



Proje Tabanlı Öğrenme Müfredatlı Endüstriyel Otomasyon Eğitim Setinin İmalatı ve Programlanması

Yazılım Mühendisliği Ana Bilim Dalı

Yüksek Lisans

İbrahim AYDIN

ORCID 0009-0001-1886-7417

Tez Danışmanı: Doç. Dr. Vahide BULUT

Haziran 2024

İzmir Kâtip Çelebi Üniversitesi Fen Bilimleri Enstitüsü öğrencisi **İbrahim AYDIN** tarafından hazırlanan **Proje Tabanlı Öğrenme Müfredatlı Endüstriyel Otomasyon Eğitim Setinin İmalatı ve Programlanması** başlıklı bu çalışma tarafımızca okunmuş olup, yapılan savunma sınavı sonucunda kapsam ve nitelik açısından başarılı bulunarak jürimiz tarafından YÜKSEK LİSANS olarak kabul edilmiştir.

ONAYLAYANLAR:

Tez Danışmanı: **Doç. Dr. Vahide Bulut**
İzmir Kâtip Çelebi Üniversitesi

Jüri Üyeleri: **Prof. Dr. Aytuğ ONAN**
İzmir Kâtip Çelebi Üniversitesi

Doç. Dr. Hatice YILDIZ DURAK
Necmettin Erbakan Üniversitesi

Savunma Tarihi: 07.06.2024

Yazarlık Beyanı

Ben, **İbrahim AYDIN** başlığı **Proje Tabanlı Öğrenme Müfredatlı Endüstriyel Otomasyon Eğitim Setinin İmalatı ve Programlanması** olan bu tezimin ve tezin içinde sunulan bilgilerin şahsıma ait olduğunu beyan ederim. Ayrıca:

- Bu çalışmanın bütünü veya esası bu üniversitede Yüksek Lisans derecesi elde etmek üzere çalıştığım süre içinde gerçekleştirilmiştir.
- Daha önce bu tezin herhangi bir kısmı başka bir derece veya yeterlik almak üzere bu üniversiteye veya başka bir kuruma sunulduysa bu açık biçimde ifade edilmiştir.
- Başkalarının yayımlanmış çalışmalarına başvurduğum durumlarda bu çalışmalara açık biçimde atıfta bulundum.
- Başkalarının çalışmalarından alıntıladığımda kaynağı her zaman belirttim. Tezin bu alıntılar dışında kalan kısmı tümüyle benim kendi çalışmamdır.
- Kayda değer yardım aldığım bütün kaynaklara teşekkür ettim.
- Tezde başkalarıyla birlikte gerçekleştirilen çalışmalar varsa onların katkısını ve kendi yaptıklarımı tam olarak açıkladım.
- Tezin donanım malzemeleri İzmir Bornova Mazhar Zorlu MTAL tarafından karşılanmıştır. Tez için üretilen deney seti, okulun eğitim öğretim materyali olarak kullanılacaktır.

Tarih: 07.06.2024

Proje Tabanlı Öğrenme Müfredatlı Endüstriyel Otomasyon Eğitim Setinin İmalatı ve Programlanması

ÖZ

Bu çalışmanın amacı, endüstriyel otomasyon teknolojileri eğitimlerinde uzaktan (online) ve yüz yüze olarak rahatlıkla kullanılabilen, geleneksel öğrenim metotlarının ötesine geçmiş, öğrencilere 21.yy. becerilerini de kazandırmaya odaklanmış proje tabanlı öğrenme (PTÖ) metodunu ve müfredatını kullanan, Milli Eğitim Bakanlığı (MEB) müfredatı ile uyumlu bir eğitim setinin imalatı ve programlanmasıdır. İmal edilen eğitim setine iki farklı marka programlanabilen mantık denetleyicisi (PLC) montajlanarak farklı PLC eğitimlerinin aynı set üzerinde verilebilmesi sağlanmıştır. Eğitim setinde, bir döner tabla ve iki adet konveyör bant kullanılarak elektropnömatik sistem yoluyla ürün ayrımı yapılabilmektedir. Eğitim setinde ayrıca motor kontrolleri, sensör uygulamaları, elektropnömatik, dokunmatik panel ve çeşitli PLC kontrolleri de yapılabilmektedir. Özellikle grafcet diyagramları metoduna odaklanılan set üzerinde, aynı senaryoların farklı PLC modellerinde nasıl programlanacağı gösterilebilmektedir. Eğitim setinde, PLC yazılım bileşenleri kullanmak yoluyla, öğrencilere görsel programlama ile kendi özgün yönetsel denetim ve veri toplama yazılımları (SCADA) geliştirmesine ve deneyimlemesine de imkân sağlanmaktadır. Bu çalışma aynı zamanda PTÖ için hazırlanmış bir örnek müfredatı da içermektedir. Bu sayede tüm eğitimler set üzerinde PTÖ metoduyla verilebilmekte ve öğrencilere 21.yy. becerilerini geliştirme fırsatı da sunulmaktadır. PTÖ, yenilikçi ve süreç odaklı bir öğrenme metodudur ve MEB tarafından da desteklenmektedir. Hazırlanan müfredat, öğrenme planı ile ölçme ve değerlendirme ölçeklerini (rubriklerini) de kapsamaktadır. Sonuç olarak tez aşamasında imal edilen bu setin, uzaktan ve yüz yüze verilecek endüstriyel otomasyon teknolojileri eğitimlerinde, öğrencilere mesleki becerilerin ve 21.yy. becerilerinin kazandırılmasında verimli ve faydalı olacağı düşünülmektedir.

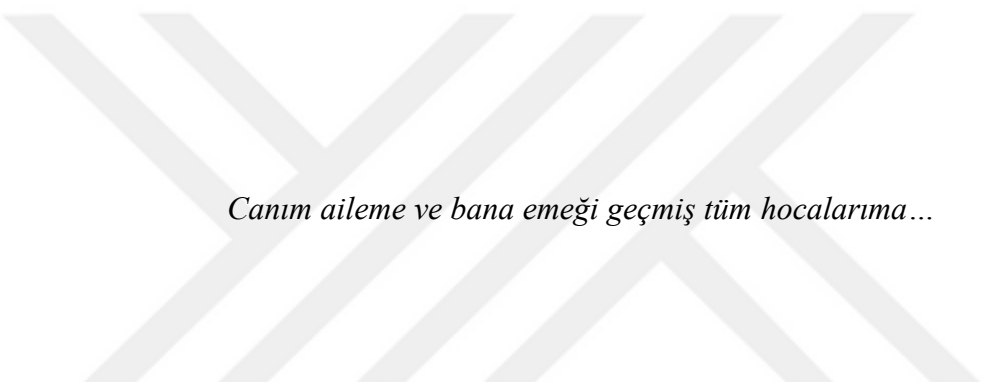
Anahtar Sözcükler: Endüstriyel otomasyon, proje tabanlı öğrenme, mesleki eğitim, PLC, hibrit eğitim

Production and Programming of Industrial Automation Training Set with Project Based Learning Curriculum

Abstract

The aim of this study is to provide students with 21st century education that can be easily used in industrial automation technologies training, both online and face-to-face, that goes beyond traditional learning methods. It is the production and programming of a training set compatible with the Ministry of National Education (MEB) curriculum, using the project-based learning (PBL) method and curriculum focused on gaining skills. Two different brands of programmable logic controllers (PLC) were mounted on the manufactured training set, allowing different PLC training to be given on the set. In the training set, product separation can be done via the electropneumatic system using a rotary table and two conveyor belts. Motor controls, sensor applications, electropneumatic, touch panel and various PLC controls can also be performed in the training set. Focusing especially on the Grafset diagrams method, the set can show how to program the same scenarios in different PLC models. By using PLC software components in the training set, students are also given the opportunity to develop and experience their own unique Supervisory Control and Data Acquisition software (SCADA) with visual programming. This study also includes a sample curriculum prepared for PBL. In this way, all training can be given on the set with the PBL method, and the students are given a 21st century experience. Opportunities to improve their skills are also provided. PBL is an innovative and process-oriented learning method and is supported by the Ministry of Education (MEB). The prepared curriculum also includes the learning plan and measurement and evaluation scales (rubrics). As a result, it is thought that this set, which was produced at the thesis stage, will be efficient and useful in providing students with professional skills and 21st century skills in industrial automation technologies trainings to be given online and face to face.

Keywords: Industrial automation, project-based learning, vocational training, PLC, hybrid education



Canım aileme ve bana emeđi gemiř tm hocalarıma...

Teşekkür

Bu tezin gerçekleştirilmesinde bilgi ve tecrübesiyle bana yol gösteren, çalışmamla yakından ilgilenen saygıdeğer akademik danışmanım Doç. Dr. Vahide Bulut'a teşekkür ederim.

Ayrıca tez süresince motivasyon kaynağım olan çok değerli eşime ve çocuklarıma, desteklerinden dolayı İzmir Bornova Mazhar Zorlu MTAL idarecilerine, özellikle tezin her safhasında tecrübe ve bilgilerinden çokça yararlandığım, desteklerini hiçbir zaman benden esirgemeyen meslek arkadaşlarım Murat Özdeveci' ye, Telat Güler'e ve Mustafa Güneş'e teşekkürlerimi sunarım.

İçindekiler

Yazarlık Beyanı	ii
Öz	iii
Abstract	iv
Teşekkür	vi
Şekiller Listesi	x
Tablolar Listesi	xi
Kısaltmalar Listesi	xii
Semboller Listesi	xiii
Bölüm 1	1
1. Giriş.....	1
1.1. Tezin Amacı ve Kapsamı.....	2
1.2. Sınırlılıklar.....	4
Bölüm 2	5
2. Kuramsal Temeller	5
2.1 Programlanabilir Lojik Kontrolör (PLC)	5
2.2. Grafcet Diyagramları.....	8
2.3. Konveyör Bant	9
2.4. Elektropnömatik Sistem.....	11
2.5. Adım (Step) Motoru.....	13
2.6. Üniversal Motor	16
2.7. Dokunmatik (Touch) Panel.....	18
2.8. Proje Tabanlı Öğrenme (PTÖ) Yöntemi.....	19

Bölüm 3	21
3. Materyal ve Yöntem.....	21
3.1. Endüstriyel Otomasyon Eğitim Seti Genel Bilgileri	21
3.2. Proje Tabanlı Eğitim Müfredatı Tasarımı	26
3.3. PLC Bağlantıları.....	43
3.4. Döner Tabla Süren Step Motoru Özellikleri	48
3.5. Döner Tabla Süren Step Motor Sürücüsü Özellikleri	52
3.6. Grafcet Diyagramı.....	60
3.7. Setin PLC Yazılımı (Programı)	64
3.8. Setin Görsel Yazılımı (SCADA).....	68
3.8.1. Form Tasarımı.....	68
3.8.2. SCADA Program Kodları	73
3.9. Dokunmatik (Touch) Panel Tasarımı.....	91
3.10. PTÖ Uygulaması.....	92
Bölüm 4	93
4. Bulgular ve Tartışma	93
4.1. Montaj Sürecindeki Bulgular.....	93
4.2. Programlama Sürecindeki Bulgular	95
4.3. PTÖ Sürecindeki Bulgular.....	96
Bölüm 5	98
5. Sonuçlar ve Öneriler	98
Kaynaklar	101

Ekler 106

Ek A: PLC-2 Ladder Programı 106

Ek B: PLC-2 SCADA Tasarımı.....125

Ek C: PLC-2 Program Kodları 126

Ek D: PLC-2 Dokunmatik (Touch) Panel Tasarımı 137

Ek E: PTÖ Uygulaması, Örnek Formlar ve Rubrikler 138

Özgeçmiş 169



Şekiller Listesi

Şekil 2.1	Farklı PLC Modelleri.....	5
Şekil 2.2	PLC Parçalarının Yapısı ve Fonksiyonları.....	6
Şekil 2.3	PLC Genişleme Birimleri.....	6
Şekil 2.4	PLC Programlama Teknikleri	7
Şekil 2.5	PLC Kullanım Alanlarına Örnek.....	7
Şekil 2.6	Örnek Bir Grafcet Diyagramı.....	9
Şekil 2.7	Çeşitli Konveyör Bant Sistemleri.....	10
Şekil 2.8	Bir Konveyör Bant Sistemi Üzerinde Kullanılan Çeşitli Algılayıcılar ...	10
Şekil 2.9	Basit Elektropnömatik Sistem	11
Şekil 2.10	Elektropnömatik Devre Örneği	11
Şekil 2.11	Elektropnömatik Devrelerde Kullanılan Bazı Semboller	12
Şekil 2.12	Çeşitli Elektropnömatik Malzemeler	12
Şekil 2.13	Elektropnömatik Devre Uygulaması Örneği.....	12
Şekil 2.14	Adım (Step) Motoru İç Yapısı.....	13
Şekil 2.15	Adım (Step) Motoru Çeşitleri	13
Şekil 2.16	Step Motor Tam Adım Çalışma (a), Yarım Adım Çalışma (b).....	14
Şekil 2.17	Unipolar Step Motor Sürme Tabloları	15
Şekil 2.18	Bipolar Step Motor Yapısı	15
Şekil 2.19	Unipolar Step Motor Yapısı	16
Şekil 2.20	Üniversal Motorun İç Yapısı.....	17
Şekil 2.21	Üniversal Motorun Ac ve Dc Gerilimde Hız Ve Tork İlişkisi.....	17
Şekil 2.22	Endüstride Kullanılan Dokunmatik Paneller	18
Şekil 3.1	Güç Katı	22
Şekil 3.2	PLC ve Kontrol Katı.....	22
Şekil 3.3	Aktüatör Katı.....	23
Şekil 3.4	Kablo Kanallarının Montajı (a), Güç Ve Kontrol Katının Montajı (b), Konveyörlerin, Motorların, Sensörlerin Ve Pnömatik Sistemin Montajı (c), Tüm Setin Montajlanmış Hali (d).	25
Şekil 3.5	Proje Tabanlı Öğrenme Yaklaşımında Öğrenenin Rolü	27
Şekil 3.6	PTÖ Planı (a), PTÖ Müfredatı (b), PTÖ Öğrenme Takvimi (c) Ve Meb Yönetmeliğine Uygun Değerlendirme Kriterleri (d).....	35

Şekil 3.7	Form-1 Öğrenci Sözleşmesi	36
Şekil 3.8	Form-2 Öğretmen Gözlem Formu	37
Şekil 3.9	Form-3 Öğrenci Öz Değerlendirme Formu	39
Şekil 3.10	Form-4 İşbirlikçi Öğrenme Formu	40
Şekil 3.11	Form-5 Proje Değerlendirme Formu	41
Şekil 3.12	Form-6 Sunum Değerlendirme Formu	42
Şekil 3.13	Form-7 Rapor Değerlendirme Formu	43
Şekil 3.14	Step Motor ile PLC Arasındaki Bağlantı Konfigürasyonu	48
Şekil 3.15	Step Motor İç Yapısı	48
Şekil 3.16	Step Motorun Açık Ve Açık Hız Karakteristik Sinyalleri	49
Şekil 3.17	Kayış Kasnak Sistemi Tork Hesabı	49
Şekil 3.18	Step Motor İçin Gerekli Pals Sinyali Hesabı	50
Şekil 3.19	Step Motor İçin Örnek Tork Hesabı	50
Şekil 3.20	Ac Ve Dc Step Motor Sürücüsü Tork-Hız Eğrileri	51
Şekil 3.21	Step Motorun Tork-Frekans Sürücü Performansı	51
Şekil 3.22	Step Motor İle Servo Motorun Hareket Sonrası Durma Davranışları	51
Şekil 3.23	Step Motor, Kayış Kasnak Sistemi ve Döner Tabla	52
Şekil 3.24	Autonics Aic-D Serisi Step Motor Sürücüsü Ve Özellikleri	52
Şekil 3.25	Autonics Aic-D Serisi Step Motor Sürücüsü Yazılımı (Atmotion)	53
Şekil 3.26	Autonics Aic-D Step Motor Sürücüsü Konfigürasyon Diyagramı	53
Şekil 3.27	Autonics Aic-D Step Motor Sürücüsü Kontrol I/O Uçları	54
Şekil 3.28	Autonics Aic-D Step Motor Sürücüsü Haberleşme Çıkışları	54
Şekil 3.29	Autonics Aic-D Serisi Step Motor Sürücüsü Konnektör Uçları	55
Şekil 3.30	Autonics Aic-D Serisi Step Motor Sürücüsü I/O Kablosu	56
Şekil 3.31	Autonics Aic-D Serisi Step Motor Sürücüsü Üniteleri	56
Şekil 3.32	Autonics Aic-D Serisi Step Motor Sürücüsü Ve Motor Bağlantıları (a), Bağlantılar (b)	58
Şekil 3.33	Autonics Aic-D Serisi Step Motor Sürücüsü Alarm Ve Uyarı Işıkları ...	58
Şekil 3.34	Autonics Aic-D Serisi Step Motor Sürücüsü Girişleri	59
Şekil 3.35	Autonics Aic-D Serisi Step Motor Program Modları	59
Şekil 3.36	Autonics Aic-D Step Motor Sürme Metotlarının Karşılaştırılması	59
Şekil 3.37	Konveyörler Üzerindeki Sensörler	61
Şekil 3.38	PLC-1 Giriş-Çıkış Adresleri	62

Şekil 3.39 PLC-1 Grafcet Diyagramı (Algoritması)	61
Şekil 3.40 PLC-1 İçin Ladder Programlama	64
Şekil 3.41 PLC-1 Programı: Adımlar (a), Eylemler (b), Zamanlayıcılar (c).....	67
Şekil 3.42 Giriş Formu Tasarımı	68
Şekil 3.43 Scada Form Tasarımı	69
Şekil 3.44 Buton Takımı	70
Şekil 3.45 Giriş-Çıkış ve Grafcet Formları	70
Şekil 3.46 Kamera Form Tasarımı.....	71
Şekil 3.47 Scada Formu Anlık Ekranı	72
Şekil 3.48 SQL Veri Tabanı Tabloları	72
Şekil 3.49 Dokunmatik Panel Tasarım Ekranı	91
Şekil 3.50 Dokunmatik Panel Test Sayfası	91

Tablolar Listesi

Tablo 3.1	PLC -1 Giriş Modüllerinin Bağlantıları.....	44
Tablo 3.2	PLC -1 Çıkış Modüllerinin Bağlantıları	44
Tablo 3.3	PLC -2 Giriş Modüllerinin Bağlantıları.....	45
Tablo 3.4	PLC -2 Çıkış Modüllerinin Bağlantıları	45
Tablo 3.5	PLC -1 Sembol Tablosu.....	46
Tablo 3.6	PLC -2 Sembol Tablosu.....	46



Kısaltmalar Listesi

PLC	Programmable Logic Controller (Programlanabilen Mantık Bilgisayarları)
SCADA	Supervisory Control And Data Acquisition (Yönetmel Denetim ve Veri Toplama)
PTÖ	Proje Tabanlı Öğrenme
I/O	Input/Output (Giriş / Çıkış)
MEB	Millî Eğitim Bakanlığı
IoT	Nesnelerin İnterneti

Semboller Listesi

ϕD	Dişli Çapı
F	Kuvvet [N]
m	Kütle [kg]
T	Tork
f	Frekans
t	Zaman



Bölüm 1

1. Giriş

İlk zamanlar çok üretim yapmak için çok işçi gerekli olması anlayışı yaygındı. Fakat ne kadar çok işçi kullanılırsa kullanılsın, istenen kalite ve hıza erişilemedi. Bu durum, insanlığı üretimde endüstriyel otomasyon sistemlerine ve teknolojilerine geçişe zorladı. İlk zamanlar elektronik kartlarla yapılan otomasyon kontrolü, sonraları yerini PLC teknolojilerine bıraktı. Bu sayede üretimde istenilen hız ve kaliteye ulaşıldı. Artık günümüzde SCADA yazılımları ile üretimin her anını kontrol, izleme ve analiz etme imkânı bulunmaktadır. Otomasyon sistemleri de son teknolojiyi kullanabilecek seviyeye ulaştı. Dolayısıyla içinde yaşadığımız çağı, “Endüstri 4.0”, “e-factory” gibi terimlerle isimlendiriyoruz.

Teknolojinin ilerlemesi ile eğitimde farklı yaklaşımlar ortaya çıkmakta böylece öğrenme-öğretme süreçlerini etkileyen, anında bilgiyi işleme, görüntüleme ve paylaşma özelliklerine sahip teknolojiler ortaya çıkmaktadır [1]. Bu çerçevede yaşadığımız çağda çevrimiçi öğrenme, işlevsel olarak daha uygulanabilir özellikler kazanmış [2], bilginin yayılmasında zaman ve mekânın sınırlarını zorlamış [3] [4] ve bu durumun herkes için eğitimde birçok fırsatlar oluşturduğu belirtilmiştir [5] [6].

Proje kavramı, genelde öğrencilerin bireysel ya da gruplar halinde verilen ödev veya görevleri yerine getirmesi; bir problemi çözmek için çalışmalar yürütmesidir [7]. Proje, hayal etme, düşünme ve kurgulama demektir [8]. Türkçe dil sözlüğünde ise; tasarı, tasarlanmış şey olarak ifade edilmektedir. (TDK, 2005).

Proje tabanlı öğrenme/öğretim ingilizce yayınlarda; “Project-Based Learning”, “Project-Based Model”, “Project-Based Teaching/Instruction”, ve “Project-Based Approach” isimleriyle geçmektedir. Türkçe yayınlarda ise; “Proje Tabanlı Öğrenme”, “Proje Temelli Öğrenme Yöntemi”, “Projeye Dayalı Öğretim”, “Proje Tabanlı Öğrenme Yaklaşımı” şeklinde ifade edilmektedir. Literatürde daha fazla tanıımı bulmak mümkündür.

Bu tanımlamalara genel olarak bakıldığında: tasarı geliştirmeye, hayal etmeye, planlamaya ve kurgulamaya dayalı bir öğrenme anlayışı [9] ve gerçekle ilişkili konu ve uygulamalara yönelik öğrenci merkezi, disiplinler arası ve etkinliklerin daha çok sürece yayıldığı veya günlük yaşamda karşılaşılan sorunları çözmeye yönelik bir öğrenme yaklaşımı [10] olarak tanımlanmaktadır. Moursund'a göre ise PTÖ, belirli bir zaman zarfında gerçekleştirilen bir gösterim, performans veya genelde bir ürünün ortaya çıkmasını sağlayan bireysel ya da grup etkileşimiyle oluşturulan etkinliktir [11]. Ayrıca PTÖ, bir ürün veya performansı planlayıp, oluşturup daha sonra sergilemek için bir derste öğrencilere bilgi ve beceri kazandıran öğretim yöntemi olarak da tanımlanmaktadır [12].

PTÖ' de sadece etkinliklerin değil, ortaya çıkan ürün ve bu ürünün ortaya çıkma sürecinin beraber değerlendirilmesi büyük önem taşımaktadır, yani süreç ve sonuç değerlendirme söz konusudur [13] [14]. Williams (1998)'in de ifadesinde; değerlendirme ölçekleri (rubrikler), öz değerlendirme, akran değerlendirme, kontrol listeleri, gözlem, günlükler, performans ve sunumlar, görüşme, kaydedilmiş öğrenci çalışmaları, yazılı görevler olabilir [15].

Bu çalışma, PTÖ temelli bir müfredat içeren iki farklı PLC barındıran bir endüstriyel otomasyon eğitim setinin imalatı ve programlanmasını hedeflemiştir. Hem çağımızın endüstri temelli isimlendirilmesini hem de eğitimde öğrenci merkezli bir model olan PTÖ modelinin yenilikçi taraflarını göz önüne aldığımızda, bu çalışmanın mesleki eğitim adına faydalı bir çalışma olacağı düşünülmektedir. Otomasyon sistemlerini ve teknolojilerini, farklı PLC modellerinin tek set üzerinde kullanma ve öğrenme imkânını sağlayacak olan bu eğitim setinin, lise ve üniversite seviyesinde atölye ve laboratuvarlarda kullanılabilecek yapıda ve yeterlilikte olması beklenmektedir.

1.1. Tezin Amacı ve Kapsamı

Endüstriyel otomasyon teknolojileri eğitimleri, Endüstri 4.0 çağını yakalama ve milli sanayi hamlesi vizyonları açısından büyük önem taşımaktadır. Meslek liselerinde ve üniversitelerde bu eğitimler için birçok deney setleri kullanılmaktadır. Bu setlerin büyük bir kısmı Avrupa ve Uzakdoğu kaynaklı ve maliyetli ürünlerdir. Bu çalışmada özgün, kullanışlı ve ekonomik bir endüstriyel otomasyon eğitim setine odaklanıldı.

Endüstriyel otomasyon alanı elektronik, makine, yazılım ve elektrik alanlarını içine alan multidisipliner bir alandır. Bu çalışmada endüstriyel otomasyonun temel bileşenleri olan PLC, konveyör bant, döner tabla, pnömotik sistem, dokunmatik panel, endüktif, kapasitif ve kızılötesi sensörler, dokunmatik panel, endüstriyel ethernet ve görsel programlama temelli SCADA yazılımlarını içeren bir deney seti üzerinde çalışılmıştır. Ayrıca eğitim müfredatı, planı ve değerlendirmesi için de proje tabanlı öğrenme (PTÖ) modeli esas alınmıştır.

Çalışmada imalatı ve programlanması yapılan bu deney seti ile benzer setlerin aksine bir değil iki farklı PLC kullanılarak aynı uygulamaların farklı PLC işlemcileriyle aynı set üzerinde yapılabilmesine olanak sağlanmıştır. Set bu yönüyle yenilikçidir. İki adet konveyör bant ve bir adet döner tabla kullanılan set ile değişik senaryolarda, grafcet diyagramları kullanılarak, kısa ve uzun olan plastik ve metal ürünlerin ayrımı pnömotik sistem yardımıyla yapılabilmektedir.

Farklı grafcet algoritmalarının set üzerinde çalıştırılabilmesi sayesinde, iki farklı PLC işlemcilerinde programlama geçişleri kolaylıkla yapılabilmektedir. Bu sayede farklı PLC yazılımlarının eğitimleri tek bir set üzerinde kolayca verilebilmektedir. Farklı PLC modelleriyle çalışma becerisini kazanmak önemlidir. Çünkü günümüzde tek bir model/marka PLC kullanılmamaktadır. Birçok PLC modeliyle çalışma zorunluluğu doğmaktadır. Maliyet planlamaları bunu gerektirmektedir.

Set üzerinde nesne tabanlı görsel programlama (C#, C++, Visual Basic.NET gibi) ile PLC yazılım kütüphanelerini kullanarak yerli ve özgün SCADA yazılımları yapılabilmektedir. Setin üzerinde istenilen senaryoya göre SCADA yazılımları geliştirilebilmektedir. Set üzerinde dokunmatik panel üzerinde görsel tasarımlar yapılabilmekte, endüstriyel ethernet teknolojisi kullanılarak endüstriyel IoT (Nesnelerin İnterneti) uygulamaları gerçekleştirilebilmektedir.

Set üzerinde verilecek olan eğitimler, öğretmen merkezli geleneksel eğitim metodunun ötesinde, öğrenci merkezli PTÖ metodu ile örnek bir müfredat kapsamında verilebilecektir. PTÖ metoduna göre hazırlanmış ölçme değerlendirme ölçekleri (rubrikler) de eğitim sonrası kullanılabilecektir. Müfredat örnek bir senaryo üzerinden geliştirilmiştir. İstenildiğinde örnek müfredat referans alınarak farklı senaryolara göre de PTÖ temelli müfredat ve plan hazırlanabilmektedir.

1.2. Sınırlılıklar

Endüstriyel otomasyon teknolojilerinin geniş bir teknoloji yelpazesine sahip olduğu düşünüldüğünde bu çalışmada; PLC, konveyör bant, sensör ve motor uygulamaları, ürün ayırıştırma, Elektropnömatik, dokunmatik panel, endüstriyel ve görsel programlama (SCADA) konularına odaklanılmıştır. Kullanılan endüstriyel malzemeler yeterli ve güncel özelliklerdedir. Çok daha gelişmiş ürün yelpazesinin piyasada olduğu açıktır. Set üzerinde, kullanılan malzemelerin özellikleri çerçevesinde sınırlı uygulamalar yapılabilmektedir.

Set, sadece eğitim amaçlı, okul atölye ve laboratuvar ortamları için hazırlanmıştır. Bu çalışmada amaç, aynı senaryoyu grafcet metodunu kullanarak farklı iki PLC ortamında gerçekleştirebilmektir. Çünkü günümüz teknolojileri çeşitlendiği için en uygun maliyetli ürün seçimi yapılmakta, hep aynı marka ürünler pratikte kullanılmamaktadır. Aynı işlemi farklı marka PLC modelleri ile yapabilme becerisine sahip olma, endüstriyel otomasyon teknolojileri alanı için artık kaçınılmaz olmuştur.

Set sınırlı özelliklere sahip olmasına rağmen geliştirilebilir özelliktedir. Yapay zekâ, görüntü işleme ve internet tabanlı uygulamalarının da üzerinde gerçekleştirilebileceği bir yapıdadır. Görsel programlamada kullanılacak olan yazılım PLC kütüphanelerinin sağladığı imkânlar dâhilindedir. Set üzerinde sanayi robotu olamamasına rağmen endüstriyel robot uygulamalarıyla uyumlu imal edilmiştir. Döner tabladan ürünler robot kol ile alınıp başka iş istasyonlarına taşınabilir. Döner tabla ise en çok sekiz ürün depolama kapasitesine sahiptir.

PTÖ uygulamaları öğretmenlerin ve öğrencilerin özverili çalışması ile sınırlıdır. PTÖ planı 16 haftalık bir süreyi kapsamaktadır. 12 hafta kısa eğitimler ve 4 haftalık PTÖ uygulamasını içerir. Plan ve müfredat öğretmen tarafından geliştirilip yenilenebilir durumdadır. Öğrenci sayısının az tutulması ve PTÖ süresinin uzatılması eğitim verimi için tavsiye edilir.

Tezin diğer bölümleri şu şekilde organize edilmiştir; ikinci bölümde, kuramsal temeller üzerinde durulmuştur. Üçüncü bölümde materyal ve yöntem, dördüncü bölümde bulgular tartışılmış ve beşinci bölümde tezin sonuçları ve öneriler sunulmuştur.

Bölüm 2

2. Kuramsal Temeller

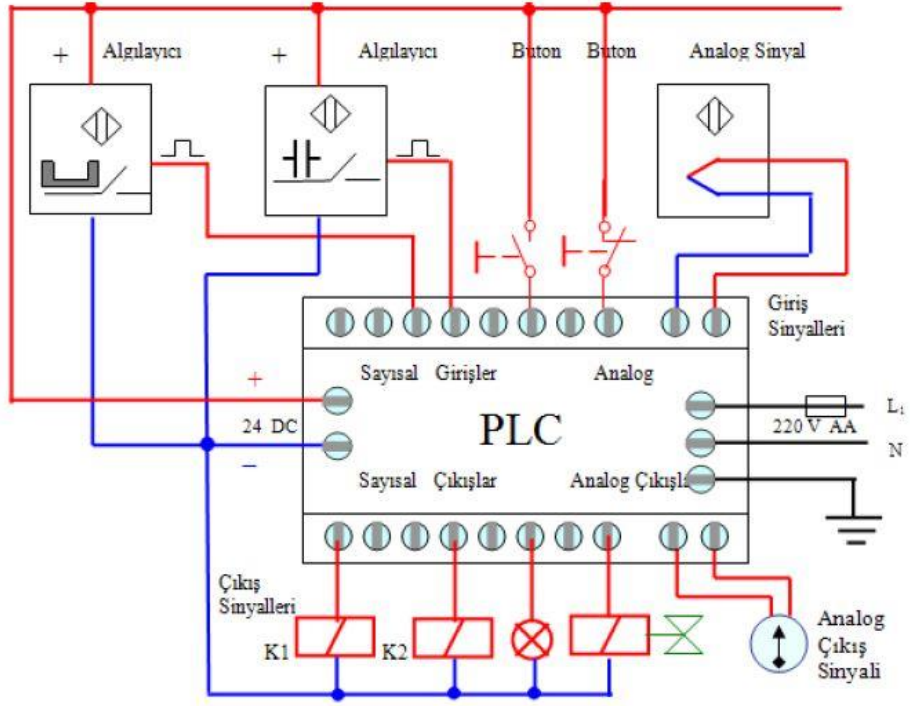
2.1 Programlanabilir Lojik Kontrolör (PLC)

Programlanabilir mantıksal denetleyici (Programmable Logic Controller, PLC) giriş-çıkış ve iletişim arabirimleri olan, bir sistem programı ile çalışan endüstriyel bir denetleyici denilebilir. PLC kullanımı, röleli kontrol kullanımının önüne geçmiş ve bu sayesinde hızlı bir gelişim göstermiştir. Günümüzde ise dev firmaların PLC modelleri yüksek performans göstermektedirler. Şekil 2.1’de çeşitli PLC modelleri görülmektedir.



Şekil 2.1: Farklı PLC Markaları ve Modelleri

Giriş ve çıkış ünitelerine sahip olan bu aygıtlara butonlar, anahtarlar, algılayıcılar (sensörler), kumanda elemanları, pnömotik valfler ve silindirler, kontaktörler, motor sürücüleri, endüstriyel kameralar, dokunmatik paneller ve sanayi robotları gibi birçok otomasyon malzemesi bağlanabilmektedir. Röle ve transistör çıkışlı olabilen bu aygıtlarda hızlı üretim için transistör çıkışlı olan modelleri tercih edilmektedir. Günümüzde üretim sahasında otomasyon yükünü kaldırabilecek yeterliliklere sahip PLC’ler bulunmaktadır. Şekil 2.2’de PLC parçalarının yapısı ve fonksiyonları görülmektedir:



Şekil 2.2: PLC Parçalarının Yapısı ve Fonksiyonları

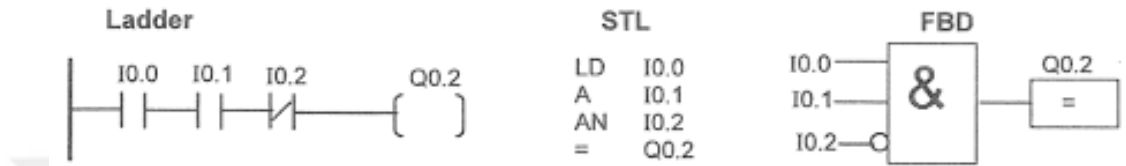
Modüler ve geliştirilebilir yapıdaki PLC'lere birçok ek modül (giriş-çıkış, analog giriş, haberleşme modülleri) eklenebilmekte ve mevcut yapı ihtiyaca göre geliştirilebilmektedir. Şekil 2.3'te PLC genişleme birimlerine örnek bir resim görülmektedir.



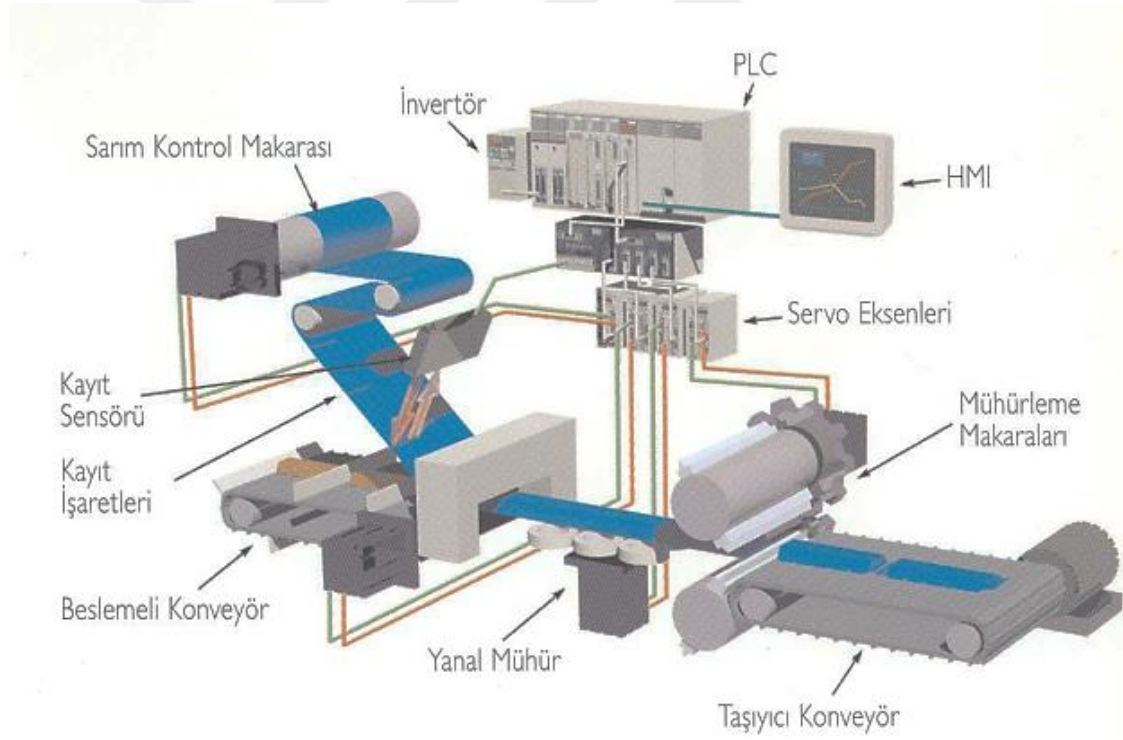
Şekil 2.3: PLC Genişleme Birimleri

Merdiven diyagramı programlama (Ladder) yöntemi PLC programlanmasında popüler bir tercihtir.

Ladder programlama haricinde diğer programlama teknikleri; ST (Structured Text), FBD (Function Block Diagram), SFC (Sequential Function) ve IL (Instruction List) olarak sıralanabilir. Şekil 2.4'te PLC' programlama tekniklerinden örnekler görülmektedir. Şekil 2.5'te [16] ise PLC kullanım alanlarına örnek bir uygulama görülmektedir.



Şekil 2.4: PLC Programlama Teknikleri



Şekil 2.5: PLC Kullanım Alanlarına Örnek

2.2. Grafcet Diyagramları

1975 yılında, karmaşık ayırık sistem kontrolüyle ilgilenen kırk araştırmacı ve endüstriyel yönetici, sıralı sistem kontrolü oluşturmaya yönelik modelleri ve yöntemleri karşılaştırmak ve araştırmak için her iki ayda bir toplanırdı. İki yıl süren zorlu ve zahmetli toplantılar, konuşmalar, sohbetler ve çalışmaların ardından GRAFCET adında bir model önerdiler. Bu isim, modelin grafik temele sahip olması nedeniyle "graf"tan ve çalışmayı destekleyen bilimsel dernekten gelen AFCET'ten (Association Française pour la Cybernétique Economique et Technique) gelmektedir. Bu ayırık sistem modelinin temel kavramları, günümüzde de oldukça açık ve basittir.

Ana Bileşenler:

- İlişkili eylemlerle adımlar:

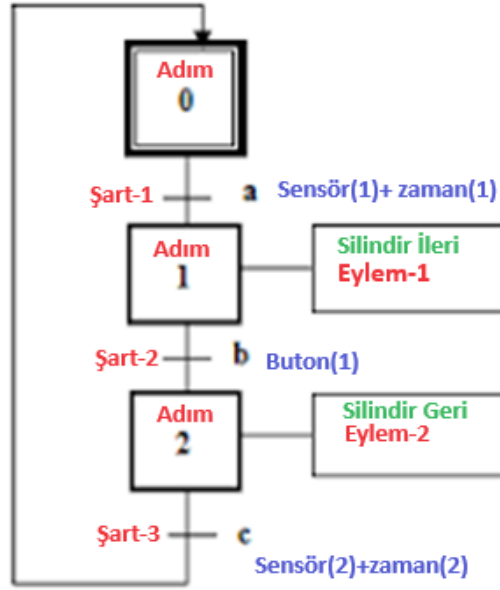
"Adım", bir eylemin gerçekleştirildiği sistemin kısmi durumunu temsil eder. Adım "aktif" veya "boşta" olabilir. İlgili "eylem", adım aktif olduğunda gerçekleştirilir ve adım boştayken uykuda kalır.

- İlişkili mantık şartlarıyla geçişler:

Geçişle ilişkili bir boole koşulu

Bir (veya birkaç) geçiş için koşullar doğrulandığında, önceki adımlar "boşta" olur ve sonraki adımlar "aktif" olur. Bir adımda çatışma halindeyken, faaliyetin aylaklığa göre önceliği vardır.

Sonuç olarak, GRAFCET diyagramları, DIN EN 60848 uyarınca kendileri için GRAFCET karakteristik dilinin (GRAPhe Fonctionnel de Commande Etapes/Transitions) kullanıldığı akış kumandalarının diyagramlar şeklinde gösterimleridir. Bir GRAFCET diyagramının temel elemanları adımlar, eylemler ve şartlar. Her adım bir önceki adımı sıfırlar. Şekil 2.6'da örnek bir grafcet diyagramı görülmektedir [17]:



Şekil 2.6: Örnek Grafcet Diyagramı

2.3. Konveyör Bant

Konveyör bantlar, katı malzemelerin uzun ve kısa mesafelerde yatay veya eğimli olarak taşınmasında kullanılırlar.

Konveyör bant sistemleri katı malzemeyi taşımak için iki döner kasnak kullanır ve çok sayıda kanca, dişli, kova ve geniş lastik kayışlardan oluşabilir.

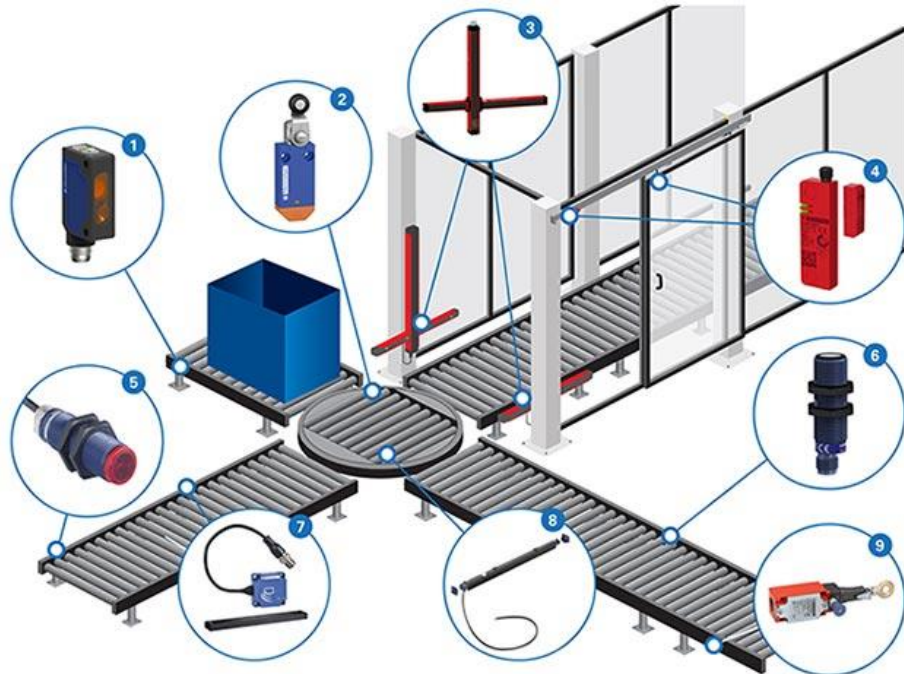
Konveyör bant sistemleri işletmelerin tercihinine göre üretilmektedir. Sabit konveyör bantlar kullanıldığı gibi tekerlekli mobil konveyör sistemleri de kullanılır ve istenilen yere kolaylıkla taşınabilir. Ayrıca bu mobil konveyör sistemleri, eğimli alanlarda kullanım için ayarlanabilir bir mekanizmaya sahiptirler.

Konveyör bantlar, üretimde ve stoklamada kullanıldığı gibi marketlerden havalimanlarına kadar günlük hayatımızda ortak kullanım alanlarına sahiptir. Şekil 2.7'de çeşitli konveyör bant sistemleri görülmektedir.



Şekil 2.7: Çeşitli Konveyör Bant Sistemleri

Konveyör bantlar üzerinde ürün algılayıcılar, sınırlayıcı mekanizmalar, silindirler ve kameralar kullanılabilir. Bu sayede ürün bir banttı diğer bant transfer edilebilir veya farklı prosesler algılayıcılardan gelen bilgilere göre bant üzerinde yaptırılabilir. Şekil 2.8’de bir konveyör bant sistemi üzerinde kullanılan çeşitli algılayıcılar görölmektedir [18]:



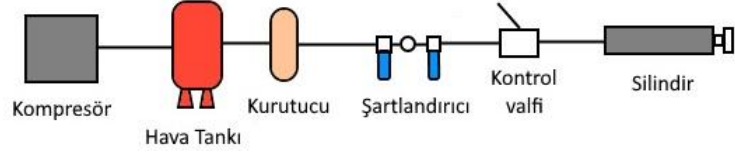
Şekil 2.8: Bir Konveyör Bant Sistemi Üzerinde Kullanılan Çeşitli Algılayıcılar

2.4. Elektropnömatik Sistem

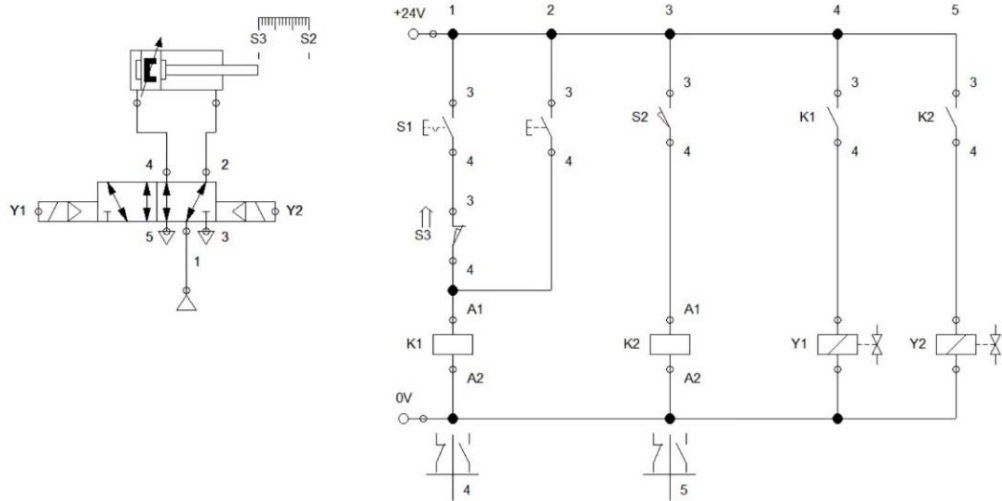
Pnömotik sistemler hava basıncıyla çalışır. Hava basıncının biri kontrol diğeri güç olmak üzere iki görevi vardır. Her enerji kaynağının dezavantajları olduğu gibi basınçlı havanın da dezavantajı mutlaka vardır. En büyük dezavantajı uzun hatlarda kullanılan havanın basıncının zayıflamasıdır.

Elektropnömatik sistemler, bu dezavantajı ortadan kaldırmak amacıyla tasarlanmış sistemlerdir. Elektropnömatik sistemlerde güç devresi ve kontrol devresi olmak üzere iki devre vardır. Güç devresi hava ile kumanda devresi ise elektrik ile kontrol edilir. Şekil 2.9 [19] ve Şekil 2.10'da Elektropnömatik sistem ve devre örneği gösterilmiştir. Elektropnömatik sistemlerin sağladığı avantajlar:

- Elektropnömatik sistemlerde elektronik valfler ile daha hızlı anahtarlama işlemleri yapılabilir.
- Elektropnömatik sistemlerde otomasyon kontrolü uzun mesafeden yapılabilir.
- Elektropnömatik sistemlerde pnömatikten daha verimli bir otomasyon sağlanır.
- Elektropnömatik sistemlerde EMK, PLC sistemler kullanarak kolay programlama ile kontrol yapılabilir.

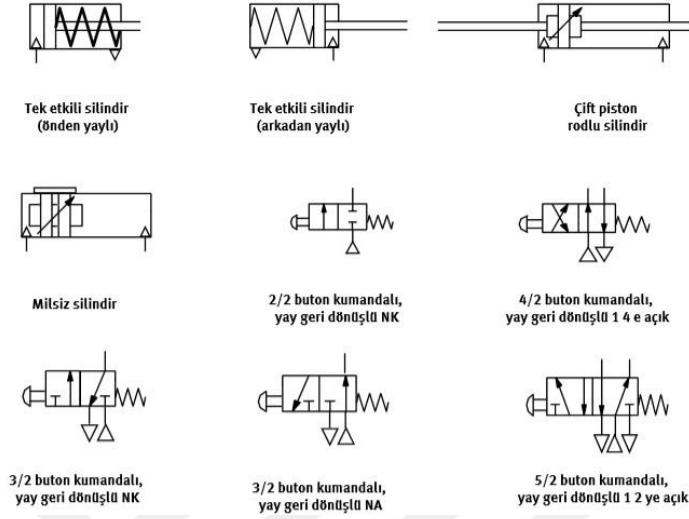


Şekil 2.9: Basit Elektropnömatik Sistem



Şekil 2.10: Elektropnömatik Devre Örneği

Şekil 2.11’de Elektropnömatik sembolleri, Şekil 2.12’de bazı Elektropnömatik malzemeler ve Şekil 2.13’te [20] ise basit bir Elektropnömatik devre örneği görülmektedir.



Şekil 2.11: Elektropnömatik Devrelerde Kullanılan Bazı Semboller



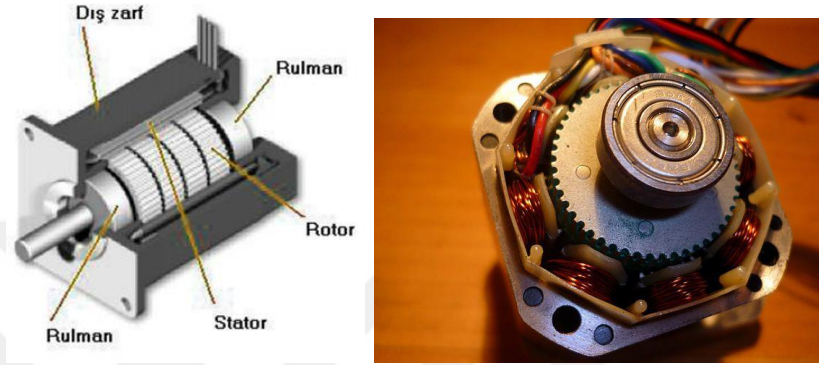
Şekil 2.12: Çeşitli Elektropnömatik Malzemeler



Şekil 2.13: Elektropnömatik Devre Uygulaması Örneği

2.5. Adım (Step) Motoru

Adım (step) motorları, kendi açısal konumunu adım adım değiştirebilen sabit mıknatıs kutuplu motorlardır. Çok çeşitleri olan step motorlarının yapıları incelendiğinde, stator ve rulmanlar olduğu görülür. Şekil 2.14'te [21] adım motorunun içyapısı ve Şekil 2.15'te adım motoru çeşitleri görülmektedir:



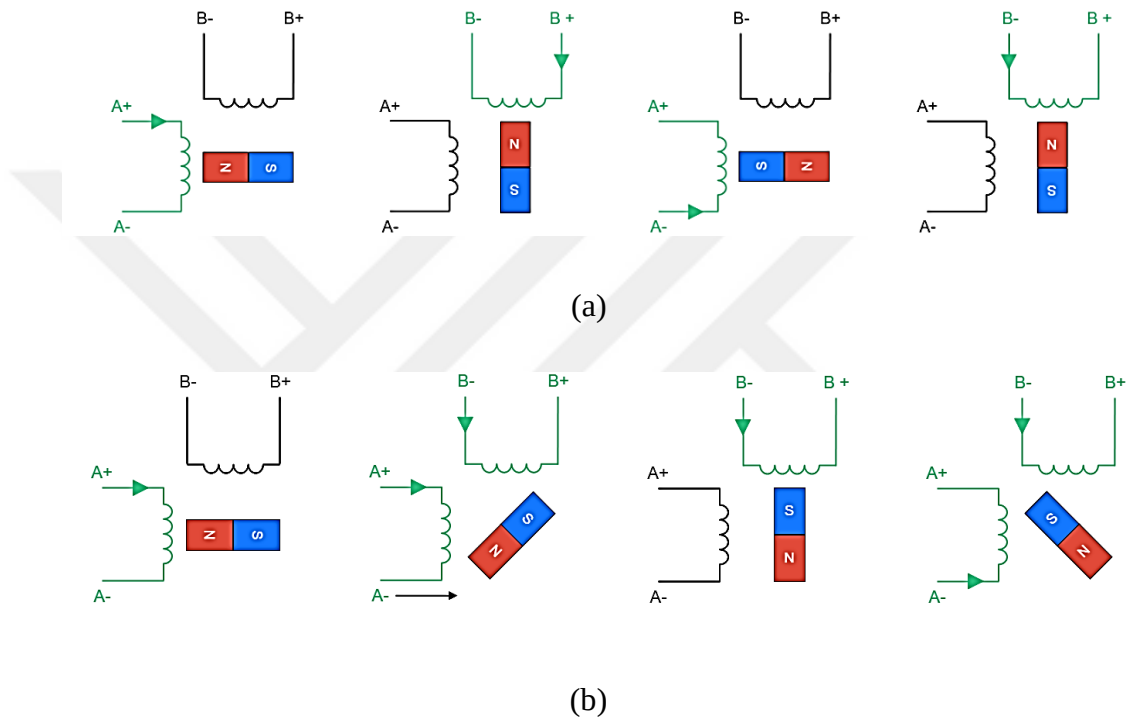
Şekil 2.14: Adım (step) Motoru İçyapısı



Şekil 2.15: Adım (step) Motoru Çeşitleri

Adım motorlarının hızı, çalışma öncesi ayarlanabilmektedir. Step motorlar endüstriyel kontrol sistemlerinde çokça kullanılır. Hareket sırasında adım kaybı yaşanmamasının nedeni rotorun açısal hızının yeterince küçük olmasıdır. Açısal mesafe biliniyorsa motor, açısal adımlarla kontrol edilebilir. Motor yapısının basit olması, motorun bakımı ve kullanım süresinin uzun olması maliyet açısından avantajdır. Step motorların ısınmadan gelen zararları da oldukça düşüktür.

Adım motoru, step motor sürücü devresi ile kolayca kontrol edilebilmektedir. Adım açısı, motorun yapısına göre 1.8, 7.5, 18, 45, 90 derece veya farklı açılarda olabilmektedir. Motorun hız kontrolü, uygulanacak sinyallerin frekansı değiştirilerek de yapılabilir. Adım motorlarının dönüş yönü, motora uygulanan sinyallerin sırası değiştirilerek yapılır. Şekil 2.16'da [22] bir step motorun çalışma mantığı görülmektedir:



Şekil 2.16: Step Motor Tam Adım Çalışma (a), Yarım Adım Çalışma (b)

Tam adım sürmede dört girişten belli olan iki uca aynı anda enerji verilir. Yarım adım sürmede ise tam adım ve tek fazlı sürmedeki işlemler birlikte arka arkaya uygulanır. Bu şekilde 90 derecelik step motor, 45 derece döndürülebilir. Bu şekilde motor, hassas ve kısa adımlarla döndürülebilir. Aşağıdaki tablolarda (Şekil 2.17) nasıl sürüldüğünü göstermektedir.

