T.C.
TRAKYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

OTOMOBİL BİLGİ TOPLAMA KUTUSU:BİR MOBİL VE GÖMÜLÜ SİSTEMLER UYGULAMASI

Hasan AYTIŞ

YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

Tez Yöneticisi: Yrd.Doç.Dr. Erdem UÇAR

2010
EDİRNE

T.C

TRAKYA ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

OTOMOBİL BİLGİ TOPLAMA KUTUSU: BİR MOBİL VE GÖMÜLÜ SİSTEMLER UYGULAMASI

HASAN AYTIŞ

YÜKSEK LİSANS TEZİ

BİLGİSAYAR MÜHENDİSLİĞİ ANA BİLİM DALI

YÖNETİCİ YRD. DOÇ. DR ERDEM UÇAR

2010 EDİRNE

T.C

TRAKYA ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

OTOMOBİL BİLGİ TOPLAMA KUTUSU: BİR MOBİL VE GÖMÜLÜ SİSTEMLER UYGULAMASI

HASAN AYTIŞ

YÜKSEK LİSANS TEZİ

BİLGİSAYAR MÜHENDİSLİĞİ ANA BİLİM DALI

Bu tez 17/ 06/2010 tarihinde Aşağıdaki Jüri Tarafından Kabul Edilmiştir

Yrd.Doç.Dr.Erdem UÇAR	Doç.Dr.Yılmaz KILIÇASLAN	Doç.Dr. Yılmaz ÇAN
Tez Yöneticisi	ÜYE	ÜYE

ÖZET

Yüksek Lisans Tezi

Otomobil Bilgi Toplama Kutusu: Bir Mobil ve Gömülü Sistemler Uygulaması

Trakya Üniversitesi

Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Ana Bilim Dalı

Danışman: Yrd. Doç. Dr. Erdem UÇAR

Gömülü sistemler için tasarlanmış ve gerçek zamanlı işletim sistemi (RTOS) olan Windows CE 6.0 detaylı olarak incelenmiştir. İşletim sisteminin yetenekleri ve çalışma şekli araştırılıp bu çalışmaya eklenmiştir. İşletim sistemi oluşturma ve derleme işlemleri örnek uygulama geliştirme yöntemlerine yer verilmiştir. İşletim sistemi oluşturken amaca uygun donanım seçmenin çok önemli olduğunu tespit ettik Aksi takdirde işletim sistemi büyüklüğü gereksiz yere fazla olmakta ve daha fazla hafızaya ihtiyacımız olmaktadır. Bu durum da iş yapma maliyetlerini artırmaktadır.

Windows CE 6.0' da yazılım geliştirme iki farklı dil ile gerçekleşmektedir. C++(Native) ve C#.NET (Managed) de uygulama geliştirilebilmektedir. Uygulamamız C#.Net ile geliştirilmiştir. C#.Net yönetilebilir bir dil olduğu için C++ a göre gerçek zamanlı yanıt verme süresi pek tatmin edici olmadı. Uygulamamızın hayati bir önemi yok ise C#.NET ile geliştirebiliriz. Fakat hayati bir önemi var ise doğal dil olan (Native) C++ da uygulama geliştirmeliyiz.

Gerçek ortamdan bilgi toplayabilmemiz ve değişkenlerin yazılımsal olarak işlenebileceği formata dönüştürmek için elektronik donımlardan yararlanılmıştır. Gerçek ortamı simule edecek bir model, Microchip ailesinden PIC 16F876 ile elektronik bir devre yapılarak sağlanmıştır. Elektronik devrede, 4 kanal analog sinyal giriş,6 kanal dijital sinyal giriş, 2 kanal dijital sinyal çıkış şeklinde tasarlanmıştır. İşletim sistemi ve elektronik devre ile haberleşme seri port üzerinden yapılmıştır. Simulator durumu soru cevap şeklinde yapılmıştır.

2010, 62 sayfa
Anahtar Kelimeler: Gömülü sistemler, gerçek zamanlı işletim sistemi, Windows CE 6.0, C#.Net,Otomobil Bilgi Toplam

ABSTRACT

Master Thesis

Automotive Data Collection Box : A Mobile and Embedded Systems Application

Trakya University

Institute of Science

Computer Engineering Department

Supervisor: Asst. Assoc. Dr. Erdem UÇAR

Windows Embedded CE 6.0, being designed for embedded systems and being a real-time operating system (RTOS), is examined in detail. The capabilities of the operating system and the way it works are investigated and the results are reported in the thesis. Operating system development methods are illustrated with examples of creation and compilation. We have seen that it is very important to employ goal-oriented hardware when developing an operating system. Otherwise , thr operating system becomes unnecessarily large and its requires more memory. This also increases the cost of doing job.

Software is developed with two different languages in Windows CE 6.0. These are C++ (Native) and C# (Managed). We have developed and implemented our application in C# .Net. Since C#.Net is a managed language, it's real-time response time was less than satisfactory compared to a response time to be achieved with C++. If an application is not of vital importance, it can be developed with C#.Net. However, in case the application is vitally important, it needs to be developed with natural development language like (native) C++.

We used and 2 output channel for digital signal processing. The operating system and the circuit electronic hardware to collect information from the real environment and to convert variables to a format that can be handled by the software. We developed a model for simulating the real environment using PIC16F876 from the Microchip family.The electronic circuit contains 4 input channels for analog signal processing, 6 input channel for digital signal processing comminicate via serial port.The simulator works in a question-answer paradigm.

2010, 62 pages
Key words: Embedded systems,Automotive, real-time operating system, Windows CE 6.0, C #. Net, Otomotive datalogger.

ÖNSÖZ

Çalışmalarımı yönlendiren, araştırmalarımın her aşamasında bilgi, öneri ve yardımlarını esirgemeyerek akademik ortamda olduğu kadar beşeri ilişkilerde de engin fikirleriyle yetişme ve gelişmeme katkıda bulunan danışman hocam Sayın Yrd.Doç.Dr.Erdem UÇAR'a, manevi desteklerini ve yardımlarını esirgemeyen Sayın Doç Dr. Yılmaz KILIÇASLAN'a, gömülü sistemler ile ilgili bilgi ve tecrübelerini benle paylaşan Çizgi Mühendislik Sayın Hasan AKGİL'e, Windows CE 6.0 işletim sistemi ile ilgili olarak özel vermiş oldukları seminerlere davet ederek gerektiğinde her hangi bir sorunumda ürün sorumluları ile irtibata geçmemi sağlayarak yardımcı olan Microsoft Türkiye Akademik'den Sayın Mehmet TUNÇKANAT'a ve Sayın Mehmet EMRE'ye, çalışmalarım süresince birçok fedakarlıklar göstererek beni destekleyen ailem ve eşime en derin duygularla teşekkür ederim.

İÇİNDEKİLER DİZİNİ

ÖZET iv

ABSTRACT v

ÖNSÖZ vi

İÇİNDEKİLER DİZİNİ vii

ŞEKİLLER DİZİNİ x

TABLOLAR DİZİNİ xii

1. GİRİŞ 1

2. GÖMÜLÜ SİSTEMLER 2

2.1. Gömülü Sistemler İçin Örnekler 3

2.2. Gerçek Zamanlı İşletim Sistemleri 6

2.3. Gömülü Sistemler İçin İşletim Sistemleri 8

2.4. Gömülü Sistem Tasarımında Kullanılan İşlemciler ve Yazılımlar 9

3. eBox2300 GÖMÜLÜ BİLGİSAYAR SİSTEMİ 16

3.1. eBox 2300 Mimari Yapısı 17

3.2. eBox'ın Boot (başlatılması) Edilmesi 19

4. PLATFORM BUILDER VE İŞLETİM SİSTEMİ DERLEME 22

4.1. Platform Builder Terimleri 24

4.2. CE İşletim Sisteminin Derlenme Süreci 25

4.3. Özelleştirilmiş CE 6.0 Çekirdeğinin Derlenmesi 27

4.3.1. Windows Embedded CE 6.0 Derleme/Hata Ayıklama 28
4.3.2. Hazırlıklar 28
4.4. Ebox BSP kullanarak yeni OS Image Kernel İnşa Etmek 29
4.4.1. OSDesign isteğimize göre değiştirmek. 31
4.4.2. Opsiyonel kurulumlar: 33
4.4.3. Configuration Manager ve Debug & Release Build 33
4.4.4. Projenin derleme opsiyonlarının ayarlanması 34
4.4.5. OS Ortam Değişkenlerinin Ayarlanması 36
4.4.6. Yeni OS Kernel 'inin Ebox 2300 e Download Ettirilmesi 37
4.4.7. eBox için ağ kurulumu 38
4.4.8. Target Device Bağlantı Ayarları 38
4.4.9. Image'ın eBox2300 e Download Edilmesi 41
5. SIMULATOR TASARIMI 43
5.1. Microchip PIC 18F876 ile Analog Dijital Sinyal Üretici Devresi (Simulator) 43
5.1.1. P I C 16F876 Çekirdek Mimarisi 43
5.1.2. Çevre Birim Özellikleri 44
5.1.3. MAX232 entegresi 45
5.1.4. Sinyal Simulator Devre Şeması 46
5.2. C#.NET Bilgi Toplama Programı 52
5.2.1. Simulator ile haberleşme uygulaması 52
5.2.2. Uygulamanın çalışması 54
5.2.3. Web Servis uygulaması 54
5.2.4. Gösterge paneli 55
6. LİTERATÜR TARAMASI 57
6.1. Fiat ecoDrive 57
6.2. Ford SYNC 58
6.3. Kia UVO 60

7. SONUÇLAR 61

7.1. Sonuçlar 61

KAYNAKLAR 62

ŞEKİLLER DİZİNİ

Şekil No **Şekil Açıklaması** **Sayfa No**

Şekil 1.1 Gömülü sistem içeren robot. Fotograf NASA/JPL-CALTECH. 3
Şekil 1.2 Cep telefonları, PDA'lar ve Multimedya oyun konsolları tüm sistemler. Fotograflar (Motorola, Microsoft ve Philips Sonicare) 4
Şekil 1.3 Gerçek-zamanlı sistem harici girişler yanıt vermek zorunda olduğu ve zaman sınırlı bir miktarda veya sisteme yeni çıkışlar üretmek başarısız olur. Yanıt süreleri genelde yaklaşık 0.5ms ile 10ms arasındadır. 6
Şekil 1.4 Bit işlemci boyutuna göre kullanılan yeni gömülü sistem tasarımları 9
Şekil 1.5 32 ve 64-bit işlemci 2002 yılı için yıllık satış verileri 10
Şekil 1.6 İşlemci sayılarına göre gömülü sistemler dağılımı 11
Şekil 1.8 Gömülü sistemlerde kullanılan ticari işletim sistemleri dağılımı. 13
Şekil 1.9 Gömülü Sistemlerde kullanılan uygulamaların geliştirilme Dillerine göre dağılımı 14
Şekil 2.1. EBox 2300 MSJK bir Windows Embedded CE düşük maliyetli X86 SoC tabanlı gömülü bir bilgisayar sistemidir. Bir dizi Giriş/Çıkış özelliklerine sahiptir. Fotoğraf ICOP Corporation'dan alınmıştır. 16
Şekil 2.2. eBox 2300 arka panel görünümü Giriş / Çıkış konektörler ve güç düğmesi 17
Şekil 2.3 eBox 2300 iç kısmı 18
Şekil 2.4.Blok EBox 2.300 Gömülü Bilgisayar Sistemi blok şeması 19
Şekil 2.6. Windows Embedded CE 6.0. ın başarılı olarak açıldıktan sonraki ekran görüntüsü 20
Şekil 3.1. Platform Builder yeni İşletim sistemi çekirdeği derlemek içn kullanılan bir araçtır. 23
Şekil 3.2 İşletim sistemi çalıştırılabilir image oluşturma süreci. 25
Şekil 3.3. CE Derleme diagramı 26
Şekil 3.4 Yeni OS Tasarım Projesi Oluşturma 29
Şekil 3.5 Catalog Items View (sol column) 32
Şekil 3.6 Configuration Manager ‘ı Debug Mode ve Release Mode durumunu seçmek için kullanıyoruz 34
Şekil 3.7 OS Property Sayfası 35
Şekil 3.8 Property Page Kullanılarak ortam değişkenlerinin eklenmesi 36
Şekil 3.9 Target Device Bağlantı Ayarları Penceresi 39
Şekil 3.10 Ethernet Download Settings penceresi 40

Şekil3.11 Target cihaz image download ediyor 41
Şekil 3.12 Yeni Image download bitti. 42
Şekil 5.1 PIC 16F876 Bacak Bağlantı Şeması 44
Şekil 5.2 MAX232 Entegresinin PIC ve COMPORT ile bağlantı şeması 46
Şekil 5.3. Simulator Devre Şeması 47
Şekil 5.4 Simulator baskı devre şeması 48
Şekil 5.5 Devrenin protoboardda test edilmesi. 48
Şekil 5.6. Baskı devre şemasının bakır plaketi 49
Şekil 5.7 Devreyi sabitleyecek kutunun montaj aşaması 49
Şekil 5.8 Devrenin kutu montajı ilk enerji verilmesi 50
Şekil 5.9 Simulator Terminal hata mesajı ve komutların listesi. 51
Şekil 5.10 “a” komutu verildikten sonra simulatorun cevap çıktısı. 51
Şekil 5.11 eBox2300 de çalıştırılan uygulamanın geliştirilme aşaması 53
Şekil 5.13 Sistemin akış dagramı 56
Şekil.6.1. Fiat ecoDrive ekran görüntüs 58
Şekil 6.2. Ford SYNC 59
Şekil 6.3. Kia UVO 60

TABLOLAR DİZİNİ

Tablo No	Tablo Açıklaması	Sayfa No
Tablo 1.1	Gömülü sistemler için örnekler	5
Tablo 2.1	eBox 2300 CE 6.0 için cihaz sürücüleri	21
Tablo 5.1	PIC16F87x ailesi karşılaştırılmalı özellikleri tablosu	45

1. GİRİŞ

Otomobiller için geliştirilecek olan bu projede otomobildeki tüm sensörlerden gelen sinyallerin bir arabirim aracılığı ile gömülü sisteme iletilmesi sağlanacaktır. İletilen tüm bu veriler gömülü sistem üzerinde işlenerek merkezdeki sunucuya internet üzerinden gönderilecektir. Elde edilen bu veriler ile araç ile ilgili teknik sorunların giderilmesinde yardımcı olacak aynı zamanda kullanıcı profili ortaya çıkartılacaktır. Elde edilen veriler yardımıyla otomobil üreticileri ülkemizdeki kullanıcı profillerine göre bir sonraki ürünlerinde geliştirmeler yapabileceklerdir. Sistemin başka bir amacı ise aracın her hangi bir kaza/arıza durumunda bir önceki durum bilgileri elimizde olacağından kaza/arıza oluşmadan önceki hatalari tespit etmemize yardımcı olacaktır.

Gömülü sistemler üzerinde geliştirilecek olan bu proje aslında dışardan veri toplayıp karar verebilmesi istenen tüm sistemler için kullanılabilir. Sadece otomobiller için değil aynı zamanda sahadan veri toplanması gereken araştırmalarda, yada bir işletmedeki kullanılan cihazlardan veri toplanması için kullanılabilir.

Proje üç farklı çalışmadan oluşmaktadır. Birinci çalışma ile işletim sisteminin amaca uygun olarak özelleştirilmesi, derlenmesi ve eBox2300 gömülü bilgisayar sisteminde işletim sisteminin çalıştırılması gerçekleştirilmiştir. İkinci çalışma ile dışardan gelecek olan verilerin gömülü sisteme iletecek olan elektronik arabirim olarak otomobildeki sensörlerin vereceği sinyal özelliklerine göre dijital ve analog sinyal üreten aynı zamanda seri port üzerinden bilgi gönderip alabilen bir simulator devresi hazırlanmıştır. Üçüncü çalışma, arabirimden gömülü sisteme gelen bilgileri işleyip sunucuya gönderecek ve gömülü sistem üzerinde çalışan c#.net uygulaması geliştirilmiştirdir. Sunucuya gelen sistem ile ilgili geçmiş ve anlık bilgilerin görüntülenip analiz edilmesine yardımcı olacak web tabanlı gösterge paneli ve geçmiş dataların filtrelenip sonuçlarını gösterecek bir web tabanlı raporlama programın hazırlanmıştır.

2. GÖMÜLÜ SİSTEMLER

Gömülü sistem kendi içinde bir bilgisayar içeren herhangi bir elektronik cihazdır. Gömülü sistem kullanıcısı çoğu zaman sistemin içinde bir bilgisayar cihazının mevcut olduğunun farkında değildir. Sistem tasarımını basitleştirmek ve esneklik kazanmak için gömülü sistemlerde bilgisayarlar kullanılmaktadır.

Kişisel bilgisayarlar gömülü sistemlerin aksine program kodu genellikle ROM ve sabit bir disk üzerinde saklanmaktadır. Tipik olarak son kullanıcı gömülü cihaz için herhangi bir yazılım geliştirmesine gerek olmamaktadır. VLSI[1] teknolojisinin ilerlemesi ile gömülü sistemler bugünün elektronik cihazlarının çoğunda bulunarak ucuz hale gelmiştir.[3]

[1] VLSI: very large scale integration uzun adı ile pek çok elektronik parçanın tek bir chip içine yerleştirilmesine verilen isim.

2.1.Gömülü Sistemler İçin Örnekler

Resim 1.1'de NASA tarafından gömülü sistem kullanılarak geliştirilmiş olan robot resmi görülmektedir. Cep telefonları, PDA veya multimedya oynatıcıları birer gömülü sistemlerdir. Resim 1.2'de görüldüğü gibi elektrikli diş fırçasında gömülü sistem kullanılmıştır. Diş fırçası küçük bir programlanabilir mikroişlemci ile hız kontrolü ve şarj göstergesi ile pil durumunu kontrol eder. Yüksek model otomobillerde yaklaşık yüz gömülü mikroişlemci içermektedir. Tipik orta sınıf bir evde bulunan cihazların ellisinde gömülü sistem içermektedir. Tüm söylenenleri göze alarak dünyada üretilen işlemcilerin çoğu gömülü sistemler için kullanılmaktadır.[1]



Şekil 1.1 Gömülü sistem içeren robot. Fotograf NASA/JPL-CALTECH.



Şekil 1.2 Cep telefonları, PDA'lar ve Multimedya oyun konsolları tüm sistemler. Fotograflar (Motorola, Microsoft ve Philips Sonicare)

Gömülü sistem içeren cihazların geniş bir yelpazede kullanıldığı tablo 1.1'de görülmektedir. Gömülü sistemler, uçakta ve askeri sistemlerde, biyomedikal sistemler, otomobiller, iletişim, giriş/çıkış bilgisayar cihazları, elektronik enstrumanlar, ev elektroniği, endüstriyel ekipmanlar, ofis makineleri, kişisel cihazlar, robotlar, akıllı oyunaklar gibi ürünlerde kullanılabilmektedir.

