

KOCAELİ ÜNİVERSİTESİ * FEN BİLİMLERİ ENSTİTÜSÜ

**PLC VE SCADA SİSTEMLERİNİN BİR MDF PRESİ DIŞ SAHA
BESLEME HATTINA UYGULANMASI**

YÜKSEK LİSANS TEZİ

Murat KILIÇ

**Anabilim Dalı: Elektrik Eğitimi
Danışman: Yrd. Doç. Dr. Şule ÖZDEMİR**

KOCAELİ, 2010

KOCAELİ ÜNİVERSİTESİ * FEN BİLİMLERİ ENSTİTÜSÜ

**PLC VE SCADA SİSTEMLERİNİN BİR MDF PRESİ DIŞ SAHA
BESLEME HATTINA UYGULANMASI**

YÜKSEK LİSANS TEZİ

Murat KILIÇ

Tezin Enstitüye Verildiği Tarih: 01 Haziran 2010

Tezin Savunulduğu Tarih: 21 Haziran 2010

Tez Danışmanı

Yrd.Doç.Dr. Şule ÖZDEMİR


(.....)

Üye

Doç.Dr. Ercüment KARAKAŞ


(.....)

Üye

Doç.Dr. Mehmet UZUNOĞLU


(.....)

KOCAELİ, 2010

ÖNSÖZ VE TEŞEKKÜR

Güç ve proses kontrolünün yapıldığı her yerde sistem koordinasyonlarını sağlamak için PLC ve SCADA sistemleri kullanılmaktadır. Endüstriyel otomasyon, üretim yapan firmaların üretim kalitesini, hızını arttırmakta ve üretim maliyetlerini düşürmektedir. PLC ve SCADA ile ülkemizde akıllı bina otomasyonu, aydınlatma otomasyonu, enerji üretimi ve taşınması otomasyonu, petrol taşınması otomasyonu ve birçok farklı üretim sistemlerinin otomasyonu yapılmaktadır.

Tez çalışmamda her zaman yanımda olan ve beni yönlendiren danışman hocam Sayın Yrd. Doç. Dr. Şule ÖZDEMİR'e, çalışmalarımı gerçekleştirmemde bana yardımcı olan EGDAŞ firması genel müdürü Hakan YARDEMİR'e, Elektrik Mühendisi İlhan YAKAY'a, Muhammet AYTEKİN'e ve çalışma arkadaşlarıma, maddi ve manevi olarak beni her zaman destekleyen değerli eşim Tuğba KILIÇ'a ve aileme teşekkürlerimi sunarım.

İÇİNDEKİLER

ÖNSÖZ VE TEŞEKKÜR	i
İÇİNDEKİLER	ii
ŞEKİLLER DİZİNİ.....	v
KISALTMALAR DİZİNİ.....	vii
SİMGELER DİZİNİ.....	vii
ÖZET.....	viii
İNGİLİZCE ÖZET.....	ix
1. GİRİŞ.....	1
2. PROGRAMLANABİLİR MANTIK DENETLEYİCİLER (PLC)	2
2.1. PLC'nin Yapısı ve Çalışma Prensipleri	2
2.1.1. Bellek yapısı	5
2.1.2. Veri belleği ve adresleme	8
2.2. Sembolik Adresleme	9
2.3. PLC Programı Yazım Dilleri	12
2.4. PLC Programlama Yöntemleri.....	12
2.5. S7-300 PLC'de Kart Yapılandırılması.....	13
3. SIMATIC S7 PROGRAMININ YAPISI.....	14
3.1. Program Blokları	15
3.1.1. Organizasyon blokları (OBn)	15
3.1.1.1. Günün belli saatlerinde oluşan kesme organizasyon bloğu (OB10..OB17)	18
3.1.1.2. Belli bir zaman sonraki kesme organizasyon bloğu (OB20..OB23).....	19
3.1.1.3. Çevrimsel kesme organizasyon bloğu(OB30..OB38).....	20
3.1.1.4. Donanım kesmesi organizasyon bloğu (OB40..OB47)	21
3.1.1.5. Zaman hatası organizasyon bloğu (OB80)	21
3.1.1.6. Besleme gerilimi hatası organizasyon bloğu (OB81)	22
3.1.1.7. Hata bulgusu uyarı organizasyon bloğu (OB82)	22
3.1.1.8. Takma/Çıkarma uyarısı organizasyon bloğu (OB83).....	23
3.1.1.9. CPU donanım hatası uyarısı organizasyon bloğu (OB84).....	23
3.1.1.10. Program yürütme hatası organizasyon bloğu (OB85)	23
3.1.1.11. Modül taşıyıcı (montaj rayı) hatası organizasyon bloğu (OB86)	24
3.1.1.12. Haberleşme hatası organizasyon bloğu (OB87)	24
3.1.1.13. Programlama hatası organizasyon bloğu (OB121).....	24
3.1.1.14. Veri erişim hatası organizasyon bloğu (OB 122)	24
3.1.2. Fonksiyon (FC) ve fonksiyon blokları (FB).....	25
3.1.2.1. Fonksiyonlar (FC).....	25
3.1.2.2. Fonksiyon blokları (FB).....	26
3.1.3. Veri blokları (DB)	26

4. SCADA SİSTEMLERİ.....	28
4.1. Giriş.....	28
4.2. SCADA Sisteminin Yapısı	30
4.2.1. Uzak uç birim (Remote terminal unit: RTU)	31
4.2.2. İletişim sistemi	33
4.2.3. Kontrol Merkezi Sistemi (Ana Kontrol Merkezi AKM – Master Terminal Unit MTU)	34
4.3. SCADA Sisteminin Yararları.....	35
4.4. SCADA Sisteminin Fonksiyonları.....	35
5. WINCC SCADA PROGRAMI	37
5.1. Giriş.....	37
5.2. WINCC SCADA Programı Ana Öğeleri	38
5.2.1. Bilgisayar (Computer).....	38
5.2.2. Etiket yöneticisi (Tag management).....	39
5.2.3. Yapısal etiket (Structure tag).....	39
5.2.4. Grafik oluşturucu (Graphics designer)	40
5.2.5. Alarm kaydedici (Alarm logging)	40
5.2.6. Etiket kaydedici (Tag logging).....	40
5.2.7. Rapor düzenleyici (Report designer).....	42
5.2.8. Global komut editörü (Global script)	42
5.2.9. Metin kütüphanesi (Text library)	43
5.2.10. Metin dağıtıcı (Text distributor)	43
5.2.11. Kullanıcı yetkisi (User administrator)	43
5.2.12. Hiyerarşi yönetimi (Picture tree manager).....	43
5.2.13. Haberleşme ağı görüntüleme (Lifebeat monitoring)	44
5.2.14. Operatör sistemi proje editörü (OS project editor)	44
6. BİR MDF PRESİ DIŞ SAHA BESLEME HATTININ SIMATIC STEP-7 İLE PROGRAMLANMASI VE WINCC İLE SCADA PROGRAMININ YAZILMASI... ..	45
6.1. MDF Presi Dış Saha Besleme Hattının Yapısı	45
6.2. MDF Presi Dış Saha Hattının Haberleşme Kurgusu.....	47
6.3. 1300 Sistemi (Yerden Besleme) Akış Diyagramları.....	49
6.3.1. 1300 Grubu (Yerden Cips Besleme) Akış Diyagramı.....	49
6.3.2. 1600 Grubu (Yerden Toz Besleme) Akış Diyagramı.....	50
6.4. 1300 Sistemi (Yerden Besleme) Siemens Simatic Step-7 ile Programının Yazılması.....	51
6.4.1. Motor kontrol bloklarının hazırlanması	54
6.4.2. 1300 Grubu (Yerden cips besleme) ve 1600 Grubunun (Yerden toz besleme) kontrolü.....	56
6.5. 1300 Sistemi (Yerden Besleme) Siemens WINCC ile SCADA Programının Yazılması.....	58
6.5.1. 1300 grubu (yerden cips besleme) kontrol sayfası.....	58
6.5.2. 1600 grubu (yerden toz besleme) kontrol sayfası	60
6.5.3. Sistem durum sayfası.....	61

6.5.4. Akım, sıcaklık ve basınç grafik sayfaları	62
6.5.5. Akım, sıcaklık ve basınç tablo sayfaları.....	64
SONUÇLAR ve ÖNERİLER.....	65
KAYNAKLAR	67
EKLER.....	69
ÖZGEÇMİŞ	138

ŞEKİLLER DİZİNİ

Şekil 2.1: PLC'nin genel yapısı	5
Şekil 2.2: PLC'nin bellek yapısı	6
Şekil 2.3: PLC'de adres yapısı	8
Şekil 2.4: Sembol tablosu.....	10
Şekil 2.5 Lokal sembol kullanımı blok iç yapısı.....	11
Şekil 2.6: Lokal sembol kullanımı blok dış yapısı.....	11
Şekil 2.7: IM Modülleri ile kart dizilimi	13
Şekil 3.1: Simatic S7 programının blok yapısı	14
Şekil 3.2: Simatic S7 programının blok yapısı	16
Şekil 4.1: SCADA sisteminin genel yapısı	31
Şekil 4.2: SCADA'da iletişim sisteminin yapısı	33
Şekil 5.1: WINCC programının genel görüntüsü.....	38
Şekil 6.1: MDF presi dış saha besleme hattı prensip şeması	46
Şekil 6.2: MDF presi dış saha besleme hattının bilgisayar – PLC ve uzak işlemci haberleşme kurgusu (Ethernet, Fiberoptik, Profibus)	47
Şekil 6.3: MDF presi dış saha besleme hattı PLC ve bilgisayarları arasındaki Fiberoptik, Ethernet haberleşmesi.....	48
Şekil 6.4: 1300 grubu (yerden cips besleme) akış diyagramı	49
Şekil 6.5: 1600 grubu (yerden toz besleme) akış diyagramı.....	51
Şekil 6.6: Net-Pro sayfasında 1300 sistemi (yerden besleme) haberleşme yapısının oluşturulması.	52
Şekil 6.7: Step-7 Hardware Config sayfasında 1300 sistemi (yerden besleme) haberleşme yapısının oluşturulması.	52
Şekil 6.8: Step-7 Sembol Tablosunda 1300 sistemi (yerden besleme) adresleri sembol tablosu.	53
Şekil 6.9: 1300 Grubu ve 1600 Grubunun OB1 yönetici bloğunda çağırılması	57
Şekil 6.10: 1300 grubu (yerden cips besleme) kontrol sayfası	58
Şekil 6.11: 1600 grubu (yerden toz besleme) kontrol sayfası.....	60
Şekil 6.12: Sistem durum sayfası	62
Şekil 6.13: Akım grafikleri sayfası	63
Şekil 6.14: Akım tabloları sayfası.....	64

TABLolar DİZİNİ

Tablo 2.1: Bellek yapılarının özellikleri	7
Tablo 3.1: Çevrimsel program işlevi.....	17
Tablo 3.2: Organizasyon blokları kesme öncelikleri	18

KISALTMALAR DİZİNİ

Config	: Configuration
CP	: Haberleşme Modülü
CPU	: Merkezi İşlem Birimi
DB	: Veri Bloğu
DOL	: Direct On Line (Direkt Yol Verme)
DP	: Distributed Processing
HW	: Hardware
FB	: Fonksiyon Bloğu
FC	: Fonksiyon
FC	: Frequency converter
IHM	: Interface Human Machine
IM	: Interface Modul
MPI	: Multi Point Interface
OB	: Organizasyon Bloğu
PC	: Bilgisayar
PLC	: Programlanabilir Mantık Denetleyici
Profibus	: Process Field Bus
PS	: Power Supply
RTU	: Uzak Terminal Birimi
S7	: Step-7
SCADA	: Gözetleyici Denetim ve Veri Toplama
SD	: Star Delta
TCP/IP	: Transmission Control Protocol and Internet Protocol
UDT	: Kullanıcı Tanımlı Veri Bloğu
WinCC	: Windows Control Center
2D	: 2 Directions

SİMGELER DİZİNİ

I	: Input
M	: Merker
Q	: Output
SM	: Special Memory
V	: Variable Memory Area

PLC ve SCADA SİSTEMLERİNİN BİR MDF PRESİ DIŐ SAHA BESLEME HATTINA UYGULANMASI

Murat KILIÇ

Anahtar Kelimeler: Simatic Step-7, PLC, WinCC SCADA, Otomasyon, MDF

Özet: Günümüzde bütün sektörlerde üretim kalitesini ve hızını arttırarak rekabet etmek zorunlu hale gelmiştir. Bunların dışında iş ve işçi güvenliğini arttırarak iş kazalarını azaltmak da çok önemlidir. Üretimin her safhasında insan eli değmeden üretim yapmak otomasyonu önemli kılmıştır. Otomasyon denince kontrol ve takip akla gelmektedir. Üretimin kontrol kısmını PLC'ler, takip kısmını ise ara yüz programları olan SCADA programları yapmaktadır.

Bu çalışmada, bir MDF presi dış saha besleme hattının otomasyonu açıklanmaktadır. MDF; Termomekanik olarak odun veya diğer lignoselüozik hammaddelerden elde edilen liflerin, sentetik yapıştırıcı ilavesiyle belirli bir rutubet derecesine kadar kurutulduktan sonra oluşturulan levha taslağının sıcaklık ve basınç altında preslenmesiyle elde edilen bir üründür. MDF üretiminde odun parçacıklarının (cipslerin) prese girmeden önce dış sahada boyutlandırılması ve içinde bulunan yabancı cisimlerden ayrıştırılması gerekmektedir. Bu işlemlerin yapılması için ise otomasyon sistemlerini kullanmak gerekmektedir.

MDF presi dış saha besleme hattı tam otomasyon ile çalışmaktadır. Otomasyon sisteminde PLC yazılımı olarak Siemens Simatic Manager Step-7 V5.4 kullanılmıştır. PLC, CPU 416-2DP, remote processor ise ET200M' dir. SCADA yazılımı olarak Siemens Simatic Wincc V7.0 kullanılmıştır. Toplam giriş/çıkış sayısı 1184 olan sistemde SCADA ile PLC arası Ethernet, CPU 416-2DP ile ET200M arasında PROFIBUS (Process Field Bus) haberleşme protokolleri kullanılmıştır.

APPLICATION OF PLC and SCADA SYSTEMS TO THE OUTFIELD FEEDER LINE OF A MDF PRESS

Murat KILIÇ

Keywords: Simatic Step-7, PLC, WinCC SCADA, Automation, MDF

Abstract: Nowadays, competing by increasing production quality and speed has become a necessity in all the sectors. Apart from these, it is also important to decrease work accidents by increasing work and worker safety. Production without human touch in all stages of production has made automation important. Automation brings to mind control and monitoring. The control of production is made through PLC's, whereas the monitoring is undertaken by the SCADA programs which are interface programs.

In this study, the automation of an outfield feeder line of an MDF press is explained. MDF; is a product which is obtained by pressing under heat and pressure the board which is formed after drying up to a certain degree of humidity the fibers which are obtained from wood or other lignocellulosic materials with the addition of synthetic glue. In the production of MDF, it is necessary to size and to separate wood chocks (the chips) at the outfield from impurities they contain before going into the press. To conduct these procedures it is necessary to apply automation systems.

The outfield feeding line of the MDF press operates in full automation. In the automation system, Siemens Simatic Manager Step-7 V5.4 is being used as the PLC software. The PLC is CPU 416-2DP and the remote processor is ET200M. Siemens Simatic Wincc V7.0 has been used as the SCADA software. In the system where the total number of input/output is 1184 Ethernet has been used between SCADA and PLC, and PROFIBUS (Process Field Bus) has been used between CPU 416-2DP and ET200M as the communication protocol.

1. GİRİŞ

İçinde bulunduğumuz bilgi ve teknoloji çağının ürünü olan ve artık birçok sektörde vazgeçilmez unsur haline gelen PLC ve SCADA sistemleri, güvenilirliğini ve fonksiyonelliğini ispatlamış, günümüzde çok daha yaygın hale gelmiş ve teknolojik açıdan büyük mesafeler kat etmiştir.

Bu çalışmada bir MDF presi dış saha besleme hattının PLC ve SCADA yazılımları kullanılarak otomasyonu tasarlanmıştır. Bu sistem MDF üretimi yapan bir endüstriyel tesise uygulanmıştır.

Tez çalışmasında, ikinci bölümde PLC sistemlerinin genel yapısı anlatılmaktadır. Üçüncü bölümde Simatic Step-7 programının genel yapısı ve programlamada kullanılan blok yapıları, dördüncü bölümde SCADA sistemleri, beşinci bölümde WinCC SCADA programı incelenmektedir. Altıncı bölümde ise bir MDF presi dış saha besleme hattının Simatic Step-7 programı ile PLC programlanması, WinCC ile SCADA programının yapılması anlatılmaktadır.

PLC ve SCADA sistemleri kullanılarak sistemde kullanılan motorların, selenoid ventillerin manüel ve otomatik kontrolü yapılmaktadır. Bir bütün olarak sistem SCADA sayesinde bilgisayar ekranından izlenebilmektedir. Sistemdeki motorların akımları, akış hattı basınçları, hidrolik yağ basınçları, hidrolik yağ tankı sıcaklıkları, motor rulman sıcaklıkları ve silo seviyeleri sürekli olarak gözlemlenmektedir. Sistemde oluşabilecek arıza durumuna göre sistem ya otomatik olarak duruşa geçmekte ya da operatörü ikaz lambaları ile uyarmaktadır. Sistemde oluşan arızalar ve uyarılar alarmlar sayfasında tarih ve saat bilgileri ile kaydedilmektedir. Meydana gelen bir arızanın hangi tarihte ve saatte oluştuğu gözlenebilmektedir. SCADA ekranından arızaların sürekli olarak izlenmesi sayesinde oluşabilecek arızalara ani müdahale edilebilmektedir. SCADA sistemi, önceden tanımlanmış veya kullanıcının tanımladığı fiziksel değerlerin geleceğe yönelik süreç kontrolünü yapabilmektedir.

2. PROGRAMLANABİLİR MANTIK DENETLEYİCİLER (PLC)

2.1. PLC'nin Yapısı ve Çalışma Prensibi

PLC, İngilizcede Programmable Logic Controller (Programlanabilir Mantık Denetleyici) teriminin baş harflerinden oluşan bir kısaltmadır. Programlanabilen kumanda ve kontrol elemanıdır.

Bir programlama yapılırken zamanlama, sayma, taşıma, mantık işlemleri, bellek fonksiyonlarına ihtiyaç vardır. Bu ve bunun gibi birçok fonksiyon PLC'lerin içerisinde kendi hafızasına bağlı olarak yer almaktadır.

PLC, diğer sayısal veri işleme makinaları gibi Merkezi İşlem Birimi (CPU), Bellek Birimi ve Giriş-Çıkış Birimlerinden oluşur. Ayrıca programı yedeklemek ya da başka bir PLC'ye aktarmak için Ayrık Genişleme Birimi, Analog Giriş Çıkış Birimi ve enerji kesilmelerinde PLC'yi besleyen Yedek Güç Kaynağı gibi birimler de bulunur.

Merkezi İşlem Birimi (CPU): PLC'nin çalışmasını düzenleyen, lojik işlemleri gerçekleyen, zamanlama, sayma, karşılaştırma, matematiksel işlemler, regülatör kontrolü, mandallama, özel tanı işlemleri, kayan sayıcıların işlenmesi, kod çevirme gibi işlevleri sağlayan en önemli birimdir.

Bellek Birimi: Giriş görüntü belleği, çıkış görüntü belleği, veri belleği, program belleği gibi kısımlara ayrılmış olup bu bellek alanları farklı işlevler için kullanılır.

Giriş Birimi: Kontrol edilen sistemle ilgili, algılama ve kumanda elemanlarından gelen elektriksel işaretleri PLC'de işlenecek lojik gerilim seviyesine dönüştüren

birimdir. Kontrol edilen sisteme ilişkin basınç, seviye, sıcaklık algılayıcıları, kumanda butonları, sınır ve yakınlık anahtarları gibi elemanlardan gelen iki değerli işaretler (1=Var, 0=Yok) giriş birimi üzerinden alınır.

Çıkış Birimi: PLC’de hesaplanan çıkış noktalarına ilişkin lojik gerilim seviyelerini, kontrol edilen sistemdeki kontaktör, röle selenoid gibi kumanda elemanlarını sürmeye uygun elektriksel işaretlere dönüştüren birimdir. Çıkış birimi, triyak, röle veya transistörlü devrelerden oluşabilir.

Giriş/Çıkış Birimleri PLC’nin beyni olan CPU için dış dünyadan veriler toplarlar ve bu veriler lojik olarak değerlendirildikten ve işlendikten sonra tekrar dış dünyaya geri gönderilirler. Bazı veriler ise eğer PLC, OP (operatör panel) veya SCADA PC ile haberleşiyorsa ekranda mimik diyagramlarında (sembollerin renk, şekil değiştirmesi veya hareketlenmesi) ile kullanıcıların anlayabileceği hale getirilir.

Giriş/Çıkış Birimlerinde her giriş ve çıkışın özel bir adresi vardır. Bu adres haritaları CPU’da tanımlanır.

Ayrı Giriş/Çıkış Birimleri de en çok kullanılan modüllerdir. Çünkü PLC’nin üzerindeki giriş ve çıkışlar yapılan otomasyon programı için çoğu zaman yeterli olmamaktadır. Bu yüzden genişleme modülleri kullanmak zorunlu hale gelir. Giriş/Çıkış modülleri CPU ile aynı raya (rack) yerleştirilir ve montajları kolaydır. CPU ile aralarında özel bir kablo ile iletişim kurulur. Kendi içindeki bu kablo aracılığıyla 5V DC gerilim ile modülün beslemesi yapılır. S7-200 PLC’ler de bu şekildedir. S7-300 PLC’lerde bus konnektörleri kullanılır ve S7-400 PLC’ler de ise işlemci ve giriş, çıkış, haberleşme modülleri aynı rayda olup bu ray elektronik karttır ve bu şekilde beslenmiş olmaktadır.

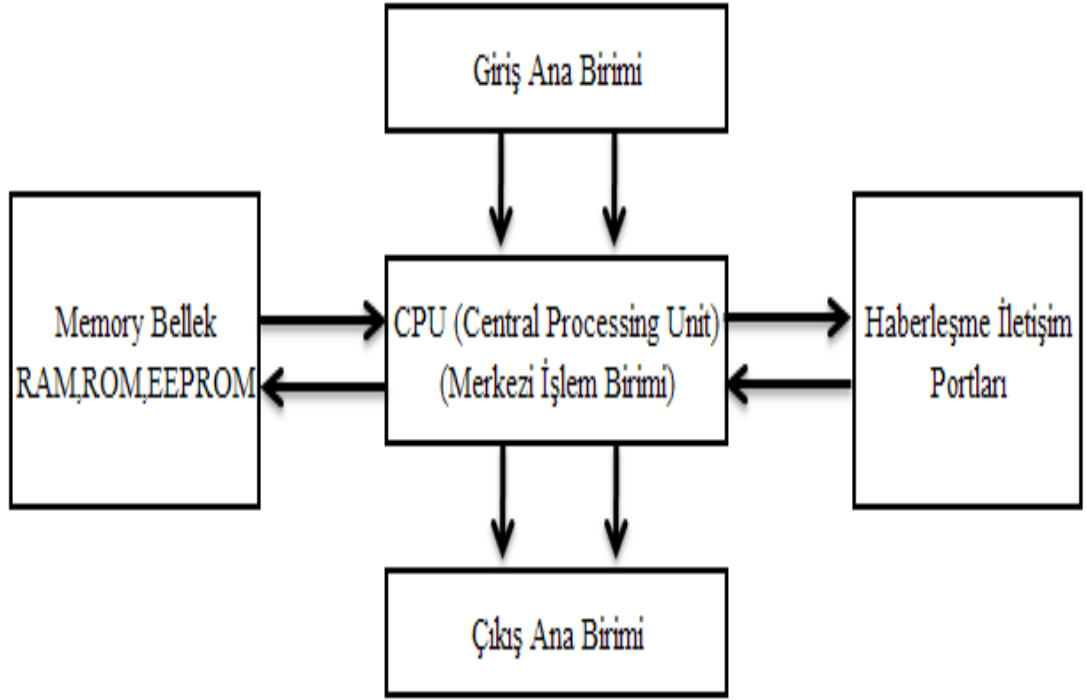
Programlayıcı Birimi: Kumanda devresine ilişkin programın yazılması, PLC'ye aktarılması ya da PLC'deki programın geri alınması ve yeniden düzenlenmesi gibi amaçlar için kullanılan bir el programlayıcısı ya da kişisel bilgisayarda çalışan bir programlama yazılımı olabilir.

İşletim Sistemi: Kalıcı ve yalnız okunabilir bir bellek alanına üretici firma tarafından yazılmış olan işletim sistemi programı, PLC'nin çalışmasını düzenler ve kullanıcı programının yürütülmesini sağlar. Genel olarak bir işletim sistemi aşağıdaki işlevleri yerine getirir.

1. Giriş noktalarındaki işaret durumlarının giriş görüntü belleğine yazılması
2. Programın yürütülmesi
3. Haberleşme isteklerinin işlenmesi
4. İşlemci, çevre birimleri ve bellek durumlarının incelenmesi
5. Çıkış görüntü belleğindeki değerlerin çıkış birimine aktarılması

Kontrolör, durma (stop) konumuna alınana kadar bu işlemler sürekli tekrarlanır. Bu işlem sırasına ise bir tarama çevrimi (SCAN TIME) denir. Bir tarama çevrimi, çıkış görüntü belleğinin çıkış birimine aktarılması ile son bulur. Aktarma işlemi tamamlandıktan sonra yeni tarama çevrimi başlar [1].

Görüntü belleğinin faydası; programların bellekten daha hızlı çalışması, belleğe erişimin giriş ve çıkışa erişimden daha kolay olmasıdır. Görüntü belleğinin sakıncası ise girişte kısa süreli değişimlerin (işlemlerin) okunamamasıdır. Şekil 2.1'de PLC'nin genel yapısı görülmektedir.



Şekil 2.1: PLC'nin genel yapısı [1]

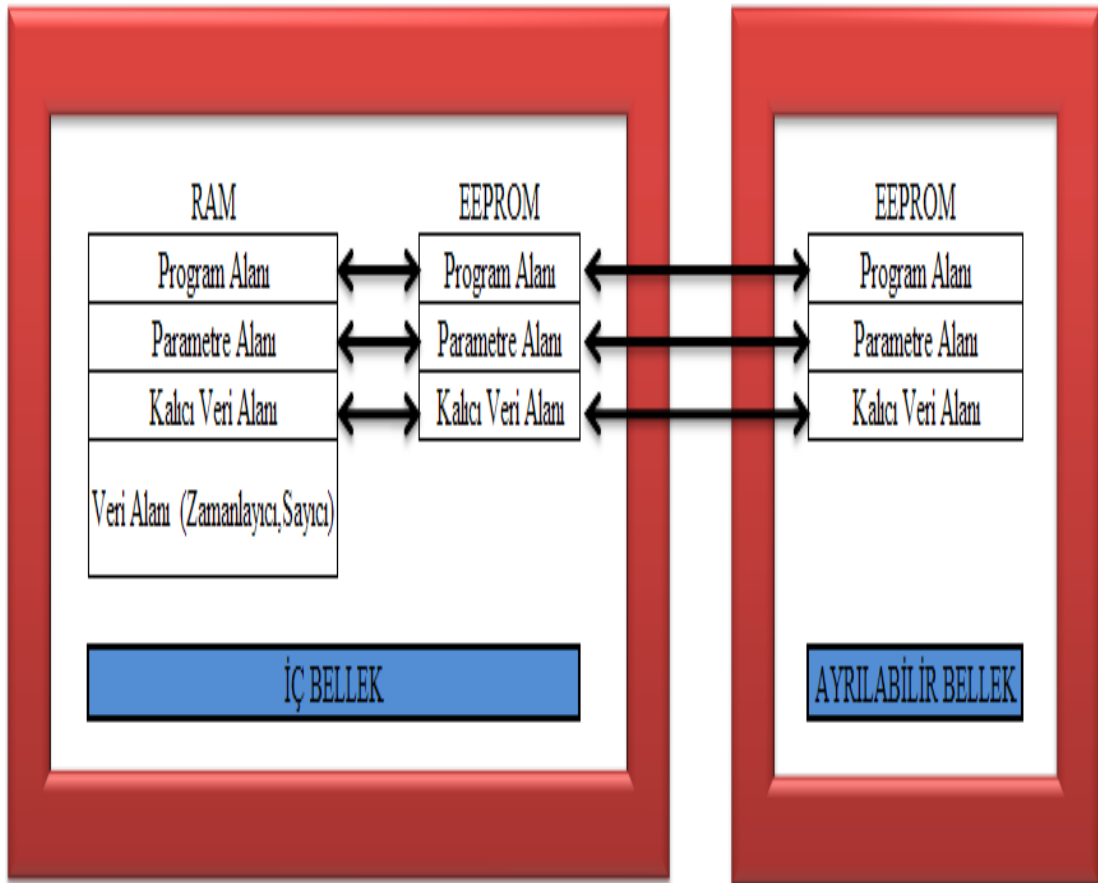
2.1.1. Bellek yapısı

PLC'lerde üç türlü bellek alanı bulunmaktadır.

- Program Alanı (Work Memory): Programlamanın yapıldığı, lojik olgu ve emirlerin saklandığı yerdir. Bir program için gerekli bellek alanı, her lojik işlem komutu için 1 Word (Kelime), zamanlayıcı ve sayıcı komutları için 3 Word'lük alan kullanılır. Program aynı zamanda yazılabilen ve okunabilir olan kalıcı bellekte de saklanır. Bunun sonucu olarak elektrik kesilmelerinde program silinmez.
- Veri Alanı (Load Memory): Denetleyici program belleğine yüklenen kullanıcı programının yürütülmesi sonucunda hesaplanan sonuçlar ara değerler, sabit değerler giriş, çıkış sayma, zamanlama ve analog giriş, çıkış verilerinin saklandığı yerdir. Veri alanı içine program kontrolünde girilebilir. Veri alanının bir kısmı kalıcı bellekte

saklandığı için enerji kesilmelerinde sabitler ve diğer bilgiler kaybolmaz. Veri alanının diğer kısmı Sıradan Erişimli Bellekte (RAM) saklandığı için PLC'nin pili tükenene kadar saklanabilir daha sonra silinir. Program PLC'ye yüklendiğinde önce Veri Belleğine (Load Memory) kaydedilir, program çalıştırıldığında Program Belleğine (Work Memory) aktarılır.

- Parametre Alanı: Kontrolöre ilişkin verilerin, şifre, istasyon adresi, kalıcı bilgiler gibi yapısal parametre seçimlerinin varsayılan ya da değiştirilmiş değerlerinin saklandığı yerdir. Parametre belleğinin içeriği, program belleği ile aynı karakteristiğe sahiptir ve kalıcı bellekte saklanır. Şekil 2.2'de PLC'nin bellek yapısı görülmektedir.



Şekil 2.2: PLC'nin bellek yapısı [1]

Tablo 2.1: Bellek yapılarının özellikleri [2]

SEMBOL	BELLEK TİPİ	SİLME	PROGRAMLAMA	BELLEK DURUMU
RAM	Random Access Memory (Okunabilir-Yazılabilir Bellek)	Elektriksel	Elektriksel	Gerilim Yoksa Geçici
ROM	Read Only Memory (Sadece Okunabilir Bellek)	Mümkün Değil	Üreticiden Alınan Maske İle	Gerilim Yoksa Kalıcı
PROM	Programable ROM (Programlanabilir Bellek)	Mümkün Değil	Elektriksel	Gerilim Yoksa Kalıcı
EPROM	Erasable ROM (Silinebilir Bellek)	Ultraviyole Işınları	Elektriksel	Gerilim Yoksa Kalıcı
EEPROM	Electrical Erasable ROM (Elektrikle Silinebilir Bellek)	Elektriksel	Elektriksel	Gerilim Yoksa Kalıcı

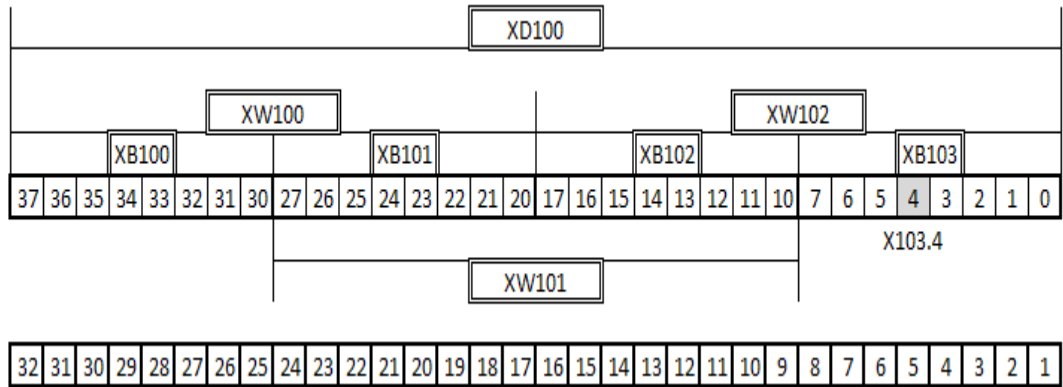
Programlar genellikle RAM Bellekte saklanır. Çünkü RAM Bellek programın kolay oluşturulmasına, değiştirilmesine ve veri girişine olanak sağlar. Diğer hafıza türlerine göre daha hızlıdır. RAM Bellekteki veriler enerji kesilmelerinde CPU'ya takılan pilin ömrü kadar bilgiyi saklayabilir. Bazı CPU'larda pilin yerine kondansatör kullanılmıştır ve değiştirilmeye gerek yoktur. ROM Bellek içinde bulunan bilgi ve komutlar sadece okunabilir. Besleme gerilimi kesilse bile ROM içindeki program korunur. PLC'deki programların yedeklenmesi veya taşınması için hafıza kartları kullanılır.

2.1.2. Veri belleği ve adresleme

Veri belleği alanına Bit, Byte, Word (Kelime) ve DWord (çift kelime) olarak doğrudan adresleme ile erişilebilir.

- Bit: Var-yok (1-0) bilgileri 1 Bit'tir. I0.0, Q2.5, M12.1, DB1.DBX0.0 gibi.
- Bayt: 8 Bitten 1 Bayt oluşur. (7.6.5.4.3.2.1.0) IB25, QB10, DB2.DBB2 gibi.
- Word: 2 Bayttan veya 16 Bitten 1 Word oluşur. IW16,QW25, DB3.DBW0 gibi.
- Doubleword: 2 Wordden veya 4 Bayttan veya 32 Bitten 1Dword oluşur. ID2,QD100,DB3.DBD4 gibi

Veri alanındaki 1 Bit'lik bellek konumunu adreslemek için bellek alanı (I,Q,M,V,SM) ve konumu (Byte.Bit) belirtilir. Örneğin; Ix.y adresi, giriş görüntü belleğinin x. Baytının y. konumunu , Qx.y adresi, çıkış görüntü belleğinin x. Baytının y. bitini tanımlamaktadır. Şekil 2.3'de PLC'nin adres yapısı görülmektedir.



Şekil 2.3: PLC'de adres yapısı

Adreslemede sorunla karşılaşılmaması için Baytlar 1'er arttırılarak, Wordler 2'ser arttırılarak Dwordler ise 4'er arttırılarak adreslenir.

Siemens PLC'lerde bayt numaraları soldan sađa dođru, bit numaraları ise tam tersine sađdan sola dođru artar.

2.2. Sembolik Adresleme

PLC'lerde genel olarak 2 tr adresleme Őekli bulunmaktadır.

1. Global sembol: PLC'nin kendi verdiđi I0.0, Q3.7 gibi adreslerin projenin tamamında ne iin kullanıldıđını hatırlamak eđer program uzun ise zor olmaktadır. Bu yzden bu mutlak (gerek) adreslere sembol tablosunda global semboller atanır. Bu semboln uzunluđu en fazla 24 karakter olup Trke karakterler ve rakamla bařlayan isimler kullanılamazlar. Bir projede en fazla 1600 sembol kullanılabilir. PLC'ye program yklendiđinde sembol tablosu yklenmez, yani CPU'nun ierisinde iřlemler sembollerle deđil kendi adresleri ile iřlenir. CPU ierisinde alıřan program programlamanın yapıldıđı bilgisayarına indirildiđinde semboller bulunmamaktadır. Eđer program bařka bir bilgisayarına tařınacak veya kopyalanacak olursa mutlaka sembollerin tařınması gerekmektedir.

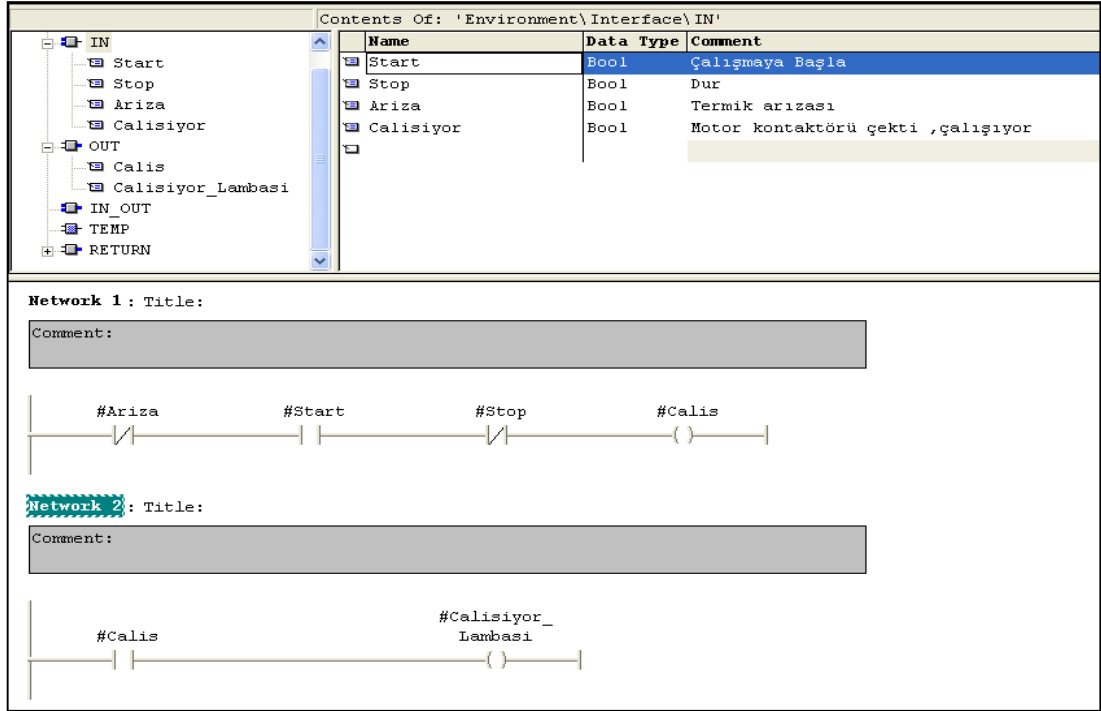
Sembol tablosu dıřarıda hazırlanarak SIMATIC programına alınabilir. rnek olarak Excel'de adresler, bunlara karřılık gelen semboller ve veri tiplerinin bulunduđu bir tablo oluřturulur. Daha sonra SIMATIC programının Import komutu ile semboller ve adresler programa dhil edilir. Bu iřlemin tersini de yapmak mmkndr. Bařka bir projenin sembol tablosunu hazırlayıp onu da Export komutunu kullanarak Excel'e gndermek mmkndr.

Program yazılırken sembollerin kullanılmasının bir diđer avantajı ise programlama blokları eđer sembol bazlı seilirse programda kullanılan input (giriř) adresi deđiřtirildiđinde kullanılan sembol sabit kaldıđından program blođundaki adresi deđiřtirilmesine gerek kalmamaktadır. Őekil 2.4'de sembol tablosu grlmektedir.

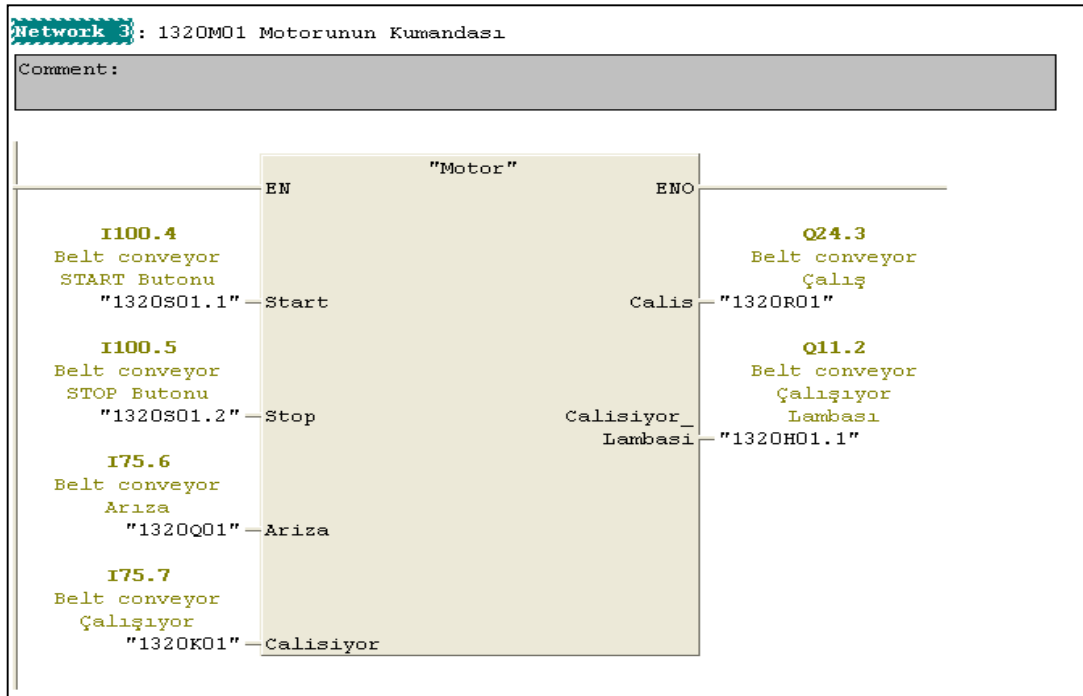
Status	Symbol	Address	Data type	Comment
641	1350S01.1	I 34.3	BOOL	Belt conveyor START Butonu
642	1350S01.2	I 34.4	BOOL	Belt conveyor STOP Butonu
643	1350S02.1	I 34.1	BOOL	Cleaning brush START Butonu
644	1350S02.2	I 34.2	BOOL	Cleaning brush STOP Butonu
645	1350S105.1	I 40.4	BOOL	1350 Belt Conveyor / Belt tracking left OK
646	1350S105.2	I 40.5	BOOL	1350 Belt Conveyor / Belt tracking left Warning
647	1350S106.1	I 40.6	BOOL	1350 Belt Conveyor / Belt tracking right OK
648	1350S106.2	I 40.7	BOOL	1350 Belt Conveyor / Belt tracking right Warning
649	1350S107.1	I 34.7	BOOL	1350 Belt Conveyor / Belt tracking left OK
650	1350S107.2	I 35.0	BOOL	1350 Belt Conveyor / Belt tracking left Warning
651	1350S108.1	I 35.1	BOOL	1350 Belt Conveyor / Belt tracking right OK
652	1350S108.2	I 35.2	BOOL	1350 Belt Conveyor / Belt tracking right Warning
653	1355H01.1	Q 8.7	BOOL	Belt conveyor Çalışıyor Lambası
654	1355K01	I 2.0	BOOL	Belt conveyor Çalışıyor
655	1355Q01	I 1.7	BOOL	Belt conveyor Arıza
656	1355R01	Q 1.0	BOOL	Belt conveyor Çalış
657	1355S01.1	I 35.3	BOOL	Belt conveyor START Butonu
658	1355S01.2	I 35.4	BOOL	Belt conveyor STOP Butonu
659	1359B0167	I 36.1	BOOL	1359 Diverter Flap / End position to 1551
660	1359B0168	I 36.2	BOOL	1359 Diverter Flap / End position to 1342
661	1359H01.1	Q 9.2	BOOL	Diverter flap İLERİ Çalışıyor Lambası
662	1359H01.2	Q 9.3	BOOL	Diverter flap GERİ Çalışıyor Lambası
663	1359H0121	I 36.3	BOOL	1359 Diverter Flap / Bakım Emniyet Şalteri-Hazır
664	1359K01.1	I 2.3	BOOL	Diverter flap İLERİ Çalışıyor
665	1359K01.2	I 2.4	BOOL	Diverter flap GERİ Çalışıyor
666	1359M01_1342M01_SECILI	M 20.4	BOOL	1359M01 DIVERTER FLAP 1342M01 BANDINI DOLDUR
667	1359M01 1551 SILO SECILI	M 20.5	BOOL	1359M01 DIVERTER FLAP 1551 SILOSUNU DOLDUR

Şekil 2.4: Sembol tablosu

2. Lokal sembol: Program modüllerinde deklârasyon tablosunda değişkenler tanımlanarak sadece o blok için geçerli olacak semboller yazılır ve bu sembollerde (#) işareti sembolün başında kullanılmaktadır. Örneğin bir motor bloğu yazılmak istendiğinde bildirge tablosunda bu motora ait start, stop, arıza=>girişler, çalış, çalışıyor =>çıkışlar bilgileri tanımlanır ve bu motor bloğunun çalışma mantığı yazılır. Bloğun dışında ise sadece bu blok için kullanılacak giriş ve çıkışlara adresler atanır. Şekil 2.5’de lokal sembol kullanımı blok iç yapısı görülmektedir. Şekil 2.6’da ise lokal sembol kullanımı blok dış yapısı görülmektedir.