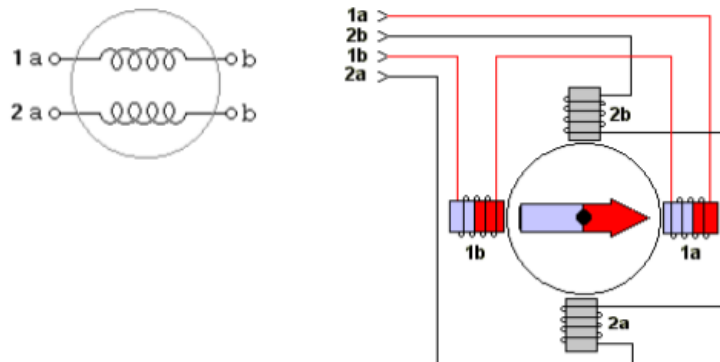
Unipolar Tek Fazlı Sürme Tablosu				
Adım	A1	B1	A2	B2
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1

Unipolar İki Fazlı Tam Adım Sürme Tablosu				
Adım	A1	B1	A2	B2
1	1	0	0	1
2	1	1	0	0
3	0	1	1	0
4	0	0	1	1

Unipolar İki Fazlı Yarım Adım Sürme Tablosu				
Adım	A1	B1	A2	B2
1	1	0	0	0
2	1	1	0	0
3	0	1	0	0
4	0	1	1	0
5	0	0	1	0
6	0	0	1	1
7	0	0	0	1
8	1	0	0	1

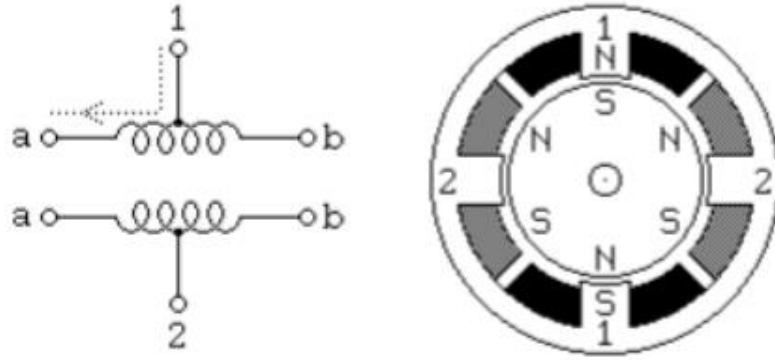
Şekil 2.17. Unipolar Step Motor Sürme Tabloları

Bipolar ve unipolar olmak üzere iki fazlı motorlarda iki farklı adım motoru çeşidi vardır. Bipolar step motorlarda; faz başına tek sargı düşer. Sargıdaki akımı ters yöne döndürmek için manyetik kutbu ters yöne döndürmek yeterlidir. Bipolar motorları ise çift kutupludur ve her iki yönden de akım alabilirler. Şekil 2.18’de [23] görüldüğü üzere bipolar motorların genelde 4 ucu bulunur:



Şekil 2.18. Bipolar Step Motor Yapısı

Unipolar step motorlar; sadece tek yönde akım ileten motorlardır. 5 ve 6 kablolu olmak üzere çeşitleri vardır. 6 Kablolu çeşidinde 2 ortak uç vardır. Bunlar genellikle birleştirilir ve tek bir besleme ucu varmış gibi kullanılırlar. (Şekil 2.19) [23]



Şekil 2.19. Unipolar Step Motor Yapısı

Günümüzde kullanılan step motorlar çeşitleri aşağıda görülmektedir:

- Sabit Mıknatıslı Step Motorlar
- Hidrolik Step Motorlar
- Hybrid Step Motorlar
- Lineer Step Motorlar
- Değişken Relüktanslı Step Motorlar

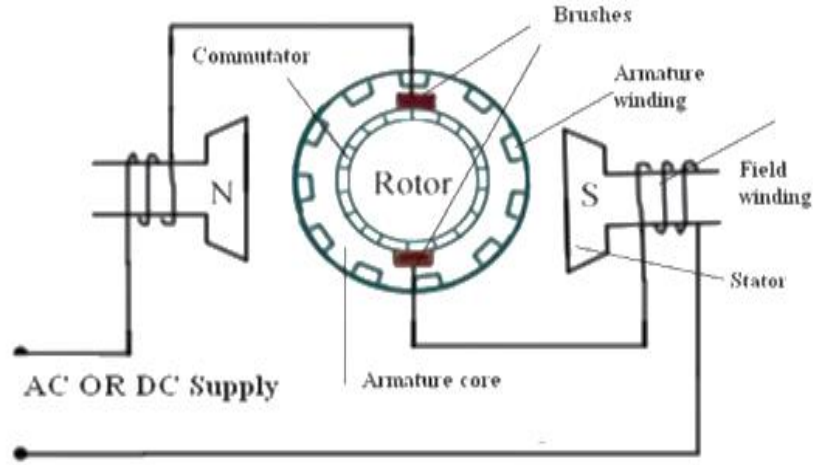
Günümüzde step motorların birçok kullanım alanlarından bazıları şunlardır:

- Yazıcılar
- Kameralar
- Teypler
- İmalat Tezgâhları
- Makinalar

2.6. Üniversal Motor

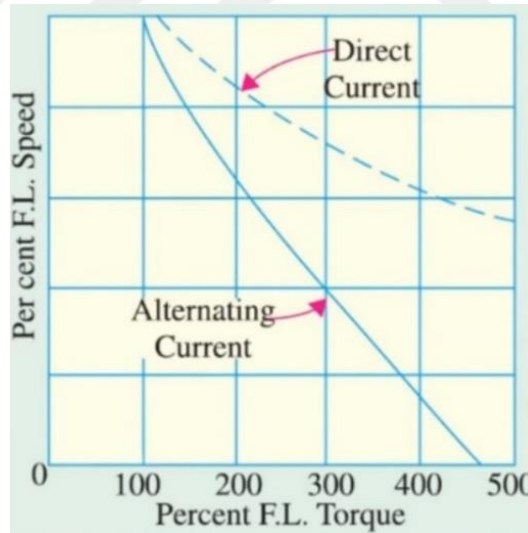
Üniversal motorlar Seri DC motorun birçok özelliğini sergileyebilirler. Üniversal motorlar, seri DC motor yapısına çok benzemekle beraber bu motorların onlardan en büyük farkı AC gerilim ile de çalışabiliyor olmalarıdır. Çamaşır makinesi, mutfak robotu, elektrikli süpürge gibi birçok aletlerde en çok kullanılan motor tipi üniversal motorlardır.

Şekil 2.20’de [24] üniversal motorun sargı bağlantıları ve içyapısı görülmektedir:



Şekil 2.20: Üniversal Motorun İçyapısı

Üniversal motorları kullanırken düşük frekanslar tercih edilir. AC çalışmada DC çalışmaya göre devir sayıları yüke bağlı olarak daha çok düşer. Şekil 2.21’de [24] üniversal motorun AC ve DC gerilimde hız ve tork ilişkisi görülmektedir:



Şekil 2.21: Üniversal Motorun AC Ve DC Gerilimde Hız Ve Tork İlişkisi

Universal motorun devir yönü değişimi, rotor uçlarını veya kutup sargısı uçlarını değiştirmek suretiyle yapılır. Fakat ikisini de aynı zamanda uyguladığımızda üniversal motorun devir yönünün değişmediği görülür.

2.7. Dokunmatik (Touch) Panel

HMI terimi “Human-Machine Interface” kelimelerinin baş harflerinden oluşur ve “İnsan-Makine Arayüzü” demektir. “Dokunmatik panel”, “touch panel”, “operatör panel” gibi isimlerini alır. Endüstriyel Otomasyon sektörünün önemli bileşenlerinden biridir.

Dokunmatik paneller sayesinde ekran kullanımı çok daha kolay bir hal alır. Herhangi bir işlem, fare ya da klavye gibi araçlara ihtiyaç duymaksızın, sadece dokunmatik panel üzerinden dilediğiniz alanın üzerine tıklayarak veya komut verilerek gerçekleştirilebilir. Dokunmatik paneller farklı alanlarda gözlemlenebilir. Özellikle Banka ATM’lerinde, endüstriyel makinelerde, güvenlik otomasyonları ve tıbbi cihazlar gibi alanlarda dokunmatik panel desteği büyük bir öneme sahiptir. Şekil 2.22’de endüstride kullanılan dokunmatik paneller (HMI) görülmektedir:



Şekil 2.22: Endüstride Kullanılan Dokunmatik Paneller

SCADA yazılımından izleme, kontrol, veri toplama, verilerin kaydı ve saklanması işlevlerini gerçekleştirmesi beklenir.

Endüstriyel Otomasyon sektöründe farklı boyutlu ekranlı, farklı modelde HMI bulunabilir. Bu HMI bileşenleri üzerinde SCADA tasarımları yapılır. Böylelikle sistemin kontrol, kumanda ve gözlemi görsel olarak dokunmatik paneller üzerinden sağlanabilir.

2.8. Proje Tabanlı Öğrenme (PTÖ) Yöntemi

Günümüzde eğitimin asıl amacı, özgürce düşünebilen, değişken dünya şartlarına uyum sağlayabilen, sorumluluk sahibi, bilgi ve becerilerini hayatı boyunca kullanabilen bireylerin yetişmesini sağlamaktır [25]. Millî Eğitim Bakanlığı, çağımızın becerilerini kazanmış bireylerin yetiştirilmesi amacıyla, proje tabanlı öğrenmeye yönelik çalıştaylar düzenleyerek, PTÖ ile ilgili makaleler yayınlamaya ve bu konuda daha birçok faaliyet ile PTÖ yöntemini okullarda teşvik etmektedir. PTÖ, geleneksel metotların ötesinde öğrenci merkezli bir öğrenme yaklaşımıdır.

Öğrencilerin gruplar halinde hayatın içinden (otantik) bir problemin çözümü için hazırladıkları bir eylem planı dahilinde araştırmalar yaparak bir ürün veya çözüm ortaya çıkarıp sunum yaptıkları öğrenme metotlarından biri de proje tabanlı öğrenmedir [26]. Proje tabanlı öğrenme, otantik soru ve sorunları çözmek için detaylı biçimde planlanmış ürün ve aktiviteleri kapsayan araştırma süreci yoluyla öğrencilerin bilgi ve becerilerini geliştirmeyi hedefleyen, sistematik bir öğretim metodu olarak tanımlanmaktadır [27].

PTÖ yaklaşımında, geleneksel öğretimde öğretme sürecinin ana ögesi olan “öğretme” faaliyeti, “öğrenme” faaliyeti ile yer değiştirilmek istenmiştir [28]. PTÖ, öğrenci merkezli aktif öğrenme yoludur. Öğrencilerin dikkatini çeken sorunlarla ilgilenmelerine ve orijinal ürünler ve sonuçlar ortaya koymalarına imkân tanır. PTÖ, gerçek hayatın içerisinde öğrenmeyi gerçekleştiren bir öğrenme yöntemidir [29].

PTÖ, öğrenci açısından bakıldığında, hayal etme, planlama, tasarlama ve kurgulama tabanlı bir öğrenme yöntemidir [30]. Öğretmenler açısından bakıldığında ise PTÖ, sınıfta gerçek yaşam ortamlarını oluşturan, öğrencilerin disiplinler arası ilişki kurmalarını sağlayan bir öğrenme metodudur [31].

PTÖ süreci öğrenci merkezlidir ve öğrenciler bu süreçte kendilerine özgü bir biçimde çalışırlar. Problemlerle karşılaşırırlar. Karşılaştıkları sorunların nedenleri araştırmaları, çözüm üretme süreçleri ve grup içi ilişkileri öğrenmenin kalitesine katkı sağlar. Öğrenciler bilgiyi kendileri keşfederek ve araştırarak elde ederler. Aktif ve keşfederek öğrenme sağlandığında derinlemesine öğrenme de gerçekleşmiş olur [32].

PTÖ yöntemi, özellikle ortaokulda ve mesleki okullarda bilişim teknolojileri, kodlama ve benzeri alanındaki alan derslerinde, günlük hayatta rastlanan problemlerin çözümünde kullanılabilir [33]. PTÖ hedeflerinden olan 21.yüzyıl becerilerinin kazanılmasında etkili olan kodlama eğitimi; birtakım şartların ve emirlerin istenilen sıralama ve düzene uygun olarak yapılmasını sağlayan komutlardır (Akkuş vd.,2019). [34]

Geleneksel kodlama dilleri yapısal olarak karmaşık olduğu için çoğu öğrenci için öğrenilmesi zor olabilmektedir. Görsel kodlama ortamı olan blok tabanlı yazımlar ise daha kolaydır. Blok tabanlı yazılımlara örnek olarak Scratch, Google Blockly ve Code.org verilebilir. Bunlar açık kaynak kodlu ve kodlamaya yeni başlayanlar için uygundur [35]. Blok tabanlı kodlamada, modelleme ile kodlama yapılabilir, doğrusal, dallanmış ve döngüsel algoritmalar kullanarak animasyonlar hazırlanabilir [36].

Bu çalışmada, üretilen deney seti üzerinde uygulanabilecek örnek bir PTÖ müfredatı hazırlanması hedeflenmiştir. Böylelikle set üzerinde eğitimler bu PTÖ müfredatı çerçevesinde verilebilecektir. MEB müfredatı ile uyumlu olacak olan bu müfredat, endüstriyel otomasyon teknolojileri alanı öğrencileri için uygulanabilir olacaktır. “Akran Değerlendirmesi”, “Öz Değerlendirme”, “Proje Süreci Değerlendirmesi”, “Ürün Değerlendirmesi”, “Sunum Değerlendirmesi”, “İşbirlikçi Çalışma Değerlendirmesi” gibi PTÖ değerlendirme ölçekleri de (rubrikler) hazırlanmıştır. Bu sayede PTÖ yöntemi başından sonuna kadar adım adım öğretmenler tarafından rahatlıkla yönetilebilecek, öğrenciler için verimli bir süreç yaşanmış olacaktır.

Bölüm 3

3. Materyal ve Yöntem

Bu çalışmada setin imalatında kullanılan malzemeler, günümüz endüstrisinde kullanılan malzemelerdir. Öğrenciler bu set ile sahada karşılaşacakları malzemeler üzerinden eğitim alabileceklerdir. Malzemeler elektronik, elektrik, bilgisayar ve mekanik tabanlı ürünlerdir. Multidisipliner bir alan olan endüstriyel otomasyon teknolojileri alanı, öğrenciler için çok yönlü eğitim yelpazesini kapsar. Öğrenci meraklı olduğu ve istediği alt alanda kendini geliştirebilir. Setin yapısı buna imkân sağlamaktadır.

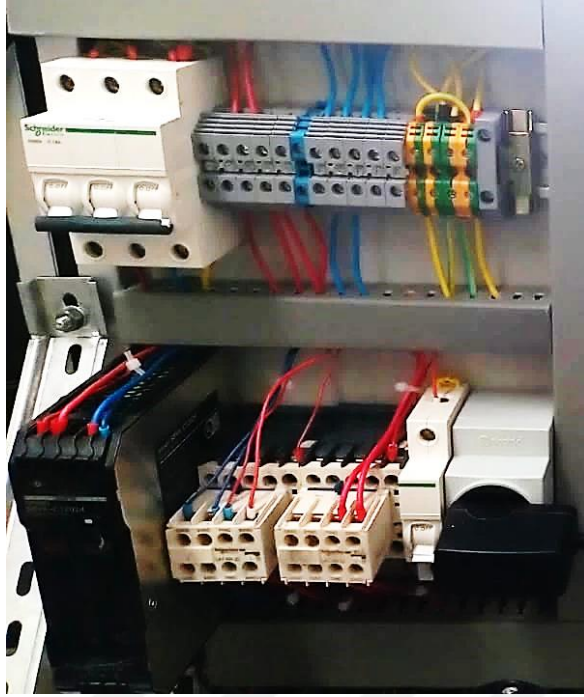
Setin imalatında öncelikle materyallerin montajı yapılmıştır. Sonrasında PLC programı için grafcet diyagramı hazırlanmış, PLC yazılımları bu diyagram üzerinden yazılmıştır. Sete montajlanan dokunmatik panel üzerinde SCADA tasarımı yapılmış, sistem test edilmiştir. Visual C# programlama dili kullanılarak görsel programlama ile SCADA yazılımı yazıldıktan sonra PTÖ müfredatı, planı ve değerlendirme ölçekleri hazırlanarak uygulanmış ve çalışmaya son verilmiştir.

3.1. Endüstriyel Otomasyon Eğitim Seti Genel Bilgileri

Bu çalışmada imal edilen set şu materyalleri içermektedir:

Güç Katı için:

- 1 adet ana şalter
- 1 adet sigorta ve 1 adet kontaktör (PLC Seçimi İçin)
- 24V güç kaynağı ve 220V/110V AC adaptör
- Klemens takımları
- 220V priz



Şekil 3.1: Güç Katı

PLC ve kontrol katı için:

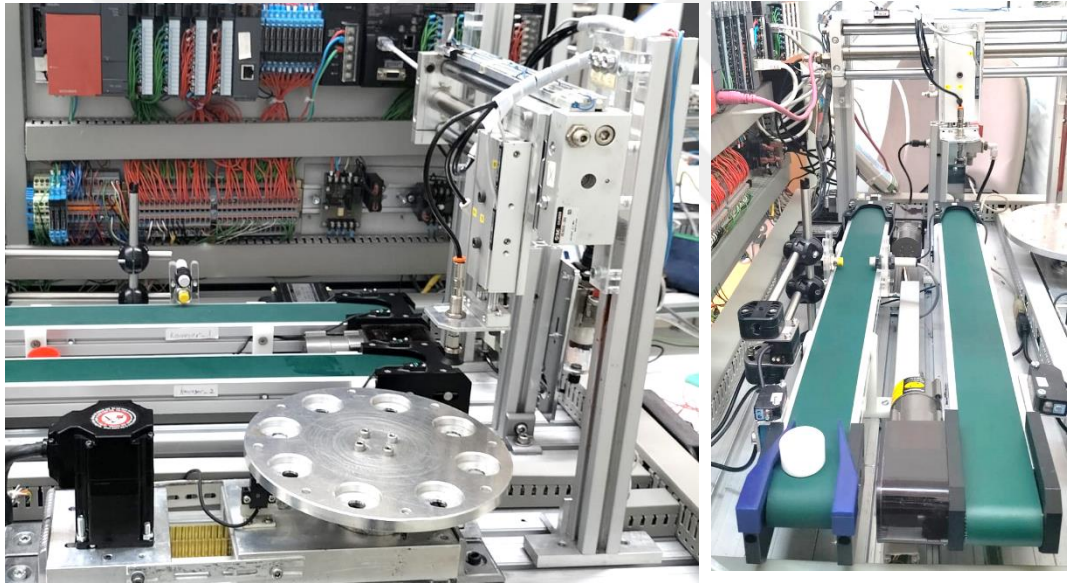
- 2 adet farklı marka PLC
- 2 adet üniversal motor sürücü (2 adet konveyör için)
- Buton takımları
- Dokunmatik panel
- Switch ve ethernet kabloları



Şekil 3.2: PLC ve Kontrol Katı

Aktüatör katı ve diğer donanım birimleri:

- 1 step (adım) motoru sürücüsü
- 2 adet üniversal motor ve konveyör bant
- 1 adet step motor, step motor sürücüsü
- Döner tabla sistemi
- 6 adet kızılötesi sensör (4 adet konveyör üstünde ve 2 adet döner tablada)
- 6 adet endüktif sensör (5 adet silindirlerde ve 1 adet konveyör üstünde)
- Pnömotik Valfler ve kompresör
- 2 adet çift etkili piston ve 1 adet vakum sistemi
- Sette kullanılacak plastik ve metal, kısa ve uzun materyaller
- Çeşitli kalınlıkta kablo, kablo kanalları ve hava hortumları
- Çeşitli mekanik parçalar



Şekil 3.3: Aktüatör Katı

Eğitim setinin montajlama işlemi için sırasıyla aşağıdaki adımlar uygulanmıştır:

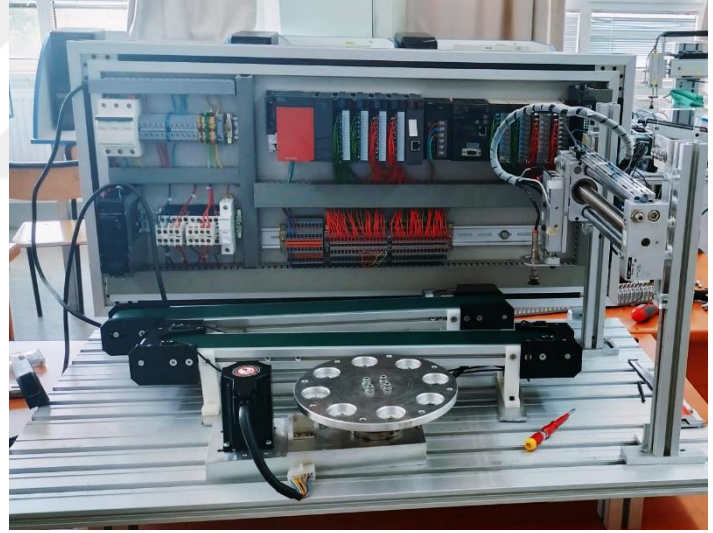
1. Setin profil aksamı L şeklinde montajlandı.
2. Kablo kanalları yerleştirildi. Dikey eksen 4 alana bölündü. (Güç alanı, PLC seçim alanı, PLC alanı ve klemens alanı dikey eksende bölündü.)
3. Güç katı için sigorta bağlantıları yapıldı.
4. PLC seçimi için sigorta ve kontaktör bağlantıları yapıldı.

5. İki farklı PLC, her PLC için iki giriş iki çıkış modülleri ile step motor sürücüsü ve ethernet switch dikey ekseninde yerleştirildi.
6. Her PLC için giriş ve çıkış bağlantıları klemenslere yapıldı.
7. Bağlantı alanına konveyör bantların motor sürücülerini yerleştirildi.
8. Konveyör bantlar, pistonlar setine montajlandı.
9. Pnömotik sistem bağlantıları yapıldı.
10. Buton takımları için pleksi malzeme kullanıldı. Butonlar yerleştirildi
11. Sensörler uygun ve gerekli yerlere yerleştirildi.
12. Döner tabla ve step motor setine eklendi.
13. Tüm malzemelerin kablo bağlantıları yapıldı.
14. Tüm giriş ve çıkışlar test edildi.
15. Çıkan problemler giderildi.

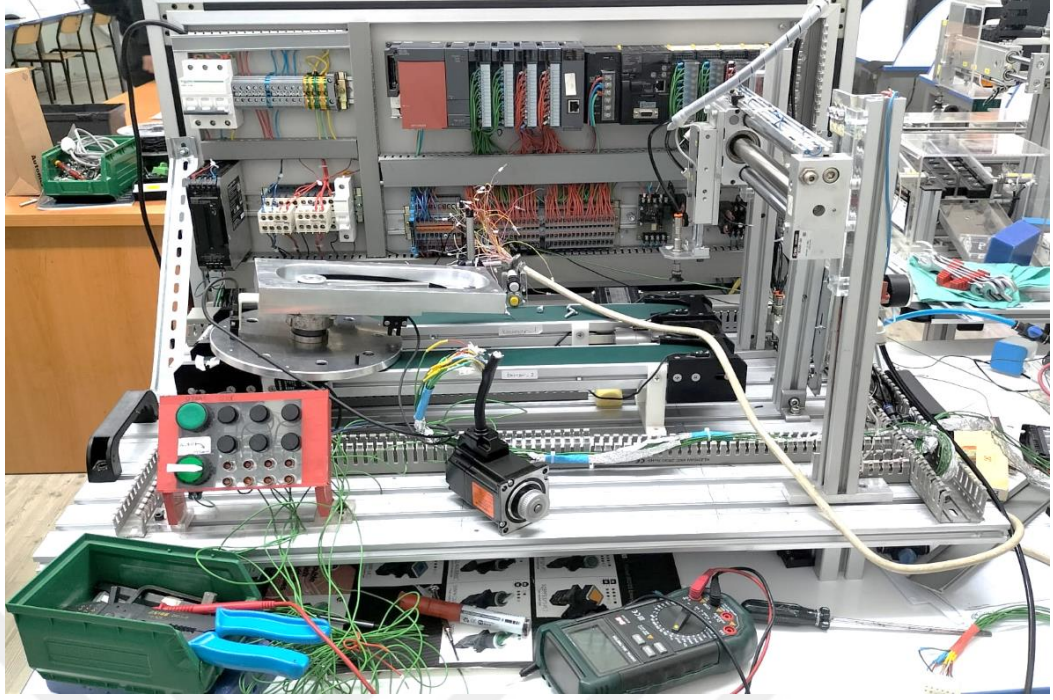
Şekil 3.4’de setin yapım aşamalarını göstermektedir:



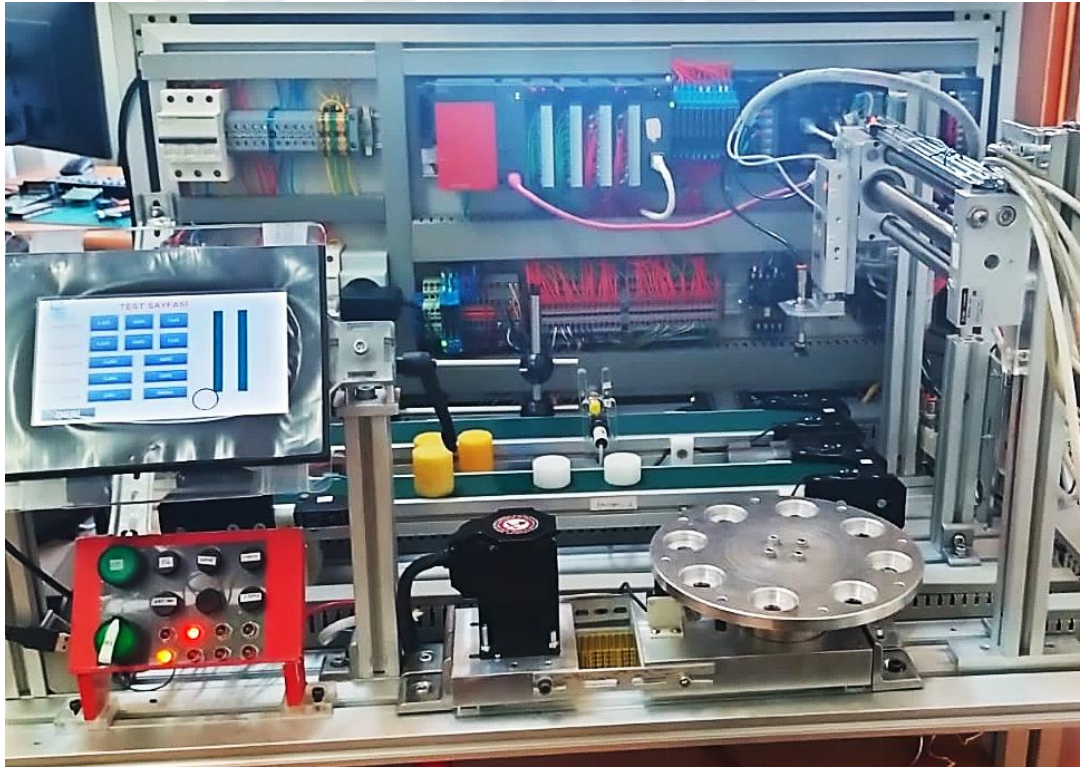
(a)



(b)



(c)



(d)

Şekil 3.4. Kablo Kanallarının Montajı (a), Güç Ve Kontrol Katının Montajı (b), Konveyörlerin, Motorların, Sensörlerin Ve Pnömotik Sistemin Montajı (c), Tüm Setin Montajlanmış Hali (d).

3.2. Proje Tabanlı Eğitim Müfredatı Tasarımı

Endüstri 4.0 çağında bilgi toplumundan bilgiyi ezberlemeleri yerine bilgiyi anlamaları, yorumlamaları, üretmeleri, bu bilgi ve becerilerini hayatlarına yerleştirmeleri beklenmektedir. PTÖ bu açıdan doğru ve uygulanabilir bir eğitim modeli olarak karşımıza çıkmaktadır.

Lucio tarafından yapılan bir sınıflandırmada projeler; öğrenme projesi, yapımı projeleri, estetik nitelikli projeler, problem projeleri, çalışma projeleri olarak adlandırılmıştır [38].

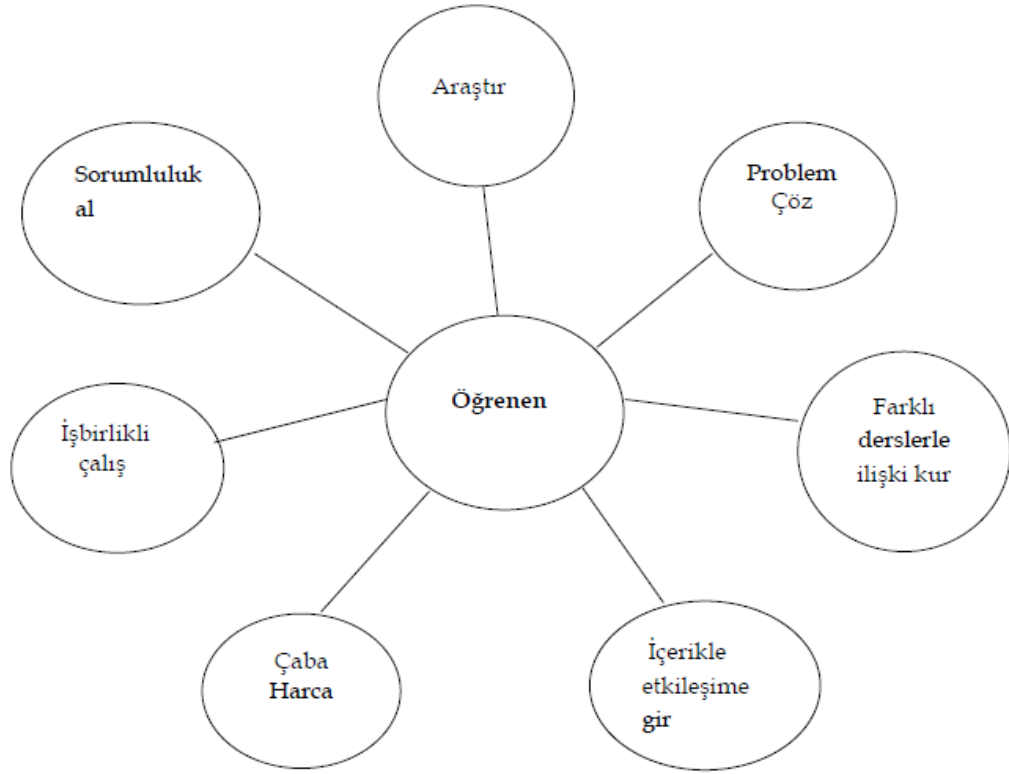
Binbaşıoğlu'na göre başlıca 4 çeşit proje vardır: 1) Estetik projeler 2) Sorun projeleri 3) Nesnel projeler 4) Beceri Projeleri [39].

Gündüz'e göre proje hazırlama aşamaları konu seçimi, zamanın planlanması, çalışma takvimi yapılması, kaynak arama ve toplama; okuma, gözlemleme, bilgi toplama, beyin fırtınası, resim, kanıt toplama ve not etme, toplanan bilgilerin ve verilerin sınıflandırılması, projenin taslağının oluşturulması, projenin yazılması, projenin değerlendirilmesi, projenin sunumu aşamalarından oluşur. [40]

Wolk, proje tabanlı öğrenmede öğretmenin rollerini aşağıdaki gibi belirtmektedir:

- Öğretmenler, öğrencilerin, yaratıcı, güncel ve ilginç projeler geliştirmelerinde çok önemli bir role sahiptir.
- Öğretmenler yaşam boyu öğrenme için öğrencilere model olmalıdır.
- Öğretmenler öğrencilerin sıkı çalışarak öğrenmeyi ciddiye almalarını, düşünceli ve sorumluluk sahibi olmalarını, kendilerini aşmalarını ve kaliteli ürünler üretmelerini beklemelidir. [41]

PTÖ yaklaşımında öğrenenin rolü ise Şekil 3.5'te görülmektedir [42]:



Şekil 3.5: PTÖ Yaklaşımında Öğrenenin Rolü

PTÖ, öğrencilerin üst düzey becerilerini geliştirir ve fiziksel ve sosyal çevrelerine karşı sorumluluk duygularının artmasını sağlar [43].

Ayrıca PTÖ, öğrencilerin eleştirel düşünme becerilerini oluşturma ve arttırmada yardımcı olabilir. Öğrencilerin aktif katılımları onların kendi fikirlerini şekillendirmelerini ve bakış açılarını ortaya koymalarını sağlar [44].

Proje tabanlı öğrenme yaklaşımının avantajlarını Hamurcu [45] şu şekilde belirtmiştir: Öğrencilerin öğrenme becerilerini geliştirir ve zenginleştirir. Yaşam boyu öğrenmeyi sağlar. Grupla çalışma ve iş birliğine dayalı öğrenme etkinliklerine katılımı sağlar. Öğrencilerin bilgilerini yansıtmaları ve katılımları için çoklu yollar önerir. Zekânın farklı boyutlarının kullanımına izin verir (kinestetik, uzamsal, mantık, dil vb.) Öğrenci performansı hakkında aileye, öğretmene ve okul yönetimine anlamlı bilgiler verir. Öğrenciler gerçek yaşamla oluşturduğu ürünleri ve performanslarını birleştirir. Problem çözme becerilerini ve probleme dayalı öğrenme becerilerini geliştirir. Değişik konularda proje yoluyla kazandığı bilgi ve becerilerini uygulama fırsatı bulurlar.

Proje tabanlı öğrenme öğrencilere çeşitli beceriler kazandırır:

- Yaşamsal Beceriler: Bir toplantı yönetmek, liderlik, bir bütçe ve plan yapmak vb.
- Teknolojiyi Kullanma Becerisi: Bilgisayar, telefon radyo, video vb. araçları kullanma
- Bilişsel Süreç Becerileri: Karar alma, eleştirel düşünme ve problem çözme becerileri
- Öz-denetim Becerileri: Hedefler belirleme, işlemleri organize etmek, zaman yönetimi
- Tutumlar: Öğrenmeye ilgi, gelecek için eğitime merak
- Eğilimler: Öz-denetim, başarı hissi
- İnançlar: Öz-yeterlik inancı

Proje tabanlı öğrenmenin dezavantajları şöyle sıralanabilir:

Proje Tabanlı Öğrenme, öğretmenin iş yükünü ve sorumluluğunu artırır. PTÖ süreci uzun olabilir. Süreç iyi planlanmazsa ve sınırları iyi çizilmezse, amaçta sapma ve dağılma olabilir.

Proje tabanlı öğrenme sürecinde öğretmenlerin uygulaması gereken altı aşama şunlardır [46]:

- 1.Proje konusuna karar verme
2. Zaman çizgisini oluşturma
- 3.Etkinlikleri planlama
- 4.Değerlendirme planı hazırlama
- 5.Öğrencilerle birlikte projeyi başlatma
- 6.Projeyi tamamlama ve ürünlerin yansıtılmasına katkıda bulunma

Proje tabanlı öğrenme yaklaşımının planlanması ve uygulanması sırasında öğrencilerin dikkat etmesi gereken noktalar şunlardır [46]:

- 1.Araştırmanın sınırları
- 2.Grup üyesi ve birey olarak sorumlulukları
- 3.Araştırma süresi
- 4.Planların uygulanabilirliğini gözden geçirme
- 5.“Projelerimizi bu planı kullanarak tamamlayabilir ve yüksek nitelikli bir iş yapabilir miyiz?”
- 6.Eskiden yapılan işler yerine orijinal ve yeni konuları tercih etme

Proje tabanlı öğrenme yaklaşımının planlanması ve uygulanması sırasında proje seçiminde dikkate alınması gereken özellikler şunlardır [46]:

- 1.Proje istendik etkinlikleri kapsayıcı olmalı ve boş uğraşılardan arındırılmalı
- 2.Projenin hazırlanması için ayrılan süre yeterli olmalı
- 3.Proje işlenen konuyla ilgili olmalı, ulaşılabilecek davranışlar açıkça belirtilmeli
- 4.Projeden elde edilecek yarar araç-gereç ve kaynaklar için yapılan yatırıma değer nitelikte olmalı
- 5.Öğrencilere etkinlikler yoluyla sorunlarını çözebilme olanağı vermeli
- 6.Öğrencinin yaratıcılık, sorumluluk ve başarı duygusunu tatmasına uygun olmalı
- 7.Proje öğrenciler normal yaşam koşulları içinde işlenmeye uygun olmalı
- 8.Proje öğrencileri düşünmeye, incelemeye ve araştırmaya yöneltmelidir.

Proje planını hazırlarken göz önüne alınması gereken planlama öğeleri şunlardır [46]:

1.Program: Projede yapılacak olan işler ve sürelerini gösteren iş takviminin hazırlanması

2.İş Bölümü: Gruptaki her bir üyenin görev tanımının yapılması

3.Bütçe: Grupta yapılacak işler için harcanacak paranın önceden belirlenmesi

4.Araştırma Planı: Bilgi toplamak için kullanılacak yöntem, araç-gereç ve kaynakların listesi, araştırmayı tanımlamak için gerekli eylemlerin ve yapılacak araştırmalara ilişkin iş bölümünün yer aldığı bir plan hazırlanması

5.Materyaller: Araştırma için gerekli olan araç gereçlerin, kontrol listelerinin belirlenmesi ve hazırlanması

6.Yayın Listesi: Araştırmada kullanılacak fotoğraf, gazete, video, radyo, televizyon vb. yayın araçlarının listesinin hazırlanması

7.Diğer: Proje süreci içerisinde ortaya çıkması muhtemel olasılıkların ve çözüm yollarının düşünülmesi

PTÖ yaklaşımında değerlendirme, ürün ve sürecin birlikte değerlendirildiği bir yapı içermektedir [47].

PTÖ yaklaşımında değerlendirme, süreç ve ürünün birlikte ele alındığı tümel değerlendirme sistemidir [48].

Bu çalışmada PTÖ planları ve müfredatı hazırlanırken, Buck Institute for Education myPBLWorks web sitesinden (<https://my.pblworks.org/>) ve Buck Institute PBL (Project Based Learning) Form şablonlarından yararlanılmıştır. Bu formlarda özellikle 21 yy. becerilerine odaklanılmıştır. PTÖ süreci ve bu süreç içinde gerekli olan her şey gayet net ve açık bir şekilde formlarda belirtilmiştir.

Bu çalışmada MEB Endüstriyel Otomasyon Teknolojileri Alanı 11. sınıflarda haftalık 10 ders saati üzerinden verilen Atölye-11 dersi için 16 haftalık PTÖ planı hazırlanmıştır. 12 hafta süreyle öğrencilere set üzerinde gerekli teorik ve uygulamalı dersler verilmekte, ardından 4 haftalık sürede PTÖ tabanlı çalışma yapmaları beklenmektedir.

Bu tez teorik bilgi ve temel uygulamalara kaynak olacaktır. Öğrencilerin kazanması beklenen bilgi ve beceri kazanımları ile yapılması gereken eğitimler müfredatta yazılmış, değerlendirme formları (rubrikler) hazırlanmış ve not kriterleri belirlenmiştir. Uygulamaya hazır hale getirilmiştir. Aşağıda Şekil 3.6’ da eğitim seti için hazırlanan 16 haftalık 9 sayfadan oluşan PTÖ planı (a), müfredatı (b), öğrenme takvimi (c) ve MEB yönetmeliğine uygun değerlendirme kriterleri (d) görülmektedir:

PROJE TABANLI ÖĞRENİM(PTÖ) PLANI							
Proje Adı:	PLASTİKTEN ARINALIM				Süre: 16 Hafta		
Konu/Ders:	ATÖLYE – 11 (PLC UYGULAMALARI)				Sınıf Seviyesi: 11 Haftalık Ders Saati: 10		
Varsa dahil edilecek diğer konu alanları:	Elektronik, Mekanik, Motorlar, Sensörler, Plastik Teknolojileri, Biyoloji, Çevre Bilgisi						
Proje Fikri Konunun özü, zorluğu, araştırma, senaryo veya problem:	Plastik maddeleri metaller arasından ayıklayarak plastik atıklardan arınmak fikri hoşunuza gitti. Elinizde iki konveyör ve bir döner tabladan oluşan bir endüstriyel otomasyon sistemi var. Plastik ile metal maddeleri bu sistem sitem ile ayırmak, plastik atıklardan arınmak istiyorsunuz.						
Yönlendirici Soru	İki konveyör ve bir döner tabladan oluşan PLC tabanlı bir otomasyon sistemi üzerinde SCADA yazılımınız ile boylarına (uzun –kısa) da dikkat ederek plastik ve metal ürün ayırımı yapabilir misiniz?						
Ele Alınacak İçerik ve Beceri Standartları:	1. PLC tabanlı kontrol devresi kullanın (İki farklı PLC sistemde bulunmakadır.) 2. Sensörleri kullanın. (indüktif , kapasitif ve kızılötesi) 3. Üniversal step motor kullanın. (Döner Tabla Kontrolü yapın) 4. Sistemi Programlamayın ve Test Edin. 5. Yazılım ve donanım becerilerinizi sergileyin. 6. Proje sunumu yapın. Proje raporu hazırlayın. 7. Proje youtube videosu yayınlayın.						
		Ö+D	T			Ö+D	T
21. Yüzyıl Becerileri açıkça öğretilip değerlendirilecek (Ö&D) veya proje çalışmasıyla teşvik edilecek (T) ama öğretilip değerlendirilmeyecek:	İşbirliği & Sorumluluk	X		Tasarımcılık & Hayal Gücü			X
	Sunum	X		İletişim (sözlü ve yazılı)			X
	Kritik Düşünme		X	Eleştirel Düşünme			X
	İş Disiplini & Devamlılık & Güvenilirlik	X		Problem Çözme			X
	Liderlik		X	Sağlık Okuryazarlığı			X
	Üretkenlik	X		Sosyal Okuryazarlık ve Kültürler arası/Çok kültürlülük Okuryazarlığı			X

Last page developed by STEM M1 Champions (MACUL) © 2008 Buck Institute for Education 1

			Tanıtım Kiti:		
Final Ürünleri ve Başarımları	Grup:	Her grup aşağıdaki adımları gerçekleştirecektir: - Her grup 4 öğrenciden oluşacaktır. - Her grup kendi grubuna ait bir isim belirleyerek logosunu tasarlayacaktır. - Her grup kendi grup başkanını seçecek ve görev paylaşımı yapacaktır. - Her grup çalışma planını, görev tablosunu ve zaman çizelgesini oluşturacaktır. - Her grup öğrenme günlüğü tutacaktır. - Her grup sunum yapacaktır. (Sunumda proje yapım aşamaları, resimler ve sonuçlar olacaktır.) - Her grup kendi çalışma videosunu youtube kanalı açarak yayınlayacaktır. - Her grup proje raporu hazırlayacaktır.	Sınıf:	X	
			Okul:		
			Web:	X	
	Ferdî:	Her öğrenci aşağıdaki etkinlikleri bağımsız olarak tamamlayacaktır: - Öğrenci iş-zaman çizelgesi - İşbirliği değerlendirilmesi. - Öz değerlendirme. - Proje değerlendirilmesi.	Uzmanlar:		
			Web:		
			Diğer:		
	Sorgulama başlatmak, öğrencileri dahil etmek için başlangıç etkinliği:	- Öğrenciler plastik atıklar ve çevre bilinci ile ilgili videolar izleyerek teknolojinin bu alanlarda kullanımını hakkında bilgi sahibi olacaklardır(Plastik teknolojileri alan uzmanından bir seminer) - Öğrenciler plastik atıkların zararları hakkında bilgi toplayarak fikirlerini söyleyeceklerdir. - Öğrenciler plastik ve metal ayrıştırmanın ekonomiye katkısını araştıracaklardır. Sahada bu işlemlerin nasıl gerçekleştirildiği hakkında uygulama videoları izleyeceklerdir. - Öğrencilere atık tesisi ziyareti yaptırılacaktır. - Öğrenciler plastik üretim tesisi ziyareti yapacaklardır.			

Last page developed by STEM MI Champions (MACUL) © 2008 Buck Institute for Education 2

Değerlendirmeler	Biçimlendirici Değerlendirmeler (Proje Esnasında)	Kısa Sınavlar/Testler	X	Sunum Alıştırmaları	
		Günce/Öğrenme Günlüğü		Notlar / Haftalık Raporlar	X
		Ön Planlar / Anahatlar / Prototipler		Kontrol Listesi	
		İlk tasarımlar		Kavram Haritası	
		Online Testler/Sınavlar		Diğer:	
	Özetleyici Değerlendirmeler (Proje Bitiminde)	Rubrik ile yazılmış ürünler: Bir kamu yararı teklifi / proje fikri	X	Rubrik ile diğer ürün ve performanslar:	X
		Rubrik ile sözlü sunumlar Proje sunumu	X	Akran Değerlendirmesi	X
		Çoktan Seçmeli/Kısa Cevaplı Testler		Öz Değerlendirme	X
Yazılı Sınav		X	Proje Nihai Raporu	X	
Gerekli Kaynaklar	Sahada insan, tesisler:	Plastik ayıklama ve üretim tesisi			
	Ekipmanlar:	Endüstriyel Otomasyon Eğitim Seti, Laboratuvar Donanımı, PC			
	Materyaller:	Teknik doküman ve el kitapçıkları, ürün katalogları, ders notları, ders kitabı			
	Topluluk Kaynakları:	İnternet ve eğitim videoları			
Fikir Paylaşım Yöntemleri	(Bireysel, Grup, ve/veya Tüm Sınıf)	Ekip Toplantıları / Grup Tartışmaları	X	Odak Grubu	
		Bütün Sınıf Tartışma		"Balık Kavanozu" Tartışması	
		Anket		Youtube	X

Last page developed by STEM MI Champions (MACUL) © 2008 Buck Institute for Education 3

(a)

PROJE ÖĞRETİMİ VE ÖĞRENME MÜFREDATI	
Proje Adı: PLASTİKTEN ARINALIM	Kurs/Dönem: 1. Dönem
Nihai ürün/performansları başarı ile tamamlamak ve özetleyici değerlendirmelerde başarılı olmak için Öğrencilerce Gerekli Bilgi ve Beceri Kazanımları	Proje öğretmeni, diğer öğretmenler, uzmanlar, mentorlar, topluluk üyeleri tarafından sağlanacak Dersler (Teorik ve Uygulamalı)
PLC ve çevre donanımı bilir ve kullanır.	PLC ve çevre donanım bilgileri dersleri verilir.
Sensorleri ve makine elemanlarını bilir ve kullanır.	Sensor çeşitleri, kullanımı ve makine elemanları hakkında dersler verilir.
Elektropnömatik malzemeleri kullanır.	Elektropnömatik dersleri verilir.
Grafcet kullanarak PLC yazılımları yazar.	Grafcet ile PLC yazılımları geliştirme dersleri verilir.
Konveyör bant kontrollerini yapar.	Konveyör bant kontrolü uygulama dersleri verilir.
İki konveyör bant üzerinde ürün taşıma işlemlerini yapar.	İki konveyör bant ile ürün taşıma dersleri verilir.
Uzun ve kısa malzemeleri konveyör üzerinde ayırır.	Uzun ve kısa malzemeleri konveyör üzerinde ayırma dersleri verilir.
Metal ve plastik malzemeleri konveyör üzerinde ayırır.	Metal ve plastik malzemeleri konveyör üzerinde ayırma dersleri verilir.
Döner tabla hareketlerini kontrol eder.	Döner tabla hareketlerini kontrol dersleri verilir.
Döner tabla ve konveyör bant arasında ürün taşımasını yapar.	Döner tabla ve konveyör bant arasında ürün taşıma dersleri verilir.
PLC modeline ait .NET (C#) kütüphanelerini kullanmasını bilir.	PLC modelinin .NET (C#) kütüphanelerinin kullanması dersleri verilir.
C# ve PLC kütüphanelerini kullanarak SCADA yazılımı yapar.	C# ve PLC kütüphanelerini kullanarak SCADA yazılımı dersleri verilir.

Last page developed by STEM MI Champions (MACUL) © 2008 Buck Institute for Education 4

(b)

PROJE TABANLI ÖĞRENME TAKVİMİ				
Proje: Plastikten Arınım			Başlangıç Tarihi:/...../.....	
PAZARTESİ	SALI	ÇARŞAMBA	PERŞEMBE	CUMA
BİRİNCİ HAFTA				
PLC nedir? PLC giriş ve çıkışları nelerdir? PLC bağlantıları nasıl yapılır?	PLC çevre donanımları nelerdir? PLC çevre donanımları arasındaki farklar ve PLC bağlantıları nasıl yapılır? MİNİ SINAV-1			
İKİNCİ HAFTA				
Sensör nedir? Sensör çeşitleri nelerdir? Sensör-PLC bağlantıları nasıl yapılır?	Makine elemanları nelerdir? Makine elemanları bağlantıları nasıldır? Mekanizma çeşitleri nelerdir? MİNİ SINAV-2			
ÜÇÜNCÜ HAFTA				
Pnömatik ve elektropnömatik sistem nedir? Pnömatik ve elektropnömatik elemanları nelerdir? Pnömatik ve elektropnömatik malzemeler nasıl bağlanır?	Pnömatik ve elektropnömatik devre çizimleri nasıl çizilir? Pnömatik ve elektropnömatik devre çizimi programı nasıl kullanılır? MİNİ SINAV-3			
DÖRDÜNCÜ HAFTA				
Grafcet nedir? Grafcet nasıl kullanılır? Grafcet örnekleri nelerdir?	PLC yazılım ortamı nasıldır? PLC programı nasıl yazılır? Grafcet ile basit PLC yazılımı nasıl yazılır? MİNİ SINAV-4			

Last page developed by STEM MI Champions (MACUL) © 2008 Buck Institute for Education 5

BEŞİNCİ HAFTA				
Konveyör nedir? Konveyör çeşitleri nelerdir? Konveyör motorları nelerdir?	PLC ile bir konveyör üzerinde ileri-geri ürün taşıma nasıl yapılır? MİNİ SINAV-5			
ALTINCI HAFTA				
PLC ile iki konveyör üzerinde ileri-geri ürün taşıma nasıl yapılır?	İki konveyör kullanarak elektropomatik sistem ve PLC ile ürün taşıma nasıl yapılır? MİNİ SINAV-6			
YEDİNCİ HAFTA				
PLC ile iki konveyör üzerinde uzun-kısa ürün ayırma nasıl yapılır?	PLC ile iki konveyör üzerinde metal-plastik ürün ayırma nasıl yapılır. MİNİ SINAV-7			
SEKİZİNCİ HAFTA				
Döner tabla nedir? Döner tabla çeşitleri ve kullanım yerleri nelerdir? Döner Tabla motorları nelerdir?	PLC ile döner tabla kontrolü nasıldır? 1. YAZILI – UYGULAMA SINAVI			

Last page developed by STEM MI Champions (MACUL) © 2008 Buck Institute for Education 6

DOKUZUNCU HAFTA				
PLC ile döner tabladan konveyör banda ürün taşıma nasıl yapılır?	PLC ile konveyör banttan döner tablaya ürün taşıma nasıl yapılır? MİNİ SINAV-8			
ONUNCU HAFTA				
C# ile temel araç uygulamaları nasıl yapılır? C# ile temel grafik uygulamaları form üzerinde nasıl yapılır?	C# ile port kontrolleri nasıl yapılır? C# ile veri tabanı işlemleri nasıl yapılır? MİNİ SINAV-9			
ONBİRİNCİ HAFTA				
SCADA nedir? SCADA programları nelerdir? Basit bir SCADA uygulaması nasıldır?	C# ile PLC component kullanımı nasıldır? C# ile PLC giriş/çıkış kontrolleri nasıldır? MİNİ SINAV-10			
ONİKİNCİ HAFTA				
C# ile basit SCADA uygulaması nasıl tasarlanır?	C# ile basit SCADA uygulaması nasıl yazılır?			

Last page developed by STEM MI Champions (MACUL) © 2008 Buck Institute for Education 7

ONÜÇÜNCÜ HAFTA				
Proje Fikri ve Sürüş Sorusunun Verilmesi Hazırlayıcı video ve bilgi aktarımı Proje Ekiplerinin Oluşturulması	Grupların isimlendirilmesi ve logo tasarımları Grup Toplantısı, Görev Paylaşımı ve İş-zaman Planlamaları			
ONDORDUNCU HAFTA				
Grafçet diyagramlarının oluşturulması	PLC yazılımının Grafçet diyagramlarına göre yazılması			
ONBEŞİNCİ HAFTA				
PLC yazılımının Grafçet diyagramlarına göre yazılması	PLC yazılımının Grafçet diyagramlarına göre yazılması ve test edilmesi			
ONALTINCI HAFTA				
PROJELERİN BİTİRİLMESİ Grup sunumların yapılması Grupların youtube videolarının izlenmesi Raporların teslim edilmesi Öğrenci ve grup değerlendirme formlarının öğrenciler tarafından doldurulması. Bir kamu yararı teklifi / proje fikri	NOT İLE SÜRECİN DEĞERLENDİRİLMESİ 2. YAZILI – UYGULAMA SINAVI			

Last page developed by STEM KI Champions (MACUL) © 2008 Buck Institute for Education 8

(c)

Ders Tasarımı/Değerlendirme Kriterleri: Engelleri kaldırmak ve tüm öğrencilere erişim sağlamak için derslerin dikkatli bir şekilde yapılandırılması.	Kontrol Noktaları:
Her öğrenci hem ferdî hem grup notu ile değerlendirilecektir. Ferdî notu NOT-1 ile Grup Notu ise NOT-2 olarak isimlendirilir.	✓ FORM-1: Öğrenci Sözleşmesi (13. Hafta)
Her öğrenci için 10 tane "Mini Sınav" ortalaması NOT-1 (Ferdî Not) olarak değerlendirilecektir.	✓ FORM-2: Öğretmen Gözlem Formu (Her hafta)
Her grup için süreç içindeki gayretli çalışmaları, işbirlikçi çalışma disiplinleri ve uyumları, haftalık öğrenme raporlarının içeriği ise NOT-2 (Grup Notu) olarak değerlendirilecektir.	✓ FORM-3: Öğrenci Öz-Akran Değerlendirme Formu (16. Hafta)
Her öğrencinin ferdî ve ekip notları (not-1 ve not-2) ve öğretmen gözlemleri <u>1. Performans notuna</u> yön verecektir. Not1 %50, not2 %30 ve <u>öğretmen</u> gözlemi %20 olarak etkilenecektir. ,	✓ FORM-4: <u>İşbirlikçi Öğrenme</u> Formu (16. Hafta)
Nihai proje çalışması <u>değerlendirmesi</u> , <u>sunum</u> ve raporlara verilecek <u>no</u> ise <u>2. Performans notu</u> olarak kullanılacaktır.	✓ FORM-5: Proje Değerlendirme Formu (16. Hafta)
Proje süresince <u>2 kez yazılı-uygulama sınavı</u> yapılacaktır. 2 yazılı ve 2 performans notu ise BAŞARI NOTU olacaktır.	✓ FORM-6: Sunum Değerlendirme Formu (16. Hafta)
BAŞARI NOTU, 2 yazılı uygulama sınavı notu ile 2 performans notunun aritmetik ortalamasıdır.	✓ FORM-7: Rapor Değerlendirme Formu (16. Hafta)

Last page developed by STEM KI Champions (MACUL) © 2008 Buck Institute for Education 8

(d)

Şekil 3.6. PTÖ Planı (a), PTÖ Müfredatı (b), PTÖ Öğrenme Takvimi (c) ve MEB Yönetmeliğine Uygun Değerlendirme Kriterleri (d)

Tez kapsamında hazırlanan 7 adet PTÖ Değerlendirme Formları (rubrikler) ise aşağıda sıralanmıştır:

1. FORM-1 Öğrenci Sözleşmesi
2. FORM-2 Öğretmen Gözlem Formu
3. FORM-3 Öğrenci Öz Değerlendirme Formu
4. FORM-4 İşbirlikçi Öğrenme Formu
5. FORM-5 Proje Değerlendirme Formu
6. FORM-6 Sunum Değerlendirme Formu
7. FORM-7 Rapor Değerlendirme Formu

FORM-1: Öğrenci Sözleşmesi

Görev Genel Değerlendirmesi
Projenizin başarısından emin olmak için hangi görevi tamamlamalıyınız?