Gömülü sistemler güvenilir olmalıdır. Birçok gömülü cihazda çökme (sistem çökmesi) olmamaktadır ve yeniden başlatmanız mümkün olmayabilmektedir. Birçok gömülü cihazda yazılımı güncellenmemektedir. Birçok cihazda kritik performans ve güç tasarım kısıtlamaları bulunmaktadır. Birçok cihazda bellek ve işlemci gücü sınırlı olmaktadır. Bu özellikleri göze alarak geliştiricilecek olan gömülü sistem, tasarım aşamasında çok dikkatli ve her aşama tekrar tekrar kontrol edilerek tasarlanmaktadır.

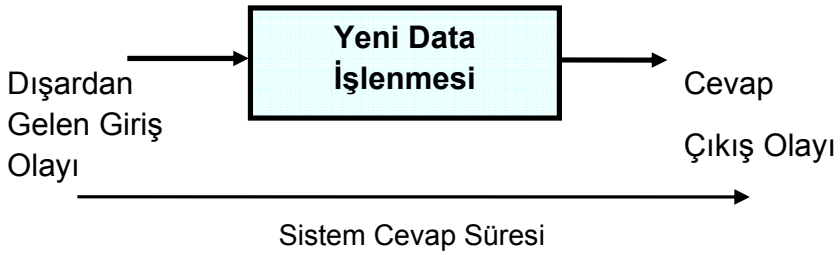
Tablo 1.1 Gömülü sistemler için örnekler

Uçak ve Askeri Sistemleri	Uçak otopilot, navigasyon sistemleri, otomatik iniş sistemleri, rehberlik sistemleri, motor kontrol eder.
Biyomedikal Sistem	CAT tarama ve görüntüleme sistemleri, hasta monitörleri, kalp Pacers Ultrason
Arabalar	Motor kontrol, anti-fren sistemi, çekiş kontrol sistemleri, hava yastığı kontrol, ısıtma ve klima kontrolleri, GPS harita.
Haberleşme	Haberleşme uyduları, ağ router, switch, hub
Tüketici Elektroniği	TV, fırınlar, bulaşık makineleri, DVD çalarlar, stereo, güvenlik sistemleri, çim sulama makinesi kontroller, termostatlar vb.
Bilgisayar I / O Cihazlar	Klavye, fare, yazıcılar, tarayıcılar, görüntüler, modemler, sabit disk sürücüleri, DVD sürücüleri, ekran kartı, USB aygıtları.
Elektronik Enstrümantasyon	Veri toplama sistemleri, Osiloskoplar, voltmetre, sinyal jeneratörleri, mantık analizörleri.
Endüstriyel Ekipmanlar	Asansör denetimleri, gözetim sistemleri, robotlar, CNC makineleri, Programlanabilir Lojik Kontrolörler, endüstriyel otomasyon ve kontrol sistemleri.
Büro Makineleri.	Faks makineleri, fotokopi makinesi, telefon, hesap makineleri, kasiyer.
Kişisel Cihazlar	Cep telefonları, taşınabilir MP3 çalarlar, video oynatıcılar, kişisel dijital yardımcılar (PDA), elektronik kol saatleri, el video oyunları, dijital kameralar, GPS sistemleri.
Robotlar	Endüstriyel robotlar, otonom araçlar, uzay araştırma robotları (yani Mars robotları)
Oyuncaklar	Video oyun sistemleri, "Aibo", "Furby", ve "Elmo" tipi robot oyuncaklar.

2.2.Gerçek Zamanlı İşletim Sistemleri

Gerçek zamanlı sistemler, izin verilen sınırlı zaman içinde harici girişlere yeni çıkışlar üretmek zorundadır. Temel olarak yapısı şekil 1.3'de görülmektedir. Sistemin tepki süresinin sınırlı olması gerekmektedir. Yanıt sürelerinin uzun olması gerçek zamanlı işletim sistemlerinin başarısız olmasına neden olmaktadır.

Otomobillerimizdeki hava yastığı denetleyicisi gerçek zamanlı bir sisteme örnek olarak gösterebiliriz. Hava yastığı hareket sensörü bir çarpışma tespit ettiğinde, sistem en fazla 10ms içinde cevap verip hava yastığını dağıtması gerekir. Aksi halde süre uzun olursa sistem başarısız olur. Yüksek hızlarda herhangi bir gecikme sürücünün bir çarpışma anında hava yastığından önce direksiyona çarpmasına sebep olur.[1]



Şekil 1.3 Gerçek-zamanlı sistem harici girişler yanıt vermek zorunda olduğu ve zaman sınırlı bir miktarda veya sisteme yeni çıkışlar üretmek başarısız olur. Yanıt süreleri genelde yaklaşık 0.5ms ile 10ms arasındadır.

Yazılımsal gerçek zamanlı sistemde, kritik görevlerin önceliği olmaktadır. Yazılımsal gerçek zamanlı sistem normal olarak gerçek zaman tepki kısıtlamasıyla karşılaşmaktadır. Multimedya oynatıcıları tipik olarak bir yazılımsal gerçek zamanlı sistem örneğidir. Bazen oynatıcı video karesi veya ses örneğini atlayabilir. Fakat bizler çoğu kez bu durumun farkına bile varamayız.

Donanımsal gerçek zamanlı sistemde, sistem hesaplanan belirli bir süre içinde yeni bir çıkış tepkisi vermektedir. Eğer bunu başaramazsa sistem başarısız olmaktadır. Donanımsal gerçek zamanlı sistem için örnek olarak, bilgisayar kontrollü uçak uçuş kontrol sistemini (fly-by-wire) ele alabiliriz. Uçuş kontrol sisteminde pilot, sistemin boyunduruğu altında geçtiği zaman, uçuş kontrol yüzeyleri hızlı bir şekilde cevap vermek zorunda kalmaktadır. Aksi takdirde sistem karasız duruma geçmekte ve sistem çökmesi yaşanmaktadır. Federal Havacılık Kurumu, kesinlikle bu sistemleri test edip gerçek zaman cevap sürelerine göre sistemlere sertifika vererek belgelendirmektedir. [1]

Sanal bellekde sayfa değiştirme ve garbage collection rutinleri her zaman nesneye yönelik diller ile çalışmakda ve bu rutinler donanımsal gerçek zamanlı sistemlerde sorun oluşturmaktadır. Pek çok gömülü sistemler birkaç giriş çıkışı olan gerçek zamanlı sistemlerdir. Birden fazla bağımsız olarak meydana gelen olaylar bulumaktadır. Programlama, görevleri ayırarak basitleştirilmektedir, fakat CPU farklı görevler arasında ileri geri geçiş yapması gerekmektedir. İşletim sistemi çoklu görev işlemeyi destekliyorsa CPU çoklu işlem süresince zaman dilimlemesi yapmaktadır. CPU senkronizasyan ilkelerine uymak için farklı görevleri paralel çalıştırarak faaliyetleri koordine etmektedir.

İşletim sistemleri gerçek-zaman karakteristiğine göre sınıflandırılmaktadır. Gerçek zamanlı işletim sistemi, gerçek zamanlı uygulamaları çalıştırmak için dikkatli olarak tasarlanmaktadır. Geçek zamanlı uygulamaların %95 i, sistemin cevap verme süresini 0.5ms ile 10ms arasında olmasını istemektedir. Sadece %10 tolerans kabul etmektedir. Bu ölçülere göre genel kullanım için kullanılan işletim sistemleri gerçek zamanlı işletim sistemi olmamaktadır. Bu ölçülere göre Windows Embedded CE gerçek zamanlı işletim sistemi kategorisine girmektedir.[4]

2.3.Gömülü Sistemler İçin İşletim Sistemleri

Yeni cihazların çoğu, çoklu işlem, işlemlerin senkronizasyonu, geniş giriş/çıkış aralığına destek, zamanlama ve giriş/çıkış işlemlerinin tamponlanması, hafıza yönetimi, grafik ekran desteği, dosya sistemi, ağ, güvenlik ve güç yönetimi desteğine ihtiyaç duyduğundan karmaşık yazılımlara sahiptirler. İşetim sistemi uygulama geliştiricilere yardımcı olabilmek için bu özelliklerin hepsine sahip olması gerekir.

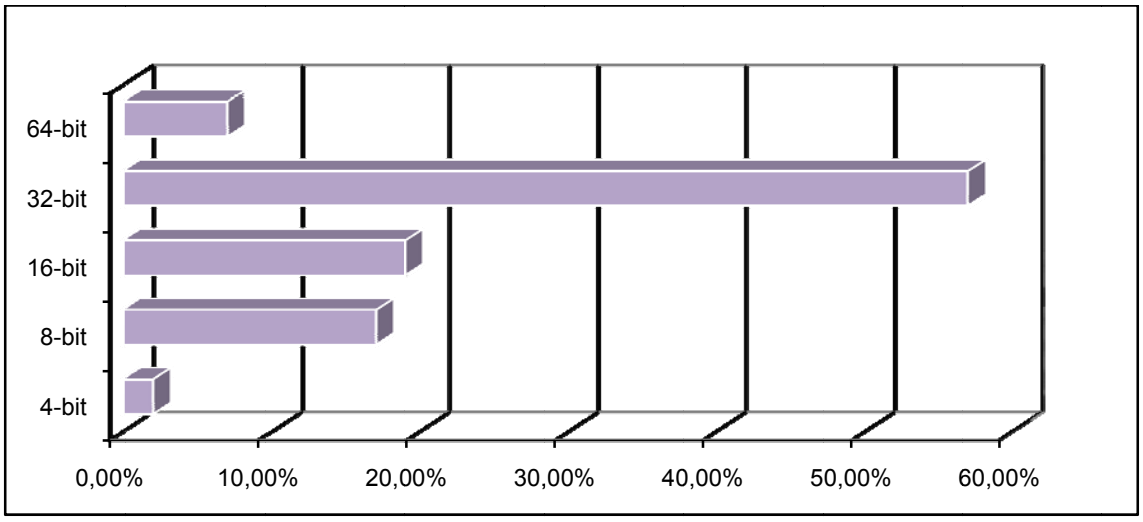
Geçtiğmiz günlerde yayınlanan bir cep telefonu tasarım kodu 5 milyon satır içeriyordu. Eğer bir proje için gerekli zaman ve finans sorunu yok ise problem değildir. Bu örnek de olduğu gibi var olan bir işletim sistemini kullanmak ekonomik olarak büyük bir etki yaratmaktadır. Böylece lisans ücreti ödeyerek daha pahalı olan geliştrme maliyetini düşürülmüş olmaktadır. Tipik olarak ticari gömülü işletim sistemi lisans ücretleri cihaz başına sadece birkaç dolardır. Bazı çok basit cihazlar işletim sistemine ihtiyaç duymayabilir fakat yeni cıkan cihazlar gittikçe daha çok karmaşıklaşmaktadır.[1],[2]

Bu sebeplerden dolayı birçok gömülü cihazlar gömülü bir işletim sistemi kullanmaktadır. Gömülü işletim sistemleri çoğunlukla C/C++'da geliştirilmektedir ve C/C++ derleyicisi, assambler ve debuggin araçaları da içinde gelir ve bu araçlar uygulama geliştirme ve test işlemlerinde programcıya yardımcı olmaktadır.

2.4.Gömülü Sistem Tasarımında Kullanılan İşlemciler ve Yazılımlar

Popüler ticaret yayınları tarafından piyasaya çıkmış gömülü sistemlerde çalışan tasarımcılarla yapılan anketler yıllık olarak yayınlanmaktadır. Bu araştırmalar piyasadaki genel eğilimleri, ilginç ve yeni gömülü sistemlerdeki gelişmeleri ortaya çıkarmakta yardımcı olmaktadır.[5],[6]

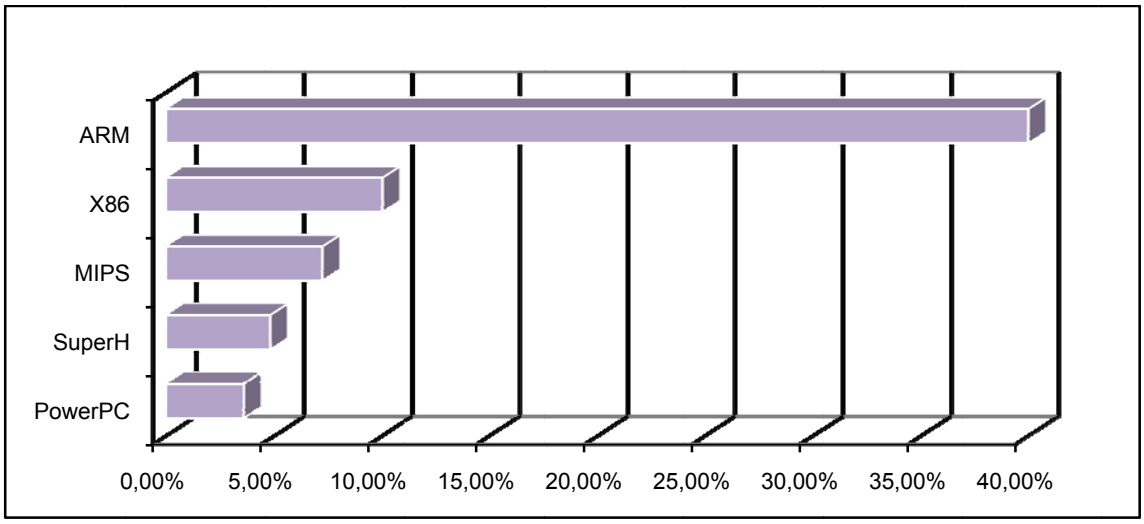
Şekil 1.4 de şu an kullanılmakta olan işlemcilerin bit büyüklüğüne göre kullanımını göstermektedir. En güncel tasarımlar 32-bitlik işlemci etrafında dönmektedir.[1]



Şekil 1.4 Bit işlemci boyutuna göre kullanılan yeni gömülü sistem tasarımları

Şekil 1.5'de yıllık olarak 32-bit ve 64-bit mikroişlemcilerinin 2002 yılına ait satış verilerini göstermektedir. En büyük pazar payına ARM ve X86 tabanlı işlemciler diğer RISC tabanlı işlemciler, MIPS, SuperH ve PowerPC takip etmektedir. Bu veriler kamuya açıklanmamıştır. Ama geçerli dağılım ile benzerlik göstermektedir. ARM işlemci ingiliz IP firması olan ARM LTD (Advanced RISC Machines) tarafında düşük güçlü RISC işlemcisi olarak tasarlanmıştır.

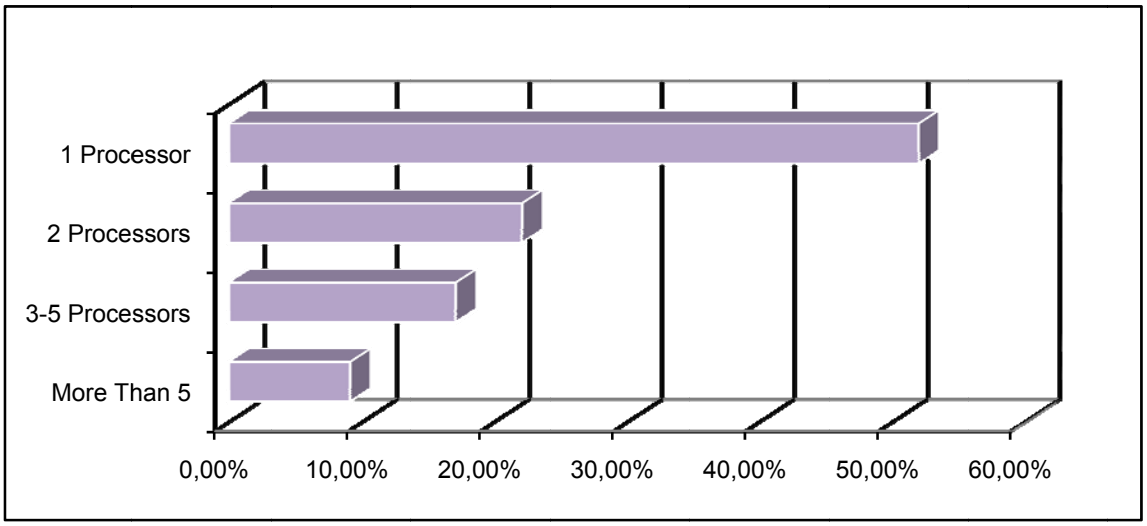
Cep telefonları, iPod, Nano, kameralar, cep oyunları, HDTV ler ve set üstü kutularda ARM işlemciler kullanılmaktadır. ARM işlemcilerin %80'i cep telefonlarında kullanılmaktadır. Performans/güç oranı düşük olması pil ile çalışan cihazlarda daha çok tercih edilme sebebidir. Bu küçük cihazlar bellek maliyetini azaltmaya yardımcı olurlar.



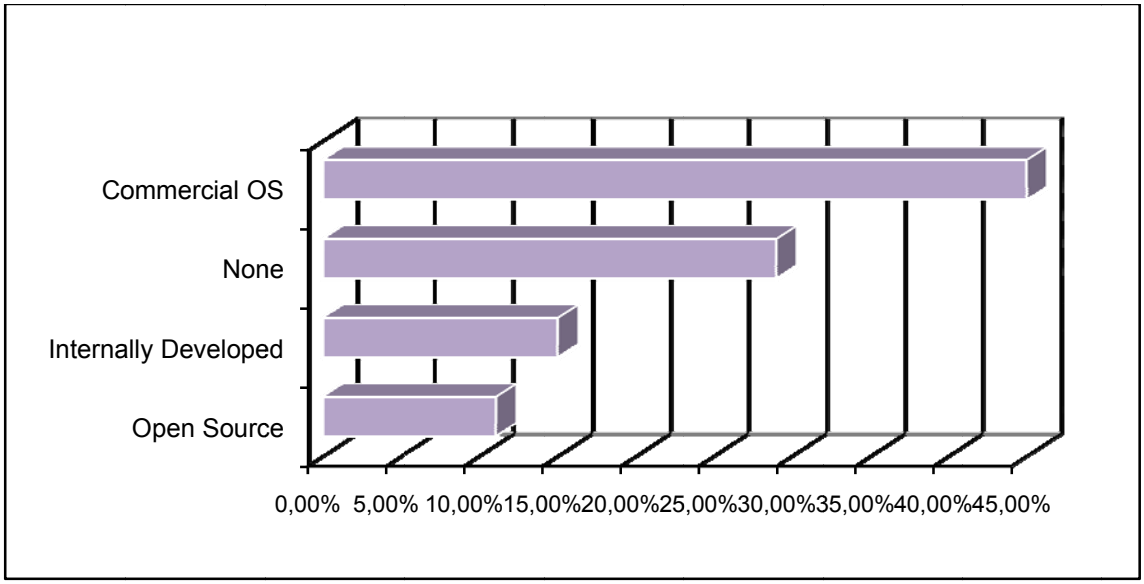
Şekil 1.5 32 ve 64-bit işlemci 2002 yılı için yıllık satış verileri

X86 işlemci ailesi Intel 80x86 CISC tabanlıdır. 1980'lerin ortasından beri oldukça düşük maliyeti olmasından dolayı masaüstü bilgisayar teknolojilerinde kullanılmaktadır. X86 işlemciler ve destek yongaları birden çok satıcılarda bulunabilmektedir. İşlemci / Fiyat aralılığında geniş ürün yelpazesi bulumaktadır. Masaüstü bilgisayarlar için x86 işlemciler performans için optimize edilip fakat güç için optimize edilmemişlerdir. Büyük masaüstü bilgisayar işlemcisi üreticileri (Intel, AMD) yeni tasarımlar ve 64-bit mimarilere yönelmektedir. Ancak diğer birçok üreticiler şimdi gömülü cihazlar için X86 işlemciler üretmektedir.

