T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Yunus SANTUR

KONTROL ALAN AĞI PROTOKOLÜ TABANLI BİR ENDÜSTRİYEL AĞIN KURULMASI VE YÖNETİLMESİ

Tez Yöneticisi:
Yrd. Doç. Dr. Hayrettin CAN

YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

ELAZIĞ,2006

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

KONTROL ALAN AĞI PROTOKOLÜ TABANLI BİR ENDÜSTRİYEL AĞIN KURULMASI VE YÖNETİLMESİ

Yunus SANTUR

YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

Bu tez, ... tarihinde aşağıda belirtilen jüri tarafından oybirliği /oyçokluğu ile başarılı / başarısız olarak değerlendirilmiştir.

Danışman : Yrd.Doç.Dr. Hayrettin CAN

Üye :

Üye :

Üye:

Üye:

Bu tezin kabulü, Fen Bilimleri Enstitüsü Yönetim Kurulu'nun ...../...../..... tarih ve ..... sayılı kararıyla onaylanmıştır.

TEŞEKKÜR

Bu tez çalışmasını yönlendiren ve çalışma süresince yardımlarını esirgemeyen danışman hocam Sayın Yrd.Doç.Dr. Hayrettin CAN'a şükranlarımı arz ederim. Ayrıca her türlü yardımlarından dolayı Arş.Gör. Erkan DUMAN'a en içten teşekkürlerimi sunarım.

İÇİNDEKİLER

TEŞEKKÜR
İÇİNDEKİLER I
ŞEKİLLER LİSTESİ V
TABLOLAR LİSTESİ VII
ÖZET VIII
ABSTRACT IX

1. GİRİŞ 1
1.1. Sayısal İletişim Kavramları 1
1.2. Seri İletişim 2
1.2.1. Senkron Seri İletişim 3
1.2.2. Asenkron Seri İletişim 3
1.3. Kontrol Ağları 3
1.4. Tezin Amacı 5
1.5. Tezin Yapısı 5
2. KONTROL ALAN AĞI PROTOKOLÜ 6
2.1. Kontrol Alan Ağı Protokolü 6
2.2. Kontrol Alan Ağı Protokolünün Kısa Tarihçesi 6
2.3. Kontrol Alan Ağı Protokolünün Özellikleri 7
2.4. Kontrol Alan Ağı Protokolü Standartları 8
2.4.1. Standart Kontrol Alan Ağı Protokolü (CAN 2.0A) 8
2.4.2. Genişletilmiş Kontrol Alan Ağı Protokolü (CAN 2.0B) 9
2.4.3. Kontrol Alan Ağı Protokolü ve OSI Referans Modeli 9
2.5. Kontrol Alan Ağı Protokolü Mesaj Çerçeve Alanları 10
2.5.1. Çerçeve Başlangıç Biti 10
2.5.2. Mesaj Tanımlayıcı Alan 11
2.5.3. İstek Mesajı Ayırt Edici 11
2.5.4. Ayrılmış Alan 11
2.5.5. Veri Uzunluk Alanı 11
2.5.6. Veri Alanı 11
2.5.7. Mesaj Kontrol Alanı 12
2.5.8. Onay Alanı 12
2.5.9. İstek Türü Ayırt Edici 13

2.5.10. Çerçeve Sonu Alanı 13
2.6. Kontrol Alan Ağı Protokolü Mesaj Türleri 14
2.6.1. Veri Mesajları 14
2.6.2. İstek Mesajları 14
2.6.3. Hata Mesajları 14
2.6.3.1 Kontrol Alan Ağı Protokolü Fiziksel Kodlama Yöntemi 14
2.6.3.2. Aktif Hata Mesajları 15
2.6.3.3. Pasif Hata Mesajları 15
2.6.3.4. Aşırı Yüklenme Mesajları 15
2.6.3.5. Bus-Off Durumu 16
2.6.4. Kontrol Alan Ağı Protokolü Fiziksel Bit Kodlama Yöntemi 17
2.7. Kontrol Alan Ağı Protokolü Fiziksel Özellikler 18
2.7.1. Kontrol Alan Ağı Protokolü Programlanabilir Veri Hızı 18
2.7.2. Kontrol Alan Ağlarında Fiziksel Kablo, Konektör ve Sonlandırma 20
2.8. Kontrol Alan Ağı Protokolü Veri Yoluna Erişim Prensibi 20
2.9. Kontrol Alan Ağı Protokolü Hata Algılama Mekanizması 22
2.9.1. CRC Kontrolü 22
2.9.2. ACK Kontrolü 22
2.9.3. Çerçeve Formatı Kontrolü 22
2.9.4. Veri Yolunun İzlenmesi 22
2.9.5. Kontrol Ağında Hata Algılanması Durumunda Yapılanlar 23
2.10. Kontrol Alan Ağı Protokolü Kontrol Ediciler ve Sınıflandırılması 23
2.11. Kontrol Alan Ağı Protokolü Tabanlı Yüksek Seviyeli Protokoller 25
3. MICROCHIP MCP2510 KAA KONTROL EDİCİ 27
3.1. Genel Bilgi 27
3.2. Entegre Bloklarının Görevleri 28
3.3. Bacak Konfigürasyonu ve Görevleri 29
3.4. MCP2510 Kontrol Edici Çalışma Modları 30
3.4.1. Konfigürasyon modu 31
3.4.2. Normal Çalışma modu 31
3.4.3. Uyuma modu 31
3.4.4. Dinleme modu 32
3.4.5. Test modu 32
3.5. MCP2510 Kontrol Edici Konfigürasyon İşlemleri 32
3.5.1. Kontrol Kaydedici (CANCTRL Register) 33

3.5.2. İletişim Hızı Kaydedicileri (CN1, CNF2, CNF3 Registers) 34
3.5.3. Kesme kaydedicisi (CANINTE Register) 35
3.5.4. Filtre ve Maske Kaydedicileri 36
3.5.5. Mesaj Gönderim Durum ve Kontrol Kaydedici(TXRTSCTRL) 36
3.5.6. Minimum Konfigürasyon Bilgisi 37
3.6. MCP2510 Mesaj Gönderim İşlemi 37
3.6.1. MCP2510 Mesaj Gönderim Tampon Bellek Alanları 38
3.6.1.1. Gönderici Tampon Bellek Kontrol Kaydedici(TXBnCTRL Register) 38
3.6.1.2. Mesaj Önceliği 40
3.6.1.3. Diğer Gönderici Tampon Bellek Kaydedicileri 40
3.6.2. Mesaj Gönderim İşleminin Başlatılması 41
3.7. MCP2510 Mesaj Alım İşlemi 41
3.7.1. MCP2510 Mesaj Alım Tampon Bellek Alanları 42
3.8. MCP2510 Mesaj Kabul Mantığı 42
3.8.1. Filtre ve Maske Kaydedicileri 43
3.8.2. Mesaj Alım Tampon Bellek Kontrol Kaydedicileri 44
3.9. MCP2510 Hata Durumlarının Değerlendirilmesi 46
4. MICROCHIP MCP2551 YOL ALICISI 48
4.1. Genel Bilgi 48
4.2. MCP2551 Yol Alıcı Bacak Konfigürasyonu 48
4.2.1. MCP2551 Çalışma Modu Seçimi 48
4.2.2 Tipik Uygulama Devresi 49
4.3. Fiziksel Veri Yolu Standartları 49
4.3.1. Kontrol Ağı Fiziksel Topoloji 49
4.3.2. Kontrol Ağı Yol Sonlandırma 50
4.3.3. Kontrol Alan Ağı Düğüm Sonlandırma 51
5. KURULAN DENEYSEL GENEL AMAÇLI ENDÜSTRİYEL AĞ 52
5.1. Genel Bilgi 52
5.2. Deney Sisteminin Blok Diyagramı 53
5.2.1. Ağ Üzerindeki İstasyonların Bağlantı Şeması 54
5.3. PIC Mikro Denetleyici ile MCP2510 Kontrol Ediciye Erişim 55
5.3.1. PIC16F877 SPI Protokolü 55
5.3.2. PIC16F877 SPI Kaydedicileri 56
5.4. Spi Haberleşme İşlemleri 57
5.4.1. Spi Aracılığıyla İstenen Kaydediciye Değer Yazma 58

5.4.2. Spi Aracılığıyla İstenen Kaydedici İçeriğini Okuma 57
5.4.3. Spi Aracılığıyla Mesaj Gönderim İşleminin Başlatılması 59
5.4.4. Spi Emir Takımı 59
5.4.5. Spi Okuma ve Yazma Zamanlama Diyagramları 60
6. SONUÇLAR 62
KAYNAKLAR 63
ÖZGEÇMİŞ..... 66
EK-1

ŞEKİLLERİN LİSTESİ

Şekil 2.1. Standart KAA Protokolü Mesaj Çerçevesi 9
Şekil 2.2. Genişletilmiş KAA protokolü mesaj çerçevesi 9
Şekil 2.3. KAA protokolü ve OSI Referans Modeli 10
Şekil 2.4. Aktif Hata Mesajı 15
Şekil 2.5. Pasif Hata Mesajı 15
Şekil 2.6. Aşırı Yüklenme Mesajı 15
Şekil 2.7. Bus-off Durumu 16
Şekil 2.8. NRZ Bit Örnekleme 17
Şekil 2.9. Bit-Stuffing Kodlama 18
Şekil 2.10. KAA Protokolü 1-bit Zamanı 19
Şekil 2.11. Örnek KAA Topolojisi 20
Şekil 2.12. Kontrol Alan Ağı Yola Erişim Önceliği 21
Şekil 2.13. BasicCAN Kontrol Edici 24
Şekil 2.14. FullCAN Kontrol Edici 25
Şekil 3.1. MCP2551 Blok Diyagram 27
Şekil 3.2. MCP2510 Bacak Konfigürasyonu 29
Şekil 3.3. MCP 2510 Konfigürasyon İşlemleri 33
Şekil 3.4. KAA Bit Zamanlaması 34
Şekil 3.5. MCP2510 Mesaj Gönderim İşlemi 37
Şekil 3.6. MCP2510 Mesaj Okuma İşlemi 41
Şekil 3.7. MCP2510 Mesaj Kabul Filtresi 43
Şekil 3.8. Hata Kaydedicilerin Güncellenmesi 46
Şekil 4.1. MCP2551 Yol Alıcısı 48
Şekil 4.2. Örnek Bir Uygulama Devresi 49
Şekil 4.3. Kontrol Ağı Fiziksel Topoloji 50
Şekil 4.4. Kontrol Alan Ağlarında Yol Sonlandırma 51
Şekil 4.5. Kontrol Alan Ağlarında Yol Sonlandırma 51
Şekil 5.1. Örnek Kontrol Alan Ağı 52
Şekil 5.2. Deney Sistemi Blok Diyagramı 53
Şekil 5.3. Deney Sistemi Bağlantı Şeması 54
Şekil 5.4. Deney Seti 54
Şekil 5.5. Örnek Spi Haberleşme 55
Şekil 5.6. Spi Aracılığıyla 1-byte Veri Gönderimi 58

Şekil 5.7. Spi Okuma Zamanlama Diyagramı 60
Şekil 5.8. Spi Yazma Zamanlama Diyagramı 61

TABLOLARIN LİSTESİ

Tablo 2.1. DLC Alanı ile Veri Mesajı Uzunluğu İlişkisi 12
Tablo 2.2. KAA Protokolü Çerçeve Alanları Değerleri 13
Tablo 2.3. KAA Protokolü Örnek Veri Hızları 19
Tablo 3.1. MCP2510 Bacak Görevleri 30
Tablo 3.2. Kontrol Kaydedici 33
Tablo 3.3. İletişim Hızı Kaydedicileri 35
Tablo 3.4. Kesme Kaydedici 36
Tablo 3.5. Mesaj Gönderim Durum ve Kontrol Kaydedici 36
Tablo 3.6. Mesaj Gönderim Tamponu 38
Tablo 3.7. Gönderici Tampon Bellek Kontrol Kaydedici 39
Tablo 3.8. TXBnSIDL Kaydedici 40
Tablo 3.9. TXBnDLC Kaydedici 40
Tablo 3.10. Mesaj Kabul Filtresi 44
Tablo 3.11. Mesaj Alım Tampon Bellek Kontrol Kaydedici 44
Tablo 3.12. Mesaj Alım Tampon Bellek Kontrol Kaydedicileri 45
Tablo 3.13. Mesajım Alım Filtre Seçme İşlemleri 45
Tablo 3.14. Hata Bayrak Kaydedici 47
Tablo 4.1. MCP2551 Bacak Görevleri 48
Tablo 4.2. Kontrol Ağı Fiziksel Mesafeler 50
Tablo 5.1. SSPCON Kaydedici 56
Tablo 5.2. SSPSTAT Kaydedici 57
Tablo 5.3. Mesaj Gönderim İşlemini Başlatma Spi Emir Kodları 59
Tablo 5.4. Spi Emir Takımı 60

ÖZET

Yüksek Lisans Tezi

KONTROL ALAN AĞI PROTOKOLÜ TABANLI BİR ENDÜSTRİYEL AĞIN KURULMASI VE YÖNETİLMESİ

Yunus SANTUR

Fırat Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı
2006, Sayfa: 65

Kontrol alan ağı çok yöneticili, yüksek güvenlik ve çarpışma çözümleyicisi özelliklerine sahip seri iletişime dayalı bir haberleşme protokolüdür. Kontrol alan ağı otomotiv endüstrisinde araç içi iletişimde kullanılmak için geliştirilmiştir. Üstün özelliklerinden dolayı birçok endüstriyel uygulamada sık sık kullanılır. Araçlar, medikal cihazlar, endüstriyel otomasyon sistemleri ve akıllı binalar diğer uygulama alanlarına örnek olarak verilebilir.

Bu çalışmada önce kontrol alan ağı protokolünün kullanıldığı çeşitli endüstriyel uygulamalar incelendi. Sonra bir deney seti tasarlandı ve kontrol programı geliştirildi. Daha sonra deney seti üzerinde çeşitli deneyler gerçekleştirildi.

Anahtar Kelimeler: Kontrol Alan Ağı Protokolü, Seri İletişim, Endüstriyel Ağlar

ABSTRACT

Master Thesis

ESTABLISHING AND MANAGING OF AN INDUSTRIAL NETWORK BASED ON CONTROLLER AREA NETWORK PROTOCOL

Yunus SANTUR

Firat University
Graduate School of Natural and Applied Sciences
Department of Computer Engineering
2006, Page: 65

Controller area network (CAN) is a communication protocol based on serial communication which has multiple administrators, high security and carrier avoidance features. CAN has been developed to use in internal vehicle communication in automotive industry. It is often used many industrial applications due to superior features. Vehicles, medical devices, industrial automation systems and intelligent buildings are given as examples to others application areas.

In this study, first various industry controller area network applications have been examined. Next an experimental setup has been designed and the control program has been developed. Then, various experiments have been preformed on the experimental setup.

Key Words: Controller Area Network, Serial Communication, Industrial Network

1. GİRİŞ

Günümüz dünyasında teknolojinin gelişmesiyle beraber haberleşme sistemlerinin önemi her geçen biraz daha artmaktadır.

Bilgisayar ve telefon haberleşmesi gibi günlük hayatta birçok insanın kullandığı haberleşme sistemlerinin yanı sıra; çeşitli algılayıcı, analog-sayısal dönüştürücü, veri toplama cihazları, programlanabilir kontrol cihazları ve mikro denetleyicilerden oluşan endüstriyel ağlarda sözü edilen birimlerin birbirleriyle haberleşmesini sağlayan endüstriyel kontrol tabanlı haberleşme protokolleri de büyük önem arz etmektedir.

Günümüzde yukarıda sözü edilen kontrol ağlarında en sık kullanılan haberleşme protokollerinden biri başlangıçta otomotiv endüstrisinde kullanılmak için tasarlanan fakat zaman içinde güçlü özellikleri nedeniyle birçok endüstriyel uygulamalarda sıkça kullanılmaya başlanan ve kullanım sahası giderek genişleyen Kontrol Alan Ağı(KAA) protokolüdür [1].

Endüstriyel uygulamalarda kullanılan kontrol ağlarında haberleşme sisteminin veri güvenliği başta olmak üzere, elektromanyetik gürültü gibi dış etkenlerden fazla etkilenmemesi, yüksek hızlı veri iletişim hızını, uzun mesafeleri desteklemesi, sistemin genişlemeye izin verir yapıda olması ve yönetilebilir olması beklenir.

KAA protokolünün haricinde kontrol ağlarında kullanılan diğer protokollere örnek olarak CANOpen, DeviceNET, PROFIBUS, WorldFIP verilebilir [2,3]. Bu protokoller arasındaki en temel fark fiziksel iletim ortamına erişim prensipleri ve protokolün güvenlik özellikleridir [2].

Kontrol ağlarında kullanılan protokoller genelde seri iletişim esasına dayalı protokollerdir, ilerleyen bölümde sayısal iletişim ile ilgili temel kavramlara yer verilmiştir.

1.1. Sayısal İletişim Kavramları

İkili tabanda kodlanmış verinin çeşitli haberleşme sistemleri arasında aktarılması sayısal iletişim olarak tanımlanır [4]. Aşağıda sırayla sayısal iletişim ile ilgili temel kavramlar verilmiştir:

Kodlama: Veri bitlerinin fiziksel iletim ortamında nasıl ifade edileceğini tanımlar, NRZ, RZ, Manchester kodlama yöntemlerine verilebilecek birkaç örnektir.

Protokol: Aynı veya farklı üreticilere ait haberleşme birimler arasında iletişim yapılabilmesi ve ortak işlemlerin yürütülebilmesi için geliştirilen bir takım kurallar kümesine protokol adı verilir.

Paralel İletişim: İki sistem arasında aynı anda n-bitlik verinin her birinin ayrı bir iletim yolu üzerinden iletildiği sayısal iletişim türüdür, yakın mesafelerde tercih edilir.

Seri İletişim: İki sistem arasında aktarılacak verinin tek bir iletim yolu üzerinden sıralı bir şekilde gönderildiği sayısal iletişim türüdür, uzak mesafelerde tercih edilir.

İletişim Hızı: Sayısal iletişim hızı *baud* birimiyle ölçülür. Birim zamanda aktarılan bit sayısı(*bps*) baud hızı olarak bilinir.

İstasyon: Fiziksel iletim yolu üzerinde bulunan veri alan ve/veya gönderen her bir haberleşme birimi istasyon olarak adlandırılır.

Topoloji: Fiziksel iletim yolunun bağlantı şeklini, kullanılan kablo ve sonlandırma prensiplerinin nasıl olacağını belirler.

Standart Belirleyen Kurumlar: Sayısal iletişim ile ilgili geliştirilen donanımsal cihazlar ve bu cihazlar arasındaki haberleşme protokolleri aşağıda bir kısmı maddeler halinde sıralanan kurumlar tarafından özel bir numara verilerek standartlaştırılır.

- ANSI
- IEEE
- ITU
- EIA
- ISO

1.2. Seri İletişim

Verinin tek bir fiziksel iletim yolu üzerinden sıralı bir şekilde aynı anda tek bir bit olarak iletildiği iletişim türüdür[4]. En yaygın olarak bilinen seri iletişim standardı ilk olarak 1960 yılında EIA tarafından benimsenen ve özellikle kişisel bilgisayar sistemlerinde yaygın olarak kullanılan RS-232 ‘dir [4].

RS-232 standardının 20Kbps ile sınırlı olması ve en fazla 15 metre uzunluğundaki iletim mesafesini desteklemesi başlangıçta kişisel bilgisayarların çevresel aygıtlarla iletişim kurmasına yeterli olmakla beraber teknolojinin gelişmesiyle beraber yerini RS-422, RS-423 ve RS-449 standartlarına bırakmıştır.

Zaman içerisinde üretici firmalar farklı ihtiyaçlarla cevap verebilecek daha uzun mesafelerde, daha yüksek iletişim hızlarını destekleyen seri iletişim esaslı standartlar geliştirmişlerdir. Günümüzde Microchip firmasının PIC mikro denetleyicilerinde olduğu birçok mikro denetleyici ailesi spi, sci ve I²C gibi seri iletişim ara yüzleri ile beraber üretilirler [5].

Seri iletişim esas olarak gönderici ve alıcı cihaz arasındaki iletişimin eş zamanlı olmasını gerektiren senkron iletişim ve eş zamanlı iletişime ihtiyaç duymayan asenkron iletişim olarak ikiye ayrılır.

1.2.1. Senkron Seri İletişim

Gönderici ile alıcı arasındaki iletişimde ortak saat hattının kullanıldığı veri bitlerinin ortak saat işaretleriyle beraber modüle edilerek gönderildiği seri iletişim türüdür. PIC mikro denetleyicilerde yaygın olarak kullanılan spi, I²C protokolleri senkron seri iletişim türüne örnektir [5].

1.2.2. Asenkron Seri İletişim

Gönderici ile alıcının ortak saat hattı kullanmasını gerektirmeyen, veri iletimim herhangi bir zamanda başlanıp bitirilebileceği seri iletişim türüdür. PIC mikro denetleyicilerde kullanılan sci protokolü asenkron seri iletişim türüne örnektir [5]. Asenkron seri iletişimde ortak saat işareti kullanılmadığı için veri paketinin başına ve sonuna özel başlama ve bitirme işaretleri konulur bu şekilde alıcı kendisine bir veri geldiğini anlar.

1.3. Kontrol Ağları

Yukarıda sözü edilen seri iletişim standartları, mikro denetleyicili sistemlerde, çeşitli kontrol cihazları, görüntüleme birimleri ve ikincil bellekler ile mikro denetleyiciler arasında ya da iki mikro denetleyicinin kendi arasında iletişim yapmasında yaygın olarak kullanılmaktadır [5].

Hatasız, gerçek zamanlı, yüksek hızlı veri iletişiminin ön planda olduğu, yönetilebilir ve genişleyebilir yapılarda olması gereken endüstriyel uygulamalarda kullanılan kontrol ağlarında bu standartlar gerekli ihtiyaçlara cevap verememektedir [6,7].

Mercedes-Benz firması var olan çarpışma tabanlı seri iletişim esaslı protokollerin kendi araçlarında kullanılmak için yetersiz olduğu düşüncesiyle yeni bir protokol arayışları içindeydi, ilk kez 1986 yılında uluslararası bir konferansta Robert Bosch [1] tarafından KAA protokolünün tanıtılmasının ardından 1992 yılında KAA protokolünü kullanan ilk araçları satışa çıkarmıştır.

Başlangıçta otomotiv endüstrisi için tasarlanan ve bu alanda ilk kez Mercedes firmasının ürettiği araçlarda kullanılan KAA protokolü takip eden yıllarda diğer otomotiv firmaları tarafından da tercih edilmiştir [8].

