T.C.
MANİSA CELAL BAYAR ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ



YÜKSEK LİSANS TEZİ
MAKİNE VE İMALAT ANABİLİM DALI
MAKİNE TEORİSİ VE DİNAMİĞİ BİLİM DALI

MATLAB OPC ARAÇ KUTUSUNU KULLANARAK DOĞRU AKIM MOTORU KONTROLÜ

Burak Çağrı DUBA

Danışman
Dr. Öğr. Üyesi Serkan ÇAŞKA

MANİSA–2024

BURAK
ÇAĞRI
DUBA

MATLAB OPC ARAÇ KUTUSUNU KULLANARAK DOĞRU AKIM MOTORU KONTROLÜ

2024

TAAHHÜTNAME

Bu tezin Manisa Celal Bayar Üniversitesi Lisansüstü Eğitim Enstitüsü Makine ve İmalat Mühendisliği Anabilim Dalı Makine Teorisi ve Dinamiği Bilim Dalı Bölümü'nde, akademik ve etik kurallara uygun olarak yazıldığını ve kullanılan tüm literatür bilgilerinin referans gösterilerek tezde yer aldığını, , tamamen kendi çalışmam olduğunu, her alıntıya kaynak gösterdiğimi, tezin yazımında akademik ve etik kurallara aykırı herhangi bir yapay zeka ve program kullanmadığımı beyan ederim. Bu tez çalışmasının konusu nedeniyle çalışmada TıaPortal, KEPServerEX, UaExpert ve MATLAB/Simulink programı kullanılmıştır.

Burak Çağrı DUBA

ÖZET

Yüksek Lisans Tezi

Burak Çağrı DUBA
Manisa Celal Bayar Üniversitesi
Lisansüstü Eğitim Enstitüsü
Makine ve İmalat Anabilim Dalı

Danışman: Dr. Ögr. Üyesi Serkan ÇAŞKA

Bu çalışmada bir doğru akım motorunun hızını ve konumunu kontrol etmek için oransal-integral-türev (PID) kontrolör kullanılmıştır. Kontrol süreci programlanabilir lojik kontrolör (PLC) kullanılarak gerçekleştirilmiştir. Denetleyici kazanç katsayıları literatürde yaygın olarak bilinen Meta-sezgisel algoritmalar aracılığıyla belirlenmiştir. Sistem tanımlama ve kontrolör kazancı belirleme işlemlerinde Matlab/Simulink ortamı kullanılmıştır. Simulink ile PLC arasındaki iletişim OPC toolbox üzerinden gerçekleştirilmiştir. Geliştirilen kontrolörlerin başarısı tablo ve grafikler aracılığıyla karşılaştırılmış ve sonuçlar tartışılmıştır.

Anahtar Kelimeler: DA Motor, Matlab OPC Toolbox, PID, OPC, Meta-sezgisel Algoritma, Optimizasyon

2024, 62 sayfa

ABSTRACT

M.Sc. Thesis

Burak Çağrı DUBA
Manisa Celal Bayar University
Graduate School of Education
Department of Mechanical and Manufacturing

Supervisor: Asst. Prof. Dr. Serkan ÇAŞKA

In this study, a proportional-integral-derivative (PID) controller was used to control the speed and position of a direct current motor. The control process was carried out using a programmable logic controller (PLC). Controller gain coefficients were determined through Meta-heuristic algorithms widely known in the literature. Matlab/Simulink environment was used for system identification and controller gain determination. Communication between Simulink and PLC was carried out via OPC toolbox. The success of the developed controllers was compared through tables and graphs and the results were discussed.

Keywords: DC Motor, Matlab OPC Toolbox, PID, OPC, Meta-heuristic Algorithm, Optimization

2024, 62 pages

ÖNSÖZ VE TEŞEKKÜR

Bu tez çalışmasında doğru akım (DA) motor kontrolü Programlanabilir lojik kontrolör (PLC) ile Matlab OPC Toolbox üzerinden oransal-integral-türev (PID) kontrolcüsü ile gerçekleştirilmiştir. Programlanabilir lojik kontrolör ile matlab/simulink arasında OPC haberleşmesi kullanılmıştır. DA motor kontrolünde darbe genişlik modülasyonu (DGM) tekniği kullanılmıştır. Kontrolcü değerleri sistem tanımlama ve üç metasezgisel algoritmalar ile analiz edilip performansları karşılaştırılmıştır.

Yüksek lisans sürecimde akademik ve sosyal anlamda desteklerini benden esirgemeyen her anlamda bana destek olan, bilgi ve deneyimleri ile yol gösteren kendisini tanımaktan mutluluk duyduğum danışman hocam Sayın Dr. Öğr. Üyesi Serkan ÇAŞKA' ya en içten teşekkürlerimi sunarım. Hayatımda her zaman yanımda olan ve desteklerini benden esirgemeyen sevgili annem Halime DUBA ve sevgili babam Ersin DUBA 'ya sonsuz teşekkürlerimi sunarım. Bu süreç de sevgili eşim yoldaşım olan Damla DEMİR DUBA 'ya teşekkürlerimi sunarım.

Burak Çağrı DUBA

Manisa, 2024

SİMGELER VE KISALTMALAR DİZİNİ

PID:	Oransal İntegral Türev
Kp:	PID Denetleyici Oransal Kazancı
Ki:	PID Denetleyici İntegral Kazancı
Kd:	PID Denetleyici Türev Kazancı
DA:	Doğru Akım
PLC:	Programlanabilir Lojik Kontrolör
PWM:	Sinyal Genişlik Modülasyonu
DGM:	Darbe Genişlik Modülasyonu
PNP:	Pozitif – Negatif - Pozitif
NPN:	Negatif - Pozitif - Negatif
HMI:	İnsan Makine Arayüzü
CNC:	Bilsayarlı Sayısal Kontrol
CPU:	Merkezi İşlem Birimi
PPR:	Devir Başına Darbe
RPM:	Dakikadaki Devir
V:	Volt
A:	Amper
STAK:	Sistem Tanımlama Araç Kutusu
OPC:	Açık Platform İletişimleri
MS:	Meta-sezgisel
GKA:	Gri Kurt Algoritması
BOA:	Balina Optimizasyon Algoritması
HŞO:	Harris Şahin Optimizasyon Algoritması
Tp(s):	Tepe Ulaşma Süresi
Tr(s):	Yükselme Süresi
Ts(s):	Yerleşme Süresi
Mp(%):	Maksimum Aşım
LAD:	Merdiven Diyagramı
FBD:	Fonksiyon Blok Diyagram
ST:	Yapılandırılmış Metin
IL:	Talimat Listesi

SKA: Sinüs Kosinus Algoritması

IOT: Nesnelerin İnterneti

OB: Organizasyon Blok

UTO: Uçan Tilki Optimizasyonu

GAO: Güve Alevi Optimizasyonu

EPROM: Silinip Programlanabilir Salt Okunur Bellek

ŞEKİLLER DİZİNİ

Sayfa

Şekil 1 : PLC İç Yapısı 5
Şekil 2 : PLC S71211C CPU 7
Şekil 3 : PLC Merdiven Diyagram(LAD) 8
Şekil 4 : PLC Fonksiyon Blok Diyagram(FBD) 9
Şekil 5 : Yapılandırılmış Metin (ST) 9
Şekil 6 : Talimat Listesi (IL) 10
Şekil 7 : DGM Sinyali ve Doluluk Boşluk Oranı 11
Şekil 8 : PNP ve NPN Bağlantı 12
Şekil 9 : Artımsal Enkoder 13
Şekil 10 : Artımsal Enkoder Çıkış Kanalları 14
Şekil 11 : Mutlak Enkoder 14
Şekil 12 : Metasezgisel Algoritmalar ve Sınıflandırılması 17
Şekil 13 : Metasezgisel Optimizasyon Algoritmalarının Kategorilendirilmesi 18
Şekil 14 : HŞO Tüm Fazlar 20
Şekil 15 : GKO Algoritma Akış Şeması 22
Şekil 16 : Uçak Tilki Optimizasyon Grafiksel Özet 23
Şekil 17 : UTO Algoritma Akış Şeması 24
Şekil 18 : PID Blok Diyagram 26
Şekil 19 : Oransal Durum 27
Şekil 20 : Oransal, İntegral ve Oransal İntegral (PI) Durum 28
Şekil 21 : PID Durum 28
Şekil 22 : DA Motor Diyagram 30
Şekil 23 : Matlab OPC Toolbox Fonksiyon 36
Şekil 24 : OPC Toolbox OPC Config KepServer Ayarı 38
Şekil 25 : OPC Toolbox OPC Read Fonksiyonu KepServer Veri Tanımlaması 39
Şekil 26 : OPC Toolbox OPC Write Fonksiyonu Veri Tanımlaması 40
Şekil 27 : Açısal Konum Kontrolü Blok Diyagramı 40
Şekil 28 : Açısal Hız Kontrolü Blok Diyagramı 41
Şekil 29 : KepServerEx OPC Sunucu Ayarları 41
Şekil 30 : KepServerEx PLC Adres Ayarları 42
Şekil 31 : UaExpert İstemci Oluşturma 43

Şekil 32 : PLC OPC UA Sunucu Etkinleştirilmesi 44
Şekil 33 : PLC PUT/GET Etkinleştirilmesi 44
Şekil 34 : PLC Enkoder Değeri Okuma 45
Şekil 35 : PLC Enkoder RPM Hesabı 46
Şekil 36 : PLC DGM Kontrol 47
Şekil 37 : Sistem Blok Diyagramı 47
Şekil 38 : Sistem Genel Gösterimi 48
Şekil 39 : DA Motor ve Enkoder 48
Şekil 40 : H Köprü 49
Şekil 41 : DA Motor Sürücü Devresi 49
Şekil 42 : DA Motor Sürücü 50
Şekil 43 : Sistem Tanımlama Araç Kutusu Model 51
Şekil 44 : Sistem Tanımlama Araç Kutusu Sonuç 51
Şekil 45 : PLC PID Fonksiyon 52
Şekil 46 : PLC PID Ayarları 53
Şekil 47 : Konum Kontrolünde PID Eğrileri 54
Şekil 48 : Hız Kontrolünde PID Eğrileri 55

TABLO DİZİNİ

Sayfa

Tablo 1 : CPU S71211C Özellik 7

Tablo 2 : Metasezgisel Algoritmalar Son 10 Yıl 18

Tablo 3 : PID Parametre Artırma Azaltma Etkileri 26

Tablo 4 : Açısal Konum Kontrolünde Kontrolörlerin Performansı 55

Tablo 5 : Açısal Konum Kontrolörlerin PID katsayıları 55

Tablo 6 : Açısal Hız Kontrolünde Kontrolörlerin Performansı 56

Tablo 7 : Açısal Hız Kontrolörlerin PID katsayıları 56

İÇİNDEKİLER

Sayfa

ÖZET II
ABSTRACT III
ÖNSÖZ VE TEŞEKKÜR IV
SİMGELER VE KISALTMALAR DİZİNİ V
ŞEKİLLER DİZİNİ VII
TABLO DİZİNİ IX
İÇİNDEKİLER X
GİRİŞ 1
1.GENEL BİLGİLER 3
1.1. Endüstriyel Otomasyon 3
1.2. PLC 4
1.3. Darbe Genişlik Modülasyonu 10
1.4. PNP ve NPN Yapısı 11
1.5. Enkoder 12
1.6. Optimizasyon 14
1.7. Metasezgisel Algoritmalar 15
1.8. PID 25
1.9. Optimal Kontrol 29
1.10. DA Motor 30
1.11. OPC 34
2. MATERYAL VE YÖNTEMLER 36
2.1. MATLAB 36
2.2. Simulink 37
2.3. KEPServerEx ve UAExpert 41
2.4. PLC Kontrol 43
2.5. DA Motor ve Sürücü 48
2.6. Sistem Tanımlama Araç Kutusu 50
2.7. PID Kontrol 52
3. ARAŞTIRMA BULGULARI TARTIŞMA 54
4. SONUÇLAR VE ÖNERİLER 57
KAYNAKLAR 59

GİRİŞ

Literatürde DA motorun kontrolünde birçok farklı yöntem bulunmaktadır. Bu tez çalışmasında DA motorun açısal konum ve hız kontrolü için ekipman olarak programlanabilir lojik kontrolör (PLC) ve denetleyici türü olarak oransal-integral-türev (PID) denetleyici kullanılmıştır. PLC çıkışlarından biri PLC üzerinde bulunan darbe genişlik modülasyonu (DGM) modülatör olarak ayarlanır. Motor kontrolü için PLC çıkışı DGM çıkış seçilir ve doğrudan güç devresine uygulanmaz. DA motor kontrolünde, sürücü devresinde kontrol kısmına PLC DGM çıkışları uygulanır. Motorun hız bilgisini okumak için enkoder kullanılmaktadır. Enkoderin ürettiği sinyal yüksek hızlı sayıcı olarak tanımlanan girişlere uygulanır. Bu tez çalışmasında kullanılan enkoder NPN yapıya, PLC ise PNP tip yapıya sahip olduğu için PLC dijital giriş için ortak uç olan "common" ucuna 24V besleme gerilimi uygulanarak o ortak uca bağlı tüm girişler NPN kanala çevrilmiştir. Motorun açısal konum ve hız bilgisi bu kanal üzerinden okunarak hesaplanır. Bu tez çalışmasında haberleşme kısmında, PLC ile OPC sunucu, Matlab OPC ile haberleştirilmiştir. PLC ile veriler KEPServerEx ve UaExpert ile node' lara aktarılmıştır. Matlab OPC üzerinden veri çekmek için OPC Toolbox ve UaClient fonksiyonu kullanılarak veriler Matlab ortamına çekilmiştir. DA motorun açısal hız ve konum modelleri ile PLC içerisindeki PID kontrolör kazançları ise Matlab ortamında meta-sezgisel (MS) algoritmalar ve sistem tanımlama ile ayarlanmıştır.

Sistem Tanımlama Araç Kutusu (STAK), MATLAB fonksiyonlarını ve Simulink bloklarını bir araya getirerek dinamik sistemlerin modellemesi, zaman serisi analizi ve tahmini için bir platform sunar. STAK kullanarak, dinamik sistemlerin zaman veya frekans alanı verilerinden yararlanarak, ölçülen değişkenler arasındaki dinamik ilişkileri analiz ederek, sürekli veya ayrık zaman alanında transfer fonksiyonları, süreç modelleri ve durum-uzay modelleri oluşturulabilir. Ayrıca doğrusal ve doğrusal olmayan otoregresif modelleme tekniklerini kullanarak zaman serilerini tahmin edebilir [1].

STAK, Gauss Süreçleri, Destek Vektör Makineleri ve diğer gösterimler gibi makine öğrenimi teknikleriyle Hammerstein-Wiener ve Doğrusal Olmayan ARX modellerini kullanarak doğrusal olmayan sistem dinamiklerini tahmin etmenize olanak tanır. Bir alternatif olarak, sinir ağlarını kullanarak derin öğrenme

yöntemlerini benimseyerek, doğrusal olmayan sistem dinamiklerini yakalayabilir ve sinirsel sıradan diferansiyel denklemler (ODE) modelleri oluşturabilirsiniz. Araç kutusu, kullanıcı tanımlı bir modelin parametrelerini tahmin etmek için gri kutu sistem tanımlaması yapılmasını sağlar [2]. Kontrol tasarımını, teşhis ve prognostik uygulamaları etkinleştirmek amacıyla hızlı simülasyonlar için tanımlanan modelleri Simulink'e entegre edilebilir [3]. Bu tez çalışmasında DA motorun açısal konum ve hız modelinin elde edilmesi için STAK kullanılmıştır.

1.GENEL BİLGİLER

1.1. Endüstriyel Otomasyon

Günümüzde, yüksek düzeyde otomatikleştirilmiş ürün gerçekleştirme ya da endüstriyel otomasyon sistemleri her sektörde rekabeti karşılaması için önemlidir. Hızla değişen teknolojiler ve ürün yaşam döngülerinin kısalması, birçok şirketi rekabet için endüstriyel otomasyona yöneltmiştir [4].

Teknolojinin hızla ilerlemesiyle birlikte hayatımızın vazgeçilmez bir parçası haline gelen yeni yenilikler her gün ortaya çıkmaktadır. Küreselleşen piyasalarda, kaliteli ve uygun maliyetli ürünlerin üretimi bir gereklilik haline gelmiştir. Günümüzde, otomasyon teknolojilerinden yararlanarak üretim maliyetlerini seri üretim yaparak düşürmek mümkündür [5].

Otomasyon, insan uzuvlarının yerine bir cihazın, bir sürecin veya bir sistemin otomatik olarak kontrol edilmesini sağlayan elektronik veya mekanik yöntemlerin kullanılmasıdır. Bu yöntemler, gözlem yapma, karar alma ve çaba gösterme gibi eylemleri insan müdahalesi olmadan gerçekleştirebilir [6].

Otomasyon, bir işin insanlar ve makineler arasında bölüştürülmesini sağlar. Bu bölüştürme oranı, otomasyonun seviyesini belirler. Eğer işin çoğu insan gücüyle gerçekleştiriliyorsa, buna yarı otomasyon denir; ancak işin büyük bir kısmı makineler tarafından yapılıyorsa, bu durumda tam otomasyondan söz edilir [7].

Endüstriyel Otomasyon, rekabetçi pazarda varlığını sürdürebilmek adına imalatta modern teknikler ve uygulamaları kullanan önemli bir üretim stratejisidir [8]. Endüstriyel otomasyon, endüstriyel süreçleri otomatikleştirmek için çeşitli kontrol sistemlerinin, makinelerin ve teknolojilerin kullanılmasını ifade eder. Buna üretim süreçleri, üretim hatları ve endüstriyel ortamlardaki diğer faaliyetler dâhildir. Endüstriyel otomasyonun temel amacı verimliliği artırmak, üretkenliği artırmak, ürün kalitesini artırmak ve tekrarlayan veya tehlikeli görevlere insan müdahalesini azaltmaktır.

Endüstriyel otomasyon sistemlerinde, programlanabilir lojik kontrolörler (PLC'ler) merkezi ve kritik bileşenlerden birini oluşturur. Modicon firması, 1969 yılında endüstride ilk ticari amaçla üretilen PLC'leri piyasaya sürdü. Röleli kumanda

devrelerinin yerini alan bu PLC'ler, ilk başlarda yalnızca temel lojik işlemleri gerçekleştirebiliyordu ve bu nedenle PLC adını aldılar. İlk PLC'lerin endüstride başarılı bir şekilde uygulanmasıyla birlikte, Allen Bradley, GEC, General Electric, Siemens ve Westinghouse gibi firmalar maliyeti ortalamada tutarak yüksek performanslı PLC'ler geliştirmeye başladılar. Daha sonra, Toshiba, Mitsubishi, Omron ve diğer firmaların düşük maliyetli ve yüksek performanslı PLC'ler üretmeleriyle bu cihazların endüstriyel otomasyon sistemlerinde kullanımı yaygınlaştı [9].

PLC'ler, esneklik, güvenilirlik, genişleme kolaylığı ve düşük güç tüketimi gibi bir dizi özellik sunar ve bunun yanı sıra daha pek çok avantaj sağlar. PLC'lerde sadece yazılım değiştirilerek, sistemdeki her türlü değişiklik ve genişleme kolaylıkla kontrol edilebilir [10].

1.2. PLC

PLC'ler, mikroişlemci tabanlıdır. Bu cihazlar, sahadaki sensörlerden gelen verileri alır ve programlarına göre bu verileri işlerler. Böylece, sahadaki otomasyon enstrümantasyon cihazlarını etkili bir şekilde kontrol edebilirler. Endüstriyel otomasyon sistemlerinde, kontrol ve kumanda işlemlerini gerçekleştirmek için yaygın olarak kullanılırlar. PLC'ler, sahadaki cihazlarla iletişim kurmak ve veri alıp göndermek için giriş ve çıkış birimlerine sahiptirler. Ayrıca, sahadaki cihazlarla etkileşim halinde olmak için haberleşme birimlerine sahiptirler ve SCADA gibi uzaktan kontrol ve izleme sistemleriyle uyumlu endüstriyel bilgisayarlar olarak da kullanılabilirler [11].

Programlanabilir Lojik Kontrolör (PLC), endüstriyel otomasyon ve kontrol uygulamaları için tasarlanmış özel bir dijital bilgisayardır. PLC'ler çeşitli endüstrilerde elektromekanik süreçleri otomatikleştirmek, girişleri izlemek ve çıkışları kontrol etmek için yaygın olarak kullanılmaktadır. PLC'ler karmaşık röle tabanlı kontrol sistemlerinin yerini almak üzere ortaya çıkmıştır. Başlangıçta endüstriyel süreçlerin programlanmasını ve kontrolünü kolaylaştırmak için geliştirilmiştir.

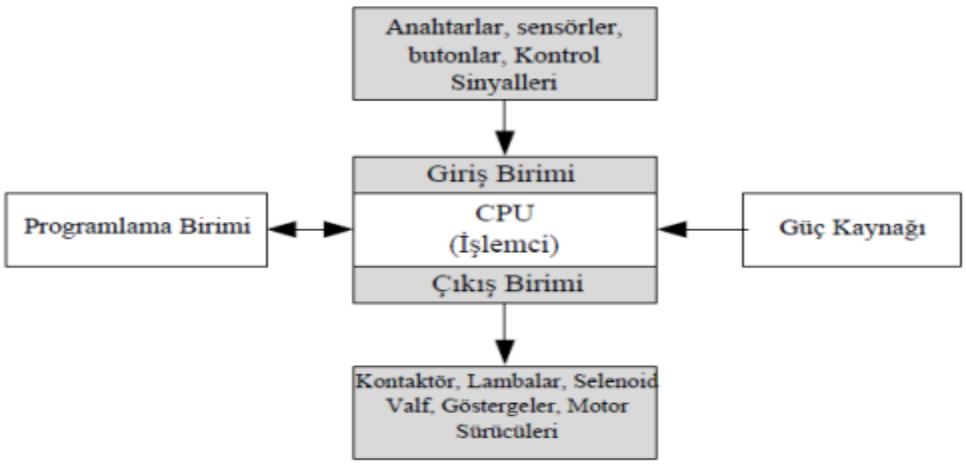
PLC'ler, bağlı sensörlerin ve anahtarların durumunu okuyarak girişlerini sürekli olarak tarar. PLC'nin kullanıcı tarafından yazılan program mantığı, önceden

belirlenmiş bir tarama döngüsü süresine göre döngüsel olarak yürütülür. Program mantığına ve giriş durumlarına bağlı olarak PLC, çıkış modüllerine bağlı aktüatörleri ve diğer cihazları kontrol etmek için çıkışlarını günceller. PLC'ler tipik olarak giriş ve çıkışların gerçek zamanlı izlenmesini sağlayarak operatörlerin sistem durumunu görüntülemesine, arızaları teşhis etmesine ve gerektiğinde ayarlamalar yapmasına olanak tanır.

PLC'ler, üretim gereksinimleri veya proses parametrelerindeki değişikliklere uyum sağlamak için kolayca yeniden programlanabilir ve yeniden yapılandırılabilir. Zorlu endüstriyel ortamlar için tasarlanmıştır ve yüksek düzeyde güvenilirlik ve dayanıklılık sunar. Farklı uygulamaların ihtiyaçlarına uyacak şekilde küçük, bağımsız ünitelerden büyük, dağıtılmış kontrol sistemlerine kadar ölçeklendirilebilir. Çeşitli cihazlar, sensörler ve ağlarla arayüz oluşturarak mevcut otomasyon sistemlerine sorunsuz entegrasyon sağlar.

PLC teknolojisi bilgi işlem, ağ ve yazılım alanlarındaki gelişmelerle birlikte evrim geçirmeye devam etmektedir. Uç bilişim, bulut bağlantısı ve Endüstriyel Nesnelerin İnterneti (IoT) gibi trendler yeni nesil PLC'lerin gelişimini etkilemektedir. PLC'ler, gelişmiş performans ve verimlilik için kestirimci bakım, makine öğrenimi ve uzaktan izleme gibi gelişmiş özelliklerle giderek daha fazla entegre edilmektedir.

PLC'ler genellikle giriş birimi, çıkış birimi ve CPU (merkezi işlem birimi) gibi üç temel bileşen içerir. Bunların yanı sıra, PLC'lerde güç kaynağı, programlama birimi ve diğer yardımcı bileşenler de bulunabilir. PLC'lerin içyapısı şekil 1 de verilmiştir [12].



