

**T.C.
MİLLÎ EĞİTİM BAKANLIĞI**

**YENİLENEBİLİR ENERJİ
TEKNOLOJİLERİ**

İNTERNET ÜZERİNDEN KONTROL
522EE0363

Ankara, 2012

- Bu modül, mesleki ve teknik eğitim okul/kurumlarında uygulanan Çerçeve Öğretim Programlarında yer alan yeterlikleri kazandırmaya yönelik olarak öğrencilere rehberlik etmek amacıyla hazırlanmış bireysel öğrenme materyalidir.
- Millî Eğitim Bakanlığınca ücretsiz olarak verilmiştir.
- PARA İLE SATILMAZ.

İÇİNDEKİLER

AÇIKLAMALAR	ii
GİRİŞ	1
ÖĞRENME FAALİYETİ-1	3
1. İNTERNET ÜZERİNDEN İLETİŞİM.....	3
1.1. Soket Programlama Mantığı	3
1.1.1. Soket kavramı	3
1.1.2. Port kavramı	4
1.2. Görsel Programlama Soket Bileşeni	4
1.3. Diğer İnternet Haberleşme Yöntemleri.....	4
1.4. İki Bilgisayarın Haberleştirilmesi	5
1.4.1. Sunucu Programı Yapımı	6
1.4.2. İstemci Programı Yapımı.....	6
1.5. Programın Test Edilmesi.....	6
UYGULAMA FAALİYETİ	8
ÖLÇME VE DEĞERLENDİRME	10
2. İNTERNET ÜZERİNDEN KONTROL.....	38
2.1. Kontrol edilecek cihazın elektronik tasarımı	38
2.2. Sunucu Program.....	40
2.3. Sunucu Program Kodları.....	41
2.4. Cihazın Sunucu Üzerinden Kontrol Programı	45
2.5. İstemci Program	47
2.5.1. İstemci Program Tasarımı.....	47
2.5.2. İstemci Program Kodları.....	49
2.6. İnternet Üzerinde Kontrolde Güvenlik.....	53
2.6.1. Kullanıcı Adı Ve Şifreye Bağlı İnternet Kontrolü.....	53
UYGULAMA FAALİYETİ	54
ÖLÇME VE DEĞERLENDİRME	58
MODÜL DEĞERLENDİRME	60
CEVAP ANAHTARLARI	63
KAYNAKÇA	64

AÇIKLAMALAR

KOD	522EE0363
ALAN	Yenilenebilir Enerji Teknolojileri
DAL/MESLEK	Dal Ortak
MODÜLÜN ADI	İnternet Üzerinden Kontrol
MODÜLÜN TANIMI	İnternet üzerinden cihazların uzaktan kontrol becerisinin kazandırıldığı bir öğrenme materyalidir.
SÜRE	40/32
ÖN KOŞUL	Seri Port Kontrolü modülünü başarmış olmak
YETERLİK	İnternet üzerinden kontrol yapmak
MODÜLÜN AMACI	Genel Amaç İnternet üzerinden elektronik bir cihazı kontrol işlemini yapabileceksiniz. Amaçlar 1. İki bilgisayarı Ethernet ağı üzerinden haberleştirebileceksiniz. 2. Bir bilgisayardan Ethernet üzerinden diğer bilgisayarın seri portuna bağlı bir cihaza bilgi gönderebileceksiniz veya alabileceksiniz.
EĞİTİM ÖĞRETİM ORTAMLARI VE DONANIMLARI	Ortam: bilgisayar laboratuvarı, bilgisayar, elektrik elektronik atölyesi, baskı devre araç gereçleri, lehimleme araç gereçleri, avometre Donanım: Bilgisayar, ethernet bağlantısı, ethernet kablosu, elektronik kontrol ünitesi (seri/paralel/usb porta bağlı)
ÖLÇME VE DEĞERLENDİRME	Modül içinde yer alan her öğrenme faaliyetinden sonra verilen ölçme araçları ile kendinizi değerlendireceksiniz. Öğretmeniniz modül sonunda ölçme aracı kullanarak modül uygulamaları ve modül değerlendirme soruları ile kazandığınız bilgi ve becerileri ölçerek sizi değerlendirecektir.

GİRİŞ

Sevgili Öğrenci,

Yenilenebilir enerji teknolojileri, günümüzün en büyük sorunu olan enerji ve çevre problemlerimize çözüm olacaktır. Gelişmekte olan ülkemizin elektrik üretiminin yarısı doğalgazla üretilmekte olup, dışa bağımlı bir yapıdadır. Elektrik üretiminin bir bölümü de kömürle yapılmakta olup, çevreyi kirleten bir yapıya sahiptir.

Ülkemizde elektrik ücretlerinin düşmesi, çevrenin korunması, enerjinin bağımsızlığı, yenilenebilir enerjilerin üretimine bağlıdır. Almanya, İspanya, Amerika ve Çin gibi gelişmiş ülkeler bunun önemini kavramış. Bu çerçevede hem elektrik üretim tesisleri kurmuş, hem de kullanılan elektriksel üreteçleri üretmiştir. Enstitüler ve üniversiteler ile bu teknolojilerini sürekli geliştirmektedirler.

Yenilenebilir enerjideki tüm teknolojiler yukarıda saydığımız ülkeler tarafından üretilmekte olup, Türkiye ve birçok ülke bu ürünleri hazır olarak almaktadır. Bu ürünlerin dışarıdan ithali de bir problemdir. Bu problemin çözümü için ODTÜ ve Ege Üniversitesi başta olmak üzere üniversitelerimizde deneysel üretim çalışmaları yapılmaktadır. Elektrik üretim uygulamalarını ve güneş enerjili araba uygulamaları ile de yenilenebilir enerji üretim sistemlerinin incelenmektedir.

Türkiye'deki bu dışa bağımlı yenilenebilir enerji ürünlerinin üretimi konusunda Enerji Bakanlığı yerli üretim ürün (yazılım, mekanik yapı v. b. uygulamalar için) kullanımına destek vermektedir.

Bu alanda yerli üretim yapılarak, elektrik üretimi yapıldığında elektrik ücretleri oldukça makul seviyeler incek ve çevre kirliliği azaltılmış olacaktır. Dünyadaki enerji yarışında Türkiye'nin de bir üretim ve ihraç merkezi olması sağlanabilecektir.

Yenilenebilir enerjilerle elektrik üretimi şehir merkezinde olabildiği gibi genellikle dağlara ve denizlere yakın konumda veya kırsal kesimlerde yapılmaktadır. Bu yapı gereği bu sistemlerin enerji üretim durumu ve çalışma durumlarının kontrolü için uzaktan izlenmesi gerekmektedir.

Bu izleme sistemleri standartlara uygun hazır olarak ithal edilmektedir. Bazı üniversitelerde bu sistemleri test amaçlı olarak kendileri yapmaktadır.

Bu modül ve bilgisayar kontrol derslerinin diğer modülleri ile yenilenebilir enerji ile ilgili uzaktan izleme işlemlerini yapabileceksiniz.

ÖĞRENME FAALİYETİ-1

AMAÇ

İnternet üzerinden iki bilgisayarın haberleşmesini sağlayabileceksiniz.

ARAŞTIRMA

Bu faaliyet öncesinde yapılacak araştırmalar şunlardır:

- Görsel programla dilleri için socket (yuva) ile ilgili internet veya programın yardım konularında bilgi araştırması yapınız.
- İki bilgisayarın internet üzerinden haberleşmesi ile ilgili kaynak araştırması yapınız.
- Sunucu (server) ve istemci (client) kavramlarını öğreniniz.

1. İNTERNET ÜZERİNDEN İLETİŞİM

1.1. Soket Programlama Mantığı

Bilgisayardaki iletişim için kullanılan bir kavram ve komut yapısıdır. Soketler, aynı veya farklı hostlar üzerindeki süreçlerin haberleşmesini sağlayan bir haberleşme (interprocess communication) yöntemidir.

Bu yöntemde değişik portlar kullanılır. Burada soketlerin alacağı veri veya göndereceği veri donanımına ait veya bilgisayarınıza ait bir veri dosyadır. Windows iletişim alt yapısında Yuva (socket) mantığı ile çalışır. İşletim sisteminiz iletişim için bazı yuvaları görevlendirir. Bu yuvalar dinleme durumunda olup, hazır beklerler. İletişim talebi geldiğinde gerekli koşullar sağlandığında iletişimi ve veri alışverişini başlatır.

1.1.1. Soket kavramı

Veri alma, tanımlama gibi birçok iletişim görevi yuva kavramı ile yerine getirilir. Her görev için yuva tanımlanır.

Yuva çeşitleri stream, datagram, raw olmak üzere 6 çeşittir. (Stream soketler verileri sıralı gönderir, datagram soketleri bir bütün olarak gönderir.). Ayrıntılı bilgi için kaynakçadan ilk kaynağa bakınız.

1.1.2. Port kavramı

Portlar işletim sistemi, uygulamaların ve protokollerin ortak kullandıkları çıkış birimleridir. Bir bilgisayarda port sayısı 65000 civarındadır. 1023'e kadar olan portlar bilinen portlardır. Örneğin; Http:(yani web) 80.porttan, Ftp:21'den, Telnet 23'den vs.

