

T.C.

GEBZE YÜKSEK TEKNOLOJİ ENSTİTÜSÜ
MÜHENDİSLİK VE FEN BİLİMLERİ ENSTİTÜSÜ



84716

KONTROLÖR ALAN AĞI ÜZERİNDE
DAĞITIK VERİ TOPLAMA VE KONTROL
SİSTEMLERİ

Nilüfer YANIK
YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ
ANABİLİM DALI

T.C. YÜKSEKÖĞRETİM KURULU
DOKÜMANTASYON MERKEZİ

TEZ DANIŞMANI
Yrd.Doç.Dr. E. Murat ESİN

GEBZE

1999

Bu tez çalışması, G.Y.T.E Mühendislik ve Fen Bilimleri Enstitüsü Yönetim Kurulu'nun ..10.8..1999.. tarih ve ..99../.22.. sayılı kararıyla oluşturulan jüri tarafından **Bilgisayar Mühendisliği** Anabilim Dalında YÜKSEK LİSANS tezi olarak kabul edilmiştir.



JÜRI

ÜYE : Yrd.Doç.Dr. E. Murat ESİN
(Tez Danışmanı)

Murat Esin

ÜYE : Yrd.Doç.Dr. İbrahim SOĞUKPINAR

Ibrahim Soğukpınar

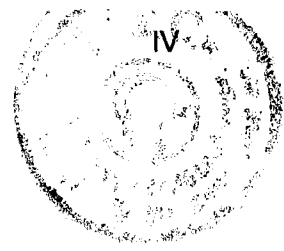
ÜYE : Yrd.Doç.Dr. Ertuğrul KARAÇUHA

Ertuğrul Karachuha

ONAY

G.Y.T.E. Mühendislik ve Fen Bilimleri Enstitüsü Yönetim Kurulu'nun ..01..11..1999.. tarih ve 99../.33.. sayılı kararı.





ÖZET

Bu tezde, veri toplama ve kontrol sistemlerinin ne olduđu, çeşitleri ve özellikleri incelenmiştir. Ek olarak, bilgisayarların bu sistemlerdeki yeri anlatılmıştır.

SCADA sistemlerinden genel olarak bahsedildikten sonra, dağıtık veri toplama ve kontrol sistemlerinde ve SCADA sistemlerinde neden ağ teknolojilerinin kullanıldığı araştırılmıştır.

Endüstride kullanılan popüler ağ teknolojilerinden biri olan Kontrolör Alan Ağı (CAN), onun özellikleri, çalışması ve uygulama alanları incelenmiştir.

Tüm yönleri ile incelenen Kontrolör Alan Ağı, diğer ağ teknolojileri ile karşılaştırılmıştır.

CAN üzerinde bir dağıtık veri toplama ve kontrol sistemi tasarlanmış ve özellikleri açıklanmıştır.



SUMMARY

In this thesis, data acquisition and control systems, their types and features were investigated. In addition, position of the computers in these systems was explained.

After a brief description was given about Supervisory Control and Data Acquisition (SCADA) systems, why network technologies are used in distributed data acquisition and control systems and SCADA systems was studied.

Controller Area Network (CAN) that is one of the popular network technology used in the industry, its specifications and application areas were investigated.

Controller Area Network that was investigated in all features was compared with other network technologies.

A distributed data acquisition and control system on CAN was designed and its specifications were explained.



TEŞEKKÜR

Bu tez konusunu seçmemde bana yardımcı olan, çalışmalarımda beni yönlendiren, tecrübelerinden faydalandığım kıymetli hocam sayın Yrd.Doç.Dr. Emin Murat ESİN'e,

Her zaman daha iyisini yapmamıza çabalayan ve bu yönde yol gösteren değerli hocam sayın Yrd.Doç.Dr. İbrahim SOĞUKPINAR'a,

Çalışmalarımı gerçekleştirirken bana sağladıkları imkan ve huzurlu aile ortamı için Gebze Yüksek Teknoloji Enstitüsü'ne ve tüm çalışma arkadaşlarıma,

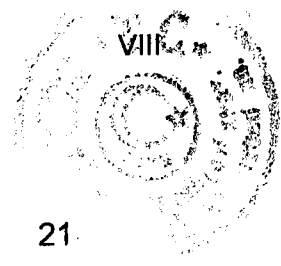
Enstitüye geldiğimde ilk karşılaştığım, güleryüz ve yardımseverliği ile burayı seçmemde vesile olan ve o günden bu yana hiç değişmeyen yardımseverliği için Mühendislik Fakültesi Dekan Yardımcısı sayın Yrd.Doç.Dr. Salim ÖNCEL'e,

Manevi katkılarından dolayı aileme ve emeği geçen herkese teşekkürlerimi bir borç bilirim.



İÇİNDEKİLER

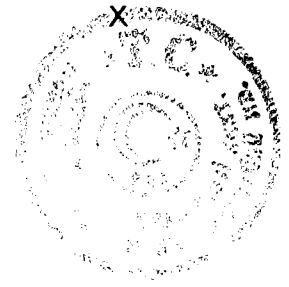
	Sayfa
ÖZET	IV
SUMMARY	V
TEŞEKKÜR	VI
İÇİNDEKİLER DİZİNİ	VII
SİMGELER VE KISALTMALAR DİZİNİ	X
ŞEKİLLER DİZİNİ	XI
ÇİZELGELER DİZİNİ	XIII
1. GİRİŞ	1
2. VERİ TOPLAMA VE KONTROL	2
2.1. Giriş	2
2.2. Mevcut Kontrol Araçları	3
2.3. Kontrol Yöntemleri: Açık Döngü ve Kapalı Döngü	4
2.4. Veri Toplama ve Kontrol Sistemi Elemanları	6
2.4.1. Algılayıcılar ve Eylemciler	7
2.4.2. Sinyal Durumlama	6
2.4.3. Veri Toplama ve Kontrol Donanımı	8
2.4.3.1. Örneksele Girişler (A/D)	8
2.4.3.2. Örneksele Çıkışlar (D/A)	9
2.4.3.3. Sayısal Girişler ve Çıkışlar	9
2.4.3.4. Sayaç/Zamanlayıcı	10
2.4.4. Yazılım	10
3. PC TABANLI KONTROL	12
3.1. Seri Port	13
3.1.1. Genel Asenkron Alıcı İletici (UART)	15
3.1.1.1. Tutma Yazmaçları	16
3.1.1.2. Hat Kontrol Yazmacı	16
3.1.1.3. Hat Durumu Yazmacı	18
3.1.1.4. Kesme Yazmaçları	18
3.1.1.5. Modem Yazmaçları	20



3.1.2. 8250 I/O Arabiriminin Programlanması	21
4. SCADA SİSTEMİ	23
4.1. Giriş	23
4.2. SCADA Sisteminin Özellikleri	23
4.3. SCADA Sistemlerinde Haberleşme	26
5. KONTROLÖR ALAN AĞI	28
5.1. Giriş	28
5.2. Kontrolör Alan Ağı (CAN)	31
5.3. CAN Standartları	32
5.3.1. A Bölümü ve B Bölümü Uyumluluğu	32
5.4. CAN'ın Çalışması	33
5.4.1. Prensip	34
5.4.2. Belirteçler ve Hakemlik	34
5.4.3. Uzak Çerçeveler	36
5.5. CAN Mesaj Formatları	37
5.5.1. 2.0A Formatı	37
5.5.2. 2.0B Formatı	38
5.6. Hata Bulma ve Faul Sınırlama	39
5.6.1. CAN Hata İşlemi	40
5.6.2. Hata Bulma	40
5.6.3. CAN Kontrolörü Hata Modları	41
5.6.4. Hata Sinyalleme	41
5.7. Bit Zamanlama	42
5.8. OSI ve CAN Modellerinin Karşılaştırılması	43
5.8.1. OSI Modeli	43
5.8.2. CAN ve OSI Modeli	45
5.8.3. CAN Veri Yolu Fiziksel Katman	46
5.8.3.1. ISO 11898	46
5.8.3.2. ISO 11519	47
5.8.3.3. Veri Yolu Uzunlukları	47
5.8.3.4. İletim Ortamı	48
5.9. CAN Uygulama Katmanları	48



5.9.1. CANOpen	48
5.9.1.1. ISO Katman Modeli ve CANOpen	48
5.9.1.2. İletişim Modeli	49
5.9.1.3. Veri Transfer Mekanizmaları	50
5.9.1.4. Ağ Yönetimi	50
5.9.1.5. Veri Takas Mekanizmaları	51
5.9.2. DeviceNet	52
5.9.2.1. Üretici/Tüketici Modeli	53
5.9.2.2. Mesaj Grupları	53
5.9.3. SDS (Smart Distributed System)	54
5.9.4. CANKingdom	54
5.10. CAN Uygulamaları	56
5.10.1. Farklı Uygulamalar- Temel CAN ve Tam CAN	56
5.10.2. Temel CAN	56
5.10.3. Tam CAN	57
5.11. Mevcut Standartlar ve CAN	59
5.12. CAN Protokolü ile Endüstriyel Uygulamalar	60
5.13. CAN ve Diğer Fieldbus'ların Karşılaştırılması	61
6. CAN ÜRÜNLERİ	62
7. CAN ÜZERİNDE DAĞITIK VERİ TOPLAMA VE KONTROL	66
SİSTEMİ TASARIMI	
7.1. Bir Tasarım Örneği	67
7.1.1. Donanımın Özellikleri ve Bağlantılar	71
7.1.2. Kullanılan Donanım	72
7.1.3. Yazılım	79
8. SONUÇ	89
KAYNAKLAR	90
ÖZGEÇMİŞ	92
EKLER	
Ek 1. Seri Port Üzerinden Haberleşme için C++Builder Programı	
Ek 2. ADAM 5000/CAN DeviceNet Fonksiyonları	



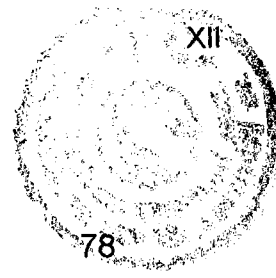
SİMGELER VE KISALTMALAR

CAN	Kontrolör Alan Ağı (Controller Area Network)
CiA	"Otomasyonda CAN" Organizasyonu(CAN in Automation)
SCADA	Veri Tabanlı Kontrol ve Gözetleme Sistemi (Supervisory Control And Data Acquisition)
PLC	Programlanabilen Mantısal Kontrolör (Programmable Logic Controller)
RTD	Direnç Sıcaklık Ölçücü (Resistance Temperature Detector)
A/D	Analog/Dijital
D/A	Dijital/Analog
UART	Genel Asenkron Alıcı İletici (Universal Asynchronous Receiver Transmitter)
LAN	Yerel Alan Ağı (Local Area Network)
WAN	Geniş Alan Ağı (Wide Area Network)
CSMA/CD+AMP	Carrier Sense Multiple Access/Collision Detection with Arbitration on Message Priority
ISO	Uluslar Arası Standart Organizasyonu (International Standard Organization)
OSI	Açık Sistem Bağlantısı (Open System Interconnection)
LLC	Mantıksal Bağlantı Kontrolü (Logical Link Control)
MAC	İletişim Araçları Ulaşım Kontrolü (Media Access Control)
SDO	Servis Veri Nesnesi (Service Data Object)
PDO	Proses Veri Nesnesi (Process Data Object)
SDS	Akıllı Dağıtık Sistem (Smart Distributed System)
LVDT	Lineer Değişken Diferansiyel Transformatör
DIP Switch	Grup ON/OFF Anahtar



ŞEKİLLER DİZİNİ

Şekil	Sayfa
2.1. Açık Döngü Kontrol Yöntemi	5
2.2. Kapalı Döngü Kontrol Yöntemi	5
2.3. Veri Toplama Sisteminin Düzeni	6
2.4. A/D Dönüşüm	8
2.5. D/A Dönüşüm	9
2.6. Bir Vananın Sayısal I/O ile Kontrolü	10
3.1. Seri İletimde Baytın Çerçevesi	14
3.2. DB25 Konnektör	14
5.1. Dağıtık Veri Toplama ve Kontrol Sisteminin Genel Yapısı	28
5.2. Bir Fieldbus Sistemi	29
5.3. CAN Genel Yapısı	33
5.4. Nokta 1, 2 ve 4 Arasındaki Bit Hakemliği	35
5.5. İletim ve Alımda Noktaların Durumları	36
5.6. CAN 2.0A Mesaj Çerçevesi	38
5.7. CAN 2.0B Mesaj Çerçevesi	39
5.8. OSI Katmanları	43
5.9. CAN Katmanları	45
5.10. CANOpen Mimarisi	49
7.1. Kontrol Edilecek Sistemin Genel Yapısı	68
7.2. P1 ve P2'nin Mantıksal Gösterimi	69
7.3. L1, L2 ve I'nın Mantıksal Gösterimi	70
7.4. Sistemin Şematik Gösterimi	70
7.5. ADAM 5000/CAN Modülünün Bilgisayar Bağlantısı	73
7.6. DIP Switch	74
7.7. ADAM 5018 Modülü ve Uygulama Bağlantısı	76
7.8. ADAM 5051 Modülü ve Uygulama Bağlantısı	76
7.9. ADAM 5060 Modülü ve Uygulama Bağlantısı	77
7.10. Pompa	77



7.11. Portatif Termokupl	78
7.12. İletkenlik Tip Seviye Anahtarı	78
7.13. Proses Isıtıcısı	79
7.14. Programın Genel Diyagramı	79
7.15. Cihaz Tanımlama Programının İş Akış Şeması	80
7.16. Cihaz Test Alt Programının İş Akış Şeması	81
7.17. Sistemin Çalışmasının İş Akış Diyagramı	82
7.18. Kullanıcı Ekranı	83
7.19. I/O Cihaz Tanımlama Penceresi	84
7.20. Haberleşme Portu Ayar Penceresi	84
7.21. Modül Seçim Penceresi	85
7.22. Modül Ekleme Penceresi	85
7.23. Sayısal Giriş Test Ekranı	86
7.24. Tarama Zamanı Penceresi	87
7.25. Sayısal Çıkış Test Ekranı	87
7.26. Örneksele Giriş Test Ekranı	88

ÇİZELGELER DİZİNİ

Çizelge	Sayfa
3.1. DB25 Konnektör Pinlerinin Görevleri	15
3.2. 8250 Yazmaç Adresleri	15
3.3. Kesme Belirtme Yazmacı Fonksiyonları	19
5.1. CAN Kontrolör Tipleri Uyumluluğu	32
5.2. ISO 11898 Özellikleri	46
5.3. ISO 11519 Özellikleri	47
5.4. Veri Yolu Uzunlukları	47
5.5. Veri Transfer Mekanizmalarının Karşılaştırılması	51
5.6. Temel CAN Özellikleri	57
5.7. Tam CAN Özellikleri	59
5.8. Seri Haberleşme Arabirimleri	59
5.9. CANbus ve Profibus'ın Karşılaştırılması	61
6.1. CAN Ürünleri Listesi	63
7.1. Girişler ve Çıkışlar Arasındaki Mantıksal İlişki	68
7.2. Ağ Adresi Set Tablosu	74
7.3. Baudrate Set Tablosu	75
7.4. CAN Portu Pin Bağlantıları	75

1. GİRİŞ

Her türlü prosesin çalışmasında çeşitli seviyelerde veri toplama ve kontrol işlemi yapılır. Teknolojinin ilerlemesiyle birlikte veri toplama ve kontrol sistemlerinde mikro işlemciler, kontrolörler ve bilgisayarlar kullanılmaya başlanmıştır. Geniş alana yayılmış veri toplama adalarını kontrol etmek için ise ağ teknolojileri kullanılmaktadır.

Dağıtık veri toplama sistemlerinin kontrolünde kullanılan ağ teknolojilerinden biri de Kontrolör Alan Ağı'dır. Kontrolör Alan Ağı'nın endüstriyel otomasyonda kullanımı gün geçtikçe artmaktadır ve bununla uyumlu kontrolörler üretilmektedir.

Bu çalışmada, Kontrolör Alan Ağı incelenerek, avantajları ve dezavantajları araştırılıp, endüstriyel proses kontrol uygulamalarında bu teknolojiye faydalanma yollarının gösterilmesi amaçlanmıştır.



2. VERİ TOPLAMA VE KONTROL

2.1. Giriş

Veri toplama ve kontrol işlemi çeşitli formlarda olabilir. En basit seviyede veri toplama, bir insan tarafından kalem ve kağıt kullanılarak voltmetre, ohmmetre, veya daha başka aygıtlardan okunanları kaydederek yapılabilir. Bazı durumlar için bu tip veri toplama uygun olabilir. Bir çok uygulamalarda, özellikle kaydedilen veri noktalarının sayısının düşük olduğu ve okumaların frekansının insanların veriyi kaydetmeleri için yeteri kadar yavaş olduğu durumlarda, bu “düşük teknoloji”, veri kayıt metodu olarak oldukça uygundur.

Bununla birlikte çok sayıda veri okuması gerektiren veri kaydetme uygulamalarında veya çok sık okumaların gerekli olduğu durumlarda, veriyi toplama ve kaydetme için değişik aygıtlara ve bilgisayarlara ihtiyaç vardır. Büyük miktarda veriyi veya hızlı gelen veriyi kaydetmek ancak bu tür aygıtlar sayesinde mümkün olabilir.

Elektriksel aygıtların kontrolü için çeşitli seviyelerde teknoloji kullanılabilir. “Düşük teknoloji” seviyesinde bir basit elektrik anahtarı, bir motoru, konveyör kayışını veya diğer elektrik/mechanik aygıtları kontrol edebilir. Bu sistem, özellikle karmaşık kontrol sistemlerinin uygulanmasının mümkün olmadığı uygulamalarda veya sistem maliyetinin düşük olması istenen durumlarda kullanılabilecek en iyi metottur. Endüstride önemli olan “İleri teknoloji” yi kullanmak değil, “uygun teknoloji” yi kullanmaktır.

Bazı kontrol uygulamaları daha gelişmiş yaklaşım gerektirebilir. Robotları kapsayan endüstriyel ekipmanlar, entegre proses kontrol sistemleri ve konveyör kayışları, gelişmiş sistemlerde ihtiyaç duyulan kontrolü sağlayan bir kişisel bilgisayarın esnek programlama kapasitesine ve işlem gücüne ihtiyaç duyabilir [Rigby and Dalby, 1995].



2.2. Mevcut Kontrol Araçları

Kontrol edilecek sistemlerin çeşitliliğine göre kontrol biçimleri de çeşitlidir. Sistemleri basitten karmaşığa doğru düşünersek kontrol biçimlerini de ona göre katı mantık, programlanabilen mantık kontrolörleri (PLC) ile, bilgisayar tabanlı ve ağ tabanlı diye dörde ayırabiliriz.

Katı mantık kontrol, kontrol edilecek sistemin o sistem için oluşturulan sabit bir elektronik kontrol devresiyle kontrolüne denir. Bu devre sadece o sistemi kontrol etmek için geliştirilmiş özel bir devredir. Bu devreler ilk başlarda basit kapı devrelerden oluşurken sonraları içlerinde program icra edilebilen mikroişlemciler ve hafıza elemanları kullanılmıştır. Bu biçimde kontrol, genelde çok fazla karmaşık olmayan basit proseslerde kullanılır.

PLC, programlanabilen bir kontrolördür. Kontrol edilecek prosesin özelliğine göre bir program yazılır. Başlangıçta mikroişlemcili devrelerle kontrol sağlanmıştır. Zamanla mikroişlemcilerin yerine mikrokontrolör kullanılmaya başlanmıştır. Fakat bunların programlanması hem zaman alıcı hem de zor olmaktaydı, bunların daha esnek kullanımı için yazılımlar geliştirilmiş bu sayede bir başka ara birimle kontrol, belli komutlar ile sağlanmıştır. İlk defa 1950'li yıllarda bir otomobil firması tarafından yaptırılan PLC, daha sonraları hem donanımların hem de yazımların artması sayesinde daha esnek ve hızlı kontrol sağlamıştır.

Bilgisayar tabanlı kontrol sistemleri kullanım şartlarının daha uygun olduğu ve daha görsel amaçların söz konusu olduğu yerlerde kullanılır. Kontrol edilecek proses ile ilgili veriler, seri port veya özel veri toplama ve kontrol kartları kullanılarak bilgisayar üzerinde çalışan kontrol yazılımına aktarılır.

Kontrol edilecek sistemin çok karmaşık olduğu durumlarda ise ağ tabanlı veri toplama ve kontrol sistemleri tercih edilir. Bu sistemde PLC'ler, bilgisayarlar ve diğer kontrol elemanları bir ağ ortamında beraber çalışarak sistemin kontrol edilmesini sağlarlar.

Bu çalışma kapsamında katı mantık ve PLC tabanlı kontrol sistemleri ile ilgilenilmeyeceği için bilgisayar ve ağ tabanlı kontrol sistemleri hakkında daha detaylı bilgi verilecektir.



2.3. Kontrol Yöntemleri: Açık Döngü ve Kapalı Döngü

Proses kontrol uygulamalarında, veri toplama ve kontrol sistemini oluşturan aygıtlar iki farklı konfigürasyonda kullanılabilir. Bu iki farklı tip konfigürasyon sistemde kullanılan kontrolün tipi ile alakalıdır. Bir başka deyişle iki mümkün kontrol sistemi, kullanılan kontrol sisteminde geri besleme döngüsünün olup olmadığıyla irtibatlıdır.

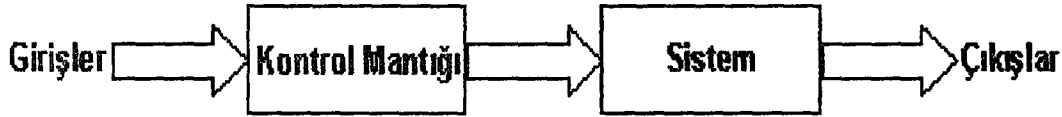
Kontrolün iki yöntemi, açık döngü veya kapalı döngü olarak belirtilir. Örneğin bilgisayar kontrollü bir açık döngü uygulamasında bilgisayarın zaman tutma yeteneğini kullanılmasıyla bir alışveriş merkezi parkındaki ışıkların her gece saat sekizde yakılması sağlanır. Böyle bir uygulamada bilgisayar, günün saatlerini içindeki sistem saatini kullanarak izler. Saat sekiz olduğunda bilgisayar üzerindeki kontrol kartı vasıtasıyla park ışıklarını kontrol eden röleye bir elektrik sinyali gönderir.

Bu tipteki bilgisayar tabanlı kontrol sisteminin avantajı, bilgisayar ve işletim sisteminin yazılım programının kolayca yenilenmesini sağlamasıdır. Örneğin, bu esneklik, gün içerisinde ışıkların açıldığı ve kapatıldığı saatlerin değiştirilebilmesine izin verir. Bilgisayarın esnekliği ayrıca park alanındaki değişik ışıkların değişik zamanlarda açılıp kapatılmasını da sağlar. Mesela bazı ışıkların saat 8 de açılması gerekirken bazılarının da 8.30 da açılması gerekebilir.

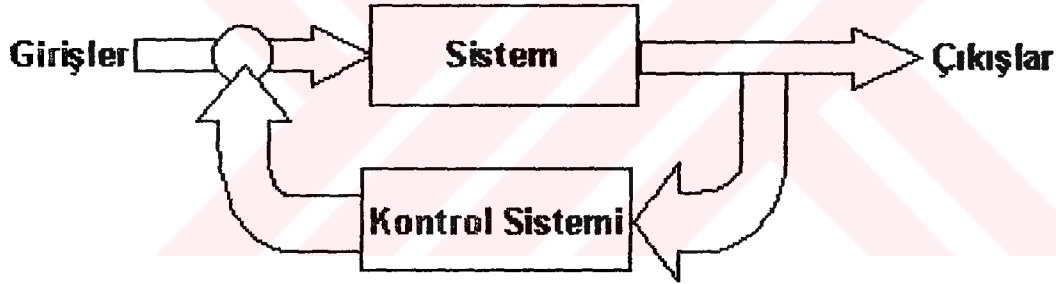
Bu açık döngü uygulaması kendine has sınırlamalara sahiptir. Bunların en önemlilerinden bir tanesi röleye elektrik sinyali gönderildikten sonra ışıkların gerçekten yakılıp yakılmadığı bilgisayar tarafından denetlenememesidir. Açık döngü devreleri, kontrol edilen sisteme ait herhangi bir geri besleme ünitesine sahip değildir.

5

Kontrol sistemine geri besleme olmaması, sistem tasarımcılarını kontrol edilen sistemden geri besleme alacak daha karmaşık düzende bir kontrol devresi düşünmeye yönlendirir. Bir kapalı döngü kontrol devresi, açık döngü kontrol devresiyle aynı kontrol kapasitelerini sağlamakla birlikte gerçek dünyadan bir geri beslemeye veya girişe izin verir. Geri besleme yeteneği kontrol sisteminin işlevini genişletir, böylece sistem sadece belirlenen saatte ışıkları yakmakla sınırlandırılmış değildir. Bu, bir kontrol sistemine ışıkları yakmak için yeteri kadar karanlık olup olmadığını anlayan sensörleri dahil etmeyi sağlar. Geri besleme sistemi, ışıkların bilgisayar tarafından kumanda edildiği gibi yanıp yanmadığını da kanıtlar. Şekil 2.1. açık döngü ve Şekil 2.2. kapalı döngü kontrol yöntemini göstermektedir.



Şekil 2.1. Açık Döngü Kontrol Yöntemi.



Şekil 2.2. Kapalı Döngü Kontrol Yöntemi.

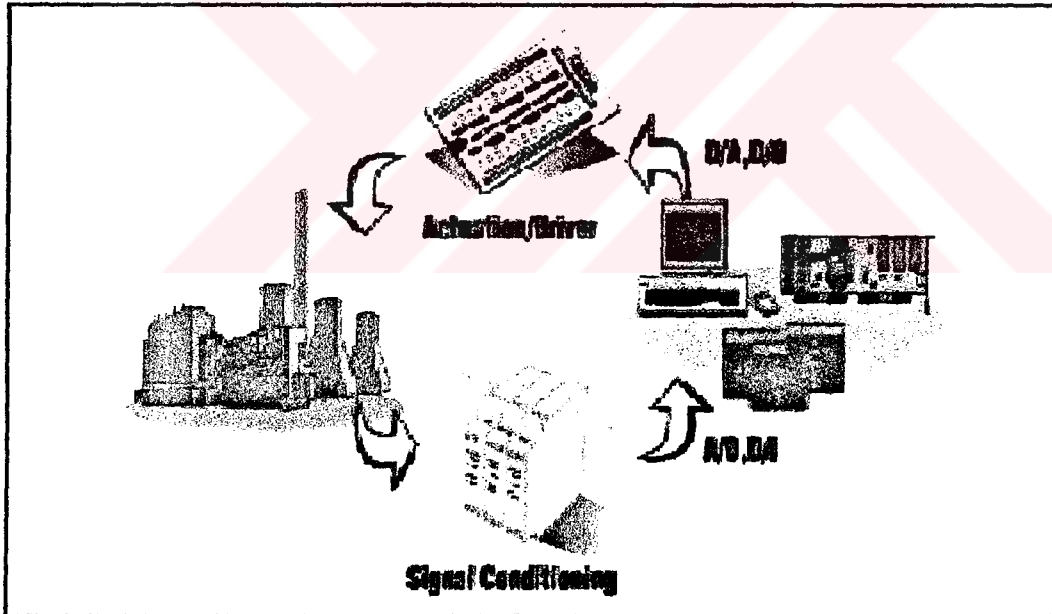
Açık döngü kontrol devreleri bir çok uygulamalara sahiptir ve endüstride yeri vardır. Kontrol edilen devreden veya aygıttan geri beslemenin gerekli olmadığı veya maliyetin etkili olduğu yerlerde basit bir açık döngü devre yeterli olabilir. Diğer yönden, kontrol durumunun denetiminin gerekli olduğu durumlarda kapalı döngü kontrol devresi uygulanmalıdır [Rigby and Dalby, 1995].

2.4. Veri Toplama ve Kontrol Sistemi Elemanları

Veri toplama ve kontrol sistemini seçmek ve inşa etmek için elektrik ve bilgisayar mühendisliğine ait bazı konularda bilgi sahibi olmak gerekir. Veri toplama ve kontrol sistemlerinin en genel parçaları şunlardır:

- Algılayıcılar ve eylemciler.
- Sinyal durumlama.
- Veri toplama ve kontrol donanımı.
- Bilgisayar sistemleri yazılımı.

Şekil 2.3., Tipik bir veri toplama sisteminin düzenini göstermektedir [Advantech,1999].



Şekil 2.3. Veri Toplama Sisteminin Düzeni.



2.4.1. Algılayıcılar ve Eylemciler

Bir algılayıcı, sıcaklık, basınç, seviye, uzunluk, pozisyon gibi değerleri gerilim, akım, frekans gibi sinyallere dönüştürür.

Termokupl, termistör ve direnç sıcaklık ölçücüler (RTD Resistance Temperature Detector), sıcaklık ölçümü için genel algılayıcılardır. Algılayıcıların diğer tipleri akış sensörlerini, basınç sensörlerini, akış hızını, basınç değişimlerini, güç ve mesafeyi ölçen LVDT'leri içerir.

Eylemci, pnomatik, hidrolik veya elektriksel güç kullanarak proses kontrol ekipmanını çalıştıran aygıttır. Örneğin bir vana eylemcisi sıvı hızını kontrol etmek için bir vanayı açar ve kapatır.

2.4.2. Sinyal Durumlama

Sinyal durumlama devreleri, algılayıcılar tarafından üretilen sinyaller PC'nin veri toplama donanımı tarafından sayısal sinyallere dönüştürülmeden önce sinyalin genliğini yükseltir. Sinyal durumlamanın örnekleri, işaret basamaklandırma, yükseltme, doğrusallaştırma, soğuk-birleşme telafisi(cold-junction compensation), süzgeçleme, zayıflatma, ortak-mod reddetme gibi işlemlerdir.