Şekil 1 : PLC İç Yapısı

PLC'nin en kritik birimi olan Merkezi İşlem Birimi, tüm mantıksal, aritmetik, sayma ve zamanlama işlemlerini gerçekleştirir. Bu birim, otomasyon sisteminin kontrolünü sağlayacak olan programın derlenmesi ve yürütülmesi süreci için sürekli olarak bellekle iletişim halindedir [13]. Merkezi İşlem Birimi (CPU) , kontrol programlarının yürütülmesinden ve sistem işlemlerinin yönetilmesinden sorumlu PLC'nin temel bileşenidir. Program talimatlarını yürütmekten, verileri işlemekten ve sistem işlemlerini kontrol etmekten sorumludur [14].

PLC'ler, kontrol programlarının depolandığı program belleği, değişkenlerin ve verilerin saklandığı veri belleği ve işletim sistemi verilerinin tutulduğu sistem belleği gibi çeşitli bellek tiplerini içerir. Program belleği, merdiven mantığı, fonksiyon blok diyagramı, yapılandırılmış metin veya komut listesi gibi programlama dillerinde yazılmış kullanıcı tarafından oluşturulan kontrol programını saklar. Veri belleği, kontrol programı tarafından kullanılan değişkenler, zamanlayıcılar, sayaçlar ve diğer verilerin yürütme sırasında depolandığı bir bellek birimidir. Sistem belleği, işletim sistemi verilerini ve PLC'nin çalışması için gereken diğer sistemle ilgili bilgileri içerir [14].

Bellek birimi, verilerin saklanması ve gerektiğinde ilgili alanlara iletilmesi görevini üstlenir. Bu birim, genellikle program belleği, sistem belleği ve veri belleği olmak üzere farklı bölümlere ayrılmıştır ve her bir bölümün farklı işlevleri vardır. PLC'lerde yaygın olarak kullanılan bellek elemanı, Programlanabilir Silinebilir Tekrar Yazılabilir Hafıza (EPROM) olarak bilinir. Bu bellek elemanı, programlanabilir ve silinebilir bir hafıza birimidir ve PLC'ler için özel olarak tasarlanmış programlar burada saklanır ve CPU'ya buradan iletilir [15].

Giriş modülleri, sensörlerden ve diğer donanımlardan gelen sinyalleri alır ve bu sinyalleri PLC tarafından işlenebilir hale getirmek için dijital veya analog sinyallere dönüştürür [14]. PLC'nin giriş birimi, sisteme yönlendirilen sinyalleri değerlendirir ve bunları lojik gerilim seviyelerine dönüştürerek kontrol işlemlerini gerçekleştirir. Bu birim, genellikle seviye, sıcaklık, basınç, yakınlık gibi farklı tipteki sensörden gelen sinyalleri alır ve bunları ikilik sistem (1 veya 0) değerlerine dönüştürür. Giriş birimi, 24, 48, 100-120, 200 ve 240 volt aralığında doğru veya alternatif akımı işleyebilir [16].

Çıkış modülleri, programlanmış PLC talimatları doğrultusunda motorlar, valfler veya röleler gibi aktüatörlere kontrol sinyalleri ileterek işlev görür [14]. PLC'nin çıkış birimi, giriş birimine uygulanan lojik gerilim voltaj sinyallerini, kontrol edilen sistemde bulunan röle, kontaktör gibi kumanda elemanlarının çalışması için uygun sinyallere dönüştüren birimdir. Bu birim genellikle transistör, röle veya triyaklı devrelerden oluşur. Yüksek hızlı açma-kapama gerektiren uygulamalarda ise transistörlü veya triyaklı çıkış devreleri tercih edilir [15].

PLC'ler diğer cihazlara, ağlara veya programlama cihazlarına bağlanmak için çeşitli iletişim arayüzlerini destekler. Yaygın iletişim protokolleri arasında Ethernet, seri iletişim (RS-232, RS-485), Profibus, Modbus, DeviceNet ve diğerleri bulunur. İletişim arayüzleri PLC'lerin diğer PLC'ler, denetleyici kontrol ve veri toplama (SCADA) sistemleri, insan-makine arayüzleri (HMI'lar) ve kurumsal ağlarla veri alışverişi yapmasını sağlar. Bu tez çalışmasında kullanılan PLC şekil 2'de ve PLC 'ye ait özellikler tablo'1 de gösterilmiştir [14].



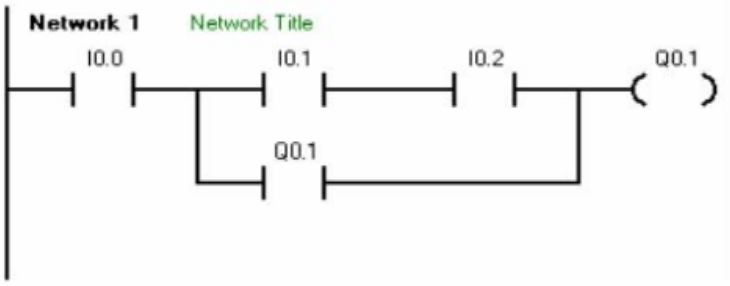
Şekil 2 : PLC S71211C CPU

Tablo 1 : CPU S71211C Özellik

ÖZELLİK	**CPU 1211C**
Çalışma Bellek	50 Kbayt
Yüklü Bellek	1 Mbayt
Kalıcı Bellek	10 Kbayt
Dijital Yerel Giriş/Çıkış	6 giriş /4 çıkış
Analog Yerel Giriş/Çıkış	2 giriş
Proses Görüntü Boyutu Giriş (I)	1024 bayt
Proses Görüntü Boyutu Çıkış (Q)	1024 bayt

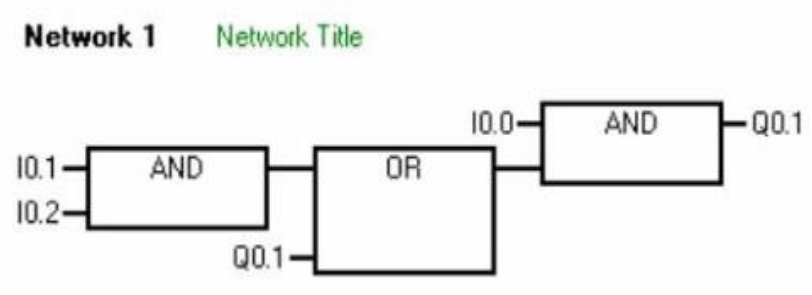
Bit Bellek (M)	4096 bayt
Sinyal Modülü (SM) genişletilmesi	Hiçbiri
Sinyal Kartı (SB), Pil Kartı (BB), veya İletişim Kartı (CB)	1
İletişim Modülü (CM) (sol taraf genişletilmesi)	3
Yüksek Hızlı Sayaçlar	100/180 kHz - Ia.0 dan Ia.5
Darbe Çıkışlar	100z - Qa.0 dan Qa.3

Merdiven mantığı veya dili PLC'ler de kullanılan en yaygın programlama dilidir. Elektrik röle diyagramlarına benzer ve elektrik devrelerine aşina olan mühendisler ve teknisyenler için sezgiseldir. Merdiven diyagramı, gerçek elektrik devrelerinde olduğu gibi, enerji akışını kullanıcıya anlaması kolay bir şekilde sembolize eder. Ladder programında, sol tarafta bulunan dikey çizgi enerji kaynağını temsil eder. Kapalı kontaklar enerji akışına izin verirken, açık kontaklar ise enerji akışını engellerler. Şekil 3' de PLC merdiven mantığı dili örnek olarak gösterilmiştir [14].



Şekil 3 : PLC Merdiven Diyagram(LAD)

Fonksiyon blok diyagramı (FBD) veya dil metodu, lojik kapıların kullanımına dayanan ve programlama işlemlerini şematik bir şekilde gösteren bir yaklaşımdır. Bu yöntemde, kullanılan lojik semboller kutu şeklinde ifade edilir. Sembollerin sol tarafında giriş sinyalleri, sağ tarafında ise çıkış sinyalleri yer alır. Şekil 4' de PLC fonksiyon blok diyagramı dili örnek olarak gösterilmiştir [14].



Şekil 4 : PLC Fonksiyon Blok Diyagram(FBD)

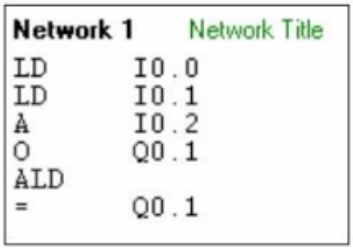
Yapılandırılmış Metin ST, karmaşık algoritmaların ve hesaplamaların kolayca ifade edilmesi için tasarlanmış yüksek seviyeli bir programlama dilidir. Bu dil, Pascal veya C diline benzer bir yapıya sahiptir ve özellikle karmaşık sistemlerin kontrolü ve yönetimi için uygundur. Şekil 5' de PLC yapılandırılmış metin dili örnek olarak gösterilmiştir [14].

```
FUNCTION_BLOCK FB100

VAR_IN_OUT
    Start:INT;  // Hedef DB deki (DB aşağıda)  başlangıç indeksi
    End  :INT;  // Hedef DB deki son indeks
END_VAR

VAR
   i,j:INT;
   Swap:REAL;  // Takas değişkeni verileri yer değiştirmede kullandım
END_VAR

BEGIN
   FOR i:=Start    TO (End-1) BY 1     DO   // İç içe iki for döngüsüyle verileri
       FOR j:= i+1 TO End     BY 1     DO     // tarıyoruz
           IF DB101.Data[i]> DB101.Data[j] THEN  // i ve j inci veriler test
edilir
               Swap         :=DB101.Data[j];    // Yer değiştirme işlemi
yapıyoruz
               DB101.Data[j]:=DB101.Data[i];
               DB101.Data[i]:=Swap;
           END_IF;
       END_FOR;
   END_FOR;

END_FUNCTION_BLOCK

DATA_BLOCK DB101    //Hedef DB DB101, 32 adet bilgi var içinde
STRUCT
   Data       : ARRAY [0..31] OF REAL;
END_STRUCT
BEGIN

END_DATA_BLOCK
```

Şekil 5 : Yapılandırılmış Metin (ST)

Talimat Listesi (IL), kontrol mantığını metinsel bir formatta ifade etmek için anımsatıcı komutlar kullanan bir düşük seviyeli programlama dilidir. Mnemonik adı verilen komutlar, bir işlemi belirtirken operantlar ise işlem yapılan bellek alanlarını

gösterir. Bu yöntem, cihazın makina koduna en yakın gösterimini sağlayarak geniş programlama olanakları sunar. Şekil 6' da PLC talimat listesi örnek olarak gösterilmiştir [14].



Şekil 6 : Talimat Listesi (IL)

1.3. Darbe Genişlik Modülasyonu

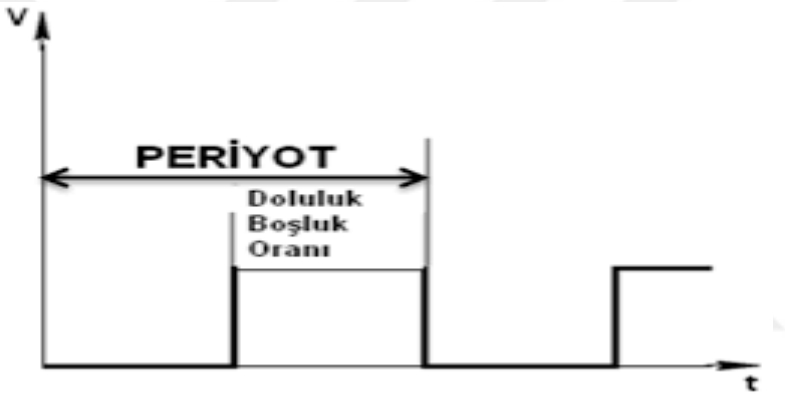
Darbe Genişlik Modülasyonu DGM, DA motor kontrolünde kullanılmasının amacı doğrusal güç amplifikatörlerindeki aşırı ısı yayılmasını önlemektir. Büyük ölçekli ısı emicileri ve bazen zorunlu soğutma sistemleri, genellikle ısı yayılma sorununu ortaya çıkarır. Ancak, DGM amplifikatörleri, çok daha yüksek bir güç dönüşüm verimliliği sunarak bu sorunu büyük ölçüde azaltır. Ayrıca, DGM sürücüsüne gelen giriş sinyali herhangi bir dijital sistemden doğrudan alınabilir ve bunun için herhangi bir D/A dönüştürücüsüne ihtiyaç duyulmaz. DGM güç amplifikatörü, bazı zayıflıklara sahip bir sistemdir. Genellikle, hedef sinyal, gerilim genliğinden ziyade darbenin zaman süresine veya iş döngüsüne dönüştürülür. Bu, açıkça doğrusal olmayan bir işlemdir. Ancak, genellikle motor kontrolünde geçerli olan birkaç varsayım ile DGM, doğrusal olarak (yani saf kazanç olarak) kabul edilebilir bir şekilde yaklaşık olarak hesaplanabilir. DGM amplifikatörünün doğrusal modeli, ortalama gerilimin voltaj dalgasının integraline eşit olması prensibine dayanır [17].

Darbe Genişlik Modülasyonu (DGM), analog elektrik sinyalleri elde etmek için kare dalga darbelerinin genişliklerini kontrol etme tekniğidir. Bu yöntemde, üretilen kare dalga darbelerinin genişliklerinin ortalama değeri, hedeflenen analog çıkışı elde etmek için kullanılır. DGM'nin en önemli avantajlarından biri, analog sinyal elde etme sürecini basitleştirmesidir, sayısal sinyali analog hale dönüştürmek için karmaşık yapılar gerekmez. Geleneksel yöntemlerde sinyalin voltaj seviyesini değiştirmek yerine, DGM darbe genişliklerini ayarlayarak aynı sonuca ulaşır.

DGM, sabit frekanslı bir dijital sinyalin üretilmesiyle başlar. Bu dijital sinyal tipik olarak sabit bir frekansta tekrar eden darbelerden oluşur. Bu darbelerin frekansı genellikle taşıyıcı frekansı olarak adlandırılır. DGM sinyalinde sabit bir zaman periyodu vardır ve görev döngüsü darbenin T_{on} açık durumda (yüksek voltaj) geçirdiği sürenin, toplam periyota olan oranını belirler. Görev döngüsü yüzde olarak ifade edilir ve darbenin genişliğinin toplam periyota olan oranını temsil eder.

$$\text{Görev Döngüsü} = \frac{\mathrm{T_{on}}}{\mathrm{T_{on}} + \mathrm{T_{off}}} \times 100$$

$$\text{DGM Frekans} = \frac{1}{\text{Periyot}}$$



Şekil 7 : DGM Sinyali ve Doluluk Boşluk Oranı

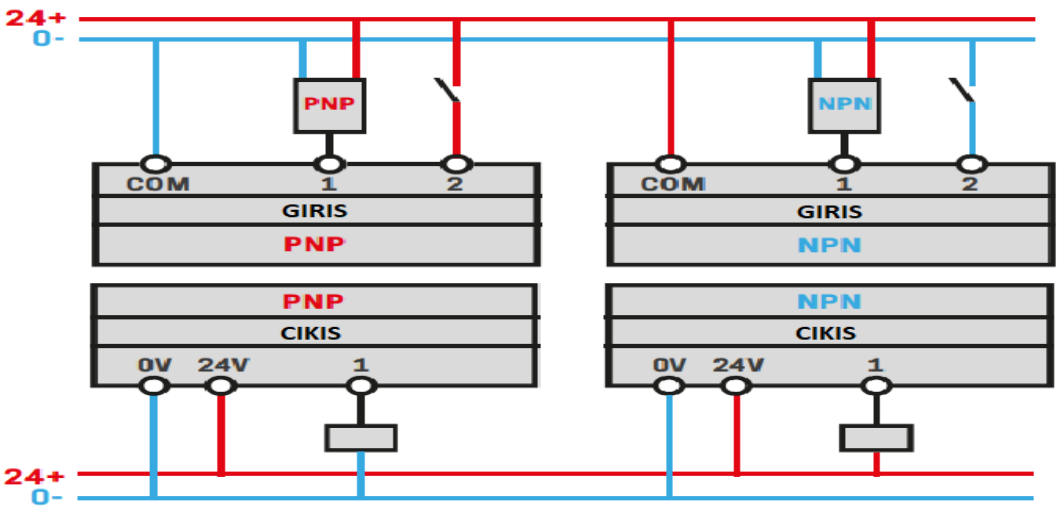
DGM yöntemi özellikle DA motorlarının hız düzenlemesinde yaygın olarak kullanılır. DGM sinyalinin görev döngüsünü ayarlayarak, DA motoruna iletilen ortalama güç kontrol edilir, bu da motorun dönme hızını etkiler. Yüksek bir görev döngüsü daha hızlı bir hıza, düşük bir görev döngüsü ise daha yavaş bir hıza karşılık gelir. Şekil 7 ‘de DGM sinyali ve doluluk boşluk oranı gösterilmiştir.

1.4. PNP ve NPN Yapısı

Endüstriyel otomasyonda (Pozitif-Negatif-Pozitif) PNP ve (Negatif-Pozitif-Negatif) NPN, kontrol sistemlerinde yaygın olarak kullanılan dijital sensör, aktüatör ve cihaz türlerini ifade eder. Bu terimler, bu cihazların elektriksel özelliklerini tanımlar ve endüstriyel kontrol devrelerinin tasarlanması ve yapılandırılmasında önemli bir rol oynar [18].

PNP sensörlerinin pozitif tarafı besleme voltajına bağlanır ve etkinleştirildiğinde pozitif sinyal sağlar. Sensör tetiklendiğinde, pozitif besleme voltajını giriş terminaline bağlayarak PLC'ye veya kontrol cihazına sensörün aktif durumda olduğunu bildirir. PNP çıkışlar etkinleştirildiğinde yüke akım sağlar. Çıkışa enerji verildiğinde pozitif besleme voltajını yüke bağlayarak akımın akmasını ve cihazın aktif hale gelmesini sağlar [18].

NPN sensörlerinin negatif tarafı toprağa (0V) bağlıdır ve etkinleştirildiğinde negatif bir sinyal sağlar. Sensör tetiklendiğinde giriş terminalini toprağa bağlayarak PLC'ye veya kontrol cihazına sensörün aktif durumda olduğunu bildirir. NPN çıkışlı PLC'nin dijital çıkışı negatif gerilim sağladığı durumu ifade eder. Aktif durumda NPN transistörü aktifleşir ve çıkışa negatif gerilim uygulanır. Bu negatif gerilim, yükü etkinleştirmek veya kontrol etmek için kullanılır. Pasif durumda ise, çıkışa herhangi bir gerilim uygulanmaz ve çıkış kapalıdır. Şekil 8 'de PNP ve NPN bağlantı gösterilmiştir [18].



Şekil 8 : PNP ve NPN Bağlantı

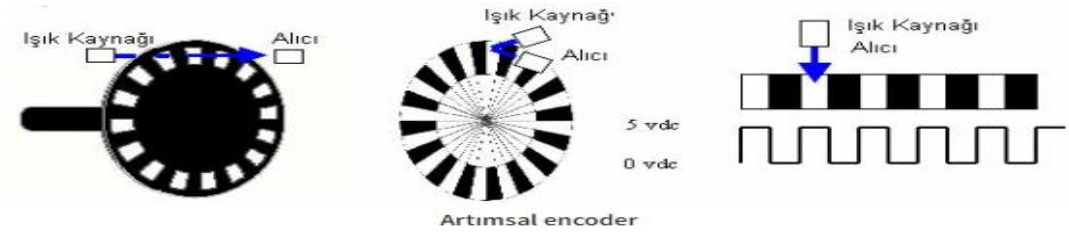
1.5. Enkoder

Enkoder, dönen hareketi ardışık sayısal sinyallere dönüştüren bir cihazdır. Bir şaftın açısal konumunu, hareketini ve yönünü analog veya dijital bir kod haline getiren elektromekanik bir cihazdır. Hem manyetik hem de optik olarak çalışabilir. Enkoder, hareketi bir sayaç veya PLC gibi bir kontrol cihazı tarafından okunabilen bir elektrik sinyaline dönüştürür. Bir motor miliyle bağlandığında, optik bir sistem

aracılığıyla motorun dönüş yönünü ve hızını algılar ve bu verileri elektronik sistemine iletir. Enkoder tarafından iletilen bu geribildirim sinyali, bir cihazın konumunu, hızını ve yönünü belirlemeye yardımcı olur [19].

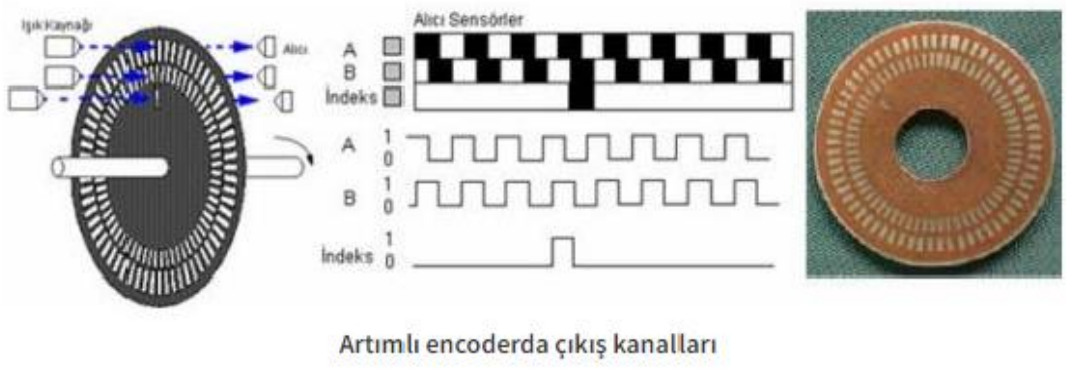
Enkoderler, doğrusal olarak çalışanlar (Linear Enkoder) ve döner olarak çalışanlar (Rotary Enkoder) olmak üzere iki ana kategoriye ayrılır. Pozisyona göre ise Artımsal (Incremental Enkoder) ve Mutlak (Absolute Enkoder) olmak üzere ikiye ayrılırlar [19].

Artımsal (Incremental) bir enkoderin diski, yarıksız ise, ışık doğrudan alıcıya ulaşır. Ancak, yarıklı bir disk kullanılıyorsa, ışık yarıklardan geçerek alıcıya gelir. Yansımalı bir enkoderde ise, disk üzerinde siyah ve beyaz çizgiler bulunur ve ışık bu çizgilerden yansır. Bu tip bir enkoder, pozisyon bilgisini sağlar ancak motor milinin dönüş yönünü belirlemez. Şekil 9 ’da Artımlı enkoder iç yapısı gösterilmiştir [19].



Şekil 9 : Artımsal Enkoder

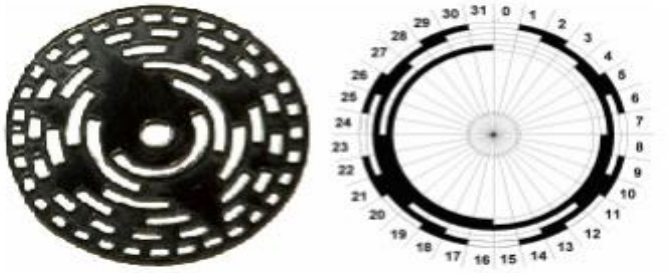
Artımsal enkoderler, dönüş yönünü belirlemek için iki sıra halinde dizilmiş yarıklardan yararlanır. Üst ve alt yarıklar arasında ¼ oranında kaçıklık bulunur. A kanalı, üst yarıktan gelen sinyali temsil ederken, B kanalı alt yarıktan gelen sinyali temsil eder. Milin dönmesiyle, her iki kanalda da mil hızına orantılı bir frekansta darbe üretilir ve bu darbeler arasındaki fark, dönüş yönünü belirler. Eğer A kanalı B kanalından önde ise, mil saat yönünde döner. Enkoder diskindeki yarık sayısı, enkoderin çözünürlüğünü belirler. Örneğin, bir enkoder diskinde 1024 yarık veya iz bulunuyorsa, enkoderin çözünürlüğü 1024 adım/tur olarak ifade edilir. Her kanalın darbe sayısı ve çözünürlüğü bilindiğinde, milin açısal konumu belirlenebilir. Enkoderler genellikle üçüncü bir kanala da sahiptir ve bu kanal Z veya indeks kanalı olarak adlandırılır. Bu kanal, her turda bir darbe üreterek milin dönme sayısı hakkında bilgi sağlar. Şekil 10 ’da artımlı enkoder çıkş kanalları gösterilmiştir [19].



Artımlı encoderda çıkış kanalları

Şekil 10 : Artımsal Enkoder Çıkış Kanalları

Mutlak tip enkoderler, milin her konumuna karşılık gelen tek bir sayısal sinyal üretir. Her konum için belirli bir sayısal desen vardır ve bu desen, ikili sayı düzeninde belirli bir bit sayısı çözünürlüğünü ifade eder. Şekil 11 ‘de mutlak enkoder gösterilmiştir [19].