Bunlar dünyada standart olarak kabul edilmiş numaralardır. İki bilgisayarın haberleşmesi için aynı port numarasını kullanmalıdır.

1.2. Görsel Programlama Soket Bileşeni

Görsel programlamada programlama yazılımları, Yuva nesnesi (socket object) ile çıkışların (portların) kontrol edilmesine internet üzerinden haberleşme ve veri gönderimine olanak verir. Yapılacak işlemler için ayrı ayrı yuvalar (socket) oluşturulur.

Visual Basic, Java programlama dilinde yuva nesnesi mevcuttur. Visual Basic'te bu bileşen WinSock ve System.Net.Sockets'dir. Java'da da socket komutu kullanılır.

Örneğin, Java'da bir yuva komutu;
Socket[addr=/127.0.0.1,port=1267,localport=8080]

1.3. Diğer İnternet Haberleşme Yöntemleri

Günümüzde kullanılan internet haberleşme yöntemleri Ethernet, Token ring, ATM, FDI, X.25, Frame Relay, DSL, VDSL'dir.

Şu an en yaygın kullanılan ethernet'tir. Kullanımı, kurulumu ve maliyetinin düşüklüğü nedeniyle, en basitten en karmaşık ağlara kadar kullanılır.10 Mbps hızına ulaşılabilir.

"Jetonlu Halka" (Token Ring) sistemde düğümler, halka şeklinde bağlanır. Veri aktarım hızı, 4 veya 16 Mbps olabilir.

ATM, bağlantı temelli çalışan, ses, video ve veri iletişimi için kullanılır. Verileri sabit uzunluklu hücreler halinde, kullanıcı sayısından bağımsız olarak, yüksek hızlı ileten teknolojidir (Kaynak 4).

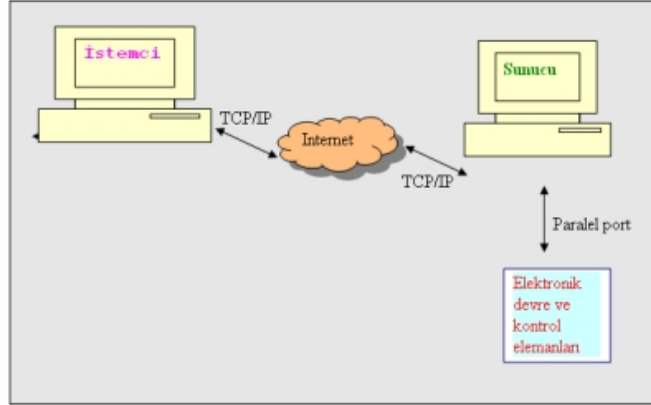
FDDI, halka geçirme (çift halkalı), yüksek hız sağlayan, fiber optik kablo ile iletişim kurulan yapıdır. 2- 100 Mbps veri aktarım hızına ulaşılabilir. İletim daha güvenilirdir.

Frame Relay, paket anahtarlamalı bir "Wan" teknoloji olup, noktadan noktadır. X.25'in geliştirilmiş halidir.

DSL, bakır kablo üzerinden yüksek hızlı iletim yapılabilen teknolojidir.1,5-8 Mbps veri hızına ulaşılabilir.

1.4. İki Bilgisayarın Haberleştirilmesi

Bilgisayarların haberleşmesi için sunucu ve istemci bilgisayar çıkışları (portları) kullanılır. Bu portlar iletişim için önceden tanımlanır. Sunucu bilgisayar portlarını açık tutup, dinleme yapar.



Şekil1.1. İki bilgisayarın haberleşmesi

İstemciler portları ile talepte bulunur ve sunucuya bağlanır. Bağlantı gerçekleştikten sonra iletişim tercihine göre ileti veya veriler TCP ise güvenli ve parça parça veya UDP ise güvenliksiz ve tek bir yapı halinde gönderilir.

Sunucu ile istemci arasında iletişim bittiğinde, aradaki bağlantı sonlandırılır. Yuva programlama mantığı sunucu ve istemci için ayrı ayrı yazılır.

Sunucu dinleme, bağlantı kurma, veri hizmeti verme işlemleri için yuva nesnesindeki komutları kullanırken, istemci sunucuya istek gönderme ve veri alma işlemleri için yuva nesnesindeki komutları kullanılır.

Visual Studio içinde Winsock programı eklenerek, alet araç kutusunda aktif edilir. Winsock programı, bilgisayardaki yuva(socket) kavramını kontrol etmek için kullanılır. Visual Studio 2010'da, Windows Sockets 2 (Winsock), kablo üzerinden Internet, intranet ve diğer ağ uygulamaları yapılabilir. Winsock, ağ uygulamaları için, C/C++ programcıları için tasarlanmıştır.

COM tabanlı Winsock kontrolü yerine, System.Net, System.Net.Sockets, kullanılması daha uygundur.

Yeni Windows yuva yapısında, güvenlik, grup ve sorgu gibi yapılar eklenmiştir. Yüksek hız içinde yapılar oluşturulmuştur. Win7 ve Server 2008 de IPv6 içinde destekler mevcuttur. Daha detaylı bilgi için kaynakçadaki Microsoft'un sitesine bakılabilir.

1.4.1. Sunucu Programı Yapımı

Sunucu dinleme, bağlantı kurma, veri hizmeti verme işlemleri için yuva nesnesindeki komutları kullanır.

Sunucu programı, istemci ile haberleşme ve veri iletimi için aşağıdaki işlemleri yapar:

- Winsock veya System. Net.Sockets eklentisi aktif edilir.
- Her işlem için yuvalar yapar, başlatır.
- Yuveyi bağlar.
- İstemciyi yuvada dinler.
- İstemciyle bağlantı talebini onaylar.
- Veri gönderimi ve alımı yapılır.
- Bağlantı kesilir.

Yukarıdaki işlem durumuna göre, sunucu programı kitapçığın ilerleyen kısmındaki gibi oluşturulur.

1.4.2. İstemci Programı Yapımı

İstemci programı, sunucuya istek gönderme ve veri alma işlemleri için yuva nesnesindeki komutları kullanılır.

İstemci, Sunucu ile haberleşme ve veri iletimi için aşağıdaki işlemleri yapar:

- Winsock veya System.Net.Sockets eklentisi aktif edilir.
- Her işlem için yuvayı yapar, başlatır.
- Sunucuya bağlanır.
- Veri gönderimi ve alımı yapılır.
- Bağlantı kesilir.

Yukarıdaki işlem durumuna göre, istemci programı kitapçığın ilerleyen kısmındaki gibi oluşturulur.

1.5. Programın Test Edilmesi

Yazılan programın test edilmesi aşağıdaki şekilde yapılır:

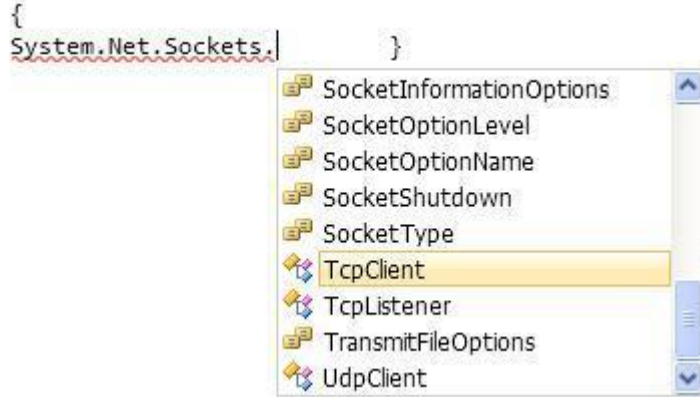
- Elektronik devre, sunucu bilgisayara seri, paralel veya USB port üzerinden bağlanır.
- Sunucu programın exe uzantılı çalıştırılabilir dosyası çalıştırılır. Sunucu beklemeye geçer.

- İstemci programın exe uzantılı çalıştırılabilir dosyası çalıştırılır. İstemci üzerindeki sunucu IP ve port bilgisi kontrol edilir. Bilgisayar hem sunucu hem de istemci olarak çalışacaksa, bilgisayarın IP adresini kendisi bulan GetIP() metodu, IP numarasını bulup, ilgili kutuya yazar. Eğer, sunucu olarak internete bağlı farklı bir bilgisayar kullanılıyorsa bu bilgisayarda MS-DOS promptta C:/>ipconfig yazarak bilgisayarın IP adresini öğrenip, istemciye kutuya girmemiz gerekmektedir. Port numarası olarak 8000 alınmıştır.
- Sunucuya bağlan düğmesine tıklanır.
- İstemci kullanıcı formunda “Bağli” yazısı gözükür.
- İstemci üzerinden step motor verisi girilir. Çalıştır butonuna tıklanır. Step motorun sunucu tarafında çalıştığı görülür.
- İstemci üzerindeki led lambaların olduğu düğmelere tıklanarak, led lambaların sunucu tarafında çalıştığı gözlemlenir.

