

**T.C.**  
**PAMUKKALE ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**  
**ELEKTRİK-ELEKTRONİK**  
**MÜHENDİSLİĞİ ANABİLİM DALI**

**İNSANSIZ HAVA ARAÇLARINDA DENGE SİSTEM**  
**KONTROLÜNÜN BULANIK MANTIKLI FPGA VE GÖMÜLÜ**  
**SİSTEMLER İLE TASARIMI VE UYGULAMASI**

**YÜKSEK LİSANS TEZİ**

**MUSTAFA MANİSALI**

**DENİZLİ, ARALIK - 2017**

**T.C.  
PAMUKKALE ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
ELEKTRİK-ELEKTRONİK  
MÜHENDİSLİĞİ ANABİLİM DALI**



**İNSANSIZ HAVA ARAÇLARINDA DENGE SİSTEM  
KONTROLÜNÜN BULANIK MANTIKLI FPGA VE GÖMÜLÜ  
SİSTEMLER İLE TASARIMI VE UYGULAMASI**

**YÜKSEK LİSANS TEZİ**

**MUSTAFA MANİSALI**

**DENİZLİ, ARALIK - 2017**

## KABUL VE ONAY SAYFASI

**Mustafa MANİSALI** tarafından hazırlanan “İNSANSIZ HAVA ARAÇLARINDA DENGE SİSTEM KONTROLÜNÜN BULANIK MANTIKLI FPGA VE GÖMÜLÜ SİSTEMLER İLE TASARIMI VE UYGULAMASI” adlı tez çalışmasının savunma sınavı 01.12.2017 tarihinde yapılmış olup aşağıda verilen jüri tarafından oy birliği / ~~oy çokluğu~~ ile Pamukkale Üniversitesi Fen Bilimleri Enstitüsü Elektrik-Elektronik Mühendisliği Anabilim Dalı Yüksek Lisans Tezi olarak kabul edilmiştir.

Jüri Üyeleri

İmza

Danışman  
Yrd. Doç. Dr. Ahmet ÖZEK  
Pamukkale Üniversitesi

Üye  
Prof.Dr. Ceyhun KARPUZ  
Pamukkale Üniversitesi

Üye  
Yrd.Doç.Dr. Ali Kürşad GÖRÜR  
Nevşehir Hacı Bektaş Veli Üniversitesi

Pamukkale Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun ~~01.12.2017~~ tarih ve ~~51/21~~.... sayılı kararıyla onaylanmıştır.

Prof. Dr. Uğur YÜCEL

Fen Bilimleri Enstitüsü Müdürü

**Bu tez çalışması Pamukkale Üniversitesi Bilimsel Araştırma Projeleri Koordinasyon Birimi tarafından 2016FEBE004 nolu proje ile desteklenmiştir.**

**Bu tezin tasarımı, hazırlanması, yürütülmesi, arařtırmalarının yapılması ve bulgularının analizlerinde bilimsel etięe ve akademik kurallara özenle riayet edildiđini; bu alıřmanın doğrudan birincil ürünü olmayan bulguların, verilerin ve materyallerin bilimsel etięe uygun olarak kaynak gösterildiđini ve alıntı yapılan alıřmalara atfedildiđine beyan ederim.**

**Mustafa MANİSALI**



## ÖZET

**İNSANSIZ HAVA ARAÇLARINDA DENGE SİSTEM KONTROLÜNÜN  
BULANIK MANTIKLI FPGA VE GÖMÜLÜ SİSTEMLER İLE  
TASARIMI VE UYGULAMASI  
YÜKSEK LİSANS TEZİ  
MUSTAFA MANİSALİ  
PAMUKKALE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ  
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI**

**(TEZ DANIŞMANI:YRD. DOÇ DR. AHMET ÖZEK)**

**DENİZLİ, ARALIK - 2017**

Sensör ve kontrol sistemlerinin gelişmesine paralel olarak son yıllarda insansız hava araçlarının kullanımı da hızla artmaktadır. Dört rotorlu insansız hava araçları (Quadrotor) yapıları gereği lineer olmayan davranışa sahip hava araçlarıdır. Yapılan çalışmada, dört motorlu insansız hava aracının bulanık mantık kullanılarak kontrol algoritması oluşturulmuş, bu algoritma National Instruments'in tek kartlı bilgisayarı olan myRIO alan programlamalı kapı dizileri (FPGA) ve gerçek zamanlı işlemcisi üzerinde Labview kullanılarak programlanmış ve test edilmiştir.

**ANAHTAR KELİMELELER:**Bulanık mantık kontrolörü, dört rotorlu insansız hava aracı, myRIO, FPGA

## **ABSTRACT**

### **DESIGN AND IMPLEMENTATION OF STABILIZATION SYSTEM ON UNMANNED AERIAL VEHICLES WITH FUZZY CONTROLLED FPGA AND EMBEDDED SYSTEMS**

**MSC THESIS**

**MUSTAFA MANİSALI**

**PAMUKKALE UNIVERSITY INSTITUTE OF SCIENCE  
ELECTRICAL AND ELECTRONICS ENGINEERING**

**(SUPERVISOR:ASST. PROF. DR. AHMET ÖZEK)**

**DENİZLİ, DECEMBER 2017**

The use of the unmanned aerial vehicles are rapidly increasing in recent years due to improvements at sensor's and control systems. Quadrotors are air vehicles with nonlinear behavior due to their construction. In this study, a control algorithm was developed using fuzzy logic and this algorithm was implemented on National Instruments's single board computer myRIO field programmeble gate arrays (FPGA) and Real Time target processor using Labview for four motor quadrotor aerial vehicle.

**KEYWORDS:**Fuzzy Logic Controller, Quadrotor, MyRIO, FPGA

# İÇİNDEKİLER

Sayfa

ÖZET.....	i
ABSTRACT .....	ii
İÇİNDEKİLER .....	iii
ŞEKİL LİSTESİ .....	iv
TABLO LİSTESİ .....	vi
SEMBOL LİSTESİ .....	vii
ÖNSÖZ.....	viii
<b>1. GİRİŞ.....</b>	<b>1</b>
<b>2. DÖRT ROTORLU İNSANSIZ HAVA ARACI TEMELLERİ .....</b>	<b>3</b>
2.1    Literatür Özeti .....	4
2.2    Koordinat Sistemi .....	8
2.3    Rotasyon Matrisi .....	10
2.4    Yön Kosinüs Matrisi .....	11
<b>3. DÖRT ROTORLU İNSANSIZ HAVA ARACI DONANIMI.....</b>	<b>14</b>
3.1    myRIO Kontrol Kartı .....	15
3.2    MPU-6050 İvme Açık Sensörü .....	22
3.2.1    Sensör füzyon .....	23
3.3    Arduino Nano .....	25
3.4    Fırçasız DC Motorlar.....	26
3.5    Elektronik Hız Kontrol Devresi .....	26
3.6    Batarya.....	27
3.7    Dört Rotorlu Hava Aracı Gövdesi.....	27
<b>4. SİSTEM TASARIMI ve PROGRAMLANMASI.....</b>	<b>29</b>
4.1    LabVIEW Geliştirme Ortamı .....	29
4.2    Kontrolör Tasarımı .....	31
4.3    Sistem Tasarımı .....	38
4.3.1    Labview Ortamı ve Programlama.....	41
<b>5. SONUÇ VE ÖNERİLER .....</b>	<b>63</b>
<b>6. KAYNAKLAR.....</b>	<b>67</b>
<b>7. EKLER.....</b>	<b>70</b>
EK A	70
EK B	72
EK C	74
EK D	75
EK E	76
EK F	77
<b>ÖZGEÇMİŞ.....</b>	<b>78</b>



# ŞEKİL LİSTESİ

## Sayfa

Şekil 2.1: Dört rotorlu hava aracı "+", "x", "H" konfigürasyonları ve pervane dönüş yönleri .....	3
Şekil 2.2: Dört rotorlu insansız hava aracı gövde koordinat sistemi ve referans koordinat sistemi .....	9
Şekil 2.3: Dört rotorlu hava aracı koordinat çerçevesinde Euler açılarının gösterimi.....	10
Şekil 2.4: Başlangıç durumu koordinat sistemi.....	12
Şekil 2.5: Xn eksenini etrafında +90 derece döndürülmesi .....	13
Şekil 2.6: Zn Eksenini etrafında -30 derece döndürülmesi .....	13
Şekil 3.1: Kendi kablosuz ağını oluşturan myRIO kontrol sistemi.....	15
Şekil 3.2: Harici kablosuz ağına bağlanan myRIO kontrol sistemi .....	15
Şekil 3.3: myRIO donanım blok diyagramı .....	19
Şekil 3.4: MXP konektörü sinyal bağlantıları.....	20
Şekil 3.5: MSP C konektörü sinyalleri.....	21
Şekil 3.6: MPU-6050 ivme açısı ölçer prototip kartı.....	22
Şekil 3.7: MPU-6050 ivme açısı sensörü bağlantı şeması.....	23
Şekil 3.8: Arduino Nano gömülü sistem geliştirme kartı.....	25
Şekil 3.9: Emax Xa2212 1400kV outrunner fırçasız DC motor .....	26
Şekil 4.1: LabVIEW geliştirme ortamı; proje gezgini, blok diyagramı ve kullanıcı ön paneli .....	30
Şekil 4.2: Dört rotorlu insansız hava aracı bulanık mantık kontrol sistemi .....	33
Şekil 4.3: Hata değişkeni üyelik fonksiyonları .....	34
Şekil 4.4: Hatadaki değişim üyelik fonksiyonları .....	34
Şekil 4.5: PWM çıkış değerleri üyelik fonksiyonu .....	35
Şekil 4.6: "Fuzzy System Designer" kural tablosu oluşturma penceresi .....	36
Şekil 4.7: Bulanık mantık kontrolör giriş değerleri test sistemi .....	37
Şekil 4.8: Bulanık mantık kontrolörü "e" ve "de/dt" giriş parametreleri ile PWM%OUT çıkış parametresi ilişkisi.....	38
Şekil 4.9: Dört rotorlu hava aracı donanım tasarımı blok diyagramı.....	39
Şekil 4.10: Dört rotorlu hava aracı şematik diyagramı .....	40
Şekil 4.11: myRIO kontrollü dört rotorlu insansız hava aracımız .....	41
Şekil 4.12: Measurements & Automation Explorer penceresi açılış yolu .....	43
Şekil 4.13: myRIO kablosuz bağlantı ayarları yapılması .....	44
Şekil 4.14: Labview projesine yeni donanım ekleme .....	45
Şekil 4.15: Ağ üzerinde paylaşımlı değişken tanımlama .....	46
Şekil 4.16: Kontrol programı için tanımlanan ağ paylaşımlı değişkenler.....	47
Şekil 4.17: Bilgisayar kontrol paneli blok diyagramı kontrol kolu ölü bölge tanımlamaları.....	48
Şekil 4.18: Dört rotorlu hava aracı yer kontrol bilgisayarı kullanıcı paneli arayüzü .....	50
Şekil 4.19: Dört rotorlu hava aracı kullanıcı paneli kumanda girdileri değişimi görüntüsü.....	51
Şekil 4.20: myRIO ana program akışı Arduino seri bağlantı ayarı blok parçası52	
Şekil 4.21: Dört rotorlu hava insansız hava aracı ana programı blok diyagramı53	
Şekil 4.22: myRIO dosyalama sistemi disk olarak açma.....	54

Şekil 4.23 : myRIO dosyalama sistemi .....	55
Şekil 4.24 : Labview bulanık mantık araçları erişimi .....	56
Şekil 4.25 : Bulanık mantık dosyasının myRIO üzerinde açan hızlı sanal ölçüm bloğu.....	57
Şekil 4.26: myRIO sensör verisini okuma alt bloğu .....	58
Şekil 4.27: “kontrol_kolu_komutlari_okuma.vi” referans komutları okunması, derece cinsine çevirme alt bloğu .....	58
Şekil 4.28 : Bulanık mantık kontrolörü alt blok diyagramı .....	60
Şekil 4.29: Motor_gorev_zamanı_hesaplama.vi alt blok diyagramı .....	62
Şekil 5.1: Dört rotorlu insansız hava aracı kontrolör sistemi yunuslama açısı 0 derece cevabı .....	63
Şekil 5.2: Dört rotorlu insansız hava aracı kontrolör sistemi yunuslama açısı -3 derece cevabı .....	64
Şekil 5.3: Dört rotorlu insansız hava aracı kontrolör sistemi yunuslama açısı +3 derece cevabı .....	64

## TABLO LİSTESİ

	<b><u>Sayfa</u></b>
Tablo 3.1: myRIO teknik özellikler .....	16
Tablo 3.2: MXP konektörü sinyal açıklamaları .....	20
Tablo 3.3: MSP konektörü sinyal açıklamaları.....	21
Tablo 4.4: Kural Tablosu .....	36
Tablo 4.5: myRIO-Arduino seri bağlantı parametreleri.....	52
Tablo 4.6: Kontrol kolu okuma alt bloğu değişken değer aralıkları .....	59

## SEMBOL LİSTESİ

<b>İHA</b>	:	İnsansız Hava Aracı
<b>WLAN</b>	:	Wireless Local Area Network
<b>LQR</b>	:	Linear Quadratic Regulator
<b>PID</b>	:	Proportional Integrative Derivative
<b>Bps</b>	:	Bits Per Second
<b>DOF</b>	:	Degree of Freedom
<b>myRIO</b>	:	My Reconfigurable Input Output
<b>AP SOC</b>	:	All Programmable System on chip
<b>FPGA</b>	:	Field Programmable gate array
<b>MEMS</b>	:	Micro Electromechanical Systems
<b>LabVIEW</b>	:	Laboratory Virtual Instrument Engineering Workbench
<b>UART</b>	:	Universal Asynchronous Receiver-Transmitter
<b>IP</b>	:	Internet Protocol
<b>VI</b>	:	Virtual Instrument
<b>HID</b>	:	Human Interface Device

## ÖNSÖZ

Tez çalışmam boyunca bilimsel katkıları ile bana destek olup eğitimim süresince yardımlarını esirgemeyen, tez danışmanım ve değerli hocam Yrd. Doç. Dr. Ahmet Özek'e, her şart altında maddi ve manevi desteklerini benden esirgemeyen aileme, her zaman beni destekleyen nişanlıma ve çalışmam için kaynak sağlayan üniversiteme teşekkürlerimi bir borç bilirim.

# 1. GİRİŞ

Günümüzde insansız hava araçları (İHA) keşif, askeri savunma – saldırı, haritalama, arama – kurtarma gibi birçok alanda kullanılmaktadır. İnsansız hava araçlarının sabit kanatlı ve döner kanatlı tipleri mevcuttur. Bu çalışmada dört rotora sahip döner kanatlı (Quadrotor) insansız hava aracı üzerinde National Instruments'in sunduğu geliştirme donanımı olan MyRIO kullanılarak bulanık mantık kontrol sistemi tasarlanmış ve donanım üzerinde çalıştırılmıştır.

Donanımın programlanması LabVIEW (Laboratory Virtual Instrument Engineering Workbench) ile görsel programlama dili kullanılarak gerçekleştirilmiştir. Labview programı veri izleme, cihaz kontrolü, test otomasyonu, veri analizi ve sinyal işleme, endüstriyel kontrol ve gömülü sistem tasarımı gibi uygulamalarda kullanılmaktadır.

Yapılan pratik çalışma sonucunda yer kontrol bilgisayarı ile IEEE 802.11 WLAN (Wireless Local Area Network) ağı üzerinden haberleşen, komutları yer bilgisayarına bağlı USB kontrol kolu (Joystick) ile alan, dört rotorlu insansız hava aracı yönelim ve açılarını da yer kontrol bilgisayarı üzerinde izlenebilen grafiksel ara yüz tasarımı yapılmıştır. Çalışmanın test edilebilmesi için hava aracı gövdesinin tek eksen üzerinde belirli sınır açıları dahilinde hareket edebileceği test standı tasarlanmıştır. Bu tasarım parçaları 3D yazıcı kullanılarak basılmış ve montajlanmıştır. Bu test standı ve hava aracı gövdesi üniversite bünyesindeki daha ileriki çalışmalar için kullanılabilir yapıdadır.

Bu tez çalışmasının ikinci bölümünde dört rotorlu insansız hava aracı hakkında genel giriş bilgileri verildikten sonra literatür bilgisi, daha sonrasında ise hava aracının kinematiği ve dinamiği hakkında bilgiler verilmiştir.

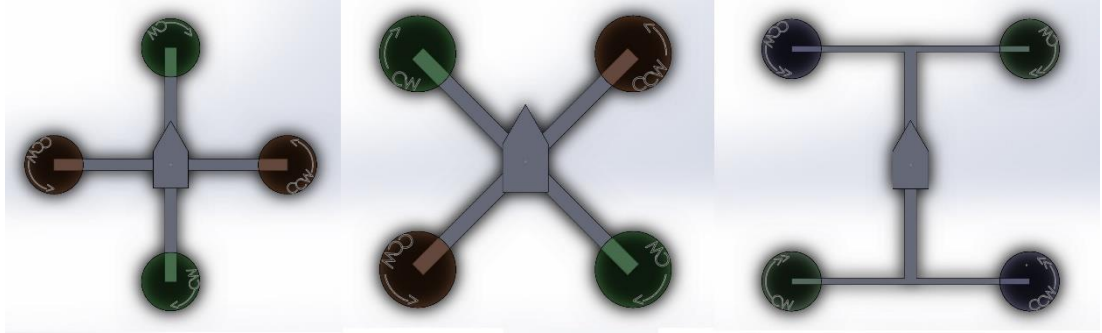
Üçüncü bölümde tasarımını yaptığımız sistemin yapısı ve üzerindeki donanımlar hakkında bilgiler verilmiştir.

Dördüncü bölümde ise, bulanık mantık kontrol sisteminin Labview kullanılarak tasarlanması, bulanık mantık kontrolcüsünün tasarlanması ve MyRIO donanımı üzerinde çalıştırılması anlatılmıştır.

Ve son bölüm olan beşinci bölümde ise çalışma sonunda elde edilen sonuçlar genel olarak değerlendirilmiş, ileride yapılabilecek çalışmalar ve iyileştirmeler için önerilerde bulunulmuştur.

## 2. DÖRT ROTORLU İNSANSIZ HAVA ARACI TEMELLERİ

Dört rotorlu insansız hava aracı; dört adet pervaneye sahip, bu pervanelerin oluşturduğu itme kuvvetiyle dikey iniş kalkış yapabilen hava araçlarıdır. Dört rotorlu hava araçları döner kanatlı insansız hava araçlarıdır. Genel olarak yapıları; uçlarında motorlar bulunan iki adet çubuğun artı "+" şeklinde ya da "x" harfi şeklinde merkezde sabitlenmiş haldedir. En çok kullanılan bu gövde tasarımlarından hariç "H" harfi şeklide olan tipleri de mevcuttur. Şekil 2.1 de dört rotora sahip insansız hava araçlarının tipik pervane diziliş konfigürasyonları görülmektedir. Bu konfigürasyonlar incelendiğinde rotor dönüş yönlerinin simetrik pervanelerde zıt olduğudur. Bu sayede rotorların dönüş hareketi ile oluşan zıt döndürme momentinin, gövdeyi; pervane dönüş yönlerinin aksi yönünde döndürmesinin önüne geçilmiş olur.



Şekil 2.1: Dört rotorlu hava aracı "+", "x", "H" konfigürasyonları ve pervane dönüş yönleri

Klasik helikopterlerde hareket, rotorun hücum açısını değiştiren karışık mekanik sistemler aracılığı ile sağlanır. Bu mekanik aksamlar karmaşık ve ağırdır. Dört rotorlu hava araçları, sabit rotorlu oluşları ve basit hareketli parçadan oluşmayan şasileri sayesinde yapıları basittir. Ayrıca tüm motor gücünün gövdeyi yukarı yönlü kaldırma için kullanılması sonucu boyutlarına kıyasla oldukça yüksek taşıma kapasitesine sahiptirler.

Dört rotorlu hava araçları klasik helikopterlere göre basit oluşları ve kolay kurulumları olmasına rağmen bazı dezavantajları vardır. Dinamiklerinin doğrusal



olmayışları nedeniyle yön kontrolü ve kararlılıđının sađlanması oldukça güçtür (Tosun 2015).

## 2.1 Literatür Özeti

Dört rotorlu hava araçlarının dikey iniş kalkış yapabilmesi ve basit yapıları nedeniyle birçok uygulamada kullanılmaktadır. Ancak sahip olduđu doğrusal olmayan dinamikleri nedeniyle de birçok pozisyon ve kontrol arařtırmalarına konu olmaktadır. Günümüzde ticari ya da hobi amaçlı birçok dört rotorlu hava araçlarının kontrol sisteminde klasik kontrol metodu olan PID kontrolör karşımıza çıkmaktadır. Dört rotorlu hava araçlarının diđer özel kullanım alanlarından olan askeri uygulamalar, gözetleme, ya da spesifik ticari uygulamalarda PID kontrolörden farklı daha doğru, daha hızlı tepki veren kontrol sistemleri kullanılmaktadır.

Erginer ve arkadaşları (2012) çalışmasında, dört rotorlu dikey iniş kalkış yapabilen insansız hava aracının kontrolünü bulanık mantık ve PD (oransal türevsel) kontrolör kullanarak gerçekleřtirmişlerdir. Yaptıkları çalışmada, Matlab Simulink, gerçekleřtirdikleri kontrol sistemlerinin testi, analizi ve karşılaştırılmasında kullanılmıştır. Karşılařtıkları test sonuçlarını denemek amacıyla gerçek donanım üzerinde testler gerçekleřtirmişlerdir. Bu testlerin yapılabilmesi amacıyla dört rotorlu hava aracına yunuslama ve yuvarlanma açılarında serbestlik sađlayan test standı tasarlanmış ve üretilmiştir. Çalışma sonucunda klasik PD kontrolör ve bulanık mantıklı hibrit PD kontrolörünün ikisinin de sistemi kontrol edebildiđi, ancak bulanık mantıklı hibrit PD kontrolörünün, sistemi, klasik PD kontrolörden biraz daha iyi kontrol ettiđi görülmüştür. Bulanık mantıklı hibrit PD kontrolörün, klasik PD kontrolöre göre, daha iyi dış etkilere karşı daha gürbüz olduđu ve kontrolörün tasarımının daha kolay olduđu gibi avantajları olduđu gösterilmiştir.

Tosun (2015) çalışmasında, Qball-X4 isimli dört rotorlu hava aracının konum ve açı kontrolünü, dinamiđi belli olmayan sistemler için kullanılan adaptif kontrol, dinamiđi belirli olan sistemlerde kullanılan LQR (Dođrusal Karesel Düzenleyici) kontrol ve klasik kontrol yöntemlerinden PID kontrol sistemleri kullanarak Matlab/Simulink benzetim programında karşılařtırmalı olarak denemiřtir. Kullanılan adaptif kontrol sistemi, gerçek sistem üzerinde de denenmiştir.

Baerveldt ve arkadaşları (1997), yaptığı çalışmasında otonom uçuş yapan helikopter sistemi için düşük ağırlıklı ve düşük maliyetli tutum tahmini yapan sistem geliştirmişleridir. Bu sistem temel olarak bir ivme-açı ölçer ve bir de gyro sensöründen oluşmaktadır. Bu iki sensörden gelen veriler, tamamlayıcı filtre ile sensör füzyon yapılarak daha doğru tutum verilerinin hesaplanması sağlanmıştır. Yapılan testlerde, öne sürülen tutum tahmin algoritmasının, sistemin gerçek tutumu ile arasındaki farkların ihmal edilebilecek kadar küçük olduğu görülmüştür.

Raza (2010) çalışmasında insansız dört rotorlu hava aracının tasarımını gerçekleştirmiş, kontrolünü ise Mamdani ve Sugeno tipi bulanık mantık kontrolörü kullanarak gerçekleştirmiştir. Matlab üzerinde gerçekleştirdiği simülasyon sonuçlarında Mamdani tipi bulanık mantık kontrolcüsünün daha hızlı tepki verdiği, bunun yanında Sugeno tipi bulanık mantık kontrolcüsünün ise rüzgâr ve titreşim gibi dış etkiler altında stabilizasyon konusunda daha iyi sonuç verdiği görülmüştür. Hazırlanan test öne sürülen bulanık mantık kontrolcüsü test edilmiş, performansı geleneksel kontrol algoritmalarıyla karşılaştırılmıştır. Sonuç olarak öne sürülen bulanık mantık kontrolcüsünün çevresel etkilerden etkilenmeden gürbüz uçuş yapabilmeye olanak verecek performansta olduğu gözlenmiştir.

Nicol (2008) yaptığı çalışmasında dört rotorlu hava aracının denge sistemi için yeni bir tür adaptif yapay sinir ağları kontrol metodu öne sürmüştür. Bu yeni metod, simülasyonlar ile hem hata modifikasyonu (e-modification), hem de ölü bölge adaptif kontrol teknikleri ile karşılaştırılmıştır. Çıkan sonuçlarda öne sürülen yeni tekniğin aracın tutumunda istenilen açılardan daha az hatalar olduğu ve ağırlık kaymasının (weight drift) da azaltıldığı gösterilmiştir.

Nicol (2010) yaptığı çalışmasında dört rotorlu hava araçlarının davranış ve yüksekliğini kontrol eden, değişken çevre etkilerine karşı sistem parametrelerini adapte edebilen yalpa ve yunus açılarını / hızlarını kontrol eden adaptif yapay sinir ağları içeren kontrol sistemi tasarımı yapmıştır. Lineer kontrolörlerin (PID ve varyasyonları ile LQR, vb.) ve bazı lineer olmayan kontrol sistemlerinin (geri adımlamalı kontrol, kayan kipli kontrol ve adaptif kontrol gibi) performansları, sistemin doğru tanımlanmış dinamik modeline dayalıdır. Lineer ve modele dayalı lineer olmayan kontrolörler aracın dinamik parametrelerine ve değişen çevre koşullarına göre iyi şekilde tanımlanmalıdır. Bu sebeple, eğer araç üzerinde değişen

çevre etkileri ya da yükte değişimler var ise sistem kontrolü beklenildiği gibi olmaya bilir ya da sistem kontrolden çıkabilir. Nicol (2010) yaptığı çalışmada adaptif yapay sinir ağı kullanarak gerçekleştirdiği kontrol sisteminin birçok durum altında geleneksel PID kontrolör ile eş ya da iyi performans verdiğini göstermiştir.

Abeywardena ve arkadaşlarının (2009) yapmış olduğu çalışmada, bulanık mantık kullanılarak dört rotorlu hava aracının denge kontrolü gerçekleştirilmiştir. Yapılan kontrol sistemi ve sistemin lineer olmayan dinamik modeli ile birlikte benzetimi yapılmıştır. Sonuç olarak, gerçekleştirilen kontrol sisteminin, insansız hava aracının havada sabit kalabilmesini sağlayabileceği gösterilmiştir.

Mehranpour ve arkadaşları (2013) yaptığı çalışmada klasik PID kontrolörün kazanç parametrelerini bulanık mantık ile sistemin cevabına göre adapte eden, bulanık mantık adaptif PID kontrolör gerçekleştirmişlerdir. Yapılan sistemde PID kontrolörü dört rotorlu hava aracı üzerindeki kontrol kartında çalışırken, bulanık mantık kontrol sistemi ise yer bilgisayarı üzerinde çalıştırılmıştır. Dört rotorlu hava aracı yer bilgisayarı ile haberleştiği her anda aracın o anki durumu ile ilgili yalpa yunus ve sapma açıları ile sistem girdi verileri yer bilgisayarına gönderilir. Daha sonra yer bilgisayarındaki bulanık mantık kontrolörü gerekli olan PID kontrolör katsayılarını hesaplayıp hava aracına geri gönderir. Sistem, yer bilgisayarı ile hava aracı arasındaki haberleşmeyi Xbee kablosuz cihazı ile gerçekleştirmiştir. Hava aracı PID hesaplama döngüsü 30  $\mu$ s ve yer bilgisayarına veri iletim hızı 57600 Bps olarak ayarlanmıştır. Öne sürülen sistemde değişin koşullar altında bulanık mantık kontrolörünün PID katsayılarını değiştirdiği, yeni PID katsayıları ile sistemin PID kontrolü sağlandığı ve sistemin kararlı hale gelmesiyle beraber ilk durumdaki PID katsayılarını tekrar kullandığı görülmüştür. Yapılan çalışmada Labview ortamının kullanımının kolaylığı ve hesaplama hızının yüksek oluşu nedeniyle Mehranpour ve arkadaşları tarafından önerilen bir program olmuştur.

Fakurian ve arkadaşları (2014) bulanık mantık kullanarak minimum kontrol girdileri kullanarak dört rotorlu hava aracını kontrol etmeye çalışmışlardır. Öne sürdükleri sistemde hava aracından alınan sensör verileri ile sisteme verdikleri girdiler arasındaki hata değeri ile hatanın zamana bağlı türevini bulanık mantık sistemi girdileri olarak kullanmışlardır. Öne sürülen kontrolör sisteminde yalpa, yunus, sapma ve yükseklik kontrolü Mamdani tipindeki dört adet bağımsız bulanık

mantık kontrolörü ile sağlanmıştır. Hazırlanan kontrolör sisteminin kontrol algoritmasını basitleştirdiği, ani değişen etkilere karşı hızlı tepki verdiği ve ani manevralar yapabildiği simülasyon ve animasyon üzerinde gözlenmiştir.

Castillo ve arkadaşları (2014), yaptıkları çalışmada dört rotorlu mini hava aracının denge kontrolü ve otonom uçuşu üzerinde çalışmışlardır. Yapılan çalışmada, hava aracının dinamik modeli Lagrange yöntemi ile çıkarılmıştır. Daha sonra, kontrol sisteminin tasarımı ise Lyapunov analizi ve yuvalanmış doygunluk algoritması (nested saturation algorithm) kullanılarak gerçekleştirilmiştir. Sonuç olarak dört rotorlu minik hava aracı, havalanma, havada sabit kalabilme ve inişi otonom olarak başarıyla gerçekleştirebilmiştir.

