleri hesaplanabilir.

3.2.1. Katı nesnenin konumu ve yönü

Bir katı nesne uzayda konumu ve yönü bir referans koordinat sistemine göre bildirilerek tanımlanır. Şekil 3.1'de görüldüğü gibi, O - xyz referans koordinat sistemi ve x, y, z sistemin birim vektörleri olsun.



Şekil 3.1. Katı nesnenin konumu ve yönü

Katı nesne üzerindeki O^1 noktasının O - xyz referans koordinat sistemine göre pozisyonu Eş.3.1'de verilmektedir. o^1_x , o^1_y , o^1_z , o^1 vektörünün bileşenleridir. O^1 noktasının pozisyonu (3×1) vektör formatında Eş. 3.2'deki gibi yazılabilir.

$$o^1 = o_x^1 x + o_y^1 y + o_z^1 z \quad (3.1)$$

$$o^1 = \begin{bmatrix} o_x^1 \\ o_y^1 \\ o_z^1 \end{bmatrix} \quad (3.2)$$

Katı nesnenin yönünü tanımlamak için katı nesne üzerine bir koordinat sistemi iliştiirmek ve bu koordinat sistemini referans koordinat sisteminin birim vektörleri cinsinden ifade etmek gerekir. O^1 merkezli, x^1, y^1, z^1 birim vektörleri olan $O^1-x^1y^1z^1$ koordinat sistemi katı nesne üzerine iliştilsin. Bu birim vektörleri O -xyz referans koordinat sistemine göre Eş. 3.3'te görüldüğü gibi ifade edilirler.

$$\begin{aligned} x^1 &= x_x^1 x + x_y^1 y + x_z^1 z \\ y^1 &= y_x^1 x + y_y^1 y + y_z^1 z \\ z^1 &= z_x^1 x + z_y^1 y + z_z^1 z \end{aligned} \quad (3.3)$$

3.2.2. Dönme matrisi

Sade bir gösterim için, Eş. 3.3'te verilen üç birim vektörü (3x3)'lük bir matris olarak birleştirilerek nesnenin yönünü referans koordinat sistemine göre tanımlayabilir. Bu matrise dönme matrisi denir (Eş. 3.4).

$$R = \begin{bmatrix} x^1 & y^1 & z^1 \end{bmatrix} = \begin{bmatrix} x_x^1 & y_x^1 & z_x^1 \\ x_y^1 & y_y^1 & z_y^1 \\ x_z^1 & y_z^1 & z_z^1 \end{bmatrix} \quad (3.4)$$

3.2.3. Temel dönmeler

Z eksenini etrafında dönme matrisi Eş. 3.5'te görülmektedir.

$$R_z(\alpha) = \begin{bmatrix} \cos \alpha & -\sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.5)$$

Y eksenini etrafında dönme matrisi Eş. 3.6'da görülmektedir.

$$R_y(\beta) = \begin{bmatrix} \cos \beta & 0 & \sin \beta \\ 0 & 1 & 0 \\ -\sin \beta & 0 & \cos \beta \end{bmatrix} \quad (3.6)$$

X eksenini etrafında dönme matrisi Eş. 3.7'de görülmektedir.

$$R_x(\gamma) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \gamma & -\sin \gamma \\ 0 & \sin \gamma & \cos \gamma \end{bmatrix} \quad (3.7)$$

3.2.4. Homojen koordinat dönüşümleri

Koordinat sistemi dönüşümünün yapılmasının nedeni bir koordinat sistemine göre koordinatları bilinen bir noktayı diğer bir koordinat sistemine göre de tanımlı hale getirmektir. Dönüşümün yapılabilmesi için koordinat sistemlerinin birbirlerine göre konumlarının ve yönlerinin bilinmesi gerekir. Şekil 3.1'de görülen uzayda bulunan P noktası ele alındığında, p^0 bu noktanın $O_0-x_0y_0z_0$ koordinat sistemine göre tanımlı konum vektörü olsun. o_1^0 O_1 merkezli koordinat sistemini O_0 merkezli koordinat sistemine göre tanımlayan vektör olsun.

R_1^0 O_1 merkezli koordinat sisteminin O_0 merkezli koordinat sistemine göre tanımlı dönme matrisi olsun. p^1 P noktasını O_1 merkezli koordinat sistemine göre tanımlı koordinat vektörü olsun. P noktasının referans koordinat sistemine göre ifadesi Eş. 3.8'de görüldüğü gibi olur. Bu eşitlik koordinat dönüşümlerini (kaydırma+dönme) ifade etmektedir.

Aynı noktanın iki ayrı çerçeve (koordinat sistemi) içindeki koordinatları arasındaki ilişkiyi sade bir şekilde gösterebilmek için p vektörünün homojen gösterimi p^1 vektörüne dördüncü bir birim eleman eklenerek tanımlanır (Eş. 3.9).

Koordinat dönüşümleri Eş. 3.10'da görüldüğü gibi (4x4) matris formatında yazılabilir ve bu matris homojen transformasyon matrisi olarak adlandırılır. Görüldüğü üzere bu matris O_1 merkezli çerçeveden O_0 merkezli çerçeveye tanımlı olan dönme matrisini ve kaydırma vektörünü içermektedir. Koordinat dönüşümleri Eş. 3.11'deki gibi sade bir şekilde gösterilebilir ve bir dizi koordinat dönüşümü Eş. 3.12'de görüldüğü gibi çarpma işlemleriyle birleştirilebilir.

$$p^0 = o_1^0 + R_1^0 p^1 \quad (3.8)$$

$$p^1 = \begin{bmatrix} p \\ 1 \end{bmatrix} \quad (3.9)$$

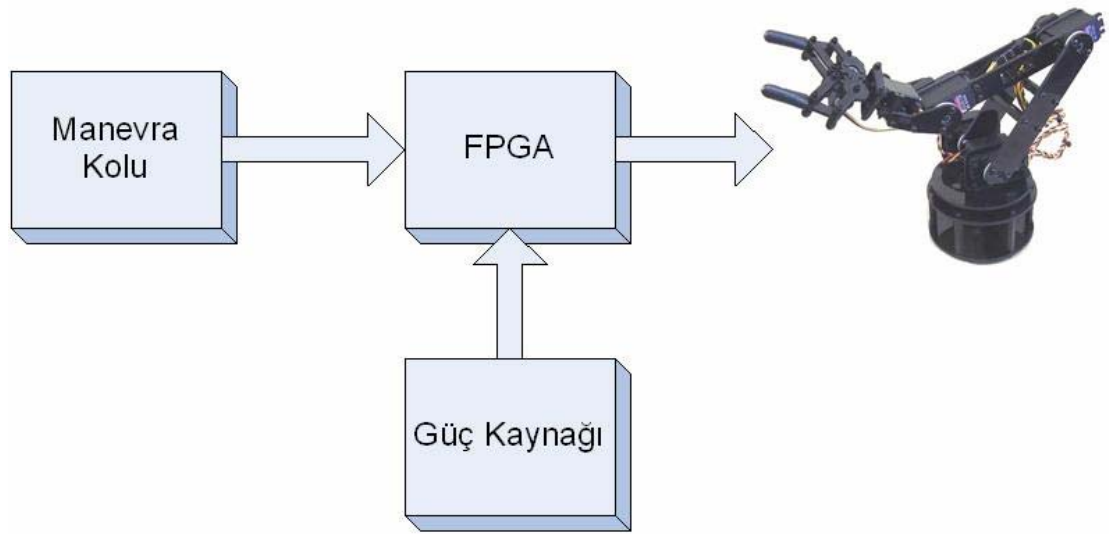
$$A_1^0 = \begin{bmatrix} R_1^0 & o_1^0 \\ 0^T & 1 \end{bmatrix} \quad (3.10)$$

$$p^{1^0} = A_1^0 p^1 \quad (3.11)$$

$$p^{1^0} = A_1^0 A_2^1 \dots A_n^{n-1} p^{1^n} \quad (3.12)$$

3.3. Geliştirilen Sistemin Genel Blok Şeması

Bu çalışmanın amacı, altı eksenli robot manipülatörü manevra kolu ile genel amaçlı FPGA tabanlı deney kartı kullanarak denetlemektir. Sistemin blok şeması Şekil 3.2'de görülmektedir.

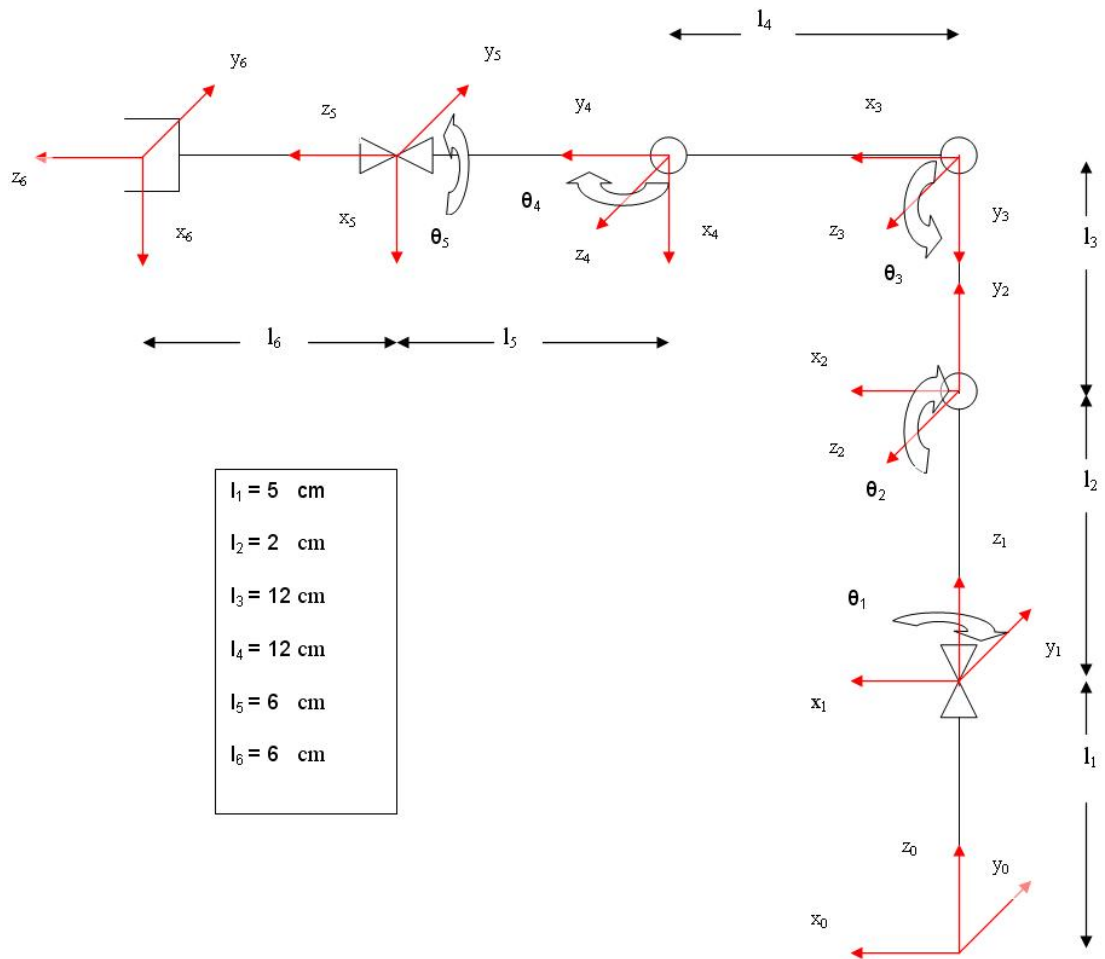


Şekil 3.2. Sistemin blok şeması

Sistem dört ana bloktan meydana gelmektedir. Bunlar manevra kolu, güç kaynağı, FPGA kartı ve robottur. Manevra kolundaki bir eksen komutu veya eksen ve butondan oluşan birleşik komut, robot manipülatörü üzerindeki belirlenmiş bir eksenin hareketini sağlar. Geliştirilen sayısal tasarım, manevra kolundan alınan denetim sinyallerini yorumlayarak gerekli denetim bilgisini ilgili eksen denetleyicisine iletir, eksen denetleyicisi de robotun denetimi için gerekli olan PWM (pulse width modulation) sinyallerini üretir. Lynx-6 robot manipülatörü, Lynx Motion firmasının bir ürünüdür, beş eksen ve bir kavrayıcıdan (gripper) oluşan toplam altı bağımsız ekleme sahiptir.

3.4. Lynx-6 Manipülâtörünün Matematiksel Modeli

Homojen koordinat sistemleri atamaları robot manipülâtörün ucunun istenilen noktada konumlandırılabilmesi için ve pozisyon denetlemesinin sağlanabilmesi için gerekli olan matematiksel analizlerin yapılmasında gerçekleştirilmesi gereken ilk aşamadır. Robot manipülâtörün homojen koordinat sistemleri atamaları Şekil 3.3'te görüldüğü gibi gerçekleştirilmiştir. Robotun sahip olduđu her bir ekleme birer koordinat sistemi atanmaktadır. Koordinat atamalarının yapılması sonucunda robotun altı adet eklemi olduđu için altı adet farklı koordinat sistemi ortaya çıkmıştır.



Şekil 3.3. Homojen koordinat sistemi atamaları

Altıncı koordinat sistemi robot kolun kavrayıcı eklemine atanan koordinat sistemidir. Beşinci koordinat sistemi ise robotun ucunun dönmesini sağlayan eklemeye atanan koordinat sistemidir. Bu koordinat sistemleri arasında tanımlanan dönüşüm matrisi Eş. 3.13'te verilmiştir.

$${}^6A_5 = \begin{bmatrix} \cos\theta_5 & -\sin\theta_5 & 0 & 0 \\ \sin\theta_5 & \cos\theta_5 & 0 & 0 \\ 0 & 0 & 1 & l_6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.13)$$

Dördüncü koordinat sistemi, robotun bilek eklemine atanan koordinat sistemidir. Beşinci ve dördüncü koordinat sistemleri arasında tanımlanan dönüşüm matrisi Eş. 3.14'te verilmiştir.

$${}^5A_4 = \begin{bmatrix} \sin\theta_4 & 0 & \cos\theta_4 & l_5 \times \cos\theta_4 \\ -\cos\theta_4 & 0 & \sin\theta_4 & l_5 \times \sin\theta_4 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.14)$$

Üçüncü koordinat sistemi, robotun dirsek eklemine atanan koordinat sistemidir. Dördüncü ve üçüncü koordinat sistemleri arasında tanımlanan dönüşüm matrisi Eş. 3.15'te verilmiştir.

$${}^4A_3 = \begin{bmatrix} -\sin\theta_3 & \cos\theta_3 & 0 & l_4 \times \cos\theta_3 \\ \cos\theta_3 & \sin\theta_3 & 0 & l_4 \times \sin\theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.15)$$

Robotun omuz kısmı iki eklemden meydana gelmektedir. Omuz kısmının ikinci eklemine atanan koordinat sistemi ikinci koordinat sistemidir. Üçüncü ve ikinci koordinat sistemleri arasında tanımlanan dönüşüm matrisi Eş. 3.16'da verilmiştir.

$${}^3A_2 = \begin{bmatrix} \cos\theta_2 & \sin\theta_2 & 0 & I_3 \times \cos\theta_2 \\ \sin\theta_2 & -\cos\theta_2 & 0 & I_3 \times \sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.16)$$

Omuz kısmının birinci eklemine atanan koordinat sistemi birinci koordinat sistemidir. İkinci ve birinci koordinat sistemleri arasında tanımlanan dönüşüm matrisi Eş. 3.17'de verilmiştir.

$${}^2A_1 = \begin{bmatrix} \cos\theta_1 & 0 & \sin\theta_1 & 0 \\ \sin\theta_1 & 0 & -\cos\theta_1 & 0 \\ 0 & 1 & 0 & I_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.17)$$

Sıfıncı koordinat sistemi robotun temelinde (base) atanan koordinat sistemidir. Referans koordinat sistemi olarak atanmıştır. Birinci ve sıfıncı koordinat sistemleri arasında tanımlanan dönüşüm matrisi Eş. 3.18'de verilmiştir.

$${}^1A_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & I_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.18)$$

Robotun kavrayıcı eklemi (altıncı koordinat sistemi) ile referans (sıfıncı) koordinat sistemi arasında tanımlanmış olan dönüşüm matrisi Eş. 3.19'da verilmiştir.

θ_i : sıralı koordinat sistemleri arasındaki açı.

$${}^6A_0 = {}^1A_0 {}^2A_1 {}^3A_2 {}^4A_3 {}^5A_4 {}^6A_5 = \begin{bmatrix} k & n & s & x \\ l & p & t & y \\ m & r & u & z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$k = -\cos\theta_1 \times \cos\theta_5 \times \cos(\theta_2 - \theta_3 + \theta_4) - \sin\theta_1 \times \sin\theta_5$$

$$l = -\sin\theta_1 \times \cos\theta_5 \times \cos(\theta_2 - \theta_3 + \theta_4) + \cos\theta_1 \times \sin\theta_5$$

$$m = -\cos\theta_5 \times \sin(\theta_2 - \theta_3 + \theta_4)$$

$$n = \cos\theta_1 \times \sin\theta_5 \times \cos(\theta_2 - \theta_3 + \theta_4) - \sin\theta_1 \times \cos\theta_5$$

$$p = \sin\theta_1 \times \sin\theta_5 \times \cos(\theta_2 - \theta_3 + \theta_4) + \cos\theta_1 \times \cos\theta_5 \quad (3.19)$$

$$r = \sin\theta_5 \times \sin(\theta_2 - \theta_3 + \theta_4)$$

$$s = \cos\theta_1 \times \sin(\theta_2 - \theta_3 + \theta_4)$$

$$t = \sin\theta_1 \times \sin(\theta_2 - \theta_3 + \theta_4)$$

$$u = -\cos(\theta_2 - \theta_3 + \theta_4)$$

$$x = \cos\theta_1 \times (l_2 \times \sin(\theta_2 - \theta_3 + \theta_4) + l_2 \times \cos(\theta_2 - \theta_3) + l_2 \times \cos\theta_2)$$

$$y = \sin\theta_1 \times (l_2 \times \sin(\theta_2 - \theta_3 + \theta_4) + l_2 \times \cos(\theta_2 - \theta_3) + l_2 \times \cos\theta_2)$$

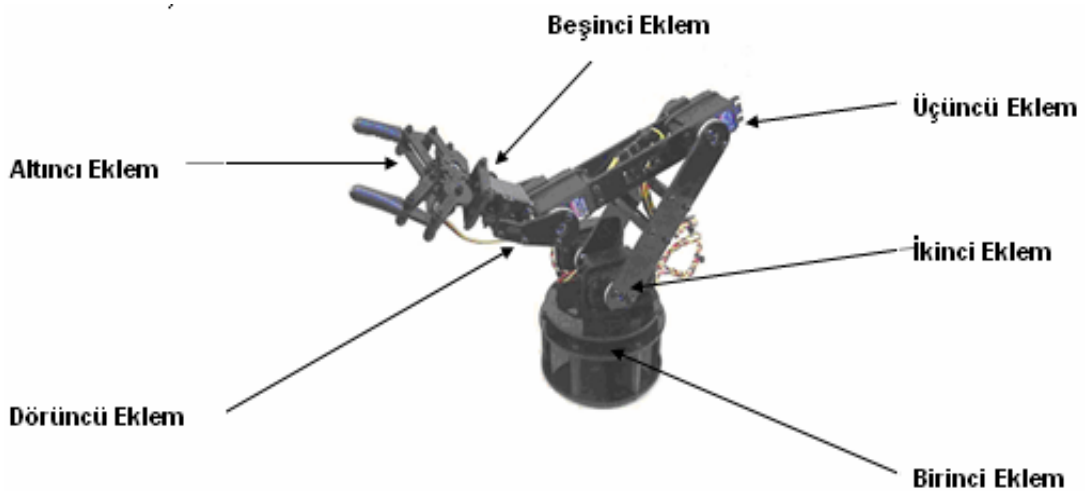
$$z = -l_2 \times \cos(\theta_2 - \theta_3 + \theta_4) + l_2 \sin(\theta_2 - \theta_3) + l_2 \times \sin\theta_2$$

Robot manipulatörün ucunun referans koordinat sistemine göre istenilen noktada konumlandırılabilmesi için robot kolunun eklem açılarının bilinmesi gerekmektedir. Robot kolun ucunun bulunması istenilen koordinatlar biliniyor ise eklem açıları 6A_0 matrisinde yer alan denklemler kullanılarak bulunabilir. Eş. 3.19'da verilen denklemler aynı zamanda manipulatörün düz kinematik denklemleridir.

3.5. Lynx-6 Manipulatörünün Özellikleri

Lynx-6 manipulatörü hızlı, doğru ve tekrarlanabilir hareket yeteneğine sahiptir. Robot; tabansal dönme, omuz, dirsek, bilek hareketi, bilek dönmesi ve fonksiyonel bir kavrayıcıya sahiptir. Lynx-6 robot manipulatörü dayanıklı, uzun çalışma ömrüne ve ekonomik bir fiyata sahiptir. Lynx-6, dört adet Hitec HS-475, bir adet Hitec HS-422, bir adet HS-85 ve bir adet HS-81 servo motorlarına sahiptir.

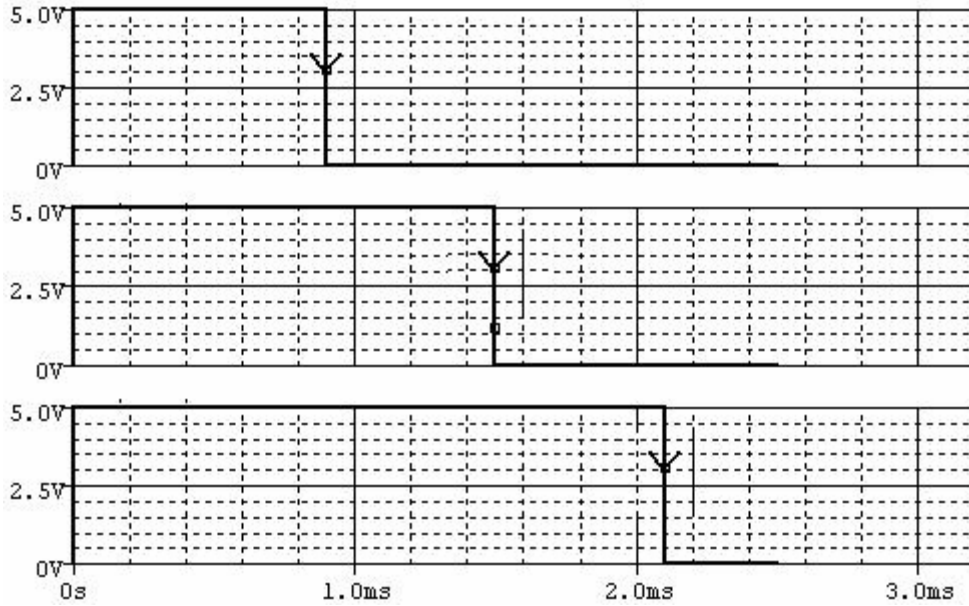
Lynx-6 beş eksen ve bir kavrayıcıdan oluşan toplam altı bağımsız ekleme sahiptir. Robot manipulatörü Şekil 3.4'te görülmektedir.



Şekil 3.4. Lynx-6 robot manipulatörü

3.6. Servo Motorlar

Robotun hareketi doğru akım servo motorlar tarafından sağlanmaktadır. Servo motorlar üç kabloya sahiptir; toprak, besleme ve denetim sinyali. Denetim sinyali bir kare dalgadır. Genişliği 0.9ms ile 2.1ms arasında değişmektedir, merkezi 1.5ms'dir. Denetim sinyalinin frekansı 50Hz'tir. Denetim sinyalinin dalga şekli Şekil 3.5'te görülmektedir. Motor milinin 0 derecede konumlanması için denetim sinyalinin genişliği 0.9ms, motor milinin 180 derecede konumlanması için denetim sinyalinin genişliği 2.1ms olması gerekmektedir. Denetim sinyalinin genişliği ve motor milinin açısı arasındaki ilişki Eş. 3.20'de görülmektedir. Bu eşitlikte Sg sinyal genişliği ve açı motor milinin açısıdır.



Şekil 3.5. Denetim sinyalinin dalga şekli

$$S_g = \frac{1,2 \times \text{açı}}{1000 \times 180} + \frac{0,9}{1000} \quad (3.20)$$

3.7. Manevra Kolu

Kullanılan manevra kolu 2 eksen ve 2 butona sahiptir. Manevra kolundaki bir eksen komutu veya eksen ve butondan oluşan birleşik komut, robot manipülatör üzerindeki belirlenmiş bir eksenin hareketini sağlar. Manevra kolu 9 adet çıkışa sahiptir. Çıkışlardan ikisi besleme, bir tanesi toprak ve diğerleri robotun denetimi için kullanılmaktadır.

Manevra kolunun hareketine göre manevra kolunun çıkış sinyalleri ve bu sinyallere karşılık gelen robotun eksen hareketleri Çizelge 3.1'de görülmektedir.

Çizelge 3.1. Manevra kolu sinyalleri-robotun eksen hareketleri

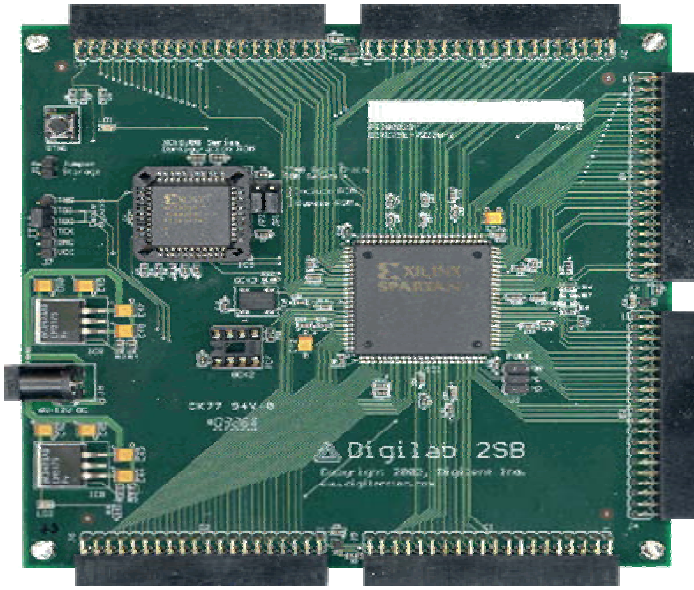
Eksen	Joystick Komutları	Eksen Hareketi	Açı	Kontrol Sinyalleri					
				Yukarı Pin 1	Aşağı Pin 2	Sol Pin 3	Sağ Pin 4	Buton1 Pin 5	Buton2 Pin 6
1	Sol	Sol	Azalt	0	0	1	0	0	0
1	Sağ	Sağ	Arttır	0	0	0	1	0	0
2	Aşağı	Yukarı	Arttır	1	0	0	0	0	0
2	Yukarı	Aşağı	Azalt	0	1	0	0	0	0
3	Aşağı	Yukarı	Azalt	1	0	0	0	0	1
3	Yukarı	Aşağı	Arttır	0	1	0	0	0	1
4	Aşağı	Yukarı	Arttır	1	0	0	0	1	0
4	Yukarı	Aşağı	Azalt	0	1	0	0	1	0
5	Sol	Sol	Arttır	0	0	1	0	0	1
5	Sağ	Sağ	Azalt	0	0	0	1	0	1
6	Sol	Aç	Arttır	0	0	1	0	1	0
6	Sağ	Kapat	Azalt	0	0	0	1	1	0

3.8. Denetim Kartı

Yapılan çalışmada Digilent D2SB FPGA tabanlı denetim kartı kullanılmıştır. Kullanılan kart ekonomik fiyata sahiptir ve ek modüller kullanılarak genişletilebilir.