Şekil 2.5: Lokal sembol kullanımı blok iç yapısı



Şekil 2.6: Lokal sembol kullanımı blok dış yapısı

Oluşturulan motor bloğu farklı giriş ve çıkışlar için yani farklı bir motor kumandası için programda birden fazla kez kullanılabilir.

2.3. PLC Programı Yazım Dilleri

PLC'lerde genel olarak 3 dil ile yazılım yapılmaktadır.

1. Ladder (Merdiven Diyagramı): Elektrik kumanda tekniğine uygun
2. FBD (Function Blok Diyagram): Elektronik kumanda tekniğine uygun
3. STL (Deyim Listesi: Statement List): Bilgisayar kumanda tekniğine uygun

Bunlardan başka Graph ve SCL dilleri de kullanılmaktadır. Ama en yaygın olarak kullanılan yukarıda verilen dillerdir.

2.4. PLC Programlama Yöntemleri

1. Linear Program: Bu programlama yönteminde komutlar sıraya göre çağrılır ve işlem yapılır. Bütün komutlar bir kez işlendikten sonra işlem tekrar başa döner ve tarama PLC, STOP konumuna alınana kadar sürekli devam eder. Bu işleme tarama çevrimi (SCAN TIME) denir. Bu programlama yöntemi genellikle basit ve az komut içeren projelerde kullanılır.

Çevrim:

1.komut
2.komut
.
.
.n.komut



2. Yapısal Program: Bu programlama yönteminde program küçük programlara ayrılır. Bu programlar da FC (Fonksiyon) veya FB (Fonksiyon Blok) içerisine yazılır. Burada işlenen veriler de DB (Data Blok: Veri Bloğu) kaydedilir. Buradan istenirse

OB1, CPU'da taranan tek program ve ana programdır. Diğer bloklar OB1 tarafından çağrılarak işletilirler. Her tarama çevriminde CPU, OB1 içerisine yazılan blokları o bloklarda kendisinin ilişkili olduğu blokları çağırarak tarama çevrimi tamamlanır. S7-300 CPU'larda iç içe 8 adet, S7-400 CPU'larda iç içe 16 adet program çağrılabilir. Ayrıca S7-300 PLC'lerde OB1 içerisindeki program işletilirken, bir OB (Organizasyon Blok) başka bir OB tarafından kesilerek, kesen OB'nin içerisindeki program işletilir. Kesme önceliği OB numarasına bağlıdır. Numarası büyük olan blok daha önceliklidir. En az önceliği bulunan blok OB90'dır. OB90 dışındaki bütün OB blokları, OB1 bloğundan daha önceliklidir. Bu OB arka program işlevi için kullanılmaktadır. Örnek verecek olursak OB20 bloğu OB10 bloğundan kesme işlemi için daha önceliklidir.

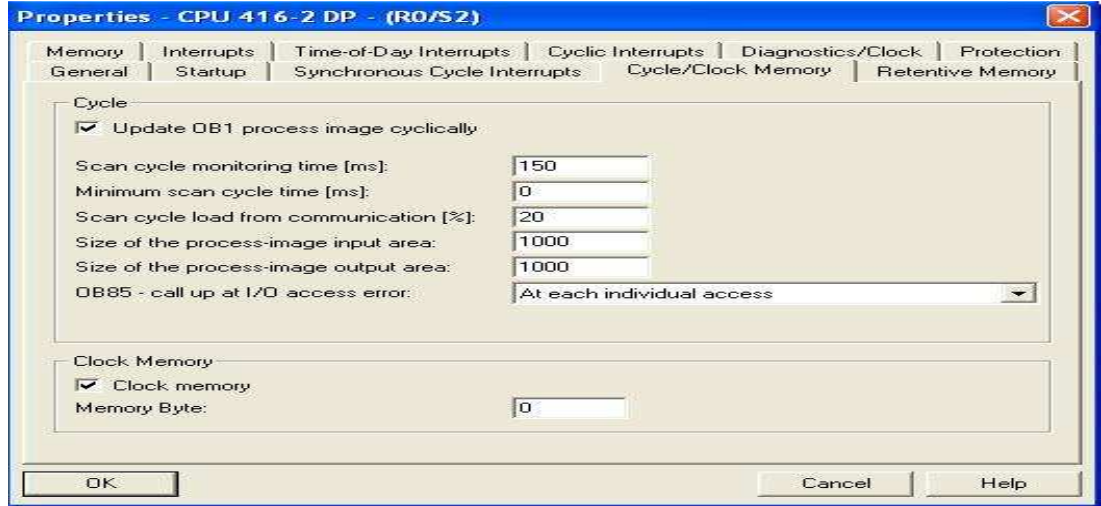
3.1. Program Blokları

3.1.1. Organizasyon blokları (OBn)

Organizasyon Blokları, program bloklarının hangi sırayla taranacağını belirler. OB1 ana modüldür. Diğer bütün program ve veri blokların hangi sırayla işleneceğini OB1 içerisinde programların çağrılmasıyla karar verilir. Diğer modüllerin hiç birini kullanmadan OB1'in içerisinde bütün program yazılabilir (Lineer Programlama). Fakat bu şekilde yazılan programın esnekliği olmadığı gibi, program işletilmeye başlandığında arıza takibi de zor olmaktadır. Programın fonksiyonel olması programın anlamlı parça programlara, yani alt programlara ayrılması ile gerçekleştirilebilir (Yapısal Programlama).

PLC yüklenen programı taramaya başladığında OB1 sürekli olarak her çevrimde taranır. Diğer bloklar ise bu modülün içerisinde kendi çağırılma sırasına göre işletilirler. CPU maksimum tarama süresini kontrol eder. Maksimum tarama süresi 150ms olarak ayarlanmıştır. İstenirse bu süre değiştirilebilir. Tarama zamanı izleyicisi programın herhangi bir yerinde resetlenebilir (SFC43''RE_TRIGER'').

OB'nin tarama süresi izin verilen maksimum süreyi geçmesi halinde işletim sistemi OB80 bloğunu çalıştırır. Eğer OB80 programlanmamışsa CPU durma konumuna geçer. Bazı CPU modellerinde minimum tarama (çevrim) süresi de ayarlanabilir. Şekil 3.2'de CPU özellikleri görülmektedir.



Şekil 3.2: Simatic S7 programının blok yapısı

OB1, OB90 ve OB100 dışındaki diğer OB'ler kesme (interrupt) programı içerirler. Üst öncelikli OB alt öncelikli OB'yi keser (alarm) ve kendi programlarını çalıştırır. Kesme OB'leri zamana veya olaya bağlı olarak çağrılırlar.

OB90 içeren CPU'larda tarama süresi maksimum tarama süresinden daha kısa ise OB90 çağrılır ve onun içerisindeki program işletilir. Eğer OB90 programlanmamışsa CPU, tarama süresi dolana kadar program işletmeyi bekletir. Süre dolduktan sonra çevrim (tarama, döngü) tekrar başlar. OB90 olmayan modellerde ise maksimum çevrim süresine bakılmadan her saykıl tarandıktan sonra yeni bir saykıla geçilir.

Start OB'si (OB100) çevrim başında bir kez (CPU durma konumundan çalışma konumuna geçerken) işlenir, daha sonra bu OB tekrar taranmaz. Enerjinin kesilip tekrar gelmesi durumunda OB100 işlenmez. OB100 içerisinde kullanılan hafıza alanları (merkerler) programın diğer modüllerinde (bloklarında) tekrar

kullanılabilirler. OB100 deki program süresi çevrim süresine dâhil değildir. Çok uzun süreli programlar çalıştırılabilir.

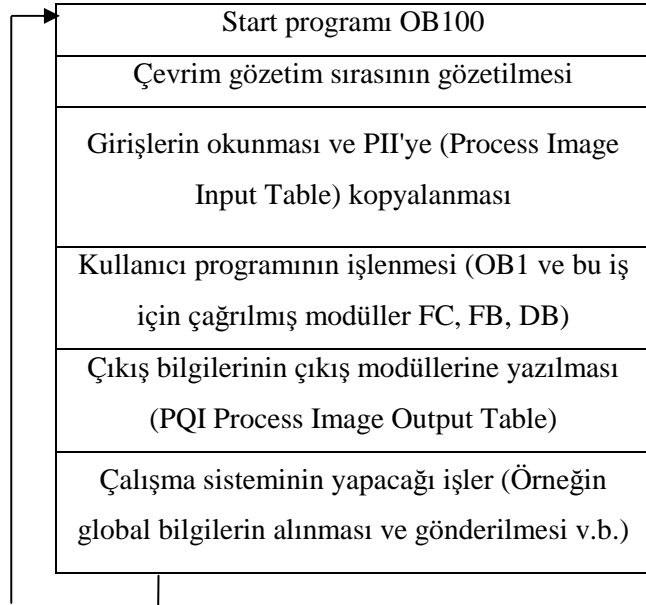
OB100 özelliikle;

- PLC ilk çalışmaya başladığında kalıcı merker, sayıcı, zamanlayıcı gibi elemanların sıfırlanmasında,
- Sistem çalışmaya başlamadan önce bazı şartların yerine getirilmesi gerektiğinde,
- Sistem çalışmaya başlamadan önce başlangıç şartlarına gelmesi istendiğinde kullanılabilir.

Tablo 3.1’de çevrimsel program işlevi, Tablo 3.2’de ise organizasyon bloklarının kesme öncelikleri yer almaktadır.

Çevrimsel program işlevi:

Tablo 3.1: Çevrimsel program işlevi [2]



Tablo 3.2: Organizasyon blokları kesme öncelikleri [2]

Adı	Açıklama	Öncelik
OB1	Serbest Çevrim	1
OB10-OB17	Saat Alarmı	2
OB20-OB23	Gecikme Alarmı	3-6
OB30-OB38	Uyandırma Alarmı	7-15
OB40-OB47	Süreç Alarmı	16-23
OB80-OB87	Asenkron Hata	26
OB90	Arka Plan Çevrimi	0.29*
OB100	Başlatma (Start)	27

3.1.1.1. Günün belli saatlerinde oluşan kesme organizasyon bloğu (OB10...OB17)

Blok parametrelerini SFC28 ve SFC30 (Sistem Fonksiyon Blokları) veya Step-7 yazılımı kullanılarak atanabilir. Bu bloklar;

- Bir defa
- Her dakika
- Saatlik
- Günlük
- Haftalık
- Aylık
- Her ayın sonunda çalıştırılabilirler.

Aylık olarak çalıştırılanlar ayın 1'inden 28'ine kadar olan günleri kullanır. Bloklar kullanılmadan önce parametreleri ayarlanmalı ve daha sonra aktif hale getirilmelidirler.

Bu blokların başlama koşulları aşağıdaki gibidir:

1. Blok otomatik olarak başlar, parametreler bir defa girilir ve blok bir defa aktif edilir.
2. Parametreler Step-7 programıyla girilir ve programda SFC30 "ACT_TINT" (Activate Time-of-Day Interrupt) çağrılarak blok aktif edilir.
3. SFC28 "SET_TINT" (Set Time-of-Day Interrupt) çağrılarak parametreler girilir ve SFC30 "ACT_TINT" (Activate Time-of-Day Interrupt) çağrılarak blok aktif edilir [2].

3.1.1.2. Belli bir zaman sonraki kesme organizasyon bloğu (OB20...OB23)

Bu bloklar belirli bir zaman sonra işletilirler. Bu blokların aktif hale getirilebilmesi için SFC32 "SRT_DINT" (Start Time-Delay Interrupt) çağrılarak kesme programı başlatılır. Bu sistem bloğu çağrıldığında OB numarası (OB_NR), bekleme zamanı (DTIME) ve kullanıcı belirli veriler (SIGN) girilir. Belirli bir zaman sonra OB çalışır. OB bloğu çalıştırılmadan önce iptal edilebilir. Bekleme zamanı dolduğunda (milisaniye) işletim sistemi belirtilen OB' yi çalıştırır.

- SFC32 "SRT_DINT" (Start Time-Delay Interrupt) çağrılmalıdır.
- OB işlemciye yüklenmelidir.

Bu blok CPU sadece RUN modunda iken çalıştırılır. Bu boğun verileri CPU'nun yarı otomatik ve elle başlatılması ile silinir. Bu blok aktif edilmeden önce SFC33 "CAN_DINT" (Cancel Time-Delay Interrupt) kullanılarak kesme iptal edilir. Bu boğun çözünürlüğü 1ms'dir. Zamanı dolan kesmenin zamanı yeniden başlatılabilir.

SFC34 “QRY_DINT” (Query Time-Delay Interrupt) kullanılarak kesme bekleme zaman durumu sorgulanabilir.

- SFC32 “SRT_DINT” (Start Time-Delay Interrupt) çağrıldığında bu bloğa yazılan OB (ilişkilendirilen) CPU’ya yüklenmeden işletim sistemi bu bloğu çalıştırdığında,
- Bu kesme OB’si bir önceki işlemini bitirmeden tekrar gerçekleşirse, İşletim sistemi bu iki durum sonucunda eşzamansız hata kesme OB’yi çağırır [2].

SFC39 “DIS_IRT” (Disable New Interrupts and Asynchronous Errors) ...SFC42 “EN_AIRT” (Enable Higher Priority Interrupts and Asynchronous Errors) sistem fonksiyonlarını kullanarak kesme OB’sinin erişimi engellenebilir, bekletilebilir veya yeniden erişimi aktif hale getirilebilir.

3.1.1.3. Çevrimsel kesme organizasyon bloğu(OB30...OB38)

Kesme çalıştırıldığında bir daha aynı kesme gelmeden önce çalıştırılması bitmiş olmalıdır. Önceki kesmenin işletilmesi bitmeden yeni bir kesme gelirse OB80 “CYCL_FLT” (Cycle Time Fault) çalıştırılır. Hataya sebep olan kesme daha sonra çalıştırılır.

SFC39 “DIS_IRT” (Disable New Interrupts and Asynchronous Errors) ...SFC42 “EN_AIRT” (Enable Higher Priority Interrupts and Asynchronous Errors) sistem fonksiyonlarını kullanarak kesme OB’sinin erişimi engellenebilir, bekletilebilir veya yeniden erişimi aktif hale getirilebilir.

Bloklarla ilgili parametreler SIMATIC S-7 programıyla değiştirilir.

3.1.1.4. Donanım kesmesi organizasyon bloğu (OB40...OB47)

Sekiz adet donanım kesmesi blokları OB40 “HW_INT0” (Hardware Interrupt 0) tarafından işletilmektedir. Modül tarafından donanım kesmesi tetiklendikten sonra işletim sistemi, slotu ve karşılık gelen OB’yi belirler. OB var olan donanım kesmesinden daha yüksek öncelikli ise çalıştırılır. Donanım kesmesi çalıştırıldıktan sonra belirtilen kanal alındı bilgisi gönderilir. Donanım kesmesi tanımlama ve alındı işlemi yapılırken aynı modülden donanım kesmesi tetiklemesi gelirse;

- Aynı kanaldan donanım kesmesi oluşursa işletilen kesme bitene kadar gelenler kaybolur.
- Aynı modülün farklı kanalından kesme gelirse hemen donanım kesmesi gerçekleşmez. İşlenmekte olan donanım kesmesi kabul edildikten sonra sıradaki donanım kesmesi aktif edilir. Kesmelerde kayıp gerçekleşmez.

SFC39 “DIS_IRT” (Disable New Interrupts and Asynchronous Errors) ...SFC42 “EN_AIRT” (Enable Higher Priority Interrupts and Asynchronous Errors) sistem fonksiyonlarını kullanarak kesme OB’sinin erişimi engellenebilir, bekletilebilir veya yeniden erişimi aktif hale getirilebilir.

Bloklarla ilgili parametreler SIMATIC S-7 programında SFC55 “WR_PARM” (Write Dynamic Parameters)... SFC57” PARM_MOD” (Assign Parameters to a Module) ile de atanabilir.

3.1.1.5. Zaman hatası organizasyon bloğu (OB80)

- Çevrim süresinin aşılmasında
- Çalışmakta olan OB’nin tekrar çağrılmasında
- Saat alarmı OB’lerinden kaynaklanan zaman hataları oluştuğunda, RUN moduna geçişte;

Yukarıda sözü edilen durumlardan herhangi biri oluştuğunda OB80 işletim sistemi tarafından çağrılır. Eğer programda OB80 bulunmuyorsa CPU STOP moduna geçer. SFC39 “DIS_IRT” (Disable New Interrupts and Asynchronous Errors)...SFC42 “EN_AIRT” (Enable Higher Priority Interrupts and Asynchronous Errors) sistem fonksiyonlarını kullanarak kesme OB’sinin erişimi engellenebilir, bekletilebilir veya yeniden erişimi aktif hale getirilebilir.

Eğer aynı çevrim süresi içinde zaman taşmasından dolayı OB80 iki kez çağrılmışsa CPU STOP konumuna geçer. Bu durum SFC43 “RE_TRIGR” (Retrigger Cycle Time Monitoring) ile engellenebilir.

3.1.1.6.Besleme gerilimi hatası organizasyon bloğu (OB81)

Aşağıdaki durumlardan herhangi biri oluştuğunda OB81 işletim sistemi tarafından çağrılır. Eğer programda OB81 programlanmamışsa, CPU besleme hatası uyarısıyla çalışmasını sürdürür.

- Ana cihaz veya genişletme cihazlarından birinin yedekleme pilinin boşalması
- Ana cihaz veya genişletme cihazlarından birinde olan besleme gerilimi ve 24V DC beslenme hatası;

3.1.1.7.Hata bulgusu uyarı organizasyon bloğu (OB82)

Hata teşhisi yapabilen bir CPU’da, teşhis alarm isteği konmuşsa, bir uyarı ile teşhis alarmı serbest bırakılır. Bunun üzerine işletim sistemi OB82’yi çağırır. OB82 teşhis verilerini 4 Bayt’lık bir lokal değişkene yazar. Eğer OB82 programlanmamışsa CPU kendiliğinden STOP moduna geçer.

SFC39 “DIS_IRT” (Disable New Interrupts and Asynchronous Errors) ...SFC42 “EN_AIRT” (Enable Higher Priority Interrupts and Asynchronous Errors) sistem

fonksiyonlarını kullanarak kesme OB'sinin erişimi engellenebilir, bekletilebilir veya yeniden erişimi aktif hale getirilebilir.

3.1.1.8. Takma/Çıkarma uyarısı organizasyon bloğu (OB83)

İşletim sistemi donanımı üzerindeki bütün kartlarını 1sn tarama süresiyle izler. Her modülün takılma ve çıkarılma işlemi (RUN, STOP veya ANLAUF konumunda) teşhis tablosuna ve sistem durum listesine eklenir. Buna ek olarak işletim sistemi, CPU RUN modunda iken OB83'ü "I/O_FLT2" (I/O Point Fault 2) çağırır. Programda OB83 oluşturulmamışsa işletim sistemi CPU'yu STOP moduna götürür.

3.1.1.9. CPU donanım hatası uyarısı organizasyon bloğu (OB84)

İşletim sistemi bir arabirim haberleşme hatası (MPI (Multi Point Interface) ağ, Profibus DP) oluştuğunda veya kaybolduğunda OB84' ü çağırır. Eğer programda OB84 programlanmamışsa donanım hatası işletim durumunu Stop konumuna götürür.

3.1.1.10. Program yürütme hatası organizasyon bloğu (OB85)

Aşağıdaki durumlardan herhangi biri oluştuğunda OB85 işletim sistemi tarafından çağrılır. Eğer OB85 oluşturulmamışsa program yürütme hatası, işletim sistemi tarafından CPU'yu STOP konumuna götürür.

- CPU' ya yüklenmeyen OB' nin işletim sistemi tarafından çağrılması
- İşletim sistemine ait veri erişim hatasında
- Tarama çevrimi işletilirken çevresel birimlerle veri iletişim hatasında (Genişleme modülleri, haberleşen motor sürücüler, haberleşen yumuşak yol vericiler (Soft Starter), haberleşen elektronik aşırı akım koruma röleleri.)

3.1.1.11. Modül taşıyıcı (montaj rayı) hatası organizasyon bloğu (OB86)

Eğer modül taşıyıcı arızası (gerilim hatası, hat kopması, hatalı IM), ağ sistemi veya merkezi olmayan çevresel birimlerde oluşan arıza sonucunda işletim sistemi OB86'yı çağırır. Eğer OB86 oluşturulmamışsa modül taşıyıcısı hatası işletim sistemi tarafından CPU'yu STOP konumuna götürür.

3.1.1.12. Haberleşme hatası organizasyon bloğu (OB87)

Aşağıdaki durumlar oluştuğunda işletim sistemi OB87'yi çağırır. Eğer OB87 programlanmamışsa haberleşme hatası CPU'yu STOP moduna götürür.

- Global veri haberleşmesinde yanlış veri tanımlaması veya veri uzunluğu
- Mümkün olmayan hata verilerinin gönderilmesi
- Saat senkronizasyonunda (uyumunda) hata

3.1.1.13. Programlama hatası organizasyon bloğu (OB121)

CPU'nun çalışması esnasında işletim sistemi tarafından bir program hatası oluştuğunda işletim sistemi OB121'i "PROG_ERR" (Programming Error) çağırır. Program içerisinde bir bloğun yazılması ve bu bloğun CPU'ya yüklenmemesi veya tanımlanmayan bir adresin girilmesi (PQW 365.0) gibi. Eğer OB121 programlanmamışsa programlama hatası CPU'yu STOP moduna götürür.

3.1.1.14. Veri erişim hatası organizasyon bloğu (OB 122)

CPU'nun çalışması esnasında tanınan bir veri erişim hatası oluştuğunda işletim sistemi OB122'yi (MODULE ACCESS ERROR) çağırır. Eğer OB122 programlanmamışsa veri erişim hatası CPU'yu STOP moduna götürür.

Programlama ve veri erişim hatası OB'leri, SFC36...SFC38' e kadara olan sistem bloklarıyla engellenebilir, geciktirilebilir ve tekrar serbest bırakılabilir [2].

3.1.2. Fonksiyon (FC) ve fonksiyon blokları (FB)

Yapısal programlamanın vazgeçilmez blokları fonksiyon ve fonksiyon bloklarıdır. Bu blokları kullanmaktaki amaç OB1 ana modül içerisinde alt programlar oluşturmaktır. Bu şekilde yazılan bir programın takibi ve hata tespiti çok daha kolay olmaktadır. Özellikle bir tesiste aynı fonksiyon işlevlerinde çalışan ve sayıları birden çok olan motor, valf, korna gibi nesnelerin programlanmasında çok kolaylıklar sağlamaktadır. Örnek verilecek olursa; bir tesiste aynı özellikte yani fonksiyon tanımları aynı olan 100 tane motorun çalışma mantığı OB1 içerisinde ayrı ayrı yazılabileceği gibi, FC veya FB bloklarının herhangi birisi içerisinde bir adet motorun çalışma mantığı lokal semboller kullanılarak oluşturulabilir. Hazırlanan bu blok 100 adet motor için kullanılır. Sadece yapılması gereken 100 adet motora ait giriş ve çıkışların adreslerinin girilmesidir. Yazılan programa ait veriler ise veri bloklarında saklanmaktadır. Bu veri blokları kendilerine erişime göre iki çeşittir:

- Bütün blokların erişebildiği veri blok (DB),
 - Sadece ilişkilendirilen fonksiyon bloğun (FB) erişebildiği özel veri blok (DI)
- [2].

3.1.2.1. Fonksiyonlar (FC)

Bir fonksiyon mantığının birden fazla kez program içerisinde çağrılması ve farklı değişkenlerle kullanılması bu bloğun parametrelendirilmesini zorunlu kılmaktadır. Bu şekilde bir fonksiyon yazabilmek için formal işlenen komut kullanılmaktadır. Program işletilirken ise aktüel işlenen komut, formal işlenen komutun yerini almaktadır. Formal işlenen komutun tanımlanması ise deklarasyon (bildirge) tablosunda yapılmaktadır. Bu şekilde hazırlanan program sonucunda sabit adresler

değil deklarasyon tablosunda tanımlanan semboller kullanılır. Bu blok OB1'in içinde veya OB1'in içerisinde taranan başka bir fonksiyonun içerisinde çağrıldığında formal işlenen adreslerin yerini gerçek adresler veya değerler almaktadır. Program fonksiyonun içinde formal işlenen adreslerle yazılırken, deklarasyon tablosunda tanımlanan sembolün başında diyez (#) işaretinin bulunduğu unutulmamalıdır. Fonksiyonların hafızaları yoktur, fonksiyon içerisindeki işlenmiş bilgiler, fonksiyon terk edilir edilmez kaybolurlar.

3.1.2.2. Fonksiyon blokları (FB)

Fonksiyon Blokları (FB), Fonksiyonlara (FC) benzer yapıdadırlar. Bir OB veya FC tarafından çağrılarak işlenirler fakat bu bloklar hafızaya sahiptirler. İçlerindeki bilgiler ilişkilendirilen veri blok (DB) içerisinde sonraki tarama çevrimine kadar tutulmaktadır. Bir fonksiyon blok (FB) mutlaka bir veri blokla ilişkilendirilmelidir. Eğer bu ilişkilendirme yapılmamışsa program CPU'ya yüklenirken PLC kendisi ilişkilendirecektir. Program yüklenirken FB ile ilişkilendirilen DB, mutlaka PLC'ye yüklenmelidir. Aksi halde program hata kodu üretecektir.

Fonksiyon blokları da fonksiyonda olduğu gibi bildirge tablosuna sahiptirler ve parametrelendirilmeleri burada yapılmaktadır.

3.1.3. Veri blokları (DB)

Veri blokları, fonksiyon ve fonksiyon modüllerinde olduğu gibi mantık işlemlerinin yazıldığı yani programlamanın yapıldığı bloklar değildirler. Sadece sabit verilerin saklandığı bloklardır. Örnek verilecek olursa sıralı şekilde çalışan ve duran yük taşıma sisteminde otomatik çalışma modunda motorlar çalış emri verildikten sonra SCADA'dan veya operatör panelinden girilen zaman değerleri kadar beklerler ve çalışmaya başlarlar. Eğer bu motor lojiği bir fonksiyon blok (FB) içerisine yazılmışsa bu FB bir veri blokla ilişkilendirilmelidir. İlişkilendirilen bu veri bloğun ismi özel

veri blok (DI) olmaktadır. DI'ya sadece kendisiyle ilişkilendirilen FB erişebilir, diğer modüller erişemezler. DI, sadece bir tek FB ile ilişkilendirilebilirken; FB, birden fazla DI ile ilişkilendirilebilir. İlişkilendirilmeyen veri bloklarına genel veri blokları denilmektedir ve bu tür veri bloklarına diğer kod modülleri erişebilirler.

Ayrıca veri bloklardaki benzer verileri defalarca yazmak gerekebilir. Örneğin, aynı tür bilgilere sahip olan bir yıldız üçgen çalışan motor devresindeki verileri tesisteki motor sayısı kadar yazmak yerine genel veri bloklar içerisinde UDT (User defined Data Type) kullanıcı tanımlı veri bloklar çağırarak bu işlem çok kolay şekilde çözülebilir. Bunu gerçekleştirmek için ise öncelikle UDT (Kullanıcı Tanımlı Veri Bloğu) içerisine; çalışıyor, arıza bilgileri, çalışma zamanı, durma zamanı set değerleri gibi standart bilgiler girilir. Daha sonra oluşturulan UDT, DB'nin içerisinde veya FB'nin deklarasyon tablosunda çağrılır. Bu şekilde birden fazla motora ait bilgiler bir seferde girilmiş olur. Eğer UDT, DB veya FB'de çağrıldıktan sonra veri eklemek veya çıkarılmak istenirse tanımlanan UDT'nin yapısı değişeceğinden DB ve FB'deki UDT de bozulacaktır. Bunu düzeltmek için mutlaka FB ve DB içerisinde, SIMATIC Step-7 programındaki kontrol ve güncelleme komutu ile bu bloklar yenilenmelidir.

Fonksiyon (FC) ve Fonksiyon Bloklarını (FB) kullanma zorunluluğu olmadığı gibi veri bloklarını da kullanma zorunluluğu yoktur. Yani bu blokları kullanmadan program yazmak mümkündür fakat kullanılması hem programa esneklik kazandırır hem de PLC'de arıza takibi yapmak kolaylaşır. Operatör tesisi gözlemlerken, PLC yerine arayüz olan operatör panel veya SCADA'dan izlemek ister. Bunu gerçekleştirmek için verilerin dış dünyadan girilebilir hale getirilmesi gerekmektedir. Veri blokları tam olarak bu tür durumlarda kullanılmaktadır. Veri blokları hafıza bitlerine (memory, merker) benzemektedirler. Hafıza bitlerinin kullanıldığı her yerde veri bitleri de kullanılabilir.

4. SCADA SİSTEMLERİ

4.1. Giriş

SCADA, Supervisory Control and Data Acquisition kelimelerinin ilk harflerinden oluşmuştur. Türkçe' ye “Danışmalı Kontrol ve Veri Toplama Sistemi” veya “Denetimli Kontrol ve Veri Toplama Sistemi” olarak çevrilebilir. SCADA sistemi geniş bir alana yayılmış cihazların bir merkezden bilgisayar aracılığıyla denetlenmesini, izlenmesini önceden tasarlanmış bir mantık içerisinde işletilmesini ve geçmiş zaman birimine ait verilerin saklanması sağlayan sistemlere verilen genel isimdir [3].

Günümüz iletişim ve bilgi dünyasında birçok süreç izlenmekte, takip ve kontrol edilmektedir. Artık cep telefonları bir kameraya bağlanıp herhangi bir yer izlenebilmekte, bir mesaj ile uzaktan bazı cihazlar kontrol edilebilmektedir.

Endüstriyel bir tesiste izleme ve kontrolün gerekliliği tartışılmazdır. O anda sistemin nasıl çalıştığı, üretimin hangi aşamada olduğu, ne kadar üretim yapıldığı veya nerede arızanın olduğu izlenmesi gereken önemli bilgilerdir. Geçmişe yönelik verilerin toplanması, grafiklerin oluşturulması sayesinde sistemin daha düzgün bir şekilde çalışmasını sağlamak ve arızaların önüne önceden geçmek mümkündür [4].

Daha geniş olarak açıklayacak olursak SCADA; süreç veya bina otomasyonunda kullanılan programlanabilir denetleyiciler (PLC), dağıtık kontrol sistemleri (DCS), I/O sistemleri ve akıllı sensörler gibi çeşitli cihazlardan saha verilerini sürekli ve gerçek zamanlı olarak toplayan, gerektiğinde kullanıcıya erken uyarı mesajları

üreten, üretimi etkileyen çeşitli etkenlerin merkezi bir noktadan grafiksel olarak gözetlenmesini sağlayan ideal bir denetleyici gözetim ve veri toplama sistemidir.

SCADA yazılım paketleri endüstriyel tesislerde alt yapı yazılım görevini üstlenmekte ve fabrika içi ile dışındaki ağlara bağlanarak şirketin bütün katmanlarının uyum içerisinde çalışmasına imkân vermektedir. SCADA işletme genelinde herkese her zaman erişebilecekleri, gerçek zamanlı ve ayrıntılı bilgiyi sağlamaktadır.

Amaç minimum maliyetle, daha kaliteli ve daha çok üretmek için gerekli yapıyı kurmaktır. İşletmedeki tesislerden maksimum verimlilikle yararlanmak, yöneticilerin işletmeye ve üretim bilgilerine tam olarak hâkim olmasını sağlamak, SCADA sistemlerini fabrikalara aktararak bu firmaların otomasyon sistemlerini modern hale getirmekle mümkündür [5].

SCADA sisteminin başlıca kullanım alanları şunlardır.

1. Tüm fabrika otomasyonunda ve modernizasyonlarında,
2. Bina ve mağaza otomasyonunda,
3. Isıtma, soğutma ve iklimlendirme otomasyonlarında,
4. Seracılık, hayvancılık, tarım sanayinde,
5. Tekstil, boya, otomotiv sanayinde,
6. Fermantasyon sistemlerinde,
7. Makine ve gıda sanayinde,
8. Fiziksel büyüklüklerin ölçülmesi, gösterilmesi, kontrolü, verilerin saklanması, iletilmesi ve raporlanmasında,
9. Bir merkezden denetlenen sistemlerde,
10. Kimya endüstrisi,
11. Petrokimya endüstrisi,
12. Demir-Çelik endüstrisi,
13. Elektrik üretim, iletim ve dağıtım sistemleri,
14. Su toplama, arıtma ve dağıtım tesisleri,

15. Çimento endüstrisi,
16. Orman ürünleri endüstrisi,
17. Otomotiv endüstrisi,
18. Trafik kontrolü,
19. Gıda endüstrisi.

SCADA'nın kullanım yerlerinin çok fazla olmasından da anlaşılacağı üzere artık günümüzde üretimin olduğu her fabrikada daha kaliteli bir ürün üretebilmek ve bütün işlemleri bir merkezden kontrol edebilmek için PLC'nin ara yüzü olan SCADA programını kullanmak şart olmuştur.

SCADA'nın temellerini mimik diyagram oluşturmaktadır. Mimik diyagramlar bir pano üzerine yerleştirilmiş ve otomasyon sistemlerindeki motor ve makinaları temsil eden ledlerden oluşmaktadır.

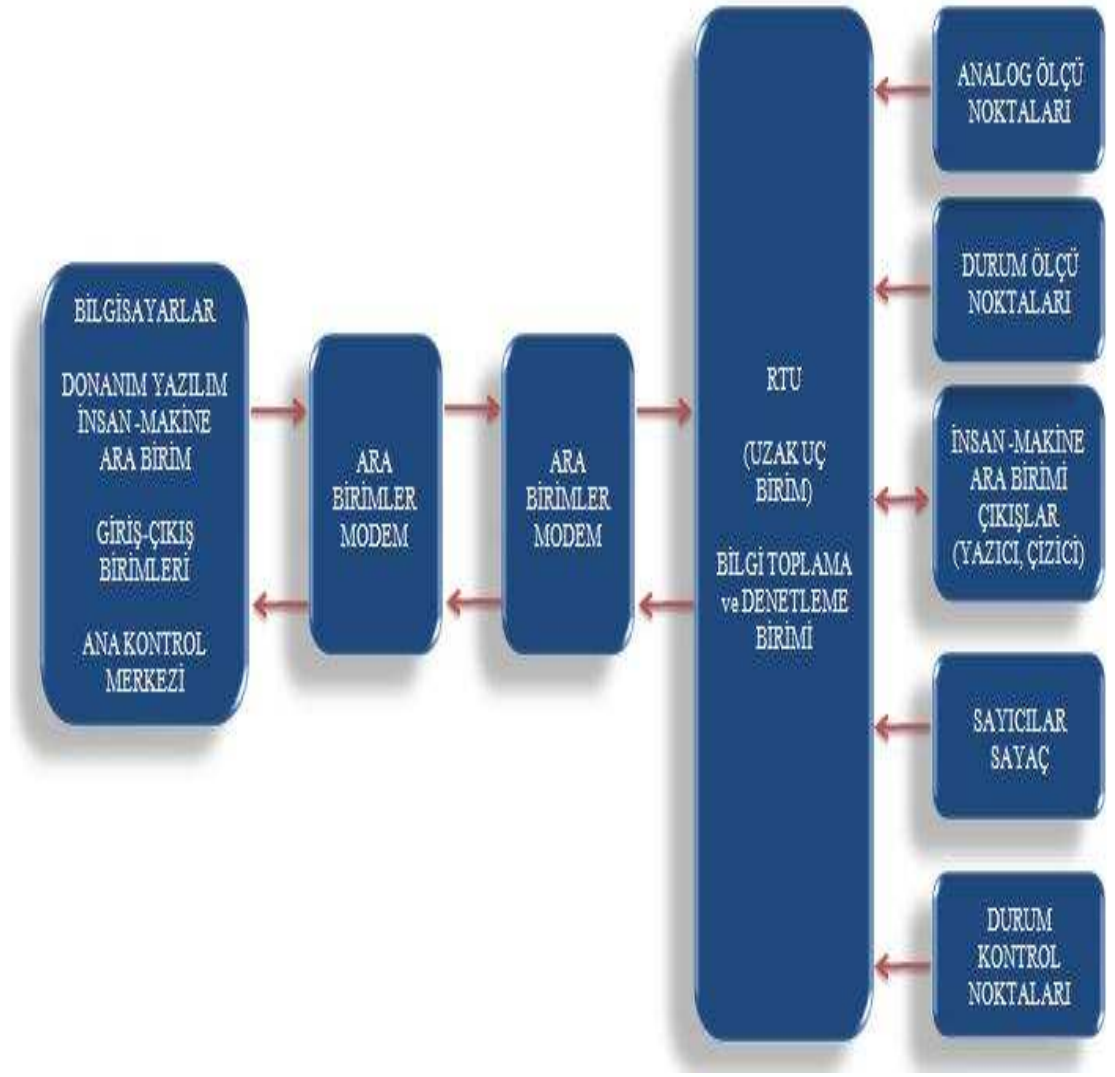
Üretimin bütün aşamalarının SCADA ile gözlemlenmesi, üretimi durdurabilecek büyük arızaların önceden tespit edilmesini ve iş kazalarının azalmasını sağlamaktadır.

4.2. SCADA Sisteminin Yapısı

SCADA esas olarak üç bölümden oluşur.

1. Uzak Uç Birim (Remote Terminal Unit: RTU)
2. İletişim Sistemi
3. Kontrol Merkezi Sistemi (Ana Kontrol Merkezi AKM: Master Terminal Unit MTU) [3]

Şekil 4.1'de SCADA sisteminin genel yapısı görülmektedir.



Şekil 4.1: SCADA sisteminin genel yapısı [3]

4.2.1. Uzak uç birim (Remote terminal unit: RTU)

RTU, SCADA sistemlerinde oldukça sık kullanılan bir kısaltmadır, İngilizcede Remote Terminal Unit kelimelerinin ilk harflerinin alınmasıyla kısaltılmıştır. RTU-Remote Terminal Unit deyimi Türkçeye tanı kelime anlamıyla "Uzak Uç Birim" veya "Uzak Giriş Birimi" olarak çevrilebilir.

RTU'nun fiziksel olarak üzerinden bilgi toplayabileceđi, gerektiđinde kumanda edebileceđi giriř ve ıkıř noktaları vardır. Elektrik tesislerinde; akım ve gerilim trafoları, ayırıcı, kesici ve rle durumları RTU tarafından izlenmekte ayrıca aynı RTU tarafından tali merkezdeki eřitli birimlerin kontrol mmkn olabilmektedir.

Bir SCADA sisteminde bir veya birkaç Kontrol Merkezi olabilirken aynı sistemde RTU sayısı yzlerce olabilmektedir. Bu nedenle RTU'lar sistemin tařınabilirliđi, gvenilebilirliđi ve zellikle maliyeti gibi nemli đelerinin dođrudan belirleyicisi olmaktadır. RTU'ların kk boyutta olması ve kullanılacak blgelerin dođal kořullarına dayanabilecek řekilde retilmesi ok nemlidir.

Gnmzde RTU'lar mikrořlemcilerin her geen gn deđiřmesi sayesinde esnek, ok fonksiyonlu, daha akıllı ve ekonomik hale gelmektedir. Temel fonksiyonları deđiřmemek kaydıyla RTU'lar gn getike artan kullanıcı isteklerine cevap verecek řekilde geliřtirilmektedir.

Tanımından da anlařılacađı gibi RTU'nun en nemli iki grevi;

1. Bilgi toplamak ve depolamak,
2. Gerekli Kumandaları gerekleřtirmektir.

RTU'lar tali merkezlerde analog deđerler, alarm ve durum bilgileri ve saya deđerlerini toplarlar. Bylece bađlı oldukları tali merkezlerin ve ait oldukları ana merkezin ihtiyacı olan tm bilgileri toplayarak otomasyonun ilk prensibini gerekleřtirmiř olurlar. Toplanan bu bilgileri kendi zerlerindeki hafızalarda saklarlar. Bu bilgiler; AKM (Ana Kontrol Merkezi) kendilerini sorgulayıncaya kadar veya ayarlanan belli sreler iin saklanır. Bilgi toplama iřini kendilerine verilen periyodik aralıklarla veya ayarlandıkları deđerlerden sapmalar oluřtuđunda yeni deđerleri kaydetmek řeklinde yerine getirirler.

