

Görüntü Tabanlı Delta Paralel Robot

Gürkan KALINAY; Erdiñ HASTEN; Ali KUTOĞLU

Mühendislik ve Mimarlık Fakültesi, Elektrik Elektronik Bölümü Erzurum Teknik Üniversitesi

Özet:

Bu çalışmada; 3 serbestlik dereceli delta paralel robotun analizi, tasarımı ve reel zamanlı yörünge takip kontrolü gerçekleştirilmiştir. Tasarlanan delta paralel robotta konum bilgisini elde etmek için ters kinematik denklemlerin analizinden faydalanılmıştır. Paralel manipülatörün konum bilgisini aldıktan sonra servo motorların dönüş açılarını belirlemek adına analizler yapılmıştır. Delta paralel robotun gerçek zamanlı konum tespitini yapması için NI LabVIEW programı ile çalışan bir kontrol kartı olan MYRIO'dan yararlanılmıştır. Yapılan çeşitli deneylerle elde edilen veriler ışığında analizlerin doğruluğu kanıtlanmıştır. Ayrıca delta paralel robotun kontrolü için labVIEW yazılımı ile MYRIO kullanılarak bu tür manipülatörleri kontrol etmek için farklı bir bakış açısı getirilmiştir.

Anahtar Kelimeler: Servo Motor; Myrio; LabVIEW

Delta Parallel Robot With Image Processing

Abstract:

In this study; Analysis, design and real-time trajectory tracking control of 3-degree-of-freedom delta parallel robot have been carried out. In the designed delta parallel robot, the analysis of the inverse kinematic equations is used to obtain the position information. After receiving the position information of the parallel manipulator, analyzes were performed to determine the turning angles of the servo motors. The Delta parallel robot uses MYRIO, a control card that works with the NI LabVIEW program to perform real-time location detection. The validity of the analysis in the data obtained by various experiments is proven. In addition, a different perspective has been introduced to control such manipulators using MYRIO with the LabVIEW software for delta parallel robot control.

Key words Servo-motor; Myrio; Labview

1.Giriş

Bir delta paralel robot sabit platform üzerine bağlanmış 3 adet kol ve bu kolları yönlendiren 3 adet motordan oluşmaktadır. Bu 3 kolun her biri birbirinden bağımsız bir şekilde hareket edebilmekte ve bu hareketler bir kinematik zincirle hareketsiz platforma bağlantısı sağlanmaktadır[1].

Günümüze kadar robotik teknolojisi seri kinematik üzerinden gelişimini sağlamıştır. Ancak sanayinin istediği hızlara ulaşabilmek bu teknoloji ile karşılanamayacak düzeye gelmiştir. Eksenlerin üst üste konulduğu yapılarda büyük mekanizmalar ve daha ağır eklemler kullanıldı. Ancak bu tasarım şekli daha büyük motorlara ihtiyaç duyulması sonucunu doğurdu. Kütleler arttıkça ataletler büyüyor, ivmeler ve hızların düşmesine sebep oluyordu. Bu tarz sorunlar tasarımda gidilen değişikliklerle düzenlendi. Paralel robot yapısında eklemler üst üste konulması yerine hepsi aynı gövde üzerine konumlanıyor. Böylece hiçbir kol bir diğerrinin yükünü sırtlamak zorunda kalmıyor. Bütün kollar artık motorun kendi gövdesini kaldırmak yerine hareket ettirilecek yükü kaldıracaktır. Robot eklemlerinin hafifletilmesi ile motorlar daha dinamik çalışmaktadır[2].

Delta robot günümüzdeki en başarılı paralel manipülatördür[3]. Gıda, ilaç, biyomedikal, ambalaj, elektrik ve kimyasalda paketlenme ve depolama gibi işlemlerde yaygın olarak kullanılmaktadır[4]. Çok hızlı olması nedeniyle üretim hatlarında pick and place işlemini gerçekleştirmek için yaygın olarak kullanılır[4].

Delta robot; eş zamanlı çalışabilme kabiliyeti ve konum hassasiyeti nedeniyle çok kullanışlı bir yapıya sahiptir[5].