Mutlak encoder

Şekil 11 : Mutlak Enkoder

Her bir dairesel sıra veya iz, disk üzerinde bir sayısal desen oluşturur. Örneğin, bir disk üzerinde 8 adet iz bulunuyorsa, enkoder toplamda $2^8 = 256$ farklı konumu temsil edebilir. Böylece, 8360 derecelik bir dönüş, 256 farklı konumu ifade eder, yani her bir konum yaklaşık olarak 1,4060 dereceyi temsil eder [19].

1.6. Optimizasyon

Optimizasyon, belirli bir problem için verilen koşullar altında en elverişli çözümü bulma sürecidir. Bu süreçte mevcut tüm alternatifler değerlendirilir [20]. Optimizasyon, temel olarak en iyi sonucu elde etmeyi amaçlayan bir iyileştirme

sürecidir. Belirli bir problemde, verilen kısıtlar altında en uygun çözümü bulma çabası olarak tanımlanır. Bir optimizasyon problemi, genellikle belirli kısıtları sağlamak için bilinmeyen parametre değerlerini bulmayı içerir [21].

Optimizasyon terimi genellikle belirli bir kriter veya kriterler altında, bu kriterleri karşılayan bir dizi çözüm arasından en iyi veya en uygun değerlerin seçilmesi sürecini ifade eder [22]. Klasik optimizasyon teknikleri olan Gradient-tabanlı [23], Newton-tabanlı [24] ve lineer programlama [25], yerel minimum noktalarda sıkışıp kalabilir ve hızlı yakınsama sorunlarıyla karşılaşıldığında başarılı olmayabilir [26][27].

Metasezgisel algoritmalar, geleneksel yöntemlerin başarısız olduğu veya yetersiz kaldığı durumlarda sıklıkla tercih edilen optimizasyon teknikleridir. Bu algoritmalar, yeni ve daha etkili çözümler geliştirmek amacıyla kullanılır ve günümüzde kaynakların en verimli şekilde kullanılmasıyla elde edilen maksimum fayda ihtiyacıyla birlikte, birçok alanda yaygın olarak kullanılmaktadır; bunlar arasında fizik, matematik, mühendislik ve ekonomi gibi alanlar bulunmaktadır [28][29].

Metasezgisel algoritmaların temel hedefi, tek amaçlı optimizasyonda problem alanındaki bir global çözüm noktasını bulmaktır. Eğer global çözüm noktası elde edilemezse, çözüm kümesindeki en yakın birey veya vektör global çözüm olarak kabul edilir. Gerçek dünya problemlerinde, çözüm alanının global noktası genellikle bilinmemektedir. Bu durumda, mevcut aday çözümler arasından kısıtları ve amaç fonksiyonunu en iyi sağlayan birey veya vektör, problemin en yakın çözümü olarak kabul edilir [30][31].

1.7. Metasezgisel Algoritmalar

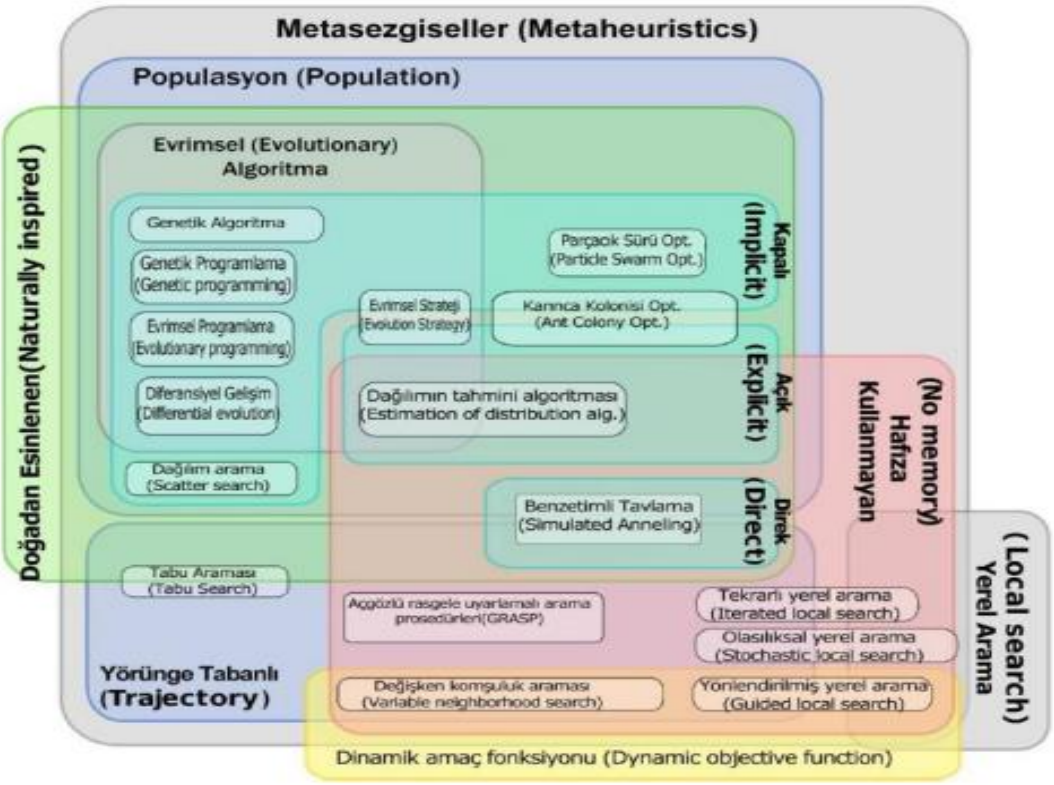
Optimizasyon, bir problemin en uygun çözümünün bulunması sürecidir, yani verilen koşullar altında en iyi sonucun elde edilmesini sağlama işlemidir. Günümüzde, birçok farklı optimizasyon algoritması, çeşitli problemlerin çözümünde kullanılmaktadır. Bu problemlerin çözümünde, hem matematiksel hem de sezgisel teknikler uygulanmaktadır. Çözüm alanı geniş olan problemlerde, matematiksel algoritmalar genellikle tüm alanı tarar, bu da daha maliyetli olabilir. Bu durumda,

sezgisel algoritmalar çözüm alanını daha hızlı ve etkili bir şekilde tarar, bu da avantajlı olabilir [32].

Temel sezgisel yöntemlerin bir araya gelerek Metasezgisel adı verilen yeni algoritmaların oluşturulmasıyla etkili bir ilerleme kaydedilmiştir. Bu algoritmalar, karmaşık çalışma ortamlarında, verimli arama süreçleri kullanarak çözüm uzayındaki en iyi çözüme daha hızlı bir şekilde ulaşabilmektedir [32].

Metasezgisel optimizasyon algoritmalarının öne çıkan avantajlarından biri, yerel optimum noktalarında sıkışıp kalmadan genel bir sonuca erişme becerisidir [33].

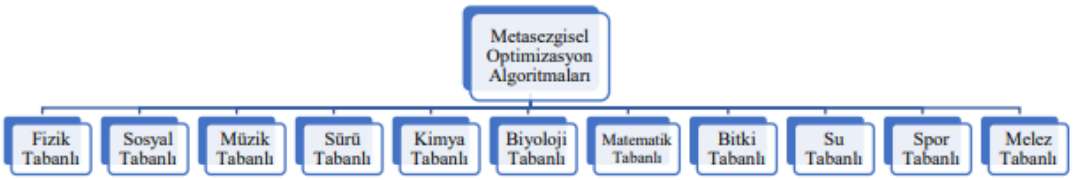
Literatürde sıkça kullanılan ve genel kabul görmüş birçok meta-sezgisel algoritma bulunmaktadır. Bu algoritmalar arasında öne çıkanlar; Genetik Algoritma (GA), Karınca Kolonisi Optimizasyonu (KKO), Parçacık Sürü Optimizasyonu (PSO), Yapay Arı Kolonisi (YAK), Diferansiyel Gelişim Algoritması (DGA), Benzetim Tavlama (BT), Yerçekimi Arama Algoritması (YAA), Gaz Brownian Hareketi Optimizasyonu (GBHO), Isı Transferi Arama (ITA), Elektromanyetik Alan Optimizasyonu (EAO), Optikten Esinlenen Optimizasyon (OEO), Ağırlıklı Süperpozisyon Çekimi (ASÇ), Orman Optimizasyonu Algoritması (OOA) olarak sayılabilir. Sürekli olarak, bu algoritmaların yanı sıra birçok yeni optimizasyon algoritması önerilmekte ve bu önerilen algoritmalar üzerinde geliştirmeler yapılmaktadır. MS algoritmalar ve sınıflandırılması şekil 12 ‘de MS algoritmalar ve sınıflandırması gösterilmiştir [32].



Şekil 12 : Metasezgisel Algoritmalar ve Sınıflandırılması

Optimizasyon problemlerinin çözümünde, Global arama ve Yerel arama stratejileri kullanılmaktadır. Yerel arama, belirli bir bölgede en iyi çözümü bulmayı hedeflerken, global arama tüm çözüm uzayını tarayarak genel en iyi çözümü bulmayı amaçlar. Yerel arama, bulunan en iyi noktanın her zaman çözüm uzayının en iyisi olduğunu garanti etmez [34].

Metasezgisel optimizasyon teknikleri, matematiksel modellerin yetersiz kaldığı büyük ölçekli optimizasyon problemlerini çözmek için sıkça başvurulan yöntemlerdir. Bu yöntemler, genellikle doğadaki süreçleri, mekanizmaları veya sosyal etkileşimleri taklit ederek, karmaşık problemlerde optimal çözümler elde etmeyi hedeflerler. Bu nedenle, metasezgisel optimizasyon, çeşitli optimizasyon problemlerine etkili bir şekilde uygulanabilen yaygın bir global optimizasyon yaklaşımı olarak kabul edilir [35]. Metasezgisel optimizasyon algoritmaları, fiziksel, sosyal, müziksel, sürü tabanlı, kimyasal, biyolojik, matematiksel, bitkisel, su temelli, spor odaklı ve melez tabanlı olmak üzere 11 farklı kategoriye ayrılarak incelenmektedir. Şekil 13 ‘de MS optimizasyon algoritmalarının kategorilendirilmesi gösterilmiştir. Tablo 2 ‘de son 10 yıl içerisinde bulunan MS optimizasyon algoritmaları gösterilmiştir [36].



Şekil 13 : Metasezgisel Optimizasyon Algoritmalarının Kategorilendirilmesi

Tablo 2 : Metasezgisel Algoritmalar Son 10 Yıl [37]

İsim	Ana Kategori	Alt Kategori	Y.Yılı
Geri İzleme Arama Algoritması	Evrimsel temelli	-	2013
Kara Delik Algoritması	Doğadan İlham Alan	Fizik Kimya Temelli	2013
Yunus Ekolokasyonu	Doğadan İlham Alan	Biyo-esinli	2013
Hayvan Göçü Optimizasyonu	Doğadan İlham Alan	Sürü tabanlı	2013
Keshtel Algoritması	Doğadan İlham Alan		2014
SDA Optimizasyon Algoritması	Doğadan İlham Alan	Biyo-esinli	2014
Yapay Kök Toplayıcı Algoritması	Doğadan İlham Alan	Bitki tabanlı	2014
Tavuk Sürü Optimizasyonu	Doğadan İlham Alan	Biyo-esinli	2014
Çarpışan Cisimler Optimizasyonu			2014
Mercan Resif Optimasyonu			2014
Çiçek Tozlaşma Algoritması	Doğadan İlham Alan	Bitki tabanlı	2014
Radyal Hareket Optimizasyonu	Doğadan İlham Alan	Sürü tabanlı	2014
Örümcek Maymun Optimizasyonu	Doğadan İlham Alan	Biyo-esinli	2014
Futbol Ligi Yarışması	Doğadan İlham Alan	İnsan temelli	2014
Yapay Alg Algoritması			2015
Uyarlanabilir Boyutlu Arama			2015
Yabancılaşmış Karınca Algoritması			2015
Yapay Balık Sürüsü Algoritması	Doğadan İlham Alan		2015
Şişe Burunlu Yunus Optimizasyonu	Doğadan İlham Alan		2015
Kriket Algoritması			2015
Fil Arama Algoritması	Doğadan İlham Alan	Biyo-esinli	2015
Gri Kurt Doktoru	Doğadan İlham Alan	Biyo-esinli	2015
Jaguar Algoritması	Doğadan İlham Alan	Biyo-esinli	2015
Çekirge Sürüsü Algoritması	Doğadan İlham Alan	Sürü tabanlı	2015
Güve Alevi Optimizasyonu	Doğadan İlham Alan	Biyo-esinli	2015
Stokastik Fraktal Arama	Evrimsel temelli	-	2015
Girdap Arama Algoritması	Doğadan İlham Alan	Fizik/Kimya temelli	2015
Su Dalgası Optimizasyonu	Doğadan İlham Alan	Fizik/Kimya temelli	2015
Karınca Aslanı Optimize Edici	Doğadan İlham Alan	Biyo-esinli	2015
Afrika Mandası Optimizasyonu	Doğadan İlham Alan	Sürü tabanlı	2015
Yıldırım Arama Algoritması	Doğadan İlham Alan	Fizik/Kimya temelli	2015

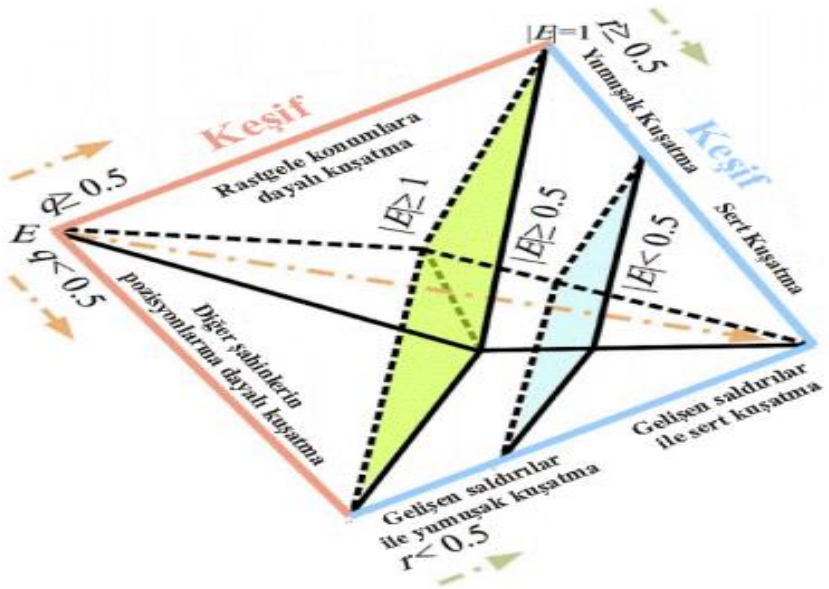
Mahalle Arayışında	Evrimsel temelli	-	2016
Karga Arama Algoritması	Doğadan İlham Alan	Biyo-esinli	2016
Elektromanyetik Alan Optimizasyonu	Doğadan İlham Alan	Fizik/Kimya temelli	2016
Ortak Operasyonlar Algoritması	Doğadan İlham Alan	Sürü tabanlı	2016
Aslan Optimizasyon Algoritması	Doğadan İlham Alan	Biyo-esinli	2016
Sinüs Kosinüs Algoritması	Doğadan İlham Alan	Fizik/Kimya temelli	2016
Virüs Kolonisi Araması	Doğadan İlham Alan	Biyo-esinli	2016
Balina Optimizasyon Algoritması	Doğadan İlham Alan	Biyo-esinli	2016
Kızıl Geyik Algoritması	Doğadan İlham Alan	Biyo-esinli	2016
Fototropik Optimizasyon Algoritması	Doğadan İlham Alan	Bitki tabanlı	2018
Coyote Optimizasyon Algoritması	Doğadan İlham Alan	Sürü tabanlı	2018
Baykuş Arama Algoritması	Doğadan İlham Alan	Biyo-esinli	2018
Sincap Arama Algoritması	Doğadan İlham Alan	Biyo-esinli	2018
Sosyal MühendislikOptimizasyon	Doğadan İlham Alan	İnsan temelli	2018
İmparator Penguen Optimizasyon	Doğadan İlham Alan	Biyo-esinli	2018
Sosyo Evrim ve Öğrenme Optimizasyonu	Doğadan İlham Alan	İnsan temelli	2018
Gelecek Arama Algoritması	Doğadan İlham Alan	İnsan temelli	2019
İmparator Penguenler Kolonisi	Doğadan İlham Alan	Sürü tabanlı	2019
Termal Değişim Optimizasyonu	Doğadan İlham Alan	Fizik/Kimya temelli	2019
Harris Hawks Optimizasyonu	Doğadan İlham Alan	Biyo-esinli	2019
Siyasi Optimize Edici	Doğadan İlham Alan	İnsan temelli	2020
Yığın Tabanlı Optimize Edici	Doğadan İlham Alan	İnsan temelli	2020
Renk Uyumu Algoritması	Sanat İlham Alan	Renk bazlı	2020
Stokastik Boya Optimizasyon	Sanat İlham Alan	Renk bazlı	2020
Gize Piramitleri İnşaatı	Antik Esinli	-	2020
Mayıs Sineği Optimizasyon Algoritması	Doğadan İlham Alan	Biyo-esinli	2020
Fire Hawk Optimazyon	Doğadan İlham Alan	Biyo-esinli	2022
Uçan Tilki Optimizasyon Algoritması	Doğadan İlham Alan	Biyo-esinli	2023

Literatürde DA motorun hızını kontrol etmek amacıyla PID kontrolörlerin kazanç parametrelerini en iyi şekilde ayarlamak için çeşitli yöntemler önerilmiştir. [38] DA motorun hız regülasyonu için bir PID kontrolör katsayılarını ayarlamak amacıyla MS algoritmalar başarıyla kullanılmıştır. Önerilen teknikler, seçilen bir amaç fonksiyonunu en aza indirerek PID denetleyicinin optimal parametrelerini elde etmek için kullanılmıştır [39].

Son yıllarda MS algoritmalarının mühendislik sistemlerinde kullanımı yaygınlaşmıştır. Harris şahinleri optimizasyon algoritması (HŞO), 2019'da Ali

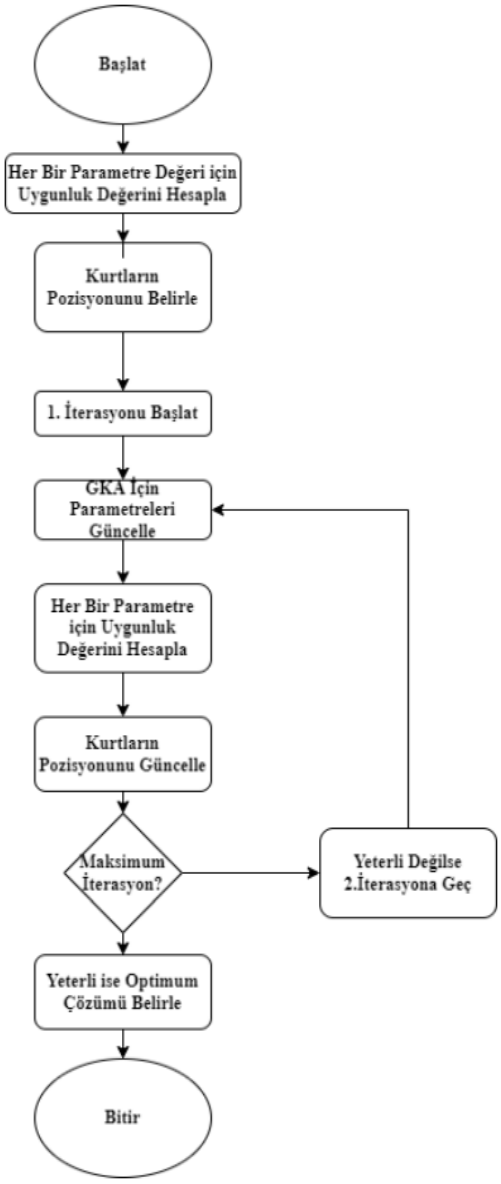
Asghar Heidari ve ekibi tarafından geliştirilen HŞO, doğadaki Harris şahinlerinin sürpriz saldırı olarak bilinen avlanma davranışlarından esinlenen, takım çalışması ve avlanma stratejilerine dayanan yeni bir popülasyon tabanlı algoritmadır. Bu algoritma, avın kaçma yetenekleri ve davranışlarıyla ilgili olarak, avın yakınında kısa süreli ve hızlı dalışlarla sürpriz saldırılar gerçekleştirebilen Harris şahinlerinin avlanma tarzlarını taklit etmektedir. Harris şahinleri, koşulların değişken doğasına ve avın kaçış kalıplarına göre çeşitli avlanma stratejileri sergileyebilmektedir. Bu işbirlikçi taktiklerin temel avantajı, Harris şahinlerinin avın tükenmişliğini tespit ederek izleyebilmesi ve kaçan avın savunma yeteneklerini geri kazanamamasını sağlamasıdır. Sonuç olarak, avın kaçma şansı kalmadığında, en güçlü ve tecrübeli Harris şahinlerinden biri tarafından kolayca yakalanır ve diğer sürü üyeleriyle paylaşılır [40].

Önerilen HŞO 'nun keşif ve sömürü aşamaları, bir avı keşfetme, sürpriz saldırı ve Harris şahinlerinin farklı saldırı stratejilerinden ilham alınarak modellenir. HŞOA, popülasyona dayalı, gradyan içermeyen bir optimizasyon tekniğidir; dolayısıyla herhangi bir şeye uygulanabilir. Optimizasyon problemi uygun bir formülasyona tabidir. Şekil 14 ‘de Harris şahin optimazyonunun tüm fazları gösterilmiştir [40].



Şekil 14 : HŞO Tüm Fazlar

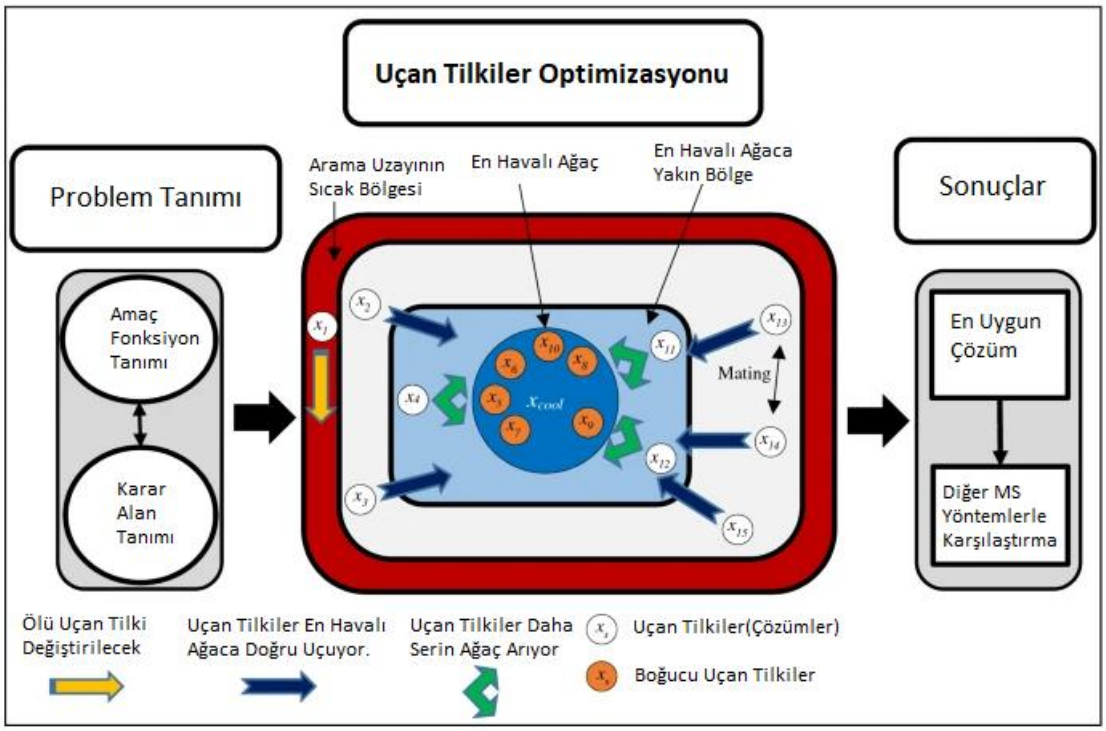
Gri Kurt Optimizasyonu (GKO) algoritması, doğadaki gri kurtların (Canis lupus) liderlik yapılarını ve avlanma stratejilerini modelleyen bir optimizasyon yaklaşımıdır. Alfa, beta, delta ve omega gibi dört farklı tür gri kurt, liderlik rollerini simüle etmek için kullanılır. Avın aranması, avın etrafını sarma ve avın saldırıya uğraması şeklinde üç temel adımdan oluşur. Algoritma daha sonra 29 tanınmış test fonksiyonu üzerinde değerlendirilmiş ve Parçacık Sürü Optimizasyonu, Yerçekimi Arama Algoritması, Diferansiyel Evrim, Evrimsel Programlama ve Evrim Stratejisi gibi diğer meta-sezgisel yöntemlerle karşılaştırılmıştır. Bu karşılaştırma, GKO algoritmasının rekabetçi sonuçlar sunduğunu ortaya koymaktadır. Ayrıca, gerilme/basma yayı, kaynaklı kiriş ve basınçlı kap tasarımı gibi üç temel mühendislik probleminin çözümü üzerinde durulmuş ve önerilen yöntemin optik mühendislik alanında gerçek bir uygulama örneği sunulmuştur. Klasik mühendislik tasarım problemlerinin sonuçları ve gerçek uygulama, bu algoritmanın zorlu problemlere uygulanabilirliğini göstermektedir. Şekil 15 ‘de Gri kurt optimizasyon algoritmasının akış şeması gösterilmiştir [41].