KAA protokolü araçlarda ABS, vites sistemi, sürücü yol bilgisayarı, sıcaklık ayarı, elektrikli camlar, park algılayıcı, merkezi kilit sistemi vb… birçok elektronik aksamlarının birbirleriyle haberleşmesinde kullanılmaktadır [8,9,10].

Bunun yanı sıra Nick Papadoglou ve Elias Stipidis [9,10] KAA protokolü kullanan araçların durum ve pozisyonlarını bilgilerini GSM üzerinden SMS aracılığı ile merkezi bir kontrol ediciye rapor etmesi üzerine bir çalışma yapmıştır.

Benzer bir çalışma Valentinos K. Kongezos [11] tarafından KAA protokolü kullanan mobil araçların birbirleriyle kablosuz haberleşme aracılığıyla haberleşebildiği bir çalışma gerçekleştirmiştir.

Zaman içerisinde kullanım alanı gittikçe artan KAA protokolüyle ilgili bir çalışma Hong-Hee Lee ve Ui-Horn Jeong [12] tarafından KAA protokolü tabanlı bir kontrol ağında çoklu motorların hız eş zamanlaması üzerine bir çalışma yapmıştır.

KAA protokolü çeşitli güvenlik, alarm ve kontrol sistemlerinden oluşan akıllı bina uygulamalarında da kullanılmıştır. Bu alanda Teoh Chee Hooi ve diğ. [13], KAA protokolü tabanlı düşük maliyetli akıllı binalar üzerine bir çalışma yapmıştır.

Yine benzer bir çalışma Kyung Chang Lee ve Hong-Hee [14] tarafından KAA protokolünün kullanılacağı akıllı binalarda yangın alarm sistemlerinin kablosuz haberleşme ortamı sayesinde uzaktan yönetimli hale getirilmesi olmuştur.

Jose C.Metrolho ve diğ. [15] KAA protokolü tabanlı bir seracılık uygulamasını kablosuz haberleşme ile uzaktan yönetilebilir olması ile ilgili bir çalışma yapmıştır.

Robotlarda çevresel ünitelerin birbirleriyle haberleşmesinde ya da mobil robotun merkezi bir kontrol ediciyle haberleşmesinde KAA protokolü kullanılmıştır. Bununla ilgili A.Valera ve Diğ. [16] bir endüstriyel robotun KAA protokolü tabanlı bir kontrol ağıyla kontrol edilmesi üzerine bir çalışma yapmıştır.

M. Wargui ve A. Rachid [17] KAA protokolünün kullanıldığı mobil robotlarla ilgili bir uygulama gerçekleştirmiştir.

Çeşitli firmalar tarafından medikal cihazlardan asansörlere, gemi alarm sistemlerinden uçuş simülatörlerine kadar benzer birçok alanda KAA protokolü kullanan cihazlar üretmişlerdir. KAA protokolünün dahada geliştirilmesi ve Ethernet ağlarda olduğu gibi kullanılabilmesi ve farklı KAA segmentlerinin birleştirilmesine çalışmalar gerçekleştirilmiştir [3,18,19,2].

Bu bağlamda KAA protokolünün performansının geliştirilmesine yönelik bir çalışma Gianluca Cena ve Adriano Valenzano [3] tarafından yapılmıştır. H.Ekiz ve diğ. [18,19] iki farklı

KAA segmentinin yine KAA protokolü tabanlı köprülerle birleştirilmesine ve bu ağların performans analizine ilişkin çalışmalar gerçekleştirmişlerdir.

Bir kontrol ağında oluşabilecek önemli durumların tespiti ve bu bilgilerin merkezi bir bilgisayarda depolanması yine gerekli durumlarda yazılımsal değişikliklerin kolayca yapılabilmesi, kontrol ağının yönetilebilir olmasını ifade eder. Bir kontrol ağının yerel ağ ve/veya internet üzerinden yönetilebilir hale getirilmesi özellikle akıllı bina, mobil robotların takibi ve çeşitli güvenlik sistemlerinde büyük önem arz etmektedir. Bu alanda Dieter Bühler ve Wolfgang Küchlin [20] Java ve XML kullanarak uzaktan yönetimli bir kontrol ağı uygulaması üzerine bir çalışma yapmıştır.

Benzer olarak yine Dieter Bühler [21] tarafından uzaktan yönetimli kontrol ağları için yüksek seviyeli bir işaretleme dili geliştirilmesi üzerine bir çalışma yapmıştır.

1.4. Tezin Amacı

KAA protokolü otomotiv sektörü başta olmak üzere birçok endüstriyel uygulamada çok geniş bir yelpazede kullanım alanına sahiptir. Bununla birlikte KAA protokolünün kullanıldığı farklı endüstriyel uygulamaların birçoğunda benzer yapılarda kontrol ağları kullanılmaktadır [2,7,12:19]. Söz konusu kontrol ağlarının üzerinde uygun donanım ve yazılım değişiklikleri yapılarak farklı endüstriyel uygulamalarda da kullanılabilmesi mümkündür.

Bu tez çalışmasında birbirleri ile KAA protokolü tabanlı haberleşen istasyonlar kullanılarak genel amaçlı deneysel bir kontrol ağı tasarlanmıştır. Deney seti olarak tasarlanan kontrol ağı üzerinde, hem KAA protokolü haberleşme işlevlerinin kavranması hem de yukarıda sözü edildiği gibi uygun donanım ve yazılım ilaveleri yapılarak farklı uygulama alanlarında kullanılabilmesi amaçlanmıştır.

1.5. Tezin Yapısı

Bu tez altı bölümden oluşmaktadır. Bölüm 2'de KAA protokolü ayrıntılı olarak ele alınmış, gelişimi, özellikleri ve protokolün çalışma prensipleri anlatılmıştır.

Bölüm 3'de Microchip KAA kontrol edicisinin yapısı, çalışması ve programlanması anlatılmıştır.

Bölüm 4'de Microchip KAA yol alıcısının yapısı, çalışması ve KAA fiziksel veri yolu özellikleri verilmiştir.

Bölüm 5'de tez kapsamında tasarlanan deney seti hakkında bilgi verilmiştir. Deney seti üzerindeki istasyonlar için yazılan program kodları ise ek-1'de verilmiştir.

Bölüm 6'da çalışma ile ilgili genel sonuçlar ve öneriler sunulmuştur.

2. KONTROL ALAN AĞI PROTOKOLÜ

2.1. Kontrol Alan Ağı Protokolü

KAA protokolü Robert Bosch [1] tarafından başlangıçta otomotiv endüstrisine yönelik olarak tasarlanmış çarpışmaların öncelik bilgisiyle önlenebildiği seri iletişime dayalı, yüksek güvenlik özelliklerine sahip, mesaj yayılma(broadcasting) tabanlı çok yöneticili bir endüstriyel iletişim protokolüdür.

2.2. Kontrol Alan Ağı Protokolünün Kısa Tarihçesi

Günümüzde başta Mercedes olmak üzere BMW, Jaguar, Opel, Renault gibi birçok otomobil markalarının elektronik aksamlarının haberleşmesinde ve yol bilgisayarlarında Kontrol alan ağı protokolü kullanılmaktadır. Araçlarda kullanılan otomatik fren sistemi(ABS), araç savrulma sistemi(ASR), anti patinaj sistemi ve akıllı park etme gibi birçok özelliklerin tasarımında kontrol alan ağı protokolü kullanılmıştır.

Bilindiği gibi Mercedes marka arabaların birçok parçası BOSCH fabrikalarında üretilmektedir Mercedes firması kendi arabalarında kullanılmak üzere daha güvenli bir haberleşme protokolü arayışı içerisindeydi klasik çarpışma tabanlı protokoller arabalarda kullanılmak için yeterince güvenli değildi.

Bu gelişmeler doğrultusunda ilk kez 1986 yılında Robert BOSCH tarafından uluslararası bir konferansta tanıtılan KAA protokolü seri iletişime dayalı, çarpışmaların öncelik bilgisiyle yok edildiği ve etkili hata tespiti yapılabilen bir haberleşme protokolü olarak doğdu. Kontrol alan ağı protokolünün 1986 yılında duyurulmasından sonra yine BOSCH firması tarafından otomotiv ve diğer endüstriyel alanlarda kullanılabilmesi için çeşitli çalışmalar yapıldı 1990 yılına gelindiğinde kontrol alan ağı protokolü bugünkü şeklini aldı ve yine bu yıllardan itibaren akademik çevreler tarafından da özel bir ağ protokolü olarak kabul edilmeye başlandı.

KAA protokolü 1992 yılından beri Mercedes marka araçların elektronik aksamlarının haberleşmelerinde kullanılmaktadır, ilk kez Mercedes marka araçlarda kullanımından çok kısa bir süre sonra BMW firması da araçlarında KAA protokolünü kullanmaya başlamıştır, günümüzde birçok otomobil üreticisi araçlarında KAA protokolünü kullanılmaktadır [8].

KAA protokolü ilk yıllarda ISO/OSI referans modelinde 2.katman protokolü olarak kabul gördü [7]. 1990'lı yıllardan itibaren ise ISO tarafından özel bir ağ protokolü olarak standartlaştırıldı(ISO 11898 ve ISO 11519).

1997 yılında Intel firması ilk KAA tümleşik devresini üretti, günümüzde ise KAA tabanlı tümleşik devre üreten 20 civarında firma bulunmaktadır [8].

1999 yılına kadar çeşitli otomotiv ve endüstriyel uygulamalarda 60 milyon civarında KAA tabanlı elektronik aygıt kullanıldı. 2000 yıllarına gelindiğinde otomotiv ve çeşitli endüstriyel alanlarda 100 milyon KAA aygıtı kullanılmıştır [8].

2.3. Kontrol Alan Ağı Protokolünün Özellikleri

Kontrol alan ağı protokolünün genel özelliklerini aşağıdaki gibi maddeler halinde sıralayabiliriz [1,6,7] ilgili konular ilerleyen bölümlerde yeri geldikçe ayrıntılarıyla açıklanacaktır.

Fiziksel Standartlar: KAA protokolü her iki ucu 120ohm'luk dirençle sonlandırılmış çift burgulu kablolarla gerçekleştirilen doğrusal yol topolojisi kullanır, iki uç düğüm arasında en fazla 1km mesafeyi destekler.

Mesaj yayılma prensibi(broadcasting): Kontrol ağında herhangi bir istasyonun gönderdiği veri, hata ya da istek mesajları diğer bütün istasyonlar tarafından alınır.

Mesaj Tanımlayıcı(Identifier): KAA protokolünde çerçeveler gönderici veya alıcı adresi içermezler bunun yerine standart KAA protokolünde 11, genişletilmiş KAA protokolünde ise 29 bitlik bir tanımlayıcı kullanırlar, her istasyon bu tanımlayıcı alana bakarak mesajın kendisine gelip gelmediğini anlar.

Çarpışma Çözümleyicisi: KAA protokolü yeterince güvenli olmadığı için Ethernet protokolünde kullanılan csma-cd tekniği yerine csma-ca tekniğini kullanır, bu yöntemde her istasyona farklı bir öncelik değeri verilerek farklı istasyonların aynı anda yola erişimini önlenir böylece çarpışma olmaması garanti edilir.

Mesaj Önceliği: KAA protokolünde tanımlayıcı alan aynı zamanda öncelik bilgisi olarak kullanılır, sayısal olarak düşük değerlikli tanımlayıcı sistemde en yüksek önceliğe sahiptir, böylece bir kontrol ağında istasyon sayısı kadar öncelik belirlenmiş olur ve çarpışma olmaması garanti altına alınmış olur.

Ayarlanabilir Veri Hızı: KAA protokolü ISO standartlarında belirtilen özelliklere göre en fazla 1MBit/s oranında iletişim yapabilmektedir, KAA aygıtları sayesinde iletişim hızı 5KBit/s ile 1MBit/s arasında programlanabilmektedir.

Yüksek Güvenlik: KAA protokolünde 4 farklı hata tespit mekanizması eş zamanlı olarak çalışmaktadır bu KAA protokolünü diğer haberleşme protokollerinden daha güvenli yapan en önemli özelliktir.

2.4. Kontrol Alan Ağı Protokolü Standartları

KAA protokolünün başlarda otomotiv endüstrisinde kullanımı için tasarlandığını biliyoruz. Araçların elektronik aksamlarının haberleşmesinde ve diğer endüstriyel uygulamalarda kullanım için 11-bitlik tanımlayıcı yeterli olmaktadır, 11-bitlik tanımlayıcı alanla 2048 farklı istasyon adreslenebilmektedir, bu birçok uygulama için yeterli olmasına rağmen tır, kamyon ve gemi otomasyonları gibi endüstrinin bu küçük alt kümesinde kullanım için yeterli olmamaktadır. KAA protokolünün tır, kamyon ve gemi uygulamalarında da kullanılabilmesi ve geleceğe yönelik bir açık kapı bırakılması amacıyla 29-bitlik tanımlayıcıda kullanılabilmektedir. Fakat KAA protokolünün sırf bu uygulamalarda da kullanılabilmesi amacıyla standart olarak 29-bitlik tanımlayıcıya sahip olması hem çerçeve boyunu arttıracağından hem de iletişim hızını düşüreceğinden iki farklı KAA protokolü standardı kullanılmıştır. ISO tarafından da iki farklı numarayla standartlaştırılmıştır.

Standart KAA daha yaygın olarak kullanılmakla beraber Genişletilmiş KAA sadece kamyon, tır ve gemi otomasyonlarında ve KAA protokolü tabanlı yüksek seviyeli protokollerin geliştirilmesinde kullanılmıştır.

2.4.1. Standart Kontrol Alan Ağı Protokolü(CAN 2.0A)

ISO 11898 numarasıyla standartlaştırılmıştır. Standart KAA protokolünde 11-bitlik tanımlayıcı alan ile teorik olarak 2032 farklı istasyon adreslenebilmektedir(16 adres ayrılmıştır). Fakat gerçek uygulamalarda bu sayı kontrol ağının donanımsal yapısı ve kullanılan yol alıcılarının özelliklerine bağlı olarak belirlenmektedir. Örneğin Microchip MCP2551 yol alıcıları kontrol ağı üzerinde en fazla 110 adet istasyon olmasına izin verir.

Standart KAA protokolü başta otomotiv olmak üzere medikal cihazlar, asansörler, uçuş simülatörleri gibi birçok endüstriyel uygulamada kullanılmıştır. Şekil 2.1'de Standart KAA protokolü mesaj çerçeve yapısı görülmektedir.

Standart KAA Protokolü Mesaj Çerçevesi

SOF	identifier	RTR	r0 r1	DLC	DATA	CRC	CRC DEL.	ACK	ACK DEL.	EOF
0	11-bit	0/1	00	4-bit	0-8 byte	15-bit	1	0/1	1	7-bit

Şekil 2.1. Standart KAA protokolü mesaj çerçevesi

2.4.2. Genişletilmiş Kontrol Alan Ağı Protokolü(CAN 2.0B)

ISO 115 numarasıyla standartlaştırılmıştır. 29(11+18) bitlik tanımlayıcı alan kullanır [1]. Standart KAA protokolü gibi yaygın bir kullanım alanı yoktur endüstride kamyon ve tır gibi araçların elektronik aksamlarının haberleşmesinde ve KAA protokolü tabanlı yüksek seviyeli protokollerin geliştirilmesinde kullanılmıştır.

Şekil 2.2'de Genişletilmiş KAA protokolü mesaj çerçeve yapısı görülmektedir.

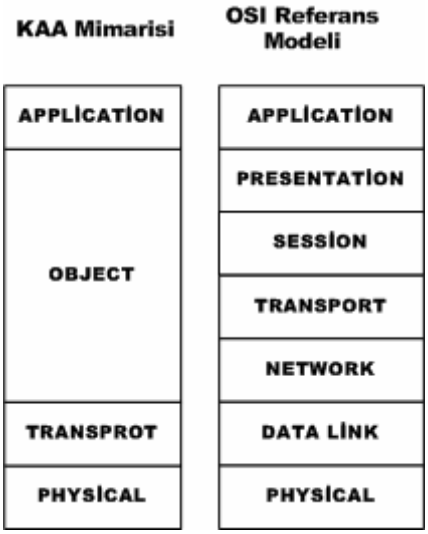
Genişletilmiş KAA Protokolü Mesaj Çerçevesi

SOF	identifier	SRR	identifier	RTR	r0 r1	DLC	DATA	CRC	CRC DEL.	ACK	ACK DEL.	EOF
0	11-bit	1	18-bit	0/1	00	4-bit	0-8 byte	15-bit	1	0/1	1	7-bit

Şekil 2.2. Genişletilmiş KAA protokolü mesaj çerçevesi

2.4.3 Kontrol Alan Ağı Protokolü ve OSI Referans Modeli

Kontrol Alan Ağı protokolü, uygulama hassasiyeti açısından yapısal olarak katmanlar halinde tanımlanmıştır. Genel olarak KAA protokolü OSI referans modelinde 2.katman protokolü olarak kabul edilir [7]. Şekil 2.3'te KAA mimarisi ile OSI referans modeli gösterilmektedir.



Şekil 2.3. KAA protokolü ve OSI referans modeli

KAA mimarisini oluşturan katmanların işlevleri sırayla şu şekildedir. Uygulama(Application) katmanı OSI referans modelinde uygulama katmanıyla eşdeğer sayılan kullanıcıyla etkileşimde bulunulan katmandır. Hedef(Object) katmanı OSI referans modelinin ara katmanlarına karşılık gelir. Hedef katmanı verinin KAA mesaj çerçevesi haline getirilmesi ve mesaj kabul filtresi gibi KAA protokolüne özgü işlemleri içerir. İletim(Transport) katmanı OSI referans modelinde 2.katmana karşılık gelir hata algılama mekanizması, hata mesajlarının üretimi, mesaj doğrulama, onay(ACK) kontrolü, öncelik belirleme, transfer hızı ve bit zamanlaması gibi fonksiyonları içerir. Fiziksel(Physical) katman ise OSI referans modelindeki 1.katman ile aynı fonksiyonları içerir veri mesajının elektriksel sinyallere dönüştürülmesi, kablo, konnektör ve fiziksel topoloji gibi fonksiyonları içerir.

2.5. Kontrol Alan Ağı Protokolü Mesaj Çerçeve Alanları

Robert Bosch [1] tarafından tanımlanan KAA protokolünün standart ve genişletilmiş her iki sürümüne ait mesaj çerçeve alanları ilerleyen bölümde sırayla anlatılmıştır.

2.5.1. Çerçeve Başlangıç Biti(Start Of Frame)

KAA protokolünde herhangi bir istasyon mesaj göndermeden önce veri yolu'nu dinler. Veri yolunun lojik-1 seviyesinde olması boş olduğunu gösterir bu durumda mesaj göndermek isteyen istasyon lojik-0 değerli çerçeve başlangıç(SOF) bitini göndererek iletime başlar. Ağ üzerinde taşıyıcı olmaması durumunda durağan olarak lojik-1 seviyesinde olmasının sebebi veri

iletim yolunu gürültü gibi dış etkenlerden korunmaktır. Ağ üzerindeki bir istasyon yedi adet lojik-1 değerli çerçeve sonlandırma alanını(EOF) gönderdikten sonra mesaj iletim işlemini bitirir böylece EOF alanını alan istasyonlar yolun artık boş olduğunu anlarlar.

2.5.2. Mesaj Tanımlayıcı Alan(Identifier)

KAA protokolünde veri mesajları gönderici ya da alıcı adresi içermezler, bunun yerine mesaj içeriğini tanımlayan bir tanımlayıcı alan kullanırlar alıcı istasyon bu tanımlayıcı alana bakarak mesajın kendisine gelip gelmediğini anlar. Genişletilmiş KAA protokolünde hem veri alanı uzunluğu(DLC) hem de veri içeriğine göre mesaj kabul edilebilir, bu yaklaşım hem daha esnek bir çözüm sunar hem de mesaj paketi boyutunu kısaltır. KAA protokolünde mesajın nereden geldiği bilgisi yani gönderici adresi ise önem taşımaz.

2.5.3. İstek Mesajı Ayırt Edici(Remote Transmission Request)

KAA protokolünde herhangi bir istasyon başka bir istasyondan veri isteyeceği zaman bir istek mesajı yayınlar, istek mesajları ile veri mesajları aynı çerçeve yapısını kullanır fakat istek mesajlarında veri alanı bulunmaz, RTR bitinin lojik-1 olması bu mesajın istek mesajı olduğunu belirtir normal değeri lojik-0'dır ve veri mesajı olduğunu gösterir.

2.5.4. Ayrılmış Alan(r0 ve r1 bitleri)

Kontrol alan ağı protokolünde bu iki bit kullanılmaz gelecekteki gelişmeler için ayrılmıştır. Bu iki bit veri yada istek mesajlarında her zaman lojik-0 değerindedir.

2.5.5. Veri Uzunluk Alanı(Data Length Code)

r0 ve r1 bitlerini takip eden 4-bitlik bu alan çerçeve içerisindeki verinin uzunluğunu belirtmek için kullanılır.

2.5.6. Veri Alanı(Data Field)

Kontrol Alan Ağı protokolünde en fazla 8-Byte uzunluğunda veri gönderilip alınabilmektedir. Veri alanı 0 ile 8-Byte arasında bir değer olabilmektedir. Tablo 2.1'de DLC alanının veri uzunluğunu nasıl ifade ettiği gösterilmektedir.

Tablo 2.1. DLC alanı ile veri mesajı uzunluğu ilişkisi

DLC3	DLC2	DLC1	DLC0	Veri alanı
0	0	0	0	0 byte
0	0	0	1	1 byte
0	0	1	0	2 byte
0	0	1	1	3 byte
0	1	0	0	4 byte
0	1	0	1	5 byte
0	1	1	0	6 byte
0	1	1	1	7 byte
1	0	0	0	8 byte

2.5.7. Mesaj Kontrol Alanı (CRC)

Kontrol Alan Ağı protokolünün hata tespit etmek için kullandığı 4 farklı yöntemden biride sınama kontrol alanıdır, mesaj içerisinde ki herhangi bir alanda meydana gelebilecek bit değişimleri bu alanın kontrol edilmesi sayesinde tespit edilebilir.