Kart üzerinde yer alan FPGA kullanılarak, tasarlanan sayısal devreler gerçekleştirilebilir. Kartta yer alan FPGA Xilinx firmasının Spartan-IIE-200 isimli, PQ208 paketli modelidir.

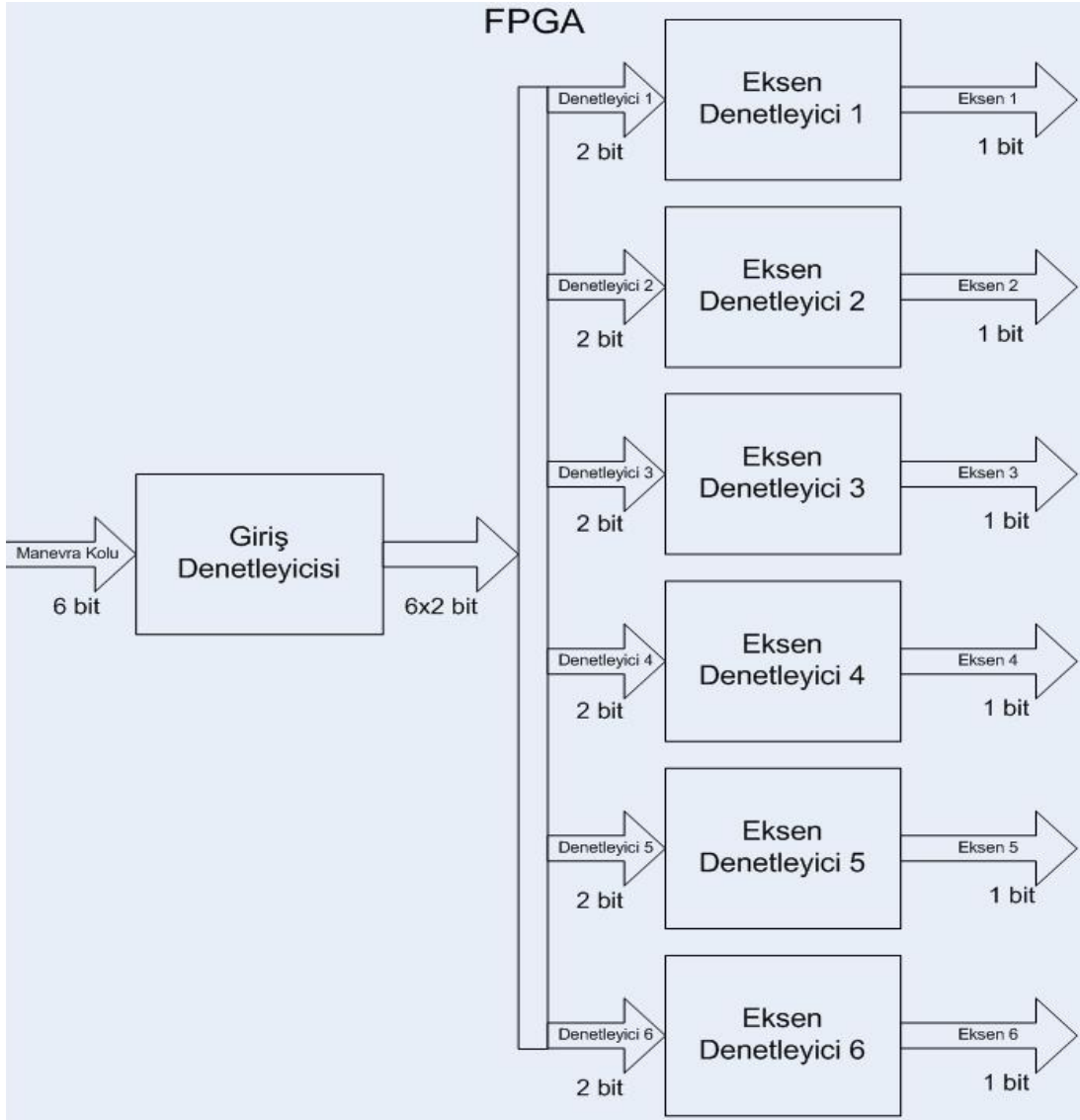
FPGA tasarlanan sayısal devreleri gerçekleştirmek için yeterli sayıda kapı ve giriş-çıkışa sahiptir. Denetim kartı Şekil 3.6'da görülmektedir. Kartın üzerinde 50MHz'lik osilatör yer almaktadır. Spartan-IIE-200 200k adet kapıya sahiptir.



Şekil 3.6. Denetim kartı

3.9. Geliştirilen Sayısal Tasarım

Gerçekleştirilen sayısal tasarımın blok şeması Şekil 3.7'de görülmektedir. Sistem yedi bloktan meydana gelmektedir. Bunlar giriş denetleyicisi ve eksen denetleyicileridir.



Şekil 3.7. Sayısal tasarımın blok şeması

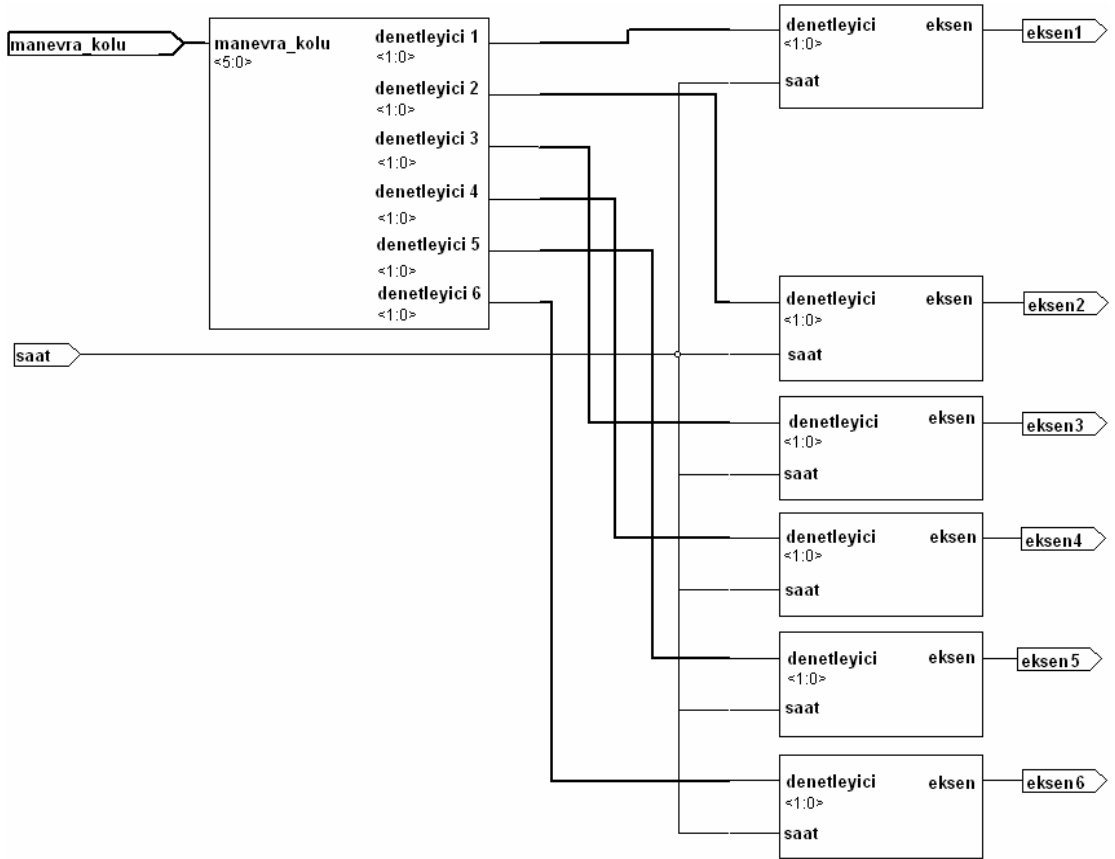
Sayısal tasarım Verilog dili kullanılarak geliştirilmiştir. Şekil 3.7'de verilen blok şemanın Verilog ile yazılmış kodunun bir bölümü Şekil 3.8'de görülmektedir. Bu kod Xilinx ISE programı kullanılarak sentezlenmiştir. Yazılan kodun sentezlenmesi sonucu oluşturulan donanım yapısı Şekil 3.9'da görülmektedir.


```

module deneme(clock,input_capture,axis1,axis2,axis3,axis4,axis5,axis6);
.
.
.
joy a (input_capture,control1,control2,control3,control4,control5,control6);
controller ax1 (clock,control1,axis1);
controller ax2 (clock,control2,axis2);
controller ax3 (clock,control3,axis3);
controller ax4 (clock,control4,axis4);
controller ax5 (clock,control5,axis5);
controller ax6 (clock,control6,axis6);
.
.
endmodule

```

Şekil 3.8. Verilog kodu örneği



Şekil 3.9. Verilog koda karşılık gelen donanım yapısı

Giriş denetleyicisi manevra kolundan gelen 6 bitlik sinyali okur ve Çizelge 3.1'de tanımlanmış denetim mantığına göre sinyali yorumlar. Hangi eksenin açılı değerinin azaltılacağını, artırılacağını ya da sabit kalacağını bulur ve gerekli eksen denetleyicisine açılı değerinin azaltılması, artırılması ya da sabit kalması için komut gönderir.

Giriş denetleyicisi ile eksen denetleyicileri arasında haberleşmek için kullanılan denetleyici sinyalleri yer almaktadır. Denetleyici sinyalinin alabileceği değerler ve bu değerlere karşılık gelen komut bilgisi Çizelge 3.2'de görülmektedir.

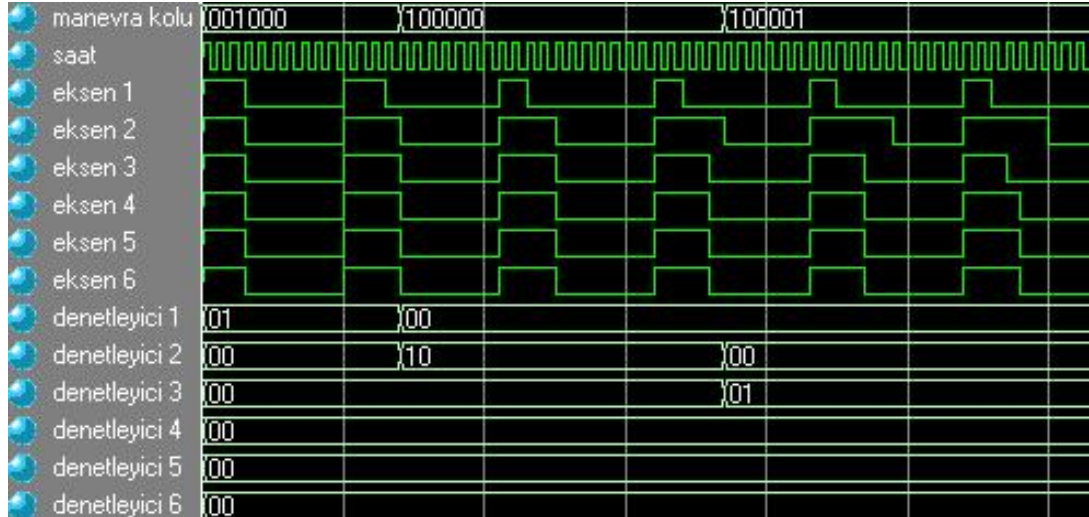
Çizelge 3.2. Denetleyici sinyalleri

1. bit	2. bit	Komut
0	0	Hareketsiz
0	1	Azalt
1	0	Arttır
1	1	Hareketsiz

Eksen denetleyicisi, robotun eksenlerinde yer alan servo motorların denetlenmesi için PWM sinyalleri üretir. Eksen denetleyicisi girişindeki denetleyici sinyalini kontrol eder, Çizelge 3.2'de belirlenmiş koşullara göre yorumlar ve çıkışında yer alan eksen sinyalini genişliğini arttırır, azaltır ya da sabit bırakır.

Gerçekleştirilen sayısal tasarımın benzetim çıktısı Şekil 3.10'da görülmektedir. Sistem saati 50 MHz'tir. Bu çalışma frekansı ile robotun eksenlerindeki servo motorların hareket doğruluğu 0.0003 derece olmaktadır. Saat ve manevra kolu giriş sinyalleridir. Eksenler sistemin çıkış sinyalleridir. Denetleyici sinyalleri giriş denetleyicisi ve eksen denetleyicileri arasındaki denetleme sinyalleridir.

Şekil 3.5'te görüldüğü üzere eksen denetleyicileri periyodu 20 ms olan ve genişliği 0.9 ms ile 2.1 ms arasında değişen eksen sinyalleri üretmektedir. Bu çalışma için geliştirilen sayısal tasarımın kodları EK-1, EK-2 ve EK-3 bölümlerinde verilmiştir.



Şekil 3.10. Sistemin giriş ve çıkışlarının dalga şekilleri

3.10. Güç Kaynağı

Güç kaynağı iki farklı voltaj çıkışına sahiptir, $V_{\text{çıkış1}}$ ve $V_{\text{çıkış2}}$. $V_{\text{çıkış1}}$ robot manipülatörün sahip olduğu servo motorların beslemesi için kullanılmaktadır. $V_{\text{çıkış1}} = 6V$ DA (doğru akım) voltaj ve 2.5A DA akım verebilmektedir. $V_{\text{çıkış2}}$ çıkışı ise elektronik kontrol kartının beslenilmesi için kullanılabilir. $V_{\text{çıkış2}} = 5V$ DA voltaj ve 150mA DA akım verebilmektedir. Ancak bu çalışmada kullanılan FPGA kartının kendi güç kaynağı mevcuttur.

4. ROBOT MANİPÜLATÖRÜN KARTEZYEN-ÖZGÜL DENETİMİ

4.1. Giriş

Gerçek zamanlı robot denetimi zor ve uğraştırıcı bir araştırma konusudur. Literatürde robot sistemlerinin çeşitli sayısal tekniklerle denetimi konusunda çalışmalar bulunmaktadır [15-19]. Eğer robot manipülatörün hareketi çalışma uzayı değişkenleri (örneğin robotun ucunun konum vektörü) cinsinden bildirilmek istenirse, bilinen eklem uzayı değişkenleri (örneğin eklem açıları) kendilerine karşılık gelen çalışma uzayı değişkenlerine düz kinematik yöntemi kullanarak dönüştürülebilirler. Eğer manipülatörün hareketi eklem uzayı değişkenleri cinsinden bildirilmek istenirse, bilinen çalışma uzayı değişkenleri kendilerine karşılık gelen eklem uzayı değişkenlerine ters kinematik yöntemi kullanılarak dönüştürülebilir.

Çalışma uzayı denetim projeleri istenilen yerde konumlanmayı başarıyla gerçekleştiren sıradan PID (Orantılı entegral türev) ve PD (Orantılı türev) denetimleri üzerine kurulmuştur [20-22]. PD denetimi basit ve kararlı olduğu için robot manipülatörlerde en çok kullanılan stratejidir. Buna ek olarak, daha ileri denetimler sıklıkla kendi denetim döngülerinde PD algotirmalarını istenilen sonucu elde etmek için kullanırlar [23]. Örneğin, hesaplanmış tork gibi model tabanlı robot denetleyicileri PD+ ile [24-27], lineer olmayan PDE denetimi PD ile [28] ve PD denetimi hesaplanmış ileri besleme ile [29] beraber kullanılmaktadır.

Alan programlanabilir mantık dizileri (FPGA) hafızasına veri akışı (bit stream) yüklenerek kofigüre edilebilir mantık kapılarından oluşan bir dizidir.

Sıradan teknolojilerle karşılaştırıldığında FPGA yüksek hız, düşük güç, kısa geliştirme zamanı ve düşük fiyat gibi artı özelliklere sahiptir. Mikroişlemci/DSP ve ASIC (uygulamaya özgül tüm devre) arasında bir imkân sunmaktadır. Mikroişlemci/DSP'den daha iyi performansa ve ASIC' ten daha kısa geliştirme zamanı ve daha ucuz fiyata sahiptir [30].

Bu çalışmada DSP (Sayısal işaret işleme) uygulamaları için geliştirilmiş bir FPGA kullanılarak üç eksenli robotun kartezyen-özgül denetimi gerçekleştirilmiştir. Kartezyen-özgül denetim ile anlatılmak istenilen robotun ucunun denetimidir. Tasarlanılan sistemde kullanıcı robotun hangi yöne doğru hareket etmesini istiyorsa manevra kolunu o yöne doğru ittirir. FPGA içerisinde yer alan sayısal tasarım girişindeki manevra kolundan gelen sinyalleri kontrol eder ve robotun istenilen hareketi yerine getirmesi için gerekli olan kare dalgaları ilgili eksenlere göndermek üzere üretir.

Bundan sonra gelen bölümlerde sırasıyla, FPGA, geliştirilen sistemin genel blok şeması, kullanılan robot manipülatörün matematiksel modeli, ters kinematik, denetleyici tasarımı, servo sürücüsü ve FPGA kartı başlıklarından bahsedilecektir.

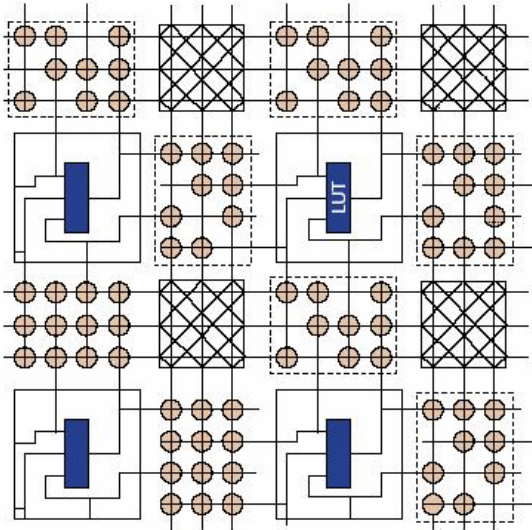
4.2. FPGA (Alan Programlanabilir Kapı Dizileri)

FPGA'lar herhangi bir sayısal fonksiyonu gerçekleştirebilmek için kullanıcı tarafından programlanabilen entegre devrelerdir. Günümüzde FPGA'lar genelleştirilmiş sayısal kaynakları, programlanabilen bağları, mikroişlemcileri, çarpıcıları, hafızaları ve diğer birçok genel çekirdek yapısını içerisinde bulunduran tek yonga yapılarıdır. İçerisindeki bağlantılar, hafızasına gerekli bit akıntısı yüklenerek yapılandırılabilen mantık kapıları dizisidir.

Mikroişlemci/DSP ve ASIC (uygulamaya özgül tüm devre) arasında bir imkân sunmaktadır. Mikroişlemci/DSP'den daha iyi performansa ve ASIC' ten daha kısa geliştirme zamanı ve daha ucuz fiyata sahiptir [30].

4.2.1. FPGA mimarisi

FPGA, mantık hücrelerinden meydana gelir. Bunlar, Girdi/Çıktı hücrelerine ya da diğer mantık hücrelerine bağlanabilirler. Mantık hücreleri programlanabilen tümleşik elemanlara sahiptirler. FPGA içine gömülecek tasarım mantık hücrelerinin fonksiyonlarını ve birbirleriyle olan bağlantılarını tanımlayan bit akıntısı ile tanımlanır. Şekil 4.1'de FPGA içerisinde bulunan mantık hücresi görülmektedir. Mantık hücresi, dikdörtgen şeklinde ve dört girişli olan LUT (look up table)'lardan ve programlanabilen yol kaynaklarından meydana gelmektedir. LUT içerisinde AND, OR, XOR gibi basit mantık devreleri yer almaktadır. FPGA'nın içine yazılacak olan programa göre bu LUT'lardan uygun fonksiyonlar seçilir ve uygun yollar oluşturularak FPGA'nın istenilen işi yapması için programlanması sağlanır [30].



Şekil 4.1. Mantık hücresi

Şimdiye kadar bahsedilen FPGA mimarisinin haricinde günümüz teknolojisinde sık kullanılan sayısal fonksiyonlar FPGA içerisinde toplanmıştır ve kullanıcılar bu fonksiyonlara sahip donanımı tek yonga içerisinde tasarlayabilirler. Gerçekleştirilen tasarımın tek yonga içinde olması, maliyetin azalması ve devrenin kapladığı yerin azalmasının yanı sıra kullanılan sayısal donanımlar aynı yonga içinde toplandığı için performansta da büyük bir artış sağlanmaktadır. FPGA içerisinde: toplayıcı, çarpıcı, bölücü, kod çözücü, mikroişlemci, programlanabilen girdi/çıkış blokları yer almaktadır. Bahsedilen donanımsal yapıların yanısıra FPGA içerisinde yer alan mantık hücreleri kullanılarak gerçekleştirilen IP (intellectual property) çekirdekleri çeşitli firmalar tarafından geliştirilmektedir ve bunlar yapılan tasarım içine entegre edilebilirler. IP'ye bus ara yüzleri, hafıza ara yüzleri, sinyal işleme fonksiyonları, mikroişlemciler örnek olarak verilebilir. IP kullanımı sonucunda tasarım zamanı düşürülebilir [30].

4.2.2. Mikroişlemci ile FPGA'nın karşılaştırılması

Mikroişlemciler, sayısal sistemler için kullanım kolaylığı, güçlü ve esnek uygulama ortamı sağlamaktadırlar. Çok sayıda değişik modele, fiyata ve performansa sahiptirler. Gömülü sistemlerdeki bu özellikleri mikroişlemcileri robotik uygulamalarında tercih haline getirmektedir. Bununla birlikte, yazılım geliştirmek için kullanılan ortamlar rahatlıkla bulunabilmektedir (compiler, debugger, library, operating systems). Fakat bu özellikleri mikroişlemcileri birçok uygulama için birinci tercih haline getirmez. Bunun sebebi gerekli sistemlerin optimize edilmiş, güçlü işlem yeteneğine sahip ve küçük kart alanı kaplamalarının istenmesidir [30].

4.2.3. ASIC (uygulamaya özgül tüm devre) ile FPGA'nın karşılaştırılması

ASIC ve FPGA, mikroişlemciden çok daha fazla seviyede paralel işlem kapasitesine ve bu yüzden birkaç kat daha fazla performansa sahiptirler. FPGA, ASIC'e göre çok daha düşük NRE' ye (tasarım, test, prototip, mühendislik masrafları) sahiptir. FPGA, ASIC'e göre daha fazla birim fiyatına sahiptir fakat robotikte kullanımları çok sayıda olmadığı için üretilecek ASIC'in maliyeti çok yüksek olacaktır bu yüzden FPGA daha ucuz bir çözümdür. Günümüzdeki uygulamalarda FPGA kullanımı artmaktadır, ASIC'ler en yüksek performansın zorunlu olduğu veya çok sayıda kullanımın olacağı zaman tercih edilirler [31].

4.2.4. FPGA ile tasarım

Kullanıcı, donanım tanımlama dilini (VHDL ya da Verilog) kullanarak FPGA ile uygulamasını yapmak istediği devreyi tasarlar. Bilgisayar destekli tasarım yazılımı (CAD) yardımıyla yazılan kod sentezlenir. Diğer bir CAD yazılımı kullanılarak sentezlenen koda karşılık uygun bağlantılar FPGA içinde gerçekleştirilir. Cad yazılımı gerekli veri akışını oluşturup FPGA içerisine aktararak FPGA'yı programlar [31].

4.2.5. FPGA ile tasarımda kullanılan araçlar

Tasarım yapılırken kullanılan birçok mühendislik aracı bulunmaktadır. Bunlar arasında yüksek seviyeli programlama dillerine örnek olarak C ve C++ dilleri ile yapılan programlar örnek olarak verilebilir. Bunlara ek olarak MATLAB/SIMULINK programı da tasarım aracı olarak kullanılmaktadır. Xilinx firmasının geliştirdiği Xilinx ISE WebPACK isimli programda VHDL ve Verilog programlama dilleri kullanılarak tasarım gerçekleştirmek mümkündür.

Aynı zamanda Altera firmasının ürünü olan Altera Quartus II isimli VHDL ve Verilog programlama dilleri kullanılarak tasarım gerçekleştirilebilir.

4.2.6. FPGA'nın uygulama örnekleri

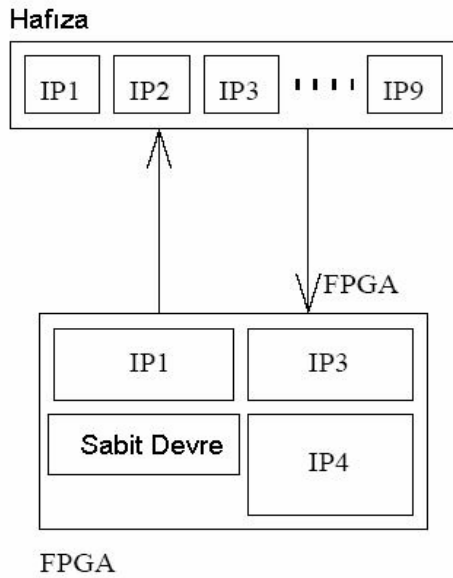
Mantık devreleri ile yer değiştirme (Logic replacement)

FPGA ilk olarak bu amaçla kullanılmıştır ve birçok sayıda küçük ve orta büyüklükteki mantık devresinin yerini almıştır. Bu kullanımın getirdiği büyük avantajlardan birisi devrede herhangi bir değişiklik yapılmak istenildiğinde ya da devre hatalı olduğunda kartta herhangi bir değişiklik yapılmadan FPGA içindeki programın değiştirilmesinin yeterli olmasıdır. Bu kullanıma örnek olarak bir mikroişlemci ile çevresel elemanlarının ara yüzünün gerçekleştirilmesi verilebilir. Gerekli olan adres dekoderleri (address decoders), hafıza denetleyicileri (memory controllers), bus ara yüzleri (interfaces) ve motor denetleyicileri (controllers) tek FPGA içerisinde tasarlanabilirler. Günümüzde FPGA büyük mantık devrelerinin tek entegre olarak gerçekleştirilmesi için kullanılmaktadır. Bu kullanım pazara sunulacak ürünün tasarım zamanını, güç tüketimini ve kapladığı yeri azaltmaktadır. FPGA ayrıca ASIC devrelerinin prototiplerini tasarlamak için kullanılmaktadır [30].