UYGULAMA FAALİYETİ

Bu uygulama ile Visual Studio içinde socket yapısını tanıyacaksınız.

İşlem Basamakları	Öneriler
<ul style="list-style-type: none">➤ Visual Studio programı çalıştırılır. Yeni – Proje - Visual C# - Konsol Uygulaması (Console Application) açılır.➤ Kod kısmına System.Net.Sockets yazılır.➤ Sockets nesnesinin altındaki komutlar incelenir. Sockets yazıldıktan sonra “.” Noktaya basıldığında alt komutlar çıkmaktadır.	<ul style="list-style-type: none">➤ Sockets komutları ile ilgili Microsoft’un sitesinden bilgi alınız.➤ Yine bu komutlarla ilgili internetten uygulamalar bularak, komutların görevini anlamaya çalışınız.



Visual Studio’da yuva(socket) nesnesi

İkinci uygulama ile sunucu ve istemci programların çalıştırılıp, test edilmesini öğreneceksiniz.

İşlem Basamakları	Öneriler
<ul style="list-style-type: none">➤ Programın kodlarını kaynakçadaki en son kaynaktaki web sitesinden indiriniz.➤ “Server control” dosyasındaki Server – Bin - Debug içindeki Server.exe dosyasını çalıştırınız.➤ Çıkan “Socket Server” formundan dinlemeye başla butonuna tıklayarak, dinlemeye başlayınız.➤ “Client control” dosyasındaki Client – Bin - Debug içindeki Client.exe dosyasını çalıştırınız.➤ User Control formundaki IP numaranız v Port bilginizi kontrol ediniz. Doğruysa “Sunucuya Bağlan” butonuna basınız.➤ İstemci Sunucuya bağlandığında İstemci formunda “Bağli” yazısını göreceksiniz.➤ Şimdi step motor adım sayısını kontrol ettikten sonra, “Motoru Çalıştır” butonuna basınız.	<ul style="list-style-type: none">➤ İnternetten seri, paralel ve USB port ile elektronik devre kontrolü uygulamalarını bularak, programları deneyiniz.

KONTROL LİSTESİ

Bu faaliyet kapsamında aşağıda listelenen davranışlardan kazandığınız becerileri Evet, kazanamadığınız becerileri Hayır kutucuğuna (X) işareti koyarak kendinizi değerlendiriniz.

Değerlendirme Ölçütleri	Evet	Hayır
Sockets nesnesini oluşturabildiniz mi?		
Sockets nesnesinin altındaki komutların işlevini anladınız mı?		
Sunucu ve istemci .exe dosyalarını çalıştırabildiniz mi?		
Sunucu ve istemci bilgisayarları birbirine bağlayabildiniz mi?		

DEĞERLENDİRME

Değerlendirme sonunda “**Hayır**” şeklindeki cevaplarınızı bir daha gözden geçiriniz. Kendinizi yeterli görmüyorsanız öğrenme faaliyetini tekrar ediniz. Bütün cevaplarınız “**Evet**” ise “Ölçme ve Değerlendirme”ye geçiniz.

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki çoktan seçmeli soruları dikkatlice okuyunuz ve doğru seçeneği işaretleyiniz.

1. Port aşağıdaki ifadelerden hangisi ile doğru ifade edilmiştir?
 - A. Port sadece kavramdır.
 - B. Port bir komut olup, kontrol için kullanılır.
 - C. Port fiziksel bir yapı olup, iletişim için kullanılır.
 - D. Hiçbiri.
2. Aşağıdakilerden hangisi internetle haberleşme yöntemlerinden **değildir**?
 - A. Ethernet
 - B. Token ring
 - C. Bus
 - D. ATM
3. Aşağıdakilerden hangisi güvenli veri iletim protokolüdür?
 - A. UD
 - B. TCP
 - C. Winsock
 - D. OS
4. İstemci programı oluşturulurken yapılacak işlemler aşağıdaki hangi seçenekte doğru olarak verilmiştir?
 - I- Winsock veya System.Net.Sockets eklentisi aktif edilir.
 - II- Her işlem için yuvayı yapar, başlatır.
 - III- Veri gönderimi ve alımı yapılır.
 - IV- Sunucuya bağlanır.
 - A. I-II-III-IV
 - B. I-II-IV-III
 - C. IV-III-II-I
 - D. IV-II-I-III

DEĞERLENDİRME

Cevaplarınızı cevap anahtarıyla karşılaştırınız. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt ettiğiniz sorularla ilgili konuları faaliyete geri dönerek tekrarlayınız. Cevaplarınızın tümü doğru ise bir sonraki öğrenme faaliyetine geçiniz.

ÖĞRENME FAALİYETİ-2

AMAÇ

İnternet üzerinden sunucu programı ve istemci programı yazabilecek ve sunucuya bağlı elektronik devreyi istemci ile internetten kontrol edebileceksiniz.

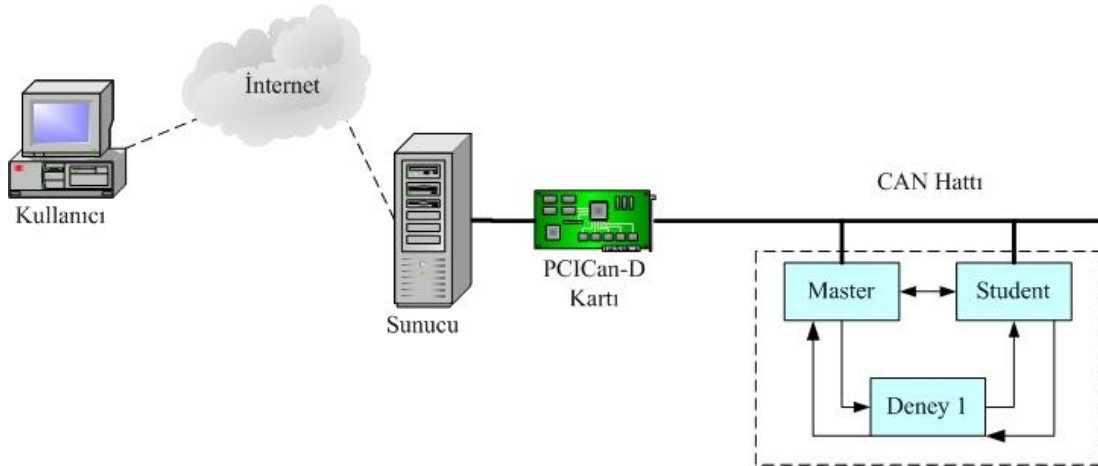
ARAŞTIRMA

Bu faaliyet öncesinde yapılacak uygulama ve araştırmalar şunlardır:

- Elektronik devre çizim (ISIS) ve baskı devre çıkartımı (ARES) için “Bilgisayarlı Devre Dizaynı Modülleri okuyarak uygulamalar yapınız.
- Paralel ve seri portla bilgisayarla iletişim kurabilen elektronik devreleri inceleyiniz.
- İki bilgisayarın internet üzerinden haberleşmesi ile ilgili kaynak araştırması yapınız.
- Sunucu (Server) ve istemci (Client) kavramlarını öğreniniz.

2. İNTERNET ÜZERİNDEN KONTROL

2.1. Kontrol edilecek cihazın elektronik tasarımı



Şekil 2.1: İnternette kontrol için örnek bir uygulama

Kontrol edilecek cihaz seri, paralel, USB port veya bilgisayar slotları kullanılarak, bilgisayara bağlanabilir.

Devre tasarımı Proteus programı ile yapılır. Baskı devresi oluşturularak, ütü yöntemi ile baskı devresi çıkartılır. “Bilgisayarlı Devre Tasarımı” dersine ait kitapçıklardan yararlanılabilir.

Seri port ve paralel port için bu dersin bir önceki kitapçıkları “Seri Port Kontrolü” ve “Paralel Port Kontrolü” kullanılmalıdır. USB Port kontrolü bir sonraki adımda yapılması daha uygun olacaktır.

Bu yöntemlerden aşağıdaki örnekte elektronik devre paralel portla bilgisayara bağlıdır.



Şekil 2.2: Paralel port ile kontrol için elektronik devre

Yukarıdaki şekildeki devrede:

Normal olarak, bir step motoru sürmek için 4 kontrol kanalına gereksinimimiz olur fakat burada özel tümleşik devre gerektirmeyen, sadece çıktı kanallarına normal transistör ve koruma diyotları bağlayarak gerekli akımı oluşturup, bir adım motorunu sürme devresi kullanılmıştır.

Bu devrede aynı zamanda 8 led kullanılarak, paralel portun 8 ucuna bağlanmıştır. İstemciden gelen her led için gelen bilgi ile led lambalar yakılmaktadır.

Donanım tarafında devrenin çalışması kontrol edildikten sonra yazılıma geçilebilir.

Kontrol devresi için bir paralel porta sahip, sunucu bilgisayar, bir istemci bilgisayar ve her iki bilgisayarın da internete veya bir ağ üzerinden birbirine bağlı olması gerekmektedir.

Yazılımda programlama için Visual Studio programı kullanılmıştır. Programlama için farklı programlar ve programlama yöntemleri kullanılabilir.