CRC var olan hata tespit yöntemleri içerisinde en etkili yöntemlerden biridir[1,7]. Gönderici çerçeve alanını matematiksel bir polinom olarak kodlar, elde edilen bu polinom özel olarak seçilen başka bir üreteç polinoma bölünerek kalan bulunur, kalan polinom üreteç polinoma eklenerek tekrar 1 ve 0'lar halinde kodlanarak mesaj çerçevesindeki CRC alanına yazılır. Alıcı istasyon kendisine gelen mesajdaki CRC alanını tekrar matematiksel bir polinoma dönüştürüp üreteç polinoma böldüğünde kalan olarak 0 değerini elde etmelidir. Bu durumda mesajın hatasız olarak geldiği kabul edilir; aksi halde gelen mesajın hatalı bir mesaj olduğuna karar verilir ve dikkate alınmaz ardından bir hata mesajı yayınlanarak ağdaki bütün istasyonlara bildirilir. KAA protokolünde CRC için 16-bitlik bir alan kullanır, bunun 0..15 numaralı bitleri CRC için 16.bit ise CRC alanını sonlandırmak için kullanılır ve lojik-1 seviyesindedir.

2.5.8. Onay Alanı (ACK)

Kontrol Alan Ağı protokolünde hata tespit etmek için kullanılan bir diğer yöntem gönderilen her mesaj için bir onay istenmesi tekniğidir. Gönderilen bir mesaja karşılık onay gelmemesi durumunda yine bir hata mesajı yayınlanır. Hangi haberleşme protokolü olursa olsun mesajın hedefine ulaşıp ulaşmadığı bilgisi çok önemlidir ve haberleşme protokollerinde en etkili hata tespit mekanizması olarak kullanılır. KAA protokolünde onay alanı 2-bit ten oluşur, ilk bit

ACK alanı ikinci bit ACK sonlandırıcısıdır. Gönderici istasyon mesaj gönderirken her iki biti lojik-1 seviyesinde gönderir alıcı mesajı başarılı olarak aldığını belirmek için ACK alanına lojik-0 yazar, ACK sonlandırıcısı değişmez her zaman lojik-1 seviyesindedir.

2.5.9. Mesaj Türü Ayırt Edici(SRR biti)

Yalnızca Genişletilmiş KAA çerçevesinde bulunan tek bitlik bir alandır. O anda gönderilmekte olan çerçevenin standart bir çerçevemi yoksa genişletilmiş bir çerçevemi olduğunu belirtir. Genişletilmiş KAA çerçevesinde SOF ve 11-bitlik standart tanımlayıcının gönderilmesinden sonra SRR-biti yer alır ve devam eden mesajın bir genişletilmiş KAA çerçevesi olduğunu belirtir, ardından da 18-bitlik genişletilmiş tanımlayıcı gelir. SRR biti aynı zamanda Standart KAA çerçevesi ile Genişletilmiş KAA çerçevesi arasında öncelik belirlenmesi için kullanılır. Bir ağ da aynı anda yola çıkmak isteyen Standart ve Genişletilmiş çerçeve kullanan 2 farklı istasyon olması durumunda öncelik Standart KAA çerçevesi kullanan istasyonundur.

2.5.10. Çerçeve Sonu Alanı(EOF)

KAA protokolünde gönderici istasyon 7 adet lojik-1 değerlikli bit göndererek kendi mesajını sonlandırır. Ağ üzerindeki tüm istasyonlar 7 adet lojik-1 değerli bit almakla aynı zamanda veri yolunun iletim yapmak için boş olduğu bilgisini elde ederler. KAA protokolünde tüm veri ve istek mesajları bu 7-bitlik lojik-1 değerli alanın gönderilmesiyle sonlandırılır. Tablo 2.2'de KAA protokolü çerçeve alanları ve alabilecekleri değerler verilmektedir.

Tablo 2.2. KAA protokolü çerçeve alanları değerleri

Çerçeve Alanı	Değer/uzunlık	Açıklama
SOF	0	Her zaman lojik-0 değeri alır
SRR	1	Sadece genişletilmiş protokolde kullanılır
RTR	0/1	İstek mesajlarında lojik-1, veri mesajlarında lojik-0 alır
R0,r1	00	Rezerve edilmiş alandır her zaman lojik-0 alırlar
DLC	0000…1000	Ayrıntıları Tablo.1'de gösterildi
CRC	15-bit	Gönderici tarafından hesaplanan 15-bit kontrol alanı
CRC Delimiter	1	CRC alanını sonlandırmak amacıyla kullanılır
ACK	0/1	Gönderici lojik-1 olarak gönderir, alıcı lojik-0 yazar
ACK Delimiter	1	Her zaman lojik-1 değerindedir
EOF	1111111	Her zaman 7 adet lojik-1 alır

2.6. Kontrol Alan Ağı Protokolü Mesaj Türleri

KAA protokolünde veri, istek, hata ve aşırı yüklenme mesajları olmak üzere 4 tür mesaj gönderilebilir [1]. Hata mesajları ise ikiye ayrılır bunlar aktif hata mesajları ve pasif hata mesajlarıdır.

2.6.1. Veri Mesajları

KAA protokolünde veri mesajları 0 ile 8-byte arasında olabilir. Şekil-2.1'de standart veri mesaj çerçevesi verilmişti önceki bölümlerde veri mesajları ve çerçeve alanları anlatılmıştı o yüzden burada tekrar değinilmeyecektir.

2.6.2. İstek Mesajları

KAA protokolünde mesaj tanımlayıcı alanı izleyen RTR-biti veri mesajları ile istek mesajlarını ayırt eder. Kontrol ağındaki bir istasyon başka bir istasyondan veri istediği zaman bir istek mesajı yayınlar. İstek mesajlarında veri alanı olmaz bunun dışında çerçeve olarak veri mesajıyla arasında bir fark yoktur. KAA protokolü OSI referans modelinde 2.katmanda tanımlanan bir seri iletişim protokolüdür ağ üzerindeki her bir kontrol edicinin bir görevi vardır kontrol ediciler arasındaki istek ya da veri mesajlarının ne zaman neye göre istenip gönderileceği ise sistem tasarımına bağlıdır.

Örneğin KAA protokolü kullanan Opel-Vectra arabalarda sürücü hızlandıkça far ışığı yükselmekte yavaşladıkça alçalmaktadır, dolayısıyla gaz ve far lamba seviyesi için kullanılan kontrol ediciler arasında veri alış verişi olmaktadır.

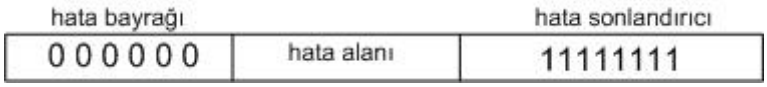
2.6.3. Hata Mesajları

KAA protokolünde iki tür hata mesajı yayınlanabilir aktif hata mesajı ve pasif hata mesajı. KAA protokolü kontrol edicilerde hata durumları için 2 kaydedici kullanılır, gönderici hata kaydedici ve alıcı hata kaydedici oluşan hata durumlarına göre bu kaydedici içerikleri güncellenir ve ilgili hata mesajı gönderilir.

KAA protokolünde veri ya da İstek mesajı iletimi esnasında hata olduğu anlaşılınca mesaj tamamlanmadan hata mesajı yayınlanarak bütün istasyonlara bildirilir ve hatalı mesaj tekrar gönderilir. Hata mesajları üç kısım içerirler, bu kısımlar sırayla hata bayrağı, hata alanı ve hata mesajı sonlandırma alanıdır.

2.6.3.1. Aktif Hata Mesajları

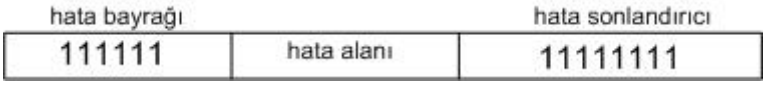
KAA kontrol edicinin aktif hata kaydedici ve pasif hata kaydedici içerikleri sayısal değerleri 128'den küçükse aktif hata bayrağı yayınlanır. Aktif hata mesajları 6-bitlik hata bayrak alanı ile başlar, hata bayrağını 6-bitlik hata alanı izler ve hata mesajı 8-bitlik hata sonlandırma alanı biter. Şekil 2.4'te aktif hata mesaj çerçevesi verilmiştir.



Şekil 2.4. Aktif hata mesajı

2.6.3.2. Pasif Hata Mesajları

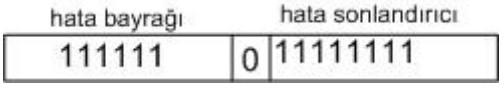
KAA kontrol edicinin aktif hata kaydedici ve pasif hata kaydedici içerikleri sayısal değerleri 127'den büyükse pasif hata bayrağı yayınlanır. Pasif hata mesajları 6-bitlik hata bayrak alanı ile başlar, hata bayrağını 6-bitlik hata alanı izler ve hata mesajı 8-bitlik hata sonlandırma alanı biter. Şekil 2.5'te pasif hata mesaj çerçevesi verilmiştir.



Şekil 2.5. Pasif hata mesajı

2.6.3.3. Aşırı Yüklenme Mesajları

Ağdaki istasyon mesaj işlerken başka bir mesaj gelmesi durumunda aktif hata bayrağına benzer bir aşırı yüklenme mesajı yayınlanır. Bir istasyonun yeniden iletime geçebilmesi için 23-bit periyodu süre geçmesi gerekmektedir. Aşırı yüklenme mesajları pasif hata mesajlarına çok benzemektedir fakat aşırı yüklenme mesajları hata alanı değeri '0' olan tek bitlik bir alandan oluşur.

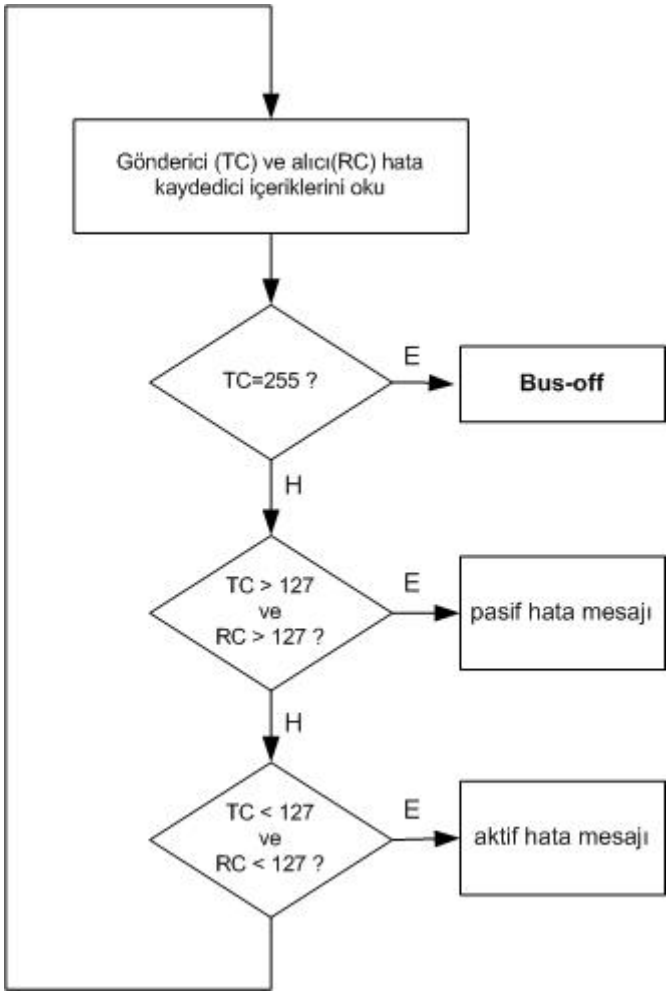


Şekil 2.6. Aşırı yüklenme mesajı

2.6.3.4. Bus-Off Durumu

KAA kontrol edicinin gönderici hata kaydedici içeriği 255 değerine ulaşmışsa o istasyon mesaj gönderim ya da alımı yapamaz ve yeniden başlatılması gerekir. KAA protokolünde meydana gelebilecek en kritik hata durumudur meydana gelme olasılığı küçük olsa bile güvenlik nedeniyle göz ardı edilemeyeceği için KAA protokolü tarafından kontrol edilir.

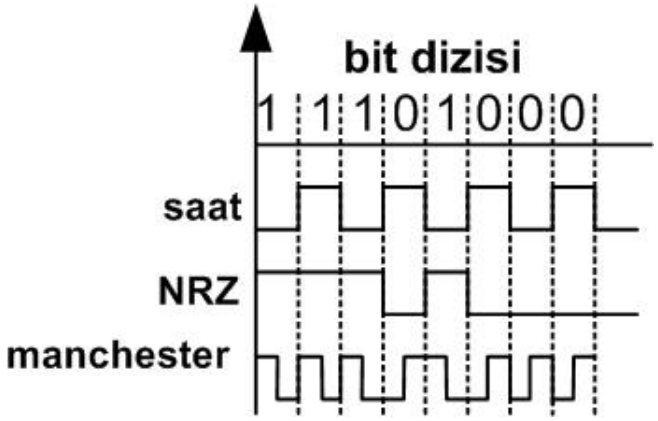
Şekil-2.7'de KAA kontrol edicilerde hata kaydedici içeriklerine göre oluşan hata mesaj türleri ve bus-off durumu verilmiştir. Bus-off durumu oluştuğunda KAA kontrol edici yeniden başlatılana kadar ağ üzerinde mesaj-alışverişi yapamaz, ağ üzerindeki herhangi bir KAA kontrol edici ilk kez ya da başlatıldığında önce kontrol ağına bir aktif hata mesajı gönderir.



Şekil 2.7. Bus-off durumu

2.6.4. Kontrol Alan Ağı Protokolü Fiziksel Bit Kodlama Yöntemi

Kontrol Alan Ağı protokolünde mesajlar sıfıra dönmeyen örnekleme (Non Return to Zero) yöntemi ile fiziksel ortam üzerinde gönderilmektedir [1,6], NRZ yöntemi hem basitlik hem de bant genişliğini daha verimli kullanmasından dolayı tercih edilmektedir. Şekil 2.8'de örnek olarak 1-Byte'lık bir dizinin NRZ ve Manchester yöntemleriyle örneklenmesi gösterilmiştir.

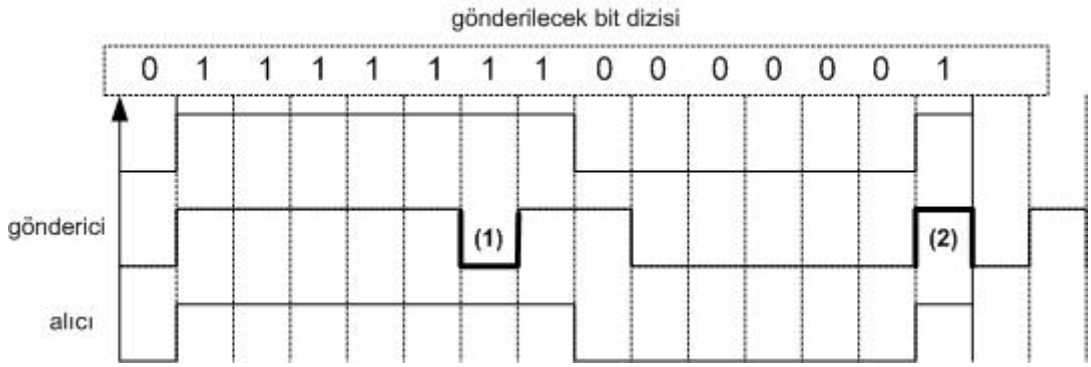


Şekil 2.8. NRZ bit örnekleme

KAA protokolünde gönderilen alınan tüm mesajlar hata tespitinin daha etkili yapılabilmesi için bit-stuffing yöntemi ile kodlanarak iletim yoluna gönderilirler [1]. KAA mesaj çerçevesinde ACK ve EOF alanları hariç bütün alanlar bu yöntemle kodlanarak gönderilir. Bu yöntem sayesinde hem daha etkili hata tespiti yapılmakta hem de KAA kontrol ediciler tarafından veri/istek mesajları ile hata mesajları birbirinden kolayca ayırt edilebilmektedir.

KAA protokolünde hata mesajları 6-adet lojik-0 seviyesinde veya 6-adet lojik-1 seviyesindeki bit dizileri ile başlar bu nedenle herhangi bir istasyon 6-adet ardışık aynı seviyede bit dizisi elde ettiğinde bunun hata mesajı olduğunu anlar, elbette ki bir KAA mesaj çerçevesinde veri yada başka bir alanda ardı ardına 6 veya daha fazla aynı seviyede bit gelebilir. İşte bit-stuffing yöntemi KAA protokolünde hata mesajları ile normal veri ya da istek mesajlarını ayırt eder.

Eğer veri ya da istek mesajının herhangi bir alanında 6 ya da daha fazla sayıda aynı seviyede bit gelecekse gönderim esnasında 5.bit ile 6.bit arasına ters işaretli bir bit yerleştirilir, stuff-bitleri normalde çerçeve alanlarında yer almaz alıcı istasyon tarafından da ihmal edilir. Böylece normal bir veri ya da istek mesajında ardışık 6 bitin aynı seviyede olmasına izin verilmez ve ardışık olarak aynı seviyedeki 6-bitin bir hata mesajı olduğu garanti edilir. Şekil 2.9'da bit-stuff yöntemi için bir örnek verilmiştir.



Şekil 2.9. Bit-Stuffing kodlama

KAA protokolünde karışıklığa sebep olmaması için alıcı istasyonlar 5 adet aynı seviyeli bit aldıklarında 6.bit'i beklerler eğer 6.bit farklı seviyedeyse stuff-bit'i olduğu için dikkate alınmaz eğer 6.bit'te aynı seviyede ise bunun bir hata mesajı olduğu anlaşılır.

Şekil 2.9'da verilen örnek incelendiğinde gönderilecek bit dizisi gönderici tarafından iletim hattına stuff bitleri eklenerek gönderiliyor. Şekilde (1) numaralı durumda gönderilecek bit dizisi ardışık olarak 7-adet lojik-1 seviyesinde bit içerdiği 5.bit gönderildikten sonra lojik-0 değerli stuff-bit yerleştiriliyor, daha sonra kalan iki adet lojik-1 değerli bit gönderiliyor. Diğer (2) numaralı durumda yine ardışık 6-adet lojik-0 değerli bit geldiği için 6.bit stuff-biti olarak otomatik olarak lojik-1'e tamamlanıyor, stuff bitinden sonra 6.bit gönderiliyor.

Alıcı istasyon tarafından (1) numaralı durumda ardışık 5-adet lojik-1 değerli bit dizisi elde edildiği için otomatik olarak 6.bitin lojik-0 seviyede olmasını bekliyor, 6.bit farklı seviyede geldiği için stuff biti olduğu anlaşılarak ihmal ediliyor, yine (2) numaralı durumda ardışık 5-adet lojik-0 değerli bit dizisi elde edildiği için 6.bitin lojik-1 seviyede olması bekleniyor.

2.7. Kontrol Alan Ağı Protokolü Fiziksel Özellikler

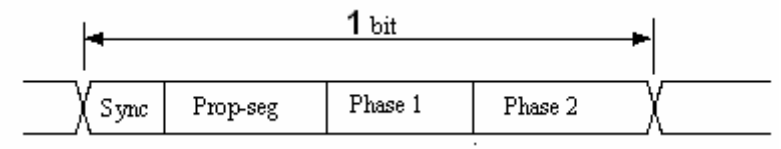
Bu bölümde KAA protokolü programlanabilir veri hızı ve fiziksel standartlar anlatılacaktır.

2.7.1. Kontrol Alan Ağı Protokolü Programlanabilir Veri Hızı

Kontrol Alan Ağı protokolü standart sürümde maksimum 1MBit/s oranında, uzatılmış sürümde ise maksimum 125KBit/s oranında veri iletişimi yapılabilir.

Standart KAA protokolünde iki istasyon arasında ki mesafenin 40 metreyi aşmaması ve iletim yolunun dış etkenlerden kaynaklanan elektromanyetik gürültüye maruz kalmaması kaydıyla 1MBit/s oranına kadar veri iletişim hızını destekler. KAA protokolü uygulamalarda

esneklik sağlanması amacıyla veri hızının belirtilen aralıklarda değişken olarak yeniden programlanabilmesini sağlar. Standart KAA protokolü kontrol edicileri 5Kbit/s ile 1MBit/s oranında veri iletişimi yapmak için yeniden programlanabilir. Şekil-2.10'da KAA protokolü için 1-bit zamanlaması gösterilmektedir.



Şekil 2.10. KAA protokolü 1-bit zamanı

Bu segmentler veri hızını programlamak amacıyla KAA kontrol edicilerde değişik isimlerle adlandırılan kaydediciler olabilirler, genel kullanım aşağıdaki şekildedir:

Sync Segment: Daima 1 zaman uzunluğundadır.
Propagation Segment: Veri yolu üzerindeki gecikmeleri telafi eder. Değer 1…8 zaman uzunluğunda olabilir.
Phase Segment1, Phase Segment2: Senkronizasyon süresini uzatmak veya kısaltmak için kullanılır. Phase Segment1 1…8 arasında, Phase Segment2 2…8 arasında programlanabilir.
BRP: Baud Rate Prescaler değeridir, veri iletişim hızını uzatmak ya da kısaltmak için kullanılır, 1…256 arasında programlanabilir.
SJW Kaydedicisi: Eş zamanlama atlama uzunluğu, maksimum zaman uzunluğuna karar veren kaydedicidir, eş zamanlama yeniden yapılması gereken durumlarda referans olarak kullanılır.

Baud Rate = $f_{crystal}$ / ((BRP)*n) olarak bulunur.
n=Sync+Prop_seg+Phase1+Phase2'dir. (n) bir bit uzunluk zamanı olarak ifade edilir.
($f_{crystal}$) KAA kontrol edicinin saat girişidir.
(n) 4 ile 25 zaman uzunluğunda olabilir.

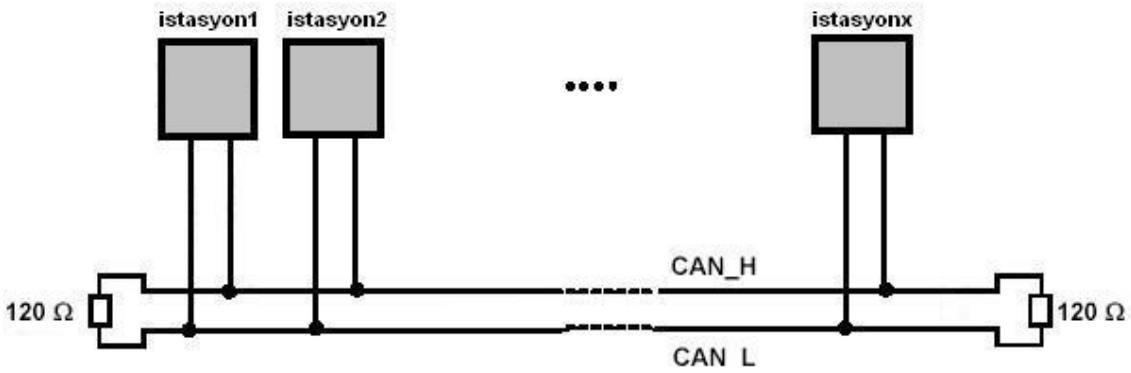
Tablo 2.3'te (BRP) ve (n) değerleri için hesaplanmış örnek veri hızları verilmiştir.