Şekillendirilebilir hesaplama (Reconfigurable computing)

FPGA genel amaçlı mantıksal eleman olduğu için yüksek performanslı hesaplama işlemleri için kullanılabilir. Geliştirilen bir algoritmanın donanım olarak gerçekleştirilmesi paralel yapıların kullanılmasına ve mikroişlemcilerden çok daha hızlı işlem yapılmasına imkân vermektedir. FPGA'nın en önemli özelliklerinden birisi de çalışma sahasında hatta çalışırken bile tekrar programlanabilmesidir. FPGA içersine bit akıntısı yüklenerek programlanabilir.

Bu işlem internet üzerinden ya da başka uzaktan erişim protokolleri kullanılarak yapılabilir. Şekil 4.2’de FPGA’nın şekillendirilebilir kullanımına bir örnek verilmektedir. Bu örnekte, FPGA içerisinde sabit devre ve IP fonksiyonları yer almaktadır. FPGA dışındaki hafızada ise gerek duyulduğunda kullanmak için IP fonksiyonları bulunmaktadır. FPGA’nın normalde alçak geçirgen filtre görevini yerine getirmesi ve bazı koşullar gerçekleştiğinde yüksek geçirgen filtre olarak çalışmasını gerektiren bir uygulama örnek olarak ele alınırsa. Normalde FPGA alçak geçirgen filtre olarak çalışacaktır fakat tanımlanmış koşullar yerine geldiğinde yüksek geçirgen filtre olarak görev yapması gerekecektir ve bunun için yüksek geçirgen filtre IP’sini hafızadan alarak kendi içerisine yazacak ve yüksek geçirgen filtre görevini yerine getirecektir [30].



Şekil 4.2. Şekillendirilebilir uygulama örneği

Mars Lander

Resim 4.1’de görülen Mars Pathfinder ve Mars Surveyor isimli araçlarda, FPGA uzay mekiğinden gelen komutları çözmek için, kameralar ile alınan görüntü bilgilerinin depolanması için ve araçların hareketini sağlayan adım (step)

motorların denetimi için kullanılmaktadır. FPGA uzay uygulamalarında büyük bir avantaja sahiptir, uzayda radyasyon dünya atmosferinden çok daha fazladır, FPGA'ların radyasyona dayanıklı üretilen paketleri bulunmaktadır ve tasarlanan sistem tek yonga olduğu için radyasyona karşı korunma sağlanır [32].



Resim 4.1. Mars Pathfinder ve Mars Surveyor

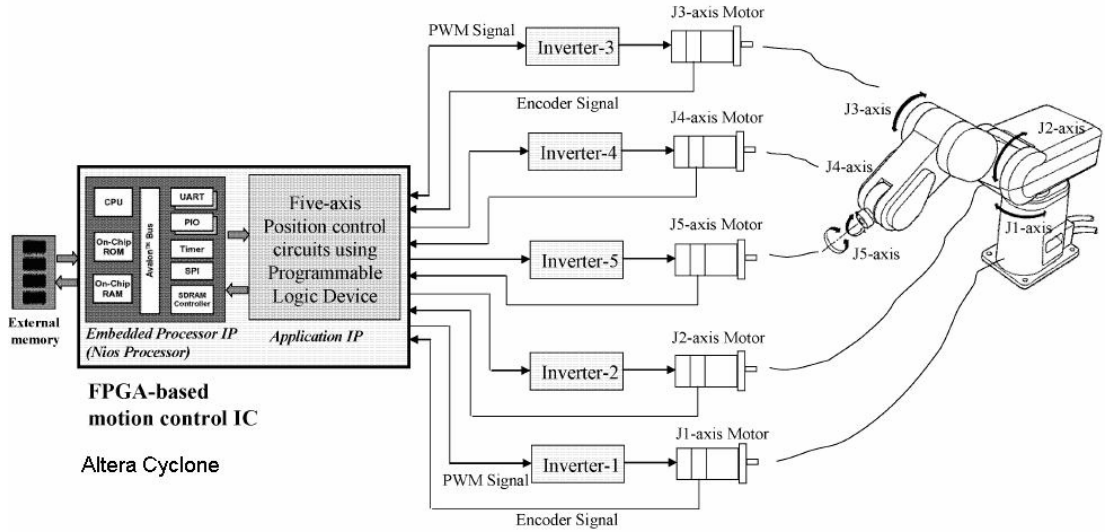
Formula 1

F1 arabaları tam anlamıyla robot olmamalarına karşın robotlarla aynı özellikleri taşımaktadırlar. 2003 BMW Williams isimli araba araç denetim ve görüntüleme sistemi için Texas Instruments DSP ve Xilinx Virtex FPGA tabanlı elektronik kartı kullanmaktadır.

Bu kart, vites deęiřtirme iřlemi, ekiři arttırmak ve lastiklerin ařınmasını nlemek iin lastiklerin kaymasını, gvenli dnř yapabilmek iin diferansiyeli, maksimum ivmeli bir kalkıř yapabilmek iin arabanın kalkıř sistemini denetler. Aracın zerinde bulunan yaklaşık 220 adet sensrden bilgiyi alarak kaydetme iřlemi yerine getirir.

Robot maniplatr iin FPGA tabanlı denetim devresi

nceden de bahsedildięi gibi yeni retilen FPGA teknolojileri gml iřlemci IP'sini ve uygulama IP'sini aynı yonga ierisinde tasarlamaya imkn vermektedir. Bu alıřmada robot maniplatrn hareketini denetlemek iin tasarlanan entegre devresi geliřtirilmiřtir. FPGA tabanlı hareket denetim entegresi iki IP'ye sahiptir. Bunlar Nios gml iřlemci IP'si ve uygulama IP'sidir. Nios iřlemci komut retimi, ters kinematik hesaplamaları ve noktadan noktaya hareket denetimini yerine getirmektedir. Uygulama IP'si robotun 5 ekseninin hareket kontroln yerine getirmektedir. Geliřtirilen sistemin blok gsterimi Őekil 4.3'te grlmektedir [34].

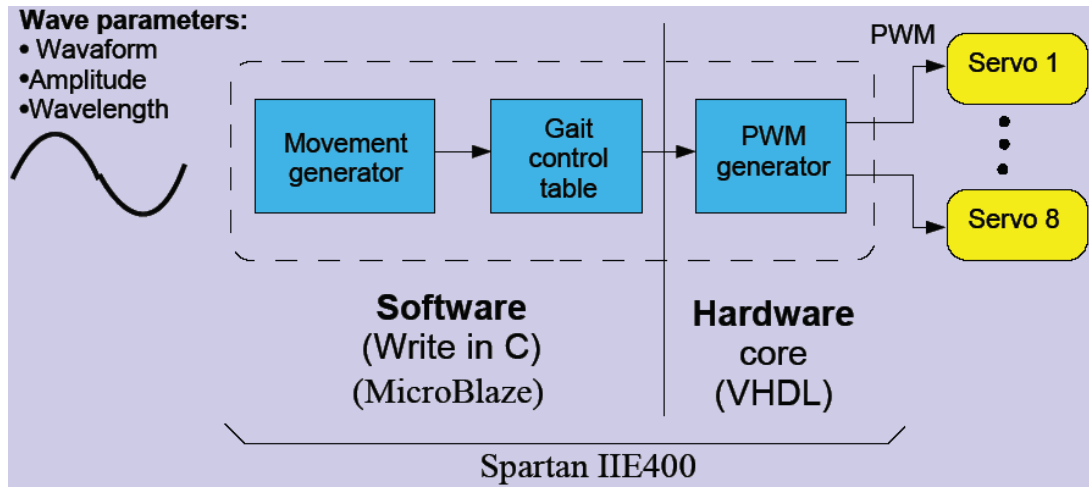


Şekil 4.3. Robot manipülatör denetim devresi

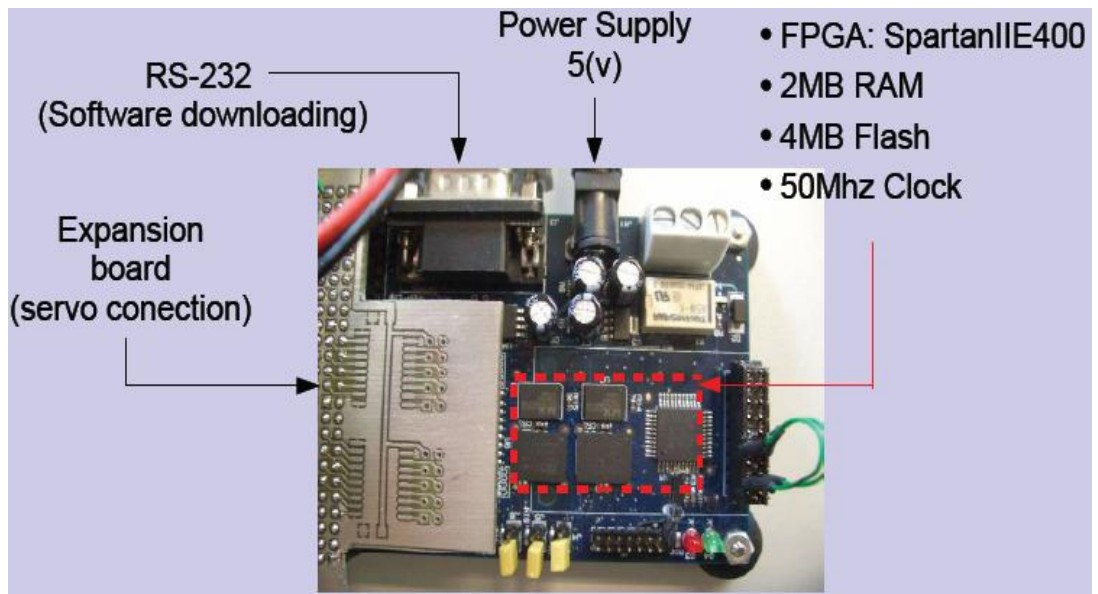
Solucansı robotun FPGA tabanlı denetimi

Şekillendirilebilir modüler robotlar çok yönlülük, sağlamlık ve düşük fiyat gibi avantajları sunmaktadır. Bu robotlar birbirlerine takılıp çıkarılabilen basit ve küçük parçalardan meydana gelmektedirler. Sekiz benzer modülden meydana gelen solucansı robotun denetimi ile ilgili yapılan çalışmada FPGA tabanlı işlemci tasarlanmış ve kullanılmıştır.

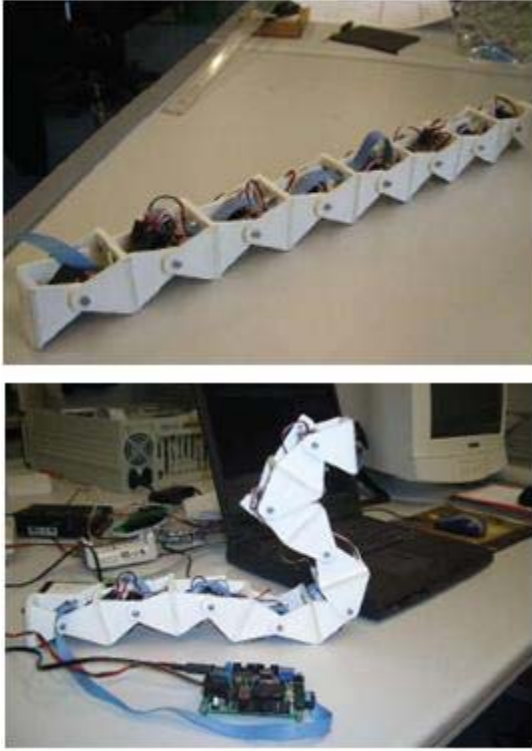
Bu işlemci VHDL donanım tanımlama dili kullanılarak geliştirilmiştir ve robotun sekiz eklemine paralel olarak denetleme yeteneğine sahiptir. Geliştirilen denetim sistemi Şekil 4.4'te, kullanılan denetim kartı Şekil 4.5'te, denetlenen solucansı robot Resim 4.3'te görülmektedir [35].



Şekil 4.4. Geliştirilen denetim sistemi



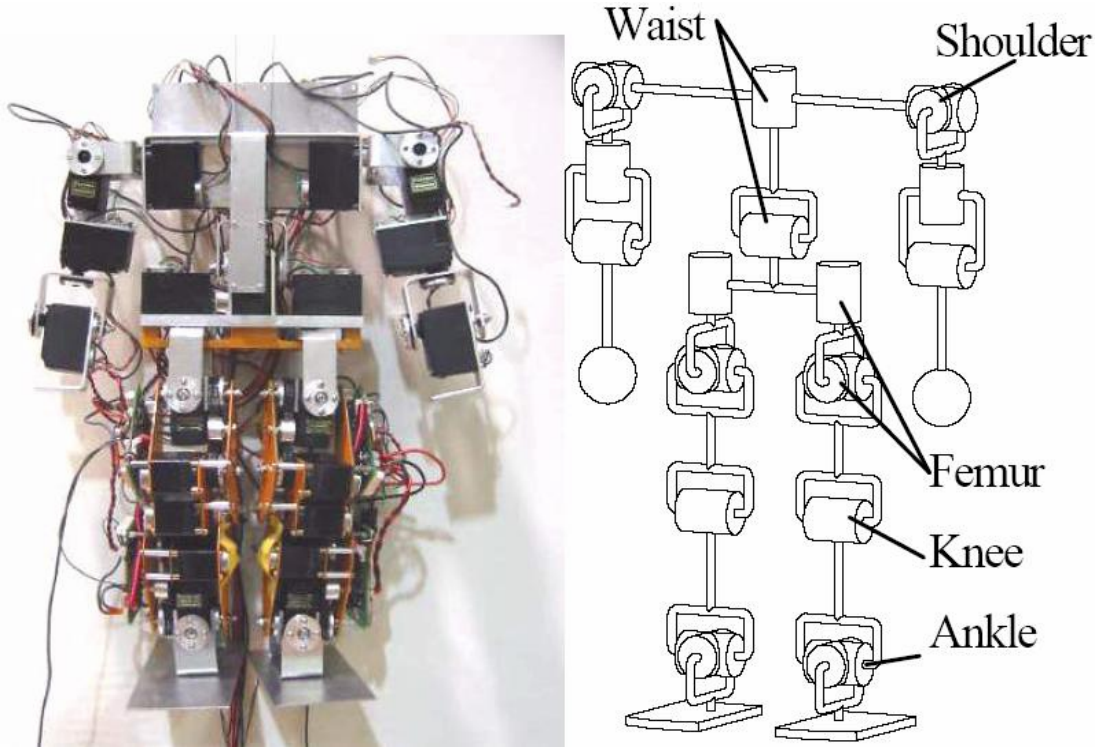
Şekil 4.5. Kullanılan denetim kartı



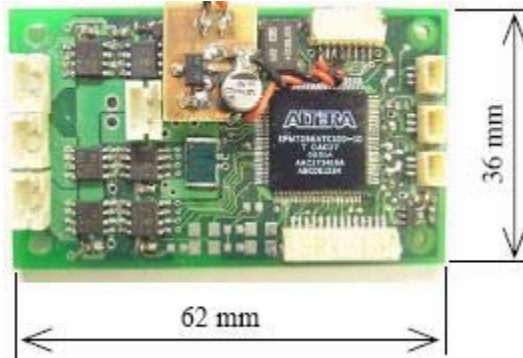
Resim 4.2. Solucansı robot

İnsansı robot için FPGA tabanlı denetim devresi

Bu çalışmada Şekil 4.6'da görülen insansı robot için denetleme sistemi geliştirilmiştir. Gerçek zamanlı seri ara yüz bağlantısı olan servo denetleyicisi gerçekleştirilmiştir. Bu servo denetleyicisi tork denetim özelliğine sahiptir. Servo denetleyici kullanılarak 22 eksenli insansı robotun denetimi yapılmıştır. Bu insansı robot 22 adet ekleme, 313mm yüksekliğe, 230mm genişliğe, 100mm derinliğe ve 2kg ağırlığa sahiptir. FPGA'nın paralel işlem yeteneği kullanılarak 22 adet servo motor gerçek anlamda paralel olarak denetlenmiştir [36].



Şekil 4.6. İnsansı robot



Resim 4.3. Kullanılan denetim kartı

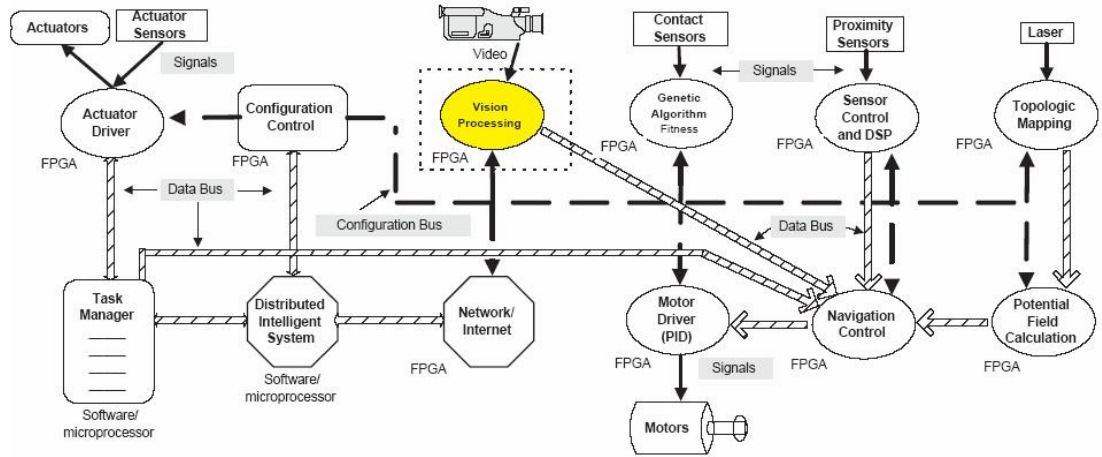
Mobil robotlar için hareket tanıma kartı

Bu çalışmada bir mobil robotun denetlenmesi için gerekli denetim sistemi ve Resim 4.5'te görülen gömülü görme sistemi FPGA kullanılarak gerçekleştirilmiştir. Görme sistemi renkli sayısal kamera kullanarak hareketleri algılamaktadır.

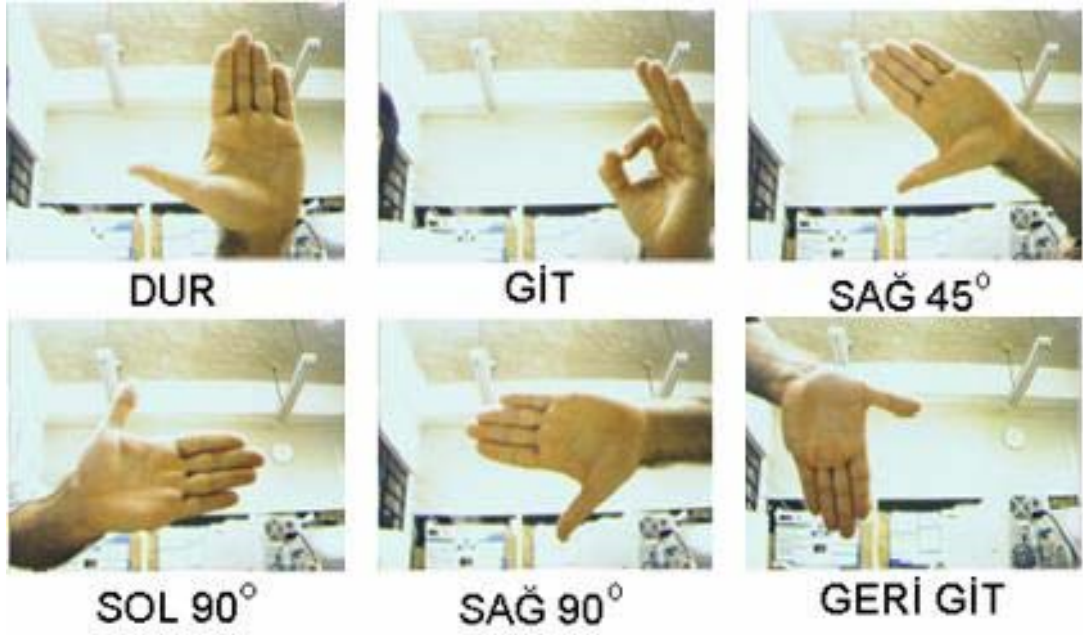
Algılanan hareketler FPGA üzerinde bulunan donanımlar kullanılarak tanımlanabilmektedir. Hem algılamaya hem de tanımlama işlemi aynı FPGA içinde tasarlanmış donanımlar tarafından gerçekleştirilmektedir. Bu çalışmada da FPGA'nın paralel işlem yeteneği kullanılmıştır ve Şekil 4.17'de görülen sistemlerden görme işlemcisi hariç diğerleri aynı FPGA içerisinde yer almaktadır. Geliştirilen sistem sayesinde mobil robot Şekil 4.8'de görülen el hareketleri ile denetlenebilmektedir [37].



Resim 4.4. Görme sistemi



Şekil 4.7. Geliştirilen sistem



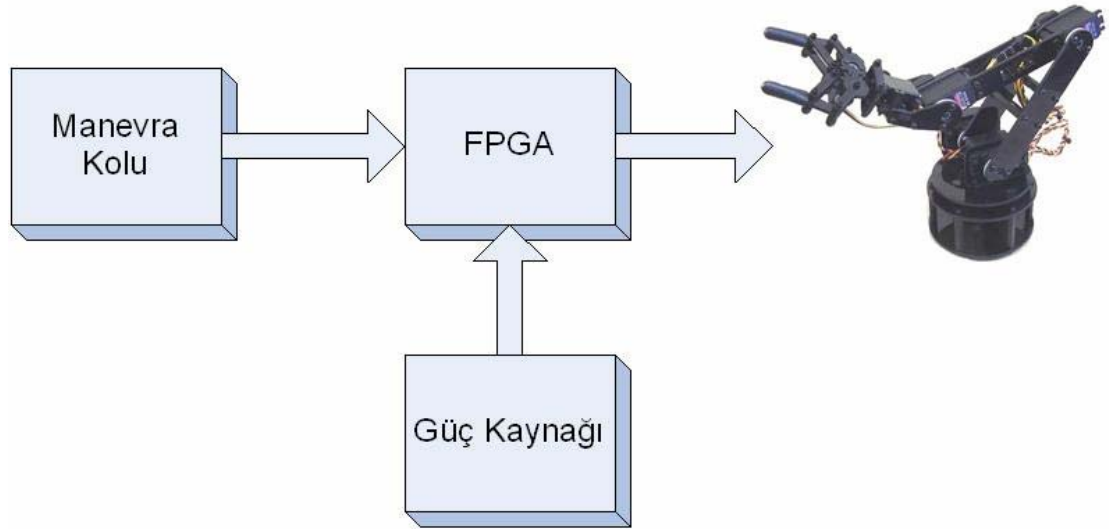
Şekil 4.8. El komutları

4.3. Geliştirilen Sistemin Genel Blok Şeması

Geliştirilen sistemin blok şeması Şekil 4.9'da görülmektedir. Sistem üç ana bloktan meydana gelmektedir. Bunlar güç kaynağı, FPGA kartı ve robottur. Lynx-6 robot manipülatörü, Lynx Motion firmasının bir ürünüdür, Bu çalışmada sadece Şekil 3.4'te verilen ikinci, üçüncü ve dördüncü eklemleri denetlenmiştir.

Bu çalışmada robot manipülatörün kartezyen-özümlü denetimi gerçekleştirilmiştir. Kartezyen-özümlü denetim ile anlatılmak istenilen robotun ucunun denetimidir. Tasarlanılan sistemde kullanıcı robotun hangi yöne doğru hareket etmesini istiyorsa manevra kolunu o yöne doğru ittirir. FPGA içerisinde yer alan sayısal tasarım girişindeki manevra kolundan gelen sinyalleri kontrol eder ve robotun istenilen hareketi yerine getirmesi için gerekli olan kare dalgaları ilgili eksenlere göndermek üzere üretir.

Bu çalışmada DSP uygulamaları için geliştirilmiş bir FPGA kullanılarak üç eksenli robotun x ve y eksenlerinde kartezyen denetimi gerçekleştirilmiştir.



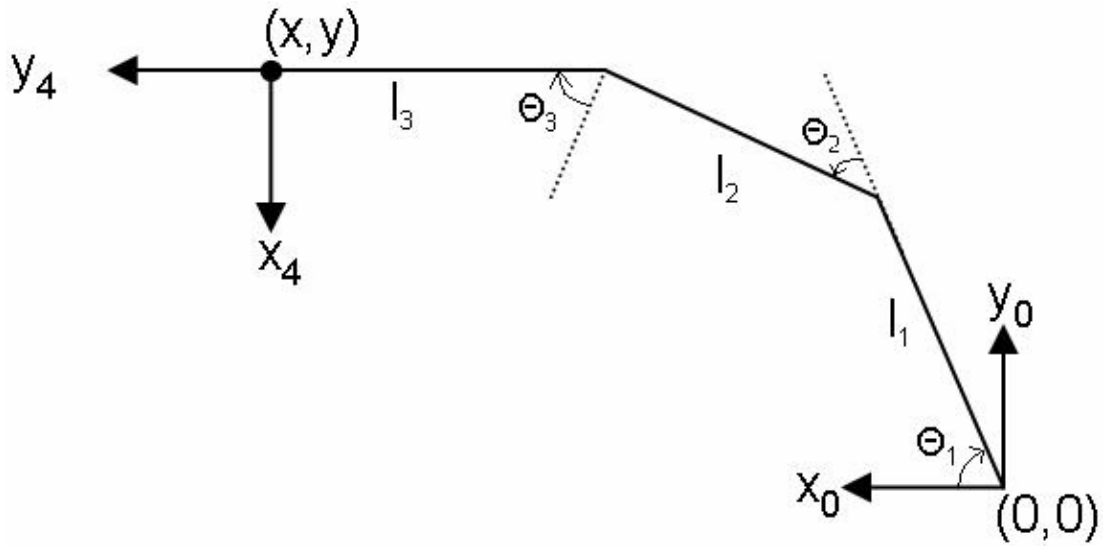
Şekil 4.9. Sistemin genel blok şeması

4.4. Manipülatörün Matematiksel Modeli

Homojen koordinat sistemleri atamaları robot manipülatörün ucunun istenilen noktada konumlandırılabilmesi için ve pozisyon denetlemesinin sağlanabilmesi için gerekli olan matematiksel analizlerin yapılmasında gerçekleştirilmesi gereken ilk aşamadır.