Şekil 1.6'da her yeni üretilen gömülü işlemci sayısını göstermektedir. Tipik olarak bir veya iki işlemcili yeni gömülü ürün tasarımı bulunmaktadır. Birçok mevcut cep telefonları hem genel amaçlı bir işlemci ve yüksek veri iletişim oranları görevleri ve protokolleri için gerekli yönetmek için bir DSP (dijital sinyal işleme) işlemcisi bulunmaktadır. İşlemciler gömülü sistemlerde kullanılan elli milyar dolar yıllık işlemci piyasasında satış üzerinde %98 almaktadır.



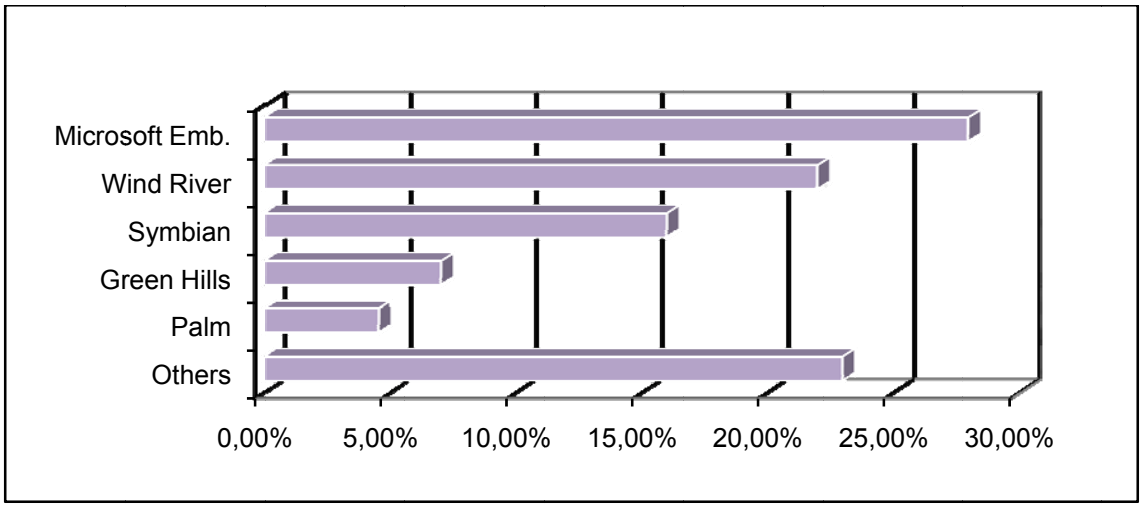
Şekil 1.6 İşlemci sayılarına göre gömülü sistemler dağılımı



Şekil 1.7 Kullanılan gerçek zamanlı işletim sistemine göre gömülü sistemler dağılımı

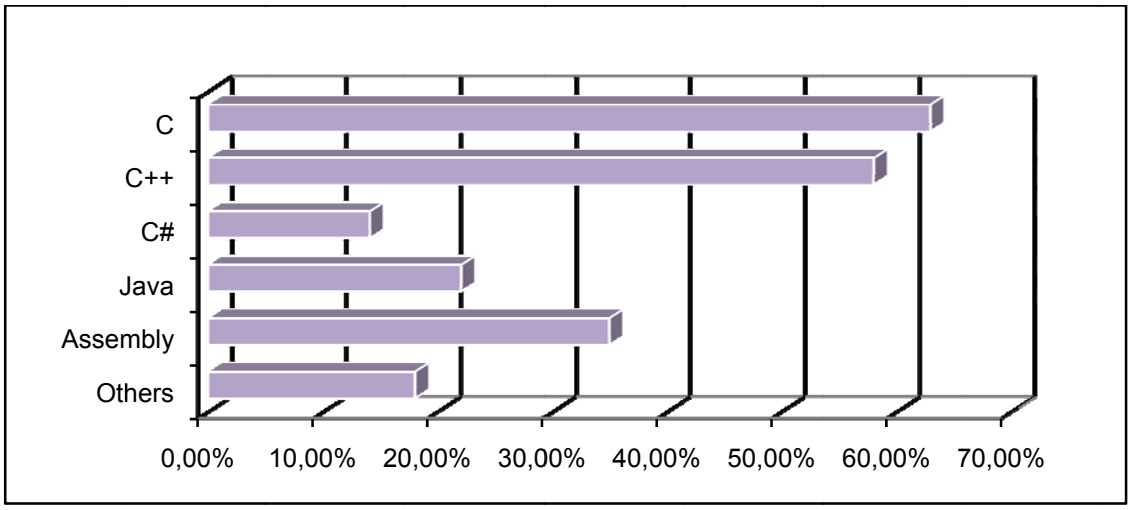
Şekil 1.7'de kullanılan işletim sistemlerine göre gömülü sistemlerin dağılımı görülmektedir. İşletim sistemine sahip olması gereken bir cihaz için en önemli özellik işletim sisteminin satın alır almaz hemen kullanılabilir olmasıdır. Çeşitli çalışmalar ve araştırmalar göstermiştir ki açık kaynak kodlu işletim sistemleri pazar payı kaybetmekte ve yerinde saydıkları görülmüştür. Bir çalışmada toplam ürün geliştirme maliyeti gerçekten zaman zaman açık kaynak kodlu bir işletim sistemi geliştirme ile harcanan maaş ve zaman lisans ücretinden çok fazla olduğu ortaya çıkmıştır. Bir başka nedeni de işletim sisteminin olgunluğa erişmesi ve geliştirme amacının ticari olması işletim sisteminin kullanımında kolaylıklar sağlamaktadır.

Ticari işletim sistemleri olarak anket sonuçlarında Şekil 1.8'de görüldüğü gibi Microsoft Windows XP Embedded ve Windows Embedded CE yi birleştirerek Microsoft Gömülü Sistemler olarak 2006'da en çok pazar payına sahiptir. Arkasından Wind River Systems VxWorks gelmektedir. Symbian işletim sistemi en çok cep telefonlarında kullanılmaktadır. Şu an Nokia Symbian'ın sahibidir.



Şekil 1.8 Gömülü sistemlerde kullanılan ticari işletim sistemleri dağılımı.

Şekil 1.9'da gömülü sistem tasarımcılarına sorulan "gömülü sistemler geliştirmek için hangi dilleri kullanıyorsunuz?" Sorusuna karşılık verilen cevapların grafiği vardır. C dili açıkça görüldüğü gibi en çok kullanılan dildir. Assembly için ise sistemin kendi tasarımlarının bazı bölümlerinde bu dili kullanmak gerekiyor diye cevap vermişlerdir. Yine de bu sonuç ankete katılanların vermiş olduğu cevaplardan çıkartılmıştır. Bu demek değildir ki gömülü sistemler sürekli assembler da yazılıyor. Sadece yapmış oldukları kendilerine özgü uygulamaların bir bölümünde assembler kullanılmaktadır. Assembler dili C Dil ailesi ile beraber kullanılmaktadır. Şu anda Büyük PC geliştiricileri PC'nin ROM BIOS 'larını yavaş yavaş C diline dönüştürülmeye başlamıştır.[1]



Şekil 1.9 Gömülü Sistemlerde kullanılan uygulamaların geliştirilme Dillerine göre dağılımı

Sonuç olarak yeni gömülü sistemlerde 32-bit işlemci kullanılmaktadır. İşletim Sistemi gerektiren sistemler kesinlikle karmaşıktırlar. Bir işletim sistemine sahip olan bu cihazların ticari bir cihaz olma özelliği yüksektir. Gömülü sistem geliştirmek için kullanılan en yaygın dil C dil ailesidir. Assembly dili hala bazı cihazlarda alt katmandaki rutinler için kullanılmaktadır. Fakat kullanımı gittikçe azalmaktadır.[1]

3. eBox2300 GÖMÜLÜ BİLGİSAYAR SİSTEMİ



Şekil 2.1. EBox 2300 MSJK bir Windows Embedded CE düşük maliyetli X86 SoC tabanlı gömülü bir bilgisayar sistemidir. Bir dizi Giriş/Çıkış özelliklerine sahiptir. Fotoğraf ICOP Corporation'dan alınmıştır.

Şekil 2.1'de görülen eBox 2300 düşük maliyetli X86 gömülü bir bilgisayar sistemidir. 4,5 x 4,5 inç büyüklüğünde ve bu küçük boyutlarına rağmen, bir masaüstü bilgisayar bulunan donanım özelliklerinin birçoğunu içermektedir. İşlemcisi, (SoC) x86 uyumlu 200Mhz Vortex86 yongalı sistemdir. İşlemcisi mevcut bilgisayarlara göre daha yavaş, fakat bir masaüstü PC'ye göre önemli ölçüde daha az güç tüketimi ve herhangi bir soğutma fanları gerektirmez. Tipik bir masaüstü bilgisayar 700W ve üstü güç isteyebilir fakat eBox maksimum 15W güç ister. 128MB SDRAM sistem belleği, PC-stili AMI BIOS bir adet 256MB dâhili flash sürücü (DOS önyükleyici ve OS için gerekli). Başlangıç donanım PC uyumlu, Windows XP, Windows CE ve Linux dâhil olmak üzere popüler işletim sistemlerini çalıştırabilir. Harici 5V DC 3A güç ile eBox çalışabilmektedir.[7]

3.1.eBox 2300 Mimari Yapısı

Ebox, tipik bir masaüstü bilgisayarda bulunana tüm giriş/çıkış birimlerine sahiptir. Arka panelde giriş/çıkış birimleri Şekil 2.2'de görülmektedir. Bir PS/2 klavye ve fare girişi vardır. (PS / 2 Y kablo gereklidir) ve standart bir VGA çıkışı vardır. Standart bir PC klavye, fare ve bir VGA monitör çıkışı vardır. 2.300 MSJK modeli (COM1 ve COM2) iki RS232C seri portu bulunmaktadır. Şekil 2.1'de görüldüğü gibi ön paneldeki AC97 ses iki adet USB 1.1 konektörleri, bir adet CF kart yuvası vardır. Ek olarak bir adet USB konektörü arka panelde mevcuttur.[7]



Şekil 2.2. eBox 2300 arka panel görünümü Giriş / Çıkış konektörler ve güç düğmesi

Arka panelde Rocker Switch ile eBox'a güç verilmektedir. Reset/güç düğmesi önde ve eBox açıp kapatmaya yarar, ama rocker düğmesi kapalı ise hiçbir hizmet görev yapamaz. Bir adet 10/100Mbs Ethernet ara yüzü vardır. Mini-PCI slot işlemci anakartın alt kısmında harici bir anten ile isteğe bağlı dâhili kablosuz kart destekleyebilmektedir.

Üst kapağı kaldırılmış olan bir adet eBox2300'ü Şekil 2.3'de ana bileşenleri ile görülmktedir. eBox2300 üst kapağı ayrıca işlemci için soğutucu olarak yardımcı olarak soğutma fanına olan ihtiyacımızı ortadan kaldırmaktadır. Ana kartın alt tarafında opsiyonel kablosuz yönetim kurulu ve CF yuvası ön panelde görülen bir iç mini PCI yuvası içermektedir.

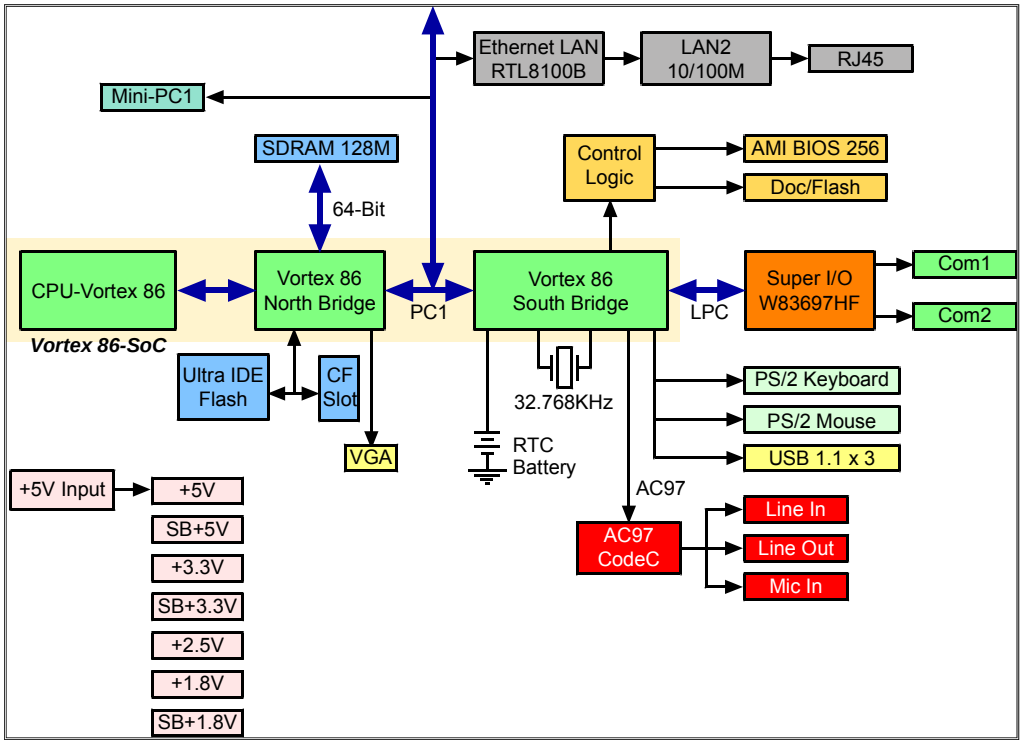


Şekil 2.3 eBox 2300 iç kısmı

Windows Embedded Ce 6.0, Visual Studio 2005 ile çalışmaktadır. Windows Embedded Ce 6.0'ın kurulu olduğu PC'de oluşturulacak olan yeni işletim sistemini eBox2300'e yüklemek ve hataları yakalamak için çapraz ethernet kablosu ile birbirine bağlanmaktadır.

Windows Embedded CE'de normal olarak debug bilgilerine COM1 portundan erişilebilmektedir. COM1'e seri null modem kablosu takılarak PC ye bağlanır ve CE 6.0 debug mesajları PC ye COM1 üzerinden gelir. Diğer giriş/çıkış cihazları normalde COM2 portuna bağlanmaktadır.

EBox sistemiyle ilgili bir blok diyagramı Şekil 2.4'de görülmektedir. Mimari bir PC'nin anakartı ile benzerlik göstermektedir, ancak Vortex86 (SiS55x) SoC dahili X86 işlemci ek olarak hem Kuzey ve Güney Köprüsü yongalarını içerir. Normal bir PC'de, Kuzey ve Güney Köprüsü yongaları anakartüzerine eklenmektedir.

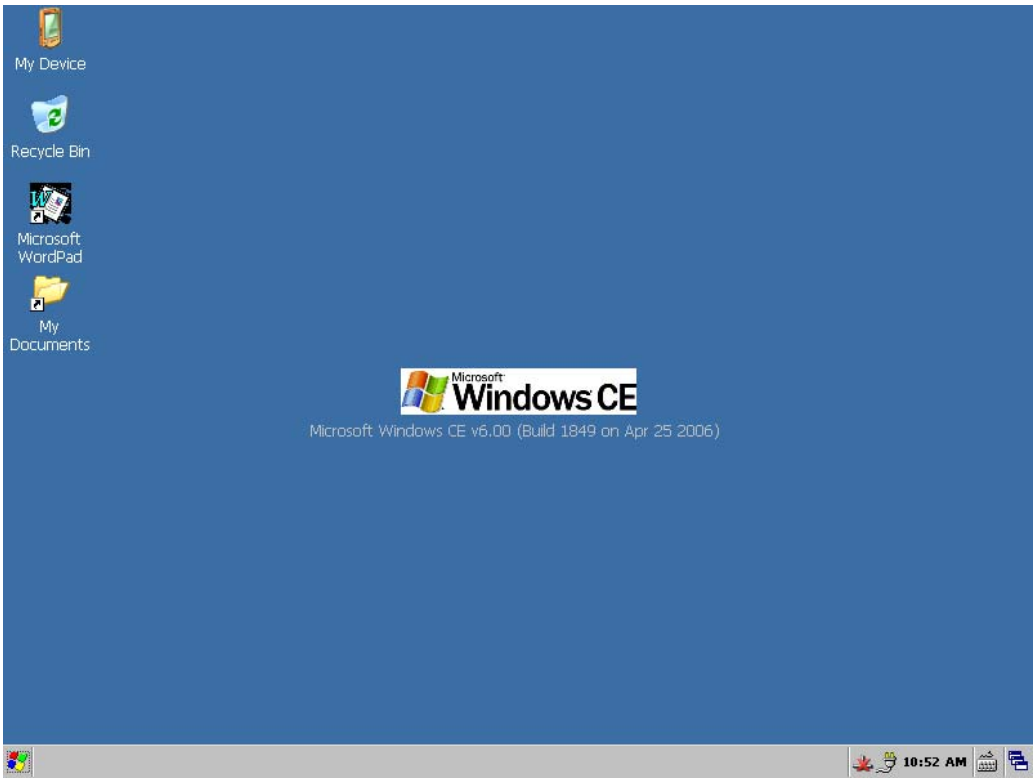


Şekil 2.4. Blok EBox 2300 Gömülü Bilgisayar Sistemi blok şeması

3.2.eBox'ın Boot (başlatılması) Edilmesi

EBox2300 'e enerji verildiğinde,, eBox'ın BIOS ROM'unda bulunan kodu bir PC gibi işlemeye başlar.Windows Embedded CE'nin açılabilmesi için fabrikada bazı dosyalar Flash sürücüsüne yüklenmiştir. Flash sürücüdeki dosyalar otomatik olarak DOS kurulumunu yapar ve AUTOEXEC.BAT dosyasını çalıştırarark kullanıcıya MENU'yü gösterir.Menü özellikleri içinde bulunan işletim sisteminin NK.BIN dosyasını yüklenmesi, DHCP ile network üzerinden PC ile bağlanı kurarak işletim sisteminin yüklenmesi veya ağdan açılmak için sitatik IP alarak boot edilmeyi beklemesi gibi özellikler vardır.

Ağ bağlantısı ve boot işlemi için eBox2300 ile PC aynı alt ağda olmalıdır. DOS programı olan LOADCEPC yeni işletim sistemini ağ üzerinden yükler. Her eBox cihazı benzersiz olarak CEPCXXXX gib bir ağ aygıt adı vardır.