Görev	Görevi tamamlayacak öğrenci(ler).	Son Tarih	Notlar

Öğrenci Anlaşmanı

Benim projem özel, gerçekçi ve ölçülebilir. Grubumun hedefimizin etkinliğini nasıl ölçebileceğini biliyorum. Bu belgede özetlenmiş olan görevin dağıtımını anlıyorum ve kabul ediyorum. Her hangi bir değişimi öğretmenime bildireceğim.

Tarih:
Adı Soyadı:
İmza:

Şekil 3.7. FORM-1 Öğrenci Sözleşmesi

FORM-2: Öğretmen Gözlem Formu**Haftalık Grup Gözlemleme Kontrol Listesi****Grup Adı:****Tarih:**

Grubu beş veya on dakika gözlemleyin. Grup üyelerinin katılımını en iyi ifade eden kutuları işaretleyin.	Üyelerin Hepsisi	Üyelerin Çoğu	Üyelerin Bazıları	Üyelerin Hiçbiri	Üyeler Hakkında Notlar (Olumsuz Durumlar Üyelerin adları yazılarak değerlendirilir)
Yeni bir göreve başlandığında, grup üyeleri:					
Gruba her gün katılır.	☐	☐	☐	☐	
Haftalık planını verip Kabul eder	☐	☐	☐	☐	
Hemen çalışmaya başlar	☐	☐	☐	☐	
Proje materyallerini çıkarır	☐	☐	☐	☐	
Öğretmenin yardımı olmadan bir şeylerin çaresine bakar	☐	☐	☐	☐	
Sorumlulukları paylaşır	☐	☐	☐	☐	
Ana kaynaklara başvurur	☐	☐	☐	☐	
Not alır	☐	☐	☐	☐	
Konuyla alakalı sohbetler yapar	☐	☐	☐	☐	
Yeni bilginin önemini değerlendirir	☐	☐	☐	☐	
Aydınlatıcı soruları sorar	☐	☐	☐	☐	
Birbirine konuşma fırsatı tanır	☐	☐	☐	☐	
Etkili kararlar verir	☐	☐	☐	☐	
Planları ve kararları kaydeder	☐	☐	☐	☐	
Temel bilgileri paylaşır	☐	☐	☐	☐	
Göreve bağlı kalır	☐	☐	☐	☐	

Değerlendirmeler:

Şekil 3.8. FORM-2 Öğretmen Gözlem Formu

FORM-3: Öğrenci Öz Değerlendirme Formu

Proje:

Grup Adı:

Tarih:

Öğrenci Adı Soyadı:

Ortak Çalışmaya Dayalı Öz/Akran Değerlendirme Formu

Talimatlar:

Lütfen grubunuzdaki katılımlarınıza dair fikirlerinizi paylaşın. Her bir kategori için, bir ekip lideri olarak etkililiğinizi en iyi yansıtan açıklamayı daire içine alın. Ardından arkadaki soruları yanıtlayın.



KATEGORI	4	3	2	1
Katkılar	Gruba ve sınıf tartışmasına katılım gösterirken <u>sürekli olarak</u> faydalı fikirler sundum. Oldukça çaba gösterip katkıda bulunan tam anlamıyla bir liderdim.	Gruba ve sınıf tartışmasına katılım gösterirken <u>genellikle</u> faydalı fikirler sundum. Oldukça çaba gösteren güçlü bir liderdim.	Gruba ve sınıf tartışmasına katılım gösterirken <u>bazen</u> faydalı fikirler sundum. Gerekeni yapan, yeterli olan bir liderdim.	Gruba ve sınıf tartışmasına katılım gösterirken <u>nadiren</u> faydalı fikirler sundum. Bazen katılım göstermeyi reddettim.
Diğerleriyle birlikte çalışma	Diğerlerini daima dinledim, onlarla paylaşımda bulundum ve onların çabalarını hep destekledim. İnsanların birlikte iyi çalışmaya devam etmesi için çaba gösterdim.	Diğerlerini çoğunlukla dinledim, onlarla paylaşımda bulundum ve onların çabalarını destekledim. Grupta "çalkantıya" sebep olmadım.	Diğerlerini sık sık dinledim, onlarla paylaşımda bulundum ve onların çabalarını destekledim, ama bazen iyi bir ekip üyesi olmadığım oldu.	Diğerlerini nadiren dinledim, onlarla nadiren paylaşımda bulundum ve onların çabalarını çok az destekledim. Çoğunlukla <u>pek iyi</u> bir ekip üyesi değildim.
Problem çözme	Aktif olarak sorunlara çözüm aradım ve tavsiyelerde bulundum.	Diğerlerinin sunduğu çözüm önerilerini geliştirdim.	Çözüm sunmadım veya diğerlerininkini geliştirmedim ancak diğerlerinin sunacakları çözüm önerilerini denemeye istekliydim.	Sorunları çözme denemedim veya diğerlerine bunun için yardımcı olmadım. Sorunları çözme başkasına bıraktım.

FORM-3: Öğrenci Öz Değerlendirme Formu

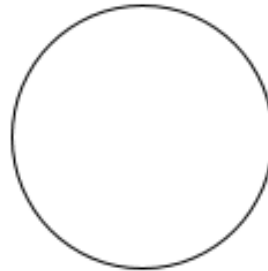
Göreve odaklanma	Göreve ve yapılması gerekene hep odaklandım. Kendimi oldukça iyi yönlendirip yönetebiliyordum.	Çoğu zaman göreve ve yapılması gerekene odaklandım. Diğer grup üyeleri bana güvenebiliyordu.	Göreve ve yapılması gerekene zaman zaman odaklandım. Diğer grup üyelerinin zaman zaman bana odaklanmam gerektiğini hatırlattığı oluyordu.	Göreve ve yapılması gerekenlere nadiren odaklanabildim. İş diğerleri yaptı.
Tavır, yaklaşım	Diğer grup üyelerinin işlerini asla ulu orta eleştirmedim. Görevle ilgili hep olumlu yaklaştım.	Diğer grup üyelerinin işlerini nadiren ulu orta eleştirdim. Görevle ilgili genellikle olumlu yaklaştım.	Diğer grup üyelerinin işlerini bazen ulu orta eleştirdim. Görevle ilgili nispeten olumlu yaklaştım.	Diğer grup üyelerinin işlerini sık sık ulu orta eleştirdim. Görevle ilgili genellikle olumsuz yaklaştım.

1. Görev boyunca üstünüze aldığınız roller ve sorumluluklar neydi? Gruptaki işiniz neydi?

2. Sizin ve grubunuzun iyi yaptığını düşündüğünüz şeyler nelerdir?

3. Siz ve grubunuz daha etkili olarak nasıl çalışabilirdi?

4. Lütfen beraber çalışırken grubunuzun her üyesinin ne ölçüde iş yaptığını temsil etmesi için aşağıdaki daire grafiğini boyayın. Lütfen her bölümde atıfta bulunduğunuz kişinin adını belirtin.



Şekil.3.9. FORM-3 Öğrenci Öz Değerlendirme Formu

İşbirlikçi Öğrenme Değerlendirme Formu : Süreç

Grup:

Tarih:

Grup Üyeleri:

	Mükemmel	Takdire Değer	Kabul Edilebilir	Amatörce
Grup Katılımı	Bütün öğrenciler istekli bir şekilde katılır	En az 4/3 öğrenci aktif olarak katılır	En az öğrencilerin yarısı bir fikir danışır veya sunar;	Sadece bir veya iki kişi aktif olarak katılır
Paylaşılan Sorumluluk	Sorumluluk görev için eşit olarak paylaşılmıştır	Sorumluluk çoğu grup üyesi tarafından paylaşılmıştır	Sorumluluk grup üyelerinin 1/2'si tarafından paylaşılmıştır	Özel olarak tek bir kişiye güvenilmesi
Etkileşimin Kalitesi	Mükemmel dinleme ve liderlik yetenekleri sergilenmiştir; öğrenciler tartışmalarında başkalarının görüş ve fikirlerini yansıtır	Öğrenciler etkileşime adapte olur; heyecanlı bir şekilde görevin ana noktalarını tartışır	Etkileşim için kısıtlı beceri; özenli dinleme; tartışma ve alternatifleri için az sayıda örnek	Az etkileşim; çok kısa sohbetler; bazı öğrenciler kayıtsız veya dikkatsizdi
Grup içindeki Roller	Her öğrenciye açıkça belirlenmiş roller atandı; grup üyeleri roller etkin olarak yerine getirdi	Her öğrenciye bir rol atandı fakat roller açık olarak belirlenmemiştir veya düzenli olarak sadık kalmadı	Öğrencilere roller atandı fakat rollere düzenli olarak sadık kalmadı	Grup üyelerine rol atamak için hiçbir çaba harcanmadı

Şekil 3.10. FORM-4 İşbirlikçi Öğrenme Formu

FORM-5: Proje Değerlendirme Formu

Grup:

Tarih:

Üyeler:



No	Standartlar	Evet	Hayır
1	Grafçet diyagramları doğru hazırlanmış mı?		
2	PLC programı Grafçet diyagramlarına uygun yazılmış mı?		
3	Konveyör üzerinde ürün sensörler ile algılanıyor mu?		
4	Konveyör ile ürün taşınabiliyor mu?		
5	Konveyör üzerinde uzun-kısa ürün ayrımı yapılıyor mu?		
6	Konveyör üzerinde plastik-metal ürün ayrımı yapılıyor mu?		
7	Konveyörler arası ürün taşıma yapılıyor mu?		
8	Konveyör-Döner Tabla arasında ürün taşıma yapılıyor mu?		
9	Döner tabla kontrolü doğru bir şekilde gerçekleştiriliyor mu?		
10	SCADA programı düzgün çalışıyor mu?		

Şekil 3.11. FORM-5 Proje Değerlendirme Formu

FORM-6: Sunum Değerlendirme Formu

POREJE SUNUMU DEĞERLENDİRMESİ

Sunumda, ekip tarafından keşfedilen bilgiler, planlar, taslak çalışmalar, ürün ve projenin birleştirilmiş son durumu, sonuç ve öneriler yer almalıdır.

KATEGORI	4	3	2	1
Sözlü sunum: Hazır olmak	Ekip tamamen hazır ve önceden prova yaptı.	Ekip hazır gözüküyordu ama birkaç provaya daha gereksinimi vardı.	Ekip biraz hazır gibiydi ama yine de prova olmalıydı.	Ekip sunum yapmaya hiç hazır değildi.
Sözlü sunum: İsteklilik	Ekibin jest ve mimikleri ve vücut dili, konuyla çok ilgili ve istekli olduklarını açıkça gösteriyordu.	Ekibin jest ve mimikleri ve vücut dili, genellikle konuyla ilgili ve istekli olduklarını gösteriyordu.	Ekibin jest ve mimikleri ve vücut dili, konuyla çok ilgili ve istekli gözükmeğe çalışıyordu ama pek başarılı değildi.	Çok az jest, mimik ve vücut dili kullanıldı. Ekip açıkça konuya ilgisizdi.
Bütünlük	Sunumda konuyla ilgili gerekebilecek ve rahatlık sağlayacak tüm materyal mevcuttu.	Sunumda konuyla ilgili gerekebilecek ve rahatlık sağlayacak çoğu materyal mevcuttu, bir iki eksik vardı.	Sunumda birden fazla kilit öge eksikti.	Sunumda pek çok eksik ve tutarsızlık vardı.
Arka plan bilgisi	Sunum, keşfedilen bilgileri özetleyen 10 slayt içeriyor. Taslak, çizim, görsel vb. ~~mevcuttu~~ .	Sunum, keşfedilen bilgileri özetleyen 7 tane slayt içeriyor. Taslak, çizim, görsel vb. mevcuttu.	Sunum, keşfedilen bilgileri özetleyen 5 tane slayt içeriyor. Taslak, çizim, görsel vb. mevcuttu.	Sunum, keşfedilen bilgileri özetleyen 1-4 adet slayt içeriyor. Sunumumuzda taslak, çizim, görsel vb. yoktu

Şekil 3.12. FORM-6 Sunum Değerlendirme Formu

Proje Raporu Değerlendirmesi

Proje Raporunda Dikkat Edilmesi Gereken Hususlar:

1. Sayfa yapısı alt, üst ve iki yandan 2 cm boşluklu olacak. Başlıklar 14 punto Times New Roman ve gövde metni Times New Roman 12 punto olacak.
2. Proje Kapağında okul logosu, ders başlığı, proje adı, grup üyeleri ve öğretmen isimleri olacak (Size format verilecektir.)
3. Size verilecek başlıklar proje raporunda olmalıdır. (Sizlere detaylı bilgi verilecektir.)
4. Proje Raporunuz ve Ürün değerlendirmeniz 2. Performans notunuza %30 etki edecektir.

No	Standartlar	Evet	Hayır
1	Rapor kapağı verilen standartlara göre mi yazılmış?		
2	Rapor içeriği istenilen başlıklara göre mi yazılmış?		
3	Rapor içeriği istenilen yazım ve sayfa formatına göre mi yazılmış?		
4	Raporda "İçindekiler" ve "Kaynakça" tabloları var mı?		
5	Raporda kullanılan malzemeler tanıtılmış mı?		
6	Raporda Grafcet diyagramları var mı?		
7	Raporda Grafcet uyumlu PLC yazılımı var mı?		
8	Raporda C# ile SCADA yazılımları var mı?		
9	Raporda resim ve şekillere yer verilmiş mi?		
10	Raporun tertip, temizlik ve düzeni iyi mi?		

Şekil 3.13. FORM-7 Rapor Değerlendirme Formu

3.3. PLC Bağlantıları

Set üzerinde iki adet PLC bulunmaktadır. İki farklı PLC modelinin giriş ve çıkışları (I/O) birbirine paralel bağlanmıştır. PLC seçim şalteri ile PLC seçimi yapılmaktadır (seçilen PLC enerjilenmektedir). Her iki PLC modelinin de iki adet giriş, iki adet çıkış ünitesi mevcuttur. NPN ve PNP bağlantıları kolaylıkla giriş ünitelerine yapılabilmektedir. Ayrıca her iki PLC de ethernet kullanmaktadır. Böylece sınıf ortamında her öğrenci ağ üzerinden sete program atarak set üzerinde çalışabilmektedir. Buton takımı, sensörler ve döner tabla motoru sürücü çıkışları giriş ünitelerine, motorlar, pnömotik valfler ve lambalar çıkış ünitelerine bağlanmıştır. Aşağıda her iki PLC için bağlantı ve sembol tabloları görülmektedir:

QX40 INPUT MODÜLÜ		QX80 INPUT MODÜLÜ	
X0	Start	X10	Kapasitif (PNP)
X1	Auto / Manual	X11	Endüktif (PNP)
X2	Stop (Normally Closed)	X12	In position
X3	b0 (Dikey silindir yukarıda)	X13	Alarm
X4	b1(Dikey silindir aşağıda)	X14	
X5	Yatay Silindir ortada (a_1)	X15	
X6	Yatay Silindir DonerTabla (a_2)	X16	
X7	Yatay Silindir (PLC Tarafı) (a_0)	X17	
X8	Konveyör1-nihayet	X18	
X9	Konveyör2- bidayet	X19	
XA	Uzun parça(Optik)	X1A	
XB	Kısa parça(Optik)	X1B	
XC	Parça _var (optik-konveyör 1)	X1C	
XD	Konveyör2-nihayet Optik	X1D	
XE	Döner Tabla _Parça Var	X1E	
XF	Döner Tabla Home Position	X1F	
Ortak uç : +24		(Ortak Uç 0v)	

Tablo 3.1: PLC-1 Giriş Modüllerinin Bağlantıları

QY50 OUTPUT MODÜLÜ		QY80 OUTPU MODÜLÜ	
Y20	Konveyör 1_Besleme	Y30	Step Servo On
Y21	Konveyör 1_Yön	Y31	Step START
Y22	Konveyör 2_Silindir	Y32	Step STOP
Y23	Konveyör 2_Silindir_Zıt	Y33	Bozuk Çıkış
Y24	Cl (Kırmızı Lamba)	Y34	Home
Y25	Bl (Sarı Lamba)	Y35	Alarm Reset
Y26	Al (Yeşil)	Y36	Step0
Y27	A_1 (X Eksen Döner Tabla))	Y37	Step1
Y28	A_0 (X Ekseni Konveyör)	Y38	Step2
Y29	Vakum	Y39	Step3
Y2A	Dikey Silindir (B)	Y3A	Mdo/Hmd0
Y2B		Y3B	Md1/Hmd1
Y2C		Y3C	Servo On Lambası
Y2D		Y3D	Lamba1(In Position)
Y2E		Y3E	Lamba2 (Alarm)
Y2F		Y3F	Lamba3
Ortak uç :0V		Ortak uç : +24V	

Tablo 3.2: PLC-1 Çıkış Modüllerinin Bağlantıları

ID211-1		ID211-2	
0.0	Start	1.0	Kapasitif (PNP)
0.1	Auto / Manual	1.1	Endüktif (PNP)
0.2	Stop (Normally Closed)	1.2	
0.3	b0 (Dikey silindir yukarıda)	1.3	
0.4	b1(Dikey silindir aşağıda)	1.4	
0.5	Yatay Silindir solda (a_2)	1.5	
0.6	Yatay Silindir ortada (a_1)	1.6	
0.7	Yatay Silindir sağda (PLC Tarafı) (a_0)	1.7	
0.8	Konveyör1-nihayet	1.8	
0.9	Konveyör2-nihayet	1.9	
0.10	Uzun parça(Optik)	1.10	
0.11	Kısa parça(Optik)	1.11	
0.12	parça _var (optik-konveyör 1)	1.12	
0.13	Konveyör2-bidayet Optik	1.13	
0.14	Döner Tabla Parça Var	1.14	
0.15	Döner Tabla Home Position	1.15	
Ortak uç : +24		Ortak uç : 0V	

Tablo 3.3: PLC-2 Giriş Modüllerinin Bağlantıları

OC211-1		OC211-2	
2.0	Konveyör 1_Besleme	3.0	Servo ON
2.1	Konveyör 1_Yön	3.1	Start
2.2	Konveyör 1_Silindir	3.2	Stop
2.3	Konveyör 1_Silindir ters	3.3	Boş
2.4	Cl (Kırmızı Lamba)	3.4	Home
2.5	Bl (Sarı Lamba)	3.5	Alarm Reset
2.6	Al (Yeşil)	3.6	Step0
2.7	A_1 (X Eksen Döner Tabla))	3.7	Step1
2.8	A_0 (X Ekseni Konveyör)	3.8	Step2
2.9	Vakum	3.9	Step3
2.10	Dikey Silindir (B)	3.10	Mdo/Hmd0
2.11		3.11	Md1/Hmd1
2.12		3.12	Servo On Lambası
2.13		3.13	Lamba1
2.14		3.14	Lamba2
2.15		3.15	Lamba3
Ortak uç : Ortak uç :0V		Ortak Uç : +24V	

Tablo 3.4: PLC-2 Çıkış Modüllerinin Bağlantıları

M0	Step0	M44	Step44	M200	SCADA_Start_(X0)	X0	PLC_Start
M1	Step1	M45	Step45	M202	SCADA_Stop_(X2)	X1	PLC_Auto_Man
M2	Step2	M46	Step46			X2	PLC_Stop
M3	Step3	M47	Step47	M300	SCADA_Konveyör-1_İLERİ	X3	b_0
M4	Step4	M48	Step48	M301	SCADA_Konveyör-1_GERİ	X4	b_1
M5	Step5	M49	Step49	M302	SCADA_Konveyör-2_GERİ	X5	a_1
M6	Step6	M50	Step50	M303	SCADA_Konveyör-2_İLERİ	X6	a_2
M7	Step7	M51	Step51	M304	SCADA_Silindir_İLERİ	X7	a_0
M8	Step8	M52	Step52	M305	SCADA_Silindir_GERİ	X8	kon1_nih
M9	Step9	M53	Step53	M306	SCADA_Vakum Silindiri_AŞAĞI	X9	kon2_bid
M10	Step10	M54	Step54	M307	SCADA_Vakum Silindiri_AŞAĞI	X0A	uzun_par_Sens
M11	Step11	M55	Step55	M308	SCADA_Vakum_ÇEK	X0B	kısa_par_Sens
M12	Step12	M56	Step56	M309	SCADA_Vakum_BIRAK	X0C	kon1_bid
M13	Step13	M57	Step57			X0D	kon2_nih
M14	Step14	M58	Step58	M1533	uzun	X0E	donTab_parca_var
M15	Step15	M59	Step59	M1534	kısa	X0F	donTab_org
M16	Step16	M60	Step60	M1535	metal	X10	kapasitif_Sens
M17	Step17	M61	Step61	M1536	derlin	X11	enduktif_Sens
M18	Step18	M62	Step62			X12	in_position
M19	Step19	M63	Step63	M1541	tabla_dolu		
M20	Step20	M64	Step64	M1542	parca_onay	Y20	Kon1_MotorAktif_Yon_1
M21	Step21	M65	Step65			Y21	Kon1_Direction_Yon_0
M22	Step22	M66	Step66	M1546	uzun_parca	Y22	Kon2_1
M23	Step23	M67	Step67	M1547	kısa_parca	Y23	Kon2_0
M24	Step24	M68	Step68	M1548	metal_parca	Y24	KL
M25	Step25	M69	Step69	M1549	derlin-parca	Y25	SL
M26	Step26	M70	Step70			Y26	YL
M27	Step27	M71	Step71	M1551	Touch_kısa_derlin	Y27	A_1
M28	Step28	M72	Step72	M1552	Touch_uzun_derlin	Y28	A_0
M29	Step29	M73	Step73	M1553	Touch_kısa_metal	Y29	Vakum
M30	Step30	M74	Step74	M1554	Touch_uzun_metal	Y2A	B_1
M31	Step31	M75	Step75	M1555	Touch_kısa		
M32	Step32	M76	Step76	M1556	Touch_uzun	Y30	StepMo Servo
M33	Step33	M77	Step77	M1557	Touch_metal	Y31	StepMo Start
M34	Step34	M78	Step78	M1558	Touch_derlin	Y32	StepMo Stop
M35	Step35	M79	Step79			Y33	Boş-Bozuk
M36	Step36	M80	Step80			Y34	StepMo Home
M37	Step37	M81	Step81			Y35	StepMo Alarm_Res
M38	Step38	M82	Step82			Y36	KonumData0
M39	Step39	M83	Step83			Y37	KonumData1
M40	Step40	M84	Step84			Y38	KonumData2
M41	Step41	M85	Step85			Y39	KonumData3
M42	Step42	M86	Step86			Y3A	MDO/HMD0
M43	Step43					Y3B	MD1/HMD1
						Y3C	ServoON_Lamba
						Y3D	Lamba1
						Y3E	Lamba2
						Y3F	Lamba3

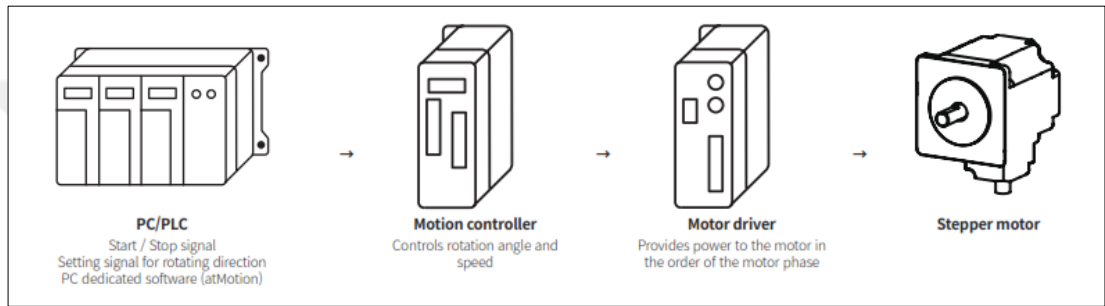
Tablo 3.5: PLC-1 Sembol Tablosu

Adim0	W0.00	Adim44	W2.12	uzun	W100.00	kapasitif_Sens	1.00
Adim1	W0.01	Adim45	W2.13	kisa	W100.01	enduktif_Sens	1.01
Adim2	W0.02	Adim46	W2.14	metal	W100.02	in_position	1.02
Adim3	W0.03	Adim47	W2.15	derlin	W100.03	alarm	1.03
Adim4	W0.04	Adim48	W3.00	tabla_dolu	W101.00	Kon1_MotorAktif_Yon_1	2.00
Adim5	W0.05	Adim49	W3.01	parca_onay	W101.01	Kon1_Direction_Yon_0	2.01
Adim6	W0.06	Adim50	W3.02	uzun_parca	W102.00	Kon2_1	2.02
Adim7	W0.07	Adim51	W3.03	kisa_parca	W102.01	Kon2_0	2.03
Adim8	W0.08	Adim52	W3.04	metal_parca	W102.02	KL	2.04
Adim9	W0.09	Adim53	W3.05	derlin_parca	W102.03	SL	2.05
Adim10	W0.10	Adim54	W3.06	Touch_kisa_derlin	W103.01	YL	2.06
Adim11	W0.11	Adim55	W3.07	Touch_uzun_derlin	W103.04	A_1	2.07
Adim12	W0.12	Adim56	W3.08	Touch_kisa_met	W103.07	A_0	2.08
Adim13	W0.13	Adim57	W3.09	Touch_uzun_met	W103.10	Vakum	2.09
Adim14	W0.14	Adim58	W3.10	Touch_kisa	W103.11	B_1	2.10
Adim15	W0.15	Adim59	W3.11	Touch_uzun	W103.12	StepMo_Servo	3.00
Adim16	W1.00	Adim60	W3.12	Touch_metal	W103.13	StepMo_Start	3.01
Adim17	W1.01	Adim61	W3.13	Touch_derlin	W103.14	Stop_Step	3.02
Adim18	W1.02	Adim62	W3.14	SCADA_Konveyor_1_ILERI	W104.00	Bos_Bozuk	3.03
Adim19	W1.03	Adim63	W3.15	SCADA_Konveyor_1_GERİ	W104.01	StepMo_Home	3.04
Adim20	W1.04	Adim64	W4.00	SCADA_Konveyor_2_GERİ	W104.02	StepMo_Alarm_Res	3.05
Adim21	W1.05	Adim65	W4.01	SCADA_Konveyor_2_ILERI	W104.03	KonumData0	3.06
Adim22	W1.06	Adim66	W4.02	SCADA_Silindir_ILERI	W104.04	KonumData1	3.07
Adim23	W1.07	Adim67	W4.03	SCADA_Silindir_GERİ	W104.05	KonumData2	3.08
Adim24	W1.08	Adim68	W4.04	SCADA_VakumSilindiri_ASA GI	W104.06	KonumData3	3.09
Adim25	W1.09	Adim69	W4.05	SCADA_Vakum_CEK	W104.08	MDO_HMD0	3.10
Adim26	W1.10	Adim70	W4.06	SCADA_Vakum_BIRAK	W104.09	MD1_HMD1	3.11
Adim27	W1.11	Adim71	W4.07	SCADA_Stop_X2	W104.10	ServoON_Lamba	3.12
Adim28	W1.12	Adim72	W4.08	SCADA_Start_X0	W104.11	Lamba1	3.13
Adim29	W1.13	Adim73	W4.09	PLC_Start	0.00	Lamba2	3.14
Adim30	W1.14	Adim74	W4.10	PLCAut_Man	0.01	Lamba3	3.15
Adim31	W1.15	Adim75	W4.11	PLC_Stop	0.02		
Adim32	W2.00	Adim76	W4.12	b_0	0.03		
Adim33	W2.01	Adim77	W4.13	b_1	0.04		
Adim34	W2.02	Adim78	W4.14	a_1	0.05		
Adim35	W2.03	Adim79	W4.15	a_2	0.06		
Adim36	W2.04	Adim80	W5.00	a_0	0.07		
Adim37	W2.05	Adim81	W5.01	kon1_nih	0.08		
Adim38	W2.06	Adim82	W5.02	kon2_bid	0.09		
Adim39	W2.07	Adim83	W5.03	uzun_par_Sens	0.10		
Adim40	W2.08	Adim84	W5.04	kisa_par_Sens	0.11		
Adim41	W2.09	Adim85	W5.05	kon1_bid	0.12		
Adim42	W2.10	Adim86	W5.06	kon2_nih	0.13		
Adim43	W2.11			donTab_parca	0.14		
				donTab_org	0.15		

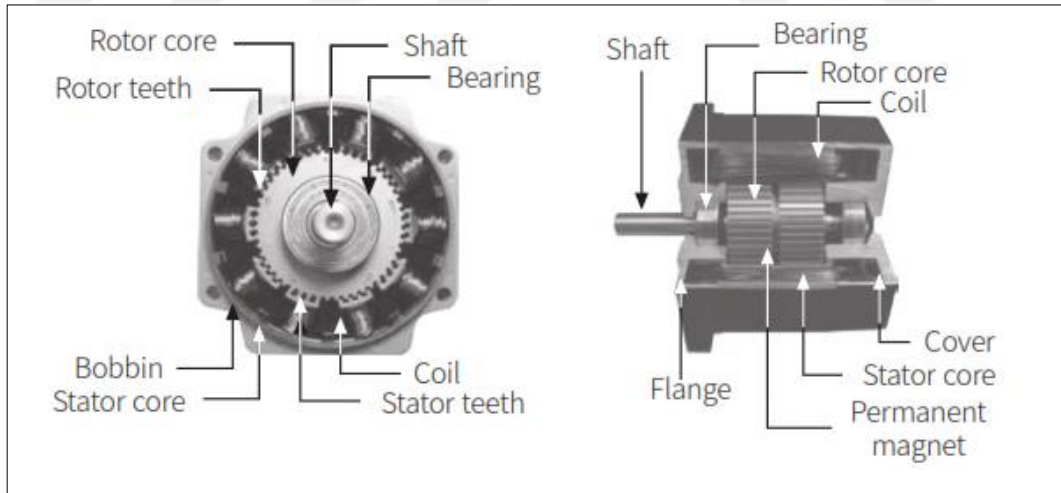
Tablo 3.6: PLC-2 Sembol Tablosu

3.4. Döner Tabla Süren Step Motoru Özellikleri

Set üzerindeki alüminyum döner tabla Autonics marka step motor ile kontrol etmektedir. Step motorun özellikleri ve fiziksel hesaplamaları motorun hareket el kitapçığında belirtilmiştir. Aşağıda bu kitapçıktan alıntılar görülmektedir. Şekil 3.14'te adım motorunun PLC ile bağlantısı görülmektedir. PLC ile adım motoru arasında motor sürücü ve hareket kontrol ünitesi kullanılmıştır. Şekil 3.15'te sette kullanılan adım motorunun içyapısı görülmektedir. Rotor (dönen kısım), sargılar, şaft ve stator (sabit kısım) gibi motor kısımlarının önden ve yandan görünüşü verilmiştir.



Şekil 3.14: Step Motor ile PLC Arasındaki Bağlantı Konfigürasyonu



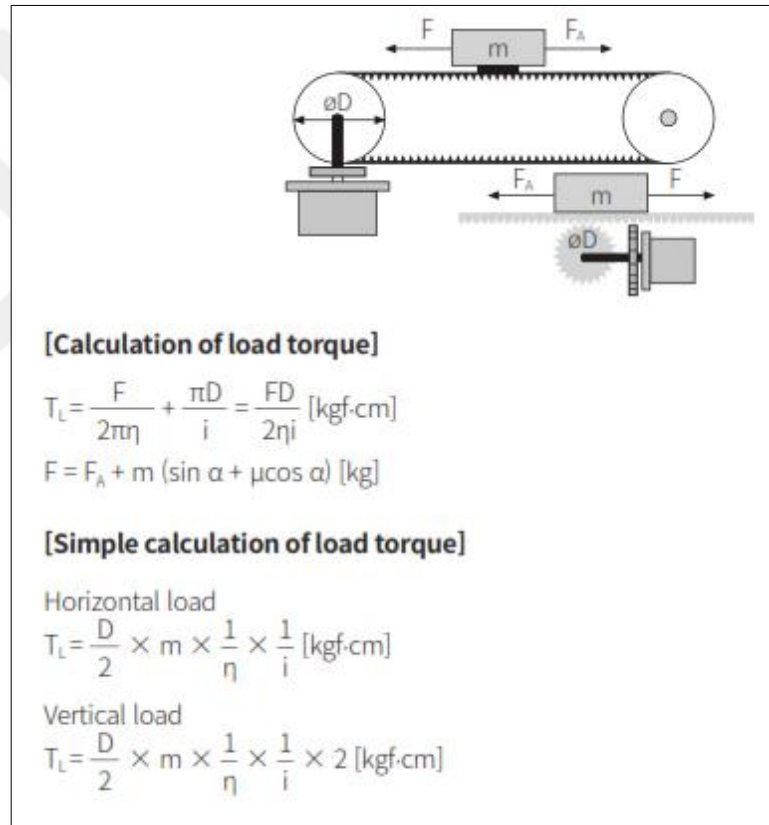
Şekil 3.15. Step Motor İç Yapısı

Şekil 3.16'da adım motorunun açı ve açısal hız karakteristikleri verilmiştir. Dönme açısı, adım açısı ile uygulanan pals sayısının çarpımıdır. Dakikadaki dönüş hızı ise (rpm: rotate per minute), pals hızı frekansının adım açısına bölümünün 60 ile çarpılmasıyla bulunur.



Şekil 3.16. Step Motorun Açık ve Açısal Hız Karakteristik Sinyalleri

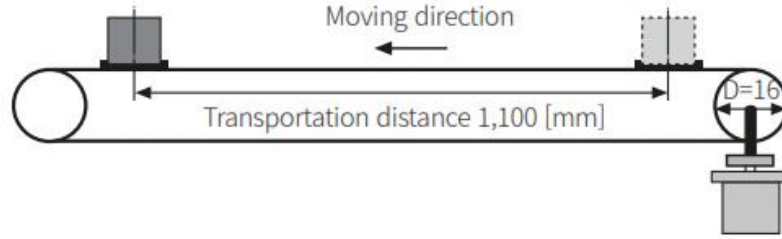
Şekil 3.17’de adım motorunun kayış kasnak sisteminde yük torku hesabı verilmiştir. Kayış kasnak sisteminde yatay ve dikey tork hesabı verilmiştir. Eğitim setinde kayış kasnak sistemi yatay olarak kullanılmıştır. Kayış kasnakta kullanılan dişli çapı ile doğru orantılıdır.



Şekil 3.17: Kayış Kasnak Sistemi Tork Hesabı

Şekil 3.18’de adım motoru için gerekli pals sayısı hesabı verilmiştir. Örnek şekilde 0,72 derece adım açısına sahip, 50mm çapında dişlisi bulunan bir sistemde, 1100 mm yol gitmek için gereken pals sayısı = $(1100/50) \times (360/0,72) = 11000$ pals olarak bulunur.

■ Driving the timing belt

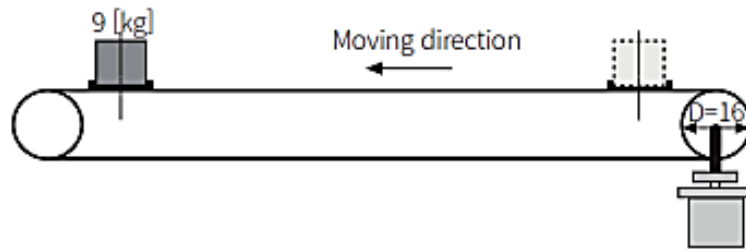


When carrying an object as following figure for 1 sec. by using 5-phase stepper motor (0.72°/step), the moving distance/revolution is approx. 50[mm] by $2\pi r$ as the circumference of the pulley. As the moving distance/revolution is 50[mm] the number of the necessary pulse is calculated as follows.

$$\text{Necessary pulses} = \frac{1,100}{50} \times \frac{360^\circ}{0.72^\circ} = 11,000 \text{ [Pulse]}$$

Şekil 3.18: Step Motor İçin Gerekli Pals Sinyali Hesabı

Şekil 3.19’da görülen örnek hesaplamada, 9kg bir yük için 16mm dişli çapı olan bir kayış kasnak sisteminde %90 verimde elde edilen tork 8kgf-cm olarak hesaplanmıştır.



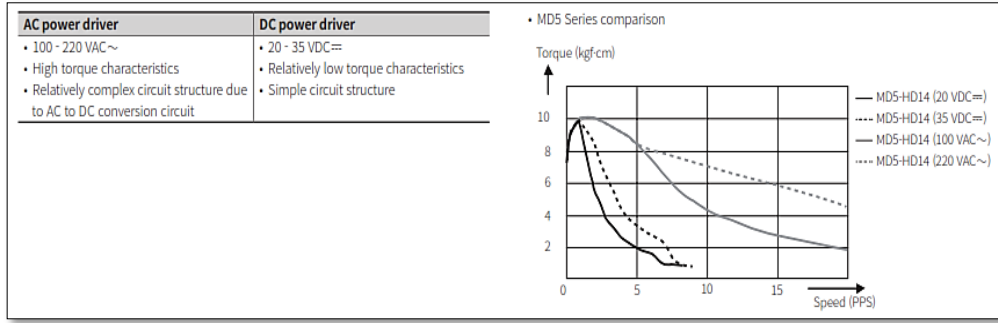
When carrying an object by using a timing belt with 90 [%] of efficiency, 16 [mm] diameter of pulley and 9 [kg] of the load weight as following figure, the load torque is calculated as follows.

$$T_L = \frac{D}{2} \times m \times \frac{1}{\eta} \times \frac{1}{i} \text{ [kgf-cm]}$$

$$T_L = \frac{1.6 \text{ [cm]}}{2} \times 9 \text{ [kg]} \times \frac{1}{0.9} \times \frac{1}{1} = 8 \text{ [kgf-cm]}$$

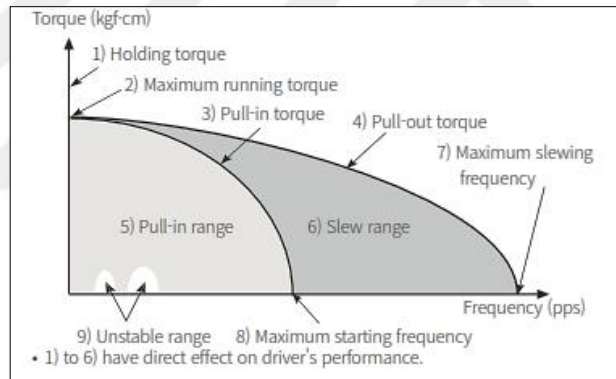
Şekil 3.19: Step Motor İçin Örnek Tork Hesabı

Şekil 3.20’de step motorun AC ve DC akınla sürülmesinde tork-hız eğrileri verilmiştir. Görüldüğü üzere hız arttıkça tork değeri düşmektedir. Diğer bir ifade ile en yüksek tork en düşük hızda elde edilmektedir. Verim AC akımda daha iyidir.

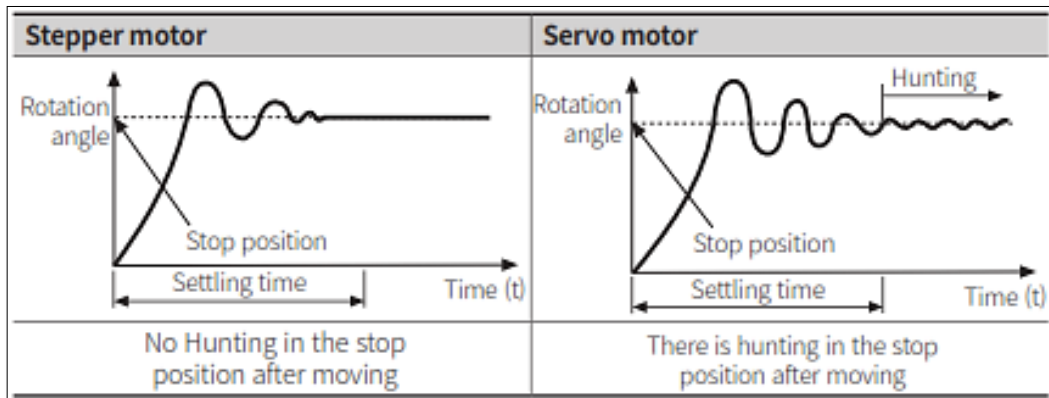


Şekil 3.20: AC ve DC Step Motor Sürücüsü Tork-Hız Eğrileri

Şekil 3.21’de tork-frekans performans eğrisi verilmiştir. Çalışma frekansı arttıkça motorun torku düşmektedir. Şekil 3.22’de ise step ve servo motorun hareket sonrası durma davranışları verilmiştir. Step motor daha kararlı (vibrasyonsuz) bir durma davranışı sergilemektedir. Servo motorda ise az da olsa durma anında bir vibrasyon olmaktadır.



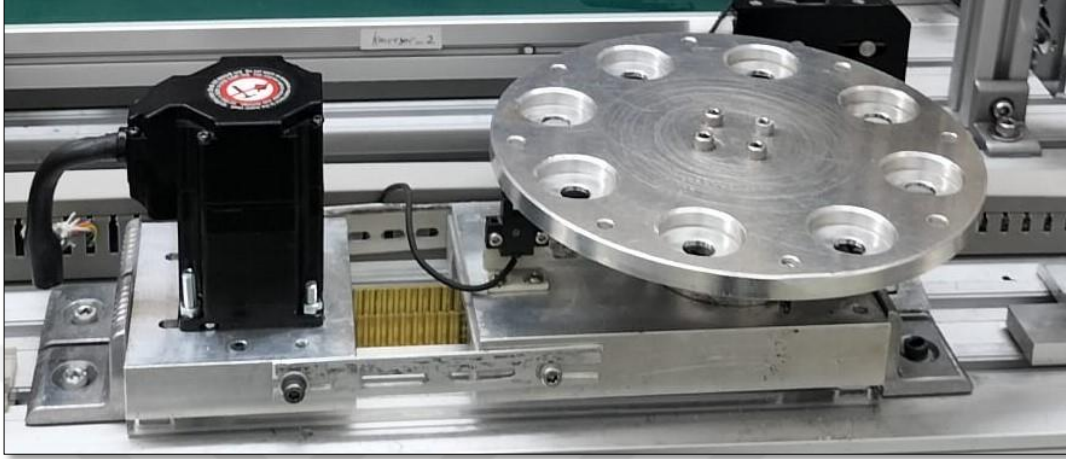
Şekil 3.21: Step Motorun Tork-Frekans Sürücü Performansı



Şekil 3.22: Step Motor ile Servo Motorun Hareket Sonrası Durma Davranışları

3.5. Döner Tabla Süren Step Motor Sürücüsü Özellikleri

Set üzerinde Autonics AiC-D sürücü modeli ile döner tabla motoru sürülmektedir. Motor miline bağlı kayış kasnak sistemi ile de döner tabla hareket ettirilmektedir.



Şekil 3.23: Step Motor, Kayış Kasnak Sistemi ve Döner Tabla

Aşağıda Autonics firmasının ürettiği AiC-D Serisi step motor sürücüsü ve sürücünün kullanım kitapçığından alıntılar görülmektedir:

■ Features

- Motor driver and controller integral type
- Competitive price compared to the servo motor and closed-loop function and fast response for short-distance continuous drive
- Controllable maximum 31 axis with RS485 communication
- Realizing a wide variety of operation up to 256 steps using 14 control commands combination
- 4 type of operation mode: jog mode, continuous mode, index mode, program mode
- Improved user convenience with providing 50 I/O pins
- C language library provided (32-bit, 64-bit)
- Dedicated Windows program (atMotion) provided
- Responding rapidly and maintaining torque in stop without hunting
- Easy to use without tuning (various gain settings via programming)
- Applicable to the precision equipment such as optical inspection equipment with the features of maintaining torque in stop and having no micro vibration (hunting)
- Containing various resolutions (electric gear)
 - : 500, 1000, 1600, 2000, 3200, 3600, 5000, 6400, 7200, 10000 (10-level)
- Various alarm functions
 - : 17 alarms; over current, over speed, over heat, motor connection error, encoder connection error
- Frame size 42mm, 56mm, 60mm supported

NEW

Şekil 3.24. Autonics AiC-D Serisi Step Motor Sürücüsü ve Özellikleri

Deney setinde sürücü ayarlarını PC üzerinden yapmak için RS485-RS232 haberleşme aparatı kullanılmıştır. Bağlantı kurulduktan sonra motorun bağlantılarının (Şekil 3.26) ve ayarlarının yapıldığı atMotion programı Şekil 3.25'te görülmektedir.

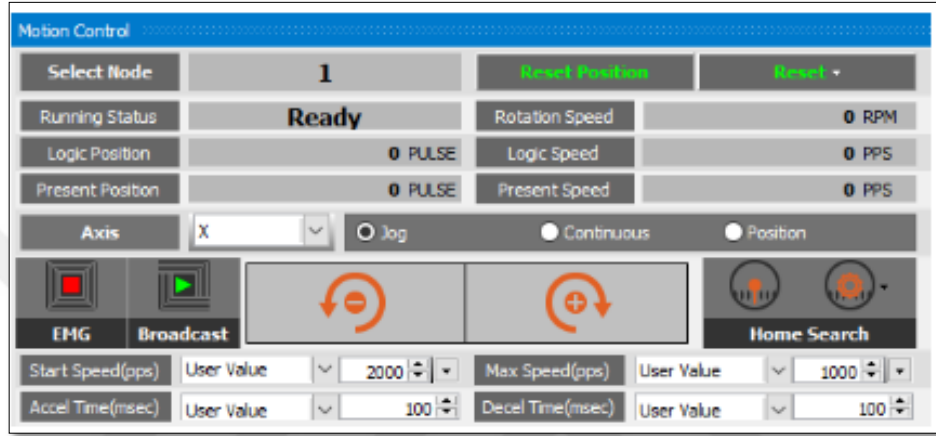
■ Software (atMotion)

- atMotion is a comprehensive motion device management program that can be used with Autronics motion controller (AiC-D, PMC-2HSP/2HSN, PMC-1HS/2HS, PMC-4B-PCI).
- atMotion provides GUI control for easy and convenient parameter setting and monitoring data management of multiple devices.
- Visit our website (www.autronics.com) to download the user manual and software.

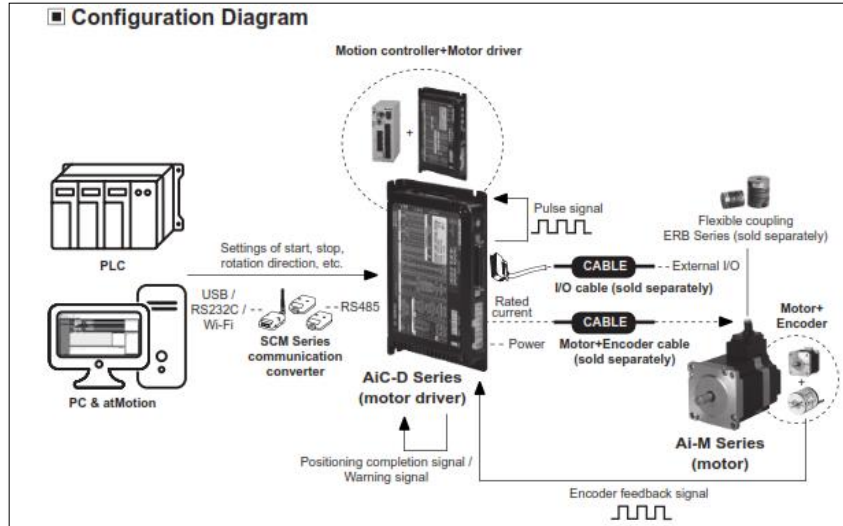
< Computer specification for using software >

Item	Minimum requirements
System	IBM PC compatible computer with Intel Pentium III or above
Operations	Microsoft Windows 98/NT/XP/Vista/7/8/10
Memory	256MB+
Hard disk	1GB+ of available hard disk space
VGA	Resolution: 1024×768 or higher
Others	RS-232 serial port (9-pin), USB port

< atMotion screen >



Şekil 3.25. Autronics AiC-D Serisi Step Motor Sürücüsü Yazılımı (atMotion)

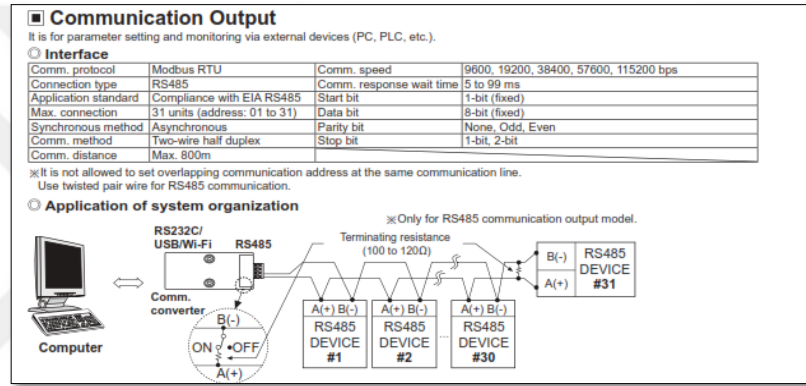


Şekil 3.26. Autronics AiC-D Step Motor Sürücüsü Konfigürasyon Diyagramı

atMotion programı ile döner tabla için en uygun açısız hız ve konumlandırma değeri denemeler sonucunda atanmıştır. Şekil 3.27’de Kontrol için motor giriş-çıkış (I/O) uçları verilmiş, Şekil 3.28’de ise haberleşme uçları görülmektedir.

Control Input/Output					
Inner signal of all input/output consists of photocoupler. ON: photocoupler power ON OFF: photocoupler power OFF					
Input					
1. Exclusive input (20)					
Signal name	Descriptions	Pin no.	Signal name	Descriptions	Pin no.
Reset	Reset command	3	MD0/HMD0	Operation mode designate 0 / Home search mode designate 0	13
Start	Drive start command	4	MD1/HMD1	Operation mode designate 1 / Home search mode designate 1	14
Stop	Drive stop command	5	Pause	Pause	15
EMG	Drive emergency stop command	6	Servo On/Off	Servo On/Off	16
Step0/+Run/+Jog	Step designate 0 / +Run / +Jog	7	Home	Home search	17
Step1/-Run/-Jog	Step designate 1 / -Run / -Jog	8	Alarm Reset	Alarm reset command	18
Step2/SSP0	Step designate 2 / Start speed designate 0	9	+Limit	+direction limit sensor	19
Step3/SSP1	Step designate 3 / Start speed designate 1	10	-Limit	-direction limit sensor	20
Step4/MSP0	Step designate 4 / Max. speed designate 0	11	ORG	Home sensor	21
Step5/MSP1	Step designate 5 / Max. speed designate 1	12	SD	Deceleration (deceleration stop) signal	22

Şekil 3.27. Autonics AiC-D Step Motor Sürücüsü Kontrol I/O Uçları



Şekil 3.28. Autonics AiC-D Step Motor Sürücüsü Haberleşme Çıkışları

Deney setinde motorun hangi uçlarının kullanıldığı PLC giriş-çıkış bağlantı tablolarında görülmektedir. Tablo 3.1, Tablo3.2, Tablo 3.3 ve Tablo 3.4 dikkatlice incelendiğinde Servo ON, Home, In position, Step 1, Alarm gibi uçların kullanıldığı görülür. Motorun hareketi için önce servo ON olması gereklidir. Ardından tabla altındaki sensör yardımıyla in position konumuna kadar motor döndürülür. Sonrasında kullanıcının atMotion programında belirlediği Home konumuna motor götürülür. Ardından kullanıcının at Motion programında belirlediği diğer adımlara (Step) motor götürülür. Tabla 8 ürün taşıma kapasitesinde olduğu için ilk ürün alma noktası Home pozisyonu, diğer 7 adım noktası ise sırayla ürün almak için konumlanma noktalarıdır. Yani tablaya ürün geldiğinde tabla diğer adım pozisyonuna konumlandırılır. Böylelikle diğer ürünü almaya hazır hale gelir. 7 adım sonra bir turu tamamlamış ve 8 ürün ile tabla dolmuş olur. In position, Home ve Alarm çıkışları buton panelindeki lambalara bağlanmıştır.

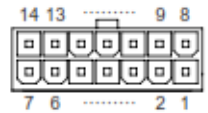
■ Connection Connectors of Driver

○ Connector function

● CN1: Power connector

Pin arrangement	Pin no.	Function
	2	GND
	1	24VDC

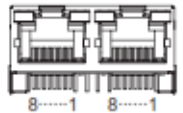
● CN2: Motor+Encoder connector

Pin arrangement	Pin no.	Function	Pin no.	Function
	1	GND	8	+5VDC
	2	Encoder A	9	Encoder A
	3	Encoder B	10	Encoder B
	4	Encoder Z	11	Encoder Z
	5	F.G.	12	N-C
	6	Motor A	13	Motor B
	7	Motor A	14	Motor B

● CN3: I/O connector

Pin arrangement	Pin no.	I/O	Function	Pin no.	I/O	Function
	1	—	N-C	26	Input	IN0
	2	—	N-C	27	Input	IN1
	3	Input	Reset	28	Input	IN2
	4	Input	Start	29	—	N-C
	5	Input	Stop	30	Input	IN3
	6	Input	EMG	31	Input	IN4
	7	Input	Step0/+Run/+Jog	32	Input	IN5
	8	Input	Step1/-Run/-Jog	33	Input	IN6
	9	Input	Step2/SSP0	34	Input	IN7
	10	Input	Step3/SSP1	35	Input	IN8
	11	Input	Step4/MSP0	36	Input	VEX
	12	Input	Step5/MSP1	37	Input	GEX
	13	Input	MD0/HMD0	38	Output	Alarm
	14	Input	MD1/HMD1	39	Output	Compare1 (Trigger)
	15	Input	Pause	40	Output	Compare2 (Trigger)
	16	Input	Servo On/Off	41	Output	OUT0
	17	Input	Home	42	Output	OUT1
	18	Input	Alarm Reset	43	Output	OUT2
	19	Input	+Limit	44	Output	OUT3
	20	Input	-Limit	45	Output	OUT4
	21	Input	ORG	46	Output	OUT5
	22	Input	SD	47	Output	OUT6
	23	Output	In-Position	48	Output	OUT7
	24	Input	VEX	49	Output	OUT8
	25	Input	GEX	50	Output	OUT9

● CN4: RS485 communication cable connector

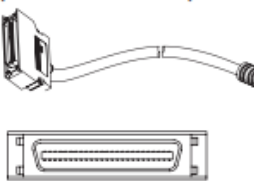
Pin arrangement	Pin no.	I/O	Function	Pin no.	I/O	Function
	1	—	N-C	5	—	N-C
	2	—	N-C	6	Input/Output	RS485 DATA-
	3	Input/Output	RS485 DATA+	7	—	N-C
	4	—	N-C	8	—	N-C

Şekil 3.29. Autonics AiC-D Serisi Step Motor Sürücüsü Konnektör Uçları

Şekil 3.29’da motor konnektör uçları verilmiştir. Deney setinde kullanılan uçlar ile PLC bağlantıları yapılmıştır.