Robot manipülatörün geometrisi Şekil 4.10'da görülmektedir. Robotun sahip olduğu her bir ekleme birer koordinat sistemi atanmaktadır. Kullanılan robot 3 serbestlik dereceli ve düzlemseldir. Robotun konumunu ve yerini hesaplamak için kullanılacak olan düz kinematik denklemleri Eş. 4.1'de görülmektedir. Θ_i değerleri ekleme açılarını, l_i uzuv uzunluklarını ifade etmektedir. x_0y_0 referans koordinat sistemini, x_4y_4 robotun ucuna atanmış koordinat sistemini göstermektedir.

(x, y) robotun ucunun referans koordinat sistemine göre tanımlı konumunu, h, k, m, n ise robotun referans koordinat sistemine göre yönünü göstermektedir.



Şekil 4.10. Robot manipülatörün geometrisi

$$A = \begin{bmatrix} h & k & x \\ m & n & y \end{bmatrix}$$

$$h = -\cos(\theta_1 - \theta_2 + \theta_3)$$

$$k = \sin(\theta_1 - \theta_2 + \theta_3)$$

$$m = -\sin(\theta_1 - \theta_2 + \theta_3) \quad (4.1)$$

$$n = -\cos(\theta_1 - \theta_2 + \theta_3)$$

$$x = l_3 \sin(\theta_1 - \theta_2 + \theta_3) + l_2 \cos(\theta_1 - \theta_2) + l_1 \cos(\theta_1)$$

$$y = -l_3 \cos(\theta_1 - \theta_2 + \theta_3) + l_2 \sin(\theta_1 - \theta_2) + l_1 \sin(\theta_1)$$

4.4.1. Ters kinematik

Ters kinematik problemi robotun ucunun bilinen konum ve yön bilgilerinden eklem değişkenlerinin saptanmasıdır. Robotun çalışma uzayında tanımlanmış olan görevini eklem uzayı hareketine çevirebilmek için çok büyük önem taşımaktadır. Ters kinematik problemi birden fazla çözüm veya ulaşılamayan tekil (singular) noktaların bulunması nedeniyle komplekstir. Birden fazla çözümü ya da ulaşılamayan noktaları gidermek için bu çalışmada ters kinematik denklemleri çözülerek elde edilen eklem açılarının robotun çalışma uzayında olup olmadığını denetleyen bir denetim mekanizması yer almaktadır. Eş. 4.2'de sistemin ters kinematik denklemleri verilmiştir. Bu eşitlikte görülen a 'nın değeri 100'dür, FPGA içerisinde ondalıklı işlemlerden kurtulmak için kullanılmıştır.

$$\theta_1 = \arctan\left(\frac{ya}{x-l_3a}\right) + \frac{\theta_2}{2}$$

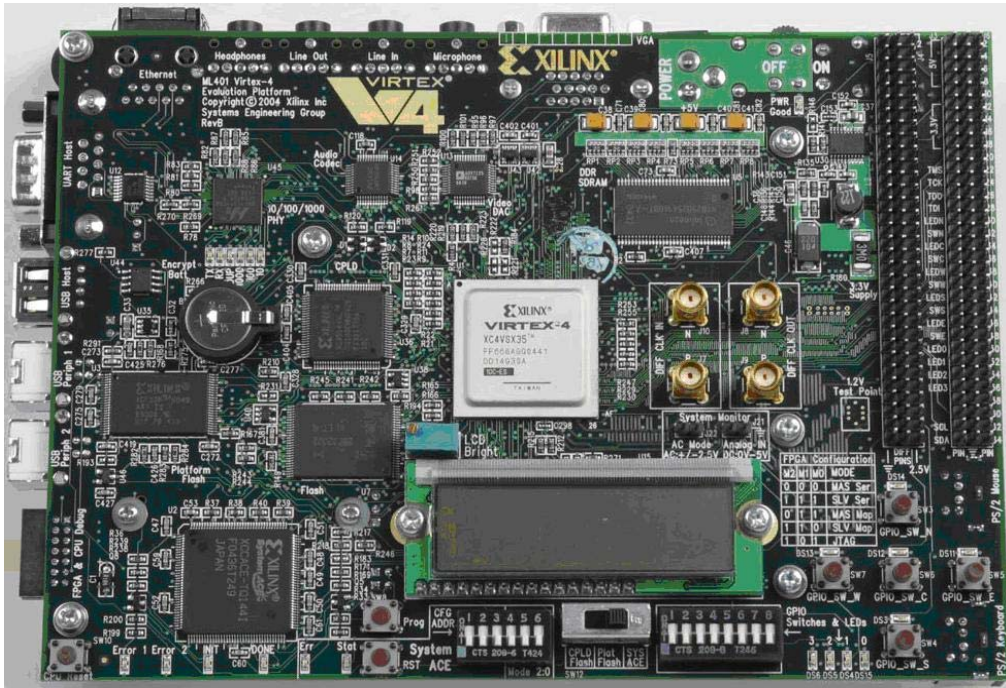
$$\theta_2 = \arccos\left(\frac{(x-l_3a)^2 + y^2}{2l_1^2a} - a\right) \quad (4.2)$$

$$\theta_3 = 90 + \theta_2 - \theta_1$$

4.5. Denetim Kartı

Bu çalışmada Virtex-4 ML402 SX kartı kullanılmıştır (Resim 4.5). Bu kart Virtex-4 SX35 tabanlıdır ve standart endüstriyel çevresel ara yüz ve bağlantıları desteklemektedir. Bu kart birçok farklı uygulama alanına hitap etmektedir. Virtex-4 SX35 200000 adet mantıksal hücreye ve çok sayıda IP'ye (Intellectual Property) sahiptir. FPGA'nın maksimum çalışma frekansı 500MHz'tir. 192 adet ExtremeDSP dilimine sahiptir.

Her bir DSP48 dilimi 2 girişli çarpıcıyı (multiplier) takip eden çoklayıcıya (multiplexer) ve 3 girişli toplayıcıya (adder)/çıkarcıya (subtractor) sahiptir. Çarpıcı iki tane 18 bitlik ikiye tamamlayıcı (two's complement) girdi sonucunda 36 bitlik ikiye tamamlayıcı formatında çıktı verir. Toplayıcı/çıkarcı 48 bitlik ikiye tamamlayıcı girdi kabul eder ve 48 bitlik ikiye tamamlayıcı formatında çıktı verir.



Resim 4.5. Denetim kartı

4.6. Denetleyici Tasarımı

Bu çalışmada bahsedilen düz ve ters kinematik denklemlerini çözmek için kullanılan Virtex SX35 isimli FPGA 192 adet DSP işlemlerinde kullanılmak için tasarlanan XtremeDSP dilimine sahiptir. Bu dilimler genel matematiksel işlemleri donanımsal olarak yerine getirmektedirler ve maksimum 500MHz'te çalışmaktadırlar. Bu çalışmada XtremeDSP dilimleri paralel olarak kullanılarak ters kinematik problemi hızlı bir şekilde çözülmektedir.

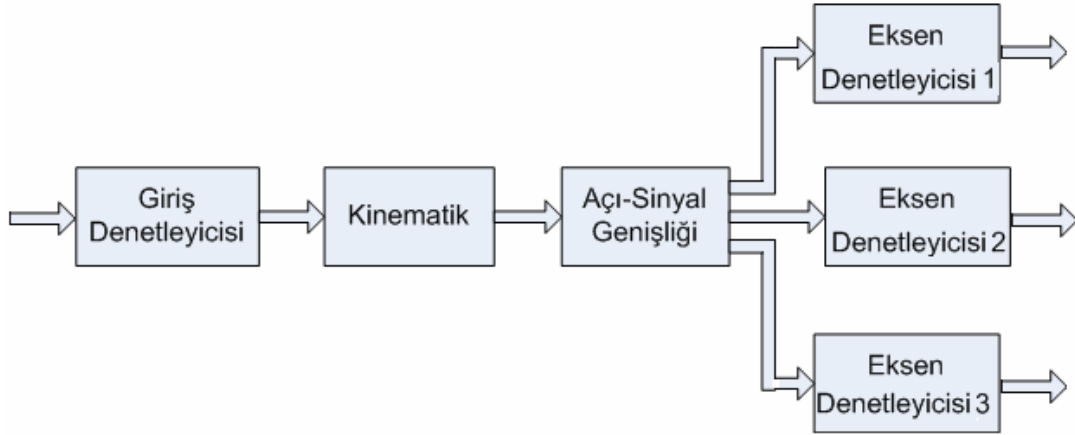
FPGA'nın çalışma frekansı 100MHz olarak seçilmiştir. Bu sayısal tasarım Verilog donanım tanımlama dili kullanılarak Xilinx ISE Foundation programı ile sentezlenmiş ve tasarımın benzetimleri gerçekleştirilmiştir.

Bütün çalışma uzayı denetleme tasarımları önemli ölçüde hesaplama işi gerektirmektedir, özellikle de ters kinematik hesaplamaları. Bu çalışmada bahsedilen problemlerin bütün matematiksel işlemleri ve denetleme algoritması FPGA içerisinde tasarlanarak uzun zaman alan işlemlerin kısa zamanda gerçekleştirilmeleri sağlanmıştır.

FPGA içerisinde yer alan sayısal tasarımın blok şeması Şekil 4.11'de görülmektedir. Sayısal tasarım Giriş Denetleyicisi, Kinematik, Açık-Sinyal Genişliği ve Eksen Denetleyicisi bloklarından oluşmaktadır.

Sayısal tasarım çalışmaya başladığında eksen denetleyicilerinin ürettiği kare dalgalar ile robot önceden tanımlanmış olan başlangıç pozisyonunda konumlandırılır. Daha sonra giriş denetleyicisi manevra kolundan gelen sinyalleri okur ve hareketin hangi yönde olacağını bulur. Örneğin hareket $x+$ yönünde olacaksa x değerini 1 artırır ve yeni x , y değerlerini çözümlenmesi için kinematik bloğuna gönderir. Kinematik bloğunun içerisinde, robotun ucunun konumlanması istenilen x , y değerleri için gerekli olan eklem açıları robotun ters kinematik denklemlerinin çözülmesi ile elde edilir. Bulunan eklem açıları robotun düz kinematik denklemlerinde yerine koyularak elde edilen sonuçların robotun çalışma alanı içinde olup olmadığı denetlenir. Eğer elde edilen sonuçlar robotun çalışma alanı içinde ise robotun eklemlerinin istenilen koordinatlarda konumlanması için gerekli açılar bulunmuş olur. Bulunan açılar, açık-sinyal genişliği bloğuna gönderilir. Açık-sinyal genişliği bloğu girişindeki açı değerlerini okur ve robotun eklemlerinde yer alan servo motorların bu açılarda konumlanması için gerekli olan darbe genişliklerini hesaplar ve bu genişlik değerlerini ilgili eksen denetleyicilerine gönderir.

Eksen denetleyicileri girişlerindeki sinyal genişlik değerlerini okurlar ve bu genişlik değerlerine sahip kare dalgaları çıkışlarında üretirler. Bu sinyaller robotun eklemlerinde yer alan motorların kontrol uçlarına gönderilir ve robotun denetimi gerçekleştirilir.



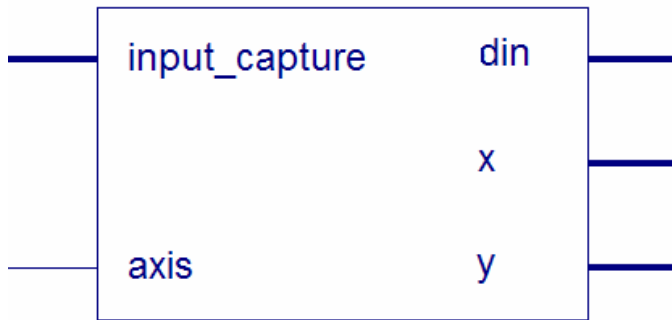
Şekil 4.11. FPGA içindeki sayısal tasarımın blok şeması

Bundan sonraki bölümlerde kinematik isimli denetleyicinin alt bölümleri hakkında sırasıyla bilgi verilecektir.

4.6.1. Giriş denetleyicisi donanımı

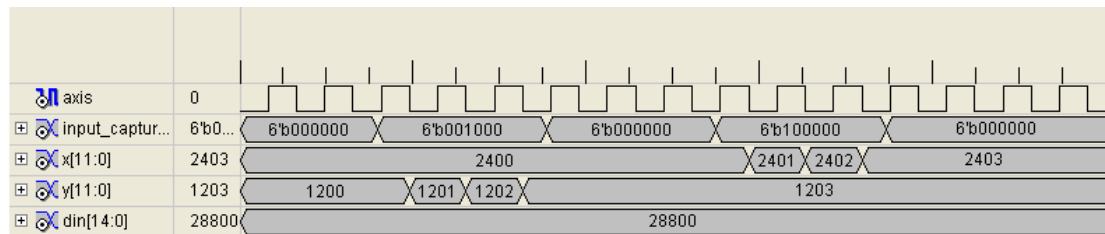
Giriş denetleyicisi Şekil 4.12'de görülmektedir, girişleri input_capture ve axis sinyalleridir, çıkışları ise din, x ve y isimli sinyallerdir. Manevra kolunun çıkış sinyalleri input_capture giriş sinyalleri tarafından alınır. Kullanıcı manevra kolundan robotu hareket ettirmek için 4 farklı komut verebilir. Manevra kolunun 1. ekseninin ileri itilmesi x+, geri çekilmesi x-, manevra kolunun ikinci ekseninin sola itilmesi y+, sağa itilmesi y- komutları olarak tanımlanmıştır. Diğer giriş sinyali olan axis eksen denetleyicisi 1'in çıkış sinyalidir. Giriş denetleyicisi, axis sinyalinin alçalan kenarında input_capture değerini okur ve manevra kolu ile verilmek istenilen komutu bulur.

Eğer komut x+ ise robotun ucunun konumlanması istenilen x değerini 1 artırır, komut x- ise x değerini 1 azaltır, komut y+ ise y değerini 1 artırır, komut y- ise y değerini 1 azaltır. Bulunan yeni x ve y değerleri aynı isme sahip çıkışlardan dışarı gönderilir. din isimli çıkış ise kinematik bloğu tarafından kullanılacak bir sabit sayıdır.



Şekil 4.12. Giriş denetleyicisi donanımı

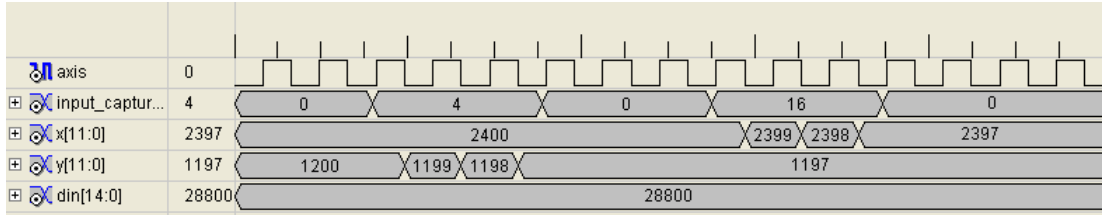
Giriş denetleyicisi donanımının Xilinx ISE benzetim programı kullanılarak benzetim sonuçları elde edilmiştir. Yapılan birinci benzetimde Manevra kolunun birinci ekseninin ileri yani x+ yönünde ve ikinci ekseninin sola yani y+ yönünde ittirildiği durumlar için benzetim gerçekleştirilmiştir. Şekil 4.13'te giriş olarak y+ uygulandığında y değerinin arttığı ve x+ uygulandığında x değerinin arttığı görülmektedir. Manevra kolu serbest bırakıldığında ise mevcut x ve y değerleri korunmaktadır.



Şekil 4.13. Giriş denetleyicisi benzetim sonucu 1

Giriş denetleyicisi ile yapılan ikinci benzetimde manevra kolunun birinci ekseninin geri yani x- yönünde ve ikinci ekseninin sağa yani y- yönünde

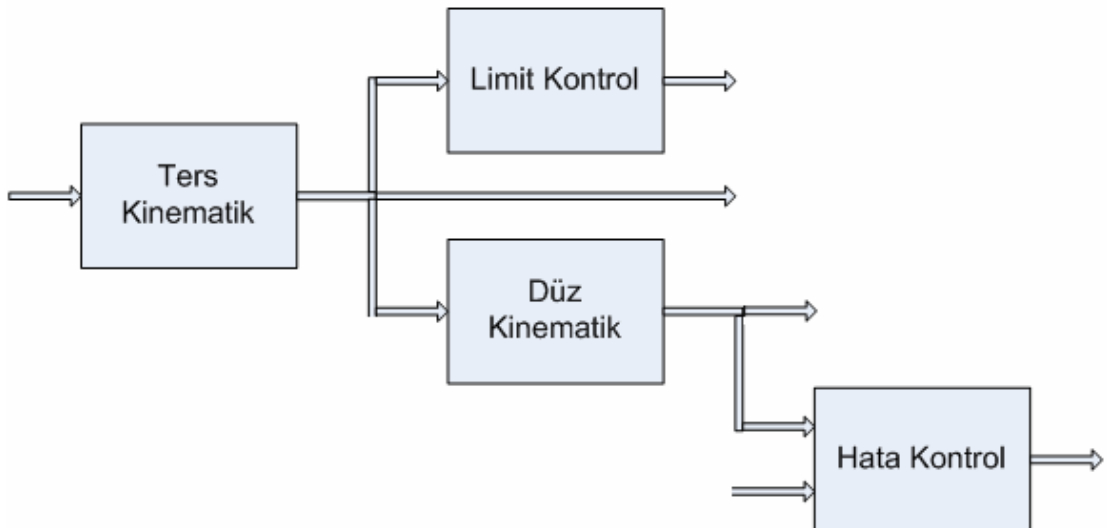
ittirildiği durumlar için benzetim gerçekleştirilmiştir. Şekil 4.14'te giriş olarak y- uygulandığında y değerinin azaldığı ve x- uygulandığında x değerinin azaldığı görülmektedir. Mavera kolu serbest bırakıldığında ise mevcut x ve y değerleri korunmaktadır.



Şekil 4.14. Giriş denetleyicisi benzetim sonucu 2

4.6.2. Kinematik donanımı

Kinematik donanımının blok şeması Şekil 4.15'te görülmektedir. Bu donanım, Ters kinematik, limit kontrol, düz kinematik ve hata kontrol bloklarından meydana gelmektedir. Kinematik donanımını meydana getiren alt denetleyiciler hakkında detaylı bilgi bu başlığın alt bölümlerinde verilmiştir.

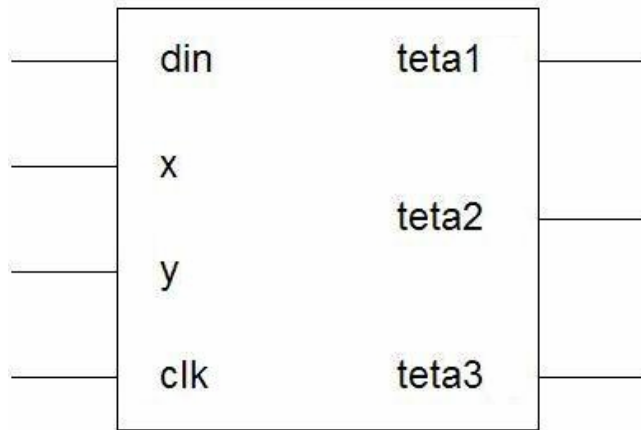


Şekil 4.15. Kinematik donanımının blok şeması

4.6.2.1. Ters kinematik donanımı

Bu donanımın girdileri saat sinyali, robot manipülatörün ucunun referans koordinat sistemine göre tanımlı olan (x, y) koordinat değerleri ve din isimli ters kinematik denklemlerin çözümünde kullanılan bir sabit değerdir. Ters kinematik donanımının çıktıları ise (x, y) koordinat değerlerine karşılık gelen robot manipülatörün eklem açıları olan $(teta1, teta2, teta3)$ açı değerleridir.

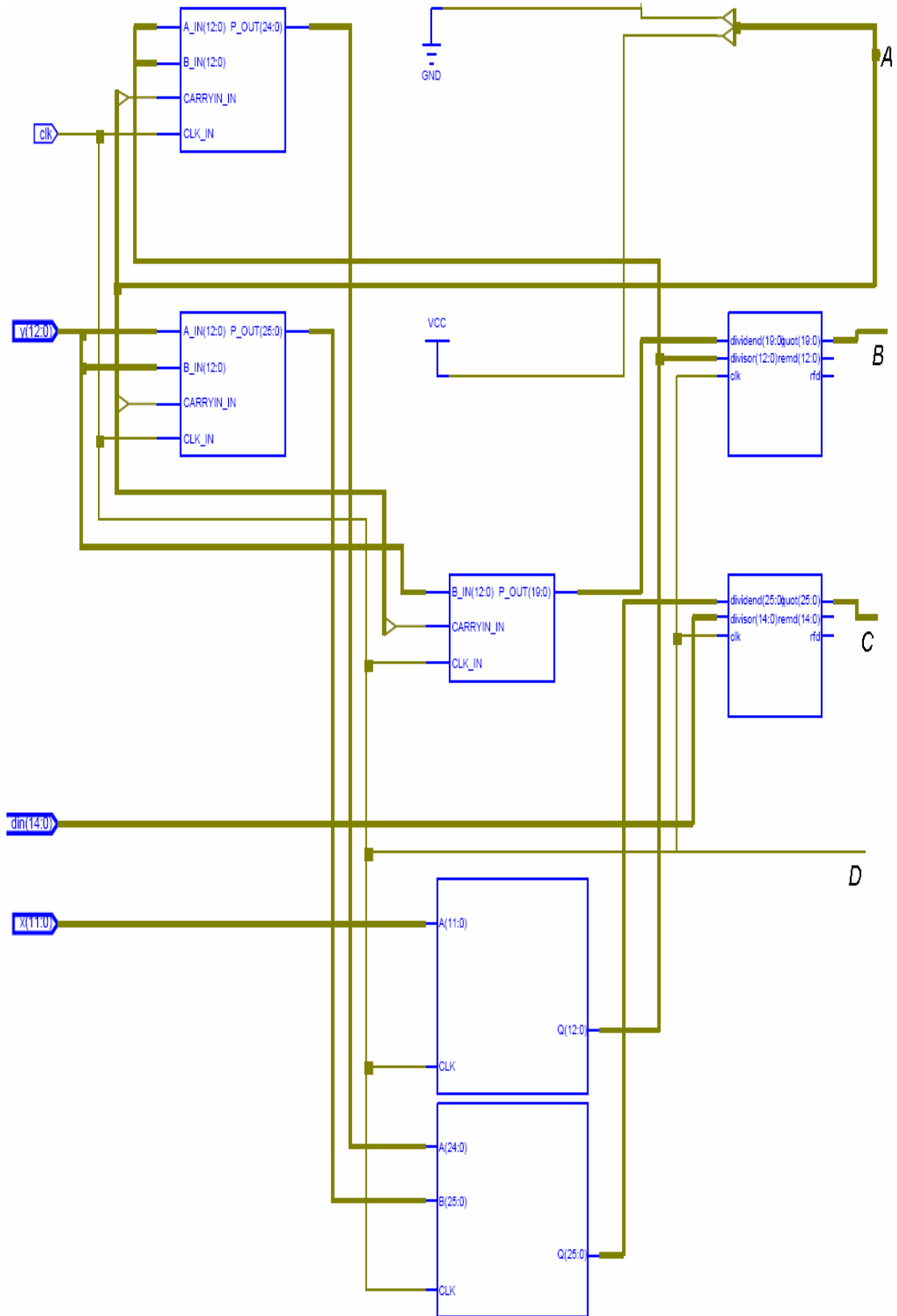
Ters kinematik sisteminin sentezlenmesi sonucu Şekil 4.16'da görülen donanım elde edilmiştir. Bu donanımda Eş. 4.2'de verilen robot manipülatörün ters kinematik denklemleri çözülmektedir.



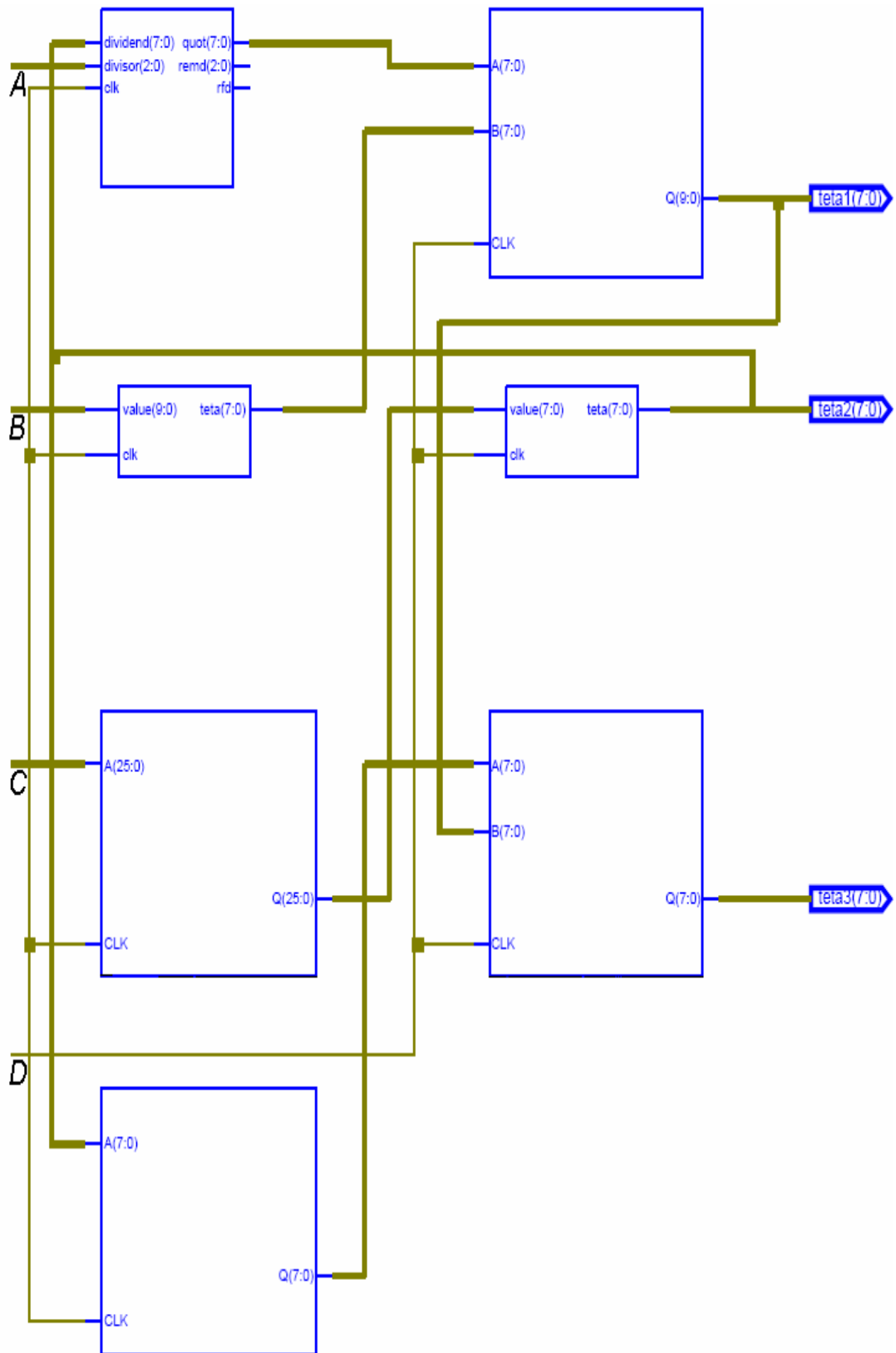
Şekil 4.16. Ters kinematik donanımı

Ters kinematik donanımı 14 alt elemandan oluşmaktadır. Bunlar 3 tane çıkarıcı, 3 tane toplayıcı, 3 tane çarpıcı, 3 tane bölücü, bir tane ters kosinüs değerlerini bulan tablo ve bir tane ters tanjant değerlerini bulan tablodur.