Şekil 2.6. Windows Embedded CE 6.0. ın başarılı olarak açıldıktan sonraki ekran görüntüsü

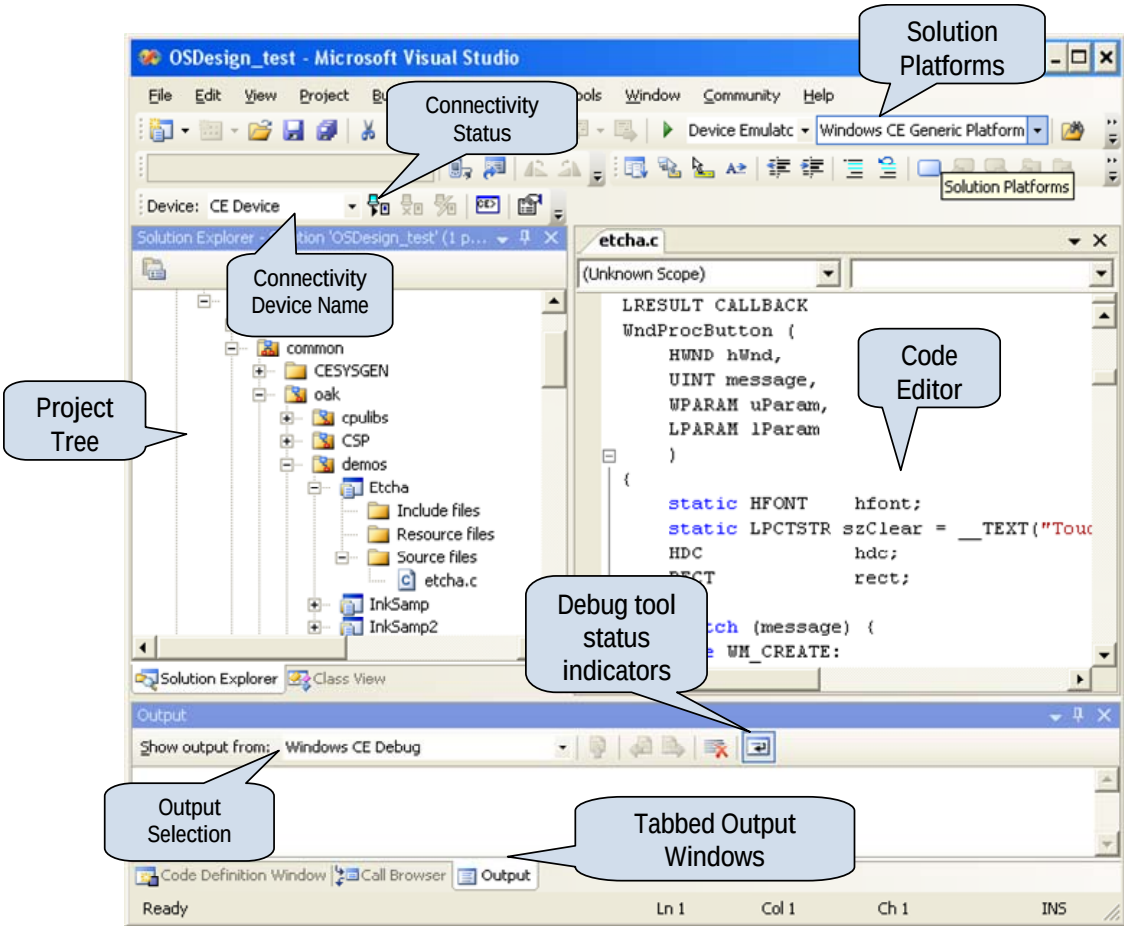
eBox2300'ün tamamlayıcısı olarak bütün giriş/çıkış birimlerinin sürücüleri mevcuttur. Tablo 2.1 de görüldüğü gibi her cihaz için gerekli olan sürücü listesi verilmiştir.

Tablo 2.1 eBox 2300 CE 6.0 için cihaz sürücüleri

eBox-2300 Özellikleri	CE 6.0 Sürücüsü
Video (SiS-550)	Vortex86_Display (Driver included with ICOP eBox BSP)
Ses (SiS-7019)	Vortex86_Audio (Driver included with ICOP eBox BSP)
Ethernet (RealTek-8100)	RTL8139 (Driver provided with CE 6.0)
Seri Port (16550 UART compatible)	Com16550 (Driver provided with CE 6.0)
USB Port (USB 1.1 Host)	OHCI (Driver provided with CE 6.0)
USB Flash Sürücü	USB Storage Class (Driver provided with CE 6.0)
USB Web Kamera (USB Video Class 1.1)	A new CE 6.0 USB camera driver is available for download at www.microsoft.com/downloads
CF storage kart	ATAPI (Driver provided with CE 6.0)
IDE Disk	ATAPI (Driver provided with CE 6.0)
MiniPCI (internal slot)	PCI Bus (Driver provided with CE 6.0)
Internal 802.11 Wireless Card - optional (VIA VT6655 - US only)	VIA VT6655 (Driver provided by VIA & ICOP)

4. PLATFORM BUILDER VE İŞLETİM SİSTEMİ DERLEME

Platforum builder aracı kullanılarak CE işletim sistemi için yeni bir çekirdek derlenmektedir. Platform Builder görüntüsü şekil 3.1 de görülmektedir. CE 6.0 için Platform Builder Visual Studio 2005 SP1 de eklenti olarak çalışmaktadır. Platform Builder da tasarımcı çeşitli özellikler ve ihtiyaç duyulan aygıt sürücüleri Catalog Items bölümünde seçebilmektedir. Daha sonra kullanıcı oluşturulmuş olan özel işletim sistemi özellikleri ile en üstten Build ederek işletim sistemi çekirdeğini derlemektedir. Yeni oluşturulan çekirdek ARM emulatorunden veya hedef aygıt da çalıştırılıp hata ayıklama işlemi yapılabilmektedir. Hedef aygıt Network, USB veya seri haberleşme üzerinden bağlanabilmektedir.[8]



Şekil 3.1. Platform Builder yeni İşletim sistemi çekirdeği derlemek için kullanılan bir araçtır.

4.1.Platform Builder Terimleri

Platform Builder'da kullanılan çok sayıda benzersiz terim bulunmaktadır. Aşağıda bu terimlerin açıklamaları bulunmaktadır.[9]

Catalog item: Herhengi bir öğeyi katalog öğeleri profilinden seçebilirsiniz.

Design template: Microsoft'un genel hedef cihazlar için seçmiş olduğu ön tanımlı bileşenlerin seçili olduğu bir taslak işletim sistemleridir. Bu taslaklar hedef cihazda çalışıtırılabilir durumdadır.

OS design: Katalog öğelerinden seçilecek olan öğelere göre işletim sisteminin karakteristiğini belirlenmektedir. OS Design'a taslak kullanarak veya kullanmayarak da başlanabilmektedir.

Run-time image: Hedef cihaza çalıştırılmak üzere gönderilecek olan yazılımdır. İşletim sistemi ve gerekli olan yazılımı içinde barındırmaktadır.

Board support package (BSP): Donanıma bağımlı olarak oluşturulan bir yazılımdır. Bu yazılım içinde boot loader, Oem Adaptation Layer (OAL) ve ana karta özel sürücüler bulundurmaktadır.

Catalog: CE nin seçilebilir işlevselliklerini taşıyan bir konteynırdır.

Component: OS tasarımına eklenebilecek en küçük işlevsellik birimidir.

Configuration: Katalog öğelerinin seçimi ve derleme özelliklerinin hepsine denir.

Hardware platform: CE işletim sisteminin çalıştırılacağı ve yazılım tarafından desteklenen donanım mimarisidir.

Module: CE işletim sisteminin bir parçası olan EXE veya DLL .

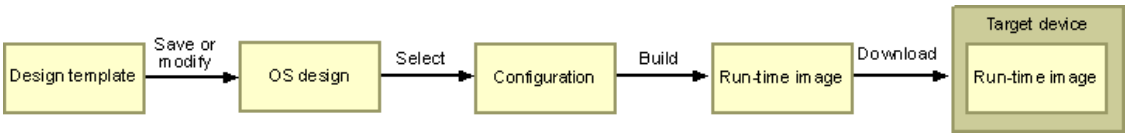
Subproject: CE işletim sistemine işlevsellik eklemek veya izleme mekanizması eklemek için derlenebilir dosyalar topluluğu OS Design Projesi birçok subproject barındırabilmektedir.

target device, CE –based device: Donanım veya kombine edlimiş donanım mimarisi örneği ve yazılım mimarisi.

Project: OS Tasarımı ile ilişkili tüm dosyalar topluluğu

4.2. CE İşletim Sisteminin Derlenme Süreci

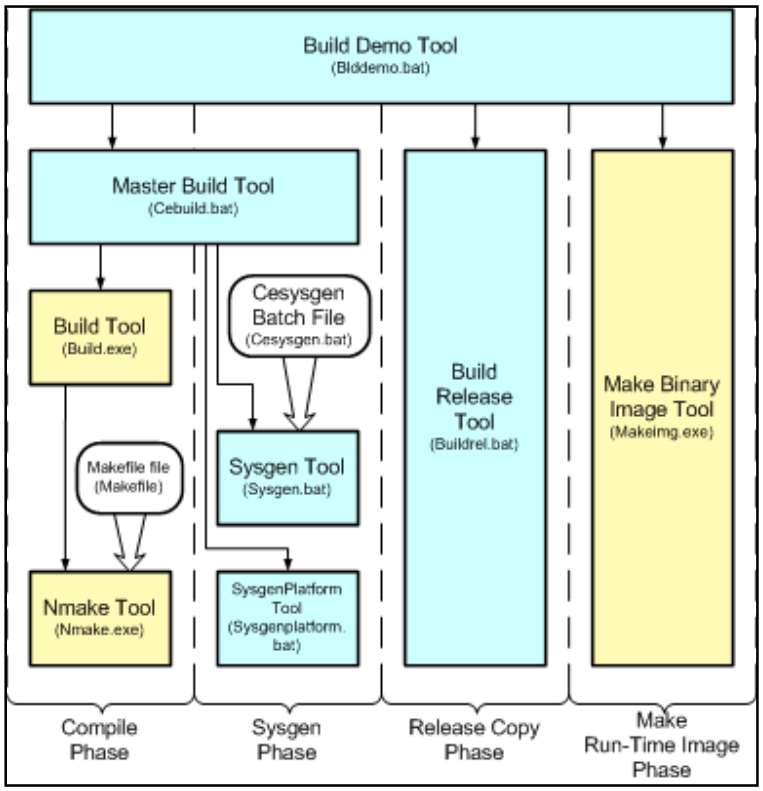
Çalıştırılabilir bir çekirdek derlenmesi için istediğimiz özelliklere sahip işletim sistemi oluşturmamız gerekmektedir. Birçok tasarım taslakları kullanarak hızlı bir şekilde işletim sistemi oluşturulabilmektedir. İşletim sistemi oluşturma adımları şekil 3.2'de verilmiştir.



Şekil 3.2 İşletim sistemi çalıştırılabilir image oluşturma süreci.

OS tasarımı derledikten sonra, **make a run-time image** seçeneği ile işletim sisteminin ihtiyacı olan tüm dosyalar ve subprojectlerin hepsi eklenerek nk.bin (New Kernel) oluşturulur. Oluşturulan bu image hedef cihaza dağıtımı yapılarak hedef cihazda işletim sistemi çalıştırılır. **Build a run-time image** sırasında Platform Builder aşağıdaki fazları işler.

- Derleme Fazı
- (System Generator) Sysgen Fazı
- Yayınlama Kopyalama Fazı
- Run-Time Image Oluşturma Fazı



Şekil 3.3. CE Derleme diagramı

Derleme sisteminin bu fazları işlemesi boyunca aşağıdaki görevleri tamamlar,

- Header dosyalarını oluşturur.
- Moduller linklenir.
- Modul sonuçlarını dağıtım klasörüne kopyalar.
- Run-time image oluşturur.

Hedef cihaz olarak çeşitli cihazları eklenebilmektedir, bu herhangi bir işlemciyi kullanan sistem donanımı veya ARM cihaz emulatorleri veya CEPC (CE çalışan PC ler) olabilir. Böylece yeni işletim sistemini herhangi bir cihaz üstünde hata ayıklama ve test edilebilmektedir.[1]

4.3.Özelleştirilmiş CE 6.0 Çekirdeğinin Derlenmesi

Embedded cihazların sabit değerlerini bir şekilde kontrol edilebilmesi için işletim sistemine ihtiyaç duyulmaktadır. Sadece cihaza sürücüler ve ihtiyaç duyalan servisler yerleştirilirse kernel in büyüklüğü küçük olur. Bu da bize düşük büyüklükteki hafızalarda (memory) çalışma imkanı sağlamaktadır. Eğer kernel yeterince küçük olmuyorsa ki bunun sebebi uygulama ve uygulamanın ihtiyaç duyduğu hafıza alanınn büyük olmasıdır. O zaman biraz daha büyük hafızalı cihazlar kullanılır. Böylece daha fazla boşta hafıza olacak ve uygulama programları ve ihyitaç duyulan depolama alanımız fazla olacaktır.

Bazı durumlarda daha az hafıza kullanarak işletim sistemini çalıştırmak cihazın stabil olmasını sağlayabilir. En düşük Windows Embedded CE kernel 400K bytes'da çalışır. En yüksek ise içinde C#.NET Compact Framework, tüm aralıkda I/O cihazları, ve full debugging desteği ile kernel 40MB üstüne çıkabilmektedir.[1]

Yeni bir işletim sistemi kerneli yapılandırma işlemlerinin tümüne **sysgen** denir. Sysgen **System Generation**'un kısaltılmışıdır. Bu işlemler sadece basit bir programı derlemek (compile) ve link yapılmasından daha fazla işlemleri kapsamaktadır. İşlemlerin kopyalamaya, kontrol etmeye, derlemeye ve yüzlerce derlenmiş bilgilerin bir birleri ile link edilmesine ihtiyaç duyar. Bu sebeple işlemlerin bitmesi için biraz zaman almaktadır.

Windows Embedded Ce'de özel tasarlanmış OS geliştirme aracı olan **Platform Builder** vardır. Bu araç Visual Studio ile yeni bir OS Kerneli düzenlememize olanak sağlar.

4.3.1. Windows Embedded CE 6.0 Derleme/Hata Ayıklama

Windows CE Run-time image oluşturma, isteğimize göre değişiklik yapma, build ve debugging işlemleri incelenmiştir. Yeni üretilen runtime image, CE destekli ARM cihazda çalıştırılmıştır.

4.3.2. Hazırlıklar

Yazılım geliştirme için kullanacağımızı PC Windows XP 512K - 1GB Ram ve en az 18 GB lik hardisk de boş alan ihtiyaç vardır. Bu bize birkaç tane OS inşa etmemiz için yeterli bir alan olacaktır. İhtiyaç duyulan yazılımlar aşağıda sıralanmıştır ve bu yazılımlar bu sıra ile kurulmalıdır.

- Visual Studio 2005
- Visual Studio 2005 Service Pack 1 (SP1).
- Windows Embedded CE 6.0 (VS 2005 in eklenti)

Kurulumu yaparken ARM ve X86 prosessorler seçili olmalıdır. Böylece hem sanal da çalışılabilinir hemde ARM CPU'lu cihazlarda çalışılabilinir. Eğer fazla disk de boş alan yok ise sadece X86 yi seçip varsayılan yere kurulumu yapılmalıdır.

- ICOP Vortex86 Rev 60B8 Board Support Package (BSP) (for CE 6.0)
- ICOP eBox SDK Rev 60B (for CE 6.0)

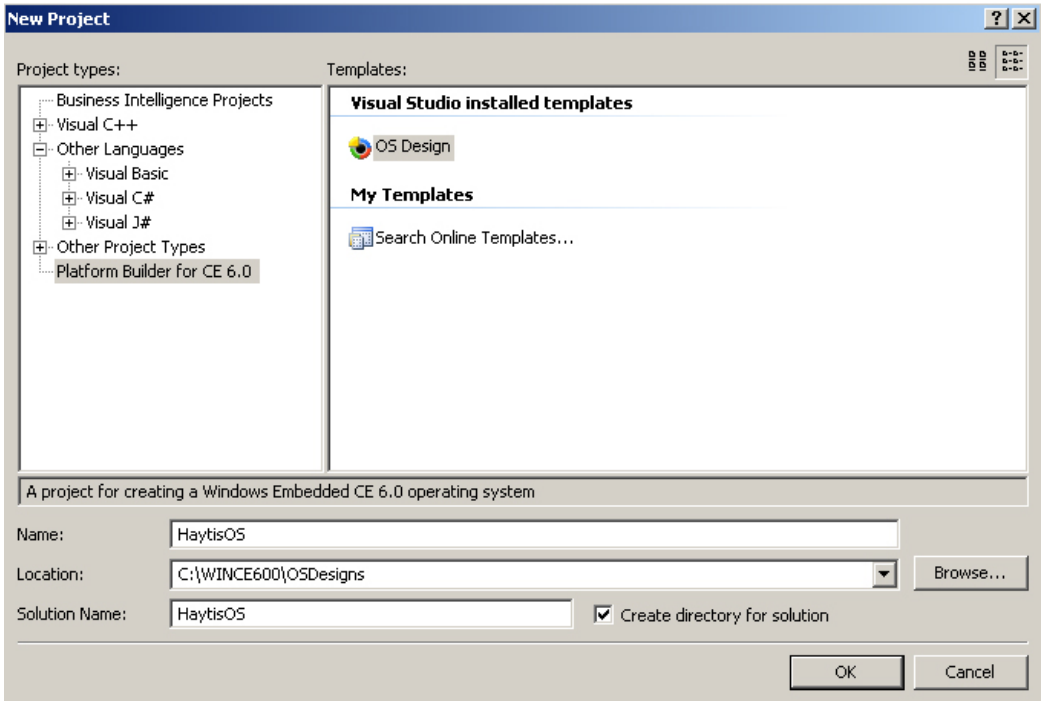
Her fazladan yeni OS inşa etmek için yaklaşık hard disk de 2GB lik yer tutmaktadır.

4.4.Ebox BSP kullanarak yeni OS Image Kernel İnşa Etmek

EBox2300'de çalışacak bütün özelliklere sahip CE Kernel in detayları ile birlikte adım adım işlenmiştir. Tüm özellkleri kernele ekleyeceğimizden kernel büyüklüğü biraz fazla olacaktır. İhtiyaç duymadığımız servis ve sürücüleri eklemediğimizde kernel küçük boyutlu hale gelecektir.

CE geliştirme araçları Visual Studio içine plug-in (eklenti) olarak eklenmektedir. Yeni bir OS kernel inşa etmemiz için Visual Studio 2005 de yeni bir proje açılır.

File→New→Project→ seçip açılan formun sol tarafındaki listeden **Platform Builder for CE 6.0**'ı seçelir.