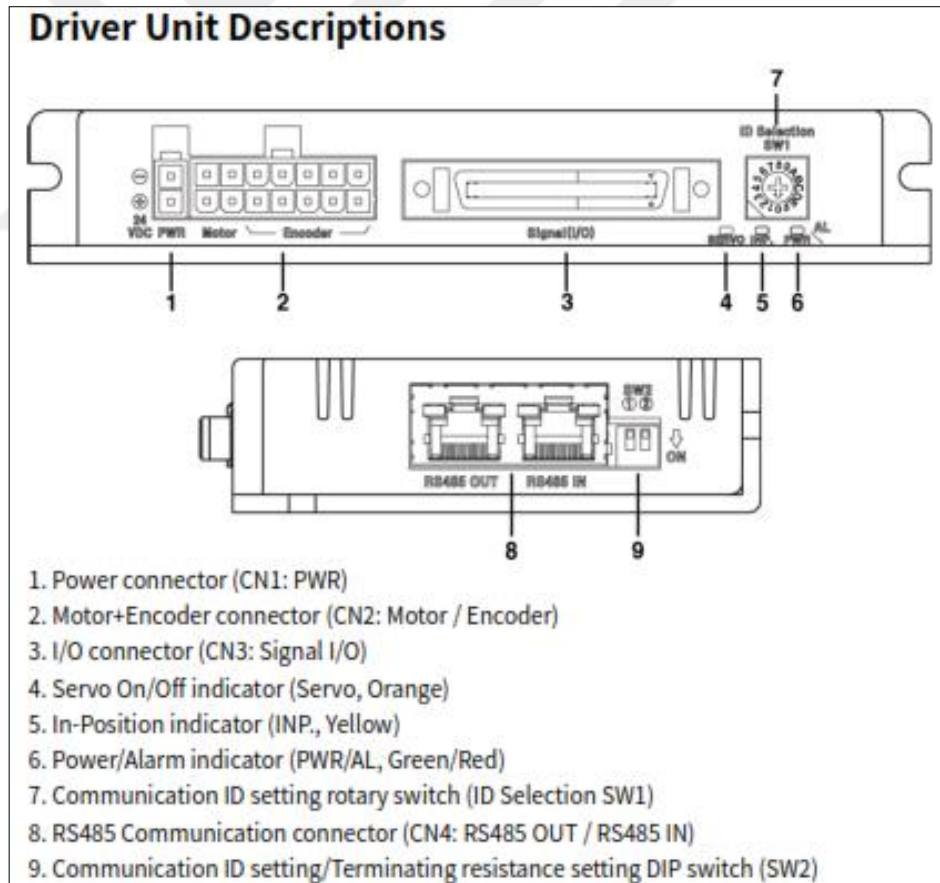
○ I/O cable

● CJ-MP50-HP□
(standard: AiC TAG)



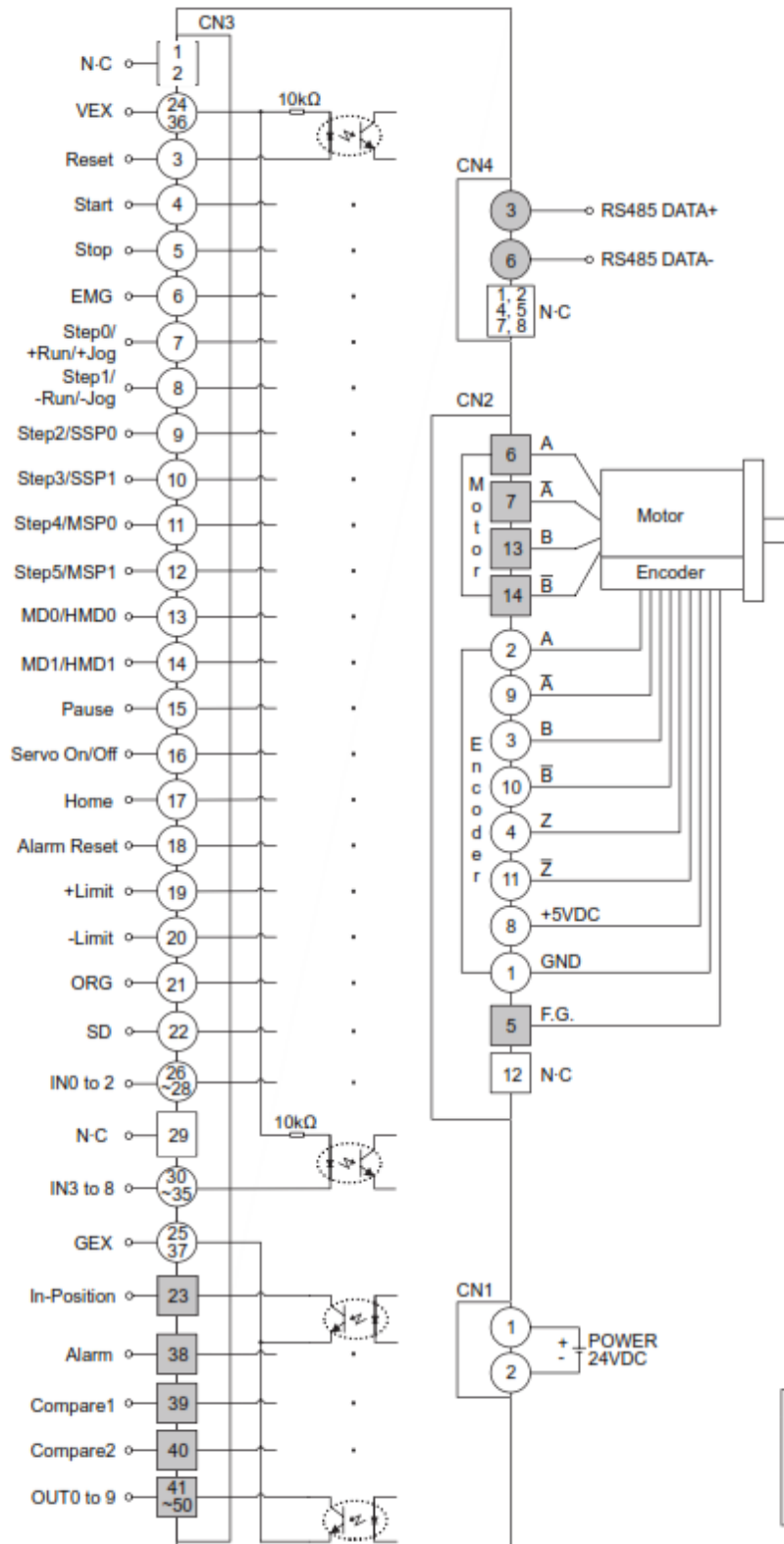
Pin no.	Function (name tag)	Cable color	Dot line color-numbers	Pin no.	Function (name tag)	Cable color	Dot line color-numbers
1	N-C	Orange	Black-1	26	IN0	White	Red-3
2	N-C		Red-1	27	IN1		Black-4
3	RESET		Black-2	28	IN2		Red-4
4	START		Red-2	29	N-C		Black-5
5	STOP		Black-3	30	IN3		Red-5
6	EMG	Yellow	Red-3	31	IN4	Gray	Black-1
7	STEP0/+RUN/+JOG		Black-4	32	IN5		Red-1
8	STEP1/-RUN/-JOG		Red-4	33	IN6		Black-2
9	STEP2/SSP0		Black-5	34	IN7		Red-2
10	STEP3/SSP1		Red-5	35	IN8		Black-3
11	STEP4/MSP0	White	Black-1	36	VEX	Pink	Red-3
12	STEP5/MSP1		Red-1	37	GEX		Black-1
13	MD0/HMD0		Black-2	38	ALARM		Red-4
14	MD1/HMD1		Red-2	39	COMPARE1		Black-5
15	PAUSE		Black-3	40	COMPARE2		Red-5
16	SERVO ON/OFF	White	Red-3	41	OUT0	Pink	Black-1
17	HOME		Black-4	42	OUT1		Red-1
18	ALARM RESET		Red-4	43	OUT2		Black-2
19	+LIMIT		Black-5	44	OUT3		Red-2
20	-LIMIT		Red-5	45	OUT4		Black-3
21	ORG	White	Black-1	46	OUT5		Red-3
22	SD		Red-1	47	OUT6		Black-4
23	IN POSITION		Black-2	48	OUT7		Red-4
24	VEX		Red-2	49	OUT8		Black-5
25	GEX		Black-3	50	OUT9		Red-5

Şekil 3.30. Autonics AiC-D Serisi Step Motor Sürücüsü I/O Kablosu

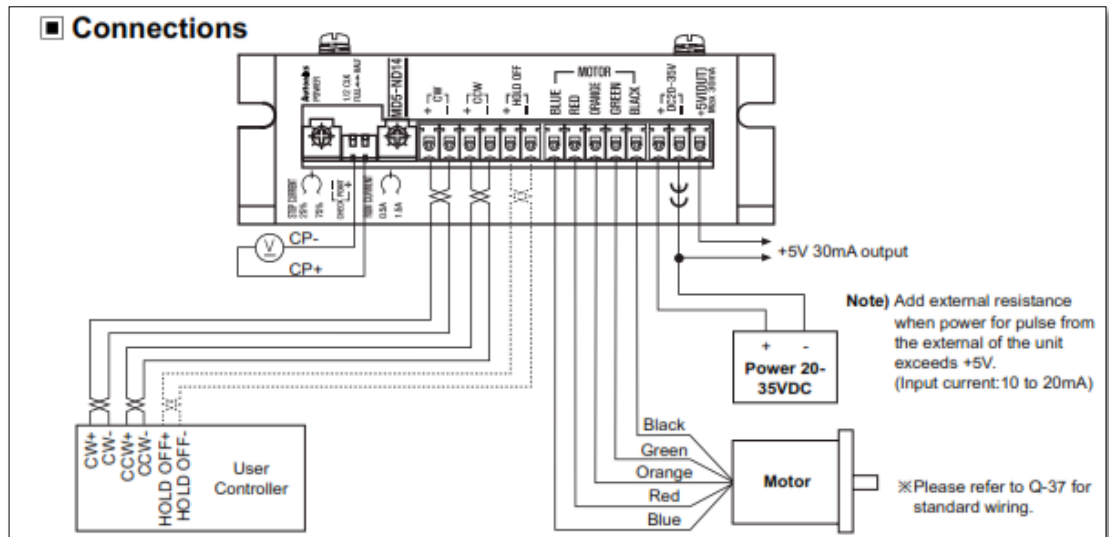


Şekil 3.31. Autonics AiC-D Serisi Step Motor Sürücüsü Üniteleri

■ Connection for Motor and Driver



(a)



Şekil 3.32. Autonics AiC-D Serisi Step Motor Sürücüsü ve Motor Bağlantıları (a),
Bağlantılar (b)

Alarm / Warning

Depending on the alarm type, it flashes for 0.4 sec interval and it turns OFF for 0.8 sec repeatedly.

■ Alarm

No. of flashing	Alarm type	No. of flashing	Alarm type
1	Overcurrent error	10	Command speed error
2	Overspeed error	11	Input voltage error
3	Position tracking error	12	In-Position error
4	Overload error	13	Memory error
5	Overheat error	14	Emergency stop
6	Motor connection error	15	Program mode error
7	Encoder connection error	16	Index mode error
8	Regenerative voltage error	17	Home search mode error
9	Motor alignment error	-	

■ Warning

No. of flashing	Warning type
1	+Software limit
2	-Software limit
3	+Hardware limit
4	-Hardware limit
5	Overload warning
6	Override warning

Exclusive Input (20)		
Signal name	Descriptions	Pin no.
Reset	Reset command	3
Start	Drive start command	4
Stop	Drive stop command	5
EMG	Drive emergency stop command	6
Step0/+Run/+Jog	Step designate0/+Continuous/+Jog	7
Step1/-Run/-Jog	Step designate1/-Continuous/-Jog	8
Step2/SSP0	Step designate2/ Start speed designate 0	9
Step3/SSP1	Step designate3/ Start speed designate 1	10
Step4/MSP0	Step designate4/ Max. speed designate 0	11
Step5/MSP1	Step designate5/ Max. speed designate 1	12
MD0/HMD0	Operation mode designate 0/Home search mode designate 0	13
MD1/HMD1	Operation mode designate 1/Home search mode designate 1	14
Pause	Pause	15
Servo On/Off	Servo On/Off	16
Home	Home search	17
Alarm Reset	Alarm reset command	18
+Limit	+ direction limit sensor	19
-Limit	- direction limit sensor	20
ORG	Home sensor	21
SD	Deceleration (Deceleration stop) signal	22

Şekil 3.34. Autonics AiC-D Serisi Step Motor Sürücüsü Girişleri

Driving by I/O Control			
Please refer to the user manual of AiC			
AiC-D series has 4 types of driving mode and home search as following.			
There are two ways to execute each driving mode that are using atMotion, PC program, and using input of input/output connector (CN3).			
Drive mode name	Descriptions	MD0/HMD0 (13)	MD1/HMD1 (14)
Jog drive	Outputs drive while instruction input signal is ON to $\pm$ direction	ON	OFF
Continuous drive	Starts drive output when instruction input signal to $\pm$ direction turns on once, and stops it when the signal turns off.	OFF	ON
Program drive	Operates by the designated program.	ON	ON
Index drive	Designates and operates the one step among the saved instructions in the program	OFF	OFF
Home search	Operates home search	-	-

Şekil 3.35. Autonics AiC-D Serisi Step Motor Program Modları

Comparison of Driving Method			
Driving method	by atMotion	by Visual Studio	by I/O control
Jog drive	●	x	●
Continuous drive	●	●	●
Program drive	●	●	●
Index drive	●	●	●
Home search	●	●	●

Şekil 3.36. Autonics AiC-D Serisi Step Motor Sürme Metotlarının Karşılaştırılması

3.6. Grafcet Diyagramı

Deney setinde ürün ayırımında kullanılmak üzere 4 çeşit malzeme seçildi:

1. Uzun plastik
2. Kısa plastik
3. Uzun metal
4. Kısa metal

Setin programlamasında örnek uygulama olarak şu işlemler yapılmaktadır:

1. Döner tablanın ve setin başlangıç konumuna alınması
2. Döner tablaya taşınacak ürün tercihinin yapılması
3. Ürün kimlik tanımlaması
4. Eğer gelen ürün istenilen ürün ise tablaya, değil ise diğer banda taşınması
5. Döner tablanın hareket ve doluluk kontrolü

Sırasıyla bu aşamaları açıklayalım:

1. Döner Tablanın ve Setin Başlangıç Konumuna Alınması:

Sete enerji verildiğinde yazmaçlar sıfırlanmakta, pnömotik silindir başlangıç konumuna gelmekte ve döner tabla, “in position” noktasını aramaktadır. Döner tabla altına konulan bir U tipi kızılötesi sensör ile “in position” noktası bulunmaktadır. Sonrasında tabla başlangıç pozisyonuna (Home) yönlendirilmektedir. Başlangıç pozisyonunda döner tabla altına yerleştirilmiş bir sensör vardır ve bu sensör ile tablada ürünün varlığı algılanmaktadır. Tabla 8 ürün almaktadır.

2. Döner tablaya taşınacak ürün tercihinin yapılması:

Setin üzerinde dokunmatik ekran ile veya PC SCADA yazılımı ile tablaya taşınacak ürün tercih edilmektedir. İstenilen ürün tercihi yapıldıktan sonra sisteme start (başlama) verilmektedir. Plastik malzeme derlin malzeme olduğu için yazılımda isimlendirme olarak derlin de kullanılmıştır.

3. Ürün kimlik tanımlaması:

Bu aşamada öncelikle birinci bandın bidayetinde (başlangıcında) bir sensör bulunmaktadır. Bu sensör, ürünün banda konduğunu algılamaktadır. Eğer ürün banda konmuş ise sırasıyla öncelikle uzun veya kısa ürün olduğu iki sensör ile tespit edilmektedir.

Sonrasında endüktif ve kapasitif sensörler ile de ürünün metal ya da plastik olduğu tespit edilmektedir. Bandın sonunda (nihayet) bulunan bir sensör ile de ürünün bandın sonuna geldiği algılanmakta ve bant durdurulmaktadır. Şekil 3.37’de konveyörler üzerindeki sensörler görülmektedir.



Şekil 3.37. Konveyörler Üzerindeki Sensörler

4. Eğer gelen ürün istenilen ürün ise tablaya, değil ise diğer banda taşınması:

Ürün seçimi ve kimlik doğrulama işlemi gerçekleştikten sonra ürün birinci konveyör nihayetinde (sonunda) durmaktadır. Pnömotik sistem harekete geçmeden önce gelen ürün ile seçilen ürün kıyaslaması yapılır. Eğer gelen ürün seçilen ürün ile aynı ise ürün tablaya taşınır ve eğer tabla dolmadıysa bir adım ileri hareket ettirilir. Eğer gelen ürün istenilen ürün değilse o zaman ürün pnömotik vakum ile ikinci banda taşınır. İkinci bant geri hareket ettirilir. İstenmeyen ürün ikinci bandın sonunda bir sensör ile algılanır ve ikinci bant durdurulur.

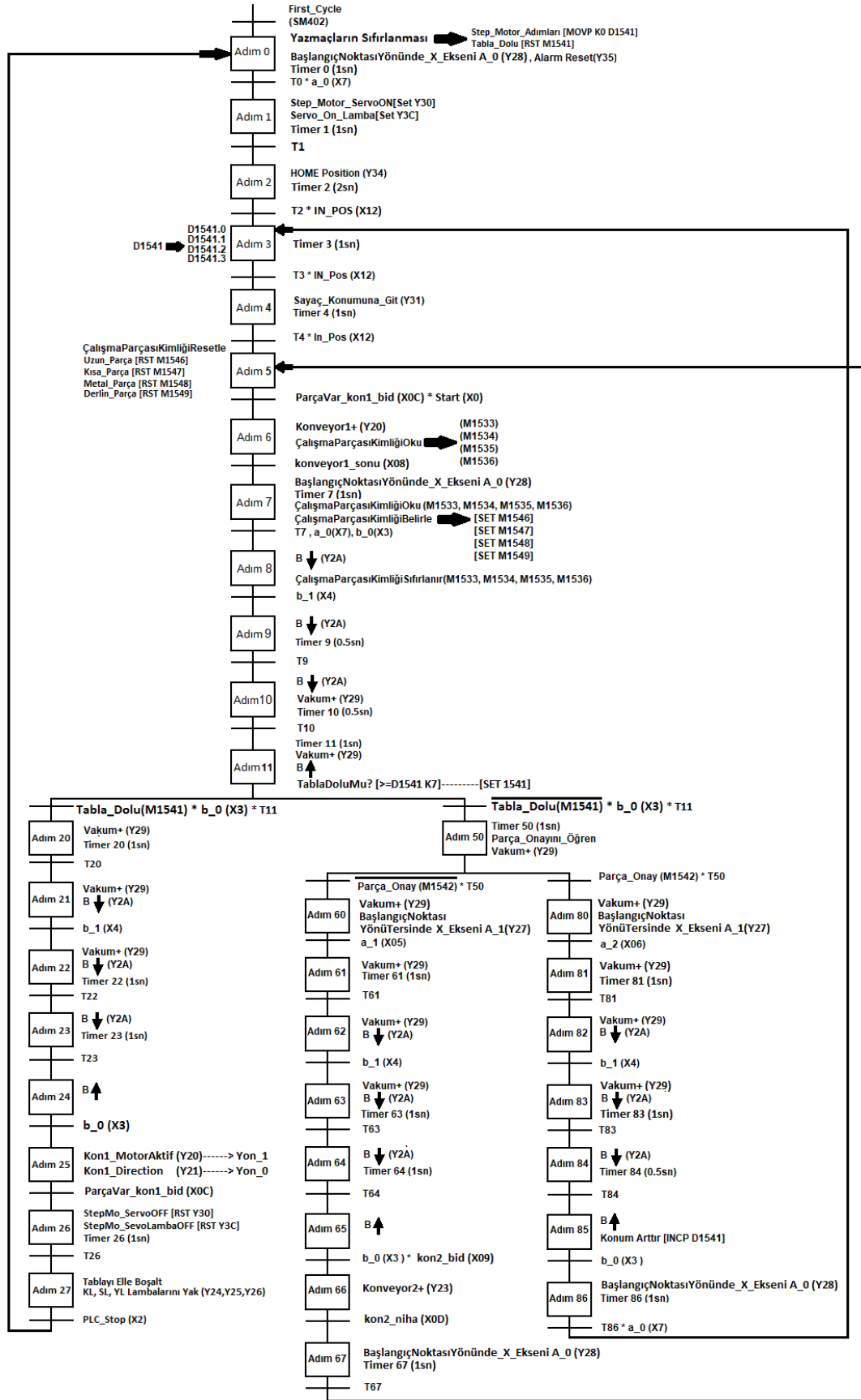
5. Döner tablanın hareket ve doluluk kontrolü:

Döner tablaya her ürün geldiğinde sayaç bir artırılır. Toplam 8 ürün sonunda döner tablanın dolduğunu ikaz eden ışıklı uyarı yapılır. Tablanın boşaltılması istenir. Tabla boşaltılıp PLC Stop (Reset) düğmesine basılarak sistem resetlenir.

Bu algoritmaya ait setin PLC-1 giriş-çıkış adresleri (Şekil 3.38) esas alınarak oluşturulan Grafcet diyagramı Şekil 3.39’de gösterilmektedir:

X0	PLC_Start	Y20	Kon1_MotorAktif_Yon_1
X1	PLC_Auto_Man	Y21	Kon1_Direction_Yon_0
X2	PLC_Stop	Y22	Kon2_1
X3	b_0	Y23	Kon2_0
X4	b_1	Y24	KL
X5	a_1	Y25	SL
X6	a_2	Y26	YL
X7	a_0	Y27	A_1
X8	kon1_nih	Y28	A_0
X9	kon2_bid	Y29	Vakum
X0A	uzun_par_Sens	Y2A	B_1
X0B	kisa_par_Sens	Y2B	
X0C	kon1_bid	Y2C	
X0D	kon2_nih	Y2D	
X0E	donTab_parca_var	Y2E	
X0F	donTab_org	Y2F	
X10	kapasitif_Sens	Y30	StepMo Servo
X11	enduktif_Sens	Y31	StepMo Start
X12	in_position	Y32	StepMo Stop
X13	alam	Y33	Boş-Bozuk
		Y34	StepMo Home
M1533	uzun	Y35	StepMo Alam_Res
M1534	kisa	Y36	KonumData0
M1535	metal	Y37	KonumData1
M1536	derlin	Y38	KonumData2
M1537		Y39	KonumData3
M1538		Y3A	MDO/HMD0
M1539		Y3B	MD1/HMD1
M1540		Y3C	ServoON_Lamba
M1541	tabla_dolu	Y3D	Lamba1
M1542	parca_onay	Y3E	Lamba2
M1543		Y3F	Lamba3
M1544			
M1545			
M1546	uzun_parca		
M1547	kisa_parca		
M1548	metal_parca		
M1549	plastik-parca		
M1550			
M1551	Touch_kisa_derlin		
M1552	Touch_uzun_derlin		
M1553	Touch_kisa_metal		
M1554	Touch_uzun_metal		
D1541	StepAdimlari		

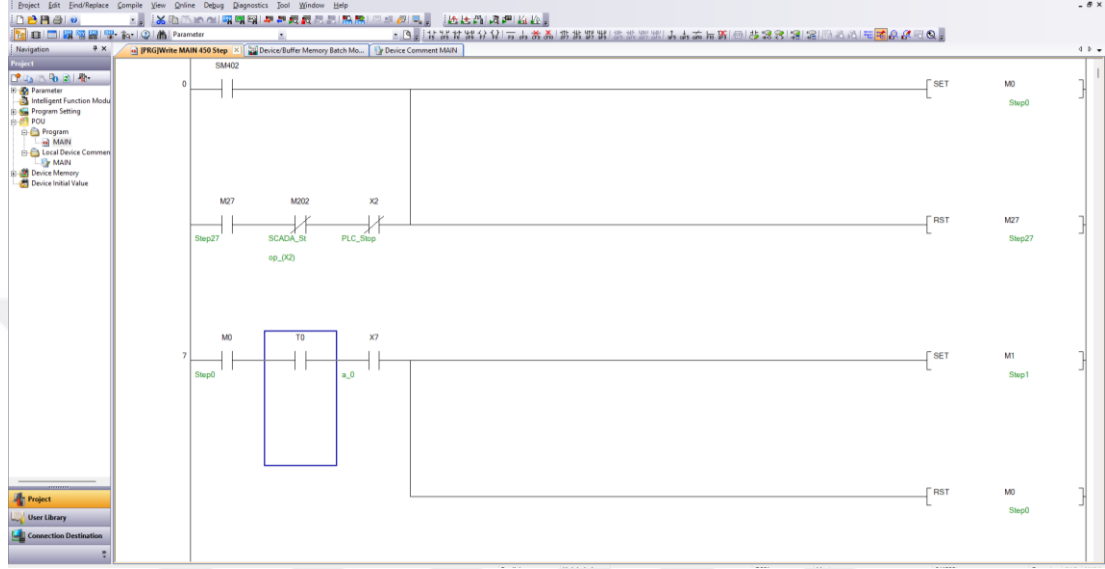
Şekil 3.38. PLC-1 Giriş-Çıkış Adresleri



Şekil 3.39. PLC-1 İçin Grafcet Diyagramı (Algoritması)

3.7. Setin PLC Yazılımı (Programı)

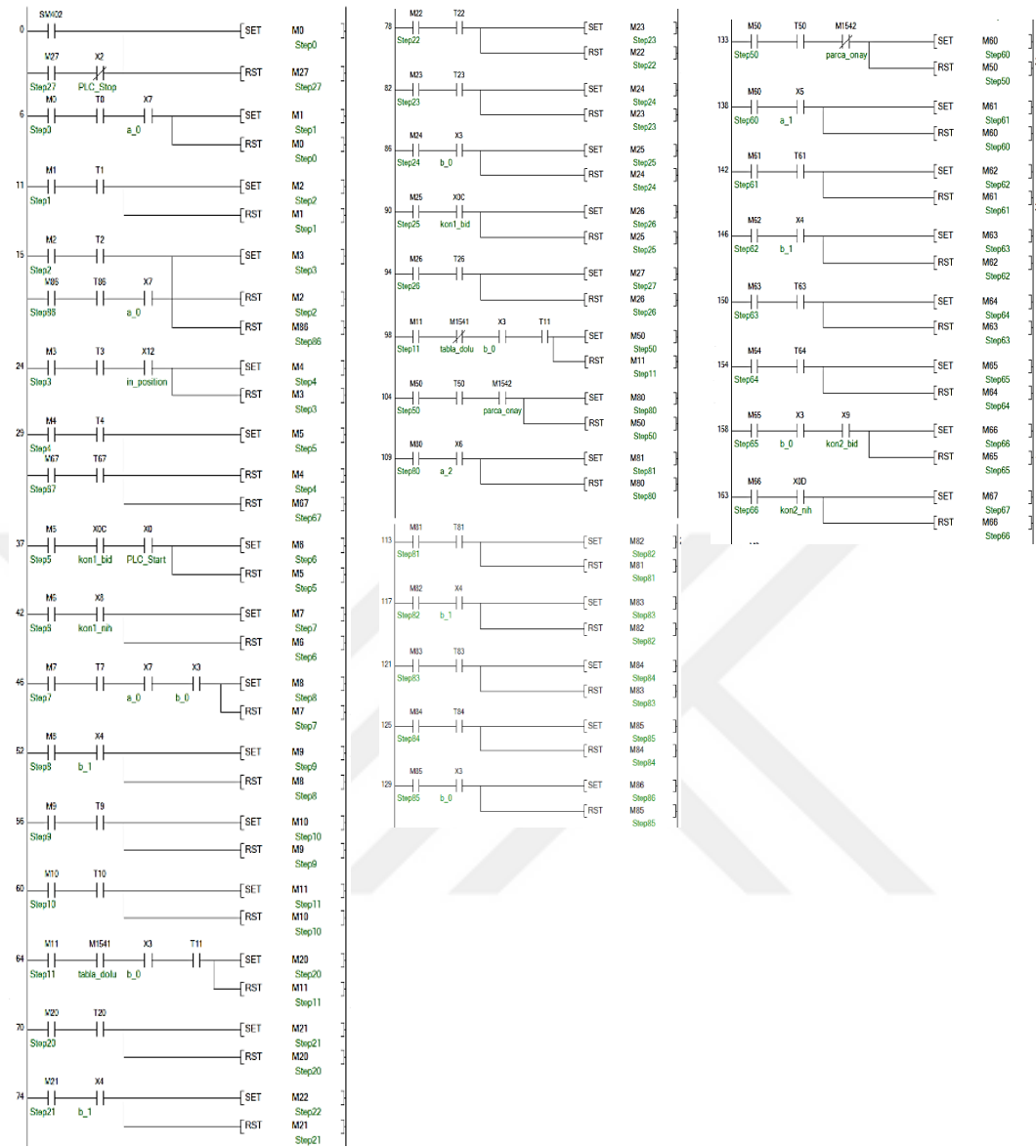
Setin üzerinde farklı işlemcili iki adet PLC bulunmaktadır. PLC seçimi bir şalter ile yapılabilmektedir. Öncelikle varsayılan olarak PLC-1 tercih edilmiş, PLC-2 ise ikincil olarak tercih edilebilmektedir.



Şekil 3.40 PLC-1 İçin Ladder Programlama

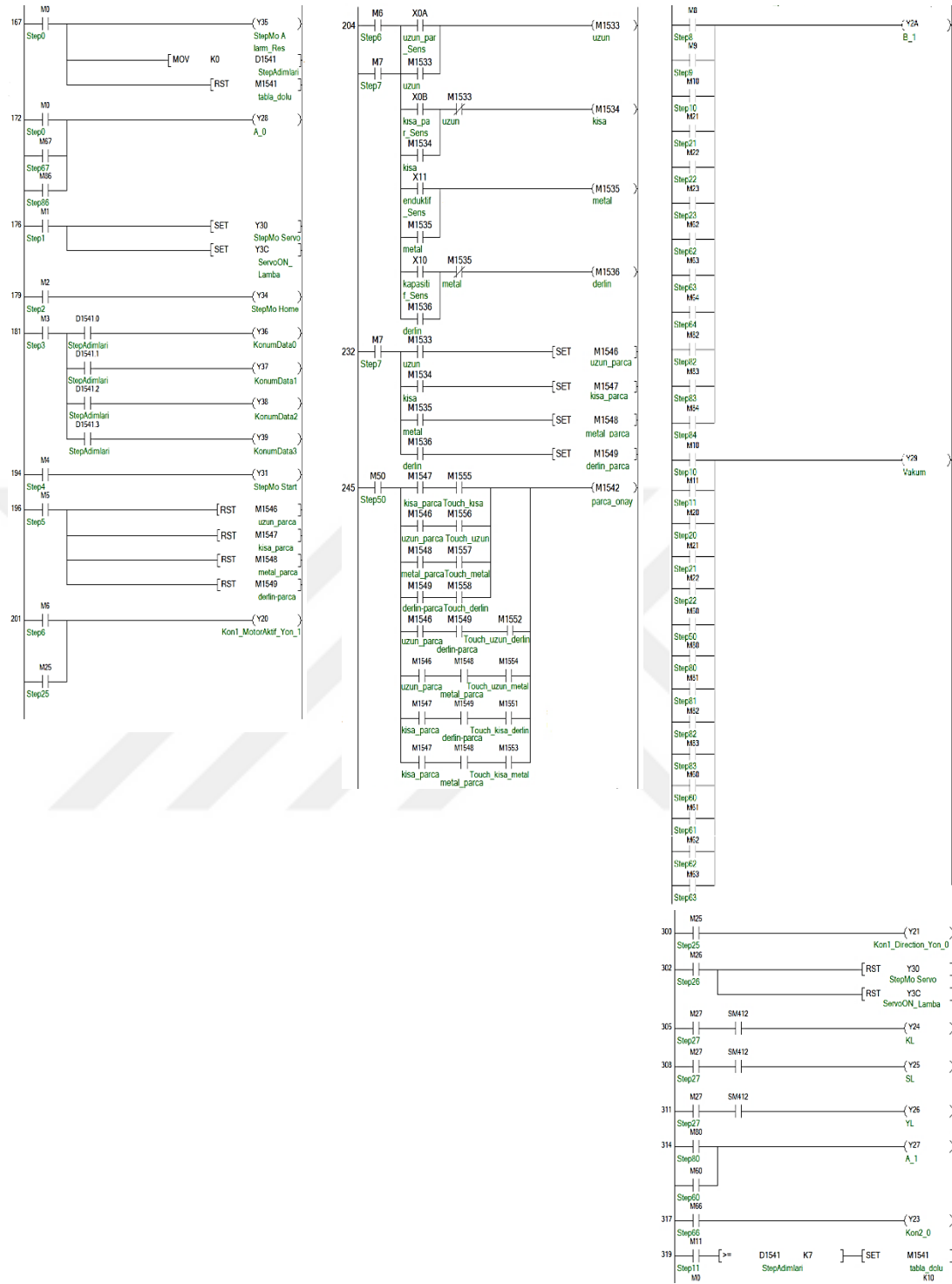
Yapılan Grafcet diyagramı her iki PLC için de geçerlidir. Bu diyagram esas alınarak PLC yazılımı yazılmaktadır. İlk olarak mevcut Grafcet diyagramına göre PLC-1 yazılımı ilgili program arayüzü kullanarak ladder programlama ile yapılmıştır. Şekil 3.41 de ise PLC-1 programı görülmektedir. Ayrıca her iki PLC için yazılan Ladder programları ekler (EK-A) bölümündedir.

Adımlar (Steps):



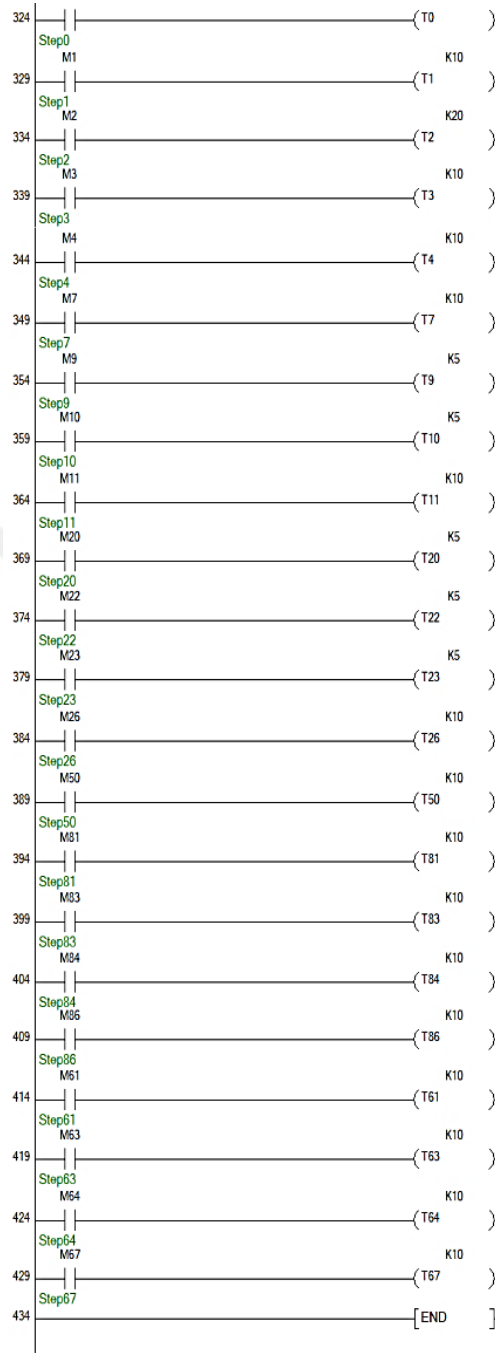
(a)

Eylemler (Actions):



(b)

Zamanlayıcılar (Timers):



(c)

Şekil 3.41. PLC-1 Programı: Adımlar (a), Eylemler (b), Zamanlayıcılar (c)

3.8. Setin Görsel Yazılımı (SCADA)

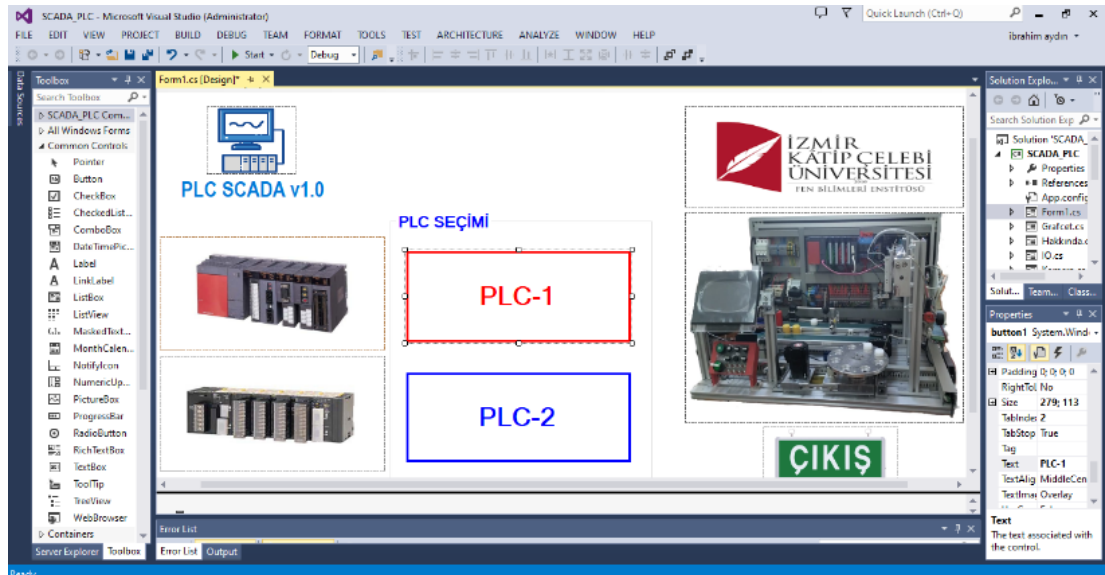
Setin görsel yazılımı (SCADA) için .NET platformunda C# yazılım dili tercih edilmiştir. Setin yazılımı için PLC kütüphaneleri projeye eklenmiştir. Öncelikle tasarım aşaması yapılmış, sonrasında kodlama aşamasına geçilmiştir.

Yazılım, SCADA mantığına göre gerçekleştirilmiştir. SQL Server ile bağlantılı olarak anlık kayıtlar veri tabanına yazılmıştır. Veri tabanında kayıtlı verileri forma yansıtmakla kalınmamış, aynı zamanda Excel dosyasına da taşıma imkânı verilmiştir. Grafikselsel olarak da ürün bilgileri form üzerinde gösterilmiştir.

SCADA içinde simülasyon olması gerektiğinden, setin çalışması anlık olarak benzetilmiştir. PLC giriş ve çıkışları (I/O) anlık olarak eşzamanlı okunarak test sayfası formuna yansıtılmıştır. AForge.Net görüntü kütüphanesi de kullanılarak setin kamera ile izlenmesi sağlanmıştır.

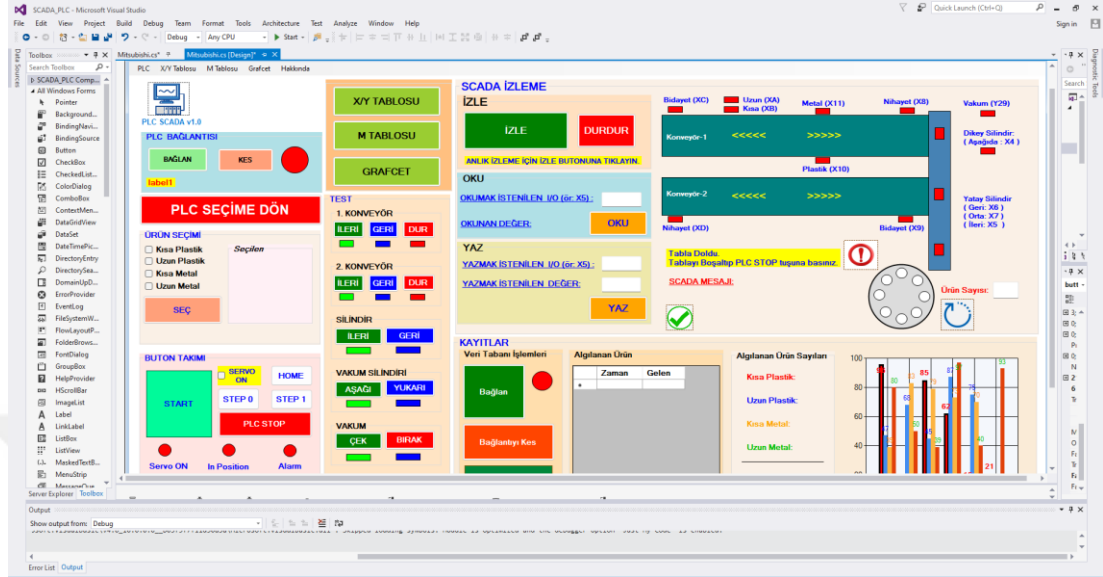
3.8.1. Form Tasarımı

Program çalıştırıldığında ilk sayfa PLC seçimi sayfası olarak tasarlandı. Şekil 3.42’de giriş formu tasarımı görülmektedir.



Şekil 3.42. Giriş Formu Tasarımı

PLC-1 seçimi yapıldığında SCADA formuna geçilmektedir. Şekil 3.43'te SCADA form tasarım ekranı görülmektedir. PLC-2 Form tasarımı ve program kodları ekler (EK-B) bölümündedir.



Şekil 3.43. SCADA form tasarımı

Formun sol üst köşesinde PLC bağlantısı için butonlar bulunmaktadır. PLC' ye bağlanıldığında veya bağlanma gerçekleşmediğinde kullanıcı yazı ve görsel ile bilgilendirilmektedir.

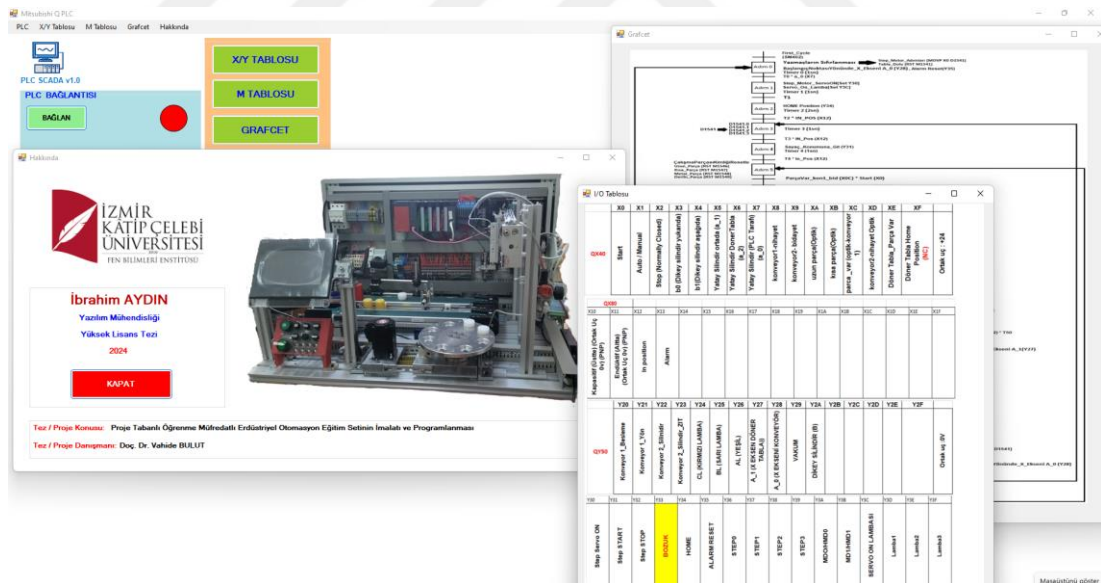
Hemen altında ürün seçimi yapılmaktadır. Kısa ve uzun plastik, kısa ve uzun metal olmak üzere set üzerinde 4 çeşit ürün tercih edilebilmektedir. Ürün kimlik okuması sensörler aracılığıyla Konveyör-1 üzerinde yapılmaktadır. Seçilen ürünler döner tablada toplanmakta, istenmeyen ürünler konveyör-2'ye taşınmaktadır. Bu taşıma işlemi pnömotik silindirler ve vakum ile yapılmaktadır.

Formun sol alt köşesine, set üzerindeki buton takımı (Şekil 3.44) ve led göstergeler eklenmiştir. Start (başlama) butonu ile set çalıştırılmaktadır. Sistem çalıştırıldığında döner tabla servo on olur, in position konumunu tabla altındaki sensör aracılığıyla bulur ve ilk konumu olan home (ürün bekleme pozisyonu) konumuna gider ürün bekler. Herhangi bir problemde alarm lambası yakılır. Formda Servo ON, In Position ve Alarm lambalarına yer verilmekle beraber, Step motor adımlarına gitmek için butonlara da yer verilmiştir. Servo ON aktif iken STEP butonlarına basılmalıdır. O yüzden yazılımda Servo ON için checkBox nesnesi tercih edilmiştir.



Şekil 3.44. Buton takımı

Formun orta bölümünde Giriş/Çıkış, Grafçet ve yardımcı kontakların listesine erişim sağlanmaktadır. Aynı erişimler form menülerinde de bulunmaktadır. Şekil 3.45'te bu listelere erişim formları görülmektedir.



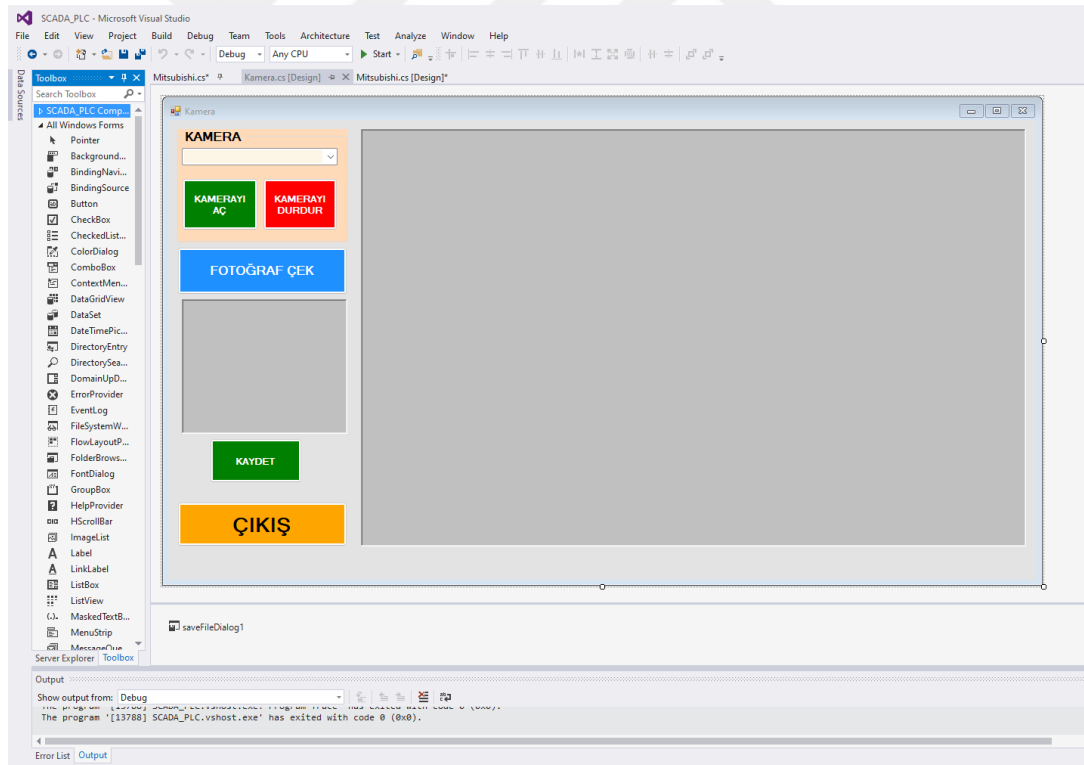
Şekil 3.45. Giriş-Çıkış ve Grafcet formları

Orta bölüm ise test alanıdır. Sistemde bulunan aktüatörler (konveyörler, silindirler ve vakum) butonlar ile test edilebilmektedir. Bu sayede sistemin başında bir problem olup olmadığı anlaşılmaktadır.

Formun sağ üst köşesi ise benzetim için ayrılmıştır. Setin görsel krokisi çizilmiş, sensörler için grafiksek uyarılar eklenmiş, anlık olarak takibi için butonlar yerleştirilmiştir. Simülasyona uyarı yazıları ve görseller eklenerek kullanıcı anlık olarak bilgilendirmesi sağlanmıştır. Aynı zamanda anlık olarak istenilen kaydedici, giriş ve çıkışlara veri okuma/yazma işlemi yapılabilmektedir.

Formun sağ alt tarafı ise gelen ürün verilerinin gözlemlenmesine ayrılmıştır. Veri tabanındaki veriler anlık olarak forma yazdırılmakta ve grafiksel olarak izlenebilmektedir. Tüm veri tabanı verileri Excel butonu ile excel sayfasına aktarılabilmektedir.

Formun altındaki kamera butonu ile setin anlık görüntüsü forma aktarılmaktadır. Bu sayede set anlık olarak izlenebilmektedir. İstenilen anda fotoğraf çekilerek kayıt altına alınabilmektedir. Görüntü kütüphanesi olarak AForge.NET Kütüphanesi kullanılmıştır. Şekil 3.46'da kamera form tasarımı görülmektedir.



Şekil 3.46. Kamera form tasarımı

Formu çalıştırdığımızda PLC bağlantısı yapılmadığı sürece diğer tasarımlar görülmemektedir (visible özellikleri false edilmiştir). PLC bağlantısı yapıldığında ilk olarak ürün seçimi yapılmakta sonrasında buton takımındaki Start butonuna basarak sistem çalıştırılmaktadır. İstenildiğinde sistem çalıştırılmadan önce test edilebilir. Sonrasında tekrar sistem kapatılıp enerjilendirilerek çalıştırılabilir. Simülasyon anlık olarak başlatılıp durdurulabilir. Şekil 3.47’de sistem çalışırken bir anlık form görüntüsü, Şekil 3.48’de ise SQL Veri tabanı tabloları görülmektedir.

The screenshot shows the SCADA form interface with the following components:

- PLC BAĞLANTISI:** A panel with a 'KES' button and a 'PLC BAĞLANTISI' indicator.
- PLC SEÇİME DÖN:** A panel with a 'SEÇ' button and a 'PLC SEÇİMİ' section showing selected items: 'Kısa Plastik', 'Uzun Plastik', 'Kısa Metal', and 'Uzun Metal'.
- BUTON TAKIMI:** A panel with 'START', 'STOP', 'HOME', 'STEP 0', and 'STEP 1' buttons, along with 'Servo ON', 'In Position', and 'Alarm' indicators.
- TEST:** A panel with '1. KONVEYÖR' and '2. KONVEYÖR' sections, each containing 'İLERİ', 'GERİ', and 'DUR' buttons.
- SCADA İZLEME:** A central panel with 'İZLE' and 'DURDUR' buttons, a 'SIMULATÖR ÇALIŞIYOR' indicator, and a 'KONVEYÖR-1' and 'KONVEYÖR-2' section showing 'Plastik (X10)' and 'Metal (X11)' indicators.
- KAYITLAR:** A panel with 'Veri Tabanı İşlemleri' and 'Alınan Ürün' sections, including a table of received products and a 'Toplam: 17' summary.
- ÜRÜN ONAYLANDI:** A panel with a 'ÜRÜN ONAYLANDI' indicator and a 'ÜRÜN SAYILARINI SIFIRLA' button.
- Bar Chart:** A bar chart showing the count of products for 'Kısa Plastik', 'Uzun Plastik', 'Kısa Metal', and 'Uzun Metal'.

Şekil 3.47. SCADA formu anlık ekranı

The screenshot shows the SQL Server Enterprise Manager interface with the following components:

- Server Explorer:** A tree view showing the database structure, including 'Databases', 'System Databases', 'PLC_SCADA', 'Tables', 'Views', 'Synonyms', 'Programmability', 'Service Broker', 'Storage', 'Security', 'Server Objects', 'Replication', and 'Management'.
- Table Designer:** A panel showing the 'Kısa_Plastik' table with columns: 'Kısa_Plastik', 'Uzun_Plastik', 'Kısa_Metal', and 'Uzun_Metal'.
- Column Properties:** A panel showing the properties for the 'Kısa_Plastik' column, including 'Data Type', 'Allow Nulls', 'Default Value or Binding', and 'Table Designer'.
- Properties:** A panel showing the properties for the 'Kısa_Plastik' table, including 'Name', 'Database Name', 'Description', 'Schema', 'Server Name', 'Identity Column', 'Identity Increment', 'Lock Escalation', 'Table', 'Regular Data Spec', 'Replicated', 'Row GUID Column', and 'Text/Image Flags'.

Şekil 3.48. SQL Veri Tabanı Tabloları

3.8.2. SCADA Program Kodları

SCADA program kodları Visual Studio.NET platformunda C# programlama dilinde yazılmıştır. Form tasarımı yapıldıktan sonra kodlamaya geçilmiştir. Kodlama öncesinde referanslar projeye yüklenmiş, “using” komutlarıyla kütüphaneler projeye eklenmiştir. Sonrasında SQL veri tabanına bağlantı sağlanmış, global değişkenler tanımlanmıştır. Aşağıda referans listesi, projede kullanılan kütüphaneler ve SQL veri tabanı bağlantısı görülmektedir.

```
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Timers;
using System.Threading;
using System.Data.SqlClient;
using Microsoft.Office.Core;
using myExcel = Microsoft.Office.Interop.Excel;
using Microsoft.Office.Interop.Excel;

namespace SCADA_PLC
{
    public partial class Mitsubishi : Form
    {
        SqlDataAdapter da;
        DataSet ds;

        public Mitsubishi()
        {
            InitializeComponent();

        }
        public SqlConnection irtibat() // SQL Veri tabanı bağlantısı
        {
            SqlConnection bag = new SqlConnection("Data Source =
BUDIN1686\\WINCCPLUSMIG2014; Initial Catalog = PLC_SCADA; Integrated
Security = True");
            bag.Open();
            return bag;
        }
        // Global Değişkenler
```

```

int st, s = 255, o = 0;
int X4, X5, X6, X7, X8, X9, XA, XB, XC, XD, X10, X11;
int oku_X4, oku_X5, oku_X6, oku_X7, oku_X8, oku_X9, oku_XA, oku_XB,
oku_XC, oku_XD, oku_X10, oku_X11;
int oku_Y29, Y29, Y20, oku_Y20, Y21, oku_Y21;
int Y22, oku_Y22;
int Y23, oku_Y23;
int Y31, oku_Y31;
int M1546, M1547, M1548, M1549, M1542;
int oku_M1546, oku_M1547, oku_M1548, oku_M1549, oku_M1542;
int D1541, oku_D1541;
string urun;
int kp, up, km, um, toplam=0;
int i, j, lt, urun_onay;

```

PLC' ye bağlanma ve bağlantıyı kesmek için butonlar kullanıldı. Ayrıca form pencere menülerine de aynı işlevler eklendi. Grafiksel olarak form zenginleştirildi. Aşağıda PLC bağlantı ve bağlantıyı kesme kodları görülmektedir.