İşletim sistemi için sunucu ve istemcide Windows XP kullanılmıştır.

Sunucu ve istemcide kullanılacak programların Visual Studio'daki kodları aşağıda verilmiştir.

2.2. Sunucu Program

Burada, belirli bir port numarası üzerinden, yaratılan bir Socket nesnesi aracılığıyla istemci dinlenir.

Bağlantı sağlandığında çalıştırılmak üzere OnClientConnect adlı Callback metodu bulunmaktadır.

Bu metot, bağlantı sağlandıktan sonra istemciden gelen bilgileri WaitForData metoduyla dinlemektedir. Bunun içerisinde de bir diğer callback metodu olan OnDataReceived callback metodu bulunmaktadır. Bu metot ile gelen bilgilerin tamamını yakalanıp değerlendirilmek için yazılmıştır. Bu metot içerisinde, istemciden gelen verinin hangi paralel port gurubuyla ve/veya kanalıyla ilgili olduğunu anlamak için başlangıçta belirlediğimiz protokol ön ve son eklerine bakılır.

Socket nesnesinin tampon alanından alınan veri önce karakter dizinine sonra da String nesnesine çevrilir. Bu nesne içerisinde sırasıyla DPort, CPort ve SPort ön ve son eklerine bakılır. Bu ekler arasındaki bilgi alınarak ilgili metoda parametre olarak geçirilir.

Örneğin, DPort ekleri bulunursa, bunların arasındaki numara (1-8 arası), hangi portun “toggle” edilmesi gerektiğini bildirir.

Inp Out sınıfındaki, Toggle Data Port Bit (adres, bit) metoduna verilen bu numara, sunucunun ilgili “bit”i toggle etmesini sağlar.

Gelen bilgi Control Portu ile ilgili ise;

Bu porta bağlı step motor kontrol devresini çalıştırmamız gerektiği anlamını verir. Veri olarak CPort ekleri arasında bulunan adım sayısı ele geçirilir; InpOut.StepMotorCont (controlportadres, beklemezamanı) metodu ile, bu sayı kadar for-next döngüsü içerisine alınır.

Step motor, Kontrol portunun 2 kanalı yardımıyla kontrol edilmektedir. Kanallardan birisi(0.bit>bit ağırlığı:1) motorun dönüş yönünü, diğeri(1.bit>bit ağırlığı:2) ise adımlama işini gerçekleştirmektedir.

Bu devreye gönderilen her bir yazma işlemi sonrasında, yazılan verinin tam olarak yerine ulaşabilmesi bilgisayarınızın hızına göre bir miktar bekleme süresi konur. Bunun için bir delay (int t) metodu yazılmıştır. Verilen parametre sayısı kadar for-next döngüsü yapan bir metod.

Aslında bu rakam, motor hızını ayarlamak için de bir parametre olabilir. Sunucuya gelen bilgi "SPort" içeriyorsa, sunucu StatusPort'un byte değerini okur ve istemciye gönderir. stPortData=InpOut.ReadStatusPort(0x378)

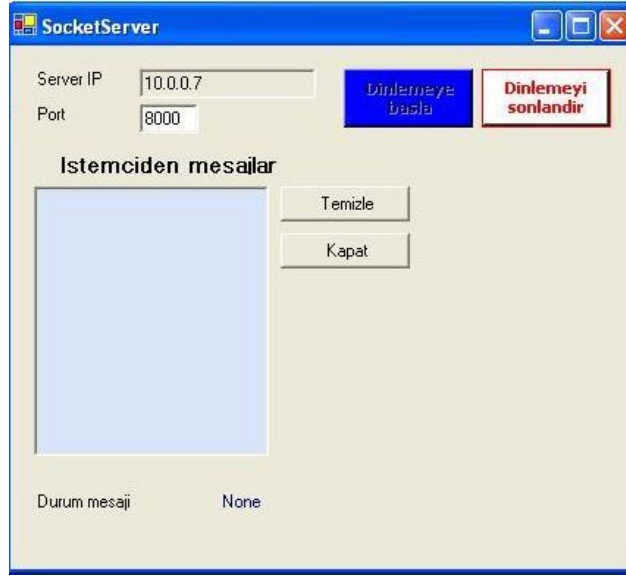
İstemciye veri, SendStatusPortData(string msg,System.Net.Sockets.Socket soc) metodu aracılığıyla gönderilir.

2.3. Sunucu Program Kodları

Visual Studio'da C# konsol uygulaması ile yeni bir proje açılır.

Sunucu programında önce kütüphane dosyaları eklenir. Yuva (Socket) programlama kullanımı için System.net. Sockets kullanılır. İstemci ile veri iletişimi için Sytem.Data uzay isimleri (kütüphaneler) eklenir. Sunucuya bağlı elektronik kart paralel portla iletişim kurduğu için paralel port uzay isimleri eklenir.

```
using System;
using System.Windows.Forms;
using System.Net;
using System.Net.Sockets;
using System.Collections;
using System.Threading;
using System.Text;
using System.Text.RegularExpressions;
using System.Data;
using System.Data.SqlClient;
using ParalelPortMethods;
```



Şekil 2.3: Sunucu kontrol programı formu

Yukarıdaki şekildeki forma göre formlar oluşturulur.

Sunucu IP adresinin alınması için aşağıdaki kod eklenir. Sunucu IP adresi alınır, ilk IP alınır. Sunucu adresi, IP adresi olarak, tanımlanır.

```
String GetIP()
{
    String strHostName = Dns.GetHostName();
    IPHostEntry iphostentry = Dns.GetHostByName(strHostName);
    String IPStr = "";
    foreach(IPAddress ipaddress in iphostentry.AddressList)
    {IPStr = ipaddress.ToString();
     return IPStr;} return IPStr;}
}
```

IP adresi, elde edilen IP adresine eşitlenerek, form üzerinde ilk açılışta gözükmesi için aşağıdaki kodla gösterilir.

```
textBoxIP.Text=GetIP();
```

Sunucu görsel formun çalışması için aşağıdaki kod programa eklenir.

```
public static void Main(string[] args)
{Application.Run(new SocketServer());}
```

İletişim için kullanılacak port bilgisi sunucu görsel formundan alınır. Port olarak tanımlanır.

```
if(textBoxPort.Text == "")
{MessageBox.Show("Port numarası girmeniz gerekiyor");
 return;}
string portStr = textBoxPort.Text;
int port = System.Convert.ToInt32(portStr);
```

İstemciyi dinlemek için yuva (Socket) oluşturulur. Kullanılacak yöntem (Stream) ve protokol (TCP) belirlenir.

```
m_mainSocket = new Socket(AddressFamily.InterNetwork,
    SocketType.Stream,
    ProtocolType.Tcp);
IPEndPoint ipLocal = new IPEndPoint (IPAddress.Any, port);
```

Yuva bind ile IP adresi ile bağlanır. Yuva listen komutu ile port dinlemeye geçilir.

```
m_mainSocket.Bind(ipLocal);
m_mainSocket.Listen(4);
```

İstemci bağlantısı için asenkron bağlantı metodu (sorgulardan cevap alınmadan diğer kodların çalışma imkânı veren yapı) kullanılır.

```
m_mainSocket.BeginAccept(new AsyncCallback (OnClientConnect),
null);
UpdateControls(true);
```

Yeni bir yuva(Socket)için m_mainSocket'i sonlandırılır.

```
Socket workerSocket = m_mainSocket.EndAccept (asyn);
WaitForData(workerSocket);
```

SocketPacket ile istemciden gelen bilgi için yeni bir yuva tanımlanır. İstemciden gelen bilgi byte olarak tutulur.

```
public SocketPacket(System.Net.Sockets.Socket socket)
    {m_currentSocket = socket;}
public System.Net.Sockets.Socket m_currentSocket;
public byte[] dataBuffer = new byte[1024];
```

Waitfordata ile istemciden veri beklenir. İstemciden veri geldiğinde asenkron bağlantı metodu kullanılır. TheSocpkt ile databuffer'a bilgi aktarılır.

```
public void WaitForData(System.Net.Sockets.Socket soc)
    {try{if ( pfnWorkerCallBack == null )
        {pfnWorkerCallBack = new AsyncCallback (OnDataReceived);}
        SocketPacket theSocPkt = new SocketPacket (soc);soc.BeginReceive
        (theSocPkt.dataBuffer, 0, theSocPkt.dataBuffer.Length,SocketFlags.None,
        pfnWorkerCallBack,theSocPkt);}
        catch(SocketException se)
        {MessageBox.Show (se.Message );}}
```

Ondatareceived veri geldiğinde Asenkron bağlantı metodu kullanılır.

```
Public void OnDataReceived(IAsyncResult asyn)
{SocketPacket socketData = (SocketPacket)asyn.AsyncState ;
Socket workerSocket = (Socket)socketData.m_currentSocket;try
{Asenkron metod sonlandırılır ve gelen veri yakalanır.
```

```
int iRx = socketData.m_currentSocket.EndReceive (asyn);
char[] chars = new char[iRx + 1];
```

Gelen karakterler yakalanır ve tampona alınır. Dport (Bilgi portu) bilgi aktarılır.

```
System.Text.Decoder d = System.Text.Encoding.UTF8.GetDecoder();
int charLen = d.GetChars(socketData.dataBuffer,
0, iRx, chars, 0);
```

```
System.String szData = new System.String(chars);
if(szData.IndexOf("DPort")>-1)
{int firstIndex=szData.IndexOf("DPort");
int lastIndex=szData.LastIndexOf("DPort");
msg=szData.Substring(firstIndex+5,lastIndex-firstIndex-5);
msg=msg.TrimEnd().TrimStart();
```

Gelen bilgi paralel porta bilgi aktarılır. Gelen veriler, Cport (kontrol portu) veya Dport (Data portu) aktarılır. Böylelikle motor çalıştırılır veya led lambalar bit bit yakıtırlabilir.