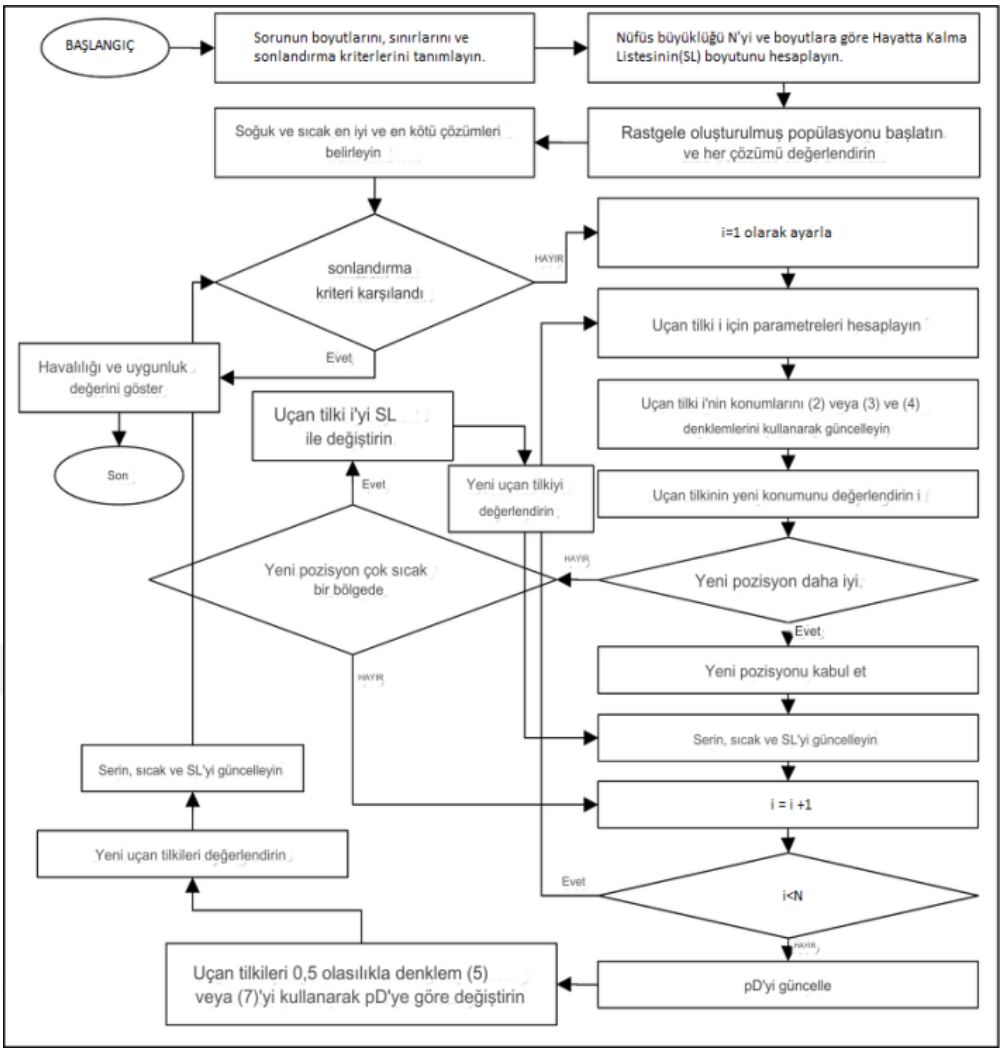
Şekil 15 : GKO Algoritma Akış Şeması

Uçan Tilkiler Optimizasyonu (UTO), uçan tilkilerin sıcak hava dalgalarında hayatta kalma stratejilerinden esinlenen bir küresel optimizasyon algoritmasıdır. Bu yeni yöntem, her bir çözüm için parametrelerin belirlenmesinde Bulanık Mantık tekniklerini kullanarak, parametresiz bir optimizasyon algoritması sunar. UTO 'nun etkinliğini değerlendirmek için, CEC2017 test fonksiyon paketi ve üç gerçek dünya mühendislik problemi gibi 56 kıyaslama fonksiyonu kullanılmıştır. Bu algoritmanın performansı, en son meta-sezgisel yöntemlerle karşılaştırıldığında, küresel optimizasyon alanında etkili ve cazip bir alternatif olduğunu göstermektedir. Şekil 16

‘da ve Şekil 17 ‘de Uçan tilki optimizasyon algoritmasının grafiksel ve akış diyagramı gösterilmiştir [42].



Şekil 16 : Uçak Tilki Optimizasyon Grafiksel Özet



Şekil 17 : UTO Algoritma Akış Şeması

Balina Optimizasyon Algoritması (BOA), kambur balinaların doğal davranışlarından esinlenen ve Balina Optimizasyonu olarak bilinen yeni bir meta-sezgisel optimizasyon yaklaşımını tanıtmaktadır. Bu algoritma, balinaların kabarcık ağları avlama stratejisine dayanmaktadır. BOA, 29 matematiksel ve 6 yapısal tasarım problemleri üzerinde test edilmiş ve elde edilen sonuçlar, bu algoritmanın geleneksel yöntemlerle kıyaslandığında oldukça etkili olduğunu ortaya koymaktadır [43].

Güve Alevi Optimizasyonu (GAO) algoritması, doğadan esinlenerek geliştirilmiş yeni bir optimizasyon paradigmasıdır. Bu algoritma, güvelerin gece uçuşlarında gösterdikleri aya göre sabit bir açı koruma stratejisinden ilham alır. Güveler, uzun mesafeler boyunca düz bir çizgide seyahat etmek için bu yöntemi kullanarak, doğadaki etkili navigasyon yeteneklerini sergilerler. Ancak, bu tür

güveler yapay ışıkların etrafında dönerek yolunu kaybedebilirler, bu da onları etkili bir şekilde hedeflerine ulaşmalarını engeller. GAO algoritması, bu doğal davranışı matematiksel olarak modelleyerek, optimizasyon problemlerini çözmek için bir araç olarak geliştirilmiştir. Bu algoritma, 29 kıyaslama ve 7 farklı mühendislik problemleri üzerinde test edilmiş ve diğer doğadan esinlenen algoritmalarla karşılaştırılarak başarılı sonuçlar elde etmiştir. Yapılan kıyaslama çalışmaları, GAO'nun çeşitli optimizasyon problemlerinde etkili ve rekabetçi bir performans sergilediğini göstermektedir. GAO algoritmasının pratik uygulamalardaki etkinliği de ele alınmaktadır. Örneğin, deniz pervanesi tasarımı gibi gerçek dünya problemleri üzerinde GAO'nun başarılı bir şekilde uygulandığı gözlemlenmiştir. Bu uygulamalar, GAO'nun kısıtlı ve belirsiz arama uzaylarına sahip zorlu problemleri çözmedeki potansiyelini vurgular [44].

1.8. PID

PID kontrolörü, oransal, integral ve türev terimlerinden oluşan bir kontrol sistemidir. Bu üç kontrolör terimi bir araya gelerek bir kontrol stratejisi oluşturur ve proses kontrolünde kullanılır. PID denetleyicisi, basınç, hız, sıcaklık, akış vb. süreç değişkenlerini yönetir. Bazı uygulamalarda, kontrolü sağlamak için iki veya daha fazla PID kontrolörü kullanılarak kademeli bir ağ oluşturulur [45].

PID kontrolcüsü, geri beslemeye dayalı bir kontrol cihazıdır ve otomatik kontrol sistemlerinde sıkça kullanılan bir kontrol tekniğidir. Giriş ve çıkış arasındaki sapmayı düzeltmek için tasarlanmıştır ve bir süreci uygun şekilde ayarlamak için kontrol sinyali üretir. PID kontrolcüsü, orantılı, integral ve türevsel yanıtları içeren üç temel bileşene sahiptir. Bu, endüstride yaygın olarak kullanılan hassas bir kapalı döngü kontrol sistemidir [46].

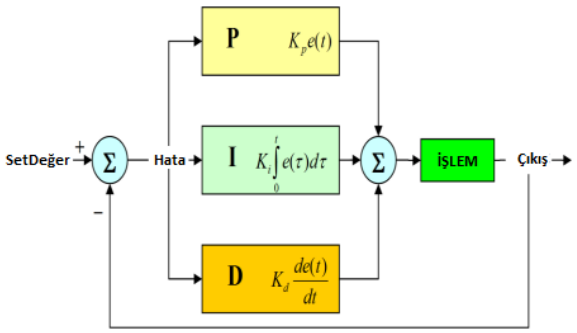
PID kontrolörün diferansiyel denklemi [46]:

$$u(t) = K_P e(t) + K_i \int e(t)d + K_d \frac{de(t)}{dt}$$

PID denetleyici, çok sayıda endüstriyel uygulamada kullanılan bu tür kontrol şemalarından biridir [47]. PID, "proportional-integral-derivative" kelimelerinin

kısaltmasıdır ve PID denetleyici, bu üç bileşeni içeren bir kontrol sistemi olarak bilinir. Bu denetleyici, kontrol edilen sistemin hatasını azaltırken türev bileşeni, daha gelişmiş bir çıktı sağlar. Bu nedenle, PID denetleyicisi, seksen yıldan fazla bir süredir tercih edilme nedenlerinden biridir [48]. Sırasıyla Kp, Ki, Kd olarak gösterilen oransal, integral ve türev kazançları parametreleri kontrollü süreçten istenen çıktıyı elde edecek şekilde ayarlanır [49].

Oransal yanıt (Kp), sistemin cevap oranını arttırırken kararlı durum hatasını azaltır, ancak tamamen ortadan kaldırmaz. İntegral yanıt (Ki), birikmiş hataları gidererek kararlı durum hatasını azaltır, ancak tamamen ortadan kaldırmaz. Türevsel yanıt (Kd), sistem kararlılığını artırır, aşırı sönümlemeyi azaltır ve sistem cevap hızını iyileştirir. Şekil 18 ‘da PID blok diyagramı gösterilmektedir.



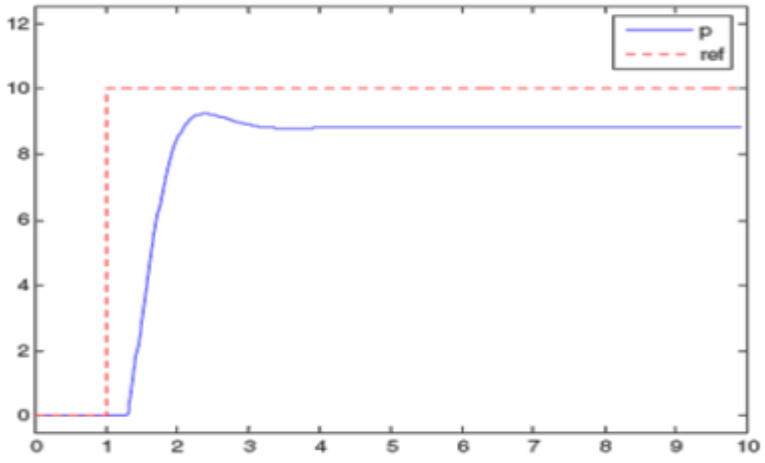
Şekil 18 : PID Blok Diyagram

Kp, Ki, Kd'nin kapalı sistem üzerindeki etkileri dikkate alındığında Tablo 3 ‘de gösterildiği gibi özetlenmiştir [46].

Tablo 3 : PID Parametre Artırma Azaltma Etkileri

Parametre	**Yükseliş Zamanı**	**Aşım**	**Yerleşme Zamanı**	**Kararlı Durum Hatası**	**Kararlılık**
Kp	Azalır	Artar	Küçük Değişim	Azalır	Geriler
Ki	Azalır	Artar	Artar	Büyük Artış	Geriler
Kd	Küçük Değişim	Azalır	Azalır	Etki Yok	İyileşir

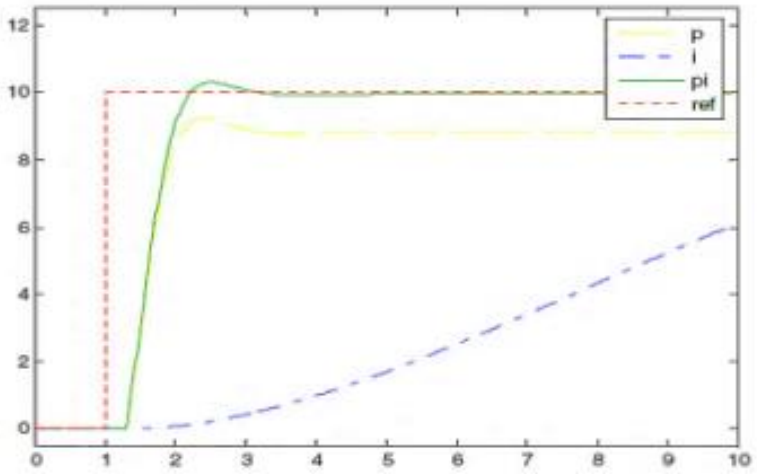
Oransal Kontrolör, mevcut hata ile orantılı bir kontrol sinyali üretir. Hata, set değer ile mevcut ölçüm arasındaki farktır. P bileşeni, hatanın büyüklüğüne göre kontrol sinyalini ayarlar. P bileşeni, sistemin hızlı bir şekilde hedefe yaklaşmasını sağlar, ancak sistemden gelen aşırı yanıtlara neden olabilir. Hatanın devam ettiği durumlarda, kontrol çıkışında herhangi bir değişiklik olmaz. Şekil 19 'da oransal durum grafiği gösterilmiştir.



Şekil 19 : Oransal Durum

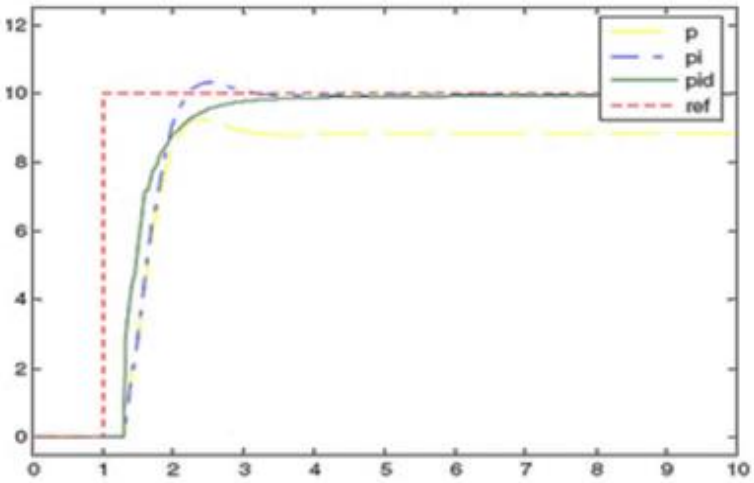
Eğer yalnızca oransal kontrol kullanılıyorsa ve sistemin giriş değeri istenen değere eşit değilse veya 0 değilse, statik hata her zaman ortaya çıkar. Kp değeri aşırı büyükse, sistem kararsız hale gelebilir.

İntegral kontrolör, önceki hataların birikimini kontrol değerlerine ekleyerek çalışır. Bu yöntem, ayarlanan değere ulaşıncaya kadar hataların toplamını sürekli olarak hesaplar ve böylece sistem dengelendiğinde hatayı minimize eder. İntegral terimi genellikle Oransal terimi ile birlikte kullanılarak bir PI denetleyici oluşturur. İntegral bileşen, sistemin tepkisini yumuşatırken, PI denetleyici kararlı durum hatasını giderir. Şekil 20 'de oransal ve integral durum grafiği gösterilmiştir.



Şekil 20 : Oransal, İntegral ve Oransal İntegral (PI) Durum

Türev kontrolörü kullanarak kapalı çevrim kararlılığı genellikle artar, ancak Integral kontrolör, hatanın gelecekteki davranışını tahmin etme yeteneğine sahip değildir. Bu nedenle, ayar noktası değiştirildiğinde, kontrolör normal bir tepki verir. Türev kontrolcüsü, hatanın gelecekteki davranışını tahmin ederek bu sorunu çözer. Çıkışı, hatanın zamanla değişim oranına bağlıdır ve türev sabiti ile çarpılır. Bu, sistem yanıtını artırarak çıkış için bir başlangıç sağlar. Şekil 21 'de oransal, integral, türev durum grafiği gösterilmiştir.



Şekil 21 : PID Durum

1.9. Optimal Kontrol

Optimal kontrolün temel amacı kontrol sinyallerini belirlemektir. Sistemlerin performansını değerlendirmeden önce simülasyon ortamında test edilmesi önemlidir. Bu, gerçek zamanlı çalışma sırasında olası hataların önceden belirlenmesine ve algoritmaların geliştirilme sürecinde sorunların tanımlanmasına yardımcı olur. Simülasyonlar, sistemin beklenen davranışlarını modelleyerek gerçek dünya senaryolarını taklit eder ve bu sayede sistemlerin doğru ve güvenilir bir şekilde işleyişini sağlar. DA motorun hız takibi için optimal kontrol tekniklerinden lineer karesel regülatör (lineer karesel regülatör-LQR) ve oransal integral diferansiyel (oransal-integral-türev PID) yöntemleri kullanılabilir [50].

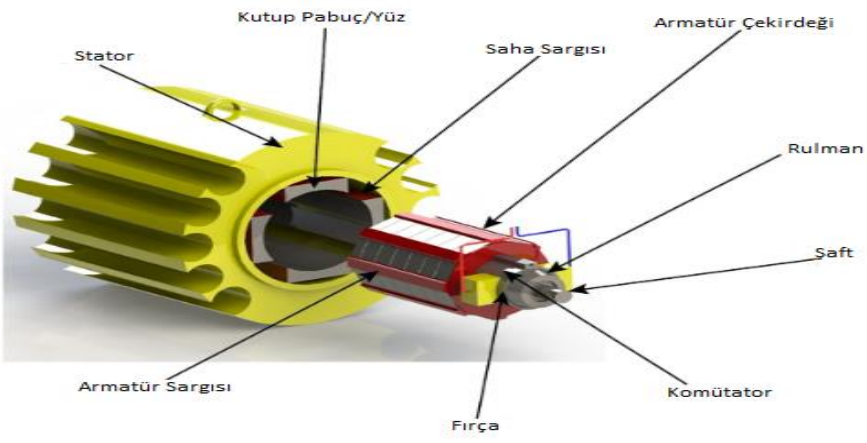
DA motorların kontrolü, özellikle havacılık ve endüstriyel uygulamalarda yaygın olan bir gereksinimdir. Bilgisayar teknolojisinin gelişimiyle birlikte, daha karmaşık kontrol yöntemlerinin kullanımı artmıştır. Bu kontrol stratejileri, motorun performansını ve enerji verimliliğini artırmayı hedefler. Optimal kontrol teorisi temelli regülatörler, bu karmaşıklıkla başa çıkmak için giderek daha fazla tercih edilmektedir. Farklı kontrol algoritmalarının etkinliği, geleneksel PID yöntemleri ile karşılaştırılarak incelenmiştir. Kontrol algoritmalarının sentezi, DA motorların güç tüketimini optimize etmeyi amaçlar. Sonuç olarak, önerilen yöntemin, diğer alternatiflere göre daha iyi performans sergilediği görülmüştür [51].

DA motorun hız kontrolü için oransal-integral türev (PID) kontrol cihazının optimal parametrelerinin belirlenmesi Sinüs Kosinüs Algoritmasının (SKA) uygulanmasıyla ilgilidir. SKA, sinüs ve kosinüs fonksiyonlarına dayalı bir matematiksel model kullanarak bunları en iyi çözüme doğru yönlendiren, birden fazla başlangıç aday çözümü oluşturan, yakın zamanda önerilen bir optimizasyon tekniğidir. Burada zamanla çarpılan mutlak hatanın (ITAE) integrali, PID kontrol cihazının parametrelerinin SKA tarafından ayarlanması için amaç fonksiyonu olarak alınmıştır. Önerilen SKA/PID şemasının mevcut diğer tekniklerle karşılaştırılması da sunulmuştur. ITAE'nin amaç fonksiyonu olduğu önerilen SKA/PID yaklaşımının karşılaştırılabilir hedef aşımı ve diğer parametreleri sağladığı gözlemlenmiştir; çökelme ve yükselme süreleri literatürdeki mevcut yaklaşımlarla karşılaştırıldığında daha azdır. DA motor parametrelerindeki değişikliklere karşı SKA/PID yaklaşımının sağlamlık analizi yapılmıştır [52].

DA motorlar, klasik kademeli kontrol mimarisiyle karşılaştırıldığında durum geri besleme kontrolü, tasarım karmaşıklığı, donanım gerçekleştirilmesi ve uyarlanabilirlik açısından avantajlar sunar. Deneysel verilerden tanımlanan durum uzayı modeli, doğrusal ikinci dereceden düzenleyici (LQR) tasarımının temelini oluşturur. Durum geri beslemeli doğrusal kontrol, Coulomb sürtünmesinin telafisi için ileri beslemeli kontrol ile güçlendirilmiştir [53].

1.10. DA Motor

Doğru akım (DA) motoru veya DC motor, elektrik enerjisini manyetik bir alan oluşturarak mekanik enerjiye dönüştüren bir elektrikli makinedir. DA motor, güç uygulandığında statorunda bir manyetik alan oluşturur. Bu alan, rotor üzerindeki mıknatısları etkileyerek rotorun dönmesine neden olur. Rotorun sürekli dönmesini sağlamak için, güç kaynağına bağlı olan fırçalar ve komütatör, motorun tel sargılarına akım sağlar. Stator, hareketsiz ana gövdeyi sağlayarak motora destek ve koruma sağlar. Stator, motorun hareket eden bölümü olan rotoru veya armatürü tahrik eden dönen bir manyetik alan üretir. Bu alan sargılarını barındırır ve elektrik beslemesini terminaller aracılığıyla alır; bu motorun sabit kısmıdır. Sargılar ve komütatör, genellikle çelik gibi sertleştirilmiş bir metal mil üzerinde bulunan motorun merkezinde yer alır ve motorun yüklerine dayanması için destek sağlar. Komütatör çubukları, plastik bir kalıp aracılığıyla mili tutan plakaya sabitlenir. Sargılar tarafından üretilen tork, stator tarafından desteklenen mili aktarır. Mil, statorun içinden çıkar ve motoru uygulamaya bağlar. Şekil 22 ‘de DA motor diyagramı gösterilmiştir [54].



Şekil 22 : DA Motor Diyagram

DA motorlar, iki terminalden oluşur: bir pozitif ve bir negatif. Pozitif kablo pozitif terminalle, negatif kablo ise negatif terminalle bağlandığında, motor saat yönünde döner. Bağlantılar ters çevrildiğinde, motorun dönüş yönü de tersine döner. Bu terminaller, güç kaynağını motorun arkasındaki fırçalara ve fırça kollarına bağlar. DA motorlarda kullanılan mıknatıslar genellikle kalıcı mıknatıslardır ve sürekli olarak manyetik alan oluşturur. Mıknatısların zıt uçları birbirini çekerken, aynı uçları birbirini iter. Bir mıknatısın manyetik alanı, güney kutbundan kuzey kutbuna doğru uzanır ve en güçlü kısmı uçlarında bulunur. İki mıknatıs bir araya geldiğinde, çok güçlü bir manyetik alan oluşturur; bu nedenle DA motorlarda genellikle rotorun etrafında iki mıknatıs bulunur, böylece güçlü manyetik alan rotor boyunca hareket eder [54].