Ters kinematik donanımının sentezlenmesi sonucunda elde edilen detaylı yapısı Şekil 4.17'de görülmektedir.

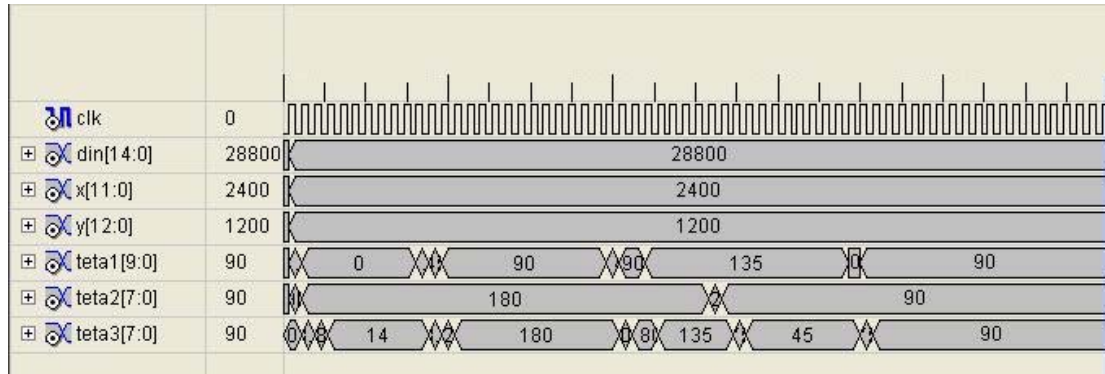


Şekil 4.17. Ters kinematik donanımının detaylı yapısı



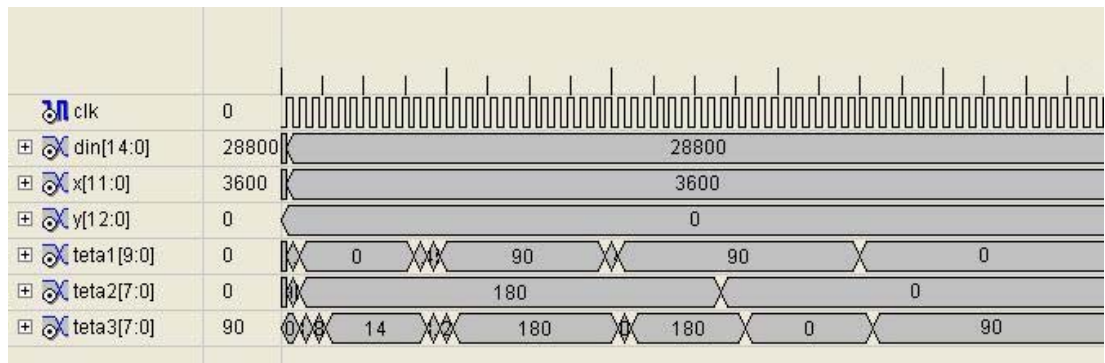
Şekil 4.17 (Devam). Ters kinematik donanımının detaylı yapısı

Ters kinematik donanımının Xilinx ISE benzetim programı kullanılarak benzetim sonuçları elde edilmiştir. Yapılan birinci benzetimde robotun ucunun referans koordinat sistemine göre tanımlı olan x değeri için 2400 ve y değeri için ise 1200 değerleri girilmiştir. Bunun sonucunda eklem açıları için sırasıyla teta1 için 90, teta2 için 90 ve teta3 için 90 değerleri bulunmuştur. Bu benzetimin sonucu Şekil 4.18’de görülmektedir.



Şekil 4.18. Ters kinematik benzetim sonucu 1

Ters kinematik donanımının ikinci benzetiminde ise robotun ucunun referans koordinat sistemine göre tanımlı olan x değeri için 3600, y değeri için 0 değerleri verilmiştir. Benzetim sonucunda eklem açıları için sırasıyla teta1 için 0, teta2 için 0 ve teta3 için 90 değerleri elde edilmiştir (Şekil 4.19).

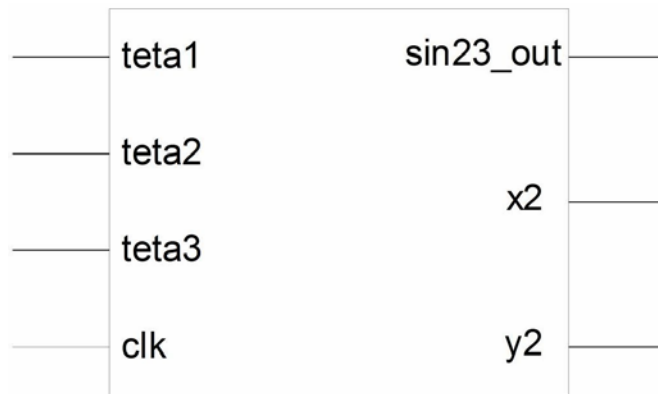


Şekil 4.19. Ters kinematik benzetim sonucu 2

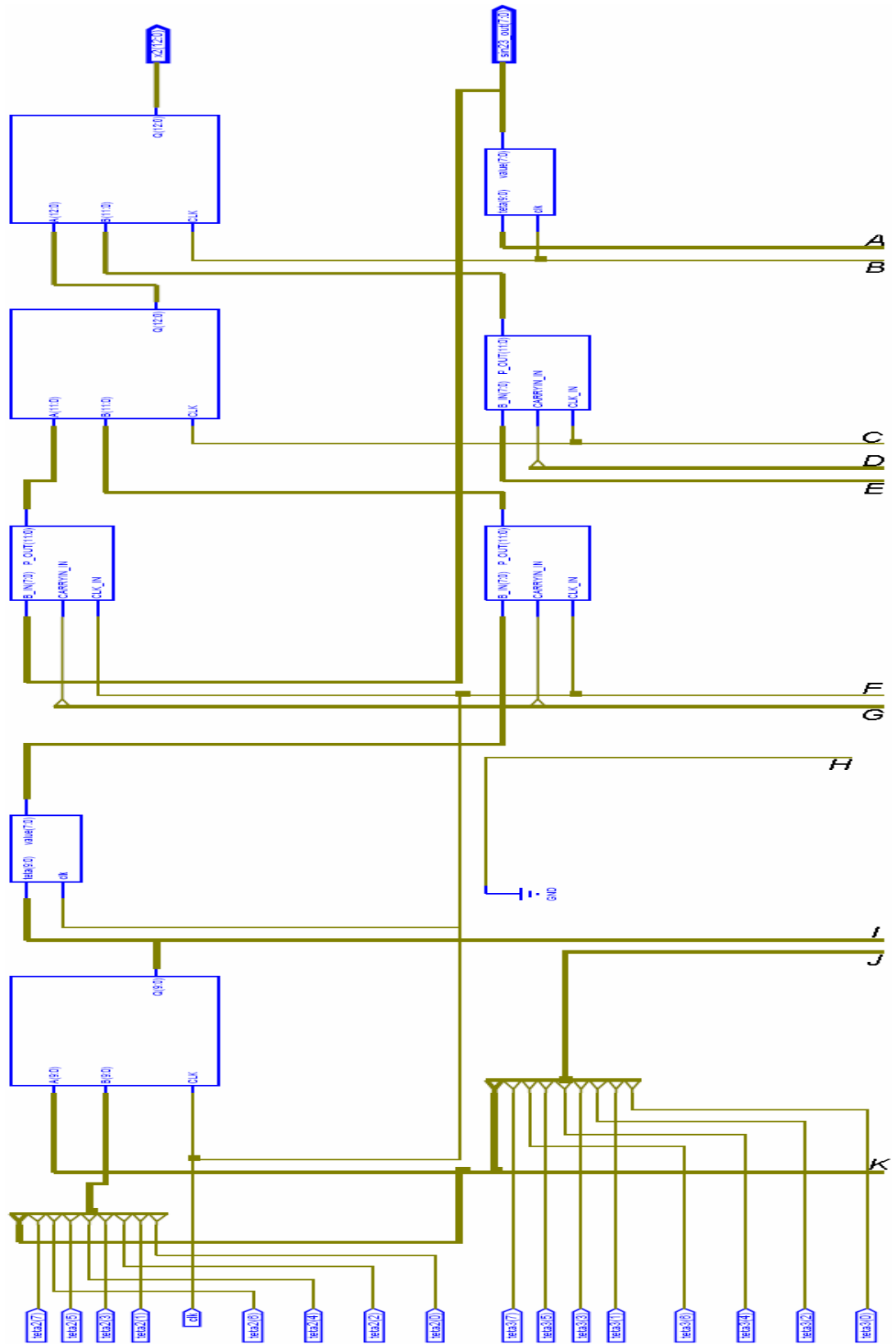
Her iki benzetim sonucundan da görüldüğü üzere ters kinematik donanımı doğru şekilde çalışmaktadır. Sisteme girdi olarak verilen (x, y) koordinat değerlerine karşılık gelen eklem açıları 50 saat darbesinden az bir süre içerisinde sistemin çıkışında görülmektedir. Benzetim sonuçlarında görülen x ve y değerlerinin belirtilenden 100 kat daha fazla olma sebebi ondalık sayılardan kurtularak sistemin daha hızlı çalışmasını sağlamaktır.

4.6.2.2. Düz kinematik donanımı

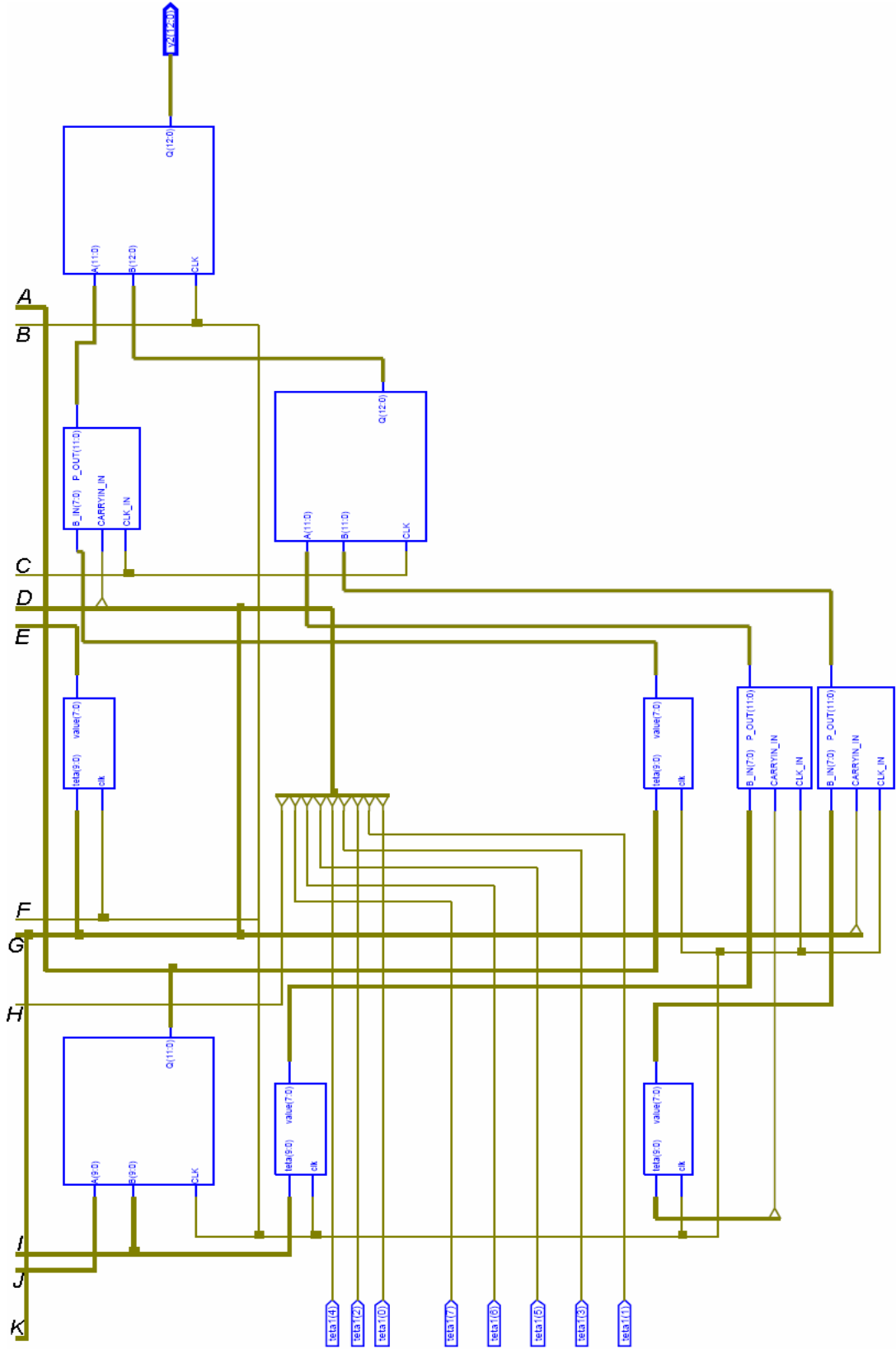
Bu donanımın girdileri saat sinyali ve ters kinematik donanımın çıktıları olan robot manipülatörün (teta1, teta2, teta3) eklem açısı değerleridir. Düz kinematik donanımının çıktıları ise (teta1, teta2, teta3) eklem açısı değerlerine karşılık gelen robot manipülatörün ucunun referans koordinat sistemine göre tanımlı olan (x2, y2) koordinat değerleri ve yine referans koordinat sistemine göre robotun ucunun yönünü bildiren sin23_out değeridir. sin23_out değeri Eş. 4.1'de verilen k değeridir. Düz kinematik tasarımının sentezlenmesi sonucunda elde edilen donanım Şekil 4.20'de görülmektedir. Düz kinematik donanımı 18 tane alt elemandan oluşmaktadır. Bunlar 2 tane çıkarıcı, 4 tane toplayıcı, 6 tane çarpıcı, 3 tane kosinüs değerlerini bulan tablo ve 3 tane sinüs değerlerini bulan tablodur. Düz kinematik donanımın sentez sonucunda elde edilen detaylı yapısı Şekil 4.21'de görülmektedir.



Şekil 4.20. Düz kinematik donanımı

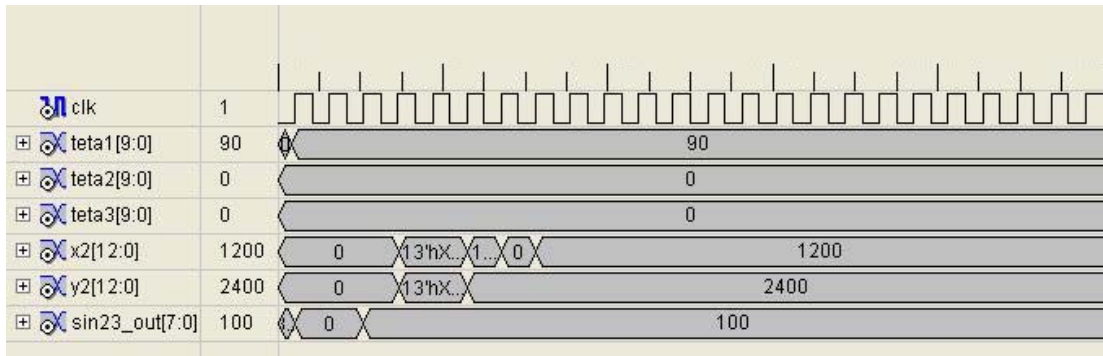


Şekil 4.21. Düz kinematik donanımının detaylı yapısı



Şekil 4.21 (Devam). Düz kinematik donanımının detaylı yapısı

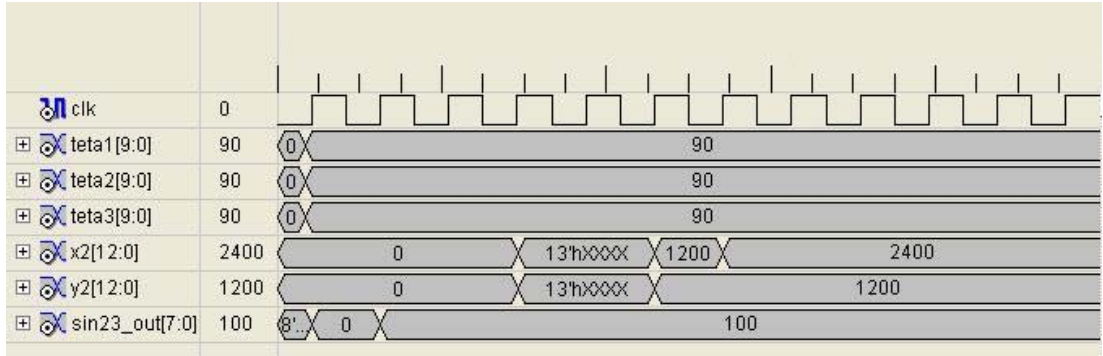
Düz kinematik donanımının Xilinx ISE programı kullanılarak benzetimleri gerçekleştirilmiştir. Yapılan birinci benzetimde robotun eklem açılarının değerleri sırasıyla teta1 için 90, teta2 için 0 ve teta3 için 0 değerleri verilmiştir. 8 saat darbesi zaman sonucunda ise robotun ucunun referans koordinat sistemine göre tanımlı x2 değeri için 1200, y2 değeri için 2400 ve robotun ucunun referans koordinat sistemine göre tanımlı sin23_out değeri için ise 100 sonuçları elde edilmiştir. Bu benzetimin sonucu Şekil 4.22'de görülmektedir.



Şekil 4.22. Düz kinematik benzetim sonucu 1

Düz kinematik donanımı ile yapılan ikinci benzetimde robotun eklem açılarının değerleri sırasıyla teta1 için 90, teta2 için 90 ve teta3 için 90 değerleri verilmiştir. 8 saat darbesi zaman sonucunda ise robotun ucunun referans koordinat sistemine göre tanımlı x2 değeri için 2400, y2 değeri için 1200 ve robotun ucunun referans koordinat sistemine göre tanımlı sin23_out değeri için ise 100 sonuçları elde edilmiştir. Bu benzetimin sonucu Şekil 4.23'te görülmektedir.

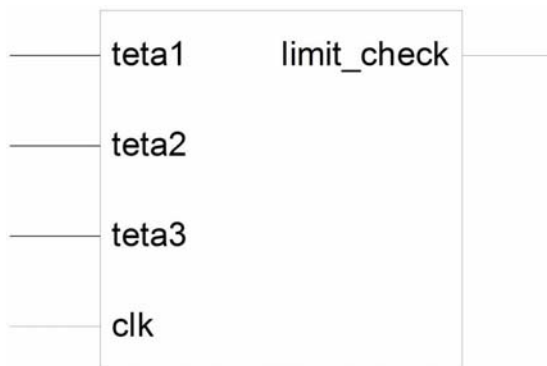
Benzetim sonuçlarında elde edilen değerler düz kinematik eşitliklerinde yerine koyulduğunda düz kinematik donanımının doğru bir şekilde çalıştığı görülmektedir.



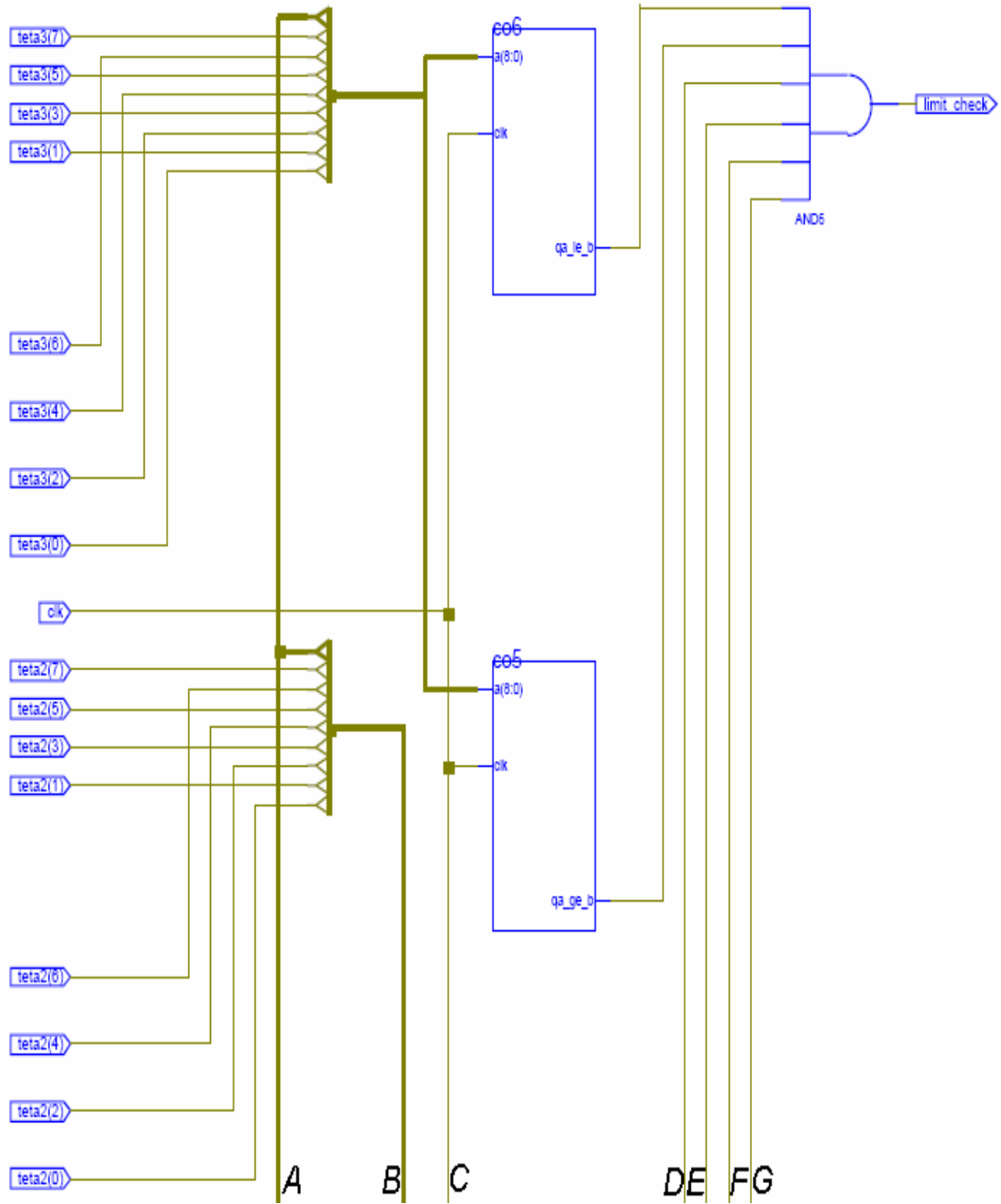
Şekil 4.23. Düz kinematik benzetim sonucu 2

4.6.2.3. Limit kontrol donanımı

Bu donanımın girdileri saat sinyali ve ters kinematik donanımının çıktıları olan robot manipülatörün (teta1, teta2, teta3) eklem açılarıdır. Limit kontrol donanımın çıkışı ise limit_check sinyalidir. Limit kontrol tasarımının sentezlenmesi sonucunda oluşan donanım Şekil 4.24'te görülmektedir. Robotun mekanik yapısından dolayı teta1 ve teta3 0 ile 180 derece arasında, teta2 ise 0 ile 150 derece arasında değerler alabilirler. Limit kontrol donanımı girdileri olan teta1, teta2 ve teta3 değerlerinin bu aralıkta olup olmadığını kontrol eder ve üç değerinde istenilen aralıktaysa çıkış sinyali olan limit_check 1 değerini alır.