4.2.2. İletişim sistemi

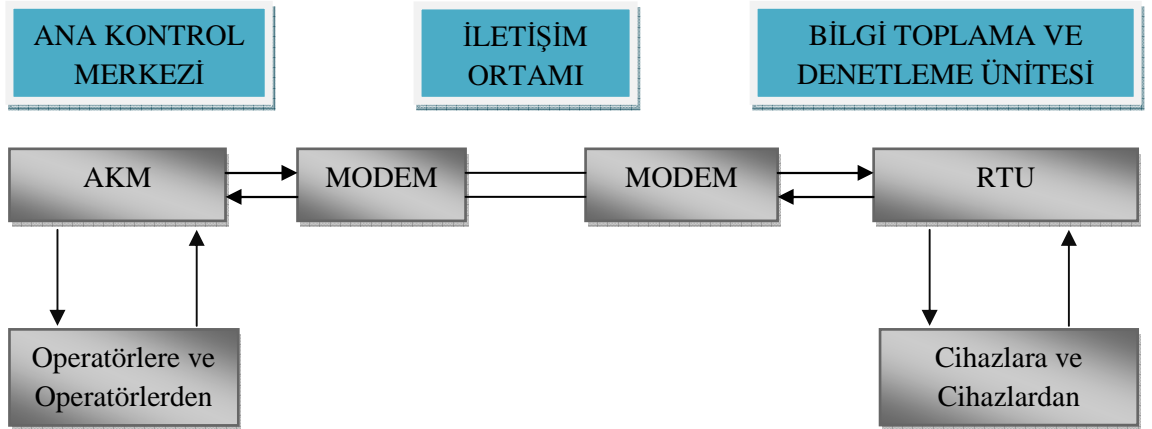
İletişim, bir bölgeden başka bir bölgeye karşılıklı olarak veri veya haberin gönderilmesi işlemidir. SCADA sisteminin en yüksek başarı düzeyi ile uygulanması iletişim sistemine bağlıdır. SCADA'nın başarılı bir şekilde uygulanabilmesi için;

- a) Güvenilir
- b) Maliyeti düşük
- c) Gerekli tüm fonksiyonlara sahip
- d) Her türlü ortamda çalışabilen bir iletişim sistemine sahip olmalıdır.

Bir SCADA sisteminde iletişim sistemi, Kontrol Merkezi ve bir RTU ünitesinin haberleşebilmesini sağlamaktadır. İletişim sisteminin elemanları şunlardır;

1. İletişim ortamı,
2. Veri İletişim Cihazı (MODEM),
3. İletişimi sağlanan Cihazlar (AKM, RTU),

Şekil 4.2'de SCADA iletişim sisteminin yapısı görülmektedir.



Şekil 4.2: SCADA iletişim sisteminin yapısı [6]

İletişim ortamları; gerilim hatları, kiralanmış hatlar, radyo frekansı, özel bir hat olabilmektedir.

İletişim mimarisi aşağıda belirtilen etkenlere göre belirlenmektedir;

1. Sistemde kullanılacak RTU'ların sayısı,
2. RTU'ya bağlı birimler ve bu birimlere ulaşım hızı,
3. RTU'ların yerleşimi,
4. Elde bulunan haberleşme kolaylıkları,
5. Ulaşılabilecek haberleşme teknikleri ve araçları,
6. AKM ve RTU arasındaki mesafe.

4.2.3. Kontrol Merkezi Sistemi (Ana Kontrol Merkezi AKM – Master Terminal Unit MTU)

Kontrol merkezi; geniş bir coğrafi alana yayılmış tesislerin, bilgisayar esaslı bir yapıyla uzaktan kontrol edildiği, izlendiği ve yönetildiği yer olarak tanımlanabilir. Kontrol Merkezleri genelde SCADA sistemlerinin ve kontrol edilecek tesislerin merkezi bir yerine kurulur.

SCADA sistemi bir insan vücuduna benzetilecek olursa, RTU'lar insan vücudunun organlarına, AKM'ler de insan vücudunun beyin ve fonksiyonlarına karşılık gelmektedir. Kontrol Merkezleri için, sistemde, SCADA sisteminin büyüklüğüne göre ayrı bir mekân olmalıdır. Bu ayrı Kontrol Merkezinden; tüm SCADA sistemine kumanda edilerek, gerekli bilgiler toplanır, uygun bir veritabanı programı ile bilgiler depolanır. Gelen veriler ve alarmlar analiz programları ile yorumlanır, veriler üzerinde işlem yapılır.

Kontrol Merkezi SCADA sistemi içinde, bir tane olabileceği gibi, sistemin büyüklüğüne göre, birkaç tane de olabilir. Çok büyük sistemlerde AKM'ler altında Alt Kontrol Merkezleri de bulunmaktadır [6].

4.3. SCADA Sisteminin Yararları

SCADA, otomasyon sistemlerinde kullanılan motorların, vanaların ve birçok makinenin işleyişini sembolize ederek bilgisayar ekranında görme olanağı sağlar. Raporlama ve alarm bilgilerini ekrana taşıma ve kaydetme olanağı sunar. Hatta alarm bilgileri cep telefonlarına mesaj şeklinde de gelebilmektedir. Ayrıca SCADA sistemindeki veri sunucuların internete bağlı olması, sistemin web üzerinden izlenmesi ve kontrol edilmesini sağlamaktadır [7].

4.4. SCADA Sisteminin Fonksiyonları

SCADA sisteminin işlevleri;

- a) İzleme işlevleri
- b) Kontrol işlevleri
- c) Veri toplama
- d) Verilerin kaydı ve saklanması olmak üzere dört grupta toplanabilir.

a) İzleme işlevleri

- Durum denetimi (açık-kapalı)
- Eşik ve limit değer denetimi (analog ölçümler)
- Olay ve alarmların rapor edilmesi, gruplandırılması, sınıflandırılması
- Trend denetimi

b) Kontrol işlevleri

- Kontrol edilecek cihazların tek tek kontrolü (ayırıcı ve kesicileri uzaktan açılıp kapatılması, trafo deęiřtirici kontrolü vb.)
- Regülatörlere veya rölelere kontrol işaretleri gönderilmesi

c) Veri toplama

- Analog ölçümler (akım, gerilim, aktif ve reaktif güçler, yağ ve sargı sıcaklıkları, kademe deęiřtirici konumu, vb.)
- Durum ölçüleri (kesici ve ayırıcıların açık kapalı konumları, röle kontak konumları vb.)
- Enerji ölçümleri (sayaç çıkışlarından alınan birim enerji işaretlerinin sayılması)

d) Verilerin kaydı ve saklanması

- Denetlemeli kontrol ve veri toplama işlevlerinden elde edilen veriler isteęe baęlı aralıklarla ve istenen şekillerde kaydedilerek istenen sürelerde saklanır [3].

5. WINCC SCADA PROGRAMI

5.1. Giriş

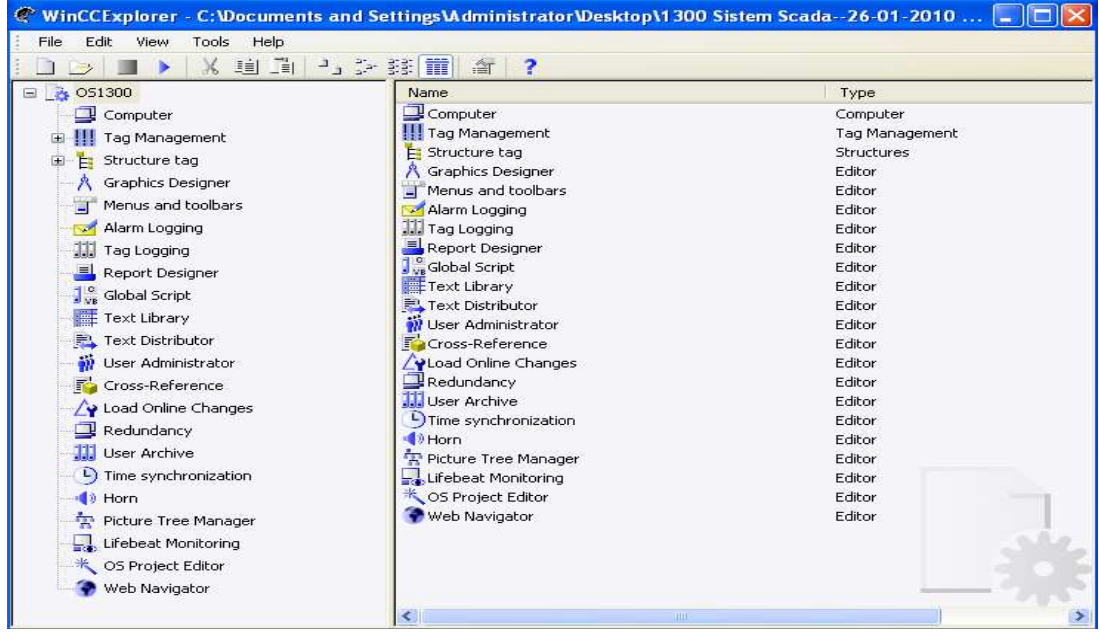
WINCC dünyanın ilk IHMI (insan arayüzlü makine bütünleşmesi) yazılım sistemidir. WINCC'nin kullanıma uyumlu parçaları, yeni uygulamalarla bütünleşmesine imkân sağlar. WINCC ile tüm işlem akışını çok kolay bir yoldan ve tüm yönleriyle otomatik olarak gözlemleyip, kontrol edilebilir. WINCC, görüntüleme işlemlerinin çözümü ve kontrol görevleri için birçok gerekli bileşen içerir [8].

WINCC'de proje oluştururken aşağıdaki sıra izlenir:

- Projeyi oluşturma,
- PLC sürücüyü yükleme,
- Etiketleri belirtme,
- Üretim şekilleri oluşturma,
- WINCC çalışma ayarını kurma,
- Şekilleri aktif hale getirme,
- Şekilleri test için simulator kullanma.

Program çalıştırıldığında proje yapılarıyla ilgili, Tekli Kullanıcı Projesi (Single-User Project), Çoklu Kullanıcı Projesi (Multi-User Project) ve Çoklu İstemci Kullanıcı Projesi (Multi-Client-User Project) olmak üzere üç seçenek bulunmaktadır.

Bu çalışmada birinci seçenek olan Tekli Kullanıcı Projesi kullanılmıştır. Şekil 5.1'de WINCC SCADA programının genel görüntüsü görülmektedir.



Şekil 5.1: WINCC programının genel görüntüsü

Sol alt pencerede WINCC Explorer'ın ana öğeleri görülmektedir. Bunlar proje kişiselleştirme adımlarında yardımcı olacaktır. Gizli bölümler “+” sembolüyle gösterilmektedir. Sağ alt pencerede, seçilen öğelerin içerikleri bulunur.

5.2. WINCC SCADA Programı Ana Öğeleri

5.2.1. Bilgisayar (Computer)

Bu kısımda programlamanın yapıldığı bilgisayarın özellikleri girilir. Ayrıca startup sekmesinde program başlatıldığında çalıştırılmak istenen runtime seçenekleri işaretlenir (Örnek: Alarm Logging Runtime, Tag Logging Runtime, Graphics Runtime...). Parameters sekmesinde runtime dili ayarlanır. Graphics sekmesinde SCADA programı çalışmaya başladığında ilk ekrana gelecek sayfa olan Başlangıç Sayfası (Start Picture) seçilir ve bu sayfanın özellikleri ayarlanır.

5.2.2. Etiket yöneticisi (Tag management)

Bu bölümde WINNCC SCADA programı ile PLC arasındaki iletişimi sağlamak için iletişim protokolleri (communication drivers) seçilir. Bu protokollerden Simatic S7 Protocol Suite seçildiğinde; Endüstriyel Ethernet, MPI (Multi Point Interface), Profibus, TCP/IP seçenekleri bir arada sunulmaktadır. Tez için hazırlanan bu çalışmada TCP/IP protokolü kullanılmıştır. Burada iletişim kurulan PLC'nin CPU'suna ait IP adresi ve bulunduğu ray numarası ile raydaki sıra numarası (slot numarası) girilmelidir.

TCP/IP protokolünün altında sisteme ait başlıca etiket grupları oluşturmakta fayda vardır. Bu grupların altına etiketler (tag) girilmelidir. Etiketler, PLC içerisindeki verilerin SCADA programına aktarılmasında kullanılır. PLC programında genel olarak SCADA ile haberleşecek veriler bir veri blokta toplanılmakta ve bu veri bloktaki adreslerin tamamı etiketlerle ilişkilendirilerek SCADA ekranındaki görsel öğelere bağlanmaktadır.

Etiketler WinCC içinde tank seviyeleri gibi gerçek değerleri sunmak için kullanılırlar. Üretim etiketleri (Process Tags) PLC veya benzer parçalar için projedeki hafıza odacıklarıdır. Örneğin su tankının su seviyesi, seviye algılayıcı sensörler yardımıyla algılanıp PLC'de kaydedilecektir ve bağlantı yolu ile tankın su seviyesi WinCC'ye gönderilecektir. İç etiketler (Internal Tags) WinCC'nin içinde bulunan hafıza odacıklarıdır. PLC ile aynı görevi yaparlar. Bu etiketler WinCC'nin içinde biçimlendirilir [8].

5.2.3. Yapısal etiket (Structure tag)

Birden fazla nesnenin etiket yapıları aynı ise yapısal etiket oluşturarak bir etiket gruplama yöntemi oluşturulabilir. Örnek olarak çift yönlü bir motor kumandası için etiket içinde motor durum bilgileri, sıralı çalışma zamanı ve sıralı durma zamanı

word (kelime) formatında girilebilmektedir. İÇerideki üç etiketin ofset ayarlarının yapılması gerekmektedir. Motor ismi girildikten sonra yapısal etiket seçilerek başlangıç adresi girilir. Bu şekilde birden fazla motor için her bir etiket içeriğini ayrı ayrı yazma problemi kalkmış olacaktır.

5.2.4. Grafik oluşturucu (Graphics designer)

Bu bölümde süreç resimleri oluşturulur. Sistemin süreç resimlerini oluşturmak için P&I (pipe and instrument-boru ve enstrüman) diyagramlarından faydalanılır. Sahada bulunan ve birbirinden uzakta bulunan makinalar bu sayede tek ekranda sembolik çizimlerle izlenebilir.

5.2.5. Alarm kaydedici (Alarm logging)

Sistemde oluşan uyarı ve hataların kullanıcıyı bilgilendirilmesi için alarm sayfasında görüntülenecek metinler burada etiketlerle ilişkilendirilir. Mesaj bloklarının içerisinde mesajlara ait sıra numarası, tarih ve saat bilgilerinin yer aldığı sistem blokları ve kullanıcının mesajı tam olarak tanımlayabilmesi ve gruplandırabilmesi için 10 adet metin blokları bulunmaktadır. Bu kullanıcı bloklarını kullanarak PLC'deki etiketin bağlı olduğu giriş-çıkış veya veri blok adresi, sembol adı, alarm kaynağı girilebilir. Ayrıca mesajların öncelik sıraları ve mesajların sınıflandırılması (uyarı, hata, alarm) yapılabilmektedir. Mesajları özelleştirerek ekranda gözüken mesajın metin ve arka planı renklendirilerek gelen mesajın uyarı veya alarm olduğu kolayca anlaşılabilir.

5.2.6. Etiket kaydedici (Tag logging)

Etiket girişini oluşturmak için kullanılır. Burada öncelikle etiketlerin PLC'den okunma sıklığını ayarlamak için zamanlayıcının ayarlanması gerekmektedir.

Zamanlayıcının ayarlanması: Zamanlayıcının içerisinde 500ms, 1sn, 1dk, 1s,1gün gibi zaman dilimleri bulunmaktadır. Zamanlayıcılar giriş yapılarak veya arşivleyerek kurulurlar. Giriş zamanları, etiket düzenleyicisi tarafından veri yöneticisinin süreç görüntülerinden alınan zaman aralıklarıdır.

Zamanların arşivlenmesi, verilerin arşivlendiği zaman aralıklarıdır. Arşivlenmiş veri her zaman için giriş zamanının tam sayı ile çarpılmışıdır. Her arşivlenmiş aralıkta depolanan değer her zaman girişi yapılmış son değerdir. İlk değer bir önceki aralığa aittir [8].

Arşiv oluşturma: Arşiv sihirbazı kullanılarak kolay bir şekilde arşiv etiketleri hazırlanabilir. Arşiv etiketinin adı verildikten sonra içerisinde sistemdeki analog değerlerin kaydının yapılacağı etiket isimleri verilir daha sonra okunup kaydedileceği döngü zamanı zamanlayıcı içerisinde seçilir, bu zaman girilecek faktör sayısı ile çarpılarak döngü zamanı ayarlanır. Örnek olarak; akımlar arşiv klasörü altında M01_AKIM (M01 motoru akım değerinin bağlandığı etiket) oluşturulduktan sonra döngü zamanı 1sn seçilir ve faktör olarak da 2 seçildiğinde döngü zamanı $2 \times 1 \text{sn} = 2 \text{sn}$ olarak ayarlanmaktadır. Oluşturulan arşiv etiketleri daha sonra oluşturulacak olan çizelge grafikleri ve tabloları için veri kaynağı sağlamaktadır.

Trend (çizelge) penceresi oluşturma: Bir çizelge (trend) penceresi, süreç etiketlerinin grafik formunda görüntülenmesini sağlar [8]. Yeni bir grafik sayfasında WINCC online Trend Control nesnesi eklendiğinde Trendler sekmesinde gösterilmek istenen çizelge isimlendirilir ve kendine ait olan etiketle ilişkilendirilir. Ayrıca eğrinin tipi ve rengi, çizelge penceresinin adı, çizelgenin yatay ve düşey eksenlerinin adlandırılması da çizelge penceresinde yapılmaktadır.

5.2.7. Rapor düzenleyici (Report designer)

Rapor düzenleyici, mesajların oluşturduğu raporların yazıcı ile alınması için bir çizelge oluşturulmasına olanak sağlar. Raporları hazırlamak ve ayrıca bu raporları istenilen formatta bastırmak için kullanılır. Rapor düzenleyicide veri çıkışı için dinamik değerler kullanılır. Statik objeler ve sistem objeleri, raporda sabit olarak görüntülenebilirler ya da tamamen hazırlanan formata bağlı olarak görünmeyebilirler.

Sistem, aşağıdaki raporlama şekillerini destekler:

1. Operatör kontrollü raporlama,
2. Zaman kontrollü raporlama,
3. Raporu yalnızca ekranda gösterme,
4. Önceden seçilen yazıcıdan çıktı alma [6].

5.2.8. Global komut editörü (Global script)

Bu bölümde C ve VBS (Visual Basic) editörleri ile yazılım yapılmaktadır. 4 ana başlık yer almaktadır. Proje fonksiyonları, standart fonksiyonlar ve intern fonksiyonlar; obje bağlantılı genel C dili aksiyonlarında kullanılır [6].

1. Proje Fonksiyonları: Yeni proje fonksiyonunun oluşturulduğu modüldür. Hazırlanan fonksiyonlar, yalnızca içinde bulunan fonksiyonda geçerlidir.
2. Standart Fonksiyonlar: Yeni standart fonksiyonların oluşturulduğu ya da var olanların değiştirildiği modüldür. Buradaki fonksiyonlar bütün projelerde çağrılabilir.
3. Intern (Dâhili) Fonksiyonlar: Bu fonksiyonlar bütün projelerde çağrılabilir fakat üzerinde herhangi bir değişiklik yapılamaz.

4. Aksiyonlar: Burada içinde bulunulan projede geçerli aksiyonlar tanımlanır. Bu aksiyonlar o proje için globaldir. Herhangi bir objeye bağlı olmaksızın arka planda çalışırlar.

5.2.9. Metin kütüphanesi (Text library)

Bu bölümde metinlerin farklı dillerdeki karşılıkları girilerek yabancı dil sözlüğü oluşturulur. Böylelikle SCADA projesi tamamlandığında eğer dil seçimi konulursa tek bir butonla bütün proje İngilizceden, Almancaya, Türkçeye kolaylıkla dönüşüm sağlanır.

5.2.10. Metin dağıtıcı (Text distributor)

Bu bölüm sayesinde projenin tamamı veya seçilen kısmı text dosyası (.txt) veya CSV dosyası (.csv) uzantılarına çevrilerek ithal ve ihraç edilebilir.

5.2.11. Kullanıcı yetkisi (User administrator)

Bu bölüm ile çalışma zamanında (runtime) bazı sayfalara şifreler konularak sadece yetkililerin bu sayfalara erişimi sağlanmış olur. Böylelikle analog değerler ve çalışma zamanları gibi herkesin değiştirmesine izin verilmeyen sayfalar korunmuş olur.

5.2.12. Hiyerarşi yönetimi (Picture tree manager)

Bu bölümde ise çalışma zamanında gösterilecek sayfalar konteynırlar içerisine yerleştirilerek hiyerarşik bir yapıda bulunmasına olanak verir. Böylelikle her sayfa içerisinde sayfaları birbirine bağlayan butonlar otomatik olarak oluşturulur.

5.2.13. Haberleşme ağı görüntüleme (Lifebeat monitoring)

Bu bölüm ile haberleşen cihazların durumları gözlemlenebilir. Örneğin; CPU Ethernet protokolü ile SCADA bilgisayarı arasında, PROFIBUS haberleşmesi ile de diğer ET modülleri ve motor sürücüleri arasındaki haberleşme durumları gözlenebilir. Böylelikle iletişim hattının koptuğu nokta bu sayfadan görüntülenebilir.

5.2.14. Operatör sistemi proje editörü (OS project editor)

Bu bölümde çalışma ekranının ayarları yapılmaktadır. Layout sekmesinde monitör sayısı ve çözünürlüğü ayarlanmaktadır. Area sekmesinde hiyerarşik konteynırların görünmesi istenmeyen sayfalar seçilerek gizlenebilir.

6. BİR MDF PRESİ DIŐ SAHA BESLEME HATTININ SIMATIC STEP-7 İLE PROGRAMLANMASI VE WINCC İLE SCADA PROGRAMININ YAZILMASI

6.1. MDF Presi DıŐ Saha Besleme Hattının Yapısı

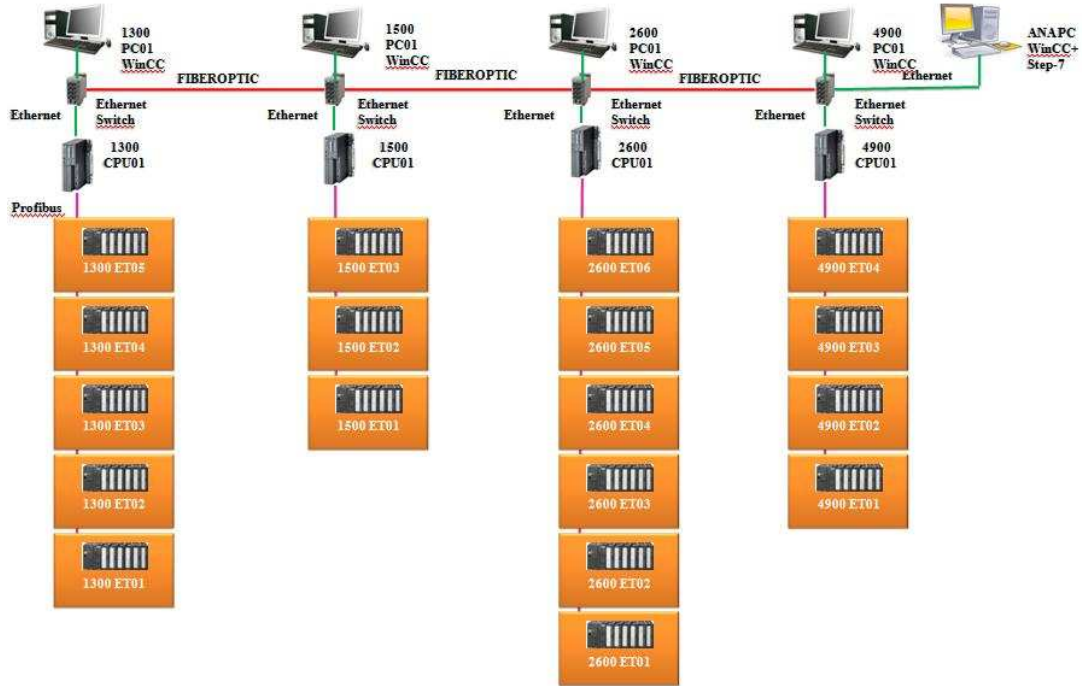
MDF; termomekanik olarak odun veya diđer lignoselüozik hammaddelerden elde edilen liflerin, sentetik yapıŐtırıcı ilavesiyle belirli bir rutubet derecesine kadar kurutulduktan sonra oluŐturulan levha taslađının sıcaklık ve basınç altında preslenmesiyle elde edilen bir üründür.

MDF'nin her noktasında liflerin eŐit dađılması ve çok yođun bulunması, levhanın her iki yüzünün olduđu kadar, kenarlarının da makinayla herhangi bir kırılma olmaksızın ya da malzeme parçacıkları arasında boşluklar ortaya çıkmaksızın işlenmesine imkân sağlamaktadır. MDF bu sayede masa tablaları, kapı panelleri ve profil yüzeyli çekmece alınları gibi parçaların üretilmesinde başarıyla kullanılabilir [9]. Şekil 6.1'de MDF presi dıŐ saha besleme hattının akıŐ diyagramı bulunmaktadır.

Otomasyonu yapılan mdf presi dıŐ saha besleme sistemi, genel olarak 1300 Sistemi (Yerden Besleme), 1500 Sistemi, 2600 Sistemi ve 4900 Sistemi olmak üzere dört ana kısımdan oluŐmaktadır. Bu dört sistemin tek bir merkezden kontrolü için PLC ve SCADA programları kullanılmaktadır. Kontrol odasındaki SCADA bilgisayarlarından bütün sistemlerden gelen uyarı ve mesajlar izlenmekte ve sistemlerin kontrolü ve birbirleri arasındaki koordinasyonları otomatik olarak yapılmaktadır. Bu tez çalışmasında 1300 Sisteminin (Yerden Besleme) otomasyonu anlatılmaktadır. Sistemde cips (odun parçacıkları, yonga) ve odun tozları ayrıŐtırılarak gerekli silolara gönderilmektedir.

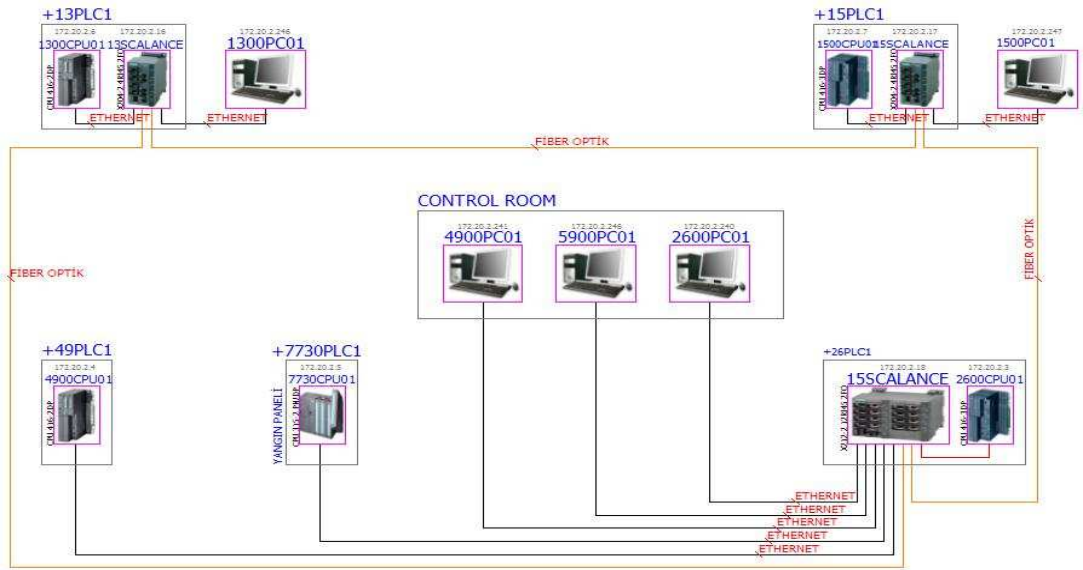
6.2. MDF Presi Dış Saha Hattının Haberleşme Kurgusu

Dış saha besleme hattı dört ana kısımdan oluşmaktadır. Her bir sistemi bir S7-400 PLC kontrol etmektedir. Her PLC, kendi SCADA bilgisayarıyla Ethernet ile haberleşmektedir. Ayrıca her PLC, Kompakt ve Yönetilebilir Endüstriyel Ethernet / Profinet Switchleri (Switch üzerinde Ethernet ve Fiberoptik portları mevcuttur) ile fiberoptik haberleşmektedirler. SCADA bilgisayarları kontrol odasında bulunmaktadır. PLC panelleri, ET panelleri; motor kontrol panelleri (Motor Control Center) anahtarlama odalarında ve yerel kontrol panelleri (Lokal Panel-LP) sahada bulunmaktadır. MCC ve LP üzerindeki dijital ve analog giriş çıkışlar ET modüllerinde toplanmaktadır. PLC, ET modülleri ve MCC içerisindeki motor sürücülerıyla Profibus ile haberleşmektedir. Ayrıca PLC'ler kendi arasında Profibus DP/DP Coupler cihazlarıyla profibus hattı üzerinden haberleşmektedirler.



Şekil 6.2: MDF presi dış saha besleme hattının bilgisayar – PLC ve uzak işlemci haberleşme kurgusu (Ethernet, Fiberoptik, Profibus)

Şekil 6.2’de MDF presi dış saha besleme hattının haberleşme kurgusu görülmektedir. Her sistemin bilgisayarı kendi sistemine ait PLC ile Ethernet ile haberleşmektedir. PLC’ler ise kendi uzak işlemcileriyle Profibus DP ile haberleşerek sahadaki verileri toplamaktadır. Her sistem arasında ise fiberoptik haberleşme vardır. Motor kontrol panelleri ve sahadaki kontrol panellerindeki veriler çok damarlı bakır kablolar ile uzak işlemcilerin olduğu panellere çekilerek PLC’lerin bu paneller üzerindeki verileri okuması ve yazması sağlanmıştır.



Şekil 6.3: MDF presi dış saha besleme hattı PLC ve bilgisayarları arasındaki Fiberoptik, Ethernet haberleşmesi

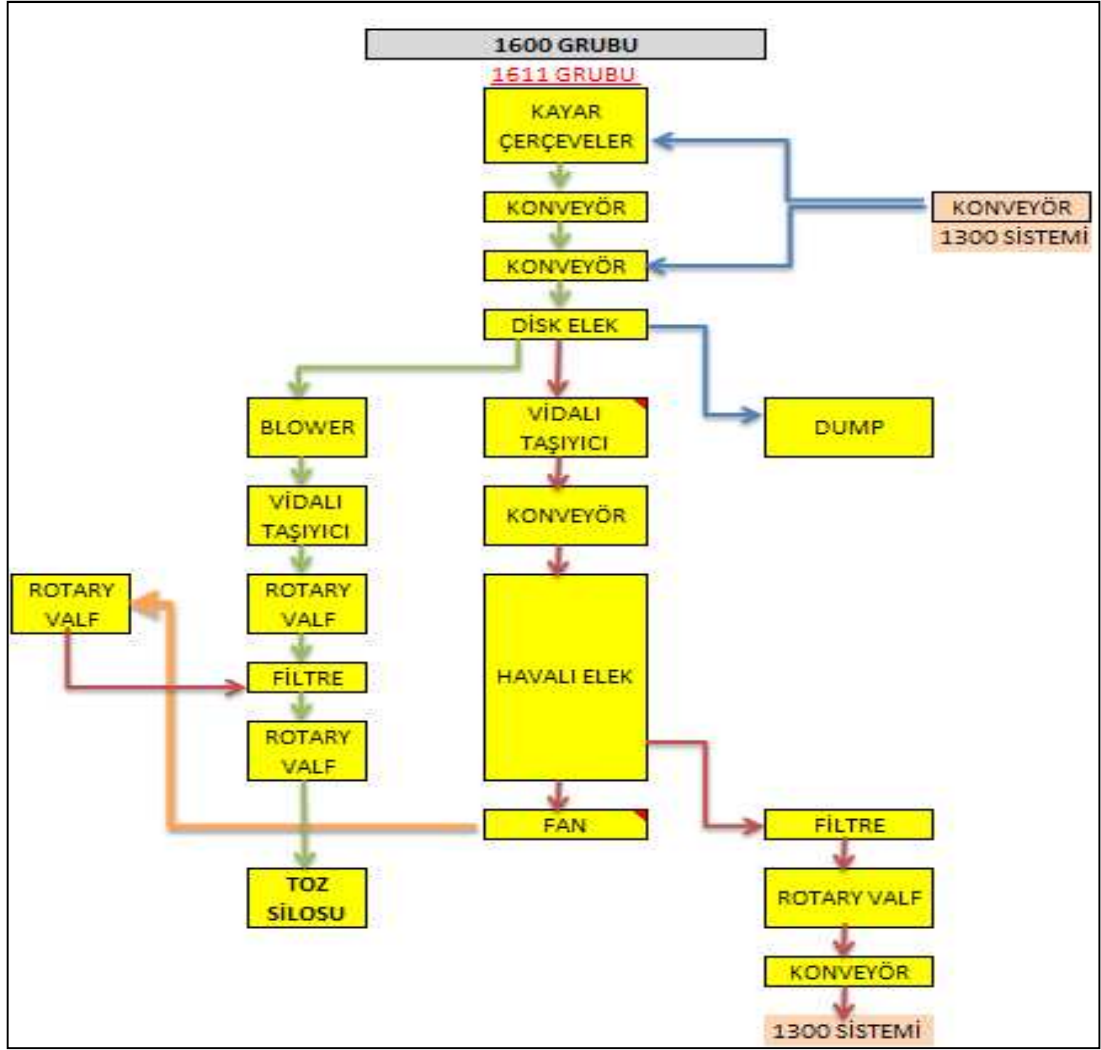
Şekil 6.3’de MDF presi dış saha besleme hattının PLC ve bilgisayarları arasındaki fiberoptik, ethernet haberleşmesi görülmektedir. Bu yapıda, dört besleme sistemi ve bir yangın söndürme sistemi bulunmaktadır. Besleme sistemlerinde S7-400 PLC, yangın söndürme sisteminde ise S7-300 PLC kullanılmıştır. PLC’ler üç ayrı kontrol merkezinden kontrol edilmektedir. Kontrol merkezleri arasında fiberoptik halka (ring) haberleşme yapısı kullanılarak kontrol odasından ana SCADA bilgisayarı ile bütün sistemler tek merkezden görüntülenebilmektedir. Yangın söndürme sistemi sadece ana SCADA bilgisayarından izlenmektedir.

kendilerine ait vidalı taşıyıcılara dökülmektedir. Daha sonra taşıma bantlarına konulan cipsler disk eleklerden geçirilerek ince, orta ve kalın olmak üzere üç boyuta getirilmektedir. Ayrıştırılan bu cipsler 1500 sistemine ait ince ve kalın cips silolarına doldurulmaktadır. Yerden besleme sisteminde cips silolarındaki seviyelerin sabit bir oranda tutulabilmesi için vidalı taşıyıcı motorları hem motor sürücüleri ile kontrol edilmekte hem de operatörün SCADA ekranından seçmesine olanak sağlanarak silo seviyesine göre vidalı taşıyıcıyı devreye alıp çıkartabilmektedir. Sistemdeki motorlar akış diyagramındaki sıra ile sıralı çalışıp sıralı durdurularak cipsin akış hattında tıkanma oluşturmadan kendine ait silolara doldurulması sağlanmıştır. Ayrıca sistemde taşıyıcı bant motorlarında ipli şalterler kullanılarak ve sahadaki bütün panellerin üzerindeki acil stop butonları ile oluşabilecek kazalara karşı can güvenliği sağlanmıştır.

6.3.2.1600 Grubu (Yerden Toz Besleme) Akış Diyagramı

1611 Toz silosu besleme noktasından kayar çerçevelerle ve bant motorlarıyla taşınan toz disk elekten geçirilerek boyutlandırılır ince parçacıklar 2400 sistemindeki toz silosuna gönderilir. Kalın parçacıklar içerisindeki yapancı cisimlerin veya büyük parçacıkların ayrıştırılması için havalı elekten geçirilir. Havalı elek içerisinde hava vardır. Yoğunluk farkına göre küçük cips parçacıkları ayrıştırılır ve mikro konveyörü ile mikro silosuna gönderilir.

Şekil 6.5’de 1600 Grubu (Yerden Toz Besleme) akış diyagramı görülmektedir.

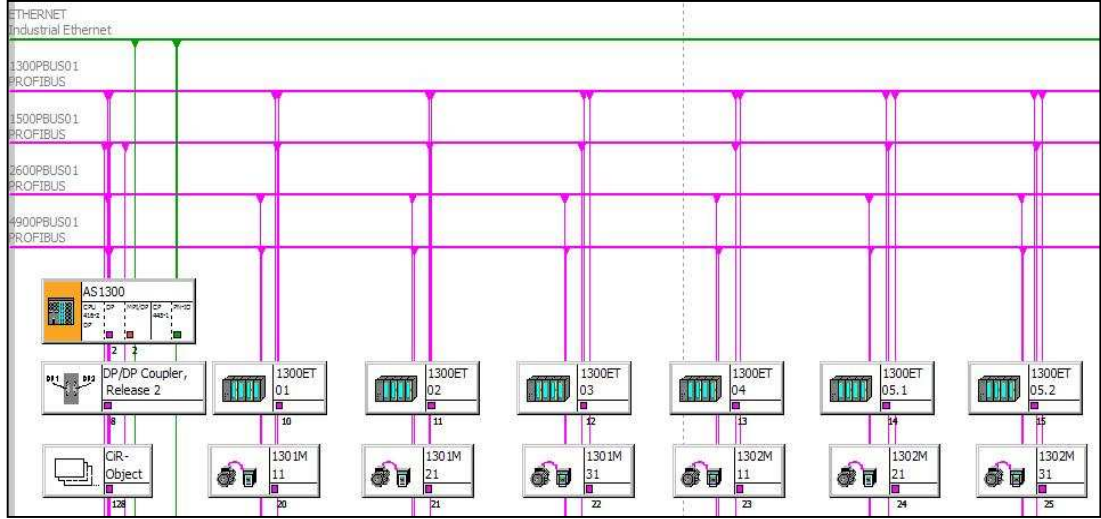


Şekil 6.5: 1600 grubu (yerden toz besleme) akış diyagramı

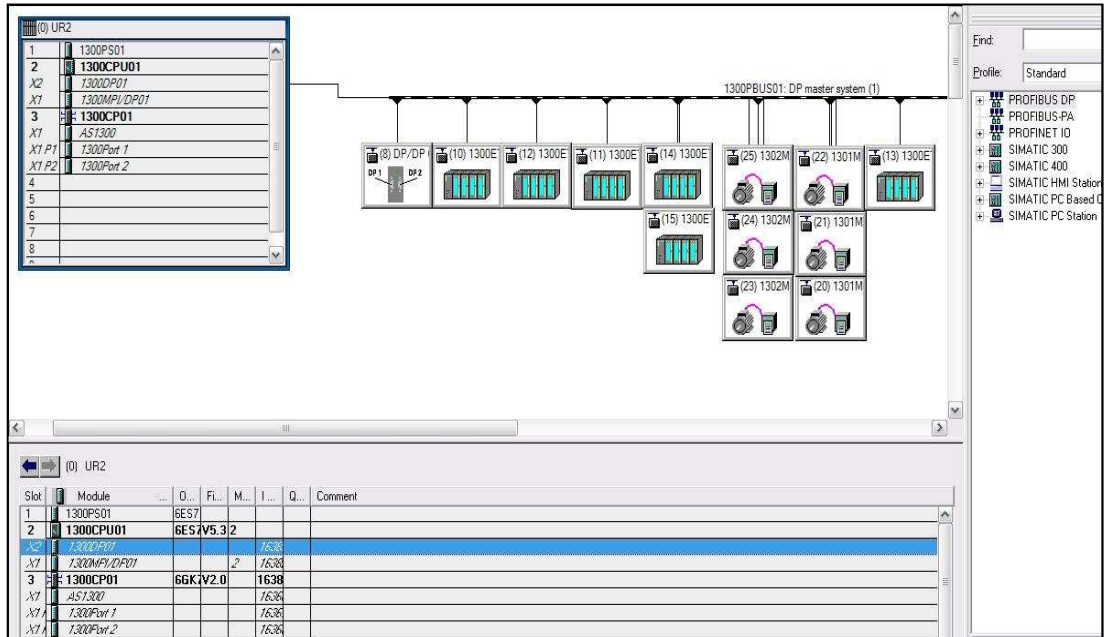
6.4. 1300 Sistemi (Yerden Besleme) Siemens Simatic Step-7 ile Programının Yazılması

PLC, bilgisayar ve uzak işlemcilerin ve giriş çıkış kartlarının yapılandırılması Step-7 Hardware Config ve Net-Pro sayfasında yapılmıştır. Burada PLC'nin CPU kartında MPI, Profibus DP adresleri ve Ethernet haberleşmesi için CP kartında IP adresi ayarlanmaktadır. Profibus haberleşme hattı üzerine de motor sürücüler ve uzak işlemciler (ET 200M, IM-153) eklenerek Profibus DP adresleri girilmektedir. Şekil

6.6'da Net-Pro sayfasında 1300 sistemi haberleşme yapısının oluşturulması görülmektedir.



Şekil 6.6: Net-Pro sayfasında 1300 sistemi (yerden besleme) haberleşme yapısının oluşturulması.



Şekil 6.7: Step-7 Hardware Config sayfasında 1300 sistemi (yerden besleme) haberleşme yapısının oluşturulması.

Şekil 6.7’deki haberleşme ağı yapısı oluşturulduktan sonra giriş çıkışları programda anlaşılır hale getirmek için sembol tablosu oluşturulur. Böylelikle sistemde kullanılan yüzlerce giriş çıkış adresleri ve iç bellek hafızaları semboller ile kullanılacaktır. Şekil 6.8’de sembol tablosu görülmektedir.