Tablo 2.3. KAA protokolü örnek veri hızları

KAA Veri Hızı	BRP	N	PS+PBS1	PBS2
1 Mbps	1	15	12	2
	2	10	7	2
500Kbps	2	20	16	3
	3	15	12	2
	4	12	9	2
	5	10	5	4

2.7.2. Kontrol Alan Ağlarında Fiziksel Kablo, Konnektör ve Sonlandırma

KAA protokolü doğrusal yol topolojisini kullanır, her iki ucu 120ohm'luk dirençlerle sonlandırılmış çift burgulu kablolar kullanılır. Standart ve genişletilmiş KAA protokolü için kullanılan kablo, konektör ve fiziksel bit seviyeleri için farklı standartlar mevcuttur [1]. Şekil 2.11'de örnek bir KAA topolojisi görülüyor.



Şekil 2.11. Örnek KAA topolojisi

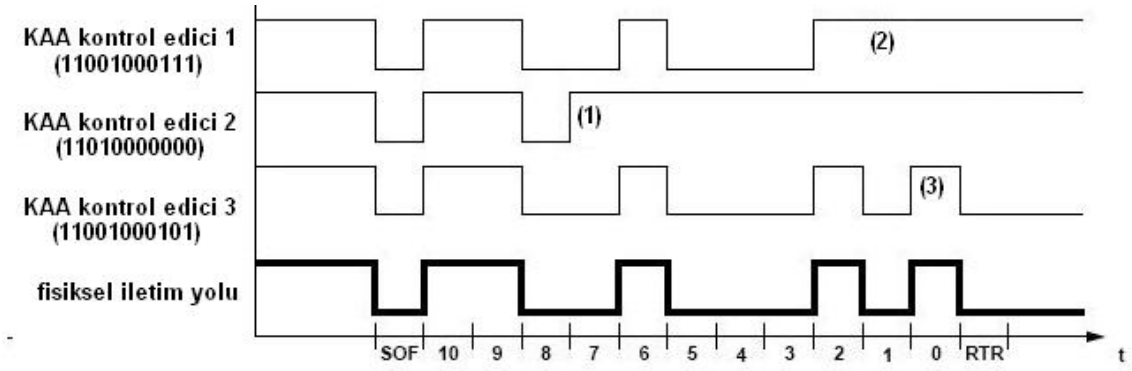
Şekil 2.11'de verilen örnek KAA topolojisi için bilinmesi gerekli fiziksel mesafe, iletim yolu ve düğüm sonlandırma prensipleri Bölüm 4'te verildiğinden dolayı burada ayrıntıya girilmeyecektir.

2.8. Kontrol Alan Ağı Protokolü Veri Yoluna Erişim Prensibi

KAA protokolü diğer haberleşme protokollerinin aksine veri yolunda iki mesajın çarpışmasına izin vermez. İki ya da daha fazla mesajın çarpışmamasını garanti altına almak için öncelik bilgisi kullanılır böylece aynı anda yola erişmek isteyen birkaç istasyon olması durumunda öncelik değeri yüksek olan istasyon yol erişimini kazanır. KAA protokolü mesaj tanımlayıcıları aynı zamanda öncelik bilgileri olarak kullanır bu teknik sayesinde ağdaki istasyon sayısı kadar öncelik belirtilmiş olur, önceliklerin belirlenmesi ise sistem tasarımcısına bağlıdır [6,7].

Aşağıdaki örnekte görüldüğü aynı anda yola erişmek isteyen bütün istasyonlar önce veri yolunu lojik-1 seviyesinde görerek yolun boş olduğu bilgisini elde ederler bu noktadan sonra bütün istasyonlar lojik-0 değerli SOF bitini veri yoluna koyarlar, SOF alanını mesaj tanımlayıcı izler, her istasyon mesaj tanımlayıcı alanı 11.bitinden başlayarak sırayla veri yoluna koyar eğer bütün istasyonların tanımlayıcı bitleri aynı seviyedeyse bir sonraki bit veri yoluna koyulur, yüksek seviyeli bit değerine sahip olan istasyon yol erişimini kaybeder böylece yola

erişim kalan istasyonlar arasında devam eder, en kötü ihtimalle tanımlayıcı alanının 1.bitine gelindiğinde kalan son iki istasyondan yüksek değerli bit seviyesine sahip olan yola erişimi kaybeder. Kontrol alan ağında mesaj tanımlayıcılar aynı zamanda öncelik belirleyici olarak kullanıldığı için bu teknik herhangi iki ya da daha fazla istasyona ait mesajların çarpışmamasını garanti altına alır.



Şekil 2.12. Kontrol alan ağı yola erişim önceliği

Şekil 2.12'de örnekte bir kontrol ağında aynı anda mesaj göndermek için iletime geçen üç adet KAA kontrol edici için yola erişimin kazanılması prensibi gösterilmiştir. Her bir kontrol ediciye ait hedef istasyon adresi parantez içinde verilmiştir, sırayla örnekteki durumları açıklayacak olursak:

- **(1):** Çerçeve başlangıç bitinin yola konulmasından bu duruma kadar üç istasyonda aynı değeri göndermiştir, (1) numaralı duruma gelindiğinde; KAA kontrol edici 2 fiziksel yola lojik-1, KAA kontrol edici 1 ve KAA kontrol edici 3 fiziksel yola lojik-0 yazmıştır. Bu nedenle KAA kontrol edici 2 yola erişim hakkını kaybetmiştir, bu noktadan sonra yola erişim yarışı 1 ve 3 numaralı KAA kontrol ediciler arasında geçecektir.
- **(2):** Bu durumda KAA kontrol edici 1 fiziksel yola lojik-1, KAA kontrol edici 3 lojik-0 yazmak istediği için KAA kontrol edici 1 yola erişim hakkını kaybeder.
- **(3):** Kontrol edici 3 tanımlayıcı alana ait birinci biti fiziksel yola koyduğu andan itibaren kontrol ağı üzerinde yola erişim yapmak isteyen başka istasyon kalmadığı için yol hakkına sahip olmuş olur.

2.9. Kontrol Alan Ağı Protokolü Hata Algılama Mekanizması

KAA protokolünde kontrol ağında oluşabilecek hataları algılamak için kullanılan yöntemleri beş farklı grupta ele alınır[1,6,7]:

- Toplam Sınama Kontrolü (Cylic Redundancy Check)
- Onay (ACK) Kontrolü
- Çerçeve Formatı Kontrolü
- Veri yolunun izlenmesi (bit mönitoring)
- Bit-stuff kodlama

2.9.1. Toplam Sınama Kontrolü (CRC)

Gönderici istasyon gönderdiği mesajın CRC alanına özel bir matematiksel işlem sonucu elde ettiği üreteç polinomu yazar. Alıcı istasyon kendisine gelen mesajdaki CRC alanını tekrar hesapladığında 0 elde etmelidir aksi halde geçersiz CRC alanı elde edilir ve hata mesajı üretilir.

2.9.2. ACK Kontrolü

Birçok seri iletişim tabanlı protokolde hata algılamak ve mesajın teslim edilip edilmediği bilgisini elde etmek için kullanılan en etkin yöntemlerden biridir. Gönderici istasyon mesajın hedefine gidip gitmediğini anlamak için hedef istasyondan onay bekler, eğer beklenen onay gelmezse ACK hatası oluşur ve hata mesajı üretilir.

2.9.3. Çerçeve Formatı Kontrolü

KAA protokolünde alınan ve gönderilen mesajlardaki çerçeve alanları KAA protokolü özelliklerine uygun olarak denetlenir. Eğer çerçeve alanlarında bir uyumsuzluk varsa çerçeve hatası oluşur ve hata mesajı üretilir. Alınan mesajda veri uzunluk alanında belirtilen sayıyla veri boyunun uyuşmaması bu türde oluşabilecek hataya örnektir.

2.9.4. Veri Yolunun İzlenmesi

KAA protokolü yayım(broadcast) tabanlı bir iletişim protokolü olduğu için mesajlar bütün istasyonlarca okunabilir. Gönderici gönderdiği mesaj ile veri yolu üzerindeki mesajı bit-bit karşılaştırır, istasyonun gönderdiği mesaj biti ile fiziksel veri yoluna koyulan bitin farklı

değerlere sahip olması sonucunda bit hatası oluşur ve gönderici göndermekte olduğu mesajın tamamlanmasını beklemeden hata mesajı yayınlamaya başlar. KAA protokolünde hata algılamak için kullanılan en etkin yöntemlerden biri bit takip yöntemidir.

Kontrol Alan Ağı protokolünde hata tespiti için kullanılan bu 4 farklı yöntem oldukça yüksek bir güvenlik sağlar. KAA protokolü başlangıçta otomotiv endüstrisi için tasarlandığı için yüksek güvenliğe ihtiyaç duymaktadır, bu şekilde bir yöntemin algılayamayacağı hata başka bir yöntem tarafından büyük ihtimalle algılanacaktır. Kontrol Alan Ağında tespit edilemeyen bir hata oluşması durumu çok düşüktür.

Bit-stuffing yöntemi Bölüm 2.6'da anlatıldığı için burada tekrar değinilmemiştir.

2.9.5. Kontrol Ağında Hata Algılanması Durumunda Yapılanlar

Kontrol Alan Ağında herhangi bir hata oluşması durumunda sırayla aşağıdaki işlemler yapılır:

- Hata gönderici istasyon tarafından sezilirse mesaj iletmeyi durdurur
- Hata alıcı istasyon tarafından sezilirse mesaj almayı durdurur
- Hatayı algılayan istasyon hemen hata mesajını yayınlar(aktif ya da pasif hata mesajı)
- Hata sayıcı kaydedici içerikleri güncellenir
- Hata mesajı yayımlama ile bütün istasyonlara duyurulur
- Yeniden iletime geçmek için 23 bit periyodu süre beklenir
- 23-bit periyodu sonunda mesaj tekrar gönderilir
- Eğer bus-off durumu oluşmuş ise iletimi tamamen kesilir, haberleşme gerçekleştirilemez

2.10. Kontrol Alan Ağı Protokolü Kontrol Ediciler ve Sınıflandırılması

Kontrol Alan Ağı tabanlı endüstriyel uygulamalarda kullanılan kontrol ediciler için birkaç farklı sınıflandırma ölçütü mevcuttur [7].

Mimariye göre: Bağımsız kontrol ediciler, tümleşik kontrol ediciler ve genişleme modülleri olarak üçe ayrılır, tümleşik kontrol edicilerde mikro denetleyici ve kontrol edici aynı devre üzerinde üretilmiştir, bağımsız kontrol edicilerde ise harici bir mikro denetleyiciye ihtiyaç

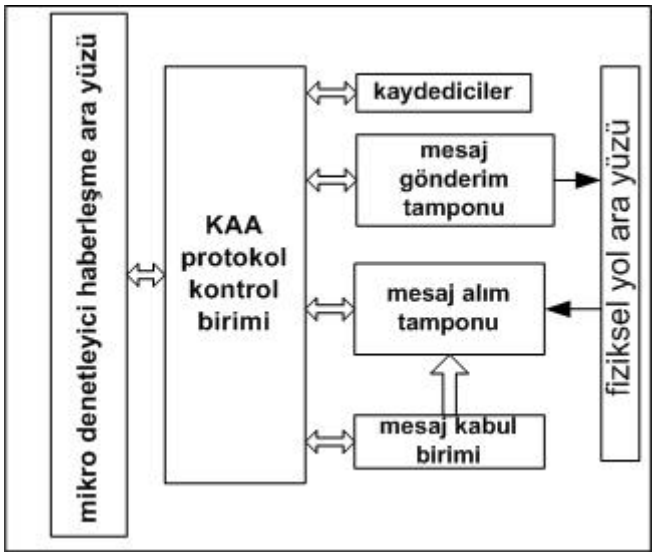
duyar, genişleme modülleri ise bağımsız kontrol edicilerden farklı olarak dahili analog-sayısal dönüştürücüler gibi çeşitli giriş/çıkış birimlerine sahiptir.

Mesaj Tanıtıcıya göre: KAA kontrol edicileri 11bitlik ya da 29bitlik mesaj tanımlayıcı kullanmasına göre de ayrılırlar, bazı kontrol ediciler ise her iki tanıtıcı düzeyini aynı anda kullanabilmektedir.

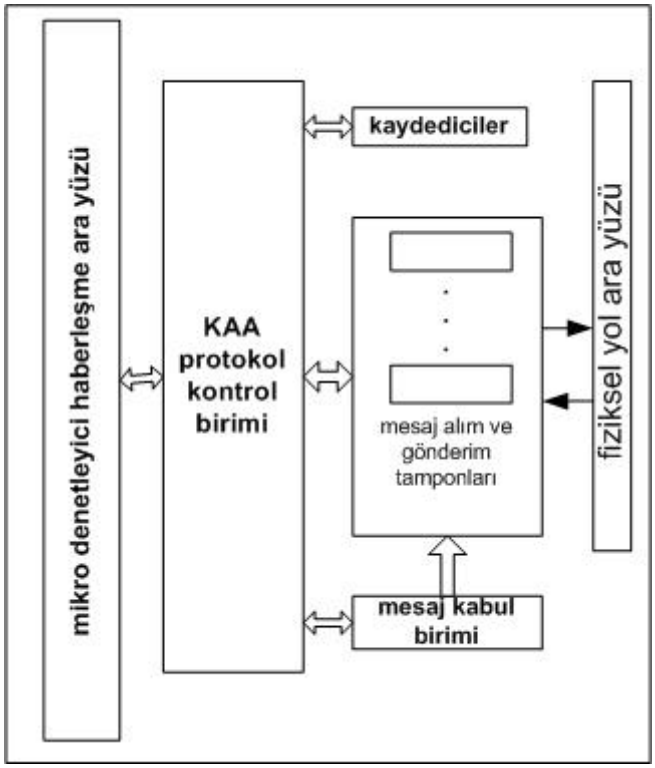
Kullanım Sahasına göre: KAA kontrol edicileri uygulanacak endüstriyel uygulamanın türüne bağlı olarak tümleşik devre üzerinde bazı ekstra algılayıcı ya da giriş-çıkış modülleri bulundurabilirler bu nedenle de kullanım sahasına göre harflerle sınıflandırılırlar bunlar:

- **A sınıfı:** Elektronik devre tasarımında
- **B sınıfı:** Bilgi paylaşımı uygulamaları
- **C sınıfı:** Gerçek zamanlı kontrol uygulamalarında
- **D sınıfı:** Mobil iletişim uygulamalarında kullanılırlar.

BasicCAN ve FullCAN: Kontrol edicinin işlem yüküne ve gelen mesajları nasıl tamponlandığına bağlı olarak yapılır. BasicCAN kontrol ediciler daha ucuz olmasına rağmen düşük işlemci gücüne sahiptir, FullCAN kontrol ediciler daha yetenekli işlemciler kullanır. BasicCAN kontrol edicilerde genel olarak birer adet gönderici ve alıcı mesaj tamponu olmasına rağmen FullCAN kontrol edicilerde iki veya daha fazla mesaj alım/gönderim tamponu içerirler [6,7].



Şekil 2.13. BasicCAN kontrol edici



Şekil 2.14. FullCAN kontrol edici

2.11. Kontrol Alan Ağı Protokolü Tabanlı Yüksek Seviyeli Protokoller

Çeşitli firmalar tarafından tasarlanan KAA protokolü tabanlı yüksek seviyeli protokoller aşağıda listelenmiştir. Yüksek seviyeli protokoller kullanıcıyı özellikle donanımsal ayrıntıları çok fazla bilmesine gerek kalmadan uygulamalar gerçekleştirebilmesini sağlaması açısından önem taşımaktadır [21]. Sözü edilen yüksek seviyeli protokoller pratikte genellikle, KAA protokolü tabanlı deney setlerinde ya da kontrol ağlarının Ethernet ağlarda olduğu gibi kullanılabilmesini olanak sağlamak amacıyla kullanılmıştır. Endüstriyel alandaki kullanımlarına örnek ise aşağıda her bir KAA tabanlı yüksek seviyeli protokol için ayrıca verilmiştir.

- **SAE J 1939:** Standart ve genişletilmiş protokol ile tümleşik çalışabilen yüksek seviyeli bir protokoldür kamyon, tır ve otobüslerde kullanılmak için tasarlanmıştır.

- **CANOpen:** KAA tabanlı yüksek seviyeli bir protokoldür, çeşitli endüstriyel uygulamalarda, yön kontrol sistemlerinde, medikal elektronik sistemlerinde ve denizcilikte kullanılmaktadır.

- **DeviceNet:** Rockwell tarafından tasarlanmıştır diğerlerinden farklı olarak bina otomasyon uygulamalarında kullanılmıştır.

- **CANKingdom:** Endüstriyel ağlar için geliştirilmiş KAA tabanlı yüksek seviyeli bir protokoldür, standart ve genişletilmiş KAA protokollerini destekler. Microchip MCP2515 KAA geliştirici deney setleri buna örnek olarak verilebilir.

3. MICROCHIP MCP2510 KAA KONTROL EDİCİ

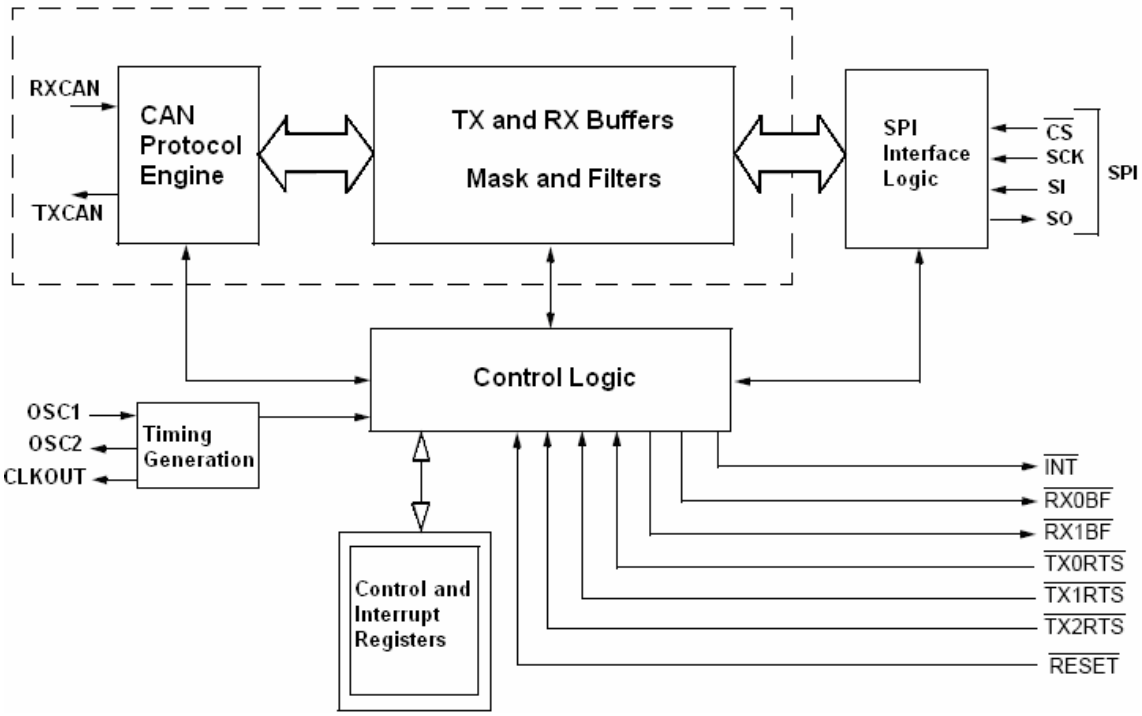
3.1. Genel Bilgi

MCP2510 Microchip firmasının özellikle deneysel eğitim amaçlı ve endüstriyel uygulamalarda KAA Protokolü işlevlerini yerine getirmek için ürettiği bağımsız kontrol edicidir [22].

KAA protokolü kontrol ediciler işlem yüküne göre genel olarak BasicCAN ve FullCAN olarak ikiye ayrılırlar. Bu ayrımdaki en belirleyici özellik kontrol edicinin standart/genişletilmiş KAA protokollerinden hangisini desteklediğidir. Microchip MCP2510 bağımsız KAA kontrol ediciler her iki protokolü de desteklediği için FullCAN sınıfında yer alırlar [22].

MPC2510 KAA kontrol edicilerde üç adet mesaj gönderim ve iki adet mesaj alım tamponuna bulunmaktadır.

Şekil 3.1'de MCP2510 kontrol edici blok diyagramı görülmektedir.



Şekil 3.1. MCP2551 blok diyagram

3.2. Entegre Bloklarının Görevleri:

Aşağıda sırayla MCP2510 kontrol edici bloklarının görevleri anlatılmıştır.

SPI Interface Logic(Spi Ara Yüzü): Mikro denetleyici MCP2510 kontrol edici ile SPI(Serial Peripheral Interface) ara yüzü üzerinden haberleşir, MCP2510 kaydedicilerinden okuma, kaydedicilere yazma ya da kontrol ediciyi yeniden başlatma, çalışma moduna geçirme vb.. işlemler SPI Protokolü üzerinden gerçekleştirilir [22].

SPI senkron seri haberleşme protokolüdür [5], PIC16F877 mikro denetleyicilerde dahili olarak bulunur senkron olduğu için saat darbesine ihtiyaç duyarlar, SPI arabirim devrelerinde 8-bitlik kayan kaydediciler gönderilecek ve alınacak 1-byte veriyi tutarlar. Kaydediciye yazma, kaydedici içeriğini okuma işlemleri için özel emir kodları kullanılır, bu tür işlemler Bölüm 5'da ayrıntılı olarak ele alınacaktır.

Mikroişlemciden gelen komutları yorumlayarak bağımsız kontrol edici kaydedicilerini setler, gönderilecek mesajın transmit buffer'a yazılması, gelen mesajın receive buffer'dan okunması, kesmelerin değerlendirilmesi ve durum bilgileri gibi kaydedici içeriklerini mikroişlemciye gönderir.

TX and RX Buffers(Mesaj Gönderim ve Alım Tampon Bellek Alanları): MCP2510 kontrol ediciler 3 adet gönderim ve 2 adet alım tampon bellek alanına sahiptir, Mikro denetleyiciden gelen mesaj önce gönderim tampon alanına yazılır. Her bir gönderici tampon alana yazılan mesaja farklı bir öncelik değeri verilebilir böylece aynı anda veri gönderilmek istendiğinde bu tampon alanlarda yer alan mesajlardan hangisinin ilk önce gönderileceği tespit edilmiş olur.