Rotor veya armatür, birbirinden izole edilmiş lamine disklerden oluşur. Bu disklerin izole edilmesi, büyük girdap akımlarının oluşmasını önler. Rotor, motorun dinamik bileşenidir ve bobin sargıları etrafına sarılır, böylece manyetik bir alan oluşturur. Her tel, içinden elektrik geçtiğinde zayıf bir manyetik alan üretir. Sarma işlemi, döndürülen her bölgenin aynı zayıf manyetik alanla etkileşime girmesini sağlar. Farklı teller bir araya geldiğinde, güçlü bir manyetik alan oluşur. Rotora daha fazla bobin eklenmesi, dönüşün daha dengeli olmasını sağlar. İki bobinli motorlar sıkışma ve durma eğilimindedir, bu nedenle tüm DA motorlarda en az üç bobin bulunur. Her bobin bir öncekine 120 derecelik bir açıyla yerleştirilir [54].

DA motorun fırçaları, enerji ileten ve yay gibi hareket eden metal parçalardır. Bunlar, bir ucu karbon malzemeden yapılmış iletken bir malzeme ile kaplıdır ve diğer ucu ise güç kaynağının motora uygulandığı bir pimdir. Fırçalar, yayın itici kuvvetiyle komütatöre doğru hareket eder, fırça kolları tarafından tutulur ve terminal veya elektrik kaynağına doğrudan bağlanır. Komütatör, rotorun dönmesiyle birlikte hareket eden küçük bakır plakalardan oluşur. Rotorun hareketi, güç kaynağının bobinlere giden kutuplarının değişmesine neden olur. Her bir bobin, elektriksel olarak izole edilmiş ancak birbirine bağlı iki komütatör plakasına bağlanır. Bu plakalar aracılığıyla, pozitif ve negatif terminaller arasında akım kolayca geçer ve bir elektromanyetik alan oluşturulur [54].

Hız kontrolü için otomatik ayarlı oransal-integral kontrolör kullanılmıştır ve kontrol algoritması programlanabilir lojik kontrolör tarafından yürütülerek orantılı

integral kazançları bulanık mantık kullanılarak belirlenmiştir. Geleneksel oransal-integral kontrolörle karşılaştırılmıştır. Bulanık mantık, MATLAB/Simulink yazılımı aracılığıyla ayrı bir bilgisayarda çalıştırılmıştır. Bu yöntem, programlanabilir lojik kontrolör üzerindeki yükü azaltarak aynı zamanda geleneksel oransal-integral denetleyicinin dezavantajlarını ortadan kaldırmıştır [55].

DA motorlar, endüstriyel ve ev kullanımlarında büyük bir öneme sahiptir. Dolayısıyla, DA şönt motorların hız kontrolü, öncelikli bir gerekliliktir. DA şönt motorların hız kontrolü için geleneksel oransal integral (PI) ve oransal integral türev (PID) kontrolcülerinin performansı karşılaştırıldığında PI kontrolörün performansının diğer kontrolörlere göre tatmin edici olduğu görülmektedir [56].

Doğru akım (DA) motorlar, kontrol edilebilirlikleri nedeniyle değişken hız ve yük gereksinimlerine sahip birçok endüstriyel uygulamada önemli bir ekipman parçasıdır. Bugün, piyasada birçok yüksek kaliteli motor hız kontrol cihazı bulunmaktadır, ancak bunların maliyetleri genellikle yüksektir. Bu durumda, düşük maliyetli ve yüksek performanslı bir hız kontrol cihazı, özellikle küçük mobilite uygulamaları için oldukça cazip olacaktır. DA motorun hızının hem ileri hem de geri yönde, tamamen kapalı durumdan tamamen açık duruma kadar kontrolü, LM324 işlemsel yükselteç modeline dayalı Darbe Genişlik Modülasyonu (PWM) kullanılarak gerçekleştirilmiştir. Önerilen sistem, basit yapısı, düşük maliyeti, doğruluğu, yüksek verimliliği, hafif ve kompakt yapısı, küçük boyutu ve çift yönlü hız kontrolü gibi birçok avantaj sunmaktadır [57].

Genetik algoritma kullanarak PID parametrelerini seçerek bir DA motorun konum kontrolörünün, PID parametrelerinin genetik algoritma ile ayarlanmasının Ziegler ve Nichols yönteminden daha iyi olduğu bulunmuştur. Önerilen yöntem, daha karmaşık sistemlerde de başarıyla kullanılabilir [58].

DA motor, elektrik enerjisini mekanik enerjiye dönüştüren bir türdür ve bu dönüşüm için alan bobini üzerine doğru akım voltajı uygulanır. DA motorun dönüş hızını sabit tutmak için, yük değişimlerine karşı PID (Proportional Integral Derivative) kontrol yöntemi kullanılarak hız stabilitesi sağlanır. Bu stabilite kontrol sistemi, yüklü DA motor için PWM (Pulse Width Modulation - Darbe Genişlik Modülasyonu) prensibine dayanır [59].

PLC birçok farklı uygulamada temel araç olarak önerilmektedir. PID denetleyici, sistemden gelen bilgilere ve otomatik ayarlamaya dayalı olarak motor hızını kontrol etmek için tasarlanmıştır. Kontrol sistemi güçlü Matlab ve Simulink yazılımı ile simüle edilmiştir. Simülasyon sonuçları aynı zamanda yükselme süresini, kararlı durum hatasını, aşımı azaltan ve sistem kararlılığını artıran daha iyi motor performansı göstermektedir [60].

Hassas hız kontrolü, verimli endüstriyel otomasyon için önemli bir gerekliliktir. Doğru akım (DA) motorları bu amaçla yaygın olarak kullanılmaktadır. Geleneksel yöntemde, DA motorun hızının kontrol edilmesi için analog devreler kullanılır. Analog devrenin AT89S52 mikrodenetleyici devresi ile değiştirilmesiyle darbe genişlik modülasyonu tekniği kullanılarak DA motorun dijital hız kontrolü kullanmıştır [61].

Hem endüstriyel hem de otomotiv sektöründe sıkça tercih edilen doğru akım (DA) motor türlerinden biri, fırçasız DA motorlar olarak bilinir. Bu motorlar, MATLAB/Simulink kullanılarak geliştirilmiş ve hızı üç farklı kontrolör tipiyle test edilmiştir. Birinci kontrol cihazı, geleneksel bir orantısal integral türev (PID) kontrolörüdür ve fırçasız DA motor hızını kontrol etmek için kullanılır. İkinci kontrolör, PID ile parçacık sürüsü optimizasyonu (PSO) yaklaşımını kullanarak fırçasız DA motor hızı için en iyi tepkiyi sağlar. Üçüncü yöntemdeki PID denetleyicisi, sinir ağına dayalı bir yapıya sahiptir ve motor hızında en iyi tepkiyi sunar [62].

DA motorun matematiksel modeliyle MATLAB/SIMULINK'te PID kullanımıyla PID denetleyicisinin simülasyonu, Ziegler-Nichols yöntemi ve deneme-yanılma yaklaşımı kullanılarak geliştirilmiştir [63].

Endüstriler, DA motorlar gibi birçok sürücü ve aktüatörü kullanarak çalışır. DA motorlar, geniş hız aralıkları ve ayarlanabilir hızları nedeniyle birçok endüstride tercih edilir. Ancak, DA motorlar harici bozulmalara ve parametre değişikliklerine hassasiyet gösterir, bu da hızlarının istikrarsız olmasına neden olur. Bu nedenle, DA motorlar, istikrarlı ve hızlı bir hız elde etmek için düşük bir durum hatasıyla çalışacak uygun bir kontrolör tasarımına ihtiyaç duyar. PID kontrol yöntemine dayalı olarak, kontrolör kazançlarının deneme-yanılma ve MATLAB Tuner ile bir tanımlama sistemi ile ayarlandığı bir kontrolör tasarlanmıştır [64].

1.11. OPC

OPC, endüstriyel otomasyonda güvenli ve güvenilir veri alışverişi için birlikte çalışabilirlik standardını temsil eder. Platformdan bağımsız olarak çalışabilen bu standart, birden fazla satıcının cihazları arasında kesintisiz bilgi akışını sağlar. OPC Vakfı, bu standartın geliştirilmesi ve sürdürülmesinden sorumludur. Endüstri satıcıları, son kullanıcılar ve yazılım geliştiricileri tarafından geliştirilen OPC standardı, gerçek zamanlı verilere erişimden alarmların ve olayların izlenmesine, geçmiş verilere ve diğer uygulamalara erişime kadar bir dizi spesifikasyonu içerir. Bu spesifikasyonlar, istemciler ve sunucular ile sunucuların kendi arasındaki ara yüzü tanımlar [65].

Endüstriyel otomasyon alanında çalışanlar için, farklı marka cihazların tek bir ortamda veri aktarımı veya cihazların birbirleriyle iletişim kurması sıklıkla zorluklarla karşılaşılan bir durumdur. Bu zorlukların temel sebebi, cihaz veya sistem üreticilerinin endüstriyel haberleşme için kendi özel protokollerini kullanmalarıdır. Bu durum, hem üreticileri hem de son kullanıcıları olumsuz etkilemektedir. OPC kavramı, bu engelleri ortadan kaldırmak için endüstriyel otomasyon sistemleri ve haberleşme sistemleri üreticileri ile otomasyon firmalarının bir araya gelmesiyle geliştirilmiştir [66].

OPC, Microsoft'un Object Linking & Embedding (OLE)/ Component Object Model (COM) standardına dayanmaktadır. Bu teknoloji, farklı uygulamalar arasında nesne tabanlı entegrasyonu sağlayan bir yapıya sahiptir. OPC ise OLE tabanlı bir haberleşme standardı olarak hızlı ve güvenilir bir bağlantı sağlar, bu da farklı otomasyon seviyeleri arasında verimli bir iletişimi mümkün kılar. OLE teknolojisinin kullanımıyla OPC, yönetim katındaki uygulamalardan proses denetimine kadar farklı alanlarda, standart bir şekilde veri alışverişini sağlar [67].

OPC, veri alışverişi için istemci-sunucu modelini benimser. OPC sunucusu, çeşitli cihazlardan gelen proses verilerini toplar ve bunları kullanıcı ara yüzü aracılığıyla istemcilere ileterek iletişimi sağlar. OPC uyumlu bir istemci, OPC sunucusuna bağlanarak gereksinim duyduğu verilere erişebilir [68].

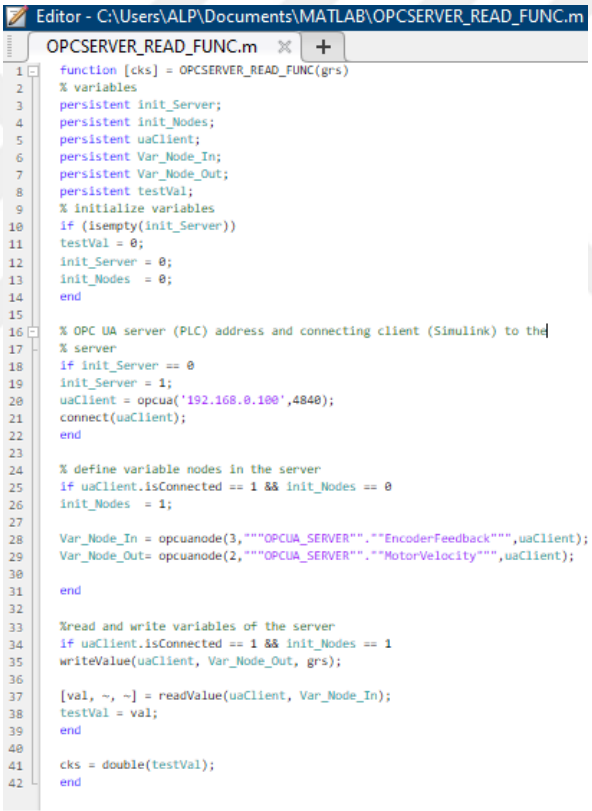
OPC, donanım üreticileri ile yazılım üreticileri arasında bir çizgi çizer. Denetleyiciler ve bilgisayarlar arasında gerçek zamanlı veri alışverişi ve paylaşım istekleri OPC ile daha standart hale getirilmiştir. Her iki taraf için de OPC standart

ara yüzünü kullandığımızda, donanım bileşenlerinde veya yazılım uygulamalarında yapılacak herhangi bir değişikliğin/iyileştirmenin veri iletişimine olumsuz bir etkisi olmayacaktır [69].

2. MATERYAL VE YÖNTEMLER

2.1. MATLAB

Matlab, endüstriyel otomasyon sistemlerinin modelleme, simülasyon, kontrol ve analizinde yaygın olarak kullanılan bir araçtır. Bu tez çalışmasında, MATLAB ile PLC arasındaki haberleşme Matlab OPCToolbox kullanılarak sağlanmıştır. PLC OPC sunucu, Matlab OPC istemci olarak kullanılmıştır. Oluşturulan haberleşme sonrasında PLC ile Matlab arasında asenkron bir iletişim sağlanmıştır. PLC den okunan DA motor durum ve kontrol verilerinin Matlab da kaydı gerçekleştirilmiştir. Şekil 23 ‘de Matlab da OPC fonksiyon gösterilmektedir.



Şekil 23 : Matlab OPC Toolbox Fonksiyon

Matlab “Command Window” komut penceresi üzerinden adımlar izlenir.

- uaClient = opcua('192.168.0.100',4840); bilgisi girilerek belirtilen IP adresi ve port numarasına sahip olan OPC sunucusuna MATLAB üzerinden bağlantı kurulmuştur.
- connect(uaClient); komutu ile bağlantı oluşturumuştur.

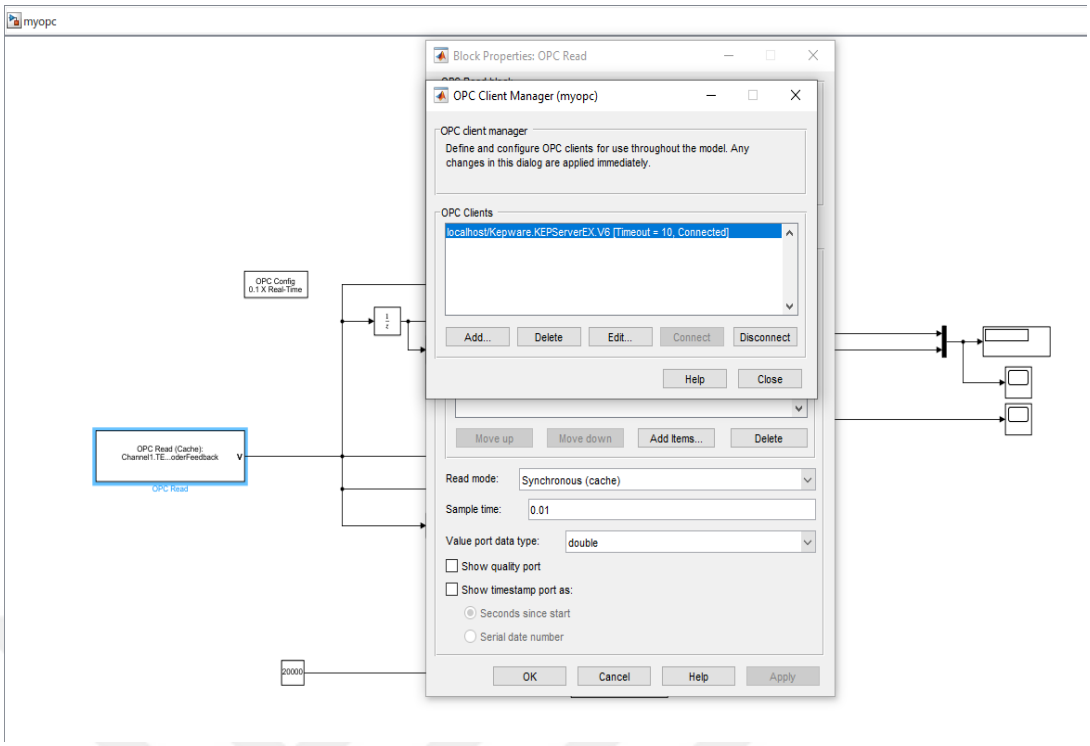
- ServerNodes = browseNamespace(uaClient); komutu ile veri isimlerini listeleme işlemi gerçekleştirilmiştir.
- readValue(uaClient,opcuanode(4,3)); komutu ile veri okuma işlemi gerçekleştirilmiştir.
- writeValue(uaClient,opcuanode(4,3)); komutu ile veri yazma işlemi gerçekleştirilmiştir.

2.2. Simulink

Matlab Simulink, MATLAB'ın bir bileşeni olan ve grafiksel modelleme ile simülasyon yetenekleri sunan bir yazılım aracıdır. Sistemleri blok diyagramları şeklinde modelleme ve simülasyon yapma olanağı sunarak karmaşık sistemlerin tasarımını ve analizini kolaylaştırır. Bu tez çalışması, DA motor kontrolü PLC ile Matlab Simulink arasındaki veri alışverişi ve kontrolünü sağlamak için OPC haberleşme protokolü kullanılarak gerçekleştirilmiştir. OPC haberleşmesi için KEPServerEx istemcisi kullanılmış ve Simulink ortamında OPC Toolbox araçlarından yararlanılmıştır.

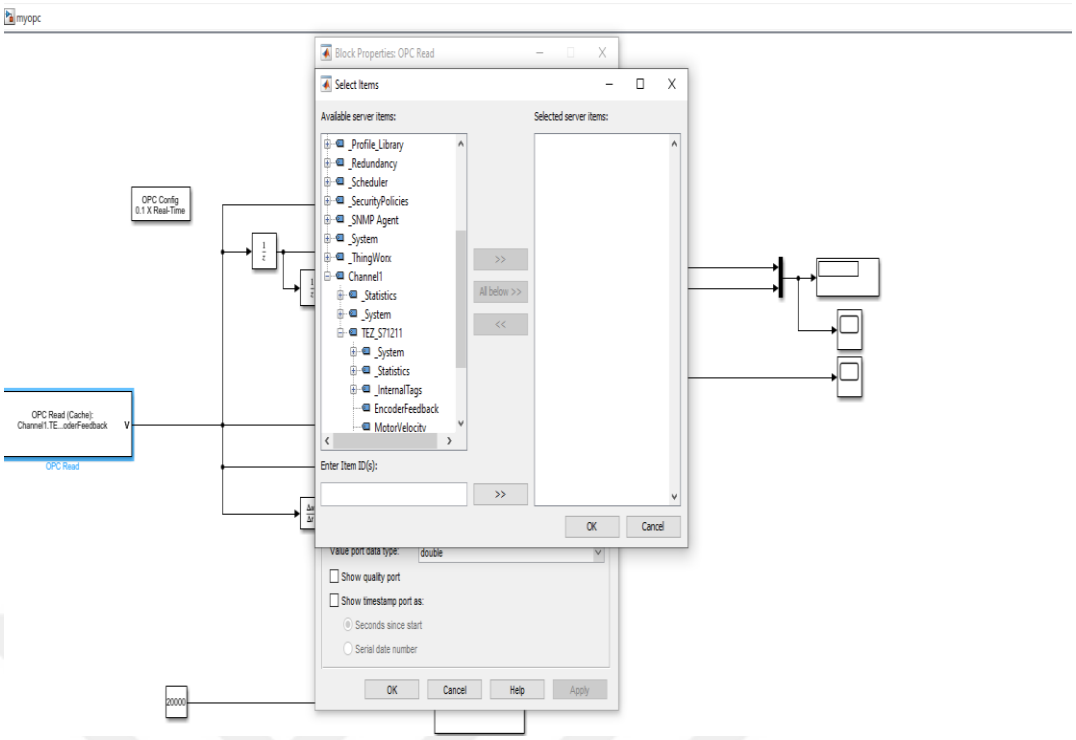
Simulink ortamında OPC Toolbox kullanılarak DA motor kontrolü gerçekleştirilmiştir. Bu, PLC'deki durum ve kontrol verilerinin KEPServerEx aracılığıyla Simulink ortamına aktarılması ve Simulink'te OPC Config, OPC Read ve OPC Write OPCToolbox'larını kullanarak veri alışverişinin sağlanmasıyla gerçekleştirilmiştir.

Simulink ortamında “Simulink Library Browser” bölümden OPC Toolbox seçilerek OPC Config fonksiyonu simulink çalışma ortamında oluşturulmuştur. OPC Config fonksiyonu OPC haberleşme protokolü kullanılarak farklı cihazlar arasında veri alışverişini yapılandırmak ve yönetmek için kullanılan bir araçtır. “OPC Config” ile KEPServer üzerinde oluşturulan OPC sunucusuyla haberleşilebilmesi için KepServerEx sunucu eklenmiştir. Şekil 24 ‘de OPC Toolbox OPC Config KEPserver ayarı gösterilmiştir.



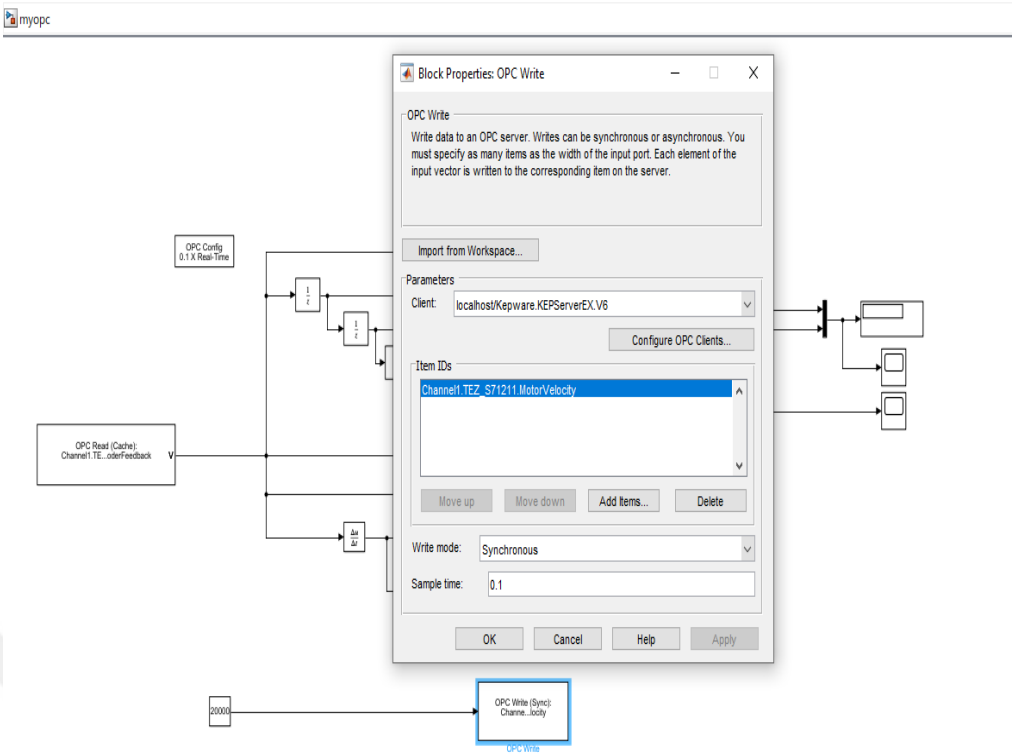
Şekil 24 : OPC Toolbox OPC Config KepServer Ayarı

"OPC Read" fonksiyon işlevi, bir OPC sunucusundan veri okuma işlemini gerçekleştirir. OPC Read fonksiyonuyla, OPC Config fonksiyon ile eklemiş olduğumuz sunucu KepServerEx üzerindeki sunucusundan DA motor enkoder verisi seçilmiş ve okuma işlemi yapılmıştır. Şekil 25 'de OPC Toolbox OPC read fonksiyonu kepserver veri tanımlaması gösterilmiştir.