// PLC Bağlantısı

```

ACTMULTILib.ActEasyIF PLC = new ACTMULTILib.ActEasyIF();

```

```

private void button1_Click(object sender, EventArgs e)
{
    PLC.ActLogicalStationNumber = 100;
    int baglanti=PLC.Open();
    button2.Enabled = true;
    label1.Visible = true;
    label1.Text = baglanti.ToString();

    if (baglanti == 0)
    {
        ovalShape1.BackColor = Color.LightGreen;
        label1.ForeColor = Color.Green;
        label1.BackColor = Color.Lime;
        label1.Text = "PLC'YE BAĞLANILDI";
        button2.Visible = true;
        button1.Visible = false;
        button5.Visible = true;
        groupBox2.Visible = true;
        groupBox4.Visible = true;
        groupBox9.Visible = true;
        groupBox10.Visible = true;
        groupBox11.Visible = true;
        groupBox12.Visible = true;
        groupBox13.Visible = true;
        groupBox16.Visible = true;
        button34.Visible = true;
    }
}

```

```

else
{
    label1.ForeColor = Color.Red;
    label1.BackColor = Color.Yellow;
    label1.Text = "HATA: BAĞLANTI YAPILAMADI.";
}

}

private void button2_Click(object sender, EventArgs e)
{
    PLC.Close();
    timer1.Stop();
    timer2.Stop();
    timer1.Enabled = false;
    label1.Visible = true;
    ovalShape1.BackColor = Color.Red;
    label1.ForeColor = Color.Red;
    label1.BackColor = Color.LightCyan;
    label1.Text = "PLC'YE BAĞLANTI KESİLDİ";
    st = PLC.WriteDeviceRandom("M300", 1, ref (o));
    st = PLC.WriteDeviceRandom("M301", 1, ref (o));

    st = PLC.WriteDeviceRandom("M5", 1, ref (o));

    button1.Visible = true;
    button2.Visible = false;
    button5.Visible = false;
    button34.Visible = false;
    groupBox2.Visible = false;
    groupBox4.Visible = false;
    groupBox9.Visible = false;
    groupBox10.Visible = false;
    groupBox11.Visible = false;
    groupBox12.Visible = false;
    groupBox16.Visible = false;

}

```

PLC bağlantısı yapıldıktan sonra dört üründen (kısa plastik, uzun plastik, kısa metal, uzun metal) tablaya aktarılmasını istediğimiz ürünleri seçen komutlar aşağıda görülmektedir:

```

// Ürün Seçme
private void button5_Click(object sender, EventArgs e)
{
    int veri1, veri1_oku, veri2, veri2_oku, veri3, veri3_oku;

```

```

groupBox12.Visible = true;

veri1_oku = PLC.ReadDeviceRandom("Y3C", 1, out (veri1));
if (veri1 == 1) { ovalShape11.BackColor = Color.Lime; }

veri2_oku = PLC.ReadDeviceRandom("X12", 1, out (veri2));
if (veri2 == 1) { ovalShape12.BackColor = Color.Lime; }

veri3_oku = PLC.ReadDeviceRandom("x13", 1, out (veri3));
if (veri3 == 1) { ovalShape11.BackColor = Color.Lime; }

if (checkBox1.Checked == true)
{
    checkBox1.ForeColor = Color.Red;
    checkBox1.BackColor = Color.Yellow;
    label2.Text = checkBox1.Text;
    label6.Text = "";
    button4.Visible = true;
    groupBox12.Visible = true;
    PLC.SetDevice("M1551", 1);
}

if (checkBox1.Checked == false)
{
    checkBox1.ForeColor = Color.Black;
    checkBox1.BackColor = Color.Linen;
    label2.Text = "";
    PLC.SetDevice("M1551", 0);
}

if (checkBox2.Checked == true)
{
    checkBox2.ForeColor = Color.Red;
    checkBox2.BackColor = Color.Yellow;
    label3.Text = checkBox2.Text;
    label6.Text = "";
    button4.Visible = true;
    groupBox12.Visible = true
    PLC.SetDevice("M1552", 1);
}

if (checkBox2.Checked == false)
{
    checkBox2.ForeColor = Color.Black;
    checkBox2.BackColor = Color.Linen;
    label3.Text = "";
    PLC.SetDevice("M1552", 0);
}

if (checkBox3.Checked == true)
{

```

```

        checkBox3.ForeColor = Color.Red;
        checkBox3.BackColor = Color.Yellow;
        label4.Text = checkBox3.Text;
        label6.Text = "";
        button4.Visible = true;
        PLC.SetDevice("M1553", 1);
    }
    if (checkBox3.Checked == false)
    {
        checkBox3.ForeColor = Color.Black;
        checkBox3.BackColor = Color.Linen;
        label4.Text = "";
        PLC.SetDevice("M1553", 0);
    }
    if (checkBox4.Checked == true)
    {
        checkBox4.ForeColor = Color.Red;
        checkBox4.BackColor = Color.Yellow;
        label5.Text = checkBox4.Text;
        label6.Text = "";
        button4.Visible = true;
        groupBox12.Visible = true;
        PLC.SetDevice("M1554", 1);
    }
    if (checkBox4.Checked == false)
    {
        checkBox4.ForeColor = Color.Black;
        checkBox4.BackColor = Color.Linen;
        label5.Text = "";
        PLC.SetDevice("M1554", 0);
    }
}

if (checkBox1.Checked == false && checkBox2.Checked == false &&
checkBox3.Checked == false && checkBox4.Checked == false)

{
    label6.Text = "EN AZ BİR ÜRÜN SEÇMELİSİNİZ...";
    button4.Visible = false;
    groupBox12.Visible = false;
    st = PLC.WriteDeviceRandom("M1551", 1, ref (o));
    st = PLC.WriteDeviceRandom("M1552", 1, ref (o));
    st = PLC.WriteDeviceRandom("M1553", 1, ref (o));
    st = PLC.WriteDeviceRandom("M1554", 1, ref (o));
}
}

```

Sonrasında SCADA programı içinde, setin çıkışlarını test edebileceğimiz bölüm kodlandı. Bu sayede setin konveyörler, silindirleri ve vakum çıkışları test edilmekte, kullanıcı grafiksel olarak da bilgilendirilmektedir. Aşağıda bu kodlar görülmektedir:

//TEST

```
private void button6_Click(object sender, EventArgs e)
{
    st = PLC.WriteDeviceRandom("M300", 1, ref (s));
    st = PLC.WriteDeviceRandom("M301", 1, ref (o));

    rectangleShape1.Visible = true;
    rectangleShape2.Visible = false;
    rectangleShape3.Visible = false;
    label13.Visible = true;
    label12.Visible = false;
}
```

```
private void button7_Click(object sender, EventArgs e)
{
    st = PLC.WriteDeviceRandom("M300", 1, ref (s));
    st = PLC.WriteDeviceRandom("M301", 1, ref (s));

    rectangleShape1.Visible = false;
    rectangleShape2.Visible = true;
    rectangleShape3.Visible = false;
    label13.Visible = false;
    label12.Visible = true;
}
```

```
private void button8_Click(object sender, EventArgs e)
{
    st = PLC.WriteDeviceRandom("M300", 1, ref (o));
    st = PLC.WriteDeviceRandom("M301", 1, ref (o));

    rectangleShape1.Visible = false;
    rectangleShape2.Visible = false;
    rectangleShape3.Visible = true;
    label13.Visible = false;
    label12.Visible = false;
}
```

```
private void button9_Click(object sender, EventArgs e)
{
    st = PLC.WriteDeviceRandom("M302", 1, ref (s));
    st = PLC.WriteDeviceRandom("M303", 1, ref (o));

    rectangleShape4.Visible = true;
    rectangleShape5.Visible = false;
    rectangleShape6.Visible = false;
    label15.Visible = true;
}
```

```

        label14.Visible = false;
    }

    private void button10_Click(object sender, EventArgs e)
    {
        st = PLC.WriteDeviceRandom("M302", 1, ref (o));
        st = PLC.WriteDeviceRandom("M303", 1, ref (s));

        rectangleShape4.Visible = false;
        rectangleShape5.Visible = true;
        rectangleShape6.Visible = false;
        label14.Visible = true;
        label15.Visible = false;
    }

    private void button11_Click(object sender, EventArgs e)
    {
        st = PLC.WriteDeviceRandom("M302", 1, ref ( o));
        st = PLC.WriteDeviceRandom("M303", 1, ref (o));

        rectangleShape4.Visible = false;
        rectangleShape5.Visible = false;
        rectangleShape6.Visible = true;
        label15.Visible = false;
        label14.Visible = false;
    }

    private void button14_Click(object sender, EventArgs e)
    {
        st = PLC.WriteDeviceRandom("M304", 1, ref (s));
        st = PLC.WriteDeviceRandom("M305", 1, ref (o));

        rectangleShape9.Visible = true;
        rectangleShape8.Visible = false;
        rectangleShape80.BackColor = Color.Lime;
        rectangleShape81.BackColor = Color.Red;
    }

    private void button13_Click(object sender, EventArgs e)
    {
        st = PLC.WriteDeviceRandom("M304", 1, ref (o));
        st = PLC.WriteDeviceRandom("M305", 1, ref (s));

        rectangleShape8.Visible = true;
        rectangleShape9.Visible = false;
        rectangleShape81.BackColor = Color.Lime;
        rectangleShape80.BackColor = Color.Red;
    }

```

```

private void button17_Click(object sender, EventArgs e)
{
    st = PLC.WriteDeviceRandom("M306", 1, ref (s));

    rectangleShape12.Visible = true;
    rectangleShape11.Visible = false;
    rectangleShape13.BackColor = Color.Lime;
}

private void button16_Click(object sender, EventArgs e)
{
    st = PLC.WriteDeviceRandom("M306", 1, ref (o));

    rectangleShape11.Visible = true;
    rectangleShape12.Visible = false;
    rectangleShape13.BackColor = Color.Red;
}

private void button15_Click(object sender, EventArgs e)
{
    st = PLC.WriteDeviceRandom("M308", 1, ref (s));

    rectangleShape7.Visible = true;
    rectangleShape10.Visible = false;
    rectangleShape90.BackColor = Color.Lime;
}

private void button12_Click(object sender, EventArgs e)
{
    st = PLC.WriteDeviceRandom("M308", 1, ref (o));

    rectangleShape7.Visible = false;
    rectangleShape10.Visible = true;
    rectangleShape90.BackColor = Color.Red;
}

```

SCADA programlarında anlık benzetim önemlidir. Sistemin izlenmesi önem arz eder. Anlık benzetim edilmiş sistemlerde anlık veri okuma ve yazma mümkün olmaktadır. Setin çalışması anlık olarak takip edildi. Bunun için timer nesneleri kullanıldı. Aşağıda timer nesnelere ait kodlamalar görülmektedir:

//Anlık Simülasyon (Benzetim)

```
private void timer1_Tick(object sender, EventArgs e)
{
    timer2.Enabled = true;
    oku_Y20 = PLC.ReadDeviceRandom("Y20", 1, out (Y20));
    oku_Y21 = PLC.ReadDeviceRandom("Y21", 1, out (Y21));
    if (Y20 == 1 && Y21 == 0) { label13.Visible = true; } else { label13.Visible
= false; }
    if (Y20 == 1 && Y21 == 1) { label12.Visible = true; } else { label12.Visible
= false; }

    oku_XC = PLC.ReadDeviceRandom("XC", 1, out (XC)); if (XC == 1) {
rectangleShape82.BackColor = Color.Lime; } else { rectangleShape82.BackColor =
Color.Red; }
    oku_XA = PLC.ReadDeviceRandom("XA", 1, out (XA)); if (XA == 1) {
rectangleShape84.BackColor = Color.Lime; } else { rectangleShape84.BackColor =
Color.Red; }
    oku_XB = PLC.ReadDeviceRandom("XB", 1, out (XB)); if (XB == 0) {
rectangleShape83.BackColor = Color.Red; }
    if (XB == 1 && XA == 0) { rectangleShape83.BackColor = Color.Lime; }
else { rectangleShape83.BackColor = Color.Red; }

    oku_X10 = PLC.ReadDeviceRandom("X10", 1, out (X10));
    oku_X11 = PLC.ReadDeviceRandom("X11", 1, out (X11));
    if (X10 == 1) { rectangleShape86.BackColor = Color.Lime; }
    if (X11 == 1) { rectangleShape85.BackColor = Color.Lime; }

    oku_X8 = PLC.ReadDeviceRandom("X8", 1, out (X8)); if (X8 == 1) {
rectangleShape87.BackColor = Color.Lime; } else { rectangleShape87.BackColor =
Color.Red; }

    oku_X4 = PLC.ReadDeviceRandom("X4", 1, out (X4)); if (X4 == 1) {
rectangleShape13.BackColor = Color.Lime; } else { rectangleShape13.BackColor =
Color.Red; }

    oku_D1541 = PLC.ReadDeviceRandom("D1541", 1, out (D1541));

    oku_Y29 = PLC.ReadDeviceRandom("Y29", 1, out (Y29)); if (Y29 == 1) {
rectangleShape90.BackColor = Color.Lime; } else { rectangleShape90.BackColor =
Color.Red; }

    oku_X5 = PLC.ReadDeviceRandom("X5", 1, out (X5)); if (X5 == 1) {
rectangleShape79.BackColor = Color.Lime; } else { rectangleShape79.BackColor =
Color.Red; }
    oku_X6 = PLC.ReadDeviceRandom("X6", 1, out (X6)); if (X6 == 1) {
rectangleShape80.BackColor = Color.Lime; } else { rectangleShape80.BackColor =
Color.Red; }
```

```

oku_X7 = PLC.ReadDeviceRandom("X7", 1, out (X7)); if (X7 == 1) {
rectangleShape81.BackColor = Color.Lime; } else { rectangleShape81.BackColor =
Color.Red; }

```

```

oku_XD = PLC.ReadDeviceRandom("XD", 1, out (XD)); if (XD == 1) {
rectangleShape89.BackColor = Color.Lime; } else { rectangleShape89.BackColor =
Color.Red; }

```

```

oku_X9 = PLC.ReadDeviceRandom("X9", 1, out (X9)); if (X9 == 1) {
rectangleShape88.BackColor = Color.Lime; } else { rectangleShape88.BackColor =
Color.Red; }

```

```

oku_Y23 = PLC.ReadDeviceRandom("Y23", 1, out (Y23));

```

```

if (Y22 == 1 && Y23 == 0) { label15.Visible = true; } else { label15.Visible
= false; }
if (Y22 == 0 && Y23 == 1) { label14.Visible = true; } else { label14.Visible
= false; }

```

```

textBox2.Text = Convert.ToString(D1541);

```

```

oku_Y31 = PLC.ReadDeviceRandom("Y31", 1, out (Y31));

```

```

timer2.Enabled = true;

```

```

if (

```

```

D1541 == 0) { ovalShape6.BackColor = Color.White;
ovalShape3.BackColor = Color.White; ovalShape7.BackColor = Color.White;
ovalShape5.BackColor = Color.White; ovalShape8.BackColor =
Color.White; ovalShape4.BackColor = Color.White;
ovalShape10.BackColor = Color.White; ovalShape9.BackColor =
Color.White; label7.Text = ""; pictureBox2.Visible = false; }

```

```

if (D1541 == 1)

```

```

{
pictureBox5.Visible = true;
ovalShape6.BackColor = Color.Blue; ovalShape3.BackColor =
Color.White; ovalShape7.BackColor = Color.White;
ovalShape5.BackColor = Color.White; ovalShape8.BackColor =
Color.White; ovalShape4.BackColor = Color.White;
ovalShape10.BackColor = Color.White; ovalShape9.BackColor =
Color.White; label7.Text = ""; pictureBox2.Visible = false;
}

```

```

if (D1541 == 2)
{
    pictureBox5.Visible = true;
    ovalShape6.BackColor = Color.Blue; ovalShape3.BackColor =
Color.Blue; ovalShape7.BackColor = Color.White;
    ovalShape5.BackColor = Color.White; ovalShape8.BackColor =
Color.White; ovalShape4.BackColor = Color.White;
    ovalShape10.BackColor = Color.White; ovalShape9.BackColor =
Color.White; label7.Text = ""; pictureBox2.Visible = false;

}

if (D1541 == 3)
{
    pictureBox5.Visible = true;
    ovalShape6.BackColor = Color.Blue; ovalShape3.BackColor =
Color.Blue; ovalShape7.BackColor = Color.Blue;
    ovalShape5.BackColor = Color.White; ovalShape8.BackColor =
Color.White; ovalShape4.BackColor = Color.White;
    ovalShape10.BackColor = Color.White; ovalShape9.BackColor =
Color.White; label7.Text = ""; pictureBox2.Visible = false;

}

if (D1541 == 4)
{
    pictureBox5.Visible = true;
    ovalShape6.BackColor = Color.Blue; ovalShape3.BackColor =
Color.Blue; ovalShape7.BackColor = Color.Blue;
    ovalShape5.BackColor = Color.Blue; ovalShape8.BackColor =
Color.White; ovalShape4.BackColor = Color.White;
    ovalShape10.BackColor = Color.White; ovalShape9.BackColor =
Color.White; label7.Text = ""; pictureBox2.Visible = false;

}

if (D1541 == 5)
{
    pictureBox5.Visible = true;
    ovalShape6.BackColor = Color.Blue; ovalShape3.BackColor =
Color.Blue; ovalShape7.BackColor = Color.Blue;
    ovalShape5.BackColor = Color.Blue; ovalShape8.BackColor =
Color.Blue; ovalShape4.BackColor = Color.White;
    ovalShape10.BackColor = Color.White; ovalShape9.BackColor =
Color.White; label7.Text = ""; pictureBox2.Visible = false;

```

```

    }

    if (D1541 == 6)
    {
        pictureBox5.Visible = true;
        ovalShape6.BackColor = Color.Blue; ovalShape3.BackColor =
Color.Blue; ovalShape7.BackColor = Color.Blue;
        ovalShape5.BackColor = Color.Blue; ovalShape8.BackColor =
Color.Blue; ovalShape4.BackColor = Color.Blue;
        ovalShape10.BackColor = Color.White; ovalShape9.BackColor =
Color.White; label7.Text = ""; pictureBox2.Visible = false;
    }

    if (D1541 == 7)
    {
        pictureBox5.Visible = true;
        ovalShape6.BackColor = Color.Blue; ovalShape3.BackColor =
Color.Blue; ovalShape7.BackColor = Color.Blue;
        ovalShape5.BackColor = Color.Blue; ovalShape8.BackColor =
Color.Blue; ovalShape4.BackColor = Color.Blue;
        ovalShape10.BackColor = Color.Blue; ovalShape9.BackColor =
Color.White; label7.Text = ""; pictureBox2.Visible = false;

        if (D1541 == 8)
        {
            pictureBox5.Visible = true;
            ovalShape6.BackColor = Color.Blue; ovalShape3.BackColor =
Color.Blue; ovalShape7.BackColor = Color.Blue;
            ovalShape5.BackColor = Color.Blue; ovalShape8.BackColor =
Color.Blue; ovalShape4.BackColor = Color.Blue;
            ovalShape10.BackColor = Color.Blue; ovalShape9.BackColor =
Color.Blue;
            label7.Text = "TABLA DOLDU. TABLAYI BOŞALTIN !!!";
            pictureBox2.Visible = true; pictureBox5.Visible = false;
            button28.Visible = true; label23.Visible = true;
        }
        pictureBox5.Visible = false;
    }

```

Gelen ürün SQL veri tabanına kaydedildi ve ekranda anlık olarak gösterildi. Grafiksel olarak da kullanıcı bilgilendirildi. Ayrıca istenilen giriş, çıkış ve kaydedici içeriği okunabilmekte ve veri yazılabilmektedir. Aşağıda PLC ye veri yazma ve PLC' den veri okuma komutları ile anlık SQL verilerini alma ve ekranda gösterme komutları görülmektedir:

//PLC Veri Okuma

```
private void button18_Click(object sender, EventArgs e)
{
    int OKU_TEXT, deger;

    OKU_TEXT= PLC.ReadDeviceRandom(textBox1.Text, 1, out (deger));

    label9.Text = Convert.ToString(deger);
}
```

//PLC Veri Yazma

```
private void button3_Click(object sender, EventArgs e)
{
    PLC.Close();
    this.Close();

    timer1.Stop();
    timer2.Stop();
    timer3.Stop();
    timer1.Enabled = false;
    st = PLC.WriteDeviceRandom("M300", 1, ref (o));
    st = PLC.WriteDeviceRandom("M301", 1, ref (o));
    st = PLC.WriteDeviceRandom("M5", 1, ref (o));
}
```

// SQL Veritabanına Bağlanma

```
private void button29_Click(object sender, EventArgs e)
{
    SqlConnection bag = irtibat();
    bag = new SqlConnection("Data Source =
BUDIN1686\\WINCCPLUSMIG2014; Initial Catalog = PLC_SCADA; Integrated
Security = True");
    bag.Open();

    dataGridView1.Visible = true;
    button29.Enabled = false;
    button30.Enabled = true;
    button32.Visible = true;
    button34.Visible = true;

    if (bag.State.ToString()=="Open")
    {

        ovalShape14.BackColor = Color.Lime;
        label25.Text = "Bağlantı Kuruldu.";
        dataGridView1.Visible = true;
    }
}
```

```

    }
    else
    {
        ovalShape14.BackColor = Color.Lime;
        label25.Text = "Bağlantı Kurulamadı...";
        dataGridView1.Visible = false;
    }

    toplam = kp + up + km + um;

    label30.Text = Convert.ToString(kp);
    label31.Text = Convert.ToString(up);
    label32.Text = Convert.ToString(km);
    label33.Text = Convert.ToString(um);
    label35.Text = Convert.ToString(toplam);

    chart1.Series["Kısa_Plastik"].Points.AddXY("Kısa Plastik", kp);
    chart1.Series["Uzun_Plastik"].Points.AddXY("Uzun Plastik", up);
    chart1.Series["Kısa_Metal"].Points.AddXY("Kısa Metal", km);
    chart1.Series["Uzun_Metal"].Points.AddXY("Uzun Metal", um);
}

// Simülasyon Mesajları

private void timer2_Tick(object sender, EventArgs e)
{
    oku_M1546 = PLC.ReadDeviceRandom("M1546", 1, out (M1546));
    oku_M1547 = PLC.ReadDeviceRandom("M1547", 1, out (M1547));
    oku_M1548 = PLC.ReadDeviceRandom("M1548", 1, out (M1548));
    oku_M1549 = PLC.ReadDeviceRandom("M1549", 1, out (M1549));

    if (M1546 == 1 && M1548 == 1) { label7.Text = "Uzun Metal Ürün Algılandı."; urun = "Uzun Metal"; um = um + 1; timer3.Enabled = true; timer3.Start(); }

    if (M1546 == 1 && M1549 == 1) { label7.Text = "Uzun Derlin (Plastik) Ürün Algılandı."; urun = "Uzun plastik"; up = up + 1; timer3.Enabled = true; timer3.Start(); }

    if (M1547 == 1 && M1548 == 1) { label7.Text = "Kısa Metal Ürün Algılandı."; urun = "Kısa Metal"; timer3.Enabled = true; km = km + 1; timer3.Enabled = true; timer3.Start(); }

    if (M1547 == 1 && M1549 == 1) { label7.Text = "Kısa Derlin (Plastik) Ürün Algılandı."; urun = "Kısa Plastik"; kp = kp + 1; timer3.Enabled = true; timer3.Start(); }

    oku_M1542 = PLC.ReadDeviceRandom("M1542", 1, out (M1542));

```

```

        if (M1542 == 1) { urun_onay = 1; }
        if (M1542 == 0) { urun_onay = 0; }
        if (urun_onay == 1) { label24.Text = "ÜRÜN ONAYLANDI";
pictureBox6.Visible = true; }
        if (urun_onay == 0) { label24.Text = ""; pictureBox6.Visible = false; }

        toplam = kp + up + km + um;

        label30.Text = Convert.ToString(kp);
        label31.Text = Convert.ToString(up);
        label32.Text = Convert.ToString(km);
        label33.Text = Convert.ToString(um);
        label35.Text = Convert.ToString(toplam);

        st = PLC.WriteDeviceRandom("M1546", 1, ref (o));
        st = PLC.WriteDeviceRandom("M1547", 1, ref (o));
        st = PLC.WriteDeviceRandom("M1548", 1, ref (o));
        st = PLC.WriteDeviceRandom("M1549", 1, ref (o));

        st = PLC.WriteDeviceRandom("M1542", 1, ref (o));

        timer2.Enabled = false;
    }

```

Veri tabanından alınan veriler excel sayfasına bir buton ile kaydedilmektedir. Aşağıda excel bağlantısı ve veri kaydetme komutları görülmektedir:

//Excel Bağlantısı ve Veri Aktarma

```

private void button32_Click(object sender, EventArgs e)
{
    myExcel._Application excel = new myExcel.Application();
    excel.Visible = true;
    object Missing = Type.Missing;
    Workbook workbook = excel.Workbooks.Add(Missing);
    Worksheet sheet1 = (Worksheet)workbook.Sheets[1];
    int StartCol = 1;
    int StartRow = 1;

    for (int j = 0; j < dataGridView1.Columns.Count; j++)
    {
        Range myRange = (Range)sheet1.Cells[StartRow, StartCol + j];
        myRange.Value2 = dataGridView1.Columns[j].HeaderText;
    }

    StartRow++;
}

```

```

        for (int i = 0; i < dataGridView1.Rows.Count; i++)
        {
            for (int j = 0; j < dataGridView1.Columns.Count; j++)
            {
                Range myRange = (Range)sheet1.Cells[StartRow + i, StartCol + j];
                myRange.Value2 = dataGridView1[j, i].Value == null ? "" :
dataGridView1[j, i].Value.ToString();
                myRange.Select();
            }
        }
    }
}

```

Son olarak setin kamera ile izlenmesi için bir buton ile kamera formuna geçilebilmektedir. Görüntü işlemleri için AForge.NET görüntü kütüphanesi kullanılmıştır. Kamera seçimi ve bağlantısı yapıldıktan sonra, istenilen anda setin kamera ile fotoğrafı çekilip kaydedilebilmektedir. Kamera formuna ait kod satırları aşağıda görülmektedir:

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using AForge.Video;
using AForge.Video.DirectShow;
namespace SCADA_PLC
{
    public partial class Kamera : Form
    {
        public Kamera()
        {
            InitializeComponent();
        }

        private FilterInfoCollection webcam; // Tüm webcam aygıtları dizisi
        private VideoCaptureDevice cam; //Kullanacak aygıt.

        private void Kamera_Load(object sender, EventArgs e)
        {
            webcam = new FilterInfoCollection(FilterCategory.VideoInputDevice);
            //webcam dizisine mevcut kameraları doldur
            foreach (FilterInfo videocapturedevice in webcam)

```

```

        {
            comboBox1.Items.Add(videocapturedevice.Name); //kameraları
        }
        comboBox1.SelectedIndex = 0; // ComboBox'taki ilk index numaralı
        kameranın ekranda görünmesi için
    }

```

// Görüntü alma

```

private void button1_Click(object sender, EventArgs e)
{
    cam = new
    VideoCaptureDevice(webcam[comboBox1.SelectedIndex].MonikerString);
    cam.NewFrame += new NewFrameEventHandler(cam_NewFrame);
    cam.Start();
    button1.Visible = false;
    button2.Visible = true;
}

```

//Kamerayı kapatma

```

private void button2_Click(object sender, EventArgs e)
{
    if (cam.IsRunning) //kamera açıksa kapatıyoruz.
    {
        cam.Stop();
        button2.Visible = false;
        button1.Visible = true;
    }
}

```

// Fotoğraf Önizleme

```

private void cam_NewFrame(object sender, NewFrameEventArgs eventArgs)
{
    Bitmap bit = (Bitmap)eventArgs.Frame.Clone();
    pictureBox1.Image = bit;
}

```

//Fotoğraf Çekme

```

private void button3_Click(object sender, EventArgs e)
{
    pictureBox2.Image = pictureBox1.Image;
}

```

```

//Fotoğraf Kaydetme
private void button4_Click(object sender, EventArgs e)
{
    if (saveFileDialog1.ShowDialog() == DialogResult.OK)
    {
        pictureBox1.Image.Save(saveFileDialog1.FileName); // Görüntüyü kaydet.
    }
}

//For kapatılırken kamera açıksa kapatma.

private void Form1_FormClosing(object sender, FormClosingEventArgs e)
{
    if (cam.IsRunning)
    {
        cam.Stop();
    }
}

// Formu Kapatma

private void button5_Click(object sender, EventArgs e)
{
    cam.Stop();