Kontrol ağından algılanan ve kabul filtresinden geçen mesajlar alıcı tampon bellek alanına kaydedilirler. Alıcı tampon alanlardan bir veya ikisi aktif olarak kullanılabileceği gibi algılanan tüm mesajların kabul filtresinden geçmeden kaydedilmesi mümkündür, mesaj kabul konusu ilerleyen bölümlerde ayrıntılı olarak ele alınacaktır.

Mask and Filters(Maske ve Filtreler): Kontrol alan ağında algılanan mesajların hangilerinin kabul edileceği maske ve filtre kaydedicilerine bakılarak karar verilir. MCP2510 kontrol edicilere ait mesaj kabul mantığı ilerleyen bölümlerde ayrıntılı olarak ele alınacaktır.

CAN Control Engine(KAA Kontrol Birimi): KAA protokolünün tüm işlevlerini yerine getirir. Gönderici tampon alana kayıtlı mesajların KAA protokolüne uygun çerçeve haline getirilerek kontrol ağına gönderilmesi, kontrol ağından algılanan mesajların kabul filtresinden geçirilerek kaydedilmesi, CRC alanın karşılaştırılması, bit zamanlaması, hata durumlarının değerlendirilmesi ve hata kaydedici içeriklerinin güncellenmesi vb.. KAA protokolüne özgü tüm işlevlerin yerine getirilmesinden sorumludur.

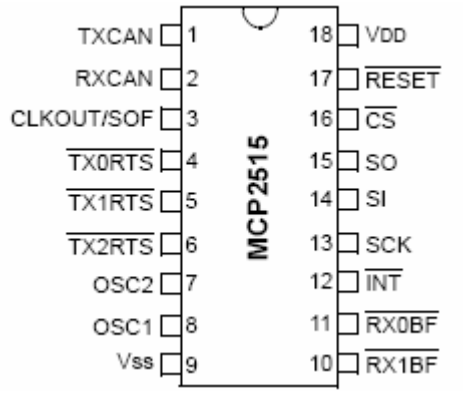
Control Logic(Kontrol Birimi): Kontrol edicinin donanımsal olarak yeniden başlatılması, gönderici tampon alanlar harici girişleri ve alıcı tampon alanlar harici çıkışlarının ve kesmelerin yönetilmesinden sorumludur. Bu kısımda yer alan işlevler mesaj gönderim ve alım işlevlerinde ayrıntılı olarak ele alınacaktır.

Control and Interrupt Registers(Kontrol ve Kesme Kaydedicileri): Maske ve filtre kaydedicileri dışında kalan KAA protokolü işlevlerinin yerine getirilmesi için gerekli olan tüm diğer kaydediciler bu kısımda yer alır, bu kaydedicilerin içerikleri ve kullanımı ilerleyen bölümlerde yeri geldikçe ayrıntılı olarak ele alınacaktır.

Timing Generation(Zamanlama Birimi): Kontrol edici harici saat girişi ve çıkışının bulunduğu kısımdır. Saat devresi mikro denetleyici ve kontrol edicilerde ortak olarak ta kullanılabilir.

3.3. Bacak Konfigürasyonu ve Görevleri

Aşağıda Şekil 3.2'de MCP2510 kontrol edici bacak konfigürasyonu görülmektedir, Tablo 3.1'de ise bacakların görevleri açıklanmıştır.



Şekil 3.2. MCP2510 bacak konfigürasyonu

Tablo 3.1. MCP2510 bacak görevleri

Numara	İsim	Açıklama	Alternatif kullanım
1	TXCAN	KAA veri yolu gönderici çıkışı	-
2	RXCAN	KAA veri yolu alıcı girişi	-
3	CLKOUT	Harici saat çıkışı	-
4,5,6	TXnRTS	Lojik-0 girildiğinde belirtilen gönderici tampon bellekteki mesaj gönderim işlemini başlatır.(n=0,1,2)	Alternatif olarak kontrol edici harici sayısal giriş olarak kullanılabilir
7,8	OSC2,OSC1	Saat çıkış ve girişi	-
9	Vss	Entegre toprak çıkışı	-
10,11	RXnBF	Aktif edilirse ilgili alıcı tampon bellek alana yeni bir mesaj kaydedildiğinde mikro denetleyiciye bir kesme üretir. (n=0,1)	Alternatif olarak kontrol edici harici sayısal çıkış olarak kullanılabilir.
12	INT	Mikro denetleyiciye kesme çıkışı	-
13	SCK	SPI ara yüzü saat girişi	-
14	SI	SPI ara yüzü veri girişi	-
15	SO	SPI ara yüzü veri çıkışı	-
16	CS	Lojik-0 girildiğinde SPI ara yüzü aktif olur	-
17	RESET	Lojik-0 girildiğinde kontrol edici yeniden başlatılır	-
18	Vdd	Entegre besleme girişi	-

3.4. MCP2510 Kontrol Edici Çalışma Modları

MCP2510 kontrol ediciler beş çalışma moduna sahiptir, bu modlar maddeler halinde aşağıdaki gibi sıralanabilir [22].

- Konfigürasyon modu
- Normal modu
- Uyuma modu
- Dinleme modu
- Test modu

Her bir çalışma modunun kendine has işlevleri bulunmaktadır, kontrol edicinin hangi çalışma modunda çalışacağı kontrol (CANCTRL) kaydedicisinden belirlenir, aşağıda sırayla bu çalışma modları ayrıntılı olarak anlatılmıştır.

3.4.1. Konfigürasyon modu

Kontrol edici ilk kez çalıştırıldığında ya da yeniden başlatıldığında standart olarak bu moda geçer. Kontrol edicinin hangi iletişim hızında çalışacağı, ağ üzerinde algılanan mesajların hangilerinin kabul edileceği ve mesaj gönderim işleminin nasıl yapılacağı bu safhada belirlenir.

Bu bilgiler ile ilgili değişiklik yapılmak istendiğinde kontrol edici tekrar bu çalışma moduna geçirilmelidir. Aşağıda maddeler halinde konfigürasyon modunda erişim yapılabilen kaydediciler sıralanmıştır [22]:

- İletişim Hızı Yapılandırma Kaydedicileri (CNF1, CNF2, CNF3)
- Mesaj Gönderim Durum ve Kontrol Kaydedici (TXRTSCTRL)
- Mesaj Alım Tampon Bellek Kontrol Kaydedicileri (RXB0CTRL, RXB1CTRL, BFPCTRL)
- Tüm filtre ve maske kaydedicileri
- Kesme kaydedici (CANINTE)

3.4.2. Normal Çalışma modu

MCP2510 Kontrol edici standart çalışma modudur. Kontrol ağı üzerinde algılanan ve kabul edilen mesajların kaydedilmesi, diğer istasyonlara mesaj gönderme, hata durumlarının değerlendirilmesi ve belirlenen durumlarda mikro denetleyiciye kesme üretilmesi gibi tüm işlevler bu çalışmada modunda gerçekleştirilir.

3.4.3. Uyuma modu

Bu çalışma modunda kontrol edici kontrol ağı üzerinde tekrar çalışma moduna geçene kadar mesaj gönderim ya da alım işlemleri yürütmez, ağ üzerindeki bir istasyon belli bir zaman dilimi içerisinde mesaj gönderim ya da alım işlemleri yapmayacaksa güç tasarrufu sağlaması amacıyla uyuma moduna geçirilir.

3.4.4. Dinleme modu

Normal moda benzer, kontrol edici dinleme modunda ağ üzerindeki tüm mesajları algılar fakat gelen mesajlar için onay(ACK) göndermez. Hata durumlarını değerlendirmez yani bir istasyon dinleme modunda çalışıyor ise hata durumu oluşturmaz, kendisine gelen aktif ya da pasif hata mesajlarıyla ilgili de işlem yapmaz.

Endüstriyel uygulamalarda veri toplama ve Ethernet ağlarda olduğu gibi tüm ağ üzerindeki mesaj trafiğinin dinlenmesi amacıyla bir istasyon dinlene modunda çalıştırılabilir.

3.4.5. Test modu

Test modu kontrol edicinin fiziksel bir veri iletim hattına veya başka bir istasyona ihtiyaç duymadan mesaj gönderim ve alma işlemi yapabileceği çalışma modudur.

Kontrol edicinin test edilebilmesini sağlar. Bu çalışma modunda gönderilen mesajlar için onay beklenmez. İstasyon kendi gönderdiği mesajı yine kendisi alır. Gönderici tampon bellekten gönderilen mesaj paralel olarak alıcı tampon bellek alana yazılır.

3.5. MCP2510 Kontrol Edici Konfigürasyon İşlemleri

MCP2510 kontrol ediciler mesaj gönderim ve alımı gibi işlemlerini normal modda gerçekleştirirler [22]. Bu işlemlerin neye göre yapılacağı ise konfigürasyon modda belirlenir. Kontrol edicinin istenen şekilde çalışabilmesi konfigürasyon modunda gerekli parametrelerin doğru bir şekilde girilebilmesiyle mümkündür.

Şekil 3.3‘te MCP2510 bağımsız kontrol edici konfigürasyon işlemleri akış diyagramı olarak verilmiştir. İlerleyen kısımlarda ise konfigürasyon modunda kullanılan kaydedici içerikleri ve fonksiyonları verilmektedir.

Şekil 3.3. MCP 2510 konfigürasyon işlemleri

3.5.1. Kontrol Kaydedici (CANCTRL Register)

En önemli fonksiyonu kontrol edici için Bölüm 3.4'te anlatılan çalışma modlarından birinin seçilmesini sağlamasıdır. Tablo 3.2'de kaydedici içeriği ayrıntılı olarak verilmiştir.

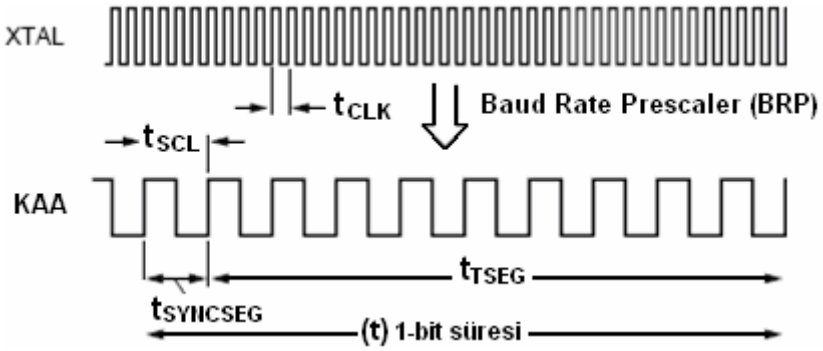
Tablo 3.2. Kontrol kaydedici

Bit pozisyonları	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Bit isimleri	REQOP2	REQOP1	REQOP0	ABAT	-	CLKEN	CLKPRE1	CLKPRE0
Fonksiyon	**000:** normal çalışma modu **001:** uyuma modu **010:** test modu **011:** dinleme modu **100:** konfigürasyon modu *Not:Diğer durumlar geçersizdir*			**1:** tüm mesaj gönderim istekleri iptal **0:** hepsi aktif		Pin.3 aktif/pasif **0:** pasif **1:** aktif	Pin.3 Saat böleni **00:** f/1 **01:** f/2 **10:** f/4 **11:** f/8 **(f):** saat frekansı	
Açıklama	Çalışma modunu değiştirir						Bit2 '0' yapılırsa önemsizdir	

3.5.2. İletişim Hızı Kaydedicileri (CN1, CNF2, CNF3 Registers)

MCP2510 KAA kontrol edicilerde iletişim hızının programlanması için üç adet kaydediciden faydalanılır. KAA kontrol edicilerin birçoğunda iletişim hızının belirlenmesi için benzer şekilde iki veya üç adet kaydedici bulunur. Farklı KAA kontrol edicilerde iletişim hızının belirlenmesi için kullanılan kaydedici içerikleri farklılık gösterse de iletişim hızının hesaplanması hepsi için aynıdır.

Şekil 3.4'te harici saat frekansı ile KAA iletişim hızı arasında ki ilişkiyi ve iletişim hızı hesaplanmasında kullanılan parametreler gösterilmektedir [22,23].



Şekil 3.4. KAA bit zamanlaması

XTAL: Harici saat giriş frekansı
tSCL: KAA dahili saat periyodu
tCLK: Harici saat giriş periyodu
BRP: Harici saat böleni
tSYNCSEG: Kontrol ağı üzerinde 1-bit süresi için zaman uyumlama süresi
tTSEG: Değişken veri hızı zamanı
tSJW: 1-bit için yeniden zaman uyumlama süresi
t: KAA protokolü 1-bit periyodu

Tablo 3.3'te MCP2510 iletişim hızı için kullanılan yukarıda sözü edilen parametrelerin yer aldığı konfigürasyon kaydedicileri içerikleri verilmiştir. Tablo 3.3'te yer alan SJW, BRP, PHSEQ0, PHSEQ1; PHSEQ2 kavramları Bölüm 2.7.1'de KAA iletişim hızının hesaplanmasında açıklandığı için burada tekrar değinilmemiştir.

Tablo 3.3. İletişim hızı kaydedicileri

Kaydedici adı	Bit pozisyonları							
	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Konfigürasyon1 (CNF1)	SJW1	SJW0	BRP5	BRP4	BRP3	BRP2	BRP1	BRP0
Konfigürasyon2 (CNF2) Açıklama	mode	SAM **1:** üç defa **0:** bir defa *Veri yolu örnekleme sayusu*	PHSEG1			PHSEG0		
Konfigürasyon3 (CNF3)	-	WAKFIL **1:** aktif **0:** pasif *Uyuma modundan dönme*	-			PHSEG2		

3.5.3. Kesme kaydedicisi (CANINTE Register)

Kontrol edicinin normal çalışma modunda mesaj alış verişi gibi KAA protokolü işlevlerini yerine getirirken hangi durumlar için mikro denetleyiciye bir kesme üreteceği konfigürasyon modunda bu kaydediciden belirlenir.

Normal çalışma modunda kesme kaydedicide aktif edilen durumlardan herhangi biri oluşursa mikro denetleyiciye bir kesme üretilir, mikro denetleyici bir kesme aldığında kesme altprogramı aktif olur. Kesme altprogramında kesme bayrak kaydedici(CANINTF) içeriğine bakılarak hangi kesme oluşmuş ise onunla ilgili hizmet yerine getirilir. Kesme bayrak kaydedici ile kesme kaydedicideki bit pozisyonları aynıdır, hangi kesme türü oluşmuş ise ilgili bit pozisyonu kontrol edici tarafından ‘1’ yapılır.

Tablo 3.4. Kesme Kaydedici

Bit no	Kısaltma	İsim	Açıklama
Bit7	MERRE	Message Error Interrupt Enable	Mesaj alışverişi sırasında hata meydana gelirse kesme üret
Bit6	WAKIE	WakeUp Interrupt Enable	Kontrol edici uyuma modundaysa normal çalışma moduna dönmesi için kesme üret
Bit5	ERRIE	Error Interrupt Enable	Hata bayrak kaydedicide değişiklik olursa kesme üret
Bit4	TX2IE	Transmit Buffer2 Emtpy Interupt Enable	Gönderici tampon bellek2 yeni bir mesaj gönderimi için hazırsa kesme üret
Bit3	TX1IE	Transmit Buffer1 Emtpy Interupt Enable	Gönderici tampon bellek1 yeni bir mesaj gönderimi için hazırsa kesme üret
Bit2	TX0IE	Transmit Buffer0 Emtpy Interupt Enable	Gönderici tampon bellek0 yeni bir mesaj gönderimi için hazırsa kesme üret
Bit1	RX1IE	Receive Buffer1 Full Interupt Enable	Alıcı tampon bellek1 ‘e yeni bir mesaj gelirse kesme üret
Bit0	RX0IE	Receive Buffer0 Full Interupt Enable	Alıcı tampon bellek0 ‘a yeni bir mesaj gelirse kesme üret
Tüm pozisyonlar için ‘1’ ilgili kesme türünü aktif ‘0’ pasif yapar			

3.5.4. Filtre ve Maske Kaydedicileri

Tüm filtre ve maske kaydedicileri ile mesaj alım tampon bellek kontrol kaydedicileri konfigürasyon modunda yapılandırılabilir, bu kaydedicileri ile ilgili ayrıntılı bilgiler Bölüm 3.7‘de ele alınmıştır.

3.5.5. Mesaj Gönderim Durum ve Kontrol Kaydedici(TXRTSCTRL)

MCP2510 kontrol edicide mesaj gönderim işlemi SPI ara yüzü aracılığıyla veya kontrol edici TXnRTS girişleri kullanılarak gerçekleştirilebilir.

Mesaj gönderim işleminin başlatılması için söz edilen bu iki yöntem Bölüm 3.6‘da ele alınmıştır. Kontrol edicinin bu yöntemlerden hangisini kullanacağı konfigürasyon modunda bu kaydedici aracılığıyla belirlenir. Tablo 3.5‘te bu kaydedici içeriği ve fonksiyonları verilmiştir.

Tablo 3.5. Mesaj gönderim durum ve kontrol kaydedici

Bit no	Kısaltma	açıklama
Bit7,6	-	Bu bitler kullanılmıyor
Bit5	B2RTS	Sayısal giriş olarak kullanılıyor ise ilgili bacağın o anki değerini verir
Bit4	B1RTS	
Bit3	B0RTS	
Bit2	B2RTSM	**0:** sayısal giriş olarak kullan **1:** mesaj gönderimini başlatmak için kullan Not: *4, 5, 6 numaralı bacakların durumunu belirler*
Bit1	B1RTSM	
Bit0	B0RTSM	

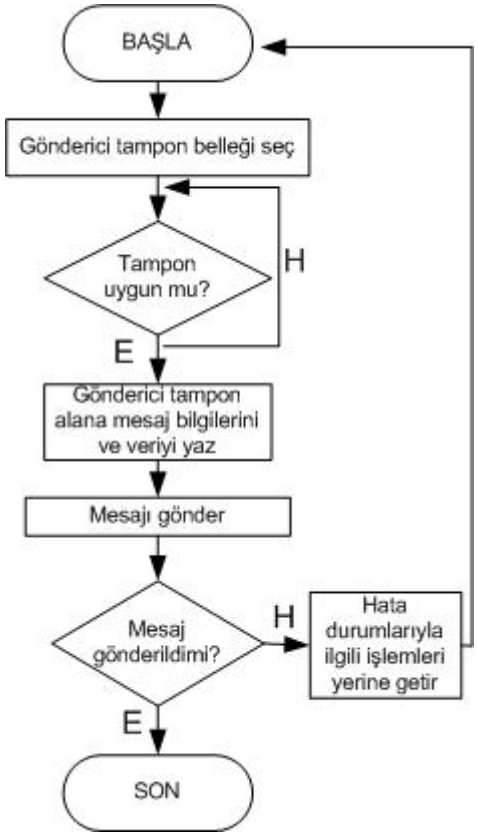
3.5.6. Minimum Konfigürasyon Bilgisi

Bir KAA kontrol edicinin kontrol ağı üzerinde normal çalışma modunda çalışabilmesi için en azından iletişim hızını belirleyen kaydedicileri yazılmalıdır bu durum minimum konfigürasyon olarak bilinir [22,23].

Örneğin bir istasyon hiçbir zaman mesaj almayacak sadece diğer istasyonlara mesaj gönderecek ise filtre ve maske kaydedicileri yazılmayabilir, benzer şekilde mikro denetleyici kesmeli hizmet yürütmeyecekse kesme kaydedicinin de yazılmasına gerek yoktur.

3.6. MCP2510 Mesaj Gönderim İşlemi

MCP2510 kaydedici için mesaj gönderim işlemi Şekil 3.5'te akış diyagramında basitçe verilmiştir, akış diyagramında yalın haliyle görülen her bir adımda yapılan işlemler ve kullanılan kaydediciler hakkında ayrıntılı bilgiler ilerleyen kısımlarda verilmiştir.



Şekil 3.5. MCP2510 mesaj gönderim işlemi

3.6.1. MCP2510 Mesaj Gönderim Tampon Bellek Alanları

MCP2510 kontrol ediciler herbiri 14-byte uzunluğunda üç adet gönderici tampon bellek alanı içermektedirler. Tablo 3.6'da gönderici tampon bellek alanların görünümü verilmektedir.

Tablo 3.6. Mesaj Gönderim Tamponu

Tampon alanları	Açıklama
TXBnCTRL	Gönderici tampon bellek kontrol kaydedici (*)
TXBnSIDH	Standart KAA mesajı için 11-bitlik tanımlayıcı alanın anlamlı 8 biti
TXBnSIDL	Standart KAA mesajı için 11-bitlik tanımlayıcı alanın anlamsız 3 biti (*)
TXBnEID8	Genişletilmiş KAA mesajı için 29-bitlik tanımlayıcı alanın anlamlı 8 biti
TXBnEID0	Genişletilmiş KAA mesajı için 29-bitlik tanımlayıcı alanın anlamsız 8 biti
TXBnDLC	Her iki KAA mesaj türü için RTR biti ve DLC alanı (*)
TXBnDm	0-8 byte uzunluğunda veri alanı
(*) bu kaydedici içerikleri farklı alanlarda içerdiklerinden aşağıda ayrıca verilmiştir, diğerleri için aynı durum geçerli olmadığından ayrıca değinilmemiştir.	

3.6.1.1. Gönderici Tampon Bellek Kontrol Kaydedici(TXBnCTRL Register)

Mesaj gönderimi için seçilen tampon belleğin durumu hakkında ayrıntılı bilgi verir. Anlamlı ilk 4-biti sadece okunabilir, anlamsız son 4-bit ise kullanıcı tarafından değiştirilebilir. Bu kaydedicinin fonksiyonları maddeler halinde aşağıdaki gibi sıralanabilir.