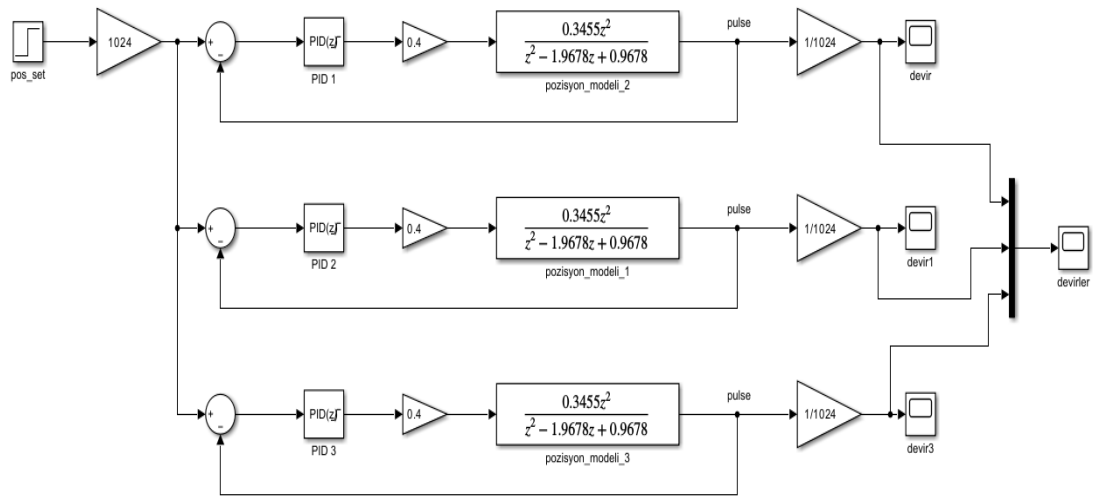
Şekil 25 : OPC Toolbox OPC Read Fonksiyonu KepServer Veri Tanımlaması

OPC Write işlevi, Simulink modelindeki bir blok aracılığıyla bir OPC sunucusuna veri yazma işlemini ifade eder. OPC Write fonksiyonu, OPC Config fonksiyon ile eklemiş olduğumuz KepServerEx üzerindeki sunucuya DA motor ayar hız değeri yazma işlemi yapılmıştır. Şekil 26 ‘de OPC Toolbox OPC Write fonksiyonu veri tanımlaması gösterilmiştir.

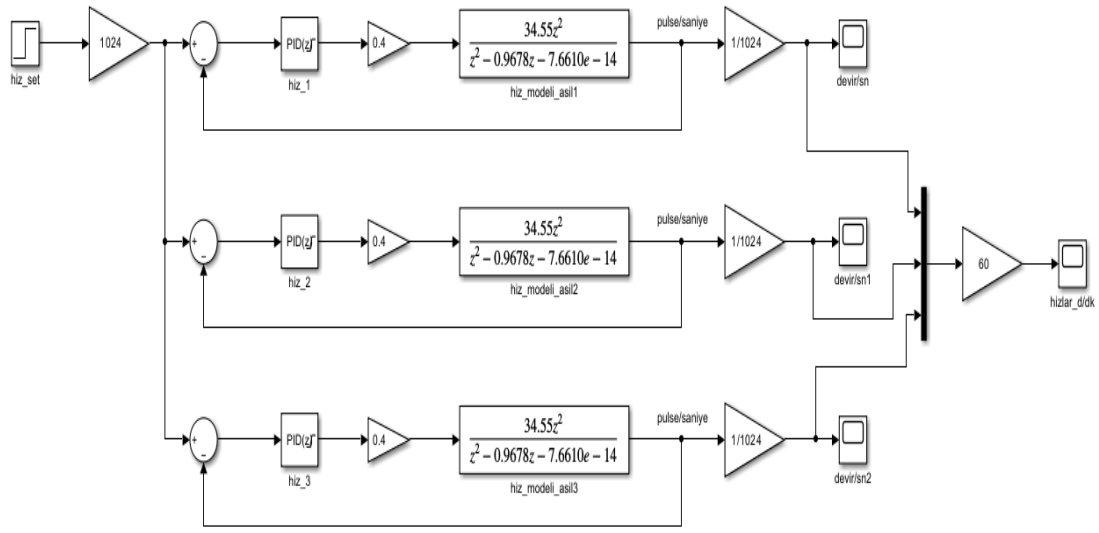


Şekil 26 : OPC Toolbox OPC Write Fonksiyonu Veri Tanımlaması

Açısal konum ve açısal hız modelleri elde edilen motorun kontrolü PID algoritması kullanılarak gerçekleştirilmiştir. Şekil 27'de açısal konum kontrolü blok diyagramı ve Şekil 28'de açısal hız kontrolü blok diyagramı gösterilmiştir.



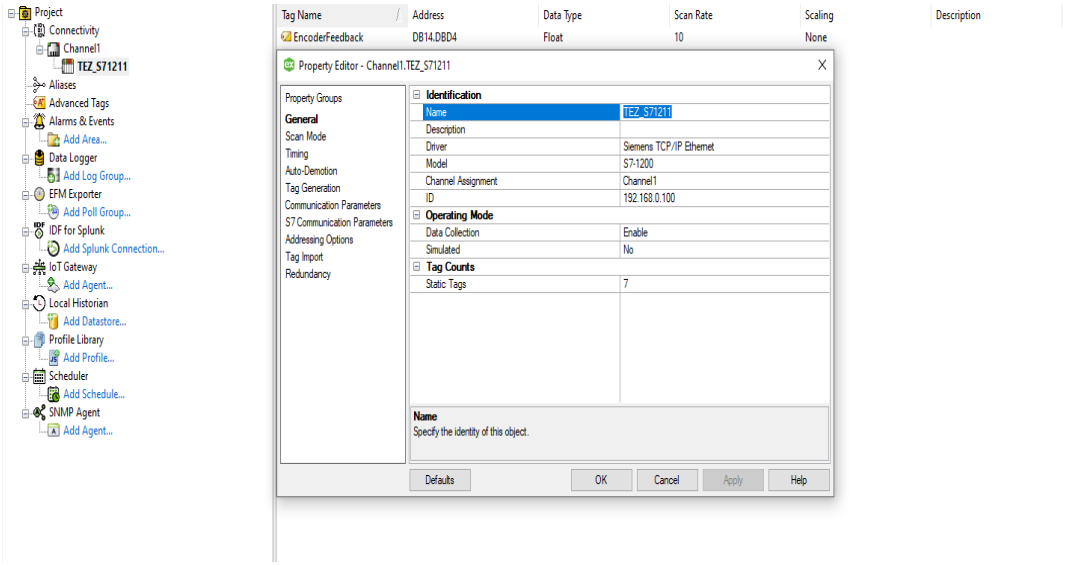
Şekil 27 : Açısal Konum Kontrolü Blok Diyagramı



Şekil 28 : Açısal Hız Kontrolü Blok Diyagramı

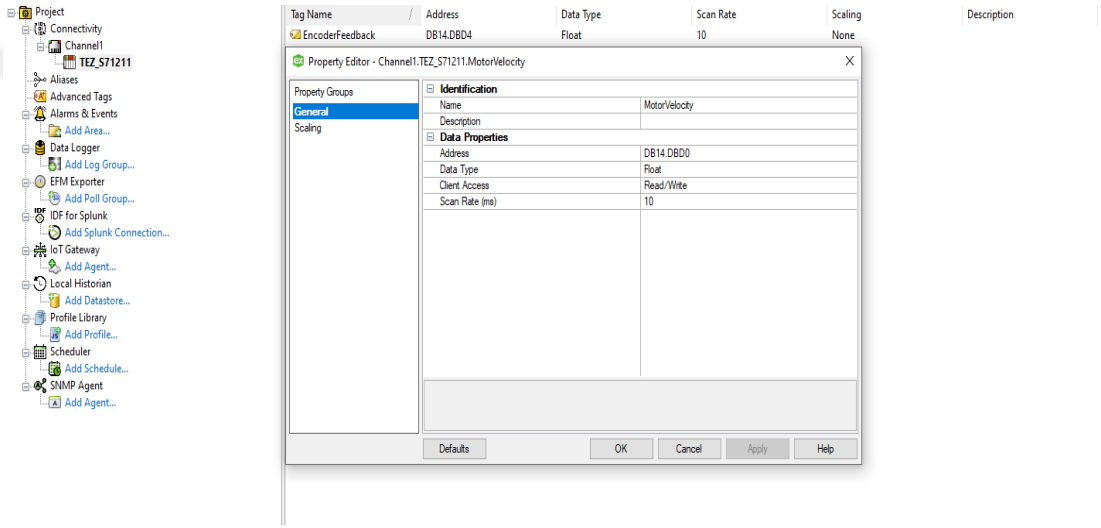
2.3. KEPServerEx ve UAExpert

KEPServer, endüstriyel otomasyon sistemlerindeki veri entegrasyonunu ve iletişimini kolaylaştıran güvenilir bir OPC sunucusudur. OPC, endüstriyel otomasyon uygulamalarında cihazlar arasındaki veri paylaşımını kolaylaştırmak için kullanılan bir iletişim protokolüdür. Bu tez çalışmasında, KEPServerEx ve UaExpert programları OPC sunucuları olarak kullanılmıştır.



Şekil 29 : KepServerEx OPC Sunucu Ayarları

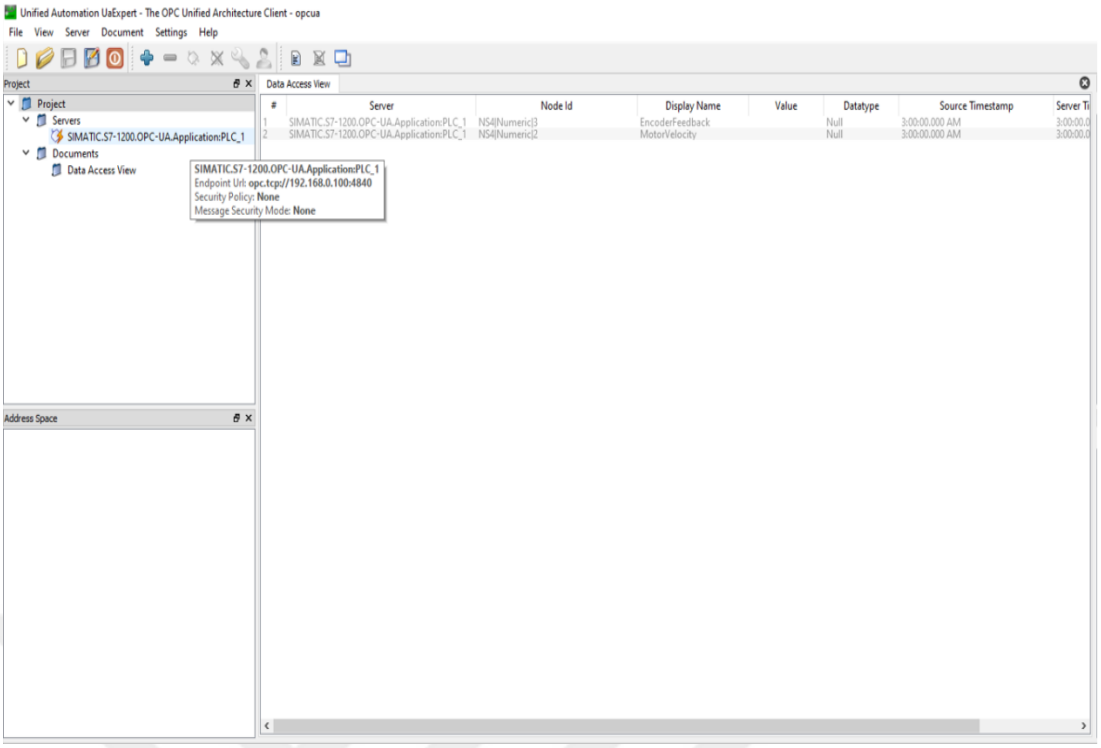
KEPServerEx, Matlab Simulink OPCTolbox üzerinde senkron haberleşme sağlanması için kullanılmıştır. OPC sunucusu etkinleştirilmiştir. KEPServerEx programı üzerinden "Siemens TCP/IP Ethernet" sürücüsü seçilerek bir kanal oluşturulmuştur. Bu kanal, endüstriyel cihazlarla veri alışverişi yapmak için bir arayüz sağlar. Aynı bilgisayar üzerinde çalıştırıldığı için, ethernet ayarları bilgisayarın ethernet adaptörüne göre yapılandırılmıştır. Tez kapsamında, PLC modeli ve IP adresi gibi gerekli ayarlar tanımlanmıştır. Bu ayarlar, PLC'nin doğru şekilde tanımlanması ve iletişim kurulması için önemlidir. Şekil 29 ’de OPC sunucu ayarı gösterilmiştir.



Şekil 30 : KepServerEx PLC Adres Ayarları

KEPServer yapılandırması tamamlandıktan sonra, PLC üzerinden okunacak olan datablok adresleri ve offset değerleri belirlenmiştir. Bu işlemler sonucunda, veri okuma işlemi başarıyla gerçekleştirilmiştir. Şekil 30 ’da PLC adres ayarı gösterilmiştir.

UaExpert, Unified Architecture (UA) standardını destekleyen bir OPC UA (Open Platform Communications Unified Architecture) istemcisidir. UaExpert, bu protokolü destekleyerek farklı cihazlarla iletişim kurmayı ve veri alışverişi yapmayı sağlar.



Şekil 31 : UaExpert İstemci Oluşturma

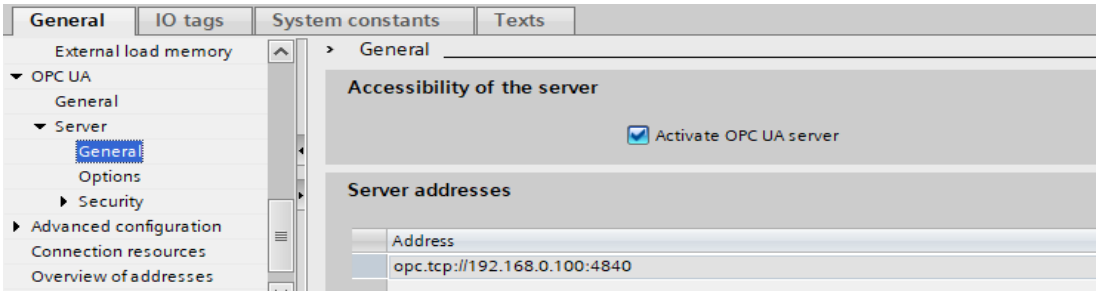
PLC sunucu ve Matlab OPC Tolbox istemci olarak oluşturulan asenkron bağlantı arasındaki haberleşmeyi test etmek için OPC UaExpert Client programı kullanımışıtr. Şekil 31 ’de UaExpert istemci oluşturma gösterilmiştir.

2.4. PLC Kontrol

Bu tez çalışmasında kullanılan Siemens CPU 1211C DC/DC/DC model PLC kullanılmıştır. Siemens CPU 1211C DC/DC/DC, Siemens'in S7-1200 serisi PLC (Programmable Logic Controller) ailesinin bir üyesidir. Bu PLC, endüstriyel otomasyon uygulamalarında sıklıkla kullanılan bir kontrol cihazıdır. CPU 1211C DC/DC/DC modeli, DA giriş voltajına sahip olan bir cihazdır. CPU üzerinde 6 dijital giriş (24V DA), 4 dijital röle çıkış (2A), 2 analog çıkış (0-10V) bulunur. Dijital girişler hızlı sayıcı (high speed counter) olarak ayarlanabilir. Dijital çıkışlar darbe üretici olarak kullanılabilir.

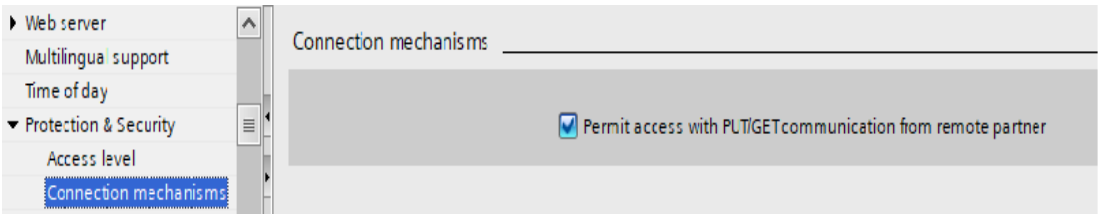
Bu tez çalışmasında kullandığımız PLC ile Matlab arasında asenkron haberleşme sağlayabilmek ve veri kaydı gerçekleştirmek için firmware V4.4 olacak şekilde versiyon güncellemesi yapılmıştır. Bu sayede PLC OPC UA sunucu

kullanılabilir hale gelmiştir. Şekil 32 ‘de PLC OPC UA sunucu etkinleştirme gösterilmiştir.



Şekil 32 : PLC OPC UA Sunucu Etkinleştirilmesi

PLC OPC UA sunucusu etkinleştirilerek ve port numaraları ayarlanarak PLC server aktif hale getirilmiştir. Ayrıca, OPC üzerinden PLC'nin Datablok adreslerinden veri okunabilmesi için gerekli ayarlamalar yapılmıştır. PLC ayarları üzerinden, "Permit Access with PUT/GET communication from remote partner" seçeneği aktif edilerek, uzaktan ortaklarla PUT/GET iletişimi ile erişim izni sağlanmıştır. Bu sayede, uzaktaki sistemlerin PLC'ye veri yazabilmesi ve veri okuyabilmesi mümkün hale gelmiştir. Şekil 33 ‘de PLC OPC UA sunucu etkinleştirme gösterilmiştir.

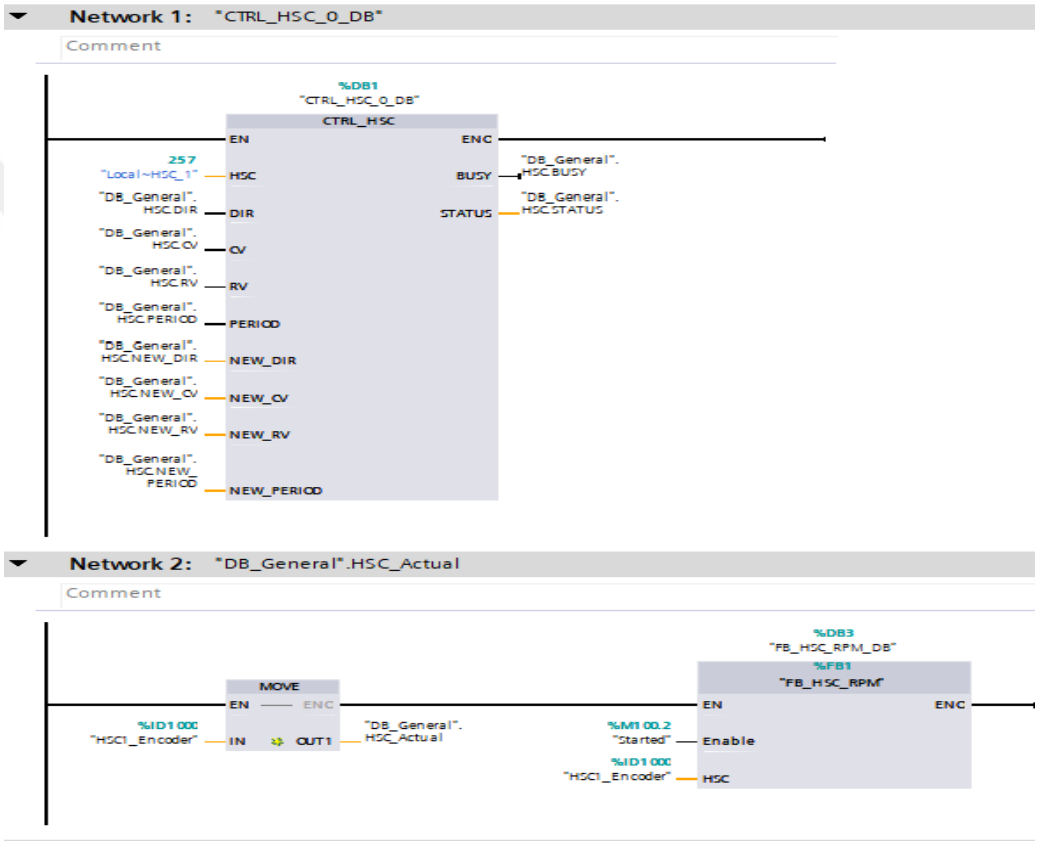


Şekil 33 : PLC PUT/GET Etkinleştirilmesi

Datablok ayarları içerisindeki "Optimized blok Access" seçeneği ile DB içerisindeki adresler erişilebilir hale getirilmiştir. Bu ayar sayesinde, OPC UA istemcisi ve MATLAB OPC Toolbox, DB içerisindeki adreslere doğrudan erişebilir hale gelip, veri okuma ve yazma işlemi gerçekleştirilmiştir.

PLC, Hızlı sayıcılar (high speed counter) genellikle döner makinelerin hızını veya dönüş sayısını izlemek için kullanılan bir tür sayaçtır. Bu sayıcılar genellikle hızlı tepki sürelerine sahip oldukları için hassas sayım işlemleri için uygundur. Bu

tezde PLC üzerinde bulunan dijital girişler I0.0 ve I0.1 100kHz bandında okuma yapabilen hızlı sayıcı olarak ayarlanmıştır. DA Motor miline bağlı enkoder tipi NPN 1024 PPR sayısına eşittir. PLC dijital giriş kanalları PNP yapıya sahiptir. Enkoder verilerinin okunabilmesini sağlamak için PLC dijital giriş için ortak uç olan common ucuna 24V besleme gerilimi uygulanarak o ortak uca bağlı tüm girişler NPN kanalına çevrilir. Enkoder üzerinde bulunan A ve B kanallarının NPN olarak ayarlanan kanallara bağlantısı yapılır.



Şekil 34 : PLC Enkoder Değeri Okuma

PLC, OB "CyclicInterrupt" içerisinde “CTRL_HSC” fonksiyonu kullanılarak enkoder başlangıç değer ayarı, okunan verinin resetlenmesi gibi ayarları yapmak için kullanılmıştır. %ID1000 adresi üzerinden enkoder değeri okunmaktadır. Şekil 34 ‘da PLC enkoder değeri okuma işlemi gösterilmiştir.

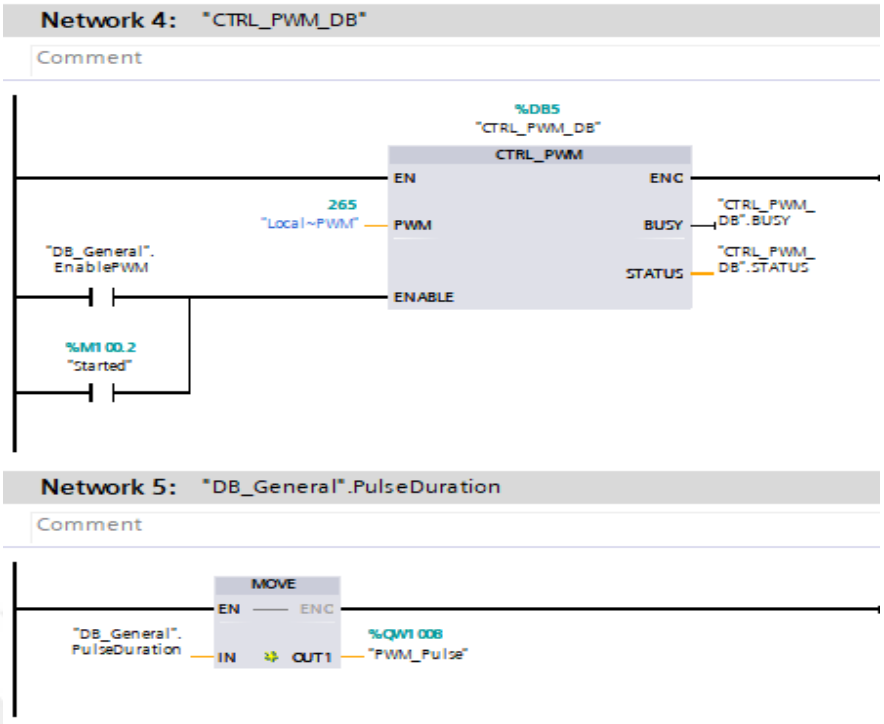
```
(*
    Speed  (RPS) =  [(C1-C2) / 0.2s ] / 1024
         or (RPM) =  [(C1-C2) / 0.2s ] * 60/ 1024  (200ms interrupt OB)
*)
REGION RPM Encoder Pulse/Rev 1024
    IF #Enable THEN
        #Current_Counter := #HSC;
        #Delta_Counter := #Current_Counter - #Record_Counter;
        #Ratio_Result_Real := DINT_TO_REAL(((#Delta_Counter*1000/#CycleTime)*60));
        #RPM := (#Ratio_Result_Real*0.1) / 1024.0;
        #Record_Counter := #Current_Counter;
        ;
    ELSE
        #Current_Counter := 0;
        #Delta_Counter := 0;
        #Ratio_Result_Real := 0;
        #RPM := 0;
        #Record_Counter := 0;
    END_IF;
END_REGION
```

Şekil 35 : PLC Enkoder RPM Hesabı

DA Motor enkoder değeri d/dk cinsinden hesaplanması için okunan değerin bir önceki değer ile farkı, okunan zaman değerleri ile işleme sokularak d/dk değerine dönüştürülmüştür. Şekil 35 ‘de PLC enkoder rpm(d/dk) hesabı gösterilmiştir.