	Status	Symbol ▾	Address	Data type	Comment
249		1636R01	Q 20.3	BOOL	HP Blower Çalış
250		1636Q01	I 81.2	BOOL	HP Blower Arıza
251		1636K01.2	I 81.4	BOOL	HP Blower Çalışıyor
252		1636H01.1	Q 3.7	BOOL	HP Blower Çalışıyor Lambası
253		1636FF01	I 81.3	BOOL	HP Blower Termistör veya Aşırı Akım Arıza
254		1636CM01	PIW 562	INT	Hp blower Akım izleme
255		1635S11.2	I 16.6	BOOL	Cleaning fan motor STOP Butonu
256		1635S11.1	I 16.5	BOOL	Cleaning fan motor START Butonu
257		1635S106.2	I 14.2	BOOL	1635 Belt Conveyor / Belt tracking right Warning
258		1635S106.1	I 14.1	BOOL	1635 Belt Conveyor / Belt tracking right OK
259		1635S105.2	I 14.0	BOOL	1635 Belt Conveyor / Belt tracking left Warning
260		1635S105.1	I 13.7	BOOL	1635 Belt Conveyor / Belt tracking left OK
261		1635S02.2	I 16.2	BOOL	Cleaning brush STOP Butonu
262		1635S02.1	I 16.1	BOOL	Cleaning brush START Butonu
263		1635S01.2	I 16.4	BOOL	Belt conveyor STOP Butonu
264		1635S01.1	I 16.3	BOOL	Belt conveyor START Butonu
265		1635R11	Q 18.3	BOOL	Cleaning fan motor Çalış
266		1635R02	Q 18.1	BOOL	Cleaning brush Çalış
267		1635R01	Q 18.2	BOOL	Belt conveyor Çalış
268		1635Q11	I 64.7	BOOL	Cleaning fan motor Arıza
269		1635Q02	I 64.3	BOOL	Cleaning brush Arıza
270		1635Q01	I 64.5	BOOL	Belt conveyor Arıza
271		1635KHZ01	I 68.5	BOOL	Belt conveyor İpli Switch Hazır
272		1635K11	I 65.0	BOOL	Cleaning fan motor Çalışıyor
273		1635K02	I 64.4	BOOL	Cleaning brush Çalışıyor
274		1635K01	I 64.6	BOOL	Belt conveyor Çalışıyor
275		1635HZ902	I 13.4	BOOL	1635 Belt Conveyor / Personal protection-R
276		1635HZ901	I 13.3	BOOL	1635 Belt Conveyor / Personal protection-L
277		1635H11.1	Q 4.6	BOOL	Cleaning fan motor Çalışıyor Lambası
278		1635H02.1	Q 4.4	BOOL	Cleaning brush Çalışıyor Lambası
279		1635H0121	I 13.6	BOOL	1635 Belt Conveyor / Bakım Emniyet Şalteri-Hazır
280		1635H01.1	Q 4.5	BOOL	Belt conveyor Çalışıyor Lambası
281		1635B0101	I 13.5	BOOL	1635 Belt Conveyor / Speed OK

Şekil 6.8: Step-7 Sembol Tablosunda 1300 sistemi (yerden besleme) adresleri sembol tablosu.

Sembollerin oluşturulmasının ardından programlanacak olan motorlar sürüş tiplerine göre ayrılarak FC (fonksiyon) yazılır. FC'lerin verileri de DB'lerde (Veri Blokları) toplanır. DB bloklarının oluşturulması için her FC için UDT (Kullanıcı Tanımlı Veri Blokları) oluşturulmuştur. Aşağıda sürüş tiplerine göre blokların oluşturulması ve programının yazılması anlatılmıştır.

6.4.1. Motor kontrol bloklarının hazırlanması

Motor bloklarını hazırlamak için ilk önce motor sürüş tiplerine göre UDT'ler hazırlanır. Motor tiplerine göre hazırlanan her bir UDT, kendi sistemine ait (1300 veya 1600) DB'de toplanarak bütün motorların verileri kısa yoldan veritabanına girilir.

Her motor tipi için ayrı fonksiyon yazılımı yapılarak sistemde kullanılan aynı tip motorlar için bir tek fonksiyon kullanılır. Sadece veri bloklarında bulunan adresler ve kendine ait giriş çıkış adresleri sembolleri ile birlikte fonksiyonun dışında girilerek yazılım her motor için defalarca yapılmamış olur.

Bütün bloklar üzerinde motorun otomatik veya manüel çalıştırılabilmesi için sistem manüel ve sistem otomatik girişleri bulunmaktadır. Sistem otomatik çalışırken motor başlangıç ve duruş gecikme zamanlarına göre sıralı kalkış ve duruş yapmaktadır. Başlangıç ve duruş gecikme süreleri SCADA'dan girilir. Motorun manüel kontrolü ise bakım için veya cipsin, sistem otomatik çalışırken taşınacağı hatta sıkışma oluşması durumunda gerekmektedir. Sistem otomatik ve manüel geçişleri bir yetkilendirme mantığında oluşturulmaktadır. Sistem otomatik çalışırken SCADA, manüelde iken ise motorun yakınında bulunan ve üzerinde motor kontrolü için çalıştırma-durdurma butonları, çalışıyor-arıza lambaları bulunan yerel panel yetkilidir. Sistem otomatik çalışırken yani SCADA yetkili ise yerel panel üzerinden motora acil durdurma butonuna basılması dışında hiçbir şekilde müdahale edilemez. Sistem manüelde ise motor yerel panel üzerinden çalıştırılıp durdurulacağı için bu

işlemler yapılırken sistem otomatik sıralı çalışması engellenir. Yetkilendirmenin amacı iş güvenliğidir.

Blokların üzerinde sistem otomatik çalışırken; otomatik çalıştırma buti, manüelde iken; çalıştırma, durdurma butonları bulunmaktadır. Motor kontrol panelinde motoru termik ve manyetik olarak koruyan motor koruma şalteri yardımcı kontağı üzerinden alınan termik arıza bilgisi, motor kumanda kontaktörü üzerinden veya yardımcı kontağı üzerinden alınan motor çalışıyor bilgisi, motor sargılarını koruyan termistörün bağlandığı termistör rölesi üzerinden alınan termistör arıza bilgisi motorun kumandasında kullanılmaktadırlar. Yerel panel üzerinde bulunan acil durdurma butonu, arıza reset butonu, çalıştırma, durdurma butonu motor kontrolü için kullanılmaktadır. Sahada motorun milinin döndüğünü gözleyen devir bekçisi algılayıcısından alınan motor çalışıyor bilgisi, motor bakıma alındığında yerel olarak 380V AC beslemesini kesen emniyet şalteri ve onun yardımcı kontağından alınan emniyet şalteri arıza bilgisi, gibi birçok giriş bilgileri ile motor kontrolü yapılmaktadır. Ayrıca başka sistemlerden, başka motorlardan veya algılayıcılardan gelen ara kilit (interlok) bilgileri de motorun çalışma şartlarını belirlemektedir. Bu giriş bilgileri fonksiyon içerisinde lojik olarak kullanıldıktan sonra SCADA'ya taşınması için veri bloklarına gönderilir.

Blokların üzerinde, motoru çalıştıracak olan kontaktörü enerjilendiren PLC çıkış adresi, yerel panel üzerindeki motor çalışıyor-arıza lambasının çıkış adresi ve motorun sorunsuz bir şekilde çalıştığını gösteren ve SCADA' ya aktarılan motor çalıştı sinyali bulunmaktadır.

Motor sürüş tiplerine göre motor blokları (FC, DB ve UDT) tezin ekler kısmında yer almaktadır. Ayrıca, motor bloklarının iç yapıları, dış yapıları ve deklarasyon tabloları da verilmektedir.

6.4.2.1300 Grubu (Yerden cips besleme) ve 1600 Grubunun (Yerden toz besleme) kontrolü

Bütün motorların sürüş tipine göre motor bloklarında adresleri girildikten sonra motorlar 1300 ve 1600 olarak gruplandırılır ve her motor kumanda bloğu bu bloklar içerisinde çağrılır.

- “DOL” (Direkt Yol Verme, FC21); “1300_DOL” (1300Grubu Direkt Yol Vermeli Motorlar, FC51), “1600_DOL”(1600Grubu DOL Motorlar, FC56) içerisinde çağrılır.
- “DOL_2D”(Direkt Yol Verme Çift Yönlü, FC22); “1300_DOL_2D” (1300Grubu Direkt Yol Vermeli Çift Yönlü Motorlar, FC52) içerisinde çağrılır.
- “SD” (Yıldız-Üçgen Yol Verme, FC23); “1300_SD” (1300Grubu Yıldız Üçgen Yol Vermeli Motorlar, FC54), “1600_SD” (1600Grubu Yıldız Üçgen Yol Vermeli Motorlar, FC57) içerisinde çağrılır.
- “FC_DOL_2D” (Motor Sürücülü, FC25); “1300_ FC_DOL_2D” (1300Grubu Motor Sürücülü Motorlar, FC53) içerisinde çağrılır.

Bu şekilde 1300 ve 1600 gruplarında bulunan bütün motorlar sürüş tipiklerine göre gruplandırılırlar.

Motor bloklarının oluşturulmasından sonra bu motorların sistem otomatik çalışırken kalkış ve duruşları, birbirlerini etkileyen arıza bağlantıları ve birbirleri arasındaki ara kilitleri (interlokları) da ayrı iki blokta yazılmıştır. “1300 Sistemi” (FC1), “1600 Sistemi” (FC2) bloklarında motor tipiklerinin bulunduğu blokları çağırılmaktadır. 1300 sistemi otomatik start stop “1300_OTOMATIK_START_STOP” (FC5) içerisinde yazılıp FC1 içerisinde çağrılmaktadır. 1600 sistemi otomatik start stop “1600_OTOMATIK_START_STOP (FC6)” içerisinde yazılıp FC2 içerisinde çağrılmaktadır.

Analog giriş çıkış bilgileri ise “ANALOG OKUMA” (FB15) bloğu ve ona bağlı olan “ANALOG KALIBRASYONU” (DB15) bloklarında kalibrasyonu yapılmaktadır. FB15 ise “Cyclic Interrupt” (OB35) içerisinde çağrılmaktadır.

Blokların gruplara ayrılarak çağırılması işleminden sonra FC1 ve FC2 gibi ana grupların oluşturulduğu bloklar yönetici blok olan ve PLC'nin taradığı OB1 içerisinde çağrılır. Şekil 6.9'da 1300Grubu ve 1600Grubu FC bloklarının OB1 içerisinde çağırılması görülmektedir.



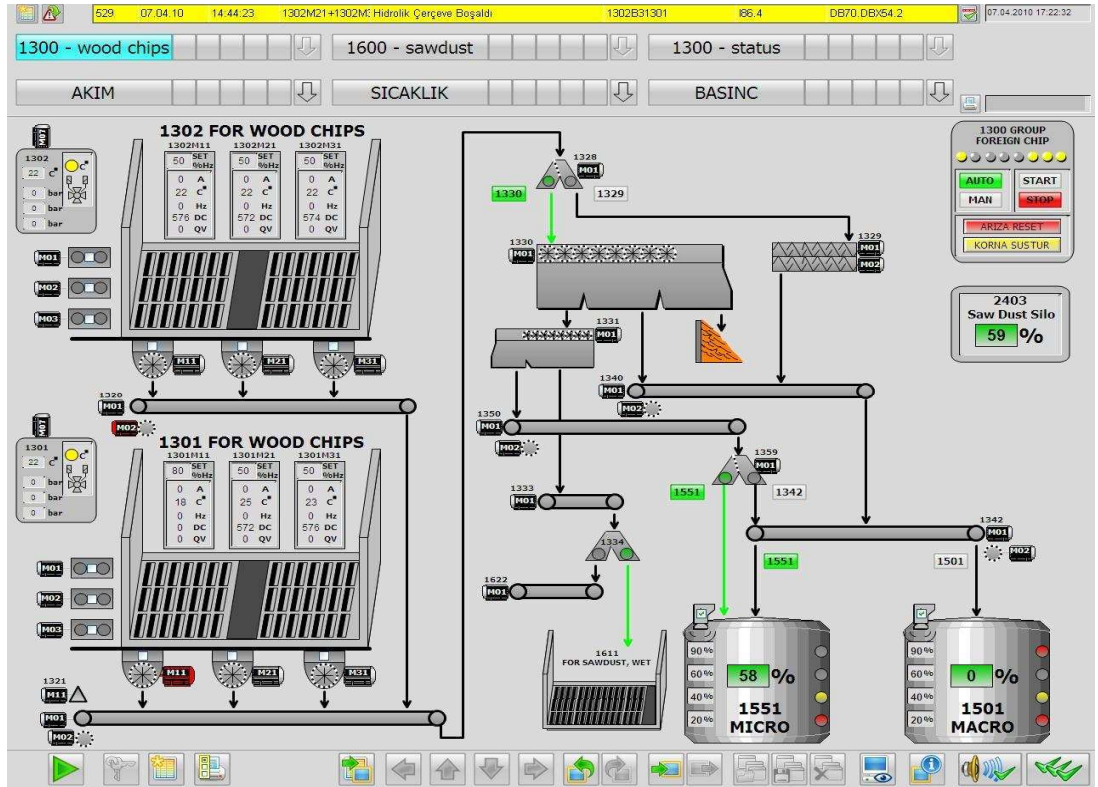
Şekil 6.9: 1300 Grubu ve 1600 Grubunun OB1 yönetici bloğunda çağırılması

6.5. 1300 Sistemi (Yerden Besleme) Siemens WINCC ile SCADA Programının Yazılması

Bu bölümde 1300 Sistemini (Yerden Besleme) oluşturan 1300 Grubu (Yerden Cips Besleme) ve 1600 Grubu (Yerden Toz Besleme) SCADA kontrol sayfaları anlatılmaktadır.

6.5.1. 1300 grubu (yerden cips besleme) kontrol sayfası

1300 Grubunda (Yerden Cips Besleme), 1301 ve 1302 cips silolarındaki cipsler disk eleklerden geçirilerek ayrıştırılmakta ve daha sonra MICRO ve MACRO silolarına gönderilmektedir. Şekil 6.10'da 1300 grubu (yerden cips besleme) SCADA ekranı bulunmaktadır.

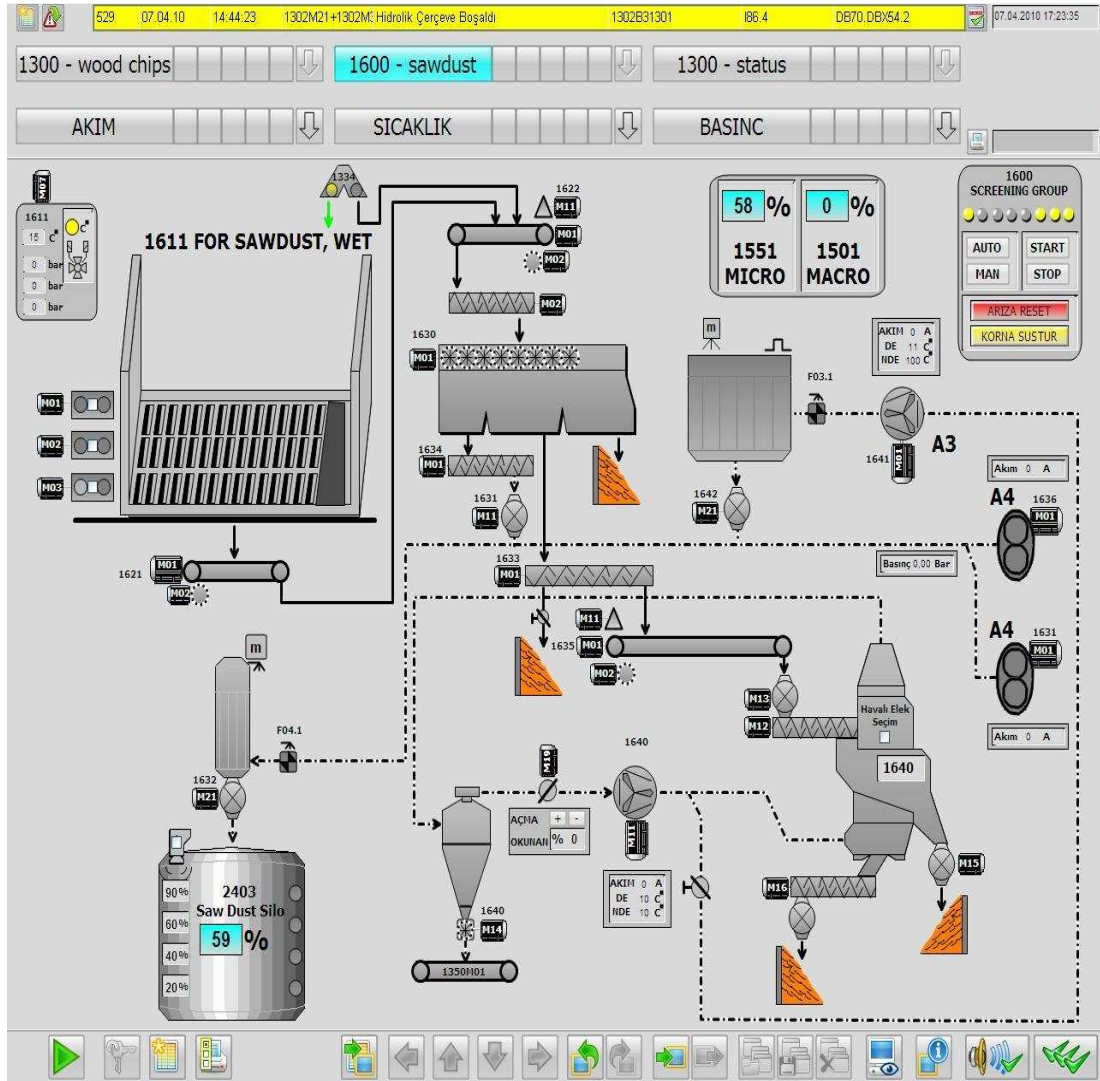


Şekil 6.10: 1300 grubu (yerden cips besleme) kontrol sayfası

1301 ve 1302 cips besleme noktalarında hidrolik motorların kontrol ettiği kayar çerçeveler ile cips ihtiyacına bağlı olarak vidalı taşıyıcılara seçim anahtarları koyulmuştur. Kayar çerçeveler (hidrolik motor grubu) cipsi vidalı taşıyıcılara dökmektedir. Vidalı taşıyıcılar bant motorlarına dökerek bant motorları ile yukarı doğru kuleye çıkartmaktadır. Kulenin üzerindeki klepe motoru, çift yönlü bir motordur. İleri yönünde cipsi inceltmek için disk elekler yönüne dönmektedir. Geri yönünde ise vidalı taşıyıcılara dönerek disk elekleri by-pass etmektedir. Disk elekler cipsin küçültülmesini yetiştiremediklerinde by-pass seçilerek cips, makro silosuna gönderilir. Disk eleklerle gelen cips önce büyük dişli disk elekten geçirilerek ince ve kalın olmak üzere boyutlandırılır. Sonrasında; istenmeyen parçacıklar elekten geçemeyeceğinden dışarı atılır. Birinci disk elekten geçen kalın cips parçacıkları; önce makro konveyör, daha sonra mikro ve makro silolarının üzerinde bulunan silo doldurma konveyörü ile makro silosuna taşınır. Bu silo doldurma konveyörünün asıl görevi makro silosunu doldurmaktır. Mikro silosunun seviyesi %80'e ulaştığında mikro silosuna gönderilen cipsler silo doldurma konveyörünün yönü değiştirilerek makro silosuna boşaltılır. İkinci disk elek, birinci disk elekten gelen ince cipsleri daha da küçültmek için kullanılır. İkinci disk elekten çıkan küçük cipsler 1611 yerden besleme kayar çerçevelerine dökülür. İkinci disk elekten çıkan orta boy cipsler, mikro konveyörüne yüklenir ve mikro silo seviyesi %80'in altında ise mikro silosuna boşaltılır, mikro silo seviyesi %80 ve üzerinde ise siloların üzerindeki çift yönlü konveyöre dökülerek makro silosuna yönlendirilir ve makro silosuna boşaltılır. Silo seviyeleri, üst seviyede yüksek ve çok yüksek, alt seviyede düşük ve çok düşük olmak üzere dört adet seviye sensörü ile dijital olarak, ultrasonik seviye sensörü (radar) ile analog olarak iki şekilde kontrol edilmektedir. Mikro ve makro silolarında üst seviyeler, sistemi otomatik sıralı duruşa geçirmektedir. Sistem, operatör seçimleri ile yarı otomatik olarak kontrol edilebilmektedir. SCADA ekranının üst tarafında bulunan alarmlar ve uyarılar satırı ile sistemden gelen anlık arıza ve uyarı mesajları operatörü uyarmaktadır. Ekranın sağ üst köşesindeki panel ile operatör; sistem otomatik/manüel seçimlerini, otomatik start/stop kontrollerini, arıza onay ve korna susturma işlemlerini yapabilir.

6.5.2.1600 grubu (yerden toz besleme) kontrol sayfası

1600 Grubunda (Yerden Toz Besleme) 1611 toz silosundaki tozlar disk elekten geçirilerek ayrıştırılmaktadır. İnce tozlar toz silosuna, büyük tozlar ise havalı elekten geçirilerek ayrıştırılır ve 1300 Grubundaki micro silosuna gönderilir. Şekil 6.11'de 1600 grubu (yerden toz besleme) SCADA ekranı bulunmaktadır.

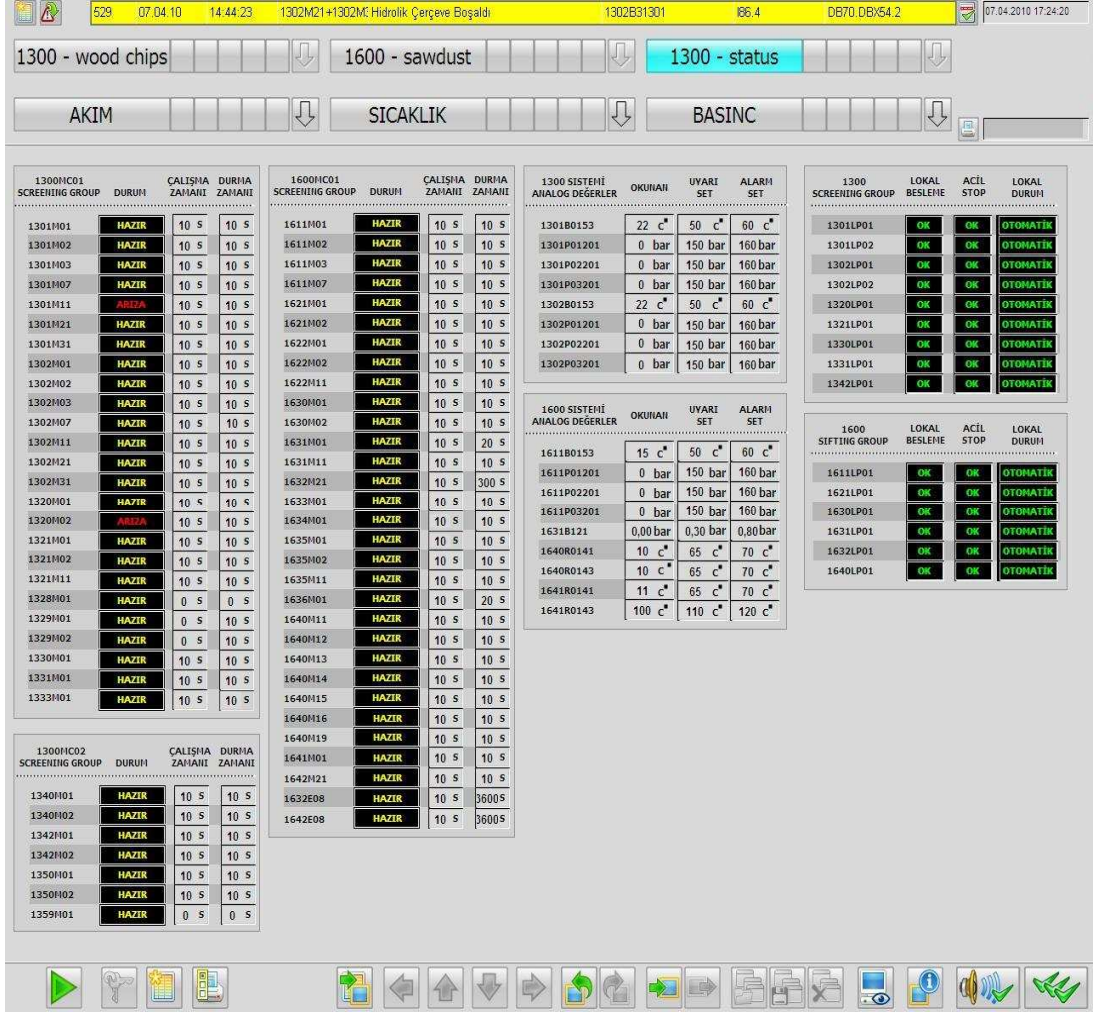


Şekil 6.11: 1600 grubu (yerden toz besleme) kontrol sayfası

1611 yerden toz besleme silosundaki tozlar konveyörlerle taşınarak vidalı taşıyıcıya boşaltılır, vidalı taşıyıcının cipsi yatay hareketi ile cipsler disk eleğin üst kısmına yayılır. Tozun boyutuna göre küçük olanlar iki adet üfleyici motorun bağlı olduğu boru içerisinde toz silosunun üzerindeki filtreye getirilir. Burada 48 adet vana bulunmaktadır. Bu vanalar 10'ar saniye aralıklarla sırayla devreye girip 200msn devrede kalmaktadır. Bu vanalar filtrenin içerisinde birbirine düşey olarak paralel duran bez borularda kalan tozları tabana indirmektedir. Filtreden aşağıya inen cips tozları, kurutucu bölümünde bulunan toz silosunda depolanır. Orta boydakiler havalı eleğe gönderilerek içerisindeki yabancı cisimler yoğunluk farkına göre ayrıştırılır ve tekrar mikro silosuna gönderilmek üzere 1300 sistemi mikro konveyörüne boşaltılır. Havalı elek ve disk elekten ayrıştırılan yabancı cisimler boşaltma kısımlarında toplanır.

6.5.3. Sistem durum sayfası

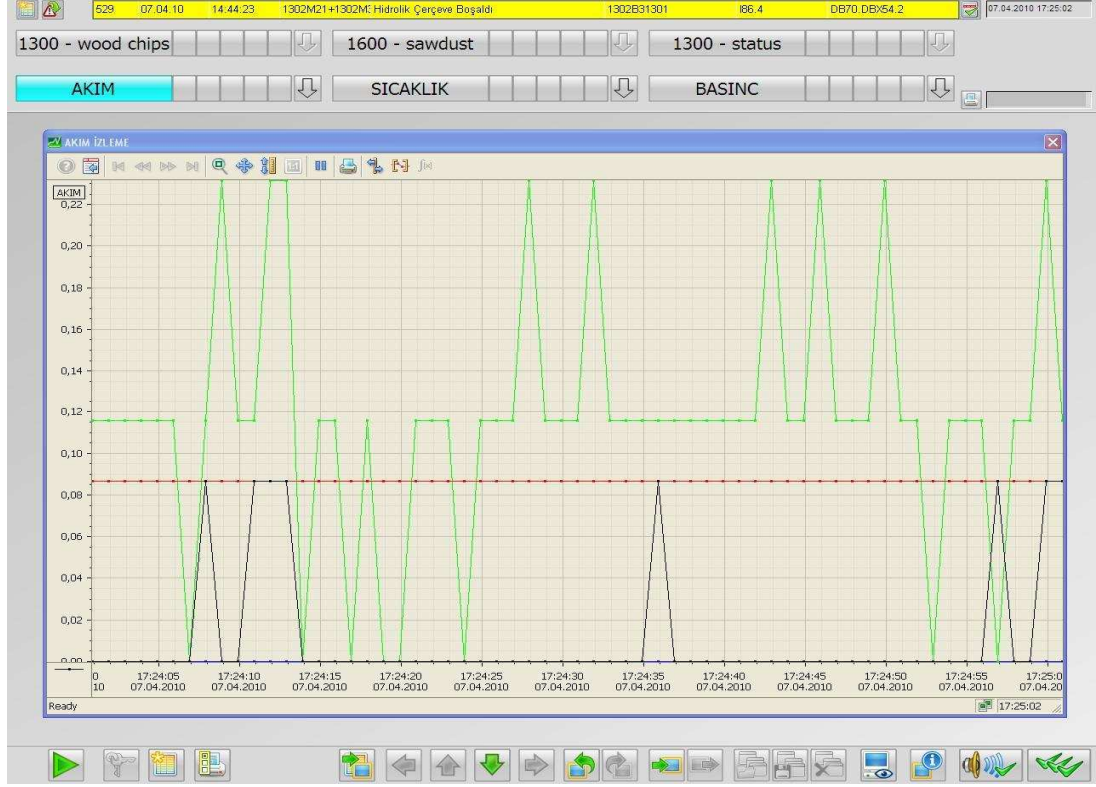
Sistem durum sayfası SCADA ekranında operatör 1300 sisteminde bulunan bütün motorların durumlarını (hazır, arıza, çalışıyor) görebilmekte ve bu motorların sistem otomatikte iken sıralı kalkış ve duruş zamanlarını girmektedir. Hidrolik motorların yağ basınçlarını, hidrolik yağ tankı sıcaklığını, motor rulman sıcaklıklarını ve yıldız üçgen yol verilen motorların akımları gözlenmektedir. Ayrıca sistemdeki bütün yerel panellerin durumları (besleme var, acil stop yok, yerel konum) da bu sayfadan görüntülenmektedir. Şekil 6.12'de sistem durum sayfası görülmektedir.



Şekil 6.12: Sistem durum sayfası

6.5.4. Akım, sıcaklık ve basınç grafik sayfaları

Akım grafikleri sayfasında akım izleme çizelge tablosu yer almaktadır. Bütün motorların akımları aynı grafikte görüntülenebileceği gibi istenirse her motorun akım grafiği ayrı sayfada da görüntülenebilir. Grafiğin dikey ekseninde motor akımı yatay ekseninde tarih ve saat bulunmaktadır. Bu çizelge ile motorların akımları anlık olarak izlenebilmekte ve sürekli olarak kayıt edilmektedir. Arızalanan motorların ne zaman ve hangi akımı çekerek arızalandığı bu grafikler ile tespit edilmektedir. Şekil 6.13'de akım grafikleri sayfası görülmektedir.



Şekil 6.13: Akım grafikleri sayfası

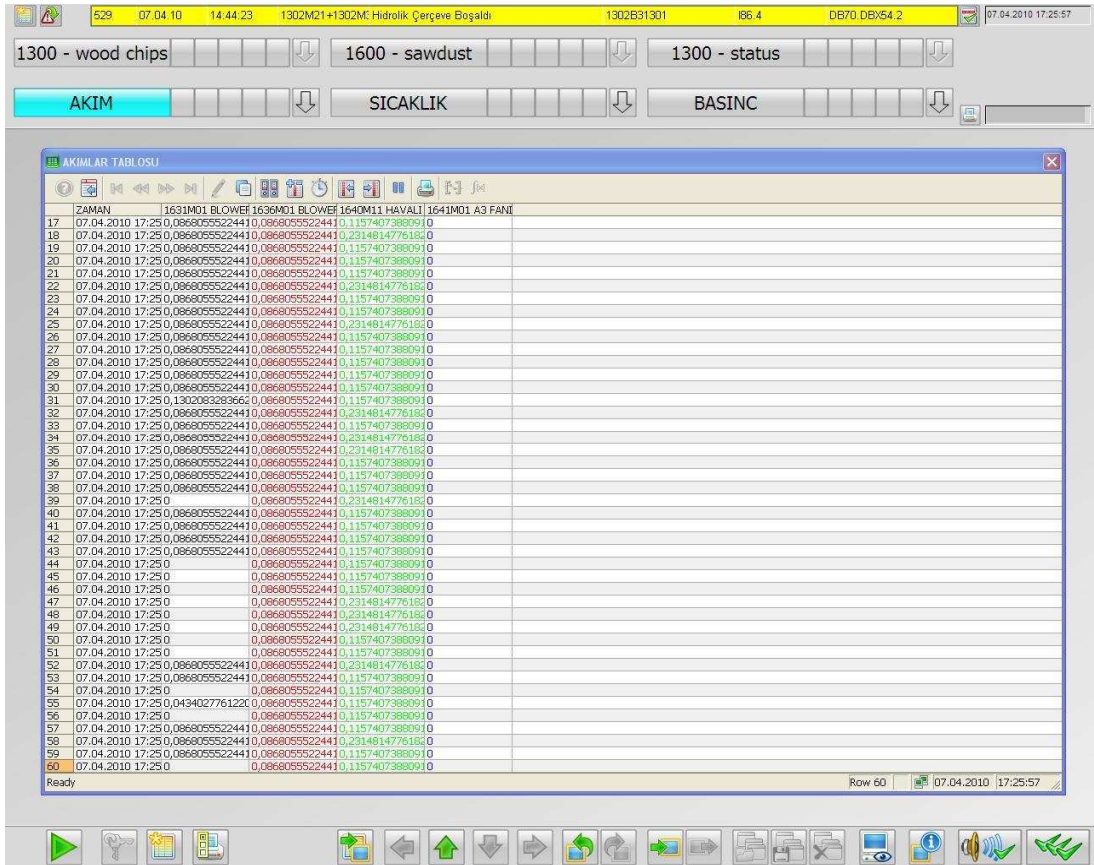
Ayrıca motor rulman sıcaklıkları ve hidrolik yağ tankı sıcaklıkları sıcaklık grafikleri sayfasında gözlemlenmektedir. Bütün sıcaklıklar aynı grafikte görüntülenebildiği gibi her birinin sıcaklığı ayrı sayfada da görüntülenebilmektedir. Grafiğin dikey ekseninde sıcaklık yatay ekseninde tarih ve saat bulunmaktadır. Sıcaklık grafikleri sayfası ile motor rulman sıcaklıkları ve tank sıcaklıkları anlık olarak izlenmekte ve kayıt edilmektedir.

İzlenebilen bir diğer değişken ise basınçtır. Basınç grafikleri sayfasında basınç izleme çizelge tablosu yer almaktadır. Bu çizelgede kayar çerçevelerin ileri geri hareketini yaptıran hidrolik motorlarının hidrolik yağ basınçları ve 1600 sisteminde tozları üfleyerek taşınmasını sağlayan üfleyici (blower) motorlarının akışkan basıncı gözlemlenmektedir. Bütün basınçlar aynı grafikte görüntülenebildiği gibi her birinin basıncı ayrı sayfada da görüntülenebilmektedir. Grafiğin dikey ekseninde basınç

yatay ekseninde tarih ve saat bulunmaktadır. Bu çizelge ile hidrolik motorların basınçları ve üfleme motorlarının hava hattı basıncı anlık olarak izlenmekte ve kayıt edilmektedir.

6.5.5. Akım, sıcaklık ve basınç tablo sayfaları

SCADA ile akım, sıcaklık ve basınç değerleri tablolar halinde görüntülenebilir. Bu tabloda akım izleme çizelge tablosundaki grafikler sayısal değerleri ile tablo halinde görüntülenmektedir. Birinci sütunda tarih ve saat bulunmakta ve diğer sütunlarda her motorun akım değerleri sayısal olarak görüntülenmektedir. Aynı şekilde motorların sıcaklıkları ve yağ basınçları ayrı tablolar halinde görüntülenebilir. Şekil 6.14'deki SCADA sayfasında akım izleme tablosu yer almaktadır.



The screenshot displays a SCADA interface with a top navigation bar and a main data table. The navigation bar includes buttons for '1300 - wood chips', '1600 - sawdust', and '1300 - status', along with tabs for 'AKIM', 'SICAKLIK', and 'BASINC'. The main window, titled 'AKIMLAR TABLOSU', shows a table with the following columns: ZAMAN, 1631M01 BLOWEF 1636M01 BLOWEF 1640M11 HAVALI, and 1641M01 A3 FANI. The table contains 60 rows of data, each representing a time stamp and flow rate values for different motors. The status of each motor is indicated by a small icon in the third column.

ZAMAN	1631M01 BLOWEF 1636M01 BLOWEF 1640M11 HAVALI	1641M01 A3 FANI
17 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
18 07.04.2010 17:25:0	0,08680555224410,08680555224410,23148147761850	
19 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
20 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
21 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
22 07.04.2010 17:25:0	0,08680555224410,08680555224410,23148147761850	
23 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
24 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
25 07.04.2010 17:25:0	0,08680555224410,08680555224410,23148147761850	
26 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
27 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
28 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
29 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
30 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
31 07.04.2010 17:25:0	13020832836650,08680555224410,11574073880910	
32 07.04.2010 17:25:0	0,08680555224410,08680555224410,23148147761850	
33 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
34 07.04.2010 17:25:0	0,08680555224410,08680555224410,23148147761850	
35 07.04.2010 17:25:0	0,08680555224410,08680555224410,23148147761850	
36 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
37 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
38 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
39 07.04.2010 17:25:0	0,08680555224410,08680555224410,23148147761850	
40 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
41 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
42 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
43 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
44 07.04.2010 17:25:0	0,08680555224410,11574073880910	
45 07.04.2010 17:25:0	0,08680555224410,11574073880910	
46 07.04.2010 17:25:0	0,08680555224410,11574073880910	
47 07.04.2010 17:25:0	0,08680555224410,23148147761850	
48 07.04.2010 17:25:0	0,08680555224410,23148147761850	
49 07.04.2010 17:25:0	0,08680555224410,23148147761850	
50 07.04.2010 17:25:0	0,08680555224410,11574073880910	
51 07.04.2010 17:25:0	0,08680555224410,11574073880910	
52 07.04.2010 17:25:0	0,08680555224410,08680555224410,23148147761850	
53 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
54 07.04.2010 17:25:0	0,08680555224410,11574073880910	
55 07.04.2010 17:25:0	04340277612200,08680555224410,11574073880910	
56 07.04.2010 17:25:0	0,08680555224410,11574073880910	
57 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
58 07.04.2010 17:25:0	0,08680555224410,08680555224410,23148147761850	
59 07.04.2010 17:25:0	0,08680555224410,08680555224410,11574073880910	
60 07.04.2010 17:25:0	0,08680555224410,11574073880910	

Şekil 6.14: Akım tabloları sayfası

SONUÇLAR ve ÖNERİLER

Endüstriyel tesislerdeki üretim süreçlerinin PLC ve SCADA sistemleri ile kontrolü kurulum maliyeti açısından klasik kumanda sistemlerine göre daha pahalı olmasına karşın uzun vadede bakıldığında fabrikaların bakım ve arızalardan dolayı duruşlardaki kayıplarını azalttığı için daha ekonomik hale gelmektedir.

Bu çalışmada, PLC ve SCADA sistemleriyle bir MDF presi dış saha besleme hattının kontrolü gerçekleştirilmiştir. Besleme hattı dört ana bölüme ayrılarak her bölüm bir Siemens S7-400 PLC ile kontrol edilmektedir. PLC'ler Siemens Simatic Step-7 programı ile programlanmıştır. PLC'ler kendilerine ait WinCC V7.0 SCADA ara yüz programlarının bulunduğu bilgisayarlar ile Ethernet hattı ile haberleşmektedir. Ayrıca bu dört sistemin PLC'leri, tek bir merkezden kontrol etmek için ana SCADA bilgisayarı ile fiberoptik ağ kurularak haberleştirilmektedir. Siemens S7-400 PLC'ler sahada bulunan ET200M (IM) modülleri kullanılarak profibus ile haberleştirilerek sahadaki veriler toplanmaktadır.

SCADA sayesinde bütün sistem bir bilgisayar ekranından izlenebilmektedir. Sistemdeki motorların akımları, akış hattı basınçları, hidrolik yağ basınçları, hidrolik yağ tankı sıcaklıkları, motor rulman sıcaklıkları ve silo seviyeleri sürekli olarak tablo ve grafikler ile gözlemlenmekte ve kaydedilmektedir. Bu grafiklerin takip edilmesi ile otomasyon sisteminde bulunan motorlar, vanalar ve makineler korunarak sistemin güvenilirliği arttırılmaktadır. SCADA ekranından arızaların sürekli olarak izlenmesi sayesinde oluşabilecek arızalara ve meydana gelmiş arızalara ani müdahale edilebilmektedir. SCADA sistemi sayesinde önceden tanımlanmış veya kullanıcının tanımladığı fiziksel değerlerin geleceğe yönelik süreç kontrolü yapılabilmektedir.

Otomasyonu yapılan MDF presi dış saha besleme sisteminde PLC ve SCADA sistemlerinin kullanılması ile sistemin kontrolü için çalışan personel sayısı ve iş kazaları en aza indirilmiştir. Sistemin üretim kapasitesi arttırılarak, işletim maliyeti azaltılmıştır.

KAYNAKLAR

- [1] ÖZDEMİR E., “Programlanabilir Mantık Denetleyiciler”, Lisans Ders Notu, *Kocaeli Üniversitesi*, (2005)
- [2] EMİNOĞLU, Yavuz, “PLC Programlama ve S7-300”, *Ege Reklam Basım Sanatları Limited Şirketi*, İstanbul, (2006)
- [3] TOPRAK, H., “Bilgisayarlı Veri Toplama ve Kontrol”, (1992)
- [4] “HMI/SCADA, Otomasyon Dergisi”, (Ocak 2010)
- [5] SALİM, S, “Bir Anaerobik Atıksu Arıtma Prosesinin PLC ve SCADA Kullanılarak Otomasyonu”, Yüksek Lisans Tezi, *Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü*, (2006)
- [6] EKE Ş., “SCADA Sisteminin İncelenmesi ve PLC Üzerinden Endüstriyel Denetim”, Proje-B Ödevi, *Kocaeli Üniversitesi*, (Mayıs-2004)
- [7] MİRZAOĞLU, İ., “PLC VE SCADA Kullanarak İrmik Üretim Sisteminin Otomasyonu”, Yüksek Lisans Tezi *Gazi Üniversitesi Fen Bilimleri Enstitüsü* (Ocak-2008)
- [8] “Simatic HMI WINCC Başlangıç el kitabı”
- [9] <http://ahsapboyama.net>
- [10] “Simatic S7-300 Programmable Controller Hardware and Installation Manual” (2002).
- [11] KURTULAN, Salman, “PLC İle Endüstriyel Otomasyon”, *Birsen Yayınevi*, İstanbul, (2001)
- [12] “Siemens Simatic Step-7 Help on ladder logic”
- [13] “Sitep-7 Micro/Win Help”
- [14] KUL, N., “1500 KVA Gücünde 6.3 Kv. Çıkış Gerilimli Generatör Grubu Ve Yüksek Gerilim Kesicilerinin PLC-SCADA İle Uzaktan İzlenmesi”, Yüksek Lisans Tezi, *Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü*, (2009).