- Daha önce gönderilen mesajın iletilip iletilmediğini gösterir.
- Mesaj gönderilirken yola erişim önceliğinin kaybolup kaybolmadığını gösterir.
- Mesaj gönderimi sırasında hata meydana gelip gelmediğini gösterir.
- Gönderici tampon bellek alanın mesaj gönderimi için uygun olup olmadığını gösterir.
- Mesaj önceliğinin belirlenmesini sağlar

Tablo 3.7. Gönderici tampon bellek kontrol kaydedici

Bit no	Kısaltma	İsim	Açıklama
Bit7	-	-	Bu bit kullanılmıyor
Bit6	ABTF	Message Aborted Flag	**0:** Mesaj iletimi tamamlanmış, onay(ACK) alınmış **1:** Mesaj için onay(ACK) alınmamış
Bit5	MLOA	Message Lost Arbitration	**0:** Mesaj iletimi sırasında yola erişim kaybedilmemiş **1:** Mesaj iletimi sırasında yola erişim kaybedilmiş
Bit4	TXERR	Transmission Error Detection	**0:** Mesaj iletimi sırasında hata oluşmamış **1:** Mesaj iletimi sırasında hata oluşmuş
Bit3	TXREG	Message Transmit Request	**0:** Tampon bellek mesaj gönderimi için uygun değil **1:** Tampon bellek mesaj gönderimi için uygun
Bit2	-	-	Bu bit kullanılmıyor
Bit1	TXP1	Message priority bit 1	**00:** düşük öncelik … **11:** yüksek öncelik
Bit0	TXP0	Message priority bit 0	

Şekil 3.5'te akış diyagramında da görüldüğü gibi mesaj gönderim işlemini başlatmadan önce söz konusu gönderici tampon bellek alanının durumu kontrol edilmelidir. Mesaj gönderim işleminin başlatılabilmesi için ilk 4-bit '0' olmalıdır, bu bitlerden herhangi biri '1' durumunda ise ilgili durum kullanıcı programı tarafından kontrol edilerek değerlendirilmelidir, aşağıda bu durumlar için yapılması gerekenler maddeler halinde listelenmiştir.

- **Bir önceki mesaj için onay(ACK) alınmamış ise:** Bu durumda istasyon gönderdiği mesaj için onay(ACK) beklemektedir, bu bitle beraber hata kaydedicileri de kontrol edilmelidir, eğer herhangi bir hata durumu oluşmamış ise gönderilen mesaj hiçbir istasyon tarafından kabul edilmemiş demektir. Bu durumda alıcı istasyonların mesaj kabulü için kullandıkları filtre ve maske kaydedicileri kontrol edilmelidir.
- **Mesaj gönderimi sırasında yola erişim kaybedilmişse:** Bu durumda ağ üzerinde iki istasyon aynı anda mesaj iletimine başlamıştır ve söz konusu istasyon diğer istasyona göre daha düşük öncelikli tanımlayıcı alana sahip olduğu için yola erişim önceliğini kaybetmiştir, diğer istasyon mesaj iletimini tamamladığında yola erişim tekrar kazanılacaktır. Bu durumda eğer bu bit '1' ise '0' olana kadar beklenmelidir.
- **Mesaj gönderimi sırasında hata meydana gelmişse:** Hata bayrak kaydedici kontrol edilerek meydana gelen hata durumuyla ilgili işlemler yerine getirilmelidir. Eğer bus-off durumu oluşmuş ise kontrol edici yeniden başlatılmalıdır.
- **Gönderici tampon bellek mesaj gönderimi için uygun değilse**: Yukarıda sözü edilen bir veya daha fazla durum oluşmuş ise söz konusu gönderici tampon bellek mesaj gönderim işlemi yapamaz. Bu durumda ilgili durumlar değerlendirilerek ya da gönderici tampon bellek kontrol kaydedici TXREG biti temizlenerek gönderici tampon bellek alan mesaj gönderimi için uygun hale getirilir.

3.6.1.2. Mesaj Önceliği

MCP2510 kontrol edici üç adet gönderici tampon bellek alan içermektedir, her bir alandaki mesaj için farklı bir öncelik değeri verilerek bu bellekler arasındaki öncelik belirlenmiş olur[22]. Yüksek önceliğe sahip bellekteki mesaj önce gönderilir. Eğer her üç alan için aynı öncelik değeri belirlenir ise sayısal olarak yüksek değerli gönderici tampon alandaki mesajın önceliği de daha yüksek olur.

3.6.1.3. Diğer Gönderici Tampon Bellek Kaydedicileri

Tablo 3.6'da yer alan kaydediciler TXBnSIDL ve TXBnDLC kaydedici içeriklerinde bit pozisyonlarının farklılığından dolayı burada ayrıca değinilmiştir. Tablo 3.8 ve Tablo 3.9'da sırasıyla bu iki kaydedici içeriği ve fonksiyonları verilmiştir.

Tablo 3.8. TXBnSIDL kaydedici

Bit pozisyonları	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Açıklama	Standart tanımlayıcı son 3 bit			-	EXIDE	-	Genişletilmiş tanımlayıcı en anlamlı ilk 2 bit	
EXIDE: 1: Genişletilmiş KAA protokolü mesajı **0:** Standart KAA protokolü mesajı								

Tablo 3.9. TXBnDLC kaydedici

Bit pozisyonları	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Açıklama	-	RTR	-		DLC3	DLC2	DLC1	DLC0
RTR: 1: istek mesajı (bu durumda veri alanı yazılmaz) **0:** veri mesajı								

Veri uzunluk alanına(DLC) yazılan değer ile veri alanı arasındaki ilişki Bölüm 2.5.6 'da anlatılmıştır. Veri uzunluk alanı ile veri alanına yazılan veri uzunluğunun farklı olması durumunda mesaj gönderimi sırasında çerçeve hatası oluşacağından [1,7] bu duruma dikkat edilmelidir. Gönderilen mesaj bir istek mesajı ise veri alanına herhangi bir şey yazılmaz, yazılması durumunda yine çerçeve hatası meydana gelecektir.

3.6.2. Mesaj Gönderim İşleminin Başlatılması

MCP2510 kontrol edici mesaj gönderim işleminin iki farklı şekilde başlatılmasını destekler, bu yöntemlerden hangisinin kullanılacağı konfigürasyon modunda belirlenir.

SPI ara yüzü aracılığıyla hangi gönderici tampon bellek alandaki mesaj gönderilmek isteniyor ise ilgili SPI emri kontrol ediciye gönderilir. SPI işlemleri Bölüm 5'te ayrıntılı olarak anlatıldığından burada fazla değinilmeyecektir. Mesaj gönderim işlemini SPI ara yüzü aracılığıyla başlatabilmek için konfigürasyon modunda TXnRTS girişleri sayısal giriş olarak yapılandırılmalıdır [22].

Diğer yöntemde ise kontrol edicinin TXnRTS (4,5 ve 6 numaralı bacaklar) girişleri alçak seviyeye çekilerek yani 1-0 geçişinde mesaj gönderimi başlatılabilir [22].

Mesajın iletilip iletilmediğinin kontrolü de iki şekilde yapılabilir, ilk yöntemde mesaj gönderim kontrol kaydedici kontrol edilerek mesajın iletilip iletilmediği kontrol edilir, ikinci yöntemde ise mesaj gönderim kesmesi konfigürasyon modunda aktif edilir, bu durumda mesaj başarılı bir şekilde gönderildiğinde mikro denetleyiciye bir kesme üretilir.

3.7. MCP2510 Mesaj Alım İşlemi

MCP2510 kontrol edici iki farklı mesaj alım tampon bellek alanına sahiptir, Şekil 3.6'da mesaj okuma işlemi akış diyagramı olarak verilmiştir.

Şekil 3.6. MCP2510 mesaj okuma işlemi

MCP2510 kontrol edicilerde kontrol ağı üzerinden algılanan ve kabul edilen mesajları okumak için üç farklı yol sunarlar, bu yöntemler maddeler halinde kısaca aşağıdaki gibi sıralayabiliriz [22]:

- SPI ara yüzü aracılığıyla: Normal çalışma modunda bir döngü içerisinde kontrol edici mesaj alım durumu spi komutlarıyla kontrol edilerek mesajın geldiği tampon bellek alandan kayıtlı mesaj okunur. Spi ara yüzü ile yapılan kontrol işleminde hangi tampon bellekte yeni bir mesaj olduğu bilgisinin yanı sıra mesajın türü(veri, istek, standart/genişletilmiş KAA protokolü) ve mesajın hangi filtreden edildiği bilgileri de elde edilir.
- Kesmeli kontrol ile: kontrol edici konfigürasyon modunda yeni bir mesaj geldiğinde mikro denetleyiciye kesme üretecek şekilde yapılandırılırsa normal çalışma modunda kontrol ediciye yeni bir mesaj gelmesi durumunda mikro denetleyiciye bir kesme üretilir, kesme altprogramında gelen kesme türü belirlenerek alıcı tampon bellek alanlardan kayıtlı mesaj okunur.
- Kontrol edici RX0BF ve RX1BF çıkışları ile: Kontrol edici konfigürasyon modunda RX0BF ve RX1BF pin çıkışları yeni bir mesaj geldiğinde harici kesme üretecek şekilde yapılandırılırsa kontrol ediciye yeni bir mesaj gelmesi durumunda hangi tampon bellek alana yeni bir mesaj geldiyse ilgili çıkıştan mikro denetleyiciye bir kesme üretilir. Bu yöntemin gerçeklenebilmesi için mikro denetleyicinin birden fazla kesme girişini destekliyor olmalıdır. Her bir alıcı tampon farklı çıkışlardan kesme üreteceği için yukarıda anlatılan ikinci yönteme kıyasla daha sadedir.

3.7. 1. MCP2510 Mesaj Alım Tampon Bellek Alanları

MCP2510 mesaj alım tampon bellek alanlarına ait kaydedici görünümleri Bölüm 3.6‘da anlatılan mesaj gönderim tampon bellek alanlarıyla aynıdır bu yüzden burada ayrıca değinilmeyecektir.

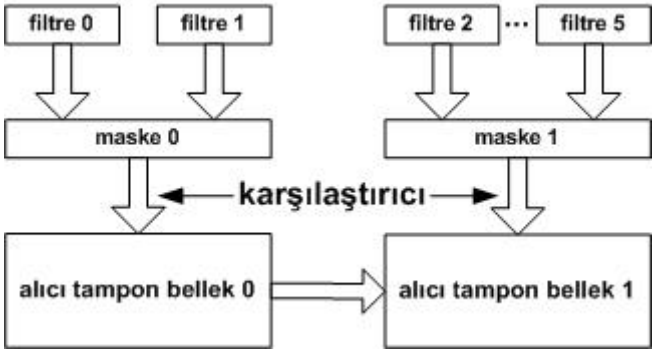
3.8. MCP2510 Mesaj Kabul Mantığı

MCP2510 kontrol edicilerde mesaj kabulü için gerekli yapılandırmalar konfigürasyon modunda yapılmalıdır. Şekil 3.7‘de görüldüğü gibi ilk alıcı tampon bellek alanı için iki farklı

filtre kabulü bulunmasına rağmen ikinci alıcı tampon bellek alanı için dört farklı filtre kabulü vardır[22]:

Alıcı tampon bellek alanlarından ilki daha yüksek önceliğe sahiptir, bu alana gelen mesajlarda taşma olması durumunda gelen mesajların ikinci alıcı tampon bellek alana kaydedilmesi şeklinde yapılandırılabilir.

Her bir alıcı tampona ait maske ve filtre kaydedicileri karşılaştırılarak mesajların kabul edilip edilmeyeceğine karar verilir. Bu esnek mesaj kabul mantığı Ethernet ağlarda olduğu gibi alıcı istasyonlar için tek bir geçerli adres yerine bir adres aralığı verilebilmesini de olanaklı kılar.



Şekil 3.7. MCP2510 mesaj kabul filtresi

3.8.1. Filtre ve Maske Kaydedicileri

Tablo 3.10'da gelen mesajların tanımlayıcı alanları ile maske ve filtre kaydedici içeriklerinin karşılaştırılması sonucu kabul edilen ve kabul edilmeyen durumlar gösterilmiştir.

Tablo özetle şöyle yorumlanabilir: Maske kaydedicideki bit pozisyonu '0' olan durumlarda filtre kaydedici ve gelen mesaj tanımlayıcı alandaki bit değerine bakılmaksızın gelen mesajlar kabul edilir. Maske kaydedicideki bit pozisyonunun '1' olması durumunda ise filtre kaydedici içeriği ile gelen mesaj tanımlayıcı alandaki bit değerleri aynı olanlar kabul, farklı olanlar reddedilir [6,22].

Tablo 3.10. Mesaj kabul filtresi

Maske kaydedici	Filtre kaydedici	Gelen mesajın tanımlayıcı alanı	Sonuç
0	0	0	kabul
0	0	1	kabul
0	1	0	kabul
0	1	1	kabul
1	0	0	kabul
1	0	1	red
1	1	0	red
1	1	1	kabul

3.8.2. Mesaj Alım Tampon Bellek Kontrol Kaydedicileri (RXB0CTRL, RXB1CTRL, BFPCTRL)

Bu bölümde konfigürasyon modunda anlatılmayan bu üç kaydedici içerikleri ve fonksiyonları ayrıntılı olarak verilecektir.

Alıcı tamponlar durum ve kontrol kaydedici RB0BF ve RB1BF çıkışlarının yapılandırılmasından sorumludur. Tablo 3.11'de bu kaydedici içeriği verilmiştir. Kaydedicideki bit0 ve bit1 pozisyonları bu iki çıkışın nasıl yapılandırılacağını belirler.

Tablo 3.11. Mesaj alım tampon bellek kontrol kaydedici

Bit pozisyonları	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Fonksiyonu	-		B1BFS	B0BFS	B1BFE	B0BFE	B1BFM	B0BFM
Açıklama	Kullanılmayan bitler		Sayısal çıkış olarak yapılandırılması durumunda çıkış değerini okur		**0:** pasif **1:** aktif Çıkış fonksiyonunu aktif yada pasif yapar		**0:** sayısal çıkış **1:** harici kesme	

Tablo 3.12'de ise sırayla her iki mesaj alım tampon bellek alan kontrol kaydedicileri verilmiştir.

Tablo 3.12. Mesaj Alım Tampon Bellek Kontrol Kaydedicileri

Bit pozisyonları	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RXB0CTRL	-	RXM1	RXM0	-	RTR	BUKT	BUKT1	FILHIT0
RXB1CTRL	-	RXM1	RXM0	-	RTR	FILHIT2	FILHIT1	FILHIT0

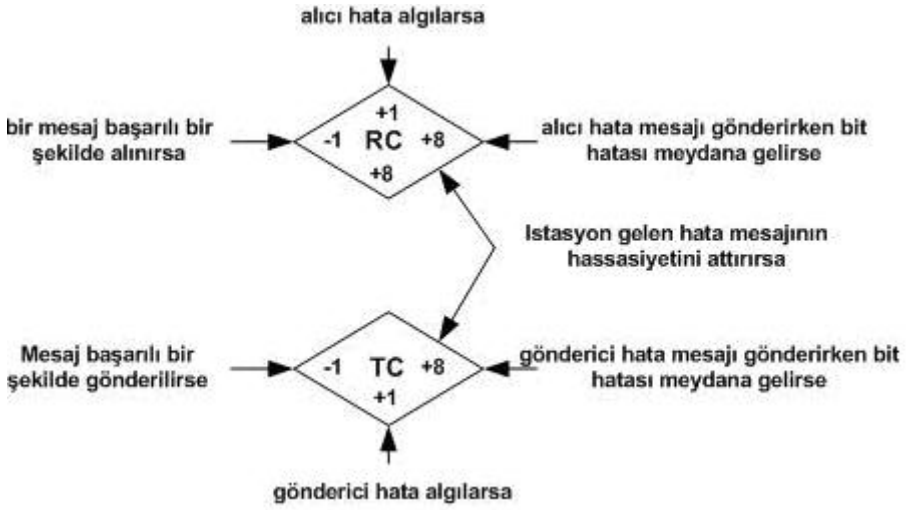
Tablo 3.12'de yer alan RTR alanı istek mesajlarının kabul edilip edilmeyeceğini belirler. Bu bit '1' yapılırsa istek mesajları kabul edilir, '0' yapılması durumunda sadece veri mesajları kabul edilir. BUKT alanı '1' yapılırsa ilk alıcı tampon bellek alanı dolu olduğu halde yeni bir mesaj gelirse ikinci alıcı tampon bellek alana kaydedilir, '0' yapılması durumunda pasiftir. İlk kontrol kaydedicide ki FILTH0 alanı '0' yapılırsa mesaj kabulü filtre 0'a göre '1' yapılırsa mesaj kabulü filtre 1'e göre yapılır. Diğer durumlar Tablo 3.13'te verilmiştir.

Tablo 3.13. Mesaj alım filtre seçme işlemleri

Bit pozisyonları		RXM1	RXM0	açıklama
		0	0	Tüm filtreleri pasif yap, gelen tüm mesajlar kabul edilir
		0	1	Sadece seçilen filtreye uyan genişletilmiş KAA protokolü mesajlarını kabul et
		1	0	Sadece seçilen filtreye standart KAA protokolü mesajlarını kabul et
		1	1	Seçilen filtreye uyan her iki mesaj türünü kabul et
	FILHIT2	FILHIT1	FILHIT0	
	1	0	1	Filtre 5 aktif
	1	0	0	Filtre 4 aktif
	0	1	1	Filtre 3 aktif
	0	1	0	Filtre 2 aktif
	0	0	1	Filtre 1 aktif (BUKT alanı '1' ise)
	0	0	0	Filtre 0 aktif (BUKT alanı '0' ise)

3.9. MCP2510 Hata Durumlarının Değerlendirilmesi

Kontrol alan ağlarında oluşabilecek hata durumları Bölüm 2.9'da verilmiştir. MCP2510 kontrol edici hata durumlarının değerlendirilmesi için üç adet kaydediciden faydalanır. Bu kaydedicilerden gönderici hata sayıcı ve alıcı hata sayıcı kaydedicileri diğer KAA kontrol edicilerde olduğu gibi kullanıcıya sadece sayısal değer olarak bilgi verirler. Bu iki kaydedici içeriğinin güncellenmesi algoritması [1,7] Şekil 3.8'de verilmiştir. Şekilde görülen RC ve TC sırasıyla alıcı/gönderici hata sayıcılarını ifade etmektedir.



Şekil 3.8. Hata kaydedicilerin güncellenmesi

Burada alıcı istasyon hatasız olarak yeni bir mesaj aldığında alıcı hata kaydedicisini 1 eksiltir. Eğer alıcı istasyon mesaj alımında bir hata olduğunu algılarsa alıcı hata kaydediciyi 1 arttırır, alınan hatalı mesajla ilgili kontrol ağına bir hata mesajı yayımlarken yeni bir bit hatası meydana gelirse 8 arttırır.

Gönderici istasyon başarılı olarak bir mesaj gönderirse gönderici hata kaydedicisini 1 eksiltir. Eğer gönderici istasyon mesaj gönderirken hata algılarsa gönderici hata kaydediciyi 1 arttırır, gönderilen hatalı mesajla ilgili kontrol ağına bir hata mesajı yayımlarken yeni bir bit hatası meydana gelirse 8 arttırır.

Ağ üzerindeki istasyon aşırı yüklenme hata mesajı alırsa hem gönderici hem de alıcı hata kaydedici içeriğini 8 arttırır.

Kullanıcıya kontrol ağı üzerinde oluşan hata durumuyla ilgili ayrıntılı bilgiyi ise hata bayrak kaydedicisi verir [22]. Tablo 3.14'te bu kaydedici içeriği verilmiştir.

Tablo 3.14 Hata bayrak kaydedici

Bit pozisyonu	Alan ismi	Açıklama
Bit7	RX1OVR	Alıcı tampon bellek 1'de aşırı yüklenme durumu
Bit6	RX0OVR	Alıcı tampon bellek 0'de aşırı yüklenme durumu
Bit5	TXB0	Bus-off durumu (gönderici hata sayıcı 255 olursa)
Bit4	TXEP	Mesaj gönderim hatası (>=128)
Bit3	RXEP	Mesaj alım hatası (>=128)
Bit2	TXWAR	Gönderici hata kaydedici limit aşım durumu (>=96)
Bit1	RXWAR	Alıcı hata kaydedici limit aşım durumu (>=96)
Bit0	EWARN	Gönderici veya alıcı hata kaydedici limit aşımı (>=96)
tüm pozisyonlar için '1' ilgili durumun oluştuğunu, '0' oluşmadığını gösterir		

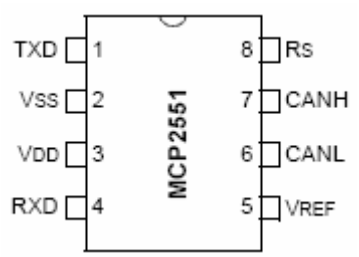
4. MICROCHIP MCP2551 YOL ALICISI

4.1. Genel Bilgi

Yol alıcıları KAA protokolü kontrol edici aygıtlar ile fiziksel iletim yolu arasında ara yüz oluştururlar. Microchip MCP2551 yol alıcısı ISO11898 standartlarına uygun olarak üretilmiştir ve 1Mbit'e kadar iletişim hızını destekler, ağ üzerinde en fazla 112 adet istasyona izin verir ve elektromanyetik gürültülerden fazla etkilenmez [24].

4.2. MCP2551 Yol Alıcı Bacak Konfigürasyonu

Şekil 4.1'de MCP2551 yol alıcısı görülmektedir, Tablo 4.1 'de ise bacak görevleri yer almaktadır.



Şekil 4.1. MCP2551 yol alıcısı

Tablo 4.1. MCP2551 bacak görevleri

Bacak	Açıklama
TXD	Veri gönderim çıkışı (kontrol edicinin TXCAN bacağına)
VSS	Toprak girişi
VDD	Besleme girişi
RXD	Veri alım girişi(kontrol edicinin RXCAN bacağına)
VREF	Elektriksel gürültüyü azaltmak için kullanılan harici çıkış
CANL	KAA iletim yolu bağlantısı
CANH	KAA iletim yolu bağlantısı
RS	Çalışma modu kontrol girişi

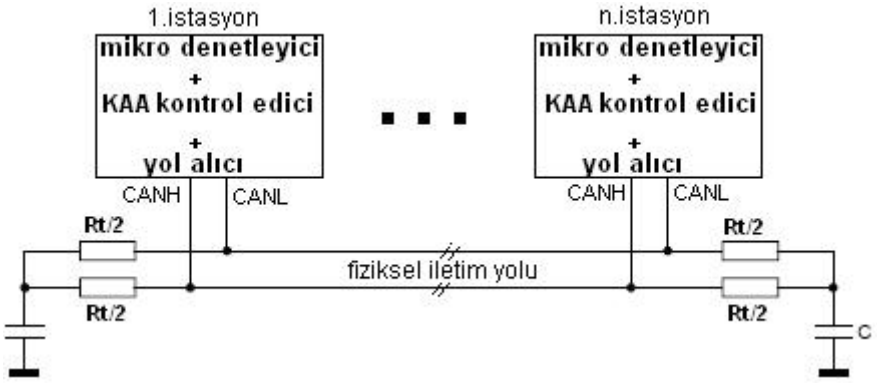
4.2.1. MCP2551 Çalışma Modu Seçimi

MCP2551 yol alıcısı çalışma modu seçimi RS bacağı tarafından kontrol edilir, normal çalışma modunda mesaj alış-verişi yapılabilmesi için bu giriş alçak seviye olmalıdır bu yüzden

direk toprağa da bağlanabilir, bekleme modunda giriş yüksek seviyededir tekrar alçak seviyeye çekildiği anda normal çalışma moduna döner [24].