En genel sinyal durumlama fonksiyonlarından biri yükseltmedir. Maksimum çözülme için gelen sinyallerin gerilim aralığı A/D dönüştürücünün maksimum giriş aralığına yaklaşık olarak eşit olmalıdır. Yükseltme algılayıcı sinyallerinin genliğini, böylece A/D dönüştürücünün giriş genliğiyle uyushurlar. Örneğin; girişinde izin verilen gerilim aralığı 0-10V olan bir A/D dönüştürücünün, 0-1V aralığında değişen bir işareti yükek hassaslıkla işlemesi için kazancı 10 olan bir yükselteç devresinden geçirilir.



2.4.3. Veri Toplama ve Kontrol Donanımı

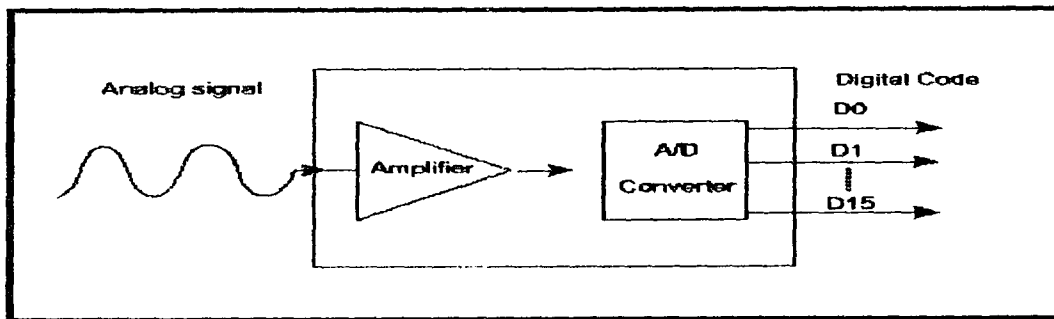
Veri toplama ve kontrol donanımı, genellikle analog giriş, analog çıkış, sayısal giriş, sayısal çıkış ve sayaç/zamanlayıcı fonksiyonlarından bir veya daha çoğunu yapar.

2.4.3.1. Örneksele Girişler (A/D)

Örneksele işareti sayısal işarete dönüştürme, örneksele gerilim veya akım büyüklüğünün sayısal işarete dönüştürülmesidir. Dönüştürme, bilgisayarın sinyalleri işleyebilmesi veya saklayabilmesi için gereklidir. Şekil 2.4., A/D dönüştürme göstermektedir [Advantech, 1999].

A/D donanımını seçerken en önemli kriterler:

1. Giriş kanallarının sayısı.
2. Tek sonlu veya diferansiyel giriş sinyalleri.
3. Örnekleme hızı (saniye bazında).
4. Çözünürlük (genellikle bit olarak ölçülür).
5. Giriş aralığı (tam skala gerilim olarak belirlenir).
6. Gürültü ve doğrusallık karakteristiği.

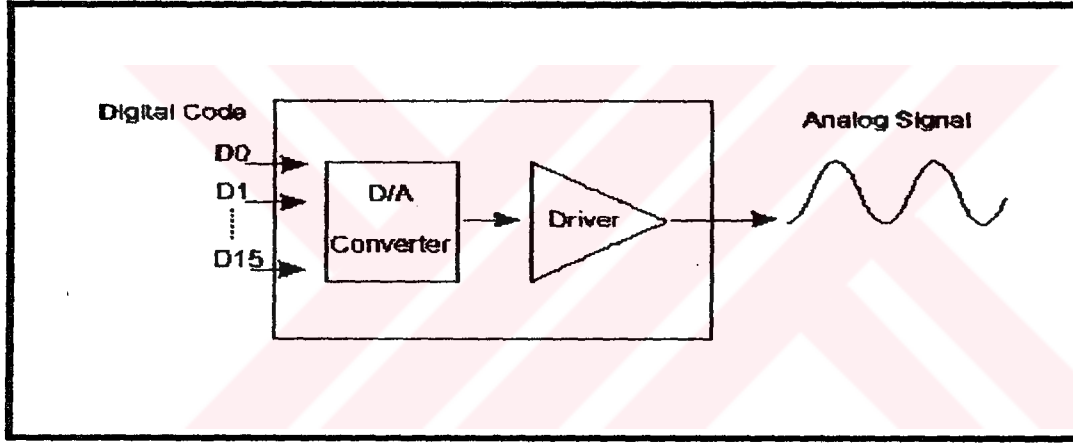


Şekil 2.4. A/D Dönüşüm.

2.4.3.2. Örneksele Çıkışlar (D/A)

Örneksele işaretten sayısal işarete dönüşümün tersi sayısal işareten örneksele işarete dönüşümdür. Bu işlem, sayısal bilgiyi örneksele gerilim veya akıma dönüştürür. D/A aygıtlar, bilgisayarın dışarıdaki olayları kontrol etmesine izin verir.

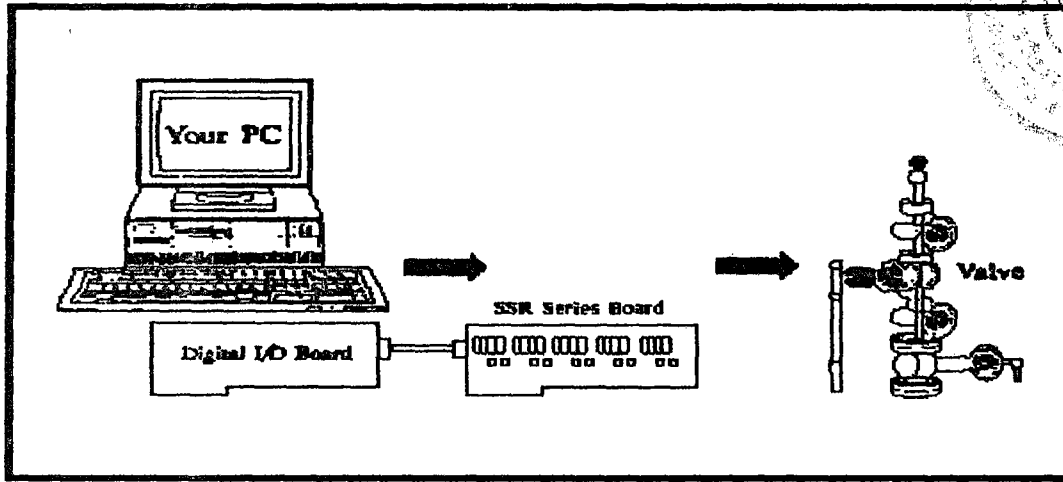
Örneksele çıkış sinyalleri direkt olarak proses ekipmanını kontrol edebilir. Proses, örneksele giriş sinyali formunda geri besleme verebilir. Bu, PID (Proportional Integral Derivative) kontrollü bir kapalı döngü kontrol sistemi olarak tanımlanabilir. Örneksele çıkışlar dalga formu üretmek için de kullanılabilirler. Bu durumda aygıt bir fonksiyon üretici gibi davranır. Şekil 2.5., D/A dönüşümü gösterir [Advantech,1999].



Şekil 2.5. D/A Dönüşüm.

2.4.3.3. Sayısal Girişler ve Çıkışlar

Sayısal giriş/çıkış fonksiyonları, anahtar durum izleme, endüstriyel On/Off kontrol ve sayısal haberleşme gibi uygulamalarda kullanışlıdır. Şekil 2.6., On/Off kontrollü bir vanayı açmak ve kapamak için sayısal giriş çıkışların kullanımını gösterir [Advantech,1999].



Şekil 2.6. Bir Vananın Sayısal I/O ile Kontrolü.

2.4.3.4. Sayaç/Zamanlayıcı

Bir sayaç/zamanlayıcı, olay sayma, akışmetre izleme, frekans sayma, darbe genişliği ölçümü ve zaman periyodu ölçümü gibi işlerde kullanılabilir.

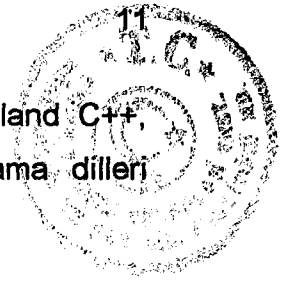
2.4.4. Yazılım

Veri toplama ve kontrol sisterninin arkasındaki güç kontrol yazılımıdır. Veri toplama ve kontrol sisteminin programlanması üç yolla başarılabilir. Bunlar donanım seviyesi programlama, sürücü seviyesi (driver level) programlama ve paket seviyesi programlamadır.

Makina seviyesi programlamada, veri toplama donanımının yazmaçları direkt olarak programlanır. Bunu yapmak için, donanım yazmaçlarına yazılacak olan kontrol kod değerlerini belirlemek gerekir. Bu kodlar, C veya Assembly dili gibi düşük seviye bit kontrol fonksiyonlarını destekleyen bir dilde üretilebilir. Bu kodların yazılması zordur. Bu yüzden bir çok veri toplama donanımı üreticileri müşterilerine sürücüler veya paket programlar sunar.

Sürücü seviyesinde programlamada, kart üreticisinin sağladığı kütüphanelerden yüksek seviyeli fonksiyonlar basitçe çağırılır. Bu fonksiyonlar, A/D dönüşümü gibi kompleks işlemlerin tek bir komutla

tanımlanmasını sağlar. Microsoft Visual C++, Visual Basic, Borland C++, Borland C++Builder ve Delphi gibi yüksek seviyeli programlama dilleri kullanılabilir.

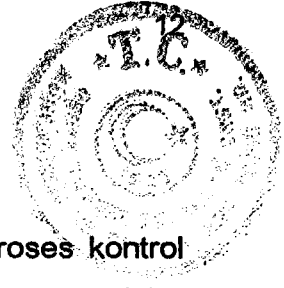


Paket seviyesinde programlamada, uygulama programı işini sizin için yapar. Yazılım paketleri, tek bir arabirimde veri analizini, sunumunu ve aygıt kontrol kapasitelerini birleştirir. Bu programlar, pull down menüler ve ikonlar, veri analizi ve eş-zamanlı grafik ekranı gibi özellikleri sunar [Advantech,1999].

İleride anlatılacak olan Veri Tabanlı Kontrol ve Gözetleme Sistemi (SCADA), her tip programlama seviyesini kapsar.



3. PC TABANLI KONTROL



Kontrol sistemlerinin geliştirilmesinden önce endüstride proses kontrol işinde daha çok insanlar görev alıyordu. Hız kalite ve geniş alan isteyen daha karmaşık prosesler olunca prosesi izleme ve kontrol etmede güvenilirlik ve hassaslık ihtiyacı ortaya çıktı. Günümüzde kullanılan bilgisayar tabanlı proses izleme, daha gelişmiş veri toplama ve kontrol sistemleri isteyen endüstriye bir çözüm sağlar. Bilgisayarlar saniyede binlerce okumayı alabilirler ve testleri uygulayabilirler. Veriyi yönetmelerindeki hassaslık ve güvenilirlik insanlardan çok fazladır.

Bilgisayarlar veri toplama ve kontrol işinde kullanılırken, ya bilgisayara bir veri toplama ve kontrol kartı takılır bunun üzerinden veri haberleşmesi yapılır ya da bilgisayarın kendi giriş/çıkış portları üzerinden veri haberleşmesi yapılır.

Veri toplama ve kontrol kartları, endüstri, araştırma ve proses izleme ve kontrolünün söz konusu olduğu yerlerde, bilgisayar ve kontrol edilecek sistem arasında veri alışverişini sağlarlar. Birçok bilgisayar kullanıcıları, kelime işleme, veri girişi veya bilgisayarla haberleşmenin klavye yoluyla olduğu tüm kullanımlar gibi geleneksel uygulamalara alışkıdır. Bu tip giriş ve çıkış bilgisayarın ön kapısını kullanmak olarak düşünülebilir. Bunlar tipik bilgisayar uygulamalarıdır. Veri toplama ve kontrol kartları yoluyla, teknisyen veya mühendis bilgisayara geleneksel olmayan bir yoldan ulaşabilir. Veri toplama ve kontrol kartı üzerinden ulaşım, bilgisayara arka kapıdan girmek olarak görülebilir. Doğal olarak bilgisayarı programlamak ve bilgisayarın elle kontrolü hala klavye yoluyla yapılır. Arka kapı, bilgisayara veri almak ve mikroişlemcinin işleme ve karar verip yapma gücünü kullanım için başka bir boyut sağlar. Bir bilgisayarın giriş çıkış kapasitelerini kullanmak, teknisyenler ve mühendisler için yeni bir bilgi bütünü yaratır [Rigby and Dalby, 1995].

Veri toplama ve kontrol sistemlerinde paralel haberleşme yerine seri haberleşme tercih edilir. Çünkü seri haberleşme paralel haberleşmeye göre

daha uzun mesafeleri destekler ve daha az kablo ile haberleşme sağlar. Bu yüzden bu bölümde sadece bilgisayarın seri portu incelenmiştir.



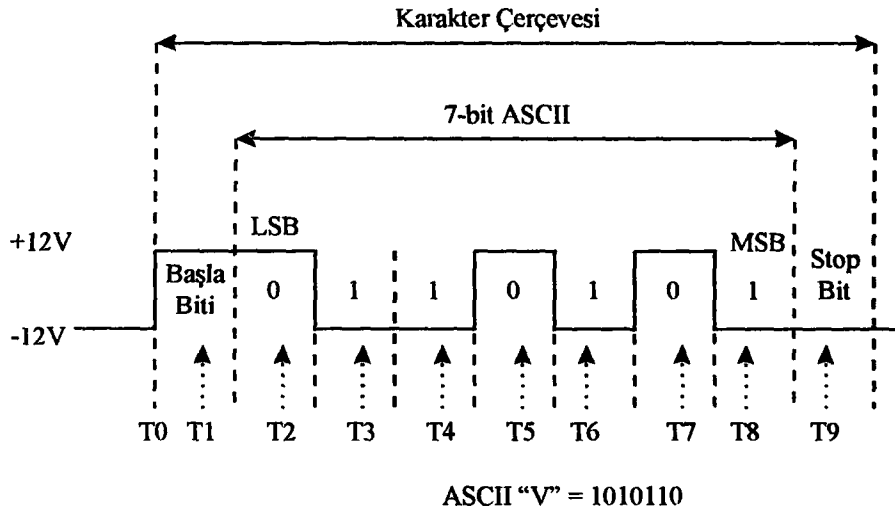
3.1. Seri Port

Seri port üzerinden asenkron haberleşme olur. Eğer veri toplama ve kontrol işinde bilgisayarın seri portu kullanılacaksa ve bunun için program yapılacaksa seri haberleşmenin özelliklerinin bilinmesi gerekir.

Seri portu kontrol etme, uygun iletim hızını, veri formatını ve el sıkışma hatlarının kontrolünü set etmeyi içerir.

Seri iletim hızları, endüstri anlaşmasına bağlı standartlar olarak kurulur. Baudrate'i set etmekle birlikte seri portun ilk ayarlanması veri bitlerinin ve stop bitlerinin sayısının belirtilmesini gerektirir. Şekil 3.1., bir seri karakterin diyagramını gösterir. Şekilde gösterildiği gibi her veri bütünü bir başla işaretiyle başlar. Bu işaret lojik 0'dır ve bir bit periyodu sürer. Başla işaretini takip eden bit, veri bütünü'nün en sağdaki (Least Significant Bit LSB) bitidir. Veri bütünü'nün tüm bitleri gönderildikten sonra bir, veya iki stop biti iletilir. Bu stop bitleri lojik 1'dir. Ek olarak, bir parity bit veri bütünü'nün geçerliliğini göstermek için en soldaki bit (Most Significant Bit MSB) ve stop bitleri arasına konabilir.

Seri iletilen bir bütünü oluşturan veri bitlerinin sayısı genellikle yedi veya sekiz bittir ve stop bitlerinin sayısı da bir, bir buçuk veya ikidir. Eğer ileten aygıt ile alan aygıt arasında iletilen stop biti ve alınan stop biti sayısı konusunda bir anlaşma yoksa iletilen veride sorun olacaktır. Örneğin, ileten aygıttan iki stop biti gönderilirse ve alan aygıt tarafından yalnızca bir stop bit bekleniyorsa bu, asıl veri transfer hızı beklenenden az olacağı şeklinde bir etki oluşturur. Eğer sadece bir stop biti gönderilirse ve alan aygıt tarafından iki stop biti bekleniyorsa bazı veriler kaybolabilir. Bu iki örnek, gönderen aygıt ile alan aygıt arasında seri iletilen veri bütünü'ndeki stop bitlerinin sayısı konusunda anlaşma olmasının önemini gösterir.

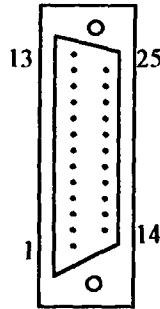


Şekil 3.1. Seri İletimde Bayt'ın Çerçevesi.

Parity, iletilen verinin hata kontrolünü sağlamakta kullanılan bir tekniktir ve veriye bir bit eklemek gerektirir. Parity biti, iletilen veriye ve daha önce belirlenen "tek parity" "çift parity" protokolüne bağlı olarak lojik 1 veya lojik 0 olabilir.

Seri portun ilk ayarlanması, verinin iletilmesinden önce veri bitlerinin sayısı (7 veya 8), stop bitlerinin sayısı (1 veya 2) ve parity seçimi parametrelerini gerektirir.

DB25 veya DB9 konnektörleri seri haberleşme kanalları için bir standart teşkil eder. DB25 konnektör Şekil 3.2.'de gösterildiği gibi 25 pindir. Çizelge 3.1.'de de pinlerin isimleri gösterilmiştir.



Şekil 3.2. DB25 Konnektör.

Çizelge 3.1. DB25 Konnektör Pinlerinin Görevleri.

Pin No	İsim	Kısaltma
1	Çerçeve Topraklama	FG
2	Veri ilet	TxD
3	Veri al	RxD
4	Gönderme isteği	RTS
5	Göndermeye açık	CTS
6	Veri seti hazır	DSR
7	Sinyal Topraklama	SG
8	Veri taşıyıcı bulma	DCD
20	Veri terminali hazır	DTR

3.1.1. Genel Asenkron Alıcı İletici (UART)

UART (Universal Asynchronous Receiver Transmitter) Genel Asenkron Alıcı İletici arabirimi, seri porttan veya seri porta veri transferini sağlar. INS8250 asenkron haberleşme elementi, asenkron haberleşme kartının kalbidir. Yalnız bu aygıt oldukça eskidir ve artık yenisi kullanılmaktadır. 8250 ilk olarak 16540 UART ile değiştirildi. 16540, 8250 ile aynı genel mimariye sahipti fakat daha hızlıydı ve veri transferinde daha yüksek baudrate'leri destekliyordu. Daha sonra bu 16540, karakterler için 16 bit genişlikte alım tamponu olan ve FIFO (First In First Out) tampon olarak inşa edilen bir 16550 UART ile değiştirildi. 16550'nin en yakını ise 32 bit genişliğinde alım tamponunu destekleyen 16560'dır [Wierzbicki,1998].

Modern seri kontrolörler geriye doğru uyumlu olduğu için 8250 I/O arabirimi hakkında öğrenilenler bugünün makinelerine uygulanabilir. Bu işlemci, çeşitli yazmaçlar içerir. Bu yazmaçlar seri portu konfigüre etmek için gerekli bilgiyi tutmakta kullanılırlar. Bunlar ayrıca gönderilen ve alınan binary karakterleri de tutarlar. Yazmaçların seçimi adres ve kontrol bloğu üzerine kurulur. Her yazmaç, seri portun ana adresine ilişkili bir adresle belirtilir. Birçok bilgisayardaki ilk iki seri portun adresleri standartlaştırılmıştır. Bu

adresler seri port 1 için (COM1) 3F8H, seri port 2 için (COM2) 2F8H'dir.

Çizelge 3.2., 8250 yazmaçlarını ve onların adreslerini listeler.

Çizelge 3.2. 8250 Yazmaç Adresleri.

DLAB	A2	A1	A0	Yazmaç	Adres
0	0	0	0	Alma tamponu (Okuma) İleten tutma yazmacı (Yazma)	3F8 (2F8)
0	0	0	1	Kesme imkanı	3F9 (2F9)
X	0	1	0	Kesme belirtici (Sadece okuma)	3FA (2FA)
X	0	1	1	Hat kontrol	3FB (2FB)
X	1	0	0	Modem kontrol	3FC (2FC)
X	1	0	1	Hat durumu	3FD (2FD)
X	1	1	0	Modem kontrol durumu	3FE (2FE)
X	1	1	1	Boş	
1	0	0	0	Bölücü Latch (LSB)	3F8 (2F8)
1	0	0	1	Bölücü Latch (MSB)	3FA (2F8)

3.1.1.1. Tutma Yazmaçları

İsimlerinin de belirttiği gibi bu yazmaçlar alınan fakat henüz okunmamış veya seri porta gönderilen fakat henüz iletilmemiş bir karakteri tutan belirli tamponlardır. INS8250, bir çift tamponlu asenkron haberleşme elementidir. Bu özellik ilk karakter daha tam iletilmeden veya işlemci tarafından okunmadan ikinci karakterin gönderilmesine izin verir.

3.1.1.2. Hat Kontrol Yazmacı

Bu yazmaç, haberleşme değişiminin formatını düzenlemek için programcı tarafından kurulan parametreleri tutar. Adresi 3FBh (2FBh) olan bu yazmacın bit fonksiyonları şöyledir:

Bit 0 ve 1: Her iletilecek karakterin veri bitlerinin sayısını tutar. Bu 5 ila 8 bit arasında olur.



Bit 1	Bit 0	Uzunluk
0	0	5 bit
0	1	6 bit
1	0	7 bit
1	1	8 bit

Bit 2 : Stop bitlerinin sayısını gösterir. Eğer bu bit lojik 0 ise 1 stop biti, lojik 1 ise 2 stop biti üretilir.

Bit 3 : Parity olup olmadığını gösterir. Eğer lojik 1 ise parity var lojik 0 ise yoktur.

Bit 4 : Parity var ise onun tipini gösterir. Eğer lojik 0 ise tek parity, lojik 1 ise çift parity seçili demektir.

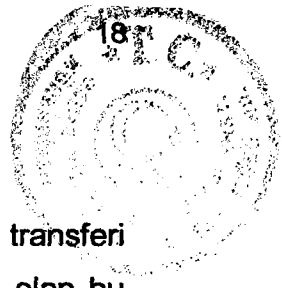
Bit 5 : Stick Parity. Eğer bit 3 ve bit 5 lojik 1 olarak set edilirse, iletici bir karakter ilettiği zaman yerel alıcı lojik 1 algılar.

Bit 6 : Bu bit kırma biti olarak adlandırılır. Bu bit lojik 1 olarak set edildiğinde 6. bit lojik 0 oluncaya kadar SOUT(TxD) nin boş mantıkta kalmasına sebep olur.

Bit 7 : Bölücü latchlerine ulaşmak için bu bit lojik 1 olarak set edilmelidir. Bu latchler seri haberleşmenin baudrate' inin ayarlandığı saat bölücüsünü tutan yazmaçlardır. Baudrate set edilmeden önce bu bit lojik 0 olmalıdır.

Baudrate Üreticini Kontrol

Baudrate'i set etmek için hat kontrol yazmacının 7. biti lojik 1 olarak set edilir. Bu bit set edildiği zaman tutma yazmaçları, yani yazmaç 0 ve 1, baudrate bölücüsünü tutmak için kullanılan bir word uzunluklu bir yazmaç olur. Baudrate bölücüsü, $0x1C200/baudrate$ olarak hesaplanır ve bir word uzunluğunda değişken olarak saklanır. Yazmaç 1 yani 3F8h adresinde bu word'ün LSB kısmı, yazmaç 2 yani 3F9h adresinde de MSB kısmı tutulur.



3.1.1.3. Hat Durumu Yazmacı

Hat durumu yazmacı, mikroişlemciye seri porta doğru veri transferi hakkında bilgi sağlayan 8 bitlik bir yazmaçtır. Adresi 3FDh (2FDh) olan bu yazmacın fonksiyonları şöyledir:

Bit 0 : Veri alındı biti. Veri alındığı zaman lojik 1 olarak set edilir.

Bit 1 : Lojik 1 olduğu zaman bir önceki alınan karakterin kaybolduğu anlamına gelir.

Bit 2 : Lojik 1 olduğunda parity hatası olduğunu gösterir yani alınan karakter yanlış paritydedir.

Bit 3 : Eğer alınan karakter geçerli stop bitine sahip değilse, bu bit lojik 1 olarak set edilir.

Bit 4 : Bu bit ara kesme biti (break interrupt bit) olarak adlandırılır. Alınan veri bütünü, veri bütününün uzunluğu için bir boş kata alındığında bu bit otomatik olarak lojik 1'e set edilir.

Bit 5 : Bu bit, iletim tutma yazmacı boş (transmit holding register empty) biti olarak adlandırılır. Bu bit, seri portun iletilecek yeni bir karakteri kabul için hazır olduğunu gösterir.

Bit 6 : Bu bit sadece okuma olarak konmuştur. Bu bit lojik 1 seviyesinde olduğunda iletili beklememektedir.

Bit 7 : Bu bit kullanılmaz ve daima lojik 0 seviyesindedir.

3.1.1.4. Kesme Yazmaçları

Kesmeler, bilgisayarın normal operasyonunu durdurmak için bir metot sağlarlar ve bilgisayarın yazılım veya donanım kaynağına dikkatini tekrar yönlendirmesine sebep olurlar. Örneğin, bir yazıcı bilgisayara kağıdın bittiğini

anlatmak istiyor, yazıcı bilgisayarla geri haberleşen bir donanım kesmesine sebep olur. Bir donanım kesmesi, donanım aygıtlarından birinin dikkat isteğini bilgisayara bildiren CPU' ya yönelik bir elektrik sinyalidir. Buna karşılık bir yazılım kesmesi, donanım aygıtlarını kontrol eden DOS ve BIOS işlevlerinin çağrılması anlamına gelir.

8250 işlemcisi, geniş kesme kapasitesine sahiptir. Kesme kaynaklarını belirlemede ve kontrol etmede iki yazmaç kullanılır. Bunlardan ilki kesme imkanı yazmacıdır (Interrupt enable register IER) ve adresi 3FGh (2FGh)'dır. Eğer işlemcinin kesme yeteneği aktifleştirilirse ve kesme oluşursa, 8250'den kesme çıkış biti lojik 1 seviyesine gelir. Bu sinyal bilgisayarın donanım kesme yoluna bağlanır. Bu yoldaki lojik 1, işlemciye seri portun ilgiye ihtiyacı olduğunu belirtir.

Bit 0 : Bir karakterin alındığı her seferde bir kesme üretilir. Bu bit, karakter işlemci tarafından okunduktan sonra resetlenir.

Bit 1 : Eğer bu bit lojik 1'e set edilirse bir boş iletim tutma yazmacı bir kesme oluşmasına sebep olacaktır.

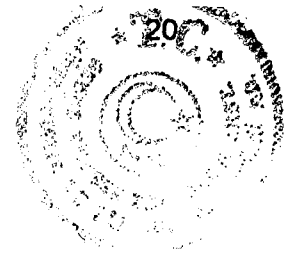
Bit 2 : Bu bit, bir kesme olduğu zaman alıcı hat durumunun değişmesini sağlar.

Bit 3 : Bu bit, işlemciye bir kesme olduğunu belirtmek için modem durumunda değişmeye sebep olur.

Bit 4-7 : Bu bitler sabit olarak lojik 0'a set edilir.

Eğer bir kesme oluşursa, yazılım hangi tip olayın kesmeye sebep olduğunu anlamak için bir yazmaç seçmelidir. Kesme belirtme yazmacı (Interrupt identification register IIR 3FAh), hangi durumun (kesmenin) ilgi istediğini belirten kodu içerir.

Bit 0 : Eğer lojik 0 ise kesme var demektir.



Bit 1: Belirteç biti (ID bit) 0.

Bit 2 : Belirteç biti (ID bit) 1.

Bit 3-7: Bu bitler kullanılmaz ve lojik 0'dır.

Çizelge 3.3., belirteç kodlarını ve hangi kesme durumlarına karşılık geldiklerini gösterir. Kesmeler önceliklendirilmiştir. Önceliklendirilme bazı kesmelerin diğerlerinden daha önemli olduğu anlamına gelir.

Çizelge 3.3. Kesme Belirtme Yazmacı Fonksiyonları.

Kesme ID Yazmacı			Kesme Set ve Reset Fonksiyonları			
Bit2	Bit1	Bit0	Öncelik seviyesi	Kesme Tipi	Kesme Kaynağı	Kesme Reset Kontrolü
0	0	1	-	Yok	Yok	-
1	1	0	En yüksek	Alıcı Hat Durumu	Üzerine hatası Parity hatası Çerçeve hatası Kırma kesmesi	Hat durum yazmacını okuma
1	0	0	İkinci	Alınan veri mevcut	Alıcı veri mevcut	Alıcı tampon yazmacını okuma
0	1	0	Üçüncü	İletici tutma yazmacı boş	İletici tutma yazmacı boş	İR yazmacını okuma. İletici tutma yazmacına yazma
0	0	0	Dördüncü	Modem durumu	Göndermeye açık Veri hazır Zil göstergesi Alındı hattı sinyali	Modem durum yazmacını okuma

3.1.1.5. Modem Yazmaçları

8250 işlemcisinin son iki yazmacı, el sıkışma sinyallerini izleme ve kontrol etmede kullanılır. Modem kontrol yazmacı çıkış el sıkışma sinyallerini kontrol eden 8 bitlik bir yazmaçtır ve adresi 3FCh (2FCh)'dır.

Bit 0 : Veri terminali hazır (Data Terminal Ready DTR).