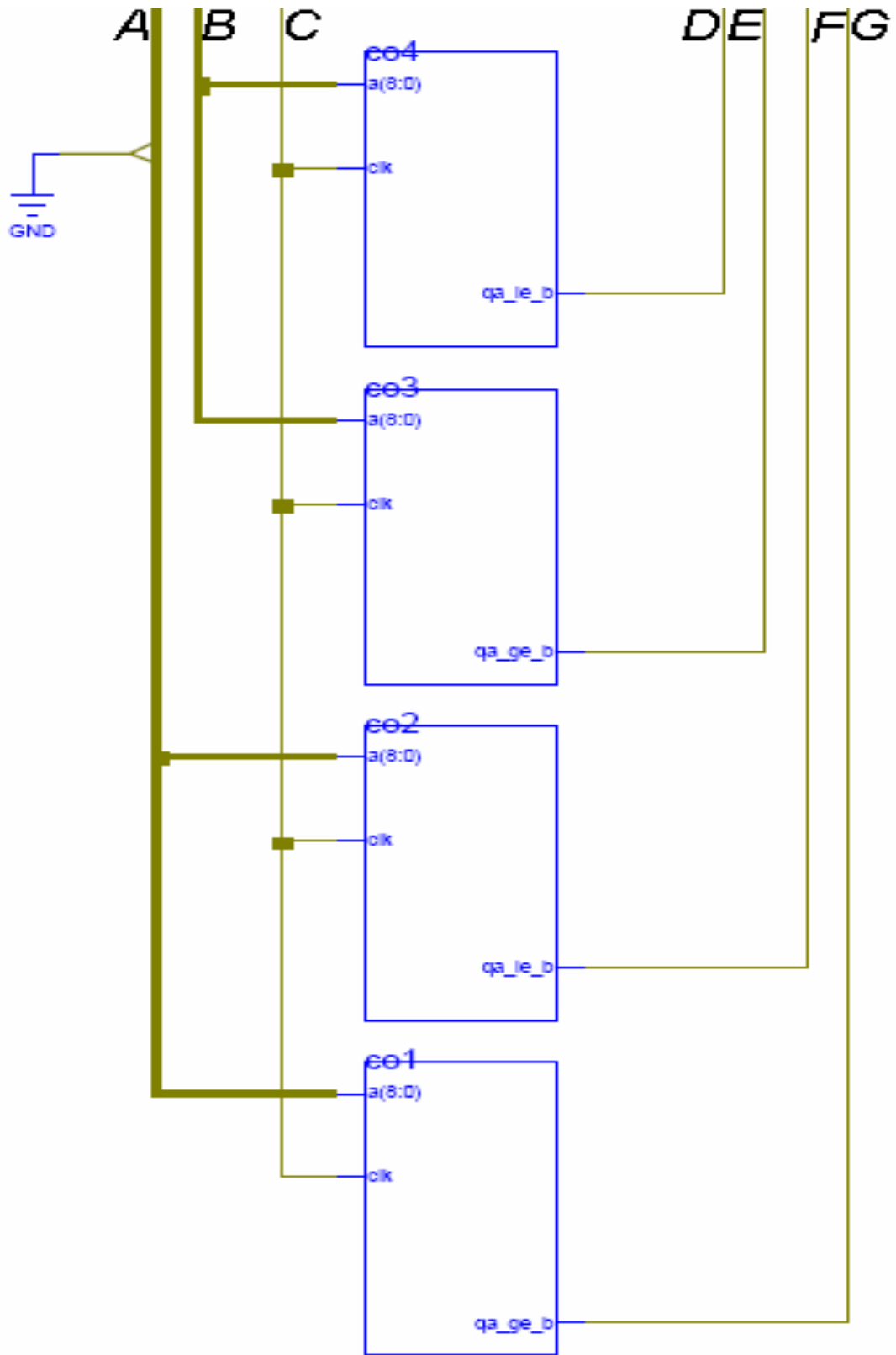


Şekil 4.24. Limit kontrol donanımı



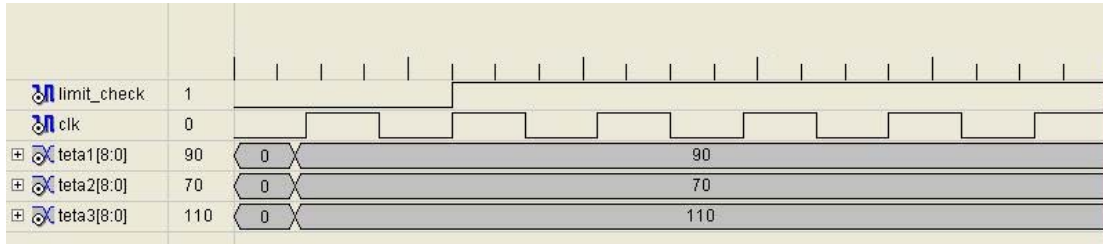
Şekil 4.25. Limit kontrol donanımının detaylı yapısı

Limit kontrol donanımı 10 tane alt elemandan oluşmaktadır. Bunlar 6 tane karşılaştırıcı (comparator) ve 4 tane VE (and) kapısıdır. Limit kontrol donanımının sentezinin detaylı sonucu Şekil 4.25'te görülmektedir.



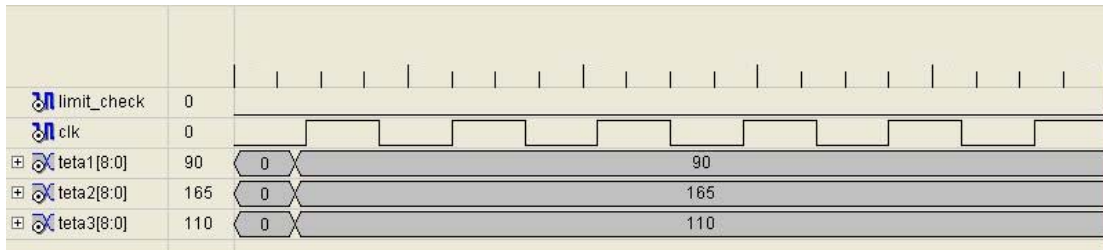
Şekil 4.25 (Devam). Limit kontrol donanımının detaylı yapısı

Limit kontrol donanımının Xilinx ISE programı kullanılarak benzetimi gerçekleştirilmiştir. Yapılan birinci benzetimde robotun eklem açıları olan teta1 için 90, teta2 için 70 ve teta3 için 110 değerleri verilmiştir. Bu üç değerde istenilen sınırlar içinde olduğu için limit_check 1 değerini almıştır. Bu benzetimin çıktısı Şekil 4.26'da görülmektedir.



Şekil 4.26. Limit kontrol benzetim sonucu 1

Limit kontrol donanımı ile yapılan ikinci benzetimde ise robotun eklem açıları olan teta1 için 90, teta2 için 165 ve teta3 için 110 değerleri verilmiştir. Bunun sonucunda girilen teta2 değeri 150'den büyük olduğu için error_check sinyali 0 değerini almıştır. Bu benzetimin sonucu Şekil 4.27'de görülmektedir.

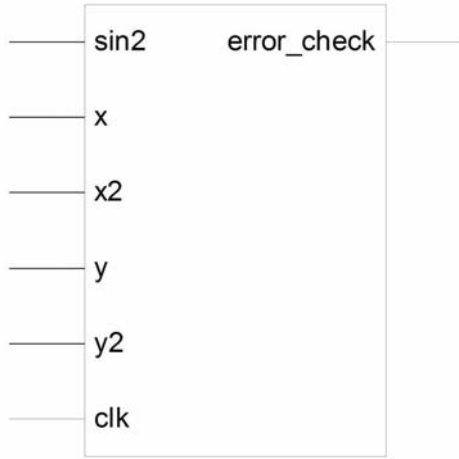


Şekil 4.27. Limit kontrol benzetim sonucu 2

Her iki benzetim sonucunda da görüldüğü üzere limit kontrol donanımı doğru şekilde çalışmaktadır. Girilen teta değerlerinin istenilen aralıkta olup olmadığını 2. saat darbesi ile birlikte limit kontrol donanımının çıkışından okunabilir.

4.6.2.4. Hata kontrol donanımı

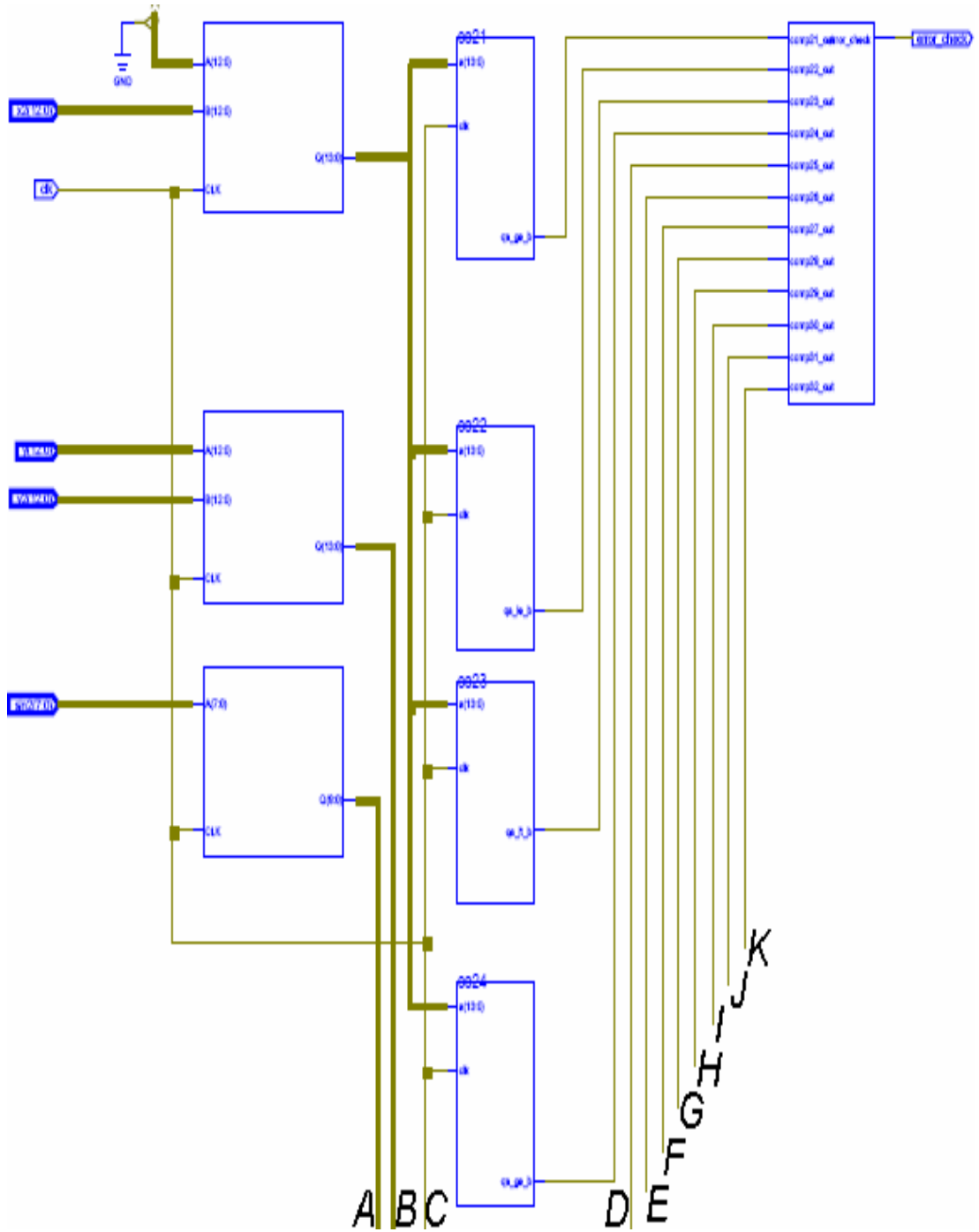
Bu donanımın girdileri saat sinyali, robot manipülatörün ucunun referans koordinat sistemine göre tanımlı (x, y) , $(x2, y2)$ koordinat değerleri ve $\sin2$ yön değeridir. Hata kontrol donanımının çıktısı ise `error_check` (hata kontrol) sinyalidir. Hata kontrol tasarımının sentezlenmesi sonucunda elde edilen donanım Şekil 4.28'de görülmektedir.



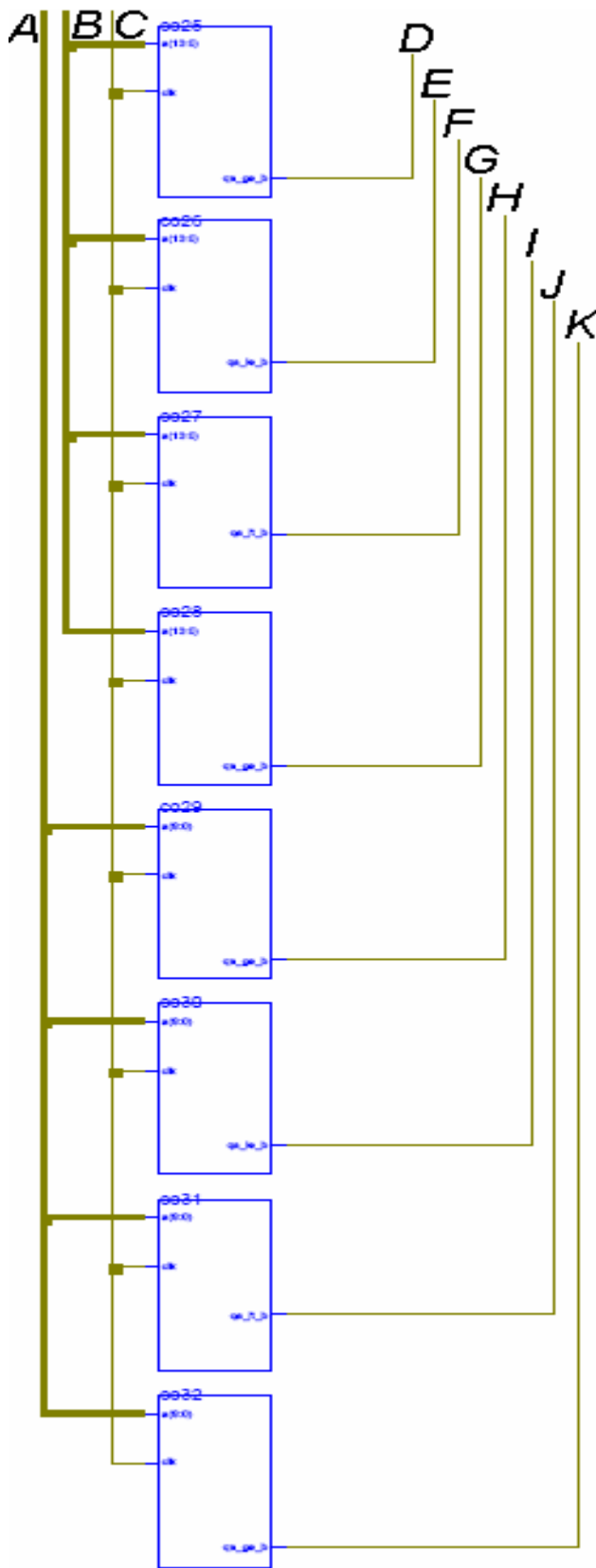
Şekil 4.28. Hata kontrol donanımı

Hata kontrol donanımı robotun ucunun referans koordinat sistemine göre tanımlı (x, y) koordinat değerlerinden düz kinematik donanımının çıktısı olan $(x2, y2)$ değerlerini çıkartır. Robotun ucunun yönü bu çalışmada sabit kalacak şekilde işlem yapılmıştır yani referans koordinat sistemine göre yön değeri önceden tanımlanmıştır. Hata kontrol donanımı bu yön değerinden düz kinematik donanımının çıktısı olan `sin23_out` değerini çıkartır. Sonuç olarak $(x-x2)$, $(y-y2)$, $(\sin2-\sin23_out)$ işlemlerini yapar, eğer farklar istenilen hata değeri içindeyse çıkış sinyali olan `error_check` değeri 1 olur. Hata kontrol donanımı 25 tane alt elemandan oluşmaktadır.

Bunlar 12 tane karşılaştırıcı, 3 tane çıkarıcı, 7 tane VE kapısı ve 3 tane VEYA (or) kapısıdır. Sentez sonucu elde edilen hata kontrol donanımının detaylı yapısı Şekil 4.29'da görülmektedir.

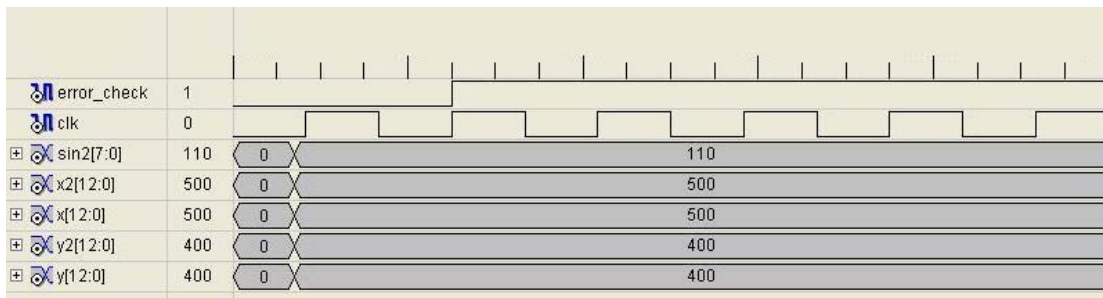


Şekil 4.29. Hata kontrol donanımının detaylı yapısı



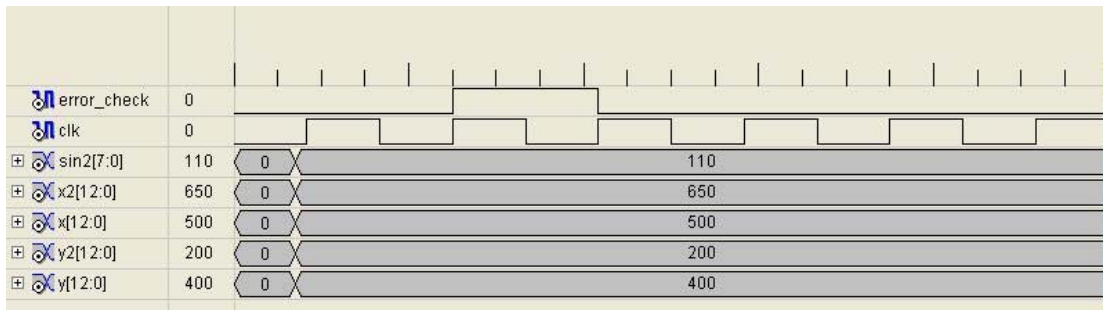
Şekil 4.29 (Devam). Hata kontrol donanımının detaylı yapısı

Hata kontrol donanımının Xilinx ISE programı kullanılarak benzetimi gerçekleştirilmiştir. Yapılan birinci benzetimde sin2 için 110, x2 için 500 ve y2 için 400, x için 500 ve y için 400 değerleri girilmiştir. Hata sınırı 100 değeri olarak seçilmiştir. Bu değer x ve y koordinatları için 1 birimlik hata anlamına gelmektedir. Benzetim sonucunda bütün fark değerleri 100'den küçük olduğu için error_check sinyali 1 değerini almıştır. Bu benzetimin sonucu Şekil 4.30'da görülmektedir.



Şekil 4.30. Hata kontrol benzetim sonucu 1

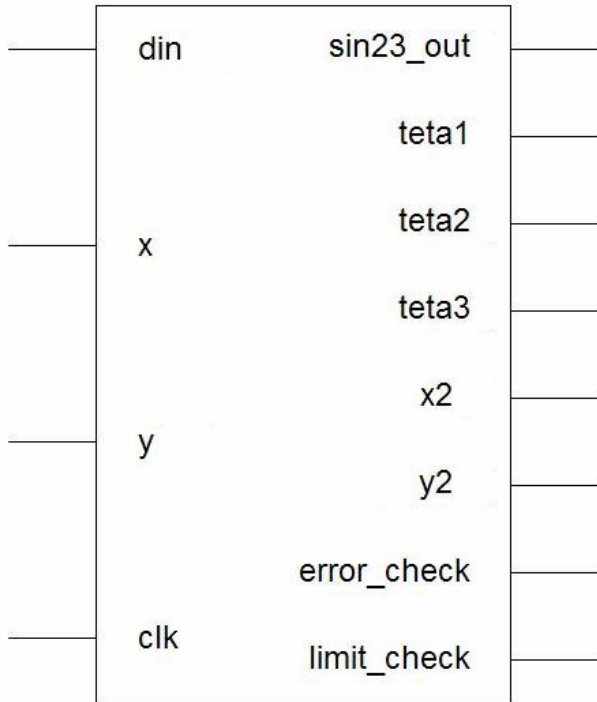
Hata kontrol donanımı ile yapılan ikinci benzetimde sin2 için 110, x2 için 650, y2 için 200, x için 500 ve y için 400 değerleri girilmiştir. (y-y2) farkı 100'den büyük olduğu için error_check sinyali 0 değerini almıştır. Bu hata kontrol donanımının error_check çıkışından 3 saat darbesi sonunda okunabilir. Bu benzetimin sonucu Şekil 4.31'de görülmektedir.



Şekil 4.31. Hata kontrol benzetim sonucu 2

Her iki benzetim sonucundan da görüldüğü üzere hata kontrol donanımı doğru bir şekilde çalışmaktadır. Girilen değerlerin sonucunda 3 saat darbesi kadar bir zaman geçtikten sonra hata donanımının çıkışı okunarak bu değerlerin istenilen hata sınırı içinde olup olmadığı görülebilir.

Şekil 4.13'te blok şeması verilen kinematik denetleyicisini meydana getiren alt denetleyiciler hakkında detaylı bilgi geçtiğimiz bölümlerde verilmişti. Kinematik denetleyicisinin sentezlenmesi sonucunda Şekil 4.32'de görülen donanım elde edilmiştir.



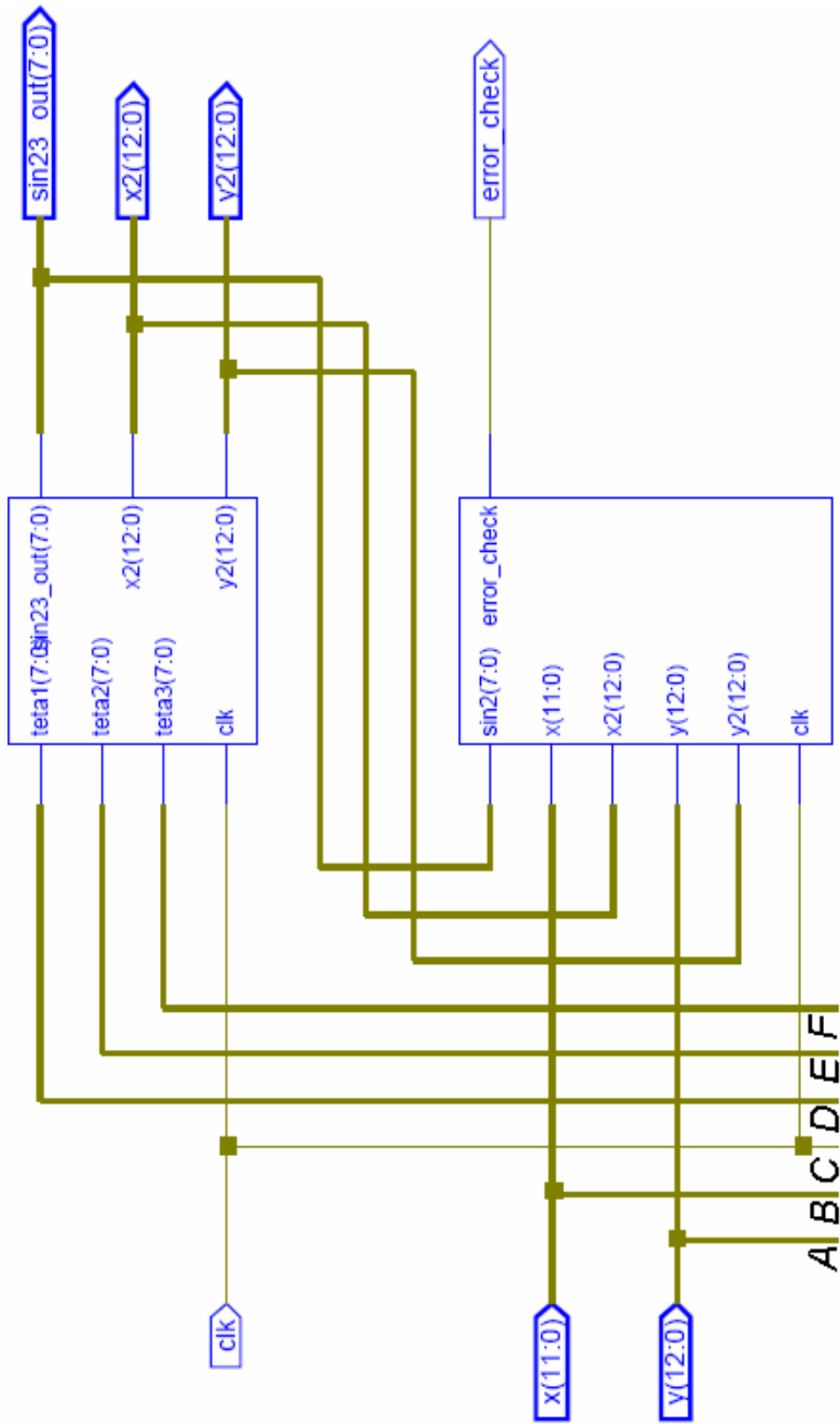
Şekil 4.32. Kinematik donanımı

Sistem dört ana bloktan meydana gelmektedir. Bunlar ters kinematik donanımı, düz kinematik donanımı, limit kontrol donanımı ve hata kontrol donanımdır. Sistemin girdileri saat sinyali, robot manipülatörün ucunun referans koordinat sistemine göre tanımlı olan (x, y) koordinat değerleri ve din isimli sabit sayıdır.