Pedro ve arkadaşları (2015), yaptıkları çalışmada Mamdani tipi bulanık mantık kontrolörü kullanarak, dış etkilere karşı gürbüz, değişken iç ve dış etkilere karşılık koyabilen akıllı kontrol sistemi tasarımı geliştirmişlerdir. Geliştirdikleri kontrol algoritması benzetim ile test edilmiştir. Test edilen sistemin gürbüz, değişken koşullara uyum sağlayabilecek esneklikte, güvenilir ve etkili olduğu görülmüştür.

Tzafestas ve arkadaşları (1990), yaptıkları çalışmada bulanık mantık kullanarak akıllı PID kontrol sistemi oluşturmuşlardır. Yaptıkları çalışmada PID kontrol sistemi parametreleri klasik Ziegler-Nichols, Kalman, metodları ile belirlenmiştir. PID kontrol sisteminin bu parametreleri sistemin çalışması sırasında bulanık mantık kontrolörü ile değiştirilerek sistemin, değişen koşullara göre hem durağan hem de değişen durum performansının iyileştirilmesi sağlanmıştır. Sistemdeki bulanık mantık kontrolörünün oluşturduğu bu ekstra hesaplama işlemleri, sistemde yardımcı mikroişlemci veya mikro kontrolör kullanılarak giderilmiştir.

Sardar ve arkadaşları (2014) yaptıkları çalışmada, akıllı kazanç artırımı PD kontrol tekniği kullanılarak dört rotorlu insansız hava aracının kontrolü otonom olarak gerçekleştirilmiştir. Yaptıkları çalışmada dört rotorlu hava aracının otonom tutum kontrolü ve yön takibi üzerinde çalışılmıştır. Çalışmada Takagi-Sugeno tipi bulanık mantık kullanılarak pozisyon ve sapma açısı kontrolü ile klasik PD kontrol algoritması benzetim ortamı üzerinde karşılaştırılarak test edilmiştir. Sonuç olarak öne sürülen kontrol sisteminin, klasik sabit kazançta sahip PD kontrol sisteminden; dış etkilere ve değişen koşullara karşı daha etkili olduğu gözlenmiştir.

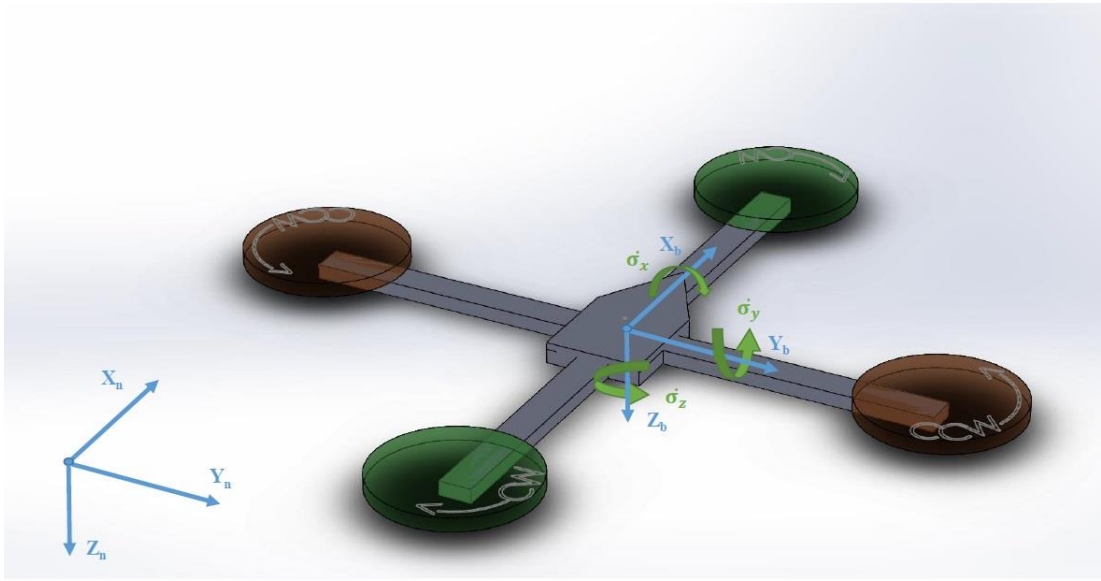
## 2.2 Koordinat Sistemi

Dört rotorlu hava aracının uçuş dinamiklerini ve sistemin kinematikliğini anlamak için çıkaracağımız denklemlerden önce hava aracı gövde koordinat sistemi ve başlangıç referansı olarak alacağımız yer düzlemini anlamamız gereklidir. Bu iki uzay koordinatları üzerinde dört rotorlu hava aracının kinematik ve dinamik denklemleri oturtulacaktır.

İki farklı koordinat sistemi kullanılmasındaki amaç dört rotorlu hava aracının 6 derece serbestlikteki hareketlerini (DOF) daha iyi tanımlayabilmektir. Eksenlerdeki bu serbestlik derecesi 3 boyutlu koordinat düzleminde x, y ve z eksenleri üzerindeki doğrusal hareket ve bu eksenlerde oluşan dönme hareketleridir. 3 boyutlu bir düzlemde tüm eksenler üzerinde özgürce hareket edebilen bir cisim 6 derece serbestliğe sahiptir. Bu hareketler; x ekseninde doğrusal hareket, x ekseninde çevresindeki dönme hareketi, y ekseninde doğrusal hareket, y ekseninde etrafındaki dönme hareketi, z ekseninde doğrusal hareket ve z ekseninde üzerindeki dönme hareketidir. 4 adet rotora sahip hava aracı 3 eksenli uzayda hareket edebilmesine rağmen motorlarının gövde üzerinde sağlayabileceği hareketler 4 derece serbestlikle sınırlıdır. Hava aracının ön-arka tarafının x ekseninde çakışık kuzeye bakar şekilde olduğu ve sağ-sol taraflarının da y ekseninde çakışık olduğu varsayıldığında 4 derece serbestlik şu şekilde tanımlanabilir. Bu hareketler; z ekseninde doğrusal hareket olan hava aracının yükselme-alçalma hareketi, z ekseninde dönme hareketi olan sapma yani “yaw” hareketi, x ekseninde dönme hareketi olan yalpa yani “roll” hareketi ve y ekseninde dönme hareketi olan yunuslama yani “pitch” hareketidir.

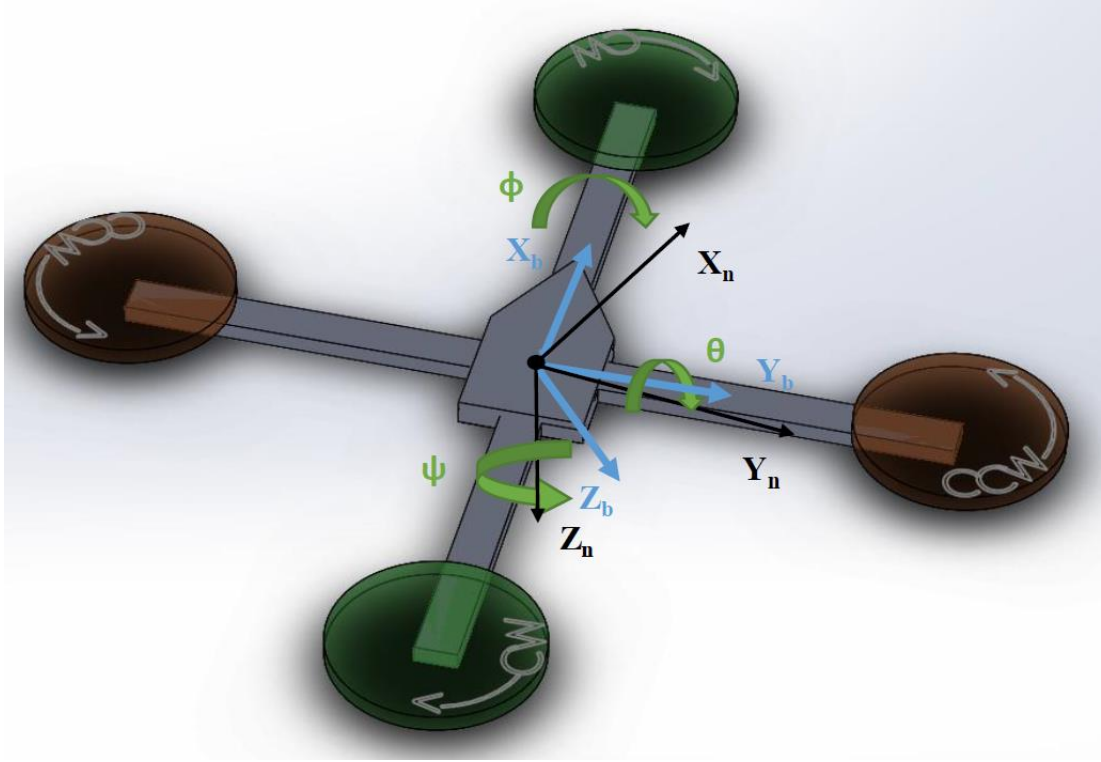
Dört rotorlu insansız hava aracının 3 boyutlu uzay üzerindeki hareketlerini tanımlarken 2 farklı koordinat sistemi tanımlamamız gereklidir. Bu koordinat sistemlerinden ilki hava aracının motorları tarafından hareket ettirilebilen gövde koordinat sistemidir. “b” alt indeksi ile gösterdiğimiz bu koordinat sistemi hava aracının gövdesine çakışık halde, z+ yönü yerçekimi vektörüne paralel şekilde tanımlanmıştır. Dört rotorlu hava aracının tutumunu incelemek için kullandığımız diğer koordinat sistemi ise “n” alt indeksi ile tanımladığımız navigasyon koordinat sistemidir. Gövde koordinat sistemi hava aracı gövdesi ile beraber hareket ederken,

navigasyon koordinat sistemi hava aracı harekete başladığı zaman başlangıç noktasında sabitlenmelidir. Bu navigasyon koordinat sistemi yeryüzü koordinat sistemi olarak da adlandırılabilir. Bu koordinat sistemi tanımlamasında yeryüzünün eğimi ihmal edilerek tanımlanır. Dört rotorlu hava aracı hareket mesafesi kısıtlı olduğu için yeryüzünün eğimi bulunduğu konum için ihmal edilebilir. Hava aracının gövde eksenleri üzerindeki dönme hareketleri  $\dot{\sigma}$  ve  $\sigma$  eksenin ismi alt indeks olacak şekilde verilmiştir. Şekil 2.2 de gövde koordinat sistemi ve navigasyon koordinat sistemi tanımlamaları gösterilmiştir (Magnussen ve ark 2011).



**Şekil 2.2:** Dört rotorlu insansız hava aracı gövde koordinat sistemi ve referans koordinat sistemi

Dört rotorlu insansız hava aracının yönelimini ifade edebilmek için yer referans koordinat sistemi ile aracın gövde koordinat sistemi arasında açısal bağıntılar verilmesi gereklidir. Bu tanımlamalar havacılıkta sıklıkla kullanılan Euler açıları kullanılarak yapılır. Euler açıları;  $\phi$ ,  $\theta$  ve  $\psi$  ile gösterilen sırası ile yalpa yani “roll”, yunuslama yani “pitch” ve sapma yani “yaw” açıları olarak bilinir. Şekil 2.3 de Euler açıları dört rotorlu hava aracı üzerinde gösterilmiştir.



Şekil 2.3: Dört rotorlu hava aracı koordinat çerçevesinde Euler açılarının gösterimi

### 2.3 Rotasyon Matrisi

Dört rotorlu insansız hava aracının dinamik modelinin oluşturulabilmesi için araca etkiyen kuvvetler ve momentler bulunmalıdır. Euler açıları kullanılarak x, y ve z eksenlerindeki dönüşleri ayrı ayrı matrislerde tanımlanabilir. Denklem (2.1), (2.2) ve (2.3)'de üç eksendeki dönüş matrisleri tanımlaması ayrı ayrı gösterilmiştir.

$$R_{\sigma_x}(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix} \quad (2.1)$$

$$R_{\sigma_y}(\theta) = \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \quad (2.2)$$

$$R_{\sigma_z}(\psi) = \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.3)$$

Gövde üzerinde oluşan toplam dönme momentlerini gövde koordinat sisteminden sabit yer koordinat sistemine çeviren rotasyon matrisi bu üç eksendeki dönme matrislerinin çarpımıyla elde edilir. Çarpım sonucunda oluşan toplam rotasyon matrisi denklem (2.4) de verilmiştir.

$$C_b^n = R_{\sigma_x}(\phi)R_{\sigma_y}(\theta)R_{\sigma_z}(\psi)$$

$$C_b^n = \begin{bmatrix} \cos \theta \cos \psi & \cos \theta \sin \psi & -\sin \theta \\ \sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi & \cos \phi \cos \psi + \sin \phi \sin \theta \sin \psi & \sin \phi \cos \theta \\ \sin \phi \sin \psi + \cos \phi \sin \theta \cos \psi & \cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi & \cos \phi \cos \theta \end{bmatrix} \quad (2.4)$$

Hesaplanan bu matris aracın gövde koordinat sisteminden navigasyon yer koordinat sistemine dönüşümü gerçekleştirir. Bazı durumlarda navigasyon koordinat sisteminden gövde koordinat sistemine dönüş gerekebilir. Bu durumda  $C_b^n$  matrisinin devriğini almamız yeterlidir. Denklem (2.5)'de  $C_b^n$  rotasyon matrisinin devriği verilmiştir.

$$C_n^b = \begin{bmatrix} \cos \theta \cos \psi & \sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi & \sin \phi \sin \psi + \cos \phi \sin \theta \cos \psi \\ \cos \theta \sin \psi & \cos \phi \cos \psi + \sin \phi \sin \theta \sin \psi & \cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi \\ -\sin \theta & \sin \phi \cos \theta & \cos \phi \cos \theta \end{bmatrix} \quad (2.5)$$

## 2.4 Yön Kosinüs Matrisi

Rotasyon matrisi iki referans çerçeve arasındaki dönel ilişkiyi belirler. Bu rotasyon matrisine aynı zamanda yönelim kosinüs matrisi adı da verilir. Yön kosinüs matrisi 3x3'lük dönüşüm matrisidir. Dört rotorlu hava aracının gövde koordinat çerçevesi ile referans yer koordinat çerçevesi arasındaki ilişkiyi tanımlar. Yön kosinüs matrisi elemanları “*i*” satır, “*j*” sütunları temsil edecek şekilde,  $r_{ij}$  elemanları yer referans çerçevesindeki “*i*”inci eksen ile gövde koordinat çerçevesinin “*j*”inci eksenini arasındaki açının kosinüsünü temsil eder. Yön kosinüs matrisinin bir özelliği, hava aracının gövde koordinat çerçevesi ile referans navigasyon çerçevesi eksenleri

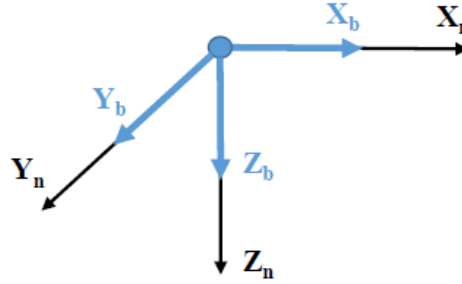


çakışık halde iken birim matris vermesidir. Denklem (2.6)'da yön kosinüs matrisinin tanımlaması verilmiştir.

$$C_b^n = \begin{bmatrix} r_{xx} & r_{xy} & r_{xz} \\ r_{yx} & r_{yy} & r_{yz} \\ r_{zx} & r_{zy} & r_{zz} \end{bmatrix} \quad (2.6)$$

Burada " $r_{xx}$ " referans yer koordinat çerçevesi x eksenine ile hava aracının gövde koordinat çerçevesi x eksenine arasındaki açının kosinüsünü tanımlar. Benzer şekilde " $r_{zy}$ " ise referans yer koordinat çerçevesi z eksenine ile gövde koordinat çerçevesi y eksenine arasındaki açının kosinüsü olarak tanımlanır. 3x3 matris üzerindeki diğer tüm elemanlar benzer şekilde tanımlanır. Magnussen (2011)

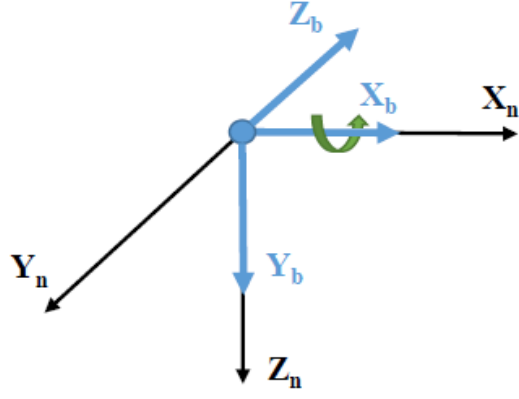
Örnek olması açısından Şekil 2.4'de dört rotorlu insansız hava aracının başlangıç durumunun koordinat sistemi örneği ve bu durumun yön kosinüs matrisi de denklem (2.7)'de görülmektedir.



Şekil 2.4: Başlangıç durumu koordinat sistemi

$$C_b^n = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.7)$$

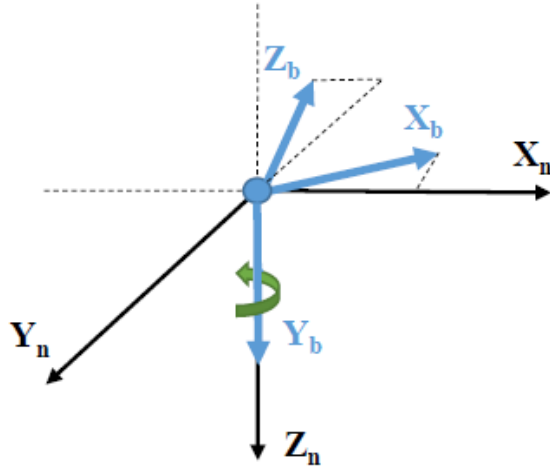
Dört rotorlu hava aracımızı başlangıç durumundan X eksenine etrafında +90 derece döndürdüğümüzü varsaydığımızda koordinat sistemimiz şekil 2.5'deki halini alacaktır. Bu durumdaki yön kosinüs matrisimiz denklem 2.8 de verilmiştir.



Şekil 2.5: Xn eksenini etrafında +90 derece döndürülmesi

$$C_b^n = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \quad (2.8)$$

Bir sonraki durumda hava aracı; Zn eksenini etrafında -30 derece döndürüldüğünü varsayarsak, koordinat çerçevemiz şekil 2.6'daki halini alır. Bu durumdaki yön kosinüs matrisi ise denklem 2.9'da verildiği gibidir.



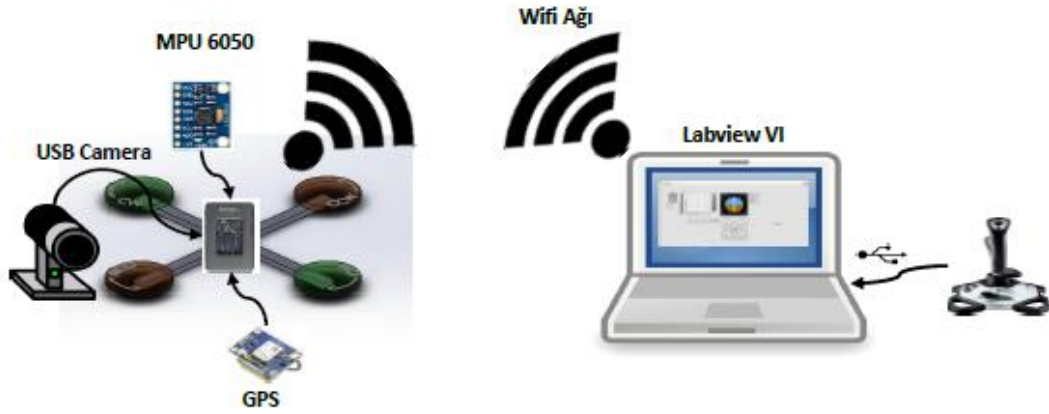
Şekil 2.6: Zn Eksenini etrafında -30 derece döndürülmesi

$$C_b^n = \begin{bmatrix} 0.866 & 0 & -0.5 \\ -0.5 & 0 & -0.866 \\ 0 & 1 & 0 \end{bmatrix} \quad (2.9)$$

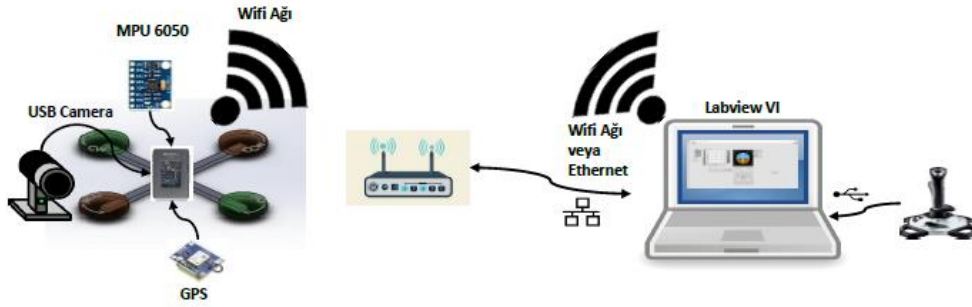
### 3. DÖRT ROTORLU İNSANSIZ HAVA ARACI DONANIMI

Bu bölümde hazırlanmış olduğumuz dört rotorlu insansız hava aracımızın donanımından ve yapısından bahsedilecektir. Bu çalışmada dört rotorlu hava aracının minimum uçuşu için gerekli olan gövde, motorlar, pervaneler, batarya, motor hız kontrol devreleri gibi elemanların seçiminde detaylı incelemelerde bulunulmamış, yapılan çalışmada daha çok kontrol sistemi hazırlaması ve kontrol kartının Labview ortamında programlanması üzerine yoğunlaşmıştır. Kullanılan sensörler, motorlar, motor hız kontrol devresi ve hava aracının şasisi hobi amaçlı kullanımlar için tasarlanmış, tak ve çalıştır tarzı hazır donanımlardan oluşmaktadır. Yapılan bu tez çalışmasındaki özgünlüğü; kullanılan myRIO geliştirme kartı, Labview üzerinde görsel programlama yapısı ve dört rotorlu hava aracının kontrol sistemi olan bulanık mantık kontrol sistemi oluşturmaktadır.

Hazırlanan dört rotorlu hava aracı uçuş sistemi temel olarak bir adet yer kontrol bilgisayarı ve hava aracından oluşmaktadır. Hava aracı ile yer kontrol bilgisayarı arasındaki haberleşme ve aracın uçuş parametrelerinin aktarımı WLAN ağı sayesinde olmaktadır. myRIO 1900 sahip olduğu WLAN modemi sayesinde ister kendi kablosuz ağını oluşturabilir, istenirse de mevcut kablosuz ağa bağlanabilir. Bu özelliği sayesinde dört rotorlu hava aracı uçuş sistemi istenirse kablosuz modeme sahip bir bilgisayarın direk myRIO'nun oluşturduğu kablosuz ağa bağlanmasıyla oluşturulabilir. Ya da daha uzak iletişim mesafesi için güçlü antenlere sahip bir kablosuz yönlendiricinin oluşturduğu ağa myRIO bağlanabilir. Bu yönlendiriciye yer kontrol bilgisayarı ister Ethernet kablosu ile, isterse de kablosuz olarak bağlanmasıyla haberleşme sağlanmış olur. Birden fazla kablosuz yönlendirici kullanılıp, örgü(mesh) ağ yapısı oluşturularak myRIO'nun kablosuz erişim mesafesi ciddi şekilde arttırılabilir. Veya birden fazla myRIO aynı haberleşme ağı üzerinde "ağ üzerinde paylaşılan değişkenler" kullanarak dataların bir myRIO'dan başka bir myRIO'ya kablosuz olarak aktarılması sağlanabilir. Şekil 3.1 ve şekil 3.2 de sırasıyla myRIO'nun oluşturduğu ağa bağlanan yer kontrol bilgisayarı ile harici kablosuz yönlendiriciye bağlanan myRIO'nun oluşturduğu kontrol sistemi diyagramları gösterilmiştir.



Şekil 3.1: Kendi kablosuz ağını oluşturan myRIO kontrol sistemi



Şekil 3.2: Harici kablosuz ağına bağlanan myRIO kontrol sistemi

Sistem diyagramlarında görüldüğü gibi myRIO donanımı istenirse kendi WLAN ağını kurarak kontrol ve izleme işlerini yapabildiği gibi başka bir WLAN ağına da bağlanabilir.

### 3.1 myRIO Kontrol Kartı

National Instruments firmasının üretmiş olduğu myRIO 1900 yeniden konfigüre edilebilir giriş çıkış (I/O) geliştirme kartı öğrencilerin ya da araştırmacıların kontrol, robotik ve mekatronik sistemlerin tasarımında kullanabilecekleri bir cihazdır. myRIO üzerinde; Xilinx firmasının üretmiş olduğu Z-7010 All Programmable SoC yongası taşımaktadır. Xilinx firmasının Z-7000 serisi AP SoC yongası, Arm® Cortex™ A9 çift çekirdekli bir işlemci ile 28nm Artix-7 ya da Kintex®-7 FPGA'yi aynı çip üzerinde sunmaktadır. Bu AP SoC yongası ile

birlikte Arm tabanlı işlemcinin yazılımsal programlama esnekliği ile birlikte, FPGA'in sağlamış olduğu donanımsal programlamanın getirdiği hız ve işlem gücü, birlikte sunulmaktadır (Xilinx 2017). myRIO donanımının sahip olduğu diğer teknik özellikler tablo 3.1 de verilmiştir (National Instruments 2016).

