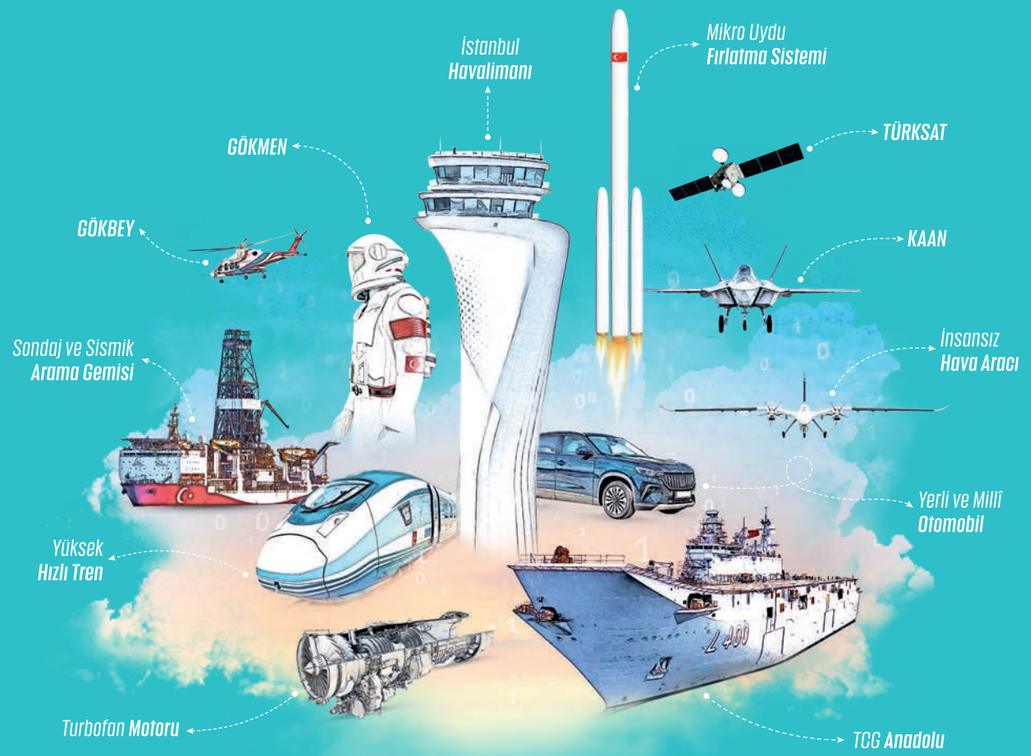
Bu kitaba sığmayan daha neler var!

eba
www.eba.gov.tr

Karekodu okut, bu kitapla ilgili EBA içeriklerine ulaş!

Teknoloji Yolculuğumuz



BU DERS KİTABI MİLLÎ EĞİTİM BAKANLIĞINCA ÜCRETSİZ OLARAK VERİLMİŞTİR.
PARA İLE SATILAMAZ.

ISBN: 978-975-11-7111-5

Bandrol Uygulamasına İlişkin Usul ve Esaslar Hakkında Yönetmeliğin Beşinci Maddesinin İkinci Fıkrası Çerçevesinde Bandrol Taşıması Zorunlu Değildir.

YENİLENEBİLİR ENERJİ TEKNOLOJİLERİ ALANI

ATÖLYE 11

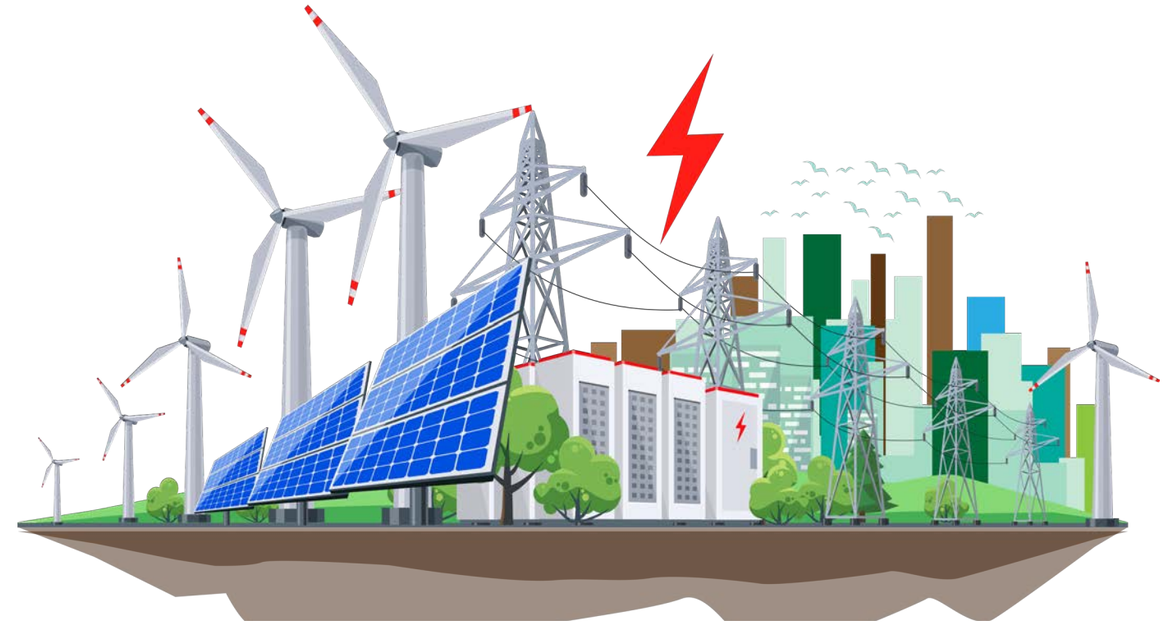
DERS MATERYALİ

MESLEKİ VE TEKNİK ANADOLU LİSESİ

YENİLENEBİLİR ENERJİ TEKNOLOJİLERİ ALANI

ATÖLYE 11

Ders Materyali





MESLEKİ VE TEKNİK ANADOLU LİSESİ

YENİLENEBİLİR ENERJİ TEKNOLOJİLERİ ALANI

ATÖLYE 11

DERS MATERYALİ

YAZARLAR

Dr. Aziz GÜNEROĞLU
Ahmet KEKİK
Aykut AKTAŞ
Cansen ALTUN
Erdal EVCİL
Harun EZER
Hayati ÖZDEMİR
Mehmet Halil KORATEŞ
Nasuf ÇALIŞKAN

MİLLÎ EĞİTİM BAKANLIĞI YAYINLARI.....: 8733
YARDIMCI VE KAYNAK KİTAPLAR DİZİSİ.....: 2618

Her hakkı saklıdır ve Millî Eğitim Bakanlığına aittir.
Ders materyalinin metin, soru ve şekilleri kısmen de olsa hiçbir surette alınıp yayımlanamaz.

HAZIRLAYANLAR

DİL UZMANI
Tekin YILMAZ

GÖRSEL TASARIM UZMANI
Nihal GENÇOL
Suat GENÇOL

ISBN: 978-975-11-7111-5

Millî Eğitim Bakanlığının 24.12.2020 gün ve 18433886 sayılı oluru ile Meslekî ve Teknik Eğitim Genel Müdürlüğünce ders materyali olarak hazırlanmıştır.



İSTİKLÂL MARŞI

Korkma, sönmez bu şafaklarda yüzen al sancak;
Sönmeden yurdumun üstünde tüten en son ocak.
O benim milletimin yıldızıdır, parlayacak;
O benimdir, o benim milletimindir ancak.

Çatma, kurban olayım, çehreni ey nazlı hilâl!
Kahraman ırkıma bir gül! Ne bu şiddet, bu celâl?
Sana olmaz dökülen kanlarımız sonra helâl.
Hakkıdır Hakk'a tapan milletimin istiklâl.

Ben ezelden beridir hür yaşadım, hür yaşarım.
Hangi çılgın bana zincir vuracakmış? Şaşarım!
Kükremiş sel gibiyim, bendimi çiğner, aşarım.
Yırtarım dağları, enginlere sığmam, taşarım.

Garbın âfâkını sarmışsa çelik zırhlı duvar,
Benim iman dolu göğsüm gibi serhaddim var.
Ulusun, korkma! Nasıl böyle bir imanı boğar,
Medeniyyet dediğin tek dişi kalmış canavar?

Arkadaş, yurduma alçakları uğratma sakın;
Siper et gövdeni, dursun bu hayâsızca akın.
Doğacaktır sana va'dettiği günler Hakk'ın;
Kim bilir, belki yarın, belki yarından da yakın.

Bastığın yerleri toprak diyerek geçme, tanı:
Düşün altındaki binlerce kefensiz yatanı.
Sen şehit oğlusun, incitme, yazıktır, atanı:
Verme, dünyaları alsan da bu cennet vatanı.

Kim bu cennet vatanın uğruna olmaz ki feda?
Şüheda fışkıracak toprağı sıksan, şüheda!
Cânı, cânânı, bütün varımı alsın da Huda,
Etmesin tek vatanımdan beni dünyada cüda.

Ruhumun senden İlâhî, şudur ancak emeli:
Değmesin mabedimin göğsüne nâmahrem eli.
Bu ezanlar -ki şehadetleri dinin temeli-
Ebedî yurdumun üstünde benim inlemeli.

O zaman vecd ile bin secde eder -varsa- taşım,
Her cerîhamdan İlâhî, boşanıp kanlı yaşım,
Fışkırır ruh-ı mücerret gibi yerden na'şım;
O zaman yükselerek arşa değer belki başım.

Dalgalan sen de şafaklar gibi ey şanlı hilâl!
Olsun artık dökülen kanlarımın hepsi helâl.
Ebediyyen sana yok, ırkıma yok izmihlâl;
Hakkıdır hür yaşamış bayrağımın hürriyyet;
Hakkıdır Hakk'a tapan milletimin istiklâl!

Mehmet Âkif Ersoy

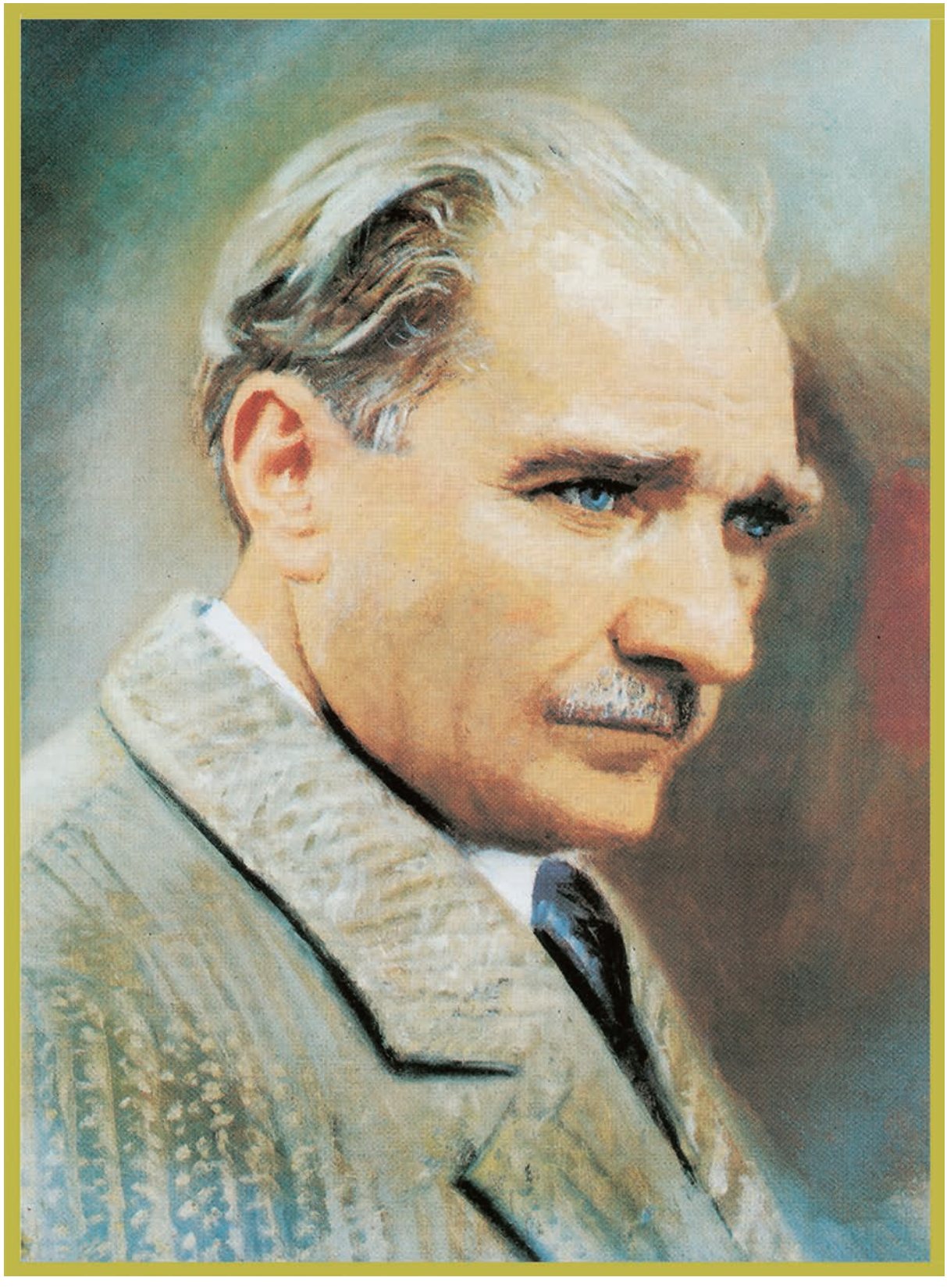
GENÇLİĞE HİTABE

Ey Türk gençliği! Birinci vazifen, Türk istiklâlini, Türk Cumhuriyetini, ilelebet muhafaza ve müdafaa etmektir.

Mevcudiyetinin ve istikbalinin yegâne temeli budur. Bu temel, senin en kıymetli hazinendir. İstikbalde dahi, seni bu hazineden mahrum etmek isteyecek dâhilî ve hâricî bedhahların olacaktır. Bir gün, istiklâl ve cumhuriyeti müdafaa mecburiyetine düşersen, vazifeye atılmak için, içinde bulunacağın vaziyetin imkân ve şeraitini düşünmeyeceksin! Bu imkân ve şerait, çok namüsait bir mahiyette tezahür edebilir. İstiklâl ve cumhuriyetine kastedecek düşmanlar, bütün dünyada emsali görülmemiş bir galibiyetin mümessili olabilirler. Cebren ve hile ile aziz vatanın bütün kaleleri zapt edilmiş, bütün tersanelerine girilmiş, bütün orduları dağıtılmış ve memleketin her köşesi bilfiil işgal edilmiş olabilir. Bütün bu şeraitten daha elîm ve daha vahim olmak üzere, memleketin dâhilinde iktidara sahip olanlar gaflet ve dalâlet ve hattâ hıyanet içinde bulunabilirler. Hattâ bu iktidar sahipleri şahsî menfaatlerini, müstevlîlerin siyasî emelleriyle tevhit edebilirler. Millet, fakr u zaruret içinde harap ve bîtap düşmüş olabilir.

Ey Türk istikbalinin evlâdı! İşte, bu ahval ve şerait içinde dahi vazifen, Türk istiklâl ve cumhuriyetini kurtarmaktır. Muhtaç olduğun kudret, damarlarındaki asil kanda mevcuttur.

Mustafa Kemal Atatürk



MUSTAFA KEMAL ATATÜRK

İÇİNDEKİLER

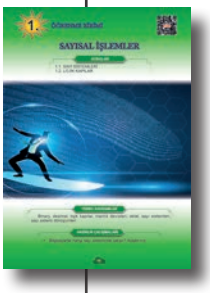
DERS MATERYALİNİN TANITIMI 13

1. ÖĞRENME BİRİMİ: SAYISAL İŞLEMLER

1.1. SAYI SİSTEMLERİ 16
1.1.1. Sayı Sistemlerinin Çeşitleri 16
1.1.1.1. Binary (İkili) Sayı Sistemleri 16
1.1.1.2. Oktal (Sekizli) Sayı Sistemleri 16
1.1.1.3. Desimal (Onlu) Sayı Sistemleri 17
1.1.1.4. Heksadesimal (Onaltılı) Sayı Sistemleri 17
1.1.2. Sayı Sistemlerinin Birbirine Dönüşümü 18
1.1.2.1. Desimal Sayının Binary Sayıya Dönüştürülmesi 18
1.1.2.2. Desimal Sayının Oktal Sayıya Dönüştürülmesi 19
1.1.2.3. Desimal Sayının Heksadesimal Sayıya Dönüştürülmesi 19
1.1.2.4. Binary Sayının Desimal Sayıya Dönüştürülmesi 20
1.1.2.5. Binary Sayının Oktal Sayıya Dönüştürülmesi 20
1.1.2.6. Binary Sayının Heksadesimal Sayıya Dönüştürülmesi 21
1.1.2.7. Oktal Sayının Desimal Sayıya Dönüştürülmesi 21
1.1.2.8. Oktal Sayının Binary Sayıya Dönüştürülmesi 22
1.1.2.9. Oktal Sayının Heksadesimal Sayıya Dönüştürülmesi 22
1.1.2.10. Heksadesimal Sayının Desimal Sayıya Dönüştürülmesi 23
1.1.2.11. Heksadesimal Sayının Binary Sayıya Dönüştürülmesi 23
1.1.2.12. Heksadesimal Sayının Oktal Sayıya Dönüştürülmesi 24
1.1.3. Binary Sayılarda Aritmetik İşlemler 25
1.1.3.1. Binary Sayılarda Toplama 25
1.1.3.2. Binary Sayılarda Çıkarma 27
1.1.3.3. Binary Sayılarda Çarpma 27
1.2. MANTIKSAL ENTEGRELER VE LOJİK KAPILAR 28
1.2.1. Lojik Entegrelerin Özellikleri 28
1.2.2. Lojik Entegrelerin Çeşitleri 29
1.2.3. Lojik Kapıların Özellikleri ve Doğruluk Tablosu 30
1.2.3.1. Mantık Anahtarlama İşlemleri 31
1.2.3.2. TAMPON (BUFFER) Kapısı 31
1.2.3.3. DEĞİL (NOT) Kapısı 31
1.2.3.4. VE (AND) Kapısı 31
1.2.3.5. VEYA (OR) Kapısı 32
1.2.3.6. VE DEĞİL (NAND) Kapısı 32
1.2.3.7. VEYA DEĞİL (NOR) Kapısı 32
1.2.3.8. ÖZEL VEYA (EX-OR) Kapısı 33
1.2.3.9. ÖZEL VEYA DEĞİL (EX-NOR) Kapısı 33
1.2.4. Pozitif ve Negatif Lojik İfadeler 33
ÖLÇME VE DEĞERLENDİRME 60

2. ÖĞRENME BİRİMİ: MİKRODENETLEYİCİ PROGRAMLAMA

2.1. MİKRODENETLEYİCİ SEÇİMİ 62
2.1.1. Mikroişlemciler ve Mikrodenetleyiciler 62
2.1.2. Mikrodenetleyici Çeşitleri 63
2.1.3. Mikrodenetleyici Donanım Özellikleri 63
2.1.4. Mikrodenetleyici Programlama Dilleri 64
2.2. MİKRODENETLEYİCİ EDİTÖRÜ 64
2.2.1. Mikrodenetleyici Program Editörü Kurulumu 64
2.2.2. Mikrodenetleyici Program Editörü Menüleri 65
2.3. MİKRODENETLEYİCİ YAZILIMI 66
2.3.1. Mikrodenetleyici Akış Diyagramı 66
ÖLÇME VE DEĞERLENDİRME 72





İÇİNDEKİLER

3. ÖĞRENME BİRİMİ: DİJİTAL VE ANALOG UYGULAMALAR.....

3.1. MİKRODENETLEYİCİ UYGULAMALARI 74
3.1.1. Mikrodenetleyici Devrelerinin Kurulması 74
3.1.2. Mikrodenetleyici Komutları 75
3.1.2.1. Değişkenler 75
3.1.2.2. Operatörler 76
3.1.2.3. Fonksiyonlar 80
3.1.2.4. Karar Verme ve Kontrol Komutları 83
3.1.2.5. Döngü Komutları 84
3.2. DİJİTAL GİRİŞ-ÇIKIŞ 85
3.2.1. Dijital Giriş-Çıkış Kavramları 85
3.3. MİKRODENETLEYİCİLERDE PORT KAVRAMI 86
3.3.1. Bit, Byte, Word, LSB, MSB 86
3.4. ANALOG GİRİŞ-ÇIKIŞ 87
3.4.1. Analog Giriş 87
3.4.2. Analog Çıkış 87
3.4.3. Seri Port Ekranı (Serial Monitor) 87
3.5. ADC-DAC UYGULAMALARI 88
3.5.1. ADC (Analog Dijital Çevirici) 88
3.5.2. DAC (Dijital Analog Çevirici) 89
ÖLÇME VE DEĞERLENDİRME 126



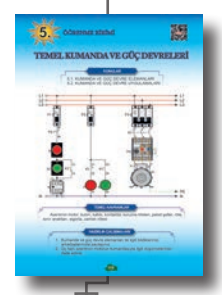
4. ÖĞRENME BİRİMİ: MİKRODENETLEYİCİ PROJE UYGULAMALARI

4.1. MOTOR KONTROL UYGULAMALARI 128
4.1.1. DC Motor ve Sürücüleri 128
4.1.1. Step Motor ve Sürücüleri 129
4.1.1. Servo Motor ve Sürücüleri 131
4.2. SENSÖR UYGULAMALARI 132
4.2.1. LM35 Sıcaklık Sensörü 132
4.2.2. LDR Işık Sensörü 132
4.2.3. DHT11 Nem ve Sıcaklık Sensörü 132
4.2.4. HC-SR501 Hareket Sensörü 132
4.2.5. HC-SR04 Ultrasonik Sensör 133
4.2.6. MQ-2 Gaz Sensörü 133
4.3. HAREKET KONTROLÜ UYGULAMALARI 132
4.3.1. Güneş Panelleri 132
4.3.2. Rüzgâr Türbinleri 132
ÖLÇME VE DEĞERLENDİRME 178



5. ÖĞRENME BİRİMİ: TEMEL KUMANDA VE GÜÇ DEVRELERİ

5.1. KUMANDA VE GÜÇ DEVRE ELEMANLARI 180
5.1.1. Kumanda Butonları 180
5.1.2. Paket Şalterler 180
5.1.3. Sınır Anahtarları 181
5.1.4. Sinyal Lambaları 181
5.1.5. Sensörler 181
5.1.6. Sigortalar 182
5.1.7. Röleler 182
5.1.7.1. Zaman Röleleri 183
5.1.7.2. Koruma Röleleri 183



İÇİNDEKİLER

5.1.8. Kontaktörler 184
5.1.9. Katı Hal Röleleri 185
5.1.10. Kumanda ve Güç Devrelerinde Kullanılan Kablolar 184
5.1.11. Asenkron Motorlar 186
5.1.11.1. Asenkron Motorların Yapısı 186
5.1.11.2. Asenkron Motorların Çalışma Prensibi 186
5.1.11.3. Asenkron Motor Çeşitleri 187
5.1.11.4. Üç Fazlı Asenkron Motorların Avantaj ve Dezavantajları 187
5.1.12. Kumanda ve Güç Devre Sembolleri 189
5.1.13. Kumanda ve Güç Devre Şeması Çizimi 191
5.1.13.1. Kumanda Devrelerinin Çizimi 191
5.1.13.2. Güç Devrelerinin Çizimi 191
5.2. KUMANDA VE GÜÇ DEVRE UYGULAMALARI 194
5.2.1. Üç Fazlı Asenkron Motor Bağlantıları 194
5.2.2. Üç Fazlı Asenkron Motorların Çalışma Özellikleri 195
ÖLÇME VE DEĞERLENDİRME 216

6. ÖĞRENME BİRİMİ: PLC SEÇİMİ



6.1. YAPI VE ÇALIŞMA PRENSİBİNE GÖRE PLC SEÇİMİ 218
6.1.1. PLC'nin Yapı Özellikleri 218
6.1.2. PLC'nin Çalışma Prensibi 220
6.1.3. PLC'nin Kullanım Alanları 220
6.1.4. PLC Seçimi 221
6.2. PLC'NİN GİRİŞ-ÇIKIŞ VE ÇEVRE BİRİMİ BAĞLANTILARI 223
ÖLÇME VE DEĞERLENDİRME 232

7. ÖĞRENME BİRİMİ: PLC PROGRAMLAMA UYGULAMALARI



7.1. TEMEL SEVİYE PLC KOMUTLARI KULLANILARAK PROGRAMLAMA YAPMA ... 234
7.1.1. PLC Programlama Mantığı ve Yazılım Dilleri 234
7.1.2. Klasik Kumanda Devrelerinin Ladder Diyagramına Dönüşümü 234
7.1.3. PLC Programlamada Hafıza Alanları ve Veri Tipleri 235
7.1.4. PLC Programlamada Giriş ve Çıkış Adreslemeleri 236
7.1.5. Temel Programlama Komutları 236
7.1.5.1. Bit Lojik Komutları 236
7.1.5.2. Zamanlayıcı (Timer) Komutları 238
7.1.5.3. Sayıcı Komutları 239
7.2. DOKUNMATİK EKRAN UYGULAMALARI 277
7.2.1. Operatör Panelinin Yapısı 277
7.2.2. Operatör Panelinin Çalışması 278
7.2.3. Operatör Paneli Çeşitleri 278
7.2.4. Operatör Panelinin Montajı ve Panel Bağlantıları 278
7.2.5. Operatör Panelinin Programlama Editörü Menüleri 279
7.2.5.1. Ana Ekran 279
7.2.5.2. Menü Alanı 280
7.2.5.3. Proje Gezgini, Nesne Denetçisi ve Programlama Alanı 281
7.2.5.4. Nesne Alanı 282
7.2.5.5. Nesne Editörü 283
7.2.6. Projeye Panel Eklenmesi ve İletişim Ayarları 284
7.2.7. Operatör Paneli Proje Uygulaması 285

İÇİNDEKİLER

7.3. FREKANS İNVERTÖRLERİ İLE ASENKRON MOTOR KONTROLÜ 296
7.3.1. Sürücülerin Güç ve Kontrol Bağlantıları 296
7.3.2. Sürücü Parametreleri ve Parametre Giriş İşlemleri 296
7.3.3. Sürücü Kontrol Yöntemleri 297
7.3.4. PLC İle Sürücü Kontrolü 298
7.4. PLC İLE SERVO MOTORLARIN KONTROLÜ 307
7.4.1. Servo Motor Sürücüleri 308
7.4.2. Sürücü Bağlantıları 308
7.4.3. Servo Motor Sürücüsünün Kontrol Paneli 310
7.4.4. Motor Parametreleri ve Parametre Ayarları 310
7.4.5. Servo Motorların Tur/Pals Hesabı 312
7.4.6. Servo Motorların PLC İle Kontrolü 312
ÖLÇME VE DEĞERLENDİRME 320

8. ÖĞRENME BİRİMİ: ENERJİ SANTRALLERİNİN ELEKTRİK VE HAREKET SİSTEMLERİ

8.1. ENERJİ SANTRALİNİN ELEKTRİK VE ŞALT SAHASI SİSTEMİ 322
8.1.1. Elektrik Santrali Sistemi 322
8.1.2. Enerji Santralinin Şalt Sahası Sistemi 323
8.2. GÜNEŞ ROTASI VE IŞINIMI, MERKEZÎ HAREKET SİSTEMİ 324
8.2.1. Güneş Pili Panelinin Açısını Hesaplama 325
8.3. RÜZGÂR SANTRALLERİ 326
8.3.1. Rüzgâr Santrallerinin Kontrol Sistemi 327
8.3.2. Rüzgâr Türbininin Kanat Açı Kontrolü 328
8.3.3. Rüzgâr Türbininin Yön Kontrolü 328
ÖLÇME VE DEĞERLENDİRME 338



9. ÖĞRENME BİRİMİ: SCADA

9.1. FOTOVOLTAİK (GÜNEŞ) PANELİN HAREKET KONTROLÜ 340
9.2. TÜRBİN HAREKET KONTROLÜ 345
9.3. ENERJİ SANTRALİNİN GÜVENLİK KONTROLÜ 349
9.4. ENERJİ SANTRALİNİN ŞALT SAHASI MANEVRA KONTROLÜ 353
ÖLÇME VE DEĞERLENDİRME 363



CEVAP ANAHTARI 364
KAYNAKÇA 365

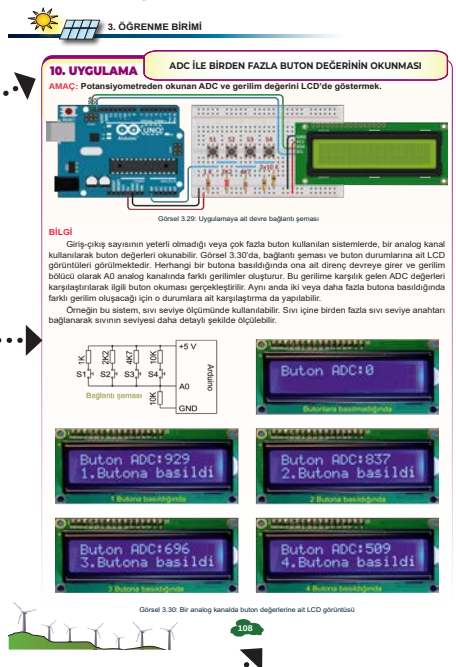
DERS MATERYALİNİN TANITIMI

Öğrenme biriminin ismini ve numarasını gösterir.

Uygulamanın adını, amacını ve numarasını gösterir.

Uygulamaya ait görselleri, bilgileri ve uygulamanın içeriğini gösterir.

Sayfa numarasını gösterir.

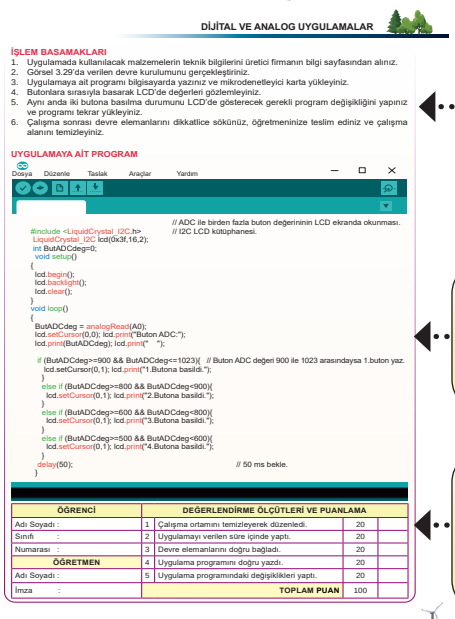


Öğrenme biriminin içeriklerini gösterir.

Uygulamanın işlem basamaklarını ve sorularını gösterir.

Uygulamaya ait programın kodlarını gösterir.

Uygulamanın değerlendirme ölçütlerini ve puanlamasını gösterir.



Öğrenme biriminin numarasını gösterir.

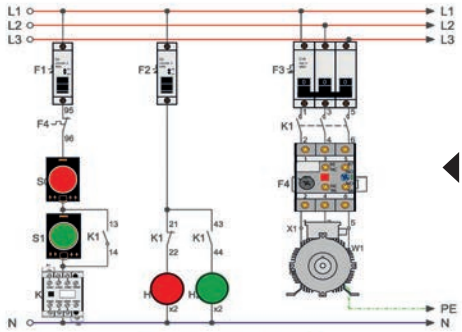
Öğrenme biriminin ismini gösterir.

Öğrenme biriminin karekodunu gösterir.

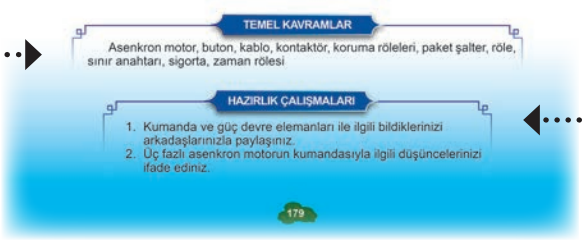
Öğrenme birimindeki konuları gösterir.



Öğrenme biriminin giriş kapağı görseli.



Öğrenme birimindeki kavramları gösterir.



Öğrenme birimine ait hazırlık çalışmalarını gösterir.

DERS MATERYALİNİN TANITIMI

Konuyla ilgili örnekleri gösterir.

ÖRNEK

$(1C8)_{16} = (?)_{10}$

$(1C8)_{16} = 8.16^0 + 12.16^1 + 1.16^2 = 8.1 + 12.16 + 1.256 = 8 + 192 + 256 = (456)_{10}$

Görsellerin kaynakçasına ulaşmak için karekodu okutunuz.



Ders materyalinin kitap.eba.gov.tr adresinde yer alan görsel kaynakçasına yönlendiren karekod son bölümde verilmiştir.

http://kitap.eba.gov.tr/karekod/Kaynak.php?KOD=1992

Öğrenme birimi ile ilgili ölçme ve değerlendirme sorularını gösterir.



4. ÖĞRENME BİRİMİ



ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki cümlelerin başındaki boşluğa cümleler doğru ise (D), yanlış ise (Y) yazınız.

1. (…) Frekans (f) saniyede oluşan periyot (T) sayısıdır.
2. (…) DC devrede güç ölçümü; gerilim, akım ve direncin çarpımına eşittir.
3. (…) Herhangi bir yük, mikrodenetleyici portlarına doğrudan bağlanabilir.

B) Aşağıdaki cümlelerde boş bırakılan yerlere uygun sözcüğü yazınız.

4. DC motorların devir yönünü değiştirmek için uygulanan DC gerilimin ..... değiştirilmelidir.
5. Servo motorlar..... sinyaliyle kontrol edilir.
6. analogReference(INTERNAL); fonksiyonuyla arduino için dâhilî AREF ..... olarak ayarlanır.

1. ÖĞRENME BİRİMİ

SAYISAL İŞLEMLER

KONULAR

1.1. SAYI SİSTEMLERİ
1.2. LOJİK KAPILAR



TEMEL KAVRAMLAR

Binary, desimal, lojik kapılar, mantık devreleri, oktal, sayı sistemleri, sayı sistemi dönüşümleri

HAZIRLIK ÇALIŞMALARI

- Bilgisayarlar hangi sayı sisteminde çalışır? Araştırınız.

1.1. SAYI SİSTEMLERİ

Sayıları temsil eden ve simgelerle matematiksel gösterim yapan sisteme **sayı sistemi** denir. Sayı sistemleri bilgisayarlarda, mikroişlemcilerde, invertörlerde ve yapay zekânın uygulandığı birçok alanda kullanılır.

1.1.1. Sayı Sistemlerinin Çeşitleri

Günümüzde birçok sayı sistemi bulunmaktadır. Sayı sistemlerinin çeşitleri ve özellikleri aşağıda verilmiştir.

1.1.1.1. Binary (İkili) Sayı Sistemleri

0 ve 1 rakamlarının kullanımıyla oluşan iki elemanlı sayı sistemine **binary (ikili) sayı sistemi** denir. Bu sayı sisteminin tabanı 2'dir ve $(xxx)_2$ şeklinde gösterilir. Her sayı dijit olarak ifade edilir ve basamaklar 2'nin kuvveti olarak yazılır. Bilgisayar ve işlemciler bu sayı sistemini kullanır.

$$\left(\underset{n}{a}\ \ldots\ \underset{2}{c}\ \underset{1}{d}\ \underset{0}{e}\right)_2 = a.2^n + \ldots + c.2^2 + d.2^1 + e.2^0$$

ÖRNEK

$(1011)_2 = 1.2^0 + 1.2^1 + 0.2^2 + 1.2^3$

$(1011)_2 = 1 + 2 + 0 + 8$

$(1011)_2 = 11$ (onlu sayı sisteminde karşılığı)

SORU

$(1001)_2 = ?$

*Örneğe uygun olarak soruyu çözünüz.

1.1.1.2. Oktal (Sekizli) Sayı Sistemleri

0, 1, 2, 3, 4, 5, 6, 7 rakamlarının tamamının kullanımıyla oluşan, toplamda 8 elemanlı sisteme **oktal (sekizli) sayı sistemi** denir. Bu sayı sisteminin tabanı 8'dir ve $(xxx)_8$ şeklinde gösterilir. Her sayı dijit olarak ifade edilir ve basamaklar 8'in kuvveti olarak yazılır.

$$\left(\underset{n}{a}\ \ldots\ \underset{2}{c}\ \underset{1}{d}\ \underset{0}{e}\right)_8 = a.8^n + \ldots + c.8^2 + d.8^1 + e.8^0$$

ÖRNEK

$(472)_8 = 2.8^0 + 7.8^1 + 4.8^2$

$(472)_8 = 2 + 56 + 256$

$(472)_8 = 314$ (onlu sayı sisteminde karşılığı)

SORU

$(720)_8 = ?$

*Örneğe uygun olarak soruyu çözünüz.

1.1.1.3. Desimal (Onlu) Sayı Sistemleri

0, 1, 2, 3, 4, 5, 6, 7, 8, 9 rakamlarının tamamının kullanımıyla oluşan, toplamda 10 elemanlı sisteme **desimal (onlu) sayı sistemi** denir. Bu sayı sisteminin tabanı 10'dur ve $(xxx)_{10}$ şeklinde gösterilir. Her sayı dijit olarak ifade edilir ve basamaklar 10'un kuvveti olarak yazılır. Normal sayma sayılarından oluşur. Günlük hayatta bu sayı sistemi kullanılır. Sayının alt tarafına $(xxx)_{10}$ ifadesi yazılmaz, genel olarak direkt sayı yazılır ve xxx şeklinde ifade edilir.

$$(a\ldots cde)_{10} = a.10^n + \ldots + c.10^2 + d.10^1 + e.10^0$$
(n … 2 1 0)

ÖRNEK

$(892)_{10} = 2.10^0 + 9.10^1 + 8.10^2$

$(892)_{10} = 2 + 90 + 800$

$(892)_{10} = 892$ (Onluk sayı sisteminde $(892)^{10}$ şeklinde kullanılmaz, direkt sayı yazılır, 892 olarak ifade edilir.)

SORU

$(254)_{10} = ?$

*Örneğe uygun olarak soruyu çözünüz.

1.1.1.4. Heksadesimal (Onaltılı) Sayı Sistemleri

0, 1, 2, 3, 4, 5, 6, 7, 8, 9 rakamlarından ve A (10), B (11), C (12), D (13), E (14), F (15) harflerinden meydana gelen 10 adet sayı, 6 adet harften oluşan toplamda 16 elemanlı sayı sistemine **heksadesimal (Onaltılı) sayı sistemi** denir. Bu sayı sisteminin tabanı 16'dır, $(xxx)_{16}$ şeklinde gösterilir. Her sayı dijit olarak ifade edilir ve basamaklar 16'nın kuvveti olarak yazılır. İkili sayı sisteminde sayılar uzun olduğundan bilgilerin daha kısa gösterilmesi amacıyla özellikle bilgisayar dilinde kullanılır.

$$(a\ldots cde)_{16} = a.16^n + \ldots + c.16^2 + d.16^1 + e.16^0$$
(n … 2 1 0)

ÖRNEK

$(8B7)_{16} = 7.16^0 + B.16^1 + 8.16^2$

$(8B7)_{16} = 7.16^0 + 11.16^1 + 8.16^2$

$(8B7)_{16} = 7 + 176 + 2048$

$(8B7)_{16} = 2231$ (onlu sayı sisteminde karşılığı)

SORU

$(ABC)_{16} = (\ ?\ )_{10}$

*Örneğe uygun olarak soruyu çözünüz.

1.1.2. Sayı Sistemlerinin Birbirine Dönüşümü

Sayı sistemleri arasında dönüşümlerin yapılması gerekir. Bilgisayarlar Onaltılı ve ikili sayı sistemini kullanır. Aşağıda her bir sayı sistemi arasındaki dönüşümler verilmiştir.

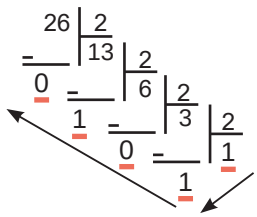
1.1.2.1. Desimal Sayının Binary Sayıya Dönüştürülmesi

Kural olarak desimal sayı başka bir sisteme dönüştürülecekse yeni oluşacak sayı, sistemin taban değerine sürekli bölünür, bölüm sonucunda elde edilen kalanların tersten sıralanmasıyla yeni sistemdeki sayı oluşturulur.

ÖRNEK

$(26)_{10} = (?)_2$

26 sayısı, bölüm 2'nin altına ininceye kadar bölünür. En sonunda, en son bölümden başlayarak kalanlar sırayla yazılır.



$(26)_{10} = (11010)_2$

SORU

$(36)_{10} = (\ ?\ )_2$

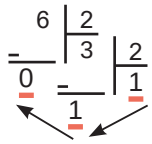
*Örneğe uygun olarak soruyu çözünüz.

Eğer sayıda, ondalık kısım var ise ondalık kısım 2 ile çarpılır, tam sayı kenara atılır, kalan kısım tekrar 2 ile çarpılır. 0 değeri bulunana dek işleme devam edilir.

ÖRNEK

$(6,65)_{10} = (?)_2$

$(6,65)^{10} = (6)^{10} + (0,65)^{10} = (?)^2 + (?)^2 = (?)^2$



$(6)_{10} = (110)_2$

0,65 x 2	=	1,30	1
0,30 x 2	=	0,60	0
0,60 x 2	=	1,20	1
0,20 x 2	=	0,40	0
:		:	:
$(0,65)_{10}$	=	$(1010....)_2$	

$(6,65)^{10} = (6)^{10} + (0,65)^{10} = (110^{)2} + (0,1010\ldots)^2 = (110,1010\ldots)^2$

SORU

$(27,45)_{10} = (\ ?\ )_2$

*Örneğe uygun olarak soruyu çözünüz.

Not: 10 tabanında gösterilen kesirli kısmın 2 tabanında gösterilebilmesi için yapılan çarpma işleminde, kesirli kısım tam şekline yani 0 değerine varmayabilir. Bu işlemin sonu yoktur. Bu sebeple bir yuvarlatma yapılarak işlem sonlandırılır.

1.1.2.2. Desimal Sayının Oktal Sayıya Dönüştürülmesi

Desimal sayı oktal sayıya çevrilirken oktal sayının tabanı olan 8'e bölünür. Kalanlar, bir kenara kaydedilir ve en sonunda tersten sekizlik sayı olarak yazılır. Eğer sayıda ondalık kısım var ise ondalık kısım 8 ile çarpılır, tam sayı kenara atılır, kalan kısım tekrar 8 ile çarpılır. 0 değeri bulunana dek işleme devam edilir.

ÖRNEK

$(69)_{10} = (?)_8$

69 | 8 → 8 | 8 → 1 | 8 → 0

5, 1, 0

$(69)_{10} = (105)_8$

ÖRNEK

$(12,84)_{10} = (?)_8$

$(12,84)_{10} = (?)_8 = (12,84)_{10} = (12)_{10} + (0,84)_{10} = (?)_8 + (?)_8 = (?)_8$

12 | 8 → 1 | 8 → 4, 1

$(12)_{10} = (14)_8$

0,84 x 8	= 6,72	6
0,72 x 8	= 5,76	5
0,76 x 8	= 6,08	6
0,08 x 8	= 0,64	0
:	:	:
$(0,84)_{10}$	= $(6560....)_8$	

$(12,84)_{10} = (12)_{10} + (0,84)_{10} = (14)_8 + (0,6560\ldots)_8 = (14,6560\ldots)_8$

SORU

$(70)_{10} = (?)_8$

*Örneğe uygun olarak soruyu çözünüz.

SORU

$(13,90)_{10} = (?)_8$

*Örneğe uygun olarak soruyu çözünüz.

1.1.2.3. Desimal Sayının Heksadesimal Sayıya Dönüştürülmesi

Desimal sayı heksadesimal sayıya çevrilirken heksadesimal sayının tabanı olan 16'ya bölünür. Kalanlar bir kenara yazılarak tersten heksadesimal sayı olarak yazılır. Eğer sayıda ondalık kısım var ise ondalık kısım 16 ile çarpılır, tam sayı kenara atılır, kalan kısım tekrar 16 ile çarpılır. 0 değeri bulunana dek işleme devam edilir.

ÖRNEK

$(59)_{10} = (?)_{16}$

59 | 16 → 3 | 16 → 11, 3

11 → B

$(59)_{10} = (3B)_{16}$

ÖRNEK

$(28,67)_{10} = (?)_{16}$

$(28,67)_{10} = (?)_{16} = (28,67)_{10} = (28)_{10} + (0,67)_{10} = (?)_{16} + (?)_{16} = (?)_{16}$

28 | 16 → 1 | 16 → 12, 1

12 → C

$(28)_{10} = (1C)_{16}$

0,67 x 16	= 10,72	10	→ A
0,72 x 16	= 11,52	11	→ B
0,52 x 16	= 8,32	8	
0,32 x 16	= 5,12	5	
:	:	:	
$(0,84)_{10}$	= $(6560....)_8$		

$(28,67)_{10} = (28)_{10} + (0,67)_{10} = (1C)_{16} + (0,AB85\ldots)_{16} = (C1,AB85\ldots)_{16}$

SORU

$(974)_{10} = (?)_{16}$

*Örneğe uygun olarak soruyu çözünüz.

SORU

$(785)_{10} = (?)_{16}$

*Örneğe uygun olarak soruyu çözünüz.

1.1.2.4. Binary Sayının Desimal Sayıya Dönüştürülmesi

Binary sayıyı desimal sayıya çevirmek için 2'li taban aritmetiği kullanılır.

ÖRNEK

$(101011)_2 = (?)_{10}$

$(101011)_2 = 1.2^0 + 1.2^1 + 0.2^2 + 1.2^3 + 0.2^4 + 1.2^5$

$(101011)_2 = 1.1+1.2+0+1.8+0+1.32 = 1+2+8+32 = (43)_{10}$

ÖRNEK

$(101,011)_2 = (?)_{10}$

$(101,011)_2 = 1.2^{-3} + 1.2^{-2} + 0.2^{-1} + 1.2^0 + 0.2^1 + 1.2^2$

$(101,011)_2 = 0,125 + 0,25 + 1 + 4 = (5,375)_{10}$

SORU

$(10111)_2 = (?)_{10}$

*Örneğe uygun olarak soruyu çözünüz.

SORU

$(111,11)_2 = (?)_{10}$

*Örneğe uygun olarak soruyu çözünüz.

1.1.2.5. Binary Sayının Oktal Sayıya Dönüştürülmesi

Binary sayıyı sekizlik (oktal) sayıya çevirmek için binary sayı, sağ taraftan yani LSD (Least Significant Dijit/En küçük değerlikli sayı) olan taraftan itibaren üçer gruba ayrılır ve her grubun oktal karşılığı yazılır. Eğer tekli veya ikili sayı denk gelirse yanına 0 ilave edilir. Ayrıca ondalık kısım varsa ondalık kısımdan itibaren üçerli gruplandırma dâhil edilir.

ÖRNEK

$(1011011)_2 = (?)_8$

$(1011011)_2 = (?)_8$

$(133)_8$

4 2 1	4 2 1	4 2 1
001	011	011
1	3	3

ÖRNEK

$(10101,11001)_2 = (?)_8$

$(10101,11001)_2 = (010101,110010)_2 = (25,62)_8$

010101,110010
2 5 6 2

SORU

$(1011101)_2 = (?)_8$

*Örneğe uygun olarak soruyu çözünüz.

SORU

$(101,1101)_2 = (?)_8$

*Örneğe uygun olarak soruyu çözünüz.

1.1.2.6. Binary Sayının Heksadesimal Sayıya Dönüştürülmesi

İkili (binary) sayıyı Onaltılı sayı sistemine çevirmek için verilen ikili sayı, sağdan başlamak üzere dörder gruba ayrılır. Ayrılan her grubun Onaltılı (heksadesimal) karşılığı yazılır. Eğer tekli, ikili veya üçlü sayı denk gelirse yanına 0 ilave edilir. Ayrıca ondalık kısım varsa ondalık kısımdan itibaren dörderli gruplandırma dâhil edilir.

ÖRNEK

$(101110011)_2 = (?)_{16}$

$(101110011)_2 = (101110011)_2$

8 4 2 1 | 8 4 2 1 | 8 4 2 1

0001 | 0111 | 0011

1 | 7 | 3

$(173)_{16}$

ÖRNEK

$(1010101,1101101)_2 = (?)_{16}$

$(1010101,1101101)_2 = (01010101,11011010)_2 = (55, DA)_{16}$

5 5 13 10

SORU

$(111110101)_2 = (?)_{16}$

*Örneğe uygun olarak soruyu çözünüz.

SORU

$(11111,101101)_2 = (?)_{16}$

*Örneğe uygun olarak soruyu çözünüz.

1.1.2.7. Oktal Sayının Desimal Sayıya Dönüştürülmesi

Oktal sayıyı desimal sayıya çevirmek için 8'li taban aritmetiği kullanılır.

ÖRNEK

$(72)_8 = (?)_{10}$

$(72)_8 = 2.8^0 + 7.8^1 = 2.1 + 7.8 = 2 + 56 = (58)_{10}$

ÖRNEK

$(64,5)_8 = (?)_{10}$

$(64,5)_2 = 5.8^{-1} + 4.8^0 + 6.8^1 = 5.0,125 + 4.1 + 6.8 = 0,625 + 4 + 48 = (52,625)_{10}$

SORU

$(714)_8 = (?)_{10}$

*Örneğe uygun olarak soruyu çözünüz.

SORU

$(55,4)_8 = (?)_{10}$

*Örneğe uygun olarak soruyu çözünüz.

1.1.2.8. Oktal Sayının Binary Sayıya Dönüştürülmesi

Oktal sayıyı binary sayıya çevirmek için oktal sayının her biri 3 bitlik binary sayıya çevrilir ya da oktal sayı desimal sayıya çevrilir, ardından bu desimal sayı binary sayıya çevrilir. Bu tür dönüştürmelerde binary dönüşümleri daha kolaydır.

ÖRNEK

$(76)_8 = (?)_2$

1. Yol: Her bir rakamın üç bit binary karşılıkları oluşturulur.

$(76)_8 = (76)_8 \rightarrow (111110)_2$

1 1 1 1 1 1
4 2 1 4 2 1

2. Yol: Her sayı 8'lik tabanla 10 tabanına ve oradan ikilik tabana çevrilir.

$(76)_8 = 6.8^0 + 7.8^1 = 6.1 + 7.8 = 56 = (62)_{10}$

$(62)_{10} = \begin{pmatrix} 64\ 32\ 16\ 8\ 4\ 2\ 1 \\ X\ X\ X\ X\ X\ X\ X \end{pmatrix}_2 = \begin{pmatrix} 64\ 32\ 16\ 8\ 4\ 2\ 1 \\ 0\ 1\ 1\ 1\ 1\ 1\ 0 \end{pmatrix}_2 = (111110)_2$

1.1.2.9. Oktal Sayının Heksadesimal Sayıya Dönüştürülmesi

Oktal sayıyı heksadesimal sayıya çevirmek için ya binary sayıya çevirdikten sonra heksadesimal sayıya çevirmek ya da desimal sayıya çevirdikten sonra heksadesimal sayıya çevirmek gerekir. Bu tür dönüştürmelerde binary sayıya dönüşümler daha kolaydır.

ÖRNEK

$(52)_8 = (?)_{16}$

1. Yol: Her bir rakamın üç bit binary karşılıkları oluşturulur.

$(52)_8 = (52)_8 \rightarrow (101010)_2 \rightarrow (101010)_2 \rightarrow (2A)_{16}$

1 0 1 0 1 0
4 2 1 4 2 1

2 10=A

2. Yol: Her sayı 8'lik tabanla 10 tabanına ve oradan ikilik tabana çevrilir.

$(52)_8 = 2.8^0 + 5.8^1 = 2.1 + 5.8 = 2 + 40 = (42)_{10}$

42 | 16
10 | 2
A

$(42)_{10} = (2A)_{16}$

SORU

$(47)_8 = (?)_8$

*Örneğe uygun olarak soruyu çözünüz.