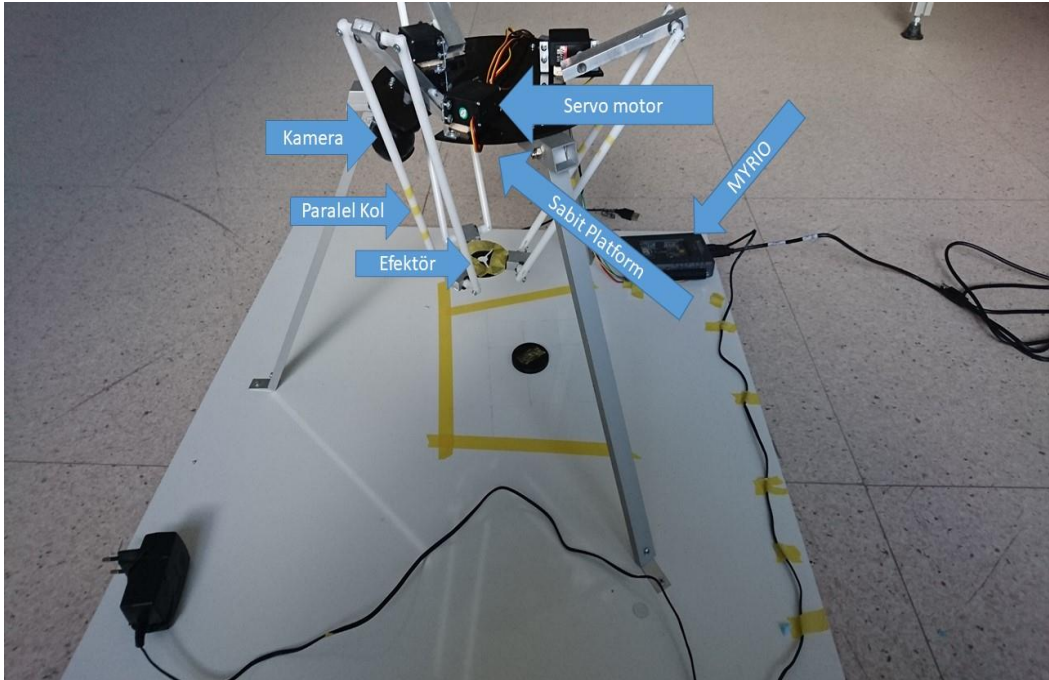
Bu çalışmada; 3 serbestlik derecesine sahip delta paralel robot analizi, tasarımı ve reel zamanlı yörünge takip kontrolü gerçekleştirilmiştir. Matematiksel analizler için ters kinematik denklemler kullanılmıştır. Bu sayede görüntü işleme ile alınan konum bilgisi için efektörün o konuma gidebilmesi için servo motorların dönmesi gereken açıların hesabı yapılmıştır. Kontrol için LabVIEW tabanlı MYRIO adı verilen kontrol kartı kullanılmıştır.

2. Materyal ve Yöntem

Delta paralel robotun boyut analizleri, ters kinematik analizleri ve LabVIEW tabanlı görüntü işleme ve yazılımı yapılarak bu optimizasyonları destekleyecek şekilde tasarım kısmı ve boyut analizi yapılmıştır. Bu sayede katı modeller kullanılarak CAM kodları çıkarılmış ve 3 eksenli CNC freze kullanılarak robotun hareketli platform, sabit platform, tutucu uç ve kolları çıkarılmıştır. Bunun yanı sıra verim açısından ve tork açısından tasarımımıza en uygun motorlarda temin edilmiştir. Bu aşamadan sonra elde edilen parçalar ve motorlar birleştirilerek robotun montajı tamamlanmıştır.

Delta paralel robotun görüntü işleme ve hareketini sağlayacak yazılımı LabVIEW tabanlı MYRIO kullanılarak yapılmıştır. Bu sayede kameranın gördüğü ve taradığı alandaki bir cismi geometrik yapısına göre algıladıktan sonra bu cismin hangi X,Y,Z kartezyen noktasında olduğunu hesaplanmakta ve yazılım kısmındaki ters kinematik eşitlikleri kullanılarak motorların bu noktaya gitmek için hangi açılarla gideceği hesaplanmıştır. Sonraki adımda hareketli platform bu noktaya giderek cismi almakta ve kullanıcının belirlediği herhangi bir noktaya götürüp bırakmaktadır.

Şekil 1. de tasarımın son hali ve tasarımı oluşturan kısımların açıklaması görülmektedir.

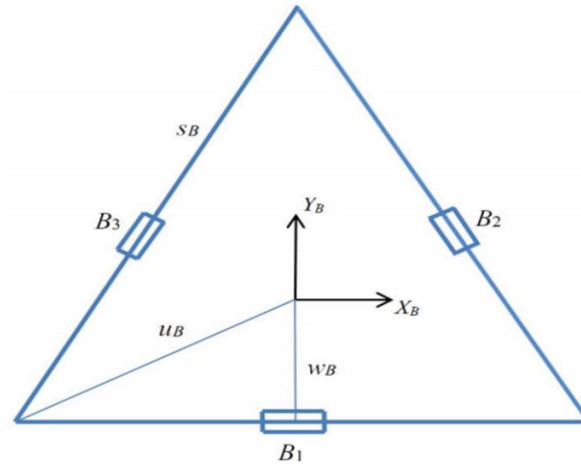


Şekil 1. Yapılan Tasarımın Son Hali

3.Analizler

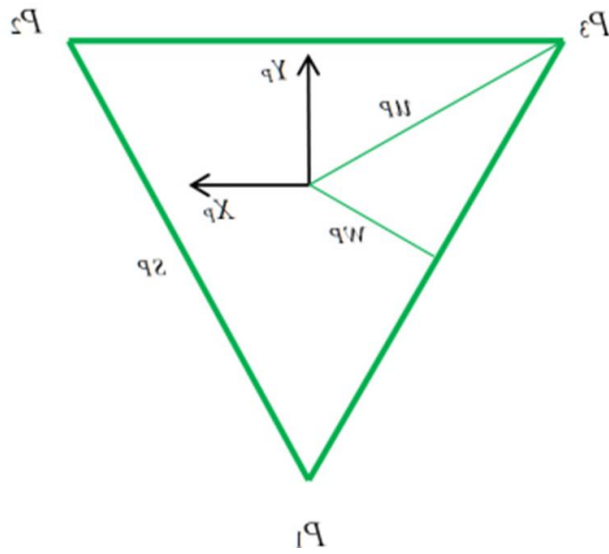
Görüntü tabanlı delta paralel robot ters kinematik hesaplamaları yapılmıştır. Bu hesaplamalar için sabit platform ve hareketli platformun eşkenar üçgen geometrisine sahip olduğu düşünülmüştür.