**Tablo 3.1:** myRIO teknik özellikler

<b>İşlemci</b>	
İşlemci tipi	Arm® Cortex™ A9
İşlemci saat frekansı	667 MHz
İşlemci çekirdek sayısı	2
<b>Bellek</b>	
Kalıcı bellek boyutu	256 MB
Ddr3 uçucu bellek boyutu	512 MB
DDR3 Saat frekansı	533 MHz
Ddr3 veri yolu genişliği	16 bit
<b>FPGA</b>	
FPGA tipi	Xilinx Z-7010
FPGA mantıksal hücre sayısı	28000
FPGA blok ram boyutu	2.1 MB
FPGA dsp birimi sayısı	80
<b>Kablosuz ağ özellikleri</b>	
Radyo modu	IEEE 802.11 b,g,n
Frekans bandı	ISM 2.4 GHz
Kanal genişliği	20 MHz
Kanal sayısı	USA 1-11, International 1-13
Tx gücü	+10 dBm max (10mw)
Açık alan çekim mesafesi	150 m'ye kadar direk görüş
Anten yönelimi	Her yöne
Güvenlik	WPA, WPA2, WPA2-Enterprise
<b>Usb</b>	
USB host portu	USB 2.0 Hi-Speed
USB device portu	USB 2.0 Hi-Speed
<b>Analog giriş</b>	

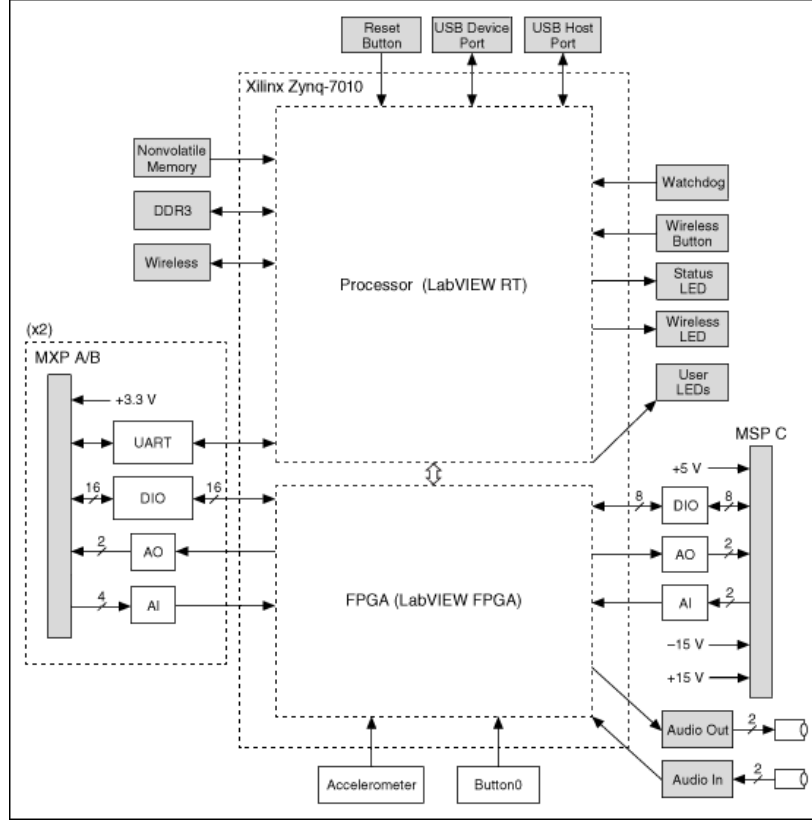
**Tablo 3.1:** myRIO teknik özellikler(devamı)

Örnekleme hızı	500 kS/s
Çözünürlük	12 Bit
Kanal sayısı	MXP konektörde konektör başına 4 adet single ended kanal 0 +5V  MPS konektörde 2 adet diferansiyel kanal $\pm 10V$
Audio girişi	Bir tane stereo giriş (iki adet single ended AC coupled kanal)
Audio giriş Empedansı	DC gerilimde 10 k $\Omega$
Audio bant genişliği	2 Hz ile 20 KHz
<b>Analog çıkış</b>	
Toplam Çıkış güncelleme hızı (Tüm Konnektörlerde)	345 kS/s
Çözünürlük	12 bit
Kanal Sayısı	MXP konektörlerde konektör başına 2 adet single ended kanal, 0 +5V  MSP konektörde 2 adet single ended kanal $\pm 10V$
<b>Dijital Giriş / Çıkış</b>	
Kanal sayısı	MXP konektörde 1 port 16 DIO kanalı her bir konektörde  MSP konektörde 1 port 8 DIO kanalı
Logic seviye	5V uyumlu LVTTTL giriş, 3.3V LVTTTL çıkış
Giriş logic seviyeleri	$V_{IL}$ 0V minimum; 0.8V maksimum  $V_{IH}$ 2.0V minimum; 5.25V maksimum
Çıkış logic seviyeleri	$V_{OH}$ 2.4V minimum; 3.465V maksimum 4 mA source akımı  $V_{OL}$ 0V minimum; 0.4V maksimum 4 mA sink akımı

**Tablo 3.1:** myRIO teknik özellikler(devamı)

Minimum pulse genişliği	20 ns
<b>İkincil dijital fonksiyonlar için maksimum frekanslar</b>	
SPI	4 MHz
PWM	100 kHz
Quadrature encoder girişi	100 kHz
I <sup>2</sup> C	400 kHz
<b>UART</b>	
Maksimum baud hızı	230.400 bps
Data bitleri	5, 6, 7, 8
Stop bitleri	1,2
Parity	Odd, Even, Mark, Space
Flow kontrolü	XON / XOFF
<b>İvme Ölçer</b>	
Eksen sayısı	3
Ölçüm aralığı	±8 g
Çözünürlük	12 bit
Örnekleme sıklığı	800 S/s
Gürültü	3.9 mg <sub>rms</sub> 25 °C’de

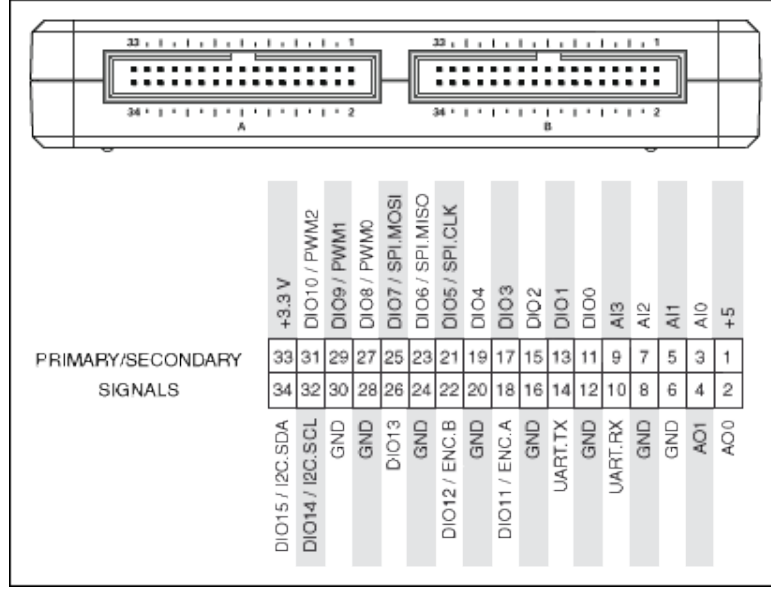
myRIO 1900 kontrol kartı analog giriş (AI), analog çıkış (AO), dijital giriş ve çıkış (DIO), ses ve güç çıkışlarına sahip kompakt gömülü geliştirme cihazıdır. myRIO programlandığı ev sahibi bilgisayara USB üzerinden yada 802.11 b,g,n kablosuz bağlantı ile bağlanabilir. Şekil 3.3’de myRIO gömülü geliştirme kartının donanım diyagramı görülmektedir. myRIO kartı; sahip olduğu Xilinx Z-7010 AP SoC yongasının sahip olduğu FPGA bloğu sayesinde, yazılımsal programlamada yapılamayan birden fazla işlem ve döngünün geçek anlamda paralel olarak yapılabilmesine olanak tanır. Bunun yanında sahip olduğu Arm Cortex A9 işlemcisi ve çalıştırdığı Labview RT işletim sistemi ile USB, 802.11 b,g,n gibi haberleşme protokollerini de kolaylıkla yerine getirebilmektedir. Bu açıdan oldukça esnek geliştirme ortamı sunmasının yanında oldukça güçlü işlem yeteneği ve veri toplama özelliklerine de sahiptir.



Şekil 3.3: myRIO donanım blok diyagramı

National Instruments myRIO iki adet genişleme portu (MXP) konektöre sahiptir. MXP A ve MXP B olarak adlandırılan bu konektörler tamamıyla aynı sıraya ve aynı isimlere sahiptir. Sinyal kanalları ayırımı yazılım tarafında konektör isminden sonra sinyal ismi verilerek ayırım sağlanır. Buna örnek olarak konektör A / DIO1 ile konektör B / DIO1 farklı sinyalleri taşıyan iki farklı kanalı temsil etmektedir. Bunun yanında MXP konektörler üzerinde dikkat edilmesi gereken bir diğer özellik ise bazı sinyal kanallarının ikincil fonksiyona sahip olmasıdır. Şekil 3.4'de MXP A ve B konektörleri myRIO üzerindeki yerleri ve sinyal isimleri görünmektedir. Tablo 3.2'de MXP konektörleri isimleri ve tanımlamaları verilmektedir.



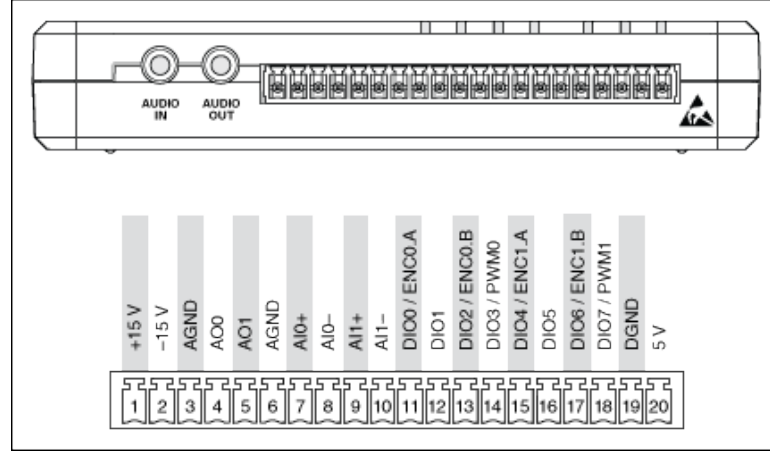


Şekil 3.4: MXP konektörü sinyal bağlantıları

Tablo 3.2: MXP konektörü sinyal açıklamaları

Sinyal ismi	Referans	Yönü	Tanımlama
+5V	DGND	Çıkış	+5V güç çıkışı
AI<0...3>	AGND	Giriş	0-5V referanslı single ended analog giriş kanalı
AO<0...1>	AGND	Çıkış	0-5V referanslı, single ended analog çıkış
AGND	Yok	Yok	Analog giriş sinyalleri için referans
+3.3V	DGND	Çıkış	+3.3V güç çıkışı
DIO<0...15>	DGND	Giriş / Çıkış	Genel amaçlı dijital sinyal kanalları 3.3V çıkış, 3.3V/5V uyumlu giriş
UART.RX	DGND	Giriş	UART alınan veri giriş kanalı
UART.TX	DGND	Çıkış	UART gönderilen veri çıkış kanalı
DGND	Yok	Yok	+5V ve +3.3V Dijital sinyaller için referans

myRIO'nun MSP C konektörü olarak adlandırılan özellikle analog sinyallerin ölçülmesi için kullanılan port bağlantısı ve sinyal isimleri şekil 3.5'de görülmektedir. Sinyallerin özellikleri ve tanımlamaları tablo 3.3'de verilmiştir.



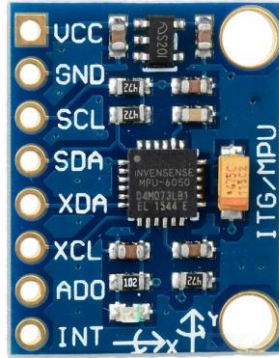
Şekil 3.5: MSP C konektörü sinyalleri

Tablo 3.3: MSP konektörü sinyal açıklamaları

Sinyal ismi	Referans	Yönü	Tanımlama
+15V / -15V	AGND	Çıkış	+15V / -15V güç çıkışı
AI0+/AI0-; AI1+/AI1-	AGND	Giriş	±10V diferansiyel analog giriş kanalları
AO<0...1>	AGND	Çıkış	±10V referanslı single-ended analog çıkış kanalı
AGND	Yok	Yok	Analog giriş ve çıkış kanalları ve +15V / -15V güç çıkışı için referans
+5V	DGND	Çıkış	+5V güç çıkışı
DIO<0...7>	DGND	Giriş / Çıkış	Genel amaçlı dijital sinyal kanalları 3.3V çıkış, 3.3V/5V uyumlu giriş
DGND	Yok	Yok	+5V Dijital sinyaller için referans

### 3.2 MPU-6050 İvme Açı Sensörü

Dört rotorlu hava aracının denge uçuş sistemi için gerekli ivme ve açı geri besleme bilgileri için Ivensense firmasının üretmiş olduğu MPU-6050 ivme açı ölçer sensörü kullanılmıştır. Ivensense'in üretmiş olduğu MPU-6050 sensörü tablet bilgisayarlar ile akıllı telefonlarında sıklıkla kullanılmaktadır. Ivensensin üretmiş olduğu MotionInterface™ gömülü yazılımı ile MPU-6050 sensörleri, ev sahibi sisteme, hareket ve açı izlemeyi kolay şekilde sunmaktadır. Bu sayede akıllı telefonlarımızdaki hareket algılama özellikleri sağlanmış olur. Bu hareket algılama özelliği sayesinde telefonlarımızda, hareket ile kontrol, yapay gerçeklik, panoramik fotoğraf çekimi ve görüntülemesi, sağlık ve form tutma için adım sayaçları, yaya ve araç navigasyonu için yönelim belirleme gibi özellikler için hareket verileri sağlanmış olmaktadır. Şekil 3.6'da MPU-6050 sensörüne sahip prototipleme kartı görülmektedir.

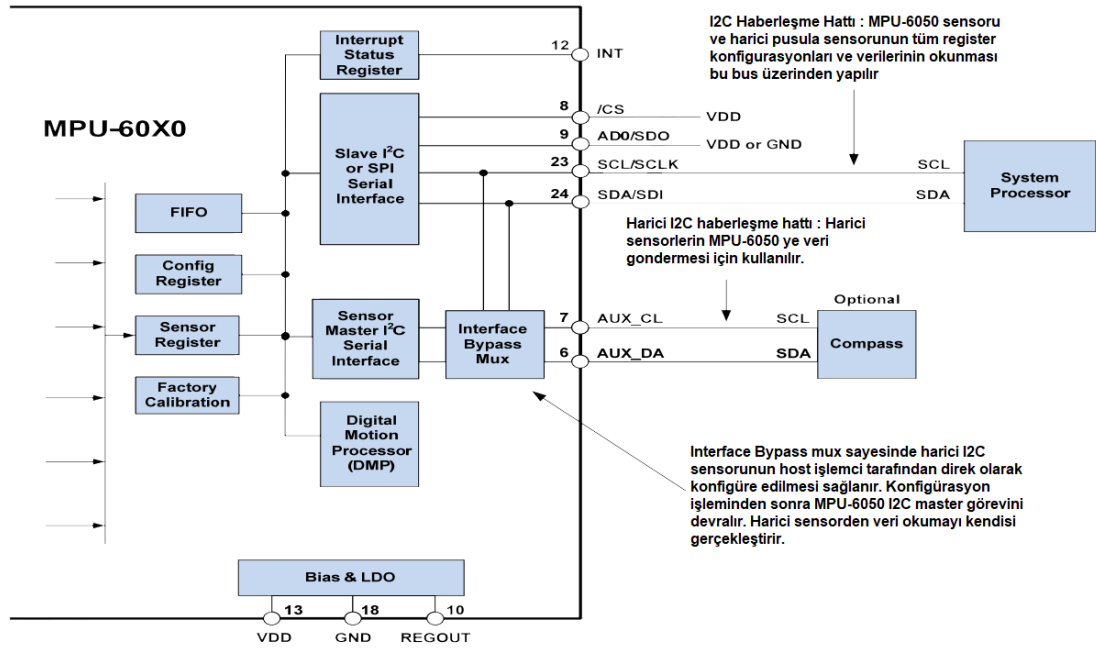


Şekil 3.6: MPU-6050 ivme açı ölçer prototip kartı

MPU-6050 sensörleri sahip olduğu MotionInterface™ teknolojisi sayesinde kesin ve doğru şekilde hareket algılamayı mümkün kılmıştır. Bu sayede akıllı telefonlarımız ve tabletlerimiz 3D uzayda nasıl hareket ettiğini algılayabilen cihazlara dönüşmüştür. MPU-6050 düşük güç tüketimi ve küçük baskı devre ayak izi sayesinde akıllı telefon ve tabletlerin içerisine sığabilecek kadar küçük ve batarya dostu sensörlerdir.

MPU-6050 sensörü I<sup>2</sup>C protokolünü kullanarak ev sahibi sistem ile haberleşir. Üzerinde bulundurduğu 16 bitlik analog dijital dönüştürücüsü ile açıölçer ve ivme ölçer verilerini yüksek hassasiyetle dijitalleştirebilmektedir. Hızlı ve yavaş

hareketleri algılayabilmek için kullanıcı tarafından açıölçer skalasını  $\pm 250$ ,  $\pm 500$ ,  $\pm 1000$  ve  $\pm 2000^\circ/s$  olarak ayarlayabilmek mümkündür. Aynı şekilde ivmeölçer skalası da  $\pm 2g$ ,  $\pm 4g$ ,  $\pm 8g$ ,  $\pm 16g$  olarak ayarlanabilmektedir. MPU-6050 ayrıca sahip olduğu harici I<sup>2</sup>C portu ile manyetik pusula sensörleri ile haberleşip bu 3 eksen manyetik veriyi de Dijital Motion Processor™'ünde işleyerek daha doğru hareket takibi verebilmektedir. Şekil 3.7'de MPU-6050 ivme ve açı sensörünün harici ev sahibi sistem ile I<sup>2</sup>C haberleşmesi blok diyagramı gösterilmiştir.



Şekil 3.7: MPU-6050 ivme açı sensörü bağlantı şeması

### 3.2.1 Sensör füzyon

İvme-açı sensörleri temel olarak bir yonga içerisinde yerleştirilmiş MEMS adı verilen mikro elektromekanik sistemlerden meydana gelmektedir. Bu elektromekanik yapılar yapıları gereği bazı kusurlara sahiptir. Dört rotorlu hava aracı sistemi veya başka hareket ve yön tayini gerektiren uygulamalarda ivme ve açı sensörlerinin birlikte kullanılmasının sebebi bu sensörlerdeki yapısal kusurlarının diğer sensör tarafından telafi edilmesi gerekliliğidir. Bazı uygulamalarda ise sapma açısının daha doğru hesaplanabilmesi için ivme-açı sensörlerine ek manyetik pusula da kullanılmaktadır. MPU-6050 sensörünün harici I<sup>2</sup>C portuna manyetik pusula bağlanabilmesinin sebebi; sapma açısının algılanmasını iyileştirmek içindir.

Sensör füzyon, sensörler tarafından elde edilen verilerin doğruluğunu arttırmak için kullanılan, birden fazla sensörden alınan verilerin bir yazılım tarafından kombine edilerek doğruluğunun artırılması ile oluşur. Sensör füzyon aynı tip sensörlerden veri alabildiği gibi farklı tip sensörlerden verilerin toplanarak belirli bir veriyi daha doğru kestirebilmek için kullanılabilir.

İvme sensörleri yapıları itibari ile ürettikleri açı vektörlerinde çok fazla miktarda gürültü barındırırlar. Bu gürültü sinyalinin filtre edebilmek için kullanılacak en basit filtre metodu olan alçak geçiren filtre kullanmak sensörün ürettiği sinyallerde iyileşme sağlar ancak, faz gecikmesine sebep olurlar. Bu faz gecikmesi hava araçları gibi yön ve açı tayininin hızla yapılmasını gerektiren sistemler için ideal değildir. Ayrıca ivme sensörünün hızlanma ve yavaşlamalar süresince de ivmeleri yanlış okuyarak yönelim açılarını hesaplamada hatalara neden olabilmektedir. İvme sensörü  $t \rightarrow \infty$  iken yönelim hesaplamada güvenilir bir sensör iken, çıkış verilerinde gürültü ve anlık ivmelenmelerde yanlış veri vermesi sebebiyle yön tayini yapmada hatalı ölçümlere yol açmaktadır.

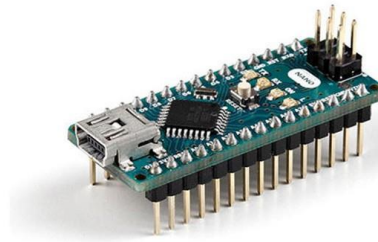
Açı sensörleri ise sensör eksenlerinin açılarının birim zamandaki değişimlerini veren sensörlerdir. Açı sensörlerinin yön tayininde kullanılabilmesi için sensör verilerinin belirli başlangıç konumundan itibaren integrali alınarak, başlangıç konumuna göre yönlerdeki değişim açıları hesaplanmış olur. Açı sensörlerinin yönelim hesaplamada tek başına kullanılamazlar. Bu durum açı sensörlerinin yapısından kaynaklanan statik hata oluşturmasıdır. Açı sensöründeki statik hata, sensör durağan halde iken bile sensör çıkış verilerinde sıfırdan farklı belirli değerler oluşturmasıdır. Bu hata değerleri sensör verilerinden integral işlemi yapılarak yönelim açıları hesaplamaya çalıştığımızda belirli zaman sonrasında hesaplanan açılarda büyük hatalara sebep olur. Açı sensöründen alınan veriler anlık ivmelenmeler, açı – yönelim değişimleri gibi kısa zaman aralıkları içerisinde güvenilirdir.

Sensör füzyon algoritması, ivme sensörü verilerinin uzun süreler sonrasındaki doğruluğu ile açı sensörünün anlık ivmelenmeler ve kısa zaman aralıklarındaki doğruluğunu kullanarak tüm zaman aralığında en doğru çıkış verilerini üretmektedir. Dört rotorlu insansız hava aracının uçuşu boyunca yönelimini ve yer çerçevesine göre olan konumunun bilenebilmesi bu algoritmanın doğruluğuna bağlıdır.

Gerçekleştirdiğimiz dört rotorlu hava aracı kontrol sistemi çalışmasında bulanık mantık kontrol sistemi tasarımına ve çalıştırılmasına ağırlık verileceğinden sensör füzyon algoritmaları üzerinde fazla durulmamış, MPU-6050 sensörü üzerinde Dijital Motion Processor™ kullanılarak sensör füzyonu ivme açısı sensörünün gömülü algoritmasına yaptırılmıştır. Tez çalışmaları başında MPU-6050 sensörü ile myRIO I<sup>2</sup>C protokolü ile haberleştirilmiş, ham ivme açısı değerleri sensör üzerinden okunabilmiştir. Ancak MPU-6050 üzerine gömülecek olan Dijital Motion Processor™ yazılımı için gerekli yazmaç datalarının yüklenmesi; Labview görsel programlamada dizi oluşturmadaki zorluk nedeniyle açık kaynak kodlu Arduino ile yapılmıştır. Arduino tarafından okunan yalpa, yunuslama ve sapma açıları seri bağlantı protokolü ile myRIO'ya aktarılmıştır.

### 3.3 Arduino Nano

Arduino nano; dört rotorlu hava aracı sistemimiz üzerinde myRIO ile MPU-6050 üzerinde çalışan Dijital Motion Processor™ algoritmasının iletişimi için kullanılmıştır. Arduino nano; açık kaynak donanımlı, üzerinde Microchip Atmel ATmega328 mikro kontrolöre sahip hızlı prototipleme yapılabilen bir geliştirme ortamıdır. Sahip olduğu geniş kullanıcı kitlesi sayesinde çok geniş kaynak kütüphaneleri ve yazılım desteği sunulabilmektedir. Çalışmamızda arduino platformunun örnek kodlarından faydalanılarak MPU-6050'nin Dijital Motion Processor™'ü çalıştırılmıştır. Bu sayede başlı başına bir tez konusu olan sensör füzyonu hazır kodlar ile yapılmış, çalışmamızda bulanık mantık sistemi üzerinde yoğunlaşmıştır. Şekil 3.14'de arduino nano gömülü sistem prototip geliştirme kartı resmi görülmektedir.



Şekil 3.8: Arduino Nano gömülü sistem geliştirme kartı

### 3.4 Fırçasız DC Motorlar

Dört rotorlu hava aracının motorlarının seçiminde myRIO ağırlığı ve eklenebilecek çevre birimleri ve farklı sensörler düşünüldüğünde toplam olarak 1.5kg olması beklenmektedir. Bu sebeple motor seçiminde uçuş süresi ve motor – pervane verimliliğinden ziyade yukarı yönlü kaldırma kuvvetinin çok olması amaçlanmıştır. Bu sebeple yüksek devirli EMAX Xa2212 1400 kV motorlar ile büyük çaplı pervaneler kullanılması planlanmıştır. Xa2212 1400 kV'luk motorun datasheeti incelendiğinde 7.4V'luk bataryalar ve 20,3 cm'lik pervaneler ile rotor başına 500gr itki kuvveti oluşturacağı görülmüştür. Bu motor ile beraber kullanılmak üzere 25.4 cm'lik daha büyük pervaneler kullanılmasına karar verilmiştir. Bu sayede motor başına 500 gr'dan daha büyük itki oluşturacağı ve ileriye dönük olarak eklenecek malzeme ve sensörlerin sorunsuz şekilde uçurulabilmesi garanti altına alınmıştır. Projede kullanılan motor ve pervaneler üniversite laboratuvarından temin edilmiştir. Şekil 3.9'da kullanılan fırçasız DC motor gösterilmiştir.



Şekil 3.9: Emax Xa2212 1400kV outrunner fırçasız DC motor

### 3.5 Elektronik Hız Kontrol Devresi

Elektronik hız kontrol devreleri(ESC) dört rotorlu hava aracının sahip olduğu fırçasız DC motorların hız kontrolünde kullanılmaktadır. Her bir motor için birer adet kullanılan elektronik hız kontrol devreleri motorların anlık akımlarını

kaldırabilecek kadar güçlü seçilmiştir. Seçilen Hobby Wing Skywalker-20A fırçasız motor sürücü kartı sürekli olarak 20A, kısa süreli olarak da 25A akımı kaldırabilecek güçte sürücülerdir. Bu sürücüler myRIO darbe genişlik modülasyonu kanalından aldığı 50 Hz frekansında 1 ila 2 ms görev döngüsü arasındaki değerler ile sürülmektedir. 1ms darbeler motorları en düşük devirde döndürürken, 2ms darbe genişliği verilmesi motorları maksimum hızda döndürmektedir.

### **3.6 Batarya**

Bataryalar elektrikli hava araçlarındaki en önemli bileşendir. Bataryaların enerji kapasiteleri ile ağırlıkları uçuş sistemlerinin uçuş sürelerine etki eden en büyük etmendir. Günümüzde robot uygulamaları ve dört rotorlu hava araçlarında Lithium Polimer pilleri sıklıkla kullanılmaktadır. Li-Po piller; NiCd, NiMH veya Alkaline pillere göre sahip olduğu yüksek enerji yoğunluğu ve hızlı şarj olmaları nedeniyle tercih edilmektedir.

Li-Po piller sahip olduğu yüksek deşarj akımları sebebiyle dört rotorlu hava aracının motorlarının ihtiyaç duyduğu yüksek akımları karşılayabilirler. Hava aracı çalışmamızda 2 adet Li-Po hücrenin seri bağlanmasıyla oluşturulan 7.4V gerilime ve 4100 mAh kapasiteye sahip, deşarj akımı olarak kapasitesinin 25 katını anlık olarak karşılayabilecek batarya seçilmiştir. Kullanılan pervaneler ile motorların maksimum olarak 25A akım çekebileceği üzerine 4 motorun toplam 100A akım çekebilmesi mümkündür. Bu sebeple 4100mAh kapasiteli bataryanın deşarj akımı anlık olarak kendi kapasitesinin 25 katını karşılayabilecek şekilde seçilmiştir.

### **3.7 Dört Rotorlu Hava Aracı Gövdesi**

Dört rotorlu hava aracı gövdesi ve pervaneler hava aracının en kırılgan parçalarıdır. Düşme veya bir objeye çarpma durumlarında ilk darbeyi bu iki parça almaktadır. Yüksek devirlerde dönen pervaneler herhangi bir çarpma durumunda çok çabuk kırılmaktadır. Bu sebeple ucuz plastik malzemelerden yapılan pervaneler seçilmiştir.



Hava aracı gövdesi ise hem kullanılan pervane aplarına gre yeterli uzunlukta kanat aıklığına, hem de stnde myRIO'yu taşıyabilecek kadar yere sahip olacak şekilde seimi yapılmıřtır. Kolay bulunabilirliėi ve ucuz olması sebebiyle piyasada F450 olarak bilinen, 45cm kol aıklığına sahip gvde tercih edilmiřtir.

## 4. SİSTEM TASARIMI ve PROGRAMLANMASI

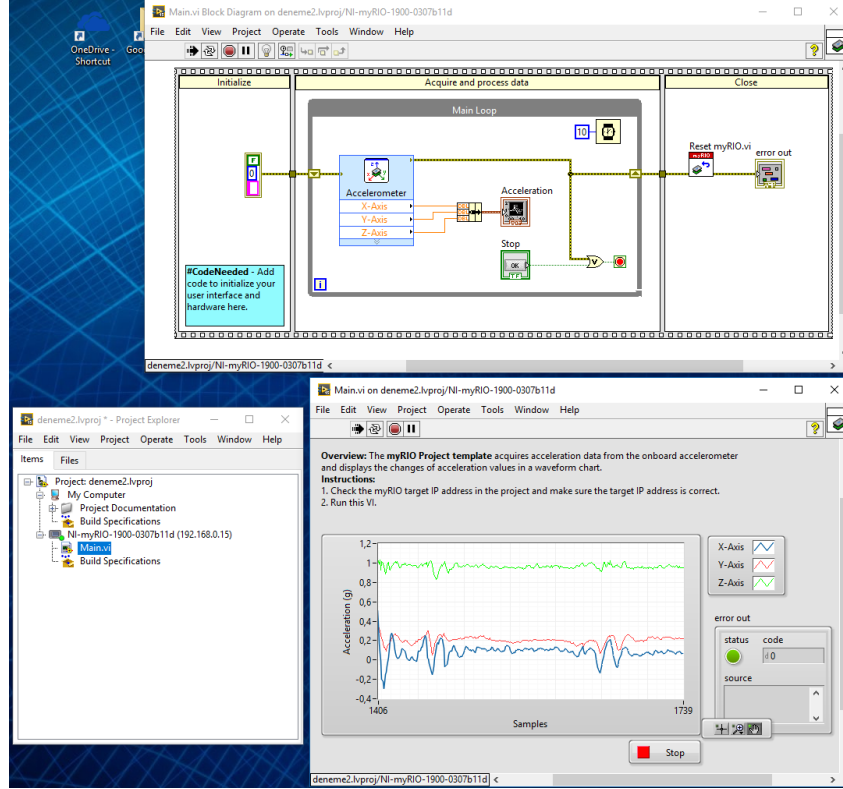
Bu bölümde dört rotorlu hava aracının tüm sisteminin nasıl oluşturulduğu ve LabVIEW ortamı kullanılarak myRIO kontrol kartının programlanması, yer kontrol bilgisayarı üzerinde çalışan LabVIEW sanal ölçüm bloklarının (VI) programlanması ile, bulanık mantık kontrol sisteminin tasarlanması ve myRIO üzerinde çalıştırılması detaylı olarak anlatılacaktır. Yapılan bu çalışmanın anlaşılabilmesi için öncelikle LabVIEW ortamı ve görsel programlama hakkında temel bilgiler verilecek, daha sonra sistemin tasarımı ve donanımların nasıl bir araya getirildiği anlatılacaktır. Daha sonra bulanık mantık kontrol sisteminin tasarımı verilecektir.

### 4.1 LabVIEW Geliştirme Ortamı

Açılımı, “Laboratory Virtual Instrument Engineering Workbench” olan LabVIEW ortamı National Instruments firmasının 1986 yılından beri geliştirdiği sistem tasarım ve görsel programlama dili geliştirme ortamıdır. LabVIEW görsel programlama dili genellikle “G” dili olarak isimlendirilir. Bu isim, makine nümerik kontrolü için kullanılan G-code’ları ile karıştırılmamalıdır. LabVIEW ve görsel programlama dili günümüzde veri izleme, cihaz kontrolü, test düzeneği otomasyonu, veri analizi ve sinyal işleme, endüstriyel kontrol ve gömülü sistem tasarımı gibi uygulamalarda çokça kullanılmaktadır.

LabVIEW görsel programlama, klasik anlamda bilgisayar yazılım kodlarından farklı olarak belirli fonksiyon ve işlevleri yerine getiren program blok ve şekillerin birbirlerine bağlanması olarak görülebilir. LabVIEW’de oluşturulan her program şemasına sanal ölçüm bloğu “Virtual Instrument(VI)” adı verilir. Bir program tek bir ölçüm bloğundan oluşabileceği gibi birden fazla ölçüm bloğunun birleşimiyle de oluşturulabilir. Oluşturulan bir blok temel olarak iki bölümden oluşur. Bunlar; kullanıcı ile program arasında ara yüzü oluşturan ön panel ve programın grafiksel kaynak kodlarını tutan blok diyagramıdır. Ön panel, program içerisindeki verilerin gösterilmesini sağlayan “indicator” yani gösterge ve kullanıcının veri girişini sağlamaya yarayan “kontrol” elemanlarını içerir. Şekil

4.1’de LabVIEW ve myRIO ile birlikte gelen örnek bir program görülmektedir. Bu program myRIO üzerinde dahili olarak bulunan ivme sensörü verilerini okuyarak “g” birimindeki çıkışını gerçek zamanlı grafik halinde göstermektedir.