Status durum bilgileri Sport (port) aktarılır.

```
ParallelPortMethods.InpOut.ToggleDataPortBit(0x378,Convert.ToInt16(msg);}
else if (szData.IndexOf("CPort")>-1)
{if(szData.IndexOf("CPort1")>-1)
{int firstIndex=szData.IndexOf("CPort1");
int lastIndex=szData.LastIndexOf("CPort1");
msg=szData.Substring(firstIndex+6,lastIndex-firstIndex-6);
msg=msg.TrimEnd().TrimStart();
int numRev=Convert.ToInt16(msg);
for(int i=0;i<numRev;i++)
{InpOut.StepMotorCont(0x378+2,1000000);}
else if(szData.IndexOf("CPort2")>-1)
{}}
else if(szData.IndexOf("SPort")>-1)
{stPortData=InpOut.ReadStatusPort(0x378);

SendStatusPortData(Convert.ToString(stPortData),workerSocket);}
AppendToRichEditControl("DataPort:"+msg+"\n");
workerSocket = (Socket)socketData.m_currentSocket;
```

Socket üzerinden veri beklemeye devam edilir.

```
WaitForData(socketData.m_currentSocket );}
    catch (ObjectDisposedException )
    {System.Diagnostics.Debugger.Log(0,"1","\nOnDataReceived:
Socket has been closed\n");}
catch(SocketException se)
{if(se.ErrorCode == 10054)
    {string msg = "Client Disconnected" + "\n";
    AppendToRichEditControl(msg);
    UpdateClientListControl();}
    Else {MessageBox.Show (se.Message );}}}
```

2.4. Cihazın Sunucu Üzerinden Kontrol Programı

Cihaz sunucu üzerinden seri, paralel port veya USB port üzerinden bağlanır. Kontrol programı sunucu üzerinden istemciden gelen bilgiler kullanarak, sunucu programı yapar.

Programımızda paralel port yöntemi kullanıldığından sunucu programında, “ParalelPortMethods” nesnesi kullanılır. Aşağıdaki program ile motor ve ledlerin kontrolü yapılır.

Gelen bilgi paralel porta bilgi aktarılır. Gelen veriler, Cport (kontrol portuna) veya Dport (Data portuna) aktarılır. Böylelikle motor çalıştırılır veya led lambalar bit bit yakıtırılabilir.

Status durum bilgileri Sport (Status-Durum portuna) aktarılır.

```
ParalelPortMethods.InpOut.ToggleDataPortBit(0x378, Convert.ToInt16(msg));
    }
    else if (szData.IndexOf("CPort")>-1)
    {if(szData.IndexOf("CPort1")>-1)
        {int firstIndex=szData.IndexOf("CPort1");
        int lastIndex=szData.LastIndexOf("CPort1");
        msg=szData.Substring(firstIndex+6,lastIndex-firstIndex-6);

        msg=msg.TrimEnd().TrimStart();
        int numRev=Convert.ToInt16(msg);
        for(int i=0;i<numRev;i++)

{InpOut.StepMotorCont(0x378+2,10000000);}}
    else if (szData.IndexOf("CPort2")>-1)
    {}
    else if (szData.IndexOf("Sport")>-1)
    {stPortData=InpOut.ReadStatusPort(0x378);
    SendStatusPortData(Convert.ToString(stPortData),workerSocket);}
    AppendToRichEditControl("DataPort:"+msg+"\n");
    workerSocket =(Socket)socketData.m_currentSocket;
```

Eğer, Seri port kullanılacaksa, aşağıdaki kütüphane eklenir.

```
using System.IO
```

Seri portla Led lamba kontrol devresi ile ilgili bir örnek kaynakçada (MCU Turkey) mevcuttur. Buradaki örnek incelenerek, seri porttan motor kontrolü veya led lamba internetten de kontrol edilmesi sağlanabilir.

Bu örnekten Seri Portla ilgili komutlardan önemli olanları aşağıda verilmiştir.

Seri porttan veri gönderilme ve bağlantı için aşağıdaki komutlar kullanılır.

```
try
{
    serialPort1.PortName = "COM" + port.Text;
    serialPort1.BaudRate = Convert.ToInt32(baudrate.Text);
    serialPort1.Open();
    baglan.Enabled = false;
    basla.Enabled = true;
    gonder.Enabled = true;
    metinGonder.Enabled = true;
    button4.Enabled = true;
    kes.Enabled = true;
    kaydirCb.Enabled = true }
catch
{ MessageBox.Show("Bağlantı kurulamadı."); }
```

Seri portla bağlantı kesme ile ilgili örnek komutlar aşağıdaki gibidir.

```
serialPort1.Close();
baglan.Enabled = true;
basla.Enabled = false;
gonder.Enabled = false;
metinGonder.Enabled = false;
button4.Enabled = false;
kes.Enabled = false;
kaydirCb.Enabled = false;
saatTimer.Stop();
kaydirTimer.Stop();
rssKontrolTimer.Stop();
rssGoruntuleTimer.Stop();
```

2.5. İstemci Program

2.5.1.İstemci Program Tasarımı

The screenshot shows a Windows-style application window titled "User Control Form". It features a light beige background and a blue title bar. The interface includes several input fields and buttons. At the top left, there are two text boxes: "Server IP Address" containing "10.0.0.7" and "Server Port" containing "8000". To the right of these is a green button labeled "Sunucuya Baglan". Below this is a red button labeled "Sunucuya Baglantiyi Kes". In the center, there is a section titled "Data Port Leds" containing eight buttons labeled "D 8" through "D 1". To the right of this is a section titled "Status Port Girdi Kanallari" containing four checkboxes labeled "1", "2", "3", and "4", and a button labeled "Girdi Kanal Durumu Al". At the bottom left, there is a section titled "ControlPorta Bagli Motor" containing a text box with "1" and a button labeled "Motor Calistir".

Şekil 2.4: İstemci iletişim formu

İstemcide çalışan programın kullanıcı önyüzünde, Dataport kanallarını temsil eden düğmelere tıklandığında sunucu paralel portuna bağlı elektronik kartın Dataport bölümündeki ledleri yakıp söndürebiliriz.

Düğmeler, NewButton adıyla oluşturulan, normal Button sınıfından türemiş fakat ondan 2 farklı özelliği olan düğmelerin oluşturduğu ButtonArray sınıfıyla form üzerindeki Dataport “groupbox”u üzerine, kullanıcı ön yüz sayfası yüklenirken otomatik olarak yerleştirilmektedir.

Tüm düğmeler için tek bir “EventHandler” vardır. EventHandler, 8 kanala denk gelen düğmelerden herhangi biri tıklandığında, düğmenin “Tag” özelliğini kullanılarak hangi düğmenin tıklandığı saptar.

NewButton düğmelerinden tek farkı, burada türettiğimiz ButtonArray içindeki düğmelerin her birinin aynı zamanda bir de System.Net.Sockets alanından alınmış bir “Socket “ sınıfı özel alanının olmasıdır. Bu alan sayesinde, düğmelere tıkladığımızda, doğrudan istemciden sunucuya mesaj gönderilebilir.

Elektronik karttaki 3 grubu temsilen, hangi porta bilgi gönderilecekse, sunucu-istemci arasındaki her gönderilen bilginin başına ve sonuna bir önek ve sonek bilgisi koyulmalıdır. Bilgi isteme durumunda ise sadece önek kullanılmalıdır. Böylece ilgili yerde alınan bilginin hangi portla ilgili olduğu bilinebilir.

Kabul edilen ekler yerine göre şöyledir:

Data port için: DPort, Control port : CPort, StatusPort : SPort

Örneğin, Data port düğmelerinden birisi tıklandığında , , sunucuya göndereceğimiz bilginin başına ve sonuna “DPort” yazısı getiriyoruz: “DPort”+düğme numarası+”DPort”; Diyelim ki “1” numaralı kanalı toggle etmek isteriz. Gidecek bilgi “DPortIDPort” olacaktır.

```
int i=(int)((NewButton)sender).Tag;
string msg = Convert.ToString(i);
msg="DPort"+msg+"DPort";
//send the data in bytes;
byte[] byData = System.Text.Encoding.ASCII.GetBytes(msg);
if(((NewButton)sender).BSocket != null)
{((NewButton)sender).BSocket.Send(byData);}
if(((NewButton)sender).Status==false)
{ ((NewButton)sender).Status=true ;
  ((NewButton)sender).BackColor= System.Drawing.Color.Red;}
else
{((NewButton)sender).Status=false;
  ((NewButton)sender).BackColor=System.Drawing.Color.White;}
```

Yukarıda verilen “EventHandler” metodundan da görüleceği üzere tıklanan düğmenin numarası “DPort” ekleri arasına konulmakta ve böylece oluşturulan veri tümcesi “byte[]” türünden veriye dönüştürülür. Düğmenin Socket alanının “Send” metodu aracılığıyla sunucuya gönderilmektedir.