DGM, bir sinyalin genlik modülasyonu yoluyla kontrol edilmesini sağlayan bir tekniktir. DGM bir sinyalin belirli bir süre boyunca yüksek veya düşük bir durumda kalmasını sağlayarak ortalama bir çıkış değeri elde etmeye yöneliktir. Bu tez çalışmasında, DA Motor hız kontrolü DGM sinyali ile yapılmıştır.

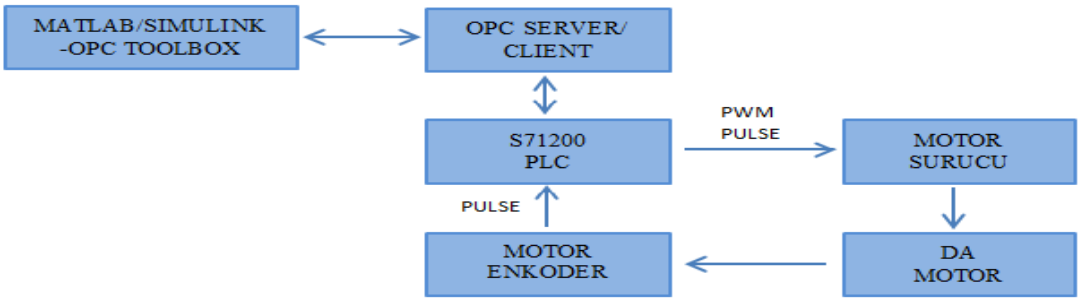
PLC üzerinde bulunan dijital çıkış Q0.0 100kHz bandında çıkış verebilen DGM sinyali olarak ayarlanmıştır. PLC donanım ayarları içerisinde puls üretici aktif edilmesi, sinyalin analog olarak seçimi, çevrim süresi ayarları gerçekleştirilmiştir.



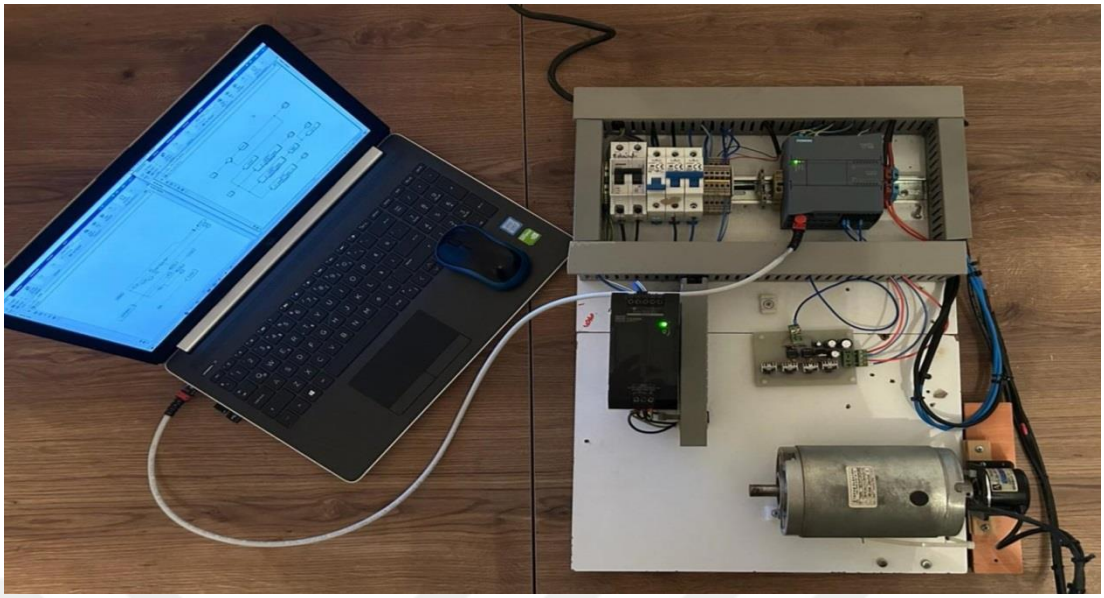
Şekil 36 : PLC DGM Kontrol

PLC içerisinde DGM çıkışını kontrol etmek için "CTRL_PWM" fonksiyonu kullanılmıştır. Bu fonksiyon DGM kanalını etkinleştirmek veya devre dışı bırakmak için kullanılmıştır. Şekil 36 'de PLC DGM kontrol gösterilmiştir.

Şekil 37 'da Sistem blok diyagramı gösterilmektedir. Sisteme 220V beslemeli sigorta üzerinden enerji verilir. Güç kaynağı beslemesi yine güç kaynağı girişinde bulanan sigorta üzerinden gerçekleştirilir. Güç kaynağı çıkışı yine sigorta üzerinden geçerek klemenslere dağıtılır. Klemensler üzerinden PLC beslemesi, sürücü beslemesi ve enkoder beslemesi gerçekleştirilir. Şekil 38 'da sistemin genel gösterimi gösterilmiştir.



Şekil 37 : Sistem Blok Diyagramı



Şekil 38 : Sistem Genel Gösterimi

2.5. DA Motor ve Sürücü

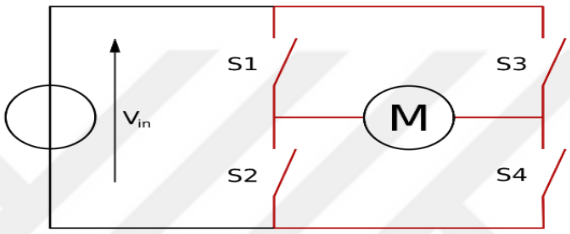
DA motorlar, doğru akım kullanarak çalışan elektrik motorlarıdır. DA motorlarındaki enerji, doğru akımın manyetik alan içinde dönmesiyle motor milinden elde edilen mekanik enerjidir. Doğru akım motorlarının kontrol edilebilmesi kolay olması sebep ile birçok alanda kullanılmaktadır. Bu tez çalışmasında 24V DA, 200W güce, 1280 d/dk etiket bilgilerine sahip DA motoru kullanılmıştır. Şekil 39 ‘de DA motor ve enkoder gösterilmiştir.



Şekil 39 : DA Motor ve Enkoder

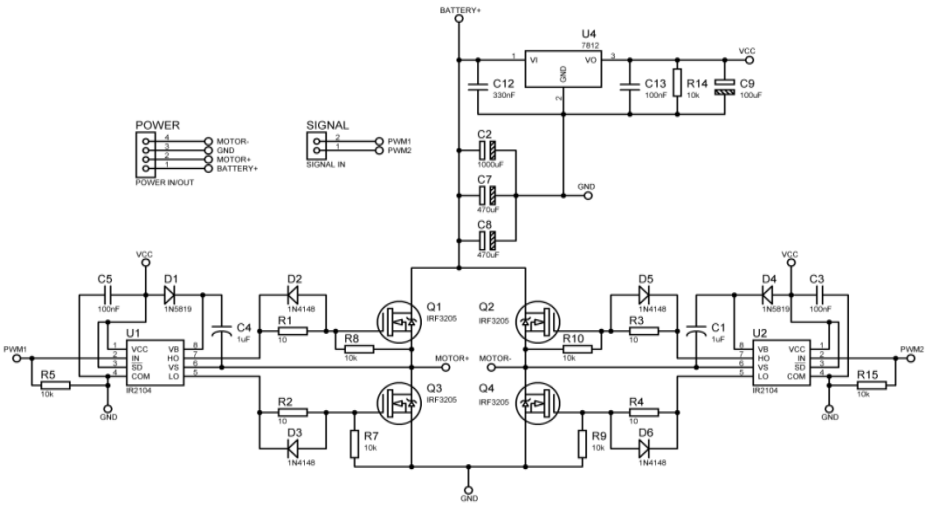
Enkoder motor şaft milinde artımsal NPN tipinde 1024 PPR çözünürlük değerine sahip A, B, Z kanallarına sahiptir. Enkoder A ve B kanalları enkoderin her bir dönüşünde darbe üretmekte, Z kanalı ise enkoder 1 turunda darbe üretmektedir. Bu tez çalışmasında A ve B kanallarının PLC üzerinde ayarlanan HSC kanalları kullanılarak bağlantıları yapılmıştır.

Bu tez çalışmasında, DA motor sürücüsü H köprülü mosfet sürücü devre topolojisi kullanılmıştır. H köprülü mosfet sürücüsü, dört ana mosfet kullanılarak oluşturulur ve bu mosfetler, motorun her iki terminaline seri olarak bağlanır.



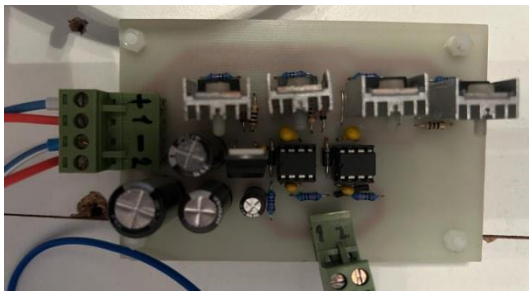
Şekil 40 : H Köprü

H köprü devresinde S1 ve S4 anahtarları aynı anda açılması ile motor saat yönünde hareket etmektedir. S3 ve S2 anahtarlarının açılması ile motor saat yönünün tersi yönde hareket etmektedir. Şekil 40 ‘de H köprü gösterilmiştir.



Şekil 41 : DA Motor Sürücü Devresi

Motor sürücü de anahtarlama elemanı olarak IRF3205 mosfetler kullanılmıştır. Mosfetlerin tetiklenebilmesi için IR2104 mosfet sürücü kullanılmıştır. Böylece güç ve kontrol kısmı ayrılmıştır. Şekil 41 ‘de ve Şekil 42 ‘de DA motor sürücü devresi ve DA motor sürücü gösterilmektedir.



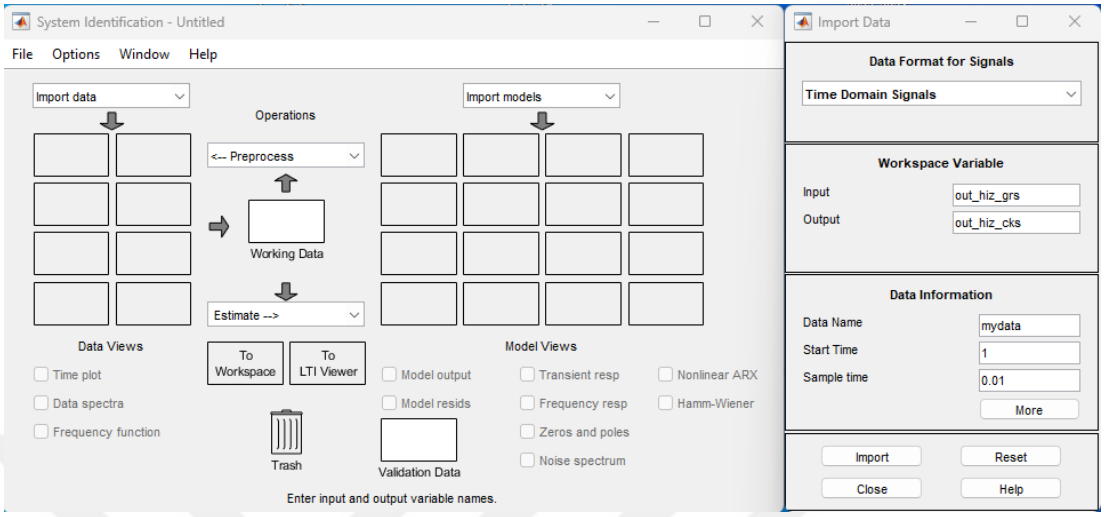
Şekil 42 : DA Motor Sürücü

Motor besleme girişleri, dört pin soket üzerinde bulunan + ve – işaretlere dikkat edilerek 12V- 48V arasında yapılmıştır. Dört pin soket üzerinde bulunan 1 ve 2 numaralı pinler DA motor bağlantı çıkışlarıdır. İki pin soket üzerinde bulunan 1 ve 2 numaralı pinler DGM girişleridir. İki pin soket üzerindeki 1 numaralı pin sol mosfet sürücüyü kontrol eder ve DGM uygulandığında motor saat yönünde döner. İki pin soket üzerindeki 2 numaralı pin ile de saat yönünün tersinde dönüş elde edilir. DGM girişlerinde güvenlik açısından max %100 görev döngüsü olmamalıdır. PLC dijital çıkışlarından ayarlanan DGM (0 → 27648) çıkışının bu pine bağlantısı yapılmıştır.

2.6. Sistem Tanımlama Araç Kutusu

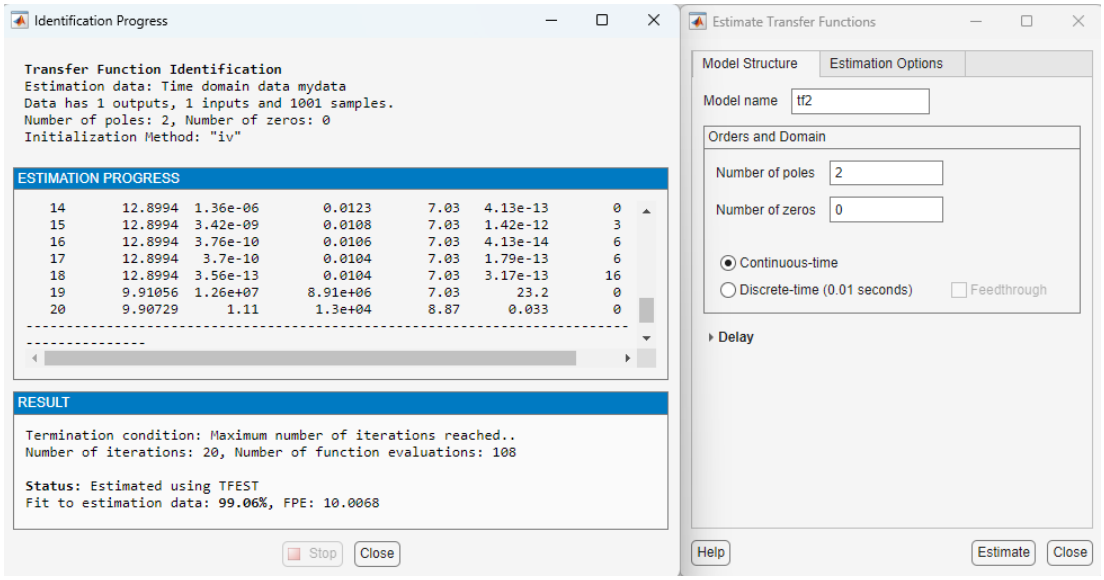
Matlab yazılımında dinamik sistemlerin deneysel yolla modellenmesi için “System Identification Toolbox” STAK isimli bir sistem tanımlama araç kutusu bulunmaktadır. Bu tezde kullanılan doğru akım motorunun kontrolünün yapılabilmesi için öncelikle matematiksel modeli elde edilmiştir. Dinamik model dikkate alınarak kontrolörler tasarlanarak optimal kontrol gerçekleştirilmiştir. Sistem tanımlama yapmanın ilk adımında doğru akım motoruna 3V genliğe sahip bir basamak giriş uygulanmış ve motorun milinde oluşan açısal konum ve açısal hız değerleri kaydedilmiştir. Elde edilen giriş-çıkış veri seti STAK üzerinde zaman

domeyni girişi kapsamında seçilmiştir. Verilerin örnekleme frekansı 0.01 Hertz'dir. Giriş-çıkış verilerinin STAK ara yüzüne girilmesi Şekil 43 'de gösterildiği gibidir.



Şekil 43 : Sistem Tanımlama Araç Kutusu Model

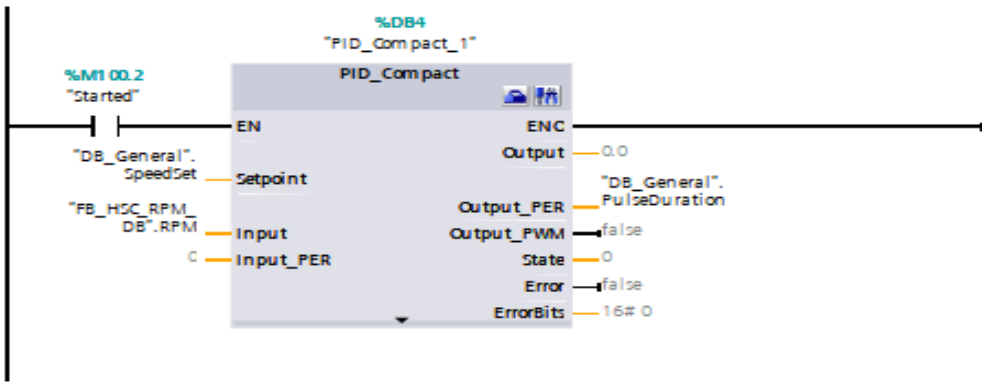
STAK ara yüzünde "estimate" başlığı altında transfer fonksiyonu, durum uzay formu, ARX gibi modelleme yöntemlerini içeren seçenekler bulunmaktadır. Bu tez çalışmasında yapılan modellemeler için transfer fonksiyonu yapısı seçilmiştir. Şekil 44 'de STAK ara yüzünde transfer fonksiyonu tanımlanmasına yönelik 2 adet kutup ve 1 adet sıfır içeren bir örnek gösterilmiştir.



Şekil 44 : Sistem Tanımlama Araç Kutusu Sonuç

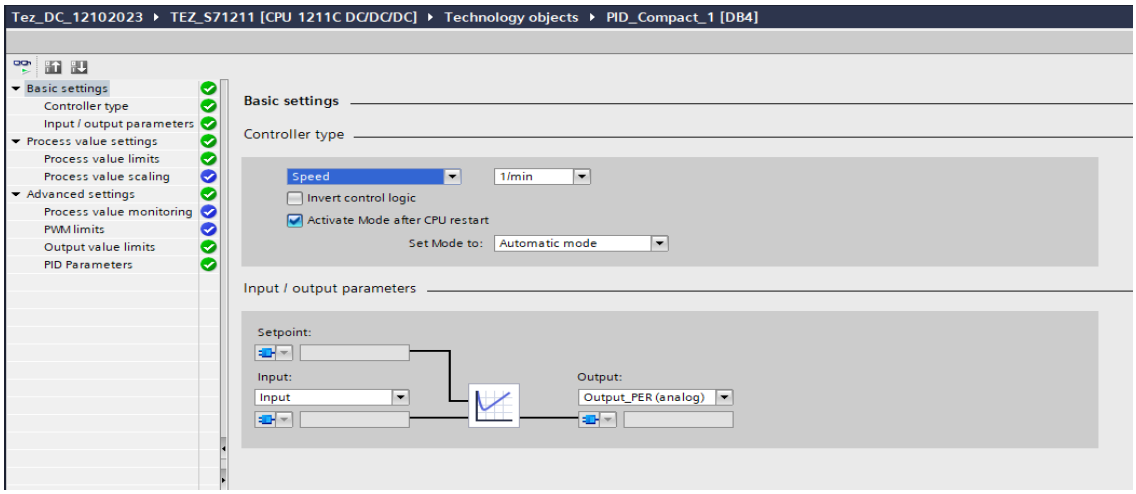
2.7. PID Kontrol

Bu tez çalışmasında, DA motor kontrolü için PLC içerisinde bulunan PID kontrolörü kullanılmıştır. "PID_Compact" fonksiyonu ölçülen proses değerini okur ve bunu ayar noktasıyla karşılaştırır. Karşılaştırma sonrası ortaya çıkan kontrol sapmasından kontrolör, ayar noktası sapmasını veya gerekirse bozulma değişkenini ayarlamak için bir çıkış değeri hesaplar. PID denetleyicinin çıkış değeri üç bileşenden oluşur. P bileşeni, çıkış değerinin P bileşeni kontrol sapması ile orantılıdır. I bileşeni, çıkış değerinin I bileşeni integral bileşeni kontrol sapması olduğu sürece artar. D bileşeni diferansiyel bileşendir ve kontrol sapmasının değişim hızı arttıkça artar. Sistem tanımlama ve metasezgisel algoritmalar ile elde edilen PID değerleri girilmiştir.



Şekil 45 : PLC PID Fonksiyon

PLC içerisinde organizasyon blok (OB) “PID_Compact” kontrol fonksiyonun çağrılması, yazılım kontrollerinin tanımlanmış bir zaman aralığında çağrılması gerektiğinden döngüsel kesme organizasyon blok OB ‘sinde gerçekleştirildi. PID yürütülmesi zaman alır ve OB1 yavaşlar bu sebep ile PLC ‘nin yanıtı azalır. OB "CyclicInterrupt" örnekleme zamanının sabit zaman aralığı olarak 100ms seçilmiştir. Şekil 45 ‘de PLC PID fonksiyonu gösterilmiştir.

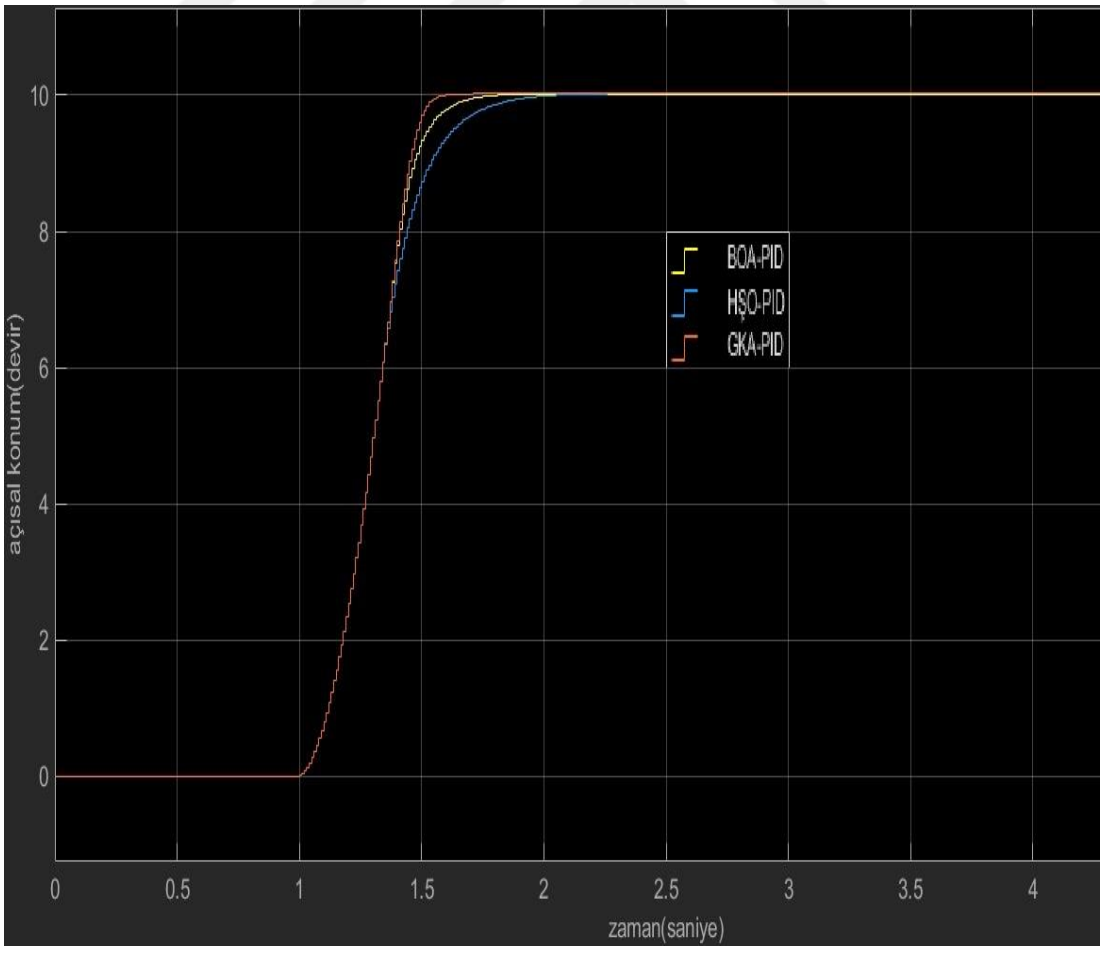


Şekil 46 : PLC PID Ayarları

"Setpoint" değeri Matlab üzerinden gelen hız ayar değeri, "Input" girdisi ise motor miline bağlı olan enkoder den okunan ayar bilgisi olarak girişleri yapılır. Çıkış ayarları PWM olduğu için PID blok çıkışı "Output_PWM" (0 → 27648) olarak çıkışa aktarılmıştır. Şekil 46 ‘de PLC PID ayarları gösterilmiştir.