Sabit platformun eşkenar üçgen geometrisi için:



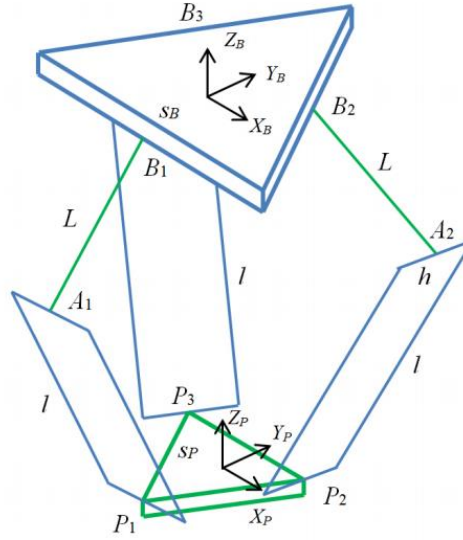
Şekil-2

Hareketli platform üçgen geometrisi için:



Şekil 3

Delta paralel robotun geometrisi için:



Şekil 4

Sabit platform, hareketli platform ve delta paralel robot için uzunluk değerleri :

ADI	ANLAMI	DEĞERİ(mm)
SB	Sabit eksen üçgen kenarı	$180\sqrt{3}$
SP	Hareketli eksen üçgen kenarı	$45\sqrt{3}$
L	Kısa bacak uzunluğu	90
l	Uzun bacak uzunluğu	260
WB	Sabit eksen merkezden kenara uzaklık	90
UB	Sabit eksen yarıçapı	180
WP	Hareketli eksen merkezden kenara uzaklık	45
UP	Hareketli eksen yarıçapı	45

Tablo-1

Ters kinematik denklemlerin çözümü için kullanılan analitik denklem:

$$(1) E_i \cos \theta_i + F_i \sin \theta_i + G_i = 0 \quad i=1,2,3$$

Yukarıdaki denklemden:

$$(2) E_1 = 2L(y + a)$$

$$(3) F_1 = 2Lz$$

$$(4) G_1 = x^2 + y^2 + z^2 + a^2 + L^2 + 2ya - l^2$$

$$(5) E_2 = -L(\sqrt{3}(x + b) + y + c)$$

$$(6) F_2 = 2Lz$$

$$(7) G_2 = x^2 + y^2 + z^2 + b^2 + c^2 + L^2 + 2(xb + yc) - l^2$$

$$(8) E_3 = L(\sqrt{3}(x - b) - y - c)$$

$$(9) F_3 = 2Lz$$

$$(10) G_3 = x^2 + y^2 + z^2 + b^2 + c^2 + L^2 + 2(-xb + yc) - l^2$$

Buradaki a,b ve c değerleri için denklemler:

$$(11) a = w_b - u_p$$

$$(12) b = \frac{s_p - \sqrt{3}w_b}{2}$$

$$(13) c = w_p - \frac{w_b}{2}$$

Buradaki denklemlerin çözümleri için tanjant yarım açısı formülleri kullanılmaktadır. Bu yarım açısı formülleri için çıkarımlar aşağıdaki gibi yapılır. Burada t_i 'yi aşağıdaki gibi tanımlanır:

$$(14) t_i = \tan \frac{\theta_i}{2}$$

Buradan:

$$(15) \cos \theta_i = \frac{(1-t_i^2)}{(1+t_i^2)}$$

$$(16) \sin \theta_i = \frac{2t_i}{(1+t_i^2)}$$

Yukarıdaki 14,15 ve 16 numarada bulduğumuz tanjant yarım açısı denklemlerini 1 numaralı denklemde yerine yazılırsa:

$$(17) E_i(1 - t_i^2) + F_i(2t_i) + G_i(1 + t_i^2) = 0$$

Buradan:

$$(18) (G_i - E_i)t_i^2 + (2F_i)t_i + (G_i + E_i) = 0$$

18 numaralı denklemden ikinci dereceden bir denklemin kökleri bulunacak olursa t_1 ve t_2 aşağıdaki gibi bulunur.

$$(19) t_{1,2} = \frac{-F_i \pm \sqrt{E_i^2 + F_i^2 - G_i^2}}{G_i - E_i}$$

Son olarak 14 numaralı denklemde tanjantın tersi alınarak θ_i açıları bulunur.

$$(20) \theta_i = 2 \tan^{-1}(t_i)$$

Denklem 20 sayesinde sırasıyla motor1, motor2 ve motor3'e ait θ_1, θ_2 ve θ_3 açı değerleri hesaplanabilmektedir. Yukarıdaki 19 numaralı denklemde bulunan iki kökte 20 numaralı denklem için kullanılabilmektedir.