SORU

$(673)_8 = (?)_{16}$

*Örneğe uygun olarak soruyu çözünüz.

1.1.2.10. Heksadesimal Sayının Desimal Sayıya Dönüştürülmesi

Heksadesimal sayıyı desimal sayıya çevirmek için 16'lı taban aritmetiği kullanılır. Burada harfler varsa her harfi kendi içinde (örneğin B=11 gibi) sayıya çevirmek gerekir.

ÖRNEK

$(1C8)_{16} = (?)_{10}$

$(1C8)_{16} = 8.16^0 + 12.16^1 + 1.16^2 = 8.1 + 12.16 + 1.256 = 8 + 192 + 256 = (456)_{10}$

ÖRNEK

$(51,F)_{16} = (?)_{10}$

$(51,F)_{16} = 15.16^{-1} + 1.16^0 + 5.16^1 = 15.0,0625 + 1.1 + 5.16 = 0,9375 + 1 + 80 = (81,9375)_{10}$

SORU

$(43,A)_{16} = (?)_{10}$

*Örneğe uygun olarak soruyu çözünüz.

SORU

$(1AB)_{16} = (?)_{10}$

*Örneğe uygun olarak soruyu çözünüz.

1.1.2.11. Heksadesimal Sayının Binary Sayıya Dönüştürülmesi

Heksadesimal (Onaltılık) sayıyı binary sayıya çevirme işlemi yapılırken düşük ağırlıklı değerden itibaren her sayı dört bitlik gruplara ayrılır ve sayının karşılığı bulunur.

ÖRNEK

$(6D1)_{16} = (?)_2$

$$(6D1)_{16} = \begin{pmatrix} 8\,4\,2\,1 & 8\,4\,2\,1 & 8\,4\,2\,1 \\ XXXX & XXXX & XXXX \end{pmatrix}_2 = \begin{pmatrix} 8\,4\,2\,1 & 8\,4\,2\,1 & 8\,4\,2\,1 \\ XXXX & XXXX & XXXX \\ 6 & 13 & 1 \end{pmatrix}_2$$

$$(6D1)_{16} = \begin{pmatrix} 8\,4\,2\,1 & 8\,4\,2\,1 & 8\,4\,2\,1 \\ 0\,1\,1\,0 & 1\,1\,0\,1 & 0\,0\,0\,1 \\ 6 & 13 & 1 \end{pmatrix}_2 = (11011010001)_2$$

SORU

$(CAB)_{16} = (?)_2$

*Örneğe uygun olarak soruyu çözünüz.

1.1.2.12. Heksadesimal Sayının Oktal Sayıya Dönüştürülmesi

Heksadesimal sayıyı oktal sayıya çevirmek için ya binary sayıya çevirdikten sonra oktal sayıya çevirmek ya da desimal sayıya çevirdikten sonra oktal sayıya çevirmek gerekir. Bu tür dönüştürmelerde binary sayıya dönüşüm daha kolaydır.

ÖRNEK

$(2E)_{16} = (?)_8$

1. Yol: 16 tabanlık değer binary değere ve oktal sayıya dönüştürülür.

$(1C8)_{16} = 8.16^0 + 12.16^1 + 1.16^2 = 8.1 + 12.16 + 1.256 = 8 + 192 + 256 = (456)_{10}$

$(2E)_{16} = \left(\overset{8\,4\,2\,1}{XXXX}\ \overset{8\,4\,2\,1}{XXXX}\right)_2 = \left(\underset{2}{\overset{8\,4\,2\,1}{XXXX}}\ \underset{14}{\overset{8\,4\,2\,1}{XXXX}}\right)_2 = \left(\underset{2}{\overset{8\,4\,2\,1}{0010}}\ \underset{14}{\overset{8\,4\,2\,1}{1110}}\right)_2 = (101110)_2$

$(101110)_2 = \left(\underset{5}{\overset{4\,2\,1}{1\,0\,1}}\ \underset{6}{\overset{4\,2\,1}{1\,1\,0}}\right)_2 = (56)_8$

2. Yol: 16 tabanlık değer desimal değere ve oktal sayıya dönüştürülür.

$(2E)_{16} = 14.16^0 + 2.16^1 = 14.1 + 2.16 = 14 + 32 = (46)_{10}$

$46 \mid 8$

$-\ \ \ \ \ 5$

6

$(46)_{10} = (56)_8$

SORU

$(1AE)_{16} = (?)_8$

*Örneğe uygun olarak soruyu çözünüz.

Tablo 1.1'de sayıların dönüşümü verilmiştir.

Tablo 1.1: Sayıların Dönüşüm Tablosu

Desimal Sayı	Binary Sayı	Oktal Sayı	Heksadesimal Sayı	Desimal Sayı	Binary Sayı	Oktal Sayı	Heksadesimal Sayı
0	0000 0000	0	0	11	0000 1011	13	B
1	0000 0001	1	1	12	0000 1100	14	C
2	0000 0010	2	2	13	0000 1101	15	D
3	0000 0011	3	3	14	0000 1110	16	E
4	0000 0100	4	4	15	0000 1111	17	F
5	0000 0101	5	5	16	0001 0000	20	10
6	0000 0110	6	6	17	0001 0001	21	11
7	0000 0111	7	7	18	0001 0010	22	12
8	0000 1000	10	8	19	0001 0011	23	13
9	0000 1001	11	9	20	0001 0100	24	14
10	0000 1010	12	A	21	0001 0101	25	15

ÖRNEK

Aşağıdaki tabloda her satır ve sütuna denk gelen kısma ilgili sayı birimini hesaplayarak yazınız.

No	Binary Sayı	Desimal Sayı	Oktal Sayı	Heksadesimal Sayı
1	……	412,50	……	……
2	……	……	……	7EB,A2
3	11100110011101110	……	……	……
4	……	……	76,41	……
5	111100111101,1	……	……	……
6	1011,01	……	……	……
7	……	203	……	……
8	……	……	……	1F2
9	……	……	256	……
10	11000001011101	……	……	……

1.1.3. Binary Sayılarda Aritmetik İşlemler

İkili sayı sistemlerinde toplama, çıkarma ve çarpma işlemlerinin yapılması devre dizaynlarında kullanılacağı için önemlidir.

1.1.3.1. Binary Sayılarda Toplama

Binary sayılarda toplama kuralları şu şekildedir:

0+0=0

0+1=1

1+0=1

1+1=0 elde 1

İkili sayıların toplanmasında toplanacak sayılar **sağdan hizalanmış olarak alt alta** yazılır. Toplama işlemi en sağdaki sütundan başlanarak yapılır. Toplama işlemi sırasında bulunan elde 1 soldaki basamağa aktarılır.

0+1=1

1+0=1

1+1=0 elde 1

İkili sayıların toplanmasında toplanacak sayılar sağdan hizalanmış olarak alt alta yazılır. Toplama işlemi en sağdaki sütundan başlanarak yapılır. Toplama işlemi sırasında bulunan elde 1 soldaki basamağa aktarılır.

ÖRNEK

$$\begin{array}{r} (11)_2 \\ +\ (10)_2 \\ \hline (101)_2 \end{array}$$

- Önce sayılar sağdan aynı hizada yazılır.
- İlk olarak en sağdan 1+0 toplama işlemi yapılır.
- Binary sayılarda toplama kuralları uygulanır.
- 1+0=1 Bulunan 1 sonuç satırına yazılır.
- Sonra sol taraftaki sayılar toplanır. 1+1=0 elde 1 işlemi sonunda, sonuç 0, elde 1 bulunur. Bulunan 0, sonuç satırına yazılır, elde 1 ise sonuç satırının soluna yazılır.

ÖRNEK

$$\begin{array}{r} (110)_2 \\ +\ (11)_2 \\ \hline (1001)_2 \end{array}$$

- Önce sayılar sağdan aynı hizada yazılır.
- İlk olarak en sağdan 0+1 toplama işlemi yapılır.
- Binary sayılarda toplama kuralları uygulanır.
- 1+0=1 Bulunan 1 sonuç satırına yazılır.
- Sonra orta sütundaki 1 ile 1 toplanır (1+1=0 elde 1), sonuç 0, elde 1 işleminin sonunda, bulunan 0 sonuç satırına yazılır. Elde 1 ise soldaki sütuna kaydırılır ve burada bulunan 1 ile toplanır (1+1=0 elde 1). Sonuç 0, elde 1 işleminin sonunda, bulunan 0 sonuç satırına yazılır, elde 1 ise soldaki sütuna kaydırılır. Burada başka rakam bulunmadığı için bu 1 en alttaki sonuç satırının sağına yazılır.

SORU

$$\begin{array}{r} (111)_2 \\ +\ (11)_2 \\ \hline (???)_2 \end{array} \qquad \begin{array}{r} (111)_2 \\ +\ (111)_2 \\ \hline (????)_2 \end{array} \qquad \begin{array}{r} (101)_2 \\ +\ (101)_2 \\ \hline (????)_2 \end{array} \qquad \begin{array}{r} (1111)_2 \\ +\ (1111)_2 \\ \hline (?????)_2 \end{array} \qquad \begin{array}{r} (10111)_2 \\ +\ (11011)_2 \\ \hline (??????)_2 \end{array}$$

*Örneğe uygun olarak soruyu çözünüz.

1.1.3.2. Binary Sayılarda Çıkarma

Binary sayılarda çıkarma kuralları şu şekildedir:

0-0=0

1-0=1

0-1=1 (Bu durumda bir soldaki sütundan 1 alınır ve bu sütunda 2 olarak adlandırılır.)

1-1=0

İkili sayıların çıkarılması işleminde çıkartılacak sayılar **sağdan hizalanmış olarak alt alta** yazılır. Çıkarma işlemi en sağdaki sütundan başlanarak yapılır. Bir sütunda üstte 0 altta 1 varsa soldaki sütundan alınan 1 bu sütuna 2 olarak aktarılır (Onlu sayılarda çıkarma işlemi sırasında soldaki basamaktan alınan 1'in sağdaki basamağa 10 olarak aktarıldığı hatırlanmalıdır.).

ÖRNEK

$$\begin{array}{r} (1101)_2 \\ -\ (11)_2 \\ \hline (1010)_2 \end{array}$$

- Önce sayılar sağdan aynı hizada yazılır.
- Çıkarma işlemine en sağdan başlanır ve 1-1=0 işlemi yapılarak 0 sonuç satırına yazılır.
- Sonra 0-1= işlemi yapılır. Bunun için bir önceki basamaktan ikilik alınır. Böylece 2-1=1 sonucu, sonuç satırına yazılır.
- Sağdan üçüncü basamaktan ikilik alındığı için burada kalan 0 sonuç satırına yazılır.
- Sağdan dördüncü basamaktaki 1 sonuç satırına yazılır.

SORU

$$\begin{array}{r} (1001)_2 \\ -\ (111)_2 \\ \hline (????)_2 \end{array} \qquad \begin{array}{r} (101)_2 \\ -\ (001)_2 \\ \hline (????)_2 \end{array} \qquad \begin{array}{r} (111)_2 \\ -\ (101)_2 \\ \hline (????)_2 \end{array} \qquad \begin{array}{r} (1101)_2 \\ -\ (1001)_2 \\ \hline (?????)_2 \end{array} \qquad \begin{array}{r} (10111)_2 \\ -\ (10011)_2 \\ \hline (??????)_2 \end{array}$$

*Örneğe uygun olarak soruyu çözünüz.

1.1.3.3. Binary Sayılarda Çarpma

Binary sayılarda çarpma kuralları şu şekildedir:

0x0=0

0x1=0

1x0=0

1x1=1

İkili sayıların çarpılmasında çarpılacak sayılar **sağdan hizalanmış olarak alt alta** yazılır. Çarpma kuralları uygulanır.

ÖRNEK

$$\begin{array}{r} (11001)_2 \\ x\ (11)_2 \\ \hline 11001 \\ +\ 11001 \\ \hline (101011)_2 \end{array}$$

- Önce sayılar sağdan aynı hizada yazılır.
- İlk olarak en sağdan 1x1 çarpma işlemi yapılır.
- Binary sayılarda çarpma kuralları uygulanır.
- Daha sonra ikinci sayı ile 1x0 çarpma işlemi yapılır.
- Bu kural tüm basamaklarda uygulanır.
- Sayılar toplanarak sonuç bulunur.

SORU

$(111)_2$	$(1110)_2$	$(10111)_2$	$(1101)_2$	$(101)_2$
x $(11)_2$	x $(10)_2$	x $(101)_2$	x $(110)_2$	x $(101)_2$
$(?????)_2$	$(?????)_2$	$(????????)_2$	$(???????)_2$	$(?????)_2$

*Örneğe uygun olarak soruyu çözünüz.

1.2. MANTIKSAL ENTEGRELER VE LOJİK KAPILAR

Sayısal (dijital) devrelerin tasarımında kullanılan temel devre elemanlarına **lojik kapılar (mantık kapıları)** adı verilir. Lojik kapıların lojik devre, mantık kapıları, mantık elemanları gibi birçok ismi bulunmaktadır. Basit bir sayısal elektronik devreden bilgisayarlara, İHA ve SİHA'lara, savunma sistemlerine kadar cihazların içindeki kontrol elemanlarının temeli çeşitli kapılara dayanmaktadır.

Lojik kapıların bir çıkışı ve birden fazla girişi vardır. Girişe gelen değere bağlı olarak çıkış ifadesi oluşur. Bir kapının çıkışını başka bir kapının girişine vererek veya kapıları birleştirerek birçok devre tasarlanabilir.

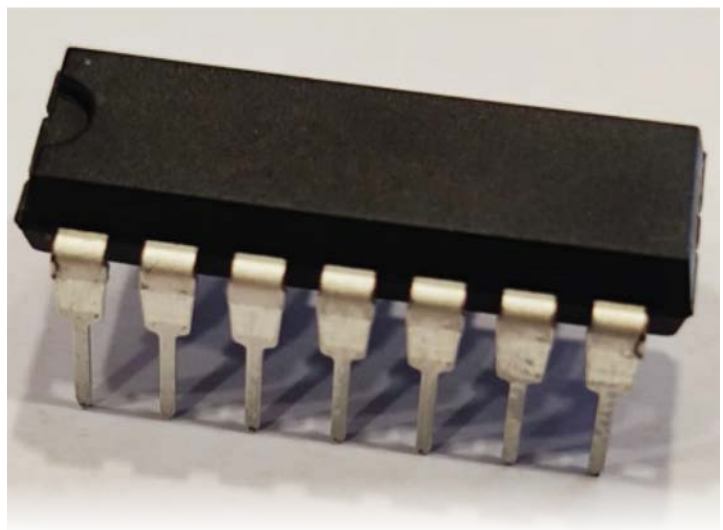
Entegre olarak imal edilen kapılar transistör, diyot ve direnç gibi elektronik devre elemanları kullanılarak yapılır (Tablo 1.3).

Lojik devrelerin en basit ve temel elemanı lojik kapılardır. VE (AND), VEYA (OR), DEĞİL (NOT), VE DEĞİL (NAND), VEYA DEĞİL (NOR), ÖZEL VEYA (EX-OR) ve ÖZEL VEYA DEĞİL (EX-NOR) kapıları vardır. Lojik kapıların kullanılması ile oluşturulan devrelere **lojik devreler** denir. Lojik kapı devrelerinde sadece iki gerilim seviyesi vardır. Birincisi lojik 1 yani yüksek seviye (+5 V), ikincisi ise lojik 0 yani düşük seviye (0 V)'dir. En basit matematik işlemleri olan toplama ve çıkarma, bu mantık kapıları yardımıyla yapılmaktadır.

Dijital elektronik sistemlerde entegreler önemli rol oynar. Özellikle temel mantık işlemleri olan toplama, çıkarma, çarpma gibi işlemlerde lojik kapılar tercih edilmektedir. Lojik kapıların bilinmesi sistemin daha kolay anlaşılmasını sağlar.

1.2.1. Lojik Entegrelerin Özellikleri

Entegre, kelime anlamı olarak bütünleştirmek demektir. Elektronik devrelerde bir veya birkaç elektronik elemanın kristal özelliğe sahip bir yarı iletken üzerinde bir araya getirilmesiyle oluşturulan malzemeye **entegre** denir (Görsel 1.1). IC (Integrated Circuit) olarak adlandırılır. Elektronik devre şemalarında entegreler "IC" ile gösterilir. Bu entegreler sayısal bilgileri işleyecek biçimde tasarlanmışsa bunlara **mantıksal (lojik) entegre** denir. Lojik 0 (Low) ve Lojik 1 (High) olarak işlem yapar.



Görsel 1.1: Lojik entegre

Birçok elektronik devre elemanının birleşmesiyle oluşan yeni yapıya **entegre devresi** denir (Tablo 1.2).

Tablo 1.2: Kapı Sayısına Göre Lojik Entegreler

Elemanlar	IEC	ANSI
SSI	Bünyesinde 1 ile 20 arasında lojik kapı bulundurur. Örneğin 7400 model entegre içerisinde 4 adet nand kapısı bulundurur (Tek IC içerisinde).	Küçük ölçekli
MSI	Bünyesinde 20 ile 100 arasında lojik kapı bulundurur. Örneğin flip-floplar, sayıcılar.	Orta ölçekli
LSI	Bünyesinde 100 ile 10.000 arasında lojik kapı bulundurur. Örneğin 4 ve 8 bitlik mikroişlemciler.	Büyük ölçekli
VLSI	Bünyesinde 10.000 den fazla lojik kapı bulundurur. Örneğin 16-32 bitlik mikroişlemciler, hafıza devreleri, bilgisayar devreleri.	
ULSI	Bünyesinde 1.000.000 ve üstü lojik kapı bulundurur. Çok hızlı bilgisayar ve işlemcilerde, SİHA ve İHA savunma sistemlerinde kullanılır.	Çok büyük ölçekli

Entegrelerin belli başlı özellikleri şunlardır:

- Bağlantıları kolay ve daha anlaşılırdır.
- Hızları yüksektir.
- Maliyetleri düşük ve boyutları küçüktür.
- Az güç tüketir ve gürültü başlıkları vardır.
- Daha sıcak ortamda çalışacaklarsa soğutulmaları için düzenekler konulmalıdır.
- Hata değerleri düşüktür hatta sıfırdır.

Entegrelerin parametreleri şunlardır:

- Çalışma ısı değeri
- Giriş gerilimi ve giriş akım değeri
- Pin ve uçlardaki sıcaklık değerleri
- Güç harcaması
- Çıkış kapasitesi ve yük sayısı
- Gürültü etkenleri ve bağışıklığı
- Yayılım gecikmesi ve yayılım hızı

1.2.2. Lojik Entegrelerin Çeşitleri

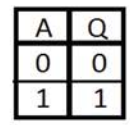
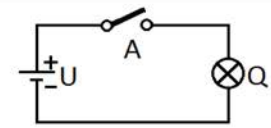
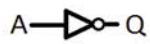
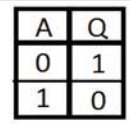
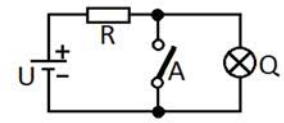
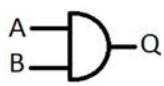
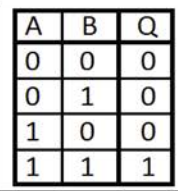
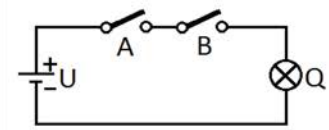
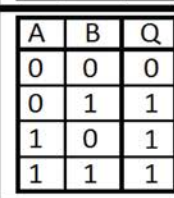
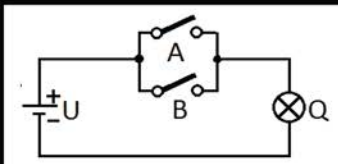
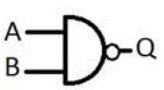
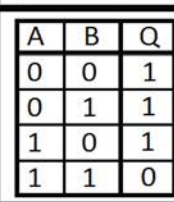
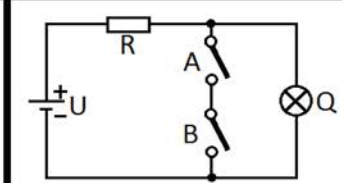
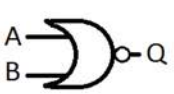
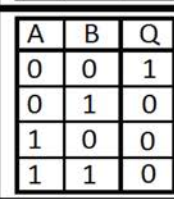
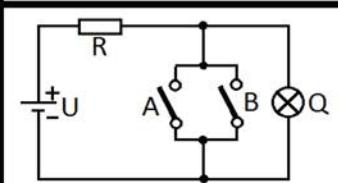
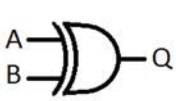
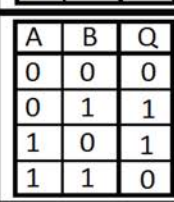
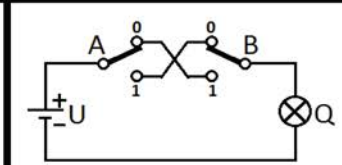
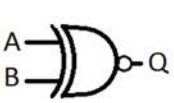
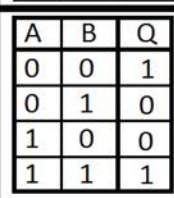
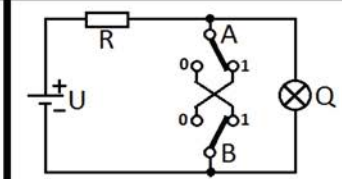
Tablo 1.3'te görüldüğü üzere çeşitli sayıda mantıksal entegre mevcuttur. Bu entegreler içerisinden en çok kullanılan iki entegre türü TTL ve CMOS'tur. Bu isimlendirmeler elektronik elemanın İngilizce baş harflerinden esinlenerek yapılmıştır.

Tablo 1.3: Lojik Entegrelerin Mantıksal Gruplandırılması

Mantık Grubu	Türü	Kullanılan Transistör
CMOS	Tümleyen Metal Oksit Yarı İletkeni	MOSFET
MOS	Metal Oksit Yarı İletkeni	
RDL	Direnç – Diyot – Lojik Tipi	BJT
RTL	Direnç – Transistör – Lojik Tipi	
DTL	Diyot – Transistör – Lojik Tipi	
I2L	Entegre – Enjeksiyonlu Lojik Tipi	
ECL	Emiter Kuplajlı Lojik	
TTL	Transistör – Transistör – Lojik Tipi	
HTL	Yüksek Eşikli Lojik Tipi	

1.2.3. Lojik Kapıların Özellikleri ve Doğruluk Tablosu

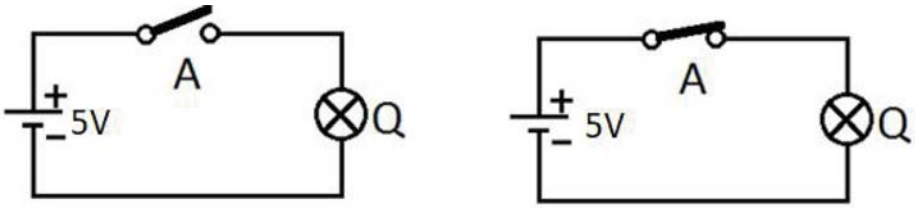
Görsel 1.2'de çeşitli lojik kapıların isimleri, sembolleri, formülleri, doğruluk tablosu ve elektrik şeması verilmiştir.

KAPI İSMİ	SEMBOLÜ	FORMÜLÜ	DOĞRULUĞU	ELEKTRİKİ ŞEMASI
TAMPON (BUFFER) KAPISI		$Q = A$		
DEĞİL (NOT) KAPISI		$Q = \overline{A}$		
VE (AND) KAPISI		$Q = A * B$		
VEYA (OR) KAPISI		$Q = A + B$		
VE DEĞİL (NAND) KAPISI		$Q = \overline{A * B}$		
VEYA DEĞİL (NOR) KAPISI		$Q = \overline{A + B}$		
ÖZEL VEYA (EX-OR) KAPISI		$Q = A \oplus B$ $Q = \overline{A}*B + A*\overline{B}$		
ÖZEL VEYA DEĞİL (EX-NOR) KAPISI		$Q = A \otimes B$ $Q = \overline{A}*\overline{B} + A*B$		

Görsel 1.2: Lojik kapıların isimleri, sembolleri, formülleri, doğruluk tablosu ve elektriksel şeması

1.2.3.1. Mantık Anahtarlama İşlemleri

Lojik kapıların çalışmasını anlayabilmek için aşağıdaki iki devrenin çalışmasını çok iyi kavramak gerekir. Bu iki devrenin çalışması kavrandığında bütün lojik kapıların çalışması anlaşılacaktır.



Görsel 1.3: Basit bir anahtarlama devresi

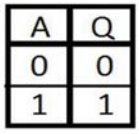
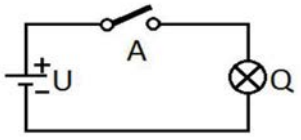
Görsel 1.3'te görülen devrede anahtar açık konumunda olduğundan Q lambası yanmayacaktır. Bu durumda lamba lojik "0" konumundadır yani lamba üzerinde "0" volt vardır. Diğer şekilde ise anahtar kapalı durumundadır, lamba yanacaktır yani lamba lojik "1" konumundadır, lamba üzerinde "5" volt vardır.

Lamba üzerinde okunacak gerilim değeri ya 5 V ya da 0 V'tur. Bunun haricinde hiçbir gerilim değeri yoktur. Başka bir ifade ile ya lojik "1"dir ya da lojik "0"dır.

Lojik kapılarda "Q" harfi her zaman çıkışı göstermektedir. Giriş harfleri ise A, B, C, D,….Y, Z'dir.

1.2.3.2. TAMPON (BUFFER) Kapısı

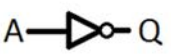
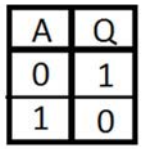
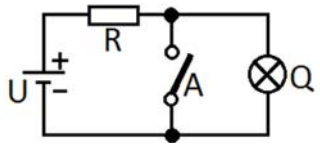
Bu kapının bir girişi ve bir çıkışı vardır. Lojik devreleri veya lojik devre katlarını birbirinden izole etmeye yarar. Girişteki sinyali olduğu gibi çıkışa aktarır (Görsel 1.4).

SEMBOLÜ	FORMÜLÜ	DOĞRULUĞU	ELEKTRİKİ ŞEMASI
	$Q = A$		

Görsel 1.4: Tampon kapısının özellikleri

1.2.3.3. DEĞİL (NOT) Kapısı

Bu kapının bir girişi ve bir çıkışı vardır. Buffer kapısına benzer, buffer kapısının çıkışına bir değil işareti (o) konulması ile elde edilir. Bu işaret (o) her zaman, konulduğu her yerde girişteki işaretin tersini alır yani tersleyen bir kapıdır. Girişten 1 verilirse çıkıştan 0 alınır. Girişten 0 verilirse çıkıştan 1 alınır (Görsel 1.5).

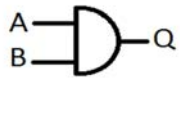
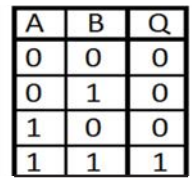
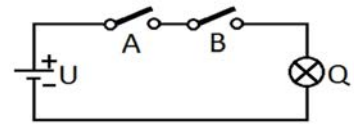
SEMBOLÜ	FORMÜLÜ	DOĞRULUĞU	ELEKTRİKİ ŞEMASI
	$Q = \overline{A}$		

Görsel 1.5: DEĞİL kapısının özellikleri

1.2.3.4. VE (AND) Kapısı

Bir çarpma kapısıdır. Bu kapının en az iki girişi vardır. A ve B gibi girişler iki veya daha fazla sayıda olabilir. Girişlerin tümü lojik 1 seviyesinde ise Q çıkışı lojik 1 seviyesindedir. Girişlerin biri veya birçoğu lojik 0 ise Q çıkışı lojik 0'dır.

Değişkenler bir VE işleminden geçiyorsa aralarına çarpma işareti konularak birbiriyle çarpılır. Q=A.B şeklinde ifade edilir (Görsel 1.6).

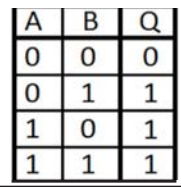
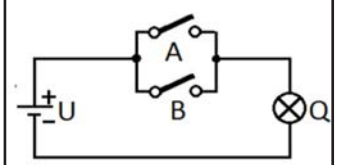
SEMBOLÜ	FORMÜLÜ	DOĞRULUĞU	ELEKTRİKİ ŞEMASI
	Q = A * B		

Görsel 1.6: VE kapısının özellikleri

1.2.3.5. VEYA (OR) Kapısı

Bir toplama kapısıdır. Birden fazla girişi olabilir. Girişlerin tümü lojik 0 seviyesinde ise Q çıkışı lojik 0 seviyesindedir. Girişlerin biri veya birçoğu lojik 1 ise Q çıkışı lojik 1'dir.

Değişkenler bir VEYA işleminden geçiyorsa aralarına toplama işareti konularak birbiriyle çarpılır. Q=A+B şeklinde ifade edilir (Görsel 1.7).

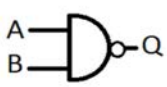
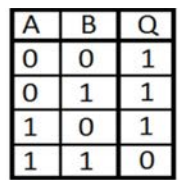
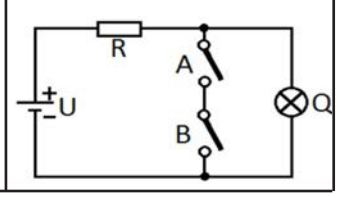
KAPI İSMİ	SEMBOLÜ	FORMÜLÜ	DOĞRULUĞU	ELEKTRİKİ ŞEMASI
VEYA (OR) KAPISI		Q = A + B		

Görsel 1.7: VEYA kapısının özellikleri

1.2.3.6. VE DEĞİL (NAND) Kapısı

VE DEĞİL kapısı VE ile DEĞİL fonksiyonlarının birleşimidir. Her iki kapının birleşmesinden oluşur. Eğer giriş değişkenlerinden biri veya birçoğu 0 ise çıkış lojik 1'dir, değişkenlerin tümü lojik 1 ise çıkış lojik 0 olur.

VE DEĞİL işleminin çıkış fonksiyonları **$Q = \overline{A.B}$** olarak yazılır. Bu ifade "Q eşit A ve B değil" diye okunur (Görsel 1.8).

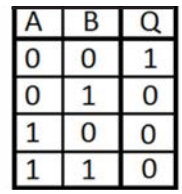
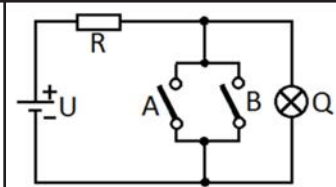
KAPI İSMİ	SEMBOLÜ	FORMÜLÜ	DOĞRULUĞU	ELEKTRİKİ ŞEMASI
VE DEĞİL (NAND) KAPISI		$Q = \overline{A * B}$		

Görsel 1.8: VE DEĞİL kapısının özellikleri

1.2.3.7. VEYA DEĞİL (NOR) Kapısı

VEYA DEĞİL kapısı VEYA ile DEĞİL fonksiyonlarının birleşimidir. Her iki kapının birleşmesinden oluşur. Eğer giriş değişkenlerinin tümü lojik 0 ise çıkış seviyesi lojik 1 olur, değişkenlerin biri veya çoğu lojik 1 ise çıkış lojik 0 olur.

VEYA DEĞİL işleminin çıkış fonksiyonları **$Q = \overline{A+B}$** olarak yazılır. Bu ifade "Q eşit A veya B değil" diye okunur (Görsel 1.9).

KAPI İSMİ	SEMBOLÜ	FORMÜLÜ	DOĞRULUĞU	ELEKTRİKİ ŞEMASI
VEYA DEĞİL (NOR) KAPISI		$Q = \overline{A + B}$		

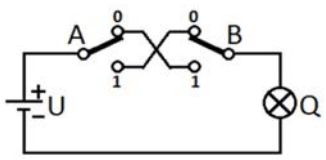
Görsel 1.9: VEYA DEĞİL kapısının özellikleri

1.2.3.8. ÖZEL VEYA (EX-OR) Kapısı

ÖZEL VEYA kapısının sadece 2 girişi ve bir çıkışı vardır. ÖZEL VEYA işlemi kısaca şöyle açıklanabilir. Eğer giriş değişkenlerinin durumu birbirinden farklı ise çıkış lojik 1'dir, giriş değişkenlerinin durumları aynı ise çıkış lojik 0'dır. Çıkış fonksiyonunun yazılışı şöyledir:

$Q = A \oplus B$ **veya** **$Q = \bar{A}B + A\bar{B}$**

İkinci yazım şekline göre ÖZEL VEYA kapısı NOT-AND-OR kapıları ile gerçekleştirilebilir (Görsel 1.10).

KAPI İSMİ	SEMBOLÜ	FORMÜLÜ	DOĞRULUĞU	ELEKTRİKİ ŞEMASI
ÖZEL VEYA (EX-OR) KAPISI		$Q = A \oplus B$ $Q = \bar{A}*B + A*\bar{B}$	A B Q 0 0 0 0 1 1 1 0 1 1 1 0	

Görsel 1.10: ÖZEL VEYA kapısının özellikleri

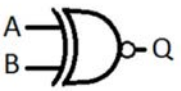
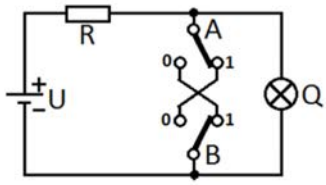
1.2.3.9. ÖZEL VEYA DEĞİL (EX-NOR) Kapısı

Sadece iki girişi, bir çıkışı olan bir EX-NOR kapısında giriş değişkenlerinin durumları aynı ise çıkış lojik 1'dir. Giriş değişkenlerinin durumu farklı ise çıkış lojik 0 olur (Görsel 1.11). Çıkış fonksiyonunun yazılışı şöyledir:

$Q = A \otimes B$

$Q = A \oplus B$

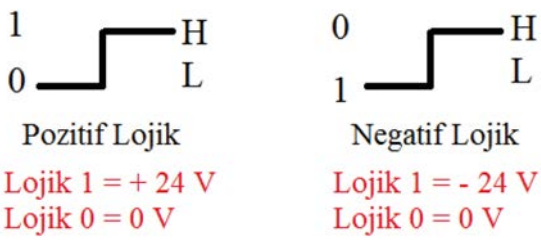
$Q = \overline{\bar{A}.B + A.\bar{B}}$

KAPI İSMİ	SEMBOLÜ	FORMÜLÜ	DOĞRULUĞU	ELEKTRİKİ ŞEMASI
ÖZEL VEYA DEĞİL (EX-NOR) KAPISI		$Q = A \otimes B$ $Q = \bar{A}*\bar{B} + A*B$	A B Q 0 0 1 0 1 0 1 0 0 1 1 1	

Görsel 1.11: ÖZEL VEYA DEĞİL kapısının özellikleri

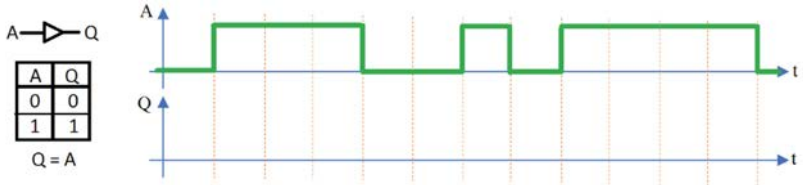
1.2.4. Pozitif ve Negatif Lojik İfadeler

Lojik devrelerde pozitif ve negatif lojik mantığı mevcuttur. Pozitif lojikte, lojik 1 değeri pozitif gerilim değerini ve lojik 0 değeri negatif veya şase gerilim değerini gösterir (Görsel 1.12).

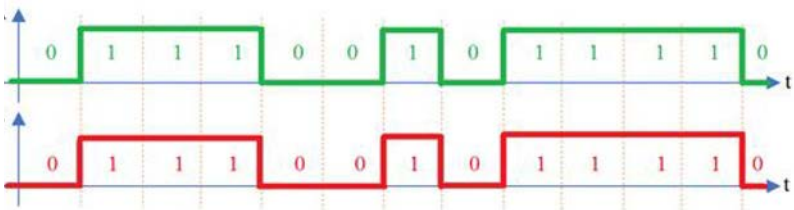


Görsel 1.12: Lojik mantık ifadeleri

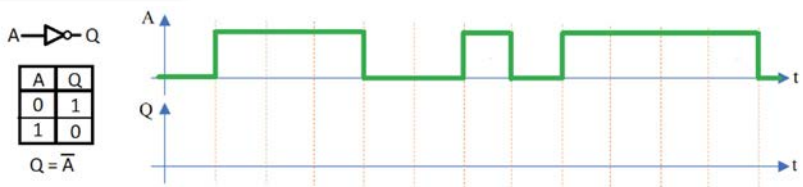
SORU



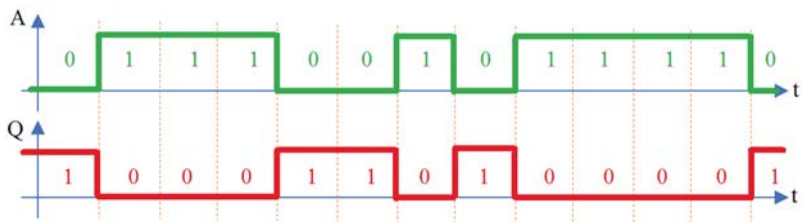
Yukarıda verilen kapı girişine yandaki sinyaller uygulandığında kapı çıkışında oluşacak dalga şeklini ilgili alana çiziniz.



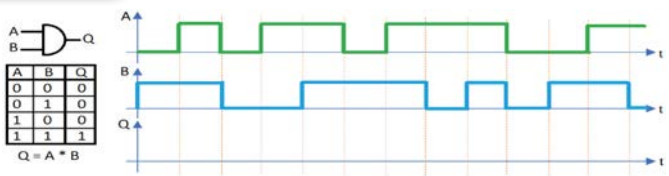
SORU



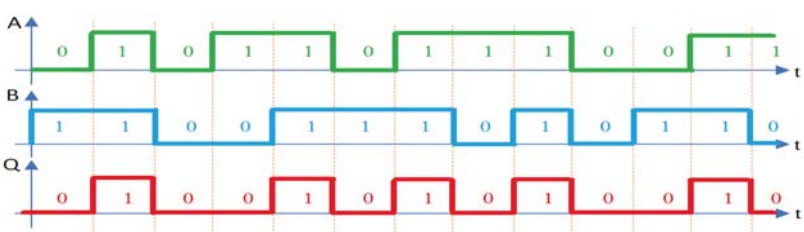
Yukarıda verilen kapı girişine yandaki sinyaller uygulandığında kapı çıkışında oluşacak dalga şeklini ilgili alana çiziniz.



SORU



Yukarıda verilen kapı girişine yandaki sinyaller uygulandığında kapı çıkışında oluşacak dalga şeklini ilgili alana çiziniz.

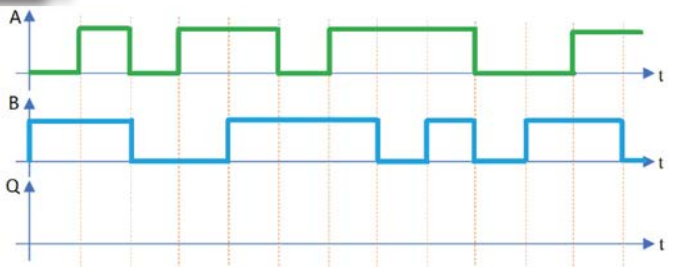


SORU

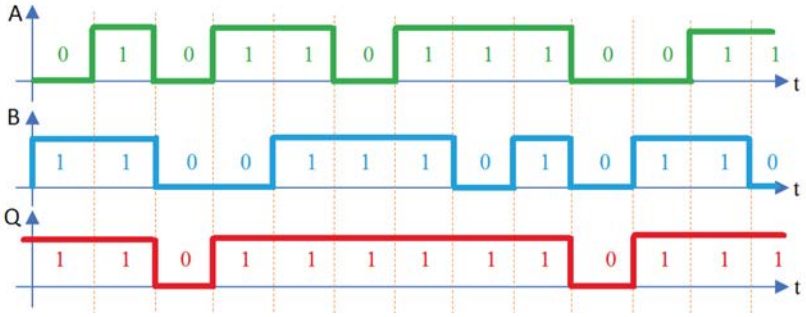


A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

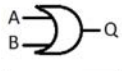
Q = A + B



Yukarıda verilen kapı girişine yandaki sinyaller uygulandığında kapı çıkışında oluşacak dalga şeklini ilgili alana çiziniz.

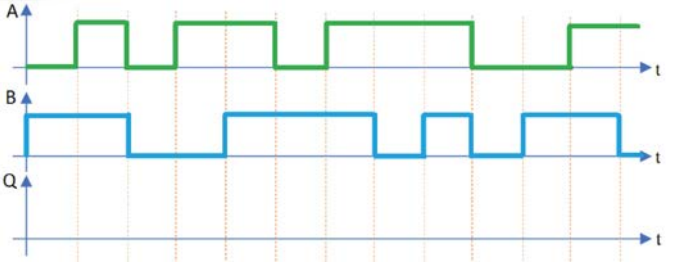


SORU

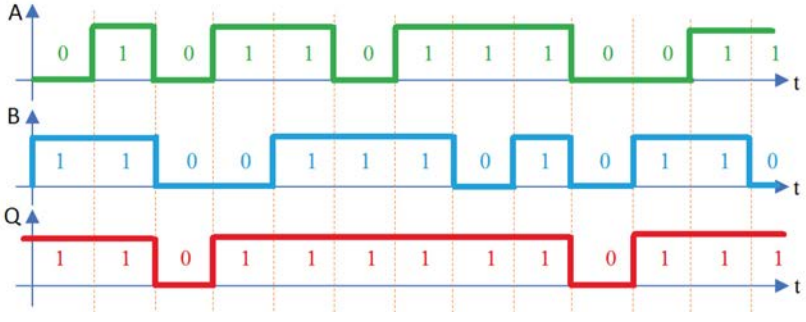


A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

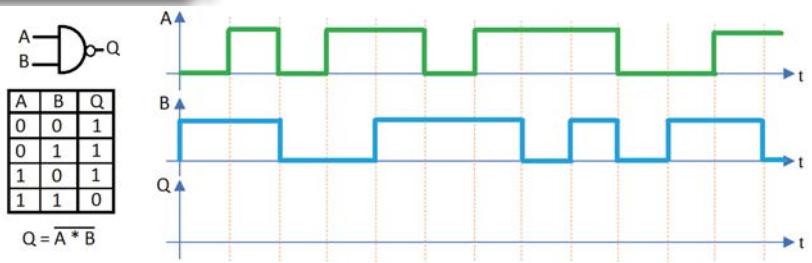
Q = A + B



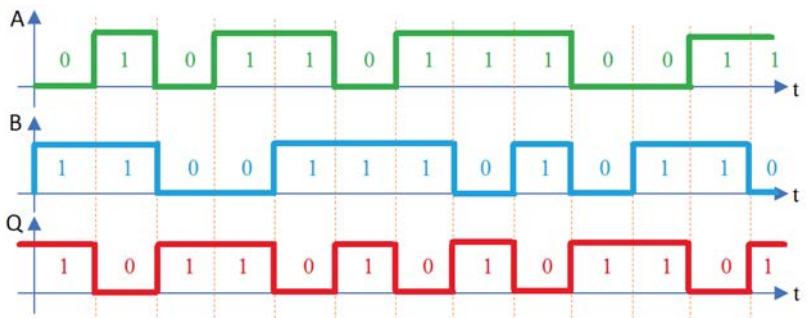
Yukarıda verilen kapı girişine yandaki sinyaller uygulandığında kapı çıkışında oluşacak dalga şeklini ilgili alana çiziniz.



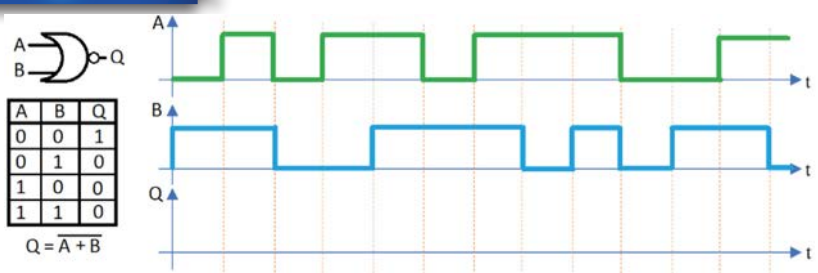
SORU



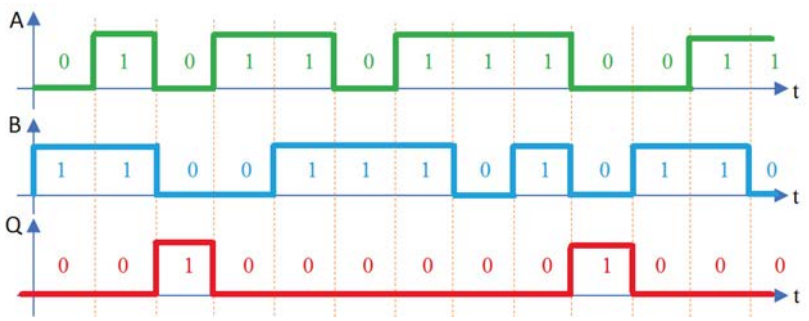
Yukarıda verilen kapı girişine yandaki sinyaller uygulandığında kapı çıkışında oluşacak dalga şeklini ilgili alana çiziniz.



SORU



Yukarıda verilen kapı girişine yandaki sinyaller uygulandığında kapı çıkışında oluşacak dalga şeklini ilgili alana çiziniz.



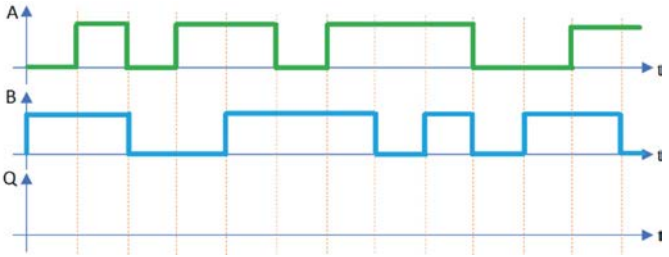
SORU



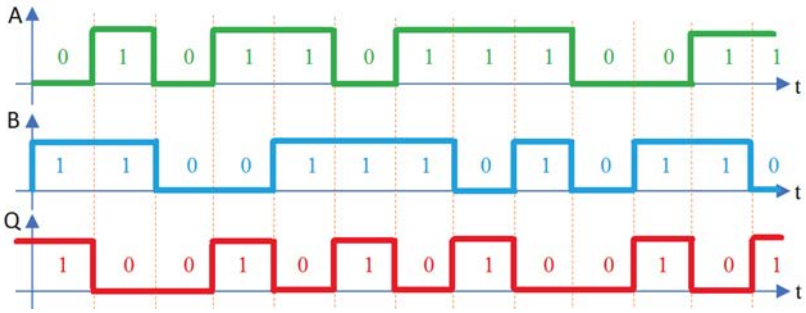
A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

$Q = A \oplus B$

$Q = \overline{A} * B + A * \overline{B}$



Yukarıda verilen kapı girişine yandaki sinyaller uygulandığında kapı çıkışında oluşacak dalga şeklini ilgili alana çiziniz.



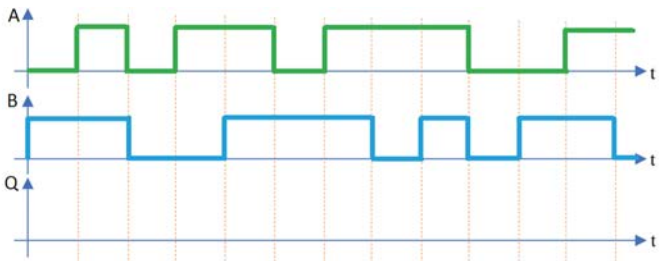
SORU



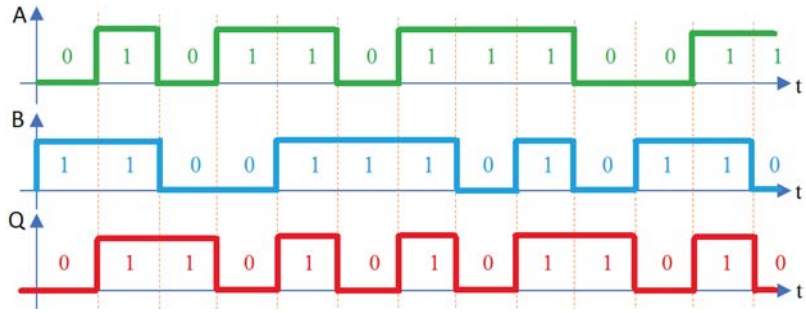
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	1

$Q = A \otimes B$

$Q = \overline{A} * \overline{B} + A * B$



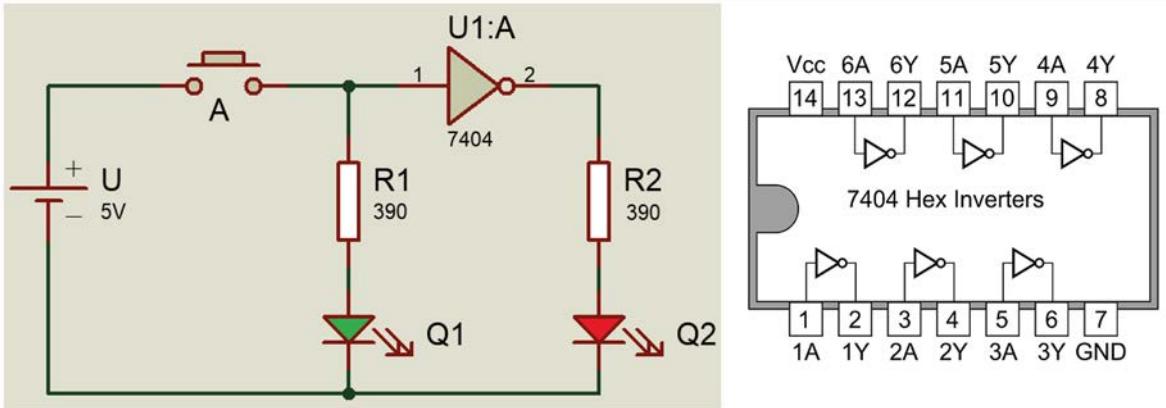
Yukarıda verilen kapı girişine yandaki sinyaller uygulandığında kapı çıkışında oluşacak dalga şeklini ilgili alana çiziniz.