Şekil 4.1: LabVIEW geliştirme ortamı; proje gezgini, blok diyagramı ve kullanıcı ön paneli

LabVIEW örnek programını incelediğimizde, “Proje Gezgini” altında projemize dahil edilen projemizde bulunan fiziksel donanımlar ve bu donanımlar altında çalışan sanal ölçüm blokları listelenmektedir. Bir projede farklı işlevleri yerine getiren birden fazla fiziksel donanım ve bu donanımların çalıştırdığı birbirleri ile haberleşen birden fazla ölçüm bloğu bulunabilir. Bu donanımlar, proje gezgini (Project Explorer) içerisinde projeye dahil edilebilirler. Ön paneli incelediğimizde karşımıza, myRIO üzerinde dahili olarak bulunan ivmeölçerden aldığı verileri grafiksel olarak gösteren göstergeler karşımıza çıkmaktadır. Bu projedeki ön paneldeki tek kontrol elemanı ölçüm bloğunun çalışmasını durduran “Stop” butonudur. Projenin blok diyagramını incelediğimizde aslında klasik yazılı kodlardan pek de farklı olmayan bir yapı karşımıza çıkmaktadır. Klasik bir gömülü sistem programında bulunan ayarlama ve başlangıç ayarları kısmı, sonrasında döngü içerisinde sürekli çalıştırılan fonksiyonlar ve program/fonksiyon bitiminde yapılacak

işler bayrak bitleri resetleme gibi yapılan işler görsel programlama kodlarında da yapılmaktadır. Görsel program kodu incelendiğinde film şeridi şekli 3 parçaya ayrılmış program şeması görülmektedir. Bu parçalar, 1-İlk ayarlama, 2-Döngü, 3-Bitiş/Reset olarak görülür. Döngü program kısmında “main loop” olarak adlandırılan sürekli olarak “true” değeri argüman olarak verilen “while” döngüsü ile programın sürekli çalışması sağlanır. Döngü çevriminin sağ üst köşesinde görülen saat ve mavi kutu içerisinde yer alan 10 rakamı ile while döngüsü her 10 milisaniyede bir çevrilir.

LabVIEW blok diyagramları içerisindeki veri bağlantıları ile değişken ve sabitlerin renkleri o değişkenin veri yolunun taşıdığı verinin türünü belirler. Turuncu ince çizgiler float tipindeki sayısal değişkenleri, turuncu tek kalın çizgiler float tipindeki tek boyutlu dizi değişkenlerini tanımlarken turuncu ince çift çizgiler ise float tipindeki 2 boyutlu dizileri tanımlar. Benzer şekilde mavi ince tek çizgiler integer tipinde değişkenleri, mavi kalın çizgiler integer tipindeki tek boyutlu dizileri, mavi ince çift çizgiler ise integer tipindeki 2 boyutlu dizileri tanımlar. Aynı düzende yeşil noktalı çizgiler bool tipi değişkenleri, pembe kesikli çizgiler ise string tipi değişkenleri ve bunların farklı boyutlardaki dizi şekillerini tanımlar.

## 4.2 Kontrolör Tasarımı

Dört rotorlu insansız hava aracı; lineer olmayan davranışları sebebiyle kullanılan kontrol metodunun aracın üzerinde uygulanması konusunda matematiksel modelinin tasarımcı tarafından iyi anlaşılması gereklidir. Bu matematiksel modeller dört rotorlu hava aracı gibi üzerinde uzun yıllar çalışılmış bir konu olduğunda çıkartılması kolay olmaktadır. Ancak yeni ve karmaşık sistemlerin matematiksel modelinin çıkartılması karmaşık ve zor olabileceğinden geleneksel PID kontrol, geri adımlamalı kontrol, yapay sinir ağları gibi kontrol sistemlerin uygulanması imkânsız olabilmektedir. Bu tür sistemlerin kontrolünde daha çok uzman bilgisi ile hazırlanan bulanık mantık kontrol sistemleri kullanılabilir.

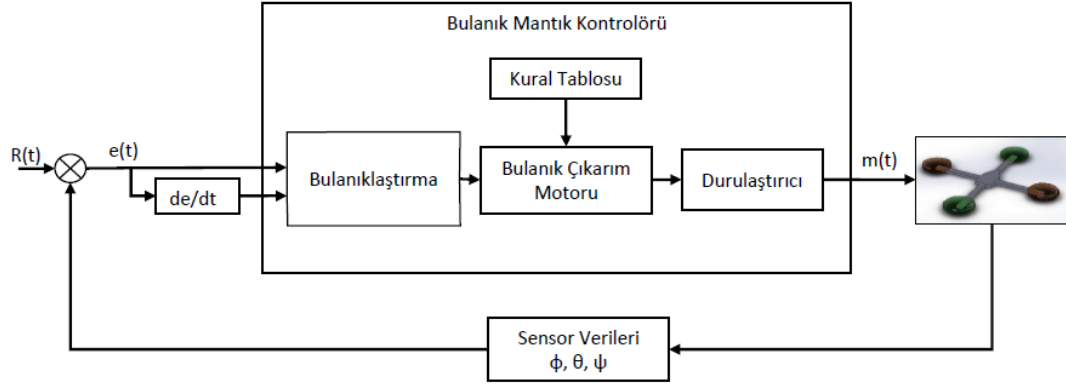
Bu tez çalışmasında dört rotorlu hava aracının pilot bilgisiyle uçurulması üzerinden yola çıkılarak, pilot uzman bilgileri ışığında bulanık mantık kontrol sistemi tasarlanmıştır. Pilot, hava aracına verdiği komutlar ve aracın o komutlara verdiği cevabı gözlemleyerek gerekli düzeltmeleri hava aracına göndererek havada sabit

kalma gibi işlevleri kolaylıkla yerine getirebilmektedir. İşte bu tip uzman bilgisi bulanık mantık kuralları şeklinde hazırlanarak dört rotorlu hava aracının denge ve uçuş kontrolü gerçekleştirilmeye çalışılmıştır. Dört rotorlu hava aracı pilotunun aracın hareketlerini gözlemleyerek havada kalma ve istenilen hareketleri yerine getirmek için kullandığı bilgiler bulanık mantık kontrolcüsüne kural tablosu olarak aktarılmıştır. Sistemin kurallarının oluşturulmasında uzman bilgisinin şu olaylara dikkat etmesine bağlı olarak kurallar oluşturulmuştur.

- 1- Hava aracının o anki açıları ve verilen komutları yerine getirip getirmediği gözlenir. Bu gözlemler hava aracına verilen açısal hareket komutları ve bu açıların değişimleri olarak gözlenir.
- 2- Pilot, hava aracına uyguladığı ani manevralar dışındaki hareketlerinde yunuslama, yuvarlanma ve yalpa kontrolleri ayrı ayrı yapılmaktadır.
- 3- Aracın havada asılı kalma durumu gibi küçük açısal değişimlerin gerçekleştirildiği durumlarda açısal değişimden çok hava aracının açıları daha önem kazanmaktadır.
- 4- Aracın hareket ederken oluşturduğu açılar büyük ise, aracın açılarındaki değişimin izlenmesi, aracın o anki açı durumlarından önemli hale gelmektedir.
- 5- Hava aracı üzerinde gözlemlenen açılar ve açılardaki değişimler; negatif büyük, negatif orta, negatif küçük, sıfır, pozitif küçük, pozitif orta ve pozitif büyük olarak pilot tarafından gözlenmektedir.

Bulanık mantık kontrol sisteminin tasarımı yukarıda bahsedilen pilot uzman bilgisine göre hazırlanmıştır. Uzman bilgisi olarak belirlenen kurallardan ilki sisteminin genel tasarımını tanımlamaktadır. Bu maddede sistemin giriş parametreleri olarak pilot tarafından gönderilen açısal komutlar ve dört rotorlu insansız hava aracı sensörü üzerinden okunan değerler olacağı tanımlanmıştır. İkinci maddede hava aracının denge kontrol sisteminin yunuslama, yuvarlanma ve yalpa kontrolörlerinin ayrı ayrı çalışacağı belirtilmiştir. Üçüncü ve dördüncü maddede bulanık mantık kontrolörünün giriş değişkenlerinin ne olacağı tanımlanmıştır. Bu değerler dört rotorlu hava aracına gönderilen komutlar ile aracın o anki açısal

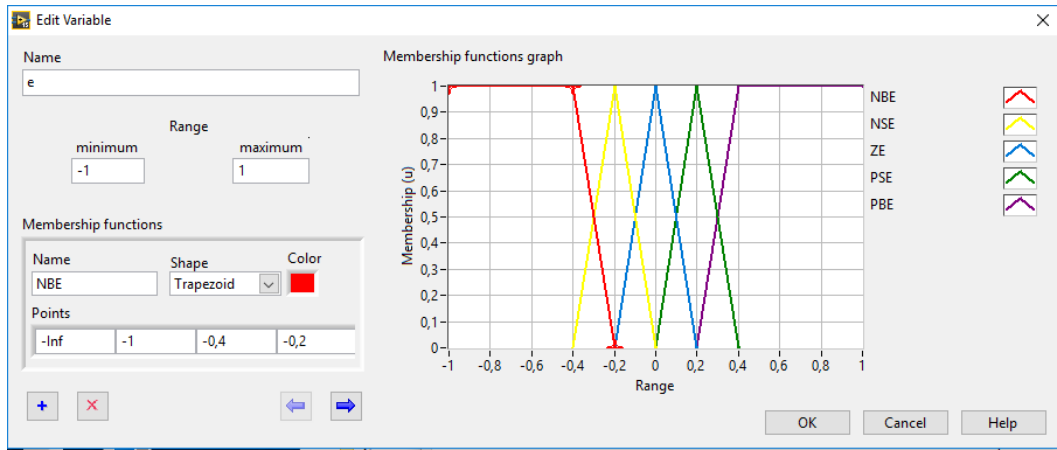
durumu arasındaki hata değeri ve bu hata değerinin belirli zaman aralıklarında örneklenmesi ile oluşan hatadaki değişim sinyalidir. Şekil 4.2’de tasarlanan bulanık mantık kontrolörü sistem diyagramı gösterilmiştir.



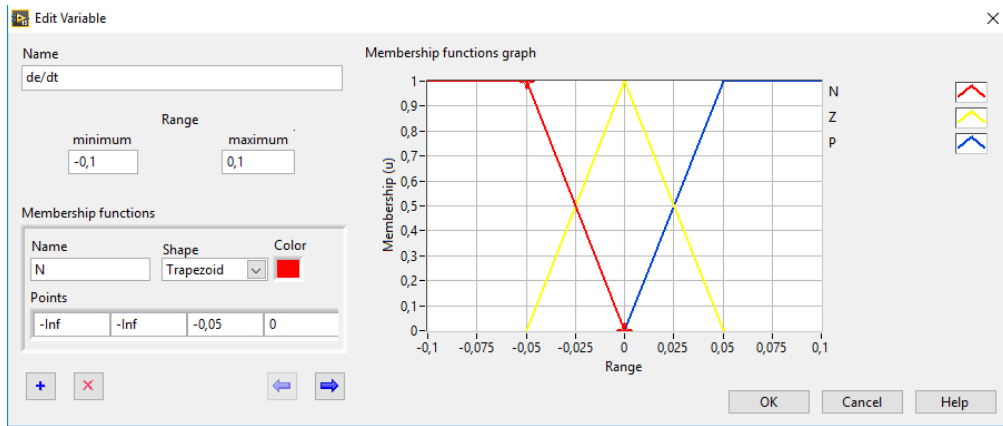
**Şekil 4.2:** Dört rotorlu insansız hava aracı bulanık mantık kontrol sistemi

Şekil 4.2’deki sistem diyagramında gösterilen  $R(t)$  giriş verileri, pilotun hava aracına gönderdiği yuvarlanma, yunuslama ve yalpa komutlarıdır. Bu komutlar yer kontrol bilgisayarı üzerine bağlı USB kontrol kolu ile gönderilen komutlardır. USB kontrol kolu üzerinden alınan bu komutlar ile o anki hava aracı üzerinden alınan açı bilgileri ile arasındaki fark hesaplanarak hava aracının açısındaki  $e(t)$  hata değeri bulunur. Dört rotorlu hava aracı üzerinde çalıştırdığımız bulanık mantık kontrol sistemi 1 Mhz frekansında çalışan ana program döngüsü içerisinde çağrılmaktadır. Her bir tekrarlama sununda hesaplanan hata değeri ile bir önceki tekrar sonucunda elde edilen hata arasındaki fark da hesaplanarak  $(de/dt)$ , yani hatadaki zamana bağlı değişim hesaplanır. Hesaplanan bu hata ve hatanın zamana göre değişimi bulanık mantık kontrolörüne giriş değişkenleri olarak verilmiştir. Hazırlanan bulanık mantık kontrolörü hata giriş değişkeni üyelik fonksiyonu üçgen ve ikizkenar yamuk giriş fonksiyonları kullanılarak hazırlanmıştır. Hata değeri üyelik fonksiyonları 5 üyeden oluşmaktadır. Hata değerinin  $\pm 10$  derece farkları arasında değişen 5 üyelik fonksiyonu olarak tanımlanmıştır. Hata değişkeninin üyelik fonksiyonları negatif büyük (NBE), negatif küçük (NSE), sıfır (ZE), pozitif küçük (PSE) ve pozitif büyük (PBE) olarak tanımlanmıştır. Bu değişkenin üyelik fonksiyonları grafiği şekil 4.3’de gösterilmiştir.

Kontrol sistemi ikinci giriş değişkeni olan hatanın zamana göre değişimi  $de/dt$ , üç üyelik fonksiyonundan oluşturulmuştur. Daha fazla sayıda üyelik fonksiyonu ile oluşturulmaya çalışılmış ancak sistemin cevabında yüksek salınımlar ve sistemde kararsızlıklar ile karşılaşmıştır. Bu sebeple parametrelerdeki değişimin kolay olması sebebiyle hatanın değişimi değişkeninde negatif (N), sıfır (Z) ve pozitif (P) olarak tanımladığımız üç üyelik fonksiyonu tanımlanmıştır.  $de/dt$ , değişkeni üyelik fonksiyonu tanımlaması şekil 4.4’de verilmiştir.



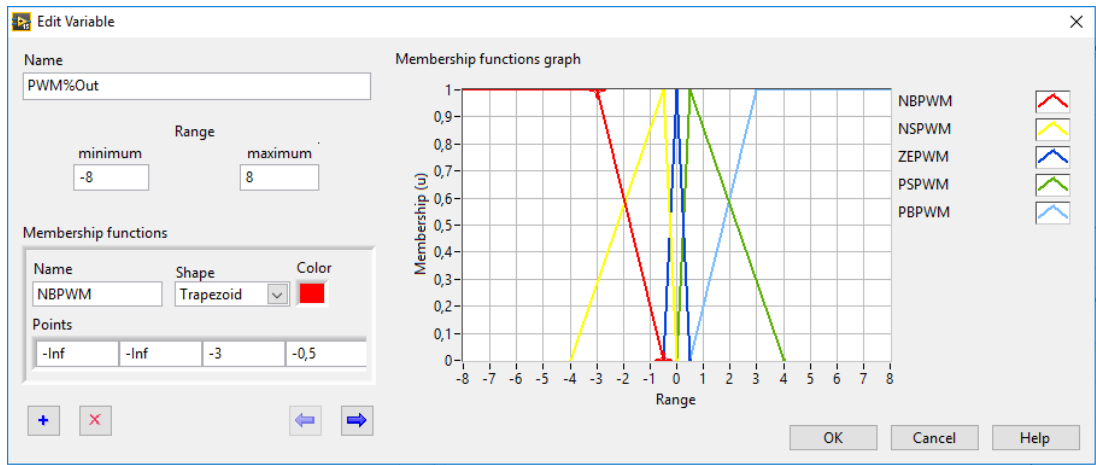
Şekil 4.3: Hata değişkeni üyelik fonksiyonları



Şekil 4.4: Hatadaki değişim üyelik fonksiyonları

Çıkış değişkeni ise motorların devir bilgisini belirleyecek olan PWM değerinin -3 ile 3 arasında normalize edilmiş halidir. PWM çıkış değişkeni 5 üyelik fonksiyonu ile tanımlanmış olup bu değerler; negatif büyük (NBPWM), negatif küçük (NSPWM), sıfır (ZEPWM), pozitif küçük (PSPWM), pozitif büyük (PBPWM) olarak tanımlanmıştır. PWM çıkışları üyelik fonksiyonları tanımlanırken sistem üzerinde denenmiş, hata ve hatadaki değişimin sıfıra yakın olması durumunda

sistem üzerindeki salınımları en aza indirebilmek için pozitif küçük ve negatif küçük üyelik fonksiyonları sifıra yakın tanımlanmıştır. Sistemdeki hatanın büyük olması durumunda ise, sistemin hızlı tepki verip yeniden stabil konuma geçebilmesi için oluşturduğu cevaplardan olan negatif büyük pwm ve pozitif büyük pwm değeri büyük pwm değerleri oluşturmaktadır. Normalize pwm çıkış değerleri, gömülü program döngüsü içinde uygun darbe genişlik modülasyon görev zamanı değerlerine çevrilerek myRIO'nun darbe genişlik çıkışı kanalından çıkış verdirilmiştir. Şekil 4.5'de bulanık mantık kontrol sistemi pwm çıkış değişkeni üyelik fonksiyonları gösterilmiştir.



Şekil 4.5: PWM çıkış değerleri üyelik fonksiyonu

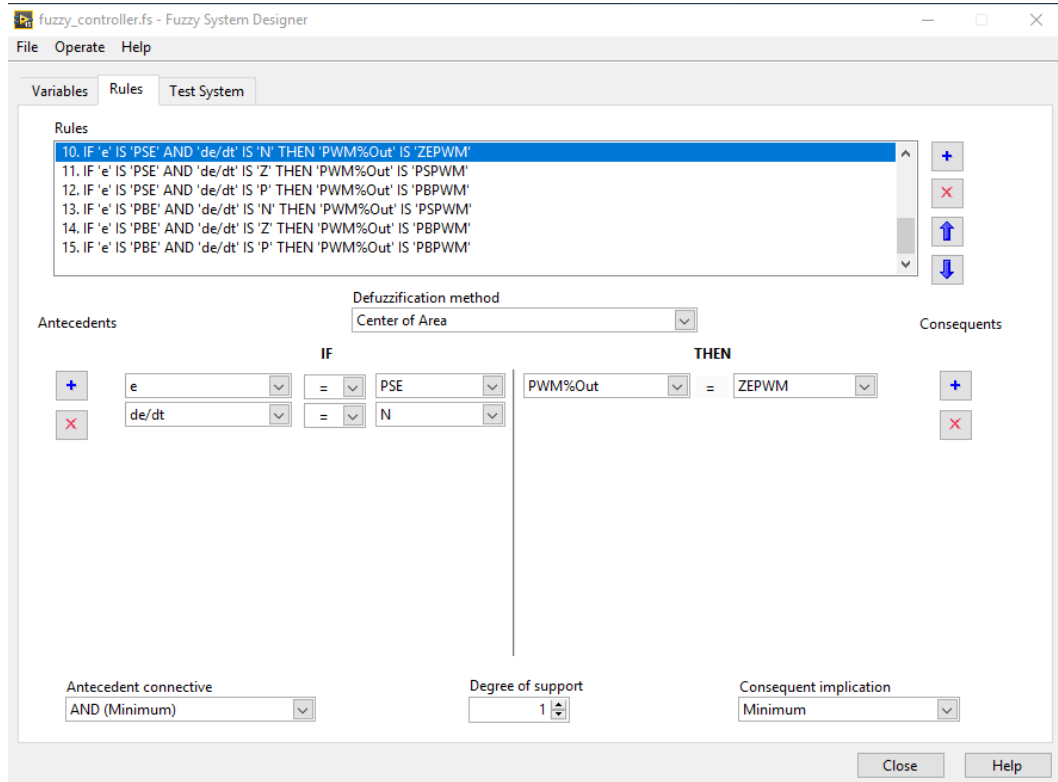
Bulanık mantık kontrol sisteminin verilen girdi değerlerine göre PWM kanallarına çıkış vermesi oluşturulan kural tablosu ile sağlanmaktadır. Sistemin kontrolü için oluşturduğumuz kural tablosu tablo 4.4'de gösterilmiştir. Kural tablosundaki kuralların temelinde, pilotun referans bir komut göndermesi sonucunda hava aracına bu komuta nasıl cevap verdiği, verilen komutu yerine getiremediği, verilen referans komut ile anlık gerçek durum arasındaki farkın ne olduğu ve nasıl değiştiğine göre karar vermek yatar.



**Tablo 4.4:** Kural Tablosu

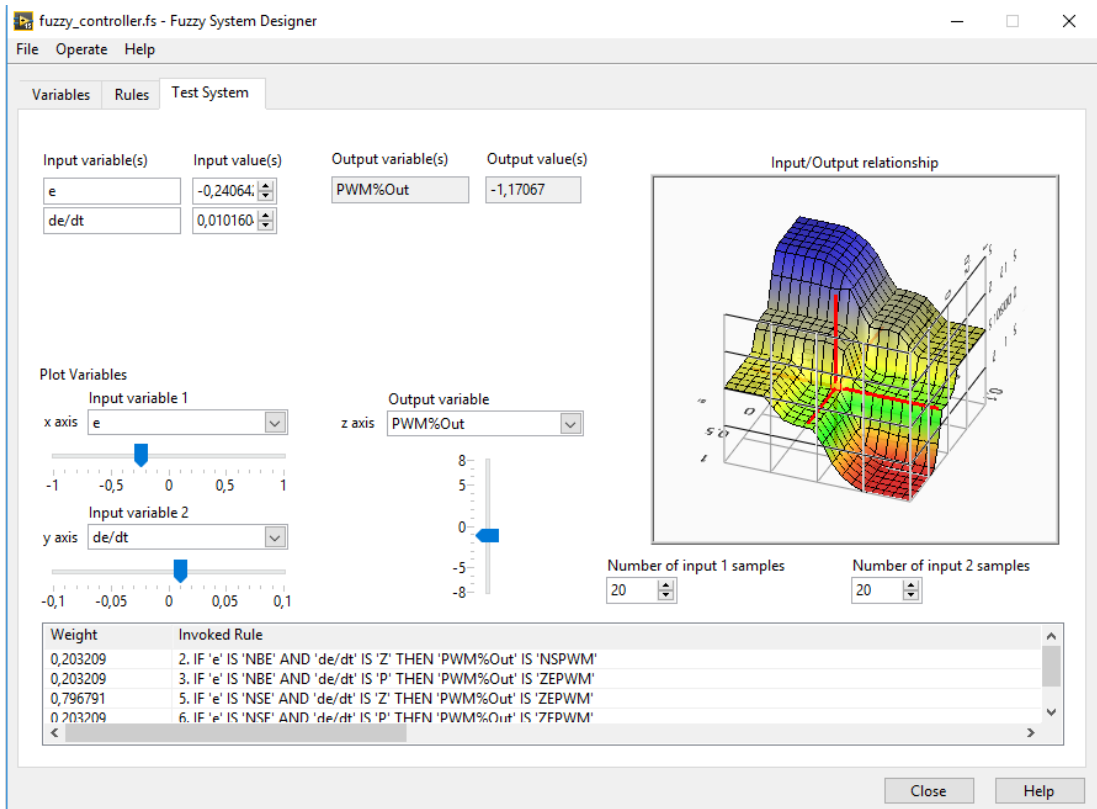
<b>e \ de/dt</b>	<b>N</b>	<b>Z</b>	<b>P</b>
<b>NBE</b>	NBPWM	NBPWM	NSPWM
<b>NSE</b>	NBPWM	NSPWM	ZEPWM
<b>ZE</b>	NSPWM	ZEPWM	PSPWM
<b>PSE</b>	ZEPWM	PSPWM	PBPWM
<b>PBE</b>	PSPWM	PBPWM	PBPWM

Sistem tasarımında kural tablosu oluşturulurken değişkenlerin birbirleri ile ilişkileri “ve” kuralı ile gerçekleştirilmiştir. Bulanık mantık sistemini oluştururken Mamdani tipi bulanık mantık kontrolörü kullanılmıştır. Mamdani tipi bulanık mantık kontrolörü üyelik fonksiyonlarında üçgen üyelik fonksiyonları kullanması sebebiyle uzman bilgisinin sistem kontrolörüne aktarılmasında kolaylık sağlamaktadır. Bulanık mantık kontrol sistemi durulaştırma metodu olarak “center of area”, yani alan merkezi metodunu kullanmaktadır. Şekil 4.6’da Labview Fuzzy System Designer üzerinde kural tablosunun oluşturulması görülmektedir.

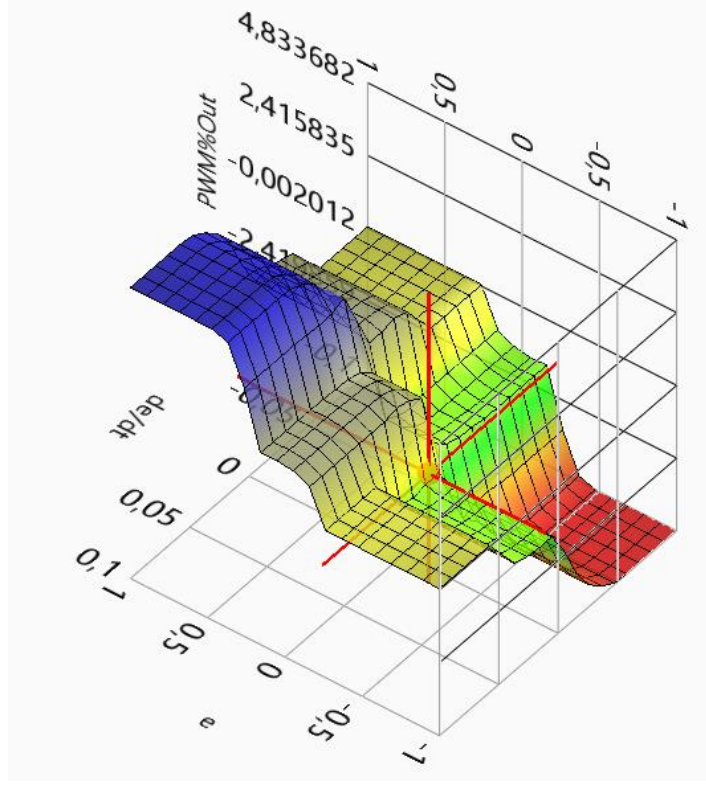


**Şekil 4.6:** “Fuzzy System Designer” kural tablosu oluşturma penceresi

Labview Fuzzy System Designer üzerinden hazırlanan bulanık mantık kontrol sistemi simülasyonu ve giriş değerlerine göre sistemin cevabı “Test System” sekmesi üzerinden görülebilmektedir. Bu sayede hazırlanan kontrol sistemi validasyonu ve kontrol sisteminin simülasyonu yapılabilir. Şekil 4.7 ve şekil 4.8’de hazırlanan bulanık mantık sistemi kural tablosunun 3D yüzey görünümü Fuzzy System Designer üzerindeki “Test System” penceresi ve sonuçları görülmektedir. Test system üzerinde giriş değerlerine karşı çağrılan kurallar ve kuralların ağırlıkları ile sistem sonucu üzerindeki etkileri de görülmektedir. Labview içerisinde sunulan Fuzzy System Designer ile kontrol sisteminin tasarımı, test edilmesi ve geliştirme donanımına yüklenmesi, çalıştırılması ve aynı zamanda programlanması Labview kullanılarak tek bir programla yapılabilmektedir.



Şekil 4.7: Bulanık mantık kontrolör giriş değerleri test sistemi



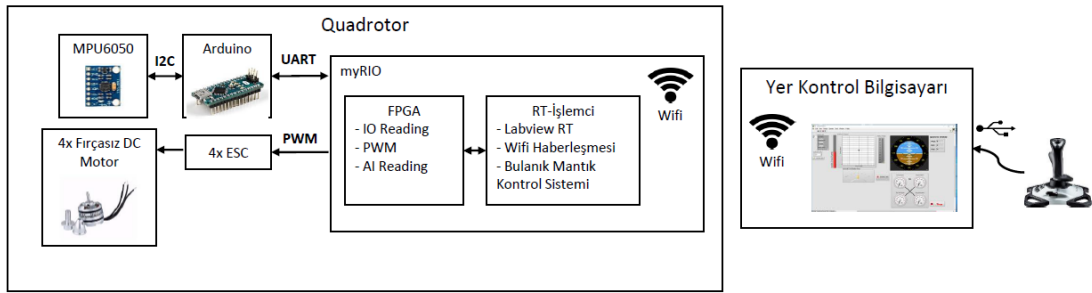
Şekil 4.8: Bulanık mantık kontrolörü “e” ve “de/dt” giriş parametreleri ile PWM%OUT çıkış parametresi ilişkisi

### 4.3 Sistem Tasarımı

Sistemin donanım tasarımında myRIO merkezi işlem birimi olarak dört rotorlu hava aracının beynini oluşturmaktadır. İvme açısı sensorundan gelen veriler arduino aracılığı ile I2C protokolü kullanılarak okunur. Okunan yunuslama, yuvarlanma ve sapma açıları UART üzerinden myRIO’ya aktarılır. MyRIO üzerindeki darbe genişlik modülasyonu çıkışları, motorları kontrol eden elektronik hız kontrol devrelerine bağlanmıştır. Yer bilgisayarı ile haberleşmesi de myRIO üzerindeki kablosuz modülü ile sağlanmaktadır. Oluşturduğumuz dört rotorlu hava aracı sistemi blok diyagramı şekil 4.9’de görülmektedir.