4. Program ve İşleyiş

Görüntü tabanlı delta paralel robotun kontrolcüsü olarak NI MyRio belirlenmiştir. Programlamada NI LabVIEW kullanılmıştır. Program algoritması 4 alt 1 ana program şeklinde oluşturulmuştur;

a-Görüntü Al

b-Görüntü İşle

c-Kinematik Hesapla

d-Motor Kontrol Et

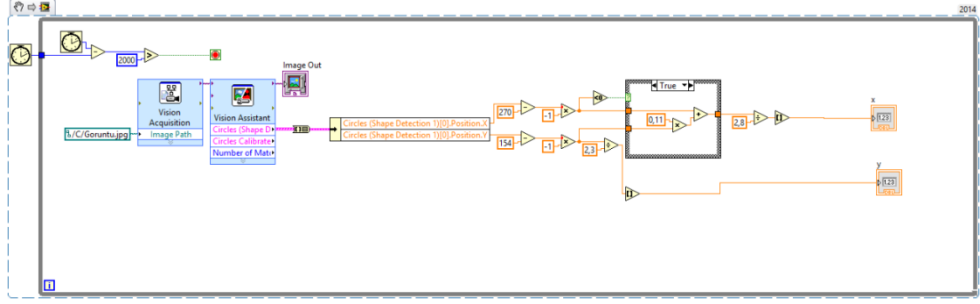
A) Görüntü Al

Bu bölümde kameradan gelen görüntü kontrolcü sunucusuna kaydeden program oluşturulmuştur. Bu işlemi gerçekleştirebilmek için sunucuya önceden bağlanması gerekmektedir. Kontrolcünün IP adresi alındıktan sonra sunucunun

bilgisayara tanımlanması yapılmıştır. Programda görüntüyü almak için program içerisindeki hazır bulunan asistanlardan faydalanılmıştır. Görüntünün kaydedileceği adres sunucu adresi olarak belirlenmiştir. Program While fonksiyonu ile sonsuz bir döngü haline getirilmiştir. Programda kullanılan zaman fonksiyonu ile programın 3 saniyelik periyotlar halinde sunucuya görüntü kaydedilmesi sağlanmıştır.

Şekil 5

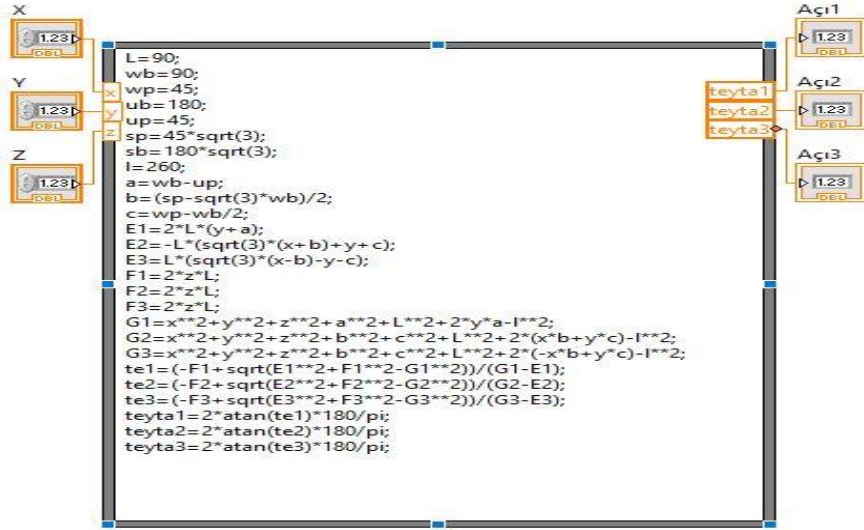
Bu bölümde daha önceden sunucuya kaydedilen görüntünün içerisindeki cismin konumunu görüntü işleme yardımıyla tespit eden bir program oluşturulmuştur. Program içerisinde görüntü işleme asistanı kullanılmıştır. Asistanın görevi görüntü içerisindeki dairesel cismin konumunu tespit etmek olarak belirlenmiştir. Bu işlemi kolaylaştırmak için renk filtreleri kullanılmıştır. Asistan tarafında tespit edilen konumlar gerçek hayata uygun hale getirebilmek için uygun çarpanlarla çarpılmıştır. Bu alt programın çıkışı X,Y ve Z koordinatları olarak belirlenmiştir.



Şekil 6

C) Kinematik Hesapla

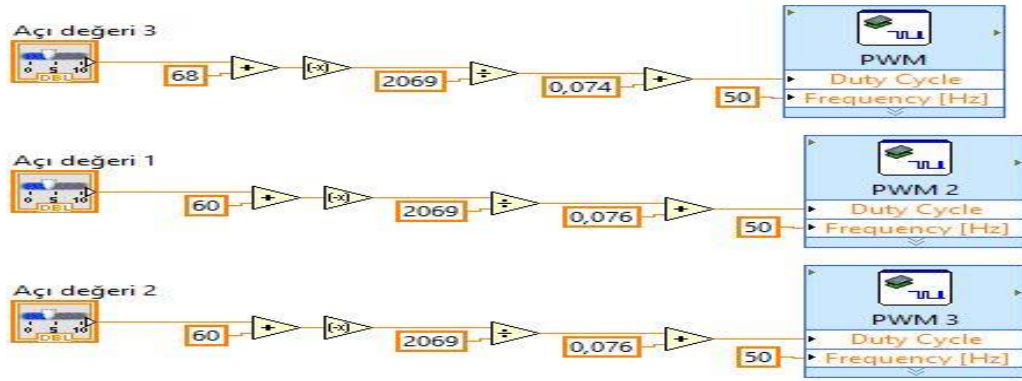
Bu bölümde bir önceki alt programın çıkışı olan konum değerlerine göre robotta bulunan motorların dönmesi gereken açıları hesaplayan program oluşturulmuştur. Ölçümlerle belirlenen sabit uzunluklar program içerisinde belirtilmiştir. Ardından kinematik denklemler yazılıp açıların hesaplanması sağlanmıştır. Hesaplanan açıların programın çıkışı olarak belirlenmiştir.