Bit 1 : Gönderme isteği (Request to Send RTS).

Bit 2 : Çıkış 1.

Bit 3 : Çıkış 2.

Bit 4 : Döngü. Lojik 1 olduğu zaman 8250 işlemcisini geri döngü moduna getirir. Bu mod donanımı test etmeye izin verir.

Bit 5-7 : Kullanılmaz ve lojik 0'dır.

Modem kontrol yazmacının programcının çıkış el sıkışma hatlarını set etmesine izin vermesi gibi modem durum yazmacı (Modem Status Register MSR 3FEh) da programcının giriş el sıkışma hatlarını denetlemesini sağlar.

Bit 0 : Lojik 1 olduğu zaman, bu bit en son okunduğundan beri göndermeye açık girişinde bir değişim olduğunu gösterir.

Bit 1 : Bit 0'da olduğu gibi lojik 1, veri hazır girişi sinyalinde bir değişim olduğunu gösterir.

Bit 2 : Bu bit, zil göstergesinin lojik 1 den lojik 0 durumuna geçtiğini işlemcinin bilmesini sağlar.

Bit 3 : Lojik 1 olduğunda, MSR en son okunduktan sonra alım hat sinyali algılama bitinde bir değişim var demektir.

Bit 4 : Bu bit CTS giriş bitinin zıttıdır.

Bit 5 : Bu bit RTS giriş bitinin zıttıdır.

Bit 6 : Zil göstergesi giriş bitinin tamamlayıcısıdır.

Bit 7 : Bu bit 8250 işlemcisine alım hattı sinyali algılama bitinin zıttıdır.

3.1.2. 8250 I/O Arabiriminin Programlanması

8250, seri port olarak ayarlanmalıdır. Bu işlem iki adımdan oluşur. İlki, programcı bölücü latchlerini set etmelidir ki böylece baudrate kurulabilir. İkincisi, veri bitlerinin ve stop bitlerinin sayısı parity ile beraber seçilmelidir.

Port ayarlandıktan sonra, gönderen aygıt ile alan aygıt arasındaki haberleşmeler başlayabilir. İleticiye bir karakter gönderilmeden önce iletim

tutma yazmacının boş olup olmadığından emin olmak için hat durum yazmacı kontrol edilmelidir. Eğer bir karakter iletim tutma yazmacında (THR) yüklü ise yazmaca gönderilen diğer bir karakter ilkinin üzerine yazılır. Ayrıca hat kontrol yazmacı, okunmayı bekleyen bir verinin olup olmadığını anlamak için de kullanılır.

Buraya kadar port ayarlarının, veri iletiminin ve alımının kontrolünün temelleri anlatılmıştır. El sıkışma sinyallerinin kontrolü, program tasarımının ve yazılımın mantığına bağlıdır.

Bu bölümde anlatılanların uygulaması olan, seriportu ayarlama, seri porta veri yollama ve alma fonksiyonlarını içeren Assembler kodlarını kullanarak yapılmış bir C++Builder programının kodu, Ekler kısmında bulunmaktadır. Bu program dosyası C++Builder'da hazırlanmak istenen herhangi bir projede kullanılarak seri port üzerinden iletişim işlerinin yapılabilmesini sağlar.



4. SCADA SİSTEMİ

4.1. Giriş

Bugünün veri toplama ve kontrol sistem kullanıcıları, gereksinimleri için güçlü, esnek, kolay uygulanabilir ve aynı zamanda pahalı olmayan çözümler aramaktadırlar. Veri Tabanlı Kontrol ve Gözetleme Sistemi (SCADA Supervisory Control And Data Acquisition); endüstriyel bilgisayarlar ve iş istasyonları için tasarlanmış bir gerçek-zamanlı (real-time), çok-görevli (multi-task) yazılım paketi olup, birçok uygulamanın gereksinimlerini karşılamaktadır. Yüksek düzey operatör iş istasyonları veya bilgisayarlarla uyarılacak SCADA paketleri, haberleşme sistemi vasıtasıyla kontrol panolarına yerleştirilmiş kontrol aygıtlarına bağlanmakta ve işletmenin kontrolü için gerekli veri, bilgi ve komut alışverişini yapabilmektedir [Avcı, 1998].

4.2. SCADA Sisteminin Özellikleri

SCADA, süreçler için gözetleyici denetim ve veri toplama işlemlerini yapan sistemler için kullanılmaktadır. SCADA sistemleri, fabrikadaki hammadde, üretim ve mamul takibi gibi süreçlerin denetiminde kullanılan çeşitli araçlarla birlikte fabrikanın üretim kontrolü ve takibine yönelik bir alt yapı oluşturur. Bu alt yapının imkan verdiği ölçüde üretim kaynakları planlaması ve işletme kaynakları planlama sistemleriyle gerekli bağlaşımlar kurularak ideal bir yapıya erişilebilir.

Amaç, minimum maliyette daha kaliteli ve daha çok üretmek için gerekli yapıyı kurmaktır. İşletmedeki tesislerden maksimum verimlilikte yararlanmak, yöneticilerin işletmeye ve üretim bilgilerine tam olarak hakim olmasıyla sağlanabilir.

SCADA yazılım paketleri, endüstriyel tesislerde alt yapı yazılım görevini üstlenmeli ve fabrika içi ile dışındaki ağlara bağlanarak şirketin bütün katmanlarının uyum içinde çalışmasına imkan vermelidir. SCADA, işletme

genelinde herkese her zaman erişebilecekleri gerçek zamanlı ve ayrıntılı bilgiyi sağlamalıdır.



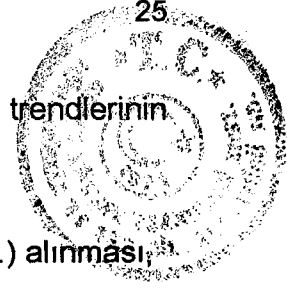
Stok, üretim, dağıtım, satış, pazarlama, servis, bayi ve müşterilerden oluşan üretim ve satış sürecinin en iyi şekilde işleyebilmesi için Internet'in sunduğu olanakların da kullanılması yerinde olacaktır [Yeltekin, 1998].

Yarının SCADA sistemi geniş teşebbüslü entegrasyona dahil edilecektir. Kontrol merkezi ve sadece ona tahsis edilen donanım ve yazılım kaynakları ortadan kalkacaktır. Bu duruma paralel olarak 1970 ve 80'lerde ortaya çıkan, birbirinden bağımsız olan mühendislik, hesaplama, üretim ve kelime işlem sistemleri de ortadan kalkacaktır. Kendini kurtarmayı amaçlayan işletmelerde, otomasyona tahsis edilen kısımlar yerlerini yönetim bilgi sistemine bırakmaktadır.

Buna benzer olarak gelişmiş sistemlerde SCADA, sanal bir sistemi oluşturmak için geniş alan ağı (WAN) üzerinden birbirine bağlanmış iş istasyonu bazlı uygulamalar tarafından gerçekleştirilen bir fonksiyon olacaktır [Pekiner, 1998].

Kuşkusuz, SCADA sistemi operatörler için ileri düzeyde kontrol ve gözetleme özellikleri sağlamalıdır. Genel olarak SCADA sistemi, uygulamada şu imkanları sağlayabilir:

- Kullanıcı tarafından tanımlanmış işletmeye ait mimikler (İşletme Simulasyonu) ve mimik ekranlarda kullanılacak nesneler vasıtası ile işletmenin takibi (seviye, sıcaklık, basınç, sayısal sinyaller, vana ve motor durumları, sistem durumu vb...)
- Reçete ekranları vasıtasıyla reçetelerin girilmesi ve işleyen reçeteler hakkında operatörlerin bilgilendirilmesi,
- Parametre ekranları vasıtasıyla, sistem için gerekli olan parametrelerin girilmesi (Setpoint, alt ve üst alarm değerleri vb.),
- PID parametrelerinin girilebilmesi ve gözetlenmesi,

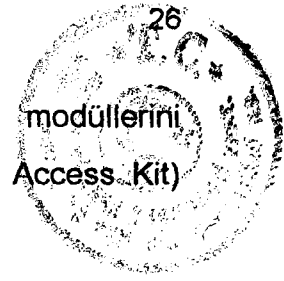


- İşletme değerlerinin tarihsel ve gerçek zamanlı trendlerinin tutulması,
- Anlık ve periyodik raporların (Üretim, Reçete, Stok vb.) alınması,
- Otomatik çalışan sisteme, SCADA ekranlarından manuel müdahale yapılabilmesi,
- Alarm ve durumlarının (event) gösterilmesi ve yazıcıya ve/veya veri tabanına kayıt edilmesi,
- İleri düzeyde kalite kontrol, örneğin İstatistiksel Proses Kontrol (Statistical Process Control SPC) desteği.

Standart SCADA paketleri, kontrol sistemlerine entegre olurken, uygulanabilir kod ve dokümantasyon özelliklerine sahip olmak zorundadır. Bunlar:

- Gerçek zamanlı nüve (Real-time engine),
- Uygulama editörü,
- Dağınık mimarisi,
- Farklı dillerde programlanabilme yeteneği,
- Farklı veri tiplerini destekleme,
- Birçok cihaz ile haberleşebilme,
- Farklı operatör girişi ve şifreleme,
- Dağıtılmış alarm yönetimi,
- Rapor üretimi,
- İnternet/İntranet desteği,

- Kullanıcı haberleşme arabirimlerini ve SCADA modüllerini yazabilme yeteneği (Open System & Programmer Access Kit) [Avcı, 1998].



4.3. SCADA Sistemlerinde Haberleşme

SCADA sisteminin hız performansını etkileyen en önemli kısım haberleşme ağıdır. Kontrolü yapılan sistemlerin çeşitli otomasyon seviyelerinde birbirine bağlanan arabirimler arasındaki veri transferi ve güncelleştirilmesini içeren tüm iletişim, haberleşme ağları üzerinde yapıldığından - kontrol edilen sistemlerin büyüklüğü de göz önüne alındığında - konunun önemi açıkça ortaya çıkmaktadır.

Bu yüzden SCADA tabanlı sistemler geniş alan haberleşme ağı (WAN), yerel alan haberleşme ağı (LAN) vb. sistemleri kullanırlar.

WAN ve LAN, kontrol sisteminde geniş alana yayılmış birden fazla operatör istasyonunun birbirine bağlanması ve işletmeye ait tüm verilerin transferi için kullanılır. Kontrol sisteminin geriye kalan kısımları, iletişim fonksiyonlarından etkin olarak izole edilmiş olmak zorundadır. Bu da haberleşme ağlarının uygulama programına herhangi bir etki yapılmadan gelişmesine veya yeniden yapılanmasına imkan vermektedir.

Hazırlanan ağ programları yardımı ile ağ üzerindeki her bilgisayar ünitesine sistemin bütün imkanları açık hale getirilmektedir. Örneğin sistemde herhangi bir bilgisayara bağlı bir yazıcıdan diğer bilgisayarlar da çıkış alabilmekte ve hatta herhangi bir bilgisayar diğerlerinin ana belleğinde mevcut bir dosyayı "görüp" aktarabilmektedir. Bu da sistemin dosyalama işlemlerini tek bir makineden kolaylıkla yapılabilmesini sağlamaktadır.

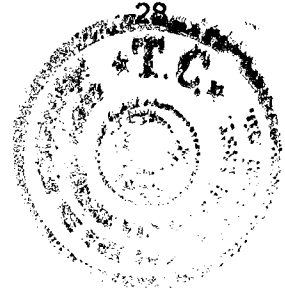
Fieldbus (Saha veri yolu), proses istasyonları ile kontrolörleri birbirine bağlayan çok hızlı bir seri veri haberleşme ağıdır. Durum bilgileri, el kumandaları, arıza ve durum ihbarları, ekran güncelleştirme bilgileri gibi prosese ait tüm veriler bu ağ üzerinden transfer edilmektedir. Fieldbus konusu bu tezin asıl konusu olan Kontrolör Alan Ağının temeli olduğu için ilerdeki bölümde ayrıntılı olarak anlatılmaktadır.

Üretim hattında kullanılan çeşitli kontrol sistemleri ve cihazları ile haberleşmek için önceki bölümlerde anlatıldığı gibi sürücü programlara ihtiyaç vardır. Eğer böyle bir program yok ise cihazın haberleşme protokolü elde edilerek veya çözülerek yeni bir programın yazılması gerekir. SCADA programlarının haberleşme sürücüsü portföylerinin geniş olması bu bakımdan önemlidir.

SCADA programları aynı anda birden fazla kontrol cihazına bağlanabilmekte ve birden fazla protokol aracılığı ile kontrol cihazlarıyla haberleşebilmektedir. Ancak sahadaki ekipmanların SCADA programlarına bağlaşımında fazla sayıda sürücü programının kullanılması hız ve güvenilirlik açısından uygun değildir. Bu bağlaşımları alt düzeyde gerçekleştirdikten sonra SCADA yazılımına aktarmak daha yerinde olacaktır [Gören, 1999].



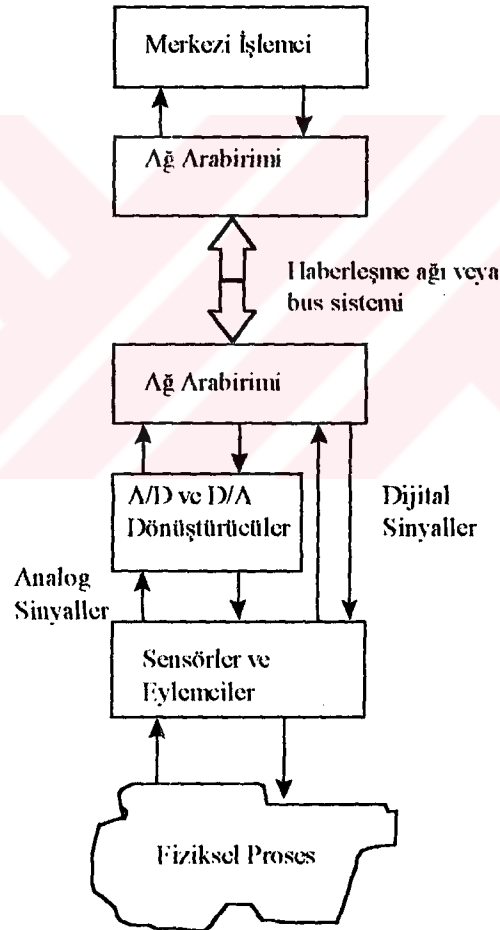
5. KONTROLÖR ALAN AĞI (CAN)



5.1. Giriş

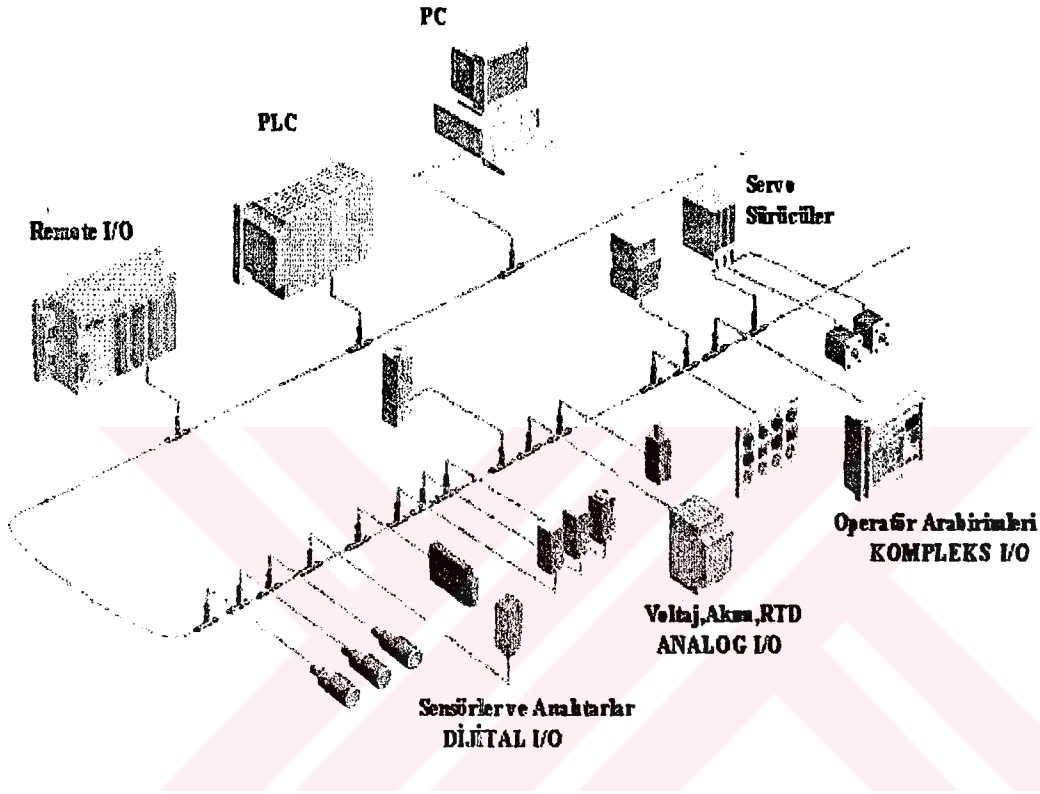
PC deki slot ve adresleme kısıtları ve uzak mesafelerde veri toplama/kumanda hatları sorun çıkartmaktadır. İçerisinde PC, PLC vb. birçok değişik türden veri toplama adası bulunan geniş alanlı uygulamalar için ağ teknolojilerinin kullanılması uygun olmaktadır.

Şekil 5.1., dağıtık veri toplama ve kontrol sistemlerinin genel yapısını göstermektedir.



Şekil 5.1. Dağıtık Veri Toplama ve Kontrol Sisteminin Genel Yapısı.

Otomasyon sahasında görülen, farklı prosesleri izlemek için kullanılan sensör, eylemci, iletilci, sürücü, PLC gibi ekipmanların sayılarının artması, bu sayısal saha cihazları ile daha yüksek seviyeli otomasyon komponentleri arasında iletişimin "Bit Serial Fieldbuses" kısa adı ile "FieldBus" (saha veri yolları) üzerinden yapılmasını gündeme getirmiştir. Şekil 5.2. bir fieldbus sistemini göstermektedir.



Şekil 5.2. Bir Filedbus Sistemi.

Günümüzde son kullanıcı bir markaya bağlı kalmaktansa farklı ürün grupları için farklı marka ürünleri birlikte kullanabilmeyi tercih etmektedir. PLC, Display giriş ve çıkış modülleri, sürücüler, valf adacıkları gibi çok çeşitli saha cihazlarını farklı firmalardan seçebilmekte, böylece endüstriyel ortamlardaki prosesleri, saha veri yollarına bağlanan çeşitli proses kontrol elemanları ve bu saha veri yolları ile direkt haberleşebilen bir önceki bölümde de bahsi geçen izleme ve kontrol yazılımları (SCADA) kullanarak kontrol edebilmektedir. Saha seviyesinde yaşanan kablolama karmaşasını da yine popüler saha veri yollarına direkt bağlanabilen dağıtılmış giriş-çıkış modülleri çözmektedir [Sarıbay,1998].

Fieldbus, yeni bir sayısal iletişim ağını tanımlayan bir terimdir. Bu yeni ağ, endüstride şu an kullanılmakta olan ve 4-20 mA analog sinyalleşme üzerine kurulu olan analog ağın alternatifi olarak geliştirilmiştir. Fieldbus ağının temel özellikleri sayısal, çift yönlü (full duplex), multidrop ve seri veri yolu olmasıdır. Birbirinden bağımsız olan aygıtlar (kontrolör, algılayıcı, eylemci, sensör) bu ağ sayesinde birbirine bağlanırlar. Her bir alan aygıtı kendi içerisinde bazı kolay fonksiyonları yerine getirebilecek şekilde dizayn edilmiştir. Bu fonksiyonlar, diagnostik ve kontrol gibi aygıtta has işlemler olabildiği gibi, diğer aygıtlarla karşılıklı iletişim işlemlerini gerçekleştirebilecek fonksiyonlar da olabilir. Bütün bu özellikleriyle birlikte bu cihazların düşük güç tüketen bir yapıya da sahip olması onları birer zeki aygıt şekline dönüştürür. Fieldbus, hem operatörlerin bu aygıtlara erişebilmelerini hem de aygıtların birbirleriyle haberleşebilmelerini temin eder. Aslında fieldbus ile şu an kullanılmakta olan analog sistemin en önemli farkı merkezi alan ağ yapısının dağıtık bir yapıya dönüştürülmüş olmasıdır. Dolayısıyla fieldbus'ın sadece analogtan sayısalge geçiş şeklinde algılanması onu gerçek değerinden uzak bir şekilde tanımlamak olur.

Fieldbus teknolojisi iletişimde kalitenin ve verimliliğin artırılmasıyla birlikte maliyetin de düşürülmesini sağlar. Bunların gerçekleşmesi büyük ölçüde analog olarak yapılan sinyalleşmenin sayısal hale dönüştürülmesiyle meydana gelir. Sayısal iletişim sayesinde her bir saha aygıtı meydana gelebilecek arızaları veya yapılması gereken ayar değişikliklerini rapor edebilecek duruma gelir. Bu sayede sistemin genel verimliliği artarken, sistemin devamlılığının sağlanması için yapılması gereken işlemler de azaltılmış ve kolaylaştırılmış olur.

Bir fieldbus aygıtı analog sistemdeki bir kaç aygıtın yerine kullanılabilir. Bu şekilde maliyetlerde büyük bir azalma olur. Bunun yanında, fieldbus aygıtlarının kurulması ve kablolama işlemleri de eskisine göre hem kolay, hem de ucuzdur. Analog sistemdeki her bir aygıtın kendine özgü kabloları ve bağlantı noktaları varken, tüm fieldbus aygıtları sadece bir bükümlü kablo ile bağlamak mümkündür [Coutinho et al, 1995].

1996'daki verilere göre, Avrupa'da proses kontrol ve endüstriyel uygulamalarında her iki projeden biri popüler saha veri yolları (CANbus, Profibus, Interbus, AS-I, LONTalk) kullanılarak yapılmaktadır ve bu rakam 98 sonlarında her üç projeden ikisi olarak değişmiştir. Saha veri yollarının en önemli özelliği; Fieldbus sektörünün 10 yıllık geçmişi sonunda 30'a yakın farklı protokolden birbirlerine göre çeşitli üstünlükleri olan 5-6 protokolün (Avrupa fieldbus pazarının %95'ine sahip) Avrupa normlarına girmesi ve konusunda söz sahibi komponent üreticilerinin tümü tarafından desteklenmesidir. Bu, kullanıcıya Proses Otomasyonu gibi ayrı otomasyon kesimlerinde farklı uygulamalar için doğru çözümler üretebilme garantisi getirmektedir.

Günümüzde CANbus, ProfiBus, InterBus, ArcNet, AS-I, IEC/ISA, Seriplex, WorldFIP, LonWorks gibi bir çok saha veri iletişim çeşidi bulunmaktadır [Sarıbay, 1998]. Bu tezde gün geçtikçe daha çok tercih edilen CANbus incelenmiştir.

5.2. Kontrolör Alan Ağı (CAN)

Kontrolör Alan Ağı (CAN), otomotiv endüstrisi için düzenlenmiş bir seri ağıdır. Fakat zamanla diğer endüstriyel otomasyon uygulamalarında da popüler hale gelmiştir. CAN veri yolu, öncelikle gömülü sistemlerde kullanılır ve isminin ifade ettiği gibi kontrolörler arasında kurulan ağıdır. Çift tel, yarı duplex, yüksek hızlı ağ sistemidir ve kısa mesajlar kullanan yüksek hızlı uygulamalara iyi uyumluluk sağlar. Güvenilirliği ve yarı iletken endüstrisi tarafından geniş çaplı desteklenmesi de CAN' in yararlarındandır.

CAN, teorik olarak bir tek ağda 2032 aygıtı bağlayabilir (Bir nokta bir belirteç varsayarak). Bununla birlikte donanımın (tranceivers) pratik limitine bağlı olarak bir ağda sadece 110 nokta bağlanabilir. Bu 1Mbit/s'e kadar yüksek hızlı haberleşme sunar böylece gerçek zamanlı kontrole izin verir. Ek olarak, hata sınırlama ve hata tespiti özelliği, onu gürültülü çevrelerde daha güvenilir yapmaktadır.

CAN ilk olarak 1986'da Almanya'da Mercedes'in taşıtlardaki uç elektronik kontrol ünitesi arasında bir haberleşme sistemi geliştirilmesi isteği üzerine Robert Bosch GmbH tarafından geliştirildi. Onlar UART'ın bu durum için uygun olmadığını çünkü onun Point to Point haberleşme kullandığını buldular. Çok sahipli haberleşme sistemi ihtiyacı zorunlu hale geldi. İlk CAN silikonu 1987 de Intel tarafından üretildi [Nilsson,1997].

5.3. CAN Standartları

Orijinal özellik Bosch tarafından sağlanmıştır. Bu özelliğin 2.0 versiyonu iki parçaya bölünmüştür:

- Standart CAN (Versiyon 2.0A). 11 bit belirteçler kullanır.
- Gelişmiş CAN (Versiyon 2.0B). 29 bit belirteçler kullanır.

İki parça, ana fark belirteç uzunluğu olarak mesaj çerçevesinin farklı formatlarını tanımlar. CAN için iki ISO standardı vardır. Fark fiziksel katmandadır, ISO 11898, 1Mbit/s kadar yüksek hızlı uygulamaları kaldırırken ISO 11519, 125 kbit/s üst limite sahiptir.

5.3.1. A Bölümü ve B Bölümü Uyumluluğu

CAN kontrolörlerinin üç tipi vardır. A bölümü, pasif B bölümü ve B bölümü. Onlar standardın farklı bölümlerini kaldırır.

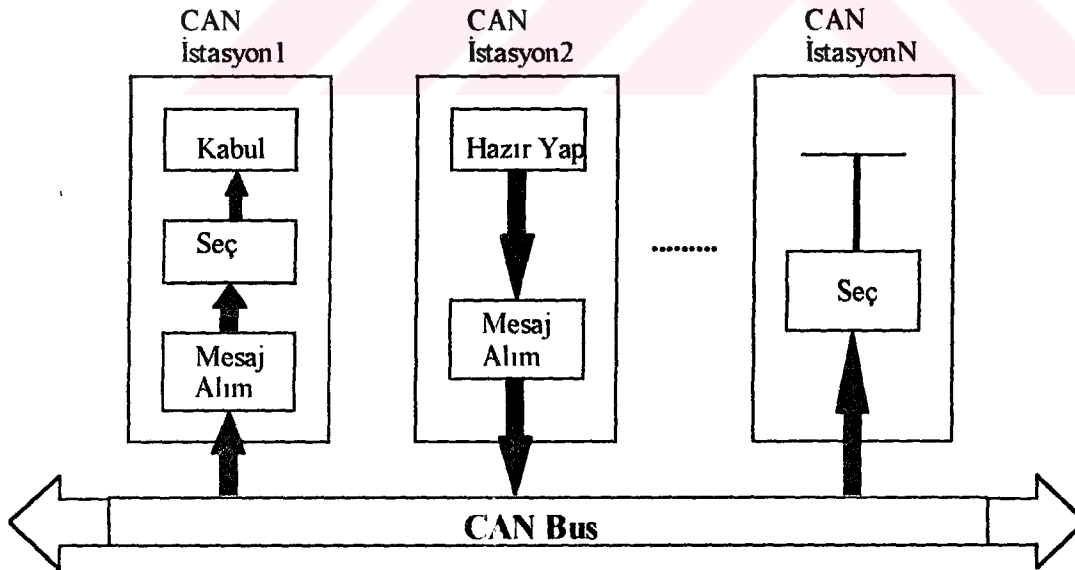
Çizelge 5.1. CAN Kontrolör Tipleri Uyumluluğu

Mesaj formatı\Can çip tipi	A bölümü	Pasif B bölümü	B bölümü
11 bit ID	Tamam	Tamam	Tamam
29 bit ID	HATA!!	Veri yolunda katlanılır, fakat önem verilmez.	Tamam

Çoğunluk 2.0A kontrolörleri sadece standart formatta mesajları iletir ve alır. Bazısı 2.0B pasif olarak bilinenler gelişmiş formatta mesajları alacaktır ama sonra onları önemsemez. 2.0B kontrolörleri her iki formatta mesajları gönderebilir ve alabilir. Şunu unutmamak gerekir eğer 29 bit belirteçler A bölümü kontrolörleri içeren bir veri yolunda kullanılırsa, veri yolu çalışmayacaktır [Bosch, 1991].

5.4. CAN'ın Çalışması

Söylendiği gibi CAN bir çok sahipli ağıdır. Veri yoluna ulaşmada CSMA/CD+AMP (Carrier Sense Multiple Access/Collision Detection with Arbitration on Message Priority) tekniğini kullanır. Bir mesaj yollamadan önce CAN noktası, veri yolunun meşgul olup olmadığını kontrol eder. Ayrıca çarpışma farketme (Collision Detection) kullanır. Bu yönleriyle Ethernet'e benzer. Bununla birlikte Ethernet ağ, çarpışma bulduğu zaman her iki gönderen noktalar iletimi durdurur. Tekrar göndermeden önce rasgele bir zaman beklerler. Bu, Ethernet ağı yüksek veri yolu yüklerine karşı çok hassas hale getirir. CAN, bu problemi hakemliğin çok akıllı bir prensibi ile çözer. Şekil 5.3., CAN genel yapısını gösterir.



Şekil 5.3. CAN Genel Yapısı.



5.4.1. Prensip

CAN veri yolu üzerindeki herhangi bir noktadan iletilen veri mesajları, ne ileten noktanın ne de amaçlanan alıcı noktanın adreslerini içermez. Bunun yerine mesaj içeriği ağ boyunca tek olan bir belirteç ile etiketlenmiştir. Ağdaki tüm diğer noktalar mesajı alır ve her biri, mesajın özel nokta ile alakalı olup olmadığını belirlemek için belirteç üzerinde bir kabul testi uygular. Eğer mesaj alakalı ise işlenir, değilse önemsenmez.