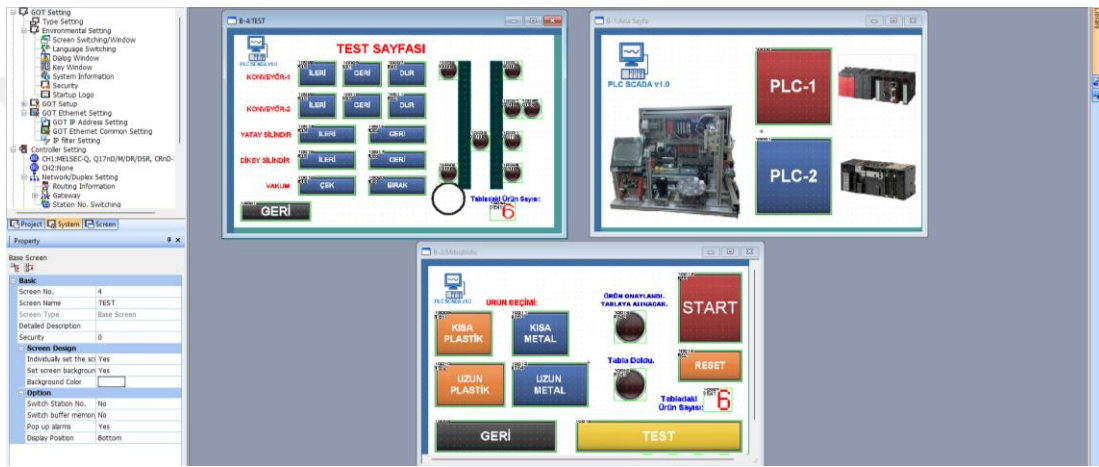
    this.Close();
}

}
}

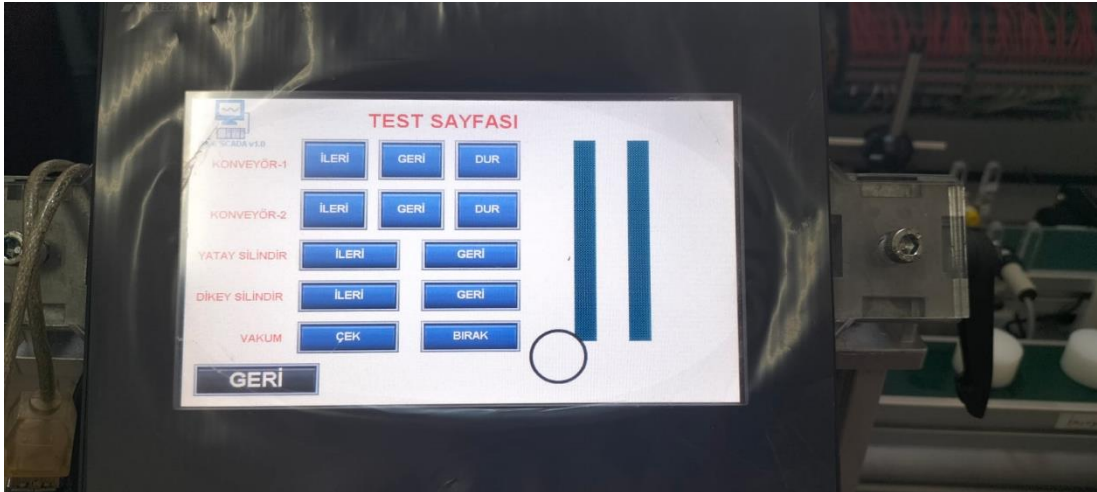
```

3.9. Dokunmatik (Touch) Panel Tasarımı

Eğitim setine montajlanan dokunmatik panel, ethernet yoluyla PLC ile haberleşmektedir. PLC-1 için Şekil 3.49’da da görüldüğü üzere bir tasarım yapılmıştır. İlk sayfada PLC seçimi yapılmaktadır. Test butonu ile test sayfasına geçildiğinde, PLC girişi ve çıkışları test edilebilmektedir. Konveyör ve silindir hareketleri, sensör verileri test edilebilmektedir. Aynı zamanda bu sayfa simülasyon olarak da kullanılmaktadır. Bant üzerindeki sensör verileri ve tabladaki ürün sayısı anlık olarak bu sayfada takip edilebilmektedir. Ürün seçme sayfasında ise tablaya yüklenmesi istenilen ürünler seçilmektedir. Ürün onaylandığında onay lambası, tabla dolduğunda tabla dolum ikaz lambası yanmaktadır. Tabladaki ürün sayısı da yine bu ekranda da görülmektedir. Şekil 3.50’de dokunmatik paneldeki test sayfası görülmektedir.



Şekil 3.49. Dokunmatik Panel Tasarım Ekranı



Şekil 3.50. Dokunmatik Panel Test Sayfası

3.10. PTÖ Uygulaması

PTÖ temelli eğitimde, öğrencilere ilk PLC için Grafcet tasarımı, PLC ladder programı ve SCADA yazılımı anlatıldıktan ve uygulandıktan sonra, aynı set üzerinde aynı Grafcet algoritması kullanılarak benzer SCADA tasarımı ve yazılımının yapılması öğrencilerden istenmiştir. Öğrencinin bu aşamaya gelmesi için bu çalışmada izah edilen PTÖ müfredatı ve metodu kullanılmıştır. Üçer kişilik dört öğrenci grubu, PTÖ faaliyetine bir ay süre ile katılmışlardır. Değerlendirmeler ilgili bölümde açıklanan PTÖ değerlendirme formları (rubrikler) ile olmuştur. Öğrenciler PTÖ sürecini başarı ile tamamlayarak ilgili formlarını doldurmuş ve sunumlarında memnuniyetlerini dile getirmişlerdir. Öğretmenin ve öğrencilerin doldurdıkları örnek formlar ekler (Ek-E) bölümündedir.

PTÖ uygulama süreci şu şekilde gerçekleştirilmiştir:

- Öğrencilere set üzerinde örnek çalışma anlatılmış, öğrenciler PTÖ için hazır hale getirilmiştir.
- Öğrenciler 3 kişilik 4 gruba ayrılmıştır.
- Öğrencilerden sanal bir şirket kurmaları, kendi şirket logolarını tasarlamaları istenmiştir.
- Öğrencilere PTÖ planı, rubrikler anlatılmış, proje fikri ve sürüş sorusu verilmiştir.
- Öğrenciler görev paylaşımı ve iş zaman çizelgeleri hazırlamak için toplantı yapmışlardır. Toplantı sonunda kendi aralarında sorumluluk alma sözleşmesini imzalayarak görev paylaşımı ve iş-zaman çizelgelerini oluşturmuşlardır.
- Öğretmen her hafta gözlem forumu ile öğrencileri gözlemlemiştir. Gözlem formu ve diğer formlar (Şekil 3.7- Şekil 3.13) daha önce hazırlanmıştır.
- Öğrenciler sete ethernet yoluyla bağlanarak kendi tasarım ve programlarını geliştirmişler ve test etmişlerdir.
- PTÖ çalışmalarını rapor ve sunum ile öğretmenlerine sunmuşlardır.
- PTÖ sonunda formlar doldurulmuş, öğrenciler hem kendilerini hem de takım arkadaşlarını değerlendirme fırsatı bulmuşlardır.
- Değerlendirme rubrikleri ile öğrenciler ürün ve süreç odaklı olarak değerlendirilmiştir.

Bölüm 4

4. Bulgular ve Tartışma

Bu çalışmada iki farklı işlemcili PLC ile bir endüstriyel otomasyon eğitim seti tasarlanmıştır. Grafcet algoritması her iki PLC için de geçerlidir. Setin yapısında mümkün olduğu kadar çeşitli endüstriyel otomasyon teknolojileri kullanılmıştır.

PLC, DC motor ve adım(step) motoru, konveyör bant, endüktif, kapasitif ve kızılötesi sensörler, elektropnömatik sistem, çeşitli silindirler, valfler, dokunmatik ekran ve endüstriyel ethernet gibi birçok endüstriyel teknolojiye set üzerinde yer verilmiştir.

Setin eğitiminde kullanılmak üzere proje tabanlı öğrenim temelli bir eğitim müfredatı da hazırlanmıştır. Müfredatın yanında değerlendirme formları (rubrikleri) de hazırlanarak çalışmaya eklenmiştir

Setin kontrolü için SCADA yazılımı Visual C# ile tasarlanmış ve kodlanmıştır. SCADA yazılımında endüstriyel Ethernet, seri haberleşme, aktüatörlerinin ve sensörlerin testi, anlık benzetim, veri tabanına kayıt, veri tabanından anlık veri gösterimi, excel sayfasına veri yazdırma, grafiksel gösterim, görsel ve yazılı alarm, bilgi mesajları, görüntü alma ve fotoğraf kayıt gibi birçok kodlama çeşidine yer verilmiştir.

4.1. Montaj Sürecindeki Bulgular

Setin tamamlanmasına kadar geçen süreçte donanımsal olarak aşağıdaki bulgulara rastlanmıştır:

- PLC ve SCADA tabanlı sistem imalatında her aşama iyi tasarlanmalı, adım adım ve notlar alarak planlı bir şekilde yapılmalıdır. Kullanılacak ürünler, PLC ve yazılıma uygun seçilmelidir. Yapmış olduğumuz set bize, planlı ve uygun malzeme seçiminin hem yazılımda hem de arıza bulmada önemini göstermiş, uygun malzeme ve yazılım ile problemlerin azaldığı gözlemlenmiştir.

- Setin montaj aşamasında çokça çeşitli el aletlerine ve mekanik aksama ihtiyaç duyulmuştur. Setin yapımında yazılım ve elektrik-elektronik bilgisi ve becerisi yanında mekanik bilgi ve becerinin de olması gerektiği görülmüştür.
- Setin montajı süresince geçmişteki bilgi, beceri ve deneyimin yeterli olmasından dolayı fazla problemle karşılaşılmamıştır. Problemler genellikle mekanik kısımlarda karşımıza çıkmıştır. Döner tablanın kayış kasnak mekanizmasında görülen titreşim sorunu, mekanik işçilik ve motor hız seviyesini ayarlayarak giderilmeye çalışılmıştır. Kayış kasnak sistemlerinde olağan olan bu problem, yapılan rulman müdahalesi ve diğer mekanik onarımlarla minimize edilmiştir.
- Döner tablanın step motorunun kontrolü için step motor kataloğu incelenmiştir. Kablo bağlantıları motor sürücüsü kitapçığında yararlanılarak PLC ve set ile irtibatlandırılmıştır. Yanlış bağlantı sonucu motorun hareket etmediği görülse de sonradan fiziksel kablo bağlantı hatası tespit edilerek ve problem giderilmiştir.
- Motor sürücüden gelen INP (in position) sinyalin düşük seviyede olmasından dolayı bir MOSFET devresi hazırlanmıştır. Düşük sinyal ile MOSFET iletime geçirilmiş ve elde edilen yüksek sinyal PLC girişine verilmiştir.
- PLC seçiminde komütatör anahtar yerine kontaktör devresi kullanılmıştır. Bu sayede atölye derslerinde verilen kontaktör eğitimi de set üzerinde uygulamalı olarak verilebilir hale getirilmiştir.
- Setin konveyör süren motor sürücüsündeki karşılaşılan arıza, optokuplör malzemesinin değişmesiyle giderilmiştir.
- Setin buton paneli, dokunmatik ekran tutacağı ve sensör tutucuları için pleksi malzeme kullanılmıştır. Tasarıma göre okulda lazer kesim yapılmıştır. Konveyör ürün hizalama aparatı ise okul imkanları ile 3D yazıcı kullanarak imal edilmiştir.
- Elektropnömatik sistem için kullanılan silindir ve pistonlar montajlanırken döner tablaya uyumlu ve ürün yerleştirme için uygun mesafede olmasına özen gösterilmiştir.
- Tüm kablolar ve hortumlar, kablo kanalları içinden geçirilerek setin iş sağlığı ve güvenliği bakımından güvenilir olması ve estetik açısından güzel görünmesi sağlanmıştır.

- Sette kullanılacak dört çeşit ürün (kısa ve uzun plastik, kısa ve uzun metal) torna ve freze kullanılarak hazırlanmıştır. Tablaya yerleşirken yaşanan zorluk, ürünlerin çaplarının küçültülmesiyle (ürün çapı 28mm) çözülmüştür,
- Mavi renkli parlak plastik ürün kullanıldığında konveyör başındaki sensörlerin ürünü algılamakta zorluk çektiği gözlemlenmiştir. Beyaz renkli mat derlin plastik malzeme ise sorun çıkarmamıştır.
- Döner tablanın en uygun dönüş açısı, denemeler sonunda atMotion programı ile istenilen değere getirilmiştir.
- Setin ağır olması nedeniyle yatay ve dikey yüzeyler, metal parçalarla birleştirilerek güçlendirilmiştir.
- Setin arkasından çıkan vida uçları spiral taş kullanılarak yüzeyle sıfırlanmıştır. Böylelikle olası el sürtmelerinden kaynaklanacak i kazaların önüne geçilmiştir.

4.2. Programlama Sürecindeki Bulgular

Setin tamamlanmasına kadar geçen yazılım sürecinde aşağıdaki bulgulara rastlanmıştır:

- PLC yazılımı grafcet diyagramına göre yapılmıştır. Grafcet diyagramındaki eksiklikler, PLC programına yansımış, gereken düzenlemeler sonucunda uygun zamanlama ve algoritmik değişikliklerle PLC yazılımı da tamamlanmıştır.
- SCADA yazılımı tasarımında PLC .NET kütüphanelerinin kullanılması için programlama kitapçıkları incelenmiş, gerekli “.dll” dosyaları yazılıma eklenerek PLC komutları kullanılmıştır.
- Form tasarımında birkaç farklı tasarım yapılmışsa da son haline karar verilmiş ve bu şekilde tezde yer almıştır.
- SCADA benzetiminde ilk başta tüm giriş ve çıkışlara yer verilmiştir. Fakat benzetimin yavaşladığı gözlemlenmiştir. Bunu düzeltmek için simülasyonda kullanılan PLC giriş ve çıkışları azaltılmıştır. Kullanılan timer nesnesinin interval değeri 1ms (en düşük değer) yapılmış olsa da PLC ve yazılım arasındaki iletişim arasında USB bağlantıda yaklaşık 3 saniyelik bir gecikme olmaktadır. Örneğin bir sensörün ürünü görmesinden 3 sn sonra ekranda sensör değeri görülebilmektedir. Ethernet iletişimde problem ortadan kalkmıştır.

- Setin üzerinde USB ile çalışırken anlık benzetim çalıştığında programın işleme hızında yavaşlama olduğu gözlemlenmiştir. En verimli kullanımın, benzetim olmadan kontrol veya en azından belirli aralıklarla simülasyonu çalıştırmak olduğu gözlemlenmiştir. Ethernet iletişimde bu sorun ortadan kalmıştır.
- Benzetimi setin ilk çalıştırılmasıyla aktif etmek şart değildir. Örneğin tablada 5 ürün varken benzetimi çalıştırdığımızda tablada 5 ürün olduğu benzetimde izlenebilmektedir.
- Veri tabanına kayıt ile ürünün tespiti arasındaki zaman uyumunun kabul edilir seviyede olduğu görüldü. Fakat grafik gösterimi anlık değil istenildiğinde buton ile güncellenmektedir.
- Tabla dolu iken yeni ürün geldiğinde sistem ürünü aynı banttı geri göndermektedir. Tablaya taşımamaktadır Tabla dolduğunda hem setin üzerindeki ışıklarla hem de benzetim ekranında alarm ile tablanın boşaltılması istenmektedir. Sistemin yeniden başlaması için buton panelindeki RESET butonuna basmak yeterlidir.

4.3. PTÖ Sürecindeki Bulgular

PTÖ sürecinde ise aşağıdaki bulgulara rastlanmıştır:

- Öğrenciler PTÖ planını, rubriklerini ve formlarını gördüklerinde şaşırmışlardır. Öğretmenlerinin bu denli planlı ve programlı olarak karşılımlarına çıkmalarının, onların da PTÖ sürecine motivasyonunu artırmıştır.
- PTÖ değerlendirmesinin sadece ürün odaklı değil süreç odaklı olması öğrenci kaygılarını azaltmıştır.
- Öğrenciler set üzerinde örnek çalışmayı incelemiş, gereken uygulamaları yapmış ve PTÖ sürecine hazır hale gelmişlerdir.
- Öğrenciler gruplara ayrılmış, ilk toplantılarını yapmış, görev paylaşımı ve iş zaman çizelgelerini hazırlamış, sanal şirketlerini kurmuş olarak sürece başlamışlardır. Bu aktivitelerin öğrencilerde olumlu karşılandığı gözlemlenmiştir.
- Proje fikri ve sürüş sorusu öğrencilere verilmiş ve öğrenciler gruplar halinde set üzerinde çalışmışlardır. Her grup ethernet yoluyla sete

bağlanmış ve kendi uygulamalarını set üzerinde gerçekleştirmişlerdir. Setin PTÖ için uygun olduğu gözlemlenmiştir.

- PTÖ sürecinde öğretmen bir rehber olarak konumlandırmıştır. Öğretmen iyi bir gözlemci olmalıdır. Öğrenciler iyi gözlemlenmediklerinde zaman kullanımı ve disiplinli çalışma konularında zaman zaman olumsuz davranışlarda bulundukları gözlemlenmiştir. O yüzden öğretmen bu süreçte önemli rol almaktadır. Öğrencilerin başında her zaman bulunmalı fakat çok müdahil olmamalıdır.
- Öğrenciler süreç sonunda çalışmalarını rapor etmiştir. Sunum yaparak çalışmalarını sunmuşlardır. Sunumlarda öğrencilerin sunum becerilerinin geliştiği gözlemlenmiştir.
- Sunumlar sonrasında öz değerlendirmelerini ilgili formlar üzerinden yapmışlardır. Öğrencilerin vicdanlı ve adil olarak değerlendirme yapmaya çalıştıkları gözlemlenmiştir. Duygusallık tamamen ortadan kalkmamıştır.
- Son olarak PTÖ sürecini değerlendirmişlerdir. Her öğrenci PTÖ modelinin geleneksel öğrenme modellerine göre daha verimli ve eğlenceli olduğunu belirtmişlerdir.
- PTÖ sürecinin öğrencilerin 21 yy. becerilerini geliştirdiği gözlemlendi. Özellikle problem çözme, işbirlikçi çalışma, liderlik, sorumluluk, sunum, eleştirel düşünme ve dijital okuryazarlık gibi becerilerinin gözle görülür bir gelişme gösterdiği gözlemlenmiştir.
- PTÖ sürecinin iyi planlanmasının önemli olduğu gözlemlenmiştir. İyi planlanmamış bir PTÖ süreci uzun zaman alabilecektir. Amaçlara ulaşmada zorluklar yaşanabilecektir.
- 4 haftalık bir PTÖ sürecinin set açısından uygun olduğu görülmüştür. 4 haftalık süreç 6 haftaya da çıkarılabileceği düşünülmektedir.
- PTÖ sürecinde gözlemlenen tüm bulgular değerlendirildiğinde imal edilen setin endüstriyel otomasyon teknolojileri alanı dersleri için verimli bir eğitim materyali olduğu kanaatine varılmıştır.

Bölüm 5

5. Sonuçlar ve Öneriler

Bu çalışmada, uzaktan (online) ve yüz yüze eğitimlerde kullanılmak üzere proje tabanlı öğrenim müfredatlı endüstriyel otomasyon eğitimi için bir eğitim setin imalatı ve programlanmasına odaklanıldı.

Öğrencileri, “Milli Sanayi Hamlesi” ve “Endüstri 4.0” vizyonlarına hazırlamak için üretim temelli eğitimlere önem verilmelidir. Piyasadaki yerli olmayan firmaların eğitim setleri oldukça maliyetlidir. Her okul bu maliyetli setlere sahip olamamaktadır. Bu çalışma piyasadaki muadil yerli olmayan setlere göre daha ucuzca imal edilmiştir.

Bu çalışma ile endüstriyel otomasyon ürünlerini kullanarak, özgün tasarımlarla kendi eğitim setlerimizi imal edebileceğimiz tecrübe edildi.

Set üzerinde aşağıdaki konular üzerinden eğitimler verilebilmektedir:

- ✓ Grafcet Diyagramları
- ✓ PLC (iki farklı model) programlama
- ✓ Elektrik-Elektronik malzemeler ve kullanımları
- ✓ Elektropnömatik
- ✓ DC motor ve adım motoru kontrolleri
- ✓ Sensörler
- ✓ Konveyör üzerinde ürün ayırıştırma
- ✓ Döner tabla üretimi ve kontrolü
- ✓ Görsel ve nesne tabanlı programlama
- ✓ SCADA programlama
- ✓ Görüntü kütüphaneleri kullanma
- ✓ Veri tabanı uygulamaları
- ✓ Dokunmatik panel uygulamaları
- ✓ PTÖ süreci ve eğitim materyalleri

Setin eğitim içeriği MEB endüstriyel otomasyon teknolojileri alanı müfredatına uygundur. Set ile aşağıdaki sınıf dersleri verilebilmektedir:

- ✓ Atölye (9. Sınıf)
- ✓ Atölye (10. Sınıf)
- ✓ Sensörler (10. Sınıf)
- ✓ Atölye (11.Sınıf)
- ✓ Mekanizmalar (11. Sınıf)
- ✓ Bilgisayarda Modelleme (11. Sınıf)
- ✓ Endüstriyel Görsel Programlama (12. Sınıf)

Bu çalışma mesleki eğitimde PTÖ temelli mesleki eğitim için başvuru kaynağı olacağı düşünülmektedir. Meslek öğretmenleri, PTÖ metoduyla bir mesleki eğitim planının nasıl hazırlanacağını, hangi formların ne şekilde olacağını, değerlendirme ölçekleri olan rubriklerin örneklerini de bu çalışmada bulabileceklerdir.

Geleneksel eğitim öğretim metotlarının ötesinde öğrenci merkezli proje tabanlı öğrenme metoduyla öğrencilerimizin tanışması gerekiyor. Bu onların 21.yy becerilerinin gelişmesini de sağlayacaktır.

Bu çalışma İzmir Bornova Mazhar Zorlu MTAL atölyesinde öğrencilerin de gözlemleri arasında gerçekleştirilmiştir. Öğrenciler, projenin her safhasını takip etmişlerdir. Gerektiğinde kablo soyma ve kablo kanallarının montajı gibi küçük el işçiliklerinde emekleri geçmiştir.

4 haftalık PTÖ uygulaması öğrenci gruplarında deneyimlenmiş, öğrenciler üzerindeki etkisi ve verimi gözlemlenmiştir. Setin amacına uygun olarak çalıştığı görülmüştür. 12 kişilik öğrenci grubunun 4 hafta süreyle PTÖ metoduyla set üzerinde uygulama geliştirmeleri, öğrencilerin mesleklerine olan ilgi, sevgi ve gelecek hedeflerini yükseltmiştir.

PTÖ rubrikleri öğrenci ve öğretmen tarafından doldurulmuş, PTÖ temelli öğrenimin geleneksel öğrenme metoduna göre mesleki eğitimde daha verimli olduğu gözlemlenmiştir.

Bu çalışma, MEB endüstriyel otomasyon teknolojileri alanı müfredatına uyumlu olarak imal edilmiş ve programlanmıştır. Setin hem online hem yüz yüze eğitim için uygun olması, iki farklı PLC barındırması, Grafcet diyagramlarını kullanması, PTÖ müfredatına sahip olması, üzerinde kullanılan malzemelerin çok çeşitli ve endüstride kullanılan malzemelerden seçilmesi, yazılan SCADA yazılımının “.NET” platformunda Visual C# nesne tabanlı programlama ile yazılması, SQL veri tabanı ve görüntü kütüphaneleri gibi birçok programlama kütüphanesini ve metodunu kullanması, PTÖ müfredatına, planına ve rubriklerine sahip olması, sete değer katmış, setin kullanışlı ve verimli bir eğitim aracı olmasını sağlamıştır.

Son olarak imal edilen setin geliştirilmeye açık olduğunu da belirtmek gerekir. Sete diğer endüstriyel ürünler eklenebilir, mevcut programlama dilli ve endüstriyel ürünler değiştirilebilir (örnek olarak başka bir PLC bağlanabilir), setin yazılımı (SCADA) geliştirilebilir, sete mobil uygulama, web tabanlı kontrol, görüntü işleme ile ürün tanıma ve yapay zekâ uygulamaları eklenebilir, set üzerinde sanal ve artırılmış gerçeklik (AR) teknolojileri kullanılabilir. Set diğer mevcut başka setlerle veya endüstriyel robotlarla birlikte kullanılabilir, endüstriyel ethernet, gömülü kartlar, endüstriyel IoT cihazları ve endüstriyel ağ teknolojileri ile geliştirilebilir. İmal edilen setin yıllarca endüstriyel otomasyon eğitimlerinde hizmet edeceği düşünülmektedir.

Kaynaklar

1. Allen, I. E., & Seaman, J. (2014). Grade change tracking online education in the united states. Babson Survey Research Group and Quahog Research Group, LLC.
2. Palvia, S., Aeron, P., Gupta, P., Mahapatra, D., Parida, R., Rosner, R., & Sindhi, S.(2018). Online education: Worldwide status, challenges, trends, and implications. Journal Of Global Information Technology Management, 21(4), 233-241. <https://doi.org/10.1080/1097198X.2018.1542262>
3. Gunawardena, C.N., & McIsaac, M.S. (2004). Distance Education. In David H.Jonassen (Ed.) Handbook Of Research on Educational Communications and Technology, Second Edition. A Project of the Association for Educational Communications and Technology. Lawrence Erlbaum Associates, Publishers, Mahwah, New Jersey London. P. 355– 397.
4. Ma, L., & Lee, C. S. (2021). Evaluating the effectiveness of blended learning using the ARCS model. Journal of computer assisted learning, 37(5), 1397-1408. <https://doi.org/10.1111/jcal.12579>
5. Heinichi, R., Molenda, M., Russel, J., & Smaldino, S. (2002). Instructional media and technologies for learning (7th ed.). Englewood Cliffs: Prentice Hall.
6. Wilson, E.K., & Marsh II, G.E. (1995). Social studies and the internet revolutions. Social Education, 59, 198-202
7. Vatansever-Bayraktar, H. (2015). Proje Tabanlı Öğrenme Yaklaşımı, Uluslararası Sosyal Araştırmalar Dergisi, 8(37), 709-718. <https://www.sosyalarastirmalar.com/articles/project-based-learning-approach.pdf>
8. Erdem, M. ve Akkoyunlu, B. (2002). İlköğretim sosyal bilgiler dersi kapsamında beşinci sınıf öğrencileriyle yürütülen ekiple proje tabanlı öğrenme üzerine bir çalışma. İlköğretim Online, 1(1). 2-11.
9. Demirel, Ö. (Ed.) (2019). Eğitimde yeni yönelimler. APegem Akademi.
10. Korkmaz, H. ve Kaptan, F. (2001). Fen eğitiminde proje tabanlı öğrenme yaklaşımı. Hacettepe Üniversitesi Eğitim Fakültesi Dergisi. 20, 193-200.

11. Moursund, D. G. (2003). Project-based learning: using information technology. (2nd ed.). International Society for Technology in Education: Eugene, OR.
12. Simkins, M. (1999). Project-based learning with multimedia. Thrust for Educational Leadership, 28(4), 10-13.
13. Korkmaz, H. (2002). Fen eğitiminde proje tabanlı öğrenme yaklaşımının ilköğretim öğrencilerinin yaratıcı düşünme, problem çözme ve akademik risk alma düzeylerine etkisi (Tez No. 313511) [Doktora Tezi, Hacettepe Üniversitesi]. YÖK Ulusal Tez Merkezi.
14. Yurtluk, M. (2003). Proje tabanlı öğrenme yaklaşımının matematik dersi öğrenme süreci ve öğrenci tutumlarına etkisi (Tez No. 132829) [Yüksek Lisans Tezi, Hacettepe Üniversitesi]. YÖK Ulusal Tez Merkezi.
15. Williams, D.A. (1998). Documenting children's learning: assesment and evaluation in the project approach. Unpublished Master's Thesis, University of Alberta. Edmonton/ Alberta, Canada.
16. PLC TANIM VE TÜRLERİ [İnternet] 2023 [erişim tarihi:27.10.2023] <https://mekatronikdunyam.wordpress.com/2017/02/13/plc-tanim-ve-turleri/>
17. File: Secuencial GRAFCET.PNG [İnternet] 2023 [erişim tarihi:27.10.2023] https://commons.wikimedia.org/wiki/File:Secuencial_GRAFCET.PNG
18. Konveyör [İnternet] 2023 [erişim tarihi:27.10.2023] <https://tesensors.com/tr/tr/applications/material-handling/conveyors>
19. Pnömatik Sistem Nedir? Nasıl Çalışır? Nerede Kullanılır? [İnternet] 2023 [erişim tarihi:27.10.2023] <https://www.maktoloji.com/2018/10/pnomatik-sistem-nedir-nasl-calsr-nerede.html>
20. Hidrolik Pnömatik Laboratuvarı [İnternet] 2023 [erişim tarihi:27.10.2023] <https://www.pau.edu.tr/mekatronikmuhendisligi/tr/sayfa/laboratuvarlar-7>
21. Step Motorlar [İnternet] 2023 [erişim tarihi:27.10.2023] <https://teslaakademi.com/step-motorlar>
22. STEP motor nedir, Nasıl çalışır ve Projelerde nasıl kullanılır. [İnternet] 2023 [erişim tarihi:27.10.2023] <https://www.mikrobotik.com/wp2/2022/06/04/step-motor-nedir-nasil-calisir-ve-projelerde-nasil-kullanilir-kutuphanesiz/>
23. Step Motor Nedir? Nasıl Çalışır? [İnternet] 2023 [erişim tarihi:27.10.2023] <https://www.mcu-turkey.com/step-motor-nedir-nasil-calisir/>

24. Universal Motors: Construction, Types, Working, and Applications? [İnternet] 2023 [erişim tarihi:27.10.2023] <https://www.electronicclinic.com/universal-motors-construction-types-working-and-applications/>
25. Shantz, D. (2005). Öğretmen Eğitiminde yenilikçi bir yöntem mi yoksa geleneksel bir anlayış mı? Ankara Üniversitesi Eğitim Bilimleri Fakültesi Dergisi, 38(2), 187-195.
26. Holm, M. (2011). A Review of the literature on effectiveness in prekindergarten through 12th grade classrooms. Rivier Academic Journal,7 (2).
27. English, M.C. ve Kitsantas, A. (2013). Supporting student self-regulated learning in problem- and project-based learning. Interdisciplinary Journal of Problem-Based Learning, 7 (2): 218-150.
28. Doymuş, K., Şimşek, Ü., & Şimşek, U. (2005). İşbirlikçi öğrenme yöntemi üzerine derleme: I. İşbirlikçi öğrenme yöntemi ve yöntemle ilgili çalışmalar. Erzincan Üniversitesi Eğitim Fakültesi Dergisi, 7(1), 59-83.
29. Kaşarcı, İ. (2013). Proje Tabanlı Öğrenme Yaklaşımının Öğrencilerin Akademik Başarı Ve Tutumlarına Etkisi: Bir Meta-Analiz Çalışması. Yüksek Lisans Tezi, Osmangazi Üniversitesi Eğitim Bilimleri Enstitüsü, Eskişehir.
30. Serttürk, M. (2008). Fen Öğretiminde Proje Tabanlı Öğrenme Yaklaşımının İlköğretim 7.Sınıf Öğrencilerinin Fen Başarısı Ve Tutumuna Etkisi. Yüksek Lisans Tezi, Sakarya Üniversitesi, Sakarya.
31. Kalaycı, N. (2010). Yükseköğretimde proje tabanlı öğrenmeye ilişkin bir uygulama projesi yöneten öğrenciler açısından analiz. Eğitim ve Bilim, 33(147), 85-105.
32. Millî Eğitim Bakanlığı (2020). Araştırma ve Uygulamalarıyla Proje Tabanlı Öğrenme.
https://yegitek.meb.gov.tr/meb_iys_dosyalar/2020_10/15171038_Arastirma_ve_Uygulamalariyla_Proje_tabanli_ogrenme_Makale.pdf, (22.02.2021)
33. Saracaloğlu, A. S., ve Çelik, B. (2018). Web tasarımı ve programlama dersi öğretiminde proje tabanlı öğrenme yöntemi kullanımının öğrenci görüşlerine göre değerlendirilmesi. Kastamonu Eğitim Dergisi, 26(2), 323-333.

34. Akkuş, İ., Özhan, U. ve Kan, A. (2019). Ortaokul öğrencileri için kodlamaya yönelik tutum ölçeği: Geçerlik ve güvenirlik çalışması. *Elementary Education Online*, 18(2), 837851.
35. Aytekin, A., Çakır, F. S., Yücel, Y. B., ve Kulaözü, İ. (2018). Geleceğe yön veren kodlama bilimi ve kodlama öğrenmede kullanılabilecek bazı yöntemler. *Avrasya Sosyal ve Ekonomi Araştırmaları Dergisi*, 5(5), 24-41.
36. Yünkül, E., Durak, G., ve Çankaya, S. (2018). Blok Tabanlı Yazılımların Kodlama Öğretiminde Kullanımı. H. Gür ve H. H. Şahan (Ed), *Bildiri Tam Metin Kitabı Proceeding Book*, (ss 533-548). Balıkesir: Uluslararası Necatibey Eğitim ve Sosyal Bilimler Araştırmaları Kongresi.
37. Kılıç, Zeynep (2004). Meslek Eğitiminde Projeye Dayalı Öğretimin Öğrencilerin Akademik Başarılarına ve Bilgilerinin Kalıcılığına Etkisi, Yüksek Lisans Tezi, Ankara: Gazi Üniversitesi Eğitim Bilimleri Enstitüsü
38. Kınık, Ayşegül (2004). Fen Bilgisi Dersinde Proje Çalışmalarının Öğrencilerin Bilim Anlayışına ve Bilimsel Süreçleri Algılamalarına Etkisi, Yüksek Lisans Tezi, İstanbul: Marmara Üniversitesi Eğitim Bilimleri Enstitüsü.
39. Binbaşıoğlu, Cavit (1986). Genel Öğretim Bilgisi Eğitim Programları Öğretim İlke, Yöntem ve Teknikleri, Ankara: Binbaşıoğlu Yayınevi
40. Gündüz, Sevim (2004). İlköğretim ve Lise Öğrencileri İçin Matematik Projeleri ve Sınıf Etkinlikleri, Toroslu Kitaplığı.
41. Wolk, Steven (2001). “What Should We Teach? The Benefits of Exploratory Time”, *Educational Leadership*. Vol. 59 (2), 56-59
42. Yurtluk, Makbule (2003). Proje Tabanlı Öğrenme Yaklaşımının Matematik Dersi Öğrenme Süreci ve Öğrenci Tutumlarına Etkisi, Yayımlanmamış Yüksek Lisans Tezi, Ankara: Hacettepe Üniversitesi Sosyal Bilimler Enstitüsü.
43. Dori, J. Yehudit ve TAL, T. Revital (2000). “Formal and Informal Collaborate Projects: Engaging In Industry With Environmental Awareness”, *Science Education*, 84(1), 95-113.
44. Zoller, Haifa Uri (1991). Problem Solving and the “Problem Solving Paradox” in Decision Making Oriented Environmental Education, InKeiny, S. and Zoller, U. (Eds), *Conceptual Issues in Enviromental Education*, New York: Peter Lang. Pub.

45. Hamurcu, H. (2000). “Okul Öncesi Eğitimde Fen Bilgisi Öğretimi Proje Yaklaşımı”, 4. Fen Bilimleri Eğitim Kongresi, Ankara.
46. Korkmaz, Hünkar (2002). Fen Eğitiminde Proje Tabanlı Öğrenmenin Yaratıcı Düşünme, Problem Çözme ve Akademik Risk Alma Düzeylerine Etkisi, Doktora Tezi, Ankara: Hacettepe Üniversitesi Sosyal Bilimler Enstitüsü
47. Demirel, Özcan (2005). Kuramdan Uygulamaya Eğitimde Program Geliştirme, Ankara: Pegema A Yayıncılık.
48. Yurtluk, Makbule (2005). Proje Tabanlı Öğrenme. Ö. Demirel içinde, Eğitimde Yeni Yönelimler. Ankara: Pegema Yayıncılık.

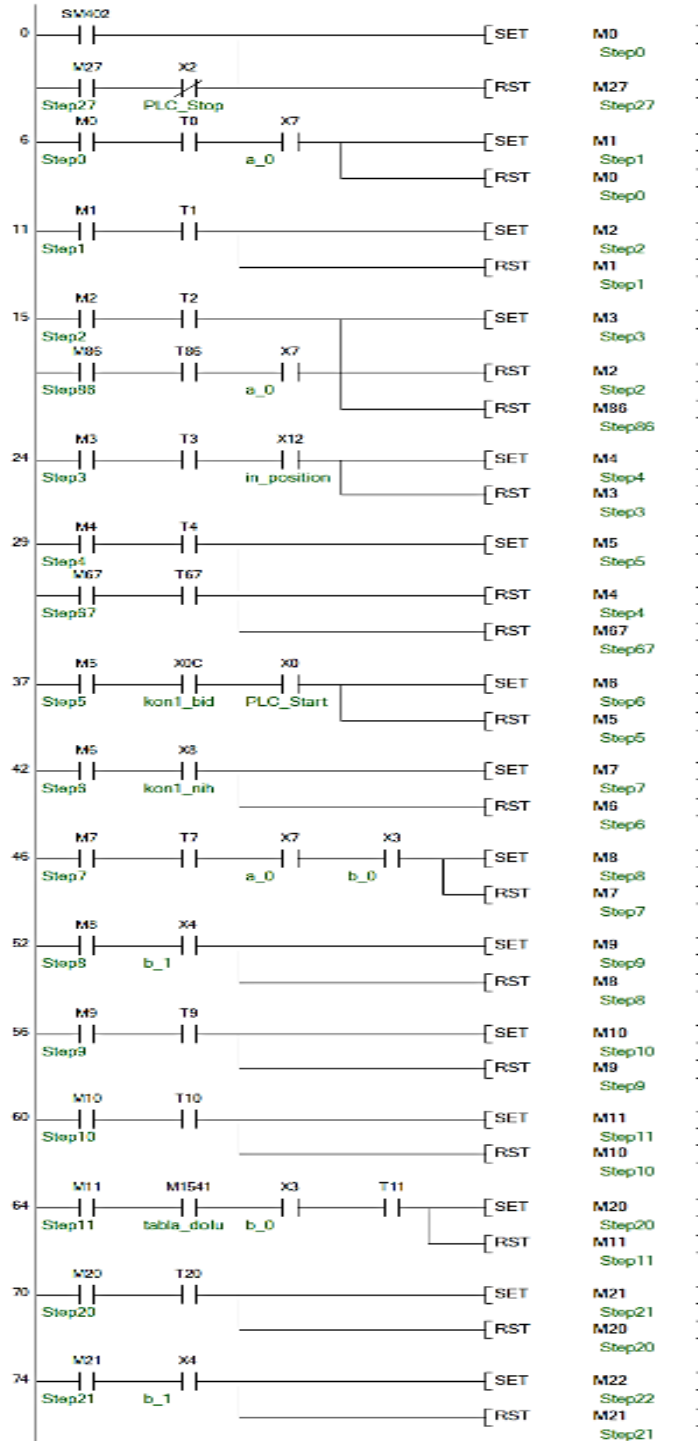


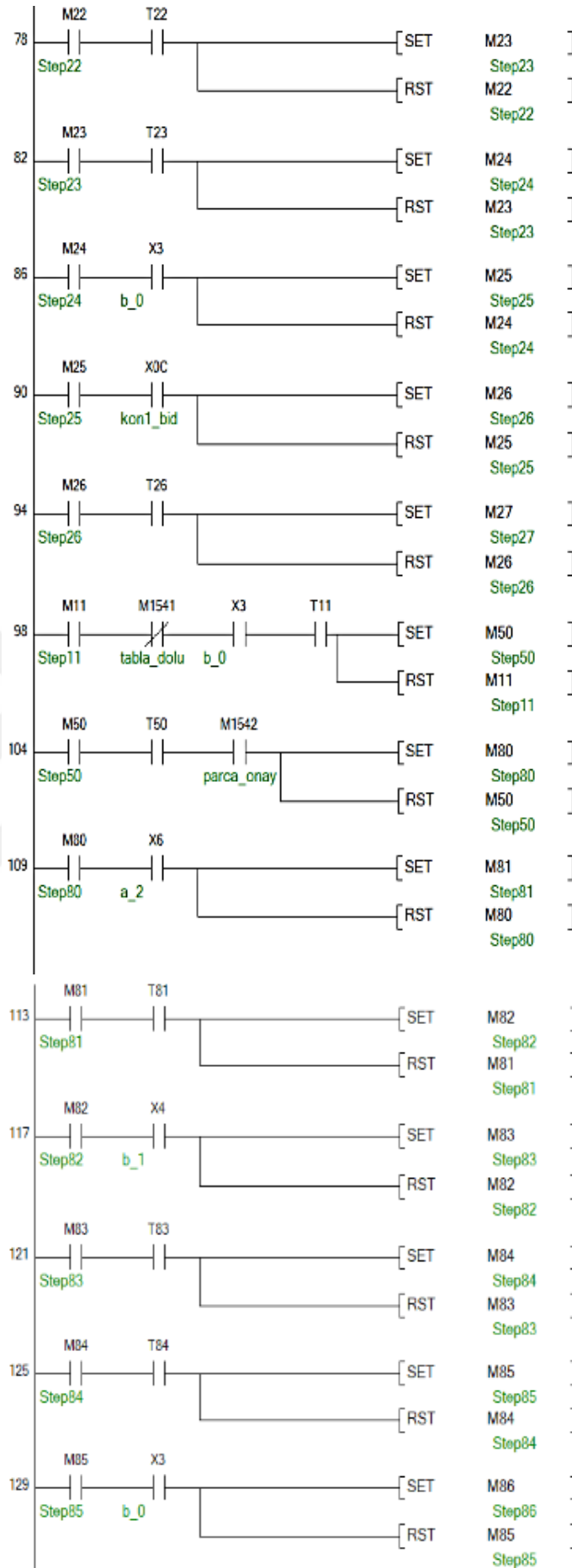
Ekler

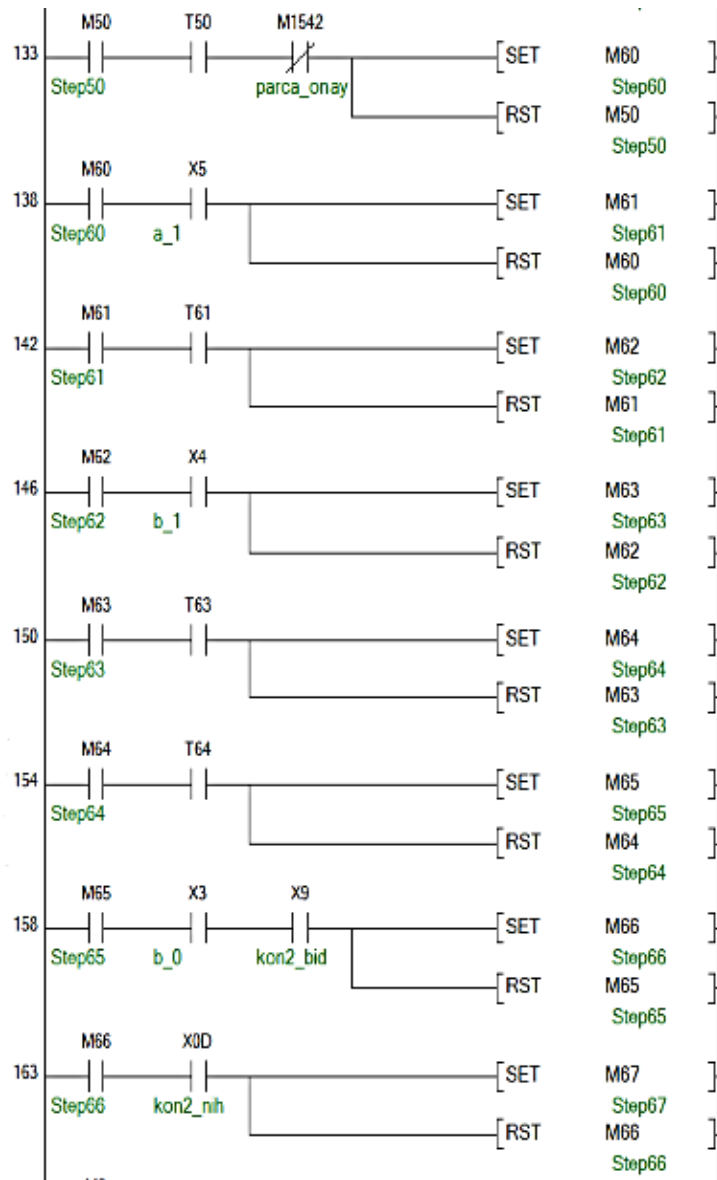
Ek A: PLC Ladder Programları

PLC-1 İÇİN LADDER PROGRAMI

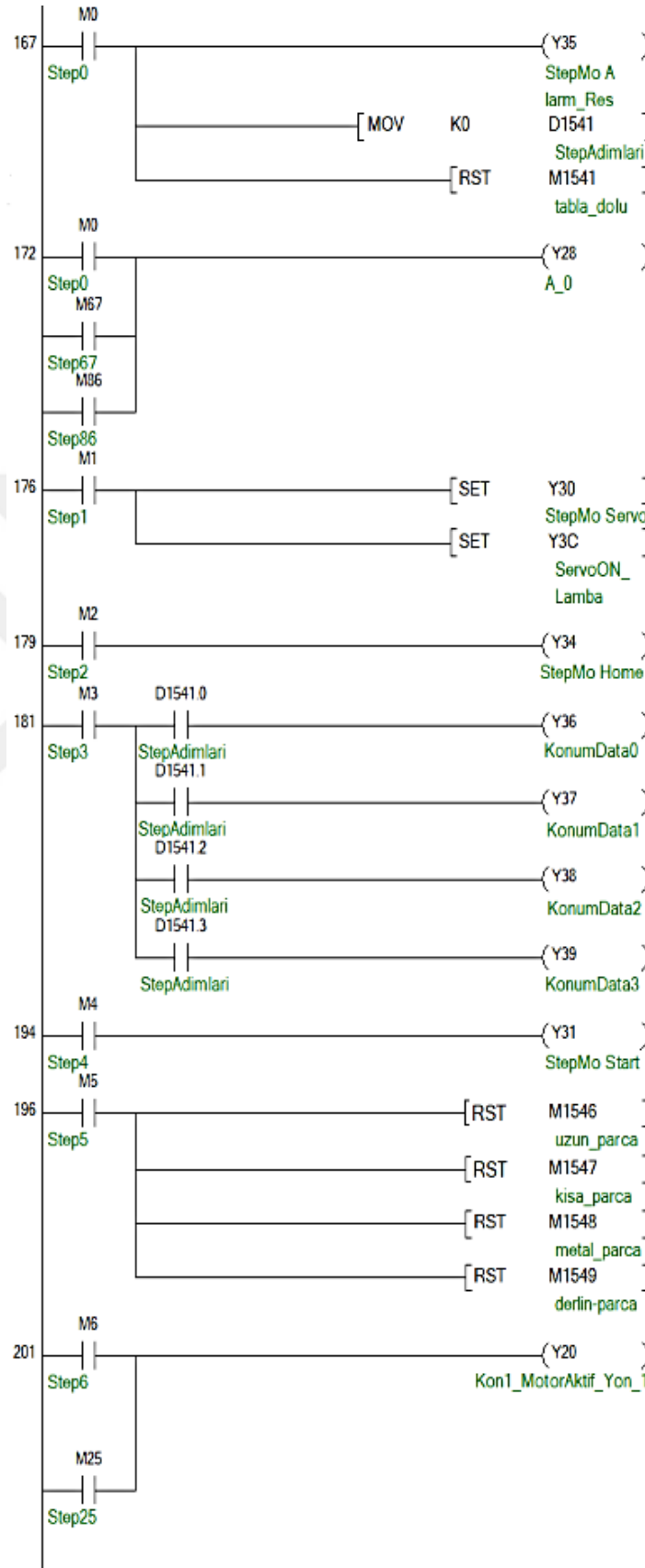
ADIMLAR(STEPS):

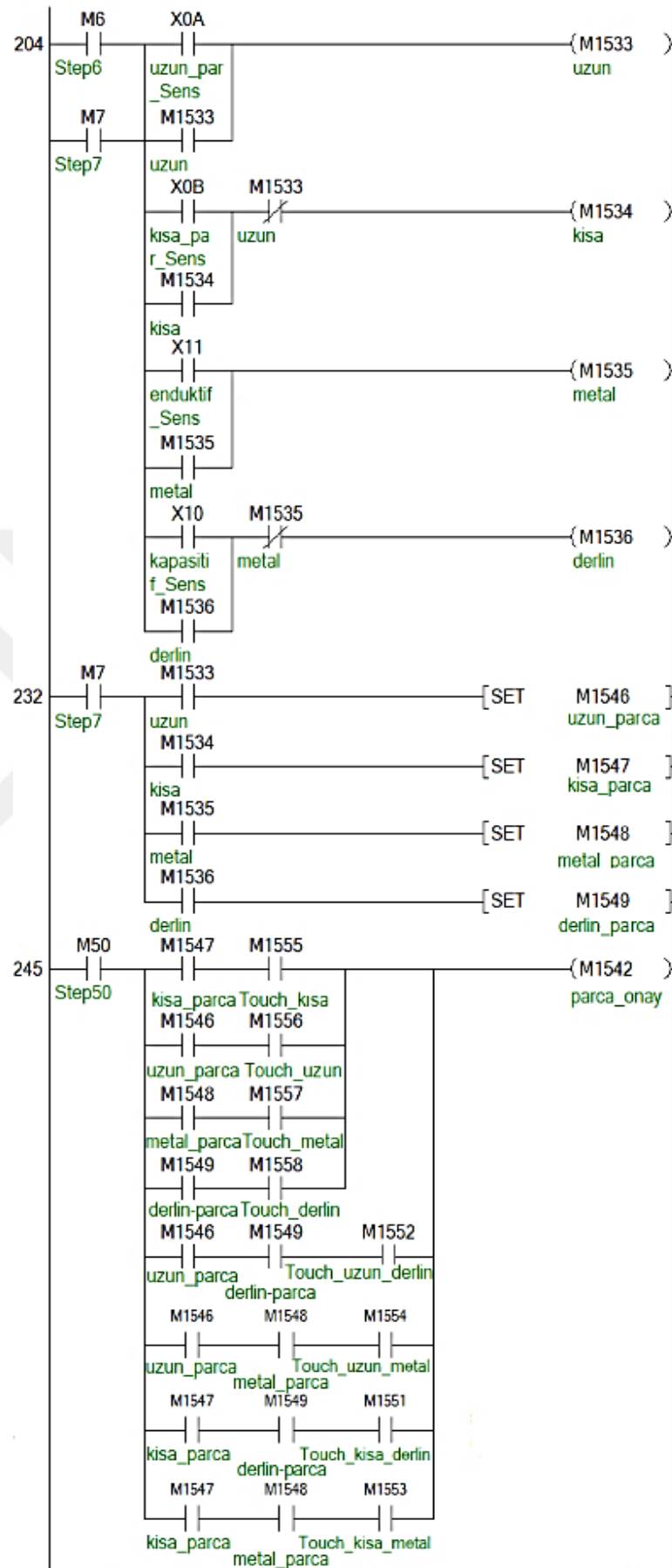


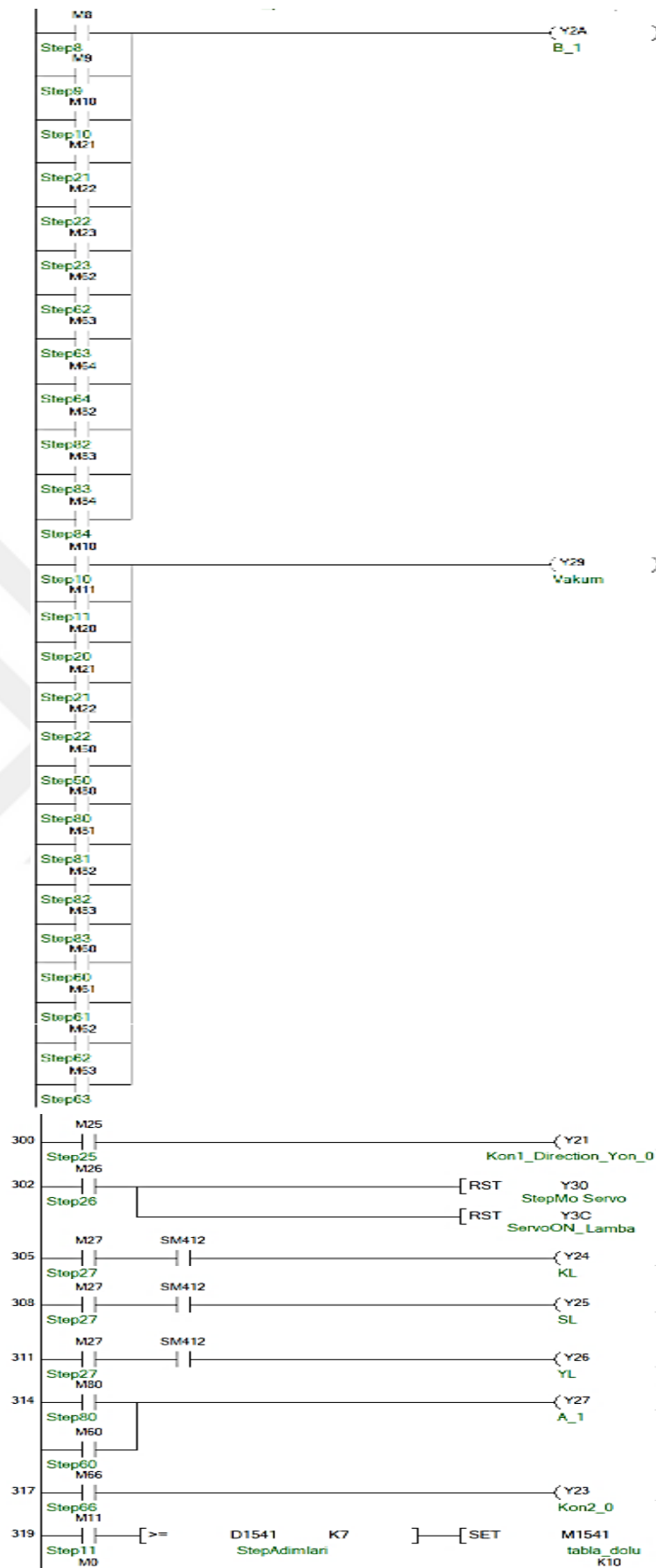




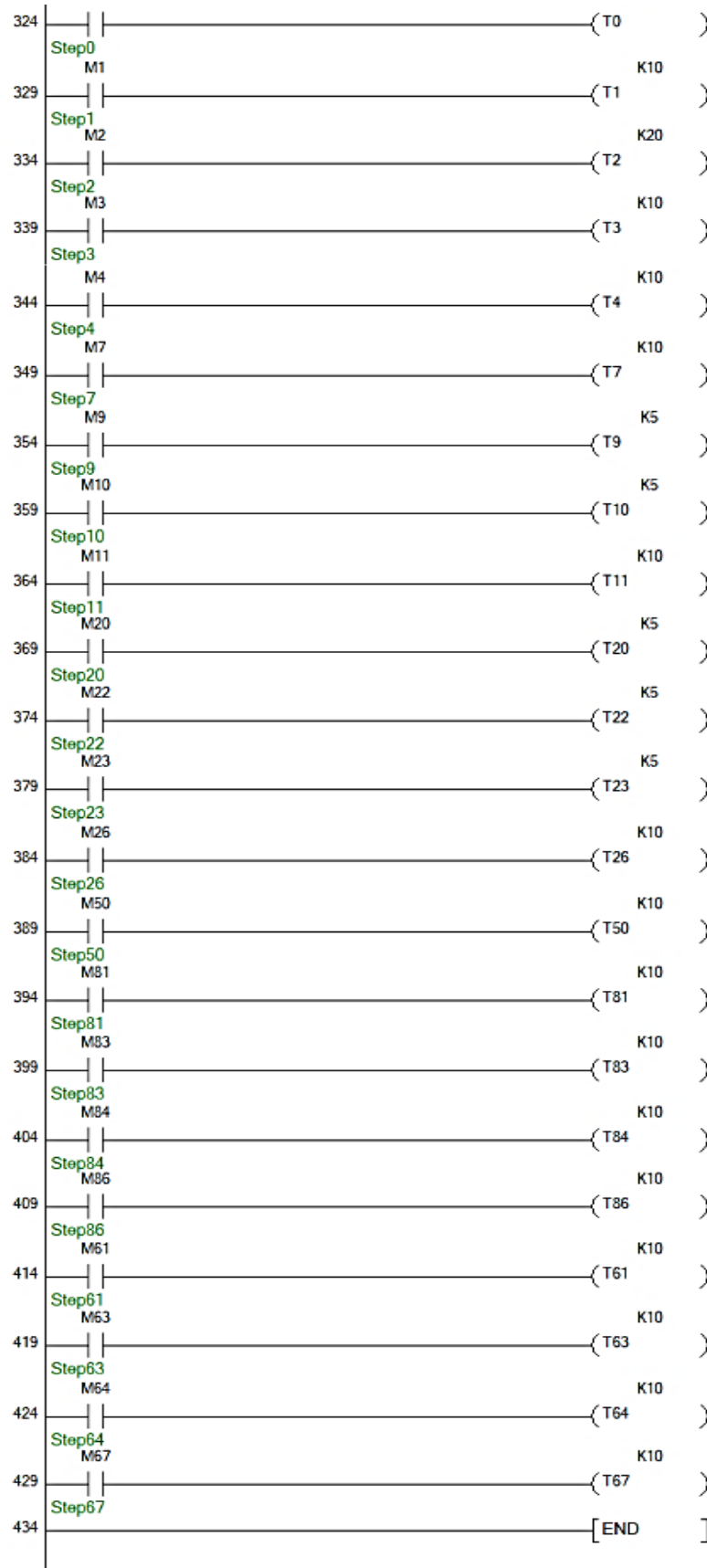
Eylemler (Actions):





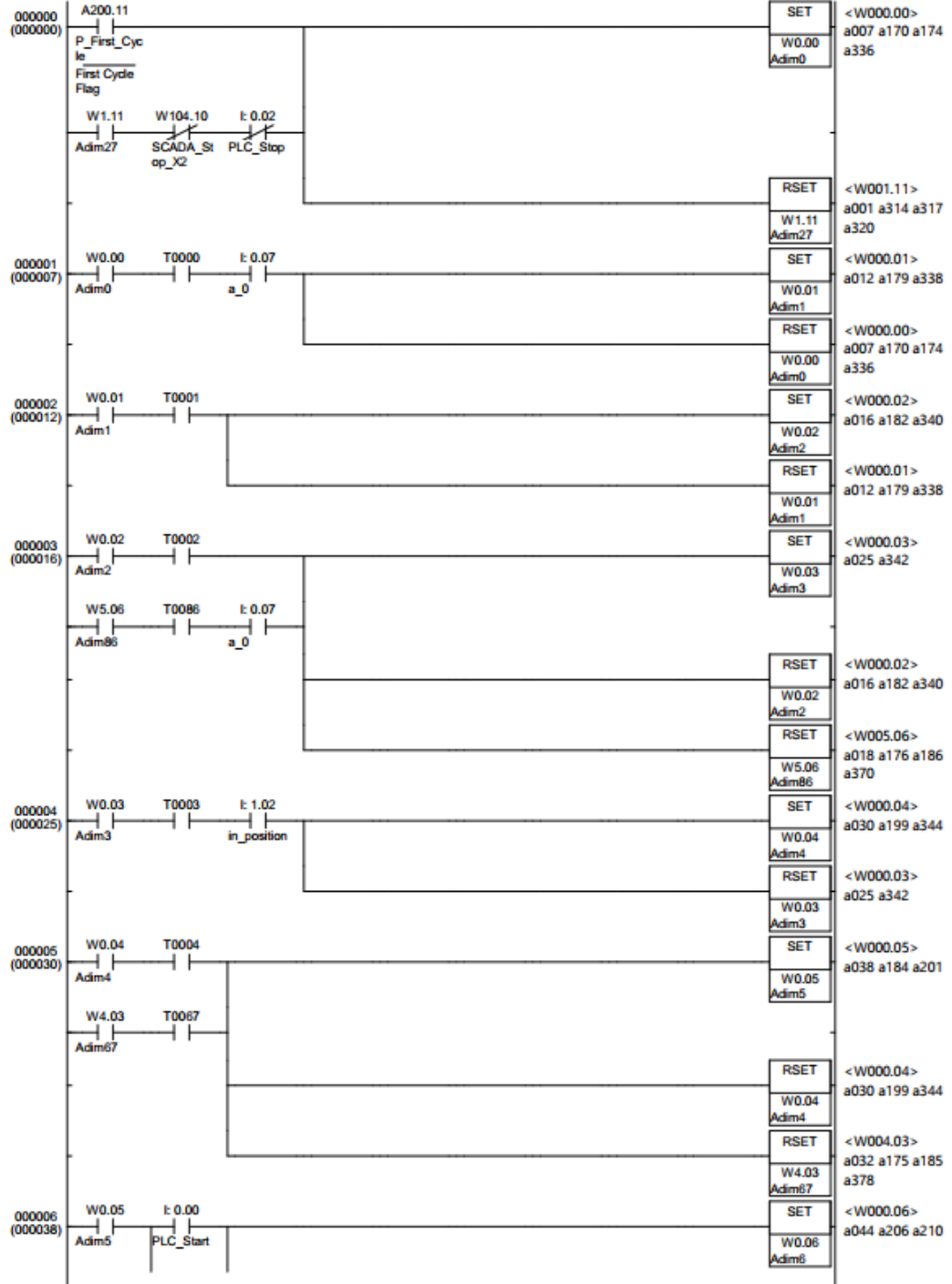


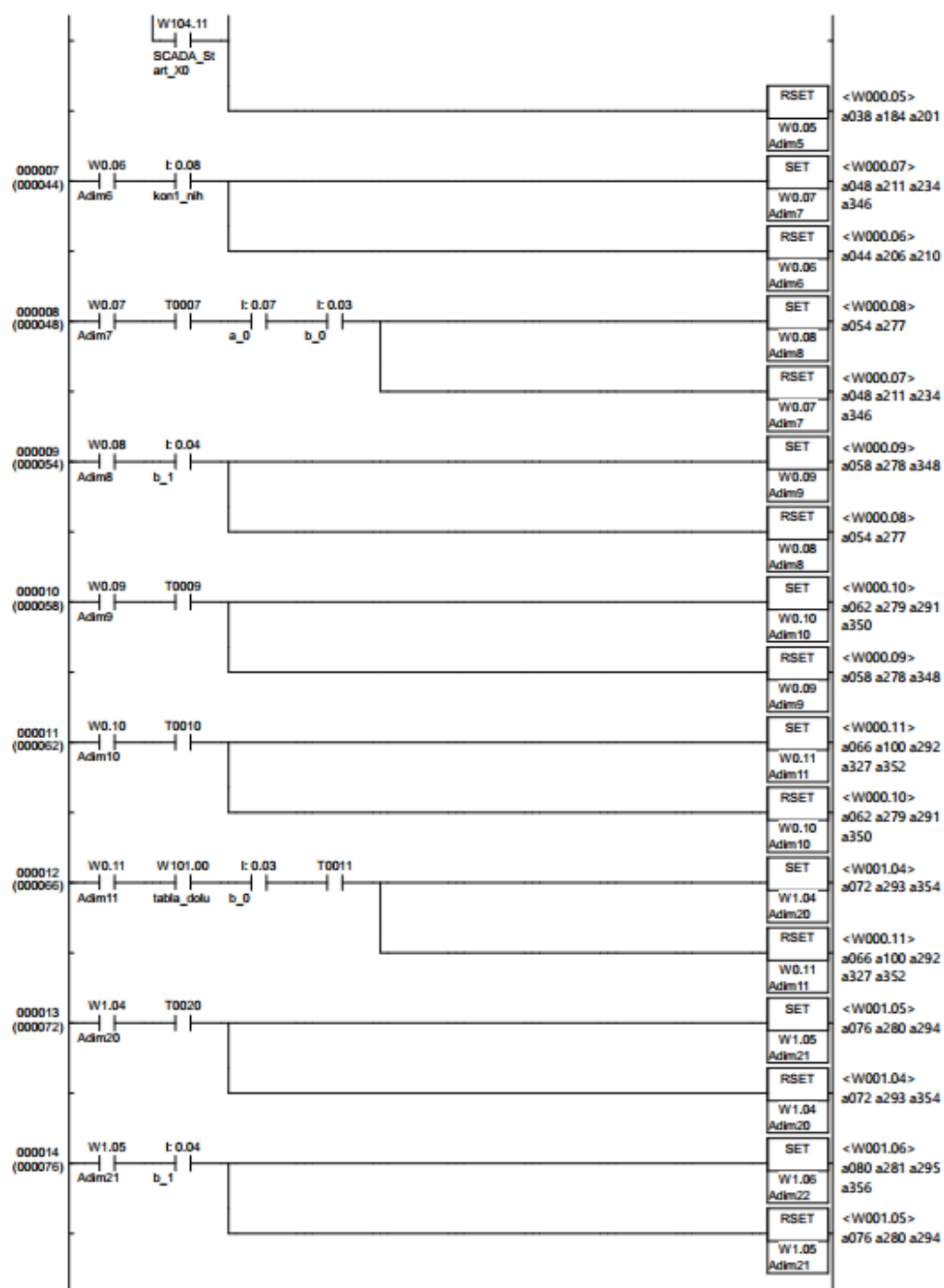
Zamanlayıcılar (Timers):

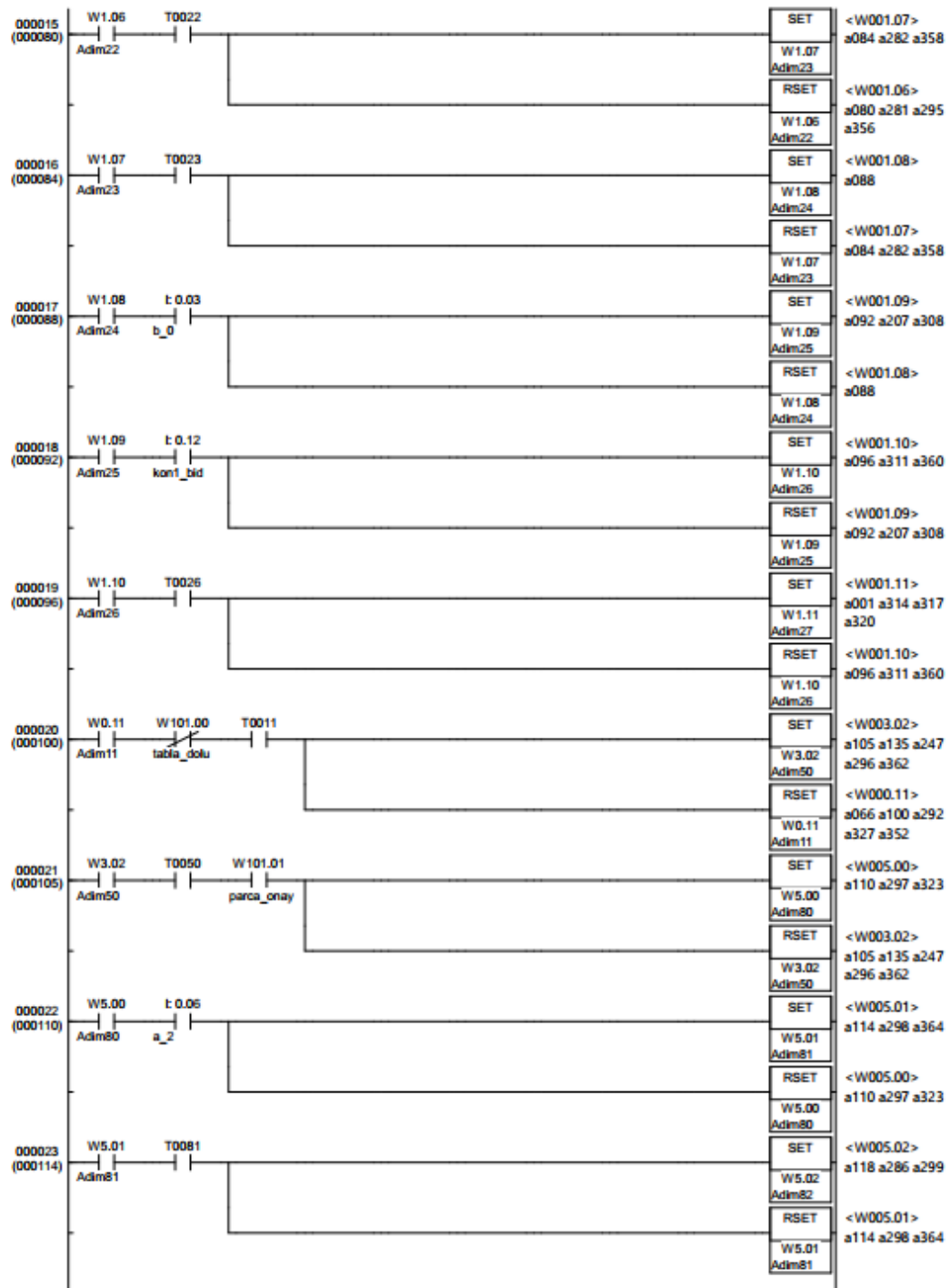


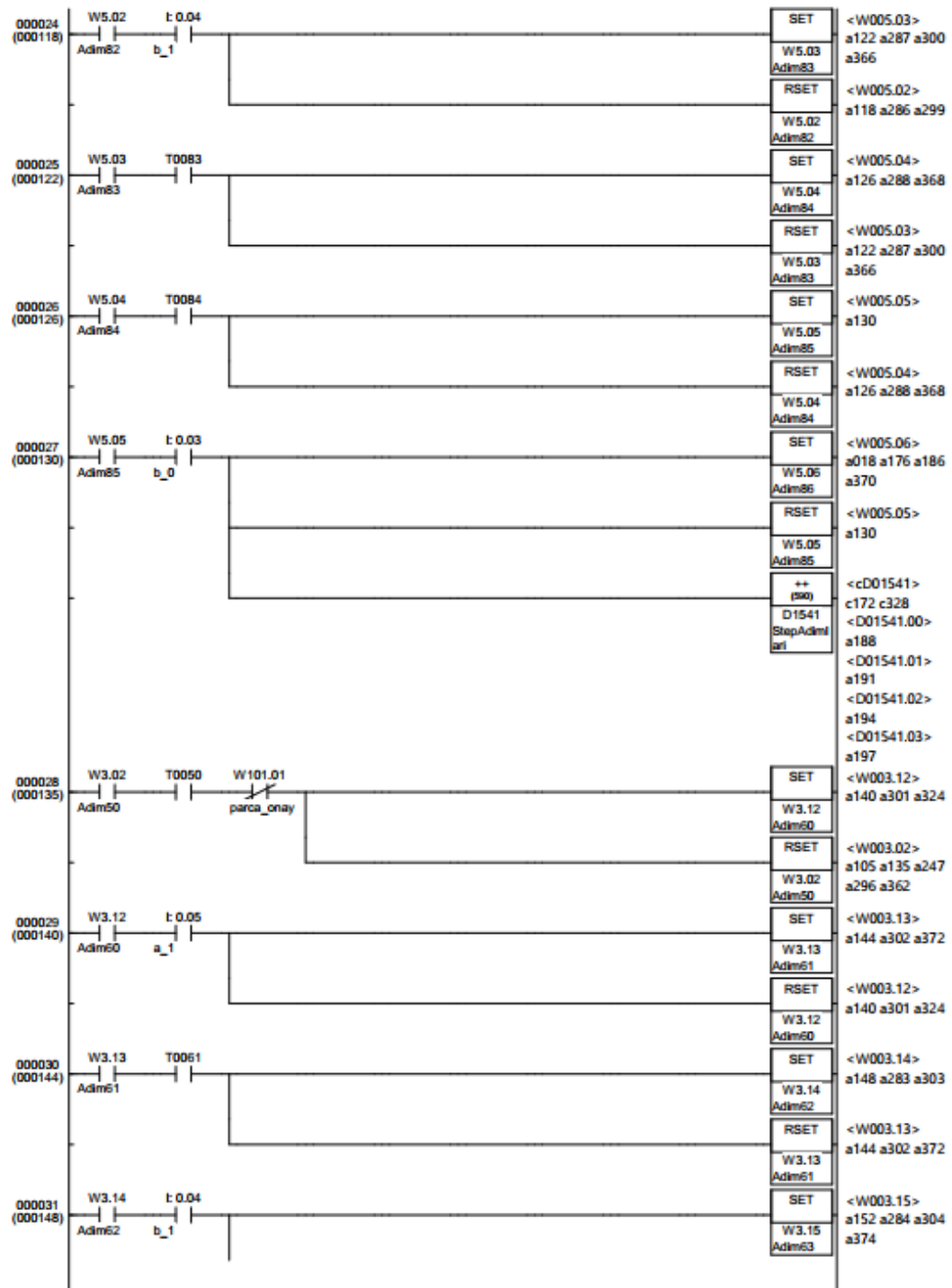
PLC-2 İÇİN LADDER PROGRAMI

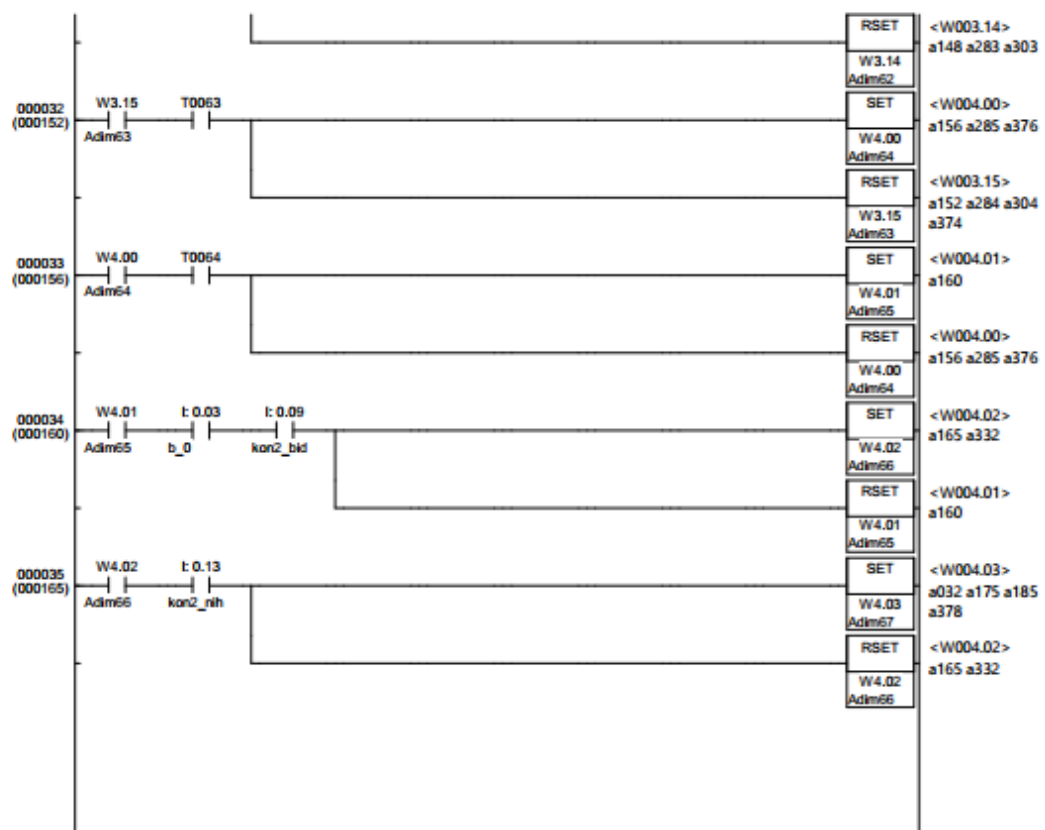
ADIMLAR(STEPS):



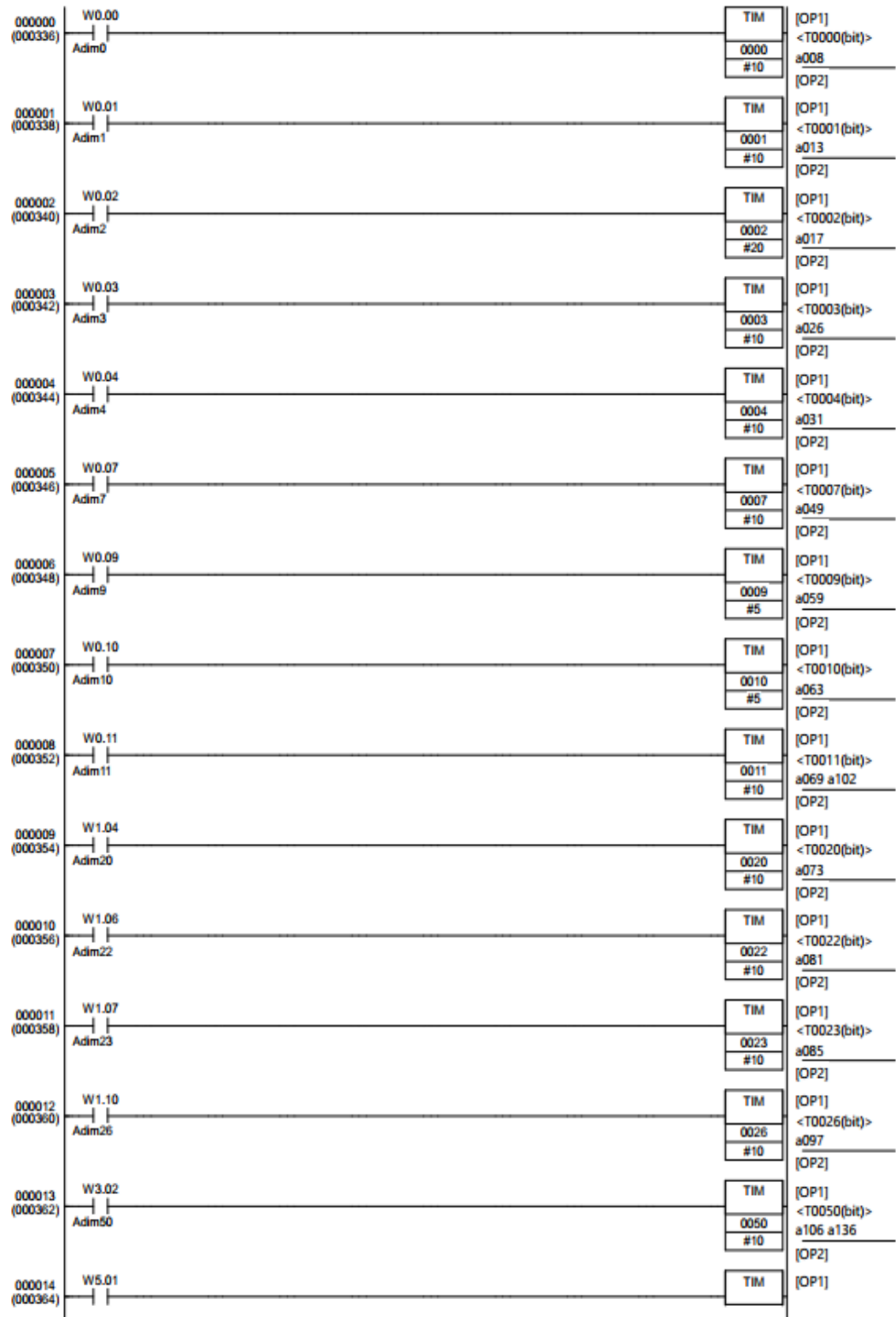


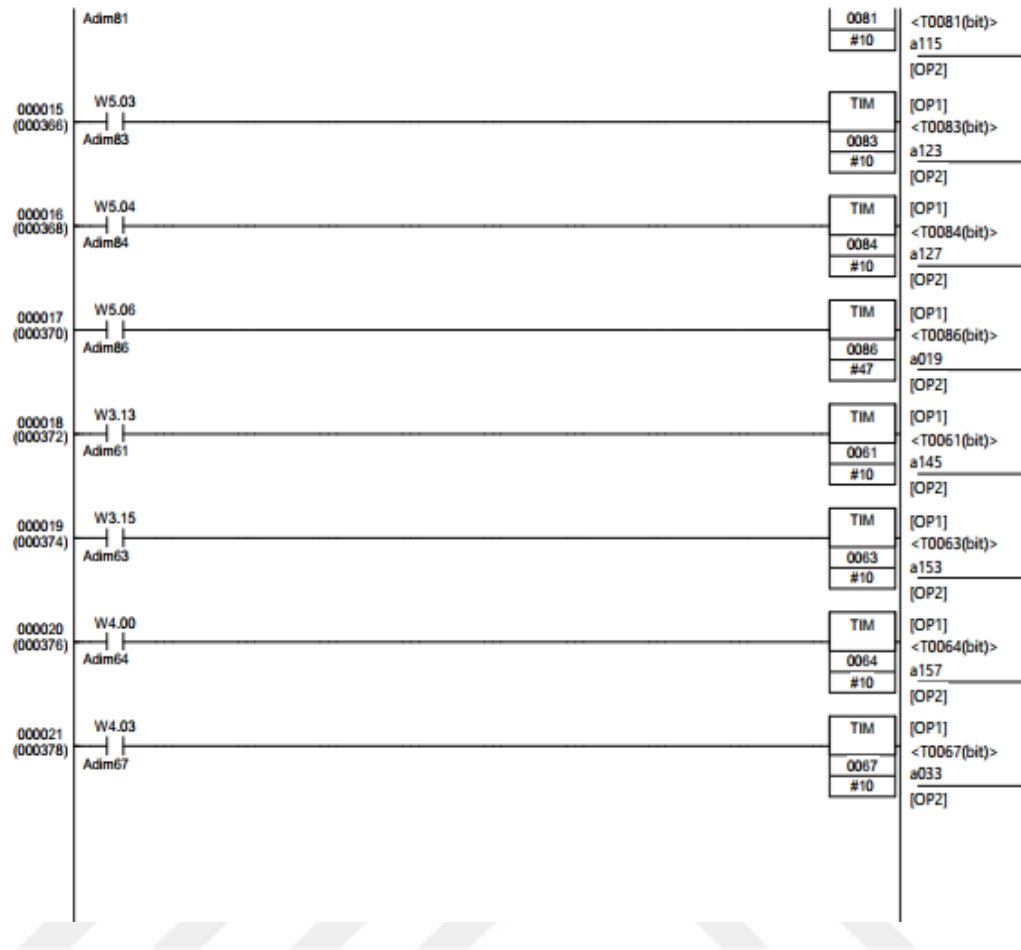




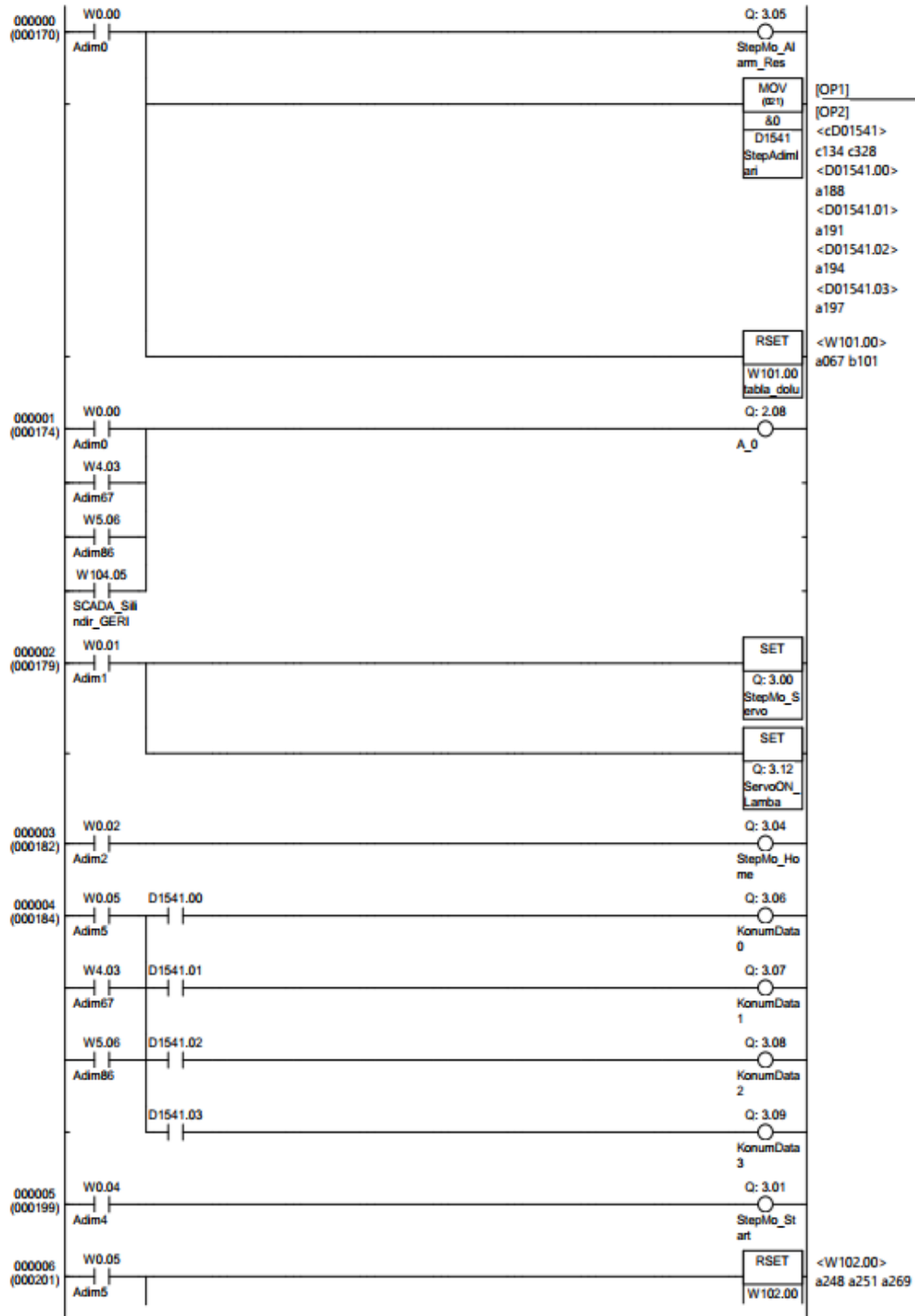


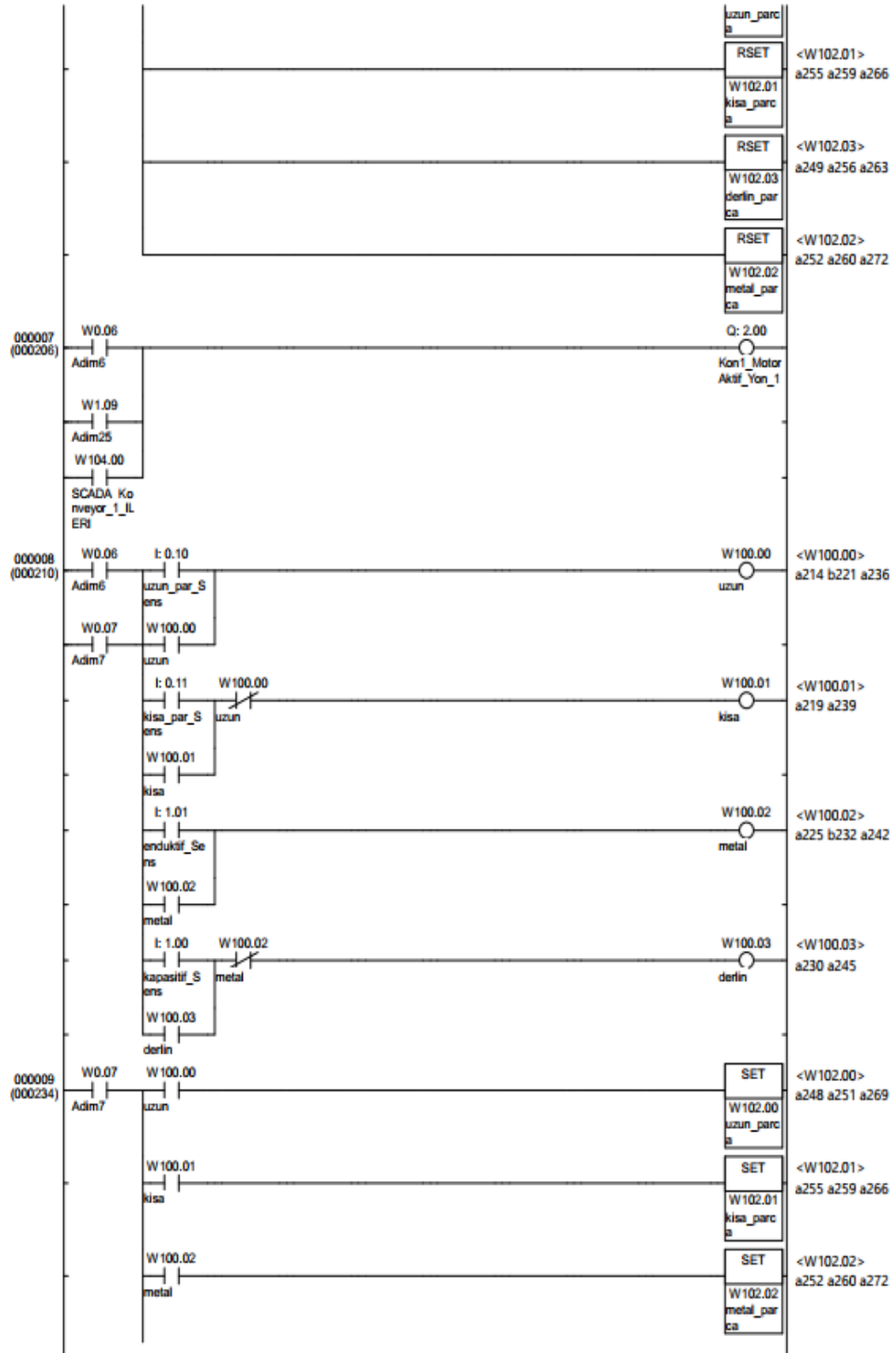
ZAMANLAYICILAR (TIMERS):

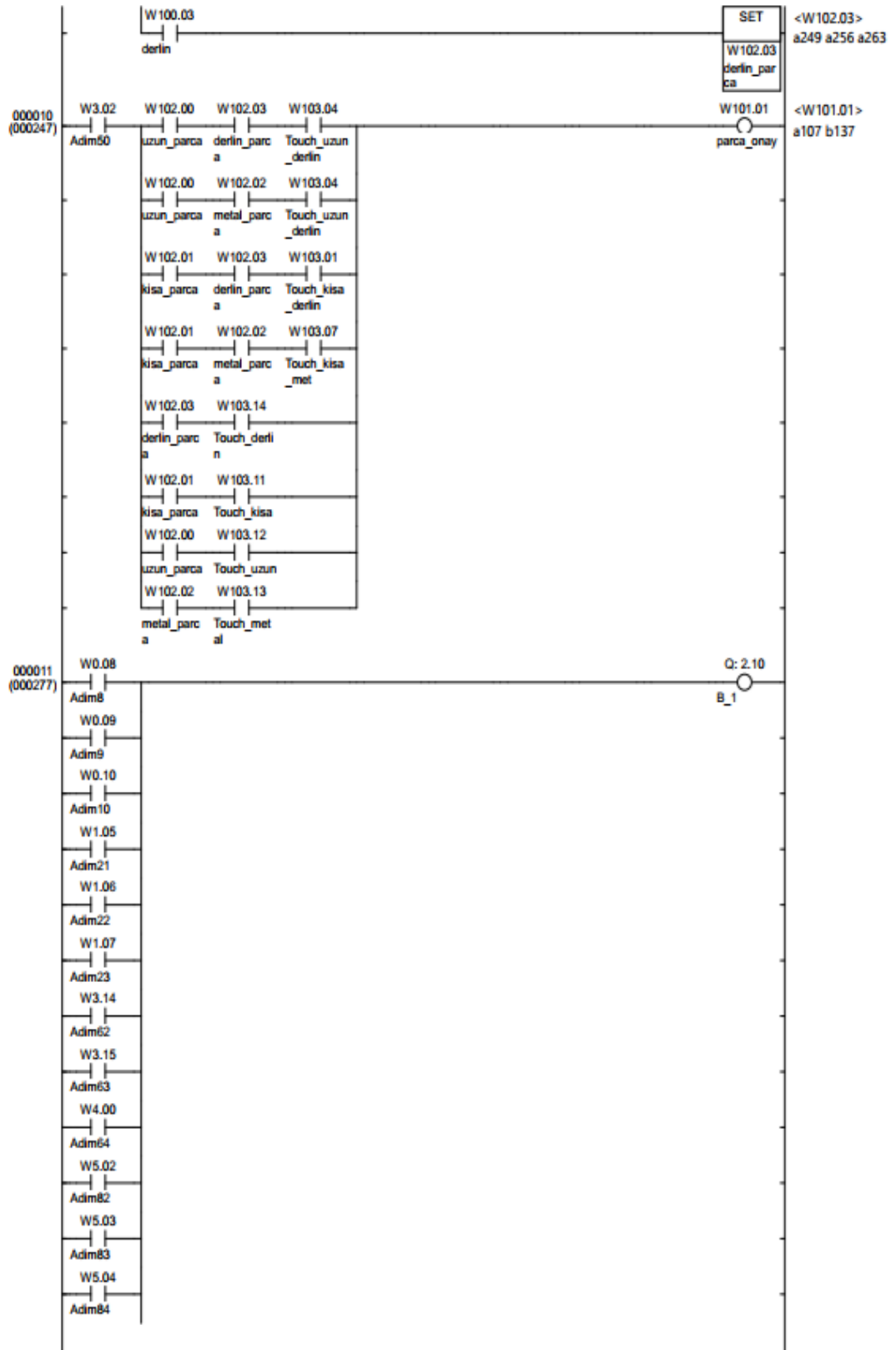


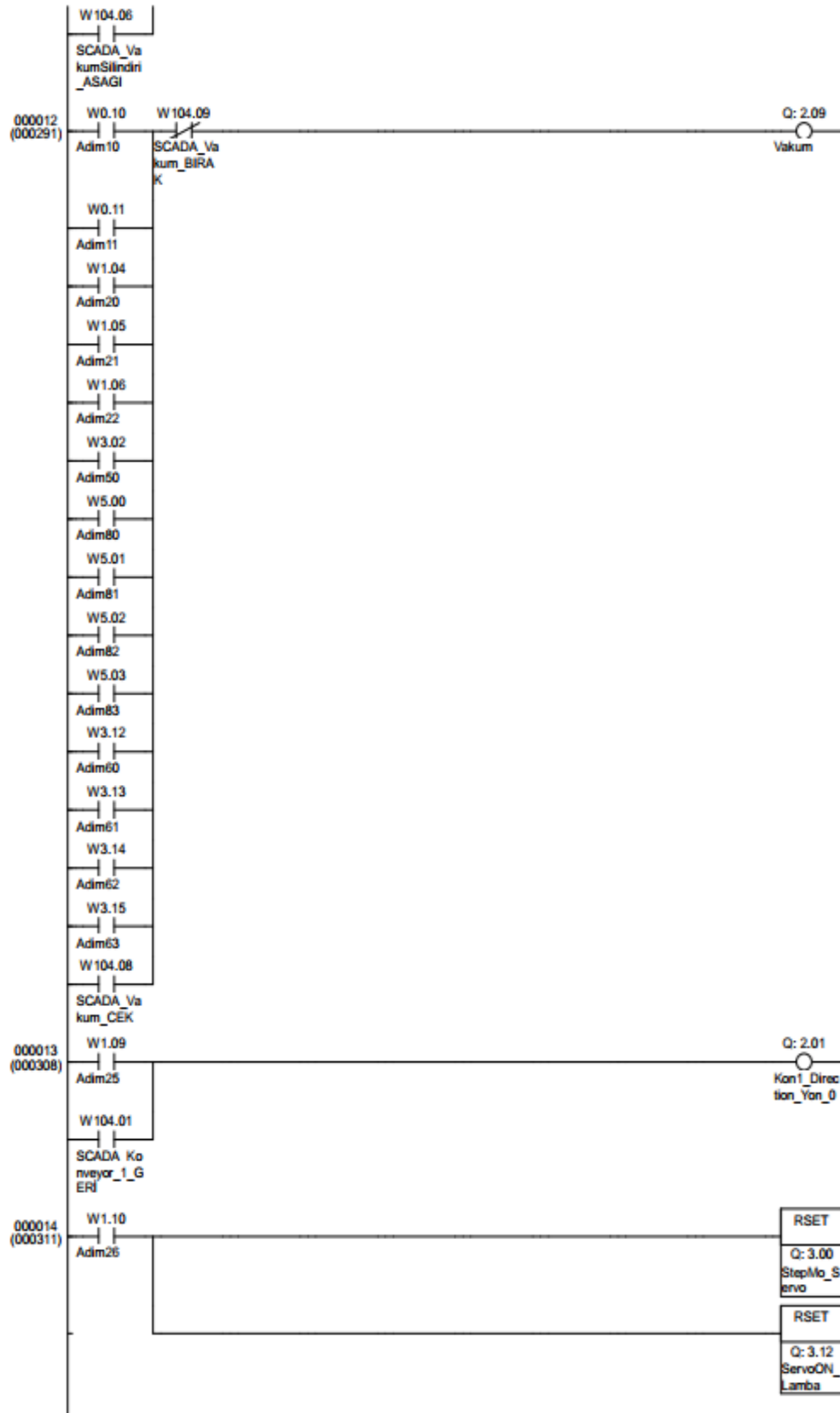


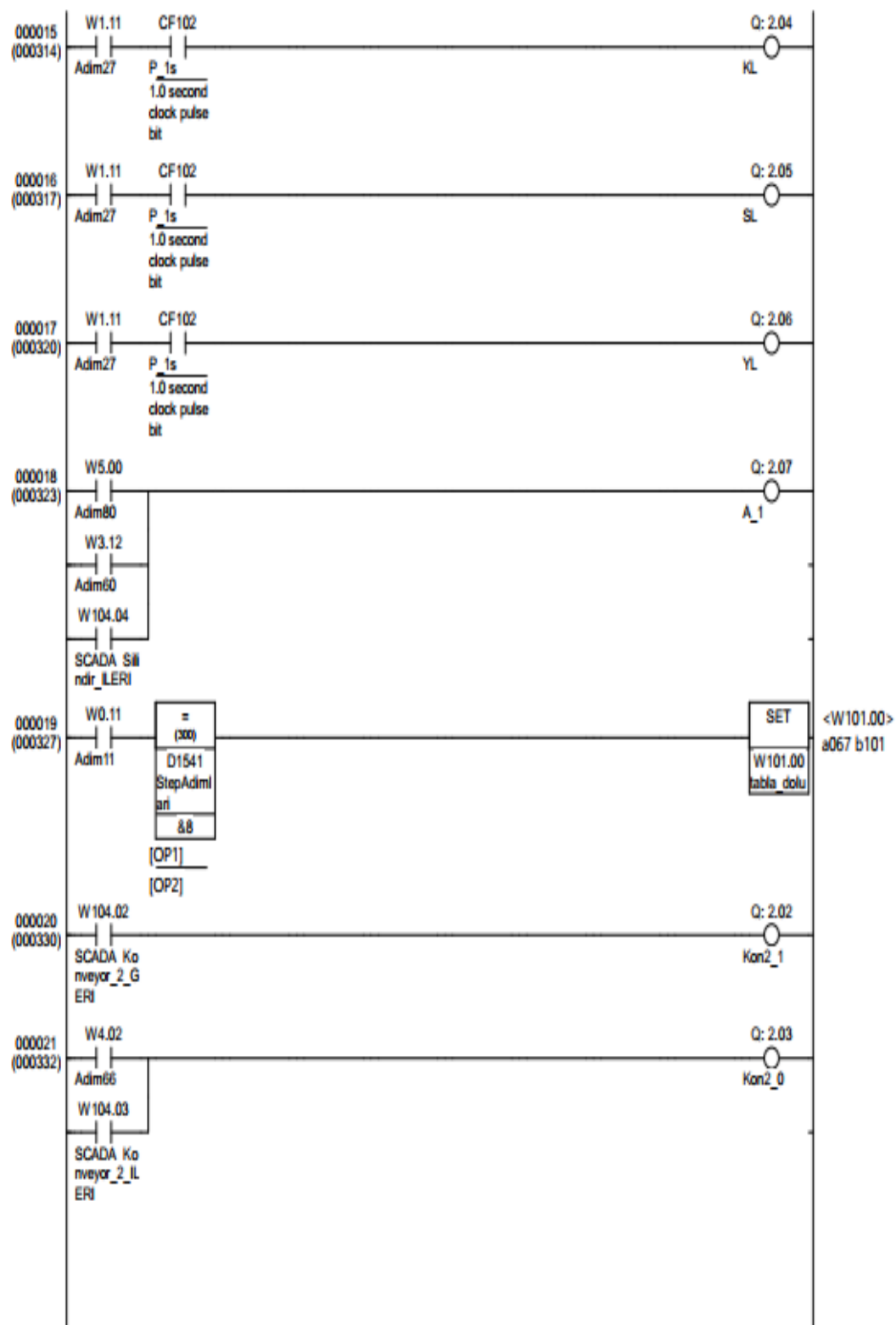
EYLEMLER (ACTIONS):



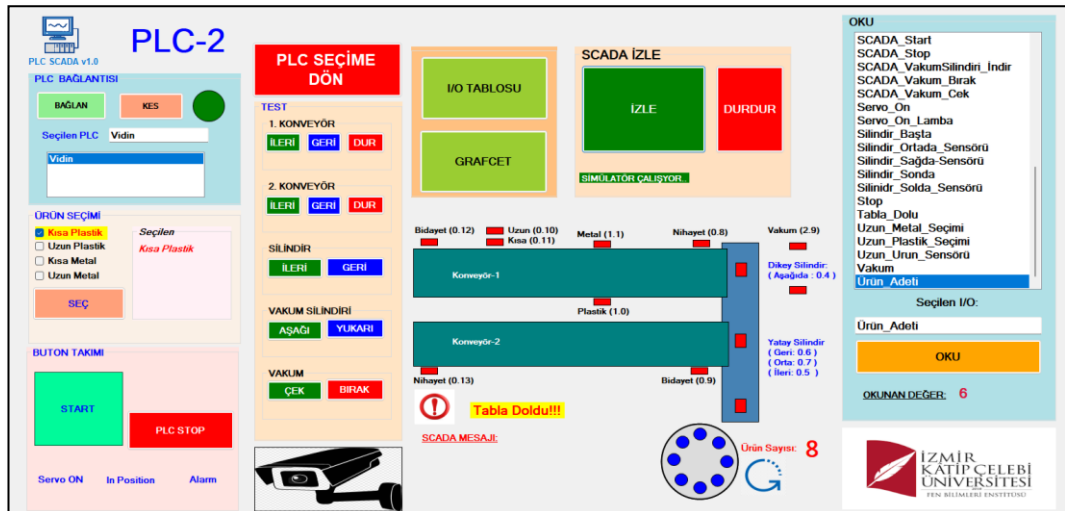
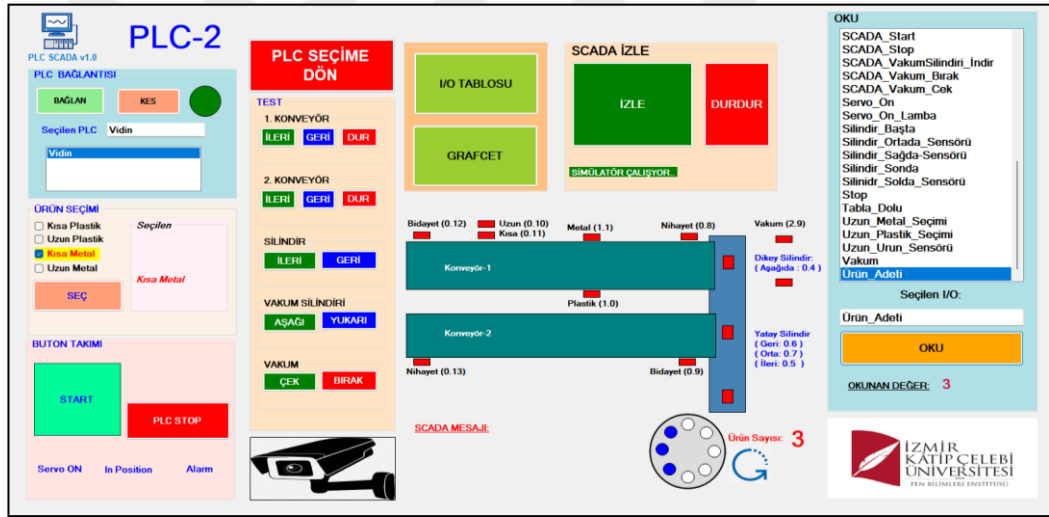
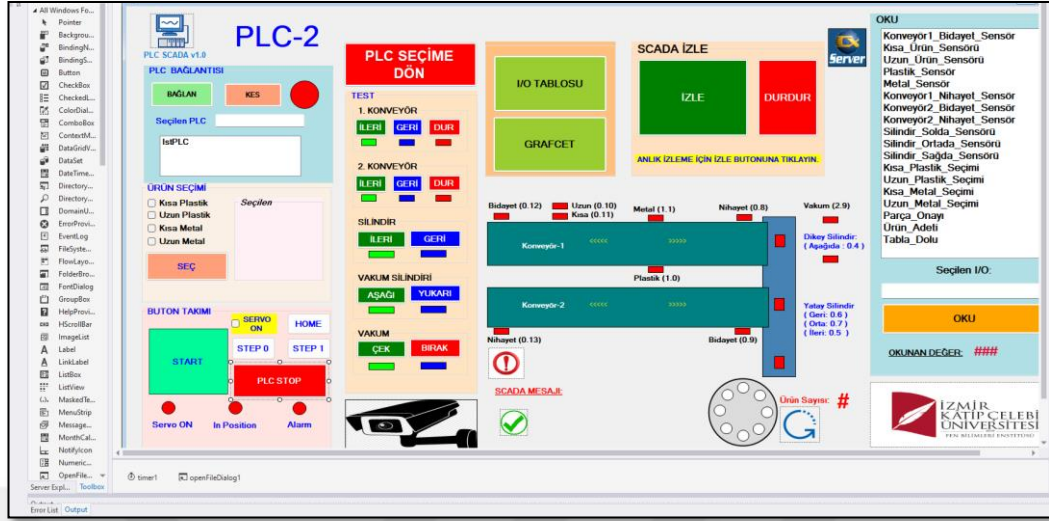








Ek B: PLC-2 SCADA Tasarımı



Ek C: PLC-2 Program Kodları

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace SCADA_PLC
{
    public partial class Omron : Form
    {
        bool PLCConnect;

        object Value = null;
        bool Quality = false;
        object Value1 = null;
        bool Quality1 = false;

        int D1541;
        public Omron()
        {
            InitializeComponent();
        }

        private void button3_Click(object sender, EventArgs e)
        {
            PLCConnect = cxsLiteCtrl1.OpenDevice(txtPLC.Text);

            if (PLCConnect == true)
            {
                ovalShape2.BackColor = Color.Green;
            }
            else
            {
                ovalShape2.BackColor = Color.Black;
            }

            lstPLC.DataSource = cxsLiteCtrl1.Devices;
        }

        private void Omron_Load(object sender, EventArgs e)
        {
            SelectProjectPath();
        }
    }
}
```

```

    }

    private void SelectProjectPath()
    {
        openFileDialog1.Filter = "CXServer Project Files(*.cdm)|*.cdm|All files
(*.*)|*. *";
        DialogResult result = openFileDialog1.ShowDialog();
        if (result == DialogResult.OK && openFileDialog1.FileName != " ")
            cxsLiteCtrl1.ProjectName = openFileDialog1.FileName.ToString();
    }

    private void lstPLC_SelectedIndexChanged_1(object sender, EventArgs e)
    {
        string[] pointler = null;
        cxsLiteCtrl1.ListPoints((string)lstPLC.Items[lstPLC.SelectedIndex], out
pointler);
        txtPLC.Text = (string)lstPLC.Items[lstPLC.SelectedIndex];
        lstPoints.DataSource = pointler;
    }

    private void button2_Click(object sender, EventArgs e)
    {
        PLCConnect = cxsLiteCtrl1.CloseDevice(txtPLC.Text);

        txtPLC.Text = "";

        lstPLC.DataSource = new string[1];
        ovalShape2.BackColor = Color.Red;
    }

    public void gecikme(int saniye)
    {
        saniye = (saniye + Convert.ToInt32(DateTime.Now.Second));
        for (;;)
        {
            if (saniye == DateTime.Now.Second) break;
        }
    }

    private void button22_Click(object sender, EventArgs e)
    {
        omron_io omron = new omron_io();
        omron.Show();
    }

    private void button23_Click(object sender, EventArgs e)