[15] KARAÇOR, M., “Cep Telefonu Tabanlı Mobil Scada Otomasyon Sisteminin Geliştirilmesi”, Yüksek Lisans Tezi, *Kocaeli Üniversitesi Fen Bilimleri Enstitüsü*, (Mayıs-2004)

[16] www.profibus.org

[17] ÇALIK, Hüseyin, “PLC ile Analog Verilerin İşlenmesi, Otomasyon Dergisi”, 164-168, (Nisan 2010)

[18] KOÇAK, H., “Enerji Dağıtım ve Otomasyon Sistemi” *İzmir Otomasyon Sempozyumu*, (11-12 Ekim 2001)

[19] www.emo.org.tr

[20] “Drive System Application, Version 2.1”, *Siemens*, (September 2009)

[21] “Micromaster 440 Parametre Listesi, Kullanıcı Dökümantasyonu V2.0”, *Siemens*, (Nisan 2002)

[22] “PROFIBUS Guideline Specification for PROFIBUS Device Description and Device Integration Volume 2: EDDL V 1.1”, (January 2001)

EKLER

EK-A MOTOR TİPLERİNE GÖRE UDT' LER (KULLANICI TANIMLI VERİ BLOĞU)

A1. DOL UDT (Tek Yönlü Direkt Yol Alan Motor Devresi)

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	OTO_START	BOOL	FALSE	
+0.1	CALISMADI_SC	BOOL	FALSE	
+0.2	DEVIRBEKCISI_SC	BOOL	FALSE	
+0.3	TERMIK_SC	BOOL	FALSE	
+0.4	TERMISTOR_SC	BOOL	FALSE	
+0.5	EMNSALTERI_SC	BOOL	FALSE	
+0.6	M_CALISTI_SC	BOOL	FALSE	
+0.7	FP1	BOOL	FALSE	
+1.0	FP2	BOOL	FALSE	
+1.1	FP3	BOOL	FALSE	
+1.2	FP4	BOOL	FALSE	
+1.3	ARIZA	BOOL	FALSE	
+2.0	SC_CALISTIRMA_SET_ZAMANI	INT	0	
+4.0	SC_DURDURMA_SET_ZAMANI	INT	0	
+6.0	CALISTIRMA_ZAMANI	INT	0	
+8.0	DURDURMA_ZAMANI	INT	0	
+10.0	GECIKME1	INT	0	
+12.0	GECIKME2	INT	0	
=14.0		END_STRUCT		

Şekil A1: DOL UDT bloğu (tek yönlü direkt yol alan motor devresi)

A2. DOL-2D UDT (Çift Yönlü Direkt Yol Alan Motor Devresi)

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	OTO_ILERI_START	BOOL	FALSE	
+0.1	OTO_GERI_START	BOOL	FALSE	
+0.2	CALISMADI_SC	BOOL	FALSE	
+0.3	DEVIRBEKCISI_SC	BOOL	FALSE	
+0.4	TERMIK_SC	BOOL	FALSE	
+0.5	TERMISTOR_SC	BOOL	FALSE	
+0.6	EMNSALTERI_SC	BOOL	FALSE	
+0.7	M_ILERICALISTI_SC	BOOL	FALSE	
+1.0	M_GERICALISTI_SC	BOOL	FALSE	
+1.1	FP1	BOOL	FALSE	
+1.2	FP2	BOOL	FALSE	
+1.3	FP3	BOOL	FALSE	
+1.4	FP4	BOOL	FALSE	
+1.5	FP11	BOOL	FALSE	
+1.6	ARIZA	BOOL	FALSE	
+2.0	SC_CALISTIRMA_SET_ZAMANI	INT	0	
+4.0	SC_DURDURMA_SET_ZAMANI	INT	0	
+6.0	CALISTIRMA_ZAMANI	INT	0	
+8.0	DURDURMA_ZAMANI	INT	0	
+10.0	GECIKME1	INT	0	
+12.0	GECIKME2	INT	0	
+14.0	GECIKME11	INT	0	
=16.0		END_STRUCT		

Şekil A2: DOL-2D UDT bloğu (çift yönlü direkt yol alan motor devresi)

A3. SD UDT (Tek Yönlü Yıldız Üçgen Yol Alan Motor Devresi)

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	OTO_START	BOOL	FALSE	
+0.1	CALISMADI_SC	BOOL	FALSE	
+0.2	DEVIRBEKCISI_SC	BOOL	FALSE	
+0.3	TERMIK_SC	BOOL	FALSE	
+0.4	TERMISTOR_SC	BOOL	FALSE	
+0.5	EMNSALTERI_SC	BOOL	FALSE	
+0.6	M_CALISTI_SC	BOOL	FALSE	
+0.7	FP1	BOOL	FALSE	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
+1.0	FP2	BOOL	FALSE	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
+1.1	FP3	BOOL	FALSE	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
+1.2	FP4	BOOL	FALSE	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
+1.3	ARIZA	BOOL	FALSE	
+2.0	SC_CALISTIRMA_SET_ZAMANI	INT	0	
+4.0	SC_DURDURMA_SET_ZAMANI	INT	0	
+6.0	CALISTIRMA_ZAMANI	INT	0	
+8.0	DURDURMA_ZAMANI	INT	0	
+10.0	GECIKME1	INT	0	
+12.0	GECIKME2	INT	0	
=14.0		END_STRUCT		

Şekil A3: SD UDT bloğu (tek yönlü yıldız üçgen yol alan motor devresi)

A4. FC UDT (Çift Yönlü Motor Sürücüsü İle Yol Alan Motor Devresi)

+0.0	OTO_ILERI_START	BOOL	FALSE	
+0.1	OTO_GERI_START	BOOL	FALSE	
+0.2	CALISMADI_SC	BOOL	FALSE	
+0.3	DEVIRBENCISI_SC	BOOL	FALSE	
+0.4	TERMİK_SC	BOOL	FALSE	
+0.5	TERMİSTOR_SC	BOOL	FALSE	
+0.6	EMNSALTERI_SC	BOOL	FALSE	
+0.7	M_ILERICALISTI_SC	BOOL	FALSE	
+1.0	M_GERICALISTI_SC	BOOL	FALSE	
+1.1	FP1	BOOL	FALSE	
+1.2	FP2	BOOL	FALSE	
+1.3	FP3	BOOL	FALSE	
+1.4	FP4	BOOL	FALSE	
+1.5	FP11	BOOL	FALSE	
+1.6	DRV_ILERICALIS	BOOL	FALSE	MOTOR ILERI ÇALIŞ
+1.7	DRV_GERICALIS	BOOL	FALSE	MOTOR GERI ÇALIŞ
+2.0	ARIZA	BOOL	FALSE	ARIZA
+2.1	M_CALIS	BOOL	FALSE	
+4.0	SC_CALISTIRMA_SET_ZAMANI	INT	0	
+6.0	SC_DURDURMA_SET_ZAMANI	INT	0	
+8.0	CALISTIRMA_ZAMANI	INT	0	
+10.0	DURDURMA_ZAMANI	INT	0	
+12.0	GEÇİME1	INT	0	
+14.0	GEÇİME2	INT	0	
+16.0	GEÇİME11	INT	0	
+18.0	SC_FREKANS_SET	REAL	0.000000e+000	
+22.0	STATUS_1	WORD	W#16#0	Status Word 1
+24.0	CONTROL	WORD	W#16#0	
+26.0	READY	BOOL	FALSE	Frequency Converter ready
+26.1	FAULT	BOOL	FALSE	Frequency Converter fault
+26.2	N_0	BOOL	FALSE	Actual speed > 0
+26.3	M_CALLISTI_SC	BOOL	FALSE	MOTOR ÇALIŞTI SKADA SINYALI
+28.0	STATUS	WORD	W#16#0	
+30.0	FREKANS_SC	REAL	0.000000e+000	DRIVERDAN GELEN SKALALANMIŞ ANLIK ÇIKIŞ FREKANS DEĞERİ
+34.0	AKIM_SC	REAL	0.000000e+000	DRIVERDAN GELEN SKALALANMIŞ ANLIK ÇIKIŞ AKIM DEĞERİ
+38.0	DC_BARA_SC	REAL	0.000000e+000	DRIVERDAN GELEN SKALALANMIŞ DC BARA GERİLİMİ DEĞERİ
+42.0	FC_SICAKLIK_SC	REAL	0.000000e+000	DRIVERDAN GELEN SKALALANMIŞ FC SICAKLIK DEĞERİ
+46.0	AC_V_CIKIS_SC	REAL	0.000000e+000	DRIVERDAN GELEN SKALALANMIŞ AC ÇIKIŞ GERİLİMİ DEĞERİ
=50.0		END_STRUCT		

Şekil A4: FC UDT bloğu (çift yönlü motor sürücüsü ile yol alan motor devresi)

EK-B MOTOR TİPLERİNE GÖRE UDT'LERİN (KULLANICI TANIMLI VERİ BLOĞU) DB'LERDE TOPLANMASI

B1. 1300MC01 Paneli Motorlarının DB' ye Girilmesi

Address	Name	Type	Init	Comment
0.0		STRUCT		
+0.0	M_1301M01	"DOL_UDT"		Hydraulic pump
+14.0	M_1301M02	"DOL_UDT"		Hydraulic pump
+28.0	M_1301M03	"DOL_UDT"		Hydraulic pump
+42.0	M_1301M11	"FC_MICROMASTR_DOL_2D_UDT"		Discharge screw 1
+92.0	M_1301M1101	"DOL_UDT"		Discharge screw 1 fan
+106.0	M_1301M21	"FC_MICROMASTR_DOL_2D_UDT"		Discharge screw 2
+156.0	M_1301M2101	"DOL_UDT"		Discharge screw 2 fan
+170.0	M_1301M31	"FC_MICROMASTR_DOL_2D_UDT"		Discharge screw 3
+220.0	M_1301M3101	"DOL_UDT"		Discharge screw 3 fan
+234.0	M_1301M07	"DOL_UDT"		Cooling circuit pump
+248.0	M_1302M01	"DOL_UDT"		Hydraulic pump
+262.0	M_1302M02	"DOL_UDT"		Hydraulic pump
+276.0	M_1302M03	"DOL_UDT"		Hydraulic pump
+290.0	M_1302M07	"DOL_UDT"		Cooling circuit pump
+304.0	M_1302M11	"FC_MICROMASTR_DOL_2D_UDT"		Discharge screw 1
+354.0	M_1302M1101	"DOL_UDT"		Discharge screw 1 fan
+368.0	M_1302M21	"FC_MICROMASTR_DOL_2D_UDT"		Discharge screw 2
+418.0	M_1302M2101	"DOL_UDT"		Discharge screw 2 fan
+432.0	M_1302M31	"FC_MICROMASTR_DOL_2D_UDT"		Discharge screw 3
+482.0	M_1302M3101	"DOL_UDT"		Discharge screw 3 fan
+496.0	M_1320M02	"DOL_UDT"		Cleaning brush
+510.0	M_1320M01	"DOL_UDT"		Belt conveyor
+524.0	M_1321M02	"DOL_UDT"		Cleaning brush
+538.0	M_1321M11	"DOL_UDT"		Cleaning fan
+552.0	M_1321M01	"SD_UDT"		Belt conveyor
+566.0	M_1325M01	"DOL_UDT"		Belt conveyor
+580.0	M_1326M01	"DOL_UDT"		Belt conveyor
+594.0	M_1328M01	"KLEPE_DOL_2D_UDT"		Diverter flap

Şekil B1: 1300MC01 paneli motorları veri bloğu

B2. 1300MC02 Paneli Motorlarının DB' ye Girilmesi

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	M_1340M02	"DOL_UDT"		Cleaning brush
+14.0	M_1340M01	"DOL_UDT"		Belt conveyor
+28.0	M_1342M02	"DOL_UDT"		Cleaning brush
+42.0	M_1342M01	"DOL_2D_UDT"		Belt conveyor
+58.0	M_1345M01	"DOL_UDT"		Belt conveyor
+72.0	M_1350M02	"DOL_UDT"		Cleaning brush
+86.0	M_1350M01	"DOL_UDT"		Belt conveyor
+100.0	M_1355M01	"DOL_UDT"		Belt conveyor
+114.0	M_1359M01	"KLEPE_DOL_2D_UDT"		Diverter flap
=132.0		END_STRUCT		

Şekil B2: 1300MC02 paneli motorları veri bloğu

B3. 1600MC01 Paneli Motorlarının DB' ye Girilmesi

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	M_1611M01	"DOL_UDT"		Hydraulic pump
+14.0	M_1611M02	"DOL_UDT"		Hydraulic pump
+28.0	M_1611M03	"DOL_UDT"		Hydraulic pump
+42.0	M_1611M07	"DOL_UDT"		Cooling circuit pump
+56.0	M_1621M02	"DOL_UDT"		Cleaning brush
+70.0	M_1621M01	"DOL_UDT"		Belt conveyor
+84.0	M_1622M11	"DOL_UDT"		Cleaning fan motor
+98.0	M_1622M02	"DOL_UDT"		Cleaning brush
+112.0	M_1622M01	"DOL_UDT"		Belt conveyor
+126.0	M_162SM01	"DOL_UDT"		Belt conveyor
+140.0	M_1630M02	"DOL_UDT"		Feeding screw
+154.0	M_1630M01	"DOL_UDT"		Roller screen
+168.0	M_1631M01	"SD_UDT"		Hp blower
+182.0	M_1631M11	"DOL_UDT"		Rotary valve
+196.0	M_1632M21	"DOL_UDT"		Rotary valve
+210.0	M_1633M01	"DOL_UDT"		Screw conveyor
+224.0	M_1634M01	"DOL_UDT"		Screw conveyor
+238.0	M_1635M02	"DOL_UDT"		Cleaning brush
+252.0	M_1635M01	"DOL_UDT"		Belt conveyor
+266.0	M_1635M11	"DOL_UDT"		Cleaning fan motor
+280.0	M_1640M11	"SD_UDT"		Radial fan
+294.0	M_1640M12	"DOL_UDT"		Screw conveyor
+308.0	M_1640M13	"DOL_UDT"		Rotary valve
+322.0	M_1640M14	"DOL_UDT"		Rotary valve
+336.0	M_1640M15	"DOL_UDT"		Rotary valve
+350.0	M_1640M16	"DOL_UDT"		Screw conveyor
+364.0	M_1641M01	"DOL_UDT"		Radial fan A3
+378.0	M_1642M21	"DOL_UDT"		Rotary valve
+392.0	M_1636M01	"SD_UDT"		Hp blower
+406.0	M_1640M19	"KONTROL_FLAP_UDT"		Kontrol Flap
+430.0	F_1632E08	"FILTRE_UDT"		FILTRE
+440.0	F_1642E08	"FILTRE_UDT"		FILTRE

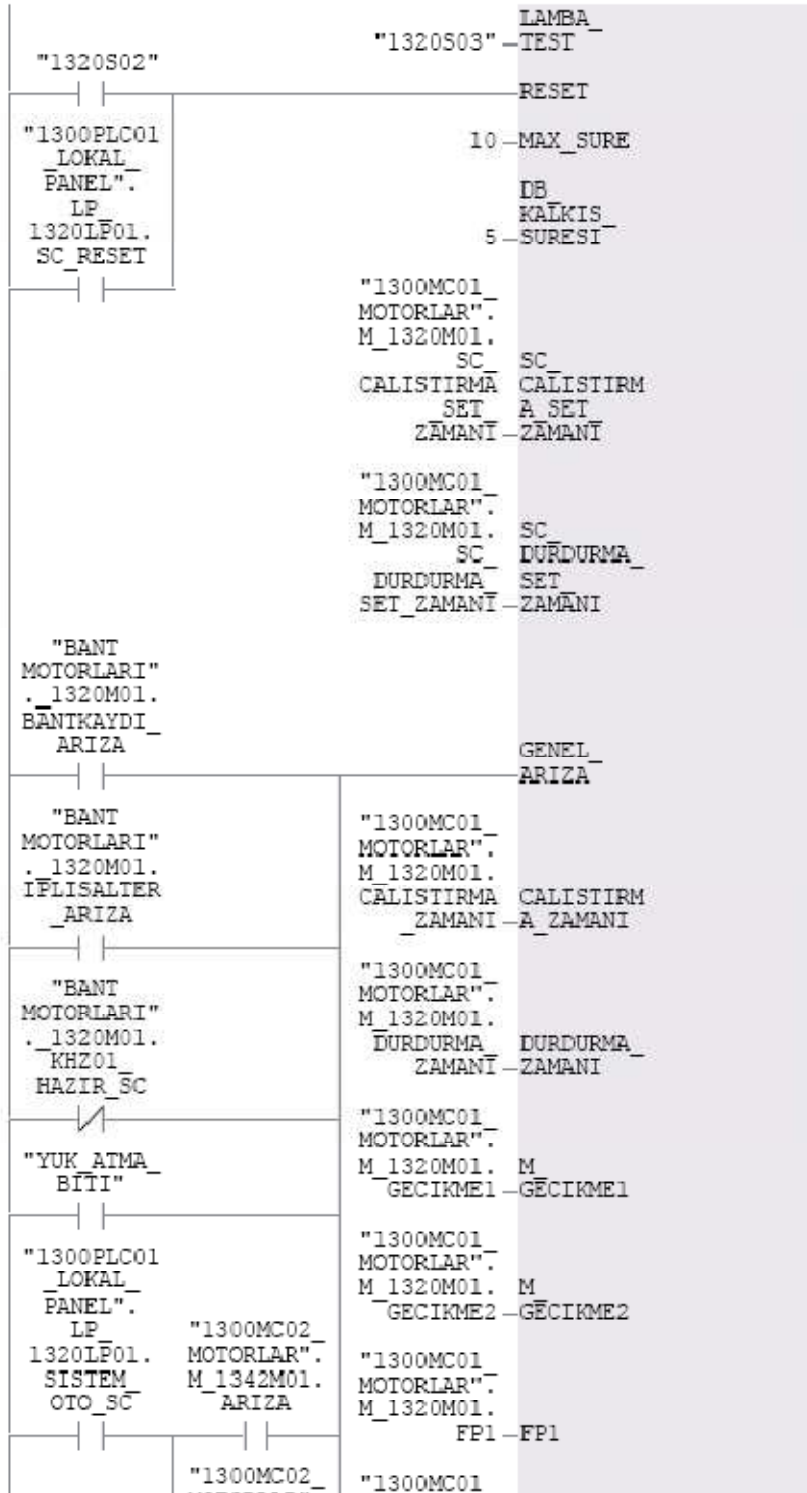
Şekil B3: 1600MC01 paneli motorları veri bloğu

EK-C MOTOR TIPLERİNE GÖRE FC'LER (FONKSİYONLAR)

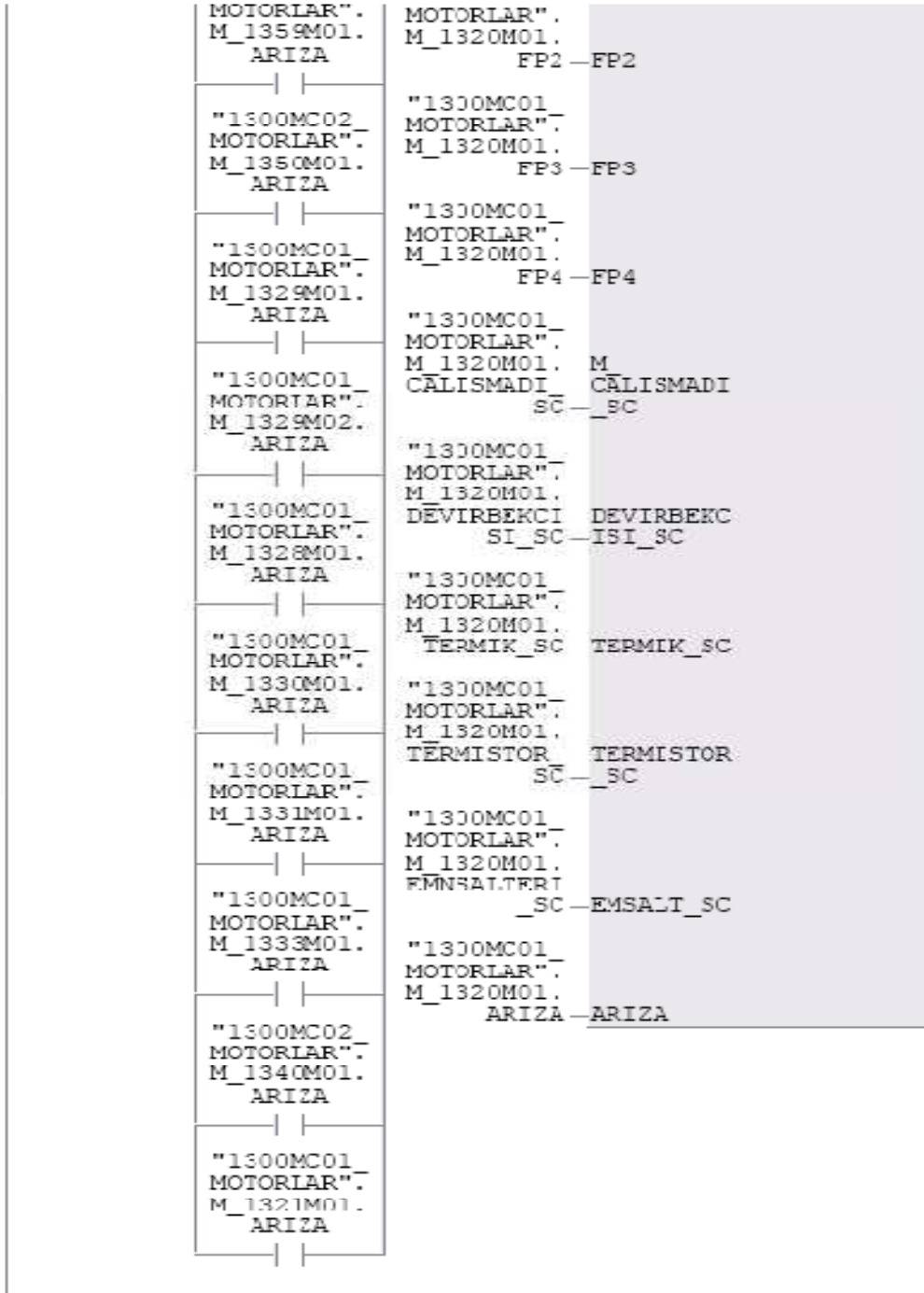
C1. DOL FC (Tek Yönlü Direkt Yol Alan Motor Devresi Bloğu)

Network: 2	1320M01 11KW DOL Belt conveyor Motoru Kontrolü		
	"DOL"		
	EN	ENO	
"1300PLC01 LOKAL PANEL". LP 1320LP01. SISTEM MAN_SC	SISTEM_ MAN	M_CALIS	"1320R01"
"1300PLC01 LOKAL PANEL". LP 1320LP01. SISTEM OTO_SC	SISTEM_ OTO	CALISIYOR ARIZA LAMBASI	"1320H01. 1"
"1300MC01 MOTORLAR". M_1320M01. OTO_START	OTO_START	M CALISTI SC	"1300MC01 MOTORLAR". M_1320M01. M_CALISTI_ SC
"1320S01. 1"	BUTON_ START		
"1320S01. 2"	BUTON_ STOP		
"1300PLC01 LOKAL PANEL". LP 1300LP01. SC1_SC	ACILSTOP		
"1320Q01"	TERMIK		
"1320R0101 "	DEVIRREKC ISI		
"1320H0121 "	EMNIYET SALTERI		
"ALLWAYS OFF"	TERMISTOR		
"1320K01"	M KONTAKTOR CEKILI		
"ALLWAYS ON"	INTERLOCK OK		

C1. (Devam) DOL FC (Tek Yönlü Direkt Yol Alan Motor Devresi Bloğu)



C1. (Devam) DOL FC (Tek Yönlü Direkt Yol Alan Motor Devresi Bloğu)



Şekil C1: DOL FC bloğu (tek yönlü direkt yol alan motor devresi) dış yapısı

C1. (Devam) DOL FC (Tek Yönlü Direkt Yol Alan Motor Devresi Bloğu)

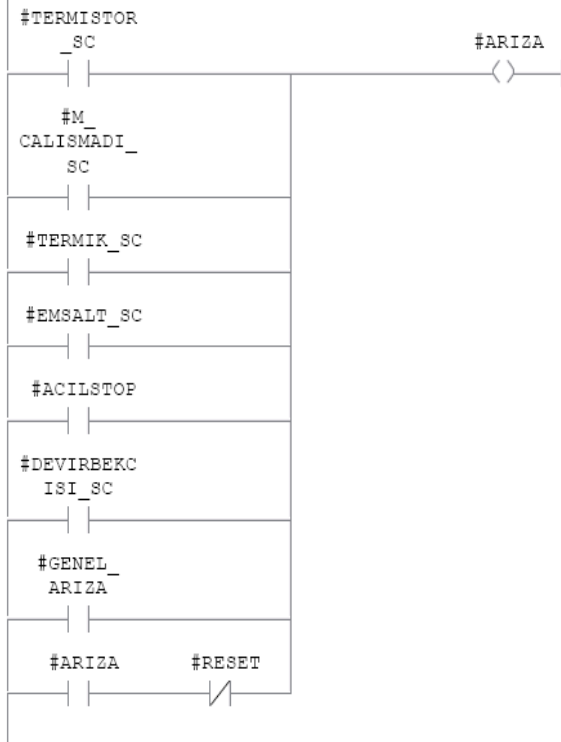
Name	Data Type	Address	Comment
IN		0.0	
SISTEM_MAN	Bool	0.0	=1 MANUAL (LP BLOĞUNUN LOJİK SONUCU)
SISTEM_OTO	Bool	0.1	=1 OTOMATİK (LP BLOĞUNUN LOJİK SONUCU)
OTO_START	Bool	0.2	=1 OTOSTART
BUTON_START	Bool	0.3	=1 PUSH START
BUTON_STOP	Bool	0.4	=0 PUSH STOP
ACILSTOP	Bool	0.5	=0 ACILSTOP OKEY (=1 ARIZA DURUMU)
TERMİK	Bool	0.6	=1 TERMİK ARIZA
DEVİRBEKÇİSİ	Bool	0.7	=1 DEVİRBEKÇİSİ ARIZA
EMNİYET_ŞALTERİ	Bool	1.0	=1 EMNİYET ŞALTERİ ARIZA
TERMİSTÖR	Bool	1.1	=1 TERMİSTÖR ARIZA
M_KONTAKTÖR_CEKİLİ	Bool	1.2	=1 KONTAKTÖR ÇEKTİ BİLGİSİ
INTERLOCK_OK	Bool	1.3	=1 OLMASI GEREKEN GENEL KOŞULLAR TAMAM
LAMBA_TEST	Bool	1.4	=1 PUSH LAMBA TEST
RESET	Bool	1.5	=1 PUSH RESET
MAX_SURE	Int	2.0	KONTAKTÖR ÇALIŞMA SİNYALİNİN GELMESİ GEREKEN SURE
DE_KALKIS_SURESI	Int	4.0	DEVİRBEKÇİSİ SİNYALİNİN MOTOR KALKTIKTAN SONRA GELMESİ GEREKEN SURE
SC_CALISTIRMA_SET_ZAMANI	Int	6.0	OTOMATİK ÇALIŞTIRMADA SET EDİLEN KALKIŞ SURESI
SC_DURDURMA_SET_ZAMANI	Int	8.0	OTOMATİK ÇALIŞTIRMADA SET EDİLEN DURUŞ SURESI
GENEL_ARIZA	Bool	10.0	GENEL ARIZA
OUT		0.0	
M_CALIS	Bool	12.0	MOTOR ÇALIŞ
CALISIYOR_ARIZA_LAMBASI	Bool	12.1	ARIZA ÇALIŞIYOR LAMBASI (=1 MOTOR ÇALIŞIYOR, =1-0 BİNG MOTOR ARIZA)
M_CALISTI_SC	Bool	12.2	MOTOR KESİN ÇALIŞTI SKADA SİNYALİ

Name	Data Type	Address	Comment
IN_OUT		0.0	
CALISTIRMA_ZAMANI	Int	14.0	OTOMATİK ÇALIŞMADA MOTOR START GECİKME ZAMANI
DURDURMA_ZAMANI	Int	16.0	OTOMATİK ÇALIŞMADA MOTOR STOP GECİKME ZAMANI
M_GECIKME1	Int	18.0	START GELDI FAKAT KONTAKTÖR ÇEKMEDİ DURUMU İÇİN GEÇEN SURE
M_GECIKME2	Int	20.0	START GELDI FAKAT DEVİRBEKÇİSİ GELMEDİ DURUMU İÇİN GEÇEN SURE
FP1	Bool	22.0	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
FP2	Bool	22.1	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
FP3	Bool	22.2	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
FP4	Bool	22.3	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
M_CALISMADI_SC	Bool	22.4	MOTOR ÇALIŞMADI SKADA SİNYALİ
DEVİRBEKÇİSİ_SC	Bool	22.5	DEVİRBEKÇİSİ SKADA SİNYALİ
TERMİK_SC	Bool	22.6	TERMİK SKADA SİNYALİ
TERMİSTÖR_SC	Bool	22.7	TERMİSTÖR SKADA SİNYALİ
EMSALT_SC	Bool	23.0	EMNİYET ŞALTERİ SKADA SİNYALİ
ARIZA	Bool	23.1	TÜM ARIZALARIN GENEL SONUCU
TEMP		0.0	
OUT_START	Bool	0.0	OTOMATİK START GECİKMESİ
OUT_STOP	Bool	0.1	OTOMATİK STOP GECİKMESİ
RETURN		0.0	
RET_VAL		0.0	

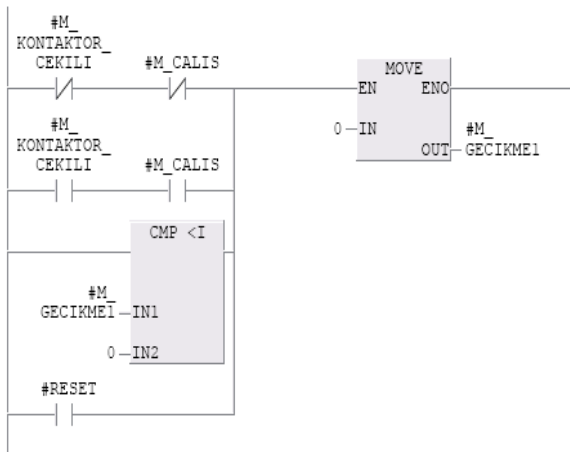
Şekil C2: DOL FC bloğu (tek yönlü direkt yol alan motor devresi) deklarasyon tablosu

C1. (Devam) DOL FC (Tek Yönlü Direkt Yol Alan Motor Devresi Bloğu)

Network: 1 ARIZA



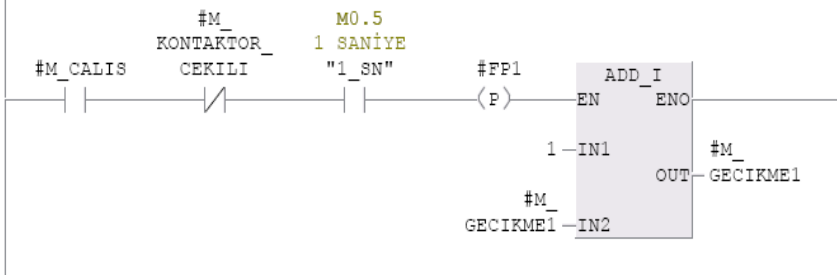
Network: 2 MOTOR GECİKME-1 SAYACININ SIFIRLANMASI



C1. (Devam) DOL FC (Tek Yönlü Direkt Yol Alan Motor Devresi Bloğu)

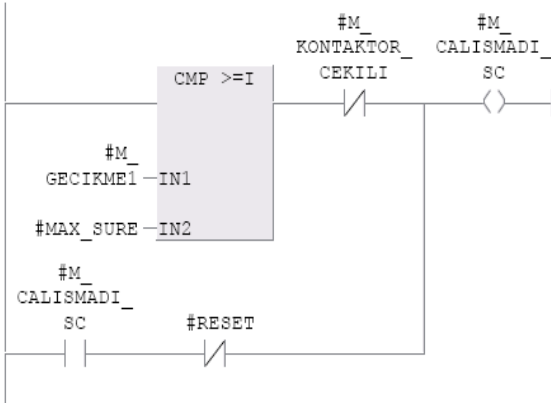
Network: 3 MOTOR GECİKME-1 SAYACI

MOTOR ÇALIŞMADI BİLGİSİNİ KONTAKTÖR'DEN ALIP SÜREYİ SAYMAYA BAŞLARIZ.

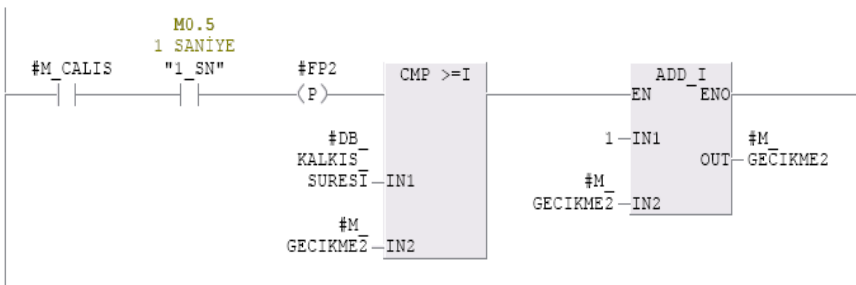


Network: 4

MOTOR ÇALIŞMADI BİLGİSİ KONTAKTÖRDEN İSE = MOTOR ÇALIŞMADI HATASI

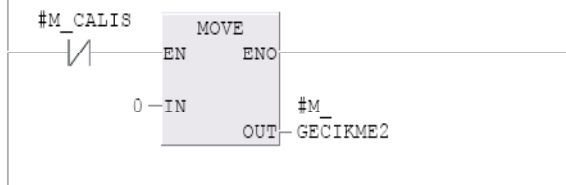


Network: 5 MOTOR GECİKME-2 SAYACI



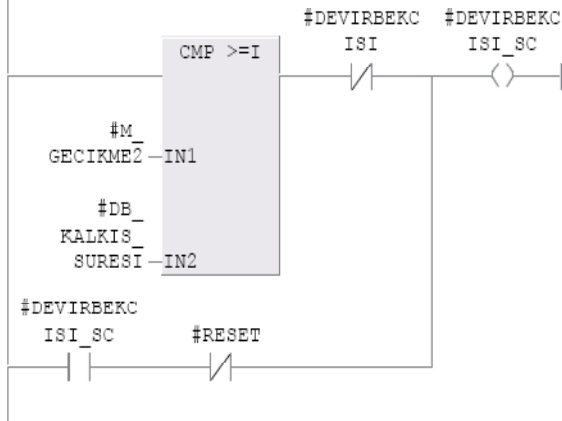
C1. (Devam) DOL FC (Tek Yönlü Direkt Yol Alan Motor Devresi Bloğu)

Network: 6 MOTOR GECİKME-2 SAYACININ SIFIRLANMASI



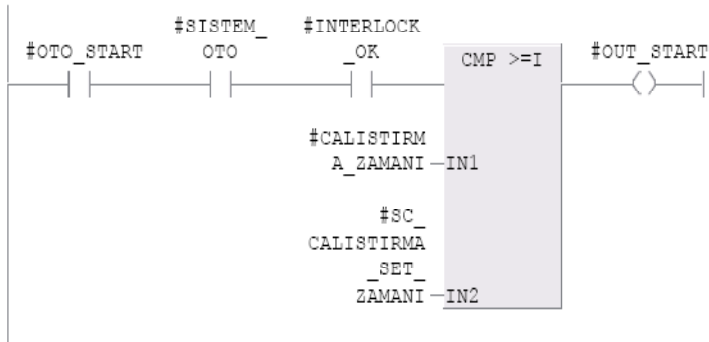
Network: 7

KONTAKTÖR BİLGİSİ VAR DEVİRBEKCİSİ YOK İSE DEVİR BEKÇİSİ ARIZA



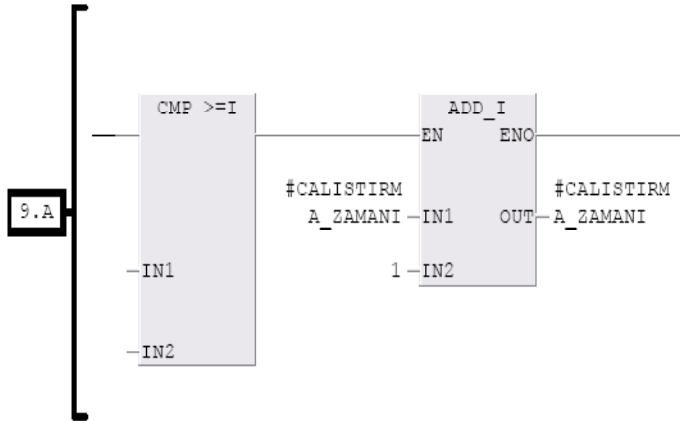
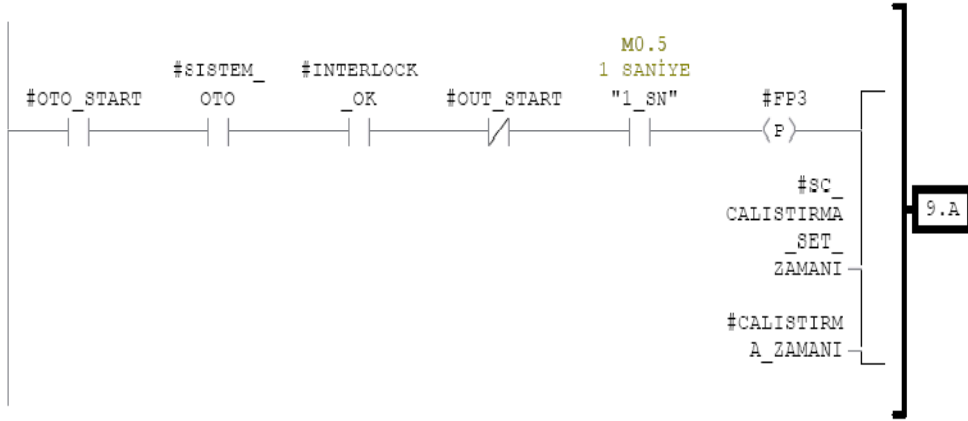
Network: 8 START GECİKMESİ

SİSTEM OTOMATİKTE İKEN START GECİKME ZAMANI

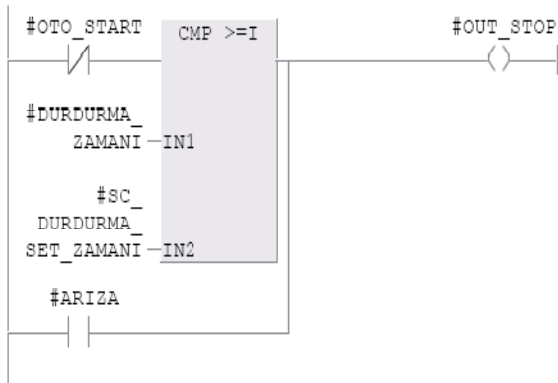


C1. (Devam) DOL FC (Tek Yönlü Direkt Yol Alan Motor Devresi Bloğu)

Network: 9 START GECİKMESİ İÇİN SAYMA

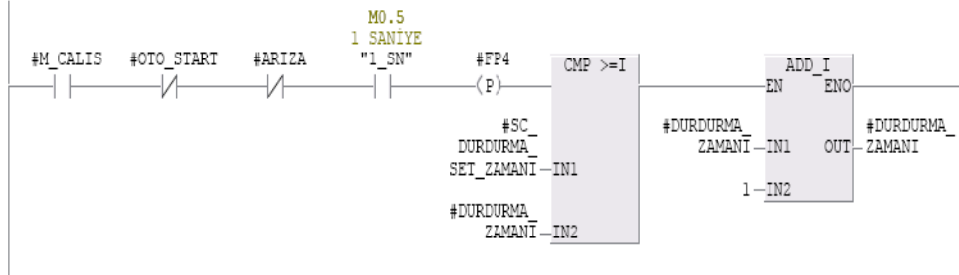


Network: 10 STOP GECİKMESİ

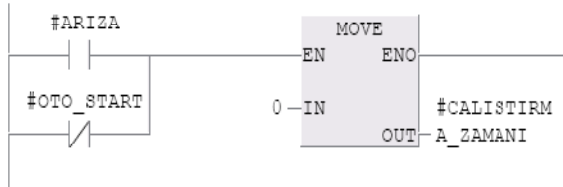


C1. (Devam) DOL FC (Tek Yönlü Direkt Yol Alan Motor Devresi Bloğu)

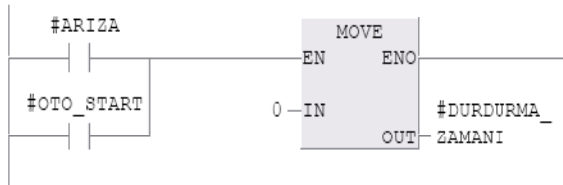
Network: 11 STOP GECİKMESİ İÇİN SAYMA



Network: 12 START ZAMANINI SIFIRLAMA

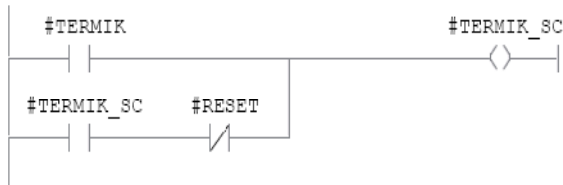


Network: 13 STOP ZAMANINI SIFIRLAMA



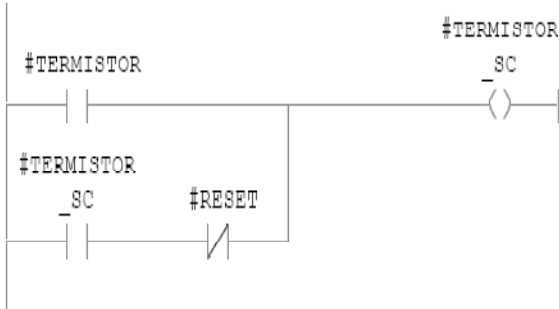
Network: 14 1 SANIYE

TERMİK ARIZA

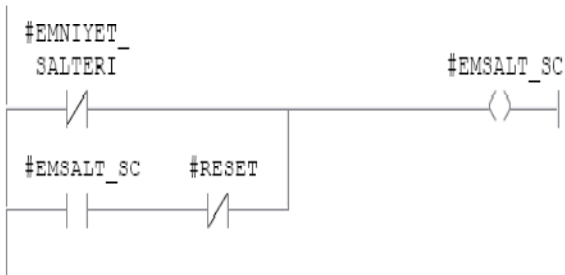


C1. (Devam) DOL FC (Tek Yönlü Direkt Yol Alan Motor Devresi Bloğu)

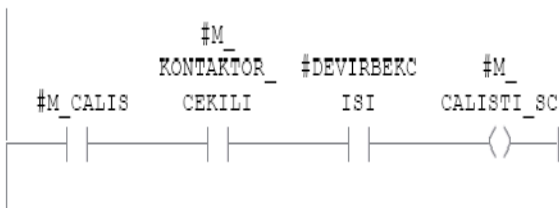
Network: 15
TERMİSTÖR ARIZA



Network: 16
EMNİYET ŞALTERİ ARIZA



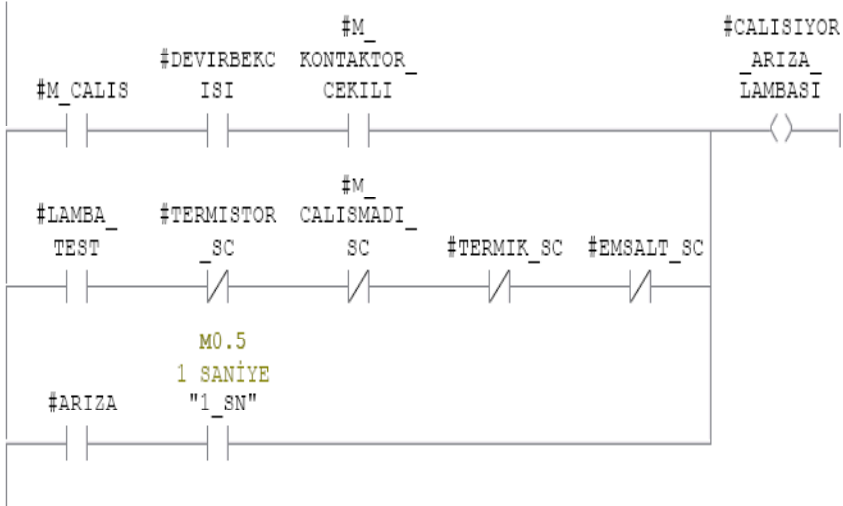
Network: 17
MOTOR ÇALIŞTI SİNYALİNİN SKADAYA AKTARILMASI