5.4.2. Belirteçler ve Hakemlik

Tek belirteç ayrıca mesajın önceliğini de gösterir. Belirtecın sayısal değerinin düşüklüğü önceliğin yüksekliğini gösterir. Bu, eğer iki veya daha fazla nokta aynı anda veri yoluna girmek için yarışırsa hakemliğe izin verir. Yüksek öncelikli mesaj eğer iletilecek tek mesajsa, veri yolu girişini kazanmayı garantiler. Düşük öncelikli mesajlar otomatik olarak bir sonraki veri yolu turuna iletilir.

Her CAN mesajı, 11 bit (Can özellik A bölümü) ya da 29 bit (B bölümü) belirtece sahiptir. Her CAN mesajının başına yerleştirilen bu belirteç, CAN hakemlik alanının prensip parçasıdır. Belirteç mesajın tipini ve ayrıca önceliğini belirtir.

Bir CAN mesajında düşük bitler daima egemendir. Bu şu anlama gelir, eğer bir nokta bir düşük göndermeye çalışır ve diğer nokta bir yüksek göndermeye çalışırsa sonuç düşük olacaktır. Bir ileten nokta iletirken daima veri yolunda dinler. Hakemlik alanında bir yüksek gönderen ve bir düşüğü bulan bir nokta hakemliği kaybettiğini bilir. İletimi durdurur daha yüksek öncelikli diğer noktaya izin verir.

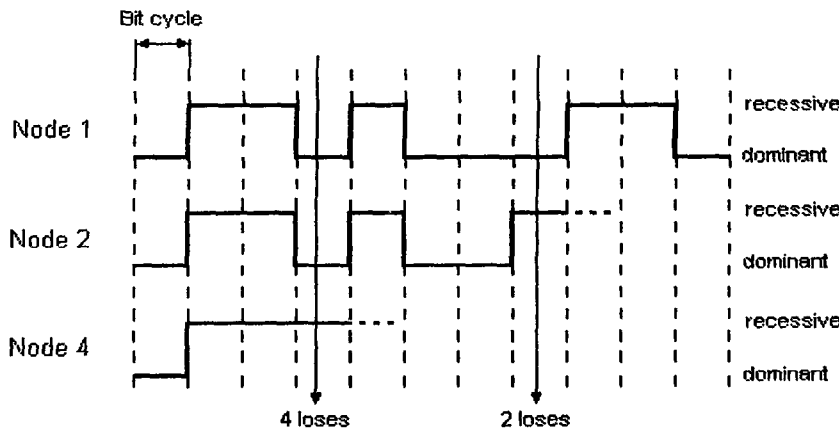
Aşağıdaki durumu örnek olarak düşünelim:

- Nokta 1, 01101000110 (838) belirteci ile veri iletmek ister.

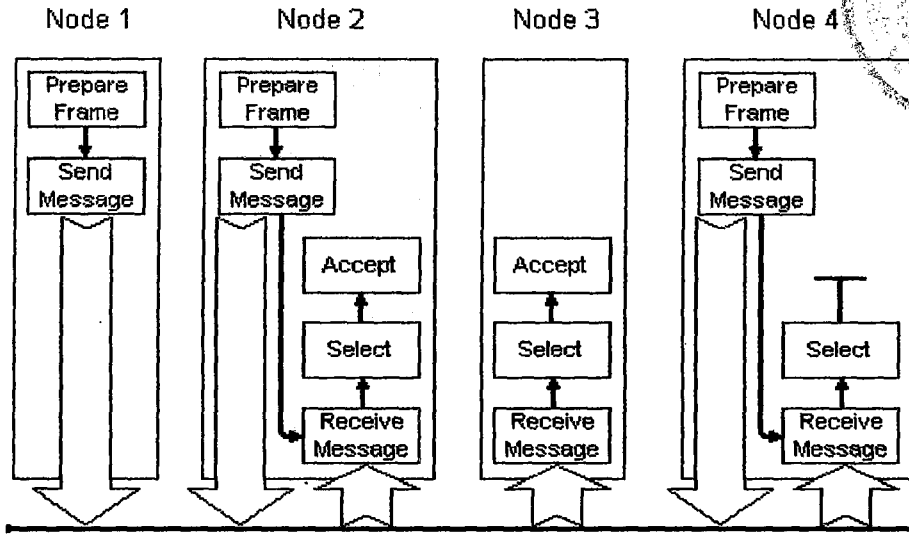
- Nokta 2, 01101001000 (840) belirteci ile veri iletmek ister ve 01101000110 belirteçli veriyi alabilir.
- Nokta 3, sadece 0110100xxxx (832-847 aralığı) ile başlayan belirteçteki verileri alır.
- Nokta 4 de 01110010010 (914) ile belirtilen veriyi iletmek ister ve sadece 0111001xxx1 (913-927 arası tüm tek sayılar) belirteçleri ile belirtilen veriyi alır.

Nokta 1, Nokta 2 ve Nokta 4 iletme aynı anda başlarlar. 4. bite kadar ileticilerden hiçbiri hatta başkasının olduğunu anlamaz. 4. bit olduğunda, Nokta 1 ve Nokta 2 sıfır iletir, Nokta 4 ise 1 iletir. 0 egemen olduğundan hatta 0 gider ve Nokta 4, bir iletmediği halde sıfır izler. Bu durumda hatta daha öncelikli bir verinin iletildiğini anlar, iletimi durdurur ve alıcı durumuna geçer. Nokta 1 ve 2, hatta 0 gördüklerinden iletim ve hakemlik devam eder. Farklı olan 8. bite gelince Nokta 1, 0 iletir fakat Nokta 2, 1 iletir. Nokta 2 hakemliği kaybettiğini anlar ve alıcı konumuna geçer. Nokta 1 iletimine devam eder ve tüm verisini iletir [Aladjem, 1997].

Şekil 5.4., Nokta1, Nokta 2 ve Nokta 4 arasındaki bit hakemliğini göstermektedir. Şekil 5.5., ise iletim ve alımda noktaların durumunu gösterir [Aladjem, 1997].



Şekil 5.4. Nokta 1, 2 ve 4 Arasındaki Bit Hakemliği.



Şekil 5.5. İletim ve Alımda Noktaların Durumları.

Ağdaki iki noktaya, aynı belirteç ile mesaj yollama izni verilmez. Eğer iki nokta bir mesajı aynı belirteçle aynı anda göndermeyi denerse hakemlik çalışmayacaktır. Bunun yerine, ileten noktalardan biri, mesajının hakemlik alanının dışında bozulduğunu bulacaktır. Bundan sonra noktalar CAN' in hata idaresini kullanacaklardır.

5.4.3. Uzak Çerçeveler

CAN' de iki tür çerçeve vardır, uzak çerçeveler ve veri çerçeveleri. Veri çerçeveleri ağda bir nokta veri iletmek istediği zaman kullanılır ve normal çerçeve tipidir.

Uzak çerçeveler bilgi için istek olarak tanımlanabilir. RTR(Uzak iletim isteği) bit setli bir çerçeve, iletim noktasının belirteçle verilen tip bilgisini sorduğu anlamına gelir. Bilginin mevcut olduğu nokta, ağa bilgiyi göndererek yanıt vermelidir. CAN kontrolörünün aletlerine bağlı olarak cevap otomatik olarak gönderilebilir. Daha basit CAN kontrolörleri (Basit CAN) otomatik olarak yanıtlayamazlar. Bu durumda baş mikrokontrolöre uzak isteğe karşı haber verilir ve baş mikrokontrolör veriyi göndermek zorundadır.



5.5. CAN Mesaj Formatları

Bir CAN sisteminde veri, mesaj çerçeveleri kullanılarak iletilir ve alınır. Mesaj çerçeveleri veriyi bir iletim noktasından bir yada daha çok alan noktaya gönderir.

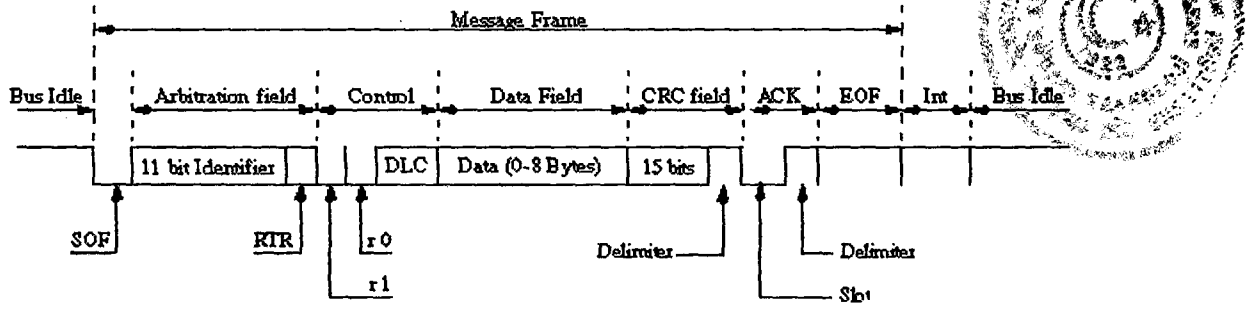
CAN protokolü iki mesaj çerçevesi formatını destekler. Bu iki format :

- Standart CAN (Versiyon 2.0A)
- Gelişmiş CAN (Versiyon 2.0B)

5.5.1. 2.0A Format

Şekil 5.6. da gösterilen bir standart CAN mesaj çerçevesi formatı yedi farklı bir alanı içerir [Nilsson,1997]:

- Bir standart çerçeve alanı (SOF). Bu mesaj çerçevesinin başlangıcını gösteren bir egemen (Lojik 0) bittir.
- 11 bit mesaj belirteci ve uzak iletim isteği (RTR Remote Transmission Request) bitini içeren bir hakemlik alanı. Bir egemen (Lojik 0), RTR biti mesajın veri çerçevesi olduğunu gösterir. Bir lojik 1 değeri mesajın bir uzak iletim isteği olduğunu gösterir. Bir uzak çerçeve, bir noktanın veri yolundaki diğer bazı noktalardan veri için isteğidir. Uzak çerçeveler veri alanı içermezler. Veri uzunluk kodu istenilen mesaj çerçevesindeki verinin baytlarının sayısını belirtir.



Şekil 5.6. CAN 2.0A Mesaj Çerçevesi

- Bir kontrol alanı 6 bit içerir:
- * gelecek uygulamalar için rezerve edilmiş iki egemen bit (r0 ve r1) ve
- * bir 4 bit veri uzunluk kodu (DLC). DLC veri alanındaki baytların sayısını gösterir.
- Veri alanı 0 dan 8'e bayt içerir.
- CRC alanı 15 bit sürekli redundancy kontrol kodu ve bir pasif delimiter bit içerir.
- Kabul alanı iki bit içerir. İlki pasif bit olarak iletilen Slot bit, ikincisi ise pasif delimiter bit.
- Çerçeve sonu alanı 7 pasif (Lojik 1) bit içerir.

Bir çerçevenin sonunu, 3 pasif bit içeren INTermission alanı izler. 3 bit INTermission periyodundan sonra veri yolunun boş olacağı kabul edilir. Veri yolunun dinleme zamanı sıfır içeren keyfi herhangi bir uzunluk olabilir.

5.5.2. 2.0B Formatı

CAN 2.0B formatı, 11 bit belirteç içeren 2.0A'nın tersine 29 bit belirteç sağlar. Versiyon 2.0B, USA'de otomotiv uygulamalarında kullanılan diğer seri

haberleşme protokolleriyle uyumluluğu sağlaması için geliştirilmiştir. 2.0A formatıyla uyumluluğu hala sağlamasına ek olarak versiyon 2.0B'deki mesaj çerçevesi gelişmiş bir formata sahiptir.

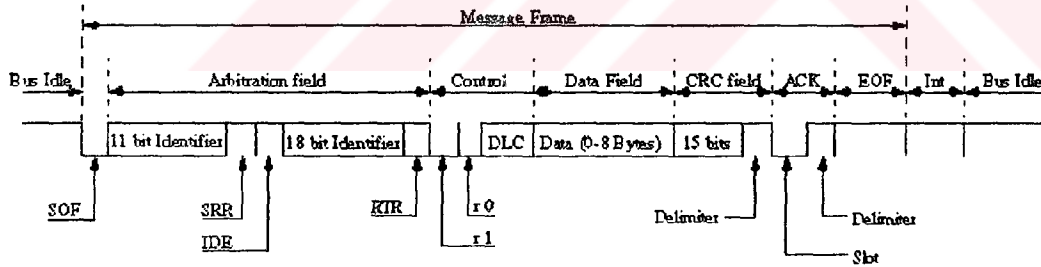
Farklar şunlardır:

- Versiyon 2.0B de hakem alanı iki belirteç bit alanı içerir. İlki (Ana Belirteç), versiyon 2.0A ile uyumlu 11 bir uzunluktadır. İkinci alan (Belirteç Ek), 18 bit uzunluktadır ve toplam olarak 29 bit uzunluk olur.

- İki mesaj formatı arasındaki fark bir Belirteç Ek (IDE) biti kullanılarak yapılır.

- Bir vekil uzak istek biti (SRR) Hakem alanındadır. Standart veri çerçevesi ve gelişmiş veri çerçevesi arasında hakemlik gerektiği zamanlarda SRR biti daima bir pasif bit olarak iletilir. Eğer her iki mesaj da aynı 11 bit belirtece sahipse standart veri çerçevesi daima önceliklidir.

Şekil 5.7. de gösterilen 2.0B mesaj çerçevesindeki diğer tüm alanlar Standart formattaki ile aynıdır [Nilsson,1997].

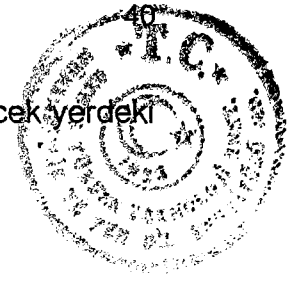


Şekil 5.7. CAN 2.0B Mesaj Çerçevesi.

5.6. Hata Bulma ve Faul Sınırlama

CAN standardında tanımlanan hata bulma, sinyalleme ve faul sınırlama CAN veri yolunu çok güvenilir yapmaktadır. Kontrolörlerdeki hata bulmayı hata sinyalleme ile beraber inşa etmek bilginin doğru ve sürekli olduğunu

kesinleştirir. Faullü noktalar, veri yolundaki trafiği rahatsız etmeyecek yerdeki modlara gideceklerdir.



5.6.1. CAN'de Hata İşlemi

1. CAN kontrolörü tarafından hata bulunur.
2. Bir hata çerçevesi iletilir.
3. Mesaj tüm noktalarda iptal edilir.
4. CAN kontrolörlerinin durumu yenilenir.
5. Mesaj tekrar iletilir. Eğer bir çok kontrolör mesaj gönderecekse, normal hakemlik kullanılır.

5.6.2. Hata Bulma

Hata bulma CAN kontrolörü tarafından otomatik olarak yapılır. Bulunan hatalar:

Bit Hataları :

1. Bit tıkama hatası : Bir ileten nokta ardışık 5 düşük bitin arkasına bir yüksek bit (ya da ardışık 5 yüksek bitin arkasına bir düşük bit) sokarsa buna bit tıkama denir. Alan nokta, ihlali bulur.
2. Bit hatası : Bir ileten nokta daima mesaj gönderilirken onu geri okur. Eğer farklı bir bit değeri bulursa gönderildikten sonra, ve bir hakemlik alanının yada kabul alanının bir parçası değilse hata bulunur.

Mesaj hataları :

1. Toplam denetleme hatası : Her alan nokta CAN mesajlarını toplam denetleme hataları için kontrol eder.



2. Çerçeve hatası : Her CAN mesaj çerçevesi içinde, belli noktalara iletilmesi gereken önceden tanımlı bit değerleri vardır. Eğer alıcı bu pozisyonlarda bir eksik bit bulursa bir Form hatası (Format hatası) işaretlenecektir.
3. Kabul hatası : Eğer bir iletilen mesajın kabul almadığını anlarsa bir kabul hatası işaretlenir.

5.6.3. CAN kontrolörü Hata Modları

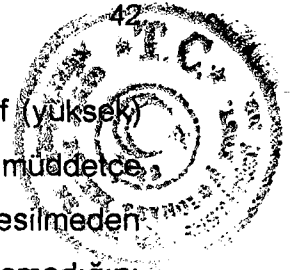
Bir Can kontrolör üç durumda olabilir :

1. Hata aktif : bir kontrolör için normal işleme modudur. Mesajlar alınabilir ve iletebilir. Bir hata bulunduğunda bir aktif hata bayrağı gönderilir.
2. Hata pasif : Mesajları iletirken veya alırken kontrolörün bazı problemleri varsa bu mod girilir. Mesajlar iletebilir ya da alınabilir. Alırken bir hata bulunduğunda bir pasif hata bayrağı gönderilir.
3. Veri yolu kapalı : Mesajları iletirken kontrolörün seri problemleri varsa girilir. Can kontrolör baş mikro kontrolör veya işlemci tarafından resetlenene kadar hiçbir mesaj iletilemez yada alınamaz.

5.6.4. Hata Sinyalleme

Bir nokta tarafından hata bulunduğu zaman, o veri yoluna bir hata bayrağı gönderir. Bu, diğer noktaların mesajı almasını önler.

Aktif hata bayrağı altı düşük bitten oluşur ve eğer hata çerçevesini ileten nokta aktif hata durumundaysa kullanılır. Düşük egemen olduğu için diğer tüm noktalar bit tıkama ihlalini bulacaktır ve kendi hata bayraklarını yollayacaktır. Bundan sonra iletim yapmak isteyen noktalar gönderime başlayacaktır. Mesajı en yüksek önceliğe sahip nokta hakemliği kazanacaktır ve mesajı gönderecektir.



Eğer kontrolör hata pasif moddaysa, hata çerçevesi altı pasif (yüksek) bittten oluşacaktır. Hata bayrağı sadece pasif bitlerden oluştuğu müddetçe, veri yolu etkilenmez. Eğer bir nokta hata bulmazsa mesaj kesilmeden gönderilecektir. Bu, alımda problemlı noktanın veri yolunu tıkayamadığını gösterir.

Tüm bu ileri hata tutmalar, CAN kontrolör tarafından baş mikrokontrolörün hiçbir şey yapmasına gerek kalmadan otomatik olarak yapılır. Bu, CAN' in büyük avantajlarından biridir.

5.7. Bit Zamanlama

CAN, uzun veri yollarındaki gecikmelerle mücadele ve veri yolu üzerindeki noktalar için zaman sayaçlarının kristal frekanslarındaki farklılıklarla mücadele için gelişmiş özelliklere sahiptir.

Bit zamanlama seçimi; bit hızı, örnek nokta ve yeniden zamandaşılama gerektiği zaman çok önemlidir.

Her bit, zamandaşılama dilimi, üreme dilimi, evre dilimi I ve II olmak üzere dört dilime bölünür. Her dilim bir veya daha çok zaman parçasından oluşur. Bir zaman parçası, CAN kontrolör saati tarafından türetilen sabit bir süreye sahiptir.

Zamandaşılama dilimi veri yolu üzerindeki çeşitli noktaları zamandaşılama için kullanılır. Veri yolu üzerinde bir bit gönderildiği zaman, en önemli kenarın bu dilimde olması beklenir. Bu dilim daima sabit bir zaman parçası uzunluğundadır.

Üreme dilimine veri yolu hatlarındaki bekleminin telafi edilmesi için ihtiyaç duyulur. Dilim büyüklüğü 1 ve 8 zaman parçası arasında programlanabilir.

Evre dilimi I ve evre dilimi II dilimleri, resenkronizasyonu uzatma veya kısaltma için kullanılabilir.

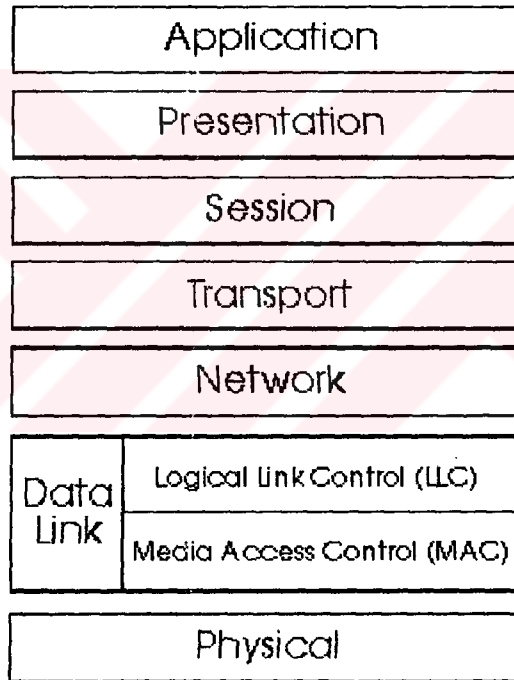


5.8. OSI ve CAN Modellerinin Karşılaştırılması

5.8.1. OSI Modeli

ISO7498, Açık Sistemler Bağlantısı (OSI Open System Interconnection) modeli olarak bilinen bir haberleşme standardı tanımlar. Bu model herhangi bir ağdaki bilgisayarlar arasındaki haberleşmenin nasıl oluşacağını açıklar ve genel bir "açık" ağ haberleşme standardı olarak benimsenmiştir.

OSI Model, bir protokol yığınının yedi bağımsız katman tanımlar. Standartla tam uyum, her katmanın diğerlerinden çok iyi tanımlanmış arabirimlerle ayrılmasını gerektirir. Şekil 5.8. de bu katmanlar gösterilir [Schofield, 1998].



Şekil 5.8. OSI Katmanları.

Her katmanın fonksiyonlarının tam açıklaması ISO7498 standardında vardır. Özetle:

- Uygulama katmanı (katman 7) en yüksek katmandır ve kullanıcıya daha aşağıdaki katmanlara arabirimi sağlar.

- Sunum katmanı (katman 6) farklı noktalar arasındaki haberleşmeye izin vermek için veri formatlama ve kod değiştirme yapar. (Veri formatlarının ve gösterimlerinin tek bir ağda olduğu CAN ile normal olarak bağlantılı değildir.)

- Oturum katmanı (5) haberleşme oturumlarındaki verinin koordinasyonunu yönetir.

- Taşıma katmanı (4) mesajların sıralanmasını kontrol eder.

- Ağ katmanı (3) adresleri set eder ve mesaj paketlerini dağıtır.

- Veri bağlantı katmanı (2) mesajlarla bit ve bayt seviyesinde uğraşır ve nokta ile ağ arasındaki veri transferini sağlar.

- Fiziksel katman (1) veri yolunun donanım elemanlarına karşılık gelir. Ağa doğru ve ağdan bit akıntılarının geçtiği katmandır.

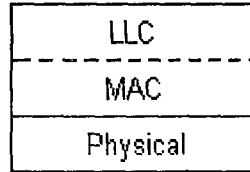
OSI modelinin nasıl işlediğine örnek olarak iki noktadan oluşan basit bir CAN sistemini düşünelim. Bir noktada bir sensör, ölçüm yapar ve bir borudan bir sıvının akış hızını ölçer; diğeri bir vanayı kontrol eden bir eylemci ve akışı ayarlar. Sensördeki sinyal kullanıcı uygulama yazılımı (uygulama katmanı seviyesinde) tarafından sürekli izlenir. Eğer akış hızı, uygulama yazılımında belirlenmiş bir eşikle belirlenen miktardan daha fazla artarsa, uygulama yazılımı belirli bir miktar vananın kapatılması için mesaj iletime başlar ("Vana X'i, Y miktar kapat"). Basit terimlerle sensördeki sinyal OSI katmanlarda aşağı doğru iner, gerekli parametreler bir CAN mesaj çerçevesinde vana X için bir belirteç ve Y'ye eşit bir veri değeriyle belirtilir. Mesaj çerçevesi kodlanır ve veri yolu üzerinde elektriksel sinyaller serisi şeklinde iletilir. Alıcı noktada elektriksel sinyaller çözülür ve vanayı kapama işlemini yapmak için alıcının uygulama katmanına gider.



5.8.2. CAN ve OSI Modeli

CAN Standartı (ISO11898) bir CAN ağ için yalnızca fiziksel ve veri bağlantı katmanlarını belirtir. Şekil 5.9., CAN'ın kullandığı katmanları göstermektedir [Aladjem, 1997].

CAN Katmanları



Şekil 5.9. CAN Katmanları.

- Veri bağlantı katmanı mesaj formatlarını anlayan ve tanıyan tek katmandır. Fiziksel katmana gönderilecek mesajları oluşturur ve fiziksel katmandan gelen mesajları çözümler.

CAN kontrolörlerinde veri bağlantı katmanı genellikle donanımda bulunur. Karmaşıklığından dolayı ve birçok genel ağların genelinde veri bağlantı katmanı kendi içinde ikiye ayrılır.

- Mantıksal Bağlantı Kontrolü Katmanı (Logical Link Control LLC) modeldeki daha yüksek katmanlardan veri mesajlarının iletim ve alımına bakar.

- İletim Ortamı Erişim Katmanı (Media Access Control MAC) katmanı iletim için mesajları kodlar ve sıralar, alınan mesajları çözer. MAC ayrıca mesaj önceliğine, hata bulmaya ve fiziksel katmana ulaşım ile ilgilidir.

- Fiziksel katman, veri yolunun ve iletilecek mesajlar için mesaj karakterlerini elektrik sinyallerine, alınan mesajlar için de elektrik sinyallerini karakterlere dönüştüren donanımın fiziksel ve elektriksel karakteristiklerini

belirtir. Diğer katmanlar donanıma veya yazılıma yerleştirilebildiği halde fiziksel katman daima gerçek donanımdır.



5.8.3. CAN Veri Yolu Fiziksel Katman

Fiziksel katman Bosch CAN standardının bir parçası değildir. Bununla birlikte, ISO standartlarında iletili karakteristikleri belirtilir. CAN, sinyalleri CAN yüksek ve CAN alçak olmak üzere iki tel içeren CAN veri yolu üzerinde iletir. Bu iki tel farklı mod'da çalışır. Bu, onların çevrilmiş voltajlar taşıyor olmasıdır. Voltaj seviyeleri, fiziksel katmanın diğer karakteristiklerinde olduğu gibi hangi standardın kullanıldığına bağlıdır.

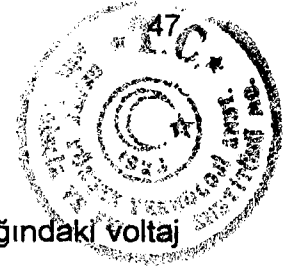
5.8.3.1. ISO 11898

ISO 11898 (CAN yüksek hız) standardını izleyen bir CAN ağ için voltaj seviyeleri aşağıdaki tabloda gösterilmiştir.

Çizelge 5. 2. ISO 11898 Özellikleri.

Sinyal		Pasif durum			Egemen durum		Birim
	min	Nominal	max	min	nominal	max	
CAN-yüksek	2.0	2.5	3.0	2.75	3.5	4.5	Volt
CAN-alçak	2.0	2.5	3.0	0.5	1.5	2.25	Volt

Pasif durum için her iki teldeki nominal voltaj aynıdır. Bu, noktalardan sonlandırma yükü dirençlerine doğru gücü düşürür. Bu dirençler 120 ohm'dur ve her iki telin sonuna yerleştirilmiştir.



5.8.3.2. ISO 11519

ISO 11519 (CAN düşük hız) standardını izleyen bir CAN ağındaki voltaj değerleri aşağıdaki çizelgede gösterilmiştir.

Çizelge 5.3. ISO 11519 Özellikleri.

Sinyal		Pasif durum			Egemen durum		Birim
	Min	nominal	max	min	nominal	max	
CAN-yüksek	1.6	1.75	1.9	3.85	4.0	5.0	Volt
CAN-alçak	3.1	3.25	3.4	0	1.0	1.15	Volt

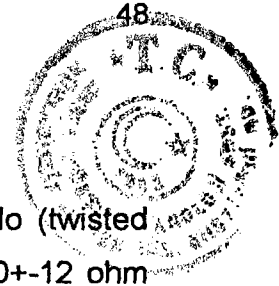
ISO 11519 terminasyon dirençleri gerektirmez. Onlar gerekli değildir çünkü limitli bit hızları (maksimum 125kB/s) veri yolunu yansımalarla karşı duyarsız yapar. CAN veri yolu üzerindeki voltaj seviyesi, veri yolu dinlemede olduğu zaman pasiftir.

5.8.3.3. Veri yolu Uzunlukları

Bir CAN ağındaki maksimum veri yolu uzunluğu kullanılan bit hızına bağlıdır. Aşağıdaki çizelgede farklı veri yolu uzunlukları ve karşıladığı maksimum bit hızları vardır.

Çizelge 5.4. Veri Yolu Uzunlukları.

Veri yolu uzunluğu (metre)	Maksimum bit hızı(bit/s)
40	1 Mbit/s
100	500 kbit/s
200	250 kbit/s
500	125 kbit/s
1 km	10 kbit/s



5.8.3.4. İletim ortamı

İletim ortamı olarak korumalı yada korumasız bükümlü kablo (twisted pair) kullanılır. ISO 11898 standardına göre, kablonun direnci 120 ± 12 ohm olacaktır. İş, tek teller için düzenlenmiş standart olan SAE J2411 e uygun olarak çalışmaktadır.