Doğal olarak, daha önceden, düğmeler form üzerine yerleştirilirken, bu düğmelerin Socket tipi alanlarına, istemci programı ilk çalıştırıldığında yaratılan “m_clientSocket” nesnesi (Programın genel yapısı aşağıda daha detaylı anlatılacaktır) atanmaktadır. Bu veri sunucuya gittiğinde, oradaki kod, ön ve sonekleri atıp, aradaki veriyi alıp, böylece gerekli veriyi Dataport’a gönderip, istenen ışığı “toggle” edecektir.

Control portuna bağlı step motoru çalıştırma:

İstemci tarafında, kullanıcı ön yüzünde bulunan 1 adet “textBox” içerisine motorun kaç adım atması gerektiğine dair rakam yazılacaktır ve sonra düğme tıklandığında şu kod çalışacaktır:

```
string msg="CPort";
msg=msg+txtStepNumber.Text+msg;
byte[] byData = System.Text.Encoding.ASCII.GetBytes(msg);
m_clientSocket.Send(byData);
```

Görüldüğü gibi, motorun dönmesini istediğimiz adım sayısı “CPort” ön ve arka ekleri arasına alınmaktadır. Oluşan yeni tümce, byte[]’e çevirilip, sunucuya gönderilmektedir.

Sunucu ise, ön ve arka ekleri atıp aradaki sayı kadar motoru adımlatacaktır. Motor kontrol devresi ve kodu sunucu programı anlatılırken verilecektir.

2.5.2.İstemci Program Kodları

İstemci programında önce kütüphane dosyaları eklenir. Yuva (Socket) programlama kullanımı için System.net.Sockets kullanılır. İstemci ile veri iletişimi için Sytem.Data uzay isimleri (kütüphaneler) eklenir.

```
using System;
using System.Drawing;
using System.Collections;
using System.ComponentModel;
using System.Windows.Forms;
using System.Data;
using System.Net.Sockets;
using System.Net;
```

İstemcide kullanılacak değişkenler (databuffer, result, socketler v.b.) tanımlanır.

Oluşturulan “Socket” nesnesinin Connect metoduna bu “IPEndPoint” nesnesi parametre olarak verilerek, sunucuyla bağlantı sağlanır.

İstemci formun içinde istemci IP’si tespit edilip, yazılır. Clientsocket isimli yuva (socket), bağlantı yuvası için oluşturulur.

Bağlantı protokolü TCP, veri aktarım yöntemi Stream seçilmiştir.

Sunucudaki Step Motora gönderilecek veri, formdaki Adım Sayısı text kutusundan alınır. Form üzerindeki buttonarray ile yuva (socket) yöntemi ile tıklama ile sunucuya bilgi gönderilir.

```
InitializeComponent();
textBoxIP.Text = GetIP();
m_clientSocket=new Socket (AddressFamily.InterNetwork,
SocketType.Stream, ProtocolType.Tcp );
bArray=new ButtonArray(dataPortGroup,m_clientSocket);
txtStepNumber.Text=Convert.ToString(1);}
protected override void Dispose( bool disposing )
{if( disposing )
    {if (components != null)
        {components.Dispose();}}
    base.Dispose( disposing );}
```

Client Control formu çalıştırılır.

```
#region Windows Form Designer generated code

#endregion
[STAThread]
static void Main()
{Application.Run(new frmClientControl());}
```

GetHost komutu ile Sunucu IP adresi alınır.

```
String GetIP()
{
    String strHostName = Dns.GetHostName();
    IPEndPoint ipHostEntry = Dns.GetHostByName(strHostName);
    String IPStr = "";
    foreach(IPAddress ipaddress in ipHostEntry.AddressList)
    {IPStr = ipaddress.ToString();
     return IPStr;return IPStr;}
}
```

Sunucuya bağlanmak için istemci, metin kutusu içine Sunucu IP adresi ve port numarasını girmelidir.

IP adresi ve port numarası kullanılarak IPEndPoint nesnesi yaratılır.

```
if(textBoxIP.Text == "" || textBoxPort.Text == "")
{MessageBox.Show("Sunucuya bağlanmak için IP Adresi ve Port
numarası girilmelidir.\n");return;}
try
{UpdateControls(false);//
 IPAddress ip = IPAddress.Parse (textBoxIP.Text);
 int iPortNo = System.Convert.ToInt16 (textBoxPort.Text);
 // Uç noktası oluştur.
 IPEndPoint ipEnd = new IPEndPoint (ip,iPortNo);
 // uzak sunucuya bağlan
 m_clientSocket.Connect (ipEnd);
 if(m_clientSocket.Connected)
 {UpdateControls(true);
 WaitForData();}}
 catch(SocketException se)
 {string str;
 str = "\n Bağlantı kurulamadı, Sunucu çalışıyor mu?\n" + se.Message;
 MessageBox.Show (str);
 UpdateControls(false);}}
```

Sonra da WaitForData() metoduyla, sunucudan gelecek mesajlar dinlenilmeye başlanır.

Arka planda veriler gidip gelirken, bilgisayarın diğer işlemlerini yapabilmesi için Asenkron haberleşme tekniği kullanılmaktadır.

Bilgi için beklenir.

```
public void WaitForData()
{try
 {if(m_pfnCallBack==null)
 {m_pfnCallBack=new AsyncCallback(OnDataReceived);}}
```

Arka planda veriler gidip gelirken, bilgisayarın diğer işlemlerini yapabilmesi için Asenkron haberleşme tekniği kullanılmaktadır.

Verileri 1024 byte'lık büyüklükte bir tampon dizin aracılığıyla alınır. SocketPacket sınıfı bu büyüklükte bir alana sahiptir. Bu sınıfın bir diğer alanı da Socket nesnemizi atadığımız bir alan. Socket nesnemizin BeginReceive metoduyla, bu SocketPacket nesnesinin tampon veri alanını doldurmaya başlarız.

Bunun için, DataReceived metodu, bir "AsyncCallback" metodu olarak tanımlanmıştır.

```
SocketPacket theSktpacket=new SocketPacket();
theSktpacket.thisSocket=m_clientSocket;

m_result=m_clientSocket.BeginReceive
(theSktpacket.dataBuffer,0,theSktpacket.dataBuffer.Length,SocketFlags.None,m_pfn
CallBack,theSktpacket);}
catch(SocketException se)
{MessageBox.Show(se.Message);}}
```

Bilgi thissocket yuvası ile databuffer'a aktarılır.

```
public class SocketPacket
{public System.Net.Sockets.Socket thisSocket;
public byte[] dataBuffer=new byte[1024];}
```

AsyncCallback nesnemizin konumunu OnDataReceived() metodu içerisinde yakalayıp, sonlandırarak, alınmış verileri önce bir karakter dizini içerisine sonra da String tipine dönüştürürüz.

```
public void OnDataReceived(IAsyncResult asyn)
{SocketPacket theSocketId=(SocketPacket)asyn.AsyncState;
int dataReceived=theSocketId.thisSocket.EndReceive(asyn);
char[] chars=new char[dataReceived+1];
System.Text.Decoder
decoder=System.Text.Encoding.UTF8.GetDecoder();
int
charLen=decoder.GetChars(theSocketId.dataBuffer,0,dataRecei
ved,chars,0);
```

Status portuna bağlı girdi kanallarını okutmak içinse, ilgili düğmeyi tıkladığımızda sadece Sport yazısını sunucuya gönderilir. Sunucu tarafına gelen mesaj "Sport" ile başlıyorsa, sunucu bundan Status port girdi kanallarına ait bilgi istendiğini anlar ve bu portun verisini okuyarak, istemciye gönderir.

İstemci de zaten bu veriyi WaitForData() metoduyla beklemektedir. Gelen veri, OnDataReceived() metoduyla işlenir. Sunucudan bu veri "SPort" ön ve arka ekleriyle geldiğinden, bu metot içerisinde bu eklerden kurtulup StatusPort'tan sunucunun okuduğu "byte" verisi deşifre edilir (herbir bit'e karşılık gelen bit ağırlık değeriyle gelen veriyi "AND"leyip sonucu kontrol ederek) ve ilgili "checkbox" , "checked" veya "unchecked" yapılır.