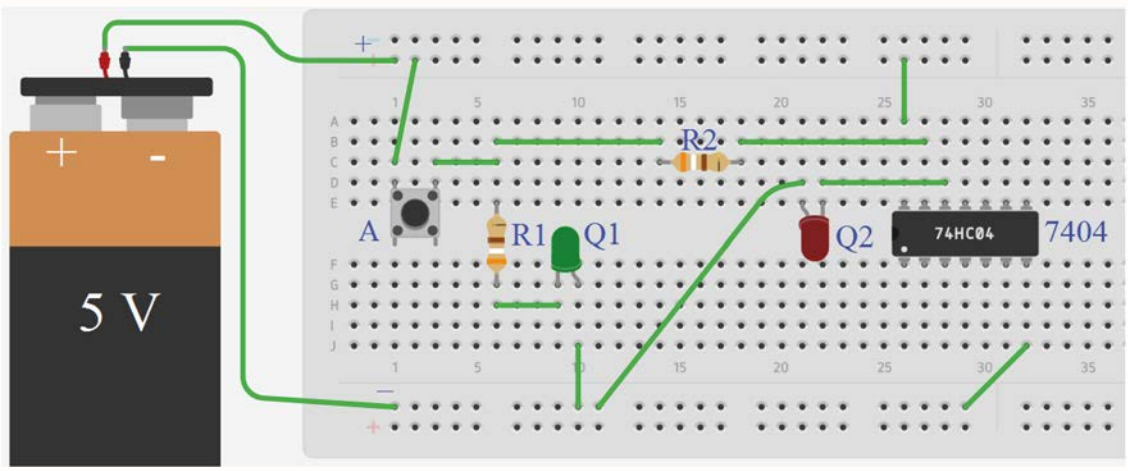
1. UYGULAMA

NOT KAPISI UYGULAMASI

AMAÇ: **NOT kapısı ile devre uygulaması yapmak.**



Görsel 1.13: Devre şeması ve entegrenin içyapısı



Görsel 1.14: Devrenin breadboard bağlantısı



Görsel 1.15: Devrenin breadboard bağlantısı

Tablo 1.4: Sonuç Tablosu

A	Q1	Q2
0		
1		

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Breadboard	-	1 adet
7404 entegresi	-	1 adet
Direnç	390 Ω	2 adet
LED diyot	Kırmızı ve yeşil renkli	2 adet
Buton	-	1 adet
Zil teli	0,5 mm^2 kablo	2-3 m
DC güç kaynağı	+5 V DC	1 adet
El aletleri	Kargaburnu, yan keski, tornavida, izole bant vb.	-

İŞLEM BASAMAKLARI

1. Görsel 1.13'teki devreyi breadboard üzerine kurunuz (Görsel 1.14).
2. Öğretmeninizin kontrolünde devreye enerji vererek devreyi çalıştırınız.
3. Devreyi Tablo 1.4'teki sıraya uygun şekilde çalıştırınız ve her bir değeri tabloya yazınız.
4. Tablodaki değerlere dikkat ederek Görsel 1.15'teki çıkış alanını oluşturunuz.
5. Enerjiyi keserek devreyi sökünüz.
6. Sonuçları yorumlayınız.

SORULAR

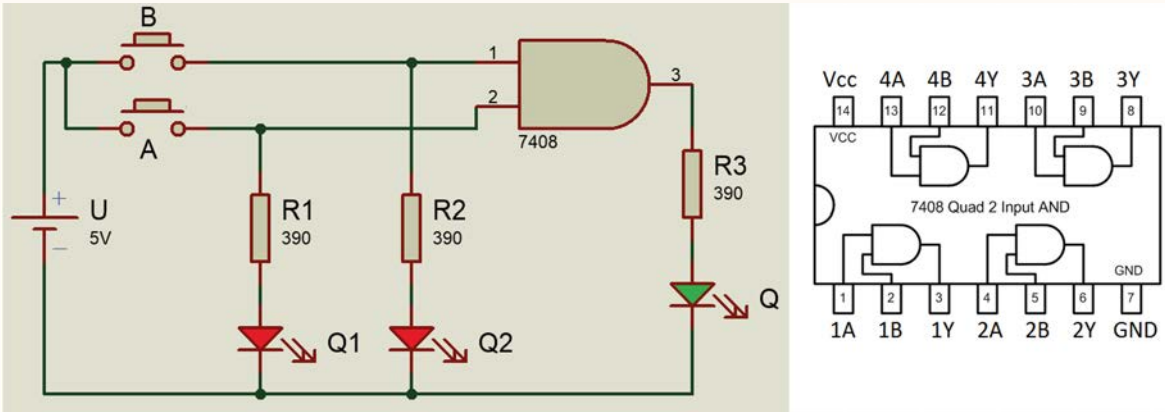
1. A anahtarı açık konumda iken hangi LED, ışık verir? Yazınız.
2. A anahtarı kapalı konumda iken hangi çıkışlardaki LED'ler ışık verir? Nedeniyle yazınız. Q2 çıkışındaki LED hangi durumlarda ışık verir? Yazınız.
3. Görsel 1.15'teki A sinyaline göre sırası ile butona basarak Q1 ve Q2 çıkış sinyallerini çizerek görsel 1.15'i tamamlayınız.
4. Güç kaynağının uçlarını değiştirerek devreye bağlayınız ve sonuçları nedenleri ile yazınız.
5. Entegre üzerindeki nokta işaretinin ne işe yaradığını belirtiniz.
6. Entegre üzerindeki VCC ve GND uçlarını bulunuz, ayak numaralarını yazınız.
7. VCC ifadesini açıklayınız.
8. GND ifadesini açıklayınız.

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Entegrenin içyapısını inceledi.	20	
Sınıfı :	2	Devreyi breadboard üzerine kurdu.	20	
Numarası :	3	Tablodaki sıraya uygun olarak devreyi çalıştırdı.	20	
ÖĞRETMEN	4	Çalışmayı tablodaki alana kaydetti.	20	
Adı Soyadı :	5	Grafiği veriler ışığında çizdi.	10	
	6	Malzemeleri teslim etti.	10	
İmza :		**TOPLAM PUAN**	100	

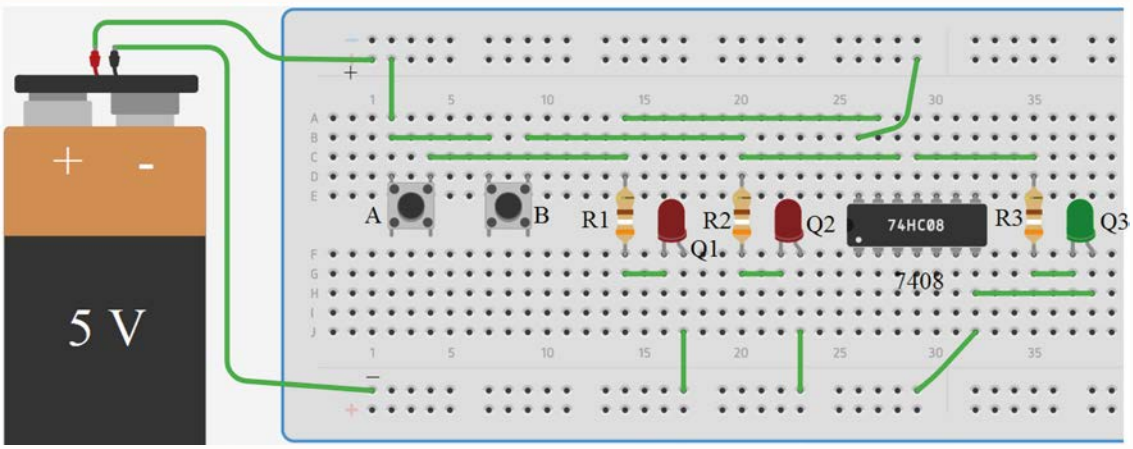
2. UYGULAMA

AND KAPISI UYGULAMASI

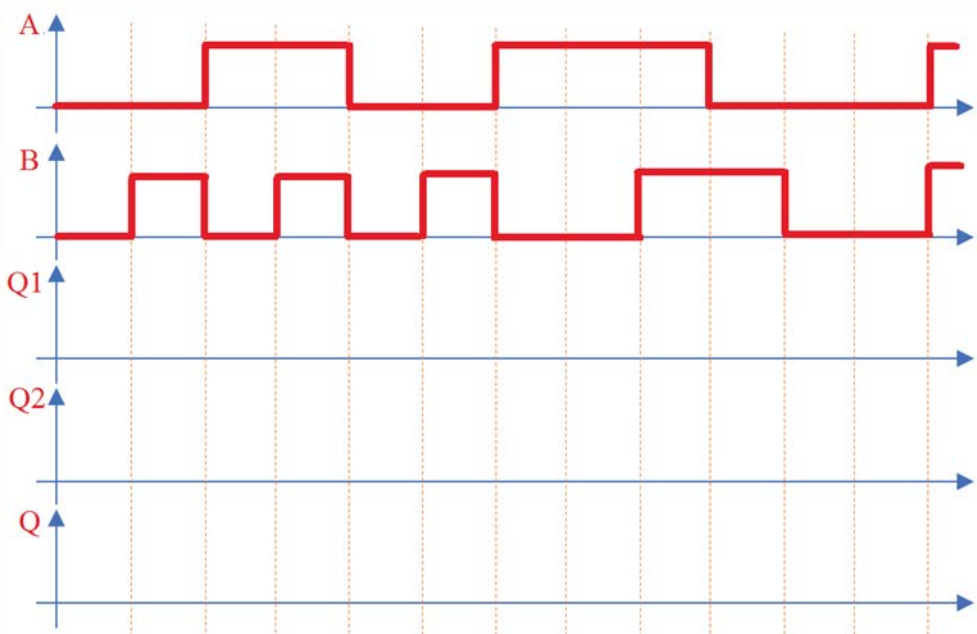
AMAÇ: **AND kapısı ile devre uygulaması yapmak.**



Görsel 1.16: Devre şeması ve entegrenin içyapısı



Görsel 1.17: Devrenin breadboard bağlantısı



Görsel 1.18: Grafik şeması

Tablo 1.5: Sonuç Tablosu

A	B	Q1	Q2	Q
0	0			
0	1			
1	0			
1	1			

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Breadboard	-	1 adet
7408 entegresi	-	1 adet
Direnç	390 Ω	3 adet
LED diyot	Kırmızı ve yeşil renkli	3 adet
Buton	-	2 adet
Zil teli	0,5 mm^2 kablo	2-3 m
DC güç kaynağı	+5 V DC	1 adet
El Aletleri	Kargaburnu, yan keski, tornavida, izole bant vb.	-

İŞLEM BASAMAKLARI

1. Görsel 1.16'daki devreyi breadboard üzerine kurunuz (Görsel 1.17).
2. Öğretmeninizin kontrolünde devreye enerji vererek devreyi çalıştırınız.
3. Devreyi Tablo 1.5'teki sıraya uygun şekilde çalıştırınız ve her bir değeri tabloya yazınız.
4. Tablodaki değerlere dikkat ederek Görsel 18'deki çıkış alanını oluşturunuz.
5. Enerjiyi keserek devreyi sökünüz.
6. Sonuçları yorumlayınız.

SORULAR

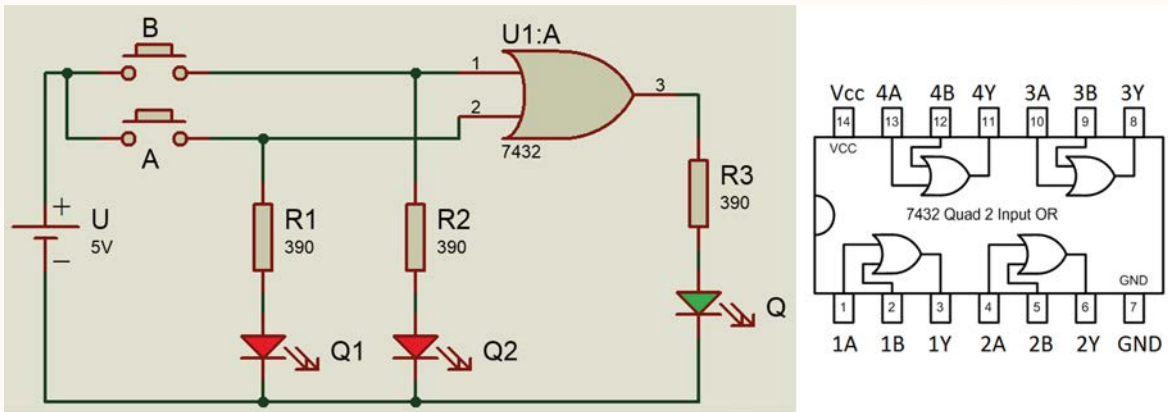
1. Q1, Q2 çıkışlarının aktif olması için A ve B girişi ne olmalıdır? Yazınız.
2. A ve B anahtarı kapalı konumda olduğunda ne olur? Açıklayınız.
3. Görsel 1.18'deki A ve B sinyaline göre sırası ile butona basarak, Q1 ve Q2 giriş sinyallerini çizip Görsel 1.18'i tamamlayınız.
4. Q çıkışının sürekli aktif olması için ne yapılmalıdır? Açıklayınız.
5. Devreye ikinci bir VE kapısı eklenmek istendiğinde 7408 entegresinin hangi ayakları kullanılmalıdır? Yazınız.
6. VCC ve GND uçları ters bağlanırsa ne gibi sonuçlar ortaya çıkar? Açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Entegrenin içyapısını inceledi.	20	
Sınıfı :	2	Devreyi breadboard üzerine kurdu.	20	
Numarası :	3	Tablodaki sıraya uygun olarak devreyi çalıştırdı.	20	
ÖĞRETMEN	4	Çalışmayı tablodaki alana kaydetti.	20	
Adı Soyadı :	5	Grafiği veriler ışığında çizdi.	10	
	6	Malzemeleri teslim etti.	10	
İmza :		**TOPLAM PUAN**	100	

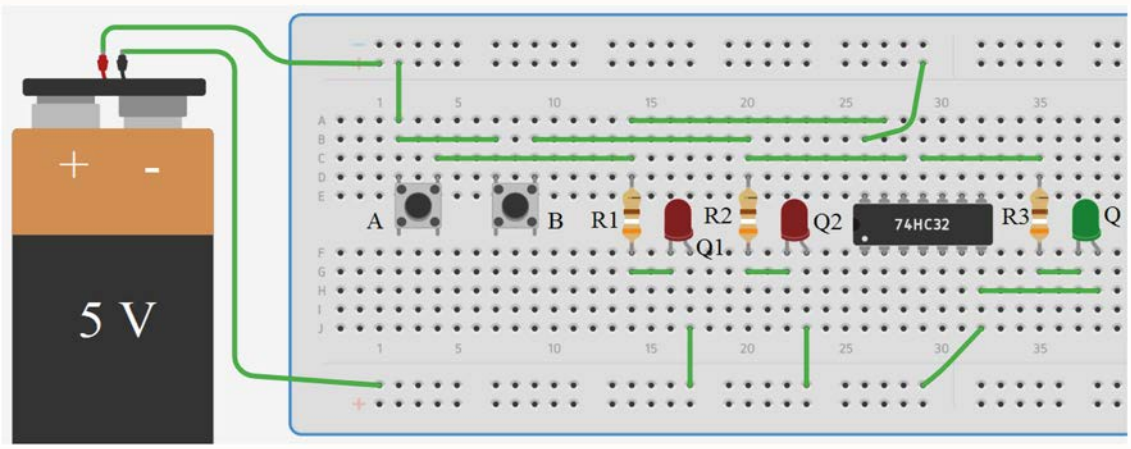
3. UYGULAMA

OR KAPISI UYGULAMASI

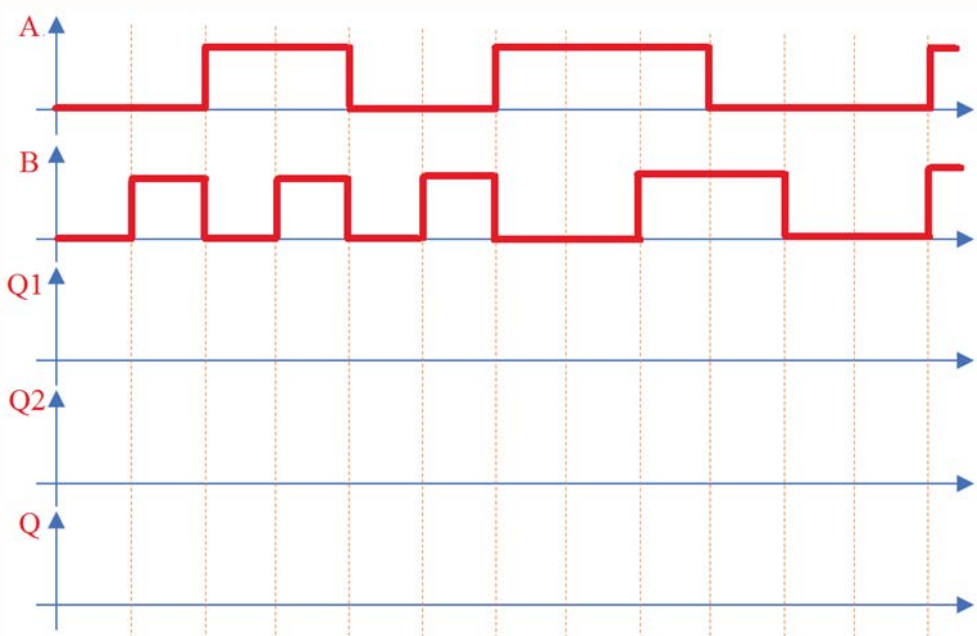
AMAÇ: **OR kapısı ile devre uygulaması yapmak.**



Görsel 1.19: Devre şeması ve entegrenin içyapısı



Görsel 1.20: Devrenin breadboard bağlantısı



Görsel 1.21: Grafik şeması

Tablo 1.6: Sonuç Tablosu

A	B	Q1	Q2	Q
0	0			
0	1			
1	0			
1	1			

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Breadboard	-	1 adet
7432 entegresi	-	1 adet
Direnç	390 Ω	3 adet
LED diyot	Kırmızı ve yeşil renkli	3 adet
Buton	-	2 adet
Zil teli	0,5 mm^2 kablo	2-3 m
DC güç kaynağı	+5 V DC	1 adet
El aletleri	Kargaburnu, yan keski, tornavida, izole bant vb.	-

İŞLEM BASAMAKLARI

1. Görsel 1.19'daki devreyi breadboard üzerine kurunuz (Görsel 1.19).
2. Öğretmeninizin kontrolünde devreye enerji vererek devreyi çalıştırınız.
3. Devreyi Tablo 1.6'daki sıraya uygun şekilde çalıştırınız ve her bir değeri tabloya yazınız.
4. Tablodaki değerlere dikkat ederek Görsel 1.21'deki çıkış alanını oluşturunuz.
5. Enerjiyi keserek devreyi sökünüz.
6. Sonuçları yorumlayınız.

SORULAR

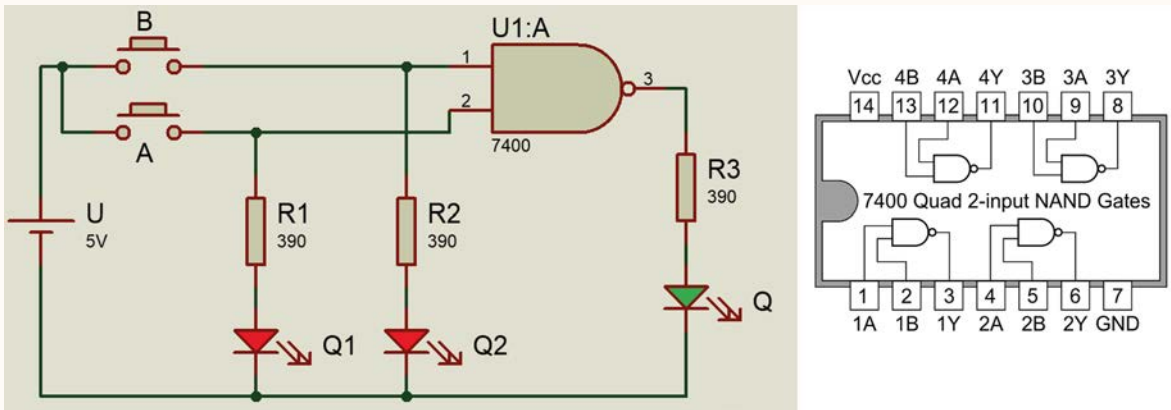
1. Q1, Q2 çıkışlarının aktif olması için A ve B girişi ne olmalıdır? Yazınız.
2. A ve B anahtarı kapalı iken Q1 ve Q2'nin durumu nedir? Açıklayınız.
3. Görsel 1.21'deki A ve B sinyaline göre sırası ile butona basarak, Q1 ve Q2 giriş sinyallerini çizip Görsel 1.21'i tamamlayınız.
4. Böyle bir devreyi günlük hayatta nerede kullanırdınız? Belirtiniz.
5. 7432 entegresine ait olan 7 numaralı ayağın adını ve devredeki bağlantı yerini yazınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Entegrenin içyapısını inceledi.	20	
Sınıfı :	2	Devreyi breadboard üzerine kurdu.	20	
Numarası :	3	Tablodaki sıraya uygun olarak devreyi çalıştırdı.	20	
ÖĞRETMEN	4	Çalışmayı tablodaki alana kaydetti.	20	
Adı Soyadı :	5	Grafiği veriler ışığında çizdi.	10	
	6	Malzemeleri teslim etti.	10	
İmza :		**TOPLAM PUAN**	100	

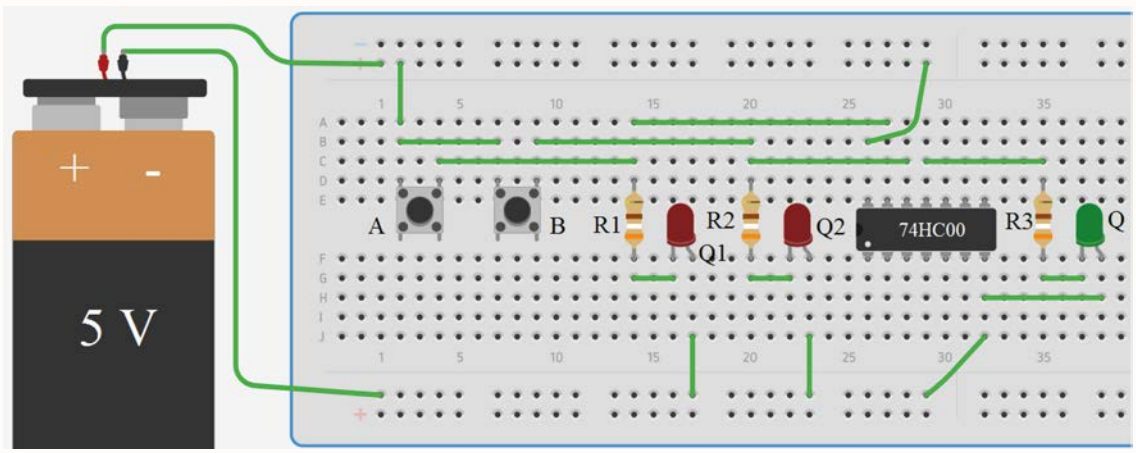
4. UYGULAMA

NAND KAPISI UYGULAMASI

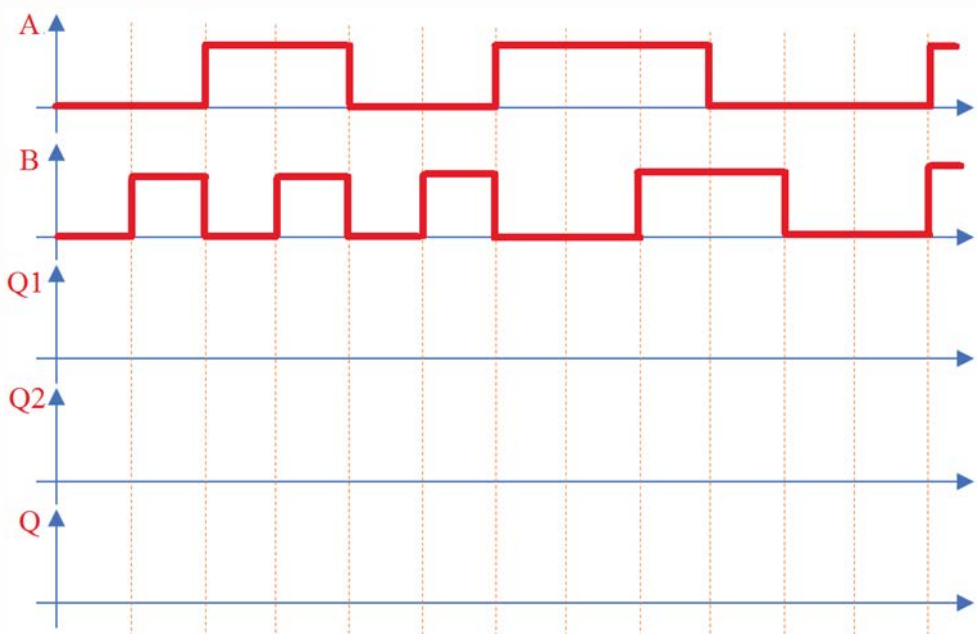
AMAÇ: **NAND kapısı ile devre uygulaması yapmak.**



Görsel 1.22: Devre şeması ve entegrenin içyapısı



Görsel 1.23: Devrenin breadboard bağlantısı



Görsel 1.24: Grafik şeması

Tablo 1.7: Sonuç Tablosu

A	B	Q1	Q2	Q
0	0			
0	1			
1	0			
1	1			

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Breadboard	-	1 adet
7400 entegresi	-	1 adet
Direnç	390 Ω	3 adet
LED diyot	Kırmızı ve yeşil renkli	3 adet
Buton	-	2 adet
Zil teli	0,5 mm^2 kablo	2-3 m
DC güç kaynağı	+5 V DC	1 adet
El aletleri	Kargaburnu, yan keski, tornavida, izole bant vb.	-

İŞLEM BASAMAKLARI

1. Görsel 1.22'deki devreyi breadboard üzerine kurunuz (Görsel 1.23).
2. Öğretmeninizin kontrolünde devreye enerji vererek devreyi çalıştırınız.
3. Devreyi Tablo 1.7'deki sıraya uygun şekilde çalıştırınız ve her bir değeri tabloya yazınız.
4. Tablodaki değerlere dikkat ederek Görsel 1.24'teki çıkış alanını oluşturunuz.
5. Enerjiyi keserek devreyi sökünüz.
6. Sonuçları yorumlayınız.

SORULAR

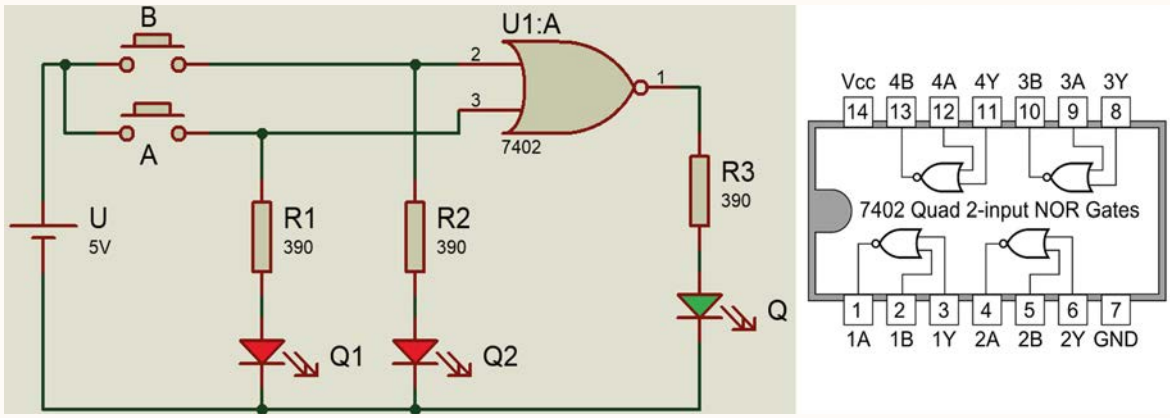
1. A ve B, iki kapılı spor bir arabanın kapılarında bulunan anahtarlar olsun, Q ise arabanın tavan lambası olsun. Buna göre arabanın her iki kapısı açık olduğunda tavan lambası yanar mı? Açıklayınız.
2. A ve B anahtarı açık iken Q1 ve Q2'nin durumu nedir? Açıklayınız.
3. Görsel 1.24'teki A ve B sinyaline göre sırası ile butona basarak, Q1 ve Q2 giriş sinyallerini çizip Görsel 1.24'ü tamamlayınız.
4. Böyle bir devreyi günlük hayatta nerede kullanırdınız? Belirtiniz.
5. 7400 entegresinde VCC ile ifade edilen ayağın görevini ve pilin hangi ucuna bağlanması gerektiğini yazınız.

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Entegrenin içyapısını inceledi.	20	
Sınıfı :	2	Devreyi breadboard üzerine kurdu.	20	
Numarası :	3	Tablodaki sıraya uygun olarak devreyi çalıştırdı.	20	
ÖĞRETMEN	4	Çalışmayı tablodaki alana kaydetti.	20	
Adı Soyadı :	5	Grafiği veriler ışığında çizdi.	10	
	6	Malzemeleri teslim etti.	10	
İmza :		**TOPLAM PUAN**	100	

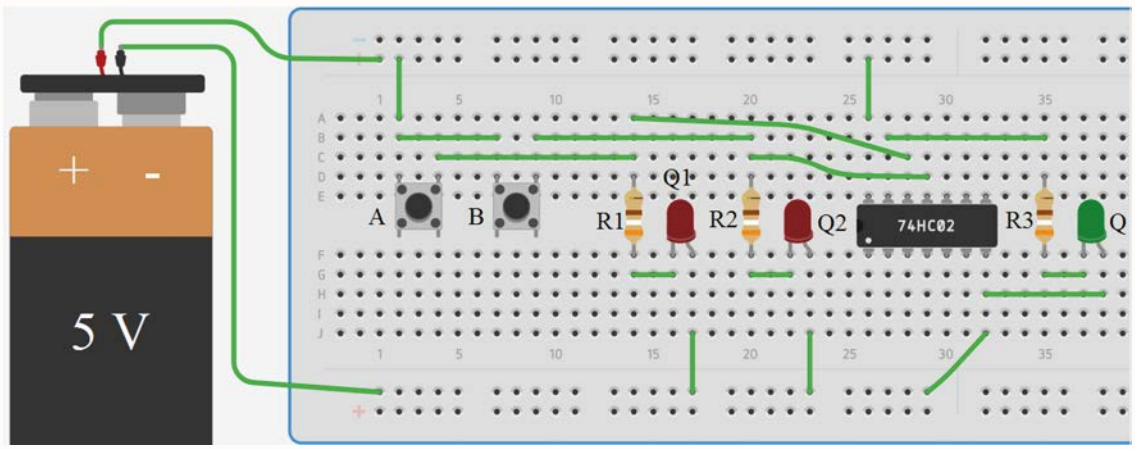
5. UYGULAMA

NOR KAPISI UYGULAMASI

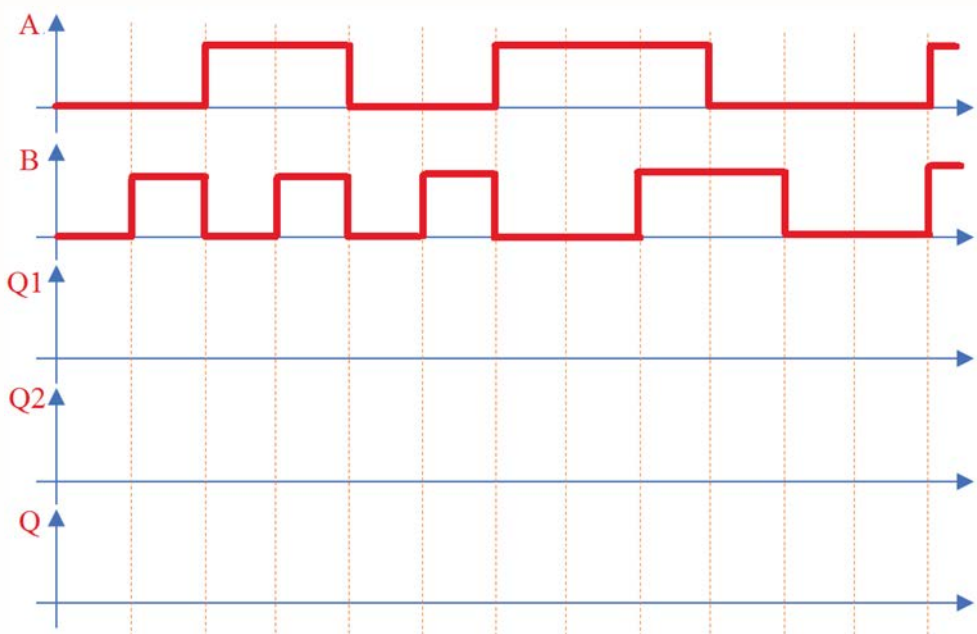
AMAÇ: **NOR kapısı ile devre uygulaması yapmak.**



Görsel 1.25: Devre şeması ve entegrenin içyapısı



Görsel 1.26: Devrenin breadboard bağlantısı



Görsel 1.27: Grafik şeması

Tablo 1.8: Sonuç Tablosu

A	B	Q1	Q2	Q
0	0			
0	1			
1	0			
1	1			

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Breadboard	-	1 adet
7402 entegresi	-	1 adet
Direnç	390 Ω	3 adet
LED diyot	Kırmızı ve yeşil renkli	3 adet
Buton	-	2 adet
Zil teli	0,5 mm^2 kablo	2-3 m
DC güç kaynağı	+5 V DC	1 adet
El aletleri	Kargaburnu, yan keski, tornavida, izole bant vb.	-

İŞLEM BASAMAKLARI

1. Görsel 1.25'teki devreyi breadboard üzerine kurunuz (Görsel 1.26).
2. Öğretmeninizin kontrolünde devreye enerji vererek devreyi çalıştırınız.
3. Devreyi Tablo 1.8'deki sıraya uygun şekilde çalıştırınız ve her bir değeri tabloya yazınız.
4. Tablodaki değerlere dikkat ederek Görsel 1.27'deki çıkış alanını oluşturunuz.
5. Enerjiyi keserek devreyi sökünüz.
6. Sonuçları yorumlayınız.

SORULAR

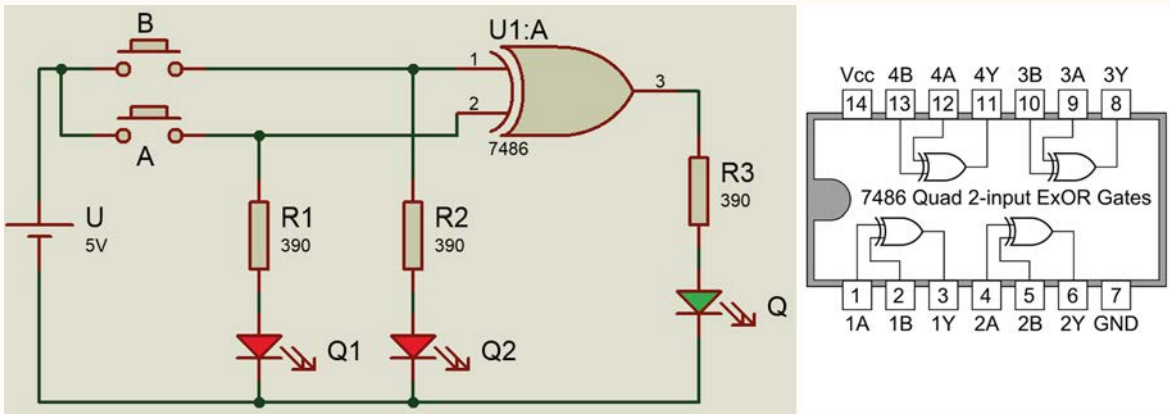
1. A ve B anahtarının her ikisi de kapalı olduğunda çıkış ne olur? Açıklayınız.
2. Görsel 1.25'teki kaynak ters çevrilip bağlandığında devrenin çalışmasında nasıl bir değişiklik olur? Deneyerek açıklayınız.
3. Görsel 1.27'deki A ve B sinyaline göre sırası ile butona basarak, Q1 ve Q2 giriş sinyallerini çizip Görsel 1.27'yi tamamlayınız.
4. Böyle bir devreyi günlük hayatta nerede kullanırdınız? Açıklayınız.
5. 7402 entegresi bulunamaması hâlinde bu devre başka hangi entegrelerle kurulabilir? Yazınız.
6. Entegrenin 7 numaralı ayağına giden kablo bağlantısını söküp, devreye tekrar enerji vererek sonuçlarını yazınız.

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Entegrenin içyapısını inceledi.	20	
Sınıfı :	2	Devreyi breadboard üzerine kurdu.	20	
Numarası :	3	Tablodaki sıraya uygun olarak devreyi çalıştırdı.	20	
ÖĞRETMEN	4	Çalışmayı tablodaki alana kaydetti.	20	
Adı Soyadı :	5	Grafiği veriler ışığında çizdi.	10	
	6	Malzemeleri teslim etti.	10	
İmza :		**TOPLAM PUAN**	100	

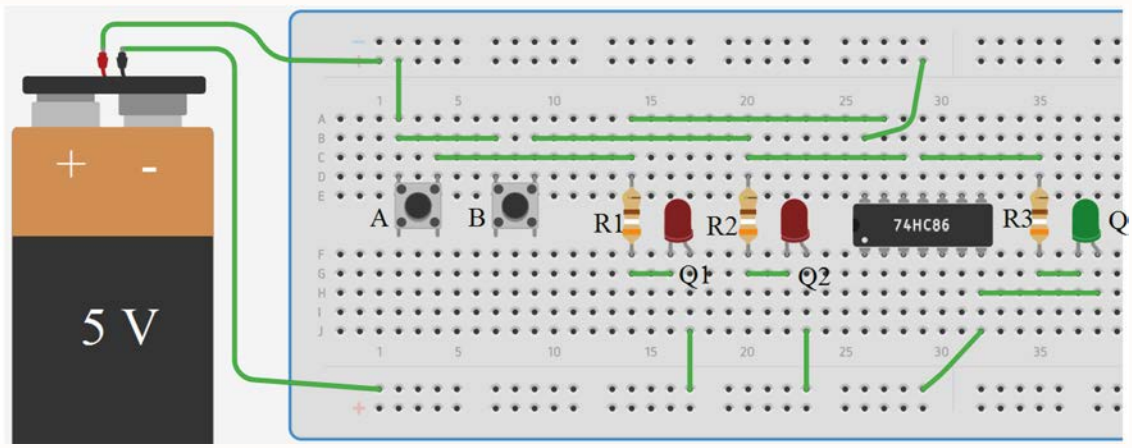
6. UYGULAMA

EX-OR KAPISI UYGULAMASI

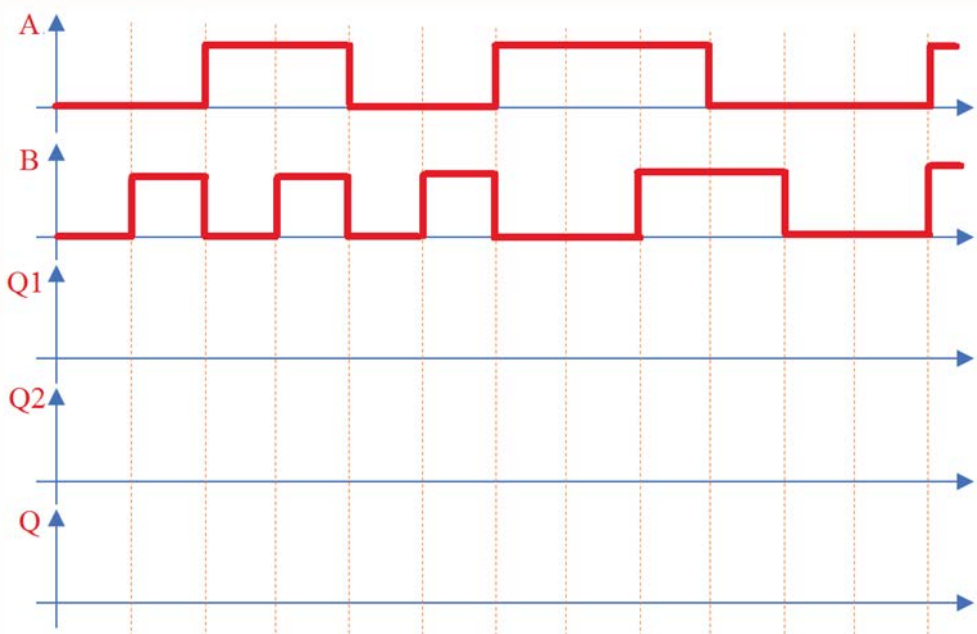
AMAÇ: **EX-OR kapısı ile devre uygulaması yapmak.**



Görsel 1.28: Devre şeması ve entegrenin içyapısı



Görsel 1.29: Devrenin breadboard bağlantısı



Görsel 1.30: Grafik şeması

Tablo 1.9: Sonuç Tablosu

A	B	Q1	Q2	Q
0	0			
0	1			
1	0			
1	1			

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Breadboard	-	1 adet
7486 entegresi	-	1 adet
LED diyot	Kırmızı ve yeşil renkli	3 adet
Direnç	390 Ω	3 adet
Buton	-	2 adet
Zil teli	0,5 mm^2 kablo	2-3 m
DC güç kaynağı	+5 V DC	1 adet
El aletleri	Kargaburnu, yan keski, tornavida, izole bant vb.	-

İŞLEM BASAMAKLARI

1. Görsel 1.29'daki devreyi breadboard üzerine kurunuz (Görsel 1.30).
2. Öğretmeninizin kontrolünde devreye enerji vererek devreyi çalıştırınız.
3. Devreyi Tablo 1.9'daki sıraya uygun şekilde çalıştırınız ve her bir değeri tabloya yazınız.
4. Tablodaki değerlere dikkat ederek Görsel 1.30'daki çıkış alanını oluşturunuz.
5. Enerjiyi keserek devreyi sökünüz.
6. Sonuçları yorumlayınız.

SORULAR

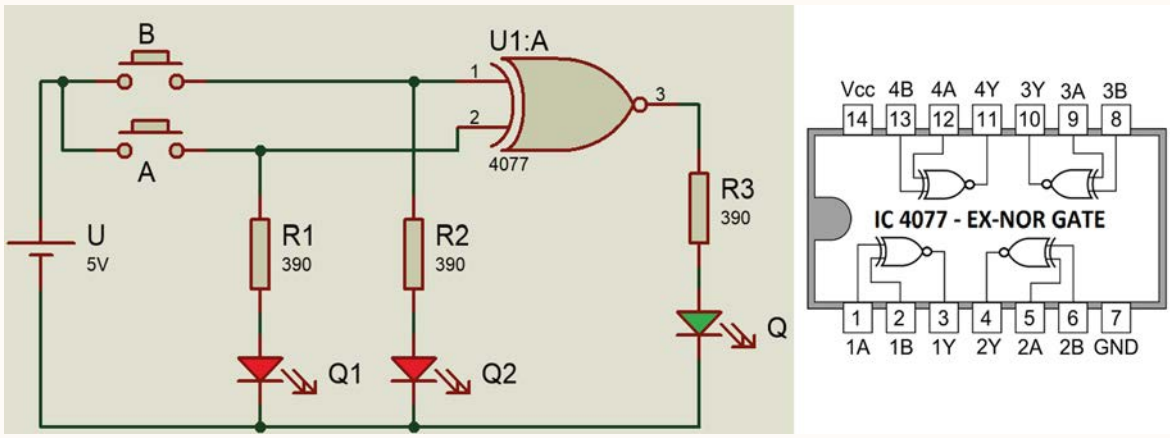
1. Aynı sonucu almak için EX-OR kapısı yerine ne kullanabilirsiniz? Belirtiniz.
2. EX-OR kapısın yerine kullandığınız yeni kapıya göre devre şemasını çiziniz.
3. Beş adet EX-OR kapılı bir devre kurmak için kullanılacak entegre adını ve adedini yazınız.

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Entegrenin içyapısını inceledi.	20	
Sınıfı :	2	Devreyi breadboard üzerine kurdu.	20	
Numarası :	3	Tablodaki sıraya uygun olarak devreyi çalıştırdı.	20	
ÖĞRETMEN	4	Çalışmayı tablodaki alana kaydetti.	20	
Adı Soyadı :	5	Grafiği veriler ışığında çizdi.	10	
	6	Malzemeleri teslim etti.	10	
İmza :		**TOPLAM PUAN**	100	

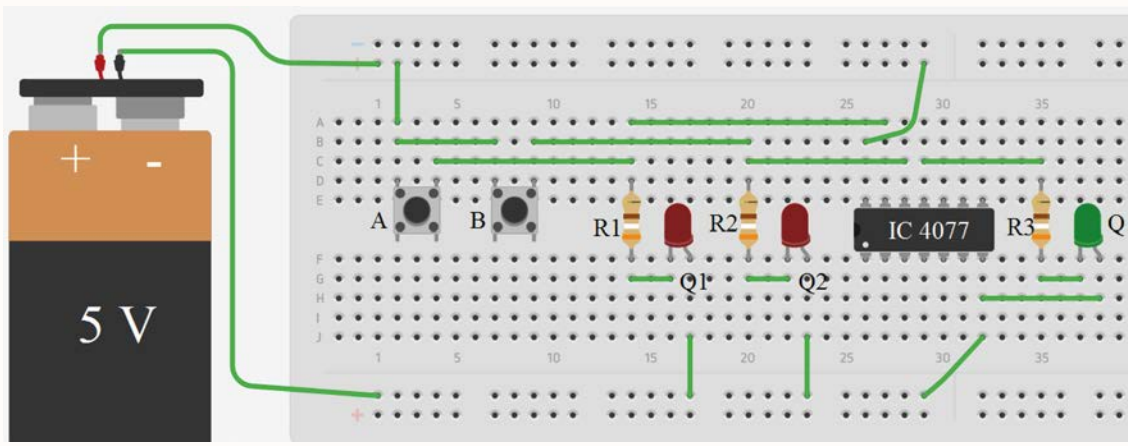
7. UYGULAMA

EX-NOR KAPISI UYGULAMASI

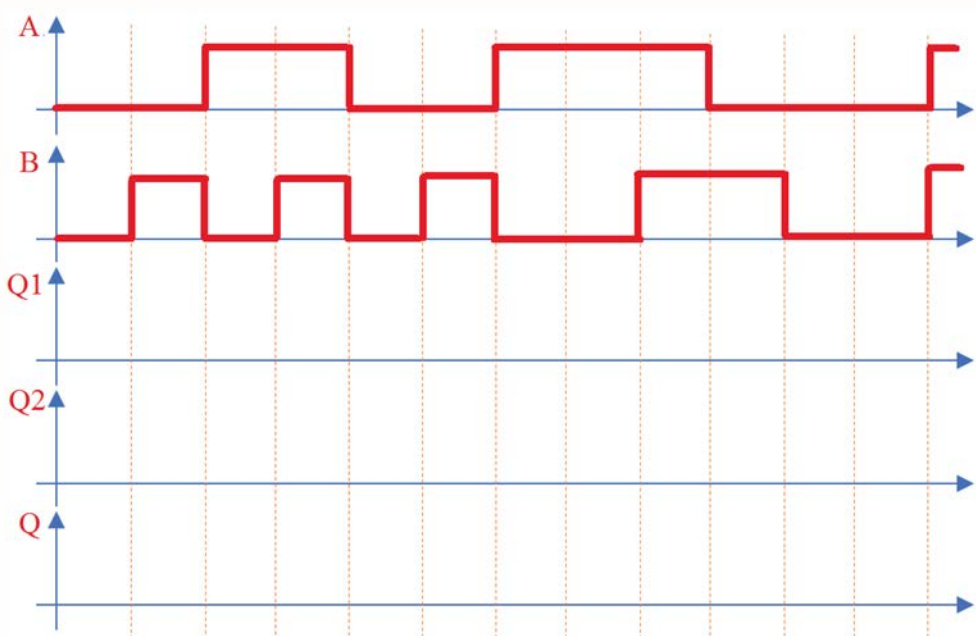
AMAÇ: **EX-NOR kapısı ile devre uygulaması yapmak.**



Görsel 1.31: Devre şeması ve entegrenin içyapısı



Görsel 1.32: Devrenin breadboard bağlantısı



Görsel 1.33: Grafik şeması

Tablo 1.10: Sonuç Tablosu

A	B	Q1	Q2	Q
0	0			
0	1			
1	0			
1	1			

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Breadboard	-	1 adet
IC4077 entegresi	-	1 adet
LED diyot	Kırmızı ve yeşil renkli	3 adet
Direnç	390 Ω	3 adet
Buton	-	2 adet
Zil teli	0,5 mm^2 kablo	2-3 m
DC güç kaynağı	+5 V DC	1 adet
El aletleri	Kargaburnu, yan keski, tornavida, izole bant vb.	-

İŞLEM BASAMAKLARI

1. Görsel 1.31'deki devreyi breadboard üzerine kurunuz (Görsel 1.32).
2. Öğretmeninizin kontrolünde devreye enerji vererek devreyi çalıştırınız.
3. Devreyi Tablo 1.10'daki sıraya uygun şekilde çalıştırınız ve her bir değeri tabloya yazınız.
4. Tablodaki değerlere dikkat ederek Görsel 1.33'teki çıkış alanını oluşturunuz.
5. Enerjiyi keserek devreyi sökünüz.
6. Sonuçları yorumlayınız.

SORULAR

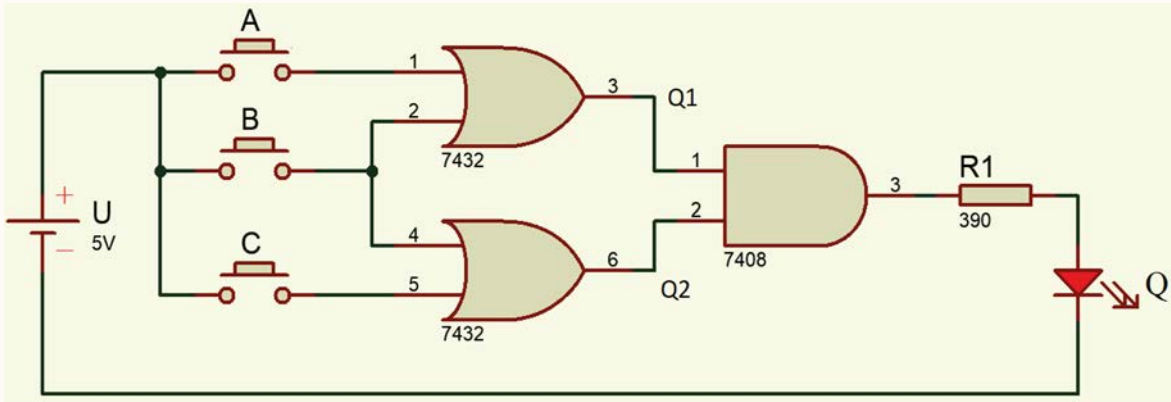
1. EX-NOR kapısının doğruluk tablosunu oluşturunuz?
2. EX-NOR kapısı yerine, aynı sonucu almak için hangi kapıları kullanırsınız? Açıklayınız.
3. Böyle bir devreyi günlük hayatta nerede kullanırdınız? Açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Entegrenin içyapısını inceledi.	20	
Sınıfı :	2	Devreyi breadboard üzerine kurdu.	20	
Numarası :	3	Tablodaki sıraya uygun olarak devreyi çalıştırdı.	20	
ÖĞRETMEN	4	Çalışmayı tablodaki alana kaydetti.	20	
Adı Soyadı :	5	Grafiği veriler ışığında çizdi.	10	
	6	Malzemeleri teslim etti.	10	
İmza :		**TOPLAM PUAN**	100	

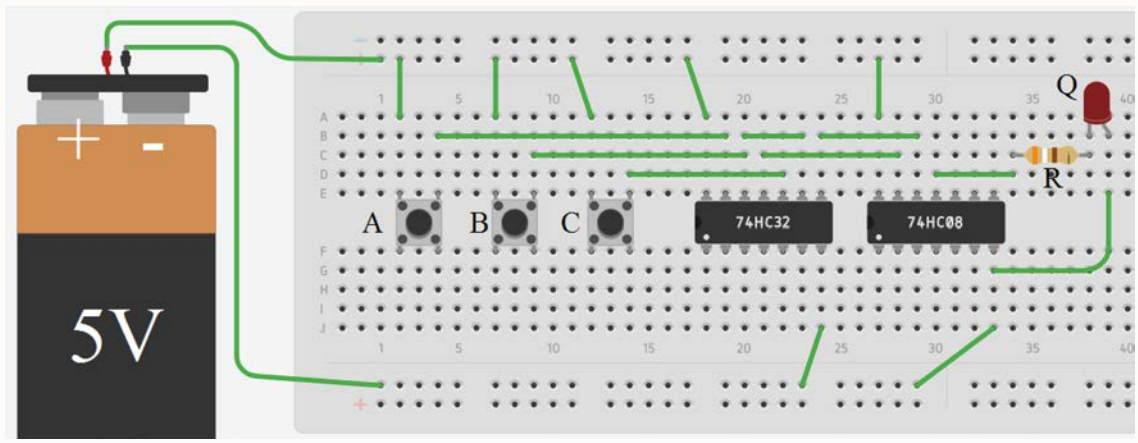
8. UYGULAMA

LOJİK KAPI UYGULAMASI 1

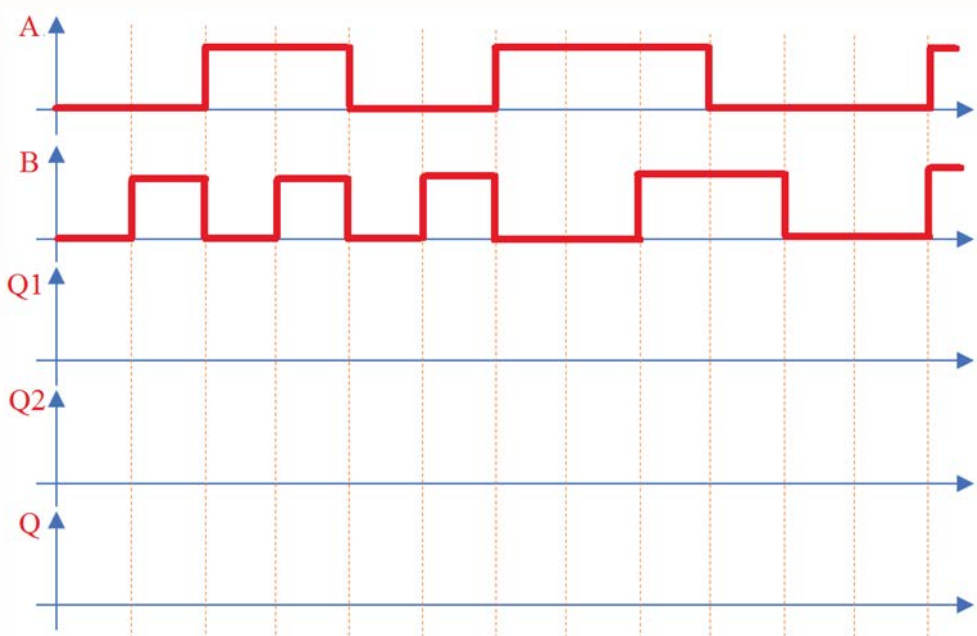
AMAÇ: **Lojik kapılar ile devre uygulaması yapmak.**



Görsel 1.34: Devre şeması



Görsel 1.35: Devrenin breadboard bağlantısı



Görsel 1.36: Grafik şeması

Tablo 1.11: Sonuç Tablosu

A	B	C	Q
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Q= .....