4.2.2. Tipik Uygulama Devresi

Şekil 4.2 'de tipik bir kontrol alan ağı görülmektedir. Burada sözü edilen her bir istasyon bir mikro denetleyici, KAA protokolü kontrol edici ve yol alıcısından oluşmaktadır, Rt fiziksel yol sonlandırma direncini C ise kapasiteyi ifade etmektedir [6,7,23].



Şekil 4.2. Örnek bir uygulama devresi

4.3. Fiziksel Veri Yolu Standartları

Kontrol alan ağlarında fiziksel yol mesafesi, maksimum iletişim hızı, yol ve istasyon sonlandırma prensipleri ağda kullanılan yol alıcısına ve uygulamaya has olarak değişebilmektedir.

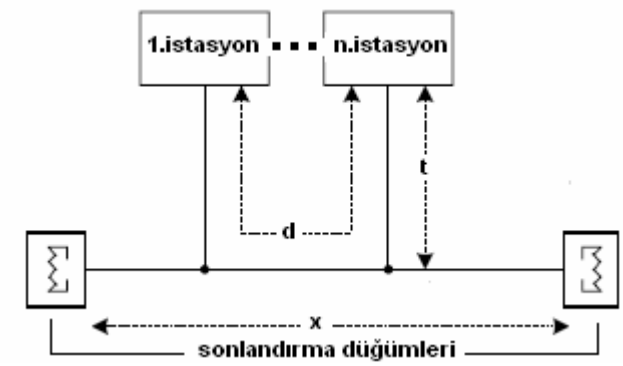
Aşağıda Microchip MCP2551 yol alıcısı için geçerli olan prensipler sırasıyla anlatılmıştır.

4.3.1. Kontrol Ağı Fiziksel Topoloji

ISO11898 standartlarına göre kontrol alan ağlarında UTP CAT5 standart çift kablo kullanılır, her bir istasyonun yol alıcısının CANH ve CANL bacakları UTP kablo çiftine bağlanır, yolun her iki ucu ise 120ohm dirençle biter.

Elektromanyetik gürültüyü azaltmak için her bir istasyonun yola bağlantısı ve yol sonlandırma için kapasite ve dirençlerden oluşan gürültü azaltıcı devreler kullanılabilir, özellikle yüksek hızlarda iletişim yapılacaksa sonlandırma prensipleri çok önemlidir çok düşük

hızlarda çalışılması durumunda ise sonlandırma yapılmaksızın haberleşme sağlanabilir. Şekil 4.3'te kontrol ağı doğrusal topoloji görülmektedir.



Şekil 4.3. Kontrol ağı fiziksel topoloji

Burada (d) mesafesi ağ üzerinde yer alan iki istasyon arasında olması gereken mesafeyi, (t) bir istasyonun fiziksel yola olan uzaklığını ve (x) ise iki sonlandırma düğümü arasındaki mesafeyi ifade eder. Tablo 4.2'de bu mesafelerin olması gereken en kısa ve en uzun değerleri verilmiştir [23].

Tablo 4.2. Kontrol ağı fiziksel mesafeler

simge	En kısa	En uzun
d	0,1metre	30metre
x	0,2metre	10km
t	0metre	1metre

Kontrol alan ağlarında iletişim hızı 5Kbit/s ile 1Mbit/s arasında programlanabilmektedir, 1Mbit/s hızında iletişim yapabilmek için mesafe en fazla 40metre olmalıdır [1,6,7] ve iletim yolu elektromanyetik gürültülerden etkilenmemelidir. Desteklenen en uzun mesafe yani 10km bir kontrol alan ağı üzerinde ise en fazla 5Kbit/s hızında iletişim yapılabilmektedir.

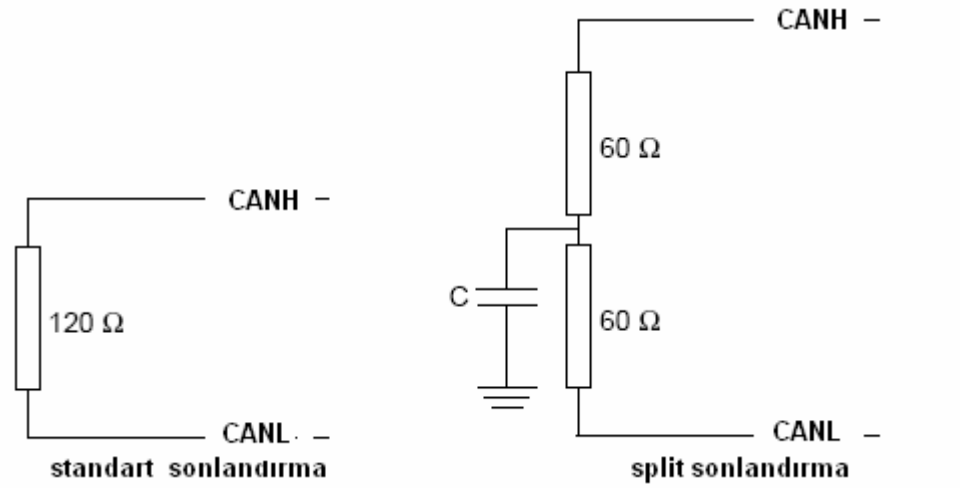
4.3.2. Kontrol Ağı Yol Sonlandırma

Kontrol alan ağlarında genel olarak iki şekilde yol sonlandırma yapılabilir. Şekil 4.4'de bu iki sonlandırma prensibi beraber görülmektedir.

Standart sonlandırma kontrol ağları için en yalın sonlandırma şeklidir ağın her iki ucu şekilde görüldüğü gibi 120ohm değerinde dirençlerle sonlandırılır, bu sonlandırma prensibi

elektromanyetik gürültüden çok etkilenmeyen ve düşük hızlarda iletişim yapılan kontrol ağları için uygundur [23,24].

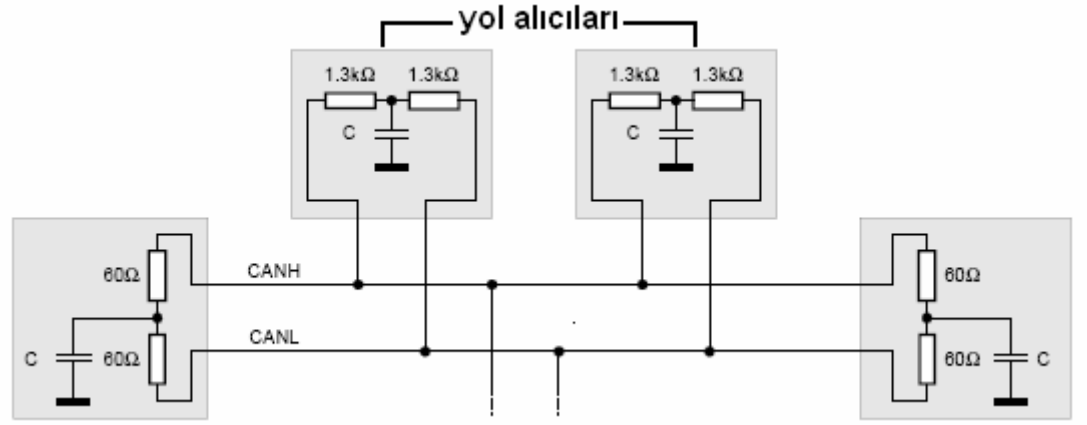
Diğer sonlandırma ise split sonlandırma olarak bilinir, bu sonlandırma şekli yüksek hızlarda iletişim yapılan kontrol ağlarında gürültüyü azaltmak için tercih edilir [23,24].



Şekil 4.4. Kontrol alan ağlarında yol sonlandırma

4.3.3. Kontrol Alan Ağı Düğüm Sonlandırma

Kontrol alan ağlarında yol sonlandırma ve ağ üzerinde her bir istasyona ait yol alıcılarının yola bağlantısı yani düğüm sonlandırma birbirleriyle yakından ilişkilidir. Aşağıda Şekil 4.5'te split sonlandırma prensibi için düğüm sonlandırma şekli verilmiştir, standart sonlandırma prensibi kullanılıyor ise dirençler 1,3Kohm yerine 60ohm olarak seçilir [23,24].



Şekil 4.5 Split sonlandırma prensibi

5. KURULAN DENEYSEL GENEL AMAÇLI ENDÜSTRİYEL AĞ

5.1. Genel Bilgi

KAA Protokolü tabanlı ağlar 1992 yılından itibaren otomotiv sektöründe yaygın olarak kullanılmaktadır. Günümüzde Mercedes, BMW, Opel, Wokswagen başta olmak üzere birçok otomotiv üreticisi ürettiği araçlarında KAA Protokolünü kullanmaktadır [7,8].

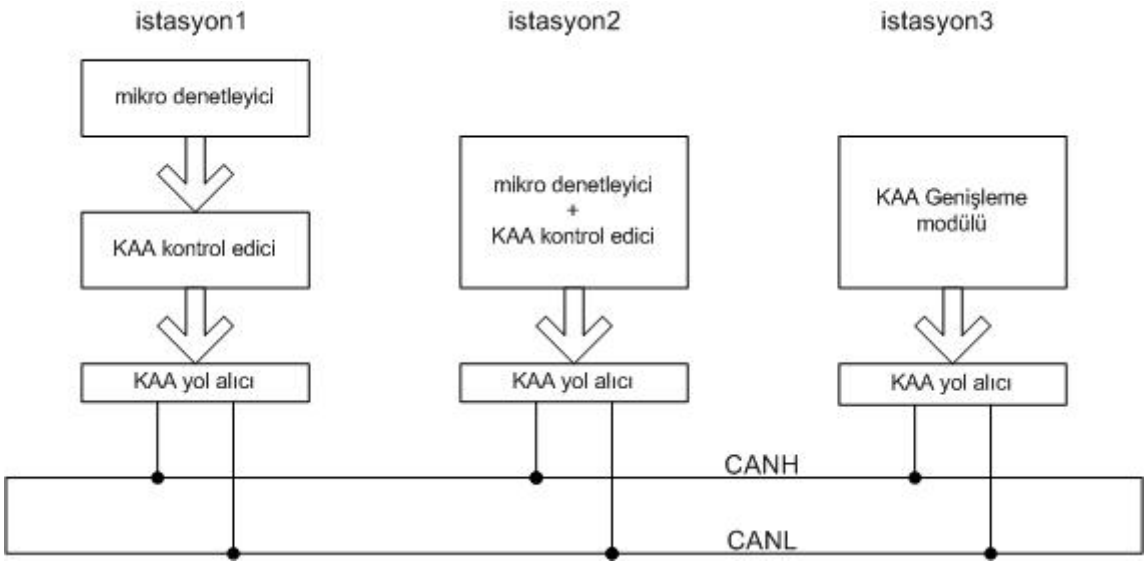
Şekil 5.1'de örnek bir kontrol ağı ve ağ üzerinde olabilecek istasyon türleri ve KAA kontrol ediciler görülmektedir.

Burada istasyon 1 harici bir mikro denetleyici, KAA kontrol edici ve yol alıcısından oluşmaktadır, burada kullanılan KAA kontrol ediciler bağımsız kontrol ediciler olarak adlandırılırlar Micrcochip MCP2510 ve Philips SJA1000 kontrol edicileri örnek olarak verilebilir [6,7,22,23].

İstasyon 2 ise mikro denetleyici ile kontrol edicinin tümleşik olarak üretildiği aygıtları simgelemektedir, Microchip PIC30F ve 18F458 ailesi bu gruba örnek olarak verilebilir [6,7,22,23].

İstasyon 3 'te kullanılan KAA genişleme modülleri üzerlerinde dahili analog-dijital çeviricileri gibi çeşitli giriş/çıkış üniteleri olan aygıtları simgelemektedir, Microchip MCP2505 ve araçlarda yaygın olarak kullanılan Philips SJA2020 kontrol ediciler buna örnek olarak verilebilir [6,7,22,23].

Tez kapsamında tasarlanan eğitim amaçlı KAA protokolü tabanlı endüstriyel ağ üzerinde kurulan istasyonlar, Şekil 5.1'de görülen istasyon 1 türünde istasyonlardır.



Şekil 5.1. Örnek kontrol alan ağı

5.2. Deney Sisteminin Blok Diyagramı

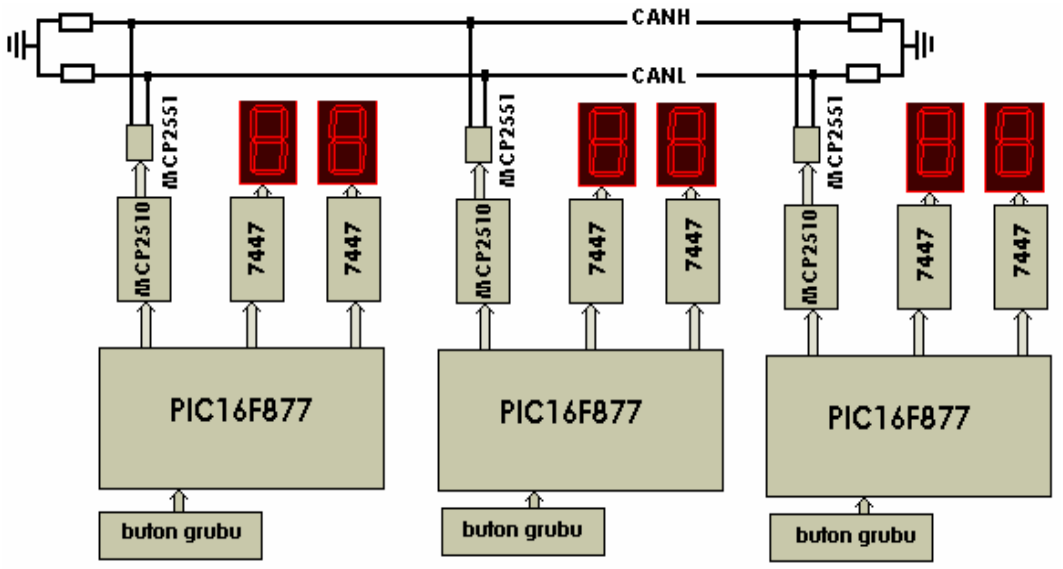
Tez kapsamında kurulan deneysel kontrol ağında 3 adet istasyon yer almaktadır, her bir istasyon PIC 16F87 mikro denetleyici, MCP2510 bağımsız kontrol edici ve MCP2551 yol alıcısından oluşmaktadır. Şekil 5.2'de tasarlanan kontrol ağı blok diyagram olarak görülmektedir.

Burada her iki ucu 120ohm dirençle sonlandırılmış UTP CAT5 standart çift kablo kullanılmıştır, her bir istasyonun yol alıcıları 60ohm dirençlerle iletim yoluna bağlanmıştır. Yalın olması nedeniyle standart sonlandırma prensibi tercih edilmiştir [23,24].

Ağ üzerindeki her bir istasyon gönderici ve/veya alıcı olarak programlanabilmektedir, bir istasyona ait alınan veri ve/veya istasyonun durumuyla ilgili bilgiler mikro denetleyiciye bağlı 7-segment görüntüleyiciler aracılığıyla takip edilebilmektedir, 7-segment görüntüleyiciler ile mikro denetleyici arasında BCD-kod çevrimi yapılabilmesi amacıyla 7447 sürücüleri kullanılmıştır.

Her bir mikro denetleyici önünde yer alan butonlar istasyonu yeniden başlatmak ve kontrol ağı üzerinde bulunan diğer iki istasyona veri gönderme işlevlerini yerine getirirler. Herhangi bir istasyon kendisine yeni bir mesaj geldiğinde kesmeli kontrol ile mesajı okumakta ve 7-segment görüntüleyicilere vermektedir.

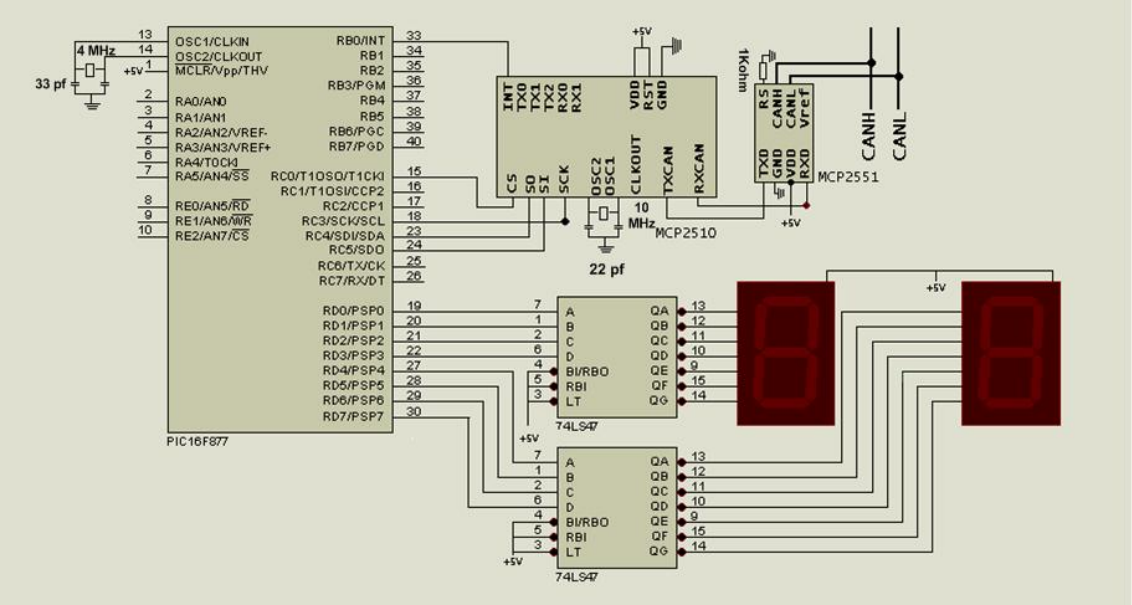
Ayrıca her bir istasyona, kontrol edicinin mesaj alış-verişi için hazır olduğunu, yeni bir mesaj okunduğunu ve hata algılandığını simgeleyen ağ üzerindeki iletişimi kontrol amaçlı ledler yerleştirilmiştir.



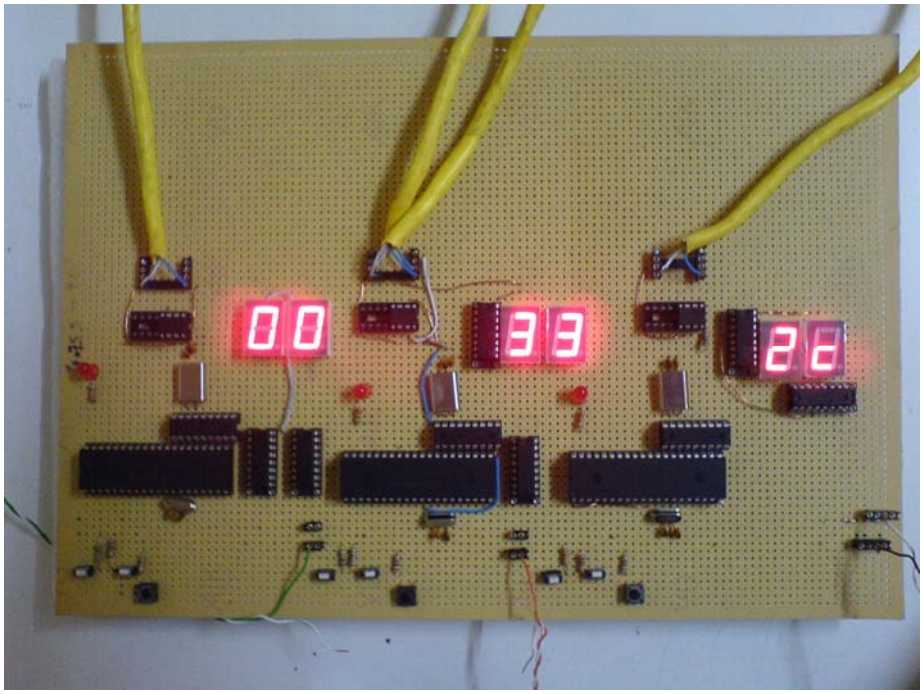
Şekil 5.2. Deney sistemi blok diyagramı

5.2. 1. Ağ Üzerindeki İstasyonların Bağlantı Şeması

Kurulan genel amaçlı deneysel kontrol alan ağında her bir istasyona ait ayrıntılı bağlantı şeması Şekil 5.3'te verilmiştir. Tasarlanan deney setinin çalışma esnasında elde edilen bir resmi ise Şekil 5.4'te verilmiştir.



Şekil 5.3. Deney sistemi bağlantı şeması



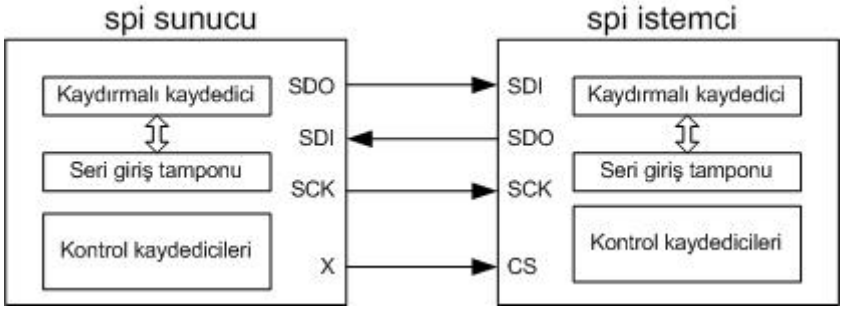
Şekil 5.4. Deney seti

5.3. PIC Mikro Denetleyici ile MCP2510 Kontrol Ediciye Erişim

MCP2510 kontrol ediciler mikro denetleyiciler ile spi aracılığıyla haberleşecek şekilde tasarlanmışlardır. Spi PIC16F877 mikro denetleyicilerde olduğu gibi birçok mikro denetleyicide dahili olarak bulunan eşzamanlı seri iletişim protokolüdür. Spi protokolü aracılığıyla haberleşecek cihazlar istemci ve sunucu olmak üzere iki farklı şekilde yapılandırılabilirler, bu yöntemler maddeler halinde aşağıda açıklanmıştır [5,22,25].