Şekil 3.4 Yeni OS Tasarım Projesi Oluşturma

Project types bölümünden Platform Builder for CE 6.0 seçilir. Templates bölümünden **OSDesign** seçilip projenin adı verilir. **Create directory for solution** check işareti koyularak **OK** basılarak devam edilir. Birkaç saniye sonra **Windows Embedded CE 6.0 Design Wizard** formu ekrana gelir. Bu sihirbaz bize basit bazı sorular soracaktır. Next'e tıklayıp devam edelir. Sihirbaz kurulu olan **Board Support Package**[2]'lardan birini yada birkaçını seçerek devam etmemiz sağlar. eBox 2300 için **ICOP_Vortex86_60B:x86** seçilir. Sihirbaz bir sonraki adımda bize hangi Design Template'ini kullanacağımızı sorar. Design Template, işletim sistemi opsiyonlarının piyasada var olan standartlara göre hazırlanmıştır. Herbir Design Template otomatik olarak sürücüleri ve özellikleri seçer. Sihirbaz sonunda yapılması gereken sadece ihtiyaç olmayan gereksiz özellikleri ve sürücüleri çıkartmak yada ihtiyaç olan diğer sürücü ve özellikleri eklemek olacaktır. **Industrial Device**, eBox için en iyi seçimdir çünkü birçok özellikler içine eklenmiş olacaktır. Next deyip ilerlenir. Sihirbazın sonraki adımında Design Template'in değişkenlerinin seçilmesi formu gelir. Seçenekler Design Template için en temel yapıyı içerir. Listeden Internet Appliance seçelir. **Internet Appliance** otomatik olarak networking desteğini içermektedir. Yüksek kapasiteli bir kernel oluşturacağımızdan aşağıda listede olanların seçilmiş olduğundan emin olduktan sonra Next deyip sonraki ayar sayfasına geçilir.

- .NET Compact Framework 2.0 (C# için gereklidir)
- Internet Explorer 6.0
- Windows Media Audio/MP3
- Windows Media Player Application
- Windows Media Player OCX
- Windows Media Video/MPEG-4 Video
- WordPad
- XML MIME Viewer

[2] BSP, temel sürücüleri ve hedef cihaz platformu için diğer ihtiyaç olan OS özelliklerini içerir.

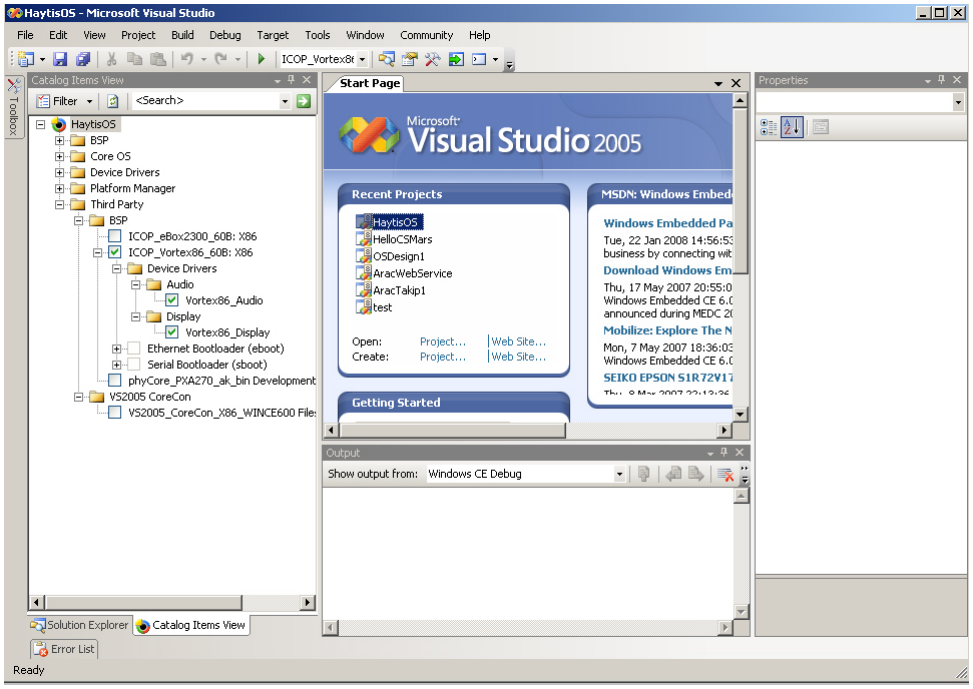
Sihirbaz bize bu aşamada bir çok network ve haberleşme opsiyonlarını göstermektedir. Varsayılan seçenekler bu durumda uygundur. Böylece Proje Yönetimi Sihirbazı sona ermiştir.

4.4.1. OSDesign isteğimize göre değiştirmek.

Bu noktada aslında OSDesign projemiz kurulmuştur. Proje dokumanları ana OSDesign klasörünün altındaki bizim vermiş olduğumuz isimdeki klasördedir. Bu design için path, C:\WINCE600\OSDesign \HaytisOS

Solution'in en üst seviyesi OS design projesini ve bazı C/C++ uygulamaları içerir. Bu uygulamalara subprojects ve C# uygulamaları olarak proje içinde belirir

Bir sonraki adımda hala OS design' ın ihtiyacı olan bazı sürücü ve compenentleri ekleme ile ilgili ayarlar yapılacaktır.



Şekil 3.5 Catalog Items View (sol column)

Catalog Item eğer bir şekilde göremiyorsanız açmak için View →Other Windows →Catalog Items View. Şeklinde ulaşabilirsiniz.

Hala OS Kernele birçok şey ekleyebiliriz. Catalog Items View'den add new item ile eklenir. Eğer bir item eklenmişse yeşil check box belirir. Yeşil kare olarak gözükenler ise yeni seçilmiş otomatik olarak eklenecektir Eklenmeyecek olanlar ise kırmızı olarak belirir. Catalog Items View sekmesinden third Party ağacını açalır. Buradan eBox tarafından desteklenen display ve audio sürücülerini işaretlenir.

Audio →Vortex86_Audio

Display→Vortex86_Display

Son olarak yazmış olduğumuz kodların deploy edilip eBox da çalıştırabilmemiz için ConMan Corecon files'lara ihtiyacımız olacaktır bu VS_2005_CoreCon_X86 yi da

seçmemiz gerekiyor. Eğer bu seçenek görünmüyorsa Setup edilmemiştir. Kurmanız gerekmektedir.

4.4.2. Opsiyonel kurulumlar:

VS 2005 kullanarak eğer uygulama geliştirmek istiyorsanız;

Core OS altındaki CEBASE altındaki Applications-End User →CAB File Installer/Uninstaller seçilmelidir.

C# Proje geliştirip çalıştırabilmek için

Applications and Services Development→.NET Compact Framework seçilmelidir.

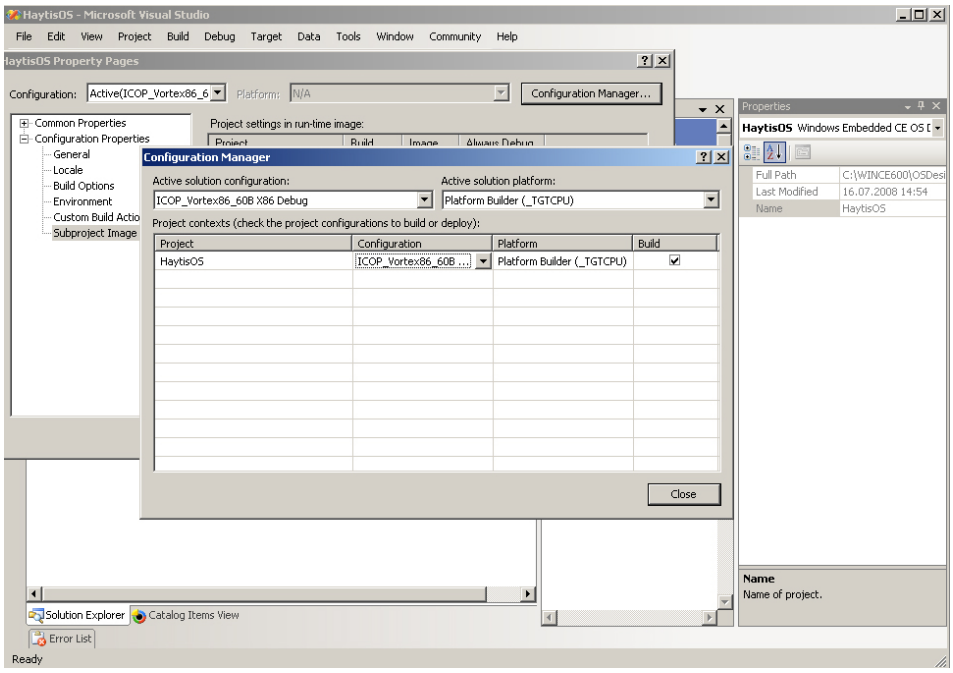
USB Host desteği sağlayabilmek için

Core OS altındaki CEBASE altındaki Core OS Service→USB Host Support→USB Storage Class Driver seçilmelidir.

4.4.3. Configuration Manager ve Debug & Release Build

Build→Configuration Manager kullanarak Configuration Manager'i açalır. Configuration Manager Şekil 3.6'da görüldüğü gibidir. ICOP_Vortex_60B_x86 Debug mode seçili ve aktif olduğundan emin olunur. Herbir proje için 2 farklı tipte OS Build desteklemektedir. Bunlar Debug Build ve Realese Build'dir. Debug Build birçok debug mesajlarını yakalamıza ve ilk OS yeni cihaza yüklenirken karşılaşılacak sorunları çözmemize yardımcı olur.Eğer Debug Mesajlarını almak istemiyorsak release Mode da Build edebiliriz.release Build de **%40** daha az memory kullanmış oluruz

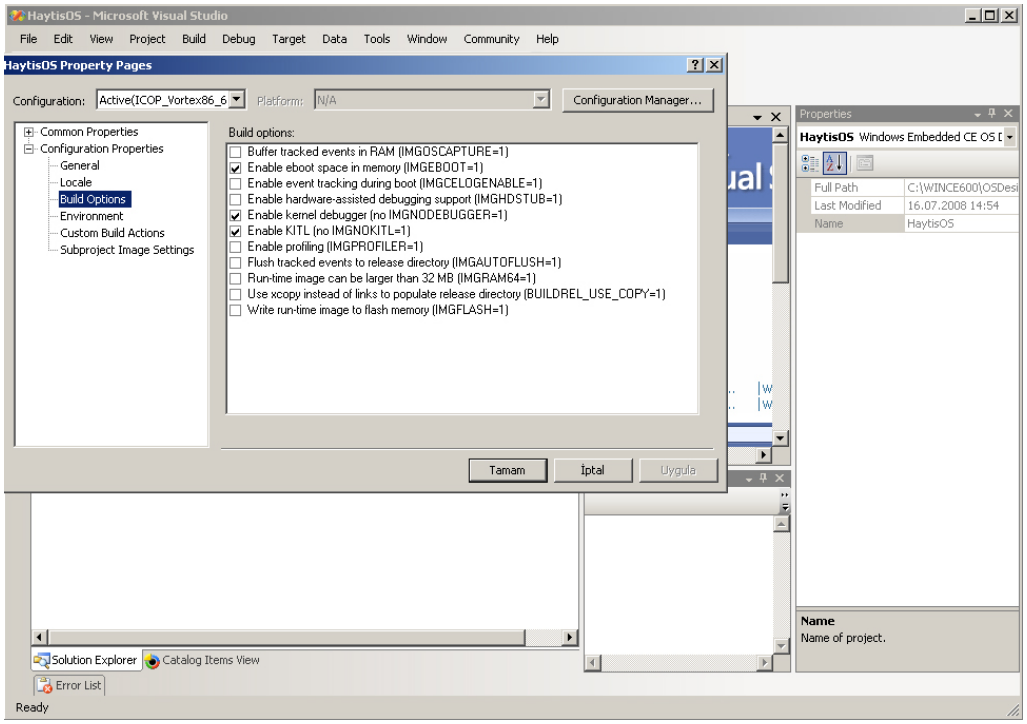
Debug mode OS design test edilip sorunlar giderildikten sonra Release Build kullanılarak proje sonlandırılır.



Şekil 3.6 Configuration Manager ‘ı Debug Mode ve Release Mode durumunu seçmek için kullanıyoruz

4.4.4. Projenin derleme opsiyonlarının ayarlanması

İşletim sistemini derlemeden önce hala birkaç ayarları seçmemize ihtiyaç vardır. Solution Explorer’i seçerek projenin solution görünümün tabına geçelir Kendi işletim sistemi adına (bu proje için HaytisOS) sağ klik yapıp **Porperties**’i açalır. Properties penceresi açılınca Configuration Properties genişletip, **Build Options** seçilir Property sayfası Şekil 3.7 deki gibi görünecektir.



Şekil 3.7 OS Property Sayfası

Bu ekranda aşağıdaki ayarları yapalır.

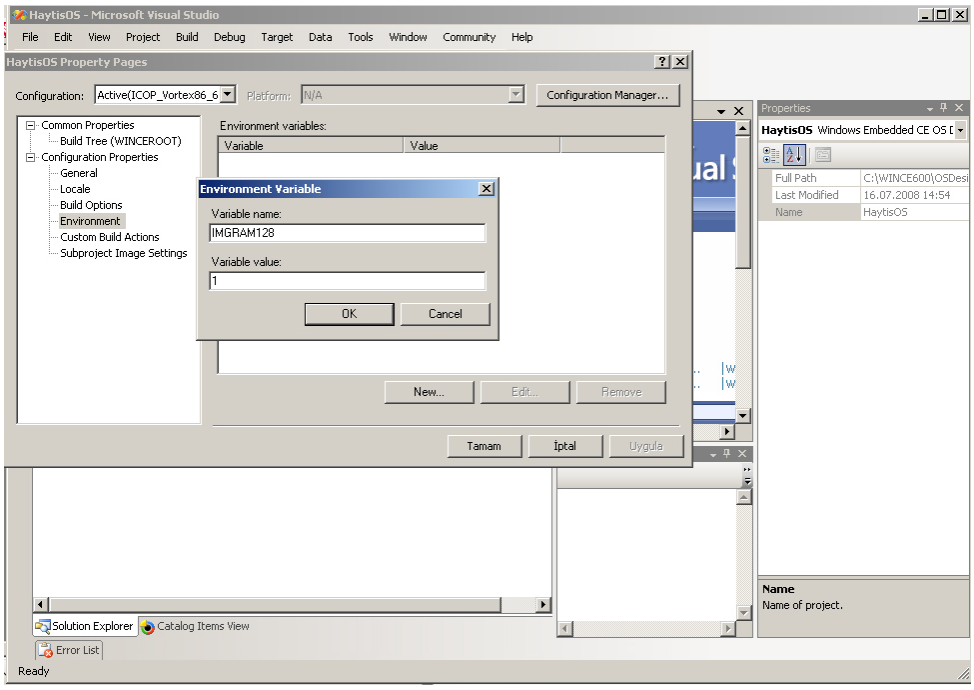
• **Enable eboot space in memory (IMGEBOOT=1) :** Bu özellik ethernet üzerinden debug ve bundling boot loder desteklemesini sağlar

• **Enable kernel debugger (no IMGNODEBUGGER=1)**. Run-time image in debugger'ının host ile target cihaz arasinda bağlantı sağlamasını ve debug mesajlarının iletmesini sağlar

• **Enable KITL (no IMGNOKITL=1) KITL,** Kernel Independent Transport Layer ın kısaltılmışıdır, hedef Windows CE ve geliştirme istasyonu arasindaki debug/download bağlantısını sağlar.

4.4.5. OS Ortam Değişkenlerinin Ayarlanması

Yapılandırma sisteminin ortam değişkenleri çeşitli maddelerin image tarafından hafıza konfigrasyonunun kapsamasını ve kapsamamasını ve ayarlamak için kullanılır. Property Page'deki Build Options altındaki Environment seçelir. Burda çok kiritik bir ortam değişkenini set edeceğiz. Şekil 3.8'de görebilirsiniz. New butonuna basalır açılan pencereden Variable Name **IMGRAM128**, **Value 1** olarak ayarlayıp OK basılır. IMGRAM128 ayarlamasını yapma sebebimiz, sonuçta çıkacak olan image in 128 M memory nin tüm alanını kullanmasını sağlayacaktır. eBox 2300 desteklediği toplam ram boyutudur.



Şekil 3.8 Property Page Kullanılarak ortam değişkenlerinin eklenmesi

Yapılandırma ile ilgili tüm ayarlamalar ve opsiyonlar seçilmiş oldu. Yeni OS image derlenebilir duruma gelmiştir. Visual Studio'nun yukardaki menüden Build→Build Solution seçilir ve Platfom Builder yeni OS image'ı tüm subprojectleri ile birlikte build etmeye başlar. İşlemlerin kontrol etmeye, derlemeye ve binlerce dosyaları birbirleri ile link etmeye ihtiyaç olduğundan Build işlemi 10-20 dk sürmektedir.

Build işlemi bittikten sonra oluşturulan dosyalar varsayılan kurulum klasörü altındaki;

Debug image:
C:\WINCE600\OSDesigns\MyOS\HaytisOS\RelDir\ICOP_Vortex86_60B_x86_Debug

Release image:

C:\WINCE600\OSDesigns\MyOS\HaytisOS\RelDir\ICOP_Vortex86_60B_x86_Release
Başarılı bir build sonrası bir NK.BIN dosyası oluşur.Bu dosya yeni Windows CE 6.0 image dosyasıdır.

4.4.6. Yeni OS Kernel 'inin Ebox 2300 e Download Ettirilmesi

EBox a elektrik adaptörünü, klavyeyi, mouse ve monitoru, ethernet kablosunu bağlayıp çalıştırılır. Herhangi bir klavye, mouse, monitor hoparlör eBox 2300 ile çalışabilmektedir. Arkadaki Adaptor kablosunun takılı olduğu soket altında power anahtarı vardır. Önce bu power anahtarını açılır, daha sonra öndeki power butonuna basılır. Böylece eBox2300 çalışır hale gelecektir. Öndeki yeşil Led Power ON oldugunu belirtip yanacaktır. Monitörde ise önce BIOS mesajı gelecek ve arkasından **starts MS DOS..** yazcak ve boot başlangıç menusu ekrana gelecektir.