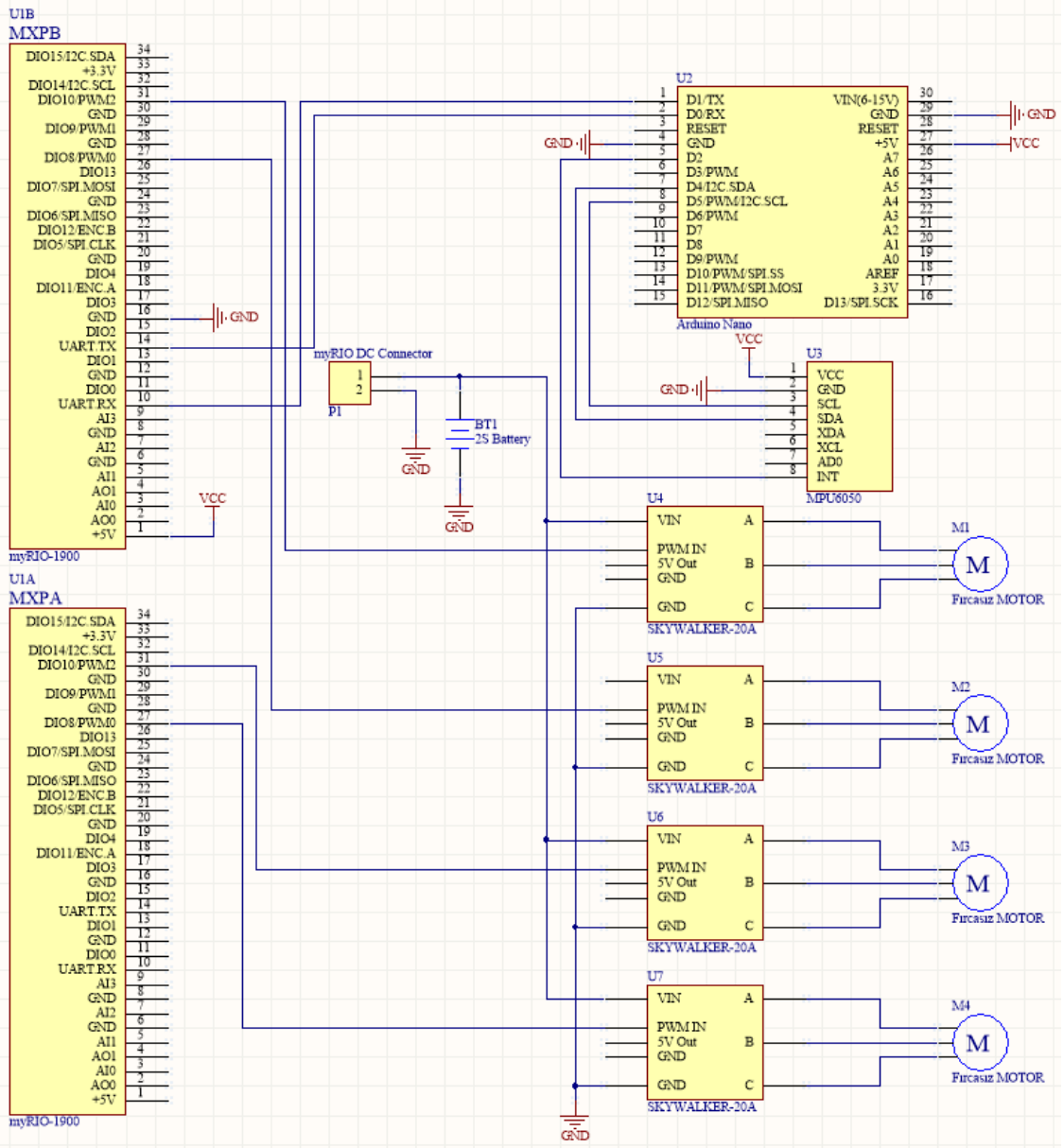
Hava aracı üzerinde 4100 mAh’lik 2 hücreli Lithium Polimer pil bulunmaktadır. Bu piller kimyaları gereği 6.6 volt ve altı durumlarda bitmiş olarak kabul edilmektedir. MyRIO üzerindeki regülatörler sayesinde çalışması için gerekli 3.3V ve 5V gerilimleri kendisi üretebilmektedir. MyRIO’nun giriş gerilimi 6-16 V arasında değişebilmektedir. Bu sayede sistemi çalışması için gerekli akım sağlandığı sürece geniş gerilim aralığında çalışabilmektedir. Kullanılan 4100mAh’lik 25C

batarya tam dolu ve tam boş gerilim aralığında MyRIO'yu sorunsuz şekilde besleyebilmektedir. Sistem üzerinde dikkat edilmesi gereken durum, myRIO'nun 6 volt gerilime kadar çalışabilir olmasıdır. Ancak 6.6 volt ve aşığı Li-Pol batarya için kritik düzeyde bitmiş anlamına geldiğinden myRIO'nun batarya durumunu sürekli gözlemlemesi ve pilota bu bilgiyi göndermesi gereklidir. Bu sebeple batarya üzerindeki gerilim iki dirençten oluşan bir gerilim bölücü ile myRIO analog kanallarından birisi üzerinden okunması ve pilota batarya durumu hakkında bilgi verilmesi sağlanmıştır.



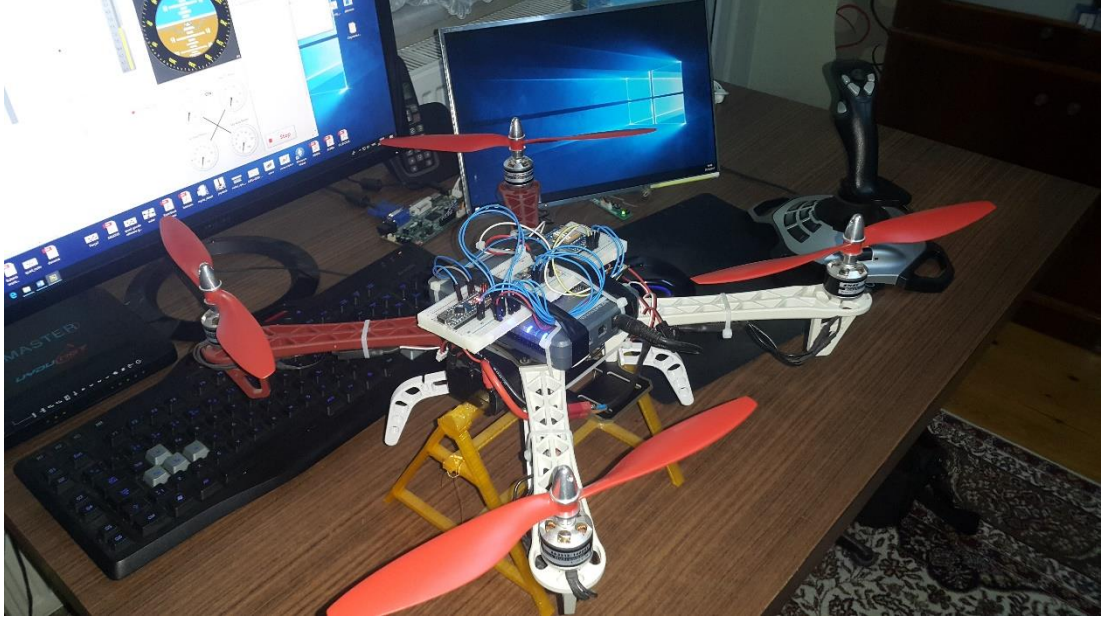
**Şekil 4.9:** Dört rotorlu hava aracı donanım tasarımı blok diyagramı

Dört rotorlu hava aracının donanım birimlerinin bağlantısı şematik diyagramı aşağıdaki şekil 4.10 de gösterildiği gibidir.



Şekil 4.10: Dört rotorlu hava aracı şematik diyagramı

Hazırladığımız dört rotorlu insansız hava aracının fotoğrafı şekil 4.11’de verilmiştir. Bilgisayara bağlı olan USB kontrol kolu ile hareket komutları bilgisayara, oradan da Ethernet kablosu vasıtasıyla kablosuz yönlendiriciye gönderilmektedir. Kablosuz ağa bağlanan myRIO bu komut verilerini ve açılış parametrelerini geri yer bilgisayarına göndermektedir.



**Şekil 4.11:** myRIO kontrollü dört rotorlu insansız hava aracımız

### **4.3.1 Labview Ortamı ve Programlama**

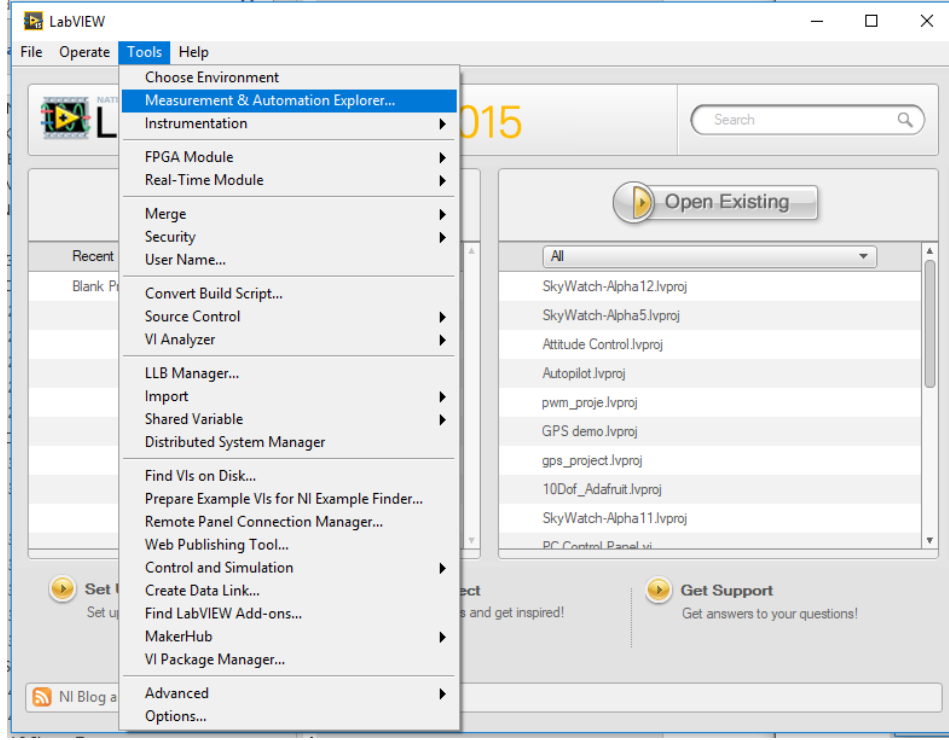
Dört rotorlu hava aracının myRIO ile kontrol edilebilmesi ve kontrol sisteminin tasarlanması için Labview üzerinde proje oluşturulması ve bu projeye myRIO donanımın eklenmesi gereklidir. Proje tasarımında myRIO ile birlikte gelen Labview myRIO 2015, 32bitlik işletim sistemleri için olan versiyonu kullanılmıştır.

Labview üzerinde projeye başlamadan önce yapılması gereken iş myRIO'nun bilgisayar ile veri bağlantısının nasıl olacağı ayarlanmalıdır. myRIO kutudan çıktığı zaman varsayılan ayarlar olarak kablosuz özelliği kapalı olarak gelmektedir. Yer kontrol bilgisayarı ile haberleşmenin sağlanabilmesi ve yazılan programların myRIO ya yüklenebilmesi için kablosuz özelliğin açılması gereklidir. Bunun için myRIO programlamanın yapılacağı bilgisayara USB kablosu ile bağlanması gereklidir. myRIO bilgisayara bağlandığında karşımıza, myRIO USB başlangıç penceresi çıkacaktır. İlk olarak "Lunch The Getting Started Wizard" ile myRIO üzerindeki Labview-RT işletim sisteminin güncellenmesi, myRIO ya bizim vereceğimiz bir adın tanımlanması yapılır. Bu konfigürasyon sihirbazı sonunda myRIO giriş-çıkışlarını ve üzerindeki ivme sensöründen bilgiler okuyan test penceresi karşımıza çıkacaktır. Bu işlemin ardından myRIO üzerinde güncel Labview-RT işletim sistemi çalışır hale gelmiş olacaktır.

myRIO'yu bilgisayarımıza ilk defa bağlamamız durumunda bilgisayar işletim sistemimiz gerekli "National Instruments USBLAN" sürücülerini otomatik olarak yükleyerek myRIO'nun bilgisayarımız üzerinde bir IP almasını sağlayacaktır. Bu IP adresi varsayılan olarak "172.22.11.10" olarak tanımlanmıştır. Bu IP adresi kullanılarak myRIO'yu Labview projelerimize dahil edebiliriz, programlamayı ve kontrolünü USB kablolu ile yapmak mümkündür.

Kontrol sistemimizin sabit olduğu ve kontrol edilecek sistemin ulaşılabilecek yakınlıkta olduğu sistemlerde, sistemin basitliği ve haberleşmenin kolaylığı açısından USBLAN üzerinden sistemin programlanması ve izlenmesi gerçekleştirilebilir. Ancak insansız hava aracı gibi hareketli sistemlerin izlenmesi ve kontrolü gibi durumlarda kablosuz olarak bağlantının sağlanması gereklidir. myRIO kablosuz modülünün kurulması, mevcut ağlara yada kablosuz ağını kendisi oluşturması için Labview Tools sekmesi içerisindeki "Measurements & Automation Explorer" açılmalıdır. Bu program mevcut bilgisayar sistemi ve bağlı olan sistemler hakkında bilgi verecektir. Şekil 4.12'de "Measurements & Automation Explorer"ın Labview üzerinden açılması görülmektedir.

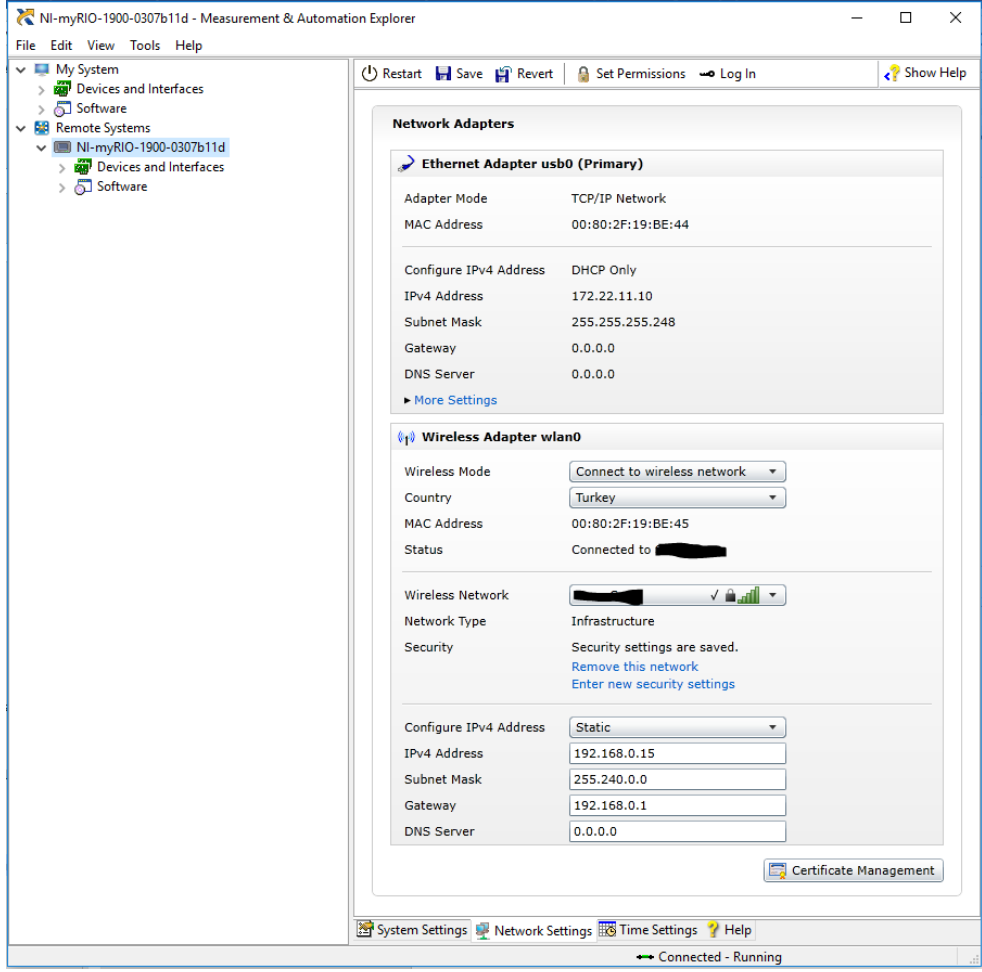
Açılan yeni pencere üzerinde sol tarafta kullandığımız bilgisayar sistemi ve buna bağlı olan uzak cihazların listesi karşımıza çıkacaktır. Programlamayı gerçekleştirdiğimiz bilgisayara şu anda sadece myRIO bağlı olduğundan tek uzak cihaz olarak myRIO görülmektedir. Uzak cihazlar listesini genişletip USB ile bağladığımız myRIO cihazını seçtiğimizde karşımıza myRIO sistemi ile ilgili; işletim sistemi sürümü, myRIO adı, aldığı IP adresi numaraları, sistem saati ve önyükleme ayarları ile ilgili detaylar gelecektir. Bu pencere üzerinde şu an için yapılacak bir değişiklik yoktur. Ancak eğer sistemin kendi başına çalışması istenirse, hazırlanan Labview programı myRIO başlangıç programı olarak gömülmesi gereklidir. Bu durumda myRIO'nun başlangıç programlarının çalışmasını engelleyecek olan "Disable RT Startup App" ayarının seçili olmaması gereklidir.



**Şekil 4.12:** Measurements & Automation Explorer penceresi açılış yolu

myRIO'nun ağ ayarlarını yapabilmek için "Measurement & Automation Explorer" penceresinde myRIO'yu seçtikten sonra pencerenin alt kısımda bulunan "Network Settings"e tıklamamız gereklidir. Bu sekme içerisinde myRIO donanımı üzerindeki "usb0" bağlantısı ile ilgili bilgiler karşımıza çıkacaktır. Bunun altında "wlan0" kablosuz modülü ile ilgili ayarlar görülmektedir. Bu ayarlardan kablosuz modülünün kendi ağını mı oluşturacağı ya da mevcut bir kablosuz ağa mı bağlanacağını ayarlanması gereklidir. Kablosuz ağ istenirse DHCP kullanılarak ağ üzerinde otomatik IP alması sağlanabileceği gibi statik ayarlanarak myRIO'nun alacağı IP sabitlenebilir. Bu sayede ağ üzerinde hangi IP'den myRIO ya bağlanacağımızı bilirsiniz. Şekil 4.13'de myRIO'nun mevcut kablosuz ağa statik IP olarak bağlandığı ayarları görülmektedir.



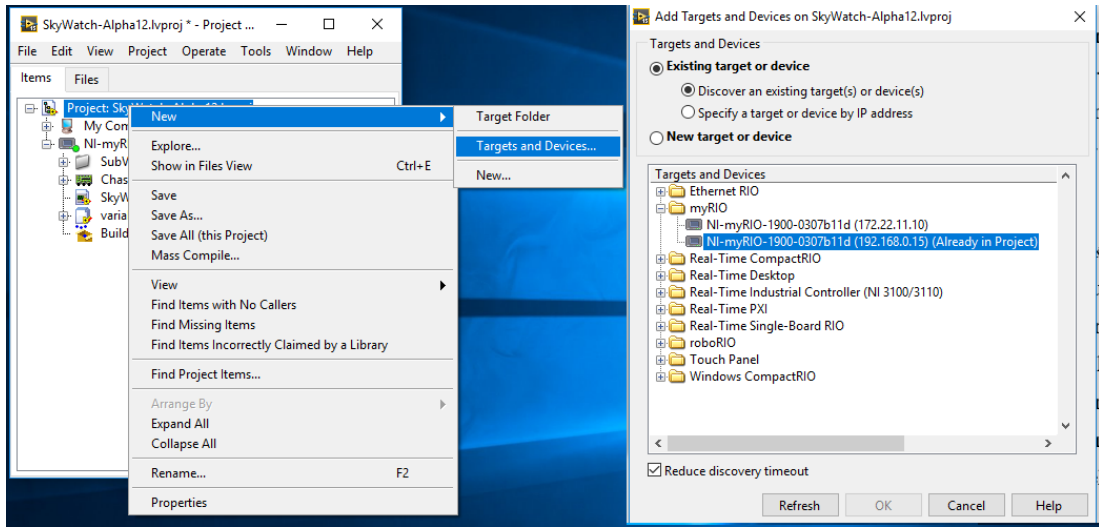


Şekil 4.13: myRIO kablosuz bağlantı ayarları yapılması

Gerekli değişiklikler ve ayarlamalar yapıldıktan sonra ayarların myRIO'ya kaydedilmesi ve sistemin yeniden başlatılması gereklidir. Yeniden başlatma olmadan değişiklikler geçerli olmayacaktır. Pencere üzerinde yer alan “Save” ve ardından “Restart” butonlarını kullanarak myRIO yeniden başlattırılır. Yeniden başlayan myRIO üzerinde kablosuz ışığının IP almaya çalışırken turuncu, IP alıp ağa bağlandıktan sonra mavi yandığı görülür. Ağ üzerinde veri trafiği var ise bu ışığın yanıp söndüğü gözlemlenebilir.

Kablosuz ağ ayarını yaptıktan sonra sistemimizin gömülü yazılımını kurmak için Labview üzerinde proje oluşturuldu ve proje ismi üzerinde sağ tıklayarak “New”, “Targets and Devices” seçilerek projeye donanım ekleme penceresi açıldı. Bu pencere üzerinde “myRIO” listesi genişletilerek sisteme bağlı olan myRIO cihazlarını otomatik olarak taraması sağlanır. Eğer istenirse myRIO ya daha önce vermiş olduğumuz statik IP adresi üzerinden arama yaptırılarak myRIO'nun

projemize dahil edilmesi de sağlanabilir. Yeni cihaz ekleme şekil 4.14’de gösterilmiştir.

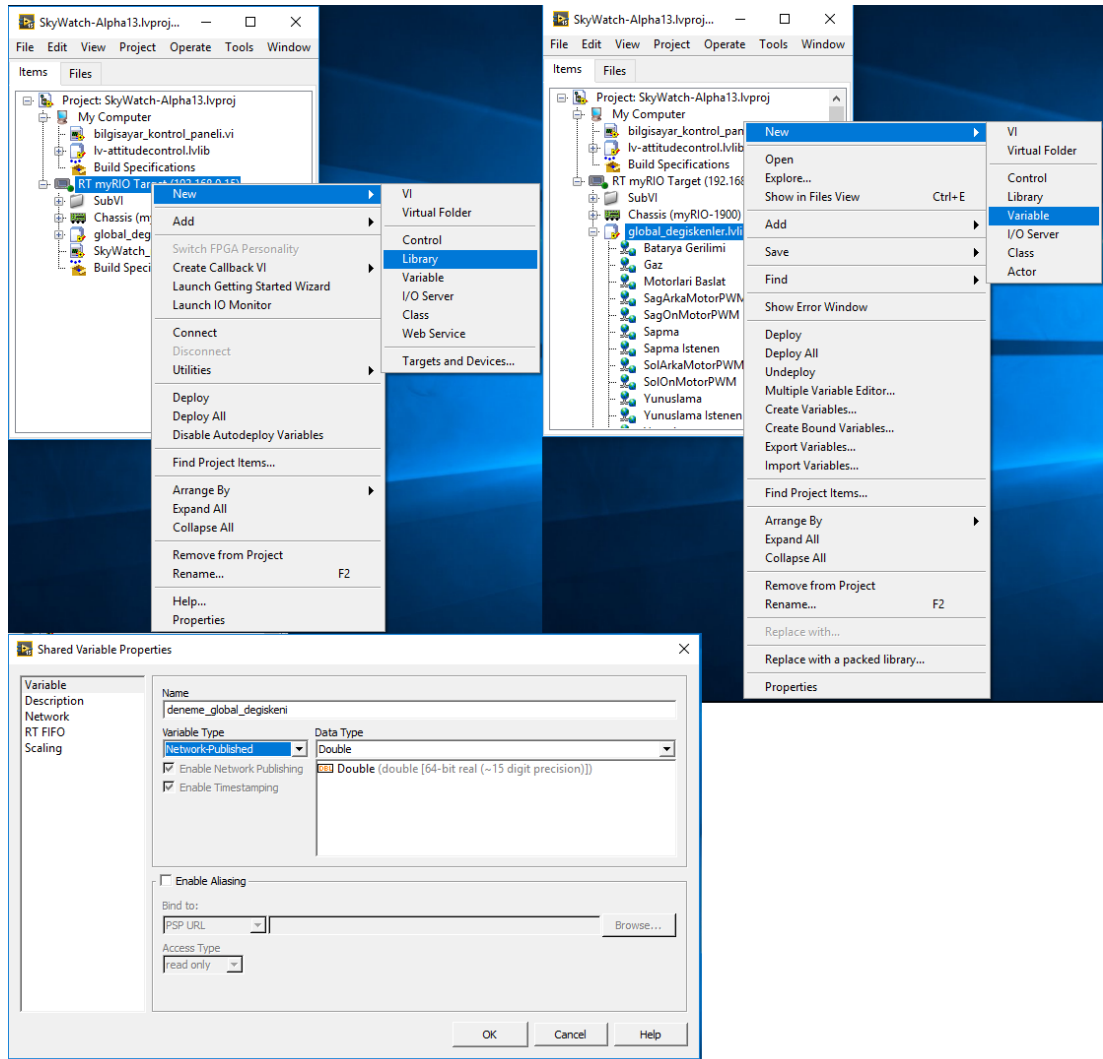


Şekil 4.14: Labview projesine yeni donanım ekleme

Gerçekleştirdiğimiz dört rotorlu hava aracı sistemi programında hem yer kontrol bilgisayarında hem de myRIO üzerinde Labview sanal ölçüm blokları çalışmaktadır. Yer kontrol bilgisayarı üzerinde çalışan sanal ölçüm bloğu, USB kontrol kolu portu ayarlaması, kontrol kolundan okunan verilerin ölü bölgelerini hesaplaması, dört rotorlu hava aracı üzerinden gelen yunuslama, yuvarlanma ve sapma açılarını gösterecek ufuk göstergesini çalıştırma, aracın batarya voltajını gösterme, araç üzerinden gelen motor darbe genişlik modülasyon değerlerine göre motor devirlerinin yüzdelik göstergelerini hesaplama ve ön panelde gösterme işlerini yapar. Hazırlanan dört rotorlu hava aracı görsel programları blok diyagramları büyük olduğundan ekler sayfasında koyulmuştur.

Hazırladığımız hava aracı programında myRIO ile yer kontrol bilgisayarı arasında paylaşılan verilerin Labview projesi üzerinde ağ üzerinde paylaşılan değişken olarak tanımlanması gereklidir. Bu sayede projeye dahil olan tüm cihazlar bu değerleri okuyup yazabilir hale gelecektir. Ağ paylaşımli değişkenleri proje içinde tanımlamadan önce tüm değişkenleri içinde tutacak bir kütüphane oluşturmamız projemizin düzenliliği açısından iyi olacaktır. Bu sebeple “global\_degiskenler.lvlib” adında değişken kütüphanesi oluşturulmuştur. Bu kütüphaneyi oluşturabilmek için, proje gezgini üzerinde myRIO üzerinde sağ tıklayıp, yeni “library” seçilir. Eklenen değişken kütüphanesinin ismi değiştirildikten sonra proje gezgini üzerinde kütüphane

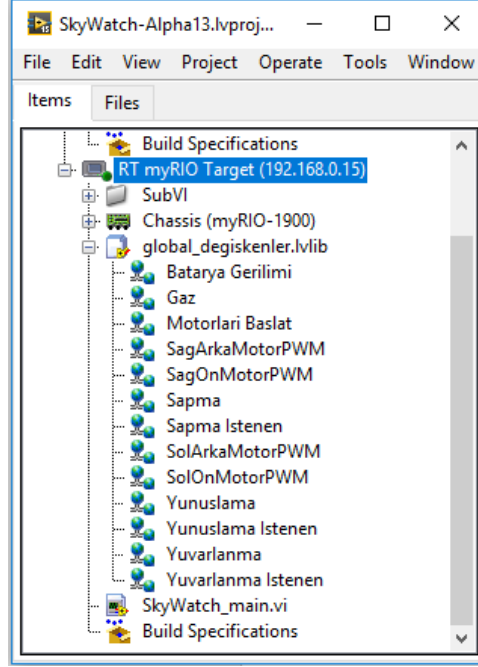
ismine sağ tıklayıp yeni “variable” seçeneği tıklandığında karşımıza eklenecek değişkenin özelliklerini tanımlayacağımız pencere açılır. Bu pencere üzerinden değişken ismi ve boyut türü gibi çeşitli tanımlamalar yapılır. Şekil 4.15’de değişken kütüphanesi ekleme ve bu kütüphaneye ağ paylaşım (network shared) değişken ekleme menüleri ve penceresi görülmektedir. Değişken ekleme sonunda açılan pencere üzerinde; değişken ismi, değişken tipi ve değişken özelliklerinin ayarlandığı paylaşım (network shared) değişkenler özellikleri penceresi görülmektedir.



Şekil 4.15: Ağ üzerinde paylaşımlı değişken tanımlama

Yer kontrol bilgisayarı ile dört rotorlu hava aracı arasındaki haberleşme için gerekli değişkenler; batarya voltajı, motor başlat biti, gerçek yunuslama açısı değeri, istenilen yunuslama açısı değeri, gerçek yuvarlanma açısı değeri, istenilen yuvarlanma açısı değeri, sağ arka motor darbe genişlik modülasyon değeri, sağ ön motor darbe genişlik modülasyon değeri, sol arka motor darbe genişlik modülasyon

değeri, sol ön motor darbe genişlik modülasyon değeri, gaz değeri, gerçek sapma açısı değeri ve istenilen sapma açısı değeridir. Şekil 4.16’de ağ paylaşımli tanımlanmış değişkenlerin proje gezgini üzerindeki tanımlamaları görülmektedir.