5.9. CAN Uygulama Katmanları

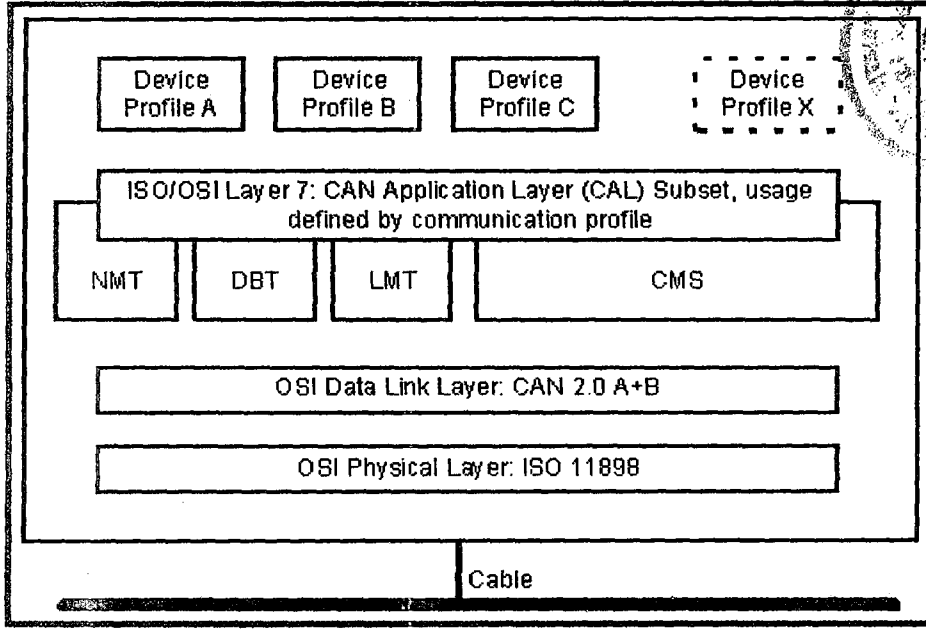
CAN' in bir çok uygulamaları, veri bağlantı katmanı ile belirtilen temel fonksiyonların ötesinde fakat uygulama katmanında işlenebilecek servisler gerektirir. Örneğin, 8 bitten daha uzun veri birimlerinin iletimi ve alımı. Bu ihtiyacı karşılamak için bir çok organizasyon uygulama katmanları geliştirdi. CiA- CAN in Automation- tarafından desteklenen 4 tane yüksek katman protokolü vardır: CANOpen, DeviceNet, CAN Kingdom ve Smart Distributed System (SDS) [CiA,1999].

5.9.1. CANOpen

CANOpen, CAN Uygulama Katmanı CAL'ın bir uygulamasıdır ve CiA DS-301'de CANOpen Haberleşme Profili tarafından tanımlanır. İlk olarak EEC ESPRIT III projesi olan ASPIC'den geliştirilmiştir. CANOpen çoğunlukla dağıtık I/O modüllerinin ve diğer aygıtların CAN tabanlı bir fieldbus ağı üzerinden iletişim kurdukları endüstriyel otomasyon uygulamalarında kullanılır. Avrupa pazarında söz sahibidir.

5.9.1.1. ISO/OSI katman modeli ve CANOpen

OSI referans modelini referans alan CANOpen mimarisi Şekil 5.10. da gösterilmiştir [NURAG,1997].



Şekil 5.10. CANOpen Mimarisi.

CANOpen, sistem entegrasyonuna ve aygıt standardizasyonuna yardımcı olan ve aygıt profilleri denen bir kavram kullanır. Bu profillerde öngörülen tanımlamaları ve ipuçlarını kullanarak, iki farklı üretici firma standartlaşmış aygıtlar üretebilirler. Bu yaklaşımın en büyük avantajı sistem kurucularını belirli bir üreticiye bağımlı kalmaktan kurtarmasıdır. Aygıt profilleri, farklı üreticilerden gelen aygıtların ağ üzerinde birbirleriyle iletişim kurabilmeleri için her bir aygıtta özel yazılım kullanma zorunluluğunu ortadan kaldırır. CANOpen aygıt profillerinde temel ağ operasyonlarının garanti edilmesi için aygıtların sahip olması gereken ana özelliklerin tanımlarıyla birlikte, isteğe bağlı veya üretici firmaya has fonksiyonelliklerin de tanımlanması mümkün hale getirilmiştir.

5.9.1.2. İletişim Modeli

CANOpen protokolü CAN veri yolu üzerindeki mesajlaşmayı yönetmek için birkaç metot tanımlar. Bu metotlar iki ana grup altında toplanır.

- 1- Zamandaş veri transferi.
- 2- Zamandaş olmayan veri transferi veya olay telegramları.

Zamandaş haberleşmede veri toplama ve gerçekleştirme işlemi tanımlanmış zamandaşlama nesnelerinin koordinasyonunda yapılır. Bu sistemde tüm ağ belirli bir zaman diliminde aktif veya pasif olan aygıtların düzenli bir şekilde haberleşmesi ile yönetilir. Zamandaş olmayan veri iletiminde ise bir aygıt herhangi bir durum değişikliğinde, kendisine ayrılan zamanı beklemeden yayın yapabilir. Mesajlaşmanın hangi tipte olacağı dinamik olarak kontrol edilebilir.

CANOpen protokolünde tanımlanmış 4 tip mesaj tipi (iletişim nesnesi) mevcuttur.

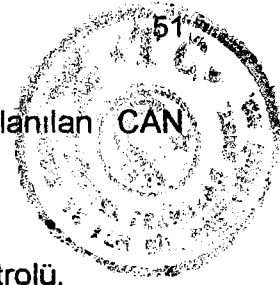
- Ağ yönetim mesajları LMT (Katman Y.), NMT (Ağ Y.), DBT (Tanımlayıcı dağıtımı).
- Servis veri mesajları.
- Proses veri mesajları.
- Özel mesajlar (Zamandaşlama telegramı, acil durum mesajları).

5.9.1.3. Veri Transfer Mekanizmaları

Servis veri mesajları çoğunlukla aygıtlar arasındaki düşük öncelikli, ve yinede miktardaki veri iletiminde kullanılır. Bu mesajlar asenkron olarak iletilirler. Normalde aygıtların konfigürasyonu için kullanılırlar. CANOpen protokolü cihazların en azından servis veri mesajlarını desteklemesi gerektiğini belirtir.

5.9.1.4. Ağ Yönetimi

Ağ yönetimi, bir CANOpen ağlarındaki cihazların durumlarını kontrol etmek için kullanılır. CANOpen ağ yönetimi hizmeti şu fonksiyonları içermektedir:



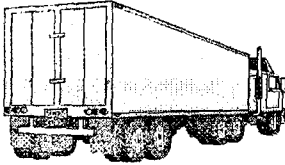
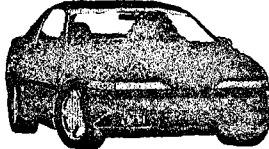
- SDO/PDO iletişimi için ve hata hizmetleri için kullanılan CAN tanımlayıcılarının dinamik ve statik olarak dağıtımı.
- Her cihazın işlem durumunun ve iletişim modlarının kontrolü.
- Cihazların fonksiyonel olup olmadıklarını tetkik etmek için ağ periyodik olarak sorgulama.

5.9.1.5. Veri Takas Mekanizmaları

CANOpen protokolünde iki veri transfer mekanizması mevcuttur. Servis veri nesneleri (SDO Service Data Object), düşük önceliğe sahip büyük miktardaki verilerin iletişimde kullanılır ve genelde ağ üzerindeki cihazların konfigürasyonlarının yapılması için gerekli verileri taşırlar. Bu moddaki veri transferi 8 baytdan fazla olabilir.

İkinci tip veri transferinde ise proses veri nesneleri (PDO Process Data Object) kullanılır. PDO metodu yüksek hızlı ve yüksek öncelikli verilerin iletişimde kullanılır. Veriler 8 bayt veya daha azdır. Çizelge 5.5. CANOpen içindeki veri transfer mekanizmalarının karşılaştırılmasını gösterir [NURAG,1997].

Çizelge 5.5. Veri Transfer Mekanizmalarının Karşılaştırılması.

Servis Veri Nesnesi (SDO) Haberleşmesi	Proses Veri Nesnesi (PDO) Haberleşmesi
• Düşük öncelikli yüksek hacimli veri transferleri. • 16 bit indeks/8bit sub-indeks nesne adresleme yapısı kullanımı. • Çoklu CAN telegramları kullanarak 8 baytdan büyük veri transferleri. 	• Yüksek öncelikli düşük hacimli veri transferleri. • Veri alanının bağımsız baytlarına bölünmüş nesneler. • Tek bir CAN telegramı kullanarak 8 bayt'a kadar veri transferi. 



5.9.2. DeviceNet™

DeviceNet, düşük maliyetli, basit ve açık bir endüstriyel ağ sistemidir. İlk olarak Rockwell/Allen-Bradley tarafından geliştirilmiştir, Amerika ve Uzakdoğu pazarlarında söz sahibidir. Dünya çapında yönetimini CiA ile birlikte "Open DeviceNet Vendor Association – ODVA" yapmaktadır.

DeviceNet, CAN altyapısını kullandığı için OSI 7-katman modelini destekler. DeviceNet, 8 baytlık bir fieldbus sistemidir. Daha çok basit I/O kontrolörlerin (sensörler, anahtarlar, barkod tarayıcılar, AC/DC sürücüler) kullanıldığı orta ölçekli endüstri uygulamalarında tercih edilir.

DeviceNet, nesne tabanlı (Object Oriented Technique OOT) bir protokoldür. Bu özelliği sayesinde genişlemeye uygun bir yapı kazanmıştır. Her üretici ODVA'nın tanımlamalarına uymak koşuluyla, kendine özgü DeviceNet cihazları üretebilir.

Temelde 4 ana DeviceNet nesnesi vardır:

- Kimlik nesnesi: Cihazın kimliğiyle ilgili genel özellikleri içerir.
- Bağlantı nesnesi: Bu nesne bir cihazda bulunan I/O portalarının sayısını tutar. Ayrıca ağ bağlantılarını oluşturur.
- DeviceNet nesnesi: Port özelliklerinin (cihaz adresi, veri transfer hızı vb.) konfigürasyonu için kullanılır.
- Mesaj yönlendirici: Bu nesne cihazların işleyişiyle alakalı değildir. Görevi alınan mesajların işlenen hedefe yönlendirilmesini sağlamaktır.

DeviceNet sisteminde, diğer CAN cihazlarının aksine düşük MAC adresine sahip olan cihazın ağa ulaşım hakkını elde etmesi diye bir şey söz konusu değildir. Hak öncelikle mesajın hangi mesaj grubuna bağlı olduğuna göre değişir.

5.9.2.1. Üretici/Tüketici Modeli

TCP/IP gibi geleneksel seri haberleşme protokolünde haberleşme mekanizması daima gönderen/gönderilen yer şeklindedir. Bu nedenle, bir nokta diğeriyle haberleşmek istediğinde, adres alanına gönderilen yerin adresini koyacaktır. Bu, sadece gönderilen yerin mesajı alacağından emin olmaktır. Bu yöntem hedef adresin ihtiyaç duyacağı kesme sayısını azalttığı için mükemmeldir. Ayrıca özel amaçlar içindir. Bununla birlikte gerçek zamanlı kontrol olduğunda gereksiz zaman kaybı olur. Böyle bir durumda çoklu hedef adreslerinin aynı anda yenilenmesi gerekir.

Üretici/Tüketici modelinde ise, mesaj iletmek isteyen nokta "hangi noktaya mesajın gönderildiğini" belirtmez, onun yerine "mesajın hangi noktadan gönderildiğini" belirtir. Önceden konfigüre edilmiş alıcılar, mesaj kendileri için yararlı veya anlamlı ise onu kabul ederler.

5.9.2.2. Mesaj Grupları

DeviceNet'de CAN belirteçleri 4 gruba ayrılır.

Mesaj Grubu	CAN ID Aralığı	Açıklaması
Grup1	0x000 – 0x3FF	Zaman Kritik I/O mesajı. Yüksek öncelikli.
Grup2	0x400 – 0x5FF	Önceden tanımlanmış Master/Slave bağlantı için Harici ve I/O mesajları.
Grup3	0x600 – 0x7C0	Zaman kritik olmayan mesajlar. İzleme ve diagnostik için kullanılır.
Grup4	0x7C0– 0x7CF	Çevrim dışı haberleşme için kullanılır.

DeviceNet'de harici ve dahili (I/O) mesajlar olmak üzere iki mesaj tipi vardır.

Nesne tabanlı tekniğin yararı ile DeviceNet, cihazların ağda herhangi bir sorun olmadan çalışmasını garanti eder. Böylece DeviceNet, endüstriyel fieldbusların gelecekteki eğilimini gösterir [Tang,1999].



5.9.3. SDS™ (Smart Distributed System)

Honeywell'in Micro Switch bölümü tarafından geliştirilen SDS zeki sensör ve eylemciler için kullanılan ileri bir veri yolu sistemidir. CAN tabanlı bu sistemin temel özellikleri şunlardır:

- Hem kontrol hem de aygıt seviyelerinde açık bir sistemdir.
- Kurulum işlemi çok kolaydır.
- Makina uygulamalarını iyileştirir.
- Giriş ve çıkış değerlerini güçlendirir.
- Endüstri için bütünleşik bir sistem sağlar.

SDS ağında 4 telli tek bir kablo üzerinden maksimum 64 farklı cihazı (126 değişik adres) bağlamak mümkündür. Bu zeki sensör ve eylemci cihazlar sadece on/off durumlarını içermekle kalmaz aynı zamanda SDS'in sağladığı cihaz seviyesinde fonksiyonları (sistem ve cihaz diagnostikleri) kullanırlar [Honeywell, 1999].

SDS'in kullanıldığı tipik uygulama alanları paketleme/yiyecek işleme sistemleri, otomobil endüstrisi, materyal işleme, taşıyıcı sistemleri ve otomatik stok geri alım sistemleridir.

5.9.4. CAN Kingdom™

CAN Kingdom isimli diğer bir CiA onaylı uygulama katmanı, Kvaser AB isimli bir İsveç firması tarafından geliştirilmiştir.

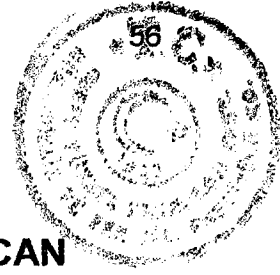
CAN Kingdom asıl olarak makina kontrolü için tasarlanmıştır. Örneğin, endüstri robotları, dokuma makinaları, hidrolik makinalar,vb.. Bazı ortak meziyetlerini sayacak olursak benzer sistemlerin, eş zamanda yüksek performans, yüksek güvenlik talepleri ve yüksek sistem performansdır. CAN



Kingdom, US D&D CDA- Askeri hedefler için ortak sayısal mimarı – 101A (Common Digital Architecture - 101A) standartlarının köşe taşlarındandır. Uygulama alanı, küçük lastik bot ve arabalardan supersonik güdümlü mermilere, Phantom uçaklara ve destroyerlere kadar geniş çaplıdır.

CAN Kingdom, 80'li yılların başından beri gelişmiş dağıtık gömülü kontrol sistemlerindeki deneylere dayanır. Mühendislerin sistem seviyelerinde ve de mümkün olan modül seviyelerinde dahi maksimum performansta çalışması için tasarlanmıştır. Aynı zamanda, sistem seviyesini modül seviyesinden ayırmak için de tasarlanmıştır. Bunun ana nedeni, sistem tasarımı daha ilk devrelerindeyken modüllerin farklı takımlar tarafından zaten paralel olarak geliştiriliyor olmasını sağlamak ve böylece geliştirme zamanını azaltmaktır. Bu da CAN Kingdom'la bir projede geç de olsa büyük kararlar alabilme imkanının olduğunu gösterir. Bu aynı zamanda şu sonuçları da doğurur: CAN sistemi kolay dokümente edilir ve de kolay geliştirilebilir. Sistemin ve modüller tabakalarının ayrılması, kritik zamanlar ve de güvenli olarak kritik sistemlerde tasarımda tam kontrol için kullanışlıdır. Sistemi dizayn eden için, CAN Kingdom açık bir protokoldür ama son kullanıcı sadece sistem dizaynında verilen limitler içinde sistemi geliştirebilir.

CiA tarafından desteklenen diğer yüksek katman protokollerinden CAN Kingdom'ı ayıran en büyük özellik onun makina kontrolü için dizayn edilmiş olmasıdır. Diğerleri daha çok üretim otomasyonuna odaklanmışlardır. CAN Kingdom diğerlerinin yanında, belirteçlerin dinamik olarak değiştirilmesini, global saati, EAN/UPC'nin modül tanımlamalarını, standart ve gelişmiş belirteçleri ve donanımın sınırladığı sistemdeki modül sayılarını destekler. DeviceNet ve SDS modülleri kolaylıkla CAN Kingdom'a entegre edilebilir, ama sistem performansında kayıplar olur [Kvaser AB, 1999].



5.10. CAN Uygulamaları

5.10.1. Farklı uygulamalar - Temel CAN ve Tam CAN

CAN kontrolörlerinin nasıl uygulanacağı yada kendilerinin baş mikro kontrolörü ile nasıl haberleşeceği için herhangi bir standart yoktur. Bugün CAN kontrolörleri için iki ana uygulama stratejisi vardır. Bunlara Temel CAN ve Tam CAN denir.

Bu stratejiler arasındaki ana fark, alakalı mesajların nasıl filtreleneceği yani hangi mesajın alakalı hangisinin alakasız olduğunun kararlaştırılmasıdır. Ayrıca uzak çerçevelerin nasıl cevap verdiği ve mesajların nasıl saklandığı da aradaki farklardandır. Farklar baş mikro kontrolöre ne kadar yükün konduğunu etkileyecektir.

5.10.2. Temel CAN

Temel CAN genellikle daha ucuz standalone CAN kontrolörlerinde veya entegre CAN kontrolörü ile birlikte daha küçük mikro kontrolörlerde kullanılır.

Bir Temel CAN kontrolörü normalde iki alım tamponu ve bir iletim tamponuna sahiptir. Alım tamponları FIFO mimarisine göre ayarlanmıştır ve bir mesaj mikrokontrolör bir tampondan bilgiyi okurken diğer tampona ulaşabilir.

Eğer bir mesaj ulaştığında her iki alım tamponları doluysa, en eski mesajlar tutulur. Bu şu anlama gelir; eğer baş mikrokontrolör mesajı yeterli kadar hızlı okuyamazsa daha yeni mesajlar kaybolabilir. Bir mesaj, iletim tamponuna yazılarak gönderilebilir.

İlgili mesajlar mesaj belirtecinde işleyen iki yazmaç kullanılarak filtrelenir. Belirteçteki her bit filtreye karşı denetlenir. Eğer mesaj filtreye uyuşursa, alım tamponlarından birine depolanır.

Filtre belirtecinin her biri 1, 0 ya da önemsiz şeklinde set edilir. Genellikle filtre belirtecinin 11 bitinin 8'inde işler (standart CAN). Bu şu anlama gelir belirtecinin düşük 3 biti daima önemsizdir.



Temel CAN kullanıldığında, belirteçlerin seçilmesi önemlidir, böylece filtrenin çerçevesi mümkün olduğu kadar küçük tutulabilir. Filtreden geçmesine izin verilen tüm mesajlar, mikro kontrolör tarafından okunmalı ve denetlenmelidir. En son filtreleme yazılım tarafından yapılır.

Bir Temel CAN kontrolörü uzak çerçevelere otomatik cevap verme desteğine sahip değildir, uygulama onları tutmak zorunda kalacaktır. Bu, mikro kontrolör veya işlemci üzerine ekstra yük koyacaktır fakat dönen değerlerin yenilenmesinden emin olunacaktır. Çizelge 5.6.'da Temel CAN özellikleri listelenmiştir.

Çizelge 5.6. Temel CAN Özellikleri.

İletim	Uygulama, ID, RTR, veri uzunluğunu kapsayan tüm Tx yazmacını doldurur.
Alım	Her CAN mesajı alınabilir. FIFO mimarisinde iki alım tamponu. Global mesaj filtreleme Sadece ilgili mesajların geçmesine izin verilir, son filtreleme uygulama tarafından yapılmalıdır.
Uzak çerçeve tutma	Uzak çerçevelere uygulama tarafından cevap verilir.
Aşırı çalışma felsefesi	En eski mesajı saklama (yenileri kaybolacak)

5.10.3. Tam CAN

Tam CAN, daha pahalı, yüksek performanslı CAN kontrolörleri ve mikrokontrolörlerinde kullanılır. Tam CAN kontrolörü, posta kutusu denen bir tampon kümesine sahiptir. Başlangıçta her posta kutusu bir belirteçle gösterilir ve iletim ve alım olarak set edilir.

CAN kontrolörü bir mesaj aldığı anda, aynı belirteçli bir alım posta kutusu var mı diye posta kutularını denetler. Böyle bir posta kutusu bulunursa mesaj ona yüklenir ve baş kontrolör haberdar edilir. Diğer halde mesaj atılır.



Bir mesaj iletildiği zaman mesaj uzunluğu ve veri, doğru belirteçle iletim kutusuna yazılır.

Bir uzak mesaj ulaştırıldığında kontrolör iletim posta kutularına karşı uzak belirteci denetler. Bir uygunluk bulunduğunda, kontrolör otomatik olarak belirteçle ve posta kutusunda tutulan veriyle bir mesaj yollar. Bu şu anlama gelir; mikrokontrolör daha az yük tutar ve yazılım uzak mesajları tutmak zorunda değildir. Bununla birlikte, posta kutusu uzun zamanda yenilenmezse, ağa yollanan bilgi eski olacaktır. Bu, yazılımı yazarken göz önünde tutulmalıdır.

Bir Tam CAN kontrolörü ile birlikte ilgili olan sadece kesin mesajları filtrelemek mümkündür. Bu tip kontrolör baş mikro kontrolör üzerinde daha az yük verecektir. Bununla birlikte posta kutuları sınırlıdır. Bugünün bir CAN kontrolöründe olan en yüksek posta kutusu sayısı bilinmiyor.

Bazı kontrolörlerle posta kutularını dinamik olarak tekrar şekillendirmek mümkündür. Yine de bu problemi tamamen çözmez. Bu yüzden bazı kontrolörler karışık CAN kontrolörleridir. Bunlar posta kutularına sahiptir fakat ayrıca Temel CAN tamponları da vardır.

Tam CAN kontrolörleri uzak çerçevelere otomatik cevap verme desteğine sahiptir. Bu, baş mikro kontrolör veya işlemci üzerindeki yükün düşeceği ama ayrıca eski bilginin gönderildiği anlamına gelir. Bunu uygulamayı yazarken göz önünde tutmak çok önemlidir. Çizelge 5.7.'de Tam CAN'ın özellikleri verilmiştir.

Çizelge 5.7. Tam CAN Özellikleri.

İletim	İletim posta kutuları önce belirlenir. İletimden önce sadece veri bitleri yazılır.
Alım	Sadece alım posta kutularında belirteçleri tanımlanan mesajlar alınabilir. Posta kutuları için çift tamponlama yoktur. Tam kabul filtreleme (Sadece kesin belirteçli mesajlara izin verilir.)
Uzak çerçeve tutma	Uzak çerçevelere kontrolör tarafından otomatik cevap verilir.
Aşırı çalışma felsefesi	Yeni mesajı tutar. (Aynı belirteçli eski mesajlar kaybolacaktır.)

5.11. Mevcut Standartlar ve CAN

Endüstriyel haberleşmede kullanılan seri haberleşme standartları RS232, RS422, RS485 ve CAN'in karşılaştırılması kısaca aşağıdaki Çizelge 5.8. de gösterilmiştir.

Çizelge 5.8. Seri Haberleşme Arabirimleri.

	RS232-C	RS-422-A	RS-485	CAN
Standart	EIA RS232-C, CCITT V.24	EIA RS-422-A, CCITT V.11	EIA RS485	ISO 11898
İletim Hızı	19.200 bps (Max)	10 Mbps (Max)	10 Mbps (Max)	1 Mbps (Max)
İletim Uzaklığı	15m (Max)	1200m (Max)	1200m (Max)	1000m (Max)
Proses	Tek-sonlu voltaj arabirimi	Diferansiyel voltaj, bükümlü kablo	Diferansiyel voltaj, bükümlü kablo	Diferansiyel voltaj, bükümlü kablo
Sürücüler	1	1	32	64
Alıcılar	1	1	32	64
Prensip	Full-duplex. Point- to-point	Full-duplex. Point- to-point	Half-duplex, Multi-drop	Half-duplex, Multi-master

5.12. CAN Protokolü ile Endüstriyel Uygulamalar



Büyük araçlardaki elektronik kontrol gereksinimleri ile endüstriyel veri yollarından beklenen teknik özellikler karşılaştırıldığında bazı benzerlikler dikkat çekicidir. Bunlar düşük maliyet, zorlu elektriksel ortamlarda çalışabilirlik, yüksek gerçek zamanda erişim kapasitesi, kablolama problemleridir. Mercedes S sınıfı arabalarında standart olarak CAN kullanılmaya başlanması US tır üreticilerinin CAN'i standart olarak kabul edip 1Mbit/s hızlarına ulaşmaları endüstriyel veri yolu uygulayıcılarının da dikkatini çekmiştir. Bunun yanında ziraat ve iş makinaları imalatçıları makine imalatında özellikle tekstil makinaları, medya sektöründeki basım makinaları, saç işleme makinaları otomasyonunda karıştırma ve dozajlama makinaları üreticileri CAN'i seçmiştir. Günümüzde BMW, Mercedes , VW Passat gibi arabalarda büyük kamyonlarda yine ulaşım sektöründe metro sistemleri trenlerde denizcilik sektöründe özellikle yatlarda, medikal cihazlarda CAN protokolü kullanılmaktadır. Yine CAN protokolü ile saha veri yoluna bağlanabilen sensör-aktör ve de I/O kartları ile bir ağ oluşturmaya yönelik birçok uygulama yapılmaktadır.

Bu uygulamalar içerisinde tekstil makinaları endüstrisi CAN protokolünün göz bebeğidir. 1990 yılının başlarında Avrupa'da bir dokuma tezgahları üreticisinin tezgahın otomasyonunda , gerçek zamanlı haberleşen modüler sisteminde CAN protokollünün kullanılması ve bunu birçok tekstil makine imalatçısının takip etmesi ile alınan olumlu sonuçlar Avrupa tekstil imalatçılarının "CAN Textile Users Group" adıyla bir araya toplanmalarına neden olmuştur. Bu grup şu anda CiA'da en aktif alt gruplardandır. Tıbbi mühendisliğin yoğun olduğu bazı ilaç ve kimyasal uygulamalarda, robot ve taşımacılık sektöründe birçok kullanıcı CAN protokollünü mevcut endüstriyel veri yolu sistemleri arasında en güvenilir olduğu için tercih etmektedir. Endüstriyel CAN uygulamalarına bakıldığında CAN'in en önemli özelliğinin düşük iletişim maliyetlerine rağmen yüksek güvenilirlik olduğu görülmektedir [Sarıbay, 1998].

5.13. CAN ve Diğer Fieldbus'ların Karşılaştırılması

Önceki bölümlerde anlatılan Kontrolör Alan Ağı'nın özelliklerinin diğer popüler saha veri yolu Profibus ile karşılaştırılması Çizelge 5.9. da gösterilmiştir.

Çizelge 5.9. CANbus ve Profibus'ın Karşılaştırılması.

	CANBUS	PROFIBUS
Genel		
Açık sistem	Açık	Açık
Uluslararası Standart	ISO, CiA	Uluslararası
Eğer belirtildiyse ruhsatlı özel sistem	Ruhsat çiplere dahildir.	
Yayınlanan özellik referansı	ISO 11898, CiA DS 201-207	DIN 19245-1-2-3-4
Kullanıcı komitesi var mı?	CAN in Automation	PROFIBUS kullanıcı grubu
İşlevsellik		
Diğer fieldbus gatewayleri	Evet	ASI/MAP
Uyumlu sensör/eylemci üreticilerinin sayısı	>500	>150
Bus Haberleşmesi		
Ağ topolojisi	Hat, yıldız, point to point	RS 485 bus
Fiziksel ortam	Bakır, fiber optik ve IR	Bakır, fiber optik
Maksimum uzunluk	1000m-50kHz, 40m-1MHz	9600m-9.6kHz ila 12MHz
Maksimum nokta sayısı	Sınırlandırılmamış	127
Maksimum yanıt zamanı	0.1mS. 4bayt-1MHz	FMS=.1S. DP<1mS-12MHz
Bus protokolü veya çeşitli haberleşme katmanlarındaki protokoller	CAN, CAL, CANOpen	FMS/DP/PA
Multi-master kabiliyetler	Evet	Evet
Broadcasting yeteneği	Evet	
Maksimum hata hızı, hata bulma ve düzeltme	10E-13 HD=6	HD=4

Çizelge 5.9. CANbus ve Profibus'ın Karşılaştırılması (Devam).

Hata üzerinde tekrar	Evet	Evet
Bir noktadaki güç kaybının etkisi	Hiç	Hiç
Bir noktaya bağlanma ve ya kopmanın etkisi	Hiç	Hiç
Network redundancy	Mümkün	
Bus tekrarlayıcılar	Evet	Evet
Standart fiber optik arabirim	İsteğe bağlı	Evet
Donanım		
Radyasyon sertliği	Mümkün	Hayır
Farklı üreticilerden chipsetler	>12 üretici	Siemens, Motorola, IAM
Çipüstü hafıza (RAM, PROM, EPROM)	Evet, isteğe bağlı	Hayır
Mikroişlemciye entegre edilen çoklu fonksiyonellik	Evet	
Ticari off-the-shelf modülleri	Evet. >1000	Evet. >700
Fieldbusdan güçlenen noktalar	Her ikisi de mümkün	Evet (PA ise).
PC'ler için yazılım: Windows 95, Windows NT, Linux		Windows 95,NT, Linux.
VME için yazılım: LynxOs, VxWorks ve OS9 sürücüler	VxWorks	OS9
Ruhsat politikası	Çip ücretine dahil	Bedava
Ağ yönetimi	CAL'da Evet	Evet
Konfigürasyon yönetimi	CAL'da Evet	Evet
Endüstriyel denetçilerle entegrasyon		Evet
Uygulama yazılımı (API)	EPICS de, Aygıt+ Haberleşme profili	ALI
Aygıt tanımlama servisleri		PA
Fonksiyon blokları	Hayır	FMS ve PA ise Evet
Diagnostik araçlar (yazılım)	Evet	Evet

6. CAN ÜRÜNLERİ



1992 Mart'ında uluslararası bazda kullanıcı ve üretici organizasyonu olan CAN in Automation (CiA) kurulmuştur. Ticari bir kuruluş olmayan CiA, 250 üyesi ile pazarlama, teknik eğitim, standartlar ve çeşitli geliştirme faaliyetlerinde bulunmakta, dünya çapında forumlar ve seminerler düzenlemektedir.