Şekil 7

D) Motor Kontrol Et

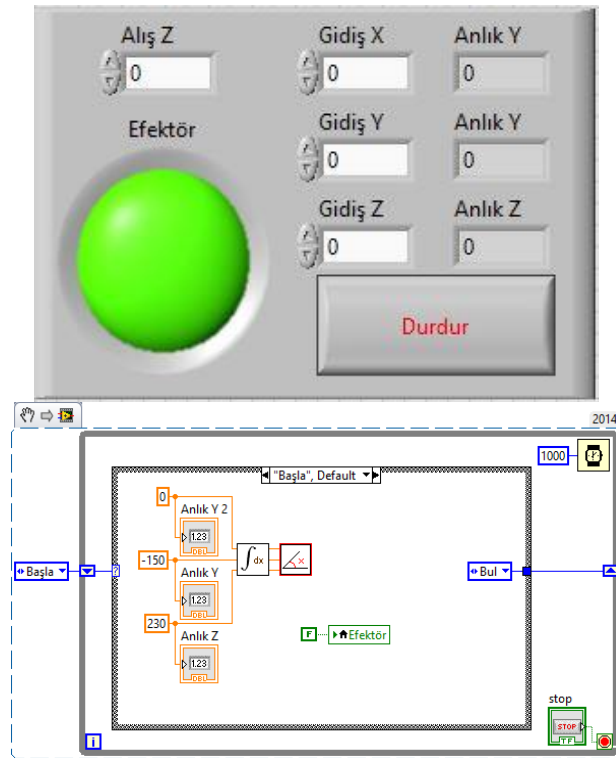
Bu bölümde motorların bir önceki programın çıkışı olan açı değerlerine gelmesini sağlayan program oluşturulmuştur. Motora gönderilen sinyal PWM formunda olduğu belirlenmiştir. Açılara gözlemlenilen çarpanlar uygulanmıştır. Çarpan uygulandıktan sonraki değerler PWM asistanı yardımıyla motor pinlerine gönderilmiştir.



Şekil 8

Ana Program

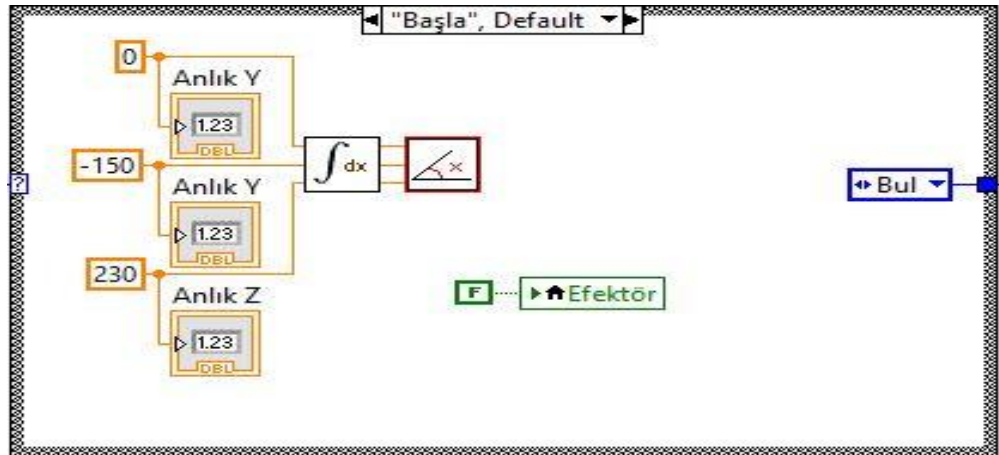
Ana program yeni bir sayfada oluşturulmuştur ve oluşturulurken LabVIEW programlama dilinde yaygın olarak kullanılan State Machine tekniği kullanılmıştır. Bu Program 7 kısımda oluşmaktadır.



Şekil 9

I) Başla

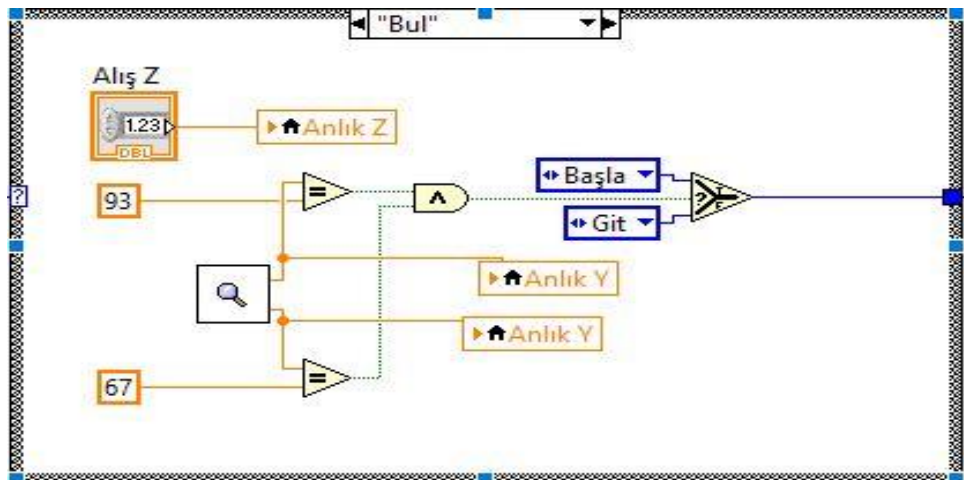
Efektörü açar, başlangıç konumuna gider ve programı Görüntü İşle'ye gönderir. Bunun için D programı kullanılır.