- **Spi sunucu:** Eş zamanlama için gerekli saat işareti spi sunucu tarafından kontrol edildiği için iletişimi herhangi bir anda başlatıp bitirebilir, veri gönderir ve spi istemcinin veriyi ne zaman göndereceğini belirler.
- **Spi istemci:** Spi sunucu kontrolünde veri alır ve gönderir, spi sunucudan gelen 1-byte uzunluğundaki spi emir kodu, kaydedici adresi veya veri önce spi istemci cihazın kaydırmalı kaydedicisine yazılır. Daha sonra MCP2510 KAA kontrol edici spi ara yüzü elde ettiği verilere göre ilgili kaydedicilere erişim işlemini gerçekleştirir.

İki cihazın spi aracılığıyla iletişime başlayabilmesi için gerekli yapılandırma ayarları spi sunucu tarafından yerine getirilir, Şekil 5.5'te spi aracılığıyla haberleşme yapacak iki cihaz için tipik bir bağlantı şekli verilmiştir. Şekilde SDO seri veri çıkışını, SDI seri alım girişini, SCK saat çıkışını ve CS spi istemci cihazı aktif/pasif etme kontrol girişidir.



Şekil 5.5. Örnek spi haberleşme

5.3.1. PIC16F877 SPI Protokolü

MCP2510 kontrol ediciler mesaj alma-gönderme kesme işlemlerinin yapılandırılması ve hata durumlarının değerlendirilmesi gibi işlevler Bölüm 3'te verildiği için burada tekrar değinilmeyecek, pic16f877 mikro denetleyicilerde spi sunucu yapılandırma ayarları ve

mcp2510 kontrol edici kaydedicilerine spi aracılığıyla yapılan okuma yazma gibi erişimler anlatılacaktır.

5.3.2. PIC16F877 SPI Kaydedicileri

Şekil 5.4'te görülebileceği gibi spi modülü haberleşme işlemlerini yerine getirebilmek için bazı kaydedicilere ve bu kaydediciler üzerinde bazı yapılandırma ayarlarına ihtiyaç duyar, bu kaydedicilerin işlevleri [25].

1. **Seri giriş tamponu:** Seri girişten alınan veya gönderilecek 1-byte uzunluğunda veriyi tutar.
2. **Kaydırmalı kaydedici:** Seri giriş tamponundan gelen 1-byte uzunluğundaki veriyi bit dizisi şeklinde diğer cihaza gönderir veya diğer cihazdan alınan bit dizisini seri giriş tamponuna yazar.
3. **Kontrol kaydedicileri:** Spi protokolüyle ilgili yapılandırma ayarlarının yerine getirilmesini sağlar, iki adet kontrol kaydedici bulunmaktadır bunlar:
    - SSPCON(Spi kontrol kaydedici)
    - SSPSTAT(Spi durum kaydedici)

Bu iki kontrol kaydedici içerikleri Tablo 5.1 ve Tablo 5.2 'de sırayla verilmiştir[25]. Tablolarda sadece spi protokolünde kullanılan bit pozisyonları açıklanmıştır, diğer bit pozisyonlarına PIC16F877 mikro denetleyicilerde sadece diğer seri iletişim protokolü türlerinde kullanıldığı için yer verilmemiştir.

Tablo 5.1. SSPCON Kaydedici

Bit pozisyonları	Bit7	Bit6	Bit5	Bit4	Bit3..Bit0
Fonksiyon	-	SSPOV (spi taşma biti) **1:** taşma var **0:** taşma yok	SSPEN (spi aktif etme biti) **1:** aktif **0:** pasif	CKP (spi okuma/yazma seçimi) **1:** yazma **0:** okuma	SSPM (spi modu ve saat böleni) 0001

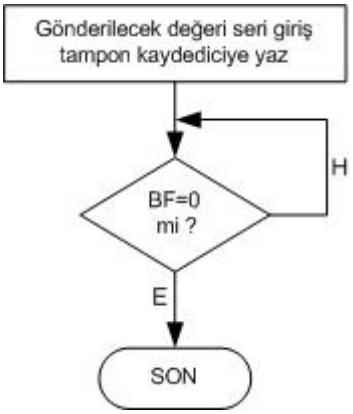
Tablo 5.2. SSPSTAT kaydedici

Bit pozisyonları	Bit7	Bit6	Bit5…Bit1	Bit0
Fonksiyon	SMP (örnekleme biti) **1:** sinyal çıkışında **0:** sinyal ortasında	CKE (saat kenarı) CKP=0 ise **1:** yükselen kenar **0:** düşen kenar (CKP=1 ise tam tersi)	-	BF (seri giriş tampon durumu) **1:** tampon dolu **0:** tampon boş

5.4. Spi Haberleşme İşlemleri

PIC16F877 MCP2510 kontrol ediciyle haberleşirken her defasında 1-byte veri gönderir ya da alır, bu veri bir spi emri, kaydedici adresi yada kaydediciye yazılacak veri olabilir, 1-byte verinin gönderim işlemi SSPSTAT kaydedici BF bitiyle kontrol edilir. Bölüm-3 'te anlatılan mesaj gönderimi, gelen mesajların okunması gibi işlemlerin gerçekleştirilebilmesi için ilgili spi işlemlerini yerine getiren altprogramlar kullanılmıştır. Şekil 5.6'da 1-byte verinin spi aracılığıyla gönderilmesi işlemi akış diyagramı olarak verilmiştir. Şekilde görülen akış diyagramı algoritması şu adımları içerir [22,25].

1. Alt programdan spi 1-byte veri gönderim isteğiyle gelen parametreyi seri giriş tampon kaydediciye yaz.
2. Bu kaydediciye değer yazıldığı anda spi aracılığıyla 1-byte veri gönderim işlemi otomatik olarak aktif olacaktır.
3. Spi durum kaydedici BF bitini kontrol et.
4. BF='0' ise gönderimin bitmesini bekle, '1' ise çık.



Şekil 5.6. Spi aracılığıyla 1-byte veri gönderimi

5.4.1. Spi Aracılığıyla İstenen Kaydediciye Değer Yazma

Mikro denetleyiciden seçilen bir kaydediciye istenen değeri yazılması işlemi şu adımları içerir[22]:

1. CS=0 yaparak kontrol ediciyi spi birimini aktif hale getir.
2. Spi kontrol kaydedici CKP=1 yaparak yazmayı aktif et.
3. Spi yazma emri için altprogramı çağır.
4. Kaydedici adresini göndermek için altprogramı çağır.
5. Kaydediciye yazılacak veriyi göndermek için altprogramı çağır.
6. CS=1 yaparak kontrol edici spi birimini pasif hale getir.

Burada sözü edilen altprogram Şekil 5.6'da gösterilen spi aracılığıyla 1-byte veri gönderim işlemidir.

5.4.2. Spi Aracılığıyla İstenen Kaydedici İçeriğini Okuma

Mikro denetleyiciden seçilen bir kaydedici içeriğinin okunması şu adımları içerir [22]:

1. CS=0 yaparak kontrol ediciyi spi birimini aktif hale getir.
2. Spi okuma emri için altprogramı çağır.
3. Kaydedici adresini göndermek için altprogramı çağır.
4. Spi kontrol kaydedici CKP=0 yaparak okumayı aktif et.
5. Altprogramdan dönen kaydedici içeriğini sakla.
6. Spi kontrol kaydedici CKP=1 yaparak yazmayı aktif et.
7. CS=1 yaparak kontrol edici spi birimini pasif hale getir.

5.4.3. Spi Aracılığıyla Mesaj Gönderim İşleminin Başlatılması

Bölüm 3.6.2'de MCP2510 kontrol edici mesaj gönderim işleminin başlatılması için iki farklı yöntem olduğu anlatılmıştı. Spi aracılığıyla mesaj gönderim işleminin başlatılması için spi mesaj gönderimini başlat emrinin verilmesi yeterlidir, bu işlemin algoritma adımlarını şöyle açıklayabiliriz [22]:

1. CS=0 yaparak kontrol edici spi birimini aktif et.
2. Spi kontrol kaydedici CKP=1 yaparak yazmayı aktif et.
3. spi gönderici tampondaki mesaj gönderimini başlat emrini göndermek için altprogramı çağır.
4. CS=1 yaparak kontrol edici spi birimini pasif hale getir.

MCP2510 kontrol edicilerde bulunan üç adet mesaj gönderim tamponunun her biri için kullanılacak mesaj gönderim işlemini başlat emri farklıdır. Tablo 5.3'te bu emirler yer almaktadır.

Tablo 5.3. Mesaj Gönderim İşlemini Başlatma Spi Emir Kodları

Gönderici tampon	İlgili Spi emri (Hexal sayı düzeninde)
Tampon 0 için mesaj gönderimini başlat	81H
Tampon 1 için mesaj gönderimini başlat	82H
Tampon 2 için mesaj gönderimini başlat	84H

5.4.4. Spi Emir Takımı

Tablo 6.4'te tüm spi işlemleri için gerekli olan emirler verilmiştir [22]. Tabloda yer alan bit değiştir emri yapı olarak diğerlerinden biraz daha farklıdır, üç adet parametresi vardır bunlar sırayla: adres, maske ve filtredir. Adres ile belirtilen kaydedicideki bit değerlerini maske ve filtre alanlarında yer alan değerlere bakarak değiştirir. Maske alanında değeri '0' olan bit pozisyonlarında filtre ve kaydedicide daha önce yer alan bit değerine bakılmaksızın aynı kalır.

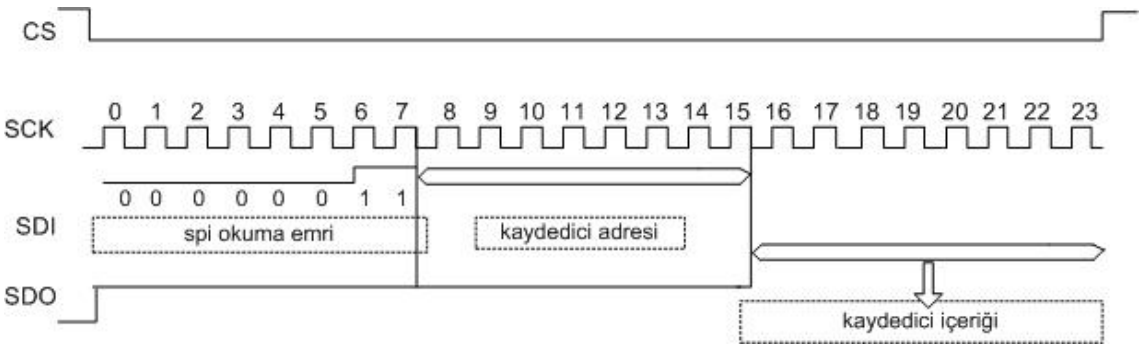
Maske ve filtre alanında '1' olan bit pozisyonlarında kaydedicideki eski bit değeri '0' 'dan '1' 'e değiştirilir. Maske alanında '1' aynı pozisyonda filtre alanında '0' olan durumlarda ise kaydedicideki eski bit değeri '1' 'den '0' 'a değiştirilir.

Tablo 5.4. Spi emir takımı

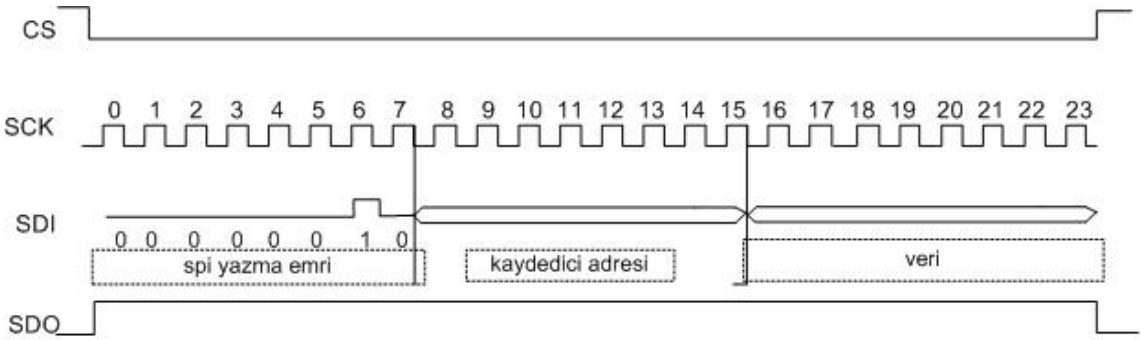
Emir	Spi emri (hexal sayı düzeninde)
Kontrol ediciyi yeniden başlat	C0H
Kaydedici oku	03H
Kaydediciye değer yaz	02H
Mesaj gönderimini başlat	81H, 82H, 84H
Kontrol edici durumunu oku	A0H
Alıcı tampon bellek durumunu oku	B0H
Bit değiştir	05H
Mesaj gönderim tamponu doldur	**40H:** 31H adresinden itibaren işaretçi diziyi yaz **41H:** 36H adresinden itibaren işaretçi diziyi yaz **42H:** 41H adresinden itibaren işaretçi diziyi yaz **43H:** 46H adresinden itibaren işaretçi diziyi yaz **44H:** 51H adresinden itibaren işaretçi diziyi yaz **45H:** 56H adresinden itibaren işaretçi diziyi yaz
Mesaj alım taponu oku	**90H:** 61H adresinden itibaren tampon 0'ı oku **92H:** 66H adresinden itibaren tampon 0'i oku **94H:** 71H adresinden itibaren tampon 1'i oku **96H:** 76H adresinden itibaren tampon 1'i oku

5.4.5. Spi Okuma ve Yazma Zamanlama Diyagramları

Şekil 5.7 ve Şekil 5.8'de sırayla spi kaydedici okuma ve kaydedici yazma işlemleri zamanlama diyagramları verilmiştir [22].



Şekil 5.7. Spi okuma zamanlama diyagramı



Şekil 5.8. Spi yazma zamanlama diyagramı

MCP2510 Kontrol ediciye erişim için birçok farklı spi emri bulunmakla beraber, kontrol ediciye yapılan işlemlerin birçoğu bir kaydediciye veri yazma veya bir kaydediciden veri okuma işlemidir. Şekil 5.7 ve Şekil 5.8'de verilen kaydedici okuma ve kaydediciye yazma işlemleri Bölüm 5.4.1 ve 5.4.2'de algoritmik olarak anlatılmıştı zamanlama diyagramlarında değinilecek bir diğer ayrıntı bu işlemlerin gerçekleştirilirken spi sunucudaki SDO çıkışının kullanımıdır.

Okuma işleminde görüldüğü gibi emir ve kaydedici adresinin gönderimi boyunca SDO çıkışı aktif değildir yani lojik-1 seviyesindedir, spi okuma emri ve kaydedici adresinin gönderiminden hemen sonra lojik-0 seviyesine geçirilerek aktif edilmiştir, bu noktadan sonra MCP2510 kaydedici içeriği spi ara yüzü ile mikro denetleyiciye gönderilir. Bir kaydediciye değer yazma işleminde ise SDO çıkışı sürekli lojik-1 seviyesinde olmalıdır.

6. SONUÇLAR

Akıllı binalar, medikal cihazlar, güvenlik otomasyon sistemleri, veri toplama uygulamaları ve araç içi elektronik aksamların birbirleriyle haberleşmesi gibi endüstriyel uygulamalar veri güvenliğinin ve gerçek zamanlı iletişimin ön planda olduğu uygulamalardır. Ethernet teknolojisi yüksek hızlı olması ve uzun mesafeleri desteklemesine karşın çarpışma tabanlı olduğu için hatasız veri iletişimini gerektiren bu tür uygulamalarda yeterli olmamaktadır.

Araç içi iletişimin örnek olarak verilebileceği çeşitli algılayıcı, kontrol edici, mikro denetleyici ve sayısal-analog dönüştürücülerden oluşan kontrol ağlarında her bir istasyonun ağ üzerindeki diğer istasyonlarla gerçek zamanlı, hızlı ve hatasız haberleşmesi istenir. KAA protokolü çok yöneticili yapısı, üstün hata algılama mekanizması, çarpışmaların öncelik bilgisiyle engellenebildiği, hızlı ve uzun mesafeleri destekleyen endüstriyel uygulamalara yönelik bir haberleşme protokolüdür.

KAA protokolü Ethernet teknolojisi ile kıyaslandığında çarpışma tabanlı olmaması ve hata algılama için 4 farklı hata tespit mekanizmasının paralel çalışması endüstriyel uygulamalarda kullanılması için avantajlı kılmaktadır. Ethernet teknolojisinde aynı anda yola çıkan iki veri paketinin fiziksel iletim yolunda çarpışması dolayısıyla veri kaybı ihtimali bulunmaktadır. KAA protokolünde aynı anda yola çıkan iki mesajdan öncelikli olan iletilir, daha düşük önceliğe sahip mesaj ise veri yolu boş olana kadar bekler, böylece kontrol ağında veri kaybı ihtimali ortadan kaldırılır.

Tez kapsamında tasarlanan deney seti üzerinde birbirleriyle KAA protokolü tabanlı haberleşebilen 3 adet istasyon bulunmaktadır. Deney seti üzerindeki her bir istasyon diğer iki istasyona rasgele bir zamanda veri gönderebilecek yada diğer iki istasyondan veri alabilecek şekilde dizayn edilmiştir. Deneysel kontrol ağı üzerindeki üç istasyon aynı anda çarpışma olmaksızın veri gönderebilmekte, ve diğer istasyonlardan gelen veriyi alabilmektedir. Deney setinin uygun donanımsal ilaveler ve yazılım değişiklikleri yapılarak KAA protokolü tabanlı farklı uygulamalarda kullanılabilmesi de mümkündür. Deney setinde karşılaşılan en büyük sorun ise elektriksel gürültü olmuştur, KAA protokolü 1Mbit'e kadar iletişim hızını desteklemesine rağmen gürültü nedeniyle bu hıza çıkılamamıştır, 250Kbit iletişim hızına kadar sorunsuz haberleşme gerçekleştirilebilmiş fakat daha yüksek hızlara çıkılamamıştır.

KAYNAKLAR

[1] Robert Bosch, 1991, CAN Specification Version 2.0

[2] Cüneyt Bayılmış, İsmail Ertürk, Celal Çeken, 2005, IEEE 802.11 KLAN Kullanarak CAN Segmentlerinin Genişletilmesi İçin Yeni Bir Çözüm, Gazi Üniv.Müh.Mim.Fak.Der., Cilt.20, No.2, sf.197-204.

[3] Gianluca Cena, Adriano Valenzano, 2000, FastCAN: A High-Performance Enhanced CAN-Like Network, IEEE Transaction on Intelligent Transportation System, Vol.47, No.4, sf. 951-963.

[4] Rıfat Çölkesen, Bülent Örencik, 1999, Bilgisayar Haberleşmesi ve Ağ Teknolojileri, Papatya Yayıncılık, 393 sf.

[5] Hikmet Şahin, Ayhan Dayanık, Caner Altınbaşak, 2006, PIC Programlama Teknikleri ve PIC16F877A, Altaş Yayıncılık, 526 sf.

[6] M Farsi, K. Ratcliff, Manuel Barbosa, 1999, An overview of Controller Area Network, 1999, Computing & Control Engineering Journal June 1999, sf.113-120.

[7] Andrew Hopkins, 2003, Controller Area Networks, Computer Science Seminar

[8] Ahmet Berkay, Murat Şeker, E.Murat Esin, 2003, Kontrolör Alan Ağı İçin Delphi Programı Aracılığı İle Geliştirilecek Olan Yazılımlarda Kullanılmak Üzere Hazırlanan Delphi Bileşenleri, Elk-Elktronik-Bilg.Müh.10.Ulusal Kongresi, sf.476-479.

[9] Nick Papadoglou, Elias Stipidis, 2001, Investigation for a Global AVL System, IEEE Transaction on Intelligent Transportation System, Vol.2, No.3, sf. 121-126.

[10] Nick Papadoglou, Elias Stipidis, 1999, Short message service link for automatic vehicle location reporting, IEEE Electronic Letters, Vol.35, No.11, sf.876-877.

[11] Valentinos K. Kongezos, 2002, Wriless communication between A.G.V.'s(Autonomous Guided Vehicle) and the industrial network CAN, International Conference on Robotics&Automation , sf.434-437.

[12] Hong-Hee Lee, Ui-Horn Jeong, 2000, A Study on Speed Synchronization for Multi-Motors Using Controller Area Network, IEEE 2000, sf.234-239.

[13] Teoh Chee Hooi, Manjit Singh, Yap Keem Siah, Abdul Rahim bin Ahmad, 2001, Building Low-Cost Intelligent Building Components with Controller Area Network Bus, IEEE 2001, sf.466-468.

[14] Kyung Chang Lee, Hong-Hee Lee, 2004, Network-based Fire-Detection System via Controller Area Network for Smart Home Automation, Transaction on Intelligent Transportation System, Vol.50, No.4, sf.1093-1100.

[15] Jose C.Metrolho, Carlos A. C. M. Couto, 1999, Carlos M. J. A. Serodio, Can Based Actuation System For Greenhouse Control, ISIE 1999, sf.945-950

[16] A. Valera, J.Salt, V.Casanova, S.Ferrus, 1999, Control of Industrial Robot With a FieldBus, IEEE 1999, sf.1235-1241.

[17] M. Wargui, A. Rachid, 1996, Application of Controller Area Network to Mobile Robots, IEEE 1996, sf.205-207.

[18] H. Ekiz, A. Kutlu, M.D. Baba, E.T. Powner, 1996, Performance Analysis of a CAN/CAN Bridge, IEEE 1996, sf.181-188.

[19] H. Ekiz, A. Kutlu, E.T. Powner, 1996, Design and Implementation of a CAN/CAN Bridge, IEEE 1996, sf.507-513.

[20] Dieter Bühler, Wolfgang Küchlin, 2000, Remote Fieldbus System Management with Java and XML, ISIE 2000.

[21] Dieter Bühler, 2000, The CANopen Markup Language-Representing Fieldbus Data with XML, IEEE 2000, sf.2449-2454.

[22] Microchip MCP2510 Stand Alone CAN Controller With Spi Intferface, 2002

[23] Philips Sja1000 Stand Alone CAN Controller Application Note, 1997

[24] Microchip MCP2551 High-Speed CAN Transceiver, 2003

[25] Microchip PIC16F87X Data Sheet, 2001

ÖZGEÇMİŞ

1979 yılında Elazığ'da doğdu. İlk, orta ve lise öğrenimi Elazığ'da tamamladı. 1996 yılında girdiği K.T.Ü. Bilgisayar Mühendisliği Bölümünü 2001 yılında bitirdi. 2004 yılında F.Ü. Bilgisayar Mühendisliği Donanım Ana Bilim Dalında yüksek lisansa başladı. 2001 yılında F.Ü. Kemaliye H.Ali Akın Meslek Yüksekokuluna Okutman olarak atandı. Halen bu görevine devam etmektedir.

Yunus SANTUR