Bu organizasyonun desteklediği yüzlerce çeşitlilikte CAN ürünleri vardır. Örneğin:

- ISA(PC uyumlu), PCMCIA(Laptops), paralel portlar ve Ethernet için adaptörler.
- CAN donanımlı sensörler (limit, fotoelektrik göz...).
- CAN donanımlı eylemciler (hidrolik, pünomatik, motor, röleler...).
- CAN tekrarlayıcılar (repeaters).

CiA'na üye firmaların CAN ürünlerinin listesi Çizelge 6.1. de verilmiştir [Miller Freeman, 1998].

Çizelge 6.1. CAN Ürünleri Listesi.

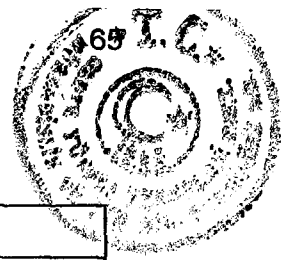
CAN Protokol Kontrolör İşlemcileri	
Advantech	ADAM5000 Series
Motorola	Motorola's CAN Microcontrollers
Motorola	MC68HC08AZ0/16/24/32 & MC68HC908AZ60 MSCAN08 On-Chip Controller Area Network
Motorola	68HC12 16-Bit Microcontrollers with On-Chip CAN Support
Motorola	68376 32-Bit Microcontrollerwith On-Chip CAN Support
Motorola	MPC555 Microcontroller with Flash EEPROM
Motorola	Motorola OSEK



Çizelge 6.1. CAN Ürünleri Listesi (Devam).

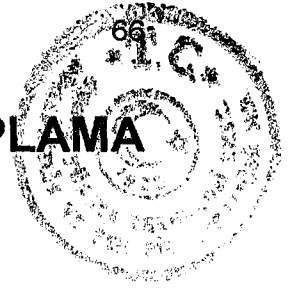
NEC Electronics	µPD78(F)081x 8-Bit Microcontrollers with DCAN
NEC Electronics	µPD78(F)082x 8-Bit Microcontrollers with DCAN
NEC Electronics	µPD78(F)094x 8-Bit Microcontrollers with DCAN
NEC Electronics	Atomic 32-Bit RISC Microcontrollers
Philips Semiconductors	8xC592/8xC598 8-Bit Microcontroller with On-Chip CAN
Philips Semiconductors	SJA1000 Stand-alone CAN Controller and XA-G3 CMOS Single-Chip 16-Bit Microcontroller
Philips Semiconductors	XA-C3 CMOS Single-Chip 16-Bit CAN 2.0B Microcontroller
Siemens	C164CI General-Purpose High-Performance MCU with OTP on Chip
Siemens	C167CR Highly Integrated Microcontroller with On-Chip CAN Module
Siemens	C505C Microcontroller with Complete CAN Capability
Siemens	C515C Multipurpose Microcontroller with On-Chip Full CAN Module
Siemens	SAE 81C90/91 Full CAN Controller
ST Microelectronics	ST10F167 Microcontroller with CAN 2.0B
ST Microelectronics	ST725x 8-Bit Microcontroller Family with CAN 2.0B
CAN İletici Çipleri	
Alcatel Microelectronics	MTC-3054 CAN Interface Chip
Philips Semiconductors	82C250/82C251/82C252 CAN Transceivers
Donanım Araçları	
Dearborn Group	Gryphon Ethernet-to-CAN
Hitex Dvelopment Tools	Toolbox for CAN Applications
I+ME Actia	CAN-WinEMU and NetTest III
I+ME Actia	Advanced CAN Technology
Ixxat Automation GmbH	Analyzing Tools, Interfaces, and Higher Layer Protocols

Çizelge 6.1. CAN Ürünleri Listesi (Devam).



Nsi SA	Traminator CAN
Vector CANtech, Inc.	CANalyzer
Vector Informatik GmbH	Vector CAN Products
Yazılım Araçları	
Keil Software	C51 Compiler Version 5.50
Keil Software	C166 Compiler Version 3.0
Tasking	C166, CAN, and Tasking: The Integrated Solution
Weidmüller Connex GmbH & Co.	DIAtool Software
Weidmüller Connex GmbH & Co.	DIAvis/Wizcon
Kart-Seviyesi Ürünler	
Arcom Control Systems	AIM104-CAN PC/104 Single Channel CAN Controller with DeviceNet Tools
National Instruments	CAN/DeviceNet Interfaces
National Instruments	AT-CAN, AT-CAN/2 Interfaces for ISA
National Instruments	PCI-CAN, PCI-CAN/2 Interfaces for PCI
Phytec	PHYTEC CAN Single Board Computers and Software
SBS GreenSpring Modular I/O	IP-DeviceNet Interface IndustryPack Module for ISA, PCI, CompactPCI, VME, and PC/104
Arabirimler ve Algılayıcılar	
Balluff	BIS-C-603-U2-DN ID Systems Interface
Balluff	Micropulse - BTL Linear Position Transducers
RM MICHAELIDES Software & Electronic GmbH	CANview Interpreting and Control Unit to Connect CAN Fieldbuses to PCs
CAN Arabirimler, Kontrolör ve Sürücüler	
Contemporary Controls	EXTEND-A-BUS Series Fieldbus Extender for DeviceNet or Smart Distributed System
Infranor	SMTBD Digital Servo Amplifier
Wago Corporation	WAGO I/O System

7. CAN ÜZERİNDE DAĞITIK VERİ TOPLAMA VE KONTROL SİSTEMİ TASARIMI



Kontrolör Alan Ağı üzerinde bir veri toplama ve kontrol sistemini tasarlarken dikkat edilmesi gereken temel hususlar vardır. Bunlar aşağıda sıra ile açıklanmıştır.

- **Prosesin tanımlanması:** Her proses önceden belirlenmiş bir dizi işlevi sağlamak için tasarlanır ve kullanılır. Prosesin genel mekanizmasının ve kullanılacak matematik model ve ilişkilerinin tanımlanması gerekir. Böylece prosesin kapasitesi, çalışma enerji domeni, sınır değerleri ve toleransları verilir.

- **Kontrol sisteminden beklenenlerin tanımlanması:** Prosesin tanımlanan işi yaparken doğru sınırlar ve toleranslarla çalışmasını, kendisinin ve çevresinin güvenliğinin sağlanmasını, beklenmeyen durumlardaki davranışını ve yeniden çalıştırılmasını kapsayan bir dizi ek işlem tanımlama zorunluluğu vardır. Bu bir taraftan süreci karmaşıktırır, diğer taraftan prosesin çalışmasına etkisi dolayısıyla başlangıçta tanımlanmış şartları değiştirebilir. Gereksinimlerin ortaya çıkarılması sırasında sorular sorulmalı ve cevaplar verilmelidir. Giriş değişkenleri, çıkış değişkenleri, kontrol sisteminden beklenen işlevler ve hesaplamalar belirlenmelidir. Kontrol sisteminin yapması gerekenler tanımlandıktan sonra, sistemin gerçekleşmesi için çözümün nasıl yapılacağına karar verilmelidir. Hangi teknolojilerin kullanılacağı seçilmelidir.

- **Kullanılacak donanımın belirlenmesi:** Bir önceki adımda belirlenen gereksinimlere ve çözüm için seçilen teknolojiye göre gerekli olan donanım belirlenmelidir. Okunacak verilerin özelliklerine göre transdüserler, bu transdüserlerin özelliklerine göre sinyal durumlama cihazları, bilgisayar kontrollü olacaksa okunan verileri bilgisayara aktarmak için veri aktarma cihazları ve kontrol edilecek elemanların özelliklerine göre de eylemlendiriciler belirlenmelidir.



• **Yazılım:** Yazılım ihtiyacı kontrol sisteminin çözümüne göre değişir. Örneğin, sistemin kontrolü bilgisayar ile yapılacaksa ve sistem ile kullanıcı arasında grafik arabirimi oluşturulması isteniyorsa, bunun için bir bilgisayar programı gerekmektedir. Bu programı hazırlamak için, veri toplama ve kontrol sistemleri için özel olarak hazırlanmış uygulama geliştirme paketlerinden yararlanılabilir veya bir programlama dili kullanılarak program yazılabilir.

7.1. Bir Tasarım Örneği

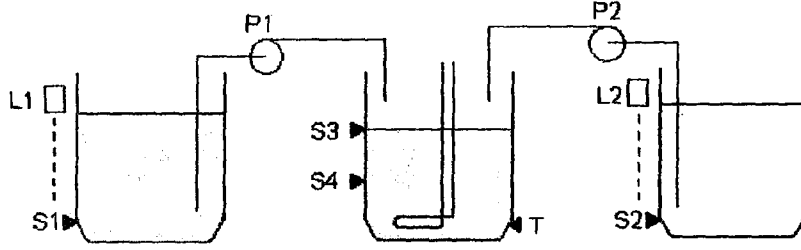
Yukarıda anlatılanların uygulaması için bir örnek tasarlanmıştır. Bu örnekte, sistem tanımlanmış, giriş ve çıkış değişkenleri belirlenmiştir. Ayrıca bu değişkenlere göre kullanılacak donanım ve özellikleri belirlenmiştir. Bu sistem ile kullanıcı arasındaki arabirimi oluşturacak bilgisayar yazılımı da genel olarak tasarlanmıştır.

Sistemin tanımı: Besleme tanklarından doldurulan bir tanktaki suyun belli bir dereceye kadar ısıtıldığı bir sistemi inceleyelim. Burada, bir su tankı ve kendisinden uzakta iki besleme tankı vardır. Besleme tanklarından ana tanka su, her besleme tankı için bir pompa aracılığı ile aktarılmaktadır. Ana tanktaki su bir ısıtıcı ile ısıtılmaktadır. Besleme tanklarına su manual olarak doldurulmaktadır. Tüm sistemin bilgisayar ile kontrolünün yapılması istenmektedir.

Sistemden beklentiler: Ana tank boş olduğu zaman birinci besleme tankından doldurulacaktır. Eğer birinci besleme tankı da boş ise ikinci besleme tankından doldurulacaktır. Ana tankta ısıtıcının çalışması için yeterli olacak kadar su olduğunda ısıtıcı çalıştırılacak ve su belli bir sıcaklığa getirildiğinde kapatılacaktır. Besleme tankları boşaldığında üzerilerine konacak birer lamba yanıp onların boş olduğunu gösterecektir ve pompa çalışıyor ise kapatılacaktır. Bu sistemin durumunu kullanıcı bilgisayar ekranından görebilecek ve sıcaklığı değiştirebilecektir.

Giriş değişkenleri: Tanklardaki su seviyeleri seviye sensörleri ile ölçülecektir. Bu seviyeler boş veya dolu şeklinde olacağı için sayısal girişler tercih edilmektedir. Ana tanktaki suyun sıcaklığı örneksel giriş olacaktır.

Çıkış değişkenleri: Besleme tankları boş olduğunda lambalar yanacağı, dolu olduğunda söneceği için sayısal çıkış olacaktır. Ana tanka su akıtılması için kullanılan pompayı çalıştıran motor açık veya kapalı olacağı için ve ayrıca su istenilen sıcaklığa gelindiğinde ısıtıcı kapatılacağı için bu çıkışlar da sayısal olacaktır.



Şekil 7.1. Kontrol Edilecek Sistemin Genel Yapısı.

Şekil 7.1.'de kontrol edilecek sistemin genel yapısı gösterilmektedir. Burada girişler; seviyeler için Sensör1 (S1), Sensör2 (S2), Sensör3 (S3), Sensör4 (S4), sıcaklık için sıcaklık sensörü (T) olarak belirtilmiştir. Çıkışlar da Pompa1 (P1), Pompa2 (P2), Lamba1 (L1), Lamba2 (L2), Isıtıcı (I) olarak belirtilmiştir.

Çizelge 7.1. Girişler ve Çıkışlar Arasındaki Mantıksal İlişki.

	T	S1	S2	S3	S4	P1	P2	L1	L2	I
0	0	0	0	0	0	0	0	1	1	0
1	0	0	0	0	1	0	0	1	1	1
2	0	0	0	1	0	X	X	X	X	X
3	0	0	0	1	1	0	0	1	1	1
4	0	0	1	0	0	0	1	1	0	0
5	0	0	1	0	1	0	1	1	0	1
6	0	0	1	1	0	X	X	X	X	X
7	0	0	1	1	1	0	0	1	0	1
8	0	1	0	0	0	1	0	0	1	0
9	0	1	0	0	1	1	0	0	1	1
10	0	1	0	1	0	X	X	X	X	X
11	0	1	0	1	1	0	0	0	1	1
12	0	1	1	0	0	1	0	0	0	0
13	0	1	1	0	1	1	0	0	0	1
14	0	1	1	1	0	X	X	X	X	X

Çizelge 7.1. Girişler ve Çıkışlar Arasındaki Mantıksal İlişki (Devam)

15	0	1	1	1	1	0	0	0	0	1
16	1	0	0	0	0	X	X	X	X	X
17	1	0	0	0	1	0	0	1	1	0
18	1	0	0	1	0	X	X	X	X	X
19	1	0	0	1	1	0	0	1	1	0
20	1	0	1	0	0	X	X	X	X	X
21	1	0	1	0	1	0	1	1	0	0
22	1	0	1	1	0	X	X	X	X	X
23	1	0	1	1	1	0	0	1	0	0
24	1	1	0	0	0	X	X	X	X	X
25	1	1	0	0	1	1	0	0	1	0
26	1	1	0	1	0	X	X	X	X	X
27	1	1	0	1	1	0	0	0	1	0
28	1	1	1	0	0	X	X	X	X	X
29	1	1	1	0	1	1	0	0	0	0
30	1	1	1	1	0	X	X	X	X	X
31	1	1	1	1	1	0	0	0	0	0

Girişler ve Çıkışlar arasındaki mantıksal ilişki yukarıdaki Çizelge 7.1.'de gösterilmiştir. Bu çizelgeden her bir çıkış için,

$$P1 = S1.S3^I + S1.S4^I$$

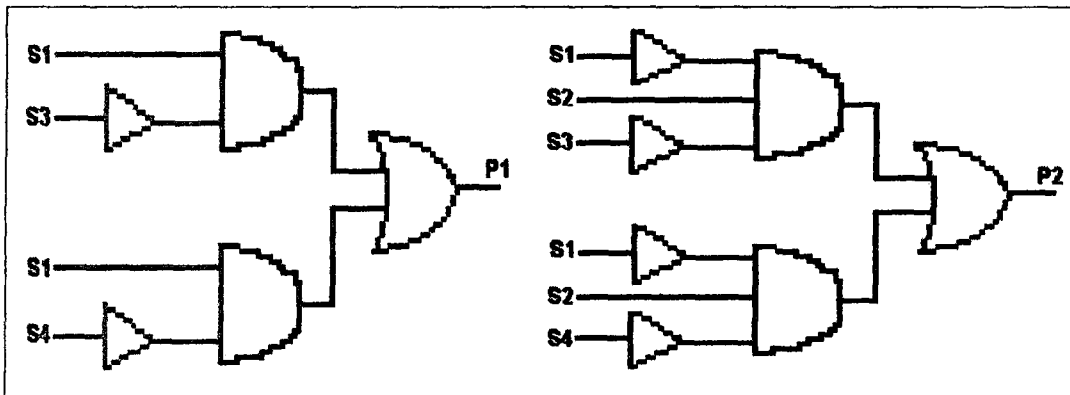
$$P2 = S1^I.S2.S3^I + S1^I.S2.S4^I$$

$$L1 = S1^I$$

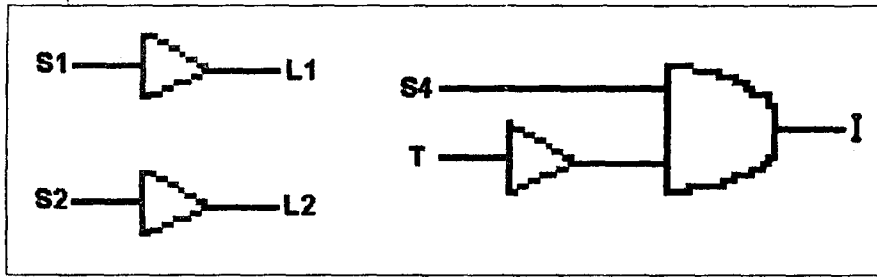
$$L2 = S2^I$$

$$I = S4.T^I$$

mantıksal bağıntıları elde edilmiştir. Şekil 7.2. ve Şekil 7.3.'de bu mantıksal bağıntılar şema olarak gösterilmiştir.



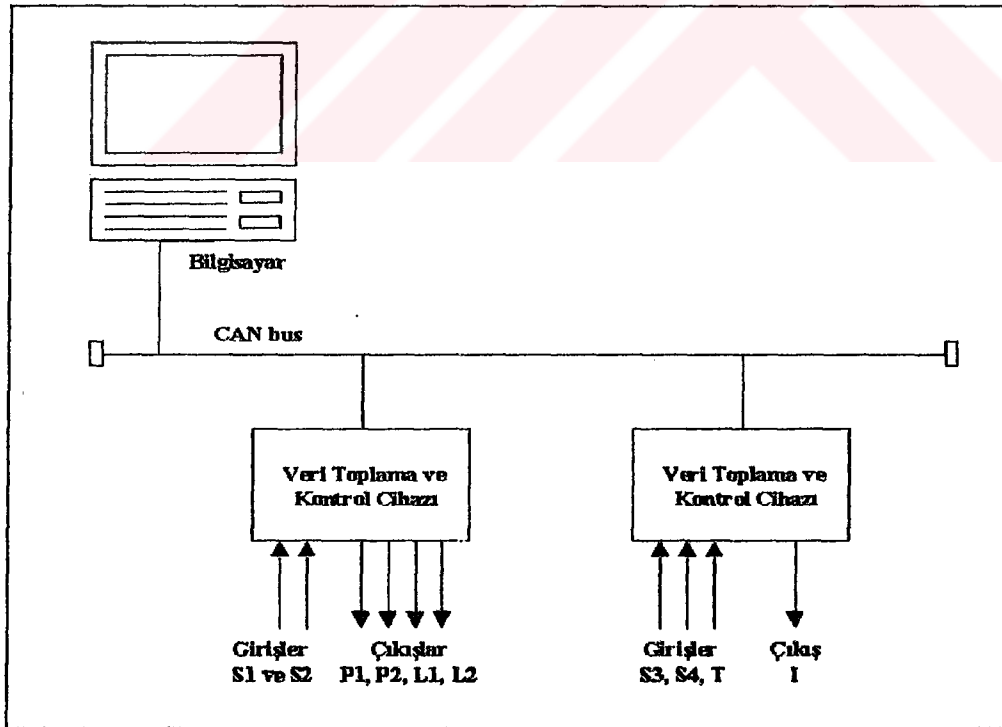
Şekil 7.2. P1 ve P2 nin Mantıksal Gösterimi.



Şekil 7.3. L1, L2 ve I'nın Mantıksal Gösterimi.

Girişler ve çıkışlar arasında gösterilen bu mantıksal bağıntılarda, hangi girişin hangi çıkışın kontrolünü etkilediği görülmektedir. Çıkışların kontrolü bilgisayar ile yapılacağından bu bağıntılardan bilgisayar programı yapılırken faydalanılacaktır.

Besleme tankları ile ana tank birbirlerinden uzak olduğu varsayıldığı için besleme tanklarından okunacak veriler ve burada kontrol edilecek veriler bir veri toplama adasına, ana tanktan okunacak ve kontrol edilecek veriler de diğer bir veri toplama adasına bağlı düşünülmüştür. Bu iki veri toplama ve kontrol adası ağ üzerinden (CANbus) bilgisayara bağlanmaktadır. Sistem Şekil 7.4.'de şematik olarak gösterilmiştir.



Şekil 7.4. Sistemin Şematik Gösterimi.



7.1.1. Donanımın Özellikleri ve Bağlantılar

Kontrolör Alan Ağı'nın özellikleri ve avantajları önceki bölümlerde incelenmiştir. Bu sistem CAN üzerinde çalışacağı için kullanılacak veri toplama ve kontrol cihazlarının CAN destekli seçilmesi gerekmektedir.

Bu sistemde seviye sensörleri sayısal girişler olarak, sıcaklık sensörü örneksel giriş olarak, pompalar, lambalar ve ısıtıcı ise sayısal çıkış olarak belirlenmiştir. Bu yüzden sensörler ve eylemlendiriciler bu kriterlere göre seçilmelidir.

Veri toplama ve kontrol cihazı olarak da CAN destekli ürünler arasından Advantech firmasının kontrolör alan ağı üzerinde dağıtık veri toplama ve kontrol sistemi kurmak için tasarladığı ADAM 5000/CAN ürün serisi seçilmiştir. Bu cihazlar bir bilgisayar tarafından uzaktan kontrol edilirler ve veri toplama ve kontrol işinde esneklik sağlarlar [Advantech, 1997].

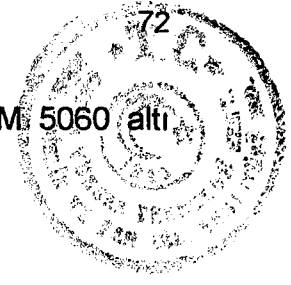
ADAM 5000/CAN ünitesi, çok kanallı I/O modüllerinden veri toplama, izleme ve kontrolünü yapabilir. Ayrıca akıllı sinyal durumlama, örneksel I/O, sayısal I/O ve CAN haberleşmesini sağlar. Bir üniteye, 64 giriş çıkış noktasına kadar 4 adet I/O modülü takılabilir. Bu ünite kendisini kontrol eden bilgisayar ile CAN ağı üzerinden haberleşir.

Çok kanallı I/O modüllerinin tek bir üniteye bağlanabilmesi ve bu ünitenin sinyal durumlama işlemini bünyesinde bulundurması bu cihazın avantajlarından biridir.

Advantech firması, bilgisayarların CAN'e bağlanması için PCL-841 iki port CAN arabirim kartı veya ADAM 4525 RS-232 – CAN dönüştürücü olmak üzere iki çözüm sunmaktadır. CAN arabirim kartının iletim hızı 1Mbps'e kadardır. RS-232 – CAN dönüştürücünün CAN hızı 1Mbps'e kadar olmasına rağmen RS232 hızı 115.2 kbp'stir.

ADAM 5000/CAN ünitesi için giriş modülleri; ADAM 5017 sekiz kanal örneksel giriş, ADAM 5018 yedi kanal termokupl giriş ve ADAM5051 onaltı kanal sayısal giriş modülleridir. Çıkış modülleri ise ADAM 5024 dört kanal

örneksel çıkış, ADAM 5051 onaltı kanal sayısal çıkış ve ADAM 5060 altı kanal röle çıkış modülüdür.



7.1.2. Kullanılan Donanım

Sistemi izleme ve kontrol için bir bilgisayar kullanılır. Bu bilgisayar Windows 9x veya NT işletim sistemi çalıştırabilmelidir. ADAM 5000/CAN serisinin bu işletim sistemleri için cihaz tanıtım yazılımı vardır.

Bilgisayarın CANbus'a bağlanabilmesi için bir adet Advantech PCL-841 CAN kartı kullanılır. Sadece RS232 portu olan bilgisayarları CANbus'a bağlamak için de ADAM 4525 RS232 – CAN dönüştürücü kullanılır.

Ana tank ve besleme tankları için ayrı olacağından iki adet Advantech ADAM 5000/CAN dağıtık veri toplama cihazı kullanılır.

Birinci ADAM 5000/CAN ünitesine bağlanmak üzere; besleme tanklarındaki seviyeleri okumak için bir adet ADAM 5051, 16 kanal sayısal giriş modülü, pompaları ve lambaları kontrol etmek için bir adet ADAM 5060, 6 kanal röle çıkış modülü gerekir.

İkinci ADAM 5000/CAN ünitesine bağlanmak üzere; sıcaklığı ölçmek için termokupl kullanılacağından bir adet ADAM 5018, 7 kanal termokupl giriş modülü ile ısıtıcıyı kontrol etmek için bir adet ADAM 5060, 6 kanal röle çıkış modülü gerekir.

ADAM 5000/CAN ünitesini beslemek için güç kaynağı (+10V - +30V) gerekir.

Tanklardaki su seviyelerini ölçmek için dört adet iletkenlik tip seviye anahtarı, suyun sıcaklığını ölçmek için bir adet termokupl kullanılır.

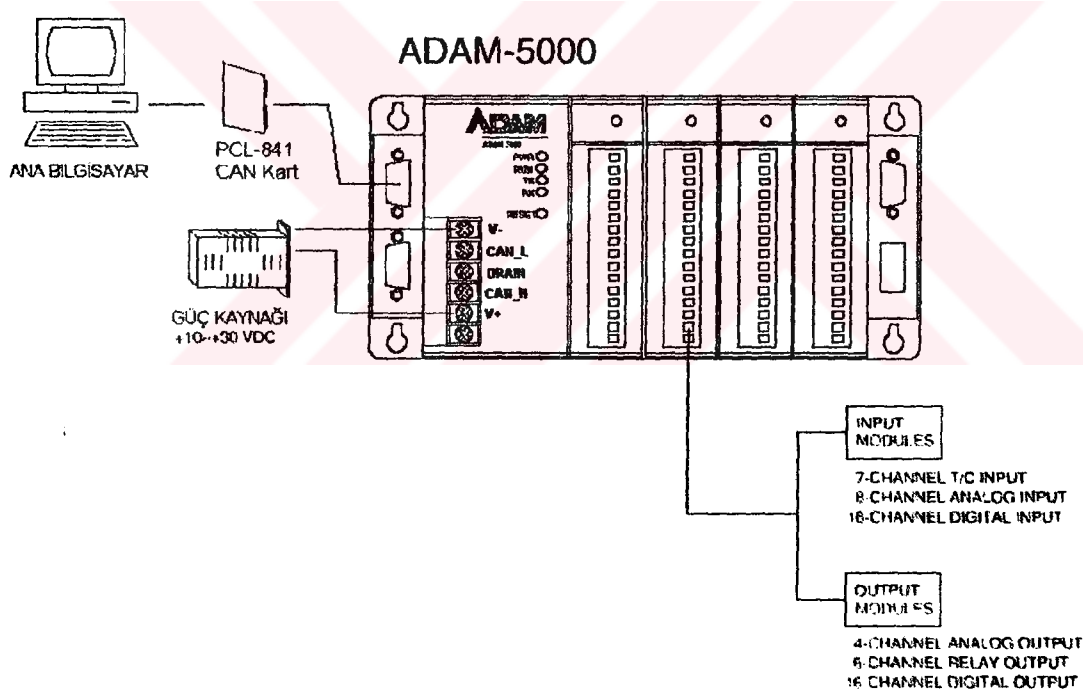
Su transferi için iki adet pompa, tanklar için iki adet lamba, suyu ısıtmak için de bir adet su ısıtıcısı gerekir.

ADAM 5000/CAN cihazları ile bilgisayar arasında oluşturulacak ağ için (CANbus) korumalı bükümlü kablo gerekir. Ayrıca cihaz bağlantıları için de DB9 konnektör gerekir.

PCL 841 CAN Arabirim Kartı: 2 adet CAN portuna sahiptir. İletim hızı maksimum 1 Mbps'tir.

ADAM 5000/CAN: Bir ana ünite ve I/O modüllerinden oluşur. Ana ünite bir CPU kartı, bir güç regülatörü, 4-slot, bir RS232 haberleşme portu ve bir çift DB9 CAN portunu kapsar. CPU sistemin kalbidir ve tüm I/O modüllerinden veri toplama ve kontrol, termokupl linearizasyon, haberleşme yazılımı ve komut seti, kalibrasyon yazılımı ve komut seti, alarm izleme, sistem parametrelerini tutan EEPROM'u yönetme, veri dönüştürme ve diagnosis fonksiyonlarına sahiptir.

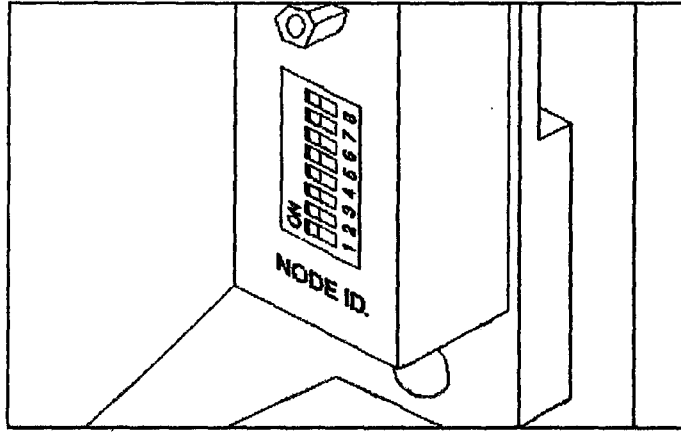
ADAM5000 sistemi standart endüstriyel regülesiz 24 V_{DC} güç kaynağı için tasarlanmıştır. +10 - +30 V_{DC} aralığındaki herhangi bir güç ünitesini kabul eder. Güç kaynağı en fazla 5 V ripple ile sınırlıdır. Şekil 7.5.'de ADAM 5000/CAN'in bilgisayar bağlantısı gösterilmiştir.