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Breadboard	-	1 adet
7408 entegresi	-	1 adet
7432 entegresi	-	1 adet
LED diyot	Kırmızı renkli	1 adet
Direnç	390 Ω	1 adet
Buton	-	3 adet
Zil teli	0,5 mm^2 kablo	2-3 m
DC güç kaynağı	+5 V DC	1 adet
El aletleri	Kargaburnu, yan keski, tornavida, izole bant vb.	-

İŞLEM BASAMAKLARI

1. Görsel 1.34'teki devreyi breadboard üzerine kurunuz (Görsel 1.35).
2. Öğretmeninizin kontrolünde devreye enerji vererek devreyi çalıştırınız.
3. Devreyi Tablo 1.11'deki sıraya uygun şekilde çalıştırınız ve her bir değeri tabloya yazınız.
4. Tablodaki değerlere dikkat ederek Görsel 1.36'da çıkış alanını oluşturunuz.
5. Enerjiyi keserek devreyi sökünüz.
6. Devrenin çıkış denklemini oluşturunuz.
7. Sonuçları yorumlayınız.

SORULAR

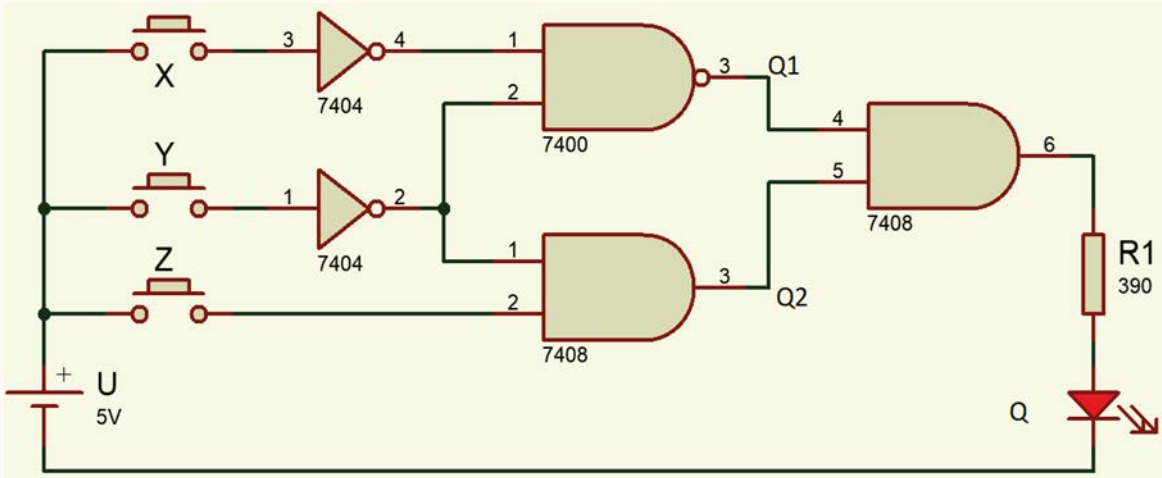
1. 7432 numaralı lojik kapı yerine hangi kapı kullanılırsa sonuç aynı olur? Açıklayınız.
2. Lojik devrenin çalışması için uygulanan gerilim en az kaç volttur? Ölçerek yazınız.
3. 7408 entegresinin özelliklerini yazınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Entegrenin içyapısını inceledi.	20	
Sınıfı :	2	Devreyi breadboard üzerine kurdu.	20	
Numarası :	3	Tablodaki sıraya uygun olarak devreyi çalıştırdı.	20	
ÖĞRETMEN	4	Çalışmayı tablodaki alana kaydetti.	20	
Adı Soyadı :	5	Grafiği veriler ışığında çizdi.	10	
	6	Malzemeleri teslim etti.	10	
İmza :		**TOPLAM PUAN**	100	

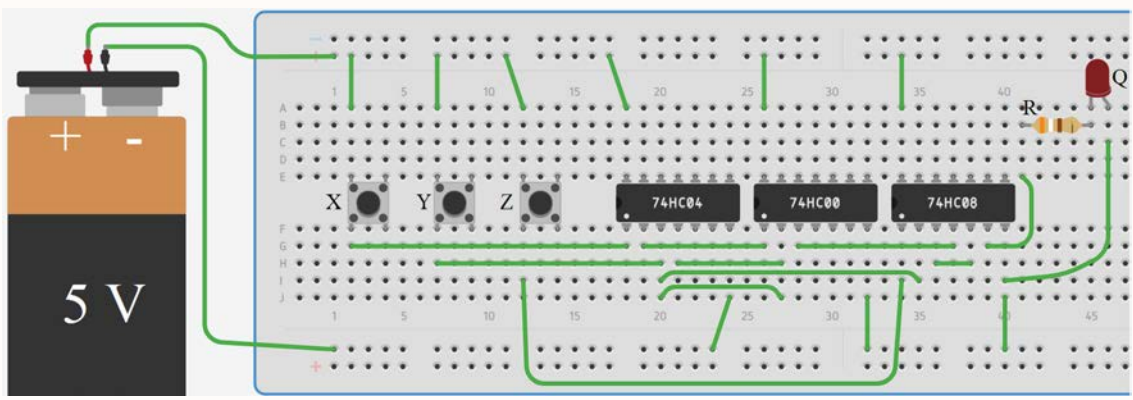
9. UYGULAMA

LOJİK KAPI UYGULAMASI 2

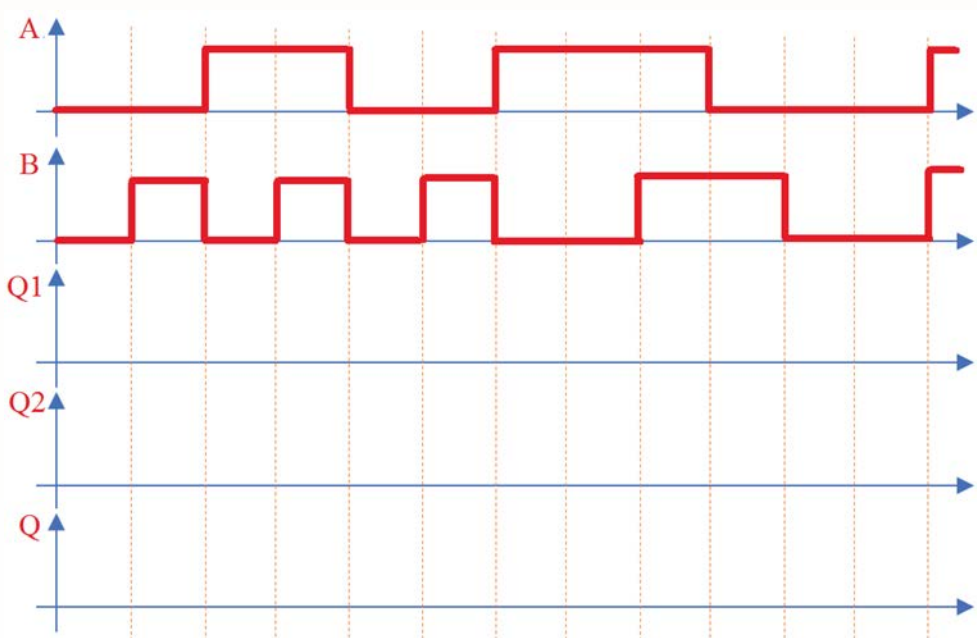
AMAÇ: **Lojik kapılar ile devre uygulaması yapmak.**



Görsel 1.37: Devre şeması



Görsel 1.38: Devrenin breadboard bağlantısı



Görsel 1.39: Grafik şeması

Tablo 1.12: Sonuç Tablosu

X	Y	Z	Q
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Q= .....

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Breadboard	-	1 adet
7400 entegresi	-	1 adet
7404 entegresi	-	1 adet
7408 entegresi	-	1 adet
LED diyot	Kırmızı renkli	1 adet
Direnç	390 Ω	1 adet
Buton	-	3 adet
Zil teli	0,5 mm^2 kablo	2-3 m
DC güç kaynağı	+5 V DC	1 adet
El aletleri	Kargaburnu, yan keski, tornavida, izole bant vb.	-

İŞLEM BASAMAKLARI

1. Görsel 1.37'deki devreyi breadboard üzerine kurunuz (Görsel 1.38).
2. Öğretmeninizin kontrolünde devreye enerji vererek devreyi çalıştırınız.
3. Devreyi Tablo 1.12'deki sıraya uygun şekilde çalıştırınız ve her bir değeri tabloya yazınız.
4. Tablodaki değerlere dikkat ederek Görsel 1.39'daki çıkış alanını oluşturunuz.
5. Enerjiyi keserek devreyi sökünüz.
6. Devrenin çıkış denklemini oluşturunuz.
7. Sonuçları yorumlayınız.

SORULAR

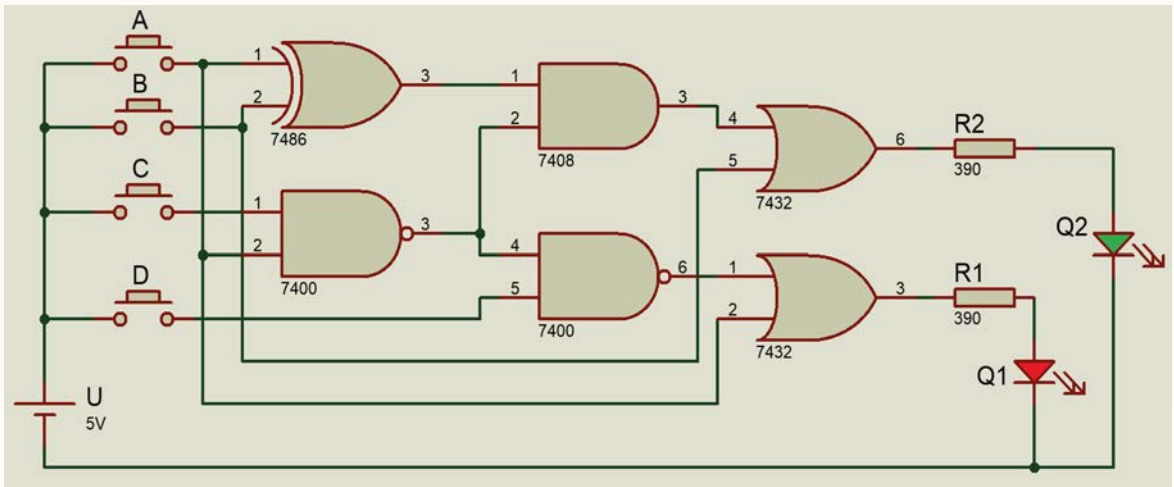
1. Görsel 1.37'de bulunan devrede kaç adet entegre kullanırsınız? Belirtiniz.
2. 7408 entegresinin içyapısında kaç adet kapı bulunur? Yazınız.
3. Q1, Q2, Q3 çıkışları hangi durumlarda lojik 1'dir? Açıklayınız.

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Entegrenin içyapısını inceledi.	20	
Sınıfı :	2	Devreyi breadboard üzerine kurdu.	20	
Numarası :	3	Tablodaki sıraya uygun olarak devreyi çalıştırdı.	20	
ÖĞRETMEN	4	Çalışmayı tablodaki alana kaydetti.	20	
Adı Soyadı :	5	Grafiği veriler ışığında çizdi.	10	
	6	Malzemeleri teslim etti.	10	
İmza :		**TOPLAM PUAN**	100	

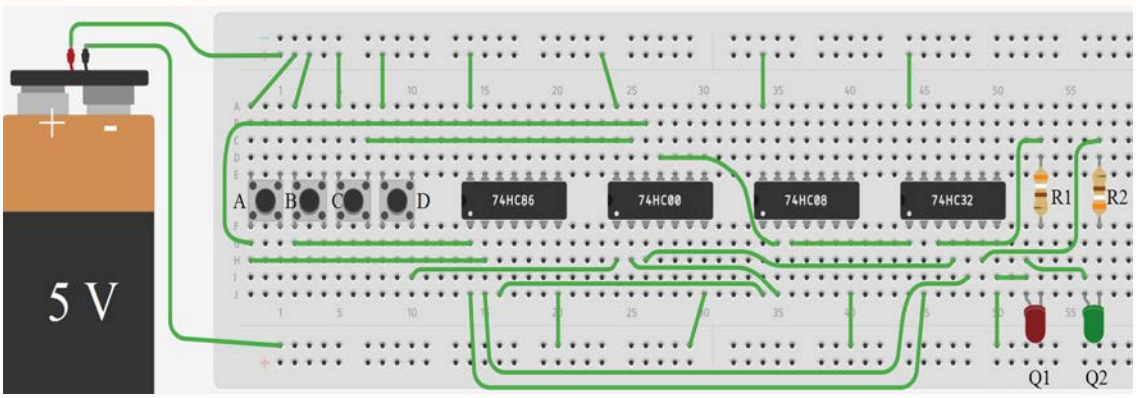
10. UYGULAMA

LOJİK KAPI UYGULAMASI 3

AMAÇ: **Lojik kapılar ile devre uygulaması yapmak.**



Görsel 1.40: Devre şeması



Görsel 1.41: Devrenin breadboard bağlantısı

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Breadboard	-	1 adet
7400 entegresi	-	1 adet
7408 entegresi	-	1 adet
7432 entegresi	-	1 adet
7486 entegresi	-	1 adet
LED diyot	Kırmızı renkli	1 adet
Direnç	390 Ω	2 adet
Buton	-	4 adet
Zil teli	0,5 mm^2 kablo	2-3 m
DC güç kaynağı	+5 V DC	1 adet
El aletleri	Kargaburnu, yan keski, tornavida, izole bant vb.	-

Tablo 1.13: Sonuç Tablosu

A	B	C	D	Q1	Q2
0	0	0	0		
0	0	0	1		
0	0	1	0		
0	0	1	1		
0	1	0	0		
0	1	0	1		
0	1	1	0		
0	1	1	1		
1	0	0	0		
1	0	0	1		
1	0	1	0		
1	0	1	1		
1	1	0	0		
1	1	0	1		
1	1	1	0		
1	1	1	1		

Q1= .....

Q2= .....

İŞLEM BASAMAKLARI

1. Görsel 1.40'taki devreyi breadboard üzerine kurunuz (Görsel 1.41).
2. Öğretmeninizin kontrolünde devreye enerji vererek devreyi çalıştırınız.
3. Devreyi Tablo 1.13'teki sıraya uygun şekilde çalıştırınız ve her bir değeri tabloya yazınız.
4. Enerjiyi keserek devreyi sökünüz.
5. Devrenin çıkış denklemini oluşturunuz.
6. Sonuçları yorumlayınız.

SORULAR

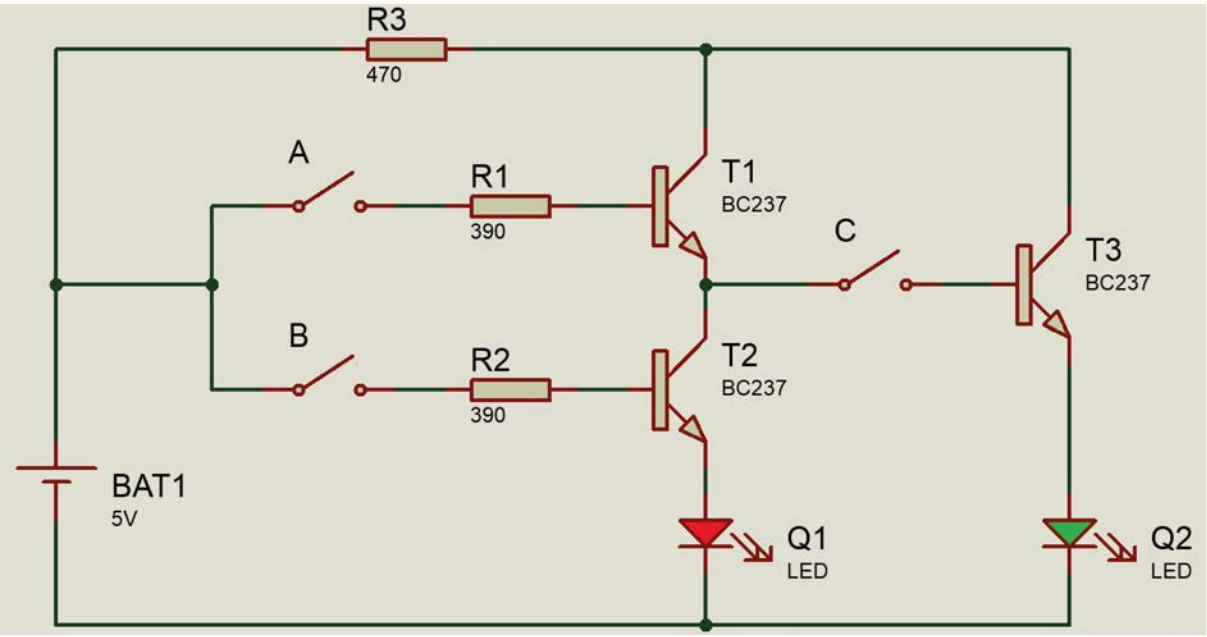
1. Görsel 1.40'daki devrenin çalışmasını anlatınız.
2. Tüm girişler "0" olduğunda çıkışlar ne olur? Belirtiniz.
3. Lojik kapı girişlerinde yazan numaralar neyi ifade eder? Açıklayınız.
4. Q1 ve Q2 çıkışlarının lojik 1 olduğu durumlarda butonların pozisyonlarını (açık-kapalı) yazınız.

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Entegrenin içyapısını inceledi.	20	
Sınıfı :	2	Devreyi breadboard üzerine kurdu.	20	
Numarası :	3	Tablodaki sıraya uygun olarak devreyi çalıştırdı.	20	
ÖĞRETMEN	4	Çalışmayı tablodaki alana kaydetti.	20	
Adı Soyadı :	5	Grafiği veriler ışığında çizdi.	10	
	6	Malzemeleri teslim etti.	10	
İmza :		**TOPLAM PUAN**	100	

11. UYGULAMA

LOJİK KAPI UYGULAMASI 4

AMAÇ: **Lojik kapılar ile devre uygulaması yapmak.**



Görsel 1.42: Devre şeması

Tablo 1.14: Sonuç Tablosu

A	B	C	Transistörlü Devre		Lojik Devre	
			Q1	Q2	Q1	Q2
0	0	0				
0	0	1				
0	1	0				
0	1	1				
1	0	0				
1	0	1				
1	1	0				
1	1	1				

Q1= .....

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Breadboard	-	1 adet
LED diyot	Kırmızı ve yeşil renkli	2 adet
Entegreler	7400, 7404, 7408, 7432 entegreleri	-
390 Ω direnç	-	2 adet
470 Ω direnç	-	1 adet
Buton	-	3 adet
Transistör	BC237 NPN transistör	3 adet
Zil teli	0,5 mm^2 kablo	2-3 m
DC güç kaynağı	+5 V DC	1 adet
El aletleri	Kargaburnu, yan keski, tornavida, izole bant vb.	-

İŞLEM BASAMAKLARI

1. Görsel 1.42'deki devreyi breadboard üzerine kurunuz.
2. Öğretmeninizin kontrolünde devreye enerji vererek devreyi çalıştırınız.
3. Devreyi Tablo 1.14'teki sıraya uygun şekilde çalıştırınız ve her bir değeri tabloya yazınız.
4. Enerjiyi keserek devreyi sökünüz.
5. Lojik kapıları kullanarak devreyi tekrardan tasarlayınız.
6. Devreyi breadboard üzerine kurunuz ve tablodaki sıraya uygun şekilde çalıştırınız.
7. Elde edilen verileri tabloya sırayla yazınız.
8. Devrenin çıkış denklemini oluşturunuz.
9. Sonuçları yorumlayınız.

SORULAR

1. Görsel 1.42'deki devrenin çalışmasını anlatınız.
2. Verilen bu devre nasıl bir lojik kapı devresidir? Açıklayınız.
3. Bu devreyi ilgili entegrelerle tekrardan çiziniz.
4. Devreyi lojik kapılarla tekrar çiziniz.
5. Devreyi çalışma hızı bakımından lojik kapılarla kıyaslayınız.
6. Bu devrenin lojik entegrelerle yapılan devreye göre avantaj ve dezavantajlarını yazınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Entegrenin içyapısını inceledi.	20	
Sınıfı :	2	Devreyi breadboard üzerine kurdu.	20	
Numarası :	3	Tablodaki sıraya uygun olarak devreyi çalıştırdı.	20	
ÖĞRETMEN	4	Çalışmayı tablodaki alana kaydetti.	20	
Adı Soyadı :	5	Grafiği veriler ışığında çizdi.	10	
	6	Malzemeleri teslim etti.	10	
İmza :		**TOPLAM PUAN**	100	

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki cümlelerin başındaki boşluğa cümleler doğru ise (D), yanlış ise (Y) yazınız.

1. (…) Lojik kapılar mantık işlemleri yapar.
2. (…) Her kapının iki girişi vardır.
3. (…) Her kapının bir çıkışı vardır.
4. (…) En basit kapı bir giriş ve bir çıkıştan oluşur.
5. (…) TTL kısaltması, transistör-transistör-lojik anlamına gelir.
6. (…) CMOS bir çeşit lojik entegredir.
7. (…) Entegrelerin çalışma hızları düşüktür.

B) Aşağıdaki cümlelerde boş bırakılan yerlere uygun sözcüğü yazınız.

8. Desimal sayı sistemi …… adet rakamdan oluşur.
9. İkili sayı sistemine …… sayı sistemi de denir.
10. Heksadesimal kodda rakamların yanında …… bulunur.
11. Lojik kapıların en az ……. ucu vardır.
12. Oktal sayı sisteminde …… adet rakam vardır.
13. Lojik entegrelerin besleme uçları …… ile …… no.lu uçlardır.
14. Lojik kapı devrelerinde kapılar yerine …… kullanılabilir.

C. Aşağıda verilen ifadeler ve bu ifadelere karşılık gelen kavramları, ifadelerin önündeki parantezlere kavramların önündeki harflerden uygun olanları yazarak eşleştiriniz.

İFADELER		
15.	(…)	Binary
16.	(…)	Lojik
17.	(…)	Entegre
18.	(…)	Tampon
19.	(…)	Değil
20.	(…)	Oktal
21.	(…)	Desimal

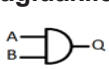
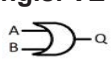
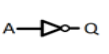
KAVRAMLAR	
A	Buffer
B	Not
C	İkili
D	Mantık
E	Birleşik
F	Sekizli
G	Onlu
Ğ	Engel

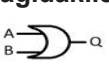
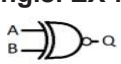
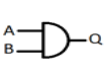
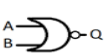
Ç) Aşağıdaki soruların doğru seçeneğini işaretleyiniz.

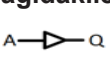
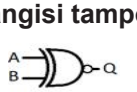
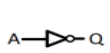
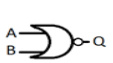
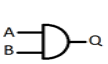
22. Desimal sistemde aşağıdakilerden hangisi yoktur?
A) 0 B) 1 C) 7 D) 9 E) 10

23. Binary sayı sisteminde yazılan aşağıdaki sayılardan hangisi yanlıştır?
A) 101 B) 100 C) 120 D) 111 E) 1000

24. Aşağıdakilerden hangisi binary sayı sisteminde 3'e karşılık gelir?
A) 0011 B) 0110 C) 1001 D) 111 E) 1100

25. Aşağıdakilerden hangisi VE kapısıdır?
A)  B)  C)  D)  E) 

26. Aşağıdakilerden hangisi EX-NOR kapısıdır?
A)  B)  C)  D)  E) 

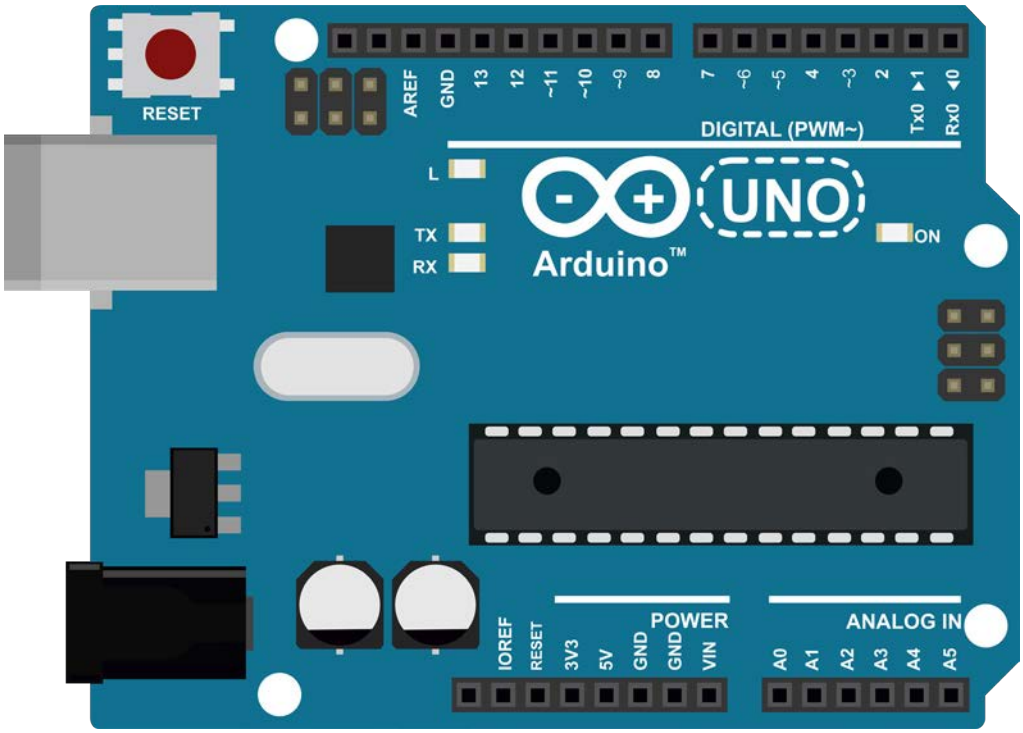
27. Aşağıdakilerden hangisi tampon olarak kullanılır?
A)  B)  C)  D)  E) 

2. ÖĞRENME BİRİMİ

MİKRODENETLEYİCİ PROGRAMLAMA

KONULAR

2.1. MİKRODENETLEYİCİ SEÇİMİ
2.2. MİKRODENETLEYİCİ EDİTÖRÜ
2.3. MİKRODENETLEYİCİ YAZILIMI



TEMEL KAVRAMLAR

Akış diyagramları, mikroişlemci, mikrodenetleyici, programlama editörü

HAZIRLIK ÇALIŞMALARI

1. Mikroişlemci hakkında bildiklerinizi arkadaşlarınızla paylaşınız.
2. Akış diyagramları hakkında düşüncelerinizi ifade ediniz.

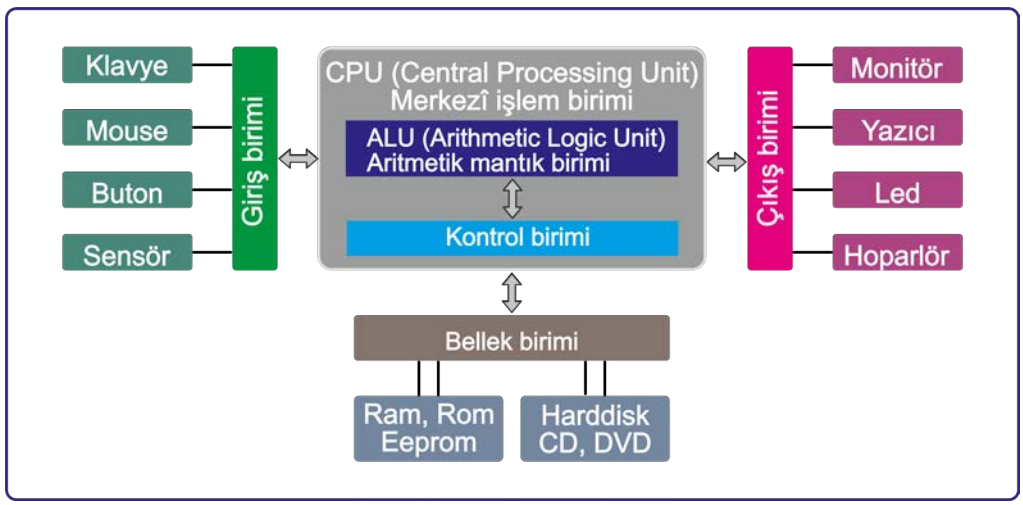
2.1. MİKRODENETLEYİCİ SEÇİMİ

Mikroişlemci ile mikrodenetleyici genellikle birbirine karıştırılan kavramlardır. Mikroişlemci ve mikrodenetleyici kavramlarının çok iyi bilinmesi ve birbirinden ayırt edilmesi gerekmektedir.

2.1.1. Mikroişlemciler ve Mikrodenetleyiciler

Mikroişlemci (Microprocessor)

Merkezî işlem birimi (CPU/Central Processing Unit) olarak da adlandırılır. Tek bir tümleşik devre içine yerleştirilmiş, kontrol ve aritmetik mantık birimlerinden (toplama, çıkarma, çarpma, bölme, sayıları karşılaştırma, mantıksal işlemler vs.) oluşan lojik bir ünitedir (Görsel 2.1). Kullanıcı tarafından yazılan komut dizelerine (program, yazılım, uygulama) göre giriş veya bellek birimlerinden aldığı bilgileri okur, değerlendirir ve çıkış birimlerine gönderir. Giriş, bellek ve çıkış birimleri, haricî entegre devre olarak bağlanmalıdır, bu nedenle genel amaçlı uygulamalarda kullanılır.



Görsel 2.1: Mikroişlemci blok diyagramı

Mikrodenetleyici (Microcontroller)

İçinde aritmetik lojik ünite, bellek, giriş ve çıkış birimlerini bir arada bulunduran tümleşik devrelerdir. Bazı mikrodenetleyicilerde ADC (Analog dijital çevirici), DAC (Dijital analog çevirici), UART (Seri haberleşme), PWM (Darbe genişlik modülasyonu), TIMERS (Zamanlayıcılar) vs. özel birimler de mevcuttur, bu nedenle özel amaçlı uygulamalarda kullanılır. Tablo 2.1'de mikroişlemci ve mikrodenetleyici arasındaki farklar verilmiştir.

Tablo 2.1: Mikroişlemci ve Mikrodenetleyicinin Karşılaştırılması

MİKROİŞLEMCİ	MİKRODENETLEYİCİ
ALU içerir. Ram, rom, giriş-çıkış birimi, timer vs. ayrıca bağlanır.	Hepsi tek bir tümleşik devre içindedir.
Tasarımcı bellek, giriş-çıkış (I/O) birimlerinin büyüklüklerini kendisi belirler ve ona göre tasarımı gerçekleştirir.	Bu birimler dâhilî olduğu için tasarımı daha kolaydır.
Tasarım çok yer kaplar.	Tasarım az yer kaplar.
Sistem maliyeti fazladır.	Sistem maliyeti ucuzdur.
Güç tüketimi yüksektir.	Güç tüketimi düşüktür.
Genellikle bilgisayar, tablet ve cep telefonlarında kullanılır.	Robotik sistemler, motor kontrol sistemleri, otomasyon sistemleri, askerî cihazlar, rüzgâr ve güneş enerjisi santralleri, otomotiv sektörü, elektrikli ev aletleri vb.lerinde kullanılır.

Birçok üretici firma ve mikrodenetleyici çeşidi vardır. Bunların arasından tasarımı yapılacak proje için en uygun seçim yapılmalıdır. Mikrodenetleyici seçiminde aşağıdaki bilgiler sıralanır ve süzülüp seçim yapılır.

- Yeterli giriş-çıkış sayısı ve birimleri olmalıdır. Örneğin projede analog ölçüm ve haberleşme kullanılacaksa bu birimlere sahip mikrodenetleyiciler bulunmalıdır.
- Program hafızası ve çalışma hızı proje için yeterli seviyede olmalıdır.
- Programlanması ve hata ayıklaması kolay olmalıdır.
- Seri üretim projeler için ulaşılabilirliği ve devamlılığı olmalıdır.
- Maliyeti düşük olmalıdır.
- Mikrodenetleyicinin pcb için kılıf seçimi yapılmalıdır (DIP veya SMD).
- Güç tüketimi az olmalıdır.
- Mikrodenetleyici özelliklerini öğrenebilmek için geliştirme kartı bulunmalıdır.

2.1.2. Mikrodenetleyici Çeşitleri

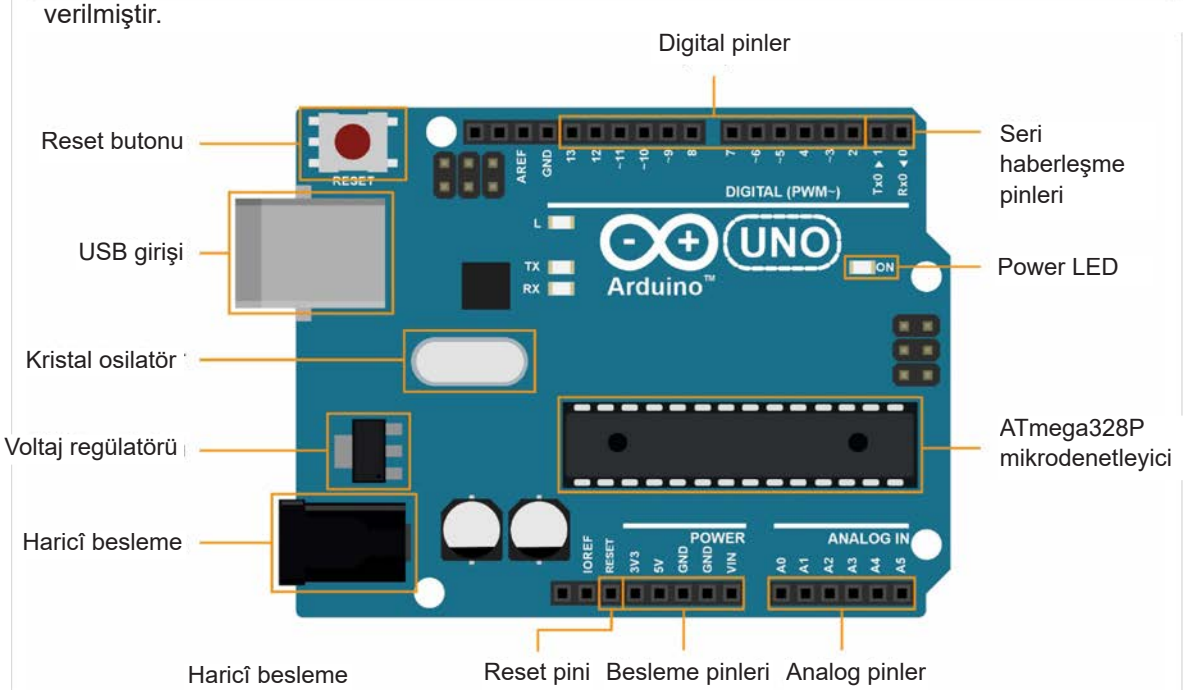
PIC, AVR, STM, MSP, PSoC, RISC-V çok kullanılan mikrodenetleyici aileleridir. Bunlar 8-16-24-32 bit veri yollarına sahiptir. İlk geliştirilen mikrodenetleyiciler 8 bit mimaride olmasına rağmen 32 bit mikrodenetleyiciler 8 bit mikrodenetleyicilerin yerini almaktadır.

2.1.3. Mikrodenetleyici Donanım Özellikleri

Mikrodenetleyici seçimine göre uygun olan ATmega328P'nin bazı donanım özellikleri şunlardır:

32 KB programlanabilir FLASH (tekrar yazılıp silinebilir) program hafızası	1 KB EEPROM	2 KB dâhilî SRAM bellek
23 adet programlanabilir giriş ve çıkış ucu	6 adet PWM kanalı	6 kanal 10-bit ADC
1 adet programlanabilir USART	1 adet I2C arayüzü	Haricî ve dâhilî kesme kaynakları
	Dâhilî ayarlı osilatör	1,8 - 5,5 V arası çalışma gerilimi
	20 MHz'e kadar çalışma hızı	

Arduino geliştirici topluluğu tarafından yapılmış ve sürekli geliştirilen arduino uno, atmel (AVR) ATmega328P işlemciye sahip bir mikrodenetleyici geliştirme kartıdır. Görsel 2.2'de donanımsal yapısı verilmiştir.



Görsel 2.2: Arduino geliştirme kartı donanımsal yapısı

2.1.4. Mikrodenetleyici Programlama Dilleri

Programlama: Mikroişlemci veya mikrodenetleyiciye bir görevi nasıl uygulayacağını söyleyen komutlar dizisidir. Programcı tarafından, kendine has biçim ve yazım kuralları olan bir programlama dili kullanılarak bilgisayarda yazılır.

Derleme (Compiler): Herhangi bir programlama diliyle yazılan programın mikroişlemci veya mikrodenetleyicinin çalıştırabileceği makine diline çevrilmesi işlemidir.

Bir programlama dili, insan konuşmasına ne kadar yakınsa seviyesi o kadar yüksek, öğrenilip anlaşılması o kadar kolay olur ancak makine dilinden uzaklaşıldığından dolayı donanıma erişim ve programın işleme hızı azalır. Bazı programlama dilleri ve yüksekten düşüğe doğru seviyeleri şu şekildedir:

Çok yüksek seviyeli görsel diller: Scratch, Simulink, Görsel programlama dilleri

Yüksek seviyeli diller: Basic, Python, Lua

Orta seviyeli diller: C, C++

Düşük seviyeli diller: Assembly

Makine dili: Kodlar 1 ve 0'lardan oluşur.

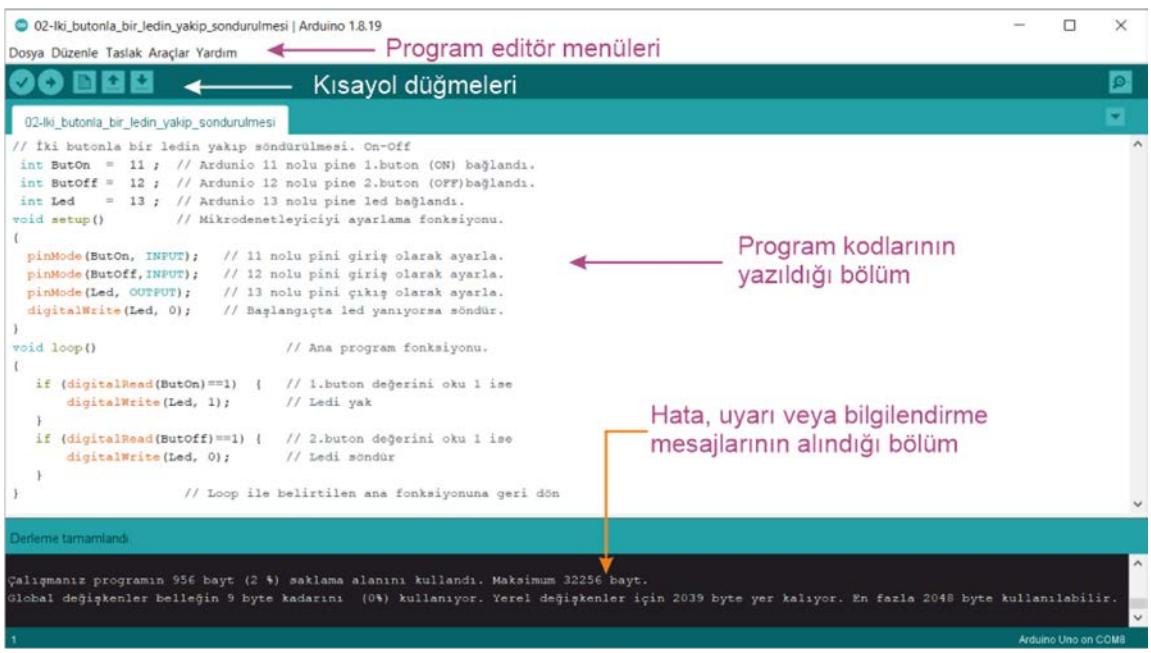
Arduino geliştirme kartlarında programlama dili olarak birden fazla alternatif bulunmaktadır. Bunlardan bazıları mBlock, Scratch, Arduino IDE vb.dir. Derleyicilerde genellikle Assembly, C, C++ veya Basic dili komutları kullanılır. Arduino IDE, temelde C dilini kullanan geliştirme aracıdır.

2.2. MİKRODENETLEYİCİ EDİTÖRÜ

Editörler, program metninin yazılması ve düzenlenmesi için geliştirilmiş uygulamalardır. Geliştirilmiş mikrodenetleyici editörleri programlama kolaylığı sağlar. Hata ayıklama, uyarıda bulunma, yazım yanlışlarını düzeltme önerileri gibi birçok kolaylık sağlar.

2.2.1. Mikrodenetleyici Program Editörü Kurulumu

Arduino programlama editörü internet sitesinden indirilir ve kurulumu gerçekleştirilir. Arduino kısayolu çalıştırılarak editör programı açılır (Görsel 2.3).



Görsel 2.3: Arduino programlama editörü

2.2.2. Mikrodenetleyici Program Editörü Menüleri

Dosya menüsü; yeni bir dosya açma, kayıt yapma, farklı kaydetme, önceden oluşturulan taslak ve örnekleri açma-kapatma, sayfa ayarlarını yapma, yazdırma ve çıkış yapma işlemlerini içermektedir.

Düzenle menüsü; geri alma, tekrarlama, kesme, kopyalama, yapıştırma, tümünü seçme, satıra gitme, yorum yapma, girintiyi artırma veya azaltma, görünümü büyültme veya küçültme ve çeşitli bulma işlemlerini içermektedir.

Taslak menüsü; doğrulama-derleme, yükleme, derlenmiş dosyayı çıkartma, çalışma klasörünü gösterme, kütüphane ve dosya ekleme işlemlerini içermektedir.

Araçlar menüsü; otomatik biçimlendirme, arşivleme, karakter kodlamasını düzeltme/tekrar yükleme, kütüphaneleri düzenleme, seri port ekranı, seri çizici, wifi ayarları, kart seçimi, port seçimi ve bilgisi, programlayıcı seçimi ve önyükleyiciyi yazdırma işlemlerini içermektedir.

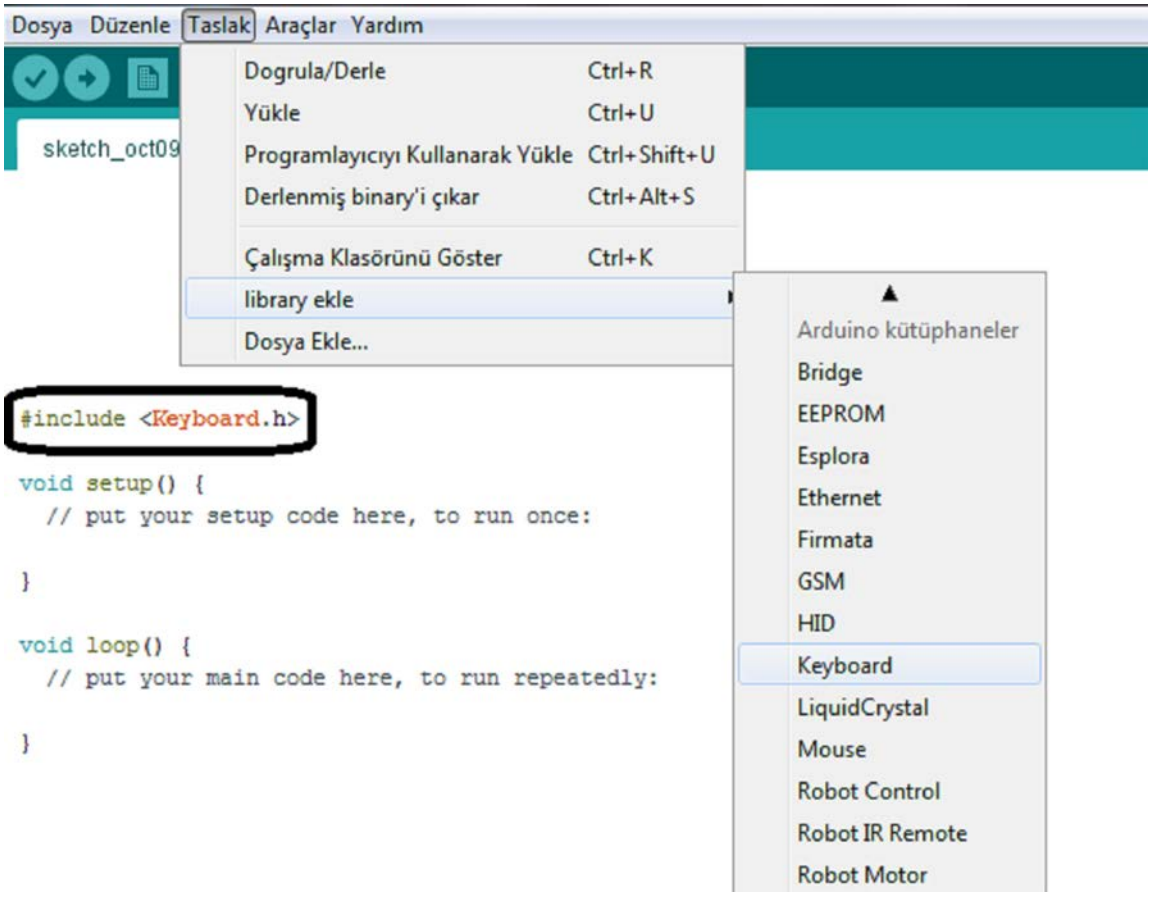
Yardım menüsü; başlangıç yardımı, sorun giderme, kaynak bulma, kaynaklarda arama yapma işlemlerini; sıkça sorulan soruları ve arduino hakkında yardımcı bilgileri içermektedir.

Yeni Bir Dosya Açmak

Arduino programında yeni bir dosya açmak için Dosya menüsü altındaki **Yeni** sekmesine tıklanır. Yeni dosya açmanın kısayolu **"Ctrl+N"** dir. Yeni bir dosya oluşturulduktan sonra editör üzerinde program kodları yazılır.

Arduino Kütüphane Ekleme

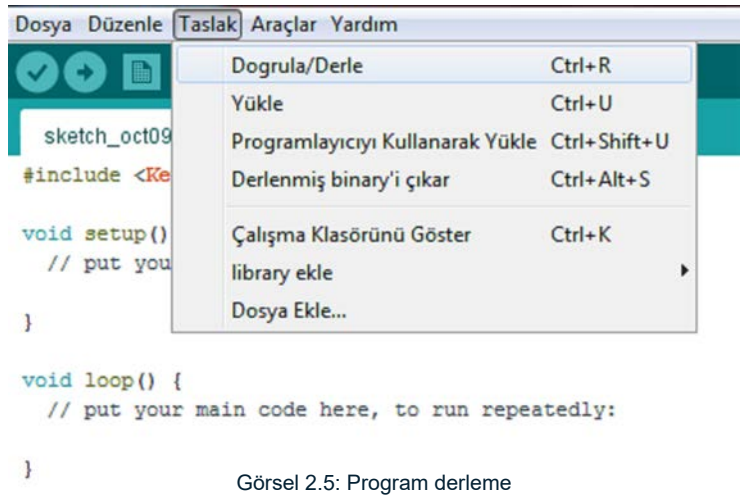
Arduino programında bazı işlemleri daha hızlı ve kolay yapabilmek amacıyla hazır yazılmış kütüphaneler bulunmaktadır. Yapılacak işleme uygun kütüphanenin bulunup eklenmesi için Arduino programında Taslak menüsü altında bulunan library (kütüphane) ekle tıklanır. Buradan ilgili kütüphane seçildiğinde program satırına ".h" uzantılı kütüphane eklenmiş olur (Görsel 2.4).



Görsel 2.4: Kütüphane ekleme

Programı Derleme

Yazılan programı kontrol etmek ve derlemek için editör programı üzerinde bulunan kontrol et düğmesine (√) veya **Taslak** menüsü altındaki **Doğrula/Derle**'ye tıklanır. Yazımda bir hata varsa bu hatalar editör programı tarafından bildirilir. Hata yoksa program derlenerek arduino'ya yüklenmek üzere hazırlanmış olur (Görsel 2.5). Program derlemenin kısayolu "**CTRL+R**" dir.



Görsel 2.5: Program derleme

Program Yükleme

Yazılan bir programı arduino'ya yüklemek için editör programı üzerinde bulunan yükle düğmesine (→) veya **Taslak** menüsü altındaki **Yükle**'ye tıklanır. Yüklemenin hatasız bir şekilde yapılması için ardunio geliştirme kartı bilgisayara bağlanmış ve programlanmaya hazır olmalıdır. Bu işlemden sonra program arduino'ya yüklenir ve çalıştırılır.