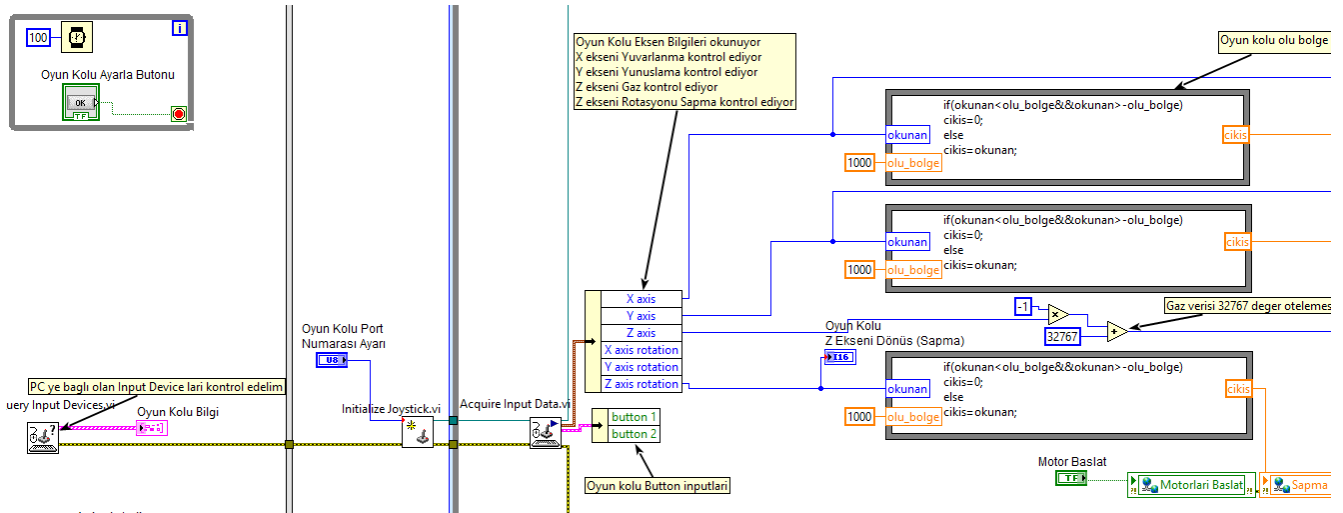
Şekil 4.16: Kontrol programı için tanımlanan ağ paylaşımli değişkenler

Dört rotorlu hava aracı yer kontrol bilgisayarı üzerinde çalışan sanal ölçüm blokları, USB kontrol kolundan aldığı verileri işler ve hava aracına gönderilen ağ paylaşımli değişkenlere yazmaktadır. Bunun yanında Labview Maker Hub üzerinden bulunan hazır ufuk çizgisi göstergesi de bu blok diyagramının içerisine gömülmüştür. Bu sayede, uçuş sistemlerinde bulunan uçağın ufuk çizgisine göre durumunu belirten gösterge sisteme uyarlanmıştır. Yer kontrol bilgisayarı üzerinde yapılan bir diğer işlem ise kontrol kolu verilerinin sıfıra yakın belirli aralıklarda yok sayılmasıdır. “Dead zone” yani ölü bölge olarak bilinen bu işlem, kontrol kolu ekseninde hiçbir komut verilmesinde dahi mekanik hatalardan dolayı sisteme belirli komutlar göndermesini önlemek için yapılmaktadır. Projemizde kullandığımız Logitech Extreme 3d Pro USB kontrol kolu eksenlerindeki hareketin konumuna bağlı olarak -32768 ile 32767 arasında bir değer oluşturmaktadır. Eğer kontrol kolu üzerine kuvvet uygulanmaz ve sabit konumda bırakılırsa dahi 0’dan farklı küçük değerler göndermektedir. Bunu engellemek için kontrol kolundan okunan değerlerin mutlak değeri tanımlanan ölü bölge değerinden küçük ise çıkış değeri olarak hava aracına sıfır değeri gönderilir. Eğer, kontrol kolundan okunan değerlerin mutlak değeri ölü

bölge değerinden büyük ise kontrol kolundan okunan değer hava aracına gönderilir. Ölü bölge hesaplaması denklem 4.1’de verilmiştir.

$$Komut = \begin{cases} 0, & |Kontrol\ kolu\ okunan| < \text{Ölü bölge değeri} \\ Kontrol\ kolu\ okunan, & |Kontrol\ kolu\ okunan| > \text{Ölü bölge değeri} \end{cases} \quad (4.1)$$

Bu sanal ölçüm bloğu üzerinde yapılan bir diğer işlem ise gaz değerinin -32768 ile 32767 arasından 0 ile 65565 aralığına ötelenmesidir. Bu sayede hava aracı üzerinde gaz değerinin 0 – 100 arasında skala edilmesi kolaylaştırılmıştır. Şekil 4.17’de kontrol kolundan okunan verilerin ölü bölge aralığında olup olmadığını kontrol eden ve gaz değerini pozitif değerlere öteleyen “bilgisayar\_kontrol\_paneli.vi”nın bir parçası görülmektedir. Bu sanal ölçüm bloğunun tamamı ekler sayfasında verilmiştir.



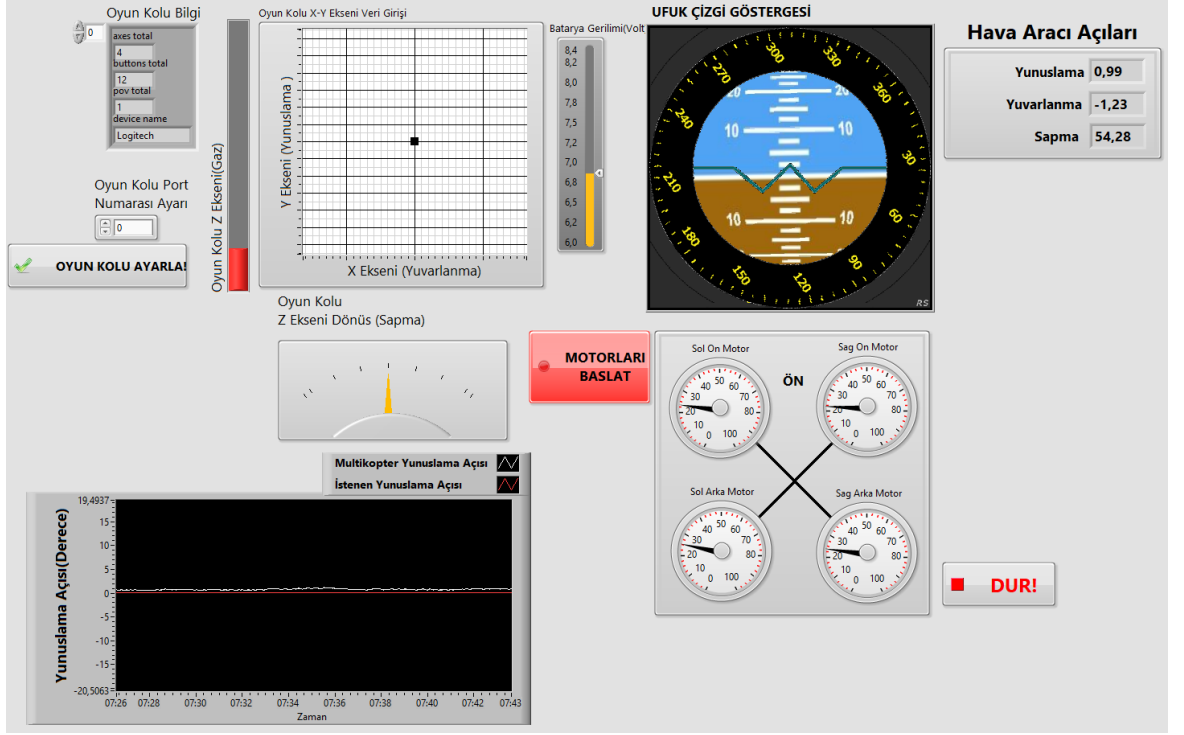
Şekil 4.17: Bilgisayar kontrol paneli blok diyagramı kontrol kolu ölü bölge tanımlamaları (Bkz. Ek A)

Bilgisayar kontrol paneli üzerinde yapılan bir diğer işlem ise hava aracı üzerinden gönderilen motor darbe genişlik değerlerinin motor devrinin yüzdelik biçime döndürülmesidir. Bu sayede kontrol bulanık mantık kontrol sisteminin hava aracını dengede tutabilmek için motor devirleri üzerinde nasıl değişiklik yapabildiği görülmektedir. Kontrol bilgisayarı üzerinde çalışan sanal ölçüm bloğu ön panelini incelediğimizde karşımıza 3 ana kısım çıkmaktadır. Pilotu bilgilendirmeye yönelik olan bu görsel ölçüm cihazları sayesinde dört rotorlu hava aracının açılı, motor devir yüzdeleri, batarya gerilimi, kontrol kolu komut girdi verileri gibi veriler ekran üzerinde görülebilmektedir.

Sanal ölçüm bloğu ön panelindeki sol tarafta yer alan kontrol kolu bilgi dizisi ve kontrol kolu ayarlama butonu ile USB portuna bağlı olan kontrol kolunun ilk ayarlaması gerçekleştirilir. Kontrol kolu bilgi üzerindeki indeks değiştirilerek USB portlarına bağlı olan HID cihazları kontrol edilir. USB klavye ve USB fare gibi cihazlar bu indeks içinde bulunabilir. İstenilen indeks bulunduktan sonra hemen altta bulunan “kontrol kolu port numarası ayarı” değişkenine USB kontrol kolunun bulunduğu indeks numarası girilir. Programı oluşturduğumuz bilgisayardaki USB cihazları sayısına göre bu indeks 0 olmuştur. Her bilgisayarda farklı olabilecek bu sayı doğru şekilde girilmesi gereklidir. kontrol kolu port numarası doğru girildikten sonra “Kontrol Kolunu Ayarla” butonuna tıklanması ile birlikte yer kontrol bilgisayarı kontrol paneli sanal ölçüm bloğu sonsuz döngü içerisinde dört rotorlu hava aracının verilerini okumaya ve veri göndermeye başlar.

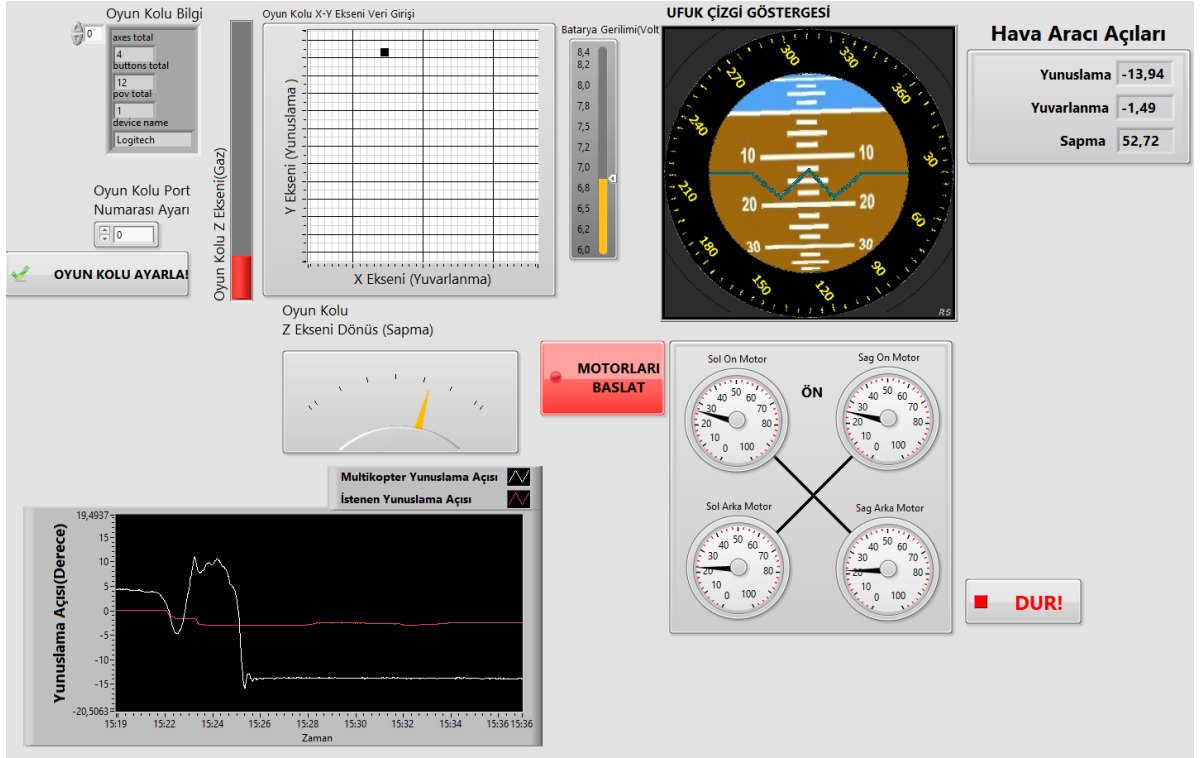
Ön paneldeki bir diğer kısım ise kontrol kolu verilerinin pilota bildirmek için hazırladığımız kontrol kolu XY eksenini veri girişi grafiği, kontrol kolu gaz girdisini gösteren çubuk grafiği ve kontrol kolu Z eksenini dönüş bilgisi sapma açısını gösteren yönelim göstergeleri mevcuttur. Bu kısım pilotun hava aracına gönderdiği verileri gösteren kısım olarak karşımıza çıkmaktadır.

Ön panel üzerindeki son kısımda ise; dört rotorlu hava aracından alınan yunuslama, yalpa ve sapma açılarını gösteren ufuk çizgi göstergesi ile bu değerleri sayısal açı değerleri olarak gösteren dizi bulunur. Ufuk çizgi göstergesinin solunda ise batarya voltajı göstergesi çubuk grafiği görülmektedir. Bu grafik batarya voltajını 8.4 ile 6 volt arasında göstermektedir. Bu göstergelerin altında 4 adet 0 ile 100 arasında değerlere sahip motorların devir sayılarını yüzdeler cinsten gösteren gösterge bulunmaktadır. Bu göstergeler myRIO'nun oluşturduğu hız kontrol devrelerine gönderilen darbe genişlik modülasyonları sinyallerinin görev zamanı oranlarına göre hesaplanmıştır. Pilotu bilgilendirme amaçlı bu göstergeler kontrol sisteminin hangi motorlara ne kadar güç verdiğini bildirme amaçlıdır. Şekil 4.18'de kullanıcı ön panelinin pilot tarafından az miktarda gaz verdiği kontrol kolundan herhangi bir yunuslama ve yuvarlanma girdisi vermediği ve hava aracının yaklaşık olarak yere paralel durduğu andaki kullanıcı ön paneli görüntüsü görülmektedir.



**Şekil 4.18:** Dört rotorlu hava aracı yer kontrol bilgisayarı kullanıcı paneli ara yüzü

Diğer bir durumda ise pilot tarafından az miktar gaz verilip kontrol kolu X ekseninde bir miktar ileri itilerek negatif yunuslama yapması istenmiştir. Diğer göstergelerin çalışması görülmesi açısından kontrol kolu Z ekseninde saat yönünde bir miktar döndürülmüştür. Bu sayede Z eksen dönüş yönü sapma açısı göstergesinin nasıl değiştiği de görülmektedir. Şekil 4.19'da kullanıcı panelinin pilot komutuna göre nasıl değiştiği, komut girdilerinin ve hava aracından okunan verilerin değişimi görülmektedir.



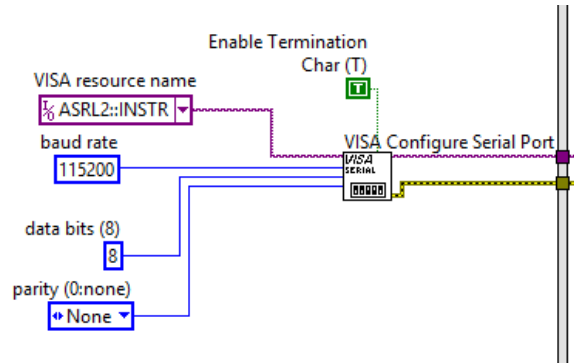
Şekil 4.19: Dört rotorlu hava aracı kullanıcı paneli kumanda girdileri değişimi görüntüsü

Dört rotorlu hava aracı üzerinde çalışan sanal ölçüm bloğunu incelediğimizde karşımıza sıralanmış düzende bir yapı ortaya çıkmaktadır. Labview görsel programlama belirli bir blok içerisindeki tüm program kodlarını karışık olarak işlemektedir. Kodların belirli bir sıraya göre işlenmesinin gerekli olduğu durumlarda “Flat Sequence Structure” kullanılarak işlemlerin birbiri ardı işlenmesi sağlanmaktadır. Bu sayede harici sensörlerin ayarlanması, kontrol sisteminin dosyalama sisteminden açılması ve ayarlanması gibi işler program ana döngüsünden önce yapılabilir. Film şeritlerine benzeyen bu yapı sayesinde program kodlarının akışı belirli düzende işletilebilmektedir. Hava aracı üzerinde çalışan gömülü yazılım sırasıyla; arduino üzerinden UART seri haberleşmesi için gerekli ayarlamaları yapar. MyRIO ile arduino arasındaki seri haberleşme protokolü ayarları tablo 4.5’de verildiği gibidir. MyRIO seri bağlantı portunun ilk ayarlamasını yapan ana program bloğu görseli şekil 4.20’de gösterilmiştir. Bu blok üzerinde myRIO MXPB portu üzerinde bulunan UART seri bağlantı kanalı kaynak adı ASRL2::INSTR olarak tanımlanmıştır. Bu referans girdi parametresi seri bağlantıyı kuran hızlı sanal ölçüm bloğuna argüman olarak verilmektedir. Bu bloğa ayrıca; seri port ile ilgili olan saniyedeki bit sayısı, veri biti sayısı, sonlandırma karakteri ve eşitlik biti gibi parametreler argüman olarak girilir.



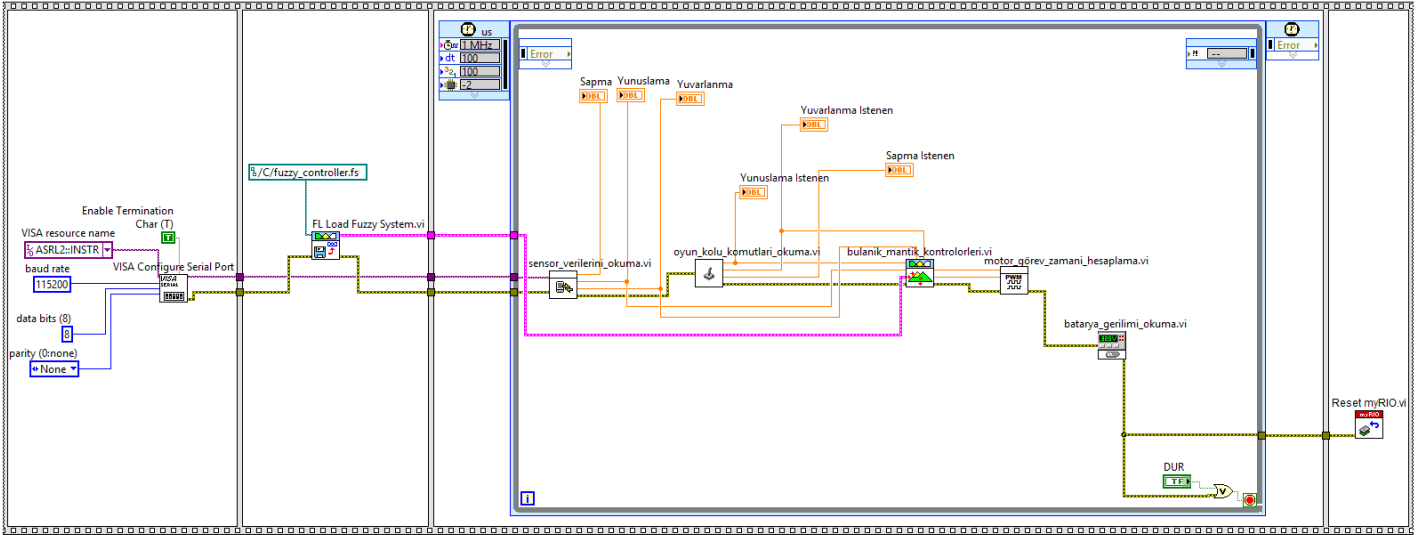
**Tablo 4.5:** myRIO-Arduino seri bağlantı parametreleri

Parametre	Değer
Veri Hız	115200 bps
Veri Bit Sayısı	8
Parite	Yok
Sonlandırma Karakteri	Var



**Şekil 4.20:** myRIO ana program akışı Arduino seri bağlantı ayarı blok parçası (Bkz Ek B)

Ana program akışında arduino ile arasında gerçekleştirilecek olan seri bağlantı parametrelerinin ayarlanmasından sonra myRIO dosyalama sistemi içerisinde daha önceden yüklemiş olduğumuz bulanık mantık kontrol sistemi tanımlama dosyasını açar. Bu bulanık mantık tanımlama dosyası, önceki bölümde anlatılan “Fuzzy System Designer” tarafından hazırlanmış bulanık mantık değişkenleri, üyelik tanımları ve kural tablosu, durulaştırma yöntemi gibi sistemi tanımlayan bilgilerin olduğu dosyadır. Daha sonrasında program akışımız sonsuz döngü içerisinde sensörlerden okuduğu veriler ile kontrol kolundan gelen verileri okur; bu veriler ile bulanık mantık kontrol sistemi çalıştırılır. Bulanık mantık kontrol sisteminin üretmiş olduğu çıkışlar alt sanal ölçüm blokları üzerinde motor hız kontrolörü görev zamanı sinyalleri değerlerine dönüştürülerek myRIO'nun darbe genişlik modülasyonu kanallarına gönderilir. Sistemin dört rotorlu hava aracı üzerinde çalışan ana sanal ölçüm bloğu şekil 4.21’de gösterilmiştir.

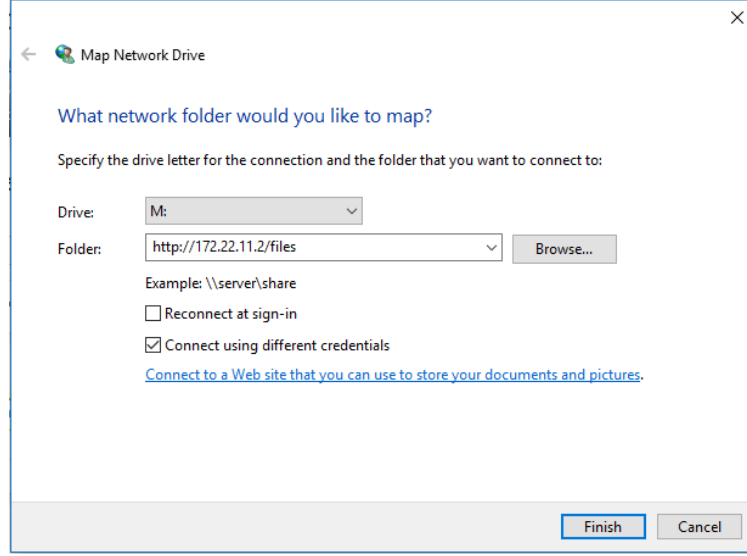


**Şekil 4.21:** Dört rotorlu hava insansız hava aracı ana programı blok diyagramı (Bkz. Ek B)

Bulanık mantık sisteminin “Fuzzy System Designer” üzerinde tasarımından sonra tasarımı yapılan sistemi tanımlayan “\*.fs” uzantılı dosya olarak kaydetmemiz gereklidir. Tasarımı yapılan ve test edilen bulanık mantık sistemi Labview-RT içerisinde bu “.fs” uzantılı dosyayı çalıştırarak sistemi kontrol edecektir. myRIO üzerinde bulanık mantık kontrol sistemi çalıştırılacak ise hazırlamış olduğumuz .fs uzantılı bulanık mantık tanımlama dosyası myRIO dosyalama sistemi üzerinde olması gereklidir. Bu işlem için myRIO usb kablosuyla bilgisayara bağlamak zorunludur. Usb kablosuyla bilgisayara bağlanan myRIO dosyalama sisteminine ulaşmak için birkaç yol vardır. Bizim kullandığımız metod, myRIO’nun bilgisayara kablo ile bağlanmasının ardından bilgisayarım üzerinden ağa bağlı bir disk olarak göstermektir. Bunun için Windows bilgisayar üzerinde Bilgisayarım > Map Network Drive seçeneğinin seçmeliyiz. Ardından açılan pencerede myRIO’nun bilgisayarım üzerinde görüleceği disk adını ve myRIO’nun USBLAN üzerinde aldığı IP adresini belirtmeliyiz. myRIO’nun dosyalama sistemi için IP adresi “172.22.11.2”dir. Şekil 4.22’de ağ sürücüsü bağlantı ayarları görülmektedir.

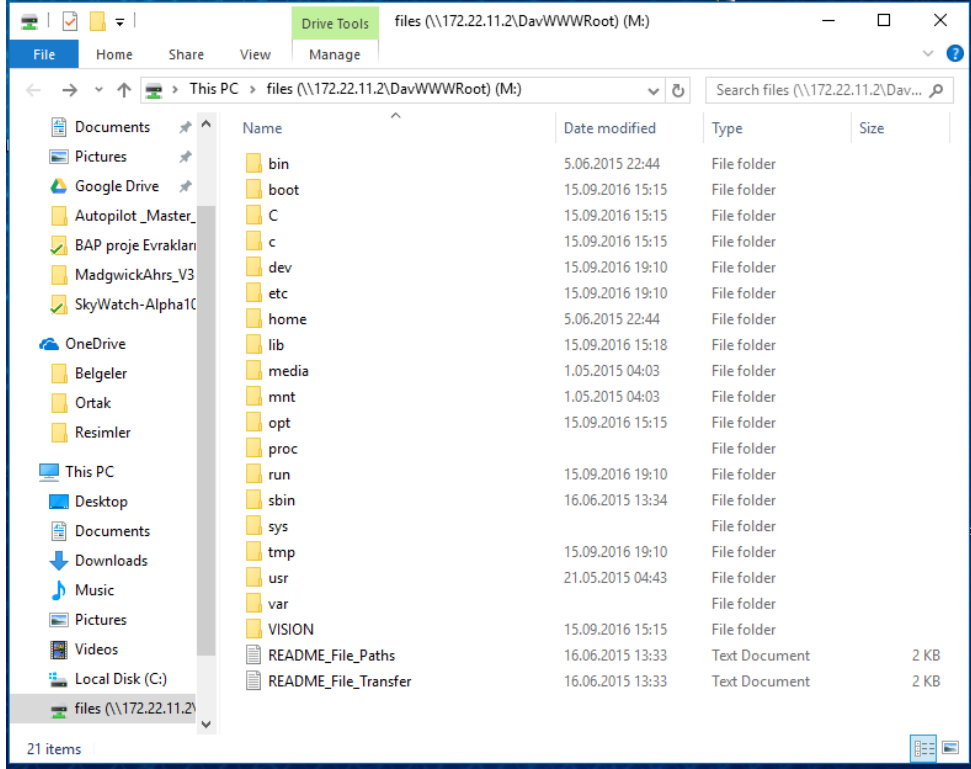
myRIO, üzerinde çalıştırdığı Linux tabanlı işletim sistemi nedeniyle dosyalama sistemi klasik Linux tabanlı işletim sistemlerine benzemektedir. Şekil 4.23’de myRIO dosyalama sistemi içeriğindeki klasörler görülmektedir. myRIO dosyalama sisteminde dikkat edilirse, Linux sistemlerinden alışık olduğumuz “bin”, “boot”, “dev”, “etc”, “home”, “lib”, “media”, “mnt”, “opt”, “sbin”, “sys”, “tmp”, “usr” ve “var” gibi klasörleri bulundurmaktadır. Bu klasörler Labview-RT işletim

sistemi için kullanılan ve Linux tabanlı işletim sistemlerinde de karşılaştığımız aynı ya da benzer işlevleri yerine getiren klasörler ve alt dosyalarıdır.



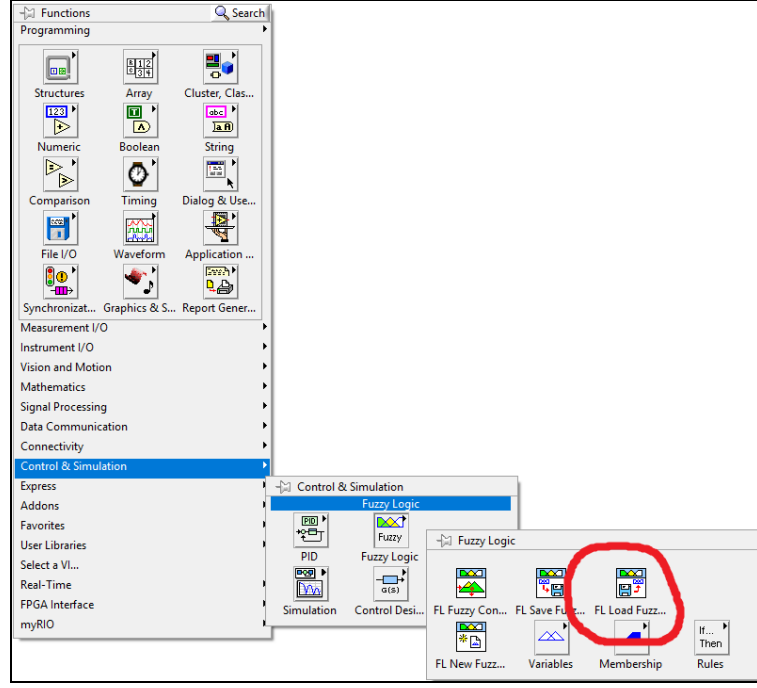
**Şekil 4.22:** myRIO dosyalama sistemi disk olarak açma

myRIO dosyalama sistemi üzerine kayıt edilmiş dosyaların erişimine Labview içerisinde de erişmek mümkündür. Bu sayede, eğer istenirse sistemin çalışması sonucunda ister myRIO üzerindeki host özellikli usb portu üzerindeki flaş belleğe, istenirse de myRIO dosyalama sistemi üzerindeki “usr” klasörü altındaki noktalara veri kayıt etme yani “Log” alma kolay şekilde yapılabilir. Yaptığımız çalışmada dosya yolunun kolay olması açısından hazırlamış olduğumuz bulanık mantık kontrol dosyasını myRIO dosyalama sistemi içerisindeki “C” klasörüne kopyalanmıştır.