Sunucudan alınan Status durum bilgisi, Strdata olarak, statusportData tanımlanır. statusPortData 1, 2, 4, 8 olmasına göre istemci programında formun üzerindeki göstereçler aktif olur.

```
System.String strData=new System.String(chars);
if(strData.IndexOf("SPort")>-1)
{
    int fIndex=strData.IndexOf("SPort");
    int lIndex=strData.LastIndexOf("SPort");
    strData=strData.Substring(fIndex+5,lIndex-fIndex-5);
    int statusPortData=Convert.ToInt16(strData);
    UncheckAll();
    if ((statusPortData &1)==1) chk1.Checked=true;
    if ((statusPortData &2)==2) chk2.Checked=true;
    if ((statusPortData &4)==4) chk3.Checked=true;
    if ((statusPortData &8)==8) chk4.Checked=true;
}
public void UncheckAll()
{
    chk1.Checked=false;
    chk2.Checked=false;
    chk3.Checked=false;
    chk4.Checked=false;}
}
```

Alttaki program da bağlantı ile ilgili sunucuya bağlılık durum bilgisi verir.

```
private void UpdateControls( bool connected )
{
    buttonConnect.Enabled = !connected;
    buttonDisconnect.Enabled = connected;
    string connectStatus = connected? "Bagli" : "Bagli degil";
    connStatus.Text = connectStatus; }
}
```

Step Motorun çalışması için Step motor adım sayısı İstemci formundaki metin kutusundan alınarak, "Cport1" başa ve sona eklenerek, msg olarak tanımlanır. Bu veri ASCII koda çevrilerek, bydata şeklinde clientsocket ile sunucuya gönderilir.

```
string msg="CPort1";
msg=msg+txtStepNumber.Text+msg;
byte[] byData = System.Text.Encoding.ASCII.GetBytes(msg);
m_clientSocket.Send(byData);
```

“Sport” deęeri yukarıdaki Form üzerindeki Status Port Girdi kanalları 1, 2, 3, 4 tuşlarının durumuna göre deęer alır.

Status Port durum bilgisinin alınması için formunda “Girdi Kanal Durum Al” butona tıkladığında, “Sport” msg olarak tanımlanır. Bu veri ASCII koda çevrilerek, bydata şeklinde clientsocket ile sunucuya gönderilir.

```
string msg="SPort";  
byte[] byData = System.Text.Encoding.ASCII.GetBytes(msg);  
m_clientSocket.Send(byData);  
waitForData();}
```

2.6. İnternet Üzerinde Kontrolde Güvenlik

İstemci ile sunucu arasında kullanıcının bilgilerinin sorgulandığı bir güvenlik formu ve kullanıcı kayıtlarını tutulduğu bir veri tabanı eklenir.

2.6.1. Kullanıcı Adı Ve Şifreye Bağlı İnternet Kontrolü

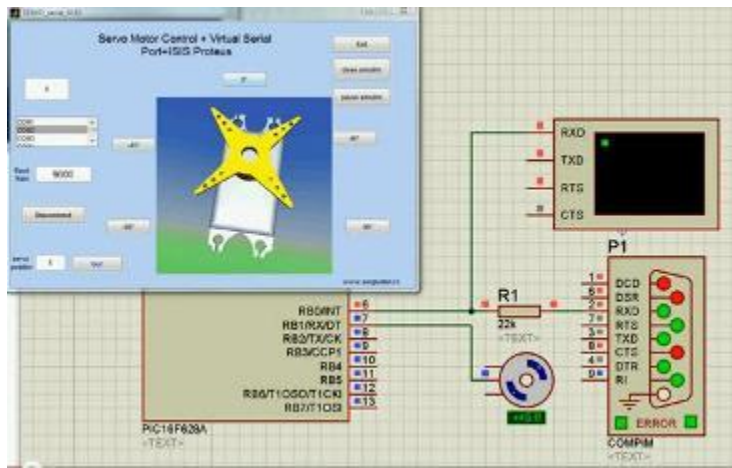
İstemci sunucuya bağlanma isteğinde bulunduğunda, sunucu tarafında bir form oluşturulur. Sunucu kullanıcı bilgileri SQL SERVER programı ile veri tabanında tutulabilir. Sunucu istemciden gelen istekle ilk önce güvenlik formunu çağırır. Kullanıcının bilgilerini veri tabanında sorgular. Sorgu sonucunda, geçerli parolaya sahip kullanıcılara istemci motor kontrol programı ekrana getirilir.

Bunun için yukarıda yazılı sunucu programı içinde bir form ve SQL veri tabanı ve sorgusu eklenir.

UYGULAMA FAALİYETİ

Elektronik devre tasarım programı ile elektronik devre tasarımı yapabileceksiniz.

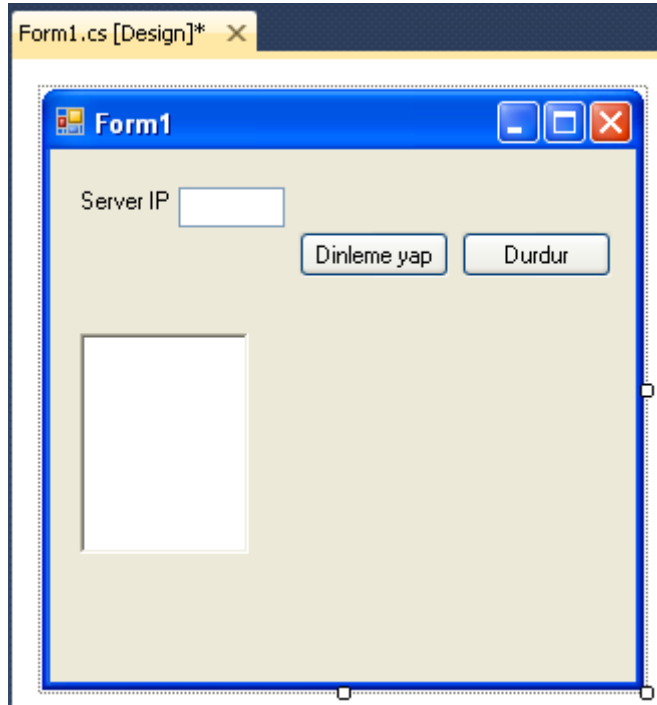
İşlem Basamakları	Öneriler
<ul style="list-style-type: none">➤ Proteus 7.6 programı içinden ISIS alt programı açılır.➤ P tuşuna basarak, aşağıdaki resimdeki elemanlar eklenerek, Com portu (COMPIM) , PIC, görsel bağlantı (virtual terminal), ekleyin. Şekildeki gibi bağlantılarını yapın.➤ Pic için gerekli diğer bağlantıları (kristal, kondansatör, besleme gerilimi gibi) yapın.➤ ARES tuşuna basılarak, devre baskı devre çizimine aktarılır.➤ Soldaki resim araç kutusundan kare seçilir. Bir baskı devre alanı seçilir.➤ Baskı devre alanı üzerinde sağ tıklama yapılır. Katmanı değiştir (Change Layer), Devre çerçevesi (Board Edge) tıklanır.➤ Araçlar(Tools) listesinden Otomatik yerleştir (Auto Placer) ile elemanlar yerleştirilir.➤ Araçlar(Tools) listesinden Otomatik yol yerleştir (Auto Router) ile elemanlar yerleştirilir.➤ -Varsa- Gerekli düzeltmeler yapılır. Output tuşundan çıktı alınır.➤ Ütüleme ve kimyasal yöntemlerle baskı devre çıkartılır.➤ Baskı devre yolları kontrol edilir.➤ Devre elemanları baskı devreye yerleştirilir.	<ul style="list-style-type: none">➤ Bilgisayarlı devre dizaynı modüllerini okuyunuz.➤ Lehimleme ve baskı devre yapımı modülünü okuyunuz.➤ Mekanik ve kimyasal işlemleri atölye ve lavaboda güvenli bir şekilde gerçekleştiriniz.



Proteus uygulaması

Bu uygulama ile Sunucu programı yazmayı öğreneceksiniz.

İşlem Basamakları	Öneriler
<ul style="list-style-type: none">➤ Visual Studio Programını açınız.➤ Dosya, yeni, proje kısmından windows form uygulamasını açınız.➤ Çıkan form üzerine Şekil 2.3.' deki gibi buton, yazı (text), etiket (label), yerleştirin.➤ Kaynakçadan sunucu programın kaynak kodlarını alarak butonların altına ve ana programa yerleştiriniz.➤ Derleme için "Debug" işlemini yapınız.➤ Hata yoksa programınızı çalıştırabilirsiniz.	<ul style="list-style-type: none">➤ Program yazmaya başlamadan bu dere ait daha önceki modülleri tekrar okuyunuz.➤ Modüllerdeki ve internette bulacağınız programı yazarak, çalıştırınız.



Sunucu(Server) programı yazmak

Bu uygulama ile İstemci programı yazmayı öğreneceksiniz.

İşlem Basamakları	Öneriler
<ul style="list-style-type: none">➤ Visual Studio Programını açınız.➤ Dosya, Yeni, Proje kısmından Windows form uygulamasını açınız.➤ Çıkan form üzerine Şekil 2.4.'deki gibi buton, yazı (text), etiket (label), yerleştirin.➤ Kaynakçadan istemci programın kaynak kodlarını alarak, butonların altına ve ana programa yerleştiriniz.➤ Derleme için "Debug" işlemini yapınız.➤ Hata yoksa programınızı çalıştırabilirsiniz.	<ul style="list-style-type: none">➤ Program yazmaya başlamadan bu dere ait daha önceki modülleri tekrar okuyunuz.➤ Modüllerdeki ve internette bulacağınız programı yazarak, çalıştırınız.