2.3. MİKRODENETLEYİCİ YAZILIMI

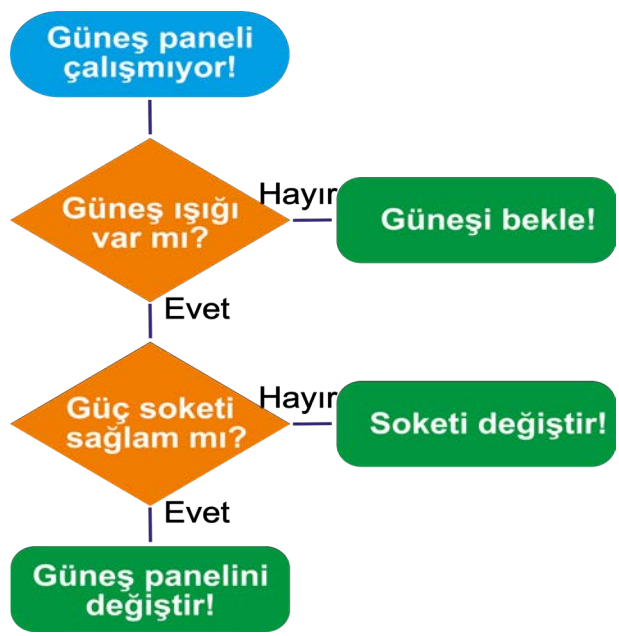
Günümüzde birçok mikrodenetleyici programlama yazılımı bulunmaktadır. Mikrodenetleyicilerin giderek çeşitlenmesi ve artmasıyla birlikte programlama yazılımları da çeşitlenip artmaktadır. İlk başlarda basit bir programlama yazılımı yapmak mümkün değilken gelişen teknolojiyle kolay ve geliştirilebilir yazılımlar piyasaya sunulmaktadır. Mikrodenetleyici seçimi yapıldıktan sonra uygun programlama dili seçilip kullanılmalıdır.

2.3.1. Mikrodenetleyici Akış Diyagramı

Akış diyagramları, bir proje veya hata çözme süreçlerinin farklı şekillerle görsel bir şema üzerinden gösterilmesidir. Hızlı ve planlı bir şekilde çözüme ulaşılmasını sağlar.

Görsel 2.6'da bir güneş panelinin kontrolüne ait örnek bir akış şeması verilmiştir. Burada güneş panelinde oluşan arızanın giderilmesi için yapılacak işlemler akış şeması şeklinde sıralanmıştır.

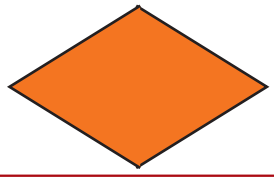
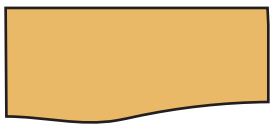
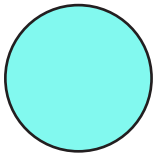
Program yazmadan önce akış diyagramını oluşturmak, program bütünlüğünü görmek ve oluşabilecek hataların önüne geçmek için büyük fayda sağlar. Yazılacak programın tüm detayları



Görsel 2.6: Örnek bir akış diyagramı

şekillerle oluşturulduğu için program yazımını kolaylaştırır. Akış diyagramı oluşturulduktan sonra görsel zemine oturtulan proje, uygun program editöründe yazılır ve yüklenir. Tablo 2.2'de programlamada en çok kullanılan akış diyagramı şekil ve işlevleri verilmiştir.

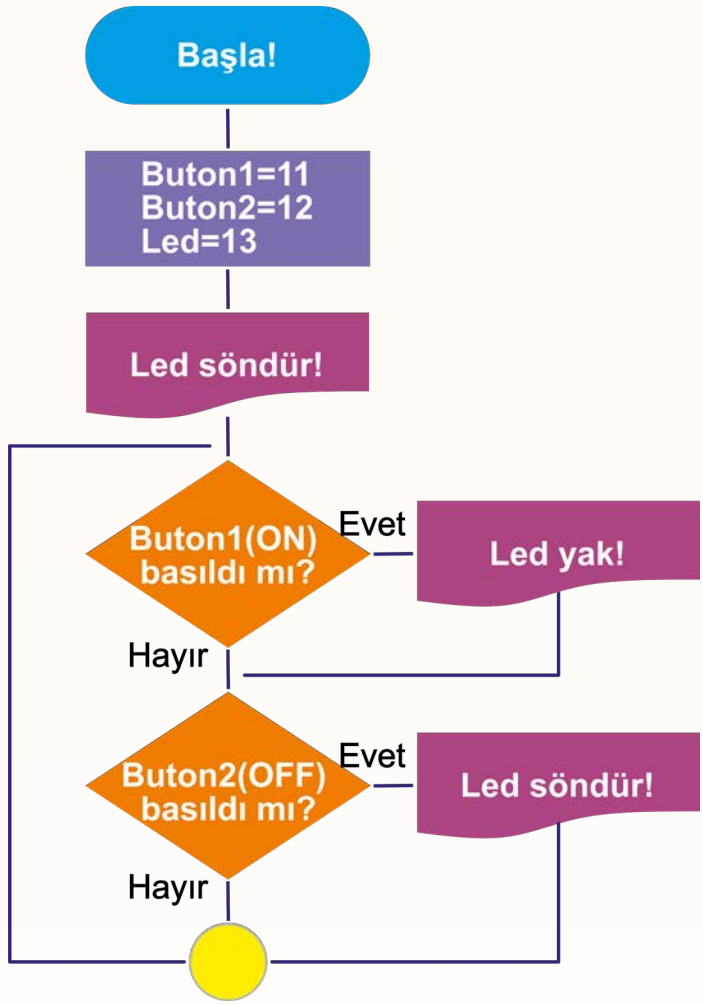
Tablo 2.2: Programlamada En Çok Kullanılan Akış Diyagramı Şekil ve İşlevleri

AKIŞ DİYAGRAMI ŞEKLİ	İŞLEVİ
	**Başla/Bitir:** Programın başlangıç veya bitişini gösterir.
	**Giriş:** Dış ortamdan girilecek veri buraya yazılır. Örneğin sıcaklık sensöründen gelen veri.
	**Atama/İşlem:** Ne yapılması gerektiği buraya yazılır. Örneğin sıcaklık sensör verisini 0,5 ile çarp gibi.
	**Karar/Kontrol:** Kontrol etme ve karşılaştırma gibi işlemler için kullanılır. Bir işlem sonucu doğru veya yanlış olacaktır. Doğruysa **Evet**, yanlışsa **Hayır** çıkışı alınır. Örneğin sensör sıcaklık değeri 50 ^{0}C'nin üzerinde mi?
	**Veri çıkışı:** İşlem sonucuna göre ekrana (monitör, lcd, seri port ekranı vs.) veri yazdırmak için kullanılır. Örneğin sensör sıcaklık değerini lcd ekrana yaz gibi.
	**Döngü:** Programlarda bazen belirlenen bir değere ulaşıncaya kadar tekrar etmesi istenir. Örneğin sensör sıcaklık değeri 30 ^{0}C'nin üzerindeyse 30 ^{0}C olana kadar fanı çalıştır gibi.
	**Akış yönü:** İşlem sonuçlarına göre akışın nereye gideceği belirtilir. Örneğin karar sonucuna göre "Evet" ise bir alta, "Hayır" ise bir üste git gibi.
	**Bağlantı:** Birleştirici bir simgedir. Farklı kollarda dallanan işlem akışlarını bir araya getirmek veya bir sayfaya sığmayan akış diyagramıyla sonraki sayfada bulunan akış diyagramlarını birleştirmek için de kullanılabilir.
	**Fonksiyon/Alt program:** Programda kullanılan alt program veya fonksiyon bu şekille ifade edilir. Örneğin LCD ekran çıkış alt programı veya fonksiyonu gibi.

1. UYGULAMA

AKIŞ DİYAGRAMINA GÖRE YAZILMIŞ PROGRAMIN MİKRODENETLEYİCİ KARTA YÜKLENMESİ

AMAÇ: **Mikrodenetleyici karta program yüklemek.**

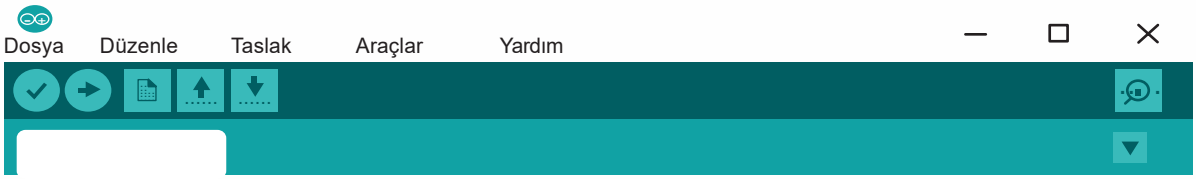


Görsel 2.7: İki butonla bir LED'in yakılıp söndürülmesine ait akış diyagramı

İŞLEM BASAMAKLARI

1. Görsel 2.7'de verilen akış diyagramını ve uygulamaya ait programı inceleyiniz.
2. Akış diyagramında bulunan tüm şekillerin işlevlerini akış yönlerini de dikkate alarak açıklayınız.
3. Uygulamaya ait programı arduino editöründe yazınız.
4. Yazdığınız programı derleyiniz ve hata oluşup oluşmadığına bakınız.
5. Hata oluşturmak için pinMode(ButOn, INPUT); fonksiyonunun sonundaki **;** işaretini kaldırınız ve tekrar derleyiniz.
6. Hatalı yazımda oluşan uyarıyı arduino editörü altından takip ediniz.
7. Hatayı düzeltiniz.
8. Mikrodenetleyici kartı bilgisayarınıza bağlayınız.
9. Yazdığınız programı mikrodenetleyici karta yükleyiniz.
10. Hata mesajı almadan programın mikrodenetleyici karta yüklendiğini gözlemleyiniz.
11. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM



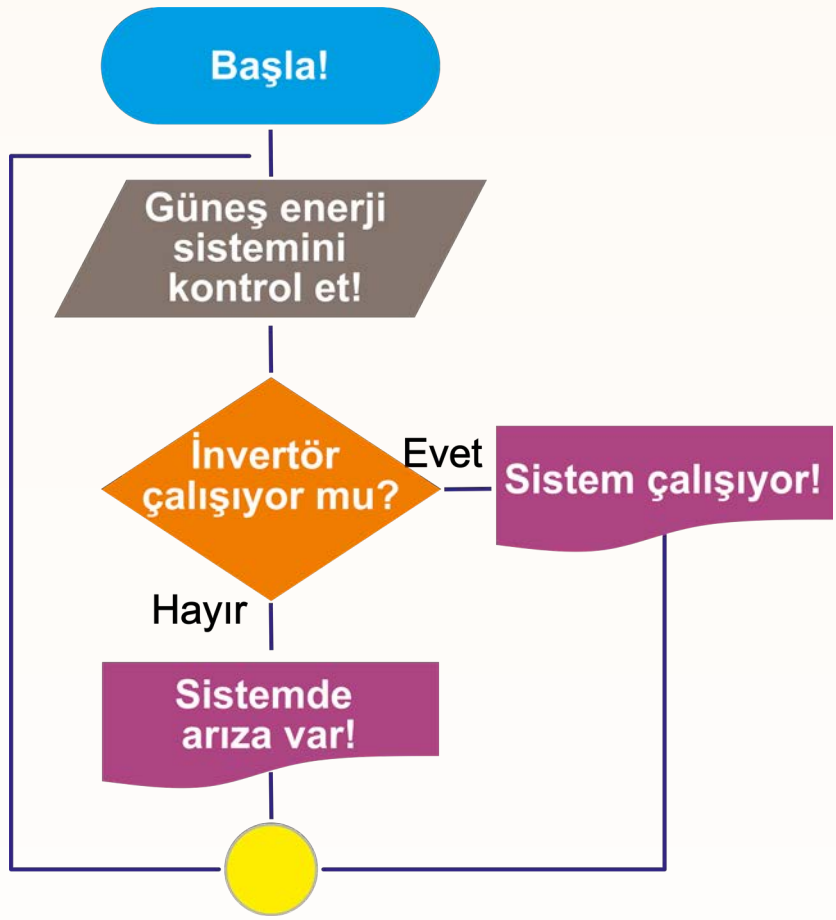

```
                                    // İki butonla bir ledin yakılıp söndürülmesi. On-Off
 int ButOn  =  11 ;                 // Ardunio 11 no.lu pine 1.buton (ON) bağlandı.
 int ButOff =  12 ;                 // Ardunio 12 no.lu pine 2.buton (OFF) bağlandı.
 int Led    =  13 ;                 // Ardunio 13 no.lu pine LED bağlandı.
void setup()                        // Mikrodenetleyiciyi ayarlama fonksiyonu.
{
  pinMode(ButOn, INPUT);            // 11 no.lu pini giriş olarak ayarla.
  pinMode(ButOff,INPUT);            // 12 no.lu pini giriş olarak ayarla.
  pinMode(Led, OUTPUT);             // 13 no.lu pini çıkış olarak ayarla.
  digitalWrite(Led, 0);             // Başlangıçta LED yanıyorsa söndür.
}
void loop()                         // Ana program fonksiyonu.
{
   if (digitalRead(ButOn)==1)  {    // 1.buton değerini oku 1 ise
      digitalWrite(Led, 1);         // LED'i yak
   }
   if (digitalRead(ButOff)==1) {    // 2.buton değerini oku 1 ise
      digitalWrite(Led, 0);         // LED'i söndür
   }
}                                   // Loop ile belirtilen ana fonksiyonuna geri dön.
```

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Akış diyagramı şekillerini açıkladı.	20	
ÖĞRETMEN	4	Editörde uygulama programını derledi.	20	
Adı Soyadı :	5	Mikrodenetleyici karta programı yükledi.	20	
İmza :		**TOPLAM PUAN**	100	

2. UYGULAMA

AKIŞ DİYAGRAMININ OLUŞTURULMASI

AMAÇ: **Akış diyagramı oluşturmak.**



Görsel 2.8: Güneş enerjisi sisteminin çalışma kontrolünü içeren akış diyagramı

İŞLEM BASAMAKLARI

1. Görsel 2.8'de verilen akış diyagramını inceleyiniz ve rüzgâr türbininin regülatörü yardımıyla çalışıp çalışmadığını kontrol edecek bir akış diyagramını aşağıdaki boş alana oluşturunuz.

2. Çalışma sonrası akış diyagramınızı öğretmeninizle birlikte kontrol ediniz ve çalışma alanını temizleyiniz.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Verilen akış diyagramını doğru oluşturdu.	60	
ÖĞRETMEN				
Adı Soyadı :		**TOPLAM PUAN**	100	
İmza :				

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki cümlelerin başındaki boşluğa cümleler doğru ise (D), yanlış ise (Y) yazınız.

1. (…) ALU, aritmetik ve mantıksal işlemlerin yapıldığı birimdir.
2. (…) Mikrodenetleyiciyle yapılan devre tasarımı mikroişlemciye göre daha çok yer kaplar.
3. (…) Akış diyagramlarında ▭ sembolü giriş işlemi olarak kullanılır.
4. (…) ATmega328P mikrodenetleyicisi tekrar yazılıp silinebilir program hafızasına sahiptir.
5. (…) Makine dili insan konuşmasına en yakın dil olduğu için anlaşılması kolaydır.

B) Aşağıdaki cümlelerde boş bırakılan yerlere uygun sözcüğü yazınız.

6. Merkezî işlem birimi …… olarak da adlandırılır.
7. Monitör veya ikaz lambası …… elemanıdır.
8. İçinde aritmetik lojik, bellek giriş ve çıkış birimlerini bir arada bulunduran tümleşik devrelere …… denir.
9. Akış diyagramlarında karar sembolü …… şeklinde gösterilir.

C) Aşağıdaki soruların doğru seçeneğini işaretleyiniz.

10. Aşağıdakilerden hangisi mikroişlemci veya mikrodenetleyiciler için giriş elemanıdır?
A) Buton B) Lamba C) Monitör
D) Röle E) Yazıcı

11. Aşağıdakilerden hangisi çok yüksek seviyeli programlama dillerindendir?
A) Assembly B) Basic C) C
D) C++ E) Scratch

12. Arduino programına kütüphane eklemek için editör programından aşağıdaki menülerin hangisi seçilmelidir?
A) Araçlar B) Dosya C) Düzenle
D) Taslak E) Yardım

13. Programı mikrodenetleyici karta yüklemek için editör üzerinde hangi düğme kullanılır?
A) → B) √ C) >
D) < E) ←

14. İşlem sonucuna göre ekrana veri yazdırmak için hangi akış diyagramı şekli kullanılır?
A) ⬭ B) ◯ C) ▱
D) ▭ E) ▭

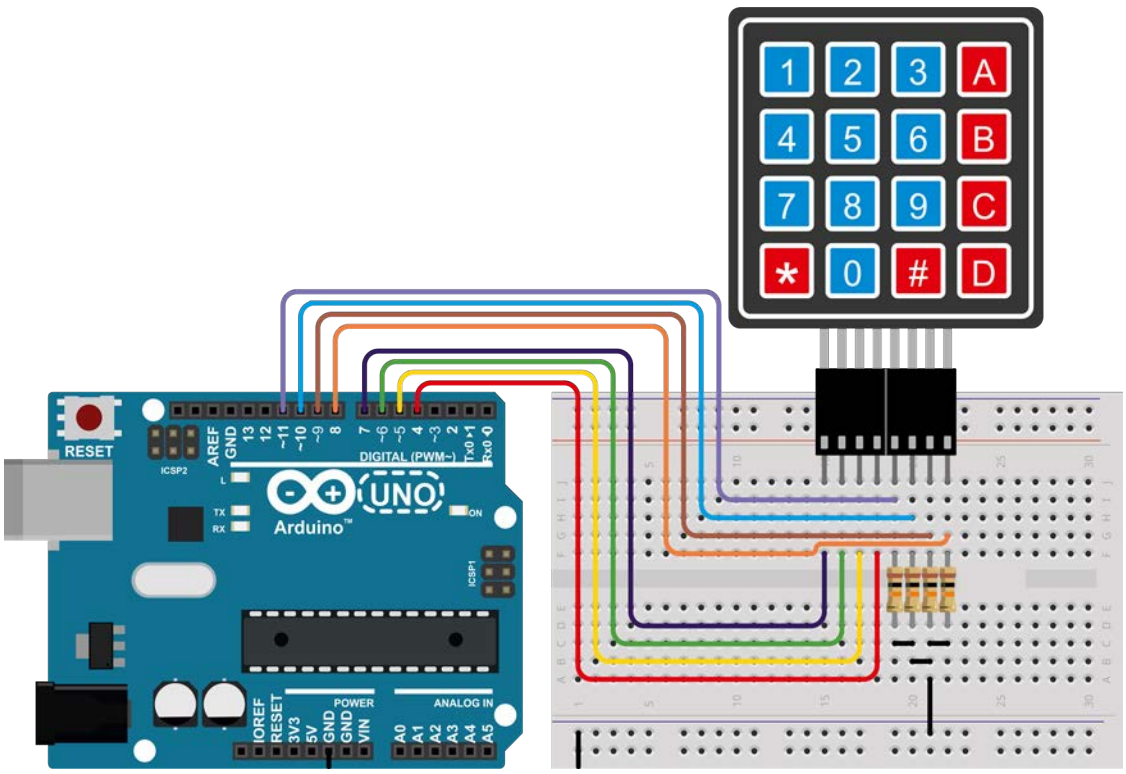
15. ⬡ **akış diyagramı şeklinin işlevi aşağıdakilerden hangisinde doğru olarak verilmiştir?**
A) Bağlantı B) Döngü C) Fonksiyon
D) Giriş E) Karar

3. ÖĞRENME BİRİMİ

DİJİTAL VE ANALOG UYGULAMALAR

KONULAR

3.1. MİKRODENETLEYİCİ UYGULAMALARI
3.2. DİJİTAL GİRİŞ-ÇIKIŞ
3.3. PORT
3.4. ANALOG GİRİŞ-ÇIKIŞ
3.5. ADC-DAC UYGULAMALARI



TEMEL KAVRAMLAR

ADC-DAC uygulamalar, analog uygulamalar, dijital uygulamalar, editör programı, mikrodenetleyici, programlama kartı, port uygulamalar

HAZIRLIK ÇALIŞMALARI

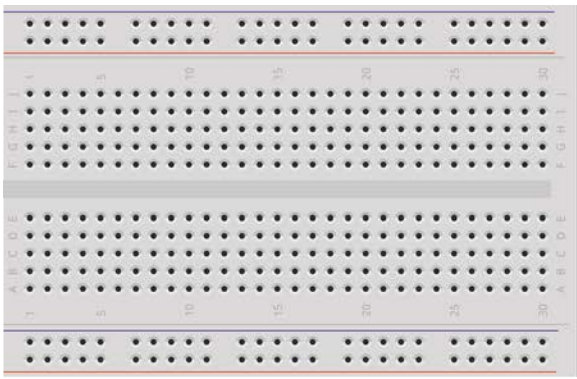
1. Mikrodenetleyici programlama ile ilgili bildiklerinizi sınıfta arkadaşlarınızla paylaşınız.
2. Analog ve dijital kavramlarını güncel örnekler vererek arkadaşlarınızla paylaşınız.

3.1. MİKRODENETLEYİCİ UYGULAMALARI

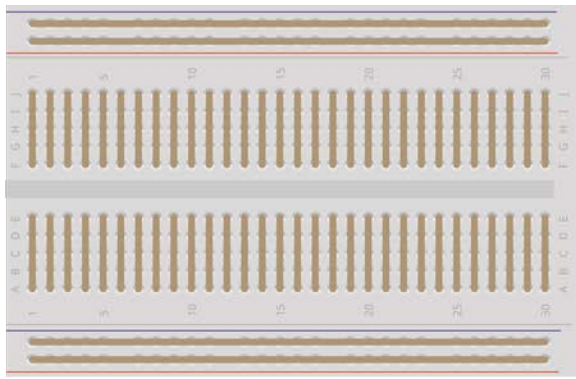
Elektronik ve mikrodenetleyici devre uygulamalarında, baskı devre ve lehimleme yapmadan önce devreleri denemek için **breadboard** kullanılır. Breadboard ile devre kurulumu basit ve hızlı olduğu için yapılan hataların düzeltilmesi açısından büyük kolaylık sağlar.

3.1.1. Mikrodenetleyici Devrelerinin Kurulması

Breadboard üzerinde birbirine paralel bağlı yollar bulunur. Alt ve üstte yatay olarak kırmızı (+) ve mavi (-) hatlara besleme gerilimi bağlanır. Orta bölümde 5'li delik grupları kendi içinde dikey bağlantılıdır. Böylelikle aynı sıradaki elektronik elemanlar birbirine bağlanmış olur (Görsel 3.1).



a) Üst görünüş

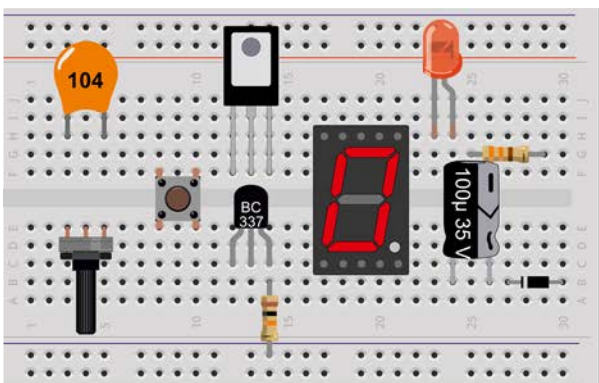


a) Breadboard'un içyapısı

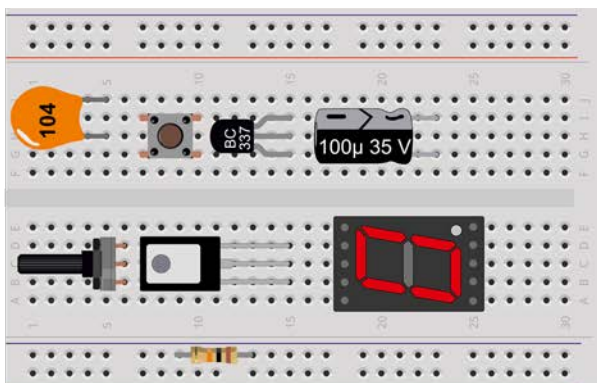
Görsel 3.1: Breadboard ve içyapısı

Elektronik devre elemanları, bacak bağlantıları tespit edildikten sonra doğru biçimde breadboard üzerine yerleştirilmelidir (Görsel 3.2). Elektronik malzemelerin breadboard üzerine yerleştirilmesinde dikkat edilmesi gereken hususlar şunlardır:

- Devre elemanının bacakları, kısa devre olmaması için aynı sütuna bağlanmaz.
- Uzun bacaklı elemanların yerleşimi sırasında kısa devre olmamasına dikkat edilir.
- Kutuplu olan devre elemanlarının (transistör, diyot, elektrolitik kondansatör vs.) bağlantı yönlerine dikkat edilmelidir.
- Entegre, display gibi devre elemanları breadbordun orta boşluk kısmına bağlanmalıdır.



✓ Doğru yerleşim



✗ Yanlış yerleşim

Görsel 3.2: Elektronik elemanların breadboard üzerine doğru ve yanlış yerleştirilmesi

3.1.2. Mikrodenetleyici Komutları

Arduino programlama dili, C programlama dilini referans alarak türetilmiştir. C programlama dilinin kendine özgü yazım kuralları ve temel özellikleri şu şekildedir:

- Program yazımı belirli kalıp ve bloklar hâlinde olur.
- Bloklar, " **{ }** " parantezleriyle oluşturulur.
- Program kodlarının satır sonları noktalı virgül (;) ile biter. Sadece blok başlatan ifadelerden sonra noktalı virgül kullanılmaz.
- Komutlar, aynı satıra veya alt alta satırlara yazılabilir. Aynı satıra yazıldıklarında aralarında noktalı virgül (**;**) işareti olmalıdır.
- Programda kullanılan tüm değişkenler ve veri türleri tanımlanır.
- Hazır kütüphaneler kullanılacaksa programın başına **#include** ifadesiyle eklenip aktifleştirilir.
- Program kodlarıyla ilgili tek satırlık açıklama yazılacaksa " ***//*** " işareti kullanılır. Bu durumda derleyici, bu işaretten sonrasını dikkate almaz.
- Birden fazla açıklama satırı kullanılacaksa açıklamanın yapılacağı yere " ***/**** " işareti, açıklamanın bittiği yere de " */ " işareti koyulur.

3.1.2.1 Değişkenler

Bir miktarı ifade etmek için kullanılan sembollerdir. Mikrodenetleyici hafızasında geçici bilgi saklamak için kullanılır. Karmaşık hafıza adres bilgileri yerine, programcı tarafından oluşturulan harf ve sembollerden oluştuğu için anlaşılması daha kolaydır.

Değişken kavramının daha iyi anlaşılması için mikrodenetleyiciye dışarıdan girilecek iki sayı toplamını bulan program için iki farklı örnek verilmiştir.

ÖRNEK

c=a+b Burada **a,b,c** birer değişken ismi olarak programcı tarafından verilmiştir. A=v+p veya afg=er+ty2 gibi rastgele harf ve sayılar da yazılabilir.

ÖRNEK

Toplam=Sayi1+Sayi2

Burada **Toplam, Sayi1, Sayi2** birer değişken ismi olarak programcı tarafından verilmiştir.

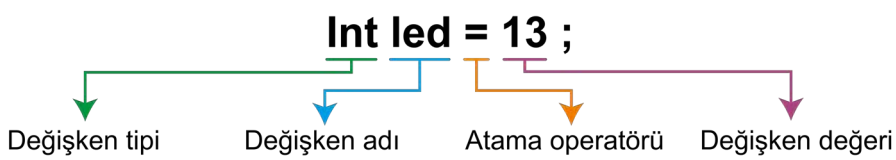
Her iki örnek de aynı işi yapar fakat 2. örneğin anlaşılması ve akılda kalması daha kolaydır. Program yazımında değişken isimlerini anlaşılır yapmak, programı incelerken veya hata takibinde büyük kolaylık sağlar.

Değişken tanımlamaları programın başında yapılırsa genel **(global)** değişken olarak tüm fonksiyonlarda geçerli olur. Fonksiyonların içinde tanımlanırsa **yerel** değişken olarak sadece o fonksiyon içinde geçerli olur. Değişken isimleri tanımlanırken tiplerinin de belirtilmesi gerekir. Tablo 3.1'de değişken türlerinin kapasitesi ve tipleri verilmiştir. Yapılacak işe özgü tip tanımlaması yapmak önemlidir. Örneğin iki sayının toplamını bulan programda **Toplam, Sayi1, Sayi2** değişkenleri byte cinsinden tanımlandığında alabilecekleri değer 255'i geçemez. Daha büyük sayılarla işlem yapmak için tip tanımlaması int, long veya double şeklinde olmalıdır. Eğer ondalıklı sayılarla ilgili işlemler varsa float şeklinde tanımlanmalıdır.

Tablo 3.1: Değişken Türleri

Tipi	Uzunluğu	Minimum Değeri	Maksimum Değeri	Anlamı
bool	1 bit	0 (False-Yanlış)	1 (True-Doğru)	Mantıksal değerler
char	8 bit	-128	127	Tek karakter
byte	8 bit	0	255	Doğal sayılar
int	16 bit	-32768	32767	Tam sayılar
unsigned int	16 bit	0	65535	Pozitif tam sayılar
word	16 bit	0	65535	Pozitif tam sayılar
long	32 bit	-2147483648	2147483647	Büyük tam sayılar
unsigned long	32 bit	0	2147483647	Büyük pozitif tam sayılar
Float	32 bit	-34028235E+38	34028235E+38	Ondalıklı sayılar
double	32 bit	-34028235E+38	34028235E+38	Ondalıklı sayılar
string	-	Bellek miktarı	Bellek miktarı	Karakter dizisi

Değişkenler tanımlanırken değişkenin tipi, değişkenin adı, varsa atama operatörü ve değişkenin değeri belirtilmelidir (Görsel 3.3).



Görsel 3.3: Örnek bir değişken tanımlama

Değişken Tanımlama Kuralları

- Değişken adlarında Türkçe karakter kullanılmaz (Sayı1 değişkenindeki ı harfi Türkçe'dir.). // açıklama işareti eklendiğinde programın bir parçası olmadığı için Türkçe karakterlerle açıklamalar yazılabilir.
- Değişken adlarında büyük ve küçük harf ayrımı vardır (Say1 ve say1 farklı değişkenlerdir.).
- Değişken adları boşluk içermez (Sayi 1 yanlış, Sayi1 doğru).
- Değişken adlarında noktalama işaretlerinden sadece alt çizgi (_) kullanılır (Sayi_1 gibi).
- Değişken adları sayı ile başlayamaz ama sayı içerebilir (1Sayi yanlış, Sayi1 doğru).
- Arduino fonksiyonları değişken adı olamaz (delay, loop gibi).

ÖRNEK

```
int deger  =  11 ;     // deger değişkeni int tipinde tanımlandı ve başlangıç değerine 11 atandı.
long Fark ;            // Fark değişkeni long tipinde tanımlandı.
```

3.1.2.2. Operatörler

Derleyiciye belirli matematiksel veya mantıksal işlevleri gerçekleştirmesini söyleyen sembollerdir. Operatör türleri şunlardır:

- Aritmetiksel operatörler (Tablo 3.2)
- Karşılaştırma operatörleri (Tablo 3.3)
- Mantıksal operatörler (Tablo 3.4)
- Bitsel operatörler (Tablo 3.5)
- Bileşik operatörler (Tablo 3.6)

Tablo 3.2: Aritmetiksel Operatörler

ARİTMETİKSEL OPERATÖRLER			
Sembolü	**Adı**	**Açıklama**	**Örnek**
=	**Atama**	Sağındaki değeri solundaki değişkene aktarır.	A=2; // (2'yi A'ya ata.)
+	**Toplama**	İki değeri toplar.	x=a+b; // (a ve b değişken değerlerini topla, x değişkenine ata.)
-	**Çıkarma**	İki değerin farkını alır.	y=x-2; // (x değişken değerinden 2 sabit sayısını çıkar, y değişkenine ata.)
*****	**Çarpma**	İki değeri çarpar.	D=c*d; // (a değişken değeriyle d değişken değerini çarp, D değişkenine ata.)
/	**Bölme**	İki değeri böler.	B=g/2; // (g değişken değerini 2'ye böl, B değişkenine ata.)
%	**Mod**	Bölümden kalanı bulur.	K=100 % 2; // (100 sabit sayısının 2'ye bölümünden kalanı K değişkenine ata.)

Tablo 3.3: Karşılaştırma Operatörleri

KARŞILAŞTIRMA OPERATÖRLERİ			
Sembolü	**Adı**	**Açıklama**	**Örnek**
==	**Eşit**	İki değerin birbirine eşit olup olmadığına bakar.	a=1,b=1,c=10; if(a==b){ c=1;} // Eğer a ve b değişken değerleri birbirine eşitse c değişken değerini 1 yap, değilse 10 kalsın (Örnekte c=1 olur.).
!=	**Eşit değil**	İki değerin birbirinden farklı olup olmadığına bakar.	a=1,b=1,c=10; if(a!=b){ c=1;} // Eğer a ve b değişken değerleri birbirinden farklıysa c değişken değerini 1 yap, değilse 10 kalsın (Örnekte c=10 olur.).
<	**Küçük**	Soldaki değerin sağdaki değerden küçük olup olmadığına bakar.	a=1,b=8,c=3; if(a<b){ c=1;} // Eğer a değişken değeri b değişken değerinden küçükse c değişken değerini 1 yap, değilse 3 kalsın (Örnekte c=1 olur.).
>	**Büyük**	Soldaki değerin sağdaki değerden büyük olup olmadığına bakar.	a=1,b=8,c=3; if(a>b){ c=1;} // Eğer a değişken değeri b değişken değerinden büyükse c değişken değerini 1 yap, değilse 3 kalsın (Örnekte c=3 olur.).
<=	**Küçük veya eşit**	Soldaki değerin sağdaki değerden küçük veya sağdaki değerle eşit olup olmadığına bakar.	a=1,b=8,c=3; if(a<=c){ c=1;} // Eğer a değişken değeri c değişken değerinden küçük veya c değişken değerine eşitse c değişken değerini 1 yap, değilse 3 kalsın (Örnekte c=1 olur.).
>=	**Büyük veya eşit**	Soldaki değerin sağdaki değerden büyük veya sağdaki değerle eşit olup olmadığına bakar.	a=12,b=11,c=10; if(a>=c){ c=1;} // Eğer a değişken değeri b değişken değerinden büyük veya b değişken değerine eşitse c değişken değerini 1 yap, değilse 10 kalsın (Örnekte c=1 olur.).

Tablo 3.4: Mantıksal Operatörler

MANTIKSAL OPERATÖRLER			
Sembolü	**Adı**	**Açıklama**	**Örnek**
&&	**Lojik VE**	İki değeri **Lojik VE** işlemine tabi tutar.	a=1,b=1,c=10; if((a==b) && (c>a)) { c=1;} // Eğer a ve b değişken değerleri birbirine eşit, **ve** c değişken değeri a değişken değerinden büyükse c değişken değerini 1 yap, değilse 10 kalsın (Örnekte c=1 olur.).
||	**Lojik VEYA**	İki değeri **Lojik VEYA** işlemine tabi tutar.	a=1,b=12,c=9; if((a==c) || (c>=b)) { c=5;} // Eğer a ve c değişken değerleri birbirine eşit, **veya** c değişken değeri b değişken değerinden büyük veya eşitse c değişken değerini 5 yap, değilse 9 kalsın (Örnekte c=9 olur.).
!	**Lojik DEĞİL**	İşlenen değeri **Lojik DEĞİL** (tersini alma) işlemine tabi tutar.	a=12,b=3,c=12; if(!(a==c)) { c=5;} // Eğer a ve c değişken değerleri birbirine eşit **değilse**, c değişken değerini 5 yap, değilse 12 kalsın (Örnekte c=12 olur.).

Tablo 3.5: Bitsel Operatörler

BİTSEL OPERATÖRLER			
Sembolü	**Adı**	**Açıklama**	**Örnek**
&	**Lojik VE**	İki değerin bitlerini **lojik VE** işlemine tabi tutar. c=a & b $a=(00001111)_2=(15)_{10}$ $b=(10001110)_2=(142)_{10}$ $c=(00001110)_2=(14)_{10}$	a=15,b=142; c=a & b; // a ve b değişken değerlerinin bitlerini lojik **ve** işlemine sokar ve sonucu c değişkenine atar (Örnekte c=14 olur.). Bitlerden herhangi biri lojik 0 ise sonuç lojik 0, değilse lojik 1 olur.
|	**Lojik VEYA**	İki değerin bitlerini **lojik VEYA** işlemine tabi tutar. c=a & b $a=(00001111)_2=(15)_{10}$ $b=(11001110)_2=(206)_{10}$ $c=(11001111)_2=(207)_{10}$	a=15,b=206; c=a | b; // a ve b değişken değerlerinin bitlerini lojik **veya** işlemine sokar ve sonucu c değişkenine atar (Örnekte c=207 olur.). Bitlerden herhangi biri lojik 1 ise sonuç lojik 1, her ikisi de lojik 0 ise sonuç lojik 0 olur.
^	**Lojik ÖZEL VEYA**	İki değerin bitlerini **lojik ÖZEL VEYA** işlemine tabi tutar. c=a & b $a=(01001011)_2=(75)_{10}$ $b=(10101110)_2=(174)_{10}$ $c=(11100101)_2=(229)_{10}$	a=75,b=174; c=a ^ b; // a ve b değişken değerlerinin bitlerini lojik **özel veya** işlemine sokar ve sonucu c değişkenine atar (Örnekte c=229 olur.). Bitlerin her ikisi de eşit olursa sonuç lojik 0, değilse lojik 1 olur.
~	**Lojik DEĞİL**	İşlenen değerin bitlerini **lojik DEĞİL** işlemine tabi tutar. c=~a $a=(01001011)_2=(75)_{10}$ $c=(10110100)_2=(180)_{10}$	a=75; c= ~ a; // a değişken değerinin bitlerini lojik **değil** işlemine sokar ve sonucu c değişkenine atar (Örnekte c=180 olur.). Bitlerin tersini alır. Lojik 0 olan biti lojik 1, lojik 1 olan biti lojik 0 yapar.
<<	**Sola kaydırma**	İşlenen değeri sağdan **sola** doğru belirlenen değer kadar kaydırır. c=a<<3 $a=(11010010)_2=(210)_{10}$ $c=(10010000)_2=(144)_{10}$	a=210; c= a<<3; // a değişken değerinin bitlerini 3 defa **sola** doğru kaydırır ve sonucu c değişkenine atar (Örnekte c=144 olur.). Sağda boşalan bitlerin yerine lojik 0 atar.

Tablo 3.5: Bitsel Operatörler

BİTSEL OPERATÖRLER			
Sembolü	**Adı**	**Açıklama**	**Örnek**
>>	**Sağa kaydırma**	İşlenen değeri soldan sağa doğru belirlenen değer kadar kaydırır. c=a>>1 $a=(11010010)_2=(210)_{10}$ $c=(01101001)_2=(105)_{10}$	a=210; c= a>>1 ; // a değişken değerinin bitlerini 1 defa **sağa** doğru kaydırır ve sonucu c değişkenine atar (Örnekte c=105 olur.). Solda boşalan bitlerin yerine lojik 0 atar.

Tablo 3.6: Bileşik Operatörler

BİLEŞİK OPERATÖRLER			
Sembolü	**Adı**	**Açıklama**	**Örnek**

++	**Arttırma**	Bağlı bulunduğu değişkenin değerini bir artırır.
		Değişkenin sağında yer alırsa ilk önce değişkenin değeri atanır sonra artırılır.
		Değişkenin solunda yer alırsa ilk önce değişkenin değeri artırılır sonra atanır.
			a=10;
			c=a++; // a değişken değerini c değişkenine ata sonra a değerini bir artır.
			d=++a ; // a değişken değerini bir artır sonra d değişkenine ata.
			
			a++ ifadesi a=a+1 anlamındadır.
			(Örnekte c=10 d=12 olur.)

--	**Azaltma**	Bağlı bulunduğu değişkenin değerini bir azaltır.
		Değişkenin sağında yer alırsa ilk önce değişkenin değeri atanır sonra azaltılır.
		Değişkenin solunda yer alırsa ilk önce değişkenin değeri azaltılır sonra atanır.
			a=10;
			c=a-- ; // a değişken değerini c değişkenine ata sonra a değerini bir azalt.
			d=--a ; // a değişken değerini bir azalt sonra d değişkenine ata.
			a-- ifadesi a=a-1 anlamındadır.
			(Örnekte c=10 d=8 olur.)

+=	**Bileşik toplama**	Sağda bulunan değişken değerini solda bulunan değişken değerine ekler ve sonucu sol değişkene atar.	a=10, c=12, d=7;
			c+=a ; // c değişken değerine a değişken değerini ekle sonra c değişkenine ata.
			d+=10 ; // d değişken değerine 10 sabit sayısını ekle sonra d değişkenine ata.
			c+=a ifadesi c=c+a anlamındadır.
			(Örnekte c=22, d=17 olur.)

-=	**Bileşik çıkarma**	Sağda bulunan değişken değerinden solda bulunan değişken değerini çıkarır ve sonucu sol değişkene atar.	a=10, c=12, d=25;
			c-=a ; // c değişken değerinden a değişken değerini çıkar sonra c değişkenine ata.
			d-=10 ; // d değişken değerinden 10 sabit sayısını çıkar sonra d değişkenine ata.
			c-=a ifadesi c=c-a anlamındadır.
			(Örnekte c=2, d=15 olur.)

***=**	**Bileşik çarpma**	Sağda bulunan değişken değerini solda bulunan değişken değeriyle çarpar ve sonucu sol değişkene atar.	a=10, c=12, d=25;
			c*=a ; // c değişken değerini a değişken değeriyle çarp sonra c değişkenine ata.
			d*=10 ; // d değişken değerini 10 sabit sayısıyla çarp sonra d değişkenine ata.
			c*=a ifadesi c= c * a anlamındadır.
			(Örnekte c=120, d=250 olur.)

Tablo 3.6: Bileşik Operatörler

BİLEŞİK OPERATÖRLER			
Sembolü	**Adı**	**Açıklama**	**Örnek**

/=	**Bileşik bölme**	Solda bulunan değişken değerini sağda bulunan değişken değerine böler ve sonucu sol değişkene atar.	a=5, c=40, d=50;
			c/=a ; // c değişken değerini a değişken değerine böl sonra c değişkenine ata.
			d/=25 ; // d değişken değerini 25 sabit sayısına böl sonra d değişkenine ata.
			
			c/=a ifadesi c= c / a anlamındadır.
			(Örnekte c=8, d=2 olur.)

%=	**Bileşik mod alma**	Solda bulunan değişken değerinin sağda bulunan değişken değerine bölümünden kalanı sol değişkene atar.	a=5, c=39;
			c%=a ; // c değişken değerinin a değişken değerine bölümünden kalanı c değişkenine ata.
			
			c%=a ifadesi c=c % a anlamındadır.
			(Örnekte c=4 olur.)

3.1.2.3. Fonksiyonlar

Programlama dillerinde bir veya daha çok eylemi gerçekleştirmek için yazılan ve ana programa parametre ve fonksiyon dönüş değeriyle bağlı olan alt programdır. Genellikle sık yapılan işlemler fonksiyon hâline getirilerek kullanılır. Fonksiyonlar yapılırken "fonksiyonun ismi (varsa paramatreler)" şeklinde tanımlanır.

Geri değer döndüren fonksiyonlar: Programda parametre alıp bu değerleri işleyen ve sonucu geri döndüren fonksiyonlardır. Aşağıda verilen program örneğinde Topla isminde fonksiyon yazılmış ve parantez içinde iki parametre alacağı bildirilmiştir. Bu parametrelerle bir toplama işlemi yaptırılmış, return komutu ile sonuç gönderilmiştir.

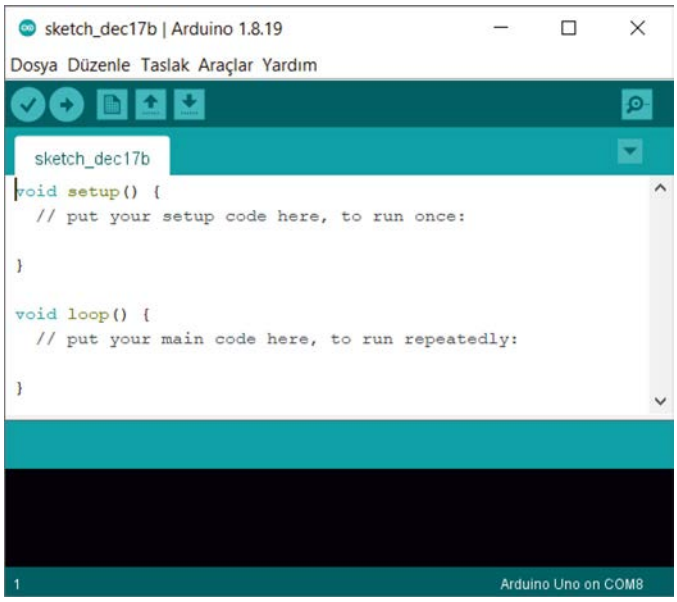
Geri döndüren fonksiyonlar bir değişkene atanarak kullanılır (Sonuc=Topla(a,b)). Burada Sonuc bir değişken, Topla(a,b) ise geri değer döndürecek fonksiyonun ismidir.

Geri değer döndürmeyen fonksiyonlar: Ana programa herhangi bir değer döndürmeyen fonksiyonlardır. Aşağıda verilen program örneğinde void PCgonder() isminde fonksiyon yazılmış ve parantez içinde herhangi bir değer yazılmamıştır. Fonksiyonun başında yazılan "void" ifadesi, bu fonksiyonun değer göndermeyecek boş bir fonksiyon olduğu anlamına gelir. Bu fonksiyon bilgisayara seri port üzerinden toplam sonucunu gönderir. Ana programa herhangi bir değer göndermez.

ÖRNEK

```
int Sonuc,a=2, b=3;          // Fonksiyon örneği
void setup(){
Serial.begin(9600); }        // Seri portu 9600 baud hız için ayarla.
void loop() {
    Sonuc = Topla(a,b);      // 2 ve 3 parametresini alıp değer döndüren fonksiyon.
    PCgonder();              // Değer döndürmeyen fonksiyon.
    delay(500); }            // 500 ms bekler.
int Topla(int x,int y) {     // Geri değer döndüren Topla fonksiyonu.
    int Hesap;
    Hesap = x + y ;          // Kendisine gelen iki değeri toplar, Hesap
    return Hesap ; }         // değişkenine atar ve return ile geri döndürür.
void PCgonder () {           // Geri değer döndürmeyen fonksiyon.
    Serial.print("Toplam="); // Sadece fonksiyon içinde yapılması istenen
    Serial.println(Sonuc); } // işlemleri yapar ve ana programa geri döner.
```

C dilinde komutlar genellikle fonksiyonlar şeklindedir. Yeni bir arduino programı yazılmaya başlandığında editör programı otomatik olarak iki boş fonksiyon oluşturur (Görsel 3.4).



Görsel 3.4: Yeni bir arduino programı başlangıç ekranı

void setup();

Bu fonksiyon, program başladığında sadece bir kez çalışır. Kullanılacak arduino karta ait ayarlama ve tanımlamaların yapıldığı kısımdır. Örneğin kullanılacak pin bağlantılarının giriş veya çıkış olarak ayarlanması, seri port kullanılacaksa haberleşme ayarları vs. bu kısımda yapılır.

ÖRNEK

```
void setup()
{
    pinMode(13, OUTPUT);          // 13 numaralı pin çıkış olarak tanımlanır.
    Serial.begin(115200);         // Seri port 115200 hızına ayarlanır.
}
```

void loop();

Asıl yapılması istenilen program kodları bu fonksiyon içine yazılır. Program kodları "**{}**" parantezleri arasında satır satır, sürekli işlenir. Sensör ölçümü, veri alışverişi, pini aktif ve pasif duruma getirme gibi kodlar içerir.

Örneklerde **void setup()** fonksiyonuyla 13 numaralı pin çıkış olarak ayarlanır. void loop() fonksiyonunda 13 numaralı pin, 1 saniye aralıklarla durumunu lojik 1 ve lojik 0 olarak değiştirir. Sürekli bir döngü olduğu için aduino enerjisi kesilince veya herhangi bir döngü kesme işlemi yapılınca bu sonsuz döngüden çıkar.

ÖRNEK

```
void loop()
{
    digitalWrite(13, HIGH);       // 13 numaralı pin çıkışı lojik 1 yapılır.
    delay(1000);                  // 1000 ms (1000 milisaniye ya da 1 saniye) beklenir.
    digitalWrite(13, LOW);        // 13 numaralı pin çıkışı lojik 0 yapılır.
    delay(1000);                  // 1000 ms (1 s) beklenir.
}
```

Tablo 3.7'de arduino programlarında en çok kullanılan fonksiyonlar yer almıştır.

Tablo 3.7: En Çok Kullanılan Arduino Fonksiyonları

Adı	Açıklama

pinMode(Pin, Durum);	Dijital bir pinin giriş veya çıkış olarak ayarlanması için kullanılır. Setup fonksiyonu içine yazılır.
	pinMode(12, OUTPUT); // 12 no. lu pini çıkış ayarla.
	pinMode(13, INPUT); // 13 no. lu pini giriş ayarla.

digitalWrite(Pin, Durum);	Dijital bir pin çıkış olarak ayarlandığında bu pine HIGH (+5 V) veya LOW (0 V) atamak için kullanılır. HIGH yerine 1, LOW yerine 0 olarak da kullanılabilir.
	digitalWrite(2, HIGH); // 2 no.lu pine lojik 1 (+5 V) gönder.
	digitalWrite(3, LOW); // 3 no.lu pine lojik 0 (0 V) gönder.

Değişken=digitalRead(Pin);	Dijital giriş olarak ayarlanmış bir pinin değerini okur. Dijital girişten +5 V gelirse 1, 0 V gelirse 0 değerini üretir. But= digitalRead(5); // 5 no.lu pinin durumunu oku ve But değişkenine ata.
Değişken=analogRead(AnalogPin);	Analog giriş olarak ayarlanmış bir pinin değerini okur ve bağlı bulunduğu değişkene atar. Sns= digitalRead(A0); // A0 no.lu pindeki analog gerilimi oku ve Sns değişkenine ata.
analogWrite(Pin, analog değer);	Analog çıkış (PWM) olarak ayarlanmış bir pinden, 0-255 arası analog değere karşı 0-5 V analog gerilim değeri alınır. analogWrite(3,127); // 3 no.lu analog çıkıştan 127 değeri kadar (Yaklaşık 2,5 V) PWM üret.
delay(Süre);	Mikrodenetleyici, parantez içinde yazan değer kadar milisaniye cinsinden bekler. delay(1500); // 1500 ms (1,5 saniye) bekleme yapar.
delayMicroseconds(Süre);	Mikrodenetleyici, parantez içinde yazan değer kadar mikrosaniye cinsinden bekler. delayMicroseconds(3000); // 3000 µs (3 ms) bekleme yapar.
Değişken=map(x,a,b,c,d);	Herhangi bir değer aralığını başka bir değer aralığına çevirir. Örneğin bir sensörün ürettiği analog değer 0-1023 aralığında olsun. Bu değeri 0-100 arasına çevirmek için; Deger=map(Sensor,0,1023,0,100); //0-1023 verisini map ile 0-100' e çevir.
Serial.begin(Baud hızı);	Mikrodenetleyicinin dijital olarak diğer cihazlarla seri port üzerinden haberleşme hız ayarıdır. Haberleşme yapacak her iki cihazın da aynı hıza ayarlanması gerekir. Baud hızı, saniyede gönderilen **bit** sayısıdır. Setup fonksiyonuna yazılır. Serial.begin(9600); // Haberleşme hızını 9600 baud ayarla.
Serial.print("karakterler");	Seri port üzerinden tırnak işareti (" ") içinde yer alan karakterleri olduğu gibi gönderir. Serial.print("Deneme"); // Deneme kelimesini seri porttan gönder.