Şekil 4.23 : myRIO dosyalama sistemi

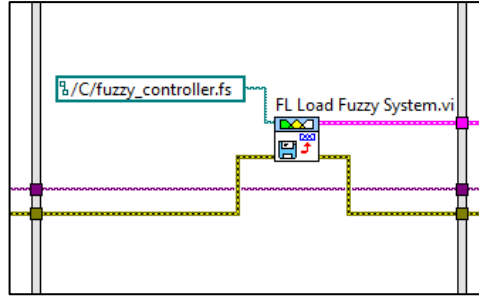
Hazırlanmış olduğumuz Labview program kodları üzerinde bulanık mantık kontrol sisteminin çalıştırılması için öncelikle Labview, bulanık mantık sistemini tanımlayan “.fs” uzantılı dosyayı verilen dosya konumundan açmalıdır. Bu dosya “Fuzzy System Designer” tarafından hazırlanan, bulanık mantık kontrol sistemi tanımlamasını tutan dosyadır. Bu dosya, myRIO dosyalama sistemi üzerinden açılabilmesi için Labview üzerinden çağırılması gereklidir. Bunun için Labview blok diyagramı üzerinde sağ tıklama> Control & Simulation > Fuzzy Logic > Load Fuzzy System.VI hazır sanal ölçüm bloğu seçilmelidir. Bu hızlı sanal ölçüm bloğu; “Fuzzy System Designer” tarafından hazırlanmış olan bulanık mantık sistemi dosyalarını açabilen Labview içerisindeki hızlı sanal ölçüm bloğudur. Şekil 4.24’da bulanık mantık araçlarına ulaşım ve palet üzerindeki hızlı sanal ölçüm blokları görülmektedir.



Şekil 4.24 : Labview bulanık mantık araçları erişimi

Labview, “Fuzzy System Designer” tarafından hazırlanan bulanık mantık sistemlerini kullanabileceği gibi, gerçek zamanlı çalışması sırasında “FL New Fuzzy System.vi” ve , bulanık mantık araçları altında yer alan değişkenler, üyelikler ve kurallar sanal ölçüm blokları kullanarak da kontrol sistemini myRIO’nun çalışması sırasında oluşturabilir. Bu sayede sistem çalışması sırasında gelen yeni değişkenleri ve kullanıcının verdiği sonuçlara göre öğrenebilir yapıya getirilebilir. Sonuç olarak çalışması sırasında öğrenebilen bir kontrol sistemi tasarımı yapılabilir.

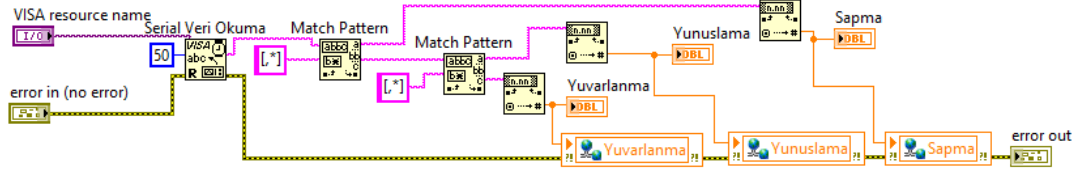
Hazırladığımız ve myRIO dosyalama sistemi üzerine kopyalayadığımız bulanık mantık sistem tanımlama dosyamızın açılması için Labview araçları üzerindeki “FL Load Fuzzy System.vi” hızlı sanal ölçüm bloğu kullanılmaktadır. Şekil 4.25’de bu dosyalama sistemi üzerinde “C” klasörüne ulaşan ve “fuzzy\_controller.fs” isimli dosya yolunu hızlı sanal blok diyagramına argüman olarak atayan blok görülmektedir.



**Şekil 4.25** : Bulanık mantık dosyasının myRIO üzerinde açan hızlı sanal ölçüm bloğu (Bkz. Ek B)

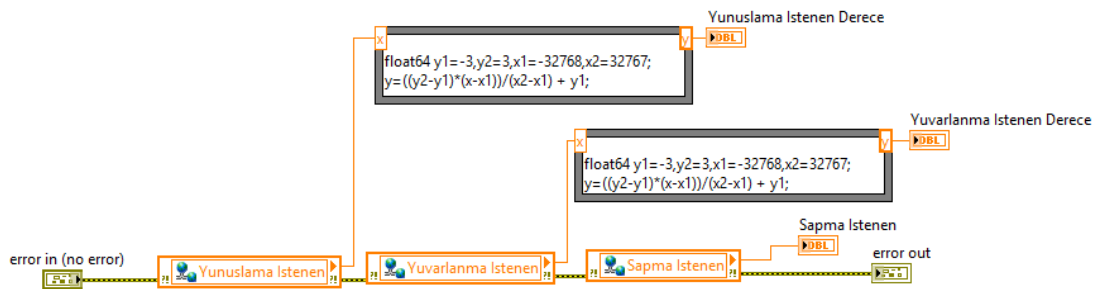
Ana program akışımızda flat sequance structure içerisindeki sonraki bölüm ise dört rotorlu hava aracının sonsuz döngü içerisinde; sensör verilerini okuduğu, kontrol kolundan gelen verileri okuduğu ve bu verileri bulanık mantık kontrol sistemine girdi olarak verdiği, çıkan sonuçlara göre de quadrotor pwm sinyallerini oluşturduğu bölümdür.

Sensörden alınan veriler virgül ile ayrılmış karakter dizilerinden oluşmaktadır. myRIO; arduino tarafından üretilen bu karakter dizilerini virgül karakterinden bölerek yuvarlanma, yunuslama ve sapma verilerini ayrıştırır. Hazırladığım “sensör\_verilerini\_okuma.vi” alt bloğu şekil 4.26’da görülmektedir. Ana program akışı başlangıcında arduino ile seri bağlantının ayarlanması sonrasında “VISA resource name” isimli değişken parametresi sensör okuma alt bloğuna argüman olarak verilir. Bu veri içerisine tanımlanan seri port adı, saniyedeki veri hızı, veri bit genişliği ve pariti durumu gibi parametreler okuma alt bloğuna argüman olarak atanmış olur. Her bir seri port okuması sonrasında 50 karakter dizisine sahip veri “,” karakterlerinden ayrıştırılır. Arduinodan gönderilen örnek bir karakter dizisi “332.554,23.345,-15.435” şeklinde gelmektedir. Karakter olarak gönderilen bu değerlerin virgüllerle ayrılmış halleri “sapma,yunuslama,yuvarlanma” şeklindedir. myRIO, okuduğu karakter dizisini virgül karakterlerinden böler, daha sonra karakter dizisi şeklindeki veriyi kayan noktalı sayı cinsine çevirir. Sensör okuma bloğunun sonunda örnek olarak verilen değerler sapma:332.554, yunuslama : 23.345,yuvarlanma : -15.435 şeklinde elde edilir.



Şekil 4.26: myRIO sensör verisini okuma alt bloğu (Bkz. Ek C)

Sensör verilerinin okunmasından sonra kontrol kolu verilerinin okunması ve açı cinsinden bir değere çevrilmesi yapılmaktadır. Sistem tasarımını yaparken kontrol kolundan alınan verilerin  $\pm 3^\circ$  olarak değişmesi tanımlanmıştır. Kontrol kolu okuması sırasında -32768 ile 32767 arasında değişen değerlerde hava aracının kontrol sistemine  $-3^\circ$  ile  $+3^\circ$  arasında referans giriş değeri girilmiştir. Bu tanımlama dört rotorlu hava aracının kontrol kolundan gönderilecek referans komutlara karşı daha az duyarlı olması amacıyla bu şekilde belirlenmiştir. Kontrol kolu XY ekseninde en uç noktalarda ürettiği değerler hava aracı üzerinde  $\pm 3^\circ$  referans değeri oluşturacaktır. Bu sayede dört rotorlu hava aracının ani hareketler yapmasının önüne geçilmesi hedeflenmiştir. Programladığımız “kontrol\_kolu\_komutlari\_okuma.vi” alt sanal ölçüm bloğu ve kontrol kolu XY eksenleri verilerinin  $\pm 3^\circ$  dereceye dönüştürülmesi işlemi şekil 4.27’de gösterilmiştir. Kontrol kolu okuma alt blok diyagramında kullanılan değişken değerleri maksimum ve minimum değer aralıkları tablo 4.6’da verildiği gibidir.



Şekil 4.27: “kontrol\_kolu\_komutlari\_okuma.vi” referans komutları okunması, derece cinsine çevirme alt bloğu (Bkz. Ek D)

**Tablo 4.6:** Kontrol kolu okuma alt bloğu değişken değer aralıkları

Değer	Minimum Değeri	Maksimum Değeri
Yunuslama istenen	-32768	32767
Yuvarlanma istenen	-32768	32767
Sapma istenen	-32768	32767
Yunuslama istenen derece	-3	+3
Yuvarlanma istenen derece	-3	+3

Dört rotorlu hava aracımızın bulanık mantık kontrol sistemi yuvarlanma kontrolörü ve yunuslama kontrolörü olarak birbiriyle özdeş iki farklı kontrol sistemi olarak çalışır. Kontrol sistemi giriş değişkenlerinden yunuslama hata ( $\Delta e_p$ ) değeri; istenilen açı değeri ( $\omega_p^d$ ) ile ölçülen açı değeri ( $\omega_p^r$ ) arasındaki fark ile hesaplanmaktadır. Yunuslama ( $\Delta e_p$ ) ve yuvarlanma ( $\Delta e_r$ ) açılarının hata değerleri ve hatadaki değişim değerleri hesaplaması denklem 4.2 - 4.5’de verildiği gibi hesaplanmıştır.

$$\Delta e_p = \omega_p^d - \omega_p^r \quad (4.2)$$

$$\Delta e_r = \omega_r^d - \omega_r^r \quad (4.3)$$

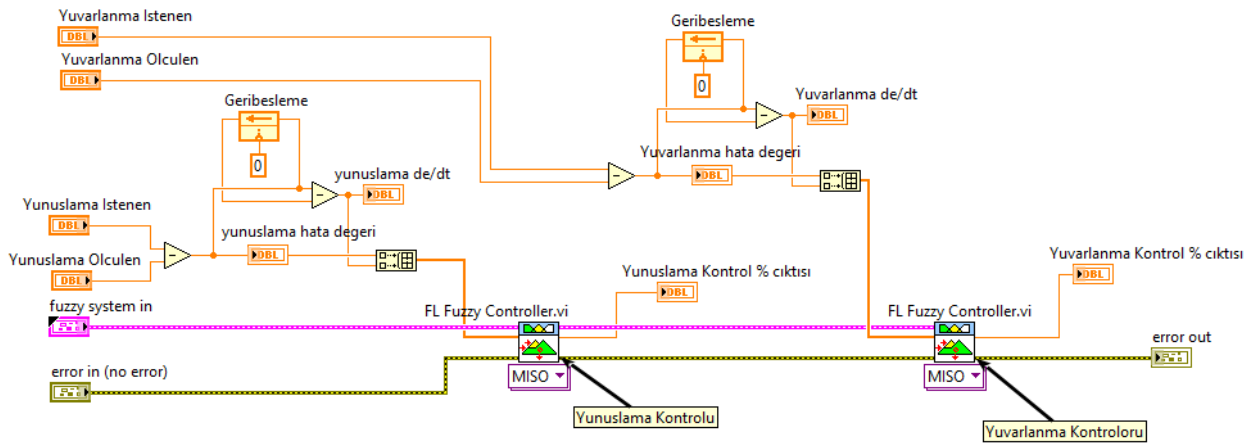
$$\frac{de_p}{dt} = \frac{e_p(t) - e_p(t-1)}{\Delta t} \quad (\Delta t : 1ms) \quad (4.4)$$

$$\frac{de_r}{dt} = \frac{e_r(t) - e_r(t-1)}{\Delta t} \quad (\Delta t : 1ms) \quad (4.5)$$

$\frac{de}{dt}$  (hatadaki değişim) değeri ise önceki fonksiyon çağrılma zamanı ile sonraki fonksiyon çağrılma zamanı arasındaki e değerinin arasındaki farktan hesaplanır. Bu hesap için Labview araçları üzerinde yer alan geribesleme düğümü kullanılmıştır. Bu sayede önceki durumdaki e değeri için geçici bir değişken oluşturmak durumunda kalınmadan değerler önceki işlem zamanındaki değeri sonraki zamanda tutulabilmektedir. Şekil 4.28’de bulanık mantık kontrol sistemini çalıştıran “bulanik\_mantik\_kontrolorleri.vi” alt sanal ölçüm blok diyagramı görülmektedir. Bulanık mantık kontrolörleri hızlı blok diyagramlarını çalıştıran bu alt blok



diyagramı, giriş argumanları olarak yuvarlanma istenen değeri, yuvarlanma ölçülen değeri, yunuslama istenen değeri, yunuslama ölçülen değeri, bulanık mantık sistem tanımlama parametresi ve myRIO üzerinde hata durumu bilgilerini giriş olarak alır. Fark operatörünü kullanarak yunuslama ve yuvarlanma açıları hata değerlerinin hesaplanması sağlanır. Daha sonra hesaplanan hata değeri tekrar fark operatörü ile kendisinin bir önceki durumdaki değerinden çıkartılır. Bu işlem geribesleme operatörü aracılığı ile yapılır. Geribesleme operatörü başlangıç durumu olarak “0” değeri ile hesaplamaya başlar. Daha sonraki ötelemelerde (t-1) zamanındaki kendi değerini hata değerinden çıkartarak de/dt değeri hesaplanmış olur.



Şekil 4.28 : Bulanık mantık kontrolörü alt blok diyagramı (Bkz. Ek E)

Hesaplanan yunuslama ve yuvarlanma hata değerleri ile hatadaki değişim değerleri bulanık mantık kontrolörü sistemi girdi parametreleri olarak verilir. Bulanık mantık kontrolörünü çalıştıran “FL Fuzzy Controller.vi” hızlı blokları oluşturduğumuz bulanık mantık sistemine göre çoklu giriş tek çıkış olarak çalıştırılmıştır. Giriş değerleri olan hata ve hatadaki değişim değerleri bulanık mantık kontrol bloklarına argüman olarak verilmeden önce birleştirilerek veri dizisi haline dönüştürülmektedir. Bulanık mantık kontrolörler çıkışı bulanık mantık sistemi tasarım dosyasında tanımladığımız PWM%OUT çıkış değişkeni içerisinde -3 ile +3 arasında tanımlanmıştır. Bulanık mantık kontrolörünü çalıştıran hızlı blok diyagramları; verilen giriş parametrelerine göre “yunuslama kontrol % çıktısı” ve “yuvarlanma kontrol çıktısı” üzerinde -3 ile +3 arasında bir değer oluşturur. Bu değerler “bulanık\_mantık\_kontrolörleri.vi” alt blok diyagramı çıkış parametreleri

olarak “motor\_gorev\_zamani\_hesaplama.vi” alt blok diyagramına argüman olarak gönderilir.

Yunuslama bulanık mantık kontrolörü ve yuvarlanma bulanık mantık kontrolörü tarafından üretilen “yuvarlanma kontrol % çıktısı” ve “yunuslama kontrol % çıktısı” değerleri, ana program akışımızda sonraki alt program bloğu olan “motor\_görev\_zamani\_hesaplama.vi” tarafından motor hız kontrol devrelerine gönderilecek olan darbe genişlik modülasyonu sinyallerine dönüştürür. Darbe genişlik değerlerine döndürülmeden önce yunuslama ve yuvarlanma kontrolörlerinden gelen yüzdelik motor devirlerini arttırma-azaltma komutu ile pilot tarafından gönderilen yüzdelik gaz değeri dört rotorlu aracın şasi üzerindeki motor dizilimine göre tanımlanmış matris kullanılarak bindirilir. Oluşturduğumuz dört rotorlu insansız hava aracı için kontrol sinyallerinin motorlara uygulanması denklem 4.6, - 4.9’da tanımlandığı gibidir.

$$\text{Sol Ön Motor Devir} = \text{Gaz} + \text{Yunuslama Kontrol Sinyali} + \text{Yuvarlanma Kontrol Sinyali} \quad (4.6)$$

$$\text{Sağ Ön Motor Devir} = \text{Gaz} + \text{Yunuslama Kontrol Sinyali} - \text{Yuvarlanma Kontrol Sinyali} \quad (4.7)$$

$$\text{Sağ Arka Motor Devir} = \text{Gaz} - \text{Yunuslama Kontrol Sinyali} - \text{Yuvarlanma Kontrol Sinyali} \quad (4.8)$$

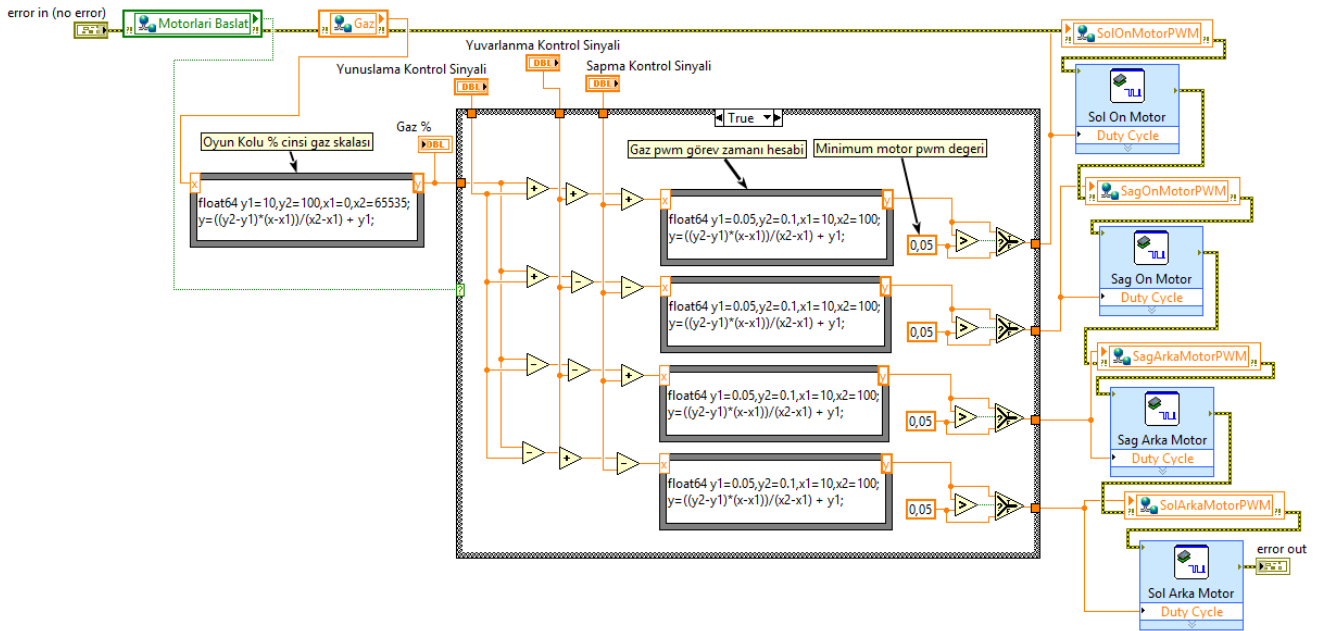
$$\text{Sol Arka Motor Devir} = \text{Gaz} - \text{Yunuslama Kontrol Sinyali} + \text{Yuvarlanma Kontrol Sinyali} \quad (4.9)$$

Yüzdelik değer olarak gelen gaz, yunuslama, yuvarlanma kontrol sinyallerinin pwn kanallarına uygulanmadan önce elektronik hız kontrol devrelerinin anlayabileceği uygun frekansda ve görev zamanında sayısal sinyallere dönüştürülmesi gerekmektedir. Elektronik hız kontrol devreleri 50 Hz frekansında 1 ile 2 ms arasında değişen görev zamanı sinyallerinde motorları minimum devir ile maksimum devir aralığında döndürebilmektedir.

MyRIO ve Labview darbe genişlik modülasyonu için, görev zamanı sinyalini yüzdelik olarak 0 ile 1 arası kayan noktalı sayı tipinde değişken kullanarak ayarlamaktadır. 0 değeri %0 görev zamanına denk gelirken 0.5 değeri %50, 0.8 ise %80 değerine denk gelmektedir. Elektronik hız kontrol devrelerimiz 50 Hz

frekansında ve 1 ile 2 ms genişliğinde darbe sinyalleri ile çalıştırıldığı için 1 ms darbe genişliği için 0.05 değeri, 2 ms darbe genişliği için 0.1 değeri almalıdır.

“motor\_gorev\_zamani\_hesaplama.vi” alt blok diyagramında motor başlat biti gelmesi ile birlikte motorlara minimum olarak 0.05’lik görev darbe genişliği uygulanarak motorların minimum devirde dönmesi sağlanır. Kontrol algoritması sonucu herhangi bir motor görev zamanı 1 ms altına düşecek olursa minimum operatörü sayesinde minimum darbe genişliği değeri 0.05’in altına düşmesi durumunda çıkış değeri olarak o motor 0.05 görev zamanı sinyalini almaya devam edecektir. Bu sayede motor çalıştır biti geldiği sürece motorların düşük devirde de olsa dönmeye devam edeceği garanti altına alınmıştır. Şekil 4.29’da “motor\_gorev\_zamani\_hesaplama.vi” alt blok diyagramı gösterilmiştir.

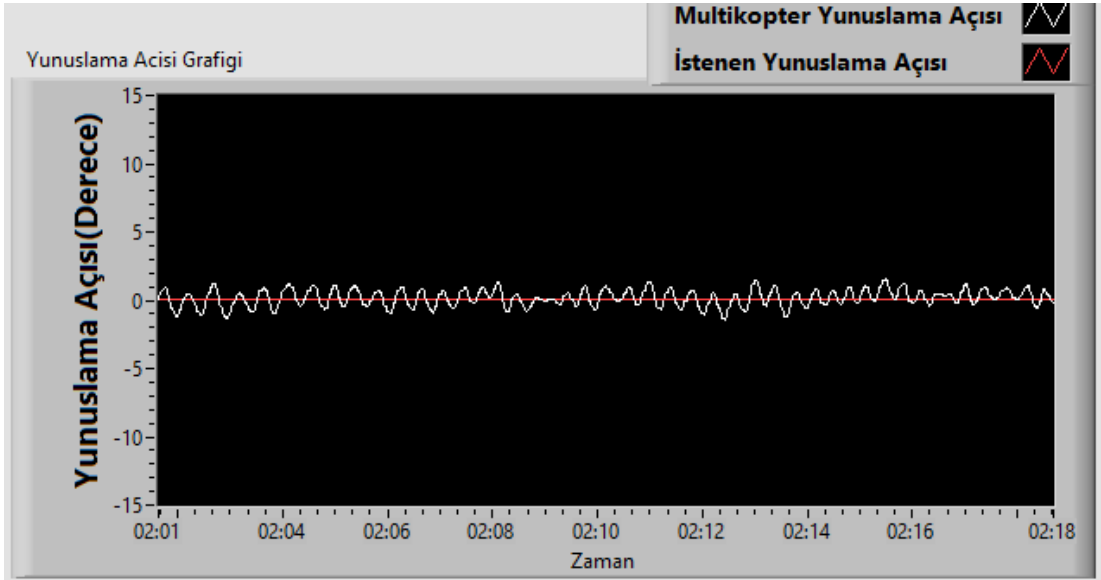


Şekil 4.29: Motor\_gorev\_zamani\_hesaplama.vi alt blok diyagramı (Bkz. Ek F)

## 5. SONUÇ VE ÖNERİLER

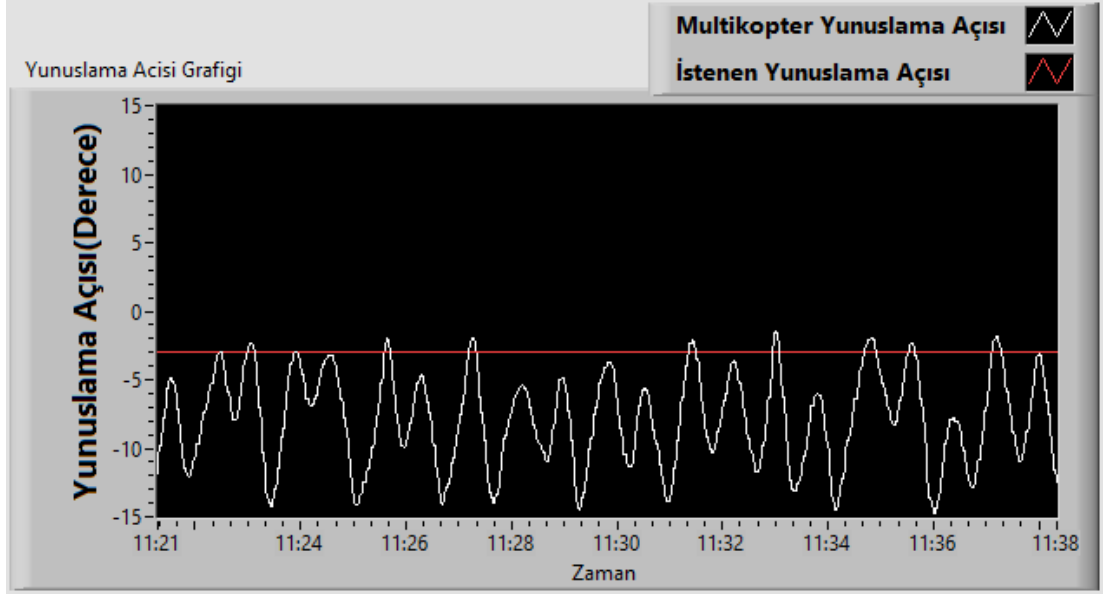
Yapılan bu tez çalışmasında dört rotorlu hava aracının uçuş denge kontrolünü sağlayan kontrol sistemi Labview kullanılarak tasarlanmış ve myRIO donanımının sağlamış olduğu FPGA ve Linux tabanlı Labview-RT üzerinde çalıştırılması sağlanmıştır. myRIO geliştirme donanımı hazırlanan hava aracı donanımı üzerinde yerleştirilmiş sistem üzerinde denenmiştir.

Yapılan çalışma sonucunda yunuslama açısı üzerinde pilot tarafından gönderilen sıfır referans değere karşın dört rotorlu hava aracı sisteminin davranışı şekil 5.1’de görüldüğü gibi gözlemlenmiştir.

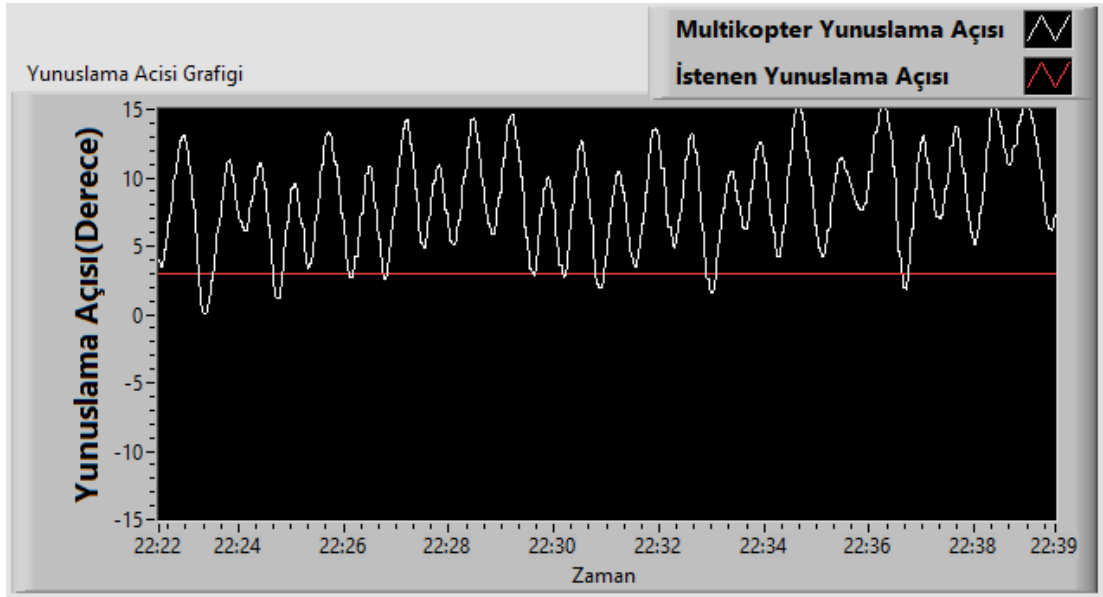


Şekil 5.1: Dört rotorlu insansız hava aracı kontrolör sistemi yunuslama açısı 0 derece cevabı

Uçuş sistemine yunuslama açısı olarak negatif yönde 3 derece referans sinyal verildiğinde sistemin cevabı şekil 5.2’deki gibi gözlemlenmiştir. Hava aracına pozitif yönde 3 derecelik referans yunuslama açısı sinyali gönderildiğinde sistemin cevabı şekil 5.3’da görüldüğü şekilde gözlemlenmiştir.