3. ARAŞTIRMA BULGULARI TARTIŞMA

Bu tez çalışmasında bir DA motorunun açısal konum ve açısal hız kontrolü gerçekleştirilmiştir. MS algoritmaların amaç fonksiyonu olarak IAE kriteri tercih edilmiştir. Kontrolörlerin başarısı grafikler ve tablolar yardımıyla incelenmiştir. Açısal konum için referans değer 10 devir ve açısal hız için referans değer 600 devir/dk olarak belirlenmiştir. Geçici ve sürekli hal performans kriterleri olarak yükselme süresi (Tr), tepe değere ulaşma süresi (Tp), sürekli hale ulaşma süresi (Ts), maksimum aşma oranı (Mp) ve sürekli hal hatası (Ess) kriterleri dikkate alınmıştır. Şekil 47 ’de açısal konum kontrolünde PID eğrileri, Tablo 4 ’de Açısal konum kontrolünde kontrolörlerin performansı ve Tablo 5 ’de Açısal konum kontrolörlerin PID katsayısı verilmiştir. Şekil 48 ’de hız kontrolünde PID eğrileri, Tablo 6 ’da açısal hız kontrolünde kontrolörlerin performansı ve Tablo 7 ‘de açısal hız kontrolörlerin PID katsayısı verilmiştir.



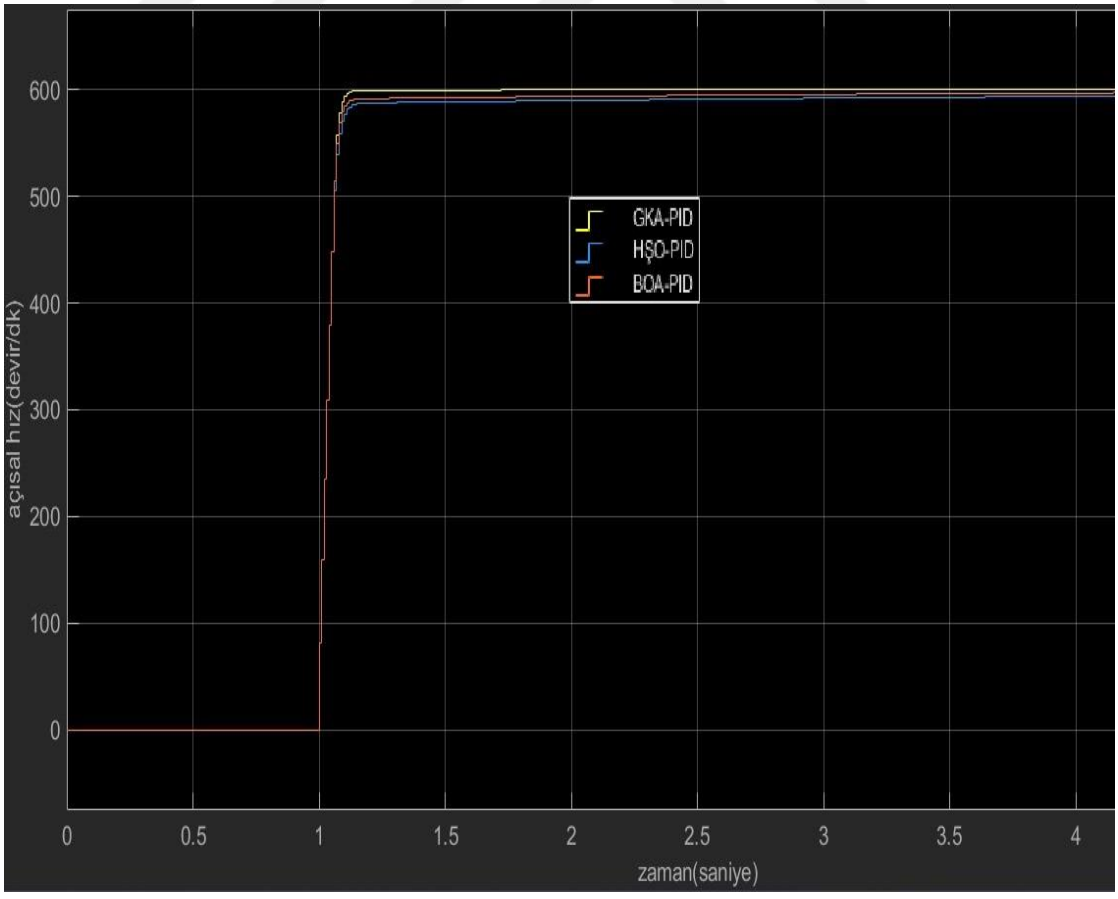
Şekil 47 : Konum Kontrolünde PID Eğrileri

Tablo 4 : Açısal Konum Kontrolünde Kontrolörlerin Performansı

Kriter	GKA-PID	BOA-PID	HŞO-PID
Tr	0.601	0.811	1.051
Tp	1.101	2.391	3.0
Ts	2.040	3.001	4.0
Mp	0.02	0.02	0.03
Ess	0.00	0.01	0.01

Tablo 5 : Açısal Konum Kontrolörlerin PID katsayıları

Katsayı	GKA	BOA	HŞO
P	0.6249020184633708	0.7949020184633708	0.5849020184633708
I	0. 0.0051662160299855	0.0054662160299855	0.0051662160299855
D	0.045522117233181307	0.071522117233181307	0.083522117233181307



Şekil 48 : Hız Kontrolünde PID Eğrileri

Tablo 6 : Açısal Hız Kontrolünde Kontrolörlerin Performansı

Kriter	GKA-PID	BOA-PID	HŞO-PID
Tr	0.174	0.188	0.201
Tp	1.690	10.381	16.630
Ts	2.530	11.860	18.861
Mp	0.01	0.01	0.01
Ess	0.001	0.002	0.002

Tablo 7 : Açısal Hız Kontrolörlerin PID katsayıları

Katsayı	GKA	BOA	HŞO
P	0.0998020184633708	0.0835020184633708	0.0751020184633708
I	0.0616662160299855	0.0295662160299855	0.0184662160299855
D	0.000310117233181307	0.000304117233181307	0.000412117233181307

MATLAB OPC Toolbox ’ın endüstriyel cihazlarla iletişim kurabilmesi, MATLAB ‘ın güçlü hesaplama ve kontrol algoritmalarını gerçekleştirmesi durumu göz önünde bulundurularak endüstrinin her alanında kullanılmaktadır. MATLAB OPC Araç Kutusu kullanılarak bir doğru akım (DA) motorunun kontrolü için OPC protokolü aracılığıyla veri alışverişi sağlanmıştır. Bu, endüstriyel otomasyon uygulamalarında uygulanabilir bir yöntemdir. OPC Araç Kutusu, MATLAB ortamında OPC sunucularına bağlanmak ve veri alışverişi yapmak için işlevler sağlar. Bu sebep ile OPC sunucusu ile iletişim kurmayı, veri okumayı ve yazmayı kolaylaştırır.

4. SONUÇLAR VE ÖNERİLER

Bu tez çalışmasında bir DA motorunun kontrolü Matlab ve OPC araç kutusu kullanılarak gerçekleştirilmiştir. Kontrolör PID yapısında olup PLC üzerinde çalıştırılmıştır. Tablo 4 ve Şekil 6 dikkate alındığında DA motorunun açısal konum kontrolünde GKA-PID yönteminin geçici ve sürekli hal yanıtı kriterlerinde en başarılı yöntem olduğu görülmektedir. Denenen 3 yöntemin sürekli hal performansları birbirine yakındır. Bununla birlikte, geçici hal performansında GKA-PID yöntemi BOA-PID ve HŞO-PID yöntemlerinden çok daha başarılı bir performans sergilemiştir. Tablo 4 ve Şekil 6 dikkate alındığında DA motorunun açısal hız kontrolünde GKA-PID yönteminin geçici ve sürekli hal yanıtı kriterlerinde en başarılı yöntem olduğu görülmektedir. Denenen 3 yöntemin geçici hal performansları birbirine yakın olmakla birlikte GKA-PID yöntemi BOA-PID ve HŞO-PID yöntemlerinden çok daha başarılı bir performans sergilemiştir.

Açısal konuma ait sonuçlar incelendiğinde GKA-PID ile elde edilen değerlerin; 0.601 Tr(s), 1.101 Tp(s), 2.040 Ts(s), 0.02 Mp(%), 0.00 Ess(%). BOA-PID ile elde edilen değerlerin; 0.811 Tr(s), 2.391 Tp(s), 3.001 Ts(s), 0.02 Mp(%), 0.01 Ess(%). HŞO-PID ile elde edilen değerlerin; 1.051 Tr(s), 3.000 Tp(s), 4.000 Ts(s), 0.03 Mp(%), 0.01 Ess(%) olduğu görülmüştür. Tr(s) değerleri incelendiğinde GKA-PID ’nin BOA-PID ’ye göre %25, BOA-PID ’nin HŞO-PID ‘ye göre %22 iyileşme sağlanmıştır. Tp(s) değerleri incelendiğinde GKA-PID ’nin BOA-PID ’ye göre %53, BOA-PID ’nin HŞO-PID ‘ye göre %20 iyileşme sağlanmıştır. Ts(s) değerleri incelendiğinde GKA-PID ’nin BOA-PID ’ye göre %32, BOA-PID ’nin HŞO-PID ‘ye göre %24 iyileşme sağlanmıştır. Hesaplanan Tr(s), Tp(s), Ts(s) değerleri incelendiğinde GKA-PID yönteminin BOA-PID ve HŞO-PID göre daha iyi performans sergilediği görülmektedir. BOA-PID de HŞO-PID göre daha iyi performans sergilemiştir

Açısal hıza ait sonuçlar incelendiğinde GKA-PID ile elde edilen değerlerin; 0.174 Tr(s), 1.690 Tp(s), 2.530 Ts(s), 0.01 Mp(%), 0.001 Ess(%). BOA-PID ile elde edilen değerlerin; 0.188 Tr(s), 10.381 Tp(s), 11.860 Ts(s), 0.01 Mp(%), 0.002 Ess(%). HŞO-PID ile elde değerlerin; 0.201 Tr(s), 16.630 Tp(s), 18.861 Ts(s), 0.01 Mp(%), 0.002 Ess(%) olduğu görülmüştür. Tr(s) değerleri incelendiğinde GKA-PID ’nin BOA-PID ’ye göre %7, BOA-PID ’nin HŞO-PID ‘ye göre %6 iyileşme

sağlanmıştır. Tp(s) değerleri incelendiğinde GKA-PID ’nin BOA-PID ’ye göre %83, BOA-PID ’nin HŞO-PID ‘ye göre %37 iyileşme sağlanmıştır. Ts(s) değerleri incelendiğinde GKA-PID ’nin BOA-PID ’ye göre %78, BOA-PID ’nin HŞO-PID ‘ye göre %37 iyileşme sağlanmıştır. Hesaplanan Tr(s), Tp(s), Ts(s) değerleri incelendiğinde GKA-PID yönteminin BOA-PID ve HŞO-PID göre daha iyi performans sergilediği görülmektedir. BOA-PID de HŞO-PID göre daha iyi performans sergilemiştir.

Sonuç olarak, bu tez çalışmasında kullanılan yöntem ve ekipmanlar ile bir DA motorun kontrolünün başarıyla yapılabildiği gösterilmektedir. Bununla birlikte, kullanılan PLC endüstriyel bir ekipmandır ve gömülü sistem ekipmanlarına göre oldukça pahalıdır. Yapılan uygulamaların daha uygun maliyetle gerçekleştirilebilmesi için 32 bitlik bir kontrol kartı kullanılabilir. Ayrıca kontrolör olarak PID ve optimize edici olarak metasezgisel algoritmalar kullanılmıştır. Sistem modelleme ve kontrolör tasarımında yapay zekâ destekli güncel yöntemler kullanılarak kontrolör başarısı arttırılabilir. Ayrıca geliştirilen kontrolörler belli referans değerlerinde başarı sağlayacak şekilde tasarlanmıştır. Daha geniş bir çalışma aralığında başarının garanti edilmesi için adaptif kontrol yöntemlerine başvurulması gerektiği açıktır.

Bu tez çalışmasında DA Motorun kontrolü stabil bir şekilde yapılabilmesi için PLC PID kontrolcüsü ile gerçekleştirilmiştir. Matlab OPC Toolbox ile OPC üzerinden DA motor kontrolü sırasında OPC ile gerçek zamanlı olarak veri alışverişinde gecikmeler söz konusu olmuştur. PLC, DGM tekniği ile motor kontrolünde çıkış frekansının daha yüksek olması kontrolün hassas olmasına sebep olacaktır. Enkoder den okunan sinyallerin PLC giriş filtrelerinin daha düşük olması ve enkoder çözünürlük değerinin daha yüksek olması yine kontrolün daha hassas olmasına sebep olacaktır.

KAYNAKLAR

[1] Çaşka, S., Dokuz, M. E., Aydın, M., İlman, M., et al. (2023). System Identification And Vibration Analysis Of Rotating Beam With Lattice Structures. International Journal of 3D Printing Technologies and Digital Industry, 7(2), 322-327.

[2] Çaşka, S., & Uysal, A. (2021). İHA Yardımcı İniş Sisteminin Meta-Sezgisel Optimizasyon Yöntemleri ile Kontrolü. Afyon Kocatepe Üniversitesi Fen ve Mühendislik Bilimleri Dergisi, 21(5), 1223-1230..

[3] MathWorks. (n.d.). System Identification ToolboxTM. Retrieved from https://www.mathworks.com/products/sysid.html

[4] Bainbridge, L. (1982). Ironies of Automation. Automatica, 19, 775-779.

[5] Yücel, İ. H. (1991). Sanayide Robot Teknolojisi: Uygulaması Ve Önemi. Ankara: DPT Yayınları.

[6] Sheridan, T. B. (1992). Telerobotics, Automation, and Human Supervisory Control. MIT Pres.

[7] Özcan, M. (2004). Otomasyon Sistemlerinde PLC Uygulamaları. Atlas Yayın Dağıtım.

[8] Acharya, V., Sharma, S. K., & Kumar Gupta, S. (2017). Analyzing the factors in industrial automation using analytic hierarchy process. Computers & Electrical Engineering. https://doi.org/10.1016/j.compeleceng.2017.08.015

[9] Kurtulan, S. (2001). PLC ile Endüstriyel Otomasyon. Birsen Yayınevi.

[10] Rehg, J. A., & Sartori, G. J. (2009). Programmable Logic Controllers. Upper Saddle River, N.J.: Pearson Prentice Hall.

[11] Mirzaoğlu, İ., & Sarıtaş, M. (2008). PLC VE SCADA Kullanarak Bir İrmik Üretim Sisteminin Otomasyonu.

[12] Bingöl, O., Aydoğan, T., Didin, H. R., Yalçiner, A. S., & Duygulu, K. (2010). PLC kontrollü otomatik katlı otopark sistemi. Süleyman Demirel Üniversitesi Uluslararası Teknolojik Bilimler Dergisi, 2(1), 65-76.

[13] Simatic, Ø. S. (2000). PLC S7-200 Öğretim Kitabı. İstanbul.

[14] Eminoğlu, Y. (2015). Plc Programlama ve S7 1200. İstanbul: Birsen Yayınevi.

[15] Acıelma, F. (2000). Programlanabilir Lojik Kontrol (PLC) Laboratuvarı: Devlet Kitapları

[16] Webb, J. W., & Reis, R. A. (2002). Programmable logic controllers: principles and applications: Prentice Hall PTR.

[17] Priya, A. Sahithi. (2021) Speed Control of DC Motor Using PWM. Kakatiya Institute of Technology and Science, Warangal, Department of Electrical and Computer Engineering.

[18] Technical.cat. (n.d.). PNP and NPN. Retrieved from www.tecnical.cat

[19] Elektrik Rehberiniz. (n.d.). Encoder. Retrieved from https://www.elektrikrehberiniz.com/elektronik/encoder-nedir-15233/

[20] Bunday, B. (1984). Basic Optimization Methods. London: Edward Arnold Ltd.

[21] Murty, K. G. (2003). Models for Decision Making. In Optimization Models For Decision Making (Vol. 1, Chapter 1, pp. 1-18). Internet Edition.

[22] Simon, D. (2013). Evolutionary Optimization Algorithms. Wiley.

[23] Ruder, S. (2016). An overview of gradient descent optimization algorithms. arXiv preprint arXiv:1609.04747

[24] Meza, J. C. (2011). Newton's method. Wiley Interdisciplinary Reviews: Computational Statistics, 3(1), 75-78. https://doi.org/10.1002/wics.129

[25] Dantzig, G. B., & Thapa, M. N. (2003). Linear Programming: Theory and Extensions. Springer
[26] Deb, K. (2001). Multi-Objective Optimization Using Evolutionary Algorithms. Wiley.
[27] Haupt R. L. (1994.) Thinned arrays using genetic algorithms. IEEE Trans Antennas Propag 42, 993–999. https://doi.org/10.1109/8.299602-5
[28] Sharma, M., & Kaur, P. (2020). A comprehensive analysis of nature-inspired metaheuristic techniques for feature selection problem. Archives of Computational Methods in Engineering. https://doi.org/10.1007/s11831-020-09412-6
[29] Meng, Z., Li, G., Wang, X., Sait, S. M., & Yıldız, A. R. (2020). A comparative study of metaheuristic algorithms for reliability-based design optimization problems. Archives of Computational Methods in Engineering. https://doi.org/10.1007/s11831-020-09443-z
[30] Bozorg-Haddad, O., Solgi, M., & Loáiciga, H. A. (2017). Meta-heuristic and evolutionary algorithms for engineering optimization. John Wiley & Sons.
[31] Kuyu, Y. Ç. (2023). Optimizasyon Problemleri İçin Yeni Metasezgisel Yaklaşımlar [Doctoral dissertation, Bursa Uludağ Üniversitesi Fen Bilimleri Enstitüsü, Elektronik Mühendisliği Anabilim Dalı].
[32] Çelik, Y. (2013). Optimizasyon Problemlerinde Bal Arılarının Evlilik Optimizasyonu Algoritmasının Performansının Geliştirilmesi [Doctoral dissertation, Selçuk Üniversitesi Fen Bilimleri Enstitüsü].
[33] Laporte, G. (2006). Classical and modern heuristics for the vehicle routing problem. International Transactions in Operational Research, 13(3), 285-300.
[34] Hart, W. E. (1994). Adaptive Global Optimization With Local Search [Doctoral dissertation, University of California, San Diego].
[35] Altay, E. V., & Alatas, B. (2020). Bird swarm algorithms with chaotic mapping. Artificial Intelligence Review, 53(2), 1373-1414.
[36] Varol, E., & Alataş, B. (2017). Sürü zekâsında yeni bir yaklaşım: Kuş sürüsü algoritması. Dicle Üniversitesi Mühendislik Fakültesi Mühendislik Dergisi, 8(1), 133-146.
[37] Metasezgisel Tablo,
https://en.wikipedia.org/wiki/Table_of_metaheuristics#cite_note-11
[38] Acharya, B. B., Dhakal, S., Bhattarai, A., & Bhattarai, N. (2021). PID speed control of DC motor using meta-heuristic algorithms. International Journal of Power Electronics and Drive Systems, 12(2), 822.
[39] Ekinci, S., Izci, D., & Hekimoğlu, B. (2020, June). PID speed control of DC motor using Harris hawks optimization algorithm. In 2020 International conference on electrical, communication, and computer engineering (ICECCE) (pp. 1-6). IEEE.
[40] Heidari, A. A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., & Chen, H. (2019, August 1). Harris hawks optimization: Algorithm and applications. Future Generation Computer Systems, 97, 849–872.
[41] Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014, March 1). Grey Wolf Optimizer. Advances in Engineering Software, 69, 46–61.
[42] Zervoudakis, Konstantinos; Tsafarakis, Stelios (2023). "A global optimizer inspired from the survival strategies of flying foxes". Engineering with Computers. 39 (2):
[43] Mirjalili, S., & Lewis, A. (2016, May 1). The Whale Optimization Algorithm. Advances in Engineering Software, 95, 51–67.
[44] Mirjalili, S. (2015). Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. Knowledge-Based Systems, 89, 228–249.

[45] Soe, Y. Y., & San, P. E. (2019). Pid closed-loop control analysis for automation with siemens plc using tia v13. 2(August), 200–208.
[46] Ang, K. H., Chong, G., & Li, Y. (2005). PID control system analysis, design, and technology. IEEE Transactions on Control Systems Technology, 13(4), 559–576.
[47] Namba, R., Yamamoto, T., & Kaneda, M. (1997, October). Robust PID controller and its application. In 1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation (Vol. 4, pp. 3636-3641). DOI: 10.1109/ICSMC.1997.633233.
[48] Bucz, Š., & Kozáková, A. (2018). Advanced methods of PID controller tuning for specified performance. In PID Control for Industrial Processes (pp. 73-119). DOI: 10.5772/intechopen.76069.
[49] Chaib, L., Choucha, A., & Arif, S. (2017). Optimal design and tuning of novel fractional order PID power system stabilizer using a new metaheuristic Bat algorithm. Ain Shams Engineering Journal, 8(2), 113-125. DOI: 10.1016/j.asej.2015.08.003.
[50] Abut, T. (2016). Modeling and optimal control of a DC motor. Int. J. Eng. Trends Technol, 32(3), 146-150.
[51] Mitin, F., & Krivushov, A. (2018, October). Application of optimal control algorithm for DC motor. In Proceedings of the 29th DAAAM International Symposium (pp. 0762-0766).
[52] Agarwal, J., Parmar, G., & Gupta, R. (2017). Application of sine cosine algorithm in optimal control of DC motor and robustness analysis. Wulfenia J, 24(11), 77-95.
[53] Ruderman, M., Krettek, J., Hoffmann, F., & Bertram, T. (2008). Optimal state space control of DC motor. IFAC Proceedings Volumes, 41(2), 5796-5801.
[54] DA Motor, https://www.iqsdirectory.com/articles/electric-motor/dc-motors.html
[55] Uysal, A., Gokay, S., Soylu, E., Soylu, T., & Çaşka, S. (2019). Fuzzy proportional-integral speed control of switched reluctance motor with MATLAB/Simulink and programmable logic controller communication. Measurement and Control, 52(7-8), 1137-1144.
[56] Suman, S. K. (2016). PI, PID Controllers with Performance Index for Speed Control of DC Motor. Compusoft, 5(12), 2241.
[57] Mohammed, J. A. (2013). Pulse width modulation for DC motor control based on LM324. Eng. and Tech. J, 31(10), 1882-1896.
[58] Thomas, N., & Poongodi, D. P. (2009, July). Position control of DC motor using genetic algorithm based PID controller. In Proceedings of the world congress on engineering (Vol. 2, pp. 1-3).
[59] Faroqi, A., Ramdhani, M. A., Frasetyio, F., & Fadhil, A. (2018, November). Dc motor speed controller design using pulse width modulation. In IOP Conference Series: Materials Science and Engineering (Vol. 434, No. 1, p. 012205). IOP Publishing.
[60] Amiri, P., & Bagheri, M. (2013). Speed control of dc motor by programmable logic control with high accuracy. Universal Journal of Control and Automation, 1(4), 91-97.
[61] Adoghe, A. U., Aliu, S. O., Popoola, S. I., & Atayero, A. A. (2017). Digital speed control of DC motor for industrial automation using pulse width modulation technique. In Proceedings of the World Congress on Engineering (Vol. 1).
[62] Shary, D. K., Nekad, H. J., & Alawan, M. A. (2023). Speed control of brushless DC motors using (conventional, heuristic, and intelligent) methods-based PID

controllers. Indonesian Journal of Electrical Engineering and Computer Science, 30(3), 1359-1368.
[63] Faruq, M. F. B. M. (2008). PID controller design for controlling DC motor speed using Matlab application. Universiti Malaysia Pahang, Electrical & Electronics Engineering.
[64]. Saputra, D., Ma'arif, A., Maghfiroh, H., Chotikunnan, P., & Rahmadhia, S. N. (2023). Design and Application of PLC-based Speed Control for DC Motor Using PID with Identification System and MATLAB Tuner. International Journal of Robotics and Control Systems, 3(2), 233-244.
[65] OPC Foundation.(n.d.). What is OPC? Retrieved from https://opcfoundation.org/about/what-is-opc/
[66] Büyük, S., & Salih, G. Ö. K. (2012). Proses Otomasyonunda OPC Teknolojisi. Mühendis ve Makina, 13-15.
[67] Ünlü, Y. (2007). OPC Standardı ve Uygulaması [Yüksek Lisans Tezi, Istanbul].
[68] Mahnke, W., Leitner, S. H., & Damm, M. (2009). OPC Unified Architecture. Berlin.
[69] Ünlü, Y. (2007). Süreç kontrolunda nesnelerin bağlaşması ve ilişkilendirilmesi (OPC) standardı ve uygulaması [Master's thesis, Fen Bilimleri Enstitüsü].