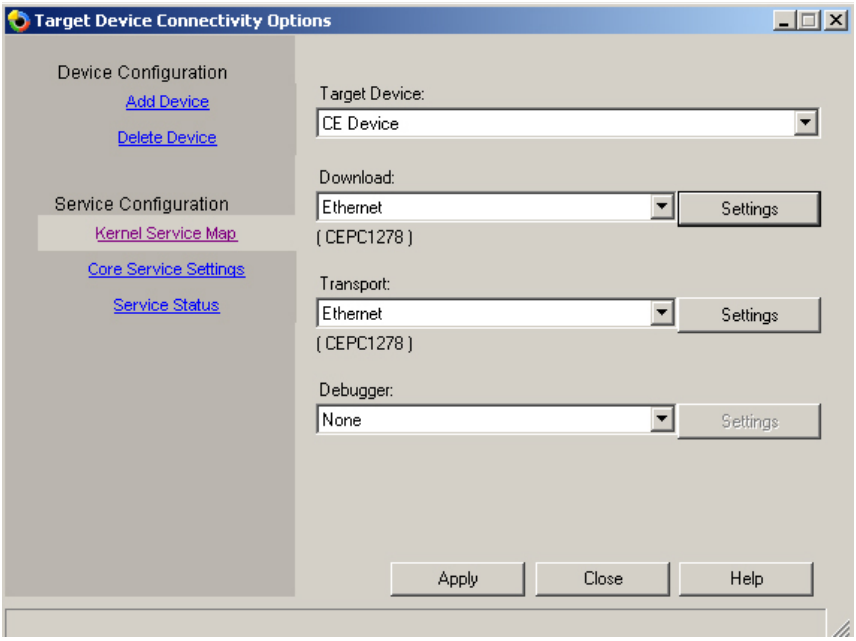
4.4.7. eBox için ağ kurulumu

Hızlı bir download için, network bağlantısı gerekmektedir. Statik IP adresi üzerinden hızlı bir şekilde bilgisayarımızdan eBox2300 kerneli download ettirebilir. Öncellikle bu işlem için bize çapraz bağlantılı bir network kablosu gerekmektedir. Çapraz netwrok kablosunun bir ucu eBox2300'e diger ucu ise bilgisayarımıza bağlı olmalıdır. Her iki cihazda sitatik IP adresi almış olmalı ve aynı subnetwork'de olmalıdır. Çünkü eBox 2300 kendi subnetworküne ağdan BOOTME talebi gönderir. Bu çalışmamızda eBox2300 e **192.168.2.232/24** Bilgisayarımıza ise **192.168.2.110/24** ip adresi verildi.

EBox2300 debug mesajlarını network üzerinden gönderebildiği gibi debug errorlarını COM port üzerinden de gönderebilmektedir. Network ile bağlantı sağlayamacağınız bir durum olursa sadece debug hatalarını almak istiyorsanız, cihaz COM portuna null modem Serial kablo takıp diger ucunu da PC ye takıp hyper terminal çalıştırıp debug mesajlarını görebilirsiniz. Hyperterminal ayarlamaları **38400 Baud,8 data bits, no parity, no handshaking ve stop bit 1.**

4.4.8. Target Device Bağlantı Ayarları

Visiual Studio 2005 de üst menude **Target→Connectivitiy** Options seçip **Connectivitiy Options** penceresini açılır. (Şekil 3.9)



Şekil 3.9 Target Device Bağlantı Ayarları Penceresi.

Bu pencere açık iken eBox2300'ü power düğmesine basılır. eBox2300 yeşil led yanacaktır ve ekranda DOS Autoexec.bat çalışıp aşağıdaki menuyu ekrana basacaktır.

1. Boot CE/PC (local nk.bin with /L:800x600x32)
2. Boot CE/PC (ether via eboot.bin with /L:1024x768x32)
3. Boot CE/PC (ether via eboot.bin with /L:800x600x32)
4. Boot CE/PC (ether via eboot.bin with /L:640x480x32)
5. Boot CE/PC (ether via eboot.bin without display settings)
6. Boot CE/PC (Static IP: 192.168.2.232, with /L:1024x768x32)
7. Boot CE/PC (Static IP: 192.168.2.232, with /L:800x600x32)
8. Boot CE/PC (Static IP: 192.168.2.232, with /L:640x480x32
9. Clean Boot (no commands)

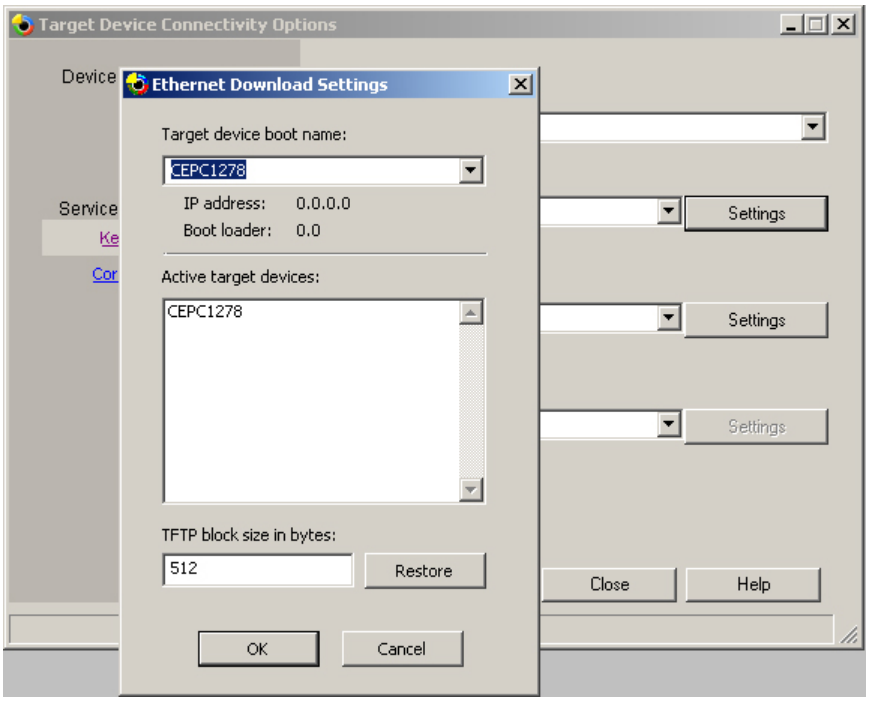
Seçenek 1: Lokal diskinde olan NK.bin (Windows CE image) yükler ve 800x600 çözünürlükte cihaz açılır.

Seçenek 2 den 5: Bu seçenek önce erişilebilir DHCP server üzerinden IP almaya çalışır ve bu IP üzerinden geliştirme PC'sine BOOTME talebi yapar.

Seçenek 6 dan 8 : Ebox2300 statik IP adresi 192.168.2.232 olarak ve Platform Builder geliştirme bilgisayarına BOOTME gönderecek şekilde açar.

Seçenek 9: Sadece DOS olarak açılır.

Target Device Connectivity hala açık olarak bekliyor olmalıdır. Burdan herhangi bir Ethernet Settings butonuna basalır. Ethernet Download Settings penceresi karşımıza gelecektir. Şekil 3.10 da görülmektedir.

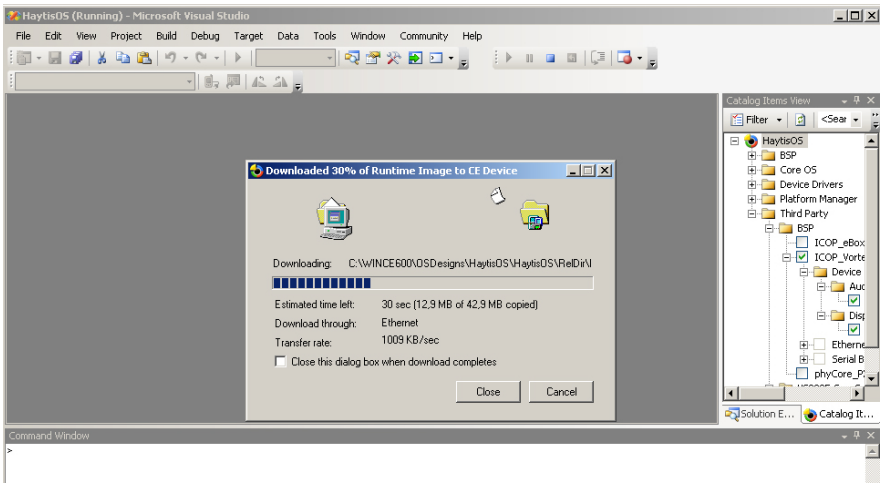


Şekil 3.10 Ethernet Download Settings penceresi

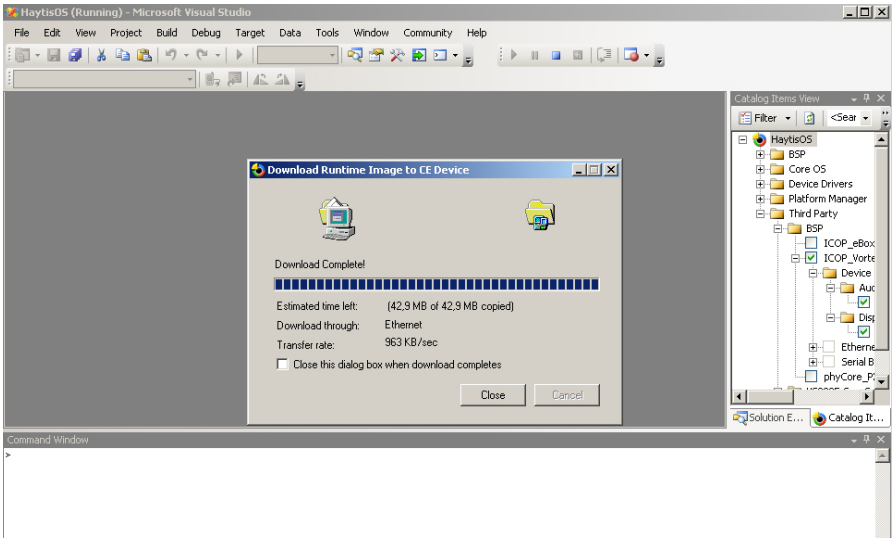
Herbir eBox sabit bir device adına sahiptir Bu ad standardı **CEPCXXXX**'dir..XXXX her zaman farklı numaraları barındırır. eBox BOOTME mesajı gönderdiğinde Bu penceredeki Acitve Target Devices listesine kendisini ekler.

4.4.9. Image'ın eBox2300 e Download Edilmesi

Visual Studio IDE'den **Target → Attach Device** seçilir. Birkaç saniye sonra Pencere etrafında açılıp kapanmalar olcak ve Şekil 3.11'deki pencere görünecektir. Bu pencere IDE nin arkasına geçebilir bu sebeple göremeyebilirsiniz. Ama Windows Çalışma Çubuğundan **Download Item** penceresini açabilirsiniz. Birkaç saniye sonra hedef cihazımız image'ı download etmeye başlayacaktır ve progres barda hareketlenme olacaktır. Eğer download başlamadıysa eBox2300'u bir restart edip yine bir önceki seçenekde tekrar açmalısınız. Şekil 3.12



Şekil3.11 Target cihaz image download ediyor



Şekil 3.12 Yeni Image download bitti.

Cihaz işletim sistemi image'ı download ettikten sonra debug mesajları VS2005 Output penceresinden görülebilmektedir. Bu işlemler bittikten sonra eBox2300'e bağlı olan monitorde işletim sisteminin çalıştığını görülür. (Şekil 3.13)

Eğer mouse ve klavye ile bişeyler yaptığnızda her bir işlemde IDE'nin debug mesajında bunları görebilirsiniz. Tabii ki bu durumda Debug modda oldugumuz için eBox2300 işlemlere bir geç cevap verebilir. Eğer mouse hareket ettirmezseniz ve bir şey yapmazsanız bir kac saniye sonra debug mesajları da gelmeyecektir.



Şekil 3.13 Hazırlanmış olan Image in ilk görüntüsü

5. SIMULATOR TASARIMI

5.1.Microchip PIC 18F876 ile Analog Dijital Sinyal Üretici Devresi (Simulator)

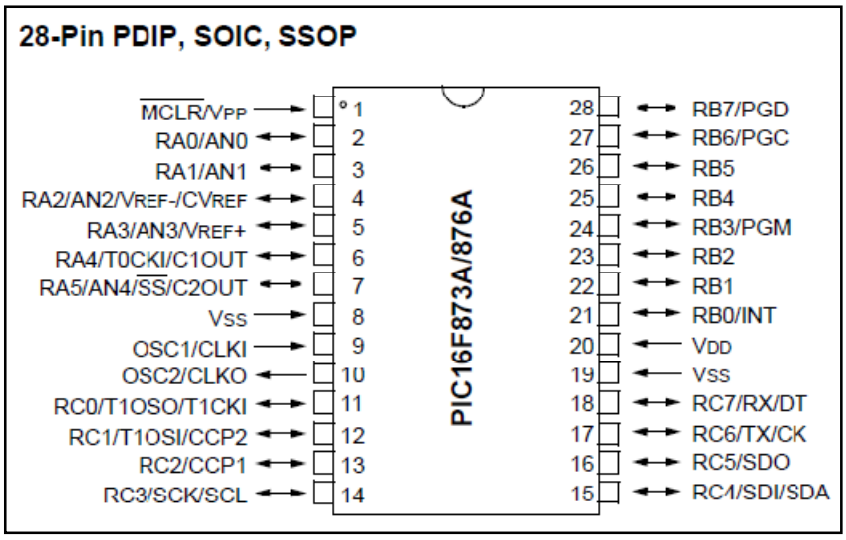
5.1.1. P I C 16F876 Çekirdek Mimarisi

PIC16F876 mikro denetleyicisinin bazı özellikleri aşağıya listelenmiştir.[13]

- 8K x 14 word kapasitesinde program hafızası (FLASH MEMORY)
- 368 x 8 byte veri hafızası (RAM)
- 256 x 8 byte kalıcı (non volatile) veri hafızası (EEPROM)
- 14 adet kesme (interrupt) kaynağı
- 8 seviyeli hardware stack memory
- Doğrudan / Dolaylı ve Bağıl Adresleme Modları
- Chip üzerinde kendi RC osilatörü olan Watchdog Timer (WDT)
- Programlanabilir kod koruma özelliği
- Güç tüketimini düşüren UYKU modu (SLEEP mode)
- Farklı osilatör seçenekleri (RC osilatör, kristal, rezonatör)
- 20 MHz maximum çalışma hızı
- Düşük güçlü ve yüksek hızlı CMOS FLASH/ EEPROM teknolojisi
- 2 pin ile devre içi (In Circuit) programlanabilme
- Yüksek kapasiteli port giriş/çıkış akımları – 25 mA sink / source

5.1.2. Çevre Birim Özellikleri

- 3 adet Timer / Counter
- Timer0 : 8 bit prescaler 8 bit Timer
- Timer1 : 16 bit Timer
- Timer2 : 8 bit prescaler ve postscaler Timer
- 2 adet Capture / Compare / PWM modülü
- 10 bit, 8 kanal ADC
- SSP, SPI ve I2C seri portları
- USART (Universal Synchronous Asyncronous Receiver Transmitter)



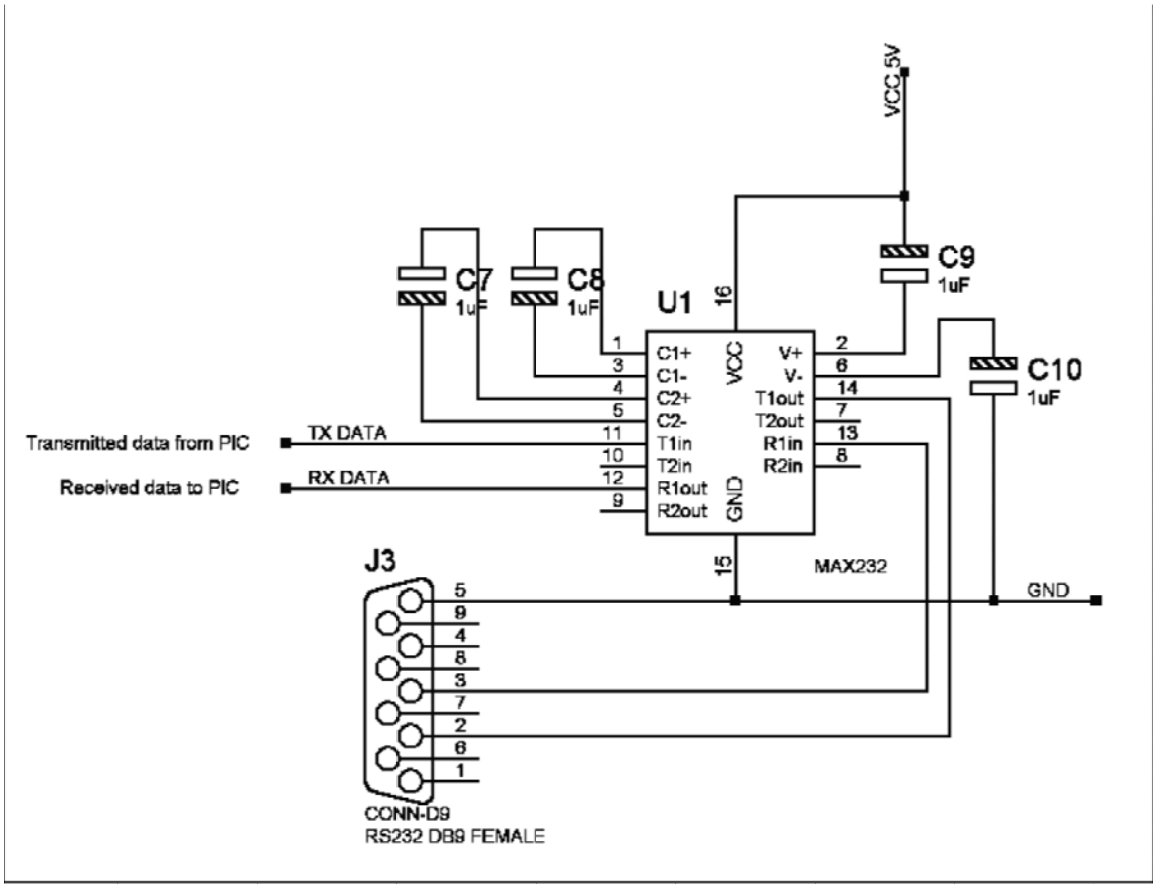
Şekil 5.1 PIC 16F876 Bacak Bağlantı Şeması

Tablo 5.1 PIC16F87x ailesi karşılaştırılmalı özellikleri tablosu

Key Features	PIC16F873A	PIC16F874A	PIC16F876A	PIC16F877A
Operating Frequency	DC – 20 MHz	DC – 20 MHz	DC – 20 MHz	DC – 20 MHz
Resets (and Delays)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)
Flash Program Memory (14-bit words)	4K	4K	8K	8K
Data Memory (bytes)	192	192	368	368
EEPROM Data Memory (bytes)	128	128	256	256
Interrupts	14	15	14	15
I/O Ports	Ports A, B, C	Ports A, B, C, D, E	Ports A, B, C	Ports A, B, C, D, E
Timers	3	3	3	3
Capture/Compare/PWM modules	2	2	2	2
Serial Communications	MSSP, USART	MSSP, USART	MSSP, USART	MSSP, USART
Parallel Communications	—	PSP	—	PSP
10-bit Analog-to-Digital Module	5 input channels	8 input channels	5 input channels	8 input channels
Analog Comparators	2	2	2	2
Instruction Set	35 Instructions	35 Instructions	35 Instructions	35 Instructions
Packages	28-pin PDIP 28-pin SOIC 28-pin SSOP 28-pin QFN	40-pin PDIP 44-pin PLCC 44-pin TQFP 44-pin QFN	28-pin PDIP 28-pin SOIC 28-pin SSOP 28-pin QFN	40-pin PDIP 44-pin PLCC 44-pin TQFP 44-pin QFN