```

```

{
    Grafcet about = new Grafcet();
    about.Show();
}

private void button34_Click(object sender, EventArgs e)
{
    Kamera cam = new Kamera();
    cam.Show();
}

private void button6_Click(object sender, EventArgs e)
{
    cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Konveyör1_İleri", 1);
    cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Konveyör1_Geri", 0);
    rectangleShape1.Visible = true;
    rectangleShape2.Visible = false;
    rectangleShape3.Visible = false;
    label13.Visible = true;
    label12.Visible = false;
    gecikme(1);
}

private void button7_Click(object sender, EventArgs e)
{
    cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Konveyör1_İleri", 1);
    cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Konveyör1_Geri", 1);
    rectangleShape1.Visible = false;
    rectangleShape2.Visible = true;
    rectangleShape3.Visible = false;
    label12.Visible = true;
    label13.Visible = false;
    gecikme(1);
}

private void button8_Click(object sender, EventArgs e)
{
    cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Konveyör1_İleri", 0);
    cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Konveyör1_Geri", 0);
    rectangleShape1.Visible = false;
    rectangleShape2.Visible = false;
    rectangleShape3.Visible = true;
    label12.Visible = false;
    label13.Visible = false;
    gecikme(1);
}

private void button9_Click(object sender, EventArgs e)
{
    cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Konveyör2_Geri", 1);
}

```

```

        cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Konveyör2_İleri", 0);
        rectangleShape4.Visible = true;
        rectangleShape5.Visible = false;
        rectangleShape6.Visible = false;
        label15.Visible = true;
        label14.Visible = false;
        gecikme(1);
    }

```

```

private void button10_Click(object sender, EventArgs e)
{
    cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Konveyör2_Geri", 0);
    cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Konveyör2_İleri", 1);
    rectangleShape4.Visible = false;
    rectangleShape5.Visible = true;
    rectangleShape6.Visible = false;
    label15.Visible = false;
    label14.Visible = true;
    gecikme(1);
}

```

```

private void button11_Click(object sender, EventArgs e)
{
    cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Konveyör2_Geri", 0);
    cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Konveyör2_İleri", 0);
    rectangleShape4.Visible = false;
    rectangleShape5.Visible = false;
    rectangleShape6.Visible = true;
    label15.Visible = false;
    label14.Visible = false;
    gecikme(1);
}

```

```

private void button14_Click(object sender, EventArgs e)
{
    cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Silindir_İleri", 1);
    cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Silindir_Geri", 0);
    rectangleShape9.Visible = true;
    rectangleShape8.Visible = false;
    rectangleShape80.BackColor = Color.Lime;
    rectangleShape81.BackColor = Color.Red;
    gecikme(1);
}

```

```

private void button13_Click(object sender, EventArgs e)
{
    cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Silindir_İleri", 0);
    cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Silindir_Geri", 1);
    rectangleShape9.Visible = false;
    rectangleShape8.Visible = true;
}

```

```

        rectangleShape80.BackColor = Color.Red;
        rectangleShape81.BackColor = Color.Lime;
        gecikme(1);
    }

    private void button17_Click(object sender, EventArgs e)
    {
        cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_VakumSilindiri_İndir", 1);
        rectangleShape12.Visible = true;
        rectangleShape11.Visible = false;
        rectangleShape18.BackColor = Color.Lime;
        gecikme(1);
    }

    private void button16_Click(object sender, EventArgs e)
    {
        cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_VakumSilindiri_İndir", 0);
        rectangleShape12.Visible = false;
        rectangleShape11.Visible = true;
        rectangleShape18.BackColor = Color.Red;
        gecikme(1);
    }

    private void button15_Click(object sender, EventArgs e)
    {
        cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Vakum_Cek", 1);
        rectangleShape7.Visible = true;
        rectangleShape10.Visible = false;
        rectangleShape17.BackColor = Color.Lime;
        gecikme(1);
    }

    private void button12_Click(object sender, EventArgs e)
    {
        cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Vakum_Cek", 0);
        rectangleShape7.Visible = false;
        rectangleShape10.Visible = true;
        rectangleShape17.BackColor = Color.Red;
        gecikme(1);
    }

    private void button18_Click(object sender, EventArgs e)
    {
        object Value = null;
        bool Quality = false;
        if (cxsLiteCtrl1.Read(txtPLC.Text, txtPoint.Text, out Value, out Quality))
            label7.Text = Value.ToString();
        gecikme(1);
    }

```

```

private void button1_Click(object sender, EventArgs e)
{
    PLCConnect = cxsLiteCtrl1.CloseDevice(txtPLC.Text);
    this.Close();
}

private void label8_Click(object sender, EventArgs e)
{
}

private void lstPoints_SelectedIndexChanged(object sender, EventArgs e)
{
    txtPoint.Text = (string)lstPoints.Items[lstPoints.SelectedIndex];
}

private void button5_Click(object sender, EventArgs e)
{
}

private void button4_Click(object sender, EventArgs e)
{
    cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Start", 1);
}

private void button28_Click(object sender, EventArgs e)
{
    cxsLiteCtrl1.Write(txtPLC.Text, "SCADA_Stop", 1);
    label17.Text = "Tablayı Boşaltın. Sistem SIFIRLANDI.. ";
    pictureBox2.Visible = false;
    //label23.Visible = false;
}

private void button24_Click(object sender, EventArgs e)
{
    timer1.Interval = 100;
    timer1.Start();
    label102.BackColor = Color.Green;
    label102.ForeColor = Color.White;
    label102.Text = "SİMÜLATÖR ÇALIŞYOR..";
}

private void button25_Click(object sender, EventArgs e)
{
    timer1.Enabled = false;
}

```

```

timer1.Stop();
label102.BackColor = Color.Red;
label102.ForeColor = Color.White;
label102.Text = "SİMÜLATÖR DURDURULDU..";
}

private void timer1_Tick(object sender, EventArgs e)
{
    if (cxsLiteCtrl1.Read(txtPLC.Text, "Parça_Onay1", out Value1, out
Quality1))

        if (Convert.ToString(Value1) == "True")
        {
            label85.Visible = true;
            label85.Text = "Parça Onaylandı...";
            pictureBox6.Visible = true;
            label85.Visible = true;
        }
        else
        {
            pictureBox6.Visible = false;
            label85.Visible = false;
        }

    if (cxsLiteCtrl1.Read(txtPLC.Text, "Ürün_Adeti", out Value, out Quality))

        D1541 = Convert.ToInt16(Value);

    if (D1541 == 0)
    {
        ovalShape15.BackColor = Color.White; ovalShape22.BackColor =
Color.White; ovalShape20.BackColor = Color.White;
        ovalShape21.BackColor = Color.White; ovalShape19.BackColor =
Color.White; ovalShape18.BackColor = Color.White;
        ovalShape17.BackColor = Color.White; ovalShape16.BackColor =
Color.White; label16.Text = "0"; pictureBox2.Visible = false;
    }

    if (D1541 == 1)
    {
        ovalShape15.BackColor = Color.Blue; ovalShape22.BackColor =
Color.White; ovalShape20.BackColor = Color.White;
        ovalShape21.BackColor = Color.White; ovalShape19.BackColor =
Color.White; ovalShape18.BackColor = Color.White;
        ovalShape17.BackColor = Color.White; ovalShape16.BackColor =
Color.White; label16.Text = "1"; pictureBox2.Visible = false;
        pictureBox5.Visible = true;
    }
}

```

```

if (D1541 == 2)
{
    ovalShape15.BackColor = Color.Blue; ovalShape22.BackColor =
Color.Blue; ovalShape20.BackColor = Color.White;
    ovalShape21.BackColor = Color.White; ovalShape19.BackColor =
Color.White; ovalShape18.BackColor = Color.White;
    ovalShape17.BackColor = Color.White; ovalShape16.BackColor =
Color.White; label16.Text = "2"; pictureBox2.Visible = false;
    pictureBox5.Visible = true;
}

if (D1541 == 3)
{
    ovalShape15.BackColor = Color.Blue; ovalShape22.BackColor =
Color.Blue; ovalShape20.BackColor = Color.Blue;
    ovalShape21.BackColor = Color.White; ovalShape19.BackColor =
Color.White; ovalShape18.BackColor = Color.White;
    ovalShape17.BackColor = Color.White; ovalShape16.BackColor =
Color.White; label16.Text = "3"; pictureBox2.Visible = false;
    pictureBox5.Visible = true;
}

if (D1541 == 4)
{
    ovalShape15.BackColor = Color.Blue; ovalShape22.BackColor =
Color.Blue; ovalShape20.BackColor = Color.Blue;
    ovalShape21.BackColor = Color.Blue; ovalShape19.BackColor =
Color.White; ovalShape18.BackColor = Color.White;
    ovalShape17.BackColor = Color.White; ovalShape16.BackColor =
Color.White; label16.Text = "4"; pictureBox2.Visible = false;
}

if (D1541 == 5)
{
    ovalShape15.BackColor = Color.Blue; ovalShape22.BackColor =
Color.Blue; ovalShape20.BackColor = Color.Blue;
    ovalShape21.BackColor = Color.Blue; ovalShape19.BackColor =
Color.Blue; ovalShape18.BackColor = Color.White;
    ovalShape17.BackColor = Color.White; ovalShape16.BackColor =
Color.White; label16.Text = "5"; pictureBox2.Visible = false;
    pictureBox5.Visible = true;
}

```

```

    }

    if (D1541 == 6)

    {

        ovalShape15.BackColor = Color.Blue; ovalShape22.BackColor =
Color.Blue; ovalShape20.BackColor = Color.Blue;
        ovalShape21.BackColor = Color.Blue; ovalShape19.BackColor =
Color.Blue; ovalShape18.BackColor = Color.Blue;
        ovalShape17.BackColor = Color.White; ovalShape16.BackColor =
Color.White; label16.Text = "6"; pictureBox2.Visible = false;
        pictureBox5.Visible = true;
    }


    if (D1541 == 7)

    {

        ovalShape15.BackColor = Color.Blue; ovalShape22.BackColor =
Color.Blue; ovalShape20.BackColor = Color.Blue;
        ovalShape21.BackColor = Color.Blue; ovalShape19.BackColor =
Color.Blue; ovalShape18.BackColor = Color.Blue;
        ovalShape17.BackColor = Color.Blue; ovalShape16.BackColor =
Color.White; label16.Text = "7"; pictureBox2.Visible = false;
        pictureBox5.Visible = true;
    }


    if (D1541 == 8)

    {

        ovalShape15.BackColor = Color.Blue; ovalShape22.BackColor =
Color.Blue; ovalShape20.BackColor = Color.Blue;
        ovalShape21.BackColor = Color.Blue; ovalShape19.BackColor =
Color.Blue; ovalShape18.BackColor = Color.Blue;
        ovalShape17.BackColor = Color.Blue; ovalShape16.BackColor =
Color.Blue;
        label16.Text = "8"; label17.Text = "Tabla Doldu!!!"; pictureBox2.Visible
= true; pictureBox5.Visible = false;
        button28.Visible = true;
        pictureBox5.Visible = true;
    }

}

```

```

private void button5_Click_1(object sender, EventArgs e)
{
    if (checkBox1.Checked == true)
    {
        checkBox1.ForeColor = Color.Red;
        checkBox1.BackColor = Color.Yellow;
        label2.Text = checkBox1.Text;
        label6.Text = "";
        button4.Visible = true;
        groupBox12.Visible = true;

        cxsliteCtrl1.Write(txtPLC.Text, "Kısa_Plastik_Seçimi", 1);

    }
    if (checkBox1.Checked == false)
    {
        checkBox1.ForeColor = Color.Black;
        checkBox1.BackColor = Color.Linen;
        label2.Text = "";
        cxsliteCtrl1.Write(txtPLC.Text, "Kısa_Plastik_Seçimi", 0);
    }
    if (checkBox2.Checked == true)
    {
        checkBox2.ForeColor = Color.Red;
        checkBox2.BackColor = Color.Yellow;
        label3.Text = checkBox2.Text;
        label6.Text = "";
        button4.Visible = true;
        groupBox12.Visible = true;
        cxsliteCtrl1.Write(txtPLC.Text, "Uzun_Plastik_Seçimi", 1);
    }
    if (checkBox2.Checked == false)
    {
        checkBox2.ForeColor = Color.Black;
        checkBox2.BackColor = Color.Linen;
        label3.Text = "";
        cxsliteCtrl1.Write(txtPLC.Text, "Uzun_Plastik_Seçimi", 0);
    }
    if (checkBox3.Checked == true)
    {
        checkBox3.ForeColor = Color.Red;
        checkBox3.BackColor = Color.Yellow;
        label4.Text = checkBox3.Text;
        label6.Text = "";
        button4.Visible = true;
        cxsliteCtrl1.Write(txtPLC.Text, "Kısa_Metal_Seçimi", 1);
    }
    if (checkBox3.Checked == false)
    {

```

```

        checkBox3.ForeColor = Color.Black;
        checkBox3.BackColor = Color.Linen;
        label4.Text = "";
        cxsLiteCtrl1.Write(txtPLC.Text, "Kısa_Metal_Seçimi", 0);
    }
    if (checkBox4.Checked == true)
    {
        checkBox4.ForeColor = Color.Red;
        checkBox4.BackColor = Color.Yellow;
        label5.Text = checkBox4.Text;
        label6.Text = "";
        button4.Visible = true;
        groupBox12.Visible = true;
        cxsLiteCtrl1.Write(txtPLC.Text, "Uzun_Metal_Seçimi", 1);
    }
    if (checkBox4.Checked == false)
    {
        checkBox4.ForeColor = Color.Black;
        checkBox4.BackColor = Color.Linen;
        label5.Text = "";
        cxsLiteCtrl1.Write(txtPLC.Text, "Uzun_Metal_Seçimi", 0);
    }

    if (checkBox1.Checked == false && checkBox2.Checked == false &&
        checkBox3.Checked == false && checkBox4.Checked == false)
    {
        label6.Text = "EN AZ BİR ÜRÜN SEÇMELİSİNİZ...";
        button4.Visible = false;
        groupBox12.Visible = false;
        cxsLiteCtrl1.Write(txtPLC.Text, "Kısa_Plastik_Seçimi", 0);
        cxsLiteCtrl1.Write(txtPLC.Text, "Uzun_Plastik_Seçimi", 0);
        cxsLiteCtrl1.Write(txtPLC.Text, "Kısa_Metal_Seçimi", 0);
        cxsLiteCtrl1.Write(txtPLC.Text, "Uzun_Plastik_Seçimi", 0);
    }
}
}
}

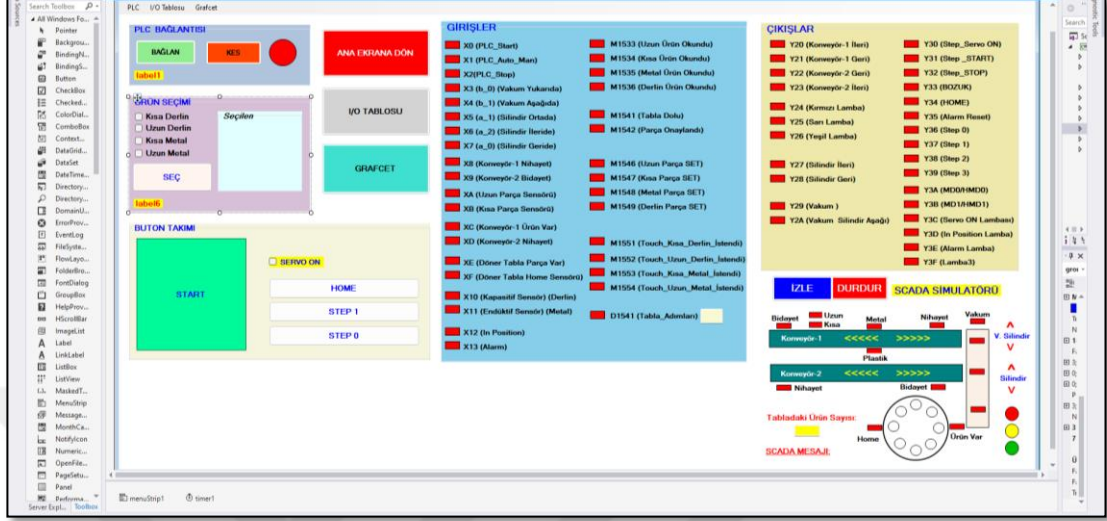
```

Ek D: PLC-2 Dokunmatik (Touch) Panel Tasarımı



Ek E: PTÖ Uygulaması, Örnek Formlar ve Rubrikler

PLC-1 SCADA Tasarımı



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Timers;
```

```
namespace SCADA_PLC
```

```
{
    public partial class Mitsubishi : Form
```

```
    {
        public Mitsubishi()
        {
            InitializeComponent();
        }
```

```
        int st, s = 255, o = 0;
        //int c = 255;
```

```
        ACTMULTILib.ActEasyIF plc = new ACTMULTILib.ActEasyIF();
        private void button1_Click(object sender, EventArgs e)
        {
```

```
            plc.ActLogicalStationNumber = 100;
            int baglanti=plc.Open();
```

```

button2.Enabled = true;
label1.Visible = true;
label1.Text = baglanti.ToString();
if (baglanti == 0)
{
    ovalShape1.BackColor = Color.LightGreen;
    label1.ForeColor = Color.Green;
    label1.BackColor = Color.Lime;
    label1.Text = "PLC'YE BAĞLANILDI";
    button2.Visible = true;
    button1.Visible = false;
    button5.Visible = true;
    groupBox2.Visible = true;
    groupBox4.Visible = true;
    groupBox9.Visible = true;
    groupBox10.Visible = true;
    groupBox11.Visible = true;
    //groupBox12.Visible = true;
    groupBox13.Visible = true;
    //timer1.Enabled = true;
}
else
{
    label1.ForeColor = Color.Red;
    label1.BackColor = Color.Yellow;
    label1.Text = "HATA: BAĞLANTI YAPILAMADI";
}

}

private void button2_Click(object sender, EventArgs e)
{
    plc.Close();
    timer1.Stop();
    timer1.Enabled = false;
    label1.Visible = true;
    ovalShape1.BackColor = Color.Red;
    label1.ForeColor = Color.Red;
    label1.BackColor = Color.LightCyan;
    label1.Text = "PLC'YE BAĞLANTI KESİLDİ";
    button1.Visible = true;
    button2.Visible = false;
    button5.Visible = false;
    groupBox2.Visible = false;
    groupBox4.Visible = false;
    groupBox9.Visible = false;
    groupBox10.Visible = false;
    groupBox11.Visible = false;
    groupBox12.Visible = false;
}

```

```

private void button1_MouseMove(object sender, MouseEventArgs e)
{
    button1.BackColor = Color.Green;
    button1.ForeColor = Color.White;
}

private void button2_MouseMove(object sender, MouseEventArgs e)
{
    button2.BackColor = Color.Red;
    button2.ForeColor = Color.White;
}

private void bağlanToolStripMenuItem_Click(object sender, EventArgs e)
{
    plc.ActLogicalStationNumber = 100;
    int baglanti = plc.Open();
    label1.Text = baglanti.ToString();
    if (baglanti == 0)
    {
        ovalShape1.BackColor = Color.Green;
        label1.ForeColor = Color.Green;
        label1.BackColor = Color.LightCyan;
        label1.Text = "PLC'YE BAĞLANTI YAPILDI";
    }
    else
    {
        label1.ForeColor = Color.Red;
        label1.BackColor = Color.Yellow;
        label1.Text = "HATA: BAĞLANTI YAPILAMADI";
    }
}

private void button1_MouseLeave(object sender, EventArgs e)
{
    button1.ForeColor = Color.Black;
    button1.BackColor = Color.LightGreen;
}

private void button2_MouseLeave(object sender, EventArgs e)
{
    button2.ForeColor = Color.Black;
    button2.BackColor = Color.LightSalmon;
}

private void kesToolStripMenuItem_Click(object sender, EventArgs e)
{
    plc.Close();
}

```

```

        ovalShape1.BackColor = Color.Red;
        label1.ForeColor = Color.Red;
        label1.BackColor = Color.LightCyan;
        label1.Text = "PLC'YE BAĞLANTI KESİLDİ";
    }

    private void button3_Click(object sender, EventArgs e)
    {
        plc.Close();
        this.Close();
        timer1.Start();
        timer1.Enabled = false;
    }

    private void button3_MouseMove(object sender, MouseEventArgs e)
    {
        button3.ForeColor = Color.Red;
        button3.BackColor = Color.Yellow;
    }

    private void button3_MouseLeave(object sender, EventArgs e)
    {
        button3.ForeColor = Color.White;
        button3.BackColor = Color.Red;
    }

    private void button4_MouseMove(object sender, MouseEventArgs e)
    {
        button4.ForeColor = Color.White;
        button4.BackColor = Color.DarkGreen;
    }

    private void button4_MouseLeave(object sender, EventArgs e)
    {
        button4.ForeColor = Color.Blue;
        button4.BackColor = Color.MediumSpringGreen;
    }

    private void checkBox1_CheckStateChanged(object sender, EventArgs e)
    {
        checkBox1.ForeColor = Color.Black;
        checkBox1.BackColor = Color.Linen;
    }

    private void button5_Click(object sender, EventArgs e)
    {
        if (checkBox1.Checked == true)
        {
            checkBox1.ForeColor = Color.Red;
        }
    }

```

```

        checkBox1.BackColor = Color.Yellow;
        label2.Text = checkBox1.Text;
        label6.Text = "";
        button4.Visible = true;
        groupBox12.Visible = true;

        plc.SetDevice("M1551", 1);
    }
    if (checkBox1.Checked == false)
    {
        checkBox1.ForeColor = Color.Black;
        checkBox1.BackColor = Color.Linen;
        label2.Text = "";
        plc.SetDevice("M1551", 0);
    }
    if (checkBox2.Checked == true)
    {
        checkBox2.ForeColor = Color.Red;
        checkBox2.BackColor = Color.Yellow;
        label3.Text = checkBox2.Text;
        label6.Text = "";
        button4.Visible = true;
        groupBox12.Visible = true;
        plc.SetDevice("M1552", 1);
    }
    if (checkBox2.Checked == false)
    {
        checkBox2.ForeColor = Color.Black;
        checkBox2.BackColor = Color.Linen;
        label3.Text = "";
        plc.SetDevice("M1552", 0);
    }
    if (checkBox3.Checked == true)
    {
        checkBox3.ForeColor = Color.Red;
        checkBox3.BackColor = Color.Yellow;
        label4.Text = checkBox3.Text;
        label6.Text = "";
        button4.Visible = true;
        plc.SetDevice("M1553", 1);
        st = plc.WriteDeviceRandom("M1553", 1, ref (s));
    }
    if (checkBox3.Checked == false)
    {
        checkBox3.ForeColor = Color.Black;
        checkBox3.BackColor = Color.Linen;
        label4.Text = "";
        plc.SetDevice("M1553", 0);
    }

```

```

    }
    if (checkBox4.Checked == true)
    {
        checkBox4.ForeColor = Color.Red;
        checkBox4.BackColor = Color.Yellow;
        label5.Text = checkBox4.Text;
        label6.Text = "";
        button4.Visible = true;
        groupBox12.Visible = true;
        plc.SetDevice("M1554", 1);
    }
    if (checkBox4.Checked == false)
    {
        checkBox4.ForeColor = Color.Black;
        checkBox4.BackColor = Color.Linen;
        label5.Text = "";
        //st = plc.WriteDeviceRandom("M1554", 1, ref (o));
        plc.SetDevice("M1554", 0);
    }
}

if (checkBox1.Checked == false && checkBox2.Checked == false &&
checkBox3.Checked == false && checkBox4.Checked == false)
{
    label6.Text = "EN AZ BİR ÜRÜN SEÇMELİSİNİZ...";
    button4.Visible = false;
    groupBox12.Visible = false;
    plc.SetDevice("M1551", 0);
    plc.SetDevice("M1552", 0);
    plc.SetDevice("M1553", 0);
    plc.SetDevice("M1554", 0);
}
}

private void checkBox2_CheckedChanged(object sender, EventArgs e)
{
}

private void checkBox2_CheckStateChanged_1(object sender, EventArgs e)
{
    checkBox2.ForeColor = Color.Black;
    checkBox2.BackColor = Color.Linen;
}

private void checkBox3_CheckStateChanged_1(object sender, EventArgs e)
{
    checkBox3.ForeColor = Color.Black;
    checkBox3.BackColor = Color.Linen;
}

```

```

}

private void checkBox4_CheckStateChanged_1(object sender, EventArgs e)
{
    checkBox4.ForeColor = Color.Black;
    checkBox4.BackColor = Color.Linen;
}

private void button5_MouseMove(object sender, MouseEventArgs e)
{
    button5.BackColor = Color.DarkSalmon;
}

private void button5_MouseLeave(object sender, EventArgs e)
{
    button5.BackColor = Color.Bisque;
}

private void pictureBox1_Click(object sender, EventArgs e)
{
}

private void Mitsubishi_Load(object sender, EventArgs e)
{
    label1.Visible = false;
    button2.Visible = false;
    groupBox2.Visible = false;
    groupBox4.Visible = false;
    groupBox9.Visible = false;
    groupBox10.Visible = false;
    groupBox11.Visible = false;

    groupBox13.Visible = false;

    groupBox12.Visible = false;
    groupBox13.Visible = false;
}

private void iOTablosuToolStripMenuItem_Click(object sender, EventArgs e)
{
    IO about = new IO();
    about.Show();
}

private void button22_Click(object sender, EventArgs e)
{
    IO about = new IO();
}

```

```

        about.Show();
    }

    private void button22_MouseMove(object sender, MouseEventArgs e)
    {
        button22.ForeColor = Color.White;
        button22.BackColor = Color.Red;
    }

    private void button22_MouseLeave(object sender, EventArgs e)
    {
        button22.ForeColor = Color.Black;
        button22.BackColor = Color.Orange;
    }

    private void grafcetToolStripMenuItem_Click(object sender, EventArgs e)
    {
        Grafcet about = new Grafcet();
        about.Show();
    }

    private void button23_MouseMove(object sender, MouseEventArgs e)
    {
        button23.ForeColor = Color.White;
        button23.BackColor = Color.Red;
    }

    private void button23_MouseLeave(object sender, EventArgs e)
    {
        button23.ForeColor = Color.Black;
        button23.BackColor = Color.Gold;
    }

    private void button23_Click(object sender, EventArgs e)
    {
        Grafcet about = new Grafcet();
        about.Show();
    }

    private void button4_Click(object sender, EventArgs e)
    {
        int veri, veri_oku;

        st = plc.WriteDeviceRandom("M200", 1, ref (s));