İstemci(Client) Programı yazmak

KONTROL LİSTESİ

Bu faaliyet kapsamında aşağıda listelenen davranışlardan kazandığınız becerileri Evet, kazanamadığınız becerileri Hayır kutucuğuna (X) işareti koyarak kendinizi değerlendiriniz.

Değerlendirme Ölçütleri		Evet	Hayır
1.	İsis programında seri port devresini çizabildiniz mi?		
2.	Ares programında seri port devresinin baskı devresini hazırladınız mı?		
3.	Sunucu programının görsel formunu Visual Studio oluşturabildiniz mi?		
4.	Sunucu programının kodlarını doğru olarak yazabildiniz mi?		
5.	İstemci programının görsel formunu Visual Studio oluşturabildiniz mi?		
6.	İstemci programının kodlarını doğru olarak yazabildiniz mi?		

DEĞERLENDİRME

Değerlendirme sonunda “**Hayır**” şeklindeki cevaplarınızı bir daha gözden geçiriniz. Kendinizi yeterli görmüyorsanız öğrenme faaliyetini tekrar ediniz. Bütün cevaplarınız “**Evet**” ise “Ölçme ve Değerlendirme”ye geçiniz.

ÖLÇME VE DEĞERLENDİRME

- Aşağıdaki çoktan seçmeli soruları dikkatlice okuyunuz ve doğru seçeneği işaretleyiniz.
- İsis programında elemanlar kütüphaneden, çalışma alanına aşağıdakilerden hangisi ile çağrılır?
 - L harfine tıklanarak, eleman ismine göre çağrılır.
 - P harfine tıklanarak, eleman ismine göre çağrılır.
 - Kaynak (Source) menüsünden isme göre eklenir.
 - Hiçbiri
 - Ares programında elemanların yerleştirilmesinden sonra yollar oluşturulması aşağıdaki komutlardan hangisi ile yapılır?
 - AutoPlacer
 - AutoPick
 - AutoLine
 - AutoRouter
 - Visual Studio programında görsel form üzerinde butonların isimlerin yerleştirilmesi aşağıdaki yöntemlerden hangisi ile yapılır?
 - Buton üzerinde farenin sağ tuşuna basılarak, çıkan menüdeki ad değiştir ile.
 - Form üzerindeki iken özelliklerden
 - Butona tıklanarak, özelliklerden, yazı (text) kutusundan
 - Butona tıklanarak, özelliklerden, isim (name) kutusundan
 - Ares programında kâğıda çıktı almak için aşağıdakilerden hangisi yapılmalıdır?
 - Dosya, çıktı (yazdır) işlemi yapılarak
 - Görünüm baskı devre çıkart işlemi ile
 - Çıkış, şekli ihraç et (Export Graphics)
 - Çıkış, yazdır.
 - Seri port devresinde kullanılan görsel bağlantı (Virtual Terminal) elemanı devreye aşağıdaki yöntemlerden hangisi ile eklenir?
 - L harfine tıklanarak, eleman ismine göre çağrılır.
 - P harfine tıklanarak, eleman ismine göre çağrılır.
 - Sol taraftaki araç çubuğundan ölçü aleti konumundan(Instruments mode) .
 - Hiçbiri
 - Visual Studio programında görsel form üzerinde butonlara ait kodlara aşağıdaki yöntemlerden hangisi ile ulaşılır?
 - Butona sağ tıklayarak,
 - Buton üzerinde çift tıklayarak
 - Butona tıklanarak, özelliklerden, komut kutusundan
 - Hiçbiri

DEĞERLENDİRME

Cevaplarınızı cevap anahtarıyla karşılaştırınız. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt ettiğiniz sorularla ilgili konuları faaliyete geri dönerek tekrarlayınız. Cevaplarınızın tümü doğru “Modül Değerlendirme”ye geçiniz.

MODÜL DEĞERLENDİRME

1. İstemci ve sunucu haberleşmesi için ön koşullardan biri aşağıdakilerden hangisidir?
 - A. Sunucunun IP adresi bilinmeli
 - B. İstemcinin IP adresi bilinmeli
 - C. Sunucuda TCP protokolü tercih edilmeli
 - D. İstemcide TCP protokolü tercih edilmeli
2. Bilgisayar portlarından internet üzerinden veri aktarımı için aşağıdaki yöntemden hangisi kullanılır?
 - A. Port
 - B. Yuva (Socket)
 - C. USB
 - D. Paralel port
3. Yuva (Socket) yöntemi ile aşağıdakilerden hangisi **gerçekleştirilemez**?
 - A. İnternet üzerinden haberleşme
 - B. İnternet üzerinden bilgi gönderimi
 - C. İnternet üzerinden uzak bilgisayara veri gönderimi
 - D. Sunucu bilgisayarın port olmaksızın elektronik devreyi kontrol etmesi
4. Yuva yapısı ile veri gönderilirken, aşağıdaki protokollerden hangisi kullanılmalıdır?
 - A. UDP
 - B. TCP
 - C. Ethernet
 - D. Hiçbiri
5. İnternette kontrol için İstemci ve sunucuda yazılan programlarda aşağıdakilerden hangisi ortak işlem **değildir**?
 - A. Yuva oluşturmak
 - B. Dinleme yapmak
 - C. Veri almak veya göndermek
 - D. IP adresini temin etmek
6. Programın test edilmesi işleminde aşağıdaki işlemlerden hangisi yapılırsa hata oluşur?
 - A. Sunucunun bilgisayar adı yanlış girilirse
 - B. Sunucunun bağlı olduğu çalışma alanı yanlış girilirse
 - C. Sunucunun IP numarası yanlış girilirse
 - D. Hiçbiri

7. GetIP() komutunun görevi aşağıdakilerden hangisinde doğru olarak açıklanmıştır?
- A. İstemcinin IP numarasını bulur
 - B. Sunucunun IP numarasını bulur
 - C. İstemci ile sunucu arasında IP ile veri aktarır
 - D. İstemci ile sunucu arasında TCP ile veri aktarır
8. İnternette elektronik cihaz kontrolü için cihazın elektronik devre ve baskı devre hazırlama programı aşağıdakilerden hangisidir?
- A. Isis
 - B. Paint
 - C. Ares
 - D. Proeus
9. Aşağıdakilerden hangisi sunucu ve istemci programının yazılımındaki kütüphanelerden ortak olan **kütüphaneden değildir?**
- A. `using System.Net.Sockets;`
 - B. `using System.Collections;`
 - C. `using System.Data;`
 - D. `using ParallelPortMethods;`
10. İstemciden gelen bilgi aşağıdaki komutlardan hangisi ile dinlenir?
- A. Onclick
 - B. WaitforData
 - C. ListenforData
 - D. SeeforData
11. İstemciden gelen veri ile sunucu yazılımında, port kontrolü için aşağıdaki kütüphanelerden hangisini kullanır?
- A. Seri port için ParallelportMethods
 - B. Paralel port için System.IO
 - C. Seri port için System.IO
 - D. Hiçbiri
12. İstemci Sunucusuyla aşağıdaki komutlardan hangisi ile bağlantı kurar?
- A. GetIP()
 - B. ServerPoint
 - C. IPPoint
 - D. IPEndpoint

13. “Buttonarray” komutunun görevi, tam ve doğru olarak aşağıdakilerden hangisinde verilmiştir?
- A. Formdaki tıklamaları algılamak
 - B. Sunucuya bilgi göndermek
 - C. Formdaki tıklamaları algılayarak, sunucuya bilgi göndermek
 - D. Hiçbiri
14. “Clientsocket.send()” komutunun görevi aşağıdakilerden hangisidir?
- A. Sunucuyla iletişim kurar
 - B. İstemcinin IP’sini sunucuya gönderir
 - C. Sunucuya veri gönderir
 - D. Sunucunun IP’sini istemciye gönderir.

DEĞERLENDİRME

Cevaplarınızı cevap anahtarıyla karşılaştırınız. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt ettiğiniz sorularla ilgili konuları faaliyete geri dönerek tekrarlayınız. Cevaplarınızın tümü doğru ise bir sonraki modüle geçmek için öğretmeninize başvurunuz.

CEVAP ANAHTARLARI

ÖĞRENME FAALİYETİ-1'İN CEVAP ANAHTARI

1	C
2	C
3	B
4	B

ÖĞRENME FAALİYETİ-2'NİN CEVAP ANAHTARI

1	B
2	D
3	C
4	D
5	C
6	B

MODÜL DEĞERLENDİRME CEVAP ANAHTARI

1	A
2	B
3	D
4	B
5	B
6	C
7	B
8	D
9	D
10	B
11	C
12	D
13	C
14	C

KAYNAKÇA

- <http://www.enderunix.org/docs/socket.html>
- Altıntaş, Altuğ B., **Java ve Yazılım Tasarımı Ders Notları**, Papatya Yayıncılık, İstanbul, 2010-3