5.1.3. MAX232 entegresi

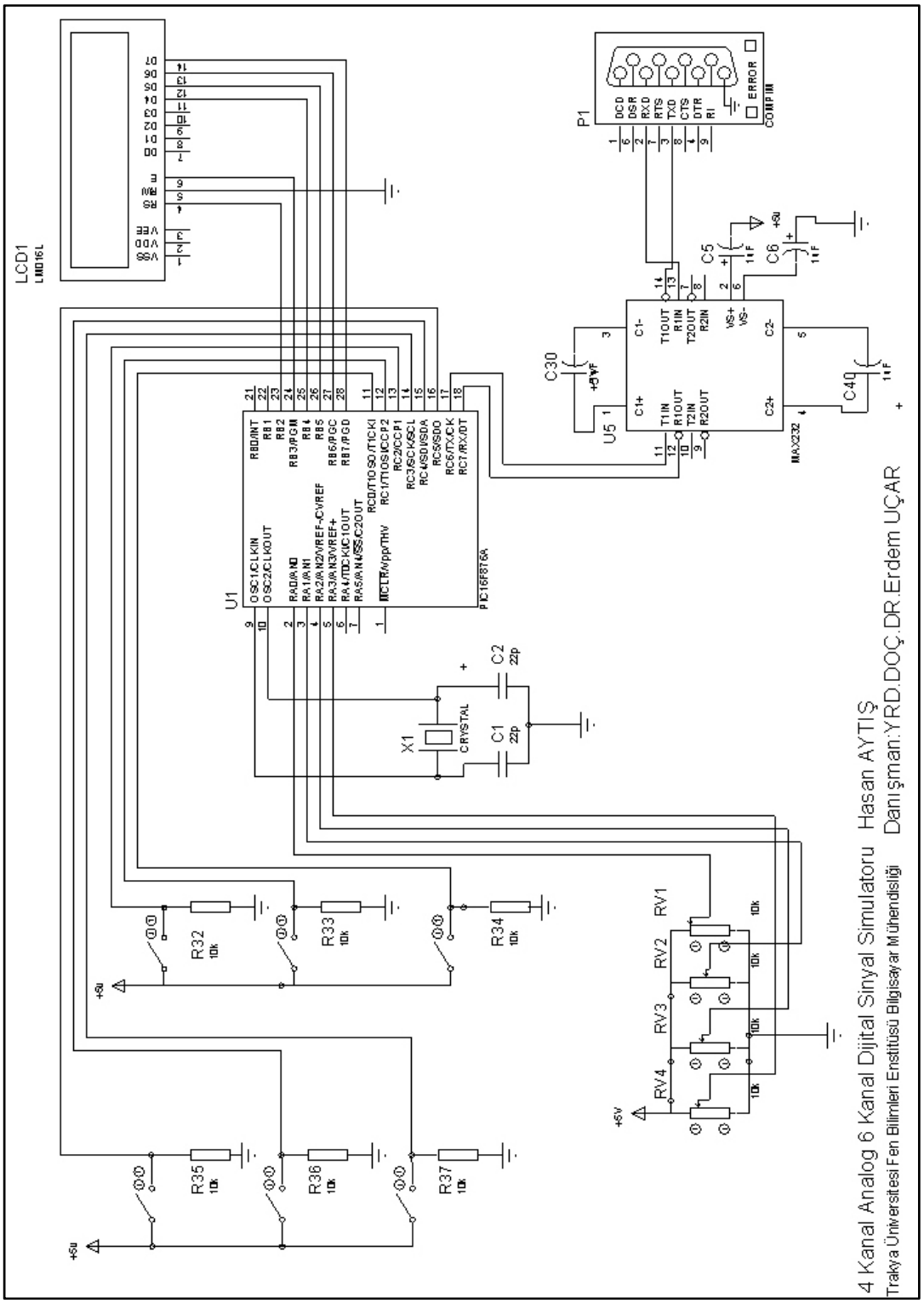
Devrede PIC ile bilgisayar haberleşmesi seri port üzerinden yapılmaktadır. Com port ların seviyesi RS232 standardında 10V civarındadır. Fakat PIC maksimum 5V çıkış vardır. Bu sebeple araya MAX232 Entegresi konularak seviye dengelenmiştir. MAX232 ile COMPORT bağlantı şeması Şekil 5.3 verilmiştir.



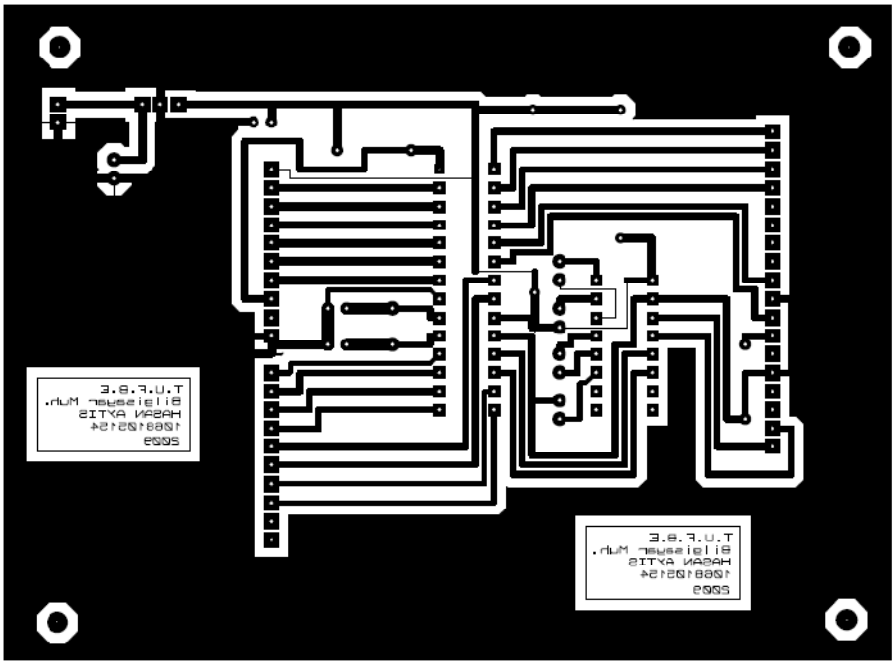
Şekil 5.2 MAX232 Entegresinin PIC ve COMPORT ile bağlantı şeması

5.1.4. Sinyal Simulator Devre Şeması

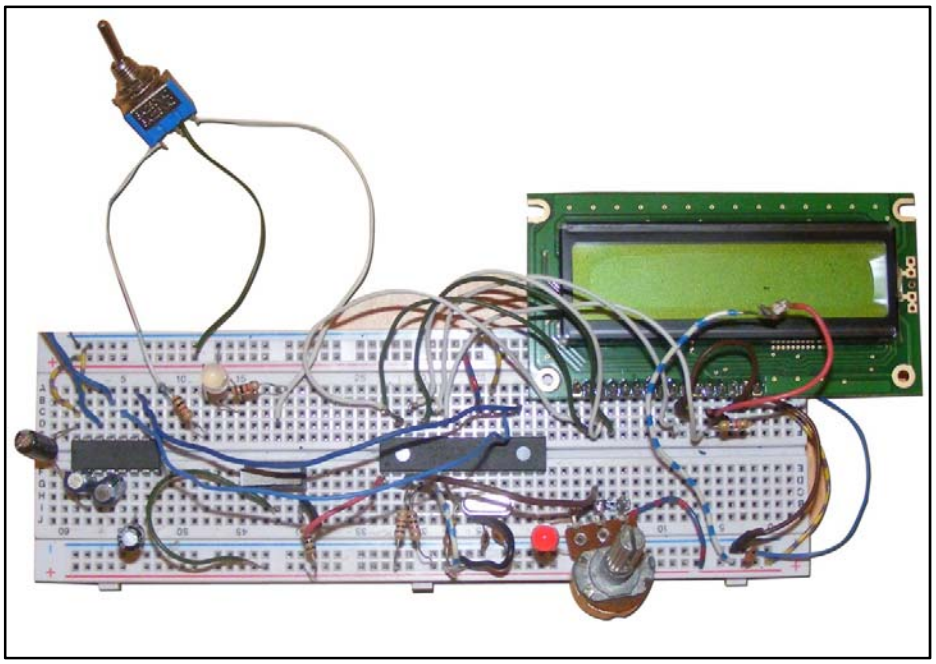
Simulator, gerçek ortamdan sensörler vasıtasıyla alınan sinyalleri simule ederek, uygulama geliştirme ve testlerimizde yardımcı olması için geliştirilimiştir. Simulatorde 4 kanal analog sinyal işleme, 6 kanal dijital sinyal giriş olarak işleme ve 2 kanal dijital sinyal çıkış olarak işleme görevi vardır. Okunan sinyallerin durumları 2 satır 16 karakter özellikli LCD ekran üzerinden mod düğmesi aracılığıyla görsel olarak da analog sinyal değerleri yüzde olarak, dijital sinyal değerleri 1 veya 0 olarak gösterilmektedir. Simulatorün şeması şekil 5.3 de verilmiştir.



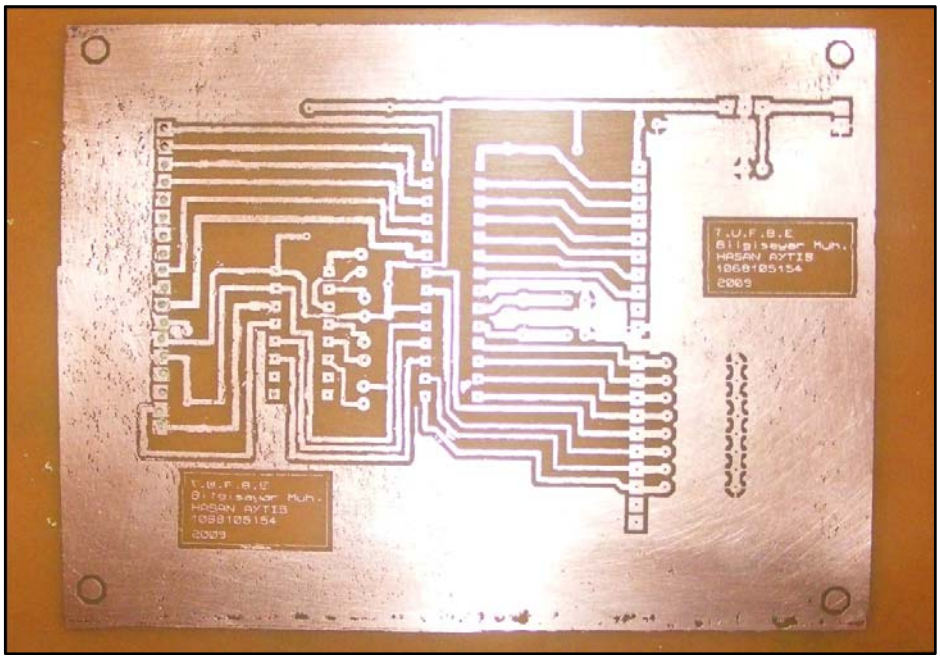
Şekil 5.3. Simulator Devre Şeması



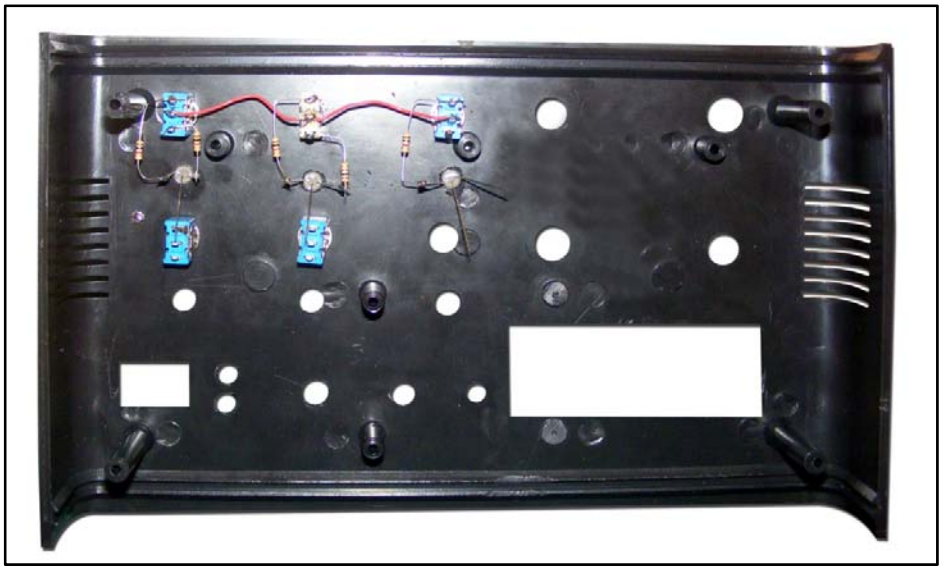
Şekil 5.4 Simulator baskı devre şeması



Şekil 5.5 Devrenin protoboardda test edilmesi.



Şekil 5.6. Baskı devre şemasının bakır plaketi



Şekil 5.7 Devreyi sabitleyecek kutunun montaj aşaması

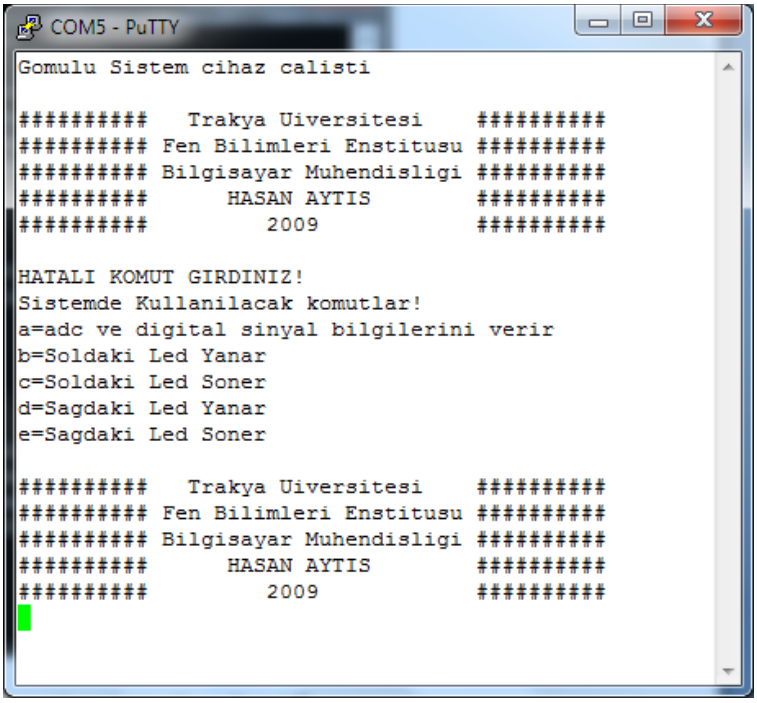


Şekil 5.8 Devrenin kutu montajı ilk enerji verilmesi

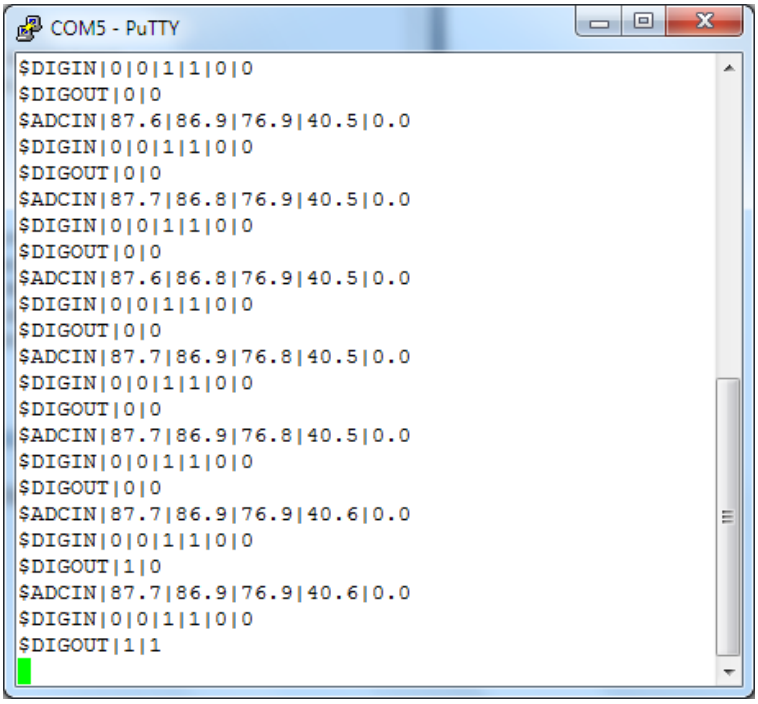
Simulator ile eBox2300 seri port üzerinden birbirileri ile haberleşmektedir. Simulator üzerindeki tüm analog ve dijital giriş çıkışları eBox2300 sorguladıkça cevap vermektedir. Sorgulama yöntemi şu şekildedir.

Simulator UART kesme yöntemi ile sürekli seri portu dinlemektedir. Seri porttan bir bilgi geldiği zaman hemen gelen bilgiyi analiz edip kendi anlayacağı dilde bir komut gelmiş ise komut işlevini yapar. Fakat farklı bir komut geldiği zaman seri port uzerinden yardım bilgisini gönderir.

Simulator seri prot üzerinden gelen sadece a, b, c, d, e komutlarında işlem yapmaktadır. Şekil 5.10 de verilmiştir.



Şekil 5.9 Simulator Terminal hata mesajı ve komutların listesi.



Şekil 5.10 “a” komutu verildikten sonra simulatorun cevap çıktısı.

Simulator Şekil 5.10'da görüldüğü gibi sensörlerin son bilgi durumlarını belli bir standarda göre vermektedir. $ işareti ve /r/n kullanılarak analog sinyal ve dijital sinyal bilgileri birbirleri ile karışmadan analiz edilmektedir.

$ADCIN : Analog giriş bilgilerini vermektedir. | işareti ile diğer analog sinyal bilgileri birbirinden ayrılmaktadır.

$DIGIN : Dijital giriş bilgilerini vermektedir. Her bir | işareti ile ayrılan değerler farklı dijital sinyalleri temsil etmektedir.

$DIGOUT : Aynı şekilde bu bilgi satır bize dijital sinyal çıkış değerini belirtmektedir.