Şekil 10

II) Görüntü İşle

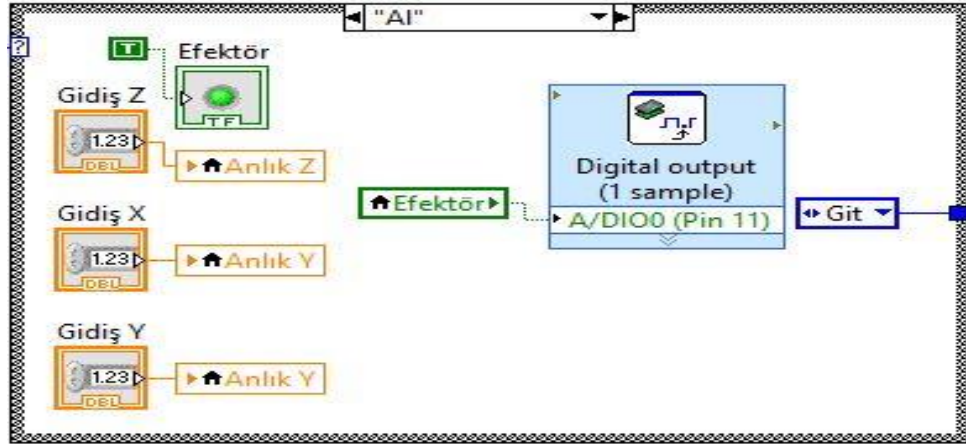
Sunucuda olan görüntüyü işler ve cismin konumunu tespit eder. Eğer cisim yoksa tekrar görüntü işleme yapar. Cisim tespit edilirse konum bilgileri ile beraber programı Hesapla'ya gönderir.



Şekil 11

III) Git

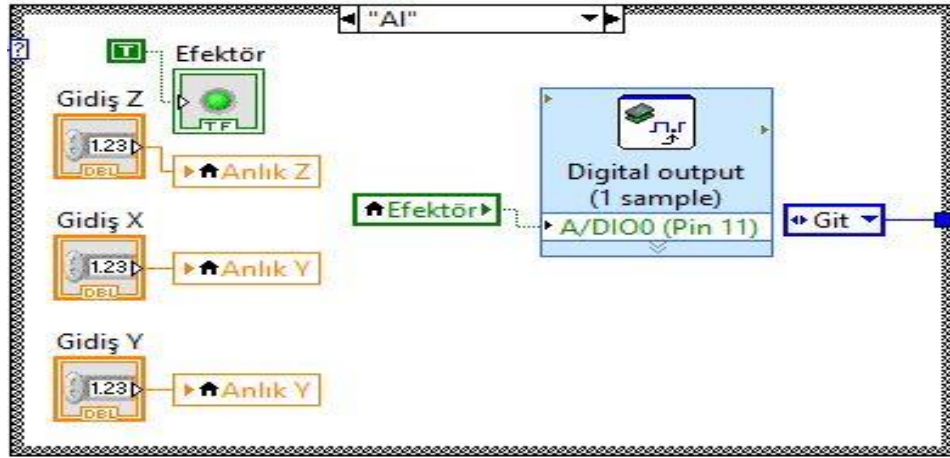
Program ile gelen konum bilgilerini kinematik denklemler yardımıyla işler ve açı değerlerini hesaplar. Hesaplanan açı bilgilerini D programı ile motorlara iletir ve motorların dönmesini sağlar. Ardında programı efektör açık ise Tut'a kapalı ise Bırak'a yönlendirir.



Şekil 12

IV) Tut

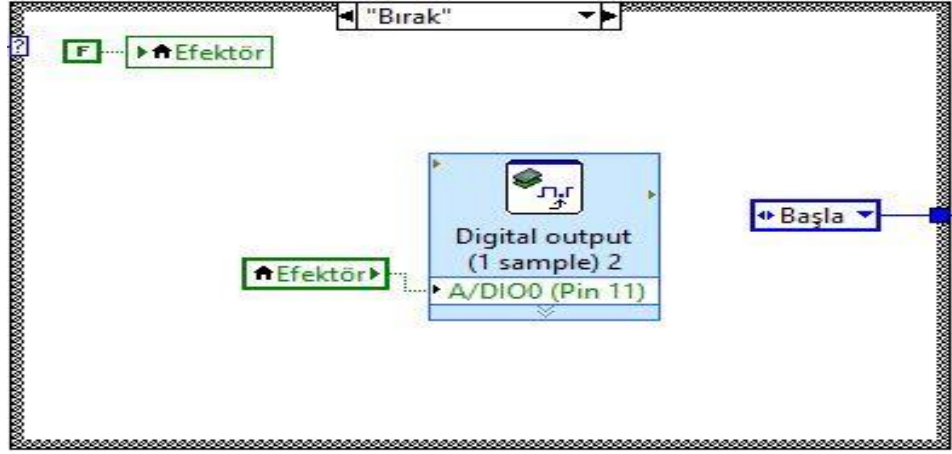
Efektörü kapatır ve kullanıcı tarafında girilen bırakma konum bilgileri ile beraber programı Hesapla'ya yönlendirir. Bu işlem basit bir digital output fonksiyonu ile gerçekleştirilir.