Şekil 7.5. ADAM 5000/CAN Modülünün Bilgisayar Bağlantısı.

ADAM 5000/CAN, uygulama katmanında iki açık protokolü destekler, DeviceNet ve CANOpen. Default haberleşme protokolü DeviceNet'tir.

Baudrate ve ağ adresini set etmek için ADAM 5000 üzerindeki 8 bitlik Şekil 7.4.'de gösterilen DIP Siviç kullanılır. 1 ila 6. bitler ağ adresini (Nokta ID) set etmek için kullanılır. 0'dan 63'e kadar (00h – 3Fh) adres kullanılabilir. ON = 1, OFF = 0 ifade eder. Bir ağdaki master için 00h adresi ayrılmıştır. 7. ve 8. bitler baudrate set etmek için kullanılır.



Şekil 7.6. DIP Siviç.

Çizelge 7.2. Ağ Adresi Set Tablosu.

Nokta ID	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6
01h	ON	OFF	OFF	OFF	OFF	OFF
02h	OFF	ON	OFF	OFF	OFF	OFF
03h	ON	ON	OFF	OFF	OFF	OFF
.						
3Fh	ON	ON	ON	ON	ON	ON

Çizelge 7.2.'de DIP siviçdeki anahtarların konumlarına göre nokta belirtecinin kaç olacağı gösterilmektedir. Çizelge 7.3.'de ise anahtar durumlarına ve kullanılan kullanıcı katmanı protokolüne göre baudrateler gösterilmektedir.

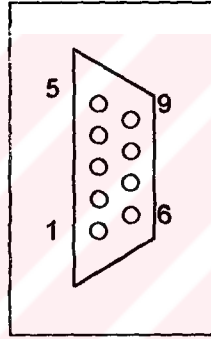
Çizelge 7.3. Baudrate Set Tablosu.

Bit 7	Bit 8	DeviceNet Baudrate	CANOpen Baudrate
OFF	OFF	125 kbps	20 kbps
ON	OFF	250 kbps	125 kbps
OFF	ON	500 kbps	500 kbps
ON	ON	-	1 Mbps

CAN portu bağlantısının pin numaraları Çizelge 7.4'da gösterilmiştir. Bu bağlantılarda bükümlü kablo kullanılır ve CAN_H pinine sarı renkli tel, CAN_L pinine yeşil renkli tel bağlanır.

Çizelge 7.4. CAN Portu Pin Bağlantıları.

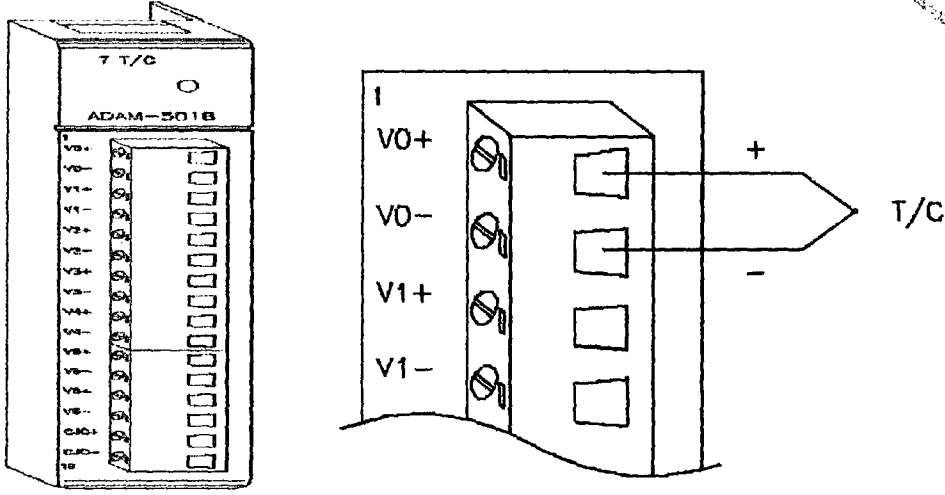
Pin No.	Tanım
Pin 1	Kullanılmıyor
Pin 2	CAN_L
Pin 3	GND
Pin 4	Kullanılmıyor
Pin 5	Kullanılmıyor
Pin 6	Kullanılmıyor
Pin 7	CAN_H
Pin 8	Kullanılmıyor
Pin 9	Kullanılmıyor



ADAM 5018: 16 bit, 7 kanal termokupl giriş modülüdür. Tüm kanallardaki giriş aralıkları programlanabilir. Milivolt girişlerini (+- 150mV, +- 500mV), voltaj girişlerini ($\pm 1V$, +-5V ve +- 10V), akım girişlerini (+-20mA, 125 Ω) ve termokupl girişlerini (J,K,T,R,S,E,B) kabul eder. Örnekleme hızı saniyede 10 örnektir.

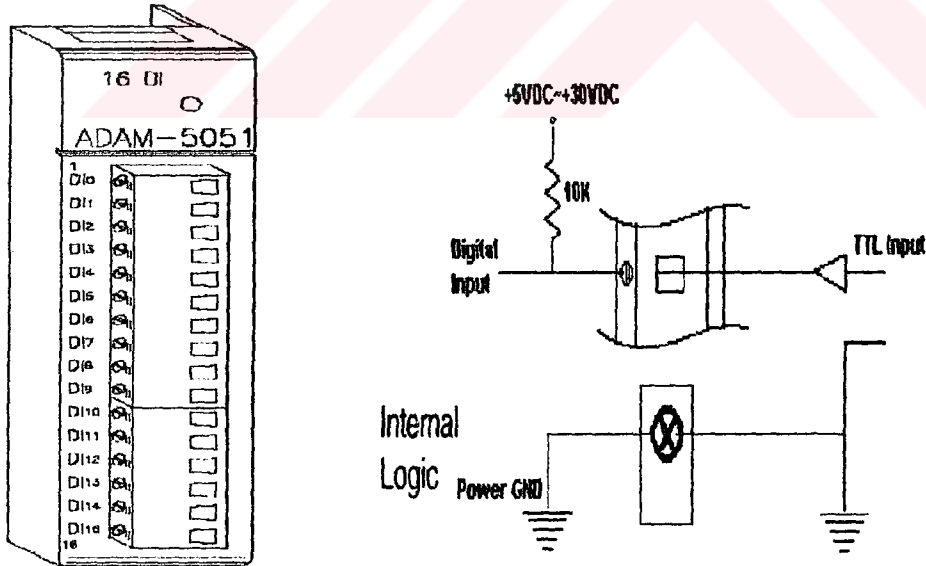
Modül, ana bilgisayara veriyi mühendislik birimleri (mV, V, mA veya °C) ile yollar. Sıcaklık ölçümünü düzeltmek için terminal üstünde bir CJC (Cold

Junction Sensor Calibration) tasarlanmıştır. Şekil 7.7., ADAM 5018 modülünün şeklini ve uygulama bağlantısını göstermektedir.



Şekil 7.7. ADAM 5018 Modülü ve Uygulama Bağlantısı.

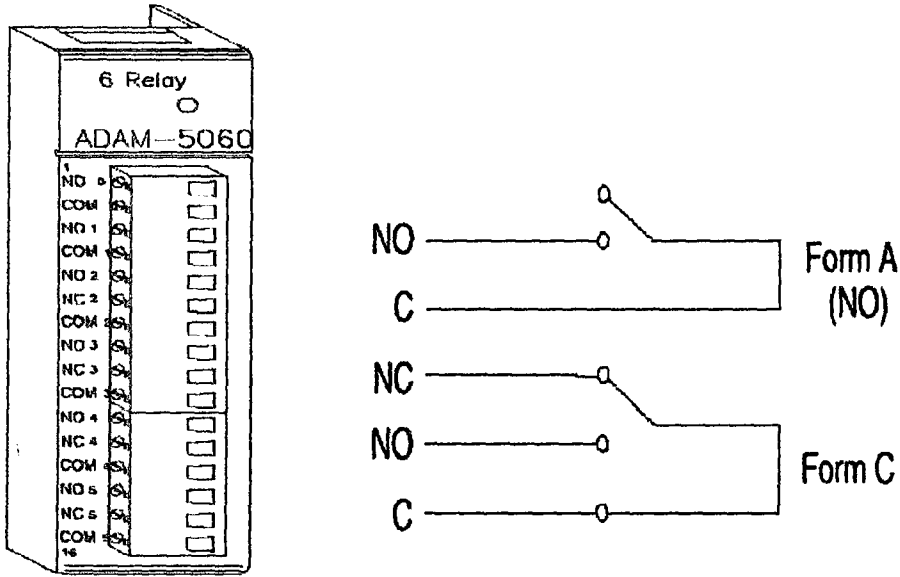
ADAM 5051: 16 kanal sayısal giriş modülüdür. Bu modül, yüksek güçteki DC veya AC hatlar ve TTL lojik sinyalleri arasındaki arabirimleri izleyebilir ve kontrol edebilir. ADAM 5051 Modülünün uygulama bağlantısı Şekil 7.8.'de gösterilmiştir.



Şekil 7.8. ADAM 5051 Modülü ve Uygulama Bağlantısı.

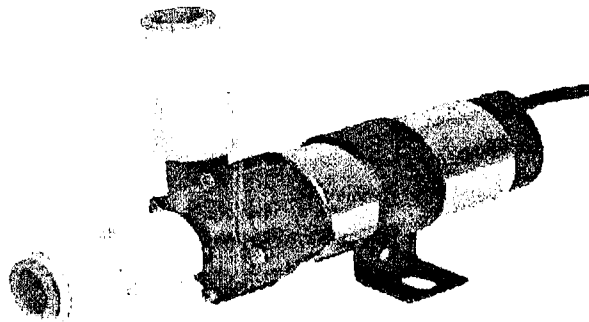
77

ADAM 5060: 6 kanal röle çıkış modülüdür. İki tanesi Form A, 4 tanesi Form C'dir. ADAM 5060'ın uygulama bağlantısı Şekil 7.9.'de gösterilmiştir.



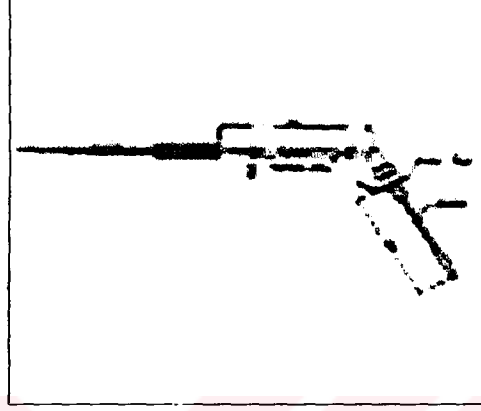
Şekil 7.9. ADAM 5060 Modülü ve Uygulama Bağlantısı.

Pompa: Bu pompa ısıtma sistemlerinde soğutulan veya ısıtılan suyun sirkülasyonu için imal edilmiştir. Motor sürekli operasyon içindir. Pompanın rotasyonu saat yönündedir ve yatay veya dikey olarak monte edilebilir. Eğer dikey olarak monte edilirse motor pompa yüzeyinin üzerine monte edilmelidir. Yatay olarak monte edilirse, çıkışı yukarıyı göstermelidir. Pompa kuru olarak çalıştırılmamalıdır. Operasyon süresince uygulanan gerilimin polaritesine dikkat edilmelidir. Terminaller renk kodludur. Pozitif besleme için mavi, Negatif besleme için kahverengi renkteki terminaller kullanılmaktadır. Şekil 7.10. pompanın resmini göstermektedir.



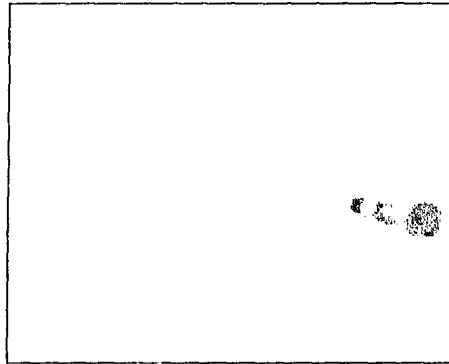
Şekil 7.10. Pompa.

Termokupl: Portatif termokupllar çok çeşitli alanlarda geniş uygulama imkanına sahiptir. Gerek ortamların, gerekse yüzeylerin sıcaklıklarının ölçülmesinde portatif termokupllar yaygın olarak kullanılırlar. Şekil 7.11 bir portatif termokupl göstermektedir. Termokupllar ile cihazlar arasındaki bağlantılar özel kablolar ile yapılır. Bu kablolar termokupl kompanzasyon kabloları olarak adlandırılır. Üretilen tüm çeşit termokupllar için değişik izolasyonlarda kompanzasyon kabloları temin edilebilir.



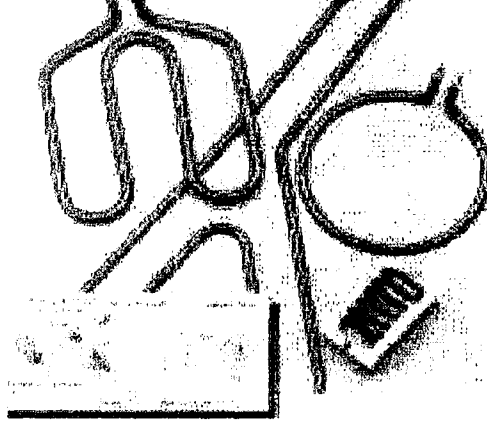
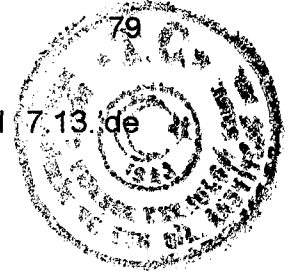
Şekil 7.11. Portatif Termokupl.

Seviye Anahtarı: İletkenlik tip seviye anahtarı, sıvı malzemelerin algılanmasında geniş kullanım alanına sahiptir. Bu seri anahtarlar iletken sıvılar için kullanılabilirler. Tanklarda, çeşitli kaplarda bulunan sıvıların alt, üst ve ara seviyelerinin algılanması için tercih edilirler. Şekil 7.12. bu anahtarı göstermektedir.



7.12. İletkenlik Tip Seviye Anahtarı.

Isıtıcı: Proses ısı kontrolü için tasarlanmış bir ısıtıcı şekil 7.13. de gösterilmektedir.

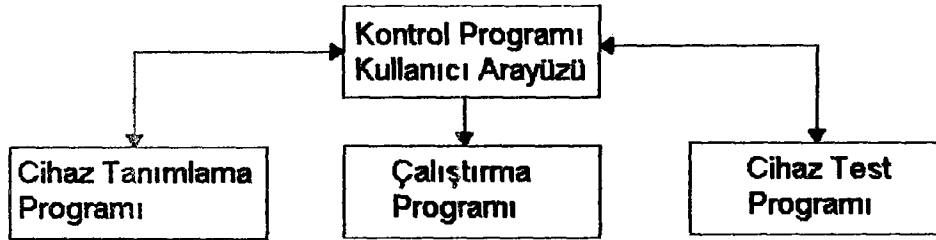


Şekil 7.13. Proses Isıtıcısı.

7.1.3. Yazılım

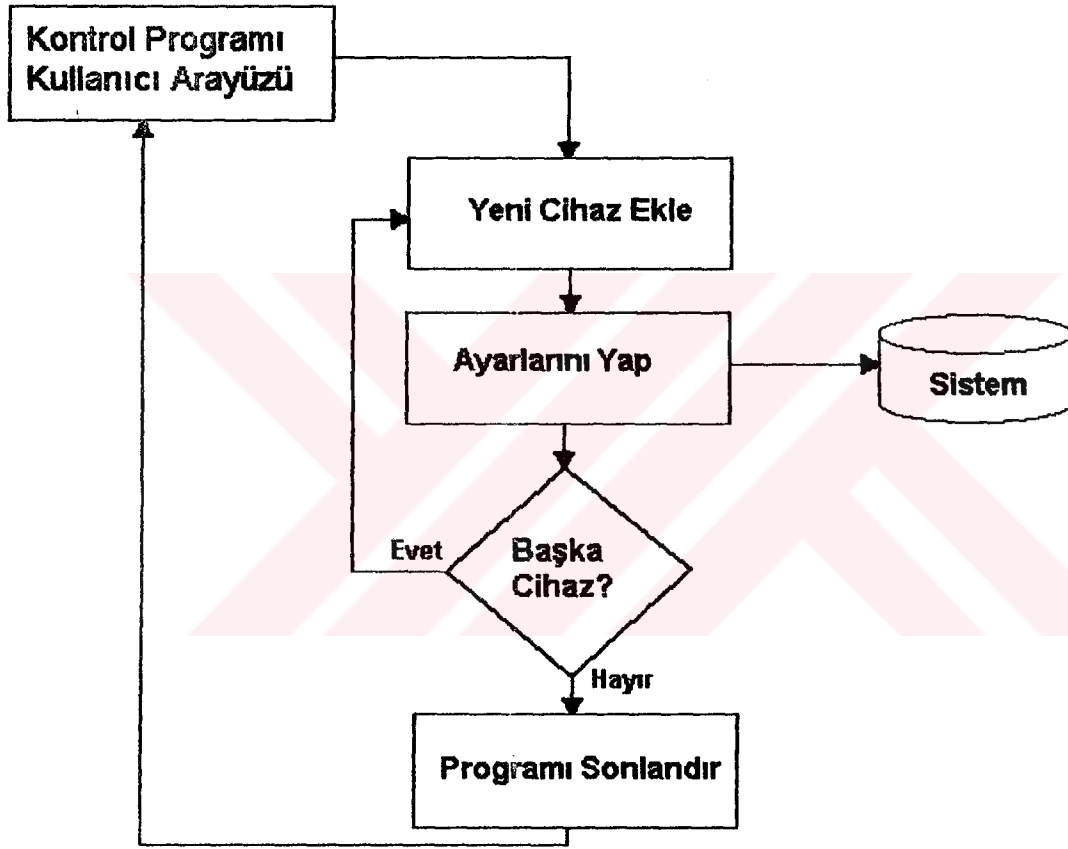
Sistem için gerekli olan kullanıcı arabirimi Borland Delphi® uygulama geliştirme aracı kullanılarak oluşturulmuştur. Arabirim 32 Bit Windows platformlarında çalışmaktadır.

Hazırlanan SCADA programında kullanıcı, firma tarafından yapılmış cihazları set eden programı çağırıp cihaz ayarlarını yeniden düzenleyebilir, cihazların çalışıp çalışmadığını test edebilir. Daha sonra da sistemin çalışmasını izleyebilir. Arabirim üzerinde tankların ve pompaların durumlarını belirten görsel araçlar ve ilgili parametrelerin izlendiği bilgi kutucukları mevcuttur.

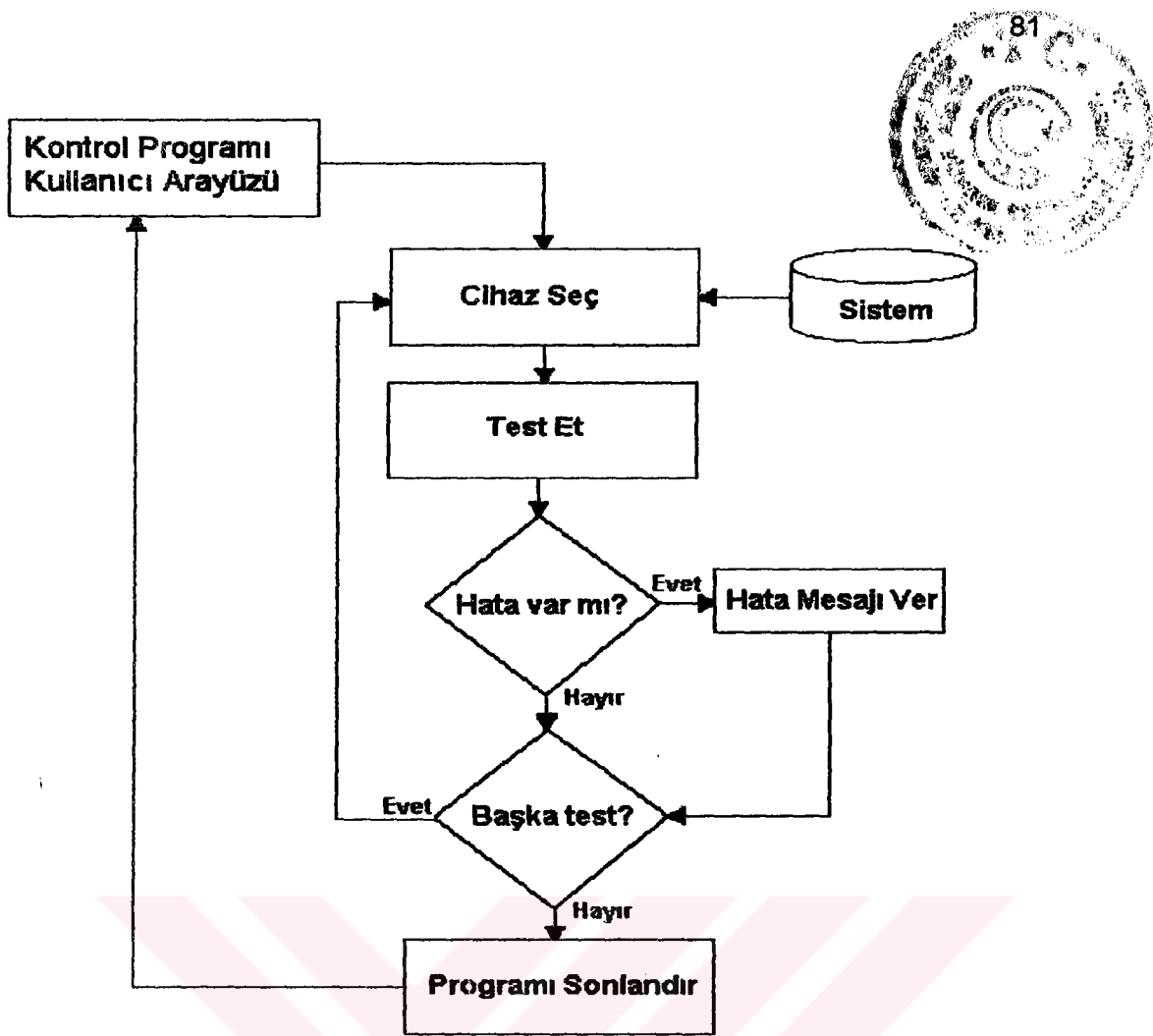


Şekil 7.14. Programın Genel Diyagramı.

Programın genel diyagramı Şekil 7.14. de gösterilmiştir. Kullanıcı grafik arayüzünden çağrılabilen üç program vardır. Bunlardan cihaz tanımlama programı, firmanın cihazlar için hazırladığı programdır. Bu programın bilgisayara önceden yüklenmiş olması gerekir. Sisteme dahil cihazların durumlarını kontrol programını çalıştırmadan önce test etmek için cihaz test alt programı vardır. Bu alt program ile sistem kontrolü çalıştırılmadan önce cihazlar ile bilgi alış verişi test edilebilir. Prosesin kontrolünü yapan ve ekran görüntüsünü yenileyen esas program şekilde çalıştırma programı olarak ifade edilmiştir.



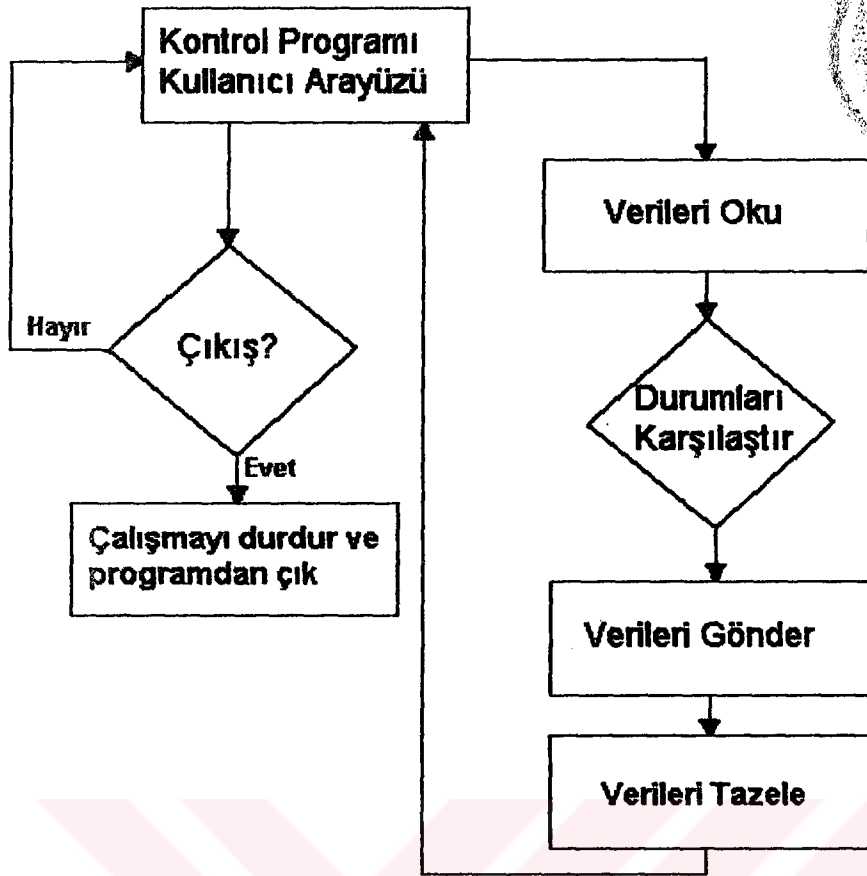
Şekil 7.15. Cihaz Tanımlama Programının İş Akış Şeması.



Şekil 7.16. Cihaz Test Alt Programının İş Akış Şeması.

Cihaz tanımlama programının ve tanımlanmış cihazları test eden alt programın iş akış şemaları Şekil 7.15. ve Şekil 7.16.'da gösterilmiştir.

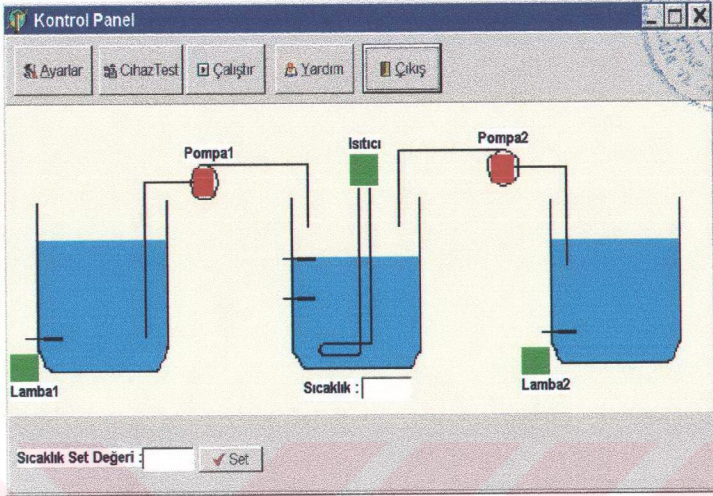
Sistemin çalışması kısmında olan işlerin iş akış diyagramı Şekil 7.17.'de gösterilmiştir. Burada, prosesden veriler okunur. Okunan bu veriler, daha önce belirtilen mantıksal bağıntılara göre hangi çıkışları etkiliyorsa, bu çıkışlara veri gönderilir. Kullanıcı ekranında da sistemdeki elemanların durumları grafik olarak gösterilir.



Şekil 7.17. Sistemin Çalışmasının İş Akış Diyagramı.

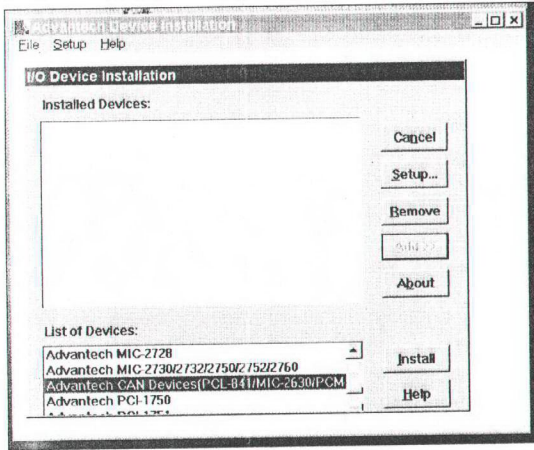
Kullanıcı ekranı, Şekil 7.18.'de gösterilmiştir. Bu ekranda, üst kısımda alt programların çağırılması için butonlar, orta kısımda fiziksel sistemi ifade eden resim, alt kısımda da kullanıcının istediği sıcaklığı set edebilmesi için bir yazı kutucuğu ve set butonu vardır.

Fiziksel sistemi ifade eden resim üzerinde cihazların kapalı veya açık olma durumlarını gösteren lambalar vardır. Pompalar ve ısıtıcı çalışıyorsa üzerlerindeki lamba yeşil, çalışmıyorsa kırmızı olacaktır. Besleme tanklarında su varsa bunu gösteren lambalar yeşil, yoksa kırmızı olacaktır. Ayrıca ana tanktaki suyun kaç derecede olduğunu gösteren bir alan vardır.

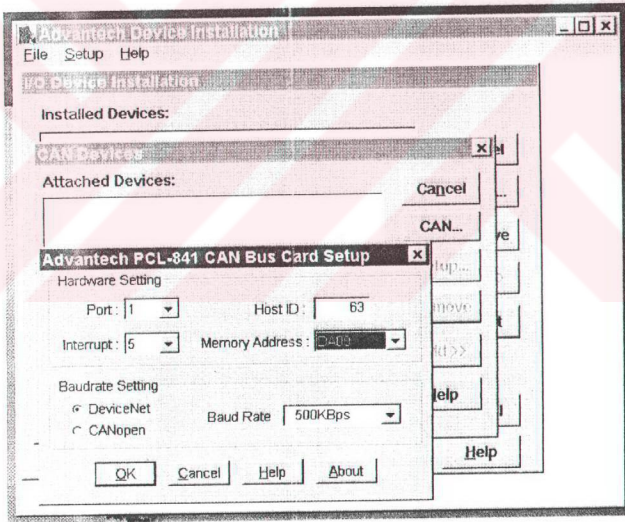


Şekil 7.18. Kullanıcı Ekranı.