Adı	Açıklama
Serial.print(Değişken);	Seri port üzerinden parantez işareti ( ) içinde yer alan değişken değerini gönderir. Serial.print(Sns); // Sns değişkeninin değerini seri porttan gönder.
Serial.println();	Seri port üzerinden gönderilecek verinin bir alt ve satır başı yazılmasını sağlar (Klavyeden enter tuşuna basılması gibi). Bu işlevi fonksiyonun sonunda kullanılan ln ifadesi gerçekleştirir.

3.1.2.4. Karar Verme ve Kontrol Komutları

Verilen şarta göre karar verip kontrol işlemini gerçekleştirir. Örneğin bir fırının sıcaklığının kontrol edilmesi istendiğinde sıcaklık sensöründen gelen verinin, ayarlanan sıcaklığa ulaşıp ulaşmadığı kontrol edilir. Eğer ulaştıysa fırın kapatılır. Arduino programlama dilinde en çok kullanılan karar verme komutları **if** ve **if else**'dir.

IF KOMUTUNUN YAZIM ŞEKLİ

```
if (şart)
{
    Şart gerçekleştiğinde yapılacak işlemler.
}
```

ÖRNEK

```
void setup()
{
    pinMode(2,INPUT);              // 2 no.lu pini giriş olarak ayarla.
    pinMode(10,OUTPUT);            // 10 no.lu pini çıkış olarak ayarla.
}
void loop()
{
    if (digitalRead(2)==1) {       // Eğer 2 no.lu pin lojik 1 ise
        digitalWrite(10,1);  }     // 10 no.lu pini lojik 1 yap.

}
```

IF ELSE KOMUTUNUN YAZIM ŞEKLİ

```
if (şart)
{
Şart gerçekleştiğinde yapılacak işlemler.
}
else
{
Şart gerçekleşmediğinde yapılacak işlemler.
}
```

ÖRNEK

```
void setup() {
pinMode(2,INPUT);                    // 2 no.lu pini giriş olarak ayarla.
pinMode(10,OUTPUT); }                // 10 no.lu pini çıkış olarak ayarla.

void loop()
{
  if (digitalRead(2)==HIGH) {        // Eğer 2 no.lu pin lojik 1 ise
     digitalWrite(10,HIGH); }        // 10 no.lu pini lojik 1 yap.
else {                               // Değilse
    digitalWrite(10,LOW); }          // 10 no.lu pini lojik 0 yap.
}
```

3.1.2.5. Döngü Komutları

Döngüler, şart sağlanıncaya kadar içerdiği fonksiyonları tekrar tekrar çalıştıran yapılardır. Şartsız yapılan döngülere **sonsuz döngüler** denir. Bu döngülerin içine yazılan fonksiyonlar sürekli satır satır işlenir. Arduino ana programının yazıldığı void loop() fonksiyonu da sonsuz bir döngü örneğidir. En çok kullanılan döngü komutları **for** ve **while**'dır. İstenildiğinde **break**; komutuyla döngülerden çıkılabilir.

FOR DÖNGÜSÜ ÖRNEĞİ

SATIR	KODLAR
1	int say; // say değişkenini int olarak tanımla.
2	void setup()
3	{
4	pinMode(2, OUTPUT); // 2 no.lu pini çıkış olarak ayarla.
5	pinMode(3, OUTPUT); // 3 no.lu pini çıkış olarak ayarla.
6	}
7	void loop()
8	{
9	digitalWrite(3, LOW); // 3 no.lu pini lojik 0 yap.
10	for (say=0;say<3;say++) { // say değişkeni 3'ten küçük olana kadar bir artır ve alt satırları tekrarla.
11	digitalWrite(2,HIGH); // 2 no.lu pini lojik 1 yap.
12	delay(500); // 500 ms bekle.
13	digitalWrite(2,LOW); // 2 no.lu pini lojik 0 yap.
14	delay(500); } // 500 ms bekle
15	// say değişkeni 3'ten küçükse 10. satıra geri dön (for döngüsü).
16	digitalWrite(3,HIGH); // 3 no.lu pini lojik 1 yap.
17	} // 8. satıra geri dön.

For döngüsü örneğinde, 10. satıra kadar program normal çalışır. 10. satırda for döngü komutu içine yazılan say=0'dan başlamak üzere şartlar kontrol edilir. Say 3'ten küçükse say bir artırılır (say=1 olur.). Sonra 11,12,13,14. satırlar işlenir ve 15. satıra gelindiğinde tekrar 10. satıra gidilir. For döngüsü şartı kontrol edilir. Bu durum, say değişkeni 3'e eşit oluncaya kadar devam eder. Say değişkeni 3 olduğunda şartın dışına çıkılmış olur. Döngü sona erer ve 16.satır işlenir.17. satıra gelindiğinde program 8. satıra gider ve yeniden başlar.

WHILE DÖNGÜSÜ ÖRNEĞİ

SATIR	KODLAR	
1	int say=0;	// say değişkenini int olarak tanımla.
2	void setup()	
3	{	
4	pinMode(2, OUTPUT);	// 2 no.lu pini çıkış olarak ayarla.
5	pinMode(3, OUTPUT);	// 3 no.lu pini çıkış olarak ayarla.
6	}	
7	void loop()	
8	{	
9	digitalWrite(3, LOW);	// 3 no.lu pini lojik 0 yap.
10	while(say<3){	//say değişkeni 3'ten küçükse while çalışsın.
11	digitalWrite(2,HIGH);	// 2 no.lu pini lojik 1 yap.
12	delay(500);	// 500 ms bekle.
13	digitalWrite(2,LOW);	// 2 no.lu pini lojik 0 yap.
14	delay(500);	// 500 ms bekle.
15	say++; }	// say değişkenini 1 artır.
16	// say değişkeni 3'ten küçükse 10. satıra geri dön (while döngüsü).	
17	digitalWrite(3,HIGH);	// 3 no.lu pini lojik 1 yap.
18	}	// 8. satıra geri dön.

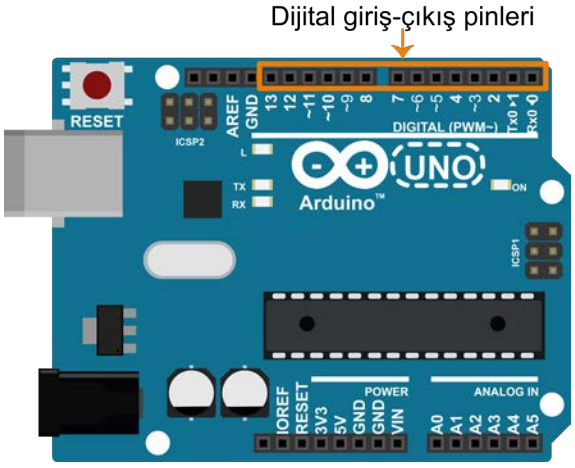
While döngüsü örneğinde, 10. satıra kadar program normal çalışır. 10. satırda while döngü fonksiyonu içinde bulunan kontrol şartı sağlandığı sürece 11,12,13,14,15. satırda yer alan program kodları işlenir ve 10. satıra gider. Say değişkeni 3 olduğunda şartın dışına çıkılmış olur. Döngü sona erer ve 17. satır işlenir. 18. satıra gelindiğinde program 7. satıra gider ve yeniden başlar.

3.2. DİJİTAL GİRİŞ-ÇIKIŞ

Mikrodenetleyicilerde dijital işlemler lojik 0 ve lojik 1'in yani 0 V ile +5 V'luk iki seçeneğin olduğu işlemlerdir.

3.2.1. Dijital Giriş-Çıkış Kavramları

Giriş işlemleri arduino pinlerinin haricî bir devreden (sensör, buton vs.) 0 V veya +5 V almasıdır. Çıkış işlemleriyse arduino pinlerinden haricî bir devreye (LED, röle kontrol vs.) 0 V veya +5 V vermesidir. Arduino mikrodenetleyici kartın dijital giriş-çıkış pinleri 0-13 numara arasıdır (Görsel 3.5). A0-A5 arası pinler analog girişler olmasına rağmen dijital giriş-çıkış olarak da kullanılabilir.



Görsel 3.5: Arduino dijital giriş-çıkış pinleri

Arduino'da dijital işlemler için kullanılan üç temel fonksiyon vardır. Bu fonksiyonlar şunlardır:

pinMode(Pin, Durum)

digitalWrite(Pin, Durum)

Değişken=digitalRead(Pin)

ÖRNEK

```
void setup()
{
pinMode(2,INPUT);              // 2 no.lu pini giriş olarak ayarla.
pinMode(3,OUTPUT);             // 3 no.lu pini çıkış olarak ayarla.
}
void loop()
{
  if (digitalRead(2)==0){      // Eğer 2 no.lu pin lojik 0 ise
     digitalWrite(3,1);        // 3 no.lu pini lojik 1 yap.
  }
}
```

3.3. MİKRODENETLEYİCİLERDE PORT KAVRAMI

Mikrodenetleyicilerin dış ortamdan veri aldıkları veya dış ortama veri gönderdikleri giriş-çıkış pinlerine **port** denir.

3.3.1. Bit, Byte, Word, LSB, MSB

Mikrodenetleyiciler ikili sayı sistemini (binary) kullanır. Lojik 0 ve lojik 1 şeklinde işlemler yapar. En küçük bilgi birimi olan bu 0 ve 1'lerin her birine **bit** denir (Görsel 3.6).

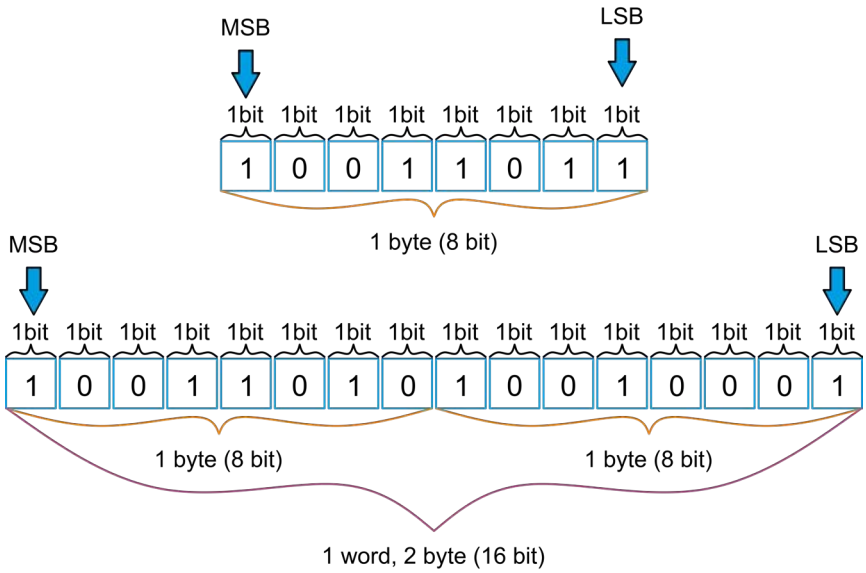
Byte: 8 bitin yan yana gelmesiyle oluşur. 8 bit 28 farklı değer alır (Değer aralığı 0-255 arasındadır.).

Word: 16 bitin yan yana gelmesiyle oluşur. 16 bit 216 farklı değer alır (Değer aralığı 0-65535 arasındadır.).

LSB (**L**east **S**ignificant **B**it)**:** En değerliksiz bit anlamındadır. Bir sayının değerini en az etkileyen bittir.

MSB (**M**ost **S**ignificant **B**it)**:** En değerlikli bit anlamındadır. Bir sayının değerini en çok etkileyen bittir.

8 bitlik bir sayının en sağındaki LSB biti 1 arttığında sayı sadece 1 değer büyür veya küçülür. En solundaki MSB biti 1 arttığındaysa sayı 128 değer birden büyür veya küçülür.



Görsel 3.6: Bit, byte, Word, LSB, MSB gösterimi

Örneğin bir değişken değeri byte olarak tanımlandığında ve değeri 120 olarak verildiğinde mikrodenetleyici tarafından, 120 sayısının binary karşılığı $(01111000)_2$ şeklinde işlenir. Byte'ın alabileceği değer en çok 255 olur. Bu değerden büyük sayılar için daha büyük değer aralığına sahip değişken tanımı yapılmalıdır. Aynı değişken unsigned int olarak tanımlandığında ve değeri 32698 verildiğinde 2 byte yani word olarak tanımlanmış olur. Bu sefer alabileceği değer en çok 65535 olur. Mikrodenetleyici tarafından, 32698 sayısının binary karşılığı $(0111111110111010)_2$ şeklinde işlenir.

3.4. ANALOG GİRİŞ-ÇIKIŞ

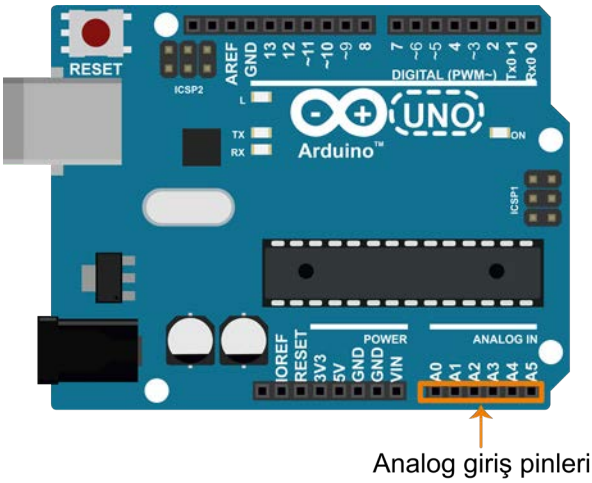
Günlük hayatta karşılaşılan her etki, bir analog sinyale karşılık gelir ve bu etkiyi ölçen sensörler analog çıkış verir.

3.4.1. Analog Giriş

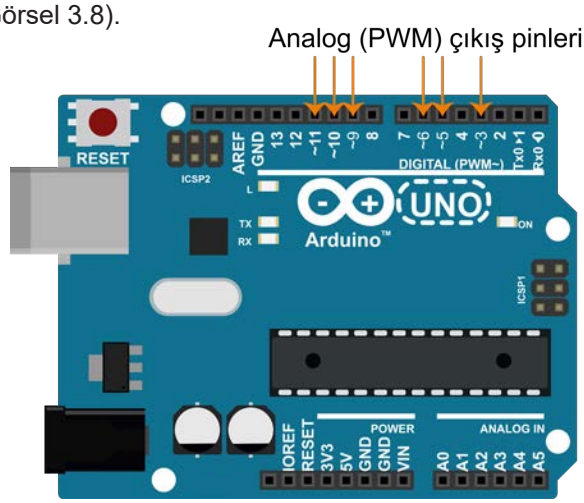
Sıcaklık değerini ölçen sensörler sıcaklıkla değişen gerilim veya akım üretir. Buna benzer değerleri değişen sinyallere kısaca **analog sinyaller** denir. Bu tip sinyaller mikrodenetleyicilerde analog girişlere bağlanıp değerlendirilir. Arduino kartta A0-A5 pinleri analog girişlerdir. İstenirse A0-A5 pinleri dijital giriş-çıkış olarak da ayarlanabilir (Görsel 3.7).

3.4.2. Analog Çıkış

Mikrodenetleyicinin dış ortama değişken gerilim verebilen çıkış pinleridir. Arduino kartta 3-5-6-9-10-11 pinleri analog çıkış (PWM) olarak kullanılabilir (Görsel 3.8).



Görsel 3.7: Arduino analog giriş pinleri

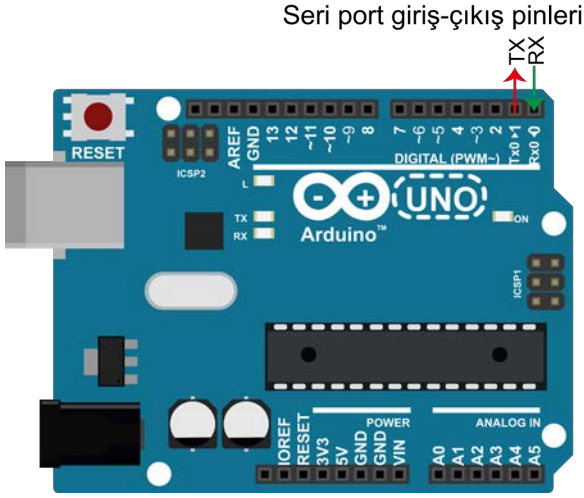


Görsel 3.8: Arduino analog çıkış pinleri

3.4.3. Seri Port Ekranı (Serial Monitor)

Arduino kartıyla bilgisayar veya diğer cihazların haberleşmesinde seri port kullanılır. Mikrodenetleyici geliştirme kartlarında farklı sayıda seri haberleşme portları bulunur. Arduino Uno kartında bir adet seri port vardır. Bunlar dijital 0 (**RX**) ve dijital 1 (**TX**) pinleridir (Görsel 3.9).

Eğer arduino kart, bağlı bulunduğu bilgisayara seri port üzerinden veri gönderecekse herhangi bir donanım değişikliği olmadan seri port ekranı açılabilir ve değerler seri port ekranı üzerinden izlenebilir. Bunun için arduino IDE programında sağ üst köşedeki **Seri Port Ekranı** ikonuna tıklanmalı veya "**Araçlar> Seri Port Ekranı**" kısmı açılmalıdır.



Görsel 3.9: Arduino seri port giriş-çıkış pinleri

Bu işlem yapıldıktan sonra Görsel 3.10'daki gibi seri port ekranı görülecektir. Seri port kullanımına dair ayrıntılı bilgiler uygulamalarda detaylı şekilde verilmiştir.



Görsel 3.10: Seri port ekranı

3.5. ADC-DAC UYGULAMALARI

Mikroişlemciler analog sinyalleri doğrudan işleyemez. Öncelikle bu sinyallerin sayısal (dijital) değerlere dönüştürülmesi gerekir. Bunun için analog dijital çevirici ADC (Analog Digital Converter) kullanılır.

3.5.1. ADC (Analog Dijital Çevirici)

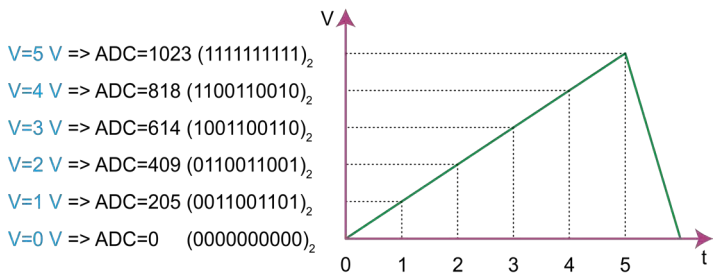
Birçok mikrodenetleyici içinde ADC modülü bulunmaktadır. Bu modül, yazılım yardımıyla analog değerleri sayısal değerlere dönüştürür. Arduino kartta A0-A5 pinlerine, analog çıkış veren (çıkışları arduino beslemesini geçmeyen) her türlü sensör ve cihaz bağlanabilir.

ADC Çevrim Kontrolü

ADC çevrimi bir referansa göre belirlenir. Yani ölçülecek analog sinyalin karşılaştırılacağı değerdir. Arduino standart referans gerilimi Vref =+5 V'tur. İstenildiğinde dâhilî veya haricî olarak farklı referans gerilimi ayarlanabilir.

ADC kullanımında ne kadar hassas ölçüm yapılırsa sinyaller o kadar detaylı işlenir. Bu duruma **çözünürlük** denir ve değeri bit şeklinde belirtilir (8 bit, 10 bit, 12 bit vs). Arduino Uno kartta 10 bit ADC çözünürlüğü mevcuttur. 10 bit çözünürlüğe sahip bir ADC, ölçülen analog sinyali 210 =1024 (0-1023) farklı değer olarak ölçebilir. Örneğin 0-100 0C ölçüme karşılık 0-5 V çıkış veren bir sıcaklık sensörü, 0 ^{0}C (0 V) = 0 ADC çevrim değeri, 50 0C (2,5 V) = 512 ADC çevrim değeri, 100 ^{0}C (5 V) = 1023 ADC çevrim değerini alır. Bu ADC değerleri gerekli hesaplamalarla gerilim, sıcaklık, basınç vs. şeklinde ekranda gösterilebilir veya diğer kontrol işlemleri için kullanılabilir.

0-5 V çıkışlı bir analog sinyalin ADC çevrim kontrol grafiği Görsel 3.11'de verilmiştir. Grafikte, her bir gerilim değerine karşılık oluşan ADC çevrimi ve 10 bit binary karşılığı görülmektedir.



Görsel 3.11: ADC çevrim kontrolü

ADC için aşağıdaki fonksiyon kullanılır.

Degişken = analogRead**(ADC kanalı);**

ADC kanalı: A0-A5 analog giriş pinlerinden birini belirtir. Bu pin giriş olarak tanımlanmalıdır.

Degişken: Okunacak ADC değerinin atanacağı değişkendir.

ADC değerini gerilim değerine çevirmek için yapılması gereken hesaplama şu şekildedir:

Gerilim=(Vref/1023) x ADC değeri, Vref Arduino için 5 V alınmıştır. 5 V / 1023 =0,00488 V. bölme işlemi sabit bir değer olduğu için her bir ADC değeri 0,00488 V. olacak şekilde kullanılabilir.

Buna göre çıkış gerilim değeri = 0,00488 x ADC değeri şeklinde bulunur.

3.5.2. DAC (Dijital Analog Çevirici)

Birçok mikrodenetleyicide analog giriş standart olarak bulunurken, analog çıkış daha gelişmiş mikrodenetleyicilerde bulunur. Bazı uygulamalarda analog çıkış sinyaline ihtiyaç duyulur. Bunun için haricî **DAC** (**D**igital **A**nalog **C**onverter / Dijital analog çevirici) entegreleri veya çoğu mikrodenetleyicilerde standart olarak bulunan PWM modülü kullanılabilir.

PWM (**P**ulse **W**idht **M**odulation / Darbe genişlik modülasyonu): Dijital bir sinyaldir. Belirli bir periyotta lojik 1 süresini (Görsel 3.12'deki gibi) artırıp azaltarak ortalama çıkış gerilimini değiştirme işlemidir. PWM çıkışları kare dalga şeklinde olduğu için anahtarlama elemanlarındaki kayıplar çok azdır. İnvertör, konvertör, anahtarlamalı güç kaynakları, DC ve AC motor sürücülerinde (hız, tork ayarı vs.) sıklıkla kullanılır.

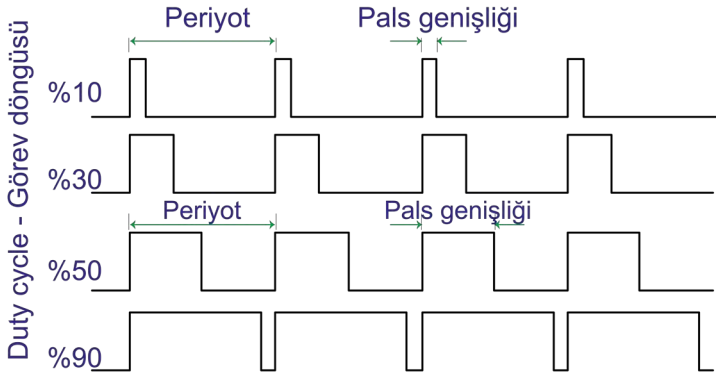
Duty cycle (Görev döngüsü): Bir periyotta sinyalin lojik 1'de kalma süresinin periyoda oranı şeklinde yüzdesel gösterimdir. Örneğin besleme gerilimi +5 V ve periyodu 100 µs olan %10 duty cycle sinyal için, 10 µs lojik 1 (+5 V), 90 µs lojik 0 (0 V) olacaktır. Bu durumda VDAC çıkışından 5 V'un %10'u 0,5 V ölçülür. Duty cycle %50 yapıldığındaysa 2,5 V ölçülür.

Çözünürlük: Daha küçük gerilim değişimlerini elde etmek için kullanılan terimdir. Diğer bir ifadeyle maksimum çıkış gerilimine kaç adımda ulaşılacağının bir ölçüsüdür. 8 bit çözünürlüğe sahip mikrodenetleyiciler maksimum çıkış gerilimine 255, 10 bit çözünürlüğe sahip mikrodenetleyiciler ise 1023 adımda ulaşabilir.

PWM için aşağıdaki fonksiyon kullanılır.

analogWrite(**PWM pin**, **PWM değeri**);

PWM pin: PWM çıkış sinyalinin alınacağı pinlerden birini belirtir. Bu pin çıkış olarak tanımlanmalıdır.



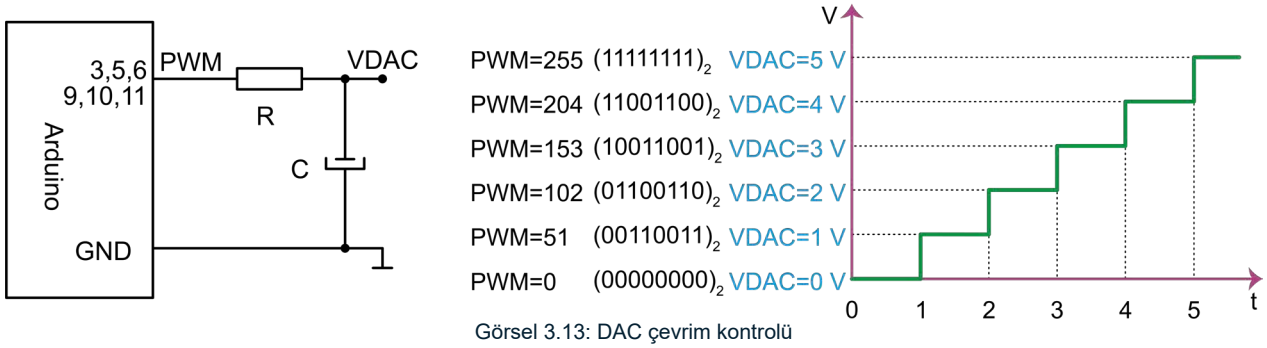
Görsel 3.12: PWM dalga şekli

Arduino kartta 3,5,6,9,10,11 numaralı pinler PWM olarak çıkış verebilir.

PWM değeri: Arduino Uno için 8 bittir (0-255 arası). Görev döngü yüzdesi, 0-255 PWM değerine karşılık %0-%100 olacaktır. Yani bu değer 127 verilirse görev döngü yüzdesi %50, 255 verilirse %100 olacaktır.

DAC Çevrim Kontrolü

PWM sinyal çıkışına RC filtre devresi bağlanarak dijital değerlere göre çıkış veren DAC elde edilebilir. Görsel 3.13'te her bir gerilim değerine karşılık oluşan PWM, VDAC çevrimi ve 8 bit binary karşılığı görülmektedir.

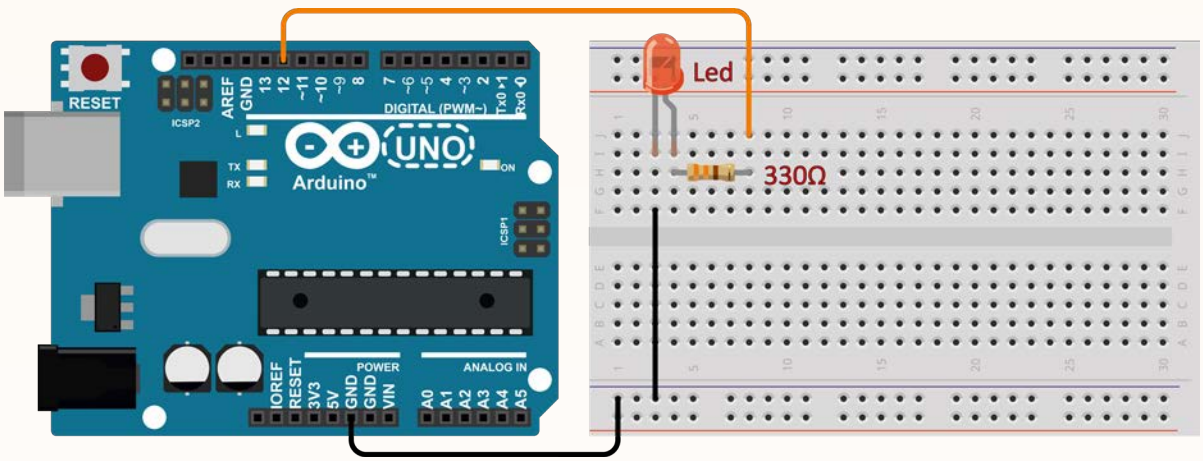


Görsel 3.13: DAC çevrim kontrolü

1. UYGULAMA

BİR LED'İN FARKLI SÜRELERDE YAKILIP SÖNDÜRÜLMESİ

AMAÇ: **Bir LED'i farklı sürelerde yakıp söndürmek.**

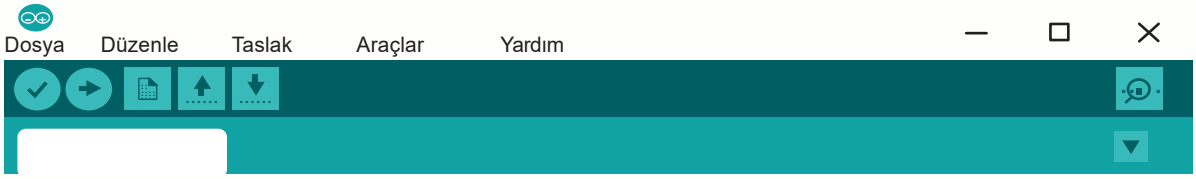


Görsel 3.14: Uygulamaya ait devre bağlantı şeması

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 3.14'te verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. LED'in, yazılımda belirlediğiniz sürelerde yanıp söndüğünü gözlemleyiniz.
5. Yazılımda verilen süreleri 1000 ms yapınız, yazılımı derledikten sonra mikrodenetleyici karta yükleyiniz ve değişikliği gözlemleyiniz.
6. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM




```
                                    // Bir adet LED'in farklı sürelerde yakılıp söndürülmesi.
                                    // Mikrodenetleyici kartın 12 no.lu pinine LED bağlanması.
void setup()                        // Mikrodenetleyiciyi ayarlama fonksiyonu.

{
    pinMode(12, OUTPUT);            // 12 no.lu pini çıkış olarak ayarla.
}
void loop()                         // Ana program fonksiyonu.

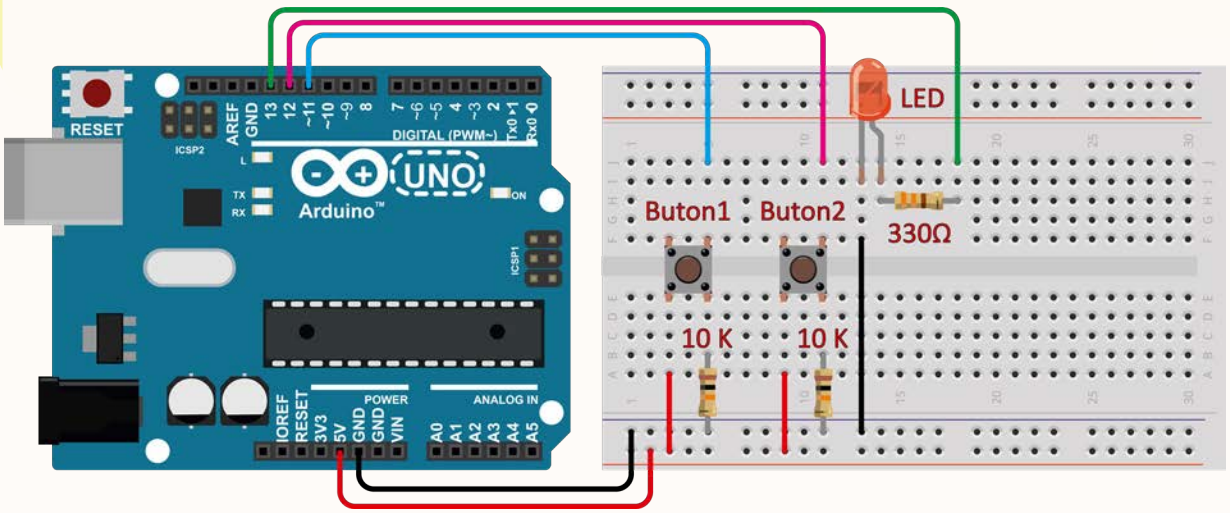
{
    digitalWrite(12, 1);            // 12 no.lu pine lojik 1 gönder.
                                    // Çıkış 5 V olur ve LED yanar.
    delay(500);                     // 500 milisaniye bekle.
    digitalWrite(12, 0);            // 12 no.lu pine lojik 0 gönder.
                                    // Çıkış 0 V olur ve LED söner.
    delay(500);                     // 500 milisaniye bekle.
}                                   // Loop ile belirtilen ana fonksiyonuna geri dön.
                                    // Program sürekli bu şekilde devam eder.
```

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Uygulama programını doğru açıkladı.	20	
İmza :		**TOPLAM PUAN**	100	

2. UYGULAMA

İKİ BUTONLA BİR LED'İN YAKILIP SÖNDÜRÜLMESİ

AMAÇ: **İki butonla bir LED'i yakıp söndürmek.**

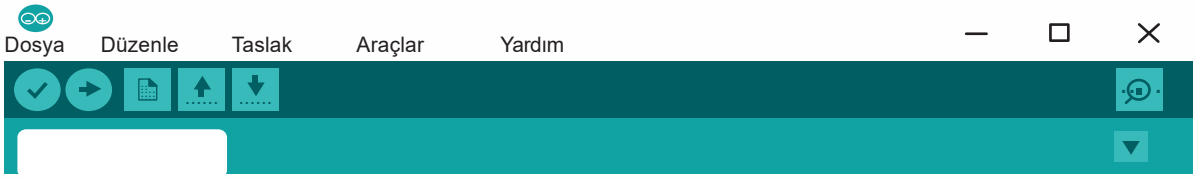


Görsel 3.15: Uygulamaya ait devre bağlantı şeması

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 3.15'te verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. 1. ve 2. butonlara sırasıyla basarak sonucu gözlemleyiniz.
5. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM



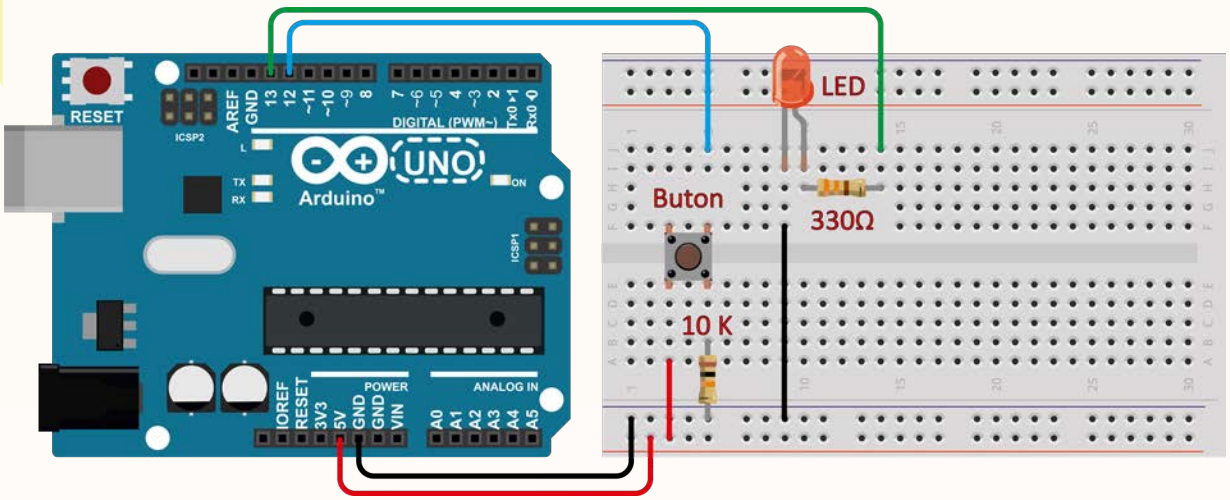

```
                                    // İki butonla bir LED'in yakılıp söndürülmesi. On-Off
 int ButOn  =  11 ;                 // Ardunio 11 no.lu pine 1.buton (ON) bağlandı.
 int ButOff =  12 ;                 // Ardunio 12 no.lu pine 2.buton (OFF) bağlandı.
 int Led    =  13 ;                 // Ardunio 13 no.lu pine LED bağlandı.
void setup()                        // Mikrodenetleyiciyi ayarlama fonksiyonu.
{
     pinMode(ButOn, INPUT);         // 11 no.lu pini giriş olarak ayarla.
     pinMode(ButOff,INPUT);         // 12 no.lu pini giriş olarak ayarla.
     pinMode(Led, OUTPUT);          // 13 no.lu pini çıkış olarak ayarla.
     digitalWrite(Led, 0);          // Başlangıçta LED yanıyorsa söndür.
}
void loop()                         // Ana program fonksiyonu.
{
     if (digitalRead(ButOn)==1)  {  // 1.buton değerini oku, 1 ise
       digitalWrite(Led, 1);        // LED'i yak.
   }
     if (digitalRead(ButOff)==1) {  // 2.buton değerini oku, 1 ise
       digitalWrite(Led, 0);        // LED'i söndür.
   }
}                                   // Loop ile belirtilen ana fonksiyonuna geri dön.
```

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Uygulama programını doğru açıkladı.	20	
İmza :		**TOPLAM PUAN**	100	

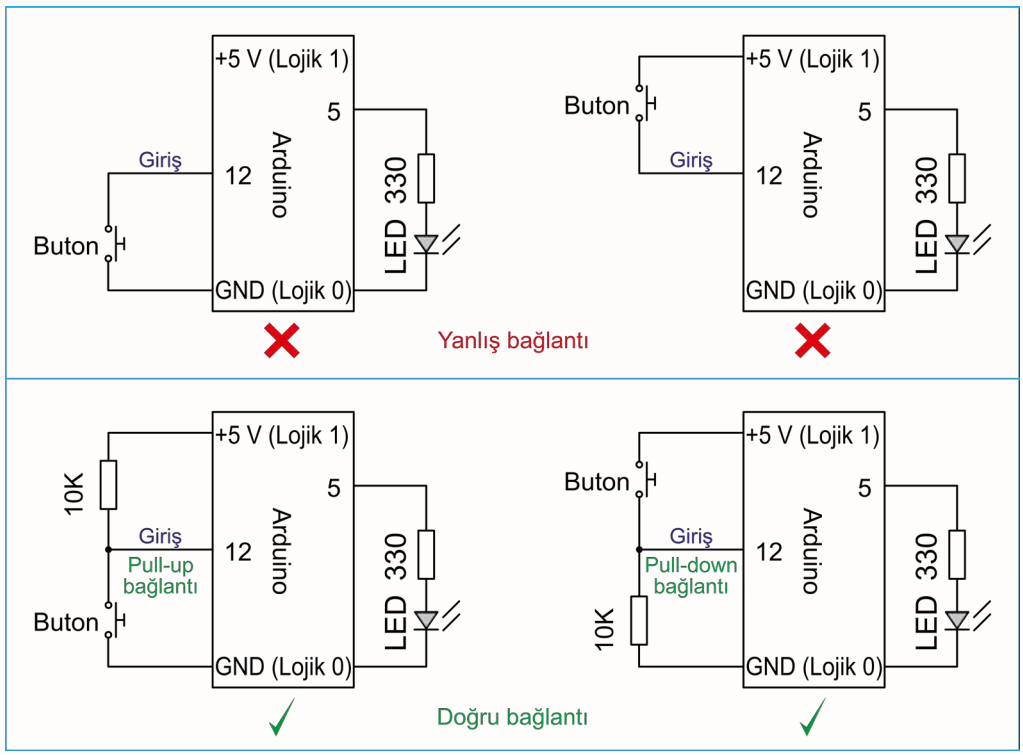
3. UYGULAMA

BİR BUTONLA BİR LED'İN YAKILIP SÖNDÜRÜLMESİ

AMAÇ: **Bir butonla bir LED'i yakıp söndürmek.**



Görsel 3.16: Uygulamaya ait devre bağlantı şeması



Görsel 3.17: Mikrodenetleyici giriş bağlantıları

BİLGİ

Mikrodenetleyici portları giriş olarak tanımlandığında bu uçlar boşta bırakılmamalıdır. Giriş empedansları çok yüksek olduğu için düşük elektriksel alan veya parazitlerden etkilenir. Bu durum giriş bilgisini yanlış algılamasına neden olur. Butona basılmasa bile oluşacak parazitlerden dolayı basılmış gibi algılayabilir. Görsel 3.17'deki yanlış bağlantı şekli buna örnektir. Doğru bağlantıysa iki şekilde olur:

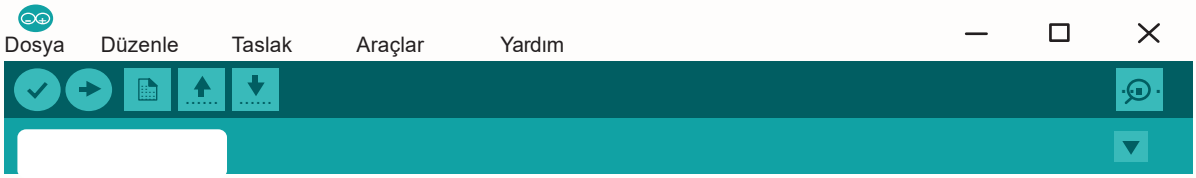
Pull-up: Girişlerin bir direnç üzerinden lojik 1 (+5 V) verilmesidir. Butona basılmadığında girişin normal durumu lojik 1(+5 V) seviyesindedir. Butona basıldığında lojik 0 (0 V) seviyesine iner.

Pull-down: Girişlerin bir direnç üzerinden lojik 0 (0 V) verilmesidir. Butona basılmadığında girişin normal durumu lojik 0 (0 V) seviyesindedir. Butona basıldığında lojik 1 (+5 V) seviyesine çıkar.

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 3.16'da verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. Butona her bastığınızda LED'in durumunu gözlemleyiniz.
5. Uygulamada butona 10K pull-down direnci bağlanmıştır, direnci pull-up yapınız. Programın aynı şekilde çalışması için gerekli donanım ve program değişikliklerini yapınız.
6. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM




```
                                        // Bir butonla bir LED'in yakılıp söndürülmesi. On-Off
 int ButOnOff = 12 ;                    // Ardunio 12 no.lu pine buton bağlandı.
 int Led      = 13 ;                    // Ardunio 13 no.lu pine LED bağlandı.
 int ButDeg=0;                          // ButDeg değişkenini integer ve değerini 0 olarak tanımla.
void setup()                            // Mikrodenetleyiciyi ayarlama fonksiyonu.
{
  pinMode(ButOnOff, INPUT);             // 12 no.lu pine giriş olarak ayarla.
  pinMode(Led, OUTPUT);                 // 13 no.lu pine çıkış olarak ayarla.
  digitalWrite(Led, 0);                 // Başlangıçta LED yanıyorsa söndür.
}
void loop()                             // Ana program fonksiyonu.
{
  if (digitalRead(ButOnOff)==0){        // Buton değeri 0 ise
    ButDeg=0;                           // ButDeg degişkenini 0 yap, kilidi aç.
  }
// Aşağıdaki program komutlarında butona basıldığında LED'in durumu okunur. Değili alınır (1 ise 0, 0 ise 1).
// ButDeg değişkeni 1 yapılır ve yazılımsal kilit yapılarak, LED'in sürekli yanıp sönmesi önlenir.
// Kilidin açılması için buton durumunun 0'a düşmesi gerekir. Bu okuma yukarıda yapılır.

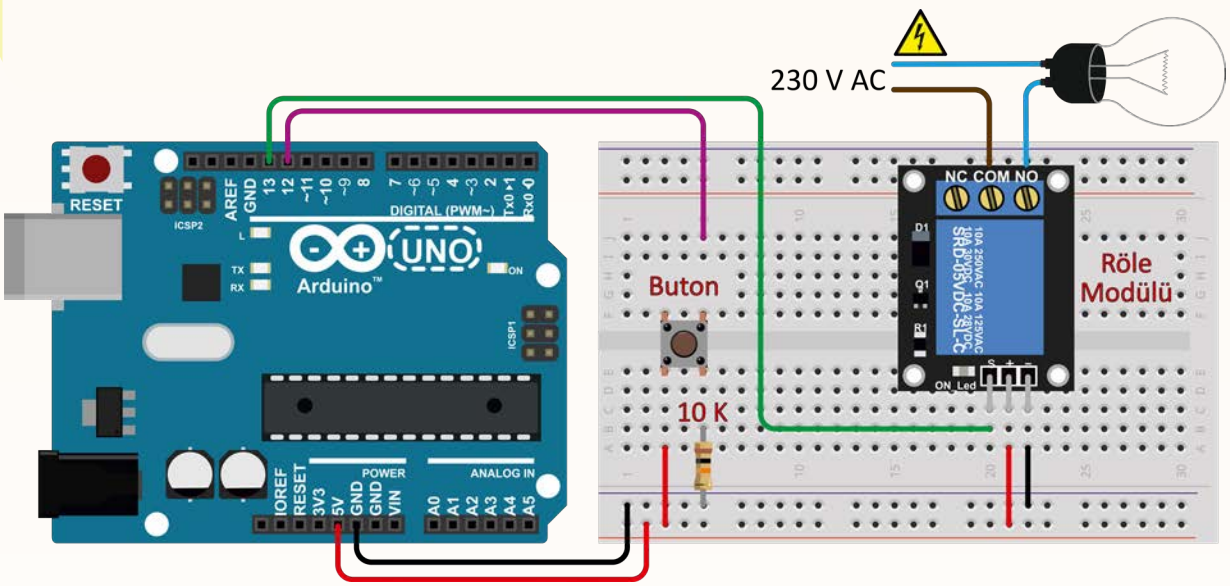
  if (ButDeg==0){
    if (digitalRead(ButOnOff)==1) {
       digitalWrite(Led, !digitalRead(Led));
       ButDeg=1;
       delay(100);                      // Buton basımında oluşan sıçramaları engellemek için 100 ms bekle.
    }
  }
}                                       // Loop ile belirtilen ana fonksiyonuna geri dön.
```

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Uygulama programını doğru açıkladı.	20	
İmza :		**TOPLAM PUAN**	100	

4. UYGULAMA

RÖLE MODÜLÜNE BAĞLI 230 V AC YÜKÜN ON-OFF ÇALIŞTIRILMASI

AMAÇ: **Röle modülüne bağlı 230 V AC yükü on-off çalıştırmak.**

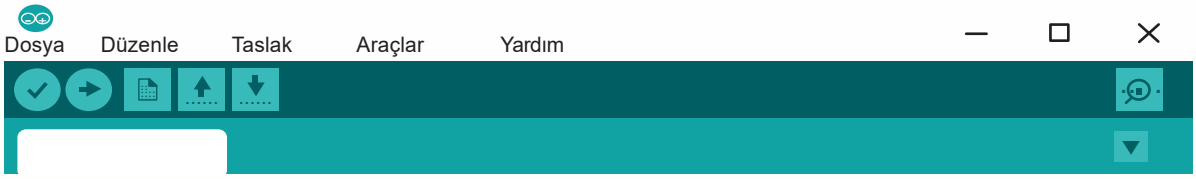


Görsel 3.18: Uygulamaya ait devre bağlantı şeması

İŞLEM BASAMAKLARI

1. İş sağlığı ve güvenliği tedbirlerini alarak çalışma ortamını hazırlayınız.
2. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
3. Görsel 3.18'de verilen devre kurulumunu gerçekleştiriniz.
4. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
5. Butona her bastığınızda rölenin on-off çalıştığını gözlemleyiniz.
6. Rölenin normalde açık kontaklarına 230 V lamba bağlayarak enerji veriniz.
7. Butona her bastığınızda lambanın on-off çalıştığını gözlemleyiniz.
8. Programda verilen delay (100) satırını iptal edip yeniden yükleyiniz ve değişikliği gözlemleyiniz.
9. Enerjiyi kesiniz.
10. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM




```
// Bir butonla röle modülünün on-off çalıştırılması.
// Aşağıdaki program komutları bir butonla LED yakıp söndürme programının aynısıdır. LED yerine röle modülü
// bağlanarak kontrol edilmiştir.

 int RoleOnOff = 12 ;                  // Ardunio 12 no.lu porta buton bağlandı.
 int Role      = 13 ;                  // Ardunio 13 no.lu porta röle modülü bağlandı.
 int RoleDeg=0;                        // RoleDeg değişkenini integer ve başlangıç değerini 0 olarak tanımla.

void setup()                           // Mikrodenetleyiciyi ayarlama fonksiyonu.
{
  pinMode(RoleOnOff, INPUT);           // 12 no.lu portu giriş olarak ayarla.
  pinMode(Role, OUTPUT);               // 13 no.lu portu çıkış olarak ayarla.
  digitalWrite(Role, 0);               // Başlangıçta röleyi kapat.
}
void loop()                            // Ana program fonksiyonu.
{
  if (digitalRead(RoleOnOff)==0){
    RoleDeg=0;
  }

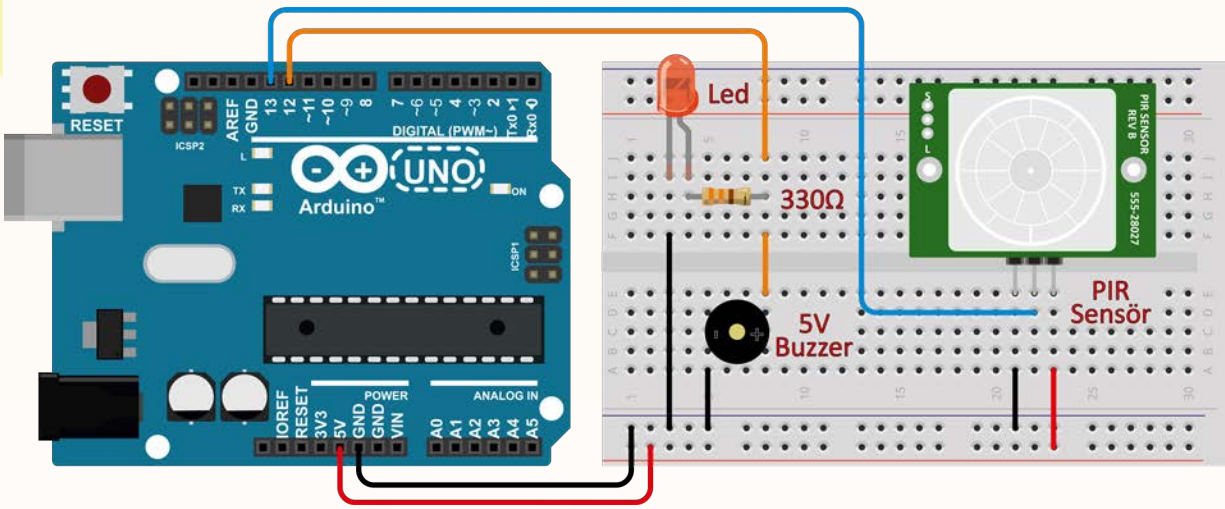
  if (RoleDeg==0){
    if (digitalRead(RoleOnOff)==1) {
        digitalWrite(Role, !digitalRead(Role));
        RoleDeg=1;
        delay(100);
    }
  }
}                                      // Loop ile belirtilen ana fonksiyonuna geri dön.
```

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Uygulama programını doğru açıkladı.	20	
İmza :		**TOPLAM PUAN**	100	

5. UYGULAMA

PIR DEDEKTÖRÜ İLE CANLI HAREKETİ ALGILANMASI

AMAÇ: PIR dedektörü ile canlı hareketi algılamak.