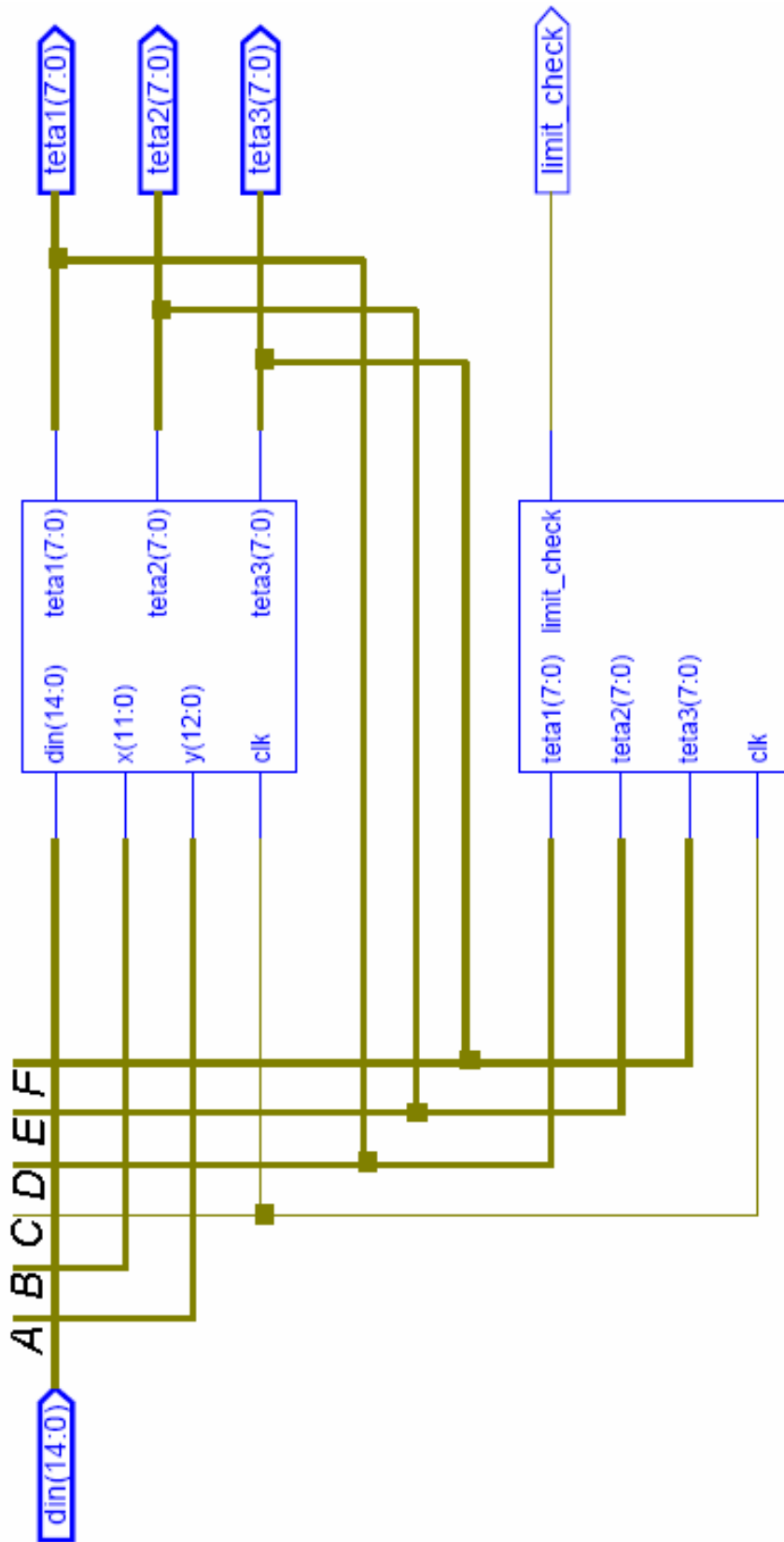
Sistemin çıktıları ise `error_check` (hata kontrolü), `limit_check` (sınır kontrolü), $(\theta_1, \theta_2, \theta_3)$ eklem açıları, (x_2, y_2) koordinat değerleri ve `sin23_out` yön değeridir.

Kinematik donanımı kendisine girdi olarak verilen robotun referans koordinat sistemine göre tanımlı olan (x,y) değerlerini ve robotun ucunun referans koordinat sistemine göre önceden tanımlanmış olan yön bilgisini alarak aşağıdaki donanım yapılarından geçirir.

Sonuçta elde edilen $\theta_1, \theta_2, \theta_3$ değerleri eğer `error_check` değeri 1 ise ve `limit_control` değeri 1 ise robotun verilen koordinat ve yön bilgilerine karşılık gelen doğru eklem açıları bulunmuş olur. Eğer `error_check` ya da `limit_check` sinyallerinden herhangi birinin değeri 1 değilse elde edilen açı değerleri yanlıştır. Kinematik donanımının sentezlenmesi sonucu elde edilen detaylı yapısı Şekil 4.33'te görülmektedir.

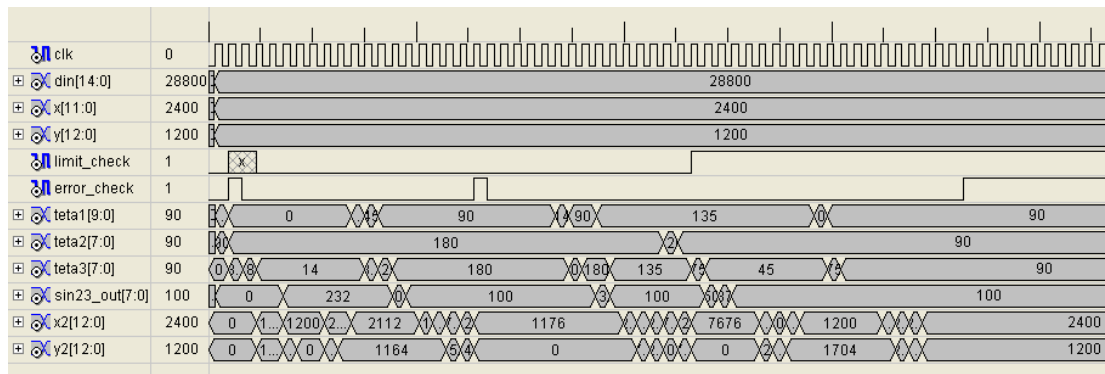


Şekil 4.33. Kinematik donanımının detaylı yapısı



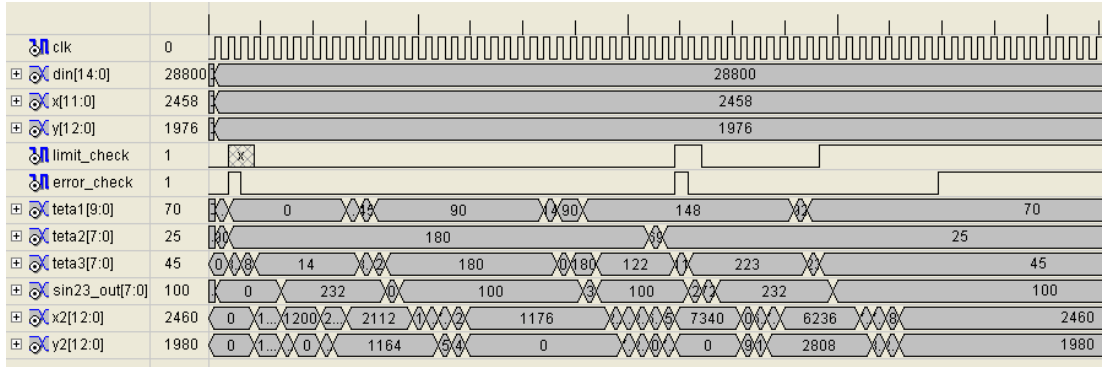
Şekil 4.33 (Devam). Kinematik donanımının detaylı yapısı

Kinematik donanımının Xilinx ISE programı kullanılarak benzetimi gerçekleştirilmiştir. Yapılan birinci benzetimde x için 2400, y için 1200, din için 28800 değerleri girilmiştir. 60 saat darbesinden az bir zaman sonucunda Kinematik donanımının çıkışında beklenen çıkış değerleri elde edilmiştir. Bunlar sırasıyla, limit_check için 1, error_check için 1, teta1 için 90, teta2 için 90, teta3 için 90, sin23_out için 100, x2 için 2400 ve y2 için 1200 değerleridir. Sonuçta elde edilen error_check ve limit_check sinyallerinin her ikisinin de değeri 1 olduğu için elde edilen açı değerleri robotun ucunu istenilen pozisyonda ve yönde konumlandırmak için kullanılabilir. Elde edilen benzetim sonucu Şekil 4.34'te görülmektedir.



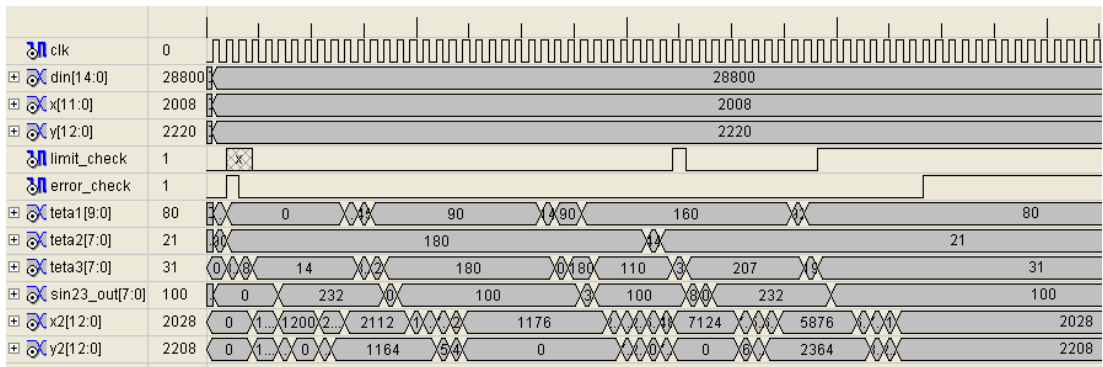
Şekil 4.34. Kinematik donanımının benzetim sonucu 1

Kinematik donanımı ile yapılan ikinci benzetimde x için 2458, y için 1976, din için 28800 değerleri girilmiştir. 60 saat darbesinden az bir zaman sonucunda sırasıyla verilen çıkış değerleri elde edilmiştir. limit_check için 1, error_check için 1, teta1 için 70, teta2 için 25, teta3 için 45, sin23_out için 100, x2 için 2460 ve y2 için 1980 değerleridir. Sonuçta elde edilen error_check ve limit_check sinyallerinin her ikisinin de değeri 1 olduğu için elde edilen açı değerleri robotun ucunu istenilen pozisyonda ve yönde konumlandırmak için kullanılabilir. Elde edilen benzetim sonucu Şekil 4.35'te görülmektedir.



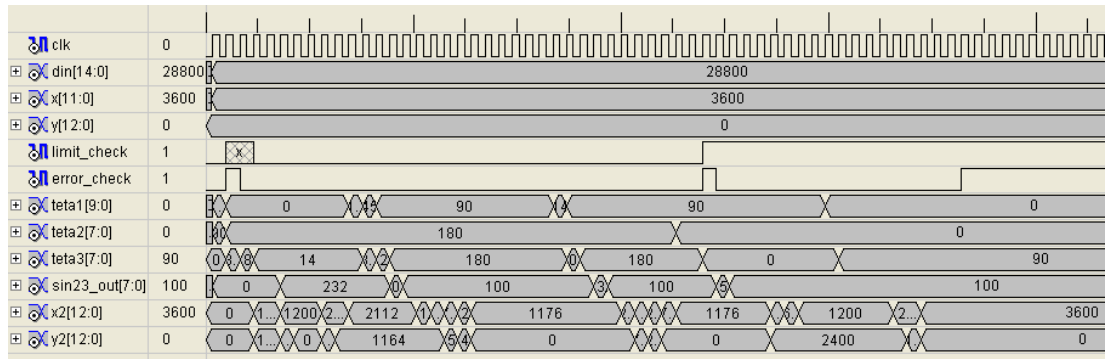
Şekil 4.35. Kinematik donanımının benzetim sonucu 2

Kinematik donanımı ile yapılan üçüncü benzetimde x için 2008, y için 2220, din için 28800 değerleri girilmiştir. 60 saat darbesinden az bir zaman sonucunda sırayla verilen çıkış değerleri elde edilmiştir. limit_check için 1, error_check için 1, teta1 için 80, teta2 için 21, teta3 için 31, sin23_out için 100, x2 için 2028 ve y2 için 2208 değerleridir. Sonuçta elde edilen error_check ve limit_check sinyallerinin her ikisinin de değeri 1 olduğu için elde edilen açı değerleri robotun ucunu istenilen pozisyonda ve yönde konumlandırmak için kullanılabilir. Elde edilen benzetim sonucu Şekil 4.36'da görülmektedir.



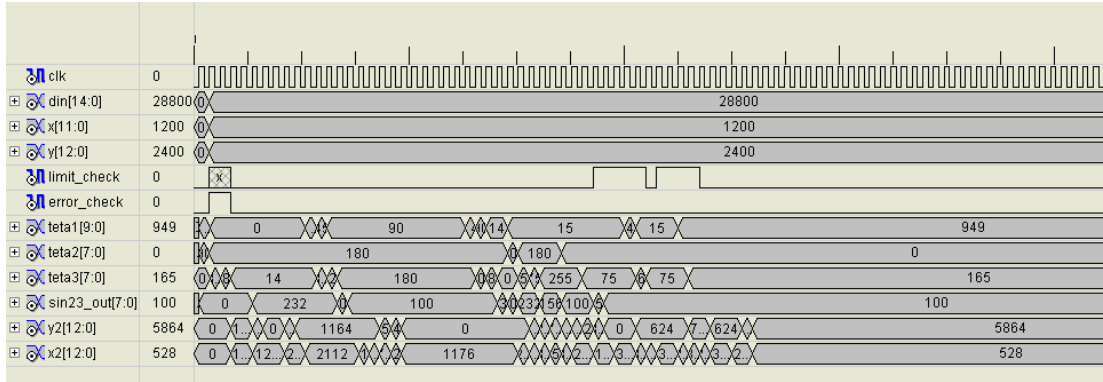
Şekil 4.36. Kinematik donanımının benzetim sonucu 3

Kinematik donanımı ile yapılan dördüncü benzetimde x için 3600, y için 0, din için 28800 değerleri girilmiştir. 60 saat darbesinden az bir zaman sonucunda sırayla verilen çıkış değerleri elde edilmiştir. limit_check için 1, error_check için 1, teta1 için 0, teta2 için 0, teta3 için 90, sin23_out için 100, x2 için 3600 ve y2 için 0 değerleridir. Sonuçta elde edilen error_check ve limit_check sinyallerinin her ikisinin de değeri 1 olduğu için elde edilen açı değerleri robotun ucunu istenilen pozisyonda ve yönde konumlandırmak için kullanılabilir. Elde edilen benzetim sonucu Şekil 4.37'de görülmektedir.



Şekil 4.37. Kinematik donanımının benzetim sonucu 4

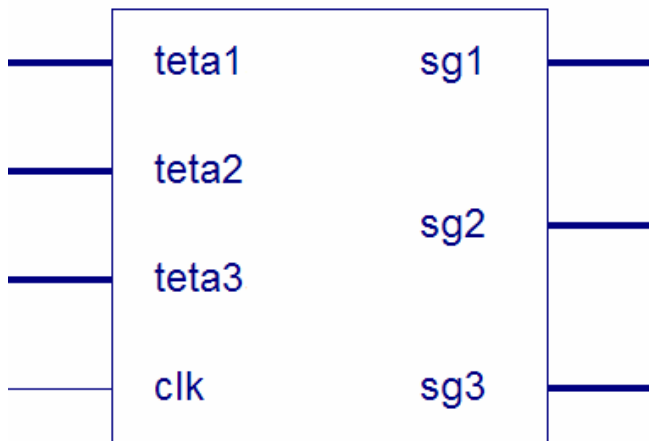
Kinematik donanımı ile yapılan beşinci benzetimde x için 1200, y için 2400, din için 28800 değerleri girilmiştir. 60 saat darbesinden az bir zaman sonucunda sırayla verilen çıkış değerleri elde edilmiştir. limit_check için 1, error_check için 1, teta1 için 0, teta2 için 0, teta3 için 90, sin23_out için 100, x2 için 3600 ve y2 için 0 değerleridir. Sonuçta elde edilen error_check ve limit_check sinyallerinin her ikisinde değeri 1 olduğu için elde edilen açı değerleri robotun ucunu istenilen pozisyonda ve yönde konumlandırmak için kullanılamaz. Bu benzetimde _check sinyalinin 0 olmasının sebebi Eş.4.2'de yer alan θ_1 eşitliğinin paydasında ki $x-l_3a$ değerinin 0 olmasıdır. Bu nokta tekil nokta (singular point) olarak adlandırılır. Elde edilen benzetim sonucu Şekil 4.38'te görülmektedir.



Şekil 4.38. Kinematik donanınin benzetim sonucu 5

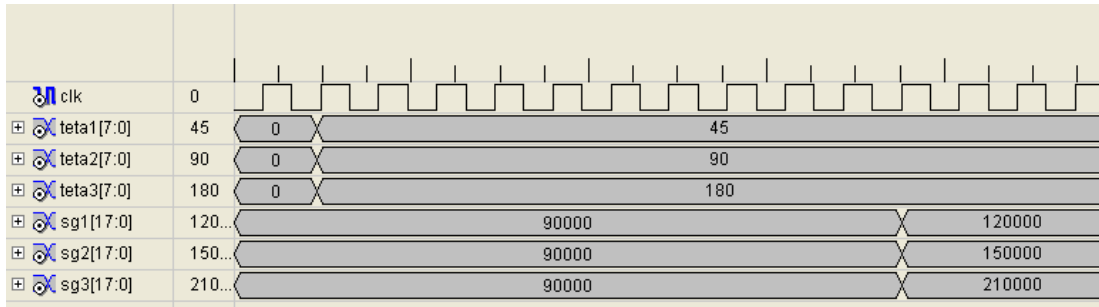
4.6.3. Açık-sinyal genişliği donanımı

Açık-sinyal genişliği donanımı Şekil 4.39’da verilmiştir. Bu donanımın girdileri, teta1, teta2, teta3 ve clk sinyalleridir. Çıktıları ise sg1, sg2, sg3 sinyalleridir. teta1, teta2 ve teta3 değerleri robotun istenilen x ve y koordinatlarında konumlanması için gerekli olan eksen açılarıdır. clk sinyali ise sistemin saat sinyalidir. Açık-sinyal genişliği donanımı girişindeki açı değerlerine karşılık gelen sinyal genişlik değerlerini Eş. 3.20’yi kullanarak hesaplar ve bulduğu sinyal genişlik değerlerini sırasıyla sg1, sg2, sg3 çıkışlarına verir.



Şekil 4.39. Açık-sinyal genişliği donanımı

Açı-sinyal genişliği denetleyicisi donanımının Xilinx ISE benzetim programı kullanılarak benzetim sonuçları elde edilmiştir. Yapılan benzetimde teta1 için 45, teta2 için 90, teta3 için 180 değerleri verilmiştir. Bunlara karşılık gelen sinyal genişlikleri Şekil 4.40'ta verilmektedir. Teta2 yani 90 değeri için hesaplanmış olan sg2 sonucuna bakılacak olursa 150000 olduğu görülür. 90 değeri Eş.3.20'de yerine koyulduğunda sinyal genişliği $1,5 \cdot 10^{-3}$ olarak hesaplanır. Benzetim sonucunda 150000 çıkmasının sebebi ise $1,5 \cdot 10^{-3}$ değerinin sistemin saatinin periyodu olan 10^{-8} bölünmesidir. Bu işlem diğer açı ve sinyal genişlikleri içinde geçerlidir.

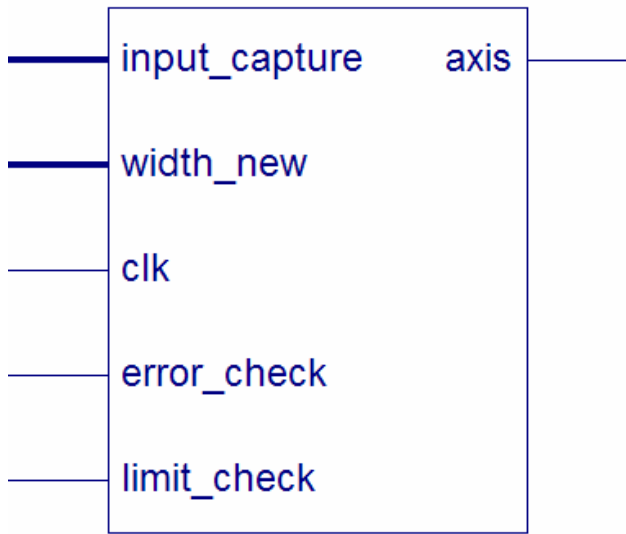


Şekil 4.40. Açı-sinyal genişliği donanımı benzetimi

4.6.4. Eksen denetleyicisi donanımı

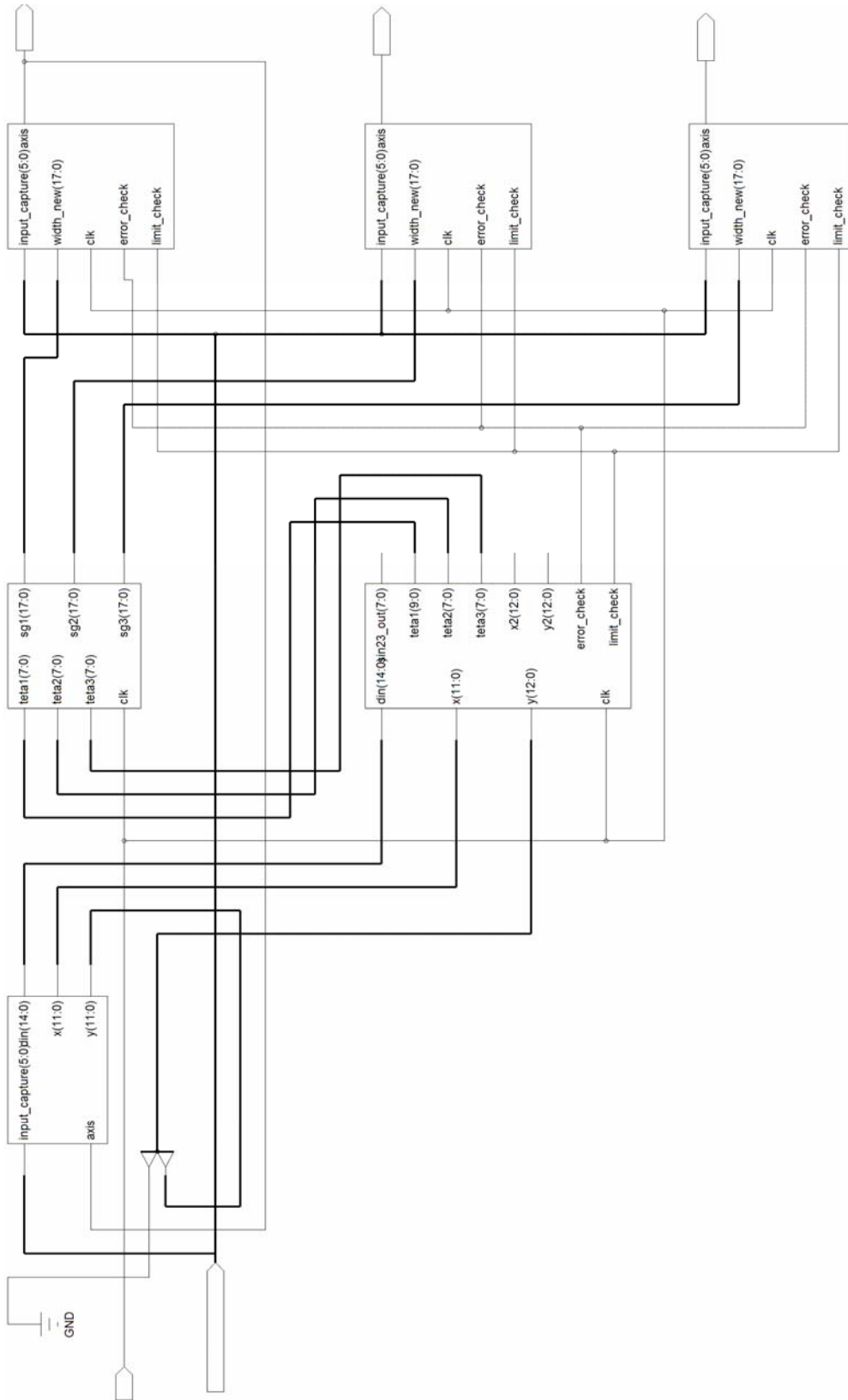
Eksen denetleyicisi donanımı Şekil 4.41'de görülmektedir. Bu donanımın girdileri input_capture, width_new, clk, error_check, limit_check sinyalleridir. Çıktısı ise axis sinyalidir. input_capture sinyali manevra kolundan gelen giriş sinyalidir, width_new sinyali açı-sinyal genişliği donanımının çıktısı olan sinyal genişliği sinyalidir, clk sinyali sistemin saat sinyalidir. error_check ve limit_check sinyalleri kinematik donanımının çıktısı olan aynı isme sahip sinyallerdir. Şekil 4.11'de sayısal tasarımın blok şemasından görüldüğü üzere 3 adet eksen denetleyicisi bulunmaktadır ancak bunların içyapıları aynı olduğu için sadece birisi anlatılacaktır. Eksen denetleyicisi girişindeki sinyalleri yorumlayarak çıkışında robotun ilgili eklemine bulunan servo

motoru denetlemek için gerekli olan Şekil 3.5'te verilen kare dalga sinyali üretir. Önceden de belirtildiği gibi kare dalganın periyodu 20ms'dir. Sinyalin genişliği ise 0.9ms ile 2.1 ms arasında değişmektedir. Eksen denetleyicisi 20ms'lik periyodun başında axis sinyalini 1 yapar ve sinyal genişliği kadar zaman süresince bu sinyalin değeri 1 olarak kalır, bu zaman dolduktan sonra axis'in değerini 0 yapar. Eksen denetleyicisi axis sinyalini 0 yaptığı zaman girişindeki sinyalleri okur. Eğer input_capture sinyalinin değeri 0 değilse yani manevra kolundan bir komut geliyorsa, error_check ve limit_check sinyallerinin değerleri 1 ise yani kinematik donanımının hesapladığı açılış değerleri istenilen hata sınırları ve mekanik sınırlar içinde ise girişindeki sinyal genişliği değerini okur ve bir sonraki periyotta bu değeri kullanarak yeni kare dalgayı üretir.



Şekil 4.41. Eksen denetleyicisi donanımı

FPGA içerisinde yer alan sayısal tasarımın blok şeması Şekil 4.11'de verilmişti. Bu bölümde ise geliştirilen sayısal tasarımın sentezlenmesi sonucunda elde edilen donanım Şekil 4.42'de verilmektedir. Bu çalışmada geliştirilmiş olan sayısal tasarımın kodları EK 4-12' de verilmiştir.



Şekil 4.42. Robot denetleyicisi donanımının detaylı yapısı

5. SONUÇ VE ÖNERİLER

FPGA'lar kart boyutunu düşürmek ve uygulamaları hızlandırmak için kullanılacak genel amaçlı hesaplama araçlarıdır. FPGA'ların başlıca avantajları: yeni mimarilerin uygulanmasına, hızlı denetleme algoritmalarının geliştirilmesine, dinamik donanım yapılandırmasına, uzaktan donanım tanımlanmasına imkân vermeleridir. FPGA'ların yoğunlukları, performansları, kullanım kolaylıkları artmaya devam etmekte ve fiyatları düşmekte olduğundan robotik uygulamalarında tercih sebebi olmaya devam edeceklerdir.

Robot manipülatörün eksen özgül denetlenmesinde geliştirilen ve uygulanan denetim algoritmasının basit olduğu görülmektedir [38]. Ancak bu tip denetim yönteminde robotun eksenleri tek tek kontrol edildiği için robotun ucunun istenilen konum ve yönde konumlanması uğraştırıcı bir iş olmaktadır.