Şekil 13

V) Bırak

Efektörü açar ve programı Başla'ya gönderir.



Şekil 14

5. Deneyler

Deney 1:

Efektörün A(0,0,230) noktasından B(50,50,250) noktasına hareket ettirmek için gereken kontrol sinyallerini oluşturma.

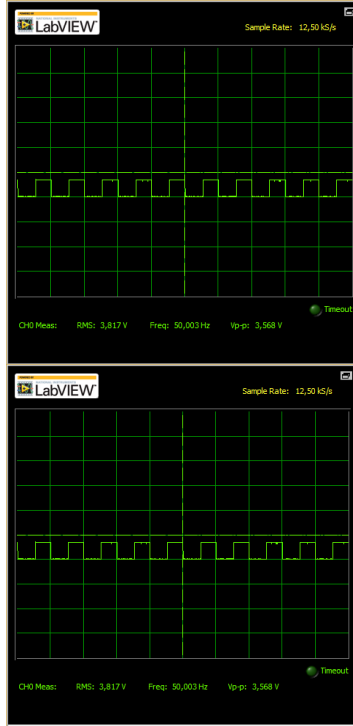
Kinematik denklemlere göre noktasal verilerin açılar hesaplanmıştır. $X^{(\theta_1, \theta_2, \theta_3)}$ şeklinde tanımlanmıştır. A ve B noktalarındayken motorların açısal değerleri A(0,0,0) ve B(-38,0,-28) dir. Kontrol sinyali motorun cinsine göre PWM olarak belirlenmiştir. Kullanılacak olan duty cycle değerleri belirlenmiştir.

$X_d(\text{Motor1}, \text{Motor2}, \text{Motor3})$

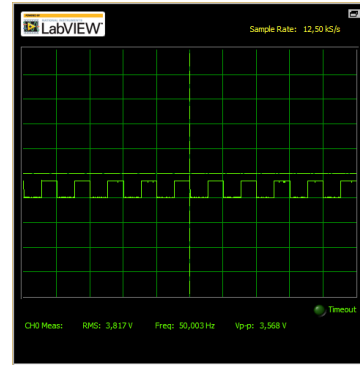
$A_d(0.47, 0.47, 0.47)$ $B_d(0.65, 0.47, 0.54)$

Bu değerler kontrolcüye çıkış olarak tanımlanmıştır ve aşağıdaki gibi osiloskoptan gözlemlenmiştir.

A noktasındayken motorlara gönderilen sinyal aşağıda gösterilmiştir ve duty cycle değerleri belirtilmiştir.

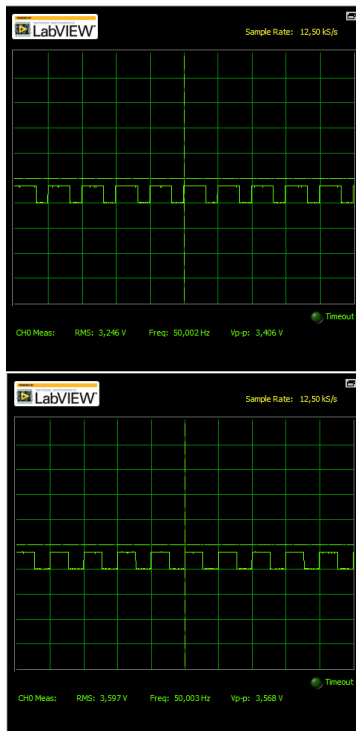


M1(0.47)
M3(0.47)

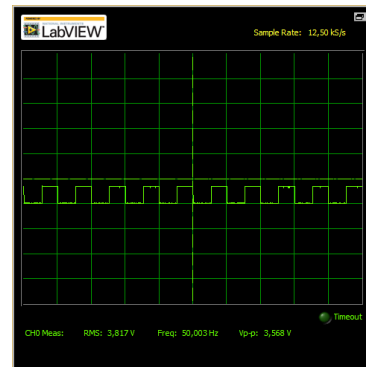


M2(0.47)

B noktasındayken motorlara gönderilen sinyal aşağıda gösterilmiştir ve duty cycle değerleri belirtilmiştir.



M1(0.65)
M3(0.54)



M2(0.47)

Deney 2:

Efektörün A(40,-40,300) noktasından B(-35,70,240) noktasına hareket ettirmek için gereken kontrol sinyallerini oluşturma.

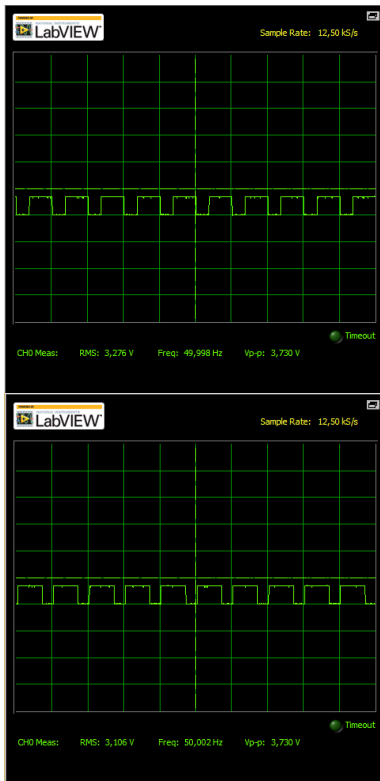
Kinematik denklemlere göre noktasal verilerin açıları hesaplanmıştır. $X(\theta_1, \theta_2, \theta_3)$ şeklinde tanımlanmıştır. A ve B noktalarındayken motorların açısal değerleri A(-37,-40,-59) ve B(-40,-18,3) dir. Kontrol sinyali motorun cinsine göre PWM olarak belirlenmiştir. Kullanılacak olan duty cycle değerleri belirlenmiştir.