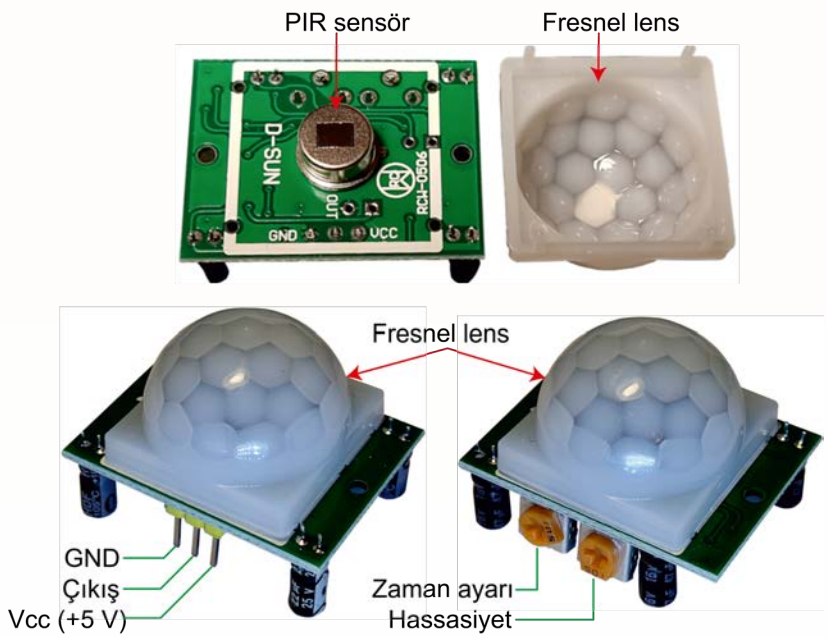
Görsel 3.19: Uygulamaya ait devre bağlantı şeması

BİLGİ

PIR sensör: Ortamdaki hareket eden canlıyı algılayan elektronik sensördür. Canlılar vücut ısılarıyla etrafa kızılötesi (infrared-IR) ışın yayar. Sensör, ortamdaki bu IR değişimi algılar ve çıkış olarak verir. Fresnel lens (mercek), oluşan kızılötesi ışığı toplayıp sensör üzerinde odaklanmasını sağlayarak algılama mesafesini artırır. Yarım küre şeklinde yapılmaları geniş algılama açısını artırmak içindir.

PIR dedektörü üzerinde iki adet ayar trimpotu mevcuttur. Bunlar, algılama hassasiyetini ayarlayan trimpot ve algılama gerçekleştiğinde çıkışın ne kadar süre aktif olacağını belirleyen zaman ayarı trimpotudur. Hareket algılandığında çıkış ucundan lojik 1 (+5 V), değilse lojik 0 (0 V) alınır (Görsel 3.20).

Buzzer: Genellikle piezoelektrik prensibine göre çalışan işitsel ikaz cihazıdır. Enerji verildiğinde doğrudan ikaz sesi çıkaranlara devreli buzzer, ikaz sesi çıkarmak için osilatöre ihtiyaç duyanlara **devresiz buzzer** denir. Uygulamada 5 V ile çalışan **devreli buzzer** kullanılmıştır.



Görsel 3.20: PIR dedektörün yapısı ve bağlantı uçları

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 3.19'da verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. PIR sensörünü uzak bir konuma yerleştiriniz ve devreye enerji veriniz.
5. Hareket ettiğinizde sensörün aktif olarak mikrodenetleyici karta sinyal gönderdiğini, çıkışındaki LED ve buzzer'ın programladığınız şekilde çalıştığını gözlemleyiniz.
6. PIR sensör üzerinde bulunan ayar trimpotlarından hassasiyet ve zaman ayarını yaparak bu ayarların nasıl bir sonuç oluşturduğunu gözlemleyiniz.
7. PIR sensör üzerinde bulunan fresnel lensi sökünüz ve çalışmasını gözlemleyiniz.
8. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM

Dosya Düzenle Taslak Araçlar Yardım

```
                                    // PIR dedektörü ile canlı hareketi algılanması
 int Led_Buzzer = 12 ;              // Ardunio 12 no.lu pine LED ve buzzer bağlandı.
 int PIR_sensor = 13 ;              // Ardunio 13 no.lu pine PIR dedektör bağlandı.

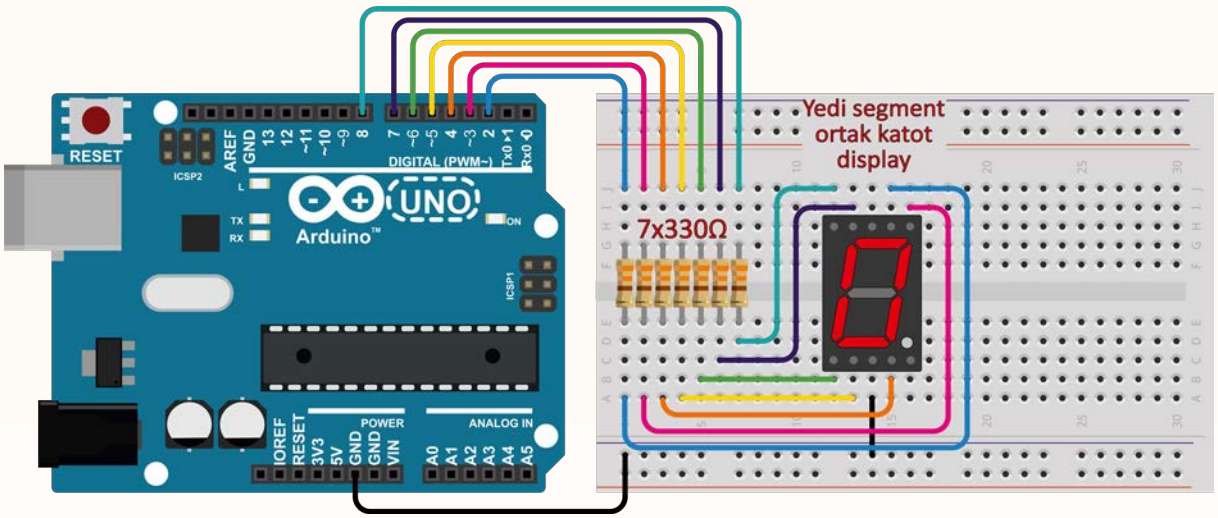
void setup()
{
    pinMode(PIR_sensor, INPUT);     // 13 no.lu pin PIR_sensor için giriş.
    pinMode(Led_Buzzer, OUTPUT);    // 12 no.lu pin Led_Buzzer için çıkış.
    digitalWrite(Led_Buzzer, 0);    // Led_Buzzer'ı başlangıçta kapat.
}
void loop()
{
  if (digitalRead(PIR_sensor)==1){  // PIR dedektörü algılama yaparsa
    digitalWrite(Led_Buzzer, 1);    // LED ve buzzer 500 ms aralıklarla
    delay(500);                     // alarm ver.
    digitalWrite(Led_Buzzer, 0);
    delay(500);
  }
}
```

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Uygulama programını doğru açıkladı.	20	
İmza :		**TOPLAM PUAN**	100	

6. UYGULAMA

YEDİ SEGMENT DİSPLAY İLE 9-0 SANİYE GERİ SAYAN ZAMANLAYICI YAPILMASI

AMAÇ: **Yedi segment display ile 9-0 saniye geri sayan zamanlayıcı yapmak.**

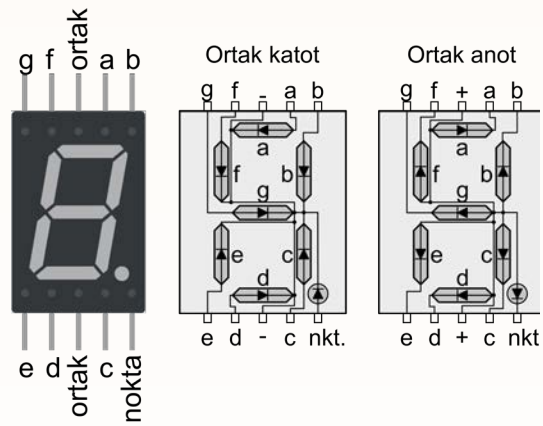


Görsel 3.21: Uygulamaya ait devre bağlantı şeması

BİLGİ

Yedi segment display, 7 adet segment ve bir nokta olmak üzere 8 adet LED bağlantısı şeklinde üretilir. Bu segmentler; a,b,c,d,e,f,g ve nokta şeklinde isimlendirilir. LED'lerin ortak uç bağlantısına göre katotlar ortaksa **ortak katot**, anotlar ortaksa **ortak anot** display denir (Görsel 3.22.)

Görsel 3.21'de olduğu gibi ortak katot bağlantılı displayde katot, mikrodenetleyici kart üzerindeki eksi (-) uca (GND); anotlar (segment) ise a-2, b-3, c-4, d-5, e-6, f-7, g-8 port uçlarına bağlanmıştır. Buna uygun rakam tablosu oluşturulur. Örneğin 1 rakamı için sadece b ve c segmentlerine lojik 1 (+5 V), diğerlerine lojik 0 (0 V) verildiğinde displayde o kısımda yer alan LED'ler yanacağı için 1 rakamı görünür. Programda her rakam için oluşturulmuş port rakam tablosu verilmiştir.



Görsel 3.22: Ortak katot ve anot iç bağlantı şeması

Bu uygulamada, tüm portlara aynı anda kolayca ulaşmak ve giriş-çıkış tanımlamaları yapmak için mikrodenetleyici kaydedicileri (register) kullanılmıştır. Bu kaydediciler şunlardır:

DDRx: x ile belirtilen port pinlerinin giriş-çıkış ayarlarının yapılması için kullanılır (Bitlere 0 verilirse giriş, 1 verilirse çıkış olarak ayarlanır.).

PORTx: x ile belirtilen port pinlerinden çıkış alınması için kullanılır.

PINx: x ile belirtilen port pininin değerinin okunması için kullanılır.

ARDUINO UNO PIN	-	-	-	13	12	11	10	9	8		7	6	5	4	3	2	1	0
PORTB	-	-	-	5	4	3	2	1	0	PORTD	7	6	5	4	3	2	1	0
DDRB	0	0	0	0	0	0	0	1	1	DDRD	1	1	1	1	1	1	0	0

DDRB = B00000011; ifadesinde başta bulunan 0b veya B harfi sayının binary olduğunu gösterir. B portunun 0. ve 1. bitlerini (arduino kartta 8. ve 9. pinler) çıkış, diğerlerini giriş olarak ayarlar.

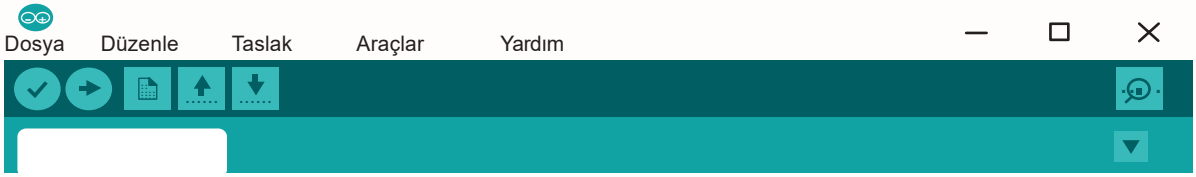
DDRD = B11111100; ifadesinde D portunun 0. ve 1. bitlerini (arduino kartta 0. ve 1. pinler) giriş, diğerlerini çıkış olarak ayarlar.

PORTB = B00000011; ifadesinde B portunun 0. ve 1. bitini aynı anda lojik 1 yapar.

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 3.21'de verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. Displayde 9-0 arası 1 sn. aralıklarla geriye doğru sayım yaptığını gözlemleyiniz.
5. Devreye ortak anot display bağlayınız, rakam tablosunu çıkartınız ve programı tekrar yükleyiniz.
6. Heksadesimal (F-d-C-b-A-9-8-7-6-5-4-3-2-1-0) sayıcı yapmak için programda gerekli değişikliği yapınız ve programı tekrar yükleyiniz.
7. Displayde 0-9 arası 1 sn. aralıklarla ileri sayıcı yapmak için programda gerekli değişikliği yapınız ve programı tekrar yükleyiniz.
8. 100 ms aralıklarla geri sayım yapması için programda gerekli değişikliği yapınız ve programı tekrar yükleyiniz.
9. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM



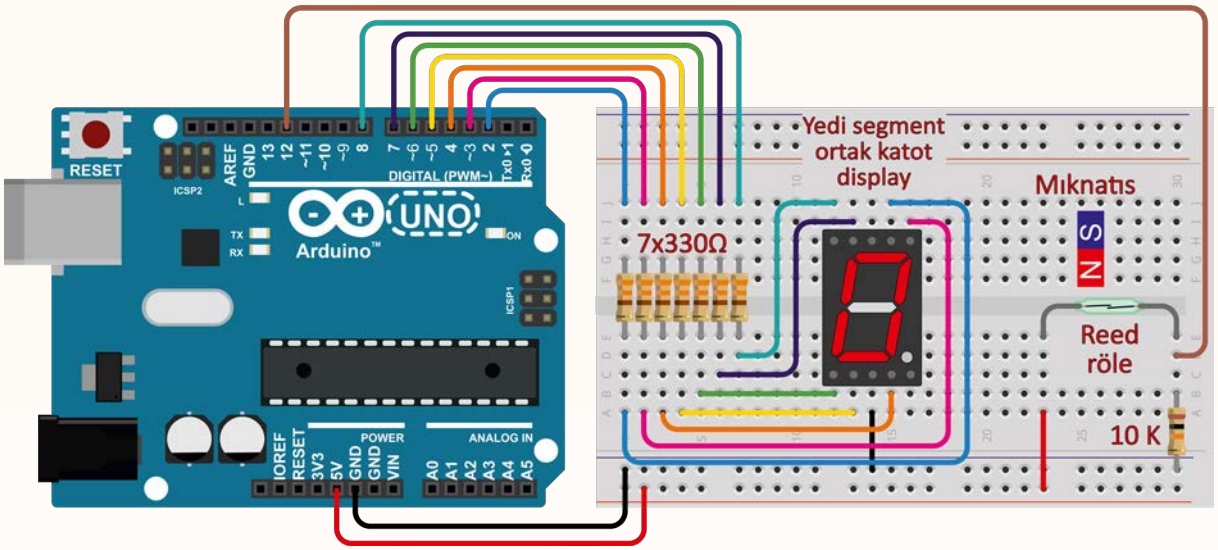

```
// Yedi segment ortak katot display ile 9-0 sn. geri sayan zamanlayıcı.
// Program çalıştığında setup içindeki port ayarlamalarını yaptıktan sonra ana programa girer.
// For döngüsü ile sayi değişkeni 9'dan 0'a kadar birer birer 1 sn. aralıklarla azalır. Her azaltma yapıldığında
// if komutu ile karşılaştırma yapılır. Sayı değişkeninin durumuna bakılarak, pinlerden gerekli çıkışlar alınıp
// displayde gösterim yapılır. Sayı 0 olduğunda tekrar baştan 9-0 sayımı yapılır.
int sayi;
void setup() {
  DDRD = 0b11111100;                    // 0,1 numaralı pinler giriş, diğerleri çıkış ayarlandı.
  DDRB = 0b00000001; }                  // 8 numaralı pin çıkış ayarlandı.
void loop() {                           // Ana program fonksiyonu.
     for(sayi=9;sayi>=0;sayi--){
                if (sayi==0){ PORTB = B00000000; PORTD = B11111100; }     // Display 0
                if (sayi==1){ PORTB = B00000000; PORTD = B00011000; }     // Display 1
                if (sayi==2){ PORTB = B00000001; PORTD = B01101100; }     // Display 2
                if (sayi==3){ PORTB = B00000001; PORTD = B00111100; }     // Display 3
                if (sayi==4){ PORTB = B00000001; PORTD = B10011000; }     // Display 4
                if (sayi==5){ PORTB = B00000001; PORTD = B10110100; }     // Display 5
                if (sayi==6){ PORTB = B00000001; PORTD = B11110100; }     // Display 6
                if (sayi==7){ PORTB = B00000000; PORTD = B00011100; }     // Display 7
                if (sayi==8){ PORTB = B00000001; PORTD = B11111100; }     // Display 8
                if (sayi==9){ PORTB = B00000001; PORTD = B10111100; }     // Display 9
   delay(1000);   }
}
```

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Uygulama programını doğru açıkladı.	20	
İmza :		**TOPLAM PUAN**	100	

7. UYGULAMA

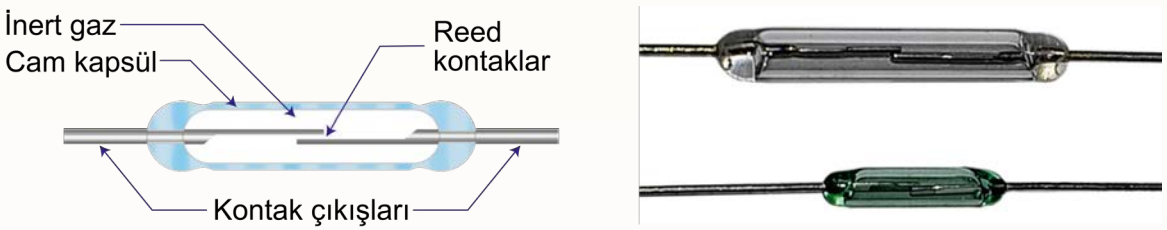
REED RÖLE İLE 0-9 SAYICI YAPIMI

AMAÇ: **Reed röle ile 0-9 sayıcı yapmak.**



Görsel 3.23: Uygulamaya ait devre bağlantı şeması

BİLGİ



3.24: Reed rölenin içyapısı ve şekli

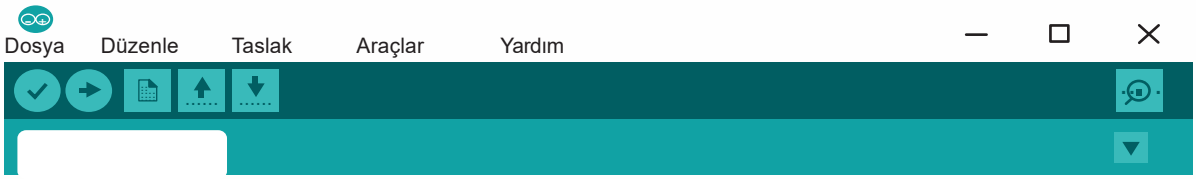
Reed röle, manyetik anahtar olarak da adlandırılır. Cam tüp içine yerleştirilmiş nikel ve demirden yapılmış kontakları bulunan bir röle çeşidir (Görsel 3.24). Kontak çıkışları normalde açıktır. Uygulanan bir manyetik alanda, reed kontakların birbirine doğru çekilmesiyle kontaklar temas eder ve kontak çıkışları kapalı konuma gelir. Manyetik alan ortadan kaldırıldığında eski durumuna geri döner. İçerisindeki inert gaz tepkimeye girmediği için kontakların oksitlenmesi ve temas anında oluşabilecek arkın düşük seviyede kalması amacıyla kullanılır.

Manyetik alan için mıknatıs kullanılabilir. Bir motorun miline mıknatıs bağlanarak tur sayısı, kapı veya pano kapağına monte edilerek açma-kapama sayısı kontrol edilebilir.

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Ölçü aletini buzzer konumuna alınız ve reed röleye mıknatıs yaklaştırarak uçlarını ölçünüz.
3. Görsel 3.23'te verilen devre kurulumunu gerçekleştiriniz.
4. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
5. Mıknatısı reed röleye yaklaştırıp uzaklaştırarak displayde artış miktarını gözleyiniz.
6. Mikrodenetleyici karta bir buton bağlayınız. Butona basıldığında displaydeki rakamı 0 yapacak (resetleme) gerekli donanım ve program değişikliklerini yapınız.
7. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM



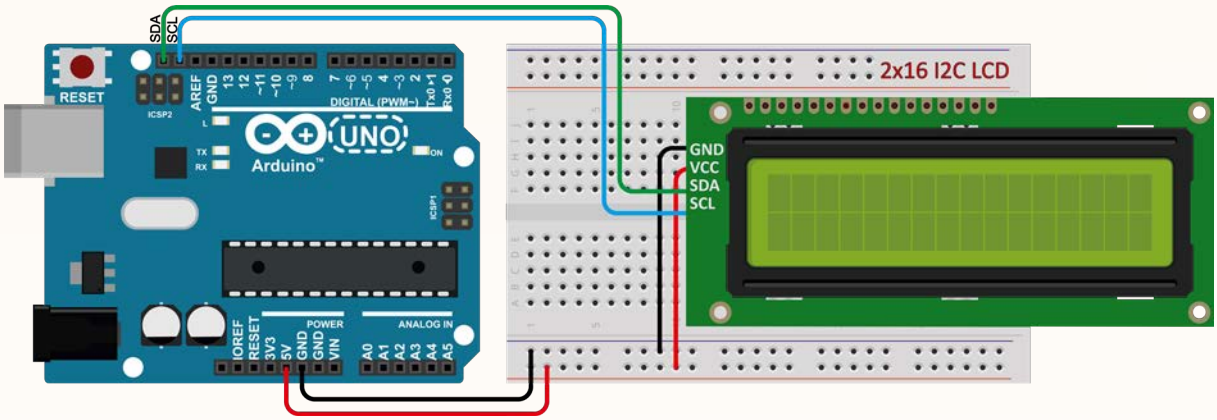

```
                                    // Reed röle ile 0-9 sayıcı yapımı.
                                    // Bir butonla on-off uygulamasında buton yerine reed röle bağlanmıştır.
                                    // Yedi segment displayli zamanlayıcıdaki yapı kullanılarak sayma yapılmıştır.
 int ReedPin = 12 ;                 // Ardunio 12 no.lu pine reed röle bağlandı.
 int sayi,ReedDeg=0;                // Değişkenleri tanımla.
 void setup()
{
 DDRD =  B11111100;                 // 0,1 numaralı pinler giriş, diğerleri çıkış ayarlandı.
 DDRB =  B00000001;                 // 8 numaralı pin çıkış ayarlandı.
 PORTB = B00000000; PORTD = B11111100;                  //İlk açılışta displayde 0 göster.
}
void loop()
{
  if (digitalRead(ReedPin)==0){
    ReedDeg=0;                      // Tekrar sayımı engellemek için değişken kilidi kullanıldı.
  }
  if (ReedDeg==0){
     if (digitalRead(ReedPin)==1) {
         sayi++;                    // Sayi değişkenini 1 artır.
         ReedDeg=1;
    if (sayi==1){ PORTB = B00000000; PORTD = B00011000; }
    if (sayi==2){ PORTB = B00000001; PORTD = B01101100; }
    if (sayi==3){ PORTB = B00000001; PORTD = B00111100; }
    if (sayi==4){ PORTB = B00000001; PORTD = B10011000; }
    if (sayi==5){ PORTB = B00000001; PORTD = B10110100; }
    if (sayi==6){ PORTB = B00000001; PORTD = B11110100; }
    if (sayi==7){ PORTB = B00000000; PORTD = B10011100; }
    if (sayi==8){ PORTB = B00000001; PORTD = B11111100; }
    if (sayi==9){ PORTB = B00000001; PORTD = B10111100; }
    if (sayi>9) { PORTB = B00000000; PORTD = B11111100;  sayi=0; }
        delay(100);
     }
  }
}
```

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Uygulama programındaki değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

8. UYGULAMA

2X16 KARAKTER LCD İLE 0-100 SANİYE ZAMANLAYICI YAPILMASI

AMAÇ: 2x16 karakter LCD ile 0-100 saniye zamanlayıcı yapmak.



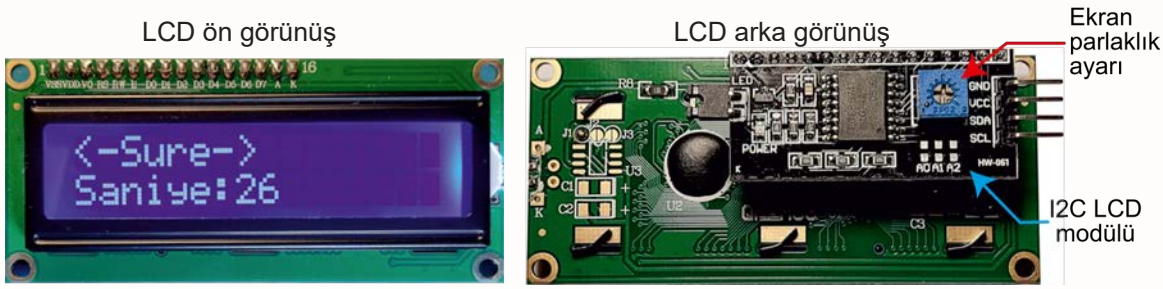
Görsel 3.25: Uygulamaya ait devre bağlantı şeması

BİLGİ

LCD: Sıvı kristal ekran anlamındadır. Yazı ve sayılar piksellerden oluşmaktadır. Piksellerin olduğu kısımda yer alan sıvı kristaller, herhangi bir elektrik alanı uygulanmadığında farklı pozisyonda bulunur ve önden gelen ışığı direkt yansıttığı için o kısım beyaz görünür. Sıvı kristallere elektrik alanı uygulandığında konum değiştirir ve ışık geri yansımaz, o bölge siyah görünür.

LCD'ler satır ve sütun sayılarına göre isim alır. Örneğin 2x16. Bu, LCD'nin 2 satır ve 16 sütunu olduğunu gösterir. 1. satırda 16 ve 2. satırda 16 olmak üzere toplam 32 ayrı karakter yazılabilir.

LCD bağlantısında Görsel 3.26'daki ön görünüşte olduğu gibi 16 pin bulunur. Çalıştırmak için en az 8 pin kullanılmalıdır. Bu, I2C protokolü kullanan bir modül ile sadece 4 pin kullanılarak sağlanabilir. Görsel 3.26'daki arka görünüşte, haberleşme uçları SDA, SCL; ortak uç GND ve besleme VCC'dir. I2C bir haberleşme protokolü olup birçok cihazın birbiriyle haberleşmesi için kullanılır.



Görsel 3.26: 2x16 I2C LCD'nin ön ve arka görünümü

Bağlantı Bilgileri

I2C Modülü üzerinde 4 adet pin vardır. Bu pinler şunlardır:

GND = Arduino GND

VCC = Arduino +5 V

SDA = Arduino SDA veya A4

SCL = Arduino SCL veya A5

Arduino için LCD I2C(Arduino-LiquidCrystal-I2C) kütüphanesinin indirilip kurulması gerekir. Bunun için Arduino ide programında;

Taslak > library ekle (kütüphane ekle) > .ZIP Kitaplığı ekle (Arduino-LiquidCrystal-I2C.zip) şeklinde yapılmalıdır.

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 3.25'te verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. 0-100 arası sayımı LCD üzerinde gözlemleyiniz.
5. Yazıların LCD'yi ortalayacak şekilde yazılması için gerekli program değişikliğini yapınız ve programı tekrar yükleyiniz.
6. Devreye buton ekleyerek herhangi bir sayım sırasında 0 (resetleme) yapmak için gerekli donanım ve program değişikliklerini yapınız.
7. Programda, 1000'e kadar 100 ms aralıklarla sayması için gerekli program değişikliğini yapınız ve sonucu gözlemleyiniz.
8. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM

Dosya Düzenle Taslak Araçlar Yardım

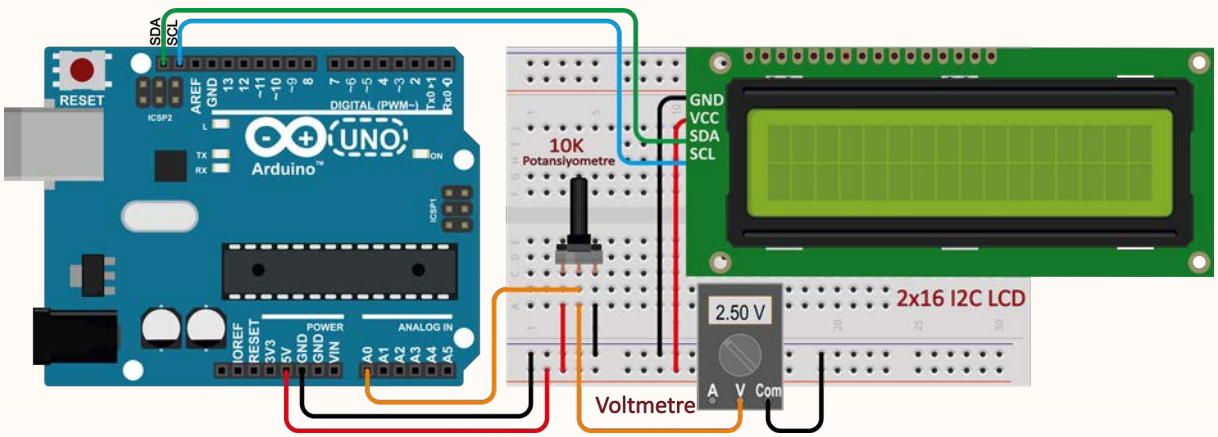
```
// I2C haberleşmeli 2x16 karakter LCD ile 0-100 sn. zamanlayıcı yapılması.
#include <LiquidCrystal_I2C.h>                  // I2C LCD kütüphanesi.
 LiquidCrystal_I2C lcd(0x3f,16,2);              // I2C LCD adres sütun ve satır tanımı
                                                // Bazı LCD'lerin I2C adresi 0x3f yerine 0x27 olabilmektedir.
int saniye=0;
void setup()
{
     lcd.begin();                               // LCD'yi başlat.
}
void loop()
{
     lcd.backlight();                           // LCD arka plan ışığını aç.
     lcd.setCursor(0,0);                        // 1.satır 1.sütun başlangıç yeri.
     lcd.print("<-Sure->");                     // 1.satırda "<-Sure->" yaz.
     lcd.setCursor(0,1);                        // 2.satır 1.sütun başlangıç yeri.
     lcd.print("Saniye:");                      // 2.satırda "Saniye:" yaz.
     lcd.print(saniye);                         // 2.satırda saniye değişken değerini yaz.
     saniye++;                                  // Saniye değişkenini 1 artır.
     delay(1000);                               // 1 saniye bekle.
      lcd.clear();                              // LCD mevcut yazıları sil.
      if (saniye>100){                          // Eğer saniye değişkeni 100'den büyükse
         saniye=0;                              // saniye değişkenini 0 yap.
          lcd.noBacklight();                    // LCD arka plan ışığını kapat.
          delay(5000); }                        // 5 sn. bekle ve yeniden saymaya başla.
  }
```

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Uygulama programındaki değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

9. UYGULAMA

ANALOG DİJİTAL ÇEVİRİCİ (ADC) UYGULAMASI

AMAÇ: Potansiyometreden okunan ADC ve gerilim değerini LCD'de göstermek.



Görsel 3.27: Uygulamaya ait devre bağlantı şeması

BİLGİ

Uygulamada sensör yerine potansiyometre kullanılmıştır. Potansiyometre çıkışı 0-5 V olarak Arduino A0'a bağlanmıştır (Görsel 3.28).

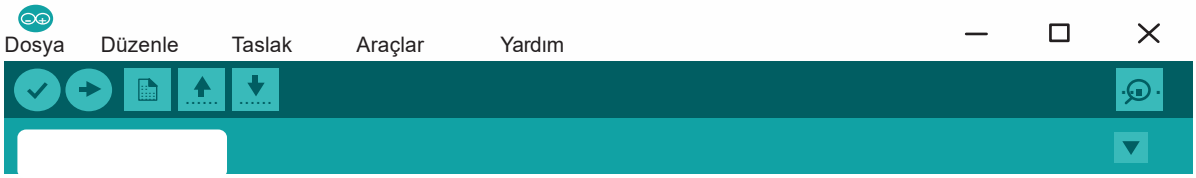


Görsel 3.28: ADC uygulamasına ait LCD görüntüsü

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 3.27'de verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. A0 kanalına voltmetre bağlayınız.
5. Potansiyometreyi önce sağa, sonra sola doğru çevirip voltmetre ve LCD'deki değişimi karşılaştırınız.
6. Bu uygulama bir voltmetre uygulamasıdır. Ampermetre için gerekli donanım ve program değişikliklerini yapınız ve sonucu gözlemleyiniz.
7. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM




```
                                        // I2C 2x16 karakter LCD ile Potansiyometreden okunan
                                        // ADC ve gerilim değerinin LCD ekranda okunması.
#include <LiquidCrystal_I2C.h>          // I2C LCD kütüphanesi.
LiquidCrystal_I2C lcd(0x3f,16,2);
int ADCdeg=0;
float Vdeg;                             // Ondalıklı işlemler için Vdeg değerini float tanımla.
void setup()
{
    lcd.begin();                        // LCD'yi başlat.
    lcd.backlight();                    // LCD arka plan ışığını aç.
}

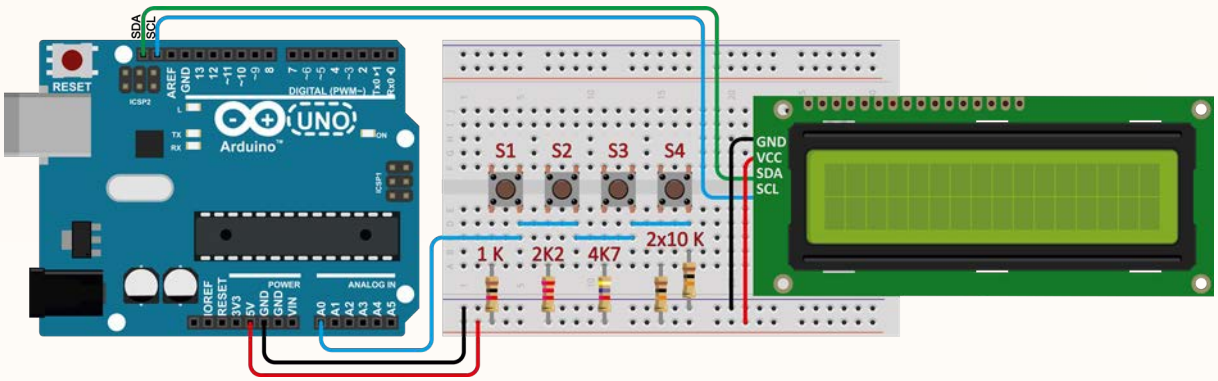
void loop()
{
    ADCdeg = analogRead(A0);            // A0 pininden okunan ADC değerini
                                        // ADCdeg değişkenine ata.
    lcd.setCursor(0,0);                 // 1.satır başlangıç yeri.
    lcd.print("ADC:");                  // 1.satıra "ADC:" ve arkasına ADCdeg
     lcd.print(ADCdeg);                 // değişkenine atanan ADC değerini yaz.
    lcd.setCursor(0,1);                 // 2.satır başlangıç yeri.
    lcd.print("Gerilim:");              // 2.satıra "Gerilim:" yaz.
    Vdeg = 0.00488*ADCdeg;              // Okunan ADC değerini, gerilim değerine çevirmek için hesapla.
     lcd.print(Vdeg);                   // 2.satırda hesaplanan gerilim değerini ve
    lcd.print(" V.");                   // arkasına" V." yaz.
    delay(500);                         // 500 ms bekle.
    lcd.clear();                        // LCD ekranı temizle.
}
```

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Donanım ve programdaki değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

10. UYGULAMA

ADC İLE BİRDEN FAZLA BUTON DEĞERİNİN OKUNMASI

AMAÇ: **Potansiyometreden okunan ADC ve gerilim değerini LCD'de göstermek.**

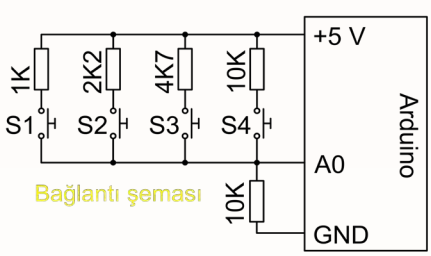


Görsel 3.29: Uygulamaya ait devre bağlantı şeması

BİLGİ

Giriş-çıkış sayısının yeterli olmadığı veya çok fazla buton kullanılan sistemlerde, bir analog kanal kullanılarak buton değerleri okunabilir. Görsel 3.30'da, bağlantı şeması ve buton durumlarına ait LCD görüntüleri görülmektedir. Herhangi bir butona basıldığında ona ait direnç devreye girer ve gerilim bölücü olarak A0 analog kanalında farklı gerilimler oluşturur. Bu gerilime karşılık gelen ADC değerleri karşılaştırılarak ilgili buton okuması gerçekleştirilir. Aynı anda iki veya daha fazla butona basıldığında farklı gerilim oluşacağı için o durumlara ait karşılaştırma da yapılabilir.

Örneğin bu sistem, sıvı seviye ölçümünde kullanılabilir. Sıvı içine birden fazla sıvı seviye anahtarı bağlanarak sıvının seviyesi daha detaylı şekilde ölçülebilir.











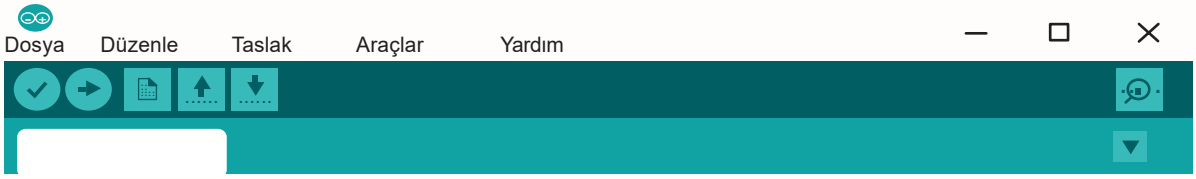


Görsel 3.30: Bir analog kanalda buton değerlerine ait LCD görüntüsü

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 3.29'da verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. Butonlara sırasıyla basarak LCD'de değerleri gözlemleyiniz.
5. Aynı anda iki butona basılma durumunu LCD'de gösterecek gerekli program değişikliğini yapınız ve programı tekrar yükleyiniz.
6. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM




```
                                        // ADC ile birden fazla buton değerininin LCD ekranda okunması.
#include <LiquidCrystal_I2C.h>          // I2C LCD kütüphanesi.
 LiquidCrystal_I2C lcd(0x3f,16,2);
 int ButADCdeg=0;
  void setup()
{
  lcd.begin();
  lcd.backlight();
  lcd.clear();
}
void loop()
{
  ButADCdeg = analogRead(A0);
  lcd.setCursor(0,0); lcd.print("Buton ADC:");
  lcd.print(ButADCdeg); lcd.print("    ");

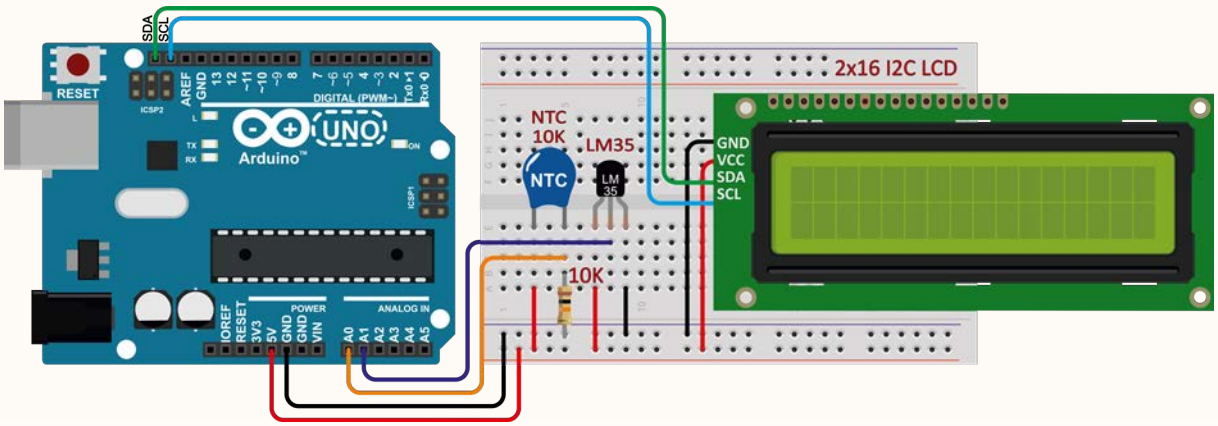
   if (ButADCdeg>=900 && ButADCdeg<=1023){   // Buton ADC değeri 900 ile 1023 arasındaysa 1.buton yaz.
      lcd.setCursor(0,1); lcd.print("1.Butona basildi.");
    }
    else if (ButADCdeg>=800 && ButADCdeg<900){
      lcd.setCursor(0,1); lcd.print("2.Butona basildi.");
    }
    else if (ButADCdeg>=600 && ButADCdeg<800){
      lcd.setCursor(0,1); lcd.print("3.Butona basildi.");
    }
    else if (ButADCdeg>=500 && ButADCdeg<600){
      lcd.setCursor(0,1); lcd.print("4.Butona basildi.");
    }
   delay(50);                                          // 50 ms bekle.
  }
```

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Uygulama programındaki değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

11. UYGULAMA

LM35 VE NTC ANALOG SICAKLIK SENSÖR DEĞERLERİNİN LCD'DE OKUNMASI

AMAÇ: LM35 ve NTC analog sıcaklık sensör değerlerini LCD'de göstermek.



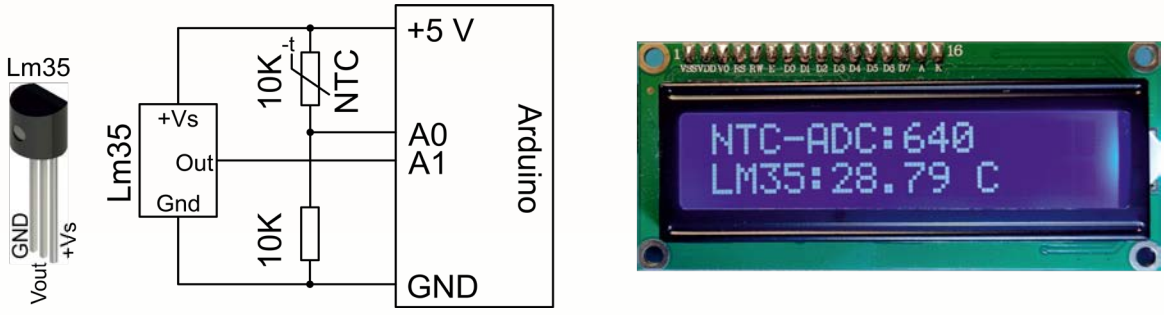
Görsel 3.31: Uygulamaya ait devre bağlantı şeması

BİLGİ

NTC: Sıcaklık arttıkça direnç değeri azalan pasif devre elemanıdır. Analog sıcaklık sensörü olarak kullanılır. −55 °C ile +200 °C'ye kadar ölçüm yapabilir. Sıcaklık değerine göre direnci doğrusal değişmediği için karmaşık matematiksel hesaplamalarla sıcaklık değeri hesaplanır. Uygulamada bu hesaplamaya girmeden sadece NTC çıkışındaki ADC değerinin sıcaklıkla değişimi gösterilecektir. Görsel 3.32'de, NTC ve 10K direnç, gerilim bölücü olarak bağlanıp çalıştırılmıştır. Sıcaklık değiştiğinde NTC direnç değişimi ve A0 analog kanalında gerilim değişimi olacaktır.

LM35: −55 °C ile +150 °C'ye kadar ölçüm yapabilen, yarı iletkenden yapılmış aktif bir analog sıcaklık sensörüdür. Her derece için 10 mV çıkış üretir. Örneğin sensör çıkışı 236 mV olarak ölçüldüğünde sıcaklık değeri 236/10=23,6 °C olacaktır. Sıcaklıkla çıkış değeri doğrusal değiştiği için matematiksel hesaplamalar daha basittir. Sensörün ölçtüğü sıcaklık değeri milivoltmetreyle doğrudan görülebilir.

LM35 sensör değerinin santigrat cinsinden hesaplama formülü için okunan ADC değeri mV cinsine çevrilir. Bunun için, Vref=5 V (5000 mV), mV=(Vref/1023) x ADC, mV= (5000/1023) x ADC mV= 4,88 x ADC olur. Her bir derece için 10 mV çıkış gerilimi elde edildiğine göre sonuç 10'a bölündüğünde santigrat cinsinden sıcaklık formülü **^{0}C= 0,488 x ADC** elde edilir.

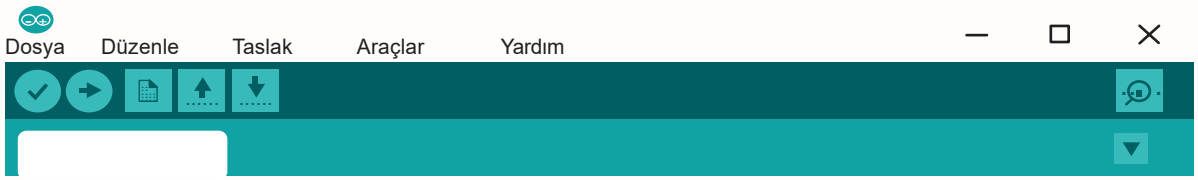


Görsel 3.32: LM35 ve NTC sıcaklık sensör devresi ve LCD görüntüsü

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 3.31'de verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. LM35 sensörünü ısıtınız. LCD'de sıcaklığın 0C cinsinden değiştiğini gözlemleyiniz.
5. NTC sensörünü ısıtınız. LCD'de ADC değerinin değiştiğini gözlemleyiniz.
6. Mikrodenetleyici dijital girişlerinden birini kullanarak ön dirençli (330Ω) bağlanmış bir LED bağlayınız.
7. Sıcaklık 30 ^{0}C'yi aştığında LED'in yanması, 29 ^{0}C'nin altındaysa LED'in sönmesi için gerekli donanım ve program değişikliklerini yapınız.
8. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM



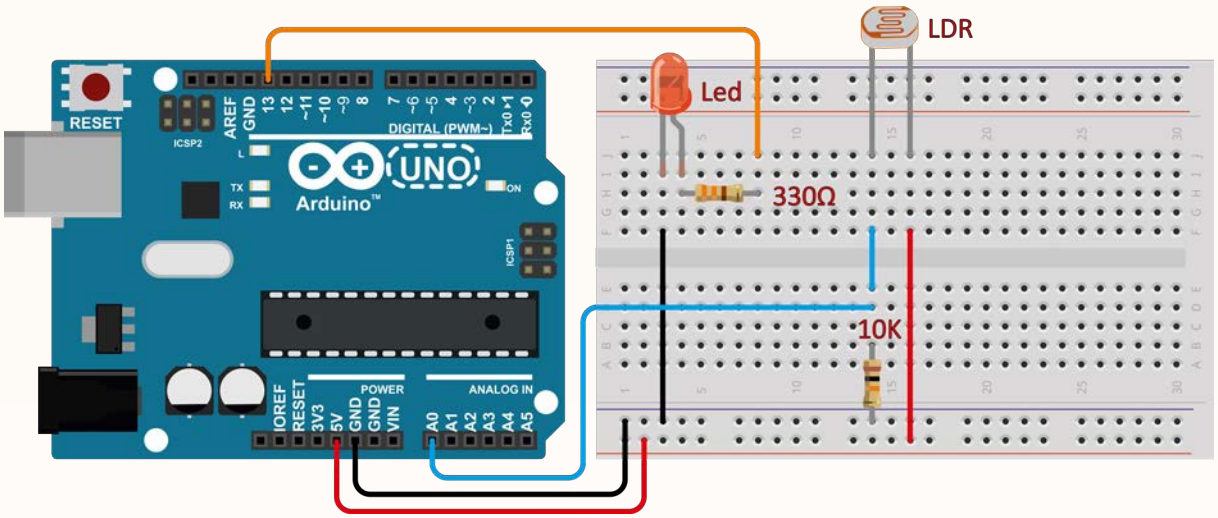

```
                                        // LM35 ve NTC sıcaklık sensör değerlerinin LCD'de okunması
#include <LiquidCrystal_I2C.h>          // I2C LCD kütüphanesi.
 LiquidCrystal_I2C lcd(0x3f,16,2);
 int NTCdeg,LMADCdeg=0;
 float LM35deg;
  void setup()
{
  lcd.begin();                          // LCD'yi başlat.
  lcd.backlight();                      // LCD arka plan ışığını aç.
}
void loop()
{
 NTCdeg  = analogRead(A0);              // NTC analog değerini A0 analog girişten oku.
 LMADCdeg= analogRead(A1);              // LM35 analog değerini A1 analog girişten oku.
 lcd.setCursor(0,0);
 lcd.print("NTC-ADC:");
 lcd.print(NTCdeg);                     // NTC sıcaklık ADC değerini LCD'ye yaz.
 lcd.setCursor(0,1);
 lcd.print("LM35:");
 LM35deg = 0.488*LMADCdeg;              // Sıcaklık hesapla ((5000/1023) x LMADCdeg)/10
 lcd.print(LM35deg);                    // LM35 sıcaklık değerini LCD'ye yaz.
 lcd.print(" C");                       // Santigrat harfini koy.
 delay(500);
 lcd.clear();
}
```

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Donanım ve programdaki değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

12. UYGULAMA

IŞIK SENSÖR DEĞERİNİN SERİ PORT ÜZERİNDEN BİLGİSAYARA GÖNDERİLMESİ

AMAÇ: **Işık sensör değerini seri port üzerinden bilgisayara göndermek.**

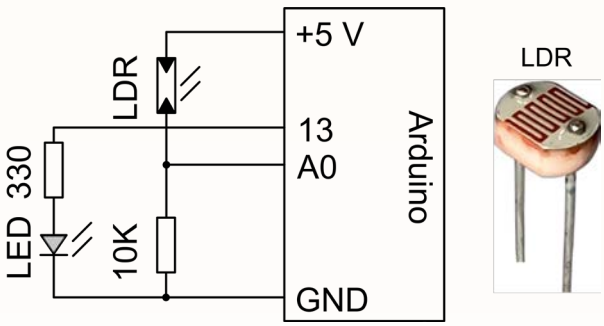


Görsel 3.33: Uygulamaya ait devre bağlantı şeması

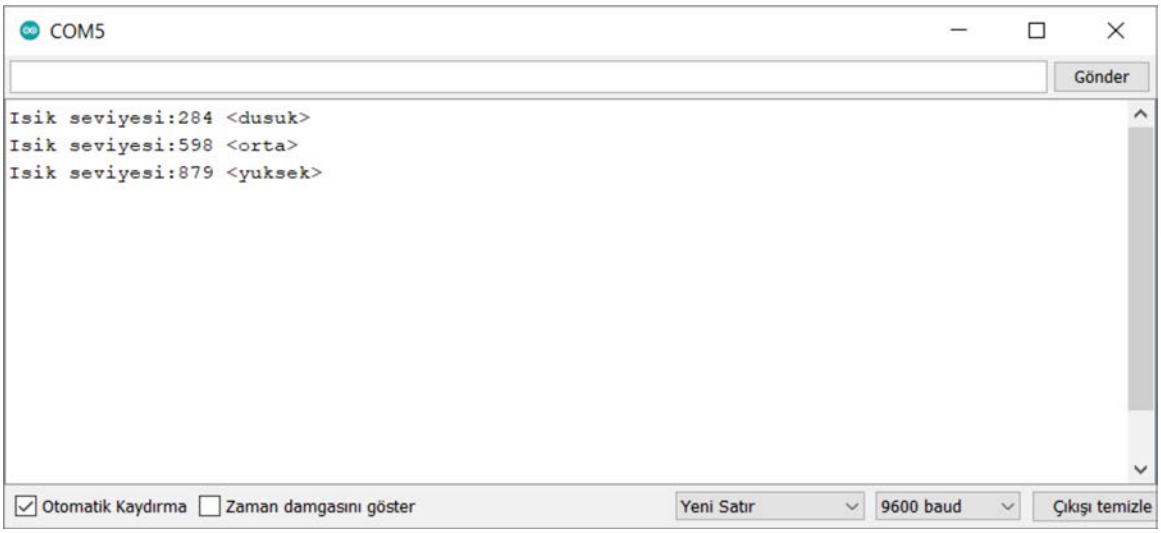
BİLGİ

Işık sensörü: Işık enerjisine göre fiziksel (direnç) veya elektriksel (fotovoltaik pil, fototransistör vs.) değişim gösteren elemandır.