Robot manipülatörün kartezyen özgül denetiminde geliştirilen denetim algoritmasının eksen özgül denetim algoritmasına göre daha karmaşık olduğu görülmektedir. Ancak bu denetim yönteminde ise robotun ucunun konum ve yönünün istenilen şekilde ayarlanabilmesi ise eksen özgül denetime göre çok daha kolaydır. Bu yöntemin esas amacı sistemi mümkün olduğunca basit tutmak ve güvenilir bir denetleyici tasarlamaktır. Denetleme algoritması FPGA içerisinde uygulandığı için matematiksel denklemlerin çözümü mikro saniyeler mertebesinde tamamlanmaktadır. Sistemin bu kadar hızlı olmasında FPGA içerisinde bulunan hazır donanımsal yapıların kullanılmasının büyük etkisi vardır.

Günümüzde endüstride bu çalışmada kullanılan eksen özgül ve kartezyen özgül denetim uygulamaları kullanılmaktadır. Her iki denetim algoritmasının tercih edildiği uygulama alanları farklıdır yani bu iki algoritma da önem taşımaktadır.

Bu alıřmada geliřtirilen kartezyen zgl denetim algoritması daha fazla ekleme sahip olan bir robotta uygulanırsa daha fazla serbestlik derecesi olan ve daha kabiliyetli bir sistem geliřtirilebilir.

Hem eksen zgl denetimde hem de kartezyen zgl denetimde robotun daha kesin ve doęru denetiminin saęlanabilmesi iin geri besleme elemanı olarak sensrler kullanılabilir ve de robotun verilen iři daha az hatayla ve daha gvenli yapması saęlanabilir.

KAYNAKLAR

1. "ISO Standartı", **ISO 8373** (1994).
2. "Robot Güvenlik Standartı", **ANSI/RIA R15.06** (1999).
3. Sciavicco, L., Siciliano, B., "Modeling and control of robot manipulators", **The McGraw-Hill Companies, Inc.**, New York, 11-26 (1996).
4. Internet: Elektronik Wikipedia Ansiklopedisi, "Industrial Robot" http://en.wikipedia.org/wiki/Industrial_robot (2007).
5. "Vizyon 2023 Projesi, Mekatronik Strateji Grubu", **Mekatronik Strateji Raporu**, Ankara, 20-21 (2004).
6. "World Robotics 2006" Raporu, **World Robotics**, 30-40 (2006).
7. Koca, H., Erdem, H., "Microcontroller Based Control of a 6-Axes Robot Manipulator", **International IEEE Conference on Mechatronics and Robotics 2004 (Student Forum)**, Acchen, 50-56 (2004).
8. Hannan Bin Azhar, M.A., Dimond, K.R., "Design of an FPGA based adaptive neural controller for intelligent robot navigation", **Digital System Design Proceedings Euromicro Symposium**, Dortmund, 283–290 (2002).
9. Kongmunvattana, A., Chongstivatana, P., "A FPGA-based behavioral control system for a mobile robot", **Circuits and Systems, IEEE APCCAS, The 1998 IEEE Asia-Pacific Conference**, Chiangmai, 759–762 (1998).
10. He, P.M.H., Jin, L., Yang, R., Wei, Y.W., Liu, H.G., Cai, H., Liu, N., Butterfass, J., Hirzinger, G., "High performance DSP/FPGA controller for implementation of HIT/DLR dexterous robot hand", **Robotics and Automation Proceedings 2004 IEEE International Conference**, New Orleans, 3397 – 3402 (2004).
11. Li, T., Chang, S.T., Chen, Y.X., "Implementation of autonomous fuzzy garage-parking control by an FPGA-based car-like mobile robot using infrared sensors", **Robotics and Automation Proceedings ICRA '03 IEEE International Conference**, Japan, 3776 – 3781 (2003).
12. Okura, M., Matsumoto, A., Ikeda, H., Murase, K., "Artificial evolution of FPGA that controls a miniature mobile robot Khepera", **SICE 2003 Annual Conference**, Fukui, 2858 – 2863 (2003).

13. Ibanez, I., Aguirre, M.A., Torralba, A., Franquelo, L.G., "A low cost 3D vision system for positioning welding mobile robots using a FPGA prototyping system", **IECON 02 Industrial Electronics Society, IEEE 2002 28th Annual Conference**, Sevilla, 2:1590 – 1593 (2002).
14. Nagata, F., Watanabe, K., "Teaching system for a polishing robot using a game joystick", **Proceedings of the 39th SICE Annual Conference**, Lizuka, 179 – 184 (2000).
15. Kabuka, M., Glaskowsky, P., Miranda, J., "Microcontrollerbased Architecture for Control of a Six Joints Robot Arm", **IEEE Transactions On Industrial Electronics**, 35 (2):217-221 (1988).
16. Sage, H.G., De Mathelin, M.F., Ostertag, E., "Robust control of robot manipulators: a survey", **International Journal of Control**, 72(16), 1498-1522 (1999).
17. Moreno, J., Kelly, R., "Velocity control of robot manipulators: analysis and experiments", **International Journal of Control**, 76(14):1420-1427 (2003).
18. Cervantes, I., Garrido, R., Alvarez-Ramirez, J., Martinez, A., "Vision-Based PID Control of Planar Robots", **IEEE/ASME Transactions on Mechatronics**, 9(1), 132-136 (2004).
19. Jatta, F., Legnani, G., Visioli, A., Ziliani, G., "On the use of velocity feedback in hybrid force/velocity control of industrial manipulators", **Control Engineering Practice**, 14:1045-1055 (2006).
20. Craig, J.J., "Introduction to robotics: Mechanics and control", **MA: Addison-Wesley**, 20-27 (1989).
21. Lewis, F.L., Abdallah, C.T., Dawson, D. M., "Control of robot manipulators", **NY: MacMillan Publishing Company**, 40-45 (1993).
22. Sciavicco, L., Siciliano, B., "Modeling and control of robot manipulators", **2nd edition, London: Springer-Verlag**, 30-40 (2000).
23. Reyes, F., Rosado, A., "Polynomial family of PD-type controllers for robot manipulators", **Control Engineering Practice**, 13(4):441-450 (2005).
24. Paden, B., Panja, R., "Globally asymptotically stable PD controller for robot manipulator", **International Journal of Control**, 47(6):1697-1712 (1988).

25. An, C.H., Atkeson, C.G., Griffiths, J.D., Hollerbach, J.M., "Experimental evaluation of feedforward and computed torque control", **IEEE Transactions on Robotics and Automation**, 5(3):368-373 (1989).
26. Tarn, T.J., Bejczy, A.K., Marth, G.T., Ramadorai, A.K., "Performance comparison of four manipulator servo schemes", **IEEE Control Systems**, 22-29 (1993).
27. De Jager, B., Banens, J., "Experimental evaluations of robot controllers", **Proceedings of the 33rd conference on decision and control**, Lake Buena Vista, 363-368 (1994).
28. Doğan, M., Morgül, Ö., "Nonlinear PDE Control of Two-Link Flexible Arm with Nonuniform Cross Section", **Proceedings of the 2006 American Control Conference**, Minneapolis, 400-405 (2006).
29. Slotine, J.J., Li, W., "On the adaptive control of robot manipulators", **The International Journal of Robotics Research**, 6(3):49-59 (1987).
30. Long, P.H.W., Tsoi, K.H., "Field Programmable Gate Array Technology for Robotics Applications", **Robio05**, 70-76 (2005).
31. DeHon, A., "The density advantage of configurable computing", **Computer**, 33(4):41-49 (2000).
32. Reynolds, R., Smith, P., Bell, L., Keller, H., "The design of mars lander cameras for mars pathfinder, mars surveyor 98 and mars surveyor 01," **IEEE Transactions on Instrumentation and Measurement**, 50(1):63-71 (2001).
33. Boland, L., "Formula 1 racing: The Xilinx advantage," **Xcell**, 47:46-49 (2003).
34. Kung, Y.S., Shu, G.S., "Development of a FPGA-based Motion Control IC for Robot Arm", **ICIT 2005**, 1397-1402 (2005).
35. González-Gómez, J., Aguayo, E., Boemo, E., "Locomotion of a Modular Worm-like Robot using a FPGA-based embedded MicroBlaze Soft-processor", **CLAWAR**, 110-116 (2004).
36. Tomokuni, N., Saiga, M., Yabuta, T., "Distributed Controller for Cooperative Joint Torque Control of Compact Humanoid Robots", **IEEE-RAS**, 259-264 (2005).

37. Bonato, V., Sanches, A., Fernandes, K., Merino, M., Cardoso, J.M.P, Eduardo, S.V., Eduardo, M., "A real time gesture recognition system for mobile robots", **ICINCO**, 207-214 (2004).
38. Koca, H., Ertay Y., Bilge, H., Taplamaciođlu C., "Altı Eksenli Robot Kolunun FPGA Tabanlı Kontrol Örneđi", **TOK 05**, 560-566 (2005).

EKLER

EK-1 Eksen özgül denetleme modülünün kaynak kodu

```
module
controller(clock,input_capture,axis1,axis2,axis3,axis4,axis5,axis6,ledg);

input [5:0] input_capture;
input clock;
output axis1;
output axis2;
output axis3;
output axis4;
output axis5;
output axis6;
output ledg;
wire [1:0] control1;
wire [1:0] control2;
wire [1:0] control3;
wire [1:0] control4;
wire [1:0] control5;
wire [1:0] control6;

assign ledg=1;

joy a (input_capture,control1,control2,control3,control4,control5,control6);
controller ax1 (clock,control1,axis1);
controller ax2 (clock,control2,axis2);
controller ax3 (clock,control3,axis3);
controller ax4 (clock,control4,axis4);
controller ax5 (clock,control5,axis5);
controller ax6 (clock,control6,axis6);

endmodule
```

EK-2 Giriş denetleyicisi modülünün kaynak kodu (eksen-özümlü)

```

module
joy(input_capture,control1,control2,control3,control4,control5,control6);
  input [5:0] input_capture;
  output [1:0] control1;
  output [1:0] control2;
  output [1:0] control3;
  output [1:0] control4;
  output [1:0] control5;
  output [1:0] control6;
  reg [1:0] control1;
  reg [1:0] control2;
  reg [1:0] control3;
  reg [1:0] control4;
  reg [1:0] control5;
  reg [1:0] control6;
  always @(input_capture)
  begin
  case (input_capture)
  6'b001000:
    begin
      control1=2'b01;           // 1. eksen sol
      control2=2'b00;
      control3=2'b00;
      control4=2'b00;
      control5=2'b00;
      control6=2'b00;
    end
  6'b000100:
    begin
      control1=2'b10;           // 1. eksen sağ
      control2=2'b00;
      control3=2'b00;
      control4=2'b00;
      control5=2'b00;
      control6=2'b00;
    end
  6'b100000:
    begin
      control2=2'b10;           // 2. eksen yukarı
      control1=2'b00;
      control3=2'b00;
      control4=2'b00;
      control5=2'b00;
      control6=2'b00;
    end
  end
  end

```

EK-2 (Devam) Giriş denetleyicisi modülünün kaynak kodu (eksen-öztgöl)

```
end  
  
6'b010000:  
begin  
    control2=2'b01;  
    control1=2'b00;  
    control3=2'b00;  
    control4=2'b00;  
    control5=2'b00;  
    control6=2'b00;  
end  
6'b100001:  
begin  
    control3=2'b01;  
    control2=2'b00;  
    control1=2'b00;  
    control4=2'b00;  
    control5=2'b00;  
    control6=2'b00;  
end  
6'b010001:  
begin  
    control3=2'b10;  
    control2=2'b00;  
    control1=2'b00;  
    control4=2'b00;  
    control5=2'b00;  
    control6=2'b00;  
end  
6'b100010:  
begin control4=2'b10;  
    control2=2'b00;  
    control3=2'b00;  
    control1=2'b00;  
    control5=2'b00;  
    control6=2'b00;  
end  
6'b010010: control4=2'b01;  
6'b001001: control5=2'b10;  
6'b000101: control5=2'b01;  
6'b001010: control6=2'b10;  
6'b000110: control6=2'b01;
```

EK-2 (Devam) Giriş denetleyicisi modülünün kaynak kodu (eksen-ölgül)

```
default:
    begin
        control1=2'b00;
        control1=2'b00;
        control2=2'b00;
        control2=2'b00;
        control3=2'b00;
        control3=2'b00;
        control4=2'b00;
        control4=2'b00;
        control5=2'b00;
        control5=2'b00;
        control6=2'b00;
        control6=2'b00;
    end
endcase
end
endmodule
```


EK-3 Eksen denetleyicisi modülünün kaynak kodu (eksen-özügöl)

```
module controller(clock,control,axis);
    input clock;
    input [1:0] control;
    output axis;

    reg axis;
    reg [17:0] width,width_count;
    reg [19:0] period;

    parameter PERYOT=1000000;
    parameter GENIS=75000;
    parameter MIN=45000;
    parameter MAX=105000;

    initial begin

        width=GENIS;
        period=PERYOT;
        axis=1;
        width_count=width;

    end

    always @(posedge clock)
    begin

        if (width_count>0) begin
            width_count=width_count-1;
        end
        else begin
            axis=0;
        end
        if (period>0) begin
            period=period-1;
        end
        else begin
            axis=1;
            period=PERYOT;
            width_count=width;
        end
    end
end
```

EK-3 (Devam) Eksen denetleyicisi modülünün kaynak kodu (eksen-özügöl)

```
always @(negedge axis)
begin
```

```
case(control)
```

```
    6'b01: if (width>MIN) begin width=width-10000; end
```

```
    6'b10: if (width<MAX) begin width=width+10000; end
```

```
endcase
```

```
end
```

```
endmodule
```

EK-4 Giriş denetleyicisi modülünün kaynak kodu (kartezyen-özümlü)

```
module inputcontroller(input_capture,axis,x,y,din);

input [5:0] input_capture;
input axis;
output [11:0] x;
output [11:0] y;
output [14:0] din;

reg [11:0] x;
reg [11:0] y;
reg [11:0] xo;
reg [11:0] yo;
reg [14:0] din;

parameter xinitial=2400;
parameter yinitial=1200;
parameter dinitial=28800;

initial
begin
    xo=xinitial;
    yo=yinitial;
    din=dinitial;
end

always @(negedge axis)
begin
    case (input_capture)
    6'b001000: begin
        yo=yo+1;
        y=yo;
        end
    6'b000100: begin
        yo=yo-1;
        y=yo;
        end
    6'b100000: begin
        xo=xo+1;
        x=xo;
        end
    end
end
```

EK-4 (Devam) Giriş denetleyicisi modülünün kaynak kodu (kartezyen-özügöl)

```
        6'b010000: begin
            xo=xo-1;
            x=xo;
            end
        default:begin
            x=xo;
            y=yo;
            end
    endcase
end
endmodule
```

EK-5 Ters kinematik modülünün verillog kodu (kartezyen-özümlü)

```
module inversekinematics(clk,x,y,din,teta1,teta2,teta3);
```

```
input clk;
input [11:0] x;
input [12:0] y;
input [14:0] din;
```

```
output [9:0] teta1;
output [7:0] teta2;
output [7:0] teta3;
```

```
wire [25:0] a1_out;
wire [12:0] s1_out;
wire [24:0] m3_out;
wire [25:0] m1_out;
wire [19:0] m2_out;
wire [25:0] d2_out;
wire [25:0] s2_out;
wire [19:0] d1_out;
wire [7:0] d3_out;
wire [7:0] a3_out;
wire [7:0] arctan_out;
```

```
s1 sub1 (x,s1_out,clk);
m1 mult1(y,y,0,clk,m1_out);
m2 mult2(y,0,clk,m2_out);
m3 mult3(s1_out,s1_out,0,clk,m3_out);
a1 add1 (m3_out,m1_out,a1_out,clk);
d1 div1 (,clk,m2_out,s1_out,d1_out,);
d2 div2 (,clk,a1_out,din,d2_out,);
s2 sub2 (d2_out,s2_out,clk);
arccos arcc (clk,s2_out[7:0],teta2);
arctan arct (clk,d1_out[9:0],arctan_out);
d3 div3 (,clk,teta2,2,d3_out,);
a2 add2 (d3_out,arctan_out,teta1,clk);
a3 add3 (teta2,a3_out,clk);
s3 sub3 (a3_out,teta1[7:0],teta3,clk);
```

```
endmodule
```

EK-6 Düz kinematik modülünün verilog kodu (kartezyen-özügöl)

```

module directkinematics(clk,teta1,teta2,teta3,sin23_out,x2,y2);
input clk;
input [9:0] teta1;
input [9:0] teta2;
input [9:0] teta3;
output [12:0] x2;
output [12:0] y2;
output [7:0] sin23_out;
wire [9:0] s21_out;
wire [7:0] c23_out;
wire [7:0] c22_out;
wire [7:0] sin22_out;
wire [7:0] c21_out;
wire [7:0] sin21_out;
wire [11:0] a21_out;
wire [12:0] a22_out;
wire [12:0] a23_out;
wire [11:0] m23_out;
wire [11:0] m25_out;
wire [11:0] m26_out;
wire [11:0] m22_out;
wire [11:0] m21_out;
wire [11:0] m24_out;

s21 sub21 (teta1,teta2,s21_out,clk);
cosine c21 (clk,teta1,c21_out);
sine sin21 (clk,teta1,sin21_out);
a21 add21 (teta3,s21_out,a21_out,clk);
cosine c22 (clk,s21_out,c22_out);
sine sin22 (clk,s21_out,sin22_out);
m21 mult21(c21_out,0,clk,m21_out);
m22 mult22(sin21_out,0,clk,m22_out);
cosine c23 (clk,a21_out,c23_out);
sine sin23 (clk,a21_out,sin23_out);
m23 mult23(c22_out,0,clk,m23_out);
m24 mult24(sin22_out,0,clk,m24_out);
m25 mult25(c23_out,0,clk,m25_out);
m26 mult26(sin23_out,0,clk,m26_out);
a22 add22 (m24_out,m22_out,a22_out,clk);
a23 add23 (m26_out,m23_out,a23_out,clk);
s22 sub22 (m25_out,a22_out,y2,clk);
a24 add24 (a23_out,m21_out,x2,clk);

endmodule

```

EK-7 Limit kontrol modülünün verilog kodu (kartezyen-öztgöl)

```
module limitcheck(clk,teta1,teta2,teta3,limit_check);
input  clk;
input [8:0] teta1;
input [8:0] teta2;
input [8:0] teta3;
output limit_check;

wire comp1_out;
wire comp2_out;
wire comp3_out;
wire comp4_out;
wire comp5_out;
wire comp6_out;
wire and1_out;
wire and2_out;
wire and3_out;
co1 comp1(comp1_out,clk,teta1);
co2 comp2(clk,comp2_out,teta1);
and and1 (and1_out,comp1_out,comp2_out);
co3 comp3(comp3_out,clk,teta2);
co4 comp4(clk,comp4_out,teta2);
and and2 (and2_out,comp3_out,comp4_out);
co5 comp5(comp5_out,clk,teta3);
co6 comp6(clk,comp6_out,teta3);
and and3 (and3_out,comp5_out,comp6_out);
and and4 (limit_check,and1_out,and2_out,and3_out);

endmodule
```

EK-8 Hata kontrol modülünün verilog kodu (kartezyen-özümlü)

```

module errorcheck(clk,x,x2,y,y2,sin2,error_check);

input clk;
input [12:0] x;
input [12:0] x2;
input [12:0] y;
input [12:0] y2;
input [7:0] sin2;
output error_check;
wire [13:0] sub31_out;
wire [13:0] sub32_out;
wire [8:0] sub33_out;
wire comp21_out;
wire comp22_out;
wire comp23_out;
wire comp24_out;
wire comp25_out;
wire comp26_out;
wire comp27_out;
wire comp28_out;
wire comp29_out;
wire comp30_out;
wire comp31_out;
wire comp32_out;
wire and21_out;
wire and22_out;
wire and23_out;
wire and24_out;
wire and25_out;
wire and26_out;
wire or1_out;
wire or2_out;
wire or3_out;

s31 sub31 (x,x2,sub31_out,clk);
co21 comp21(comp21_out,clk,sub31_out);
co22 comp22(clk,comp22_out,sub31_out);
and and21 (and21_out,comp21_out,comp22_out);
co23 comp23(clk,comp23_out,sub31_out);
co24 comp24(comp24_out,clk,sub31_out);
and and22 (and22_out,comp23_out,comp24_out);
or or1 (or1_out,and21_out,and22_out);
s32 sub32 (y,y2,sub32_out,clk);

```


EK-8 (Devam) Hata kontrol modülünün verilog kodu (kartezyen-özüml)

```
co25 comp25(comp25_out,clk,sub32_out);
co26 comp26(clk,comp26_out,sub32_out);
and and23 (and23_out,comp25_out,comp26_out);
co27 comp27(clk,comp27_out,sub32_out);
co28 comp28(comp28_out,clk, sub32_out);
and and24 (and24_out,comp27_out,comp28_out);
or or2 (or2_out,and23_out,and24_out);
s33 sub33 (sin2,sub33_out,clk);
co29 comp29(comp29_out,clk,sub33_out);
co30 comp30(clk,comp30_out,sub33_out);
and and25 (and25_out,comp29_out,comp30_out);
co31 comp31(clk,comp31_out,sub33_out);
co32 comp32(comp32_out,clk,sub33_out);
and and26 (and26_out,comp31_out,comp32_out);
or or3 (or3_out,and25_out,and26_out);
and and27 (error_check,or1_out,or2_out,or3_out);

endmodule
```

EK-9 Kinematik modülün verilog kodu (kartezyen-özügöl)

```
module kinematics
(clk,x,y,din,error_check,limit_check,teta1,teta2,teta3,x2,y2,sin23_out);

input    clk;
input  [11:0] x;
input  [12:0] y;
input  [14:0] din;
output   error_check;
output   limit_check;
output  [9:0] teta1;
output  [7:0] teta2;
output  [7:0] teta3;
output  [7:0] sin23_out;
output  [12:0] x2;
output  [12:0] y2;

inversekinematics ik (clk,x,y,din,teta1,teta2,teta3);
directkinematics  dk (clk,teta1,{0,0,teta2},{0,0,teta3},sin23_out,x2,y2);
limitcheck        lc (clk,teta1[8:0],{0,teta2},{0,teta3},limit_check);
errorcheck        ec (clk,{0,x},x2,y,y2,sin23_out,error_check);

endmodule
```

EK-10 Açık-sinyal genişliği modülünün verilog kodu (kartezyen-özümlü)

```
module angeltosignal(clk,teta1,teta2,teta3,sg1,sg2,sg3);

input clk;
input [7:0] teta1;
input [7:0] teta2;
input [7:0] teta3;
output [17:0] sg1;
output [17:0] sg2;
output [17:0] sg3;
wire [7:0] t1;
wire [7:0] t2;
wire [7:0] t3;
wire [17:0] sg1;
wire [17:0] sg2;
wire [17:0] sg3;
wire [8:0] tet1;
wire [8:0] tet2;
wire [8:0] tet3;
wire [8:0] te1;
wire [8:0] te2;
wire [8:0] te3;

assign tet1=teta1+teta1;
assign tet2=teta2+teta2;
assign tet3=teta3+teta3;

dts1 divts1(,clk,tet1,3,te1,);
dts1 divts2(,clk,tet2,3,te2,);
dts1 divts3(,clk,tet3,3,te3,);

assign t1=te1+90;
assign t2=te2+90;
assign t3=te3+90;

assign sg1=t1*1000;
assign sg2=t2*1000;
assign sg3=t3*1000;

endmodule
```

EK-11 Eksen denetleyicisi modülünün verilog kodu (kartezyen-özügöl)

```

module
axiscontroller(clk,input_capture,width_new,error_check,limit_check,axis);

input clk;
input [17:0] width_new;
input [5:0] input_capture;
input error_check;
input limit_check;
output axis;
reg axis;
reg [17:0] width,width_count;
reg [20:0] period;
parameter PERYOT=2000000;
parameter GENIS=150000;
parameter MIN=90000;
parameter MAX=210000;
initial
begin
width=GENIS;
period=PERYOT;
axis=1;
width_count=width;
end

always @(posedge clk)
begin
if (width_count>0) begin
width_count=width_count-1;
end
else begin
axis=0;
if(input_capture!=0 && error_check==1 && limit_check==1)
width=width_new;
end
if (period>0) begin
period=period-1;
end
else begin
axis=1;
period=PERYOT;
width_count=width;
end
end
endmodule

```

EK-12 Kartezyen-özügöl denetleyicinin verilög kodu

```

module robotcontroller(clk,input_capture,axis1,axis2,axis3);

input clk;
input [5:0] input_capture;
output axis1;
output axis2;
output axis3;
wire [11:0] x;
wire [12:0] y;
wire [14:0] din;
wire error_check;
wire limit_check;
wire [9:0] teta1;
wire [7:0] teta2;
wire [7:0] teta3;
wire [7:0] sin23_out;
wire [12:0] x2;
wire [12:0] y2;
wire [17:0] sg1;
wire [17:0] sg2;
wire [17:0] sg3;

inputcontroller incon(input_capture,axis1,x,y,din);
kinematics      kine
  (clk,x,y,din,error_check,limit_check,teta1,teta2,teta3,x2,y2,sin23_out);
angeltoSignal atos (clk,teta1,teta2,teta3,sg1,sg2,sg3);
axiscontroller ax1 (clk,input_capture,sg1,error_check,limit_check,axis1);
axiscontroller ax2 (clk,input_capture,sg2,error_check,limit_check,axis2);
axiscontroller ax3 (clk,input_capture,sg3,error_check,limit_check,axis3);

endmodule

```

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : KOCA, Hikmet
 Uyuğu : T.C.
 Doğum tarihi ve yeri : 30.04.1982 Eskişehir
 Telefon : 0 (533) 343 49 67
 E-mail : hikmetkoca@gmail.com

Eğitim

Derece	Eğitim Birimi	Mezuniyet tarihi
Yüksek lisans	Gazi Üniversitesi / Elektrik Elektronik Müh.	2007
Lisans	Başkent Üniversitesi/ Elektrik Elektronik Müh.	2004
Lise	Anıttepe Lisesi	1999

İş Deneyimi

Yıl	Yer	Görev
2005-(devam)	Gazi Üniversitesi	Uzman
2004-2005	Atılım Üniversitesi	Araştırma Görevlisi

Yabancı Dil

İngilizce

Yayınlar

- Koca, H., Erdem H. ,“Microcontroller Based Control of a 6-axes Robot Manipulator”, *International IEEE Conference on Mechatronics and Robotics*, 1:70-76 (2004).
- Koca, H., Ertaş Y., Bilge, H., Taplamacıoğlu C., “Altı Eksenli Robot Kolunun FPGA Tabanlı Kontrol Örneği”, *TOK 05*, 560-566 (2005)

Hobiler

İnternette gezinti, Robotik, Yüzmek ve balık avlamak