$X_d(\text{Motor1}, \text{Motor2}, \text{Motor3})$

$A_d(0.64, 0.66, 0.69)$ $B_d(0.66, 0.55, 0.39)$

Bu değerler kontrolcüye çıkış olarak tanımlanmıştır ve aşağıdaki gibi osiloskoptan gözlemlenmiştir.

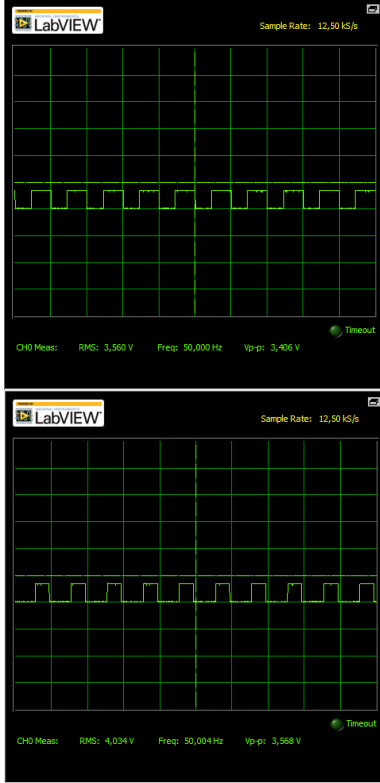
A noktasındayken motorlara gönderilen sinyal aşağıda gösterilmiştir ve duty cycle değerleri belirtilmiştir.



M1(0.64)
M3(0.69)

M2(0.66)

B noktasındayken motorlara gönderilen sinyal aşağıda gösterilmiştir ve duty cycle değerleri belirtilmiştir.



M1(0.66)

M2(0.55)

M3(0.39)

6. Araştırma ve Bulgular

Yapılan araştırmalar ve uygulamalar sonucu tutucu ucun belirlenen konuma geçebilmesi için yapılan hesaplamaların doğruluğunu göstermekle birlikte teorik olarak elde edilen verilerin uygulama sırasında ideal şartlar oluşturulamamasından (tasarımdaki uzuv uzunluklarındaki çok küçük hatalar, pleksiglasın esnekliği vb.) tutucu ucumuzun 1 cm ve daha az bir sapma ile belirlenen konuma ulaştığı gözlemlenmiştir. Bu sorun görüntü işlemede yapılan ofset ile halledilerek giderilmiştir. Ayrıca motor seçiminin plastik dişli servo motor olmasından kaynaklanan, yükü taşıyamama gibi bir sorunla karşılaşınca anlaşılmış ve gerekli düzenlemeler yapılarak bu sorun da halledilerek proje sonuçlandırılmıştır. Hız konusunda bir problem gözlemlenmemiştir. Bu parametreler ile de anlaşıldığı üzere proje istenilen düzeye getirilerek sonuca ulaşılmıştır.

7. Sonular

Görüntü işlemeli delta paralel robot tasarlanarak üretilmiştir ve sonuçları aşağıdaki gibi özetlenmiştir:

1-Delta Paralel Robot kontrolü Myrio adı verilen Labview tabanlı bir kontrolcü ile yapılarak farklı bir bakış açısı kazandırılmıştır.

2-Malzeme seçimindeki küçük hataların doğurduğu çalışma hatalarının giderilmesi konusunda yöntemler ortaya konulmuştur.

3-Yapılan uygulamalar sonucu matematiksel analizlerin ve program kurgusunun doğruluğu ispatlanmıştır.

Teşekkür

Yapılan çalışmada laboratuvar ve malzeme desteğini esirgemeyen Erzurum Teknik Üniversitesi Mühendislik ve Mimarlık Fakültesi Dekanlığı'na ve Elektrik Elektronik Mühendisliği Bölüm Başkanlığı'na teşekkür ederiz.

Kaynakça

- [1] Merlet J. M., 2006. *Introduction, Chapter 1, Parallel Robots*. Springer, Dordrecht Netherlands, 1-18.
- [2] Manar Lashin; Mohamed Fanni; Mahmuod Magdy; Abdelfatah M. Mohamed, “PD type of fuzzy controller for a new 3DOF fully decoupled translational manipulator”,2016.
- [3] F. Pierrot, C. Reynaud, and A. Fournier, “DELTA: a simple and efficient parallel robot,” *Robotica*, vol. 8, no. 2, pp. 105-109, 1990.
- [4] Ting-Ting Su; Long Cheng; Yun-Kuan Wang; Xu Liang; Xu Liang; Jun Zheng; Hao-Jian Zhang, “Time-Optimal Trajectory Planning for Delta Robot Based on Quintic Pythagorean-Hodograph Curves”,2018.
- [5] Ahmet Dumlu; Koksal Erenturk, “Trajectory Tracking Control for a 3-DOF Parallel Manipulator Using Fractional-Order $h_{box}P(l)_{lambda}d_{hbox}D_{mu}$ Control”, 2014.