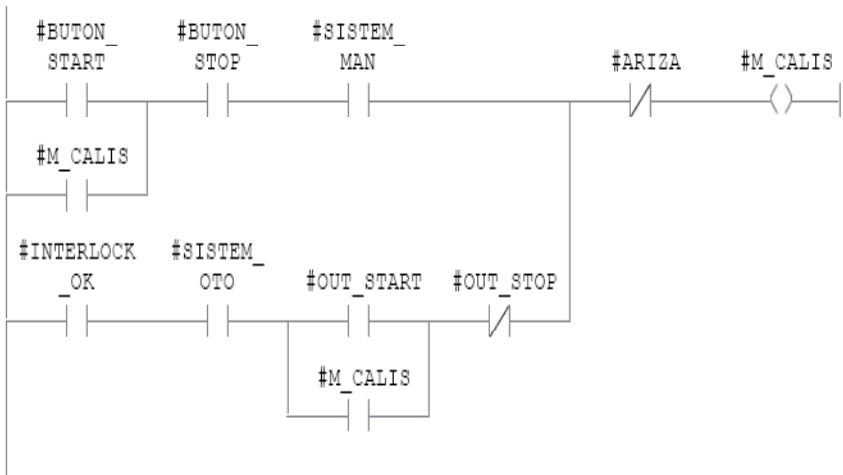
C1. (Devam) DOL FC (Tek Yönlü Direkt Yol Alan Motor Devresi Bloğu)

Network: 18 ARIZA VE ÇALIŞIYOR LAMBASI

SİSTEM ÇALIŞIYOR İSE LAMBA YANSIN.LAMBA TEST BUTONU BASILI İKEN'DE YANSIN.ARIZA VAR İKEN LAMBA TEST BUTONU'NA BASILSA DAHİ LAMBA SÜREKLİ YANMAMALI.ARIZA GELDİĞİNDE LAMBA BLİNG YAPSIN



Network: 19 MOTOR START BİTİNİN ÇEKTİRİLMESİ



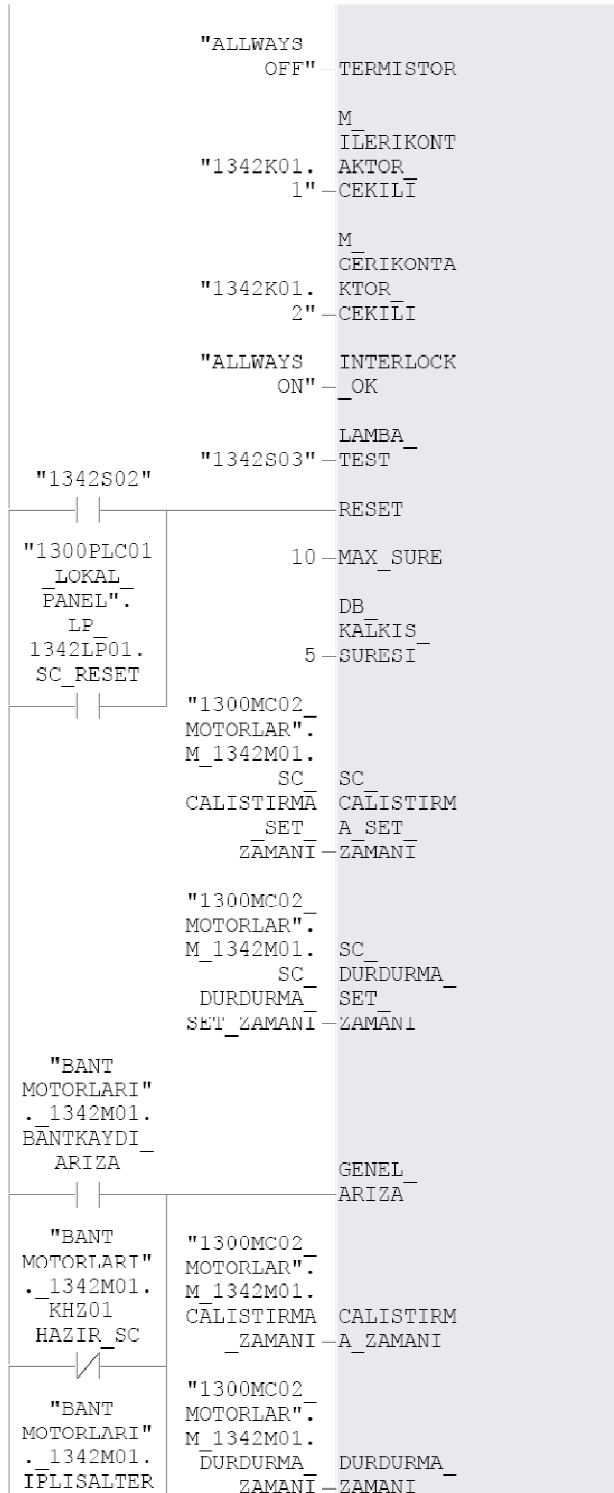
Şekil C3: DOL FC bloğu (tek yönlü direkt yol alan motor devresi) içyapısı

C2. DOL-2D FC (Çift Yönlü Direkt Yol Alan Motor Devresi Bloğu)

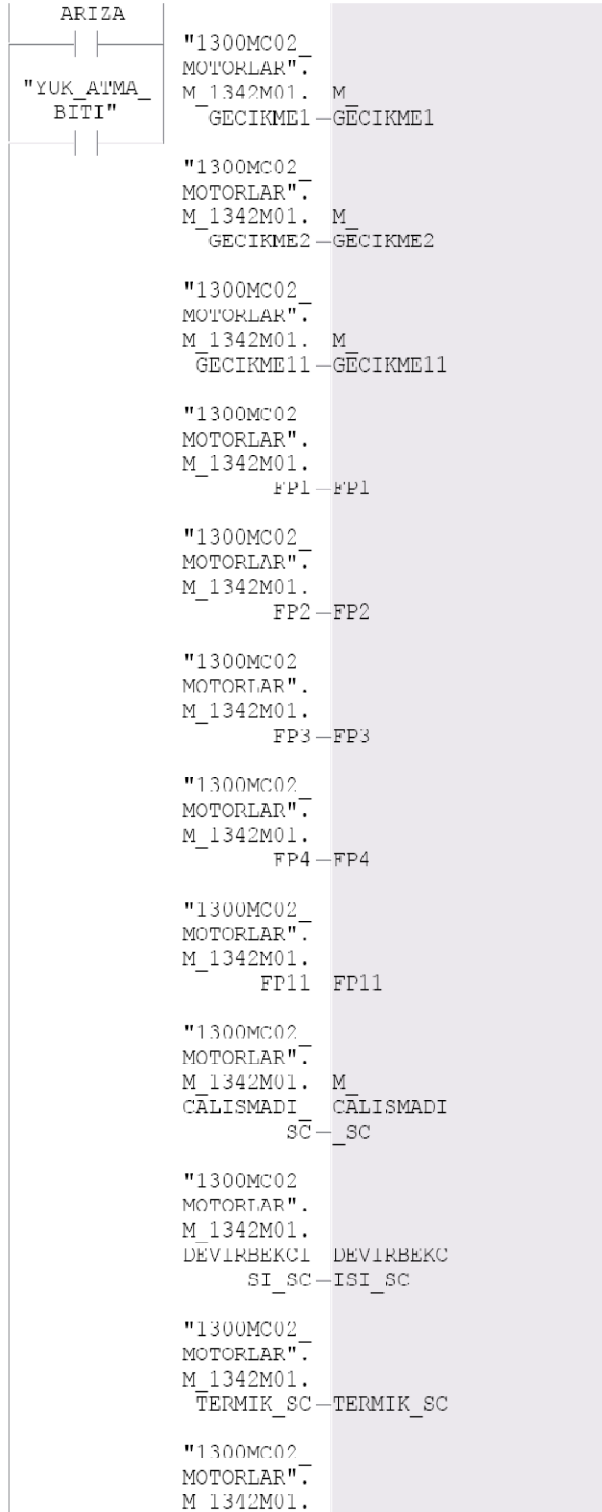
Network: 2 1342M01 7.5KW DOL-2D Belt conveyor Motoru Kontrolü

		"DOL_2D"			
		FN		FNO	
"1300PLC01 _LOKAL_ _PANEL". LP_ 1342LP01. SISTEM_	SISTEM_ MAN_SC-	MAN		ILERICALI S	"1342R01. 1"
"1300PLC01 _LOKAL_ _PANEL". LP_ 1342LP01. SISTEM_	SISTEM_ OTO_SC-	OTO		ILERICALI S	"1342R01. 2"
"1300MC02 MOTORLAR". M_1342M01. OTO ILERI START-T	OTO ILERISTAR	ILERISTAR		GERI CLSYR ARZ LAMBASI	"1342H01. 1" "1342H01. 2"
"1300MC02 MOTORLAR". M_1342M01. OTO_GERI START-	OTO GERISTAR	GERISTAR		GERI CLSYR ARZ LAMBASI	"1300MC02 MOTORLAR". M_1342M01. M_ M_ ILERICALI STI_SC
"1342S01. 1"-T	BUTON_ ILERISTAR	ILERISTAR		GERI CLSYR ARZ LAMBASI	"1300MC02 MOTORLAR". M_1342M01. M_ M_ ILERICALI STI_SC
"1342S01. 2"-GERISTART	BUTON_ GERISTART	GERISTART		GERI CLSYR ARZ LAMBASI	"1300MC02 MOTORLAR". M_1342M01. M_ M_ ILERICALI STI_SC
"1342S01. 3"-STOP	BUTON_ STOP	STOP		GERI CLSYR ARZ LAMBASI	"1300MC02 MOTORLAR". M_1342M01. M_ M_ ILERICALI STI_SC
"1300PLC01 _LOKAL_ _PANEL". LP_ 1300LP01. SC2_SC-	ACILSTOP	ACILSTOP		GERI CLSYR ARZ LAMBASI	"1300MC02 MOTORLAR". M_1342M01. M_ M_ ILERICALI STI_SC
"1342Q01"-TERMIK	TERMIK	TERMIK		GERI CLSYR ARZ LAMBASI	"1300MC02 MOTORLAR". M_1342M01. M_ M_ ILERICALI STI_SC
"1342B0101 "-TST	DEVIRBEKC "-TST	DEVIRBEKC "-TST		GERI CLSYR ARZ LAMBASI	"1300MC02 MOTORLAR". M_1342M01. M_ M_ ILERICALI STI_SC
"1342H0121 "-SALTERI	EMNIYET_ "-SALTERI	EMNIYET_ "-SALTERI		GERI CLSYR ARZ LAMBASI	"1300MC02 MOTORLAR". M_1342M01. M_ M_ ILERICALI STI_SC

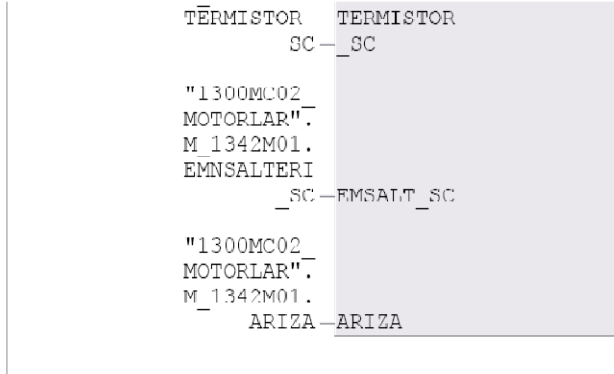
C2. (Devam) DOL-2D FC (Çift Yönlü Direkt Yol Alan Motor Devresi Bloğu)



C2. (Devam) DOL-2D FC (Çift Yönlü Direkt Yol Alan Motor Devresi Bloğu)



C2. (Devam) DOL-2D FC (Çift Yönlü Direkt Yol Alan Motor Devresi Bloğu)



Şekil C4: DOL-2D FC (Çift Yönlü Direkt Yol Alan Motor Devresi Bloğu) blok dış yapısı

C2. (Devam) DOL-2D FC (Çift Yönlü Direkt Yol Alan Motor Devresi Bloğu)

Name	Data Type	Address	Comment
IN		0.0	
SISTEM_MAN	Bool	0.0	=1 MANUAL(LP BLOĞUNUN LOJİK SONUCU)
SISTEM_OTO	Bool	0.1	=1 OTOMATİK(LP BLOĞUNUN LOJİK SONUCU)
OTO_ILERISTART	Bool	0.2	=1 ILERİ OTOSTART
OTO_GERISTART	Bool	0.3	=1 GERİ OTOSTART
BUTON_ILERISTART	Bool	0.4	=1 PUSH START ILERİ
BUTON_GERISTART	Bool	0.5	=1 PUSH START GERİ
BUTON_STOP	Bool	0.6	=0 PUSH STOP
ACILSTOP	Bool	0.7	=0 ACİLSTOP OKEY(=1 ARIZA DURUMU)
TERMIK	Bool	1.0	=1 TERMİK ARIZA
DEVIRBEKCISI	Bool	1.1	=1 DEVİRBEKÇİSİ ARIZA
EMNIYET_SALTERI	Bool	1.2	=1 EMNİYET ŞALTERİ ARIZA
TERMISTOR	Bool	1.3	=1 TERMİSTÖR ARIZA
M_ILERIKONTAKTOR_CEKILI	Bool	1.4	=1 KONTAKTÖR ILERİ ÇEKTİ BİLGİSİ
M_GERIKONTAKTOR_CEKILI	Bool	1.5	=1 KONTAKTÖR GERİ ÇEKTİ BİLGİSİ
INTERLOCK_OK	Bool	1.6	=1 OLMASI GEREKEN GENEL KOŞULLAR TAMAM
LAMBA_TEST	Bool	1.7	=1 PUSH LAMBA TEST
RESET	Bool	2.0	=1 PUSH RESET
MAX_SURE	Int	4.0	KONTAKTOR ÇALIŞMA SINYALİNİN GELMESİ GEREKEN SÜRE
DB_KALKIS_SURESI	Int	6.0	DEVİRBEKÇİSİ SINYALİNİN MOTOR KALKTIKTAN SONRA GELMESİ GEREKEN SÜRE
SC_CALISTIRMA_SET_ZAMANI	Int	8.0	OTOMATİK ÇALIŞTIRMADA SET EDİLEN KALKIŞ SÜRESİ
SC_DURDURMA_SET_ZAMANI	Int	10.0	OTOMATİK ÇALIŞTIRMADA SET EDİLEN DURUŞ SÜRESİ
GENEL_ARIZA	Bool	12.0	GENEL ARIZA
OUT		0.0	
M_ILERICALIS	Bool	14.0	MOTOR ILERİ ÇALIŞ
M_GERICALIS	Bool	14.1	MOTOR GERİ ÇALIŞ

C2. (Devam) DOL-2D FC (Çift Yönlü Direkt Yol Alan Motor Devresi Bloğu)

Name	Data Type	Address	Comment
ILERI_CLSYR_ARZ_LAMBASI	Bool	14.2	ARIZA ILERI ÇALIŞIYOR LAMBASI(=1 MOTOR ÇALIŞIYOR,=1-0 BLİNG MOTOR ARIZA)
GERI_CLSYR_ARZ_LAMBASI	Bool	14.3	ARIZA GERI ÇALIŞIYOR LAMBASI(=1 MOTOR ÇALIŞIYOR,=1-0 BLİNG MOTOR ARIZA)
M_ILERICALISTI_SC	Bool	14.4	MOTOR KESİN İLERİ ÇALIŞTI SKADA SİNYALİ
M_GERICALISTI_SC	Bool	14.5	MOTOR KESİN GERİ ÇALIŞTI SKADA SİNYALİ
IN_OUT		0.0	
CALISTIRMA_ZAMANI	Int	16.0	OTOMATİK ÇALIŞMADA MOTOR START GECİKME ZAMANI
DURDURMA_ZAMANI	Int	18.0	OTOMATİK ÇALIŞMADA MOTOR STOP GECİKME ZAMANI
M_GECIKME1	Int	20.0	START GELDI FAKAT KONTAKTOR CEKMEDİ DURUMU İÇİN GEÇEN SÜRE
M_GECIKME2	Int	22.0	START GELDI FAKAT DEVİRBEKCİSİ GELMEDİ DURUMU İÇİN GEÇEN SÜRE
M_GECIKME11	Int	24.0	START GELDI FAKAT KONTAKTOR CEKMEDİ DURUMU İÇİN GEÇEN SÜRE
FP1	Bool	26.0	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
FP2	Bool	26.1	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
FP3	Bool	26.2	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
FP4	Bool	26.3	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
FP11	Bool	26.4	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
M_CALISMADI_SC	Bool	26.5	MOTOR ÇALIŞMADI SKADA SİNYALİ
DEVİRBEKCİSİ_SC	Bool	26.6	DEVİRBEKCİSİ SKADA SİNYALİ
TERMIK_SC	Bool	26.7	TERMIK SKADA SİNYALİ
TERMİSTOR_SC	Bool	27.0	TERMİSTÖR SKADA SİNYALİ
EMSALT_SC	Bool	27.1	EMNİYET ŞALTERİ SKADA SİNYALİ
ARIZA	Bool	27.2	TÜM ARIZALARIN GENEL SONUCU
TEMP		0.0	
OUT_ILERI_START	Bool	0.0	OTOMATİK İLERİ START GECİKMESİ
OUT_GERI_START	Bool	0.1	OTOMATİK GERİ START GECİKMESİ
OUT_ILERI_STOP	Bool	0.2	OTOMATİK İLERİ STOP GECİKMESİ
OUT_GERI_STOP	Bool	0.3	OTOMATİK GERİ STOP GECİKMESİ
RETURN		0.0	
RET_VAL		0.0	

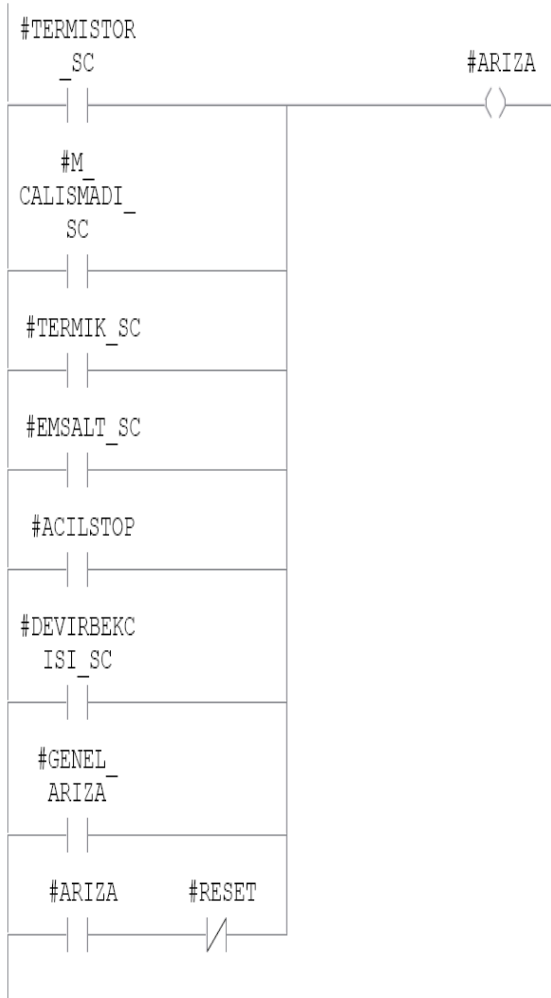
Şekil C5: DOL-2D FC (Çift Yönlü Direkt Yol Alan Motor Devresi Bloğu) blok deklarasyon tablosu

C2. (Devam) DOL-2D FC (Çift Yönlü Direkt Yol Alan Motor Devresi Bloğu)

Block: FC22

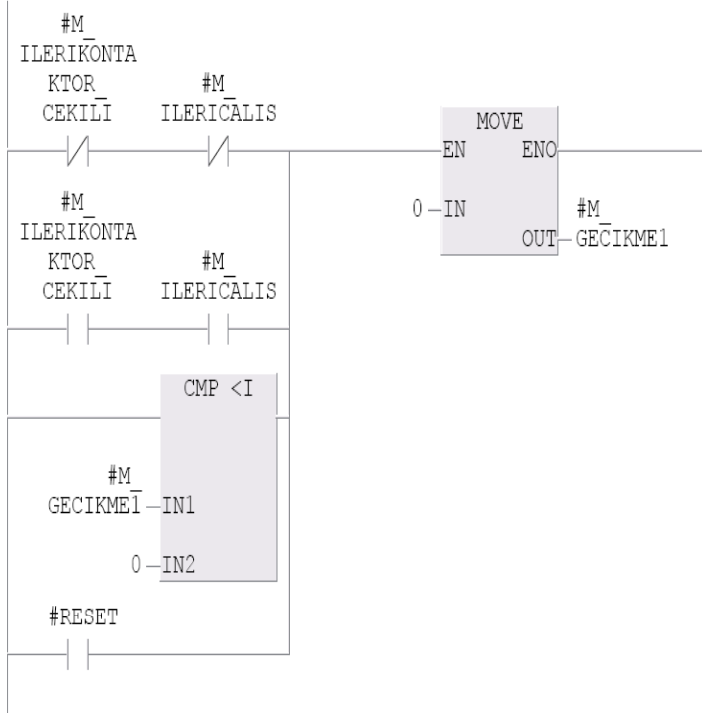
TEK YÖN MOTOR START-STOP BUTON'LU(DİREKT VE YILDIZ ÜÇGEN DEVRELİ)

Network: 1 ARIZA



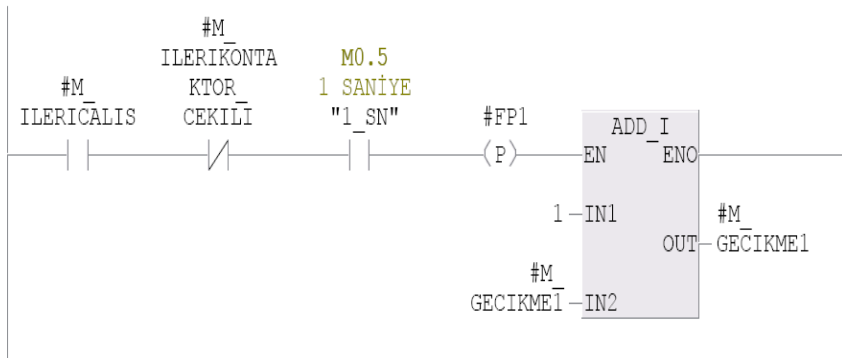
C2. (Devam) DOL-2D FC (Çift Yönlü Direkt Yol Alan Motor Devresi Bloğu)

Network: 2 MOTOR GECİKME-1 SAYACININ SIFIRLANMASI



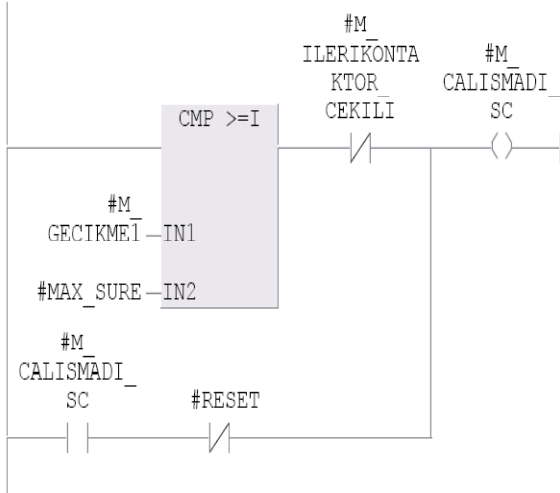
Network: 3 MOTOR GECİKME-1 SAYACI

MOTOR ÇALIŞMADI BİLGİSİNİ KONTAKTÖR'DEN ALIP SÜREYİ SAYMAYA BAŞLARIZ.

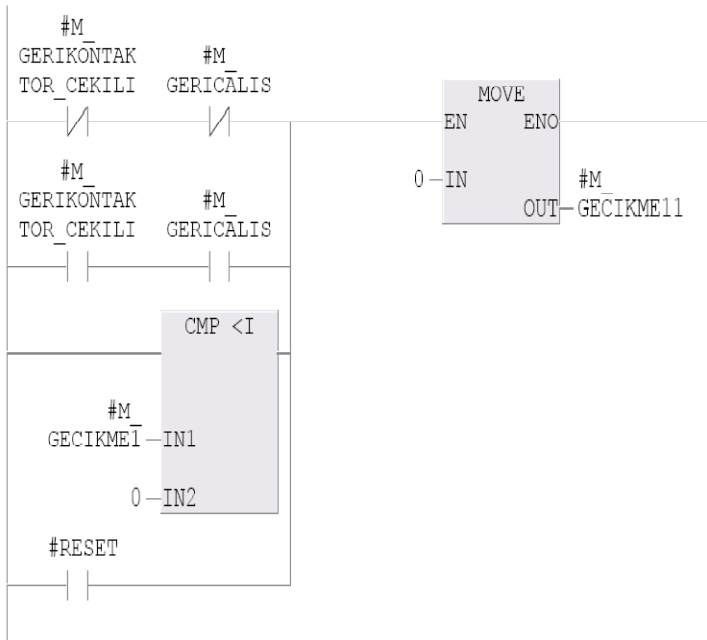


C2. (Devam) DOL-2D FC (Çift Yönlü Direkt Yol Alan Motor Devresi Bloğu)

Network: 4
MOTOR ÇALIŞMADI BİLGİSİ KONTAKTÖRDEN İSE = MOTOR ÇALIŞMADI HATASI



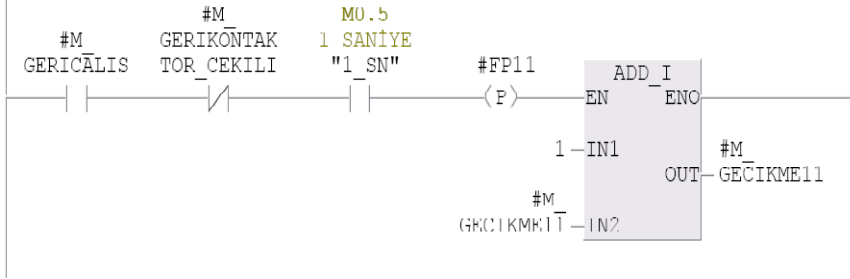
Network: 5 MOTOR GECİKME-1 SAYACININ SIFIRLANMASI



C2. (Devam) DOL-2D FC (Çift Yönlü Direkt Yol Alan Motor Devresi Bloğu)

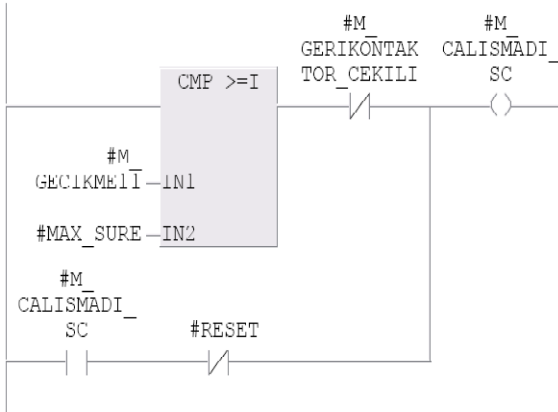
Network: 6 MOTOR GECİKME-11 SAYACI

MOTOR ÇALIŞMADI BİLGİSİNİ KONTAKTÖR'DEN ALIP SÜREYİ SAYMAYA BAŞLARIZ.

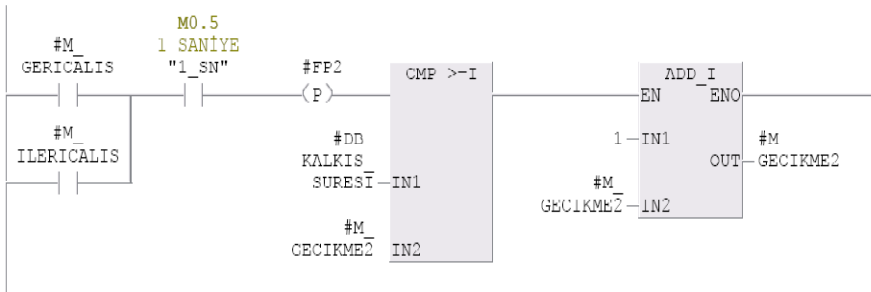


Network: 7

MOTOR ÇALIŞMADI BİLGİSİ KONTAKTÖRDEN İSE = MOTOR ÇALIŞMADI HATASI

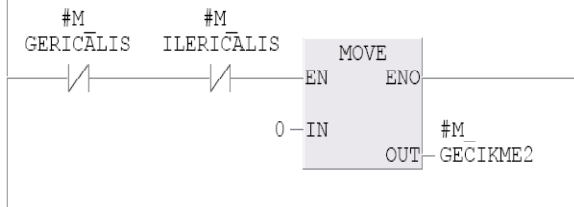


Network: 8 MOTOR GECİKME 2 SAYACI



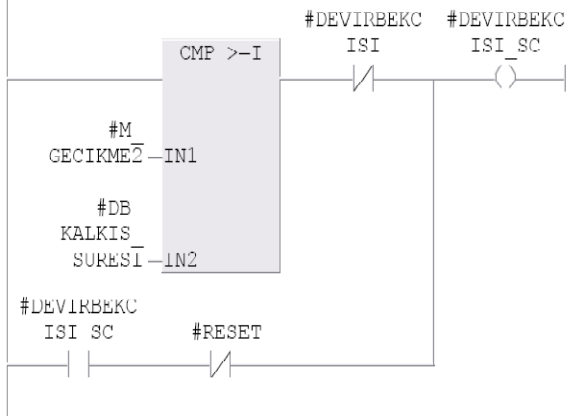
C2. (Devam) DOL-2D FC (Çift Yönlü Direkt Yol Alan Motor Devresi Bloğu)

Network: 9 MOTOR GEÇİKME-2 SAYACININ SIFIRLANMASI



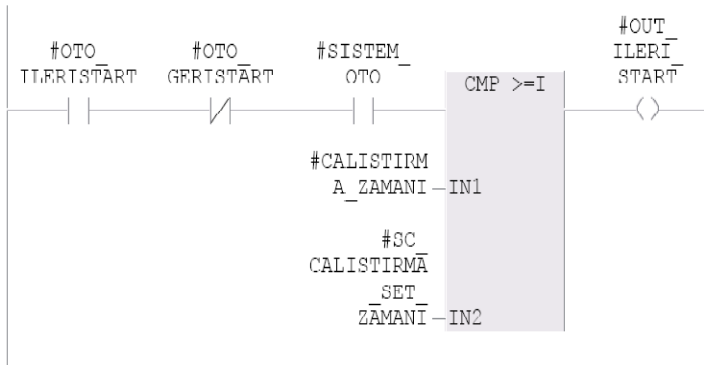
Network: 10

KONTAKTÖR BİLGİSİ VAR DEVİRBEKCİSİ YOK İSE DEVİR BEKÇİSİ ARIZA



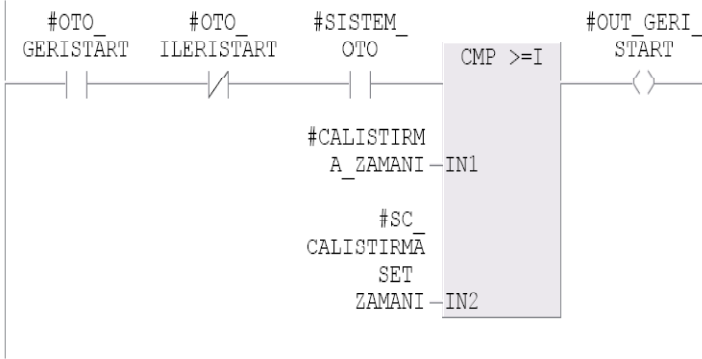
Network: 11 İLERİ START GECİKMESİ

SİSTEM OTOMATİKTE İKEN START GECİKME ZAMANI

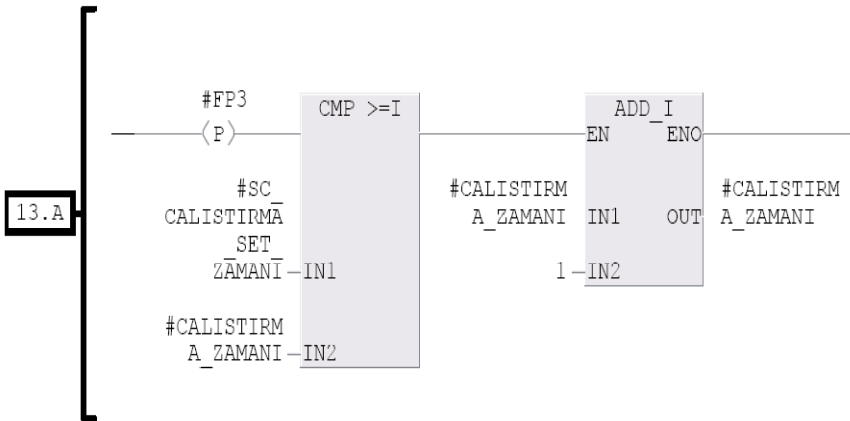
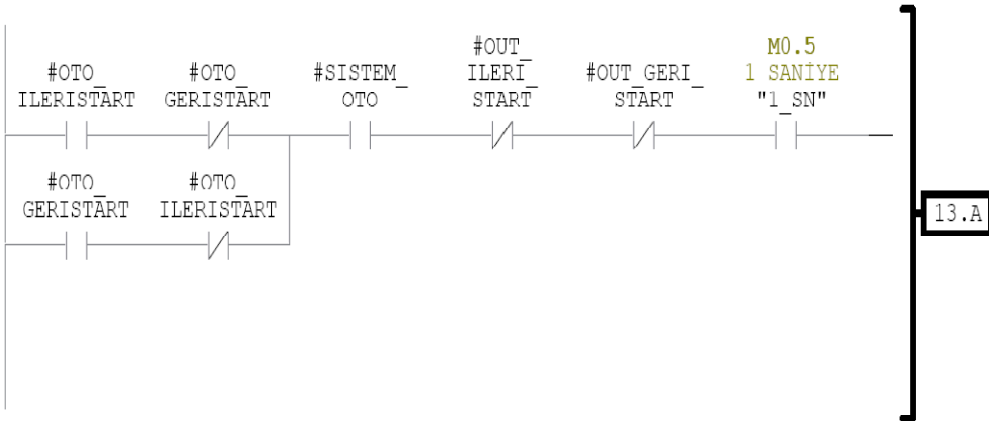


C2. (Devam) DOL-2D FC (Çift Yönlü Direkt Yol Alan Motor Devresi Bloğu)

Network: 12 GERİ START GECİKMESİ
SİSTEM OTOMATİKTE İKEN START GECİKME ZAMANI

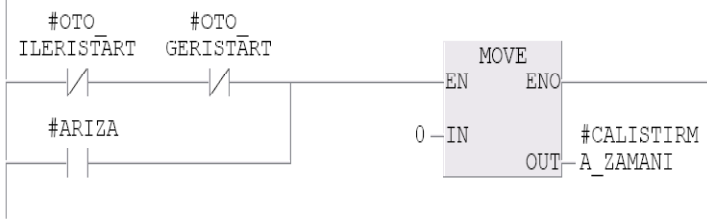


Network: 13 START GECİKMESİ İÇİN SAYMA

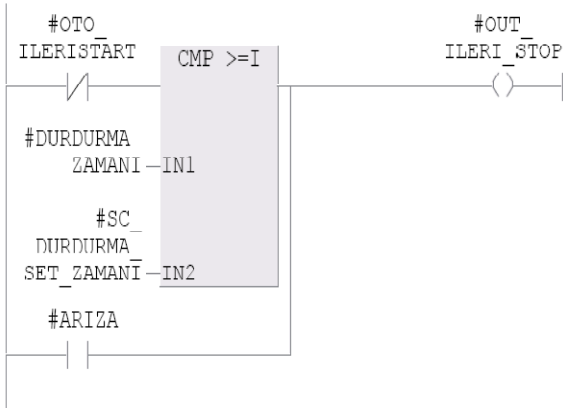


C2. (Devam) DOL-2D FC (Çift Yönlü Direkt Yol Alan Motor Devresi Bloğu)

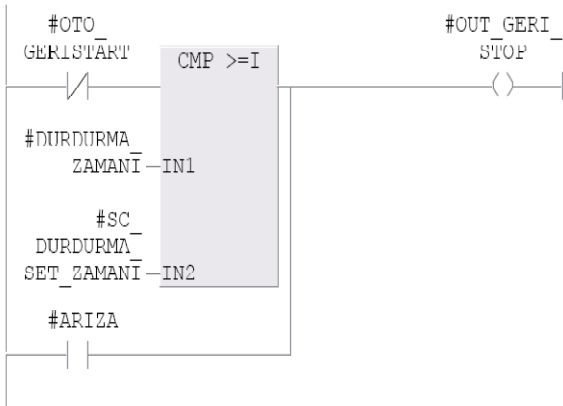
Network: 14 START ZAMANINI SIFIRLAMA



Network: 15 İLERİ STOP GECİKMESİ

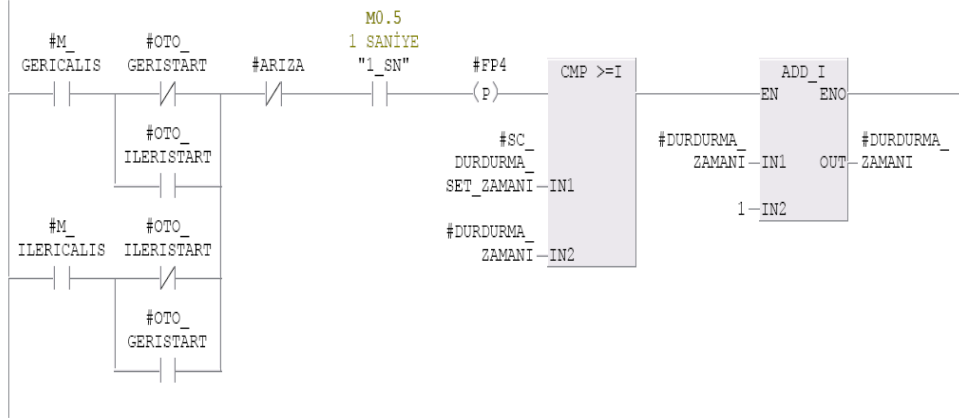


Network: 16 GERİ STOP GECİKMESİ

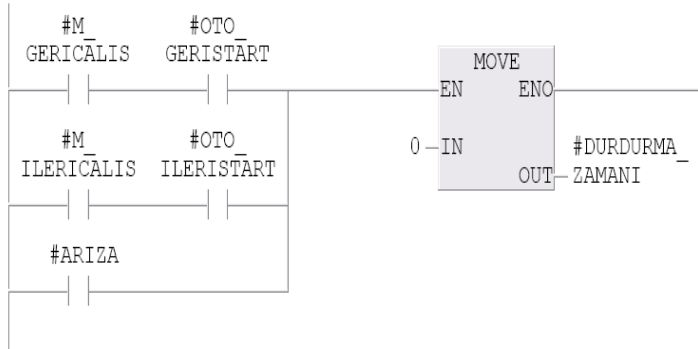


C2. (Devam) DOL-2D FC (Çift Yönlü Direkt Yol Alan Motor Devresi Bloğu)

Network: 17 STOP GECİKMESİ İÇİN SAYMA

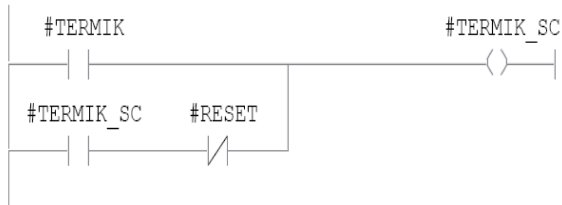


Network: 18 STOP ZAMANINI SIFIRLAMA



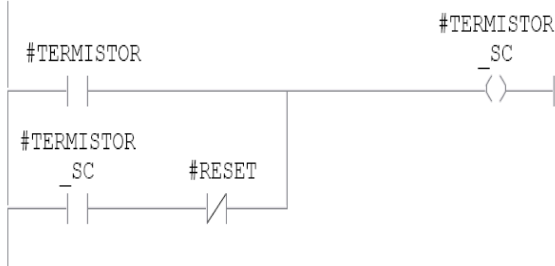
Network: 19 1 SANİYE

TERMİK ARIZA

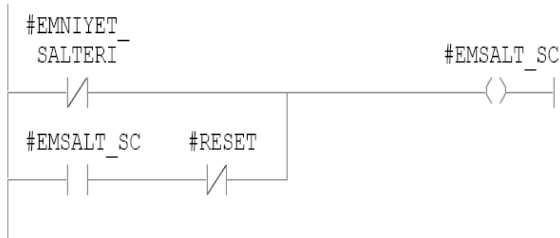


C2. (Devam) DOL-2D FC (Çift Yönlü Direkt Yol Alan Motor Devresi Bloğu)

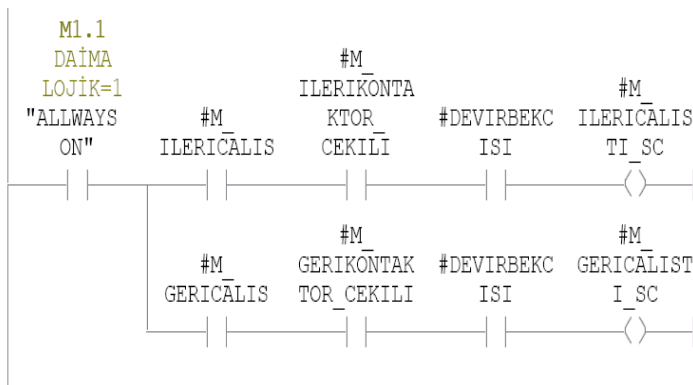
Network: 20
TERMİSTÖR ARIZA



Network: 21
EMNİYET ŞALTERİ ARIZA

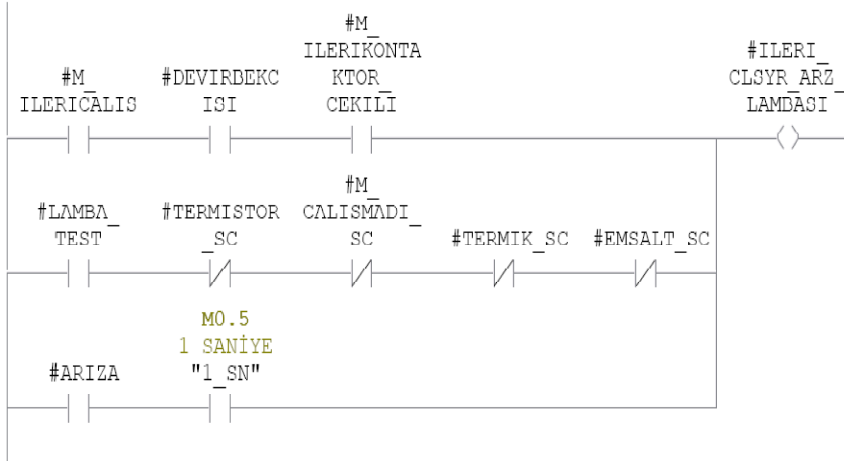


Network: 22
MOTOR ÇALIŞTI SİNYALİNİN SKADAYA AKTARILMASI

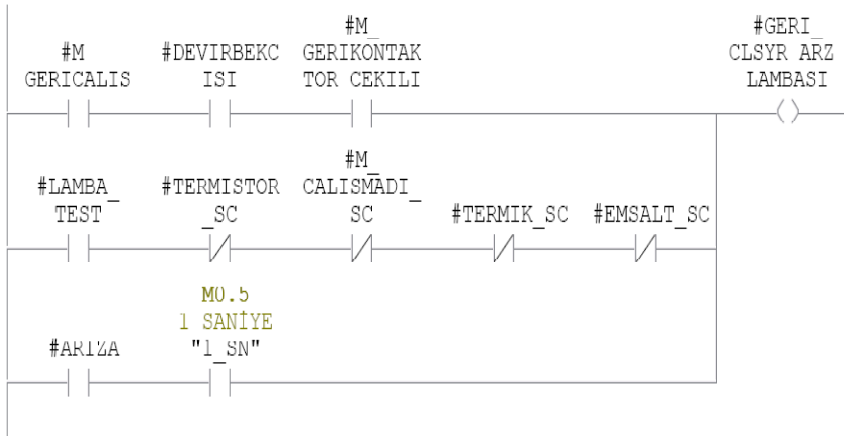


C2. (Devam) DOL-2D FC (Çift Yönlü Direkt Yol Alan Motor Devresi Bloğu)