        //plc.SetDevice("M200", 1);
        //plc.SetDevice("M202", 0);

        button19.Visible = true;
    }

```

```

button20.Visible = true;
button21.Visible = true;
checkBox5.Visible = true;
groupBox4.Visible = false;
groupBox11.Visible = false;

veri_oku = plc.ReadDeviceRandom("XC", 1, out (veri));

label85.Text = "Sistem Çalışıyor...";
}

private void button20_Click(object sender, EventArgs e)
{
    if (checkBox5.Checked)
    {
        st = plc.WriteDeviceRandom("Y30", 1, ref (s));
        st = plc.WriteDeviceRandom("Y36", 1, ref (s));
        label85.Text = "STEP 0 Konumu";
        checkBox1.BackColor = Color.Yellow;
    }

    if (!checkBox5.Checked)
    {
        st = plc.WriteDeviceRandom("Y30", 1, ref (o));
        st = plc.WriteDeviceRandom("Y36", 1, ref (o));

        label85.Text = "";
    }
}

private void button21_Click(object sender, EventArgs e)
{
    if (checkBox5.Checked)
    {
        st = plc.WriteDeviceRandom("Y30", 1, ref (s));
        st = plc.WriteDeviceRandom("Y37", 1, ref (s));

        label85.Text = "STEP 1 Konumu";
        checkBox1.BackColor = Color.Yellow;
    }

    if (!checkBox5.Checked)
    {
        st = plc.WriteDeviceRandom("Y30", 1, ref (o));
        st = plc.WriteDeviceRandom("Y37", 1, ref (o));

        label85.Text = "";
    }
}

```

```

}

private void button19_Click(object sender, EventArgs e)
{
    if (checkBox5.Checked)
    {
        st = plc.WriteDeviceRandom("Y30", 1, ref (s));
        st = plc.WriteDeviceRandom("Y34", 1, ref (s));

        label85.Text = "Home Position...";
        checkBox1.BackColor = Color.Yellow;
    }

    if (!checkBox5.Checked)
    {
        st = plc.WriteDeviceRandom("Y30", 1, ref (o));
        st = plc.WriteDeviceRandom("Y34", 1, ref (o));
        label85.Text = "";
    }
}

private void button6_Click(object sender, EventArgs e)
{
    st = plc.WriteDeviceRandom("M300", 1, ref (s));
    st = plc.WriteDeviceRandom("M301", 1, ref (o));
    rectangleShape1.Visible = true;
    rectangleShape2.Visible = false;
    rectangleShape3.Visible = false;
}

private void button7_Click(object sender, EventArgs e)
{
    st = plc.WriteDeviceRandom("M300", 1, ref (s));
    st = plc.WriteDeviceRandom("M301", 1, ref (s));

    rectangleShape1.Visible = false;
    rectangleShape2.Visible = true;
    rectangleShape3.Visible = false;
}

```

```

private void button8_Click(object sender, EventArgs e)
{
    st = plc.WriteDeviceRandom("M300", 1, ref (o));
    st = plc.WriteDeviceRandom("M301", 1, ref (o));
    rectangleShape1.Visible = false;
    rectangleShape2.Visible = false;
    rectangleShape3.Visible = true;
}

private void button9_Click(object sender, EventArgs e)
{
    st = plc.WriteDeviceRandom("M302", 1, ref (s));
    st = plc.WriteDeviceRandom("M303", 1, ref (o));

    rectangleShape4.Visible = true;
    rectangleShape5.Visible = false;
    rectangleShape6.Visible = false;
}

private void button10_Click(object sender, EventArgs e)
{
    //plc.SetDevice("Y22", 1);
    st = plc.WriteDeviceRandom("M302", 1, ref (o));
    st = plc.WriteDeviceRandom("M303", 1, ref (s));

    rectangleShape4.Visible = false;
    rectangleShape5.Visible = true;
    rectangleShape6.Visible = false;
}

private void button11_Click(object sender, EventArgs e)
{
    st = plc.WriteDeviceRandom("M302", 1, ref ( o));
    st = plc.WriteDeviceRandom("M303", 1, ref (o));
    rectangleShape4.Visible = false;
    rectangleShape5.Visible = false;
    rectangleShape6.Visible = true;
}

private void button14_Click(object sender, EventArgs e)
{
    st = plc.WriteDeviceRandom("M304", 1, ref (s));
    st = plc.WriteDeviceRandom("M305", 1, ref (o));

    rectangleShape9.Visible = true;
}

```

```

        rectangleShape8.Visible = false;
    }

    private void button13_Click(object sender, EventArgs e)
    {
        st = plc.WriteDeviceRandom("M304", 1, ref (o));
        st = plc.WriteDeviceRandom("M305", 1, ref (s));
        rectangleShape8.Visible = true;
        rectangleShape9.Visible = false;
    }

    private void button17_Click(object sender, EventArgs e)
    {
        st = plc.WriteDeviceRandom("M306", 1, ref (s));
        rectangleShape12.Visible = true;
        rectangleShape11.Visible = false;
    }

    private void button16_Click(object sender, EventArgs e)
    {
        st = plc.WriteDeviceRandom("M306", 1, ref (o));
        rectangleShape11.Visible = true;
        rectangleShape12.Visible = false;
    }

    private void button15_Click(object sender, EventArgs e)
    {
        st = plc.WriteDeviceRandom("M307", 1, ref (s));
        rectangleShape7.Visible = true;
        rectangleShape10.Visible = false;
    }

    private void button12_Click(object sender, EventArgs e)
    {
        st = plc.WriteDeviceRandom("M307", 1, ref (o));
        rectangleShape7.Visible = false;
        rectangleShape10.Visible = true;
    }

    private void timer1_Tick(object sender, EventArgs e)
    {
        int X0, X1, X2, X3, X4, X5, X6, X7, X8, X9, XA, XB, XC, XD, XE, XF,
        X10, X11, X12, X13;
        int oku_X0, oku_X1, oku_X2, oku_X3, oku_X4, oku_X5, oku_X6, oku_X7,
        oku_X8, oku_X9, oku_XA, oku_XB, oku_XC, oku_XD, oku_XE, oku_XF,
        oku_X10, oku_X11, oku_X12, oku_X13;
    }

```

int M1533, M1534, M1535, M1536, M1541, M1542, M1546, M1547,
M1548, M1549, M1551, M1552, M1553, M1554;

int oku_M1533, oku_M1534, oku_M1535, oku_M1536, oku_M1541,
oku_M1542, oku_M1546, oku_M1547, oku_M1548, oku_M1549, oku_M1551,
oku_M1552, oku_M1553, oku_M1554;

int Y20, Y21, Y22, Y23, Y24, Y25, Y26, Y27, Y28, Y29, Y2A;

int oku_Y20, oku_Y21, oku_Y22, oku_Y23, oku_Y24, oku_Y25, oku_Y26,
oku_Y27, oku_Y28, oku_Y29, oku_Y2A;

int Y30, Y31, Y32, Y33, Y34, Y35, Y36, Y37, Y38, Y39, Y3A, Y3B, Y3C,
Y3D, Y3E, Y3F;

int oku_Y30, oku_Y31, oku_Y32, oku_Y33, oku_Y34, oku_Y35, oku_Y36,
oku_Y37, oku_Y38, oku_Y39, oku_Y3A, oku_Y3B, oku_Y3C, oku_Y3D,
oku_Y3E, oku_Y3F;

int D1541, oku_D1541;

oku_X0 = plc.ReadDeviceRandom("X0", 1, out (X0));
oku_X1 = plc.ReadDeviceRandom("X1", 1, out (X1));
oku_X2 = plc.ReadDeviceRandom("X2", 1, out (X2));
oku_X3 = plc.ReadDeviceRandom("X3", 1, out (X3));
oku_X4 = plc.ReadDeviceRandom("X4", 1, out (X4));
oku_X5 = plc.ReadDeviceRandom("X5", 1, out (X5));
oku_X6 = plc.ReadDeviceRandom("X6", 1, out (X6));
oku_X7 = plc.ReadDeviceRandom("X7", 1, out (X7));
oku_X8 = plc.ReadDeviceRandom("X8", 1, out (X8));
oku_X9 = plc.ReadDeviceRandom("X9", 1, out (X9));
oku_XA = plc.ReadDeviceRandom("XA", 1, out (XA));
oku_XB = plc.ReadDeviceRandom("XB", 1, out (XB));
oku_XC = plc.ReadDeviceRandom("XC", 1, out (XC));
oku_XD = plc.ReadDeviceRandom("XD", 1, out (XD));
oku_XE = plc.ReadDeviceRandom("XE", 1, out (XE));
oku_XF = plc.ReadDeviceRandom("XF", 1, out (XF));
oku_X10 = plc.ReadDeviceRandom("X10", 1, out (X10));
oku_X11 = plc.ReadDeviceRandom("X11", 1, out (X11));
oku_X12 = plc.ReadDeviceRandom("X12", 1, out (X12));
oku_X13 = plc.ReadDeviceRandom("X13", 1, out (X13));

oku_M1533 = plc.ReadDeviceRandom("M1533", 1, out (M1533));
oku_M1534 = plc.ReadDeviceRandom("M1534", 1, out (M1534));
oku_M1535 = plc.ReadDeviceRandom("M1535", 1, out (M1535));
oku_M1536 = plc.ReadDeviceRandom("M1536", 1, out (M1536));
oku_M1541 = plc.ReadDeviceRandom("M1541", 1, out (M1541));
oku_M1542 = plc.ReadDeviceRandom("M1542", 1, out (M1542));
oku_M1546 = plc.ReadDeviceRandom("M1546", 1, out (M1546));
oku_M1547 = plc.ReadDeviceRandom("M1547", 1, out (M1547));
oku_M1548 = plc.ReadDeviceRandom("M1548", 1, out (M1548));
oku_M1549 = plc.ReadDeviceRandom("M1549", 1, out (M1549));
oku_M1551 = plc.ReadDeviceRandom("M1551", 1, out (M1551));
oku_M1552 = plc.ReadDeviceRandom("M1552", 1, out (M1552));

```
oku_M1553 = plc.ReadDeviceRandom("M1553", 1, out (M1553));  
oku_M1554 = plc.ReadDeviceRandom("M1554", 1, out (M1554));
```

```
oku_Y20 = plc.ReadDeviceRandom("Y20", 1, out (Y20));  
oku_Y21 = plc.ReadDeviceRandom("Y21", 1, out (Y21));  
oku_Y22 = plc.ReadDeviceRandom("Y22", 1, out (Y22));  
oku_Y23 = plc.ReadDeviceRandom("Y23", 1, out (Y23));  
oku_Y24 = plc.ReadDeviceRandom("Y24", 1, out (Y24));  
oku_Y25 = plc.ReadDeviceRandom("Y25", 1, out (Y25));  
oku_Y26 = plc.ReadDeviceRandom("Y26", 1, out (Y26));  
oku_Y27 = plc.ReadDeviceRandom("Y27", 1, out (Y27));  
oku_Y28 = plc.ReadDeviceRandom("Y28 ", 1,out (Y28));  
oku_Y29 = plc.ReadDeviceRandom("Y29", 1, out (Y29));  
oku_Y2A = plc.ReadDeviceRandom("Y2A", 1, out (Y2A));
```

```
oku_Y30 = plc.ReadDeviceRandom("Y30", 1, out (Y30));  
oku_Y31 = plc.ReadDeviceRandom("Y31", 1, out (Y31));  
oku_Y32 = plc.ReadDeviceRandom("Y32", 1, out (Y32));  
oku_Y33 = plc.ReadDeviceRandom("Y33", 1, out (Y33));  
oku_Y34 = plc.ReadDeviceRandom("Y34", 1, out (Y34));  
oku_Y35 = plc.ReadDeviceRandom("Y35", 1, out (Y35));  
oku_Y36 = plc.ReadDeviceRandom("Y36", 1, out (Y36));  
oku_Y37 = plc.ReadDeviceRandom("Y37", 1, out (Y37));  
oku_Y38 = plc.ReadDeviceRandom("Y38", 1, out (Y38));  
oku_Y39 = plc.ReadDeviceRandom("Y39", 1, out (Y39));  
oku_Y3A = plc.ReadDeviceRandom("Y3A ", 1,out (Y3A));  
oku_Y3B = plc.ReadDeviceRandom("Y3B", 1, out (Y3B));  
oku_Y3C = plc.ReadDeviceRandom("Y3C", 1, out (Y3C));  
oku_Y3D = plc.ReadDeviceRandom("Y3D", 1, out (Y3D));  
oku_Y3E = plc.ReadDeviceRandom("Y3E ", 1,out (Y3E));  
oku_Y3F = plc.ReadDeviceRandom("Y3F", 1, out (Y3F));
```

```
oku_D1541 = plc.ReadDeviceBlock("D1541", 1, out (D1541));
```

```
if (D1541 == 1) { ovalShape10.BackColor = Color.Blue; }
```

```
if (D1541 == 2) { ovalShape10.BackColor = Color.Blue;  
ovalShape4.BackColor = Color.Blue; }
```

```
if (D1541 == 3) { ovalShape10.BackColor = Color.Blue;  
ovalShape4.BackColor = Color.Blue; ovalShape8.BackColor = Color.Blue; }
```

```
if (D1541 == 4) { ovalShape10.BackColor = Color.Blue;  
ovalShape4.BackColor = Color.Red; ovalShape8.BackColor = Color.Blue;  
ovalShape5.BackColor = Color.Blue;}
```

```
if (D1541 == 5) { ovalShape10.BackColor = Color.Blue;  
ovalShape4.BackColor = Color.Blue; ovalShape8.BackColor = Color.Blue;  
ovalShape5.BackColor = Color.Blue; ovalShape8.BackColor =  
Color.Blue; }
```

```

        if (D1541 == 6) { ovalShape10.BackColor = Color.Blue;
        ovalShape4.BackColor = Color.Blue; ovalShape8.BackColor = Color.Blue;
            ovalShape5.BackColor = Color.Blue; ovalShape8.BackColor =
Color.Blue; ovalShape3.BackColor = Color.Blue; }

        if (D1541 == 7) { ovalShape10.BackColor = Color.Blue;
        ovalShape4.BackColor = Color.Blue; ovalShape8.BackColor = Color.Blue;
            ovalShape5.BackColor = Color.Blue; ovalShape8.BackColor =
Color.Blue; ovalShape3.BackColor = Color.Blue;
            ovalShape3.BackColor = Color.Blue; }

        if (D1541 == 8) { ovalShape10.BackColor = Color.Blue;
        ovalShape4.BackColor = Color.Blue; ovalShape8.BackColor = Color.Blue;
            ovalShape5.BackColor = Color.Blue; ovalShape8.BackColor =
Color.Blue; ovalShape3.BackColor = Color.Blue;
            ovalShape3.BackColor = Color.Blue; ovalShape9.BackColor =
Color.Blue; }

        textBox1.Text = Convert.ToString(oku_D1541);
        textBox2.Text = Convert.ToString(oku_D1541);

        if (X0 == 1) { rectangleShape13.BackColor = Color.Lime; }
        if (X1 == 1) { rectangleShape14.BackColor = Color.Lime; }
        if (X2 == 1) { rectangleShape15.BackColor = Color.Lime; }
        if (X3 == 1) { rectangleShape16.BackColor = Color.Lime; label98.Visible =
true; }
        if (X4 == 1) { rectangleShape17.BackColor = Color.Lime; label101.Visible =
true; }
        if (X5 == 1) { rectangleShape18.BackColor = Color.Lime;
        rectangleShape81.BackColor = Color.Lime; }
        if (X6 == 1) { rectangleShape19.BackColor = Color.Lime;
        rectangleShape79.BackColor = Color.Lime; }
        if (X7 == 1) { rectangleShape20.BackColor = Color.Lime;
        rectangleShape80.BackColor = Color.Lime; }
        if (X8 == 1) { rectangleShape21.BackColor = Color.Lime;
        rectangleShape88.BackColor = Color.Lime; }
        if (X9 == 1) { rectangleShape22.BackColor = Color.Lime;
        rectangleShape82.BackColor = Color.Lime; }
        if (XA == 1) { rectangleShape23.BackColor = Color.Lime;
        rectangleShape84.BackColor = Color.Lime; }
        if (XB == 1) { rectangleShape24.BackColor = Color.Lime;
        rectangleShape83.BackColor = Color.Lime; }
        if (XC == 1) { rectangleShape25.BackColor = Color.Lime;
        rectangleShape82.BackColor = Color.Lime; }
        if (XD == 1) { rectangleShape26.BackColor = Color.Lime;
        rectangleShape87.BackColor = Color.Lime; }
        if (XE == 1) { rectangleShape27.BackColor = Color.Lime;
        rectangleShape91.BackColor = Color.Lime; }

```

```

        if (XF == 1) { rectangleShape28.BackColor = Color.Lime;
rectangleShape92.BackColor = Color.Lime; }
        if (X10 == 1) { rectangleShape29.BackColor = Color.Lime;
rectangleShape86.BackColor = Color.Lime; }
        if (X11 == 1) { rectangleShape30.BackColor = Color.Lime;
rectangleShape85.BackColor = Color.Lime; }
        if (X12 == 1) { rectangleShape31.BackColor = Color.Lime; }
        if (X13 == 1) { rectangleShape32.BackColor = Color.Lime; }

        if (M1533 == 1) { rectangleShape33.BackColor = Color.Lime; label71.Text
= "UZUN ÜRÜN..."; }
        if (M1534 == 1) { rectangleShape34.BackColor = Color.Lime; label71.Text
= "KISA ÜRÜN..."; }
        if (M1535 == 1) { rectangleShape52.BackColor = Color.Lime; label71.Text
= "METAL DOLDU..."; }
        if (M1536 == 1) { rectangleShape35.BackColor = Color.Lime; label71.Text
= "DERLİN (PLASTİK) ÜRÜN..."; }
        if (M1541 == 1) { rectangleShape36.BackColor = Color.Lime; label71.Text
= "TABLA DOLDU..."; }
        if (M1542 == 1) { rectangleShape37.BackColor = Color.Lime; label71.Text
= "ÜRÜN ONAYLANDI..."; }
        if (M1546 == 1) { rectangleShape38.BackColor = Color.Lime; }
        if (M1547 == 1) { rectangleShape39.BackColor = Color.Lime; }
        if (M1548 == 1) { rectangleShape40.BackColor = Color.Lime; }
        if (M1549 == 1) { rectangleShape41.BackColor = Color.Lime; }
        if (M1551 == 1) { rectangleShape42.BackColor = Color.Lime; }
        if (M1552 == 1) { rectangleShape44.BackColor = Color.Lime; }
        if (M1553 == 1) { rectangleShape45.BackColor = Color.Lime; }
        if (M1554 == 1) { rectangleShape46.BackColor = Color.Lime; }

        if (Y20 == 1) { rectangleShape58.BackColor = Color.Lime; label91.Visible =
true; }
        if (Y21 == 1) { rectangleShape57.BackColor = Color.Lime; label94.Visible =
true; }
        if (Y22 == 1) { rectangleShape56.BackColor = Color.Lime; label95.Visible =
true; }
        if (Y23 == 1) { rectangleShape55.BackColor = Color.Lime; label93.Visible =
true; }
        if (Y24 == 1) { rectangleShape54.BackColor = Color.Lime;
ovalShape11.Visible = true; }
        if (Y25 == 1) { rectangleShape53.BackColor = Color.Lime;
ovalShape12.Visible = true; }
        if (Y26 == 1) { rectangleShape51.BackColor = Color.Lime;
ovalShape13.Visible = true; }
        if (Y27 == 1) { rectangleShape50.BackColor = Color.Lime; label97.Visible =
true; }
        if (Y28 == 1) { rectangleShape49.BackColor = Color.Lime; label99.Visible =
true; }

```

```

        if (Y29 == 1) { rectangleShape48.BackColor = Color.Lime;
rectangleShape90.BackColor = Color.Lime; }
        if (Y2A == 1) { rectangleShape47.BackColor = Color.Lime; }
        if (Y2A == 1) { rectangleShape47.BackColor = Color.Lime; }

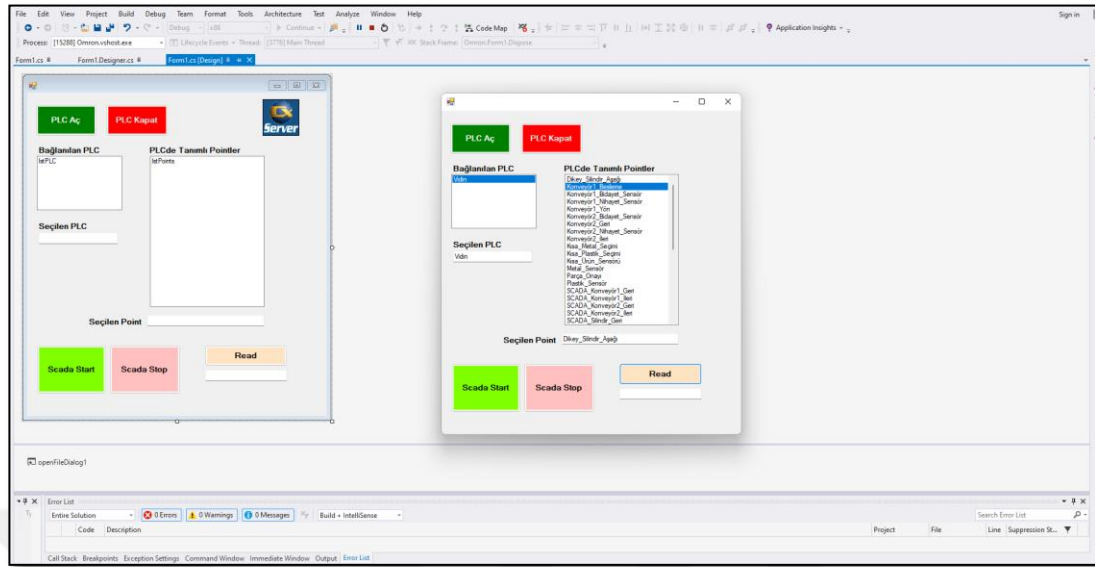
        if (Y30 == 1) { rectangleShape73.BackColor = Color.Lime; }
        if (Y31 == 1) { rectangleShape72.BackColor = Color.Lime; }
        if (Y32 == 1) { rectangleShape71.BackColor = Color.Lime; }
        if (Y33 == 1) { rectangleShape70.BackColor = Color.Lime; }
        if (Y34 == 1) { rectangleShape69.BackColor = Color.Lime; }
        if (Y35 == 1) { rectangleShape68.BackColor = Color.Lime; }
        if (Y36 == 1) { rectangleShape67.BackColor = Color.Lime; }
        if (Y37 == 1) { rectangleShape66.BackColor = Color.Lime; }
        if (Y38 == 1) { rectangleShape65.BackColor = Color.Lime; }
        if (Y39 == 1) { rectangleShape64.BackColor = Color.Lime; }
        if (Y3A == 1) { rectangleShape63.BackColor = Color.Lime; }
        if (Y3B == 1) { rectangleShape62.BackColor = Color.Lime; }
        if (Y3C == 1) { rectangleShape61.BackColor = Color.Lime; }
        if (Y3D == 1) { rectangleShape60.BackColor = Color.Lime; }
        if (Y3E == 1) { rectangleShape59.BackColor = Color.Lime; }
        if (Y3F == 1) { rectangleShape74.BackColor = Color.Lime; }
    }

    private void button24_Click(object sender, EventArgs e)
    {
        //timer1.Tick += new EventHandler(timer1_Tick);
        timer1.Interval = 1;
        timer1.Start();

        button24.BackColor = Color.Green;
        button24.ForeColor = Color.White;
        label102.Text = "SİMÜLATÖR ÇALIŞYOR..";
    }
    private void button25_Click(object sender, EventArgs e)
    {
        timer1.Enabled = false;
        timer1.Stop();
        button25.BackColor = Color.Red;
        button25.ForeColor = Color.White;
        label102.Text = "SİMÜLATÖR DURDURULDU..";
    }
}
}
}

```

PLC-2 SCADA Tasarımı



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
```

```
namespace Omron
```

```
{
    public partial class Form1 : Form
    {
        bool PLCCConnect;
```

```
        public Form1()
        {
            InitializeComponent();
        }
```

```
        private void btnOpenDev_Click(object sender, EventArgs e)
        {
```

```
            PLCCConnect = cxsLiteCtrl1.OpenDevice(txtPLC.Text);
```

```
            if (PLCCConnect == true)
            {
                rectangleShape1.BackColor=Color.Red;
            }
```

```

else
{

rectangleShape1.BackColor=Color.Black;
}
lstPLC.DataSource = cxsLiteCtrl1.Devices;

}

private void Form1_Load(object sender, EventArgs e)
{
    SelectProjectPath();
}

private void SelectProjectPath()
{
    openFileDialog1.Filter = "CXServer Project Files(*.cdm)|*.cdm|All files
(*.*)|*. *";
    DialogResult result = openFileDialog1.ShowDialog();
    if (result == DialogResult.OK && openFileDialog1.FileName != " ")
        cxsLiteCtrl1.ProjectName = openFileDialog1.FileName.ToString();
}

private void btnCloseDev_Click(object sender, EventArgs e)
{
    PLCConnect = cxsLiteCtrl1.CloseDevice(txtPLC.Text);

    txtPLC.Text = "";
    txtPoint.Text = "";

    //DataRowView Secilen = lstPLC.SelectedItem as DataRowView;
    //if (null == Secilen)
    //    return;
    //Secilen.Row.Delete();

    //DataRowView Secilenpoint = lstPLC.SelectedItem as DataRowView;
    //if (null == Secilenpoint)
    //    return;
    //Secilenpoint.Row.Delete();
    lstPLC.DataSource=new string[1];
}

private void lstPLC_SelectedIndexChanged(object sender, EventArgs e)
{

```

```

        string[] pointler = null;
        cxsLiteCtrl1.ListPoints((string)lstPLC.Items[lstPLC.SelectedIndex], out
pointler);
        txtPLC.Text = (string)lstPLC.Items[lstPLC.SelectedIndex];
        lstPoints.DataSource = pointler;

    }

    private void lstPoints_SelectedIndexChanged(object sender, EventArgs e)
    {
        txtPoint.Text = (string)lstPoints.Items[lstPoints.SelectedIndex];
    }

    private void btnScadaStart_Click(object sender, EventArgs e)
    {
        //cxsLiteCtrl1.Write(txtPLC.Text, txtPoint.Text, txtWrite.Text);
        cxsLiteCtrl1.Write(txtPLC.Text, "servoOn", 1);
        gecikme(1);
        cxsLiteCtrl1.Write(txtPLC.Text, "servoOn", 0);
    }

    public void gecikme(int saniye)
    {
        saniye = (saniye + Convert.ToInt32(DateTime.Now.Second));
        for (; )
        {
            if (saniye == DateTime.Now.Second) break;
        }
    }

    private void btnStop_Click(object sender, EventArgs e)
    {
        cxsLiteCtrl1.Write(txtPLC.Text, "ScadaservoOff", 1);
        gecikme(1);
        cxsLiteCtrl1.Write(txtPLC.Text, "ScadaservoOff", 0);
    }

    private void btnRead_Click(object sender, EventArgs e)
    {
        object Value = null;
        bool Quality = false;
        if (cxsLiteCtrl1.Read(txtPLC.Text, txtPoint.Text, out Value, out Quality))
            txtReadVal.Text = Value.ToString();
        gecikme(1);
    }

}
}

```

PTÖ Formları ve Rubrikleri

FORM-2: Öğretmen Gözlem Formu

Haftalık Grup Gözlemleme Kontrol Listesi

Grup Adı: VENTUS Tarih: 23/05/2024

Grubu beş veya on dakika gözlemleyin. Grup üyelerinin katılımını en iyi ifade eden kutuları işaretleyin.	Üyelerin Hepsini	Üyelerin Çoğu	Üyelerin Bazıları	Üyelerin Hiçbiri	Üyeler Hakkında Notlar (Olumsuz Durumlar Üyelerin adları yazılarak değerlendirilir)
Yeni bir göreve başlandığında, grup üyeleri:					
Gruba her gün katılır.	☒	☐	☐	☐	
Haftalık planını verip Kabul eder	☒	☐	☐	☐	
Hemen çalışmaya başlar	☒	☐	☐	☐	
Proje materyallerini çıkarır	☒	☐	☐	☐	
Öğretmenin yardım olmadan bir şeylerin çaresine bakar	☒	☐	☐	☐	
Sorumlulukları paylaşır	☒	☐	☐	☐	
Ana kaynaklara başvurur	☒	☐	☐	☐	
Not alır	☐	☐	☒	☐	
Konuyla alakalı sohbetler yapar	☐	☒	☐	☐	
Yeni bilginin önemini değerlendirir	☐	☒	☐	☐	
Aydınlatıcı soruları sorar	☒	☐	☐	☐	
Birbirine konuşma fırsatı tanır	☒	☐	☐	☐	
Etkili kararlar verir	☐	☒	☐	☐	
Planları ve kararları kaydeder	☐	☐	☒	☐	
Temel bilgileri paylaşır	☒	☐	☐	☐	
Göreve bağlı kalır	☒	☐	☐	☐	

Değerlendirmeler: Genel olarak uyumlu çalıştılar. 15 dakikalık ve özverili bir çalışma yaptılar

FORM-4: İşbirlikçi Öğrenme Formu

İşbirlikçi Öğrenme Değerlendirme Formu... Süreç

Grup: VENTUS

Tarih:

Grup Üyeleri:

	Mükemmel	Takdire Değer	Kabul Edilebilir	Amatörce
Grup Katılımı	Bütün öğrenciler istekli bir şekilde katılır	En az 4/3 öğrenci aktif olarak katılır	En az öğrencilerin yarısı bir fikir danışır veya sunar;	Sadece bir veya iki kişi aktif olarak katılır
Paylaşılan Sorumluluk	Sorumluluk görev için eşit olarak paylaşılmıştır	Sorumluluk çoğu grup üyesi tarafından paylaşılmıştır	Sorumluluk grup üyelerinin 1/2'si tarafından paylaşılmıştır	Özel olarak tek bir kişiye güvenilmesi
Etkileşimin Kalitesi	Mükemmel dinleme ve liderlik yetenekleri sergilenmiştir; öğrenciler tartışmalarında başkalarının görüş ve fikirlerini yansıtır	Öğrenciler etkileşime adapte olur; heyecanlı bir şekilde görevin ana noktalarını tartışır	Etkileşim için kısıtlı beceri; özenli dinleme; tartışma ve alternatifleri için az sayıda örnek	Az etkileşim; çok kısa sohbetler; bazı öğrenciler kayıtsız veya dikkatsizdir
Grup içindeki Roller	Her öğrenciye açıkça belirlenmiş roller atandı; grup üyeleri roller etkin olarak yerine getirdi	Her öğrenciye bir rol atandı fakat roller açık olarak belirlenmemiştir veya düzenli olarak sadık kalmadı	Öğrencilere roller atandı fakat rollere düzenli olarak sadık kalmadı	Grup üyelerine rol atamak için hiçbir çaba harcanmadı

FORM-5: Proje Değerlendirme Formu

Grup: VENTUS

Tarih: 23/5/24

Üyeler:

No	Standartlar	Evet	Hayır
1	Grafçet diyagramları doğru hazırlanmış mı?	X	
2	PLC programı Grafçet diyagramlarına uygun yazılmış mı?	X	
3	Konveyör üzerinde ürün sensörler ile algılanıyor mu?	X	
4	Konveyör ile ürün taşınabiliyor mu?	X	
5	Konveyör üzerinde uzun-kısa ürün ayrımı yapılıyor mu?	X	
6	Konveyör üzerinde plastik-metal ürün ayrımı yapılıyor mu?	X	
7	Konveyörler arası ürün taşıma yapılıyor mu?	X	
8	Konveyör-Döner Tabla arasında ürün taşıma yapılıyor mu?	X	
9	Döner tabla kontrolü doğru bir şekilde gerçekleştiriliyor mu?	X	
10	SCADA programı düzgün çalışıyor mu?	X	

Proje Raporu Değerlendirmesi

Proje Raporunda Dikkat Edilmesi Gereken Hususlar:

1. Sayfa yapısı alt_üst ve iki yandan 2 cm boşluklu olacak. Başlıklar 14 punto Times New Roman ve gövde metni Times New Roman 12 punto olacak.
2. Proje Kapağında okul logosu, ders başlığı, proje adı, grup üyeleri ve öğretmen isimleri olacak (Size format verilecektir.)
3. Size verilecek başlıklar proje raporunda olmalıdır. (Sizlere detaylı bilgi verilecektir.)
4. Proje Raporunuz ve Ürün değerlendirmeniz 2. Performans notunuza %30 etki edecektir.

No	Standartlar	Evet	Hayır
1	Rapor kapağı verilen standartlara göre mi yazılmış?	☒	☐
2	Rapor içeriği istenilen başlıklara göre mi yazılmış?	☒	☐
3	Rapor içeriği istenilen yazım ve sayfa formatına göre mi yazılmış?	☒	☐
4	Raporda "İçindekiler" ve "Kaynakça" tabloları var mı?	☒	☐
5	Raporda kullanılan malzemeler tanıtılmış mı?	☒	☐
6	Raporda <u>Grafçet</u> diyagramları var mı?	☒	☐
7	Raporda <u>Grafçet</u> uyumlu PLC yazılımı var mı?	☒	☐
8	Raporda C# ile SCADA yazılımları var mı?	☒	☐
9	Raporda resim ve şekillere yer verilmiş mi?	☒	☐
10	Raporun tertip, temizlik ve düzeni iyi mi?	☒	☐

FORM-6: Sunum Değerlendirme Formu

POREJE SUNUMU DEĞERLENDİRMESİ

Sunumda, ekip tarafından keşfedilen bilgiler, planlar, taslak çalışmalar, ürün ve projenin birleştirilmiş son durumu, sonuç ve öneriler yer almalıdır.

KATEGORİ	4	3	2	1
Sözlü sunum: Hazır olmak	Ekip tamamen hazır ve önceden prova yaptı.	Ekip hazır gözüküyordu ama birkaç provaya daha gereksinimi vardı.	Ekip biraz hazır gibiydi ama yine de prova olmalıydı.	Ekip sunum yapmaya hiç hazır değildi.
Sözlü sunum: İsteklilik	Ekibin jest ve mimikleri ve vücut dili, konuyla çok ilgili ve istekli olduklarını açıkça gösteriyordu.	Ekibin jest ve mimikleri ve vücut dili, genellikle konuyla ilgili ve istekli olduklarını gösteriyordu.	Ekibin jest ve mimikleri ve vücut dili, konuyla çok ilgili ve istekli gözükmeğe çalışıyordu ama pek başarılı değildi.	Çok az jest, mimik ve vücut dili kullanıldı. Ekip açıkça konuya ilgisizdi.
Bütünlük	Sunumda konuyla ilgili gerekebilecek ve rahatlık sağlayacak tüm materyal mevcuttu.	Sunumda konuyla ilgili gerekebilecek ve rahatlık sağlayacak çoğu materyal mevcuttu, bir iki eksik vardı.	Sunumda birden fazla kilit öge eksikliği vardı.	Sunumda pek çok eksik ve tutarsızlık vardı.
Arka plan bilgisi	Sunum, keşfedilen bilgileri özetleyen 10 slayt içeriyor. Taslak, çizim, görsel vb. mevcuttu.	Sunum, keşfedilen bilgileri özetleyen 7 tane slayt içeriyor. Taslak, çizim, görsel vb. mevcuttu.	Sunum, keşfedilen bilgileri özetleyen 5 tane slayt içeriyor. Taslak, çizim, görsel vb. mevcuttu.	Sunum, keşfedilen bilgileri özetleyen 1-4 adet slayt içeriyor. Sunumumuzda taslak, çizim, görsel vb. yoktu

FORM-3: Öğrenci Öz Değerlendirme Formu

Proje: Deney setinde grasscot ve c# ile geliştirme

Grup Adı: Ventus Automation

Tarih: 23.05.2024

Öğrenci Adı Soyadı: Batuhan Özcan

Ortak Çalışmaya Dayalı Öz/Akran Değerlendirme Formu

Talimatlar:

Lütfen grubunuzdaki katılıma dair fikirlerinizi paylaşın. Her bir kategori için, bir ekip lideri olarak etkililiğinizi en iyi yansıtan açıklamayı daire içine alın. Ardından arkadaki soruları yanıtlayın.



KATEGORİ	4	3	2	1
Katkılar	Gruba ve sınıf tartışmasına katılım gösterirken <u>sürekli olarak</u> faydalı fikirler sundum. Oldukça çaba gösterip katkıda bulunan tam anlamıyla bir liderdim.	Gruba ve sınıf tartışmasına katılım gösterirken <u>genellikle</u> faydalı fikirler sundum. Oldukça çaba gösteren güçlü bir liderdim.	Gruba ve sınıf tartışmasına katılım gösterirken <u>bazen</u> faydalı fikirler sundum. Gerekeni yapan, yeterli olan bir liderdim.	Gruba ve sınıf tartışmasına katılım gösterirken <u>nadiren</u> faydalı fikirler sundum. Bazen katılım göstermeyi reddettim.
Diğerleriyle birlikte çalışma	Diğerlerini daima dinledim, onlarla paylaşımda bulundum ve onların çabalarını hep destekledim. İnsanların birlikte iyi çalışmaya devam etmesi için çaba gösterdim.	Diğerlerini çoğunlukla dinledim, onlarla paylaşımda bulundum ve onların çabalarını destekledim. Grupta "çalkantıya" sebep olmadım.	Diğerlerini sık sık dinledim, onlarla paylaşımda bulundum ve onların çabalarını destekledim, ama bazen iyi bir ekip üyesi olmadığım oldu.	Diğerlerini nadiren dinledim, onlarla nadiren paylaşımda bulundum ve onların çabalarını çok az destekledim. Çoğunlukla pek iyi bir ekip üyesi değildim.
Problemleri çözme	Aktif olarak sorunlara çözüm aradım ve tavsiyelerde bulundum.	Diğerlerinin sunduğu çözümleri önerilerini geliştirdim.	Çözüm sunmadım veya diğerlerininkini geliştirmedim ancak diğerlerinin sunacakları çözüm önerilerini denemeye istekliydim.	Sorunları çözmeyle denemedim veya diğerlerine bunun için yardımcı olmadım. Sorunları çözmeyle başkasına bıraktım.

FORM-3: Öğrenci Öz Değerlendirme Formu

Göreve odaklanma	Göreve ve yapılması gerekene hep odaklandım. Kendimi oldukça iyi yönlendirip yönetebiliyordum.	Çoğu zaman göreve ve yapılması gerekene odaklandım. Diğer grup üyeleri bana güvenebiliyordu.	Göreve ve yapılması gerekene zaman zaman odaklandım. Diğer grup üyelerinin zaman zaman bana odaklanmam gerektiğini hatırlattığı oluyordu.	Göreve ve yapılması gerekenlere nadiren odaklanabildim. İş diğerleri yaptı.
Tavır, yaklaşım	Diğer grup üyelerinin işlerini asla ulu orta eleştirmedim. Görevle ilgili hep olumlu yaklaştım.	Diğer grup üyelerinin işlerini nadiren ulu orta eleştirdim. Görevle ilgili genellikle olumlu yaklaştım.	Diğer grup üyelerinin işlerini bazen ulu orta eleştirdim. Görevle ilgili nispeten olumlu yaklaştım.	Diğer grup üyelerinin işlerini sık sık ulu orta eleştirdim. Görevle ilgili genellikle olumsuz yaklaştım.

1. Görev boyunca üstünüze aldığınız roller ve sorumluluklar neydi? Gruptaki işiniz neydi?

C# ve agortadim grafceti çizdik yaptık scada'yı yaptık PLC yaptık

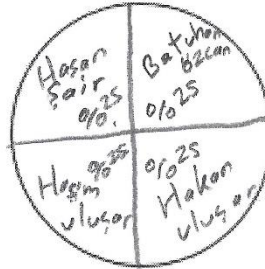
2. Sizin ve grubunuzun iyi yaptığını düşündüğünüz şeyler nelerdir?

C# (visual studio) , grafcet

3. Siz ve grubunuz daha etkili olarak nasıl çalışabilirdi?

Zamanı daha iyi kullanabiliydik

4. Lütfen beraber çalışırken grubunuzun her üyesinin ne ölçüde iş yaptığını temsil etmesi için aşağıdaki daire grafiğini boyayın. Lütfen her bölümde atıfta bulunduğunuz kişinin adını belirtin.



FORM-3: Öğrenci Öz Değerlendirme Formu

Proje: C#ta SCADA
Grup Adı: Ventus Automation
Tarih: 23.05-2024
Öğrenci Adı Soyadı: Huban ULUÇAR

Ortak Çalışmaya Dayalı Öz/Akıran Değerlendirme Formu

Talimatlar:

Lütfen grubunuzdaki katılımlarınıza dair fikirlerinizi paylaşın. Her bir kategori için, bir ekip lideri olarak etkililiğinizi en iyi yanıtıtan açıklamayı daire içine alın. Ardından arkadaki soruları yanıtlayın.

KATEGORI	4	3	2	1
Katkılar	Gruba ve sınıf tartışmasına katılım gösterirken sürekli olarak faydalı fikirler sundum. Oldukça çaba gösterip katkıda bulunan tüm anılarıyla bir liderdim.	Gruba ve sınıf tartışmasına katılım gösterirken genellikle faydalı fikirler sundum. Oldukça çaba gösteren güçlü bir liderdim.	Gruba ve sınıf tartışmasına katılım gösterirken bazen faydalı fikirler sundum. Gerekeni yapan, yeterli olan bir liderdim.	Gruba ve sınıf tartışmasına katılım gösterirken nadiren faydalı fikirler sundum. Bazen katılım göstermeyi reddedtim.
Diğerleriyle birlikte çalışma	Diğerlerini daima dinledim, onlarla paylaşımda bulundum ve onların çabalarını hep destekledim. İnsanların birlikte iyi çalışmaya devam etmesi için çaba gösterdim.	Diğerlerini çoğunlukla dinledim, onlarla paylaşımda bulundum ve onların çabalarını destekledim. Grupta "çalkantiya" sebep olmadım.	Diğerlerini sık sık dinledim, onlarla paylaşımda bulundum ve onların çabalarını destekledim, ama bazen iyi bir ekip üyesi olmadığım oldu.	Diğerlerini nadiren dinledim, onlarla nadiren paylaşımda bulundum ve onların çabalarını çok az destekledim. Çoğunlukla pek iyi bir ekip üyesi değildim.
Problemi çözme	Aktif olarak sorunlara çözüm aradım ve tavsiyelerde bulundum.	Diğerlerinin sunduğu çözüm önerilerini geliştirdim.	Çözüm sunmadım veya diğerlerininkini geliştirmedim ancak diğerlerinin sunacakları çözüm önerilerini denemeye istekliydim.	Sorunları çözme denemedim veya diğerlerine bunun için yardımcı olmadım. Sorunları çözme başkasına bıraktım.

FORM-3: Öğrenci Öz Değerlendirme Formu

Göreve odaklanma	Göreve ve yapılması gerekene hep odaklandım. Kendimi oldukça iyi yönlendirip yönetebiliyordum.	Çoğu zaman göreve ve yapılması gerekene odaklandım. Diğer grup üyeleri bana güvenebiliyordu.	Göreve ve yapılması gerekene zaman zaman odaklandım. Diğer grup üyelerinin zaman zaman bana odaklanmam gerektiğini hatırlattığı oluyordu.	Göreve ve yapılması gerekenlere nadiren odaklanabildim. İşleri diğerleri yaptı.
Tavır, yaklaşım	Diğer grup üyelerinin işlerini asla ulu orta eleştirmedim. Görevle ilgili hep olumlu yaklaştım.	Diğer grup üyelerinin işlerini nadiren ulu orta eleştirdim. Görevle ilgili genellikle olumlu yaklaştım.	Diğer grup üyelerinin işlerini bazen ulu orta eleştirdim. Görevle ilgili nispeten olumlu yaklaştım.	Diğer grup üyelerinin işlerini sık sık ulu orta eleştirdim. Görevle ilgili genellikle olumsuz yaklaştım.

1. Görev boyunca üstünüze aldığınız roller ve sorumluluklar neydi? Gruptaki işiniz neydi?

Bütün görevler ortak. Herseyi beraber yaptık.

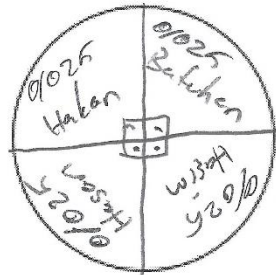
2. Sizin ve grubunuzun iyi yaptığını düşündüğünüz şeyler nelerdir?

PLC, SCADA, Grafset, KazimCC#, Python-

3. Siz ve grubunuz daha etkili olarak nasıl çalışabilirdi?

Zamanımızı daha doğru kullanabilirdik.

4. Lütfen beraber çalışırken grubunuzun her üyesinin ne ölçüde iş yaptığını temsil etmesi için aşağıdaki daire grafiğini boyayın. Lütfen her bölümde atıfta bulunduğunuz kişinin adını belirtin.



FORM-3: Öğrenci Öz Değerlendirme Formu

Proje: C#'da Sılda
Grup Adı: Ventus Automation
Tarih: 23/05/2024
Öğrenci Adı Soyadı: Hasan S.Ör

Ortak Çalışmaya Dayalı Öz/Akran Değerlendirme Formu

Talimatlar:

Lütfen grubunuzdaki katılımlarınıza dair fikirlerinizi paylaşın. Her bir kategori için, bir ekip lideri olarak etkililiğinizi en iyi yansıtan açıklamayı daire içine alın. Ardından arkadaki soruları yanıtlayın.



KATEGORİ	4	3	2	1
Katkılar	Gruba ve sınıf tartışmasına katılım gösterirken <u>sürekli olarak</u> faydalı fikirler sundum. Oldukça çaba gösterip katkıda bulunan tam anlamıyla bir liderdim.	Gruba ve sınıf tartışmasına katılım gösterirken <u>genellikle</u> faydalı fikirler sundum. Oldukça çaba gösteren güçlü bir liderdim.	Gruba ve sınıf tartışmasına katılım gösterirken <u>bazen</u> faydalı fikirler sundum. Gerekeni yapan, yeterli olan bir liderdim.	Gruba ve sınıf tartışmasına katılım gösterirken <u>nadiren</u> faydalı fikirler sundum. Bazen katılım göstermeyi reddedtim.
Diğerleriyle birlikte çalışma	Diğerlerini daima dinledim, onlarla paylaşımda bulundum ve onların çabalarını hep destekledim. İnsanların birlikte iyi çalışmaya devam etmesi için çaba gösterdim.	Diğerlerini çoğunlukla dinledim, onlarla paylaşımda bulundum ve onların çabalarını destekledim. Grupta "çalkantiya" sebep olmadım.	Diğerlerini sık sık dinledim, onlarla paylaşımda bulundum ve onların çabalarını destekledim, ama bazen iyi bir ekip üyesi olmadığım oldu.	Diğerlerini nadiren dinledim, onlarla nadiren paylaşımda bulundum ve onların çabalarını çok az destekledim. Çoğunlukla pek iyi bir ekip üyesi değildim.
Problemi çözüme	Aktif olarak sorunlara çözüm aradım ve tavsiyelerde bulundum.	Diğerlerinin sunduğu çözüm önerilerini geliştirdim.	Çözüm sunmadım veya diğerlerininkini geliştirmedim ancak diğerlerinin sunacakları çözüm önerilerini denemeye istekliydim.	Sorunları çözmeyi denemedim veya diğerlerine bunun için yardımcı olmadım. Sorunları çözmeyi başkasına bıraktım.

FORM-3: Öğrenci Öz Değerlendirme Formu

Göreve odaklanma	Göreve ve yapılması gerekene hep odaklandım. Kendimi oldukça iyi yönlendirip yönetebiliyordum.	Çoğu zaman göreve ve yapılması gerekene odaklandım. Diğer grup üyeleri bana güvenebiliyordu.	Göreve ve yapılması gerekene zaman zaman odaklandım. Diğer grup üyelerinin zaman zaman bana odaklanmam gerektiğini hatırlattığı oluyordu.	Göreve ve yapılması gerekene nadiren odaklanabildim. İş diğerleri yaptı.
Tavır, yaklaşım	Diğer grup üyelerinin işlerini asla ulu orta eleştirmedim. Görevle ilgili hep olumlu yaklaştım.	Diğer grup üyelerinin işlerini nadiren ulu orta eleştirdim. Görevle ilgili genellikle olumlu yaklaştım.	Diğer grup üyelerinin işlerini bazen ulu orta eleştirdim. Görevle ilgili nispeten olumlu yaklaştım.	Diğer grup üyelerinin işlerini sık sık ulu orta eleştirdim. Görevle ilgili genellikle olumsuz yaklaştım.

1. Görev boyunca üstünüze aldığınız roller ve sorumluluklar neydi? Gruptaki işiniz neydi?

Grup Lideri, Grup Sorumlusu, Grup Sorumlusu

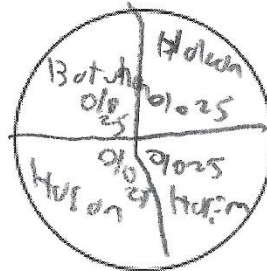
2. Sizin ve grubunuzun iyi yaptığını düşündüğünüz şeyler nelerdir?

Konuyla ilgili bir şey yaptığımızı düşünmüştük

3. Siz ve grubunuz daha etkili olarak nasıl çalışabilirdi?

Biraz daha verimli oluna bilirdi

4. Lütfen beraber çalışırken grubunuzun her üyesinin ne ölçüde iş yaptığını temsil etmesi için aşağıdaki daire grafiğini boyayın. Lütfen her bölimde atıfta bulunduğunuz kişinin adını belirtin.



FORM-3: Öğrenci Öz Değerlendirme Formu

Proje: SCADA

Grup Adı: Ventus Automation

Tarih: 23.05.2024

Öğrenci Adı Soyadı: Haşim Vluğar

Ortak Çalışmaya Dayalı Öz/Akran Değerlendirme Formu

Talimatlar:

Lütfen grubunuzdaki katılımlarınıza dair fikirlerinizi paylaşın. Her bir kategori için, bir ekip lideri olarak etkililiğinizi en iyi yansıtan açıklamayı daire içine alın. Ardından arkadaki soruları yanıtlayın.



KATEGORI	4	3	2	1
Katkılar	Gruba ve sınıf tartışmasına katılım gösterirken <u>sürekli olarak</u> faydalı fikirler sundum. Oldukça çaba gösterip katkıda bulunan tam anlamıyla bir liderdim.	Gruba ve sınıf tartışmasına katılım gösterirken <u>genellikle</u> faydalı fikirler sundum. Oldukça çaba gösteren güçlü bir liderdim.	Gruba ve sınıf tartışmasına katılım gösterirken <u>bazen</u> faydalı fikirler sundum. Gerekeni yapan, yeterli olan bir liderdim.	Gruba ve sınıf tartışmasına katılım gösterirken <u>nadiren</u> faydalı fikirler sundum. Bazen katılım göstermeyi reddettim.
Diğerleriyle birlikte çalışma	Diğerlerini daima dinledim, onlarla paylaşımda bulundum ve onların çabalarını hep destekledim. İnsanların birlikte iyi çalışmaya devam etmesi için çaba gösterdim.	Diğerlerini çoğunlukla dinledim, onlarla paylaşımda bulundum ve onların çabalarını destekledim. Grupta "çalkanıya" sebep olmadım.	Diğerlerini sık sık dinledim, onlarla paylaşımda bulundum ve onların çabalarını destekledim, ama bazen iyi bir ekip üyesi olmadığım oldu.	Diğerlerini nadiren dinledim, onlarla nadiren paylaşımda bulundum ve onların çabalarını çok az destekledim. Çoğunlukla pek iyi bir ekip üyesi değildim.
Problem çözme	Aktif olarak sorunlara çözüm aradım ve tavsiyelerde bulundum.	Diğerlerinin sunduğu çözüm önerilerini geliştirdim.	Çözüm sunmadım veya diğerlerininkini geliştirmedim ancak diğerlerinin sunacakları çözüm önerilerini denemeye istekliydim.	Sorunları çözme denemedim veya diğerlerine bunun için yardımcı olmadım. Sorunları çözme başkasına bıraktım.

FORM-3: Öğrenci Öz Değerlendirme Formu

Göreve odaklanma	Göreve ve yapılması gerekene hep odaklandım. Kendimi oldukça iyi yönlendirip yönetebiliyordum.	Çoğu zaman göreve ve yapılması gerekene odaklandım. Diğer grup üyeleri bana güvenebiliyordu.	Göreve ve yapılması gerekene zaman zaman odaklandım. Diğer grup üyelerinin zaman zaman bana odaklanmam gerektiğini hatırlattığı oluyordu.	Göreve ve yapılması gerekenlere nadiren odaklanabildim. İş diğerleri yaptı.
Tavır, yaklaşım	Diğer grup üyelerinin işlerini asla ulu orta eleştirmedim. Görevle ilgili hep olumlu yaklaştım.	Diğer grup üyelerinin işlerini nadiren ulu orta eleştirdim. Görevle ilgili genellikle olumlu yaklaştım.	Diğer grup üyelerinin işlerini bazen ulu orta eleştirdim. Görevle ilgili nispeten olumlu yaklaştım.	Diğer grup üyelerinin işlerini sık sık ulu orta eleştirdim. Görevle ilgili genellikle olumsuz yaklaştım.

1. Görev boyunca üstünüze aldığınız roller ve sorumluluklar neydi? Gruptaki işiniz neydi?

Grafet çizimi ve kod üzerinde çalışma

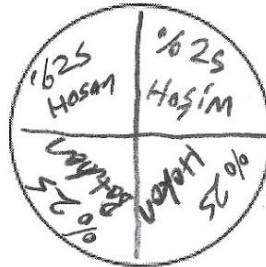
2. Sizin ve grubunuzun iyi yaptığını düşündüğünüz şeyler nelerdir?

plc kodunu yazma

3. Siz ve grubunuz daha etkili olarak nasıl çalışabilirdi?

Zamanı daha iyi değerlendire bilirdik

4. Lütfen beraber çalışırken grubunuzun her üyesinin ne ölçüde iş yaptığını temsil etmesi için aşağıdaki daire grafiğini boyayın. Lütfen her bölümde atıfta bulunduğunuz kişinin adını belirtin.



Özgeçmiş

Adı Soyadı: **İbrahim Aydın**

Eğitim:

1996–2000 Marmara Ü. Teknik Eğitim Fak. Bilgisayar ve Kontrol Öğretmenliği

2022–2024 İzmir Kâtip Çelebi Üniversitesi, Fen Bilimleri Enstitüsü, Yazılım Müh.

İş Deneyimi:

2002 – 2006 Bayrampaşa Saraybosna Mesleki ve Teknik Anadolu Lisesi

2006 – 2010 Ödemiş İlkurşun Mesleki ve Teknik Anadolu Lisesi

2010– Bornova Mazhar Zorlu Mesleki ve Teknik Anadolu Lisesi

Yayınlar:

1. Aydın, İ. (2023, Temmuz). Ağırlığına Göre Ürün Ayırabilen Endüstri 4.0 Tabanlı Akıllı Fabrika SCADA Eğitim Setinin Tasarımı ve Gerçekleştirilmesi. III. Uluslararası Yapay Zekâ ve Veri Bilimi Kongresi'nde sunulan bildiri, İzmir Kâtip Çelebi Üniversitesi, Fen Bilimleri Enstitüsü, İzmir.

2. Aydın, İ. , Bulut, V. (2024, Mayıs). Proje Tabanlı Öğrenme Müfredatlı Endüstriyel Otomasyon Eğitim Setinin İmalatı ve Programlanması. IV. Uluslararası Eğitim Teknolojisi ve Çevrimiçi Öğrenme Konferansı'nda sunulan bildiri, Anadolu Üniversitesi, Eskişehir.