LDR (**L**ight **D**ependent **R**esistor): Işığa bağımlı dirençtir. Foto direnç olarak da adlandırılır. En çok kullanılan ışık sensörüdür. Üzerine düşen ışık miktarı arttıkça direnç değeri azalır. Ortamdaki ışık yoğunluğunu ölçmek için kullanılır. Sokak lambalarında kullanılan fotosel röle içinde bu sensörden mevcuttur (Görsel 3.34).



Görsel 3.34: LDR Işık sensörü devresi



Görsel 3.35: Işık sensör bilgilerine ait seri port ekranı

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 3.33'te verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. Görsel 3.35'teki gibi seri port ekranını açınız.
5. LDR'nin bulunduğu ortamdaki ışık miktarını yavaş yavaş artırınız. Seri port ekranından değerleri ve LED'in durumunu gözlemleyiniz.
6. Mikrodenetleyici dijital girişlerinden üç tane kullanarak ön direnci (330Ω) bağlanmış üç LED bağlayınız.
7. Her bir ışık seviyesinde bir LED yanacak şekilde (düşük, orta ve yüksek kademeleri için) gerekli donanım ve program değişikliklerini yapınız. Sonucu gözlemleyiniz.
8. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM

Dosya Düzenle Taslak Araçlar Yardım

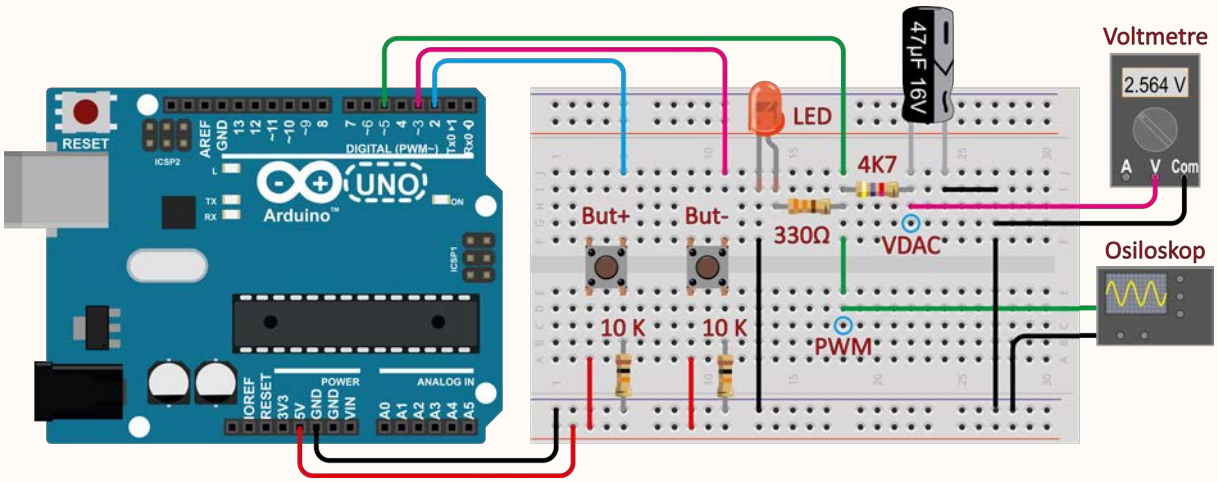
```
                                    // LDR ışık sensörü değerinin seri port üzerinden bilgisayara gönderilmesi.
                                    // Düşük ve orta seviyede LED'in yakılması, yüksek seviyede söndürülmesi.
 int LDRdeg=0;
 int Led=13;
   void setup() {                   // LED 13 no.lu pine bağlandı.
 pinMode(Led, OUTPUT);              // 13 no.lu pini çıkış olarak ayarla.
 Serial.begin(9600); }              // Seri portu 9600 baud hız için ayarla.
void loop()
{
 LDRdeg  = analogRead(A0);          // LDR analog değerini A0 analog girişten oku.
 Serial.print("Isik seviyesi:");Serial.print(LDRdeg);
   if (LDRdeg>=0 && LDRdeg<350){
                                    // LDR ADC değeri 0-350 arasındaysa seri porta "ışık seviyesi düşük" yaz.
     Serial.print(" <dusuk>"); Serial.println(); // Satırbaşı verisi.
     digitalWrite(Led, 1);          // 0-350 arasındaysa LED'i yak.
    }
    else if (LDRdeg>350 && LDRdeg<700){
     Serial.print(" <orta>"); Serial.println();
     digitalWrite(Led, 1);          // 350-700 arasındaysa LED'i yak.
    }
    else if (LDRdeg>700 && LDRdeg<1023){
     Serial.print(" <yuksek>"); Serial.println();
     digitalWrite(Led, 0);          // 700-1023 arasındaysa LED'i söndür.
    }
 delay(500);
}
```

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Donanım ve programdaki değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

13. UYGULAMA

PWM İLE BİR LED'İN PARLAKLIĞININ AYARLANMASI VE DEĞERLERİN BİLGİSAYARA GÖNDERİLMESİ

AMAÇ: PWM ile bir LED'in parlaklığını değiştirmek.



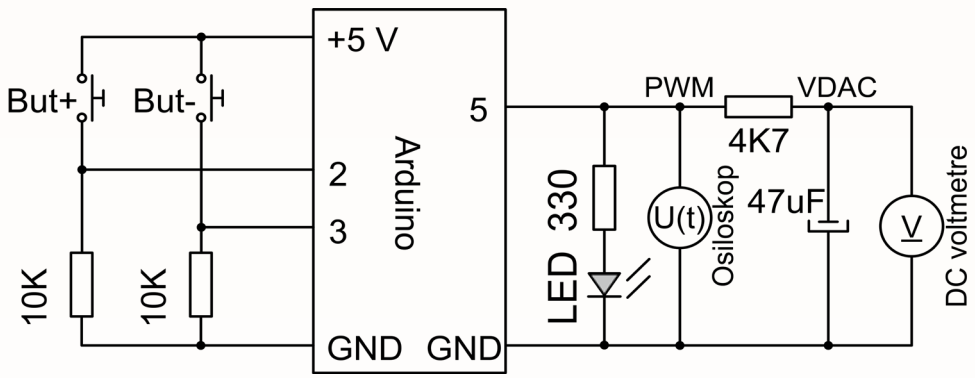
Görsel 3.36: Uygulamaya ait devre bağlantı şeması

BİLGİ

Programda PWM pininden çıkış almak için analogWrite(5,**LedPWM**); fonksiyonu kullanılır. Bu fonksiyon 5 numaralı pinden **LedPWM** değişkeni kadar görev döngüsü oluşturur.

Program yüklendikten sonra seri port ekranı açılır. Butonlara basıldığında ekranda LED yüzdesi ve PWM değeri görülür. LED parlaklık değeri %0-%100 arası yapılırken 8 bit PWM için görev döngüsünün 0-255'e çevrilmesi gerekir. Bunun için **LedPWM**=map**(LedPar,0,100,0,255);** fonksiyonu kullanılır.

Görsel 3.37'deki DAC devre şemasında, osiloskopla PWM dalga şekli gözlenebilir. Ayrıca RC (4,7K - 47µF) filtre elemanlarıyla VDAC çıkışından butonlarla dijital kontrollü 0-5 V ayarlı DAC yapılmış olur.

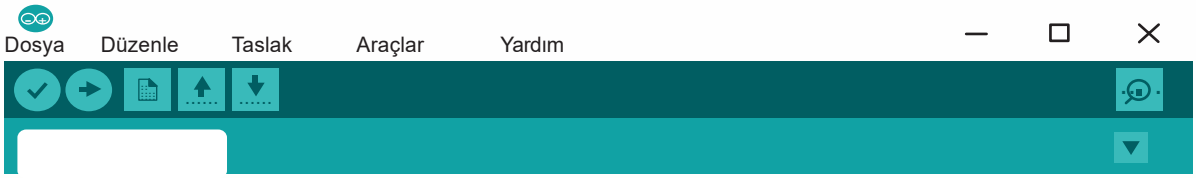


Görsel 3.37: DAC devre şeması

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 3.36'da verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. Görsel 3.37'deki PWM çıkışına osiloskop, VDAC çıkışına voltmetre bağlayınız.
5. But+ butonuna basıldığında PWM pals genişliğinin ve LED parlaklığının arttığını, But- butonuna basıldığındaysa PWM pals genişliğinin ve LED parlaklığının azaldığını gözlemleyiniz.
6. Osiloskop ve VDAC çıkışındaki voltmetreye bakarak, pals genişliğiyle doğru orantılı olarak çıkış gerilimi verdiğini gözlemleyiniz.
7. VDAC çıkışına osiloskop bağlayınız ve çıkış geriliminin pürüzsüz olduğunu gözlemleyiniz.
8. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM



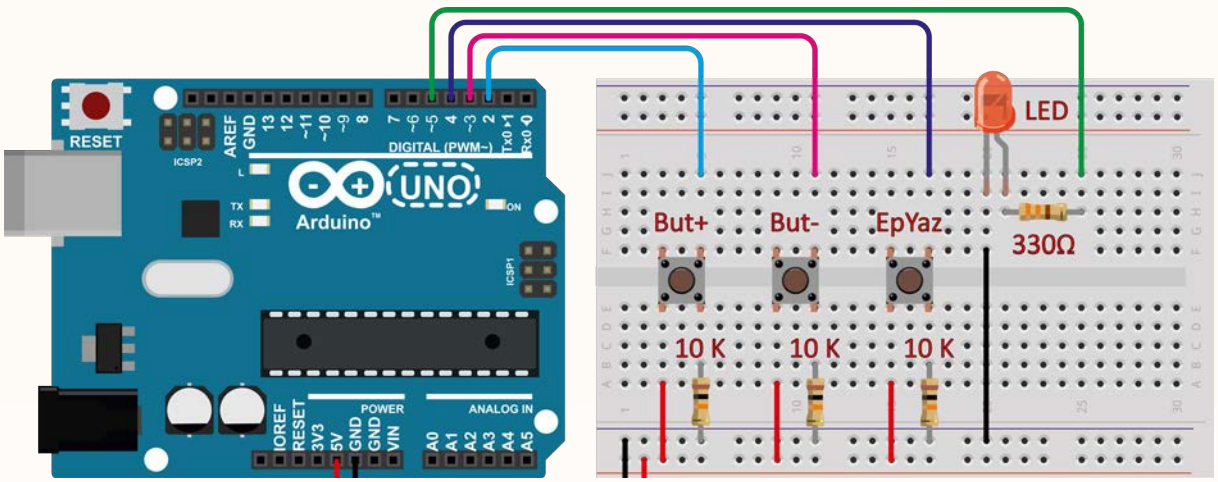

```
                        // PWM ile bir LED'in parlaklığının ayarlanması ve değerlerin PCye gönderilmesi.
 int LedPar,LedPWM=0;
 void setup()
{
 pinMode(5, OUTPUT);                    // 5 no.lu PWM pinine LED bağlandı.
 pinMode(2, INPUT);                     // 2 no.lu pine Buton artı bağlandı.
 pinMode(3, INPUT);                     // 3 no.lu pine Buton eksi bağlandı.
 Serial.begin(9600);                    // Seri portu 9600 baud hız için ayarla.
 VeriYaz();                             // Ortak kullanılan veri fonksiyonunu çağır.
}
void VeriYaz(){                         // Ortak kullanılan verileri tek fonksiyonla yaz.
  LedPWM=map(LedPar,0,100,0,255);       // 0-100 verisini map ile 0-255'e çevir.
  Serial.print("Led:% "); Serial.print(LedPar);
  Serial.print(" Led PWM:"); Serial.print(LedPWM);
  Serial.println();
  analogWrite(5,LedPWM);                // 5 no.lu çıkıştan LedPWM değeri kadar PWM üret.
  delay(50);
 }
void loop()
{
   if (digitalRead(2)==1){              // 2 no.lu pinden Buton artı değerini oku.
      LedPar++;
        if(LedPar>100){ LedPar=100; }   // 100'den büyükse 100'e sabitle.
        VeriYaz();                      // Ortak kullanılan veri fonksiyonunu çağır.
   }
   if (digitalRead(3)==1){              // 3 no.lu pinden Buton artı değerini oku.
      LedPar--;
        if(LedPar<0){ LedPar=0; }       // 0'dan küçükse 0'a sabitle.
        VeriYaz();                      // Ortak kullanılan veri fonksiyonunu çağır.
   }
}
```

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Uygulama programını doğru açıkladı.	20	
İmza :		**TOPLAM PUAN**	100	

14. UYGULAMA

PWM İLE BİR LED'İN PARLAKLIK DEĞERİNİN EEPROM'A KAYDEDİLİP OKUNMASI

AMAÇ: EEPROM'a veri kaydedip okumak.



Görsel 3.38: Uygulamaya ait devre bağlantı şeması

BİLGİ

Bellek, bilgilerin saklandığı veya okunduğu küçük depolama birimidir. Birçok mikrodenetleyicide dâhilî olarak RAM, ROM ve EEPROM bellekler bulunur. Arduino Uno kartında kullanılan mikrodenetleyici, 2 KB RAM ve 1 KB EEPROM belleğe sahiptir.

RAM: Enerji olduğu sürece okunup yazılabilen bellek türüdür. Programdaki değişkenler bu tip belleklerde tutulur.

ROM: Sadece okunabilen, yazma yapılamayan belleklerdir. Enerjisi kesilse bile tekrar verildiğinde içindeki bilgiler okunabilir (CDROM gibi).

EEPROM: Hem yazılabilir hem de okunabilir belleklerdir (Flash bellek gibi). Enerjisi gitse bile tekrar verildiğinde içindeki bilgiler okunabilir. Yazılımda saklanmak istenen bilgiler bu belleğe yazılabilir. Örneğin bu uygulamada LED parlaklık değeri dâhilî EEPROM'a kaydedilirse enerji gittiğinde veya resetlendiğinde kaydedilen bilgiler okunup parlaklık değeri son yapılan hâliyle görülür (Görsel 3.39).

Eeprom belleğe veri yazmak için "EEPROM.h" kütüphanesi eklenmelidir.

Verileri kaydetmek için EEPROM.write(kayitAdresi, kaydedilecekVeri);

Okumak için Degisken = EEPROM.read(okunacakAdres); fonksiyonları kullanılır.



Görsel 3.39: Eeprom kayıt bilgilerine ait seri port ekranı

UYGULAMAYA AİT PROGRAM

Dosya Düzenle Taslak Araçlar Yardım

```
                                  // İki buton ve PWM ile LED'in parlaklık değerini eepromа kaydet, oku.
#include <EEPROM.h>               // EEPROM fonksiyonları için kütüphane dâhil edildi.
 int LedPar,LedPWM,EpAdres=0;
 int ButArti = 2, ButEksi = 3,EpYaz=4 ;          // Butonları tanımla.
 int Led = 5;                                    // LED pinini tanımla.
                                  // EpAdres=0 olduğu için eeprom'un 0.adresine yazma ve okuma yapar.
   void setup()
{
 pinMode(Led, OUTPUT);                            // 5 no.lu PWM pinine LED bağlandı.
 pinMode(ButArti, INPUT);                         // 2 no.lu pine Buton artı bağlandı.
 pinMode(ButEksi, INPUT);                         // 3 no.lu pine Buton eksi bağlandı.
 pinMode(EpYaz,   INPUT);                         // 4 no.lu pine eeproma yaz butonu bağlandı.
 Serial.begin(9600);                              // Seri portu 9600 baud hız için ayarla.
 EepromOku();                                     // Açılışta eeproma yazılan bilgileri oku.
}
void EepromOku(){                                 // Eepromdan  bilgi okuma fonksiyonu.
  LedPar = EEPROM.read( EpAdres );                // 0. adreste bulunan bilgiyi oku.
  if(LedPar<0 || LedPar>100 ){
    LedPar=0;}                                    // LedPar 0'dan küçük veya 100'den büyükse LedPar=0 yap.
    PWMYap();                                     // PWMyap fonksiyonunu çağır.
}
void EepromYaz(){                                 // Eeproma  bilgi yazma fonksiyonu.
  EEPROM.write(EpAdres, LedPar);                  // 0. aderese LedPar değerini yaz.
  Serial.println("Bilgiler eeproma kaydedildi...");
  delay(500);
}
void PWMYap(){                                    // Ortak kullanılan verileri tek fonksiyonla yaz.
  LedPWM=map(LedPar,0,100,0,255);                 // 0-100 verisini map ile 0-255'e çevir.
  Serial.print("Led:% "); Serial.print(LedPar);
  Serial.print(" Led PWM:"); Serial.print(LedPWM);
  Serial.println();
  analogWrite(Led,LedPWM);                        // 5 no.lu çıkıştan LedPWM değeri kadar PWM üret.
  delay(50);
 }
void loop()
{
   if (digitalRead(ButArti)==1){                  // 2 no.lu pinden Buton artı değerini oku.
      LedPar++;
        if(LedPar>100){ LedPar=100; }             // 100'den büyükse 100'e sabitle.
        PWMYap();                                 // Ortak kullanılan PWMYap fonksiyonunu çağır.
   }
   if (digitalRead(ButEksi)==1){                  // 3 no.lu pinden Buton eksi değerini oku.
      LedPar--;
        if(LedPar<0){ LedPar=0; }                 // 0'dan küçükse 0'a sabitle.
        PWMYap();                                 // Ortak kullanılan PWMYap fonksiyonunu çağır.
   }
   if (digitalRead(EpYaz)==1){                    // 4 no.lu pine bağlı butona basıldığında
        EepromYaz();                              // EepromYaz fonksiyonu ile eeproma yaz.
   }
}
```

İŞLEM BASAMAKLARI

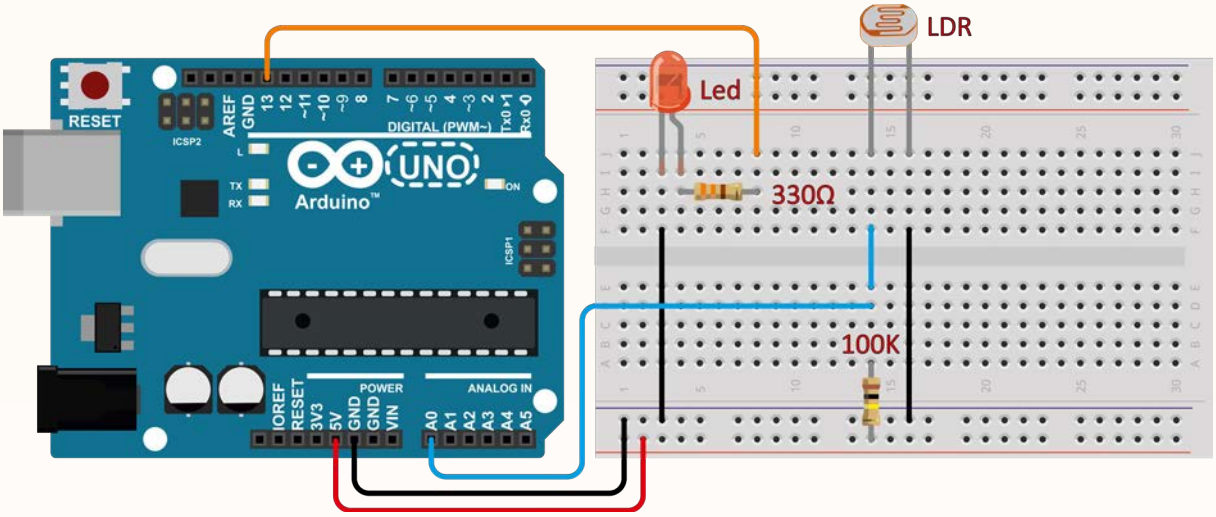
1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 3.38'de verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. Görsel 3.39'daki gibi seri port ekranını açınız.
5. But+ butonu ile LED parlaklık değerinin arttığını, But- butonu ile azaldığını seri port ekranından takip ediniz.
6. Görsel 3.39'daki gibi EepromYaz butonuna basıldığında değerlerin kaydedildiğini gözlemleyiniz.
7. Mikrodenetleyici kartın enerjisini kesiniz ve yeniden veriniz.
8. Yaptığınız son değişikliklerin saklandığını ve başlangıçta bu değerlerin okunarak seri port ekranına yazıldığını gözlemleyiniz.
9. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Uygulama programını doğru açıkladı.	20	
İmza :		**TOPLAM PUAN**	100	

15. UYGULAMA

IŞIK SENSÖRÜNE GÖRE LED'İN PARLAKLIĞININ AYARLANMASI

AMAÇ: **Işık sensörüne göre LED'in parlaklığını ayarlamak.**

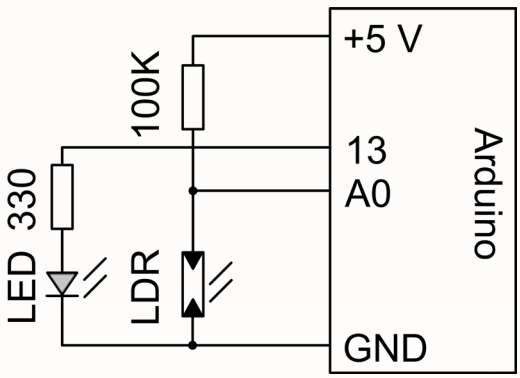


Görsel 3.40: Uygulamaya ait devre bağlantı şeması

BİLGİ

Uygulamada, LDR sensöründen okunan değere göre PWM görev döngüsünün değiştirilmesiyle LED parlaklığı değiştirilir. Görsel 3.41'deki şemaya göre, ışık miktarı artınca LDR direnci ve A0 analog girişteki gerilim azalır. Analog değer azalacağı için PWM ve LED'in parlaklık değeri de azalır. Kısacası ortam aydınlıktan karanlığa doğru geçerken ışık miktarı artacak, tersi durumdaysa ışık miktarı azalacaktır. LED yerine güç transistörlü devre kullanılarak daha güçlü otomatik aydınlatma sistemi yapılabilir.

Görsel 3.42'de uygulamaya ait LDR analog ve PWM çıkış değerleri bilgisayar seri port ekranından görülebilir.



Görsel 3.41: Uygulamaya ait devre şeması

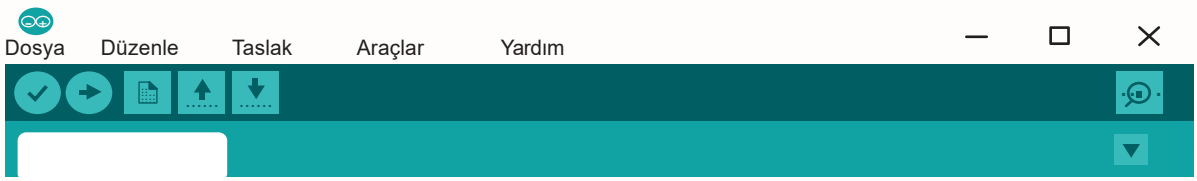


Görsel 3.42: Işık sensör bilgilerine ait seri port ekranı

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 3.40'ta verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. Görsel 3.42'deki gibi seri port ekranını açınız.
5. LDR'nin bulunduğu ortamdaki ışık miktarını yavaş yavaş artırınız.
6. Seri port ekranından değerleri ve LED'in parlaklık durumunu gözlemleyiniz.
7. Bu uygulamada ortamdaki ışık miktarı azalırken LED'in parlaklığı artar. Ortamdaki ışık miktarı artarken LED'in parlaklığını da artıran program değişikliğini yapınız ve programı tekrar yükleyiniz.
8. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM



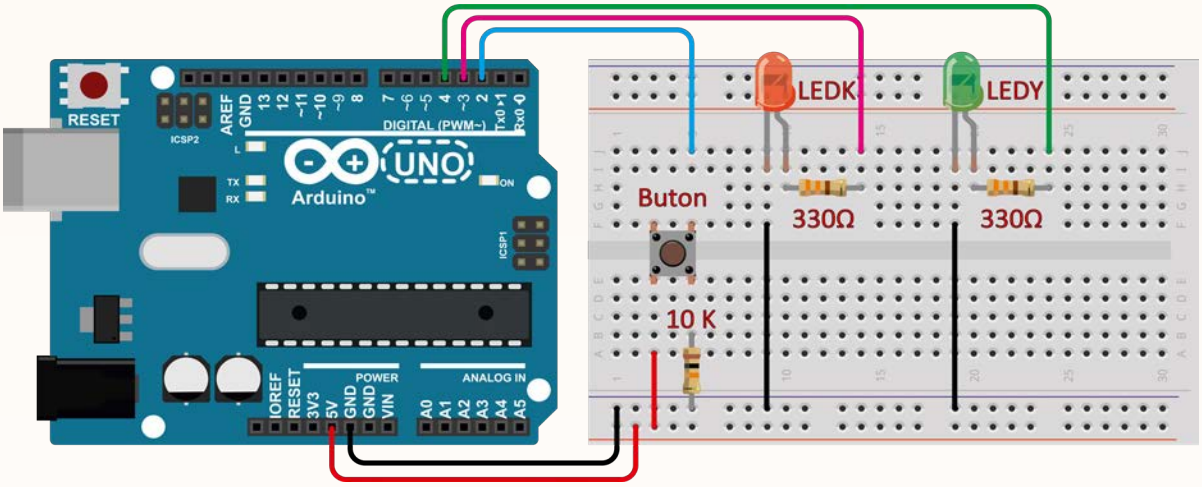

```
// Işık sensörü LDR ile LED parlaklık değerinin PWM ile değiştirilmesi.
 int Led,LDRdeg,LedPWM,seri=0;
  void setup()
{
 pinMode(Led, OUTPUT);                  // 13 no.lu pini LED için çıkış olarak ayarla.
 pinMode(5, OUTPUT);                    // 5 no.lu PWM pinine LED bağlandı.
 Serial.begin(9600);                    // Seri portu 9600 baud hız için ayarla.
}
void loop()
{
  LDRdeg  = analogRead(A0);             // LDR analog değerini A0 analog girişten oku.
  LedPWM=map(LDRdeg,0,1023,0,255);      // 0-1023 verisini map ile 0-255'e çevir.
  analogWrite(5,LedPWM);                // 5 no.lu çıkıştan LedPWM değeri kadar PWM üret.
  seri++;
  if (seri>250){                        // 250 ms aralıklarla seri porta bilgi gönder.
    seri=0;
    Serial.print("LDR: "); Serial.print(LDRdeg);
    Serial.print(" Led PWM:"); Serial.print(LedPWM);
    Serial.println();
  }
delay(1);
}
```

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Uygulama programını doğru açıkladı.	20	
İmza :		**TOPLAM PUAN**	100	

16. UYGULAMA

DIŞ KESME (INTERRUPT) KULLANILMASI

AMAÇ: Işık sensörüne göre LED'in parlaklığını ayarlamak.



Görsel 3.43: Uygulamaya ait devre bağlantı şeması

BİLGİ

Kesme (Interrupt): Mikroişlemcilerde ana program akışı devam ederken, önceliği yüksek bir sinyal geldiğinde ana program akışının kesilerek kesme fonksiyonunun çalıştırılmasıdır. Kesme fonksiyonunda yapılması gereken işlemler bittiğinde ana program kaldığı yerden devam eder. Örneğin bilgisayarda film izlendiği sırada kapı zilinin çalınmasıyla film durdurulur ve kapı açılır, tekrar gelindiğinde filme kaldığı yerden devam edilir. Burada film, ana programın işleyişi, kapı ziliyse kesme sinyaline benzetilebilir.

Dış kesmeler: Mikroişlemciye dışarıdan gelecek sinyal kaynağı ile oluşturulan kesmelerdir (buton, sensörlerden gelen sinyaller vs.). Örneğin bir optokuplör ile AC sinyalin 0 geçişleri algılanarak frekans ve periyodu ölçülebilir. Buna bağlı olarak bir triyak tetiklenerek kontrol yapılabilir.

İç kesmeler: Mikroişlemcinin kendi içinde oluşturduğu kesmelerdir (zamanlayıcı kesmesi, seri haberleşme bilgi alındı veya gönderildi kesmesi, ADC tamamlandı kesmesi vs.). Örneğin zamana bağlı fonksiyonlarda (frekans, periyot, geçen zaman ölçümü vs.) işlem yapmak için zamanlayıcı (timer) kesmesi kullanılabilir.

Arduino Uno'da dış kesme ayarlamak için kullanılacak fonksiyon şu şekildedir:

attachInterrupt (dış kesme giriş no, kesme fonksiyonun adı, kesme sinyali kenar durumu)

Dış kesme giriş no: Arduino Uno' da iki tanedir: 0. kesme girişi (2 no.lu pin) ve 1. kesme girişi (3 no.lu pin). Buton pin 2'ye bağlandığı için dış kesme giriş no=0 yapılmıştır.

Kesme fonksiyonun adı: Kesme durumunda hangi fonksiyon çalıştırılacaksa onun adı yazılır.

Kesme sinyali kenar durumu: 5 farklı durum vardır. Bu durumlar şunlardır:

LOW	Dijital pindeki gerilim 0 V olduğunda kesme oluşturur.
HIGH	Dijital pindeki gerilim 5 V olduğunda kesme oluşturur.
CHANGE	Her gerilim değişiminde kesme oluşur. Pindeki gerilim 0'dan 5 V'a yükseldiğinde veya 5 V'tan 0 V'a düştüğünde kesme oluşturur.
RISING	Yükselen kenar. Pindeki gerilim 0'dan 5 V'a yükseldiğinde kesme oluşturur.
FALLING	Düşen kenar. Pindeki gerilim 5 V'tan 0 V'a düştüğünde kesme oluşturur.

Ana program incelendiğinde başlangıçta **led_deg=0** olduğu için LED 0,5 sn. aralıklarla yanıp sönmektedir. Bu, ana program akışıdır. Kesme sinyali gelmediğinde bu akış sürekli devam eder. Herhangi bir anda butona basıldığında (dış kesme sinyali), ana program işlemini durdurup "AcilDurumKesmesi" fonksiyonunu çağırır ve orada bulunan kodları çalıştırır. Buna göre **led_deg=1** olacaktır. Yeşil LED

söndürülecek, kırmızı acil durum LED'i yanacak ve kesme programı sona ererek ana program fonksiyonuna gidecektir. **led_deg=1** olduğu için ana program yeşil LED ile ilgili işlem yapmayacaktır. Tekrar butona basıldığında (kesme sinyali geldiğinde) kesme fonksiyonuna gidip işlemlerin tersini yapacak ve başlangıçtaki çalışma şartlarına geri dönecektir.

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 3.43'te verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. LED'in 0,5 sn. aralıklarla yanıp söndüğünü gözlemleyiniz.
5. Kesme sinyali için kullanılan butona basınız ve sonuçları gözlemleyiniz.
6. Programda, kesme sinyali kenar durumunu beş farklı duruma göre değiştiriniz ve mikrodenetleyici karta yükleyiniz. Sonuçlarını değerlendiriniz.
7. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM

Dosya Düzenle Taslak Araçlar Yardım

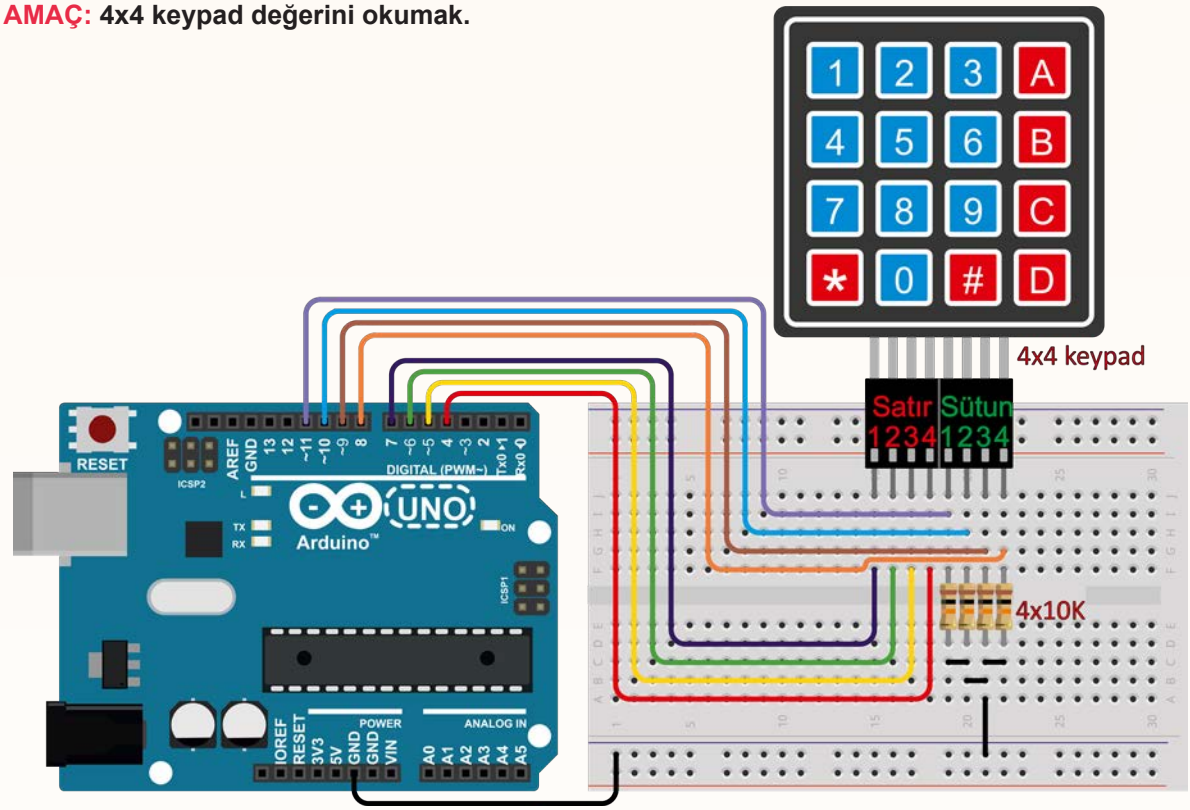
```
int Buton = 2;                          // Dış interrupt kullanılması.
int ledKirmizi = 3;
int ledYesil = 4;
boolean led_deg = 0;                    // led_deg'i mantıksal değişken olarak ayarla.
void setup() {
  pinMode(ledKirmizi, OUTPUT);
  pinMode(ledYesil, OUTPUT);
  pinMode(Buton,INPUT);
  attachInterrupt(0, AcilDurumKesmesi, RISING);
                                        // Uno 0.dış kesme, Kesme olduğunda AcilDurumKesmesi fonksiyonunu çağır.
                                        // Butona basıldığında giriş 0'dan 5 V'a yükseldiği için RISING kullanıldı.
}
void loop() {
  if (led_deg==0){                      // Bu değişken 0 olduğu sürece yeşil LED'i yak söndür.
    digitalWrite(ledYesil, 1);
    delay(500);
    digitalWrite(ledYesil, 0);
    delay(500);
  }
}
void AcilDurumKesmesi () {              // Kesme oluştuğunda çağrılacak fonksiyon.
  led_deg = !led_deg;                   // led_deg değerini değille (0 ise 1, 1 ise 0)
  digitalWrite(ledYesil, 0);            // Yeşil LED'i söndür.
  digitalWrite(ledKirmizi, led_deg); }  // Kırmızı LED'i yak.
```

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Uygulama programındaki değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

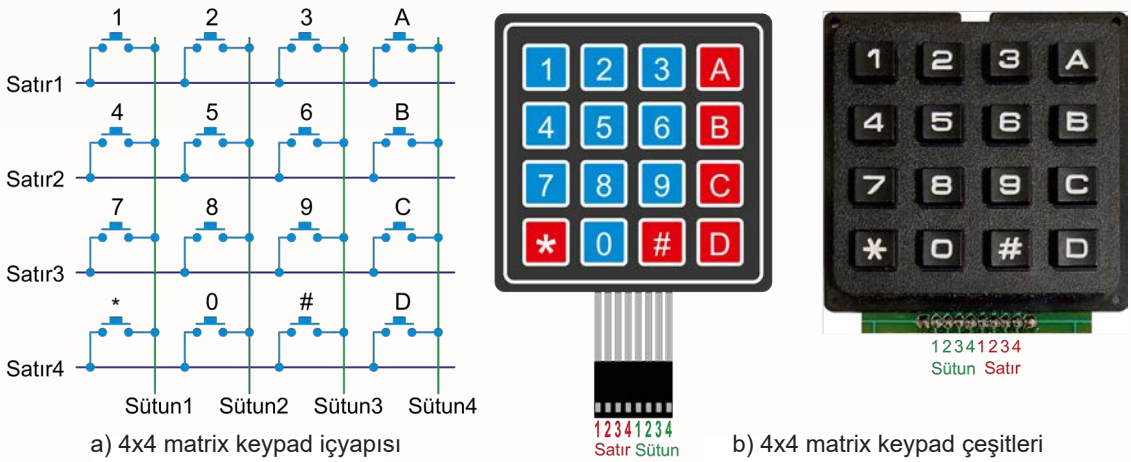
17. UYGULAMA

4X4 MATRİX KEYPAD DEĞERİNİN SERİ PORT ÜZERİNDEN BİLGİSAYARA GÖNDERİLMESİ

AMAÇ: 4x4 keypad değerini okumak.



Görsel 3.44: Uygulamaya ait devre bağlantı şeması



a) 4x4 matrix keypad içyapısı

b) 4x4 matrix keypad çeşitleri

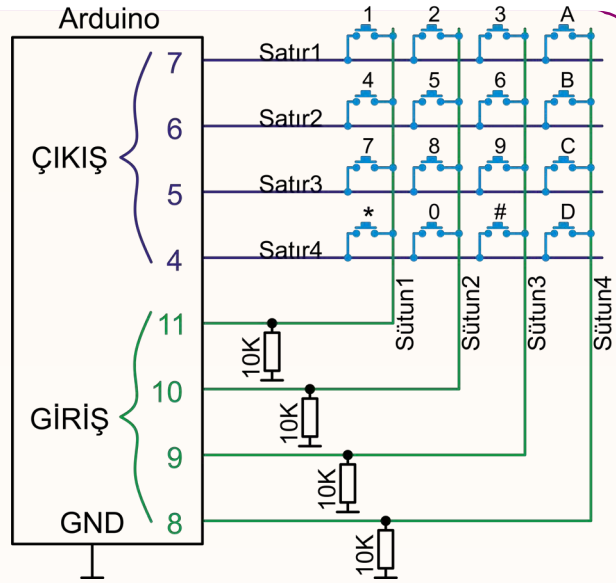
Görsel 3.45: 4x4 matrix keypad'ın içyapısı ve çeşitleri

BİLGİ

Matrix keypad: Mikroişlemcilerde giriş sayısının yeterli olmadığı veya daha az yer kaplamasının istendiği durumlarda kullanılmak için matrix şeklinde oluşturulmuş butonlar dizisidir. Görsel 3.45'te, matrix keypad'ın içyapısında butonların satır ve sütunlar şeklinde (4 satır x 4 sütun) 16 adet olarak yerleştirildiği görülür. Bu 16 buton, mikroişlemcilere doğrudan bağlansaydı 16 girişe ihtiyaç duyulurdu. Matrix bağlantıda satır ve sütun sayısı kadar giriş ve çıkış sayısı yeterli olur. 4x4 matrix bağlantıda 4 adet çıkış ve 4 adet giriş, toplamda 8 adet giriş-çıkış pini ile 16 buton değeri okunabilir. Örneğin 8x8 matrix bağlantıda 16 adet giriş- çıkış pini yeterli olurken buna karşılık 64 buton değeri okunabilir. Bilgisayarların klavye tuşları da matrix bağlantı şeklinde oluşturulmuştur.

Çalışma şekli: Matrix keypad satır veya sütunlardan biri giriş, diğeri çıkış olarak seçilir. Uygulamaya ait programda satırlar çıkış, sütunlar giriş seçilmiştir (Görsel 3.46).

İlk önce satır1'e lojik 1, diğer satırlara lojik 0 gönderilir. Sütunlar okunur. Herhangi bir butona basılmadığında sütunlardan gelen bilgi lojik 0 olacaktır. Örneğin 3 butonuna basıldığı düşünülürse satır1'den gelen lojik 1, 3 no.lu buton üzerinden sütun3'e ve buradan giriş 9'a iletilir. Bu durumda sütunlardan gelen bilgi Sutun=B00000010 olacak ve dizi değişkenden ilgili karakter alınarak seri port üzerinden bilgisayara gönderilecektir. İşlemler bu şekilde satır ve sütun tarama şeklinde gerçekleştirilir. Her satır taramaya bir sütun bilgisi okunur ve eğer butona basılmışsa ilgili buton değeri alınıp bilgisayara Görsel 3.47'deki gibi gönderilir.



Görsel 3.46: 4x4 matrix keypad devre şeması

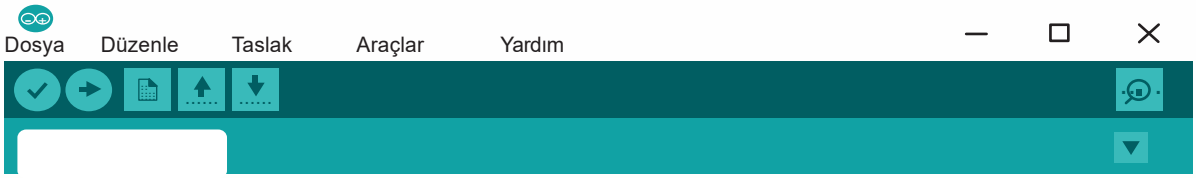


Görsel 3.47: 4x4 matrix keypad bilgilerine ait seri port ekranı

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 3.44'te verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. Görsel 3.47'deki gibi seri port ekranını açınız.
5. Tuşlara basınız. Görsel 3.47'de olduğu gibi seri port ekranından bastığınız tuşlara ait doğru bilgileri okuyunuz.
6. Satır ve sütun bağlantılarını değiştirdikten sonra tuşlara basınız ve sonuçları gözlemleyiniz.
7. Uygulamada, Görsel 3.46'daki gibi pull-down dirençleri kullanılarak sütun girişleri lojik 0'a çekilmiştir. Bu dirençleri pull-up şeklinde bağlayınız ve gerekli donanım ve program değişikliklerini yapınız.
8. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM




```
                                        // 4x4 matrix keypad uygulaması.
int Basdeg,Satir,Sutun,Say=0;
char KeyPadNo;
char KeyPadDizi[] = {
  '1','2','3','A','4','5','6','B','7','8','9','C','*','0','#','D'};
                                        // KeyPadDizi değişkeni 16 elemanlıdır. Örneğin bu dizinin 0.elemanı
                                        // KeyPadDizi[0]=1 karakteri, 8.elemanı KeyPadDizi[7]=B karakteridir.
void setup()
{
  DDRD = B11110000;                     // 7,6,5,4 no.lu pinleri satır için çıkış ayarla.
  DDRB = B00000000;                     // 11,10,9,8 no.lu pinleri sütun için giriş ayarla.
   Serial.begin(9600);                  // Seri portu 9600 baud hız için ayarla.
 }
void PCgonder(){                        // Ortak kullanılan verileri tek fonksiyonla yaz.
   if (Basdeg==1){ return ;}            // Tuşlara art arda basmayı engelle.
   Serial.print(KeyPadNo);              // Basılan tuşu seri porttan gönder.
   Serial.print(" => Tuşuna basıldı."); Serial.println();
   delay(100);
   Basdeg=1;                            // Tuşlara art arda basmayı engelleme değişkeni.
 }

void loop()                             // Ana program fonksiyonu.
{
  for (Satir=1;Satir<=4;Satir++){
    if (Satir==1){ PORTD = B10000000;Say=0;}    // 1.satır tarama
    if (Satir==2){ PORTD = B01000000;Say=4;}    // 2.satır tarama
    if (Satir==3){ PORTD = B00100000;Say=8;}    // 3.satır tarama
    if (Satir==4){ PORTD = B00010000;Say=12;}   // 4.satır tarama
    delay(1);
   Sutun=PINB & B00001111;
                // Sutun=PINB & B00001111; PINB port kaydedicisi B portundaki pin bilgilerini almak için kullanıldı.
                // B00001111 sabit sayısıyla and işlemi(maskeleme) yapılarak gerekli olan ilk 4 bit alındı.
    if (Sutun==B1000){ KeyPadNo=KeyPadDizi[Say] ; PCgonder();break;}
    if (Sutun==B0100){ KeyPadNo=KeyPadDizi[Say+1];PCgonder();break;}
    if (Sutun==B0010){ KeyPadNo=KeyPadDizi[Say+2];PCgonder();break;}
    if (Sutun==B0001){ KeyPadNo=KeyPadDizi[Say+3];PCgonder();break;}
                // Bu bölüm satır taramasına göre ilgili sütunlarda hangi tuşa basılmışsa "KeyPadDizi"
                //  değişkeninden ilgili karakteri alır. "KeyPadNo" değişkenine atar, bilgisayara gönderir
                //  ve for döngüsünden break; komutu ile çıkar.
  }
    if (Sutun==B0000){ Basdeg=0;}       // Tuşlara art arda basma yoksa engelleme değişkenini 0 yap.
}
```

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Donanım ve programdaki değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki cümlelerin başındaki boşluğa cümleler doğru ise (D), yanlış ise (Y) yazınız.

1. (…) DDRD=B11111100 ifadesinde “0.” ve “1.” bitler giriş olarak ayarlanır.
2. (…) Reed rölenin kontakları normalde kapalıdır.
3. (…) Analog dijital çeviriciye kısaca ADD denir.

B) Aşağıdaki cümlelerde boş bırakılan yerlere uygun sözcüğü yazınız.

4. Pinleri giriş-çıkış olarak ayarlayan …… komuttur.
5. Gerilim, akım gibi değerleri değişen sinyale …… sinyal denir.
6. Sıcaklık arttıkça direnç değeri azalan pasif elemana …… denir.

C. Aşağıda verilen ifadeler ve bu ifadelere karşılık gelen kavramları, ifadelerin önündeki parantezlere kavramların önündeki harflerden uygun olanları yazarak eşleştiriniz.

İFADELER		
7.	(…)	Işığa bağımlı dirençtir.
8.	(…)	Darbe genişlik modülasyonu
9.	(…)	Enerji olduğu sürece okunup yazılabilen bellektir.
10.	(…)	Dış kesme
11.	(…)	İşitsel ikaz cihazıdır.

KAVRAMLAR	
A	PWM
B	Interrupt
C	Buzzer
D	LDR
E	RAM
F	ROM

Ç) Aşağıdaki soruların doğru seçeneğini işaretleyiniz.

12. Ortamdaki canlıları algılayan sensör aşağıdakilerden hangisidir?
A) Basınç B) Ldr C) Manyetik
D) Mesafe E) Pir

13. Aşağıdaki bekletme komutlarından hangisi programın 1 saniye beklemesini sağlar?
A) delay (1) B) delay (10) C) delay (100)
D) delay (1000) E) delay (10000)

14. 10 no.lu pine lojik 1 bilgisini gönderen komut satırı aşağıdakilerden hangisidir?
A) digitalWrite(10, 1); B) digitalWrite(1, 10); C) digitalWrite(10, 0);
D) digitalWrite(0, 10); E) digitalWrite(10, 10);

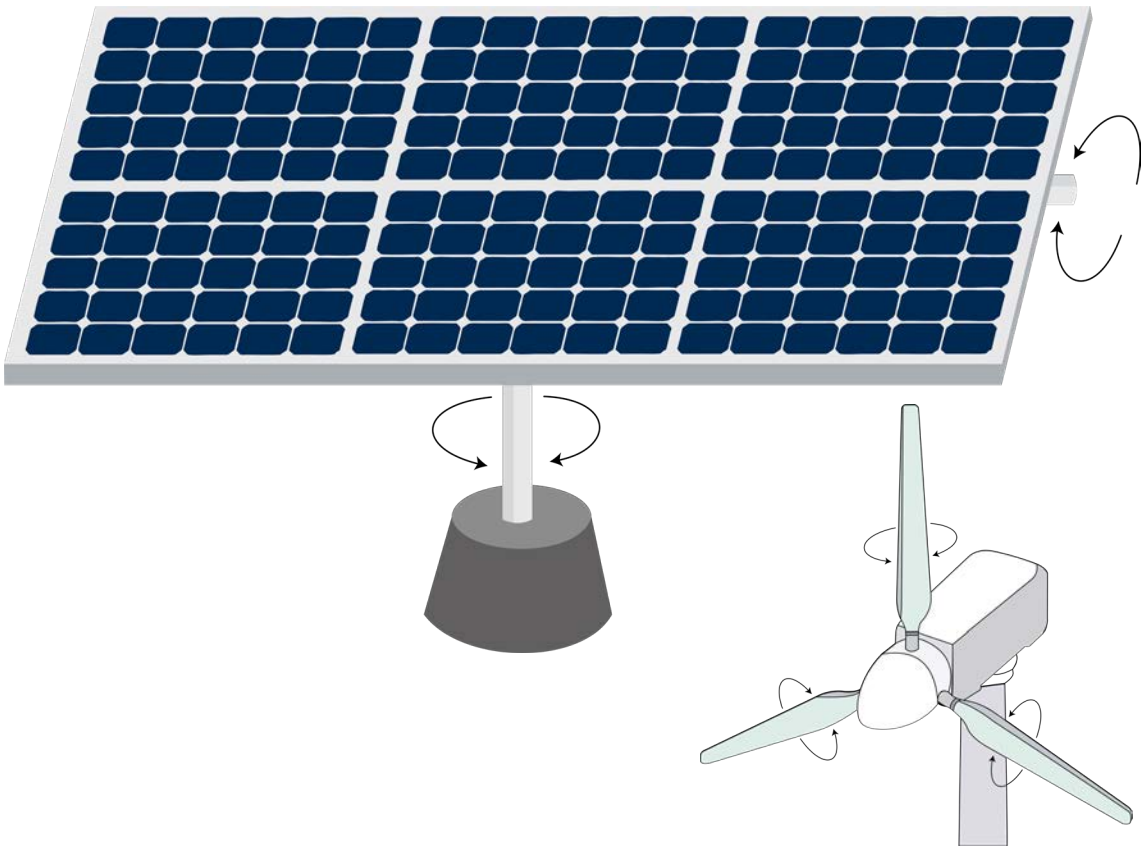
15. Aşağıdakilerden hangisi dış kesme kullanılırken pindeki gerilim 0 V’tan 5 V’a yükseldiğinde kesme oluşturur?
A) Change B) Falling C) High
D) Low E) Rising

4. ÖĞRENME BİRİMİ

MİKRODENETLEYİCİ PROJE UYGULAMALARI

KONULAR

4.1. MOTOR KONTROL UYGULAMALARI
4.2. SENSÖR UYGULAMALARI
4.3. HAREKET KONTROLÜ UYGULAMALARI



TEMEL KAVRAMLAR

Bilgisayar, breadboard, mikrodenetleyici editörü, motorlar, kodlama yazılımı, sensörler

HAZIRLIK ÇALIŞMALARI

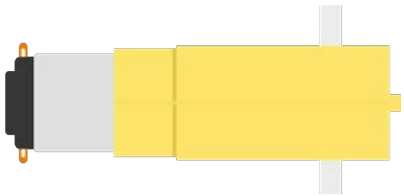
1. Doğru akım motorları hakkında bildiklerinizi arkadaşlarınızla paylaşınız.
2. Ortamdaki nem, sıcaklık, ışık gibi değişkenleri elektronik devreler nasıl algılar? Tartışınız.

4.1. MOTOR KONTROL UYGULAMALARI

Devrelerde otomatik kontrol hareketlerini yaptırmak için motorlar kullanılmaktadır. Motorlara gelen emirlere göre istenilen hareketi yaptırabilmek mümkündür. Bu şekilde motorlar ile kontrol yapılabilmektedir.