Şekil 5.2: Dört rotorlu insansız hava aracı kontrolör sistemi yunuslama açısı -3 derece cevabı



Şekil 5.3: Dört rotorlu insansız hava aracı kontrolör sistemi yunuslama açısı +3 derece cevabı

Sonuçlar incelendiğinde dört rotorlu hava aracının bulanık mantık kontrolörü yunuslama açısı sıfır olması durumunda stabil kalabildiği, ancak sistemde belirli bir miktar salınımların olduğu görülmektedir. Motor gaz değeri yükseltilmesi durumunda bu salınımların miktarları arttığı, sistemin kararsızlaştığı görülmüştür. Sistem üzerine referans sinyal olarak +3 derecelik yunuslama açısı verildiğinde sistemin hedef sinyal seviyesini aşarak yüksek değerler göstermiştir. Hava aracı üstünde durduğu test ayakları üzerine çarparak geri istenilen konuma sıçramalar yaptığı görülmüştür. Benzer şekilde negatif yönde verilen referans sinyal sonucunda

hava aracının aşırı tepki verdiği görülmüştür. Oluşan salınımlar, kural tablosu ve üyelik fonksiyonların ayarlanması ile düzeltilebilecek sorunlar olup sistemin hiçbir matematiksel modeline bağlı kalınmaksızın belirli seviyelerde kontrol edilebilir olması sağlanmıştır.

Gerçekleştirilen bulanık mantık kontrol sistemi ile sistemin matematiksel modeline gerek duymadan, pilot uzman bilgisine dayanarak kontrolör sistemi tasarımı yapılmıştır. Uygulamada bulanık mantık kontrol sistemi seçilmesi ile kontrol sistemi üzerindeki değişiklik kolay şekilde hızlıca yapılabılır, bilgisayar üzerinde çıkışlar direk olarak test edilebilmektedir. Daha sonra dört rotorlu hava aracı üzerindeki myRIO üzerine programlamalar da hızlı şekilde gerçekleştirilebilmektedir. Bu şekilde tüm sistem tasarımının tek bir program üzerinde yapılması ile; sistemin analizi, yeniden program oluşturulması, sistemin simülasyonu ve programlanması hızlı şekilde gerçekleştirilebilmiştir.

Matlab gibi bilgisayar üzerinde çalışan simülasyon yazılımları yerine Labview kullanmak hem sistemin programlanması hem de sistem parametrelerinin gerçek donanım üzerinde izlenebilmesine olanak tanımıştır.

myRIO donanımı ile FPGA ve ARM tabanlı işlemcinin birlikte çalışması sonucu hem paralel işlem yapabilme yeteneği, hem de kablosuz bağlantı, USB cihaz, USB host gibi bağlantılara sahip geliştirme kartı kullanılmıştır. Bu sayede başka donanımlar ile USB, wireless gibi gelişmiş haberleşme protokolleri kullanılabilmiş, FPGA ile de I2C, UART ve PWM gibi basit haberleşmeler de hızlı şekilde gerçekleştirilebilmiştir.

myRIO ve Labview grafiksel programlama kullanılması sonucu lengüistik programlama dillerinde sayfalarca sürececek kodlamalar kolay şekilde yapılabilmektedir. Yapılacak küçük değişiklikler ve eklemeler grafik programlama içerisinde kolaylıkla gerçekleştirilebilmektedir.

Gömülü sistem tasarımında en çok zorlanılan kısım olan ara yüz oluşturma ve okunan verilerin grafiksel ara yüz ile kullanıcıya gösterilmesi Labview üzerinde kolaylıkla yapılabilmektedir. Labview kullanıcılarının oluşturduğu topluluklar

üzerinden indirilen kütüphaneler ile farklı görsel araçlar, kendi projemizde kolaylıkla kullanılabilir.

Yapılan tez projemizde dört rotorlu hava aracı kontrol algoritması üzerinde ve Labview programlanması üzerinde daha çok durulmuştur. Sistemin mekanik tasarımı ve sağlamlığı üzerinde çalışmalar az kalmıştır. Sistemin tek eksen üzerinde testi sırasında görülmüştür ki hazırlanan test düzeneği üzerindeki sürtünmelerden ve hava aracı üzerindeki yük dengesizliğinden aşırı derecede etkilenmiştir. Bu donanım tasarımımyRIO ve batarya hava aracının gövdesi üzerine plastik kelepçeler ile bağlanmıştır. Sonraki yapılacak çalışmalarda dört rotorlu hava aracı gövdesi yeniden tasarlanabilir, myRIO'nun montaj yuvalarını kullanabilecek şekilde daha sağlam bir gövde tasarımı gerçekleştirilebilir. Hava aracının en kırılgan parçası olan pervaneleri korumak için pervane çevrelerine koruyucu halkaların tasarlanması önerilmektedir.

Kontrol sistemi üzerinde optimizasyonlar ile kontrol sistemi daha stabil olacak şekilde geliştirilebilir.

Ayrıca myRIO üzerindeki USB host girişi kullanılarak USB kamera cihazları ve FPGA kullanılarak görüntü işleme yapılabilir. Bu sayede görüntü üzerinden hedef takibi, insansız uçuş gibi görevleri otonom olarak gerçekleştirebilecek sistem tasarımı yapılmış olur.

## 6. KAYNAKLAR

Passino, K. M., Yurkovich, S., *Fuzzy Control*, Addison-Wesley, (1998).

Erginer, B., Altuğ, E., “Design And Implementation of a Hybrid Fuzzy Logic Controller for a Quadrotor VTOL Vehicle”, *International Journal of Control, Automation, and Systems*, 10(1), 61-70, (2012).

Tosun, D. C., “Dört Rotorlu Bir İnsansız Hava Aracı İçin Adaptif Kontrol Sistem Tasarımı”, Yüksek Lisans Tezi, *Anadolu Üniversitesi Fen Bilimleri Enstitüsü*, Havacılık Elektrik Elektronik Anabilim Dalı, Eskişehir, (2015).

Baerveldt, A. J., Klang, R., “A Low-cost and Low-weight Attitude Estimation System for an Autonomous Helicopter” *Proceedings of IEEE International Conference on Intelligent Engineering Systems*, Budapest, Hungary, 391-395, (1997).

Raza, S. A., “Design and Control of a Quadrotor Unmanned Aerial Vehicle”, Master's Thesis, *University of Ottawa, School of Information Technology and Engineering, Faculty of Engineering*, , Canada, (2010).

Nicol, C., Macnab, C. J. B., Ramirez-Serrano, A., “Robust Neural Network Control of a Quadrotor Helicopter”, *2008 Canadian Conference on Electrical and Computer Engineering*, Niagara Falls, Ontario, Canada, 1233-1238, (2008).

Nicol, C. E., “A Robust Adaptive Neural Network Control for Quadrotor Helicopter”, Master's Thesis, *University Of Calgary, Department of Electrical and Computer Engineering*, Alberta, Canada, (2010).

Abeywardena, D. M. W., Amaratunga, L. A. K., Shakoor, S. A. A., Munasinghe, S.R., “A Velocity Feedback Fuzzy Logic Controller for Stable Hovering of a Quad Rotor UAV”, *2009 International Conference on Industrial and Information Systems (ICIIS)*, Sri Lanka, 558-562, (2009).

Mehranpour, M. R., Emamgholi, O., Shahri, A. M., and Farrokhi, M., “A New Fuzzy Adaptive Control for a Quadrotor Flying Robot”, *13th Iranian Conference on Fuzzy Systems (IFSC)*, (2013).

Fakuran, F., Menhaj, M.B., and Mohammadi, A., “Design of a Fuzzy Controller by Minimum Controlling Inputs for a Quadrotor”, *2nd RSI/ISM*



*International Conference on Robotics and Mechatronics*, Tehran, Iran, 619-624, (2014).

Castillo, P., Dzul, A., Lozano, R., “Real-Time Stabilization and Tracking of a Four-Rotor Mini Rotorcraft”, *IEEE Transactions on Control Systems Technology*, 12, (4), 510-516, (Temmuz 2014).

Pedro, J. O., Mathe, C., “Nonlinear Direct Adaptive Control of Quadrotor UAV Using Fuzzy Logic Technique”, *2015 10th Asian Control Conference (ASCC)*, 1-6, (2015).

Tzafestas, S., Papanikolopoulos, N. P., “Incremental Fuzzy Expert PID Control”, *IEEE Transactions on Industrial Electronics*, 37, (5), 365-371, (Ekim 1990).

Sardar, S., Kadri, M. B., “Autonomous Control of a Quadcopter via Fuzzy Gain Scheduled PD Control”, *2014 12th International Conference on Frontiers of Information Technology*, Islamabad, Pakistan, 73-78, (2014).

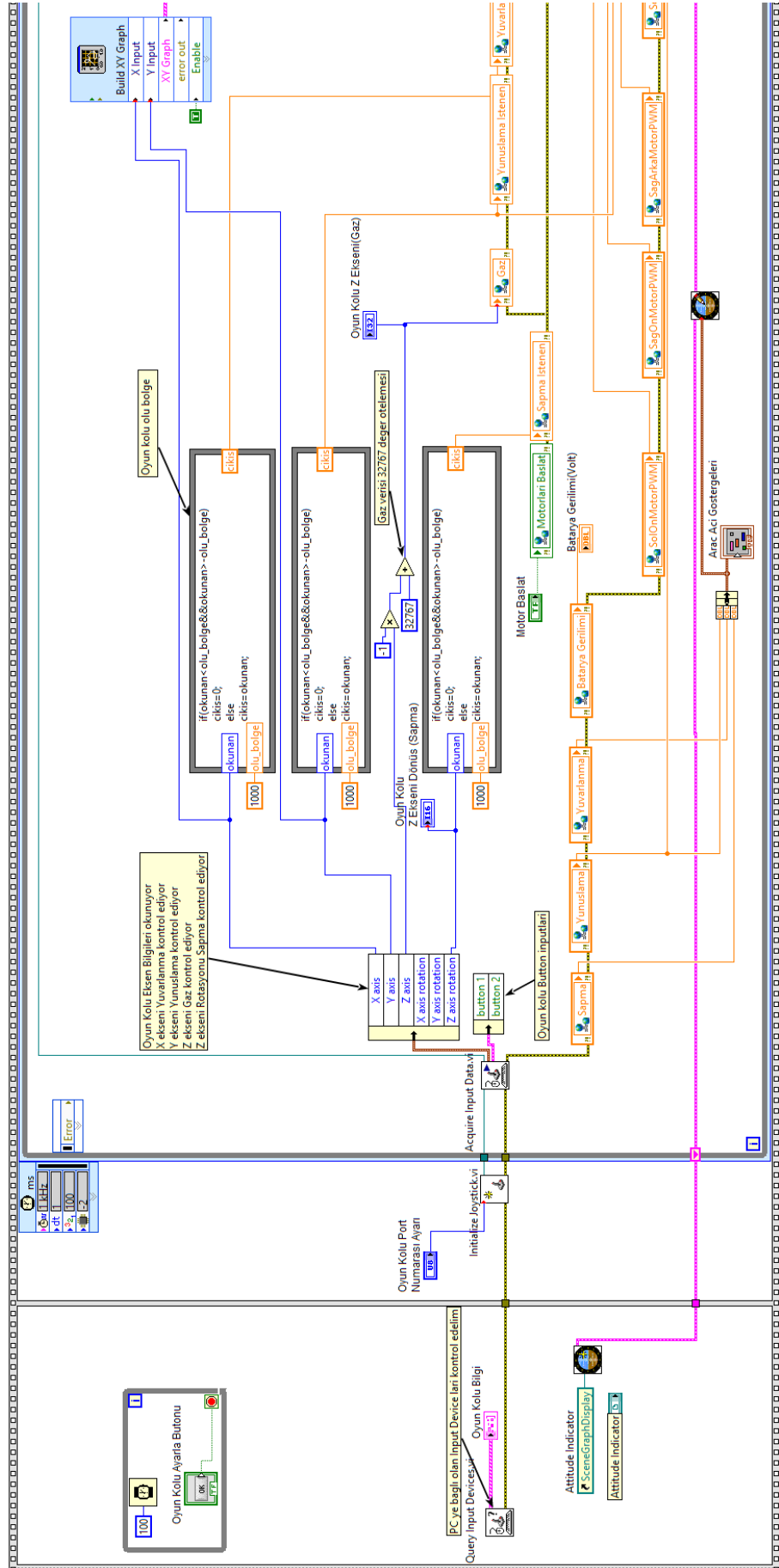
Magnussen, Ø., Skjønhaug, K. E., “Modelling, Design and Experimental Study for a Quadcopter System Construction”, Master’s Thesis, *University of Agder, Faculty of Technology and Science, Department of Engineering*, Norway, (2011).

National Instruments, “User guide and specifications NI myRIO-1900[Online]”,(Mays 2016), “<http://www.ni.com/pdf/manuals/376047c.pdf>”

Xilinx , “Zynq-7000 All Programmable SoC Data Sheet: Overview[Online], (Haziran 2017), “[https://www.xilinx.com/support/documentation/data\\_sheets/ds190-Zynq-7000-Overview.pdf](https://www.xilinx.com/support/documentation/data_sheets/ds190-Zynq-7000-Overview.pdf)”

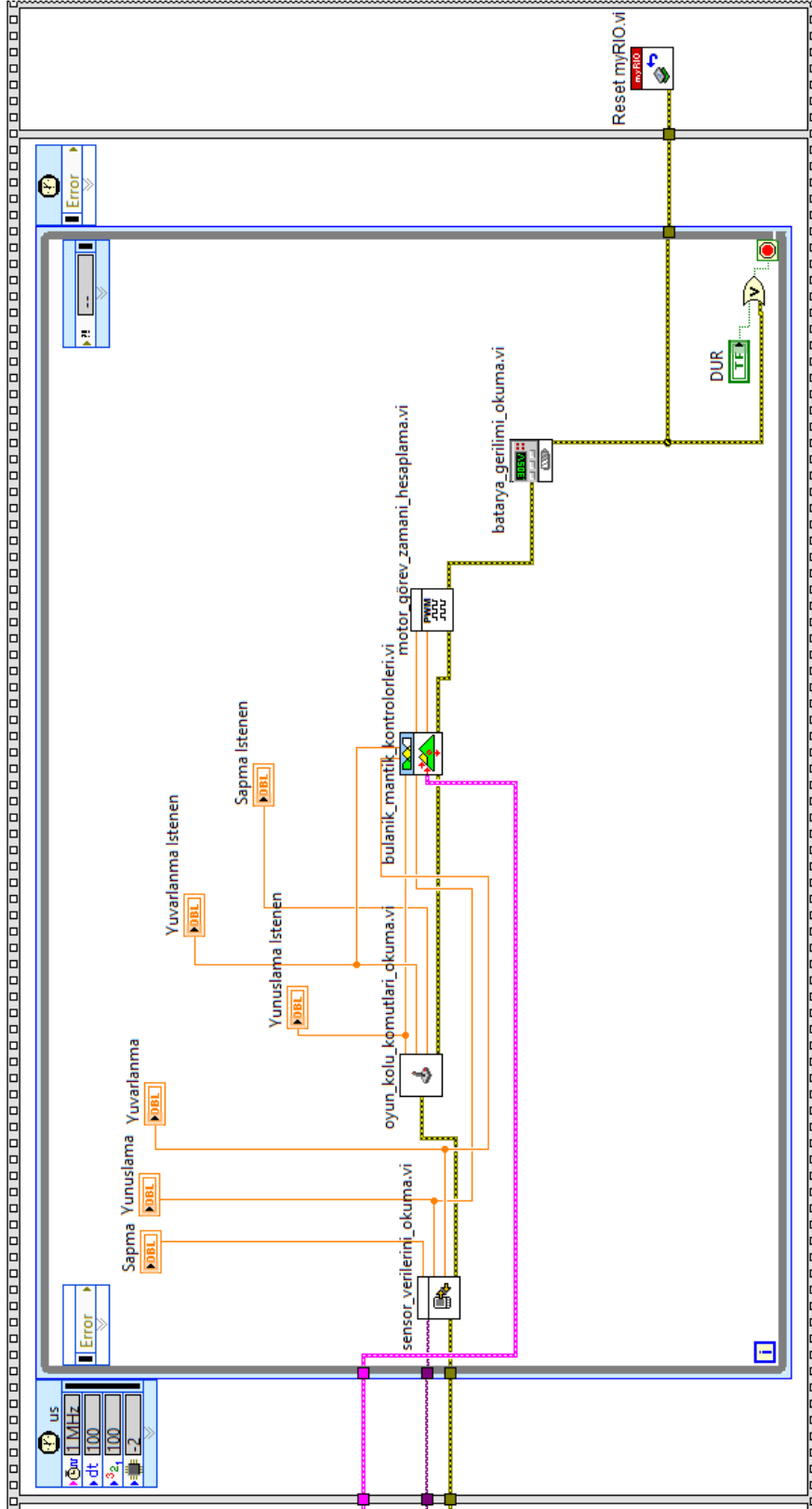
# **EKLER**

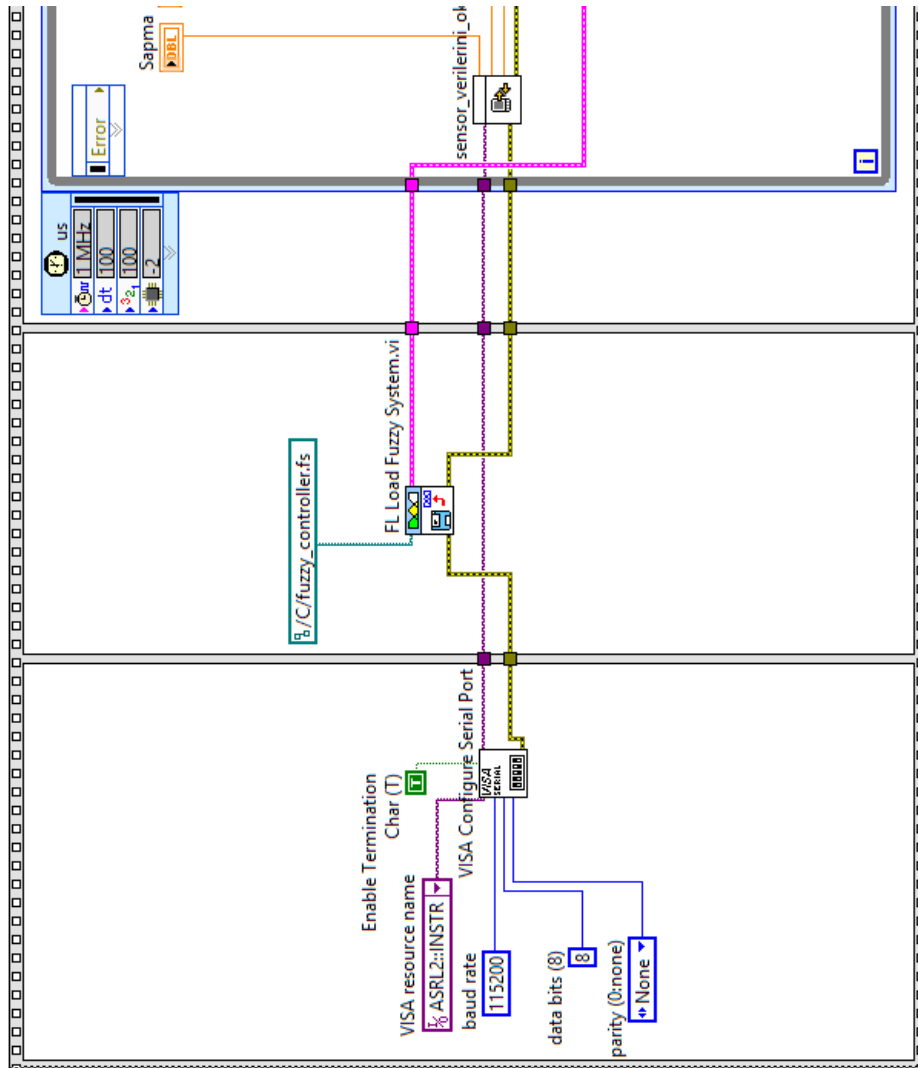




## EK B

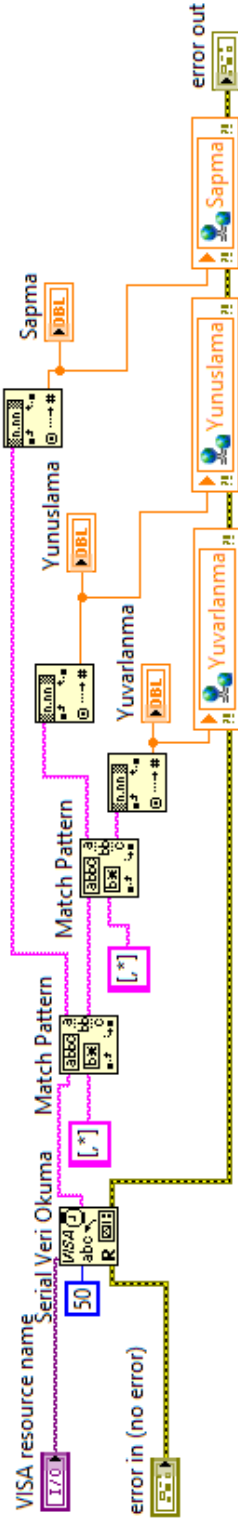
myRIO üzerinde çalışan ana program blok diyagramı (2 Sayfa)





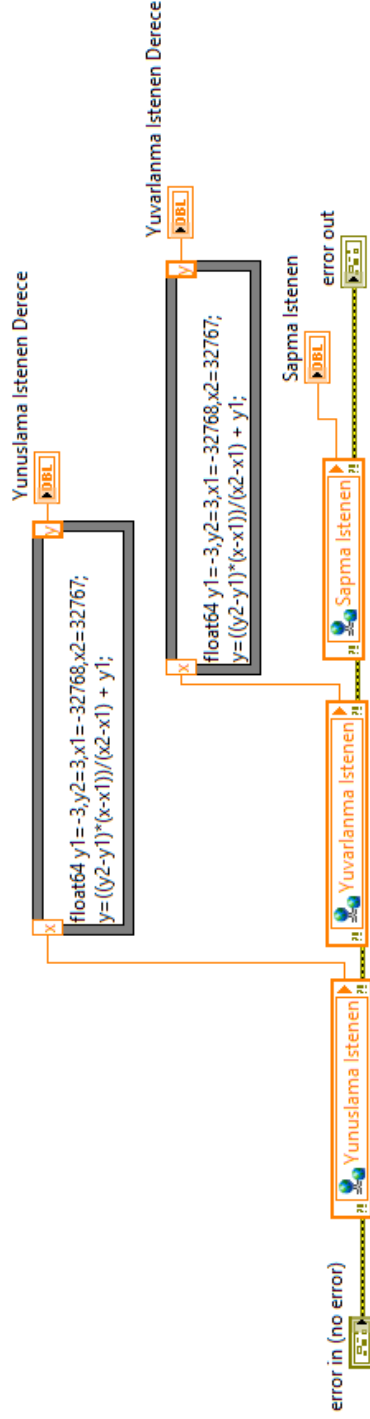
## EK C

myRIO sensör verisini okuma alt bloğu



## EK D

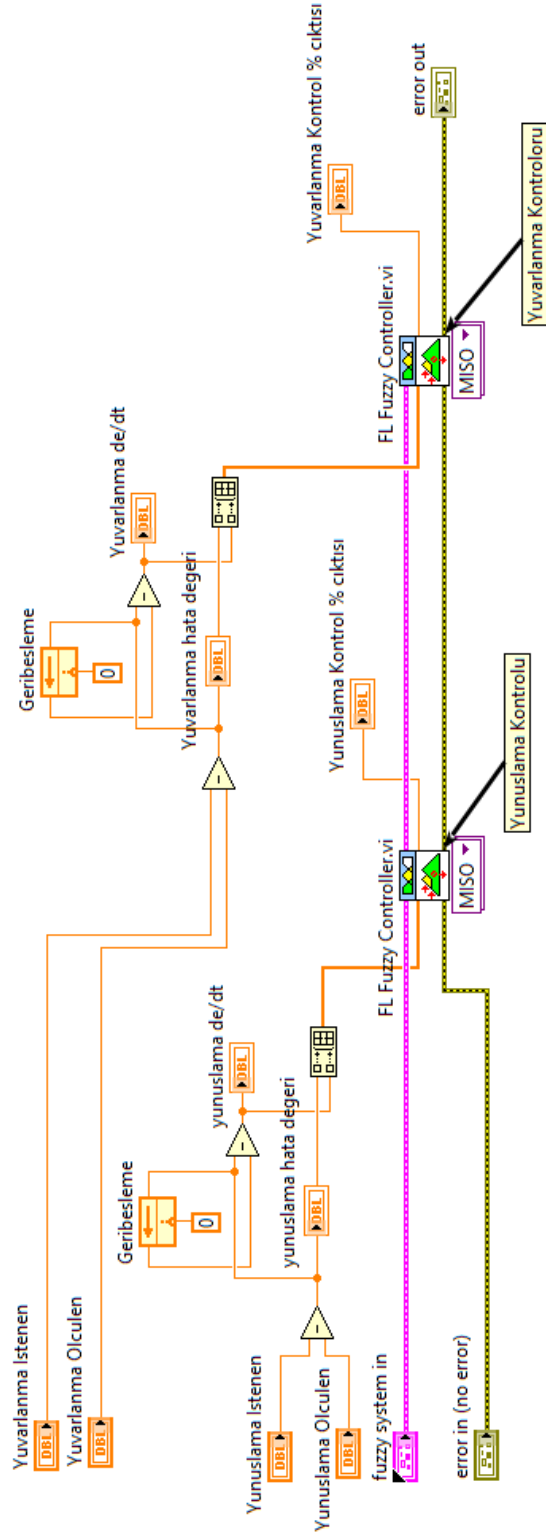
“kontrol\_kolu\_komutlari\_okuma.vi” referans komutları okunması, derece cinsine çevirme alt bloğu





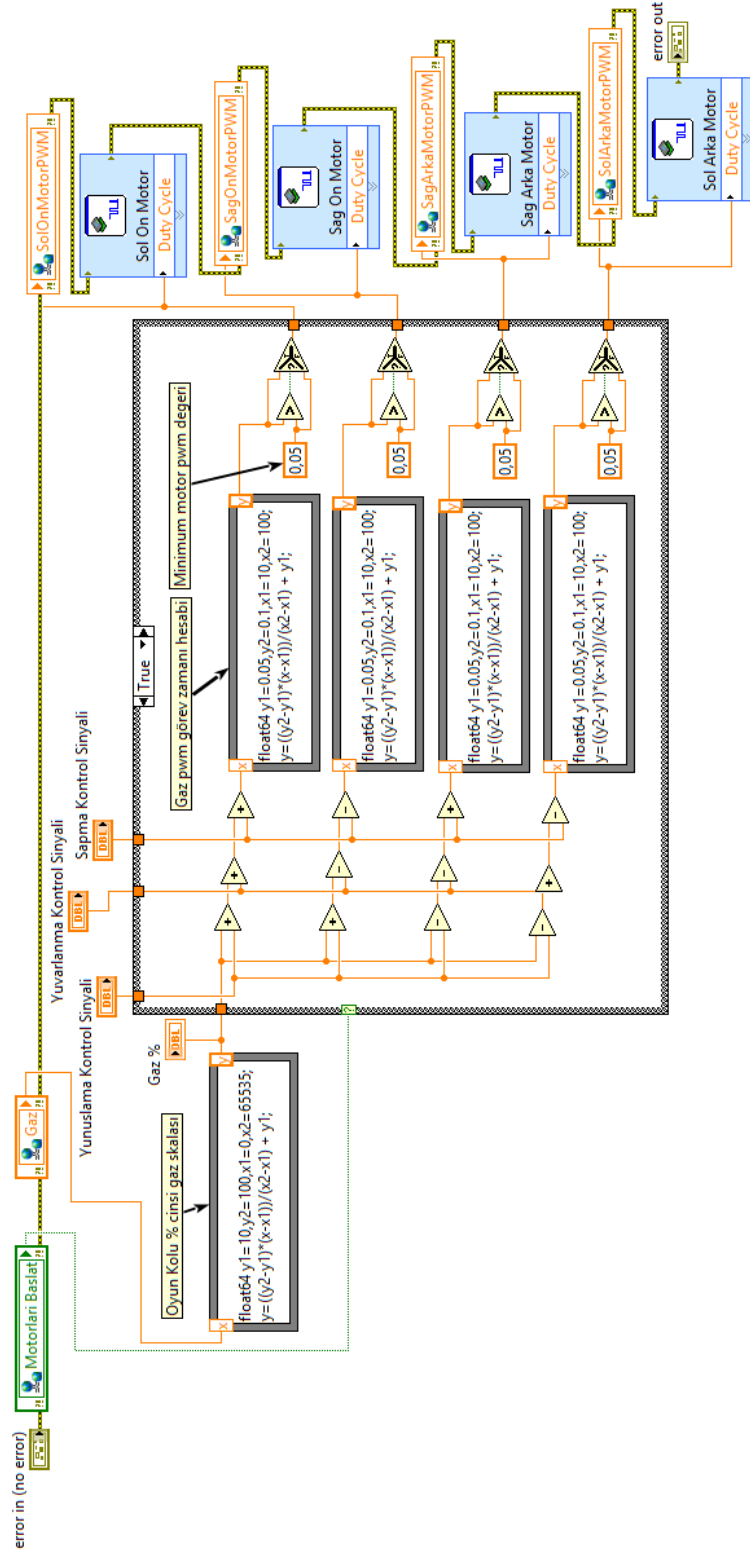
## EK E

Bulanık mantık kontrolörü alt blok diyagramı



# EK F

Motor\_gorev\_zamani\_hesaplama.vi alt blok diyagramı



## ÖZGEÇMİŞ

Adı Soyadı : Mustafa Manisalı

Doğum Yeri ve Tarihi : Denizli 06/05/1989

Lisans Üniversite : Pamukkale Üniversitesi

Elektronik posta : mstfmanisali@gmail.com

İletişim Adresi : Fesleğen Mah. 988 Sk. No.1 Pamukkale /  
Denizli.