Network: 23	ARIZA VE ÇALIŞIYOR LAMBASI
SİSTEM ÇALIŞIYOR İSE LAMBA YANSIN.LAMBA TEST BUTONU BASILI İKEN'DE YANSIN.ARIZA VAR İKEN LAMBA TEST BUTONU'NA BASILSA DAHİ LAMBA SÜREKLİ YANMAMALI.ARIZA GELDİĞİNDE LAMBA BLİNG YAPSIN	



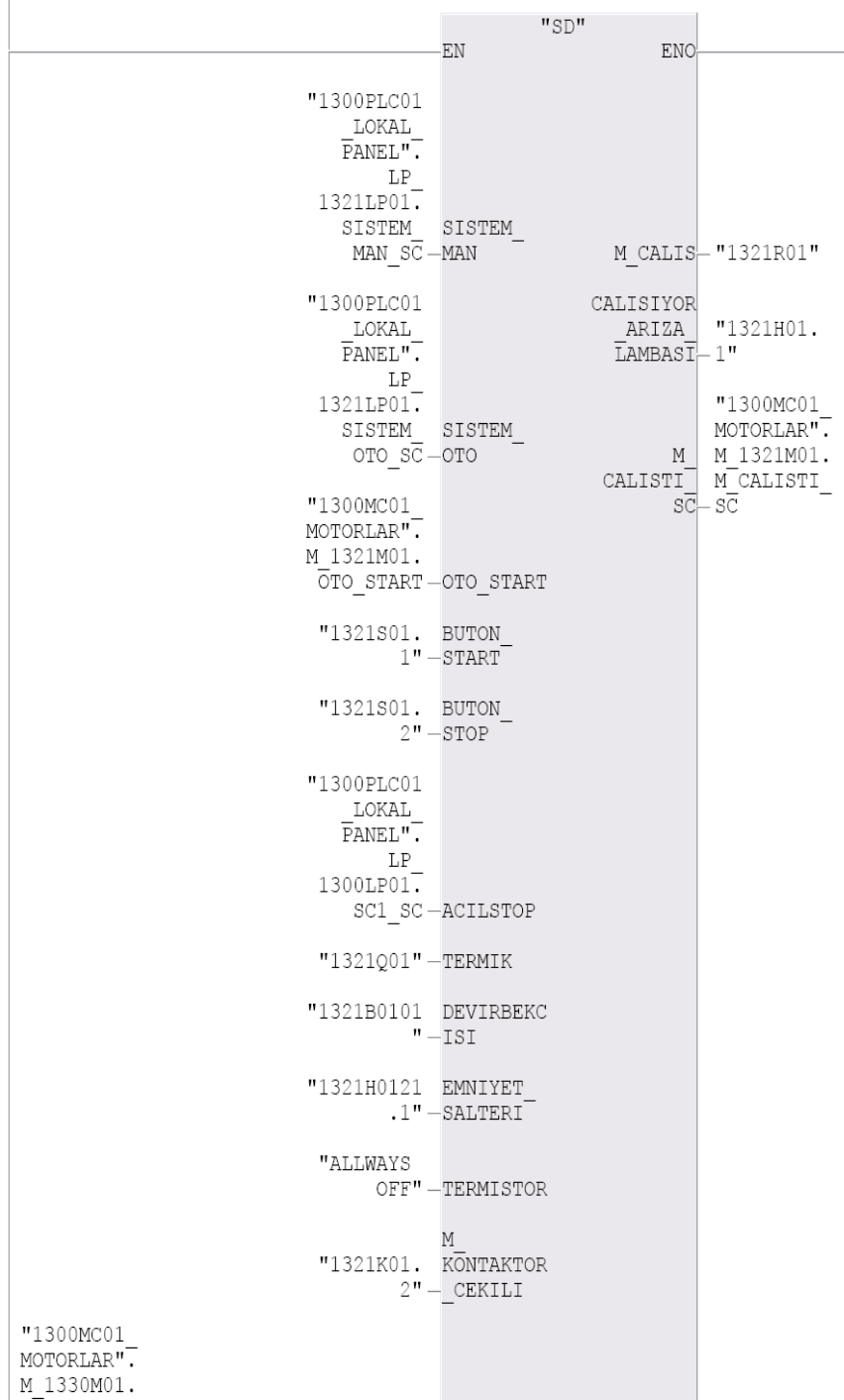
Network: 24	ARIZA VE ÇALIŞIYOR LAMBASI
SİSTEM ÇALIŞIYOR İSE LAMBA YANSIN.LAMBA TEST BUTONU BASILI İKEN'DE YANSIN.ARIZA VAR İKEN LAMBA TEST BUTONU'NA BASILSA DAHİ LAMBA SÜREKLİ YANMAMALI.ARIZA GELDİĞİNDE LAMBA BLİNG YAPSIN	



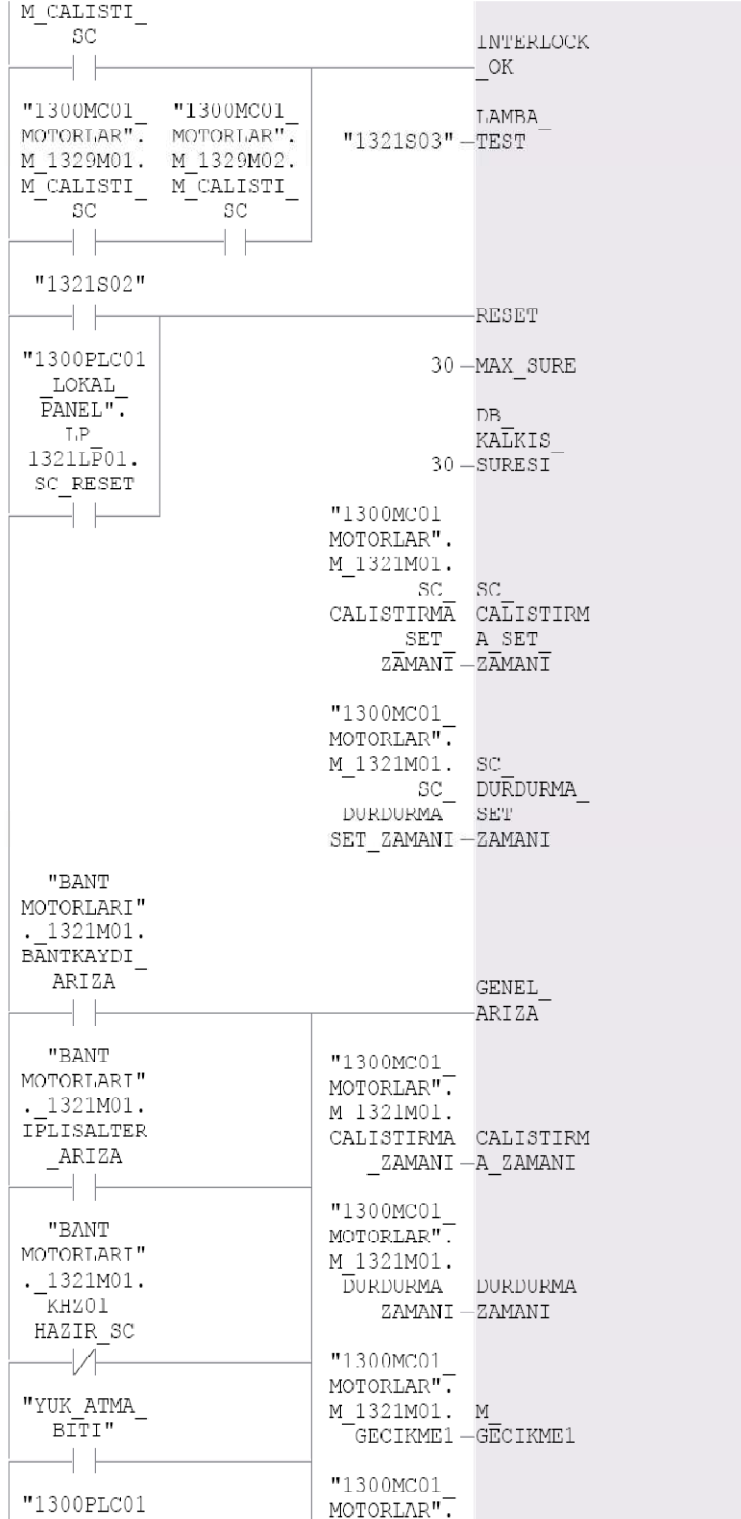
Şekil C6: DOL-2D FC (Çift Yönlü Direkt Yol Alan Motor Devresi Bloğu)

C3. SD FC (Tek Yönlü Yıldız Üçgen Yol Alan Motor Devresi Bloğu)

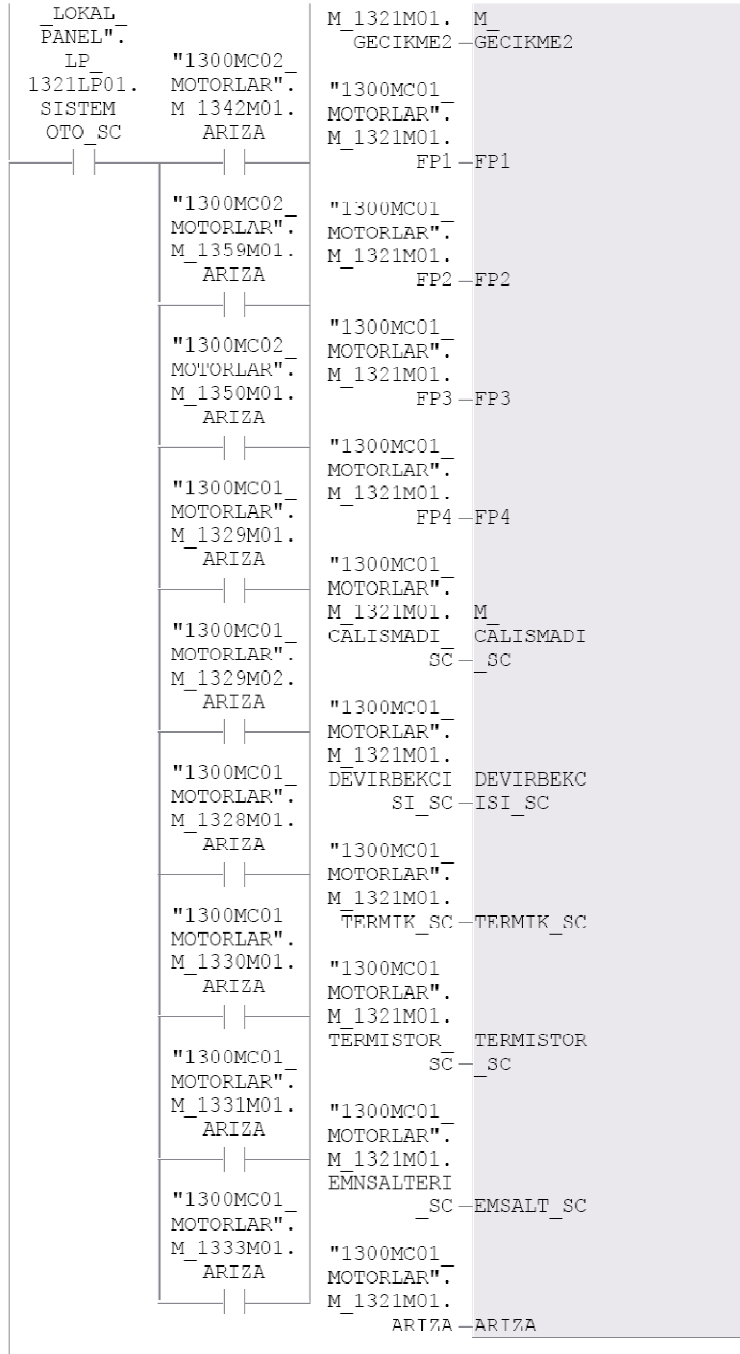
Network: 1 1321M01 22KW SD Belt conveyor Motoru Kontrolü



C3. SD FC (Devam) (Tek Yönlü Yıldız Üçgen Yol Alan Motor Devresi Bloğu)



C3. (Devam) SD FC (Tek Yönlü Yıldız Üçgen Yol Alan Motor Devresi Bloğu)



Şekil C7: SD FC (Tek Yönlü Yıldız Üçgen Yol Alan Motor Devresi Bloğu) blok dış yapısı

C3. (Devam) SD FC (Tek Yönlü Yıldız Üçgen Yol Alan Motor Devresi Bloğu)

Name	Data Type	Address	Comment
IN		0.0	
SISTEM_MAN	Bool	0.0	=1 MANUAL (LP BLOĞUNUN LOJİK SONUCU)
SISTEM_OTO	Bool	0.1	=1 OTOMATİK (LP BLOĞUNUN LOJİK SONUCU)
OTO_START	Bool	0.2	=1 OTOSTART
BUTON_START	Bool	0.3	=1 PUSH START
BUTON_STOP	Bool	0.4	=0 PUSH STOP
ACILSTOP	Bool	0.5	=0 ACILSTOP OKEY (=1 ARIZA DURUMU)
TERMIK	Bool	0.6	=1 TERMİK ARIZA
DEVIRBEKCISI	Bool	0.7	=1 DEVİRBEKÇİSİ ARIZA
EMNIYET_SALTERI	Bool	1.0	=1 EMNİYET ŞALTERİ ARIZA
TERMISTOR	Bool	1.1	=1 TERMİSTÖR ARIZA
M_KONTAKTOR_CEKILI	Bool	1.2	=1 KONTAKTÖR ÇEKTİ BİLGİSİ
INTERLOCK_OK	Bool	1.3	=1 OLMASI GEREKEN GENEL KOŞULLAR TAMAM
LAMBA_TEST	Bool	1.4	=1 PUSH LAMBA TEST
RESET	Bool	1.5	=1 PUSH RESET
MAX_SURE	Int	2.0	KONTAKTOR ÇALIŞMA SİNYALİNİN GELMESİ GEREKEN SÜRE
DB_KALKIS_SURESI	Int	4.0	DEVİRBEKÇİSİ SİNYALİNİN MOTOR KALKTIKTAN SONRA GELMESİ GEREKEN SÜRE
SC_CALISTIRMA_SET_ZAMANI	Int	6.0	OTOMATİK ÇALIŞTIRMADA SET EDİLEN KALKIŞ SÜRESİ
SC_DURDURMA_SET_ZAMANI	Int	8.0	OTOMATİK ÇALIŞTIRMADA SET EDİLEN DURUŞ SÜRESİ
GENEL_ARIZA	Bool	10.0	GENEL ARIZA
OUT		0.0	
M_CALIS	Bool	12.0	MOTOR ÇALIŞ
CALISIYOR_ARIZA_LAMBASI	Bool	12.1	ARIZA ÇALIŞIYOR LAMBASI (=1 MOTOR ÇALIŞIYOR, =1-0 BLİNG MOTOR ARIZA)
M_CALISTI_SC	Bool	12.2	MOTOR KESİN ÇALIŞTI SKADA SİNYALİ

C3. (Devam) SD FC (Tek Yönlü Yıldız Üçgen Yol Alan Motor Devresi Bloğu)

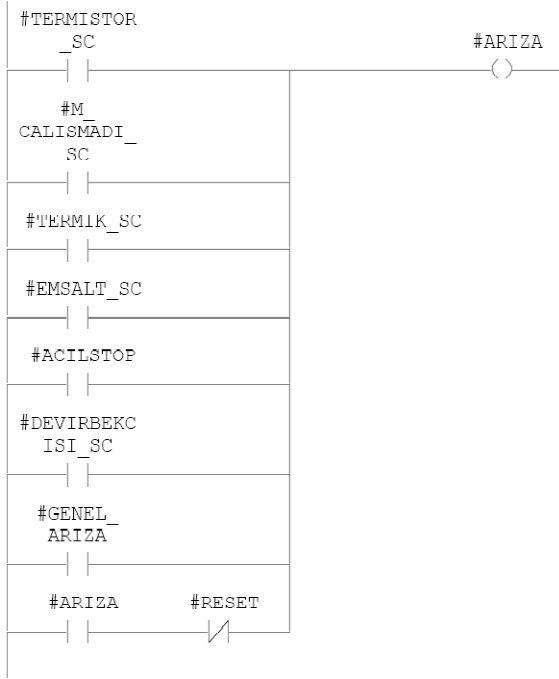
Name	Data Type	Address	Comment
IN_OUT		0.0	
CALISTIRMA_ZAMANI	Int	14.0	OTOMATİK ÇALIŞMADA MOTOR START GECİKME ZAMANI
DURDURMA_ZAMANI	Int	16.0	OTOMATİK ÇALIŞMADA MOTOR STOP GECİKME ZAMANI
M_GECIKME1	Int	18.0	START GELDI FAKAT KONTAKTOR CEKMEDE DURUMU İÇİN GEÇEN SÜRE
M_GECIKME2	Int	20.0	START GELDI FAKAT DEVİRBEKCİSİ GELMEDİ DURUMU İÇİN GEÇEN SÜRE
FP1	Bool	22.0	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
FP2	Bool	22.1	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
FP3	Bool	22.2	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
FP4	Bool	22.3	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
M_CALISMADI_SC	Bool	22.4	MOTOR ÇALIŞMADI SKADA SİNYALİ
DEVİRBEKCİSİ_SC	Bool	22.5	DEVİRBEKCİSİ SKADA SİNYALİ
TERMİK_SC	Bool	22.6	TERMİK SKADA SİNYALİ
TERMİSTÖR_SC	Bool	22.7	TERMİSTÖR SKADA SİNYALİ
EMSALT_SC	Bool	23.0	EMNİYET ŞALTERİ SKADA SİNYALİ
ARIZA	Bool	23.1	TÜM ARIZALARIN GENEL SONUCU
TEMP		0.0	
OUT_START	Bool	0.0	OTOMATİK START GECİKMESİ
OUT_STOP	Bool	0.1	OTOMATİK STOP GECİKMESİ
RETURN		0.0	
RET_VAL		0.0	

Şekil C8: SD FC (Tek Yönlü Yıldız Üçgen Yol Alan Motor Devresi Bloğu) deklaryasyon tablosu

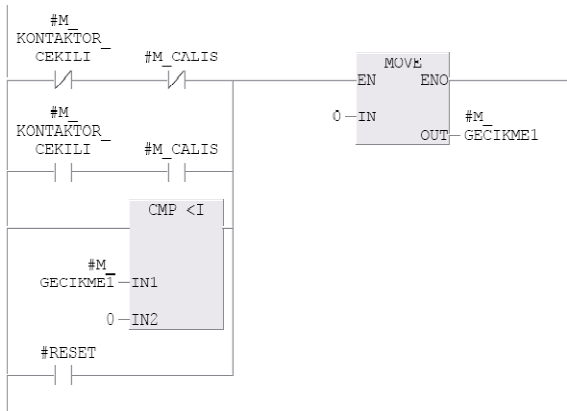
C3. (Devam) SD FC (Tek Yönlü Yıldız Üçgen Yol Alan Motor Devresi Bloğu)

Block: FC23
TEK YÖN MOTOR START-STOP BUTON'LU (DİREKT VE YILDIZ ÜÇGEN DEVRELİ)

Network: 1

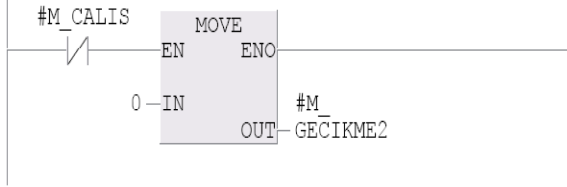


Network: 2 MOTOR GECİKME-1 SAYACININ SIFIRLANMASI



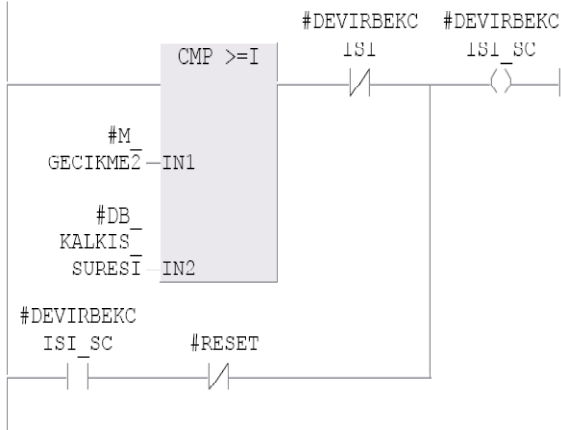
C3. (Devam) SD FC (Tek Yönlü Yıldız Üçgen Yol Alan Motor Devresi Bloğu)

Network: 6 MOTOR GECİKME-2 SAYACININ SIFIRLANMASI



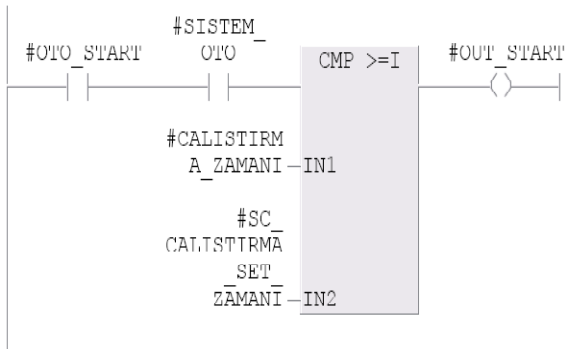
Network: 7

KONTAKTÖR BİLGİSİ VAR DEVİRBEKCİSİ YOK İSE DEVİR BEKÇİSİ ARIZA



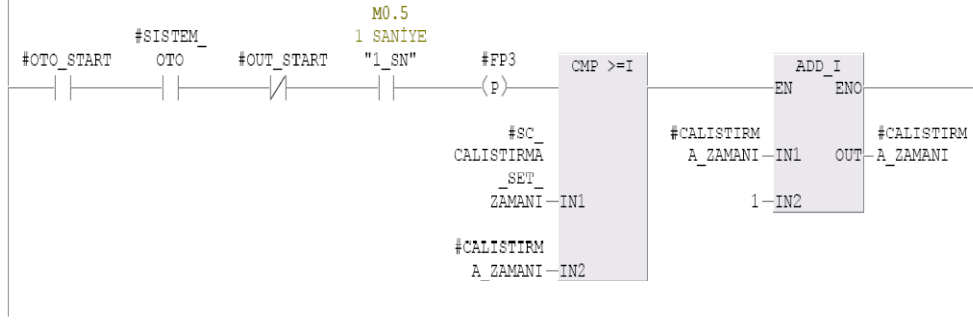
Network: 8 START GECİKMESİ

SİSTEM OTOMATİKTE İKEN START GECİKME ZAMANI

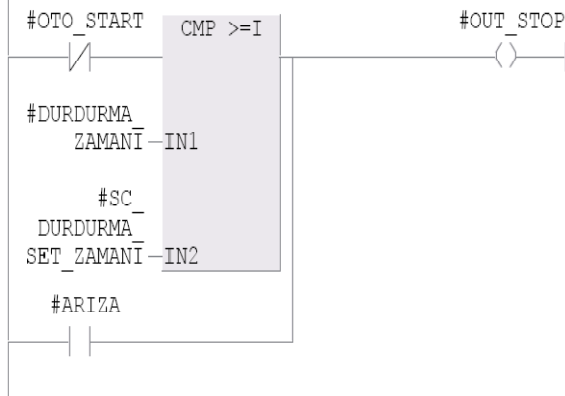


C3. (Devam) SD FC (Tek Yönlü Yıldız Üçgen Yol Alan Motor Devresi Bloğu)

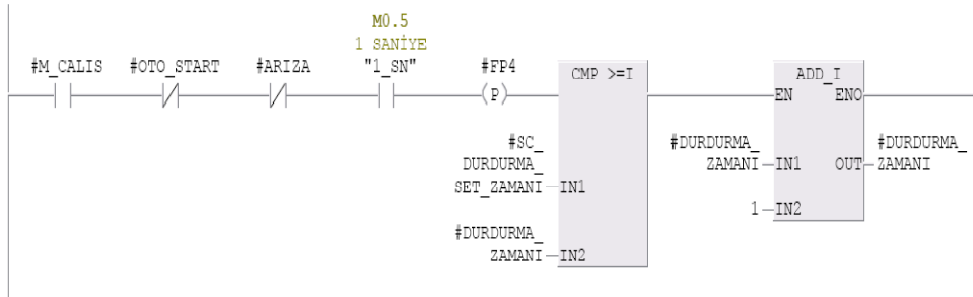
Network: 9 START GECİKMESİ İÇİN SAYMA



Network: 10 STOP GECİKMESİ

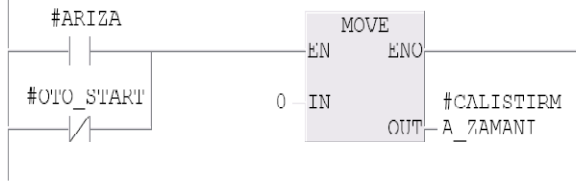


Network: 11 STOP GECİKMESİ İÇİN SAYMA

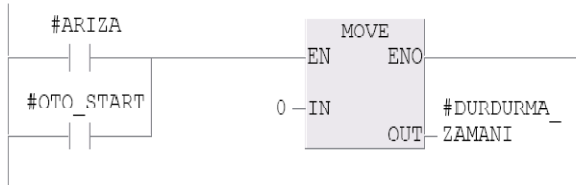


C3. (Devam) SD FC (Tek Yönlü Yıldız Üçgen Yol Alan Motor Devresi Bloğu)

Network: 12 START ZAMANINI SIFIRLAMA



Network: 13 STOP ZAMANINI SIFIRLAMA



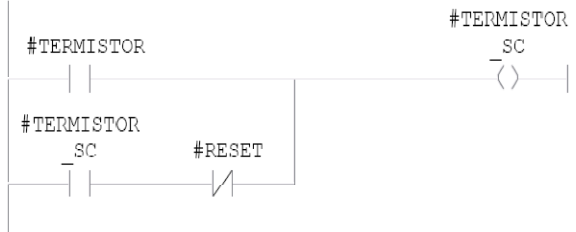
Network: 14

TERMİK ARIZA



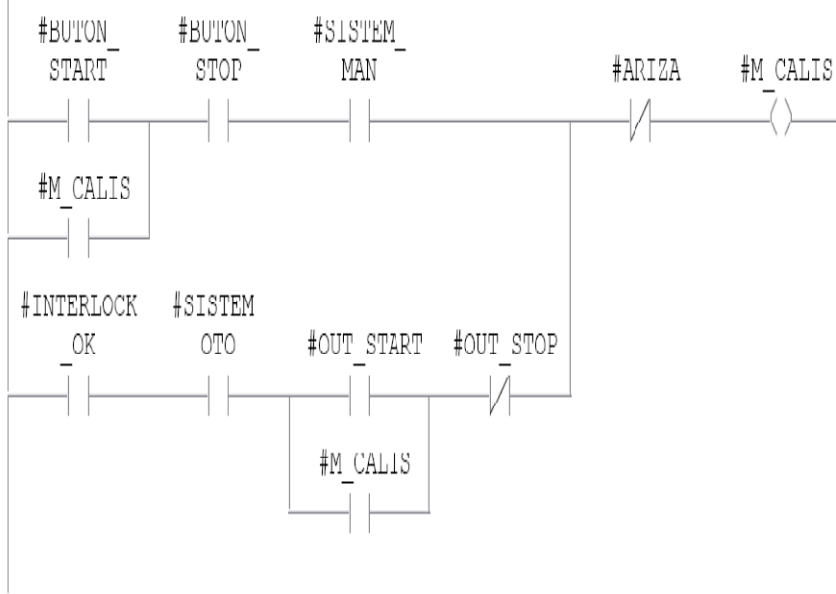
Network: 15

TERMİSTÖR ARIZA



C3. (Devam) SD FC (Tek Yönlü Yıldız Üçgen Yol Alan Motor Devresi Bloğu)

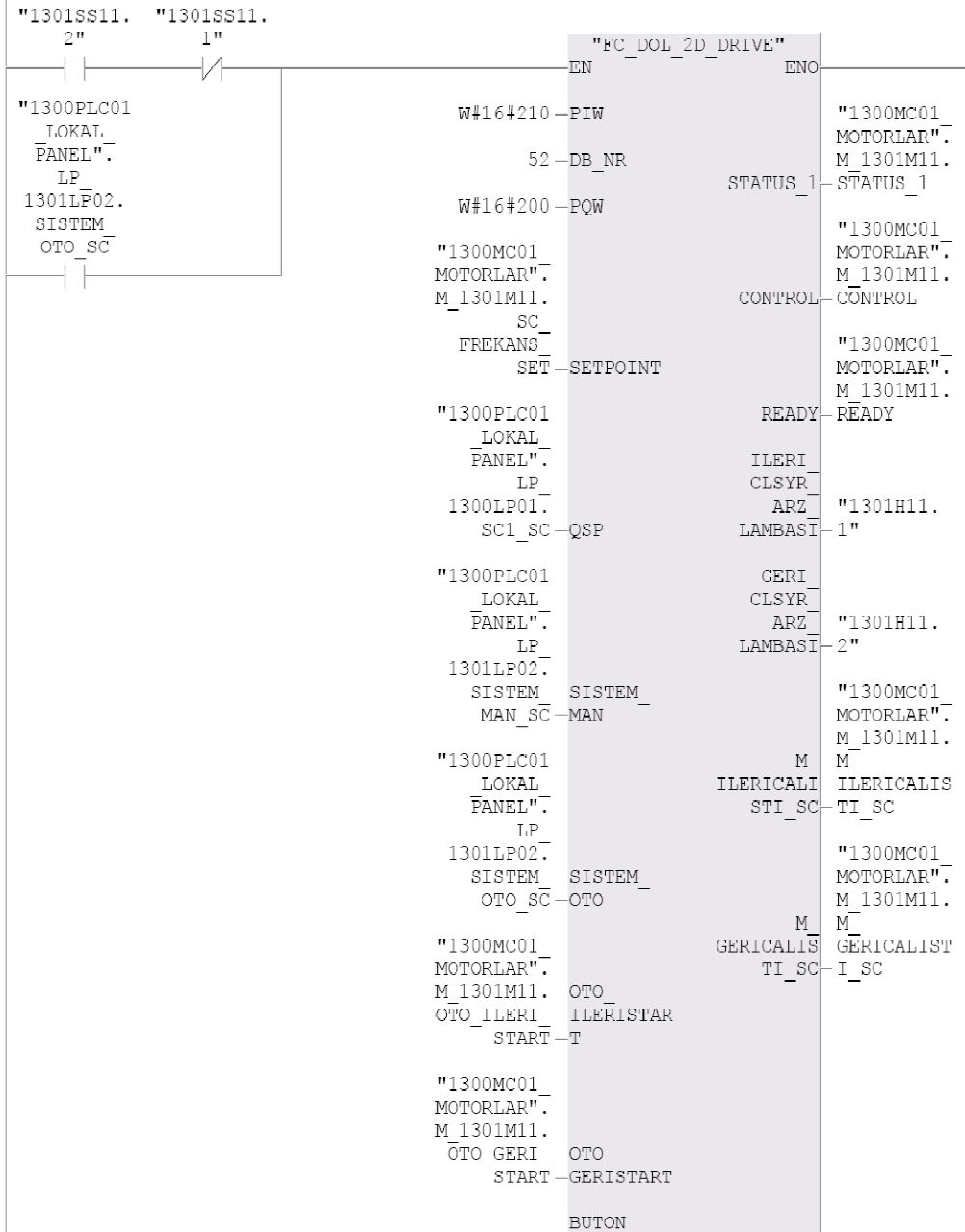
Network: 19 MOTOR START BİTİNİN ÇEKTİRİLMESİ



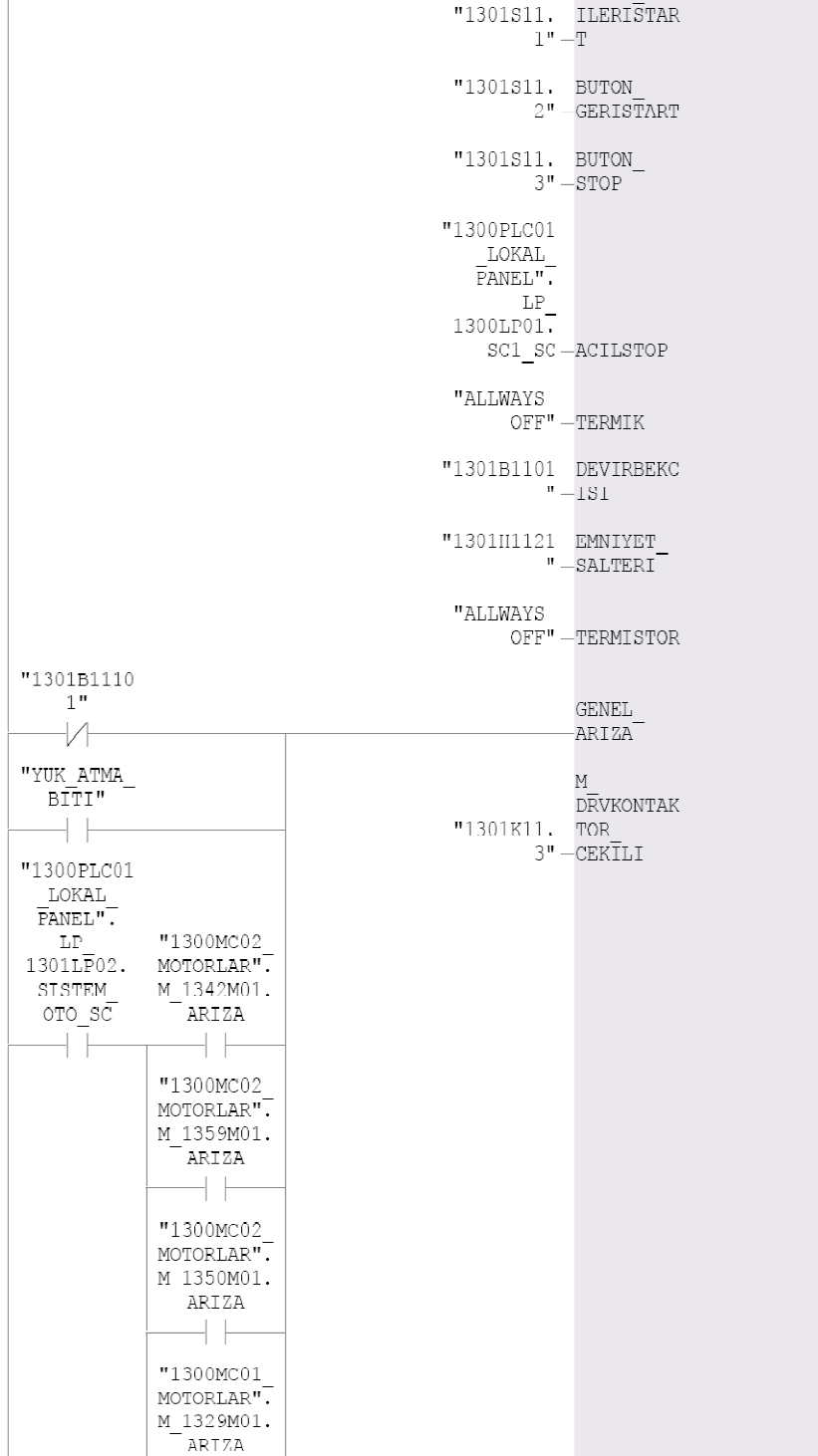
Şekil C9: SD FC (Tek Yönlü Yıldız Üçgen Yol Alan Motor Devresi Bloğu) blok içyapısı

C4. FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

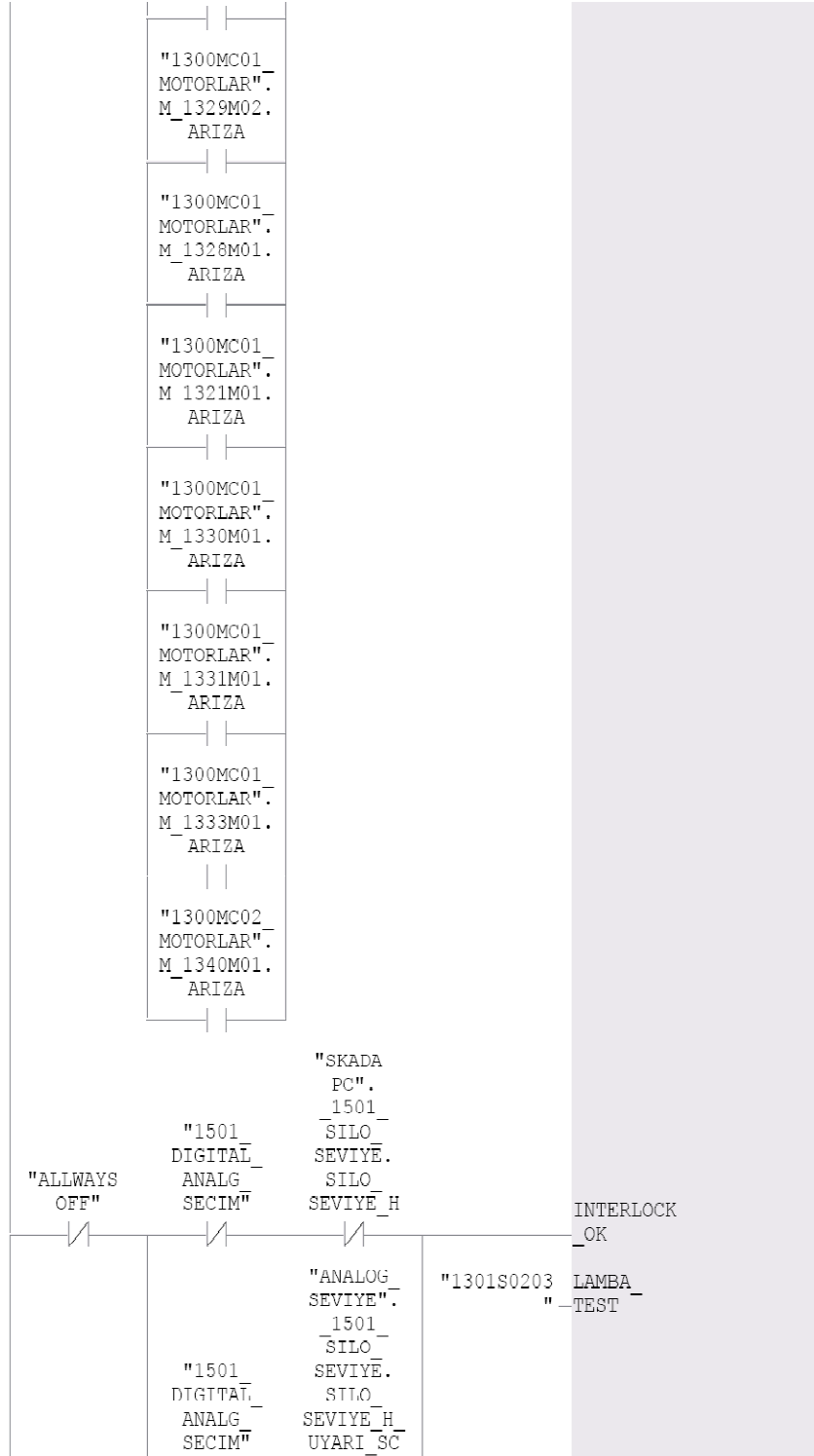
Network: 2	1301M11 18.5KW FC Discharge screw 1 Motoru Kontrolü
1301SS11.2 seçim anahtarı DRIVER pozisyonunda iken 1301M11 motoru FC (MICROMASTER) çalışıyor.	



C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)



C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)



C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

```
GERICALIS - GERICALIS
"1300MC01
MOTORLAR":
M_1301M11.
  FAULT - DRV_FAULT

"1300MC01
MOTORLAR":
M_1301M11. M
CALISMADI - CALISMADI
  SC - SC

"1300MC01
MOTORLAR":
M_1301M11.
DEVIRBEKCI - DEVIRBEKCI
  SI_SC - ISI_SC

"1300MC01
MOTORLAR":
M_1301M11.
  TERMİK_SC - TERMİK_SC

"1300MC01
MOTORLAR":
M_1301M11.
TERMİSTOR - TERMİSTOR
  SC - SC

"1300MC01
MOTORLAR":
M_1301M11.
EMNSALTERI
  _SC - EMSALT_SC

"1300MC01
MOTORLAR":
M_1301M11.
CALİSTIRMA - CALİSTIRM
  _ZAMANI - A_ZAMANI

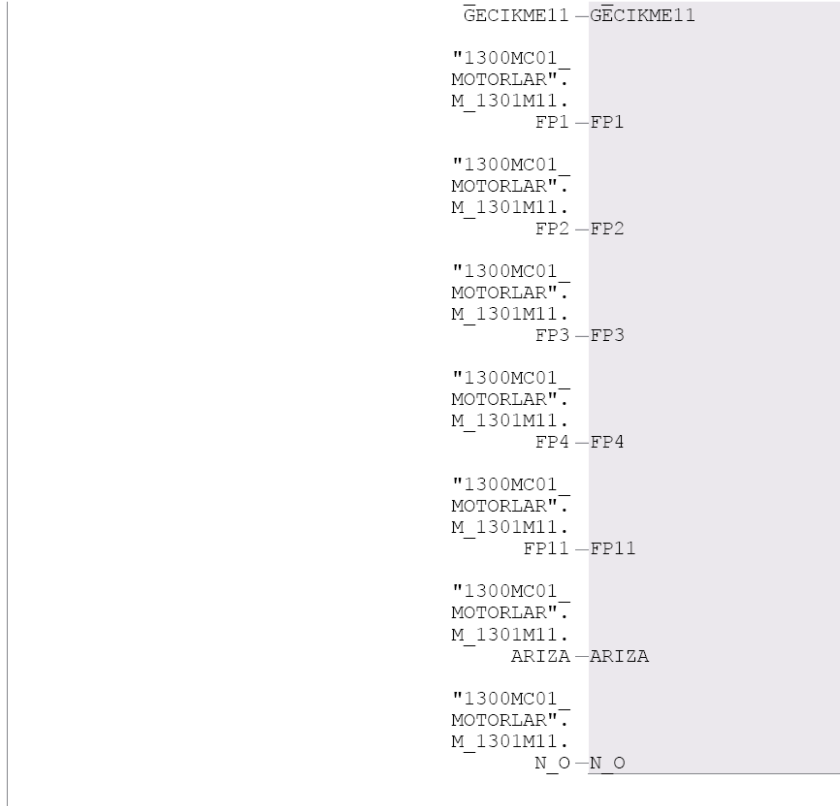
"1300MC01
MOTORLAR":
M_1301M11.
  DÜRDURMA - DÜRDURMA
  ZAMANI - ZAMANI

"1300MC01
MOTORLAR":
M_1301M11. M
  GECİKME1 - GECİKME1

"1300MC01
MOTORLAR":
M_1301M11. M
  GECİKME2 - GECİKME2

"1300MC01
MOTORLAR":
M_1301M11. M
```

C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)



Şekil C10: FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu) blok dış yapısı

C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

Name	Data Type	Address	Comment
IN		0.0	
PIW	Word	0.0	Address for reading
DB_NR	Int	2.0	Data block number for each different motor
PQW	Word	4.0	Address for writing
SETPOINT	Real	6.0	Setpoint for motor kontrol
QSP	Bool	10.0	Quickstop on
SISTEM_MAN	Bool	10.1	=1 MANUAL (LP BLOĞUNUN LOJİK SONUCU)
SISTEM_OTO	Bool	10.2	=1 OTOMATİK (LP BLOĞUNUN LOJİK SONUCU)
OTO_ILERISTART	Bool	10.3	=1 ILERİ OTOSTART
OTO_GERISTART	Bool	10.4	=1 GERİ OTOSTART
BUTON_ILERISTART	Bool	10.5	=1 PUSH START ILERİ
BUTON_GERISTART	Bool	10.6	=1 PUSH START GERİ
BUTON_STOP	Bool	10.7	=0 PUSH STOP
ACILSTOP	Bool	11.0	=0 ACILSTOP OKEY (=1 ARIZA DURUMU)
TERMIK	Bool	11.1	=1 TERMİK ARIZA
DEVIRBEKCISI	Bool	11.2	=1 DEVİRBEKÇİSİ ARIZA
EMNIYET_SALTERI	Bool	11.3	=1 EMNİYET ŞALTERİ ARIZA
TERMISTOR	Bool	11.4	=1 TERMİSTÖR ARIZA
GENEL_ARIZA	Bool	11.5	=1 GENEL ARIZA
M_DRVKONTAKTOR_CEKILI	Bool	11.6	=1 KONTAKTÖR İLERİ ÇEKTİ BİLGİSİ
INTERLOCK_OK	Bool	11.7	=1 OLMASI GEREKEN GENEL KOŞULLAR TAMAM
LAMBA_TEST	Bool	12.0	=1 PUSH LAMBA TEST
RESET	Bool	12.1	=1 PUSH RESET
MAX_SURE	Int	14.0	KONTAKTOR ÇALIŞMA SİYALİNİN GELMESİ GEREKEN SÜRE
DB_KALKIS_SURESI	Int	16.0	DEVİRBEKÇİSİ SİYALİNİN MOTOR KALKTIKTAN SONRA GELMESİ GEREKEN SÜRE
SC_CALISTIRMA_SET_ZAMANI	Int	18.0	OTOMATİK ÇALIŞTIRMADA SET EDİLEN KALKIŞ SÜRESİ

C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

Name	Data Type	Address	Comment
SC_DURDURMA_SET_ZAMANI	Int	20.0	OTOMATİK ÇALIŞTIRMADA SET EDİLEN DURUŞ SÜRESİ
OUT		0.0	
STATUS_1	Word	22.0	Status Word 1
CONTROL	Word	24.0	
READY	Bool	26.0	Frequency Converter ready
ILERI_CLSYR_ARZ_LAMBASI	Bool	26.1	ARIZA ILERI ÇALIŞIYOR LAMBASI(=1 MOTOR ÇALIŞIYOR,=1-0 BLING MOTOR ARIZA)
GERI_CLSYR_ARZ_LAMBASI	Bool	26.2	ARIZA GERI ÇALIŞIYOR LAMBASI(-1 MOTOR ÇALIŞIYOR,-1-0 BLING MOTOR ARIZA)
M_ILERICALISTI_SC	Bool	26.3	MOTOR ILERI ÇALIŞTI SCADA SINYALI
M_GERICALISTI_SC	Bool	26.4	MOTOR GERI ÇALIŞTI SCADA SINYALI
IN_OUT		0.0	
DRV_CALIS	Bool	28.0	DRIVER ÇALIŞ BİTİ
DRV_ILERICALIS	Bool	28.1	
DRV_GERICALIS	Bool	28.2	
DRV_FAULT	Bool	28.3	Frequency Converter fault
M_CALISMADI_SC	Bool	28.4	MOTOR ÇALIŞMADI SKADA SINYALI
DEVIRBEKCISI_SC	Bool	28.5	DEVİRBEKCİSİ SKADA SINYALI
TERMIK_SC	Bool	28.6	TERMİK SKADA SINYALI
TERMISTOR_SC	Bool	28.7	TERMİSTÖR SKADA SINYALI
EMSALT_SC	Bool	29.0	EMNİYET ŞALTERİ SKADA SINYALI
CALISTIRMA_ZAMANI	Int	30.0	OTOMATİK ÇALIŞMADA MOTOR START GECİKME ZAMANI
DURDURMA_ZAMANI	Int	32.0	OTOMATİK ÇALIŞMADA MOTOR STOP GECİKME ZAMANI
M_GECIKME1	Int	34.0	START GELDI FAKAT KONTAKTOR ÇEKMEDİ DURUMU İÇİN GEÇEN SÜRE
M_GECIKME2	Int	36.0	START GELDI FAKAT DEVİRBEKCİSİ GELMEDİ DURUMU İÇİN GEÇEN SÜRE
M_GECIKME11	Int	38.0	START GELDI FAKAT KONTAKTOR ÇEKMEDİ DURUMU İÇİN GEÇEN SÜRE
FP1	Bool	40.0	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
FP2	Bool	40.1	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
FP3	Bool	40.2	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
FP4	Bool	40.3	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)
FP11	Bool	40.4	YÜKSELEN KENAR TETİKLEYİCİ İÇİN (PULSE BIT)

C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

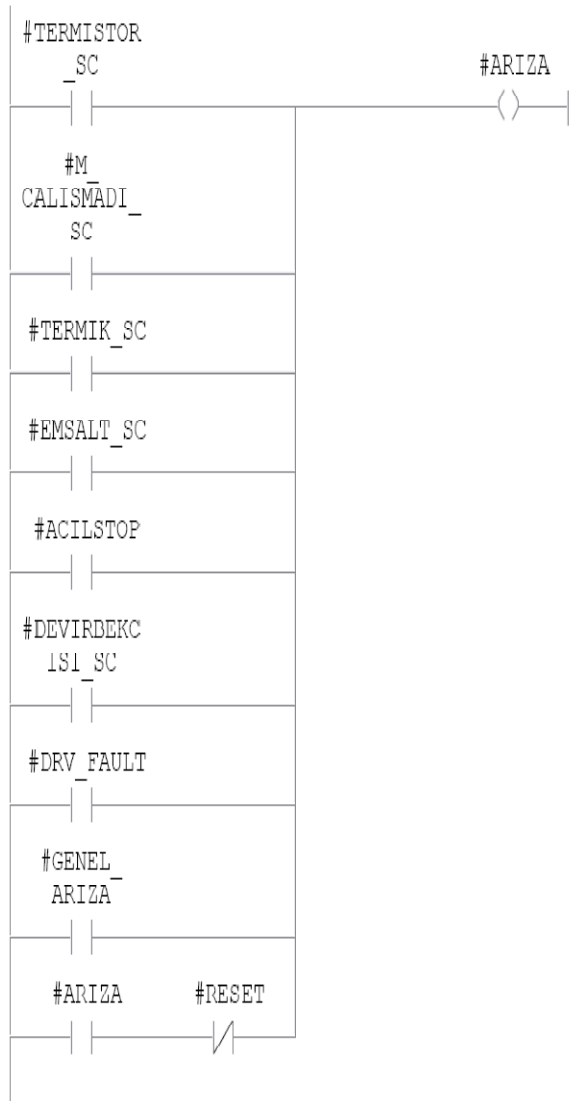
Name	Data Type	Address	Comment
ARIZA	Bool	40.5	TÜM ARIZALARIN GENEL SONUCU
N_0	Bool	40.6	Actual speed > 0
TEMP		0.0	
HW_ANY	Any	0.0	Temp Any
HW_INT	Int	10.0	Temp Integer
Temp_DB_NR	Word	12.0	
TEMP_STATUS_1	Word	14.0	
TEMP_STATUS_2	Word	16.0	
TEMP_CONTROL	Word	18.0	
TEMP_N	Word	20.0	
TEMP_I	Word	22.0	
TEMP_SETPOINT	Word	24.0	
TEMP_REAL	Real	26.0	
OUT_START	Bool	30.0	OTOMATİK START GECİKMESİ
OUT_STOP	Bool	30.1	OTOMATİK STOP GECİKMESİ
STATUS_2	Word	32.0	Status Word 2
N_ACTUAL	Real	34.0	Actual speed (1.....36000 1/min)
I_ACTUAL	Real	38.0	Actual current
RETURN		0.0	
RET_VAL		0.0	

Şekil C11: FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu) deklarasyon tablosu

C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

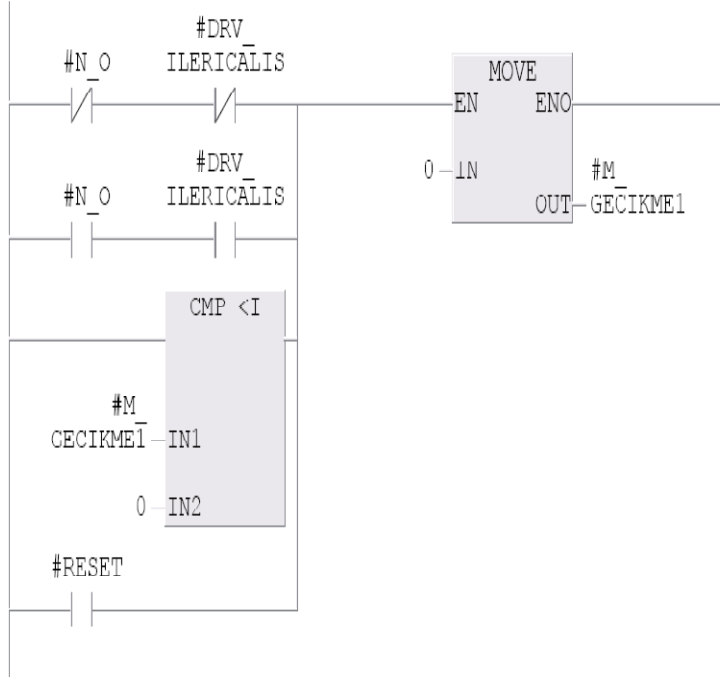
Block: FC19
ÇİFT YÖN MOTOR START-STOP BUTON'LU VE MOTOR SÜRÜCÜLÜ YOL VERME DEVRESİ

Network: 1	ARIZA
------------	-------



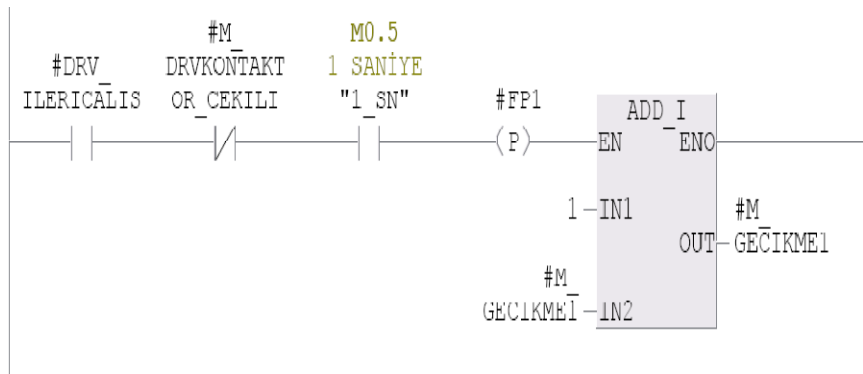
C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

Network: 2 MOTOR GECİKME-1 SAYACININ SIFIRLANMASI



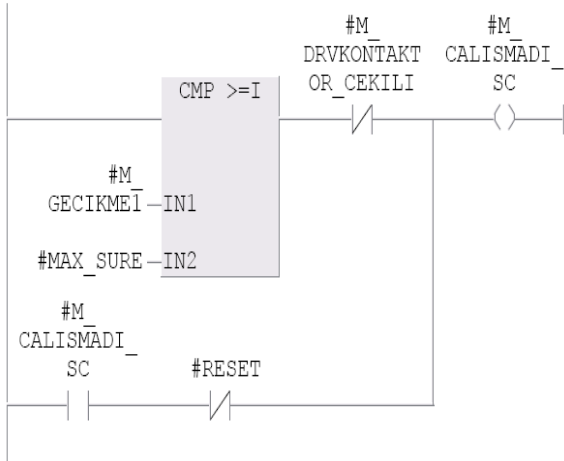
Network: 3 MOTOR GECİKME-1 SAYACI

MOTOR CALISMADI BİLGİSİNİ KONTAKTÖR'DEN ALIP SÜREYİ SAYMAYA BAŞLARIZ.

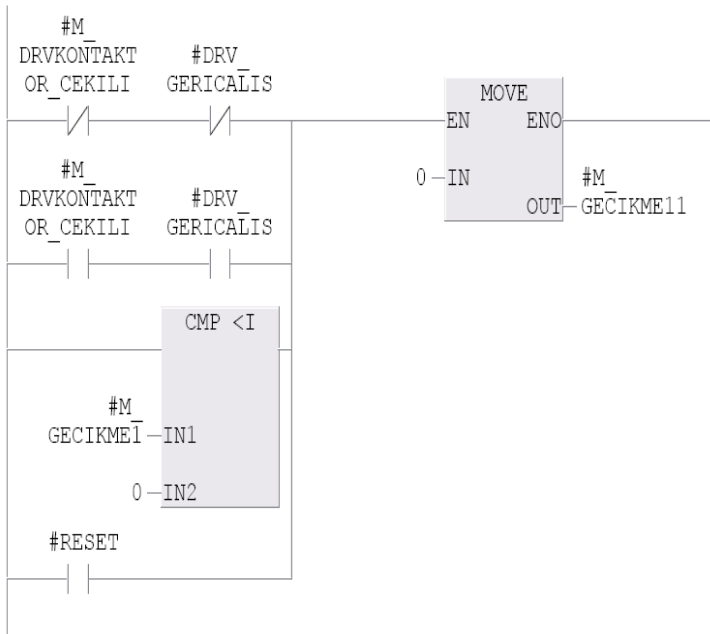


C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

Network: 4
MOTOR ÇALIŞMADI BİLGİSİ KONTAKTÖRDEN İSE = MOTOR ÇALIŞMADI HATASI



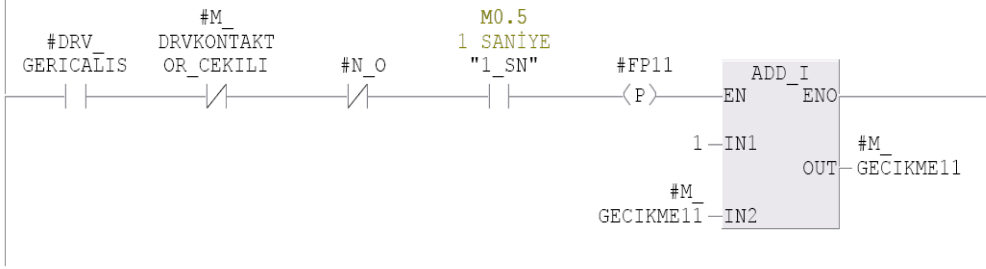
Network: 5 MOTOR GECİKME-1 SAYACININ SIFIRLANMASI



C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

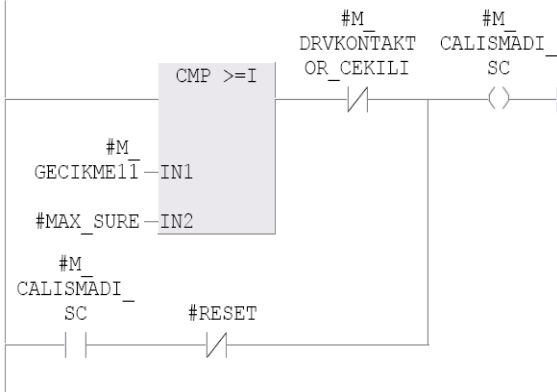
Network: 6 MOTOR GECİKME-1 SAYACI

MOTOR ÇALIŞMADI BİLGİSİNİ KONTAKTÖR'DEN ALIP SÜREYİ SAYMAYA BAŞLARIZ.

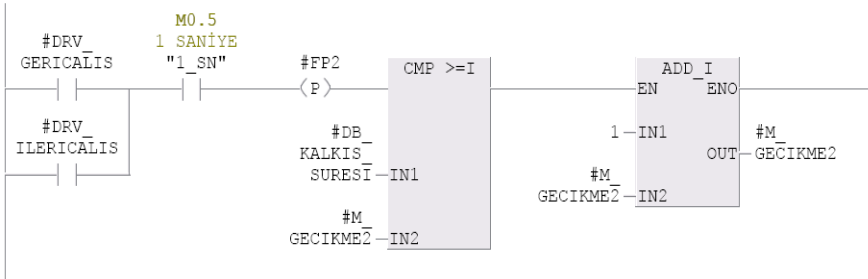


Network: 7

MOTOR ÇALIŞMADI BİLGİSİ KONTAKTÖRDEN İSE = MOTOR ÇALIŞMADI HATASI

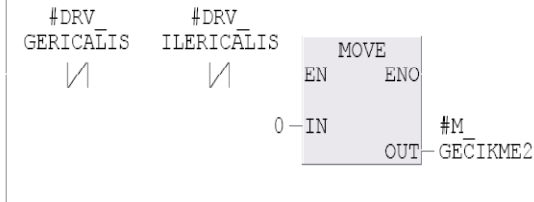


Network: 8 MOTOR GECİKME-2 SAYACI



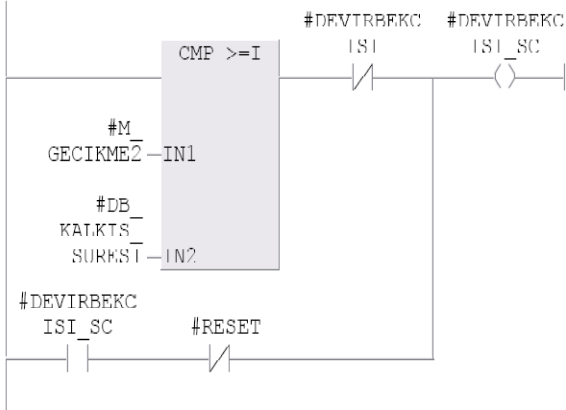
C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

Network: 9 MOTOR GECİKME-2 SAYACININ SIFIRLANMASI



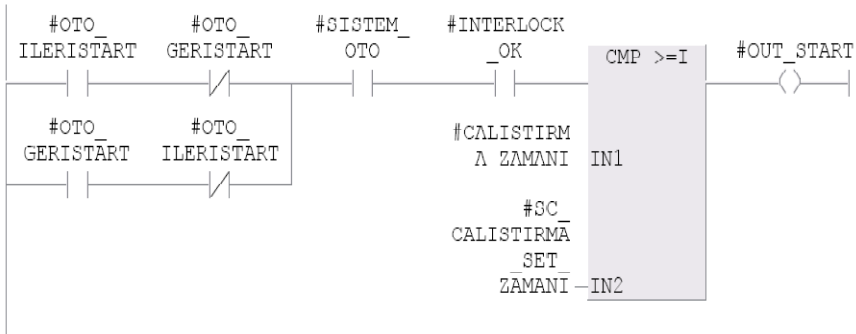
Network: 10

KONTAKTÖR BİLGİSİ VAR DEVİRBEKCİSİ YOK İSE DEVİR BEKÇİSİ ARIZA



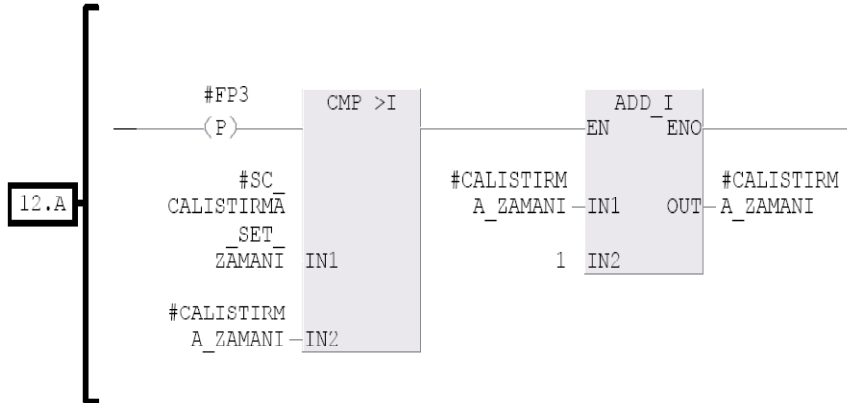
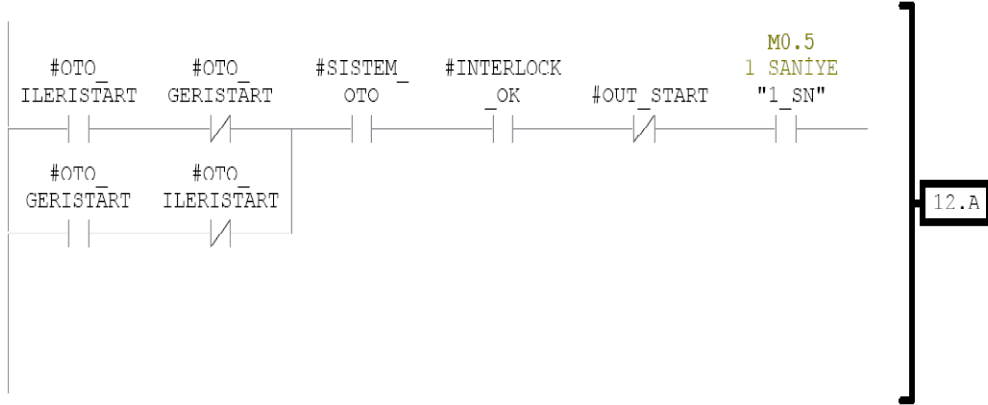
Network: 11 START GECİKMESİ

SİSTEM OTOMATİKTE İKEN START GECİKME ZAMANI

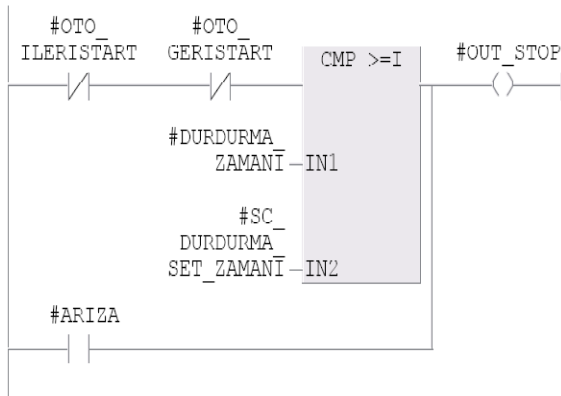


C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

Network: 12 START GECIKMESİ İÇİN SAYMA

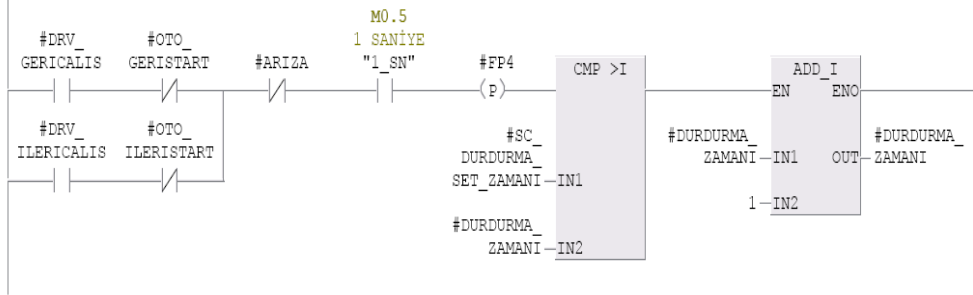


Network: 13 STOP GECİRMESİ

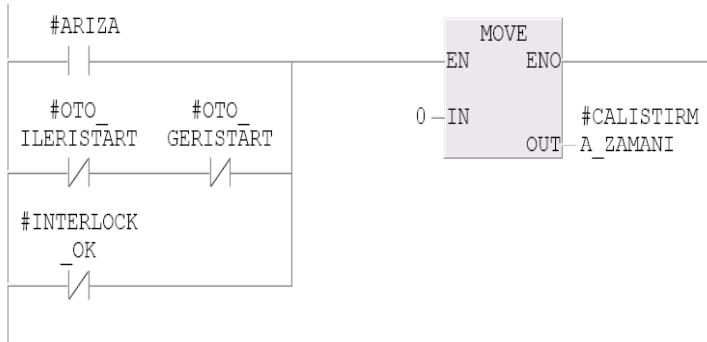


C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

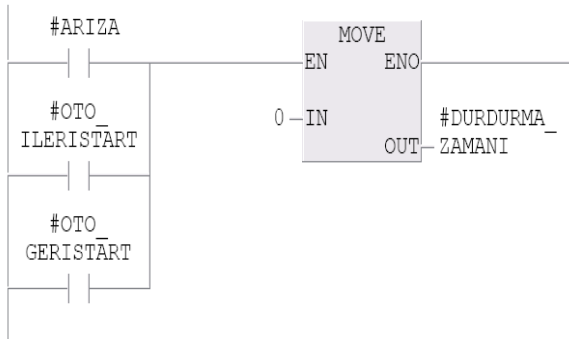
Network: 14 STOP GECİKMESİ İÇİN SAYMA



Network: 15 START ZAMANINI SIFIRLAMA

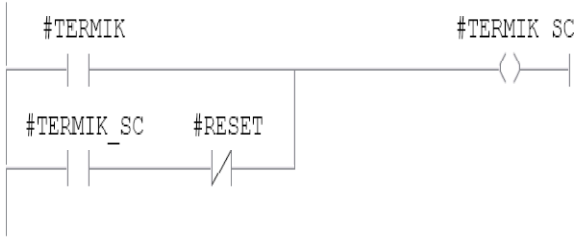


Network: 16 STOP ZAMANINI SIFIRLAMA

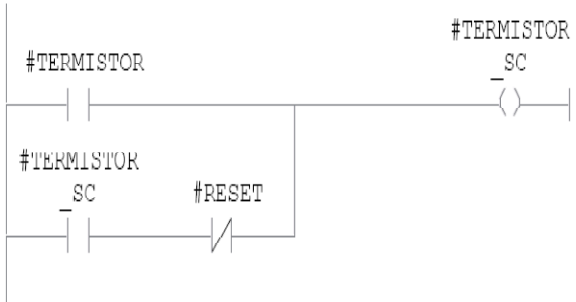


C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

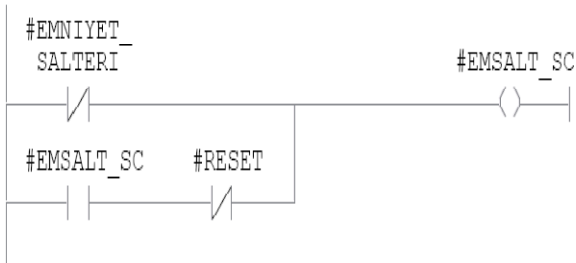
Network: 17	1 SANİYE SAYACI
TERMİK ARIZA	



Network: 18
TERMİSTÖR ARIZA



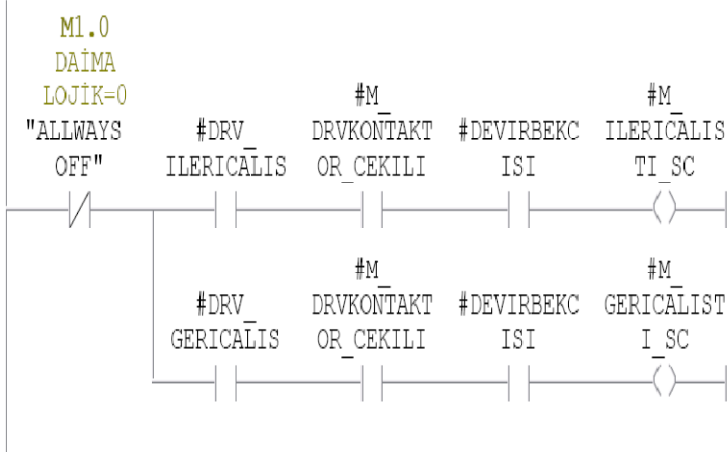
Network: 19
EMNİYET ŞALTERİ ARIZA



C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

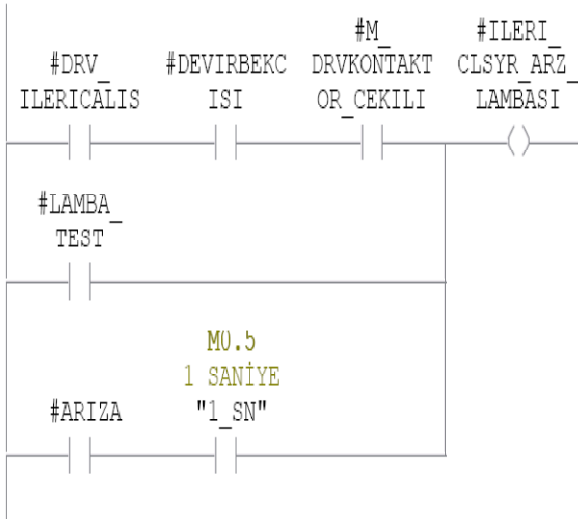
Network: 20

MOTOR ÇALIŞTI SİNYALİNİN SKADAYA AKTARILMASI



Network: 21 ARIZA VE ÇALIŞIYOR LAMBASI

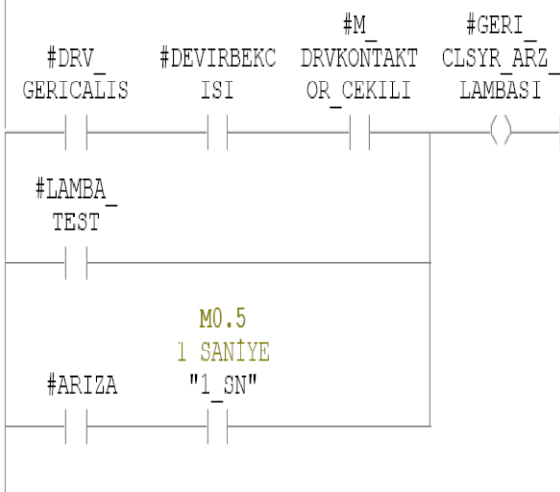
SİSTEM ÇALIŞIYOR İSE LAMBA YANSIN.LAMBA TEST BUTONU BASILI İKEN'DE YANSIN.ARIZA VAR İKEN LAMBA TEST BUTONU'NA BASILSA DAHİ LAMBA SÜREKLİ YANMAMALI.ARIZA GELDİĞİNDE LAMBA BLİNG YAPSIN



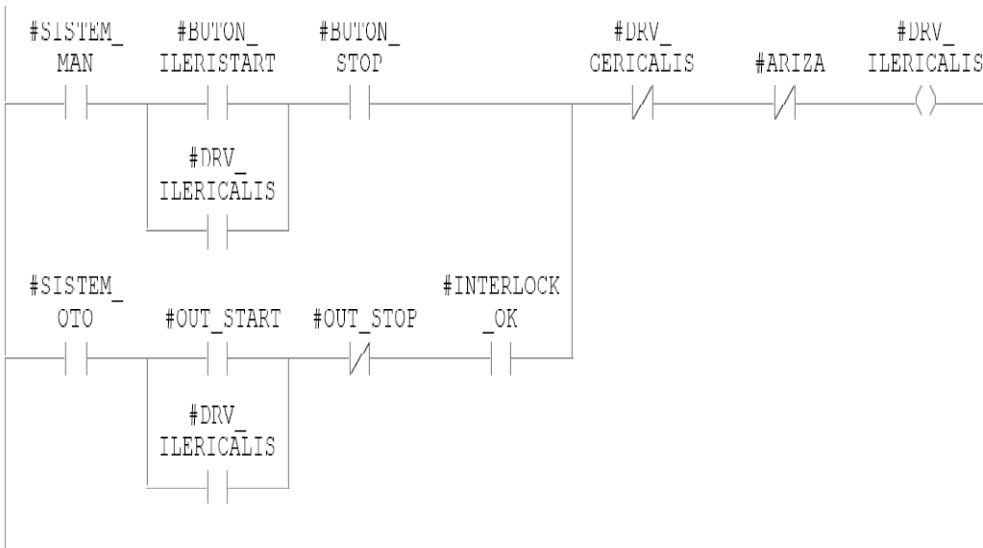
C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

Network: 22 ARIZA VE ÇALIŞIYOR LAMBASI

SİSTEM ÇALIŞIYOR İSE LAMBA YANSIN.LAMBA TEST BUTONU BASILI İKEN'DE YANSIN.ARIZA VAR İKEN LAMBA TEST BUTONU'NA BASILSA DAHI LAMBA SÜREKLİ YANMAMALI.ARIZA GELDİĞİNDE LAMBA BLİNG YAPSIN

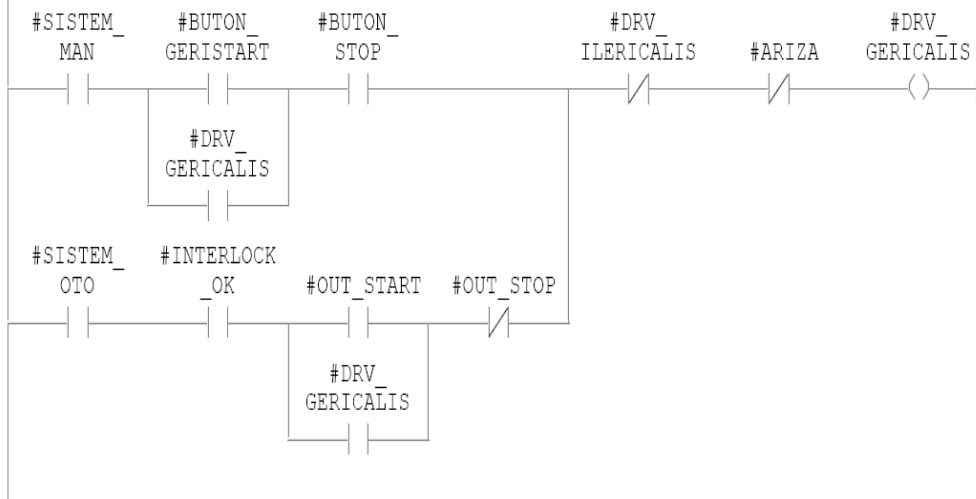


Network: 23 MOTOR START BİTİNİN ÇEKTİRİLMESİ

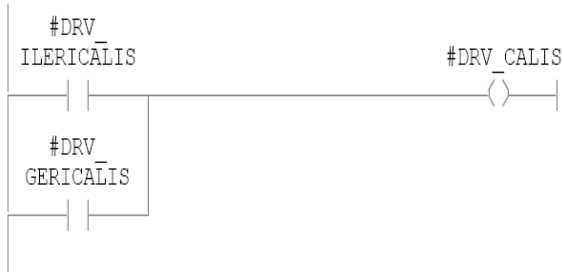


C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

Network: 24 MOTOR START BİTİNİN ÇEKTİRİLMESİ



Network: 25



Network: 26 initialization

```
L 0
T #TEMP_STATUS_1
T #TEMP_STATUS_2
T #TEMP_N
T #TEMP_I
T #TEMP_SETPPOINT
T #TEMP_REAL
```


C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

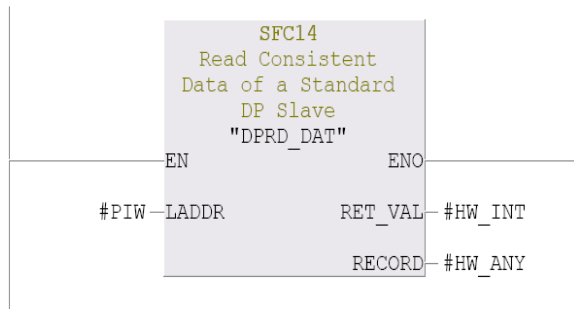
Network: 27

```
L   #DB_NR
T   #Temp_DB_NR
OPN DB [#Temp_DB_NR]
```

Network: 28 Pointer initialization -Data read-

```
L   B#16#10 // S7 Data -HEX- 10h for S7
T   LB      0 //
L   B#16#2  // Data type -HEX-for byte B#16#2
T   LB      1 //
L   12      // Length -INT- repetition
T   LW      2 //
L   #DB_NR  // DB Address -INT-
T   LW      4 //
L   0       // Bit Address -INT-
ITD
SLD 3
T   LD      6 //
L   B#16#84 // Coding of the Memory Areas -HEX- #16#84 for DB
T   LB      6 //
```

Network: 29 Data read from VFD



Network: 30 Status and actual value

```
L   DBW    0 // Status 1
T   #TEMP_STATUS_1

// L   DBW    2 // Status 2
// T   #TEMP_STATUS_2
```

C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

```

L   DEW   2           // Actual value 1   Speed
T   #TEMP_N

L   DEW   4           // Actual value 2   Current
T   #TEMP_I

```

Network: 31	Status
-------------	--------

```

L   #TEMP_STATUS_1 // Status 1
T   #STATÜS_1
L   #TEMP_STATUS_2
T   #STATÜS_2      // Status 2

L   #TEMP_N
ITD
DTR
L   1.638400e+002 // 16384 => 100% Speed
/R
ABS
T   #N_ACTUAL      // Actual speed

L   #TEMP_I
ITD
DTR
L   1.638400e+002 // 16384 => 100% power
/R
T   #I_ACTUAL      // Actual current   (1.....36000 1/min)

// O   L   15.0           ready to switch on
// O   L   15.2           run (voltage at output terminals)
AN  L   15.3           // fault active
=   #READY            // Frequency Converter ready

A   L   15.3
AN  L   15.0           // ready to switch on:1
=   #DRV_FAULT        // Frequency Converter fault

A   L   15.2           // comparison value reached:1
=   #N_O              // Actual speed > 0

```

Network: 32	Control Bits to Frequency Converter
-------------	-------------------------------------

```

//----- Reset Control Word -----
L   2#10000000000    // BIT 10 always 1!
T   #TEMP_CONTROL    // 1 = Converter with Profi-BUS

//----- Motor on/off -----
A   #QSP             // Quick stop on
JC  QSP
A   #DRV_CALIS       // Enable Motor on
JC  RUN
JU  HALT             // Standby

```

C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

```
//----- Control Bits -----
QSP: A L 14.2 // Actual speed > 0
JC QSTP
T 2#1000010001111010 //
T #TEMP_CONTROL
JU RES
QSTP: L 2#1000010001111011 // Control bits Quickstop
T #TEMP_CONTROL
JU RES

RUN: T 2#1000010001111111 // Control bits Enable motor on
T #TEMP_CONTROL
JU RES

HALT: L 2#1000010001111110 // Control bits Standby (Stop)
T #TEMP_CONTROL

RFS: A #RFSFT // Control bit Reset Fault
= L 19.7

AN #DRV_GERICALIS // Control bit Forward
A #DRV_ILERICALIS
- L 18.4
A #DRV_GERICALIS // Control bit Reverse
AN #DRV_ILERICALIS
- L 18.3

// for monitoring
T #TEMP_CONTROL
T #CONTROL
```

Network: 33	Setpoint control and norm
-------------	---------------------------

```
//----- Setpoint control -----

L #SETPOINT
T #TEMP_REAL

L 1.000000e+002 // max. 100%
<R
JC neg
T #TEMP_REAL
JU sw
neg: L #SETPOINT
L 0 // min. 0%
>-R
JC sw
T #TEMP_REAL

// Setpoint norm
sw: L #TEMP_REAL
L 1.638400e+002 // 16384 = 100% speed
*R

RND
T #TEMP_SETPOINT
```

C4. (Devam) FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu)

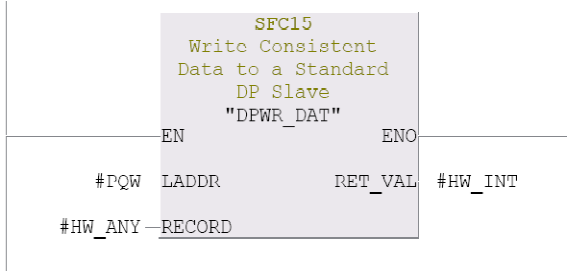
Network: 34 Control Bits and Setpoint to Frequency Converter

```
L #TEMP_CONTROL
T DBW 28
L #TEMP_SETPOINT
T DBW 30
```

Network: 35 Pointer initialization -Data send-

```
L 28 //Bit address -INT-
ITD
SLD 3
T LD 6
L B#16#84 //Coding of the Memory Areas -HEX-
T LB 6
```

Network: 36 Data write to VFD



Network: 37

```
SET
SAVE
```

Şekil C12: FC FC (Çift Yönlü Direkt ve Motor Sürücüsüyle Yol Alan Motor Devresi Bloğu) blok içyapısı

ÖZGEÇMİŞ

1981 yılında Kütahya'nın Emet ilçesinde doğdu. 1992 yılında Çamlıca Mahallesi İlkokulu, 1995 yılında Bağlık Mahallesi İlköğretim Okulu (Orta Okul Kısım), 1999 yılında Tavşanlı Teknik Lisesinden okul birincisi olarak mezun oldu. 2001-2006 yılları arasında Kocaeli Üniversitesi Teknik Eğitim Fakültesi Elektrik Öğretmenliği bölümünden lisans eğitimini sınıf birincisi olarak tamamladı. 2006'da mezun olduktan sonra Kocaeli Üniversitesi bünyesindeki Leonardo Da Vinci projesi ile Almanya'nın Mannheim kentinde bir aylık Siemens PLC kursunu tamamladı. 2006-2008 yılları arasında EREL GRUP firmasında otomasyon sorumlusu olarak çalıştı. Haziran 2009'dan beri EGDAŞ firmasında proje yöneticisi olarak görev yapmakta olup evlidir.