Kullanıcı ekranındaki ayarlar butonuna tıklayınca Advantech Cihaz Tanımlama programı çalışır. Bu çalışan programda menüden Setup'a tıklayınca Şekil 7.19.'da gösterilen I/O cihaz tanımlama penceresi açılır. Açılan bu pencerede Add butonuna tıklayınca pencerede cihazların listesi belirir. Bu listeden tanıtılacak cihaz seçilir ve Install butonuna tıklanır. Bu sefer seçilen cihaza göre yeni bir pencere açılır. Biz örneğimizde CAN arabirim kartı kullanacağımız için Advantech CAN devices seçtiğimizi seçiyoruz ve "CAN Devices" penceresi karşımıza çıkıyor. Bu pencerede CAN butonuna tıklıyoruz ve CAN portumuzu seçip ayarlarını yapıyoruz. Bu ayarlarda kullanacağımız portun numarasını, kartın kullandığı kesme numarasını hafıza adresini ve Host ID yazan alan ile ana makine numarasını ayarlıyoruz. Kullanılacak uygulama katmanı protokolünü seçip, buna göre baudrate'i ayarlıyoruz. Varsayılan olarak DeviceNet protokolü ayarlıdır. Şekil 7.20.'de tüm bunların yapıldığı pencere gösterilmektedir.



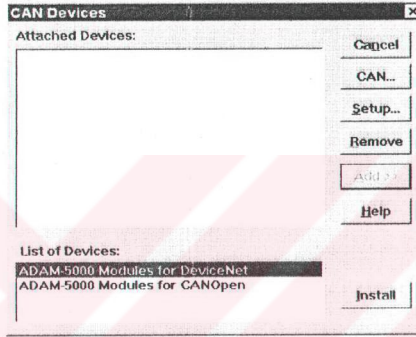
Şekil 7.19. I/O Cihaz Tanımlama Penceresi.



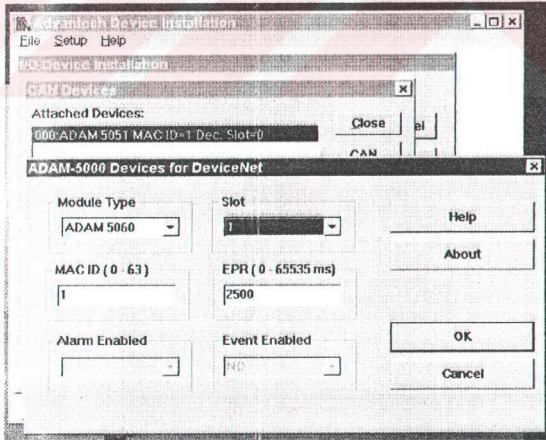
Şekil 7.20. CAN Arabirim Kartı Ayar Penceresi.

Kart ayarları yapıldıktan sonra "CAN Devices" penceresinde Add butonu aktif olur ve buna tıklayınca yeni bir cihaz listesi gelir. Şekil 7.21.'de gösterilmiştir. Bu listeden ADAM 5000 DeviceNet Modülleri seçeneği seçilir

ve Install butonuna tıklanır. Modül isimlerinin ve parametrelerinin olduğu Şekil 7.22'de gösterilen yeni bir pencere açılır. Bu pencereden hangi modülü kullanacağımızı seçip MAC ID alanına bağlı olduğu ADAM 5000 cihazının MAC ID'sini, slot alanına da hangi slota bağlı olduğunu yazıyoruz. Alarm özelliği olanlarda bu özelliği aktif veya pasif konuma getiriyoruz. Event Enabled alanı evet seçilirse, giriş modüllerinde modül, kendisine bağlı olan olay konum değiştirdiğinde yani ON ise OFF veya tam tersi olduğunda bunu tüm ağa yayınlayacaktır. Modül ayarlama işlemi her modül için tek tek yapılır.



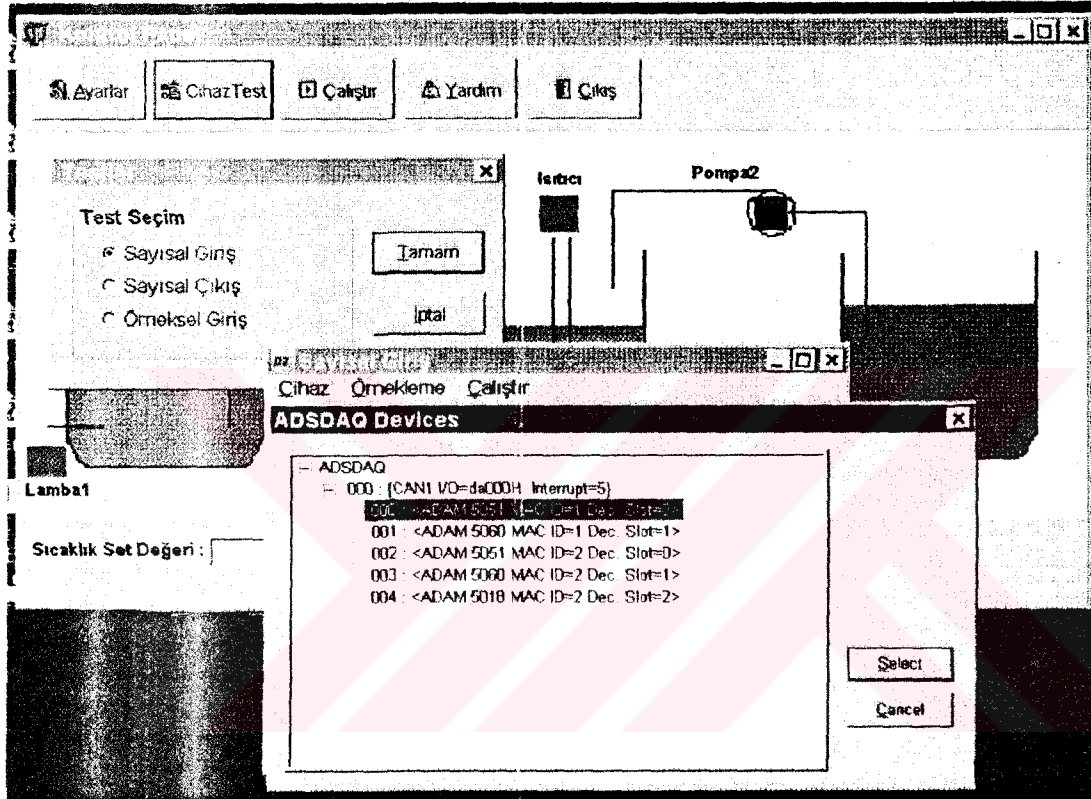
Şekil 7.21. Modül Seçim Penceresi.



Şekil 7.22. Modül Ekleme Penceresi.

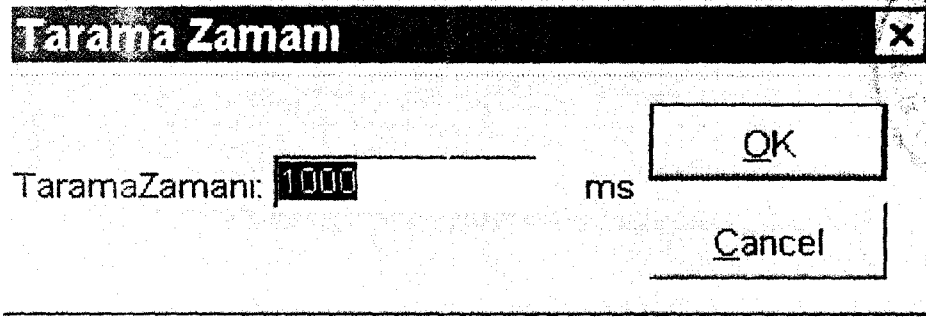
Tüm cihazlar tanımlandıktan sonra Close butonuna tıklanarak pencereler kapatılır ve programdan çıkılır. Ayarlarda değişiklik yapılmak istendiğinde program tekrar çağırılıp yukarıdaki işlemler tekrarlanabilir.

Kontrol panelinden cihaz test butonuna tıklanınca hangi tür test isteniyor ise onun seçimi için bir pencere çıkar. Bu pencerede istenilen seçilip tamam butonuna tıklanırsa o teste ait cihaz seçim için başka bir pencere karşınıza çıkar.



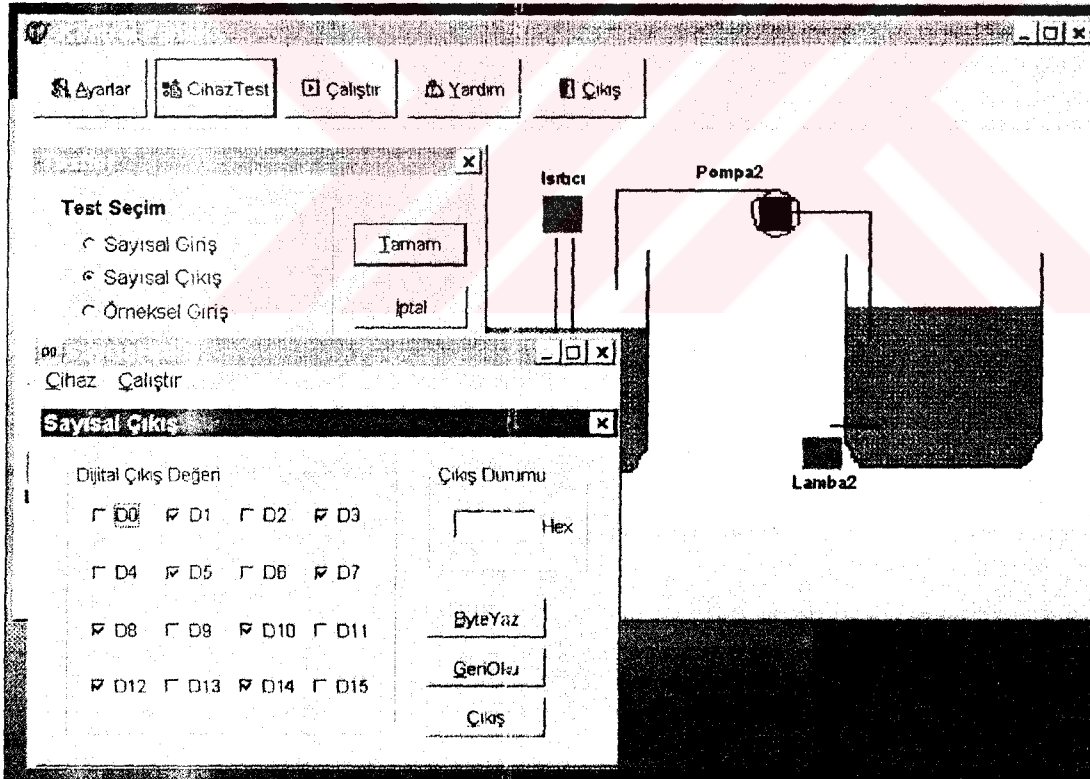
Şekil 7.23. Sayısal Giriş Test Ekranı.

Test seçim penceresinde sayısal giriş testi seçildiğinde Şekil 7.23.'de gösterilen sayısal giriş cihaz seçim penceresi çıkar. Burada menüden cihaza tıklayınca daha önce bilgisayara tanıtılmış cihazların listesi çıkar. Bu listeden test edilmek istenen sayısal giriş modülü seçilir. Tarama seçeneğine tıklanınca Şekil 7.24.'de gösterilen bilgisayarın kaç milisaniye ara ile porttan veri okumasının ayarlanacağı pencere açılır. Menüden çalıştır seçeneği tıklanınca porttan veri okunmaya başlar eğer bir hata varsa hata mesajı verir.



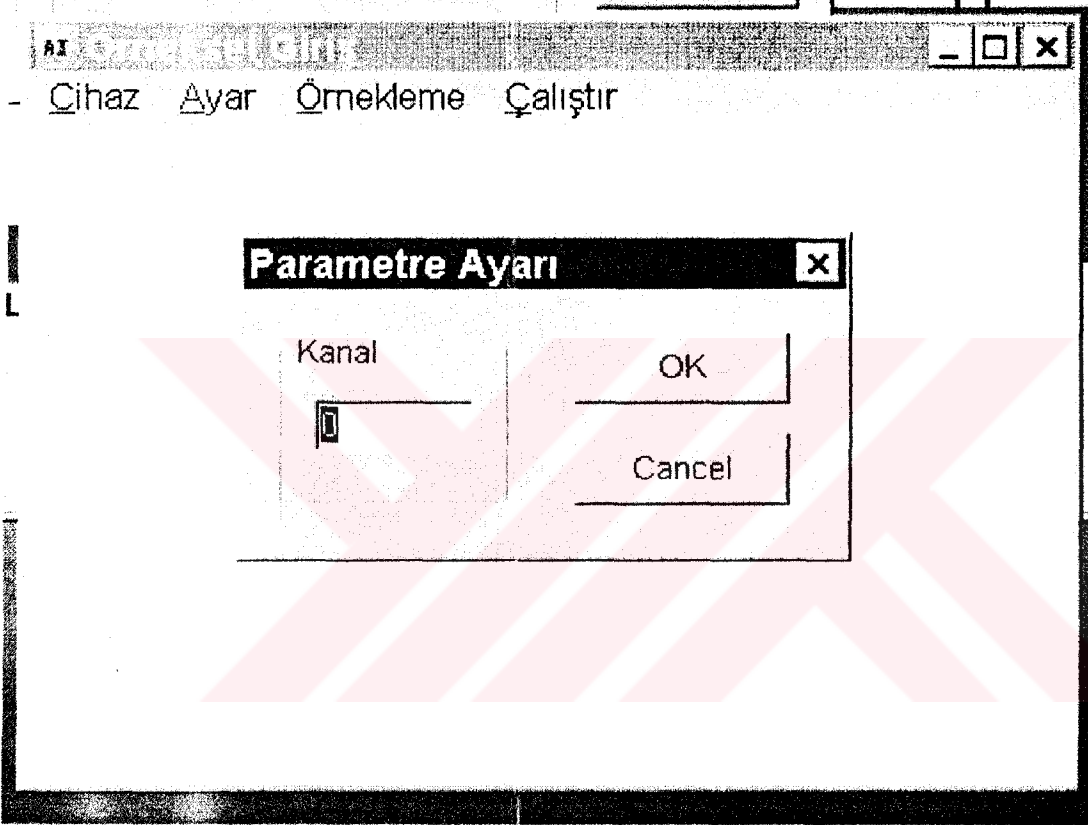
Şekil 7.24. Tarama Zamanı Penceresi.

Sayısal çıkış testi seçildiğinde sayısal girişte olduğu gibi önce cihaz ve modül seçim penceresi ekrana çıkar. Bu pencereden dijital çıkış modülü seçilir. Menüden çalıştırı tıklayınca ekrana ikinci gelen sayısal çıkış penceresinde bitleri temsil eden kutucuklar vardır. Bu kutucuklar seçilerek modüle gönderilecek veri ayarlanır. ByteYaz butonuna tıklayınca bir baytlık veri gönderilir. Gerioku butonuna tıklayınca da çıkıştaki veriyi geri okur. Şekil 7.25.'de gösterilmiştir. Göndermede hata varsa hata mesajı çıkar.



Şekil 7.25. Sayısal Çıkış Test Ekranı.

Örneksel giriş test edilmek istendiğinde çıkan örneksel giriş penceresinde menüden cihaza tıklanarak açılan cihaz seçim penceresinde örneksel giriş modülü seçilir. Menüden ayara tıklayınca Şekil 7.26'da gösterilen parametre ayarı penceresi çıkar. Burada modülün kanalı seçilir ve tamama tıklanır. Tarama seçeneğinde de sayısal giriş menüsünde olduğu gibi bir tarama zamanı penceresi açılır ve bilgisayarın ne kadar sıklıkla modülden veri okuyacağı ayarlanır. Çalıştırı tıklayınca veri okumaya başlanır. Hata durumunda hata mesajı çıkar.



Şekil 7.26. Örneksel Giriş Test Ekranı.

Kontrol panelde çalıştır butonuna tıklanınca ana program çalışır. Ana program çalışmaya başladığında ekrana yeni bir pencere açılmayacak sadece kontrol panelinde resim üzerindeki durum belirtme lambaları çalışma durumlarına göre kırmızı veya yeşil olacaktır. Suyun kaç derecede olduğu oradaki alanda sürekli gösterilecektir. Kullanıcı programdan çıkmak isteyip çıkış butonuna tıkladığında programın çalışması durdurulacak ve programdan çıkılacaktır.



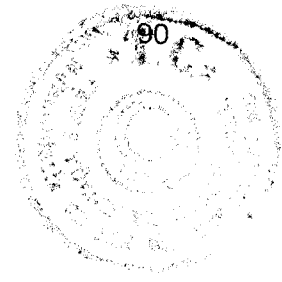
8. SONUÇ

Bu tezde, dağıtık veri toplama ve kontrol sistemlerinde kullanılan ağ teknolojilerinden Kontrolör Alan Ağı (CAN) tüm yönleriyle araştırılıp diğer ağ teknolojileri ile karşılaştırılmıştır.

Bunun sonucunda CAN'in güvenilir, hızlı ve yaygınlaşmakta olan bir teknoloji olduğunu görülmüştür. CAN, henüz ülkemizde pek yaygınlaşmamasına rağmen özellikle Avrupa'da yaygınlaşmaktadır ve 500'den fazla üretici CAN uyumlu malzeme üretmektedir. Bu yüzden CAN'e endüstriyel uygulamalarda geleceğin teknolojisi de denilebilir.

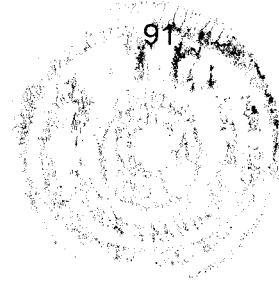
CAN üzerinde dağıtık veri toplama ve kontrol sistemi örneği tasarlanmıştır ve böyle bir sistemin kolayca tasarlanabildiği gösterilmiştir. Bunun için bir kullanıcı arayüzü programı yazılmıştır.

Bu çalışmanın bölümümüzde kurulmakta olan proses kontrol laboratuvarında CAN teknolojilerinin seçiminde önemli bir rehber olacağı düşünülmektedir.



KAYNAKLAR

- 1- Advantech, ADAM 5000/CAN User Manual, Taiwan, 1997.
- 2- Advantech, Help on: ADAM 5000 DeviceNet Driver, 1999.
- 3- Advantech, Total Solution for PC-Based Industrial Automation, Solution Guide IA Vol.81, 1999.
- 4- Aladjem, S., CAN - Networking for Real-Time Systems, <http://www.rad.com/networks/1997/can/can.htm>, 1997.
- 5- Avcı, O., SCADA Sistemleri, Eksen Yayıncılık, İstanbul, 1998.
- 6- Bosch, R., CAN Specifications Version 2.0, BOSCH, Stuttgart, 1991.
- 7- CAN in Automation , <http://www.can-cia.de/>, 1999.
- 8- Coutinho, J., Martin, S., Samata, G., Tapley, S. and Wilkin, D., What is Fieldbus, <http://www.kdtek.com/filedbus-tutor.htm>, 1995.
- 9- Fredriksson, L.B., Controller Area Networks and The Protocol CAN For Machine Control Systems, Kvaser AB, <http://www.kvaser.se>, Sweden, 1996.
- 10- Fredriksson, L.B., Distributed Embedded Control Systems in Robotics, Kvaser AB, <http://www.kvaser.se>, Sweden, 1996.
- 11- Gören, M., "OTOMASYON 99" Seminer Notları, 1999.
- 12- Honeywell, Smart Distributed System (SDS) , <http://www.honeywell.com/sensing/prodinfo/sds/> , 1999.
- 13- Kvaser AB, Kvaser CAN Kingdom Pages (CAN), <http://www.kvaser.se/canking/index.htm>, Sweden, 1999.
- 14- Lennartsson, K. and Fredriksson, L.B., Fundamental Parts in SDS, DeviceNet and CAN Kingdom, Kvaser AB, Sweden, 1996.
- 15- McLaughlin, R.T. and Khoh, S.B., Controller Area Network (CAN) For Industrial Applications, <http://www.warwick.ac.uk/devicenet/indus.htm>.
- 16- Miller Freeman Inc, CAN Solutions Directory 1998, <http://www.directories.mfi.com/embedded/can/> , 1998.
- 17- NewCastle University Robotics and Automation Group (NURAG), The CANOpen Protocol, <http://www.ncl.ac.uk/~nrauto/canproto.htm> , 1997.



- 18- Nilsson, S., Controller Area Network - CAN Information,
<http://www.algonet.se>, 1997.
- 19- Olsson, G., Piani, G., Computer Systems for Automation and Control,
Prentice Hall Inc, New York, 1992.
- 20- Pekiner, F.O., 3e Dergisi Sayı: 53, Bileşim Yayıncılık, İstanbul, Ekim
1998.
- 21- Rigby, W. H. And Dalby, T., Computer Interfacing a Practical Approach to
Data Acquisition and Control, Prentice-Hall Inc., New Jersey, 1-5, 152-
173, 1995.
- 22- Sarıbay, T., Endüstriyel Saha Veri Yolu - CANBus, Otomasyon Dergisi
Sayı: 73, Bileşim Yayıncılık, İstanbul, 54-56, Temmuz 1998.
- 23- Schofield, M. J., Controller Area Network – CAN Application Layers,
<http://www.compulink.co.uk/>, 1998.
- 24- Tang, K.H., Introduction to DeviceNet, University of Warwick,
<http://www.warwick.ac.uk/devicenet/dnet1.htm> , 1999.
- 25- Wierzbicki, J., Programming The 8250 UART, <http://www.perplexed.com>,
<http://www.byterunner.com/16550.html>.
- 26- Yeltekin, A., Otomasyon Dergisi Sayı:71, Bileşim Yayıncılık, İstanbul,
Mayıs 1998.



ÖZGEÇMİŞ

Nilüfer YANIK, 24 Ocak 1976 tarihinde İzmit'te dünyaya geldi. İlk ve orta öğrenimini İzmit'in Tavşancıl beldesinde 1989 yılında tamamladıktan sonra lise öğrenimini Gebze STFA Anadolu Teknik Lisesi Bilgisayar Bölümü'nde sürdürdü. 1993 yılında Marmara Üniversitesi Teknik Eğitim Fakültesi Elektronik ve Bilgisayar Eğitimi Bölümüne girdi ve 1997 yılında Bilgisayar Teknolojileri Eğitimi Anabilimdalından mezun oldu. Aynı yıl Gebze Yüksek Teknoloji Enstitüsü'nün yüksek lisans sınavını kazandı. Halen bu kurumda araştırma görevlisidir.



EK-1

SERİ PORT ÜZERİNDEN HABERLEŞME İÇİN C++BUILDER
PROGRAMISericomm.cpp dosyası:

```
//-----  
#include <vcl.h>  
#pragma hdrstop  
  
#include "sericomm.h"  
//-----  
#pragma package(smart_init)  
  
//-----  
  
void UART_init(void)  
{  
/*Set wordformat and data rate*/  
asm{  
    mov dx,COM1+LCR //8-n-1 and DLAB = 1  
    mov al,0x83  
    out dx,al  
  
    mov dx,COM1+DL_L //9600 bps  
    mov al,0xC  
    out dx,al  
    mov al,10  
    in al,dx  
  
    mov dx,COM1+DL_H  
    mov al,0  
    out dx,al
```

```
    mov dx,COM1+LCR //reset DLAB =0
    mov al,0x3
    out dx,al
    }
}
```

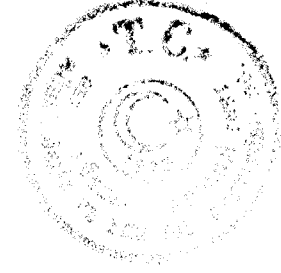
```
//-----
```

```
void UART_putchar(unsigned char ch)
```

```
{
    unsigned char bb = 0;
    short chi;

    /* polling the port until it is empty, then send the char */
    while (((int)bb & THRE) == 0)
    {
        asm{
            mov dx,COM1+LSR
            in  al,dx
            mov bb,al
        }
    }
}
```

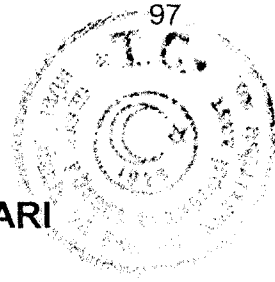
```
if (((int)bb & THRE) == 32) {
    chi=(short) ch;
    asm{
        mov dx,COM1
        mov al,byte ptr [chi]
        out dx,al
    }
}
}
```



```
//-----  
  
unsigned char UART_getchar(void)  
{  
    unsigned char bb=0;  
    unsigned char veri = 0;  
  
    /* polling the port until a char is available and then read it */  
    while (((int)bb & RDR ) == 0){  
        asm{  
            mov dx,COM1+LSR  
            in  al,dx  
            mov bb,al  
        }  
    }  
    if (((int)bb & RDR) ==1)  
    asm{  
        mov dx,COM1  
        in  al,dx  
        mov veri,al  
    }  
    return veri;  
}
```

Sericomm.h dosyası:

```
//-----  
  
#ifndef sericommH  
#define sericommH  
  
#define RBR    0  
#define THR    0  
#define DL_L    0  
#define DL_H    1  
#define LCR    3  
#define MCR    4  
#define LSR    5  
#define MSR    6  
#define RDR    1  
#define THRE    32  
#define CTS    16  
#define DSR    32  
#define RI     64  
#define DCD    128  
#define DTR    1  
#define RTS    2  
#define EOT    4  
#define COM2    0x2F8  
#define COM1    0x3F8  
  
void UART_init(void);  
void UART_putchar(unsigned char ch);  
unsigned char UART_getchar(void);  
  
//-----  
#endif
```



ADAM 5000/CAN DEVİCENET FONKSİYONLARI

Fonksiyon Özeti [Advantech, 1999]

1. Configuration Functions (ALL)

DNOpen: Opens CAN driver

DNClose: Closes CAN driver

DNInit: Initializes DeviceNet driver

DNReset: Resets DeviceNet driver

2. Connection or Disconnection Functions (ALL)

DNAllocateGrpMsg: Allocates Available Message ID

DNFreeGrpMsg: Frees Allocated Message ID

DNUcmmOpenReq: Sends Open Explicit Message Request

DNUcmmCloseReq: Sends Close Explicit Message Request

DNSetAttrib: Modifies an attribute

DNGetAttrib: Retrieves an attribute

DNSendEXPReq: Sends Explicit Message Request

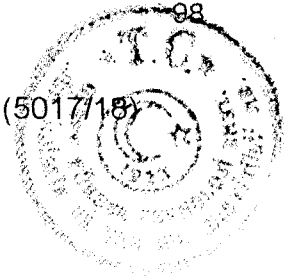
DNGetEXPRsp: Gets Explicit Message Request

3. I/O Functions

DNGetAi: Gets Analog Input from Device (5017/18)

DNGetCJC: Gets CJC Value from Device (5018)

DNSetChanMask: Enable/Disable Analog Input Channel (5017/18)



DNGetChanMask: Gets AI Channel Enable/Disable Mask (5017/18)

DNGetAo: Gets Analog Output status device (5024)

DNSetAO: Sets Analog Output on Devices (5024)

DNGetAoStart: Gets Analog Output Start Value on Device (5024)

DNSetAoStart: Sets Analog Output Start Value on Device (5024)

DNGetDo: Gets Digital Output Status from Device (5056)

DNGetDi: Gets Digital Input Status from Device (5051/52)

DNSetDo: Sets Digital Output to Device (5056)

4. Alarming Functions (5017/18)

DNGetAlarmState: Gets Alarm Status

DNSetHiAlarm: Sets High Alarm Limit

DNSetLoAlarm: Sets Low Alarm Limit

DNSetAlarmEnabled: Enables/Disables Alarm

DNGetHiAlarm: Gets High Alarm Limit

DNGetLoAlarm: Gets Low Alarm Limit

DNGetAlarmEnabled: Gets Enables/Disables Alarm Type

5. Calibration Functions

DNSetSlew: Sets Slew Rate (5017/18)

DNGetSlew: Gets Slew Rate (5017/18)

DNSpanCali: Span Calibration (5017/18)

DNZeroCali: Zero Calibration (5017/18)

DNCJCCali: CJC calibration for ADAM-5018 (5018)

DNAO4mACali: Analog Output calibration for 4mA (5024)

DNAO20mACali: Analog Output calibration for 20mA (5024)

6. Miscellaneous Functions

DNGet5kVersion: Gets ADAM-5000 firmware version (ALL)

DNSetEPR: Set EPR value (ALL)

DNGet5kType: Gets ADAM-5000 Device Type (ALL)

DNSetHz: Sets Analog Input filter (5017/18)

DNGetHz: Gets Analog Input filter (5017/18)

DNSetRange: Sets Range Code for AI or AO (5017/18/24)

DNGetRange: Gets Range Code for AI or AO (5017/18/24)

7. Event Handling Function

DNSetEventEnabled: Sets Event handling function enable

DNGetEventData: Gets Event Data

8. Device Function

DNDeviceGetParamOfReg: Gets configure data installed in registry

DNDeiveGetNumOfList: Returns the number of installed devices

DNDeviceGetList: Gets the number and type of installed device, not including the devices attached to COM port or CAN

DNDeviceGetSubList: Returns a list of installed devices attached to a specified COM port or CAN.