5.2. C#.NET Bilgi Toplama Programı

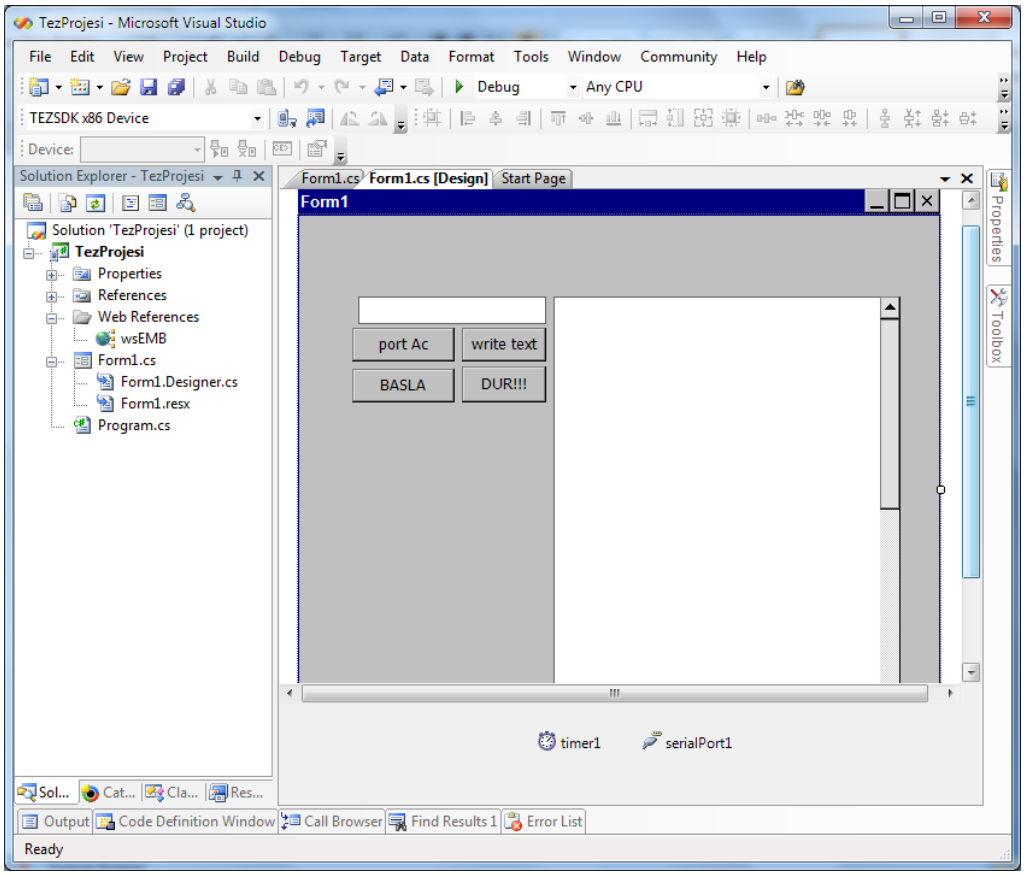
Bu yazılım CE 6.0'da RS232'den gelen analog ve dijital sinyallerin yorumlanıp sisteme internet üzerinden web servis arabirimi ile göndermesi sağlanmıştır.

Bir önceki bölümde tasarlamış olduğumuz simulatorun çalışma şekline göre C#:NET teknolojisi kullanarak iki farklı uygulama geliştirildi. Bu uygulamardan biri direk simulatore sorgu gönderip almış olduğu cevabı işleyerek hazırlamış olduğu bilgiyi diğer uygulama olan web servisine göndermektedir. Diğer uygulama olan web servisi ilgili metod çalıştırılarak hem anlık hemde sistemin çalışmaya başladığı andan itibaren XML formatında gelen bilgileri depolamaktadır.

5.2.1. Simulator ile haberleşme uygulaması

Uygulama Visual Studio 2005 IDE si kullanılarak C#.NET Smat Device Application olarak geliştirildi. Uygulamada timer modulu ve seriport modulu kullanılarak

gerçekleştirildi. Uygulamamızın aslında bir forma ihtiyacı yoktu fakat proje geliştirme aşamasında anlık müdaheleler yapabilmemiz için form kullanıldı ve bazı test değerleri için gerek gördüğümüz araçlar eklendi. Şekil 5.12 de uygulama geliştirme görüntüsü görülmektedir.



Şekil 5.11 eBox2300 de çalıştırılan uygulamanın geliştirilme aşaması

Uygulamaya ek olarak simulatorden almış olduğu bilgileri kayıt edebilmesi için hazırlamış olduğumuz web servis web referans olarak eklenmiştir.

5.2.2. Uygulamanın çalışması

Uygulamamız ilk çalıştığında hiçbir işlem yapmamaktadır.

Port Ac butonuna bastığımızda eğer seri port açık değilse açmaya çalışacaktır. Açık ise uyarı mesajı verecektir.

BASLA butonuna bastığımızda timer1 aktif hale getirip her timer1 interval değeri kadar bekleyerek GetDataFromPic() metodu tetiklenecektir. Timer1 in interval değeri 1000ms olarak set edilmiştir.

getDataFromPic metodu seriporttan gelen ham bilgileri 3 farkli dizi içine yerleştirip dizi halinde web servisin metoduna göndermektedir.

5.2.3. Web Servis uygulaması

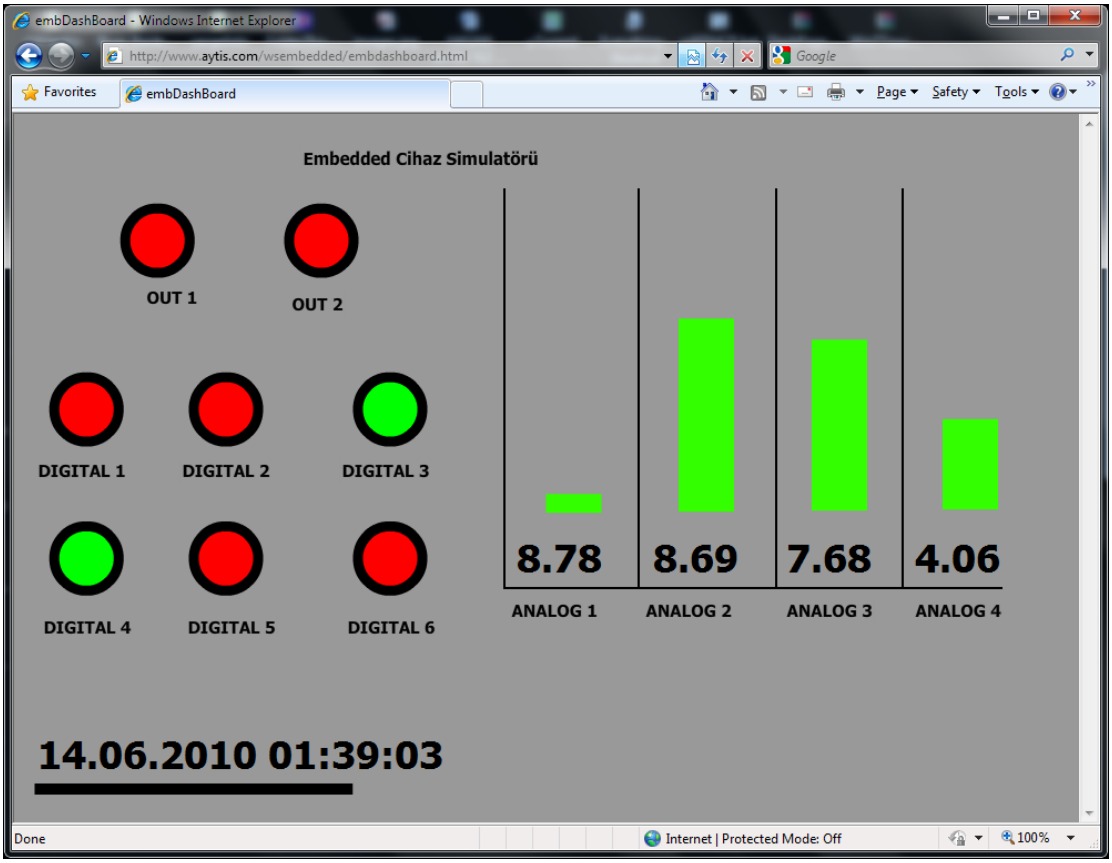
Web servisde 2 farklı web method yazılmıştır. Bunlar XmlYaz ve GelenToXML dir.XmlYaz metodu 3 tane dizi tipinde değişken almaktadır. Gelen bu diziler XmlTextWriter yönetimi ile web sunucusunda embeddedNow.xml isimli dosyaya yazılmaktadır. Eğer sistemin anlık bilgisini öğrenmek istiyorsak yani en son durumu nedir bilgisini öğrenmek istediğimizde embeddedNow.xml dokumanını okumamız yeterli olacaktır.

Web serviste XmlYaz metodundan başka gelenToXML metodu da eklenmiştir. Bu metodun eklenme amacı sistemin geçmiş bilgilerini tutmaktır. Sistem çalışmaya başladığı andan itibaren tüm değişiklikleri ve sistemin geçmiş bilgilerini analiz etmek istediğimizde bu xml dökümanını incelememiz yeterli olacaktır.

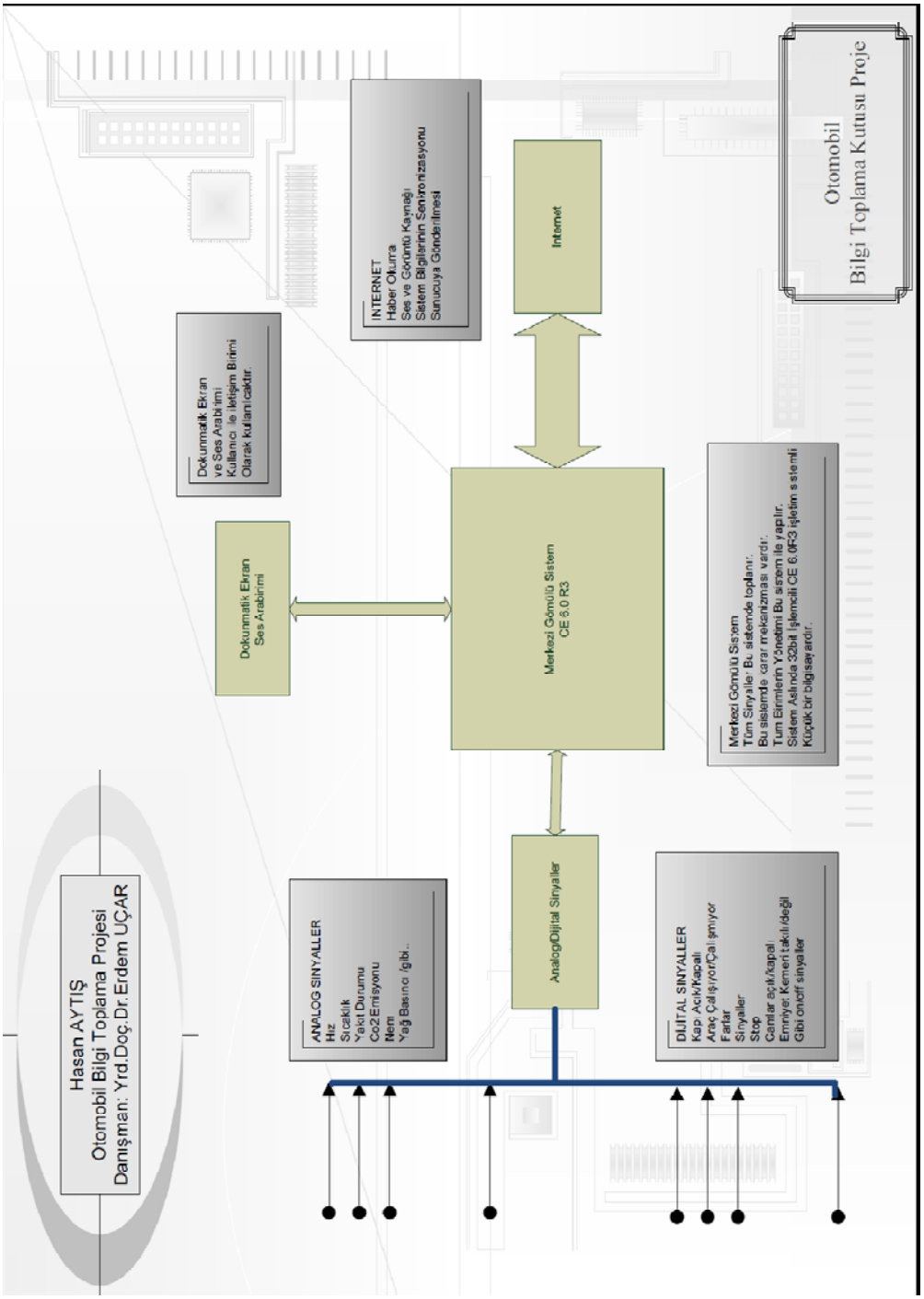
Bu uygulamara ek olarak web üzerinde çalışan bir gösterge paneli hazırlanmıştır.

5.2.4. Gösterge paneli

Gösterge paneli Adobe'nin Flash CS3 teknolojisi kullanılarak geliştirilmiştir.Action Script dilinde uygulama geliştirilerek arka planda her 2 saniyede bir xml dökümanı okuyarak ilgili panel bilgilerini güncellemektedir. Böylece sistemi internete erişelen herhangi bir yerden o anki durumu gözlemlenebilmektedir. Şekil 5.13 da gösterge panelinin ekran görüntüsü görülmektedir.



Şekil 5.12 Gösterge paneli ekran görüntüsü



Şekil 5.13 Sistemin akış dagramı

6. LİTERATÜR TARAMASI

6.1.Fiat ecoDrive

Fiat firmasının otomobiller için geliştirmiş olduğu Blue&Me uygulaması 2010 yılında tanıtılmıştır. Fiat, Blue&Me için kişiselleştirilebilir mobil aygıtların otomobillere entegrasyonu şeklinde tanımlamıştır. Blue & Me uygulaması ile sürücüler bluetooth bağlantısı veya USB bağlantısı kullanarak kendi cep telefonlarına bağlanabiliyorlar. USB bağlantı aracılığı ile sürücüler disklerindeki müzikleri ve videoları izleyebiliyorlar.Bunların yanında bazı özelleştirilmiş özellikler şunlardır.

Blue &Me Nav : Bu uygulama ses ile çalışmaktadır.İngilizce dil kullanılarak gidilmek istenen yer söylenir ve navigasyon programı sürücüyü sesli olarak yönlendirir.Bunun yanında SOS veya sigorta ile ilgili servislerde sürücüye yardımcı olmaktadır.

Fiat EcoDrive: Bu uygulama aracın performansı ile ilgili ve araçla ilişkili tüm bilgiui toplayarak Blue&Me USB kapısı aracılığı ile USB diske tüm verileri aktarır.Sürücü bu USB diski kendi PC sine takarak daha önce kurmuş olduğu programdan her bir seyahat için CO2 emisyon seviyesini görebilmektedir.Uygulama aynı zamanda kullanıcıya daha az CO2 emisyonu için nasıl aracı kullanması gerektiği konusunda bilgiler vermektedir.Bu bilgileri sürücünün daha az yakıt harcaması ve daha az ücretle seyahat etmesine yardımcı olmaktadır.[13]



Şekil.6.1. Fiat ecoDrive ekran görüntüs

6.2.Ford SYNC

Ford SYNC uygulaması otomotik olarak mikrofon ve ses sistemi ilebağlantılı cep telefonları ve medya oynatıcılarını ahizesiz olarak sesli aktif olarak kontrolünü sürüclere sağlamaktadır.Ford tarafından yayınlanan yazılım güncellemeleri USB port üzerinden sistem güncellenebilmektedir.Kullanıcılar kendi istedikleri gibi sistemi özelleştirebilmektedir. Sistemin öne çıkan özellikleri şunlardır:

Ford Work Solutions : Tam olarak yüksek hızda internet erişimi ve navigasyon sağlar. Evdeki veya ofisteki dökümanlara araç içinden erişimi kolaylaştırır.

SYNC Trafik, Talimatlar ve Bilgi: Gerçek zamanlı olarak trafik raporları,Haberler, hava durumu ve çeşitli özelleştirilebilir bilgilere erişimi sağlar.

911 Assist : Hiç bir obenelik gerektirmeden herhangi bir durumda hava yastığı açıldığı zaman otoamtik olarak mobil telefon üzerinden 911 aranarak konum bilgisi gönderilme hizmetidir.[14]



Şekil 6.2. Ford SYNC

6.3.Kia UVO

Kia UVO uygulaması Microsoft'un Speech Engine üzerine kurulmuştur. Sesli olarak komutları alıp işleyebilmesi yanında dokunmatik ekran üzerinden de işlemleri gerçekleştirebilmektedir.Kia UVO ile hızlı bir şekilde müzik dosyalarına erişmek, arka kamerayı kontrol etmek,ses ile telefon görüşmelerini başlatmak gibi birçok özelliği bulunmaktadır.Bazı özellikleri şunlarıdır:

Gelişmiş konuşma tanıma sistemine sahiptir. Microsoft'un Speech engine uygulaması sisteme entegre dilmiştir.

Geliştirilmiş doğal arayüze sahiptir.UVO gösterge ekranı tam renkli, kullanımı kolaydır.

Arka yedek kamera sadece UVO nun ekranında geniş bir açı ile otomobilin arkasını görmeye olanak sağlar.

Kia Motors tarafından yazılım sürekli güncellenmektedir.[15]



Şekil 6.3. Kia UVO

7. SONUÇLAR

7.1.Sonuçlar

Bu tez çalışmasında gömülü sistemler donanımsal olarak yetenekleri kullanım alanları ve üzerinde çalıştırılan işletim sistemlerinden biri olan Windows Embedded CE 6.0 işletim sistemi incelenmiştir. Gömülü sistem kullanılarak ortamdan bilgi toplama ve bu bilgilerin gösterilmesi ile ilgili elektronik ve web teknolojilerinden faydalanılmıştır.

Gömülü sistem kullanılarak oluşturulacak projelerde her durumun en baştan incelenip işletim sistemi ve uç birimler ona göre tasarlanmalıdır. Uygulama geliştirme aşamasında projenin hayatilik değerine göre uygulama dili seçilmelidir. Yönetilebilir diller ile uygulama geliştirme maliyeti düşük olmaktadır. Fakat bu avantajı yanında real-time işlemler yapmada başarısız olmaktadır. Eğer uygulama real-time çalışma anları gerekiyorsa doğal dil ile uygulama geliştirilmelidir.

KAYNAKLAR

[1] Hamblen, J.O, 2007,. Introduction to Embedded Systems using Windows Embedded CE ,Georgia Intstitute of Technology

[2] http://www.embedded.com

[3] Kahraman,Nihan,VLSI Tasarım http://www.yildiz.edu.tr/~nicoskun/VLSI%20TASARIM.ppt

[4] http://msdn.microsoft.com/en-us/library/ms836791.aspx

[5] http://www.embedded.com

[6] http://www.embedded.com/1999/9905/9905turley.htm

[7] http://www.embeddedpc.net/eBox2300/tabid/110/Default.aspx

[8] http://channel9.msdn.com/posts/scobleizer/Mike-Hall-Build-a-new-Windows-CE-operating-system-in-less-than-10-minutes/

[9] http://msdn.microsoft.com/en-us/library/ee482154.aspx

[10] http://www.faqs.org/docs/ldev/0130091154_263.htm What are Embedded Systems

[11] Microchip, PIC 16F87X Data Sheet, 2001

[12] http://focus.ti.com/lit/ds/symlink/max232.pdf

[13] http://www.fiat.co.uk/blueandme/

[14] http://www.fordvehicles.com/technology/sync/

[15] http://www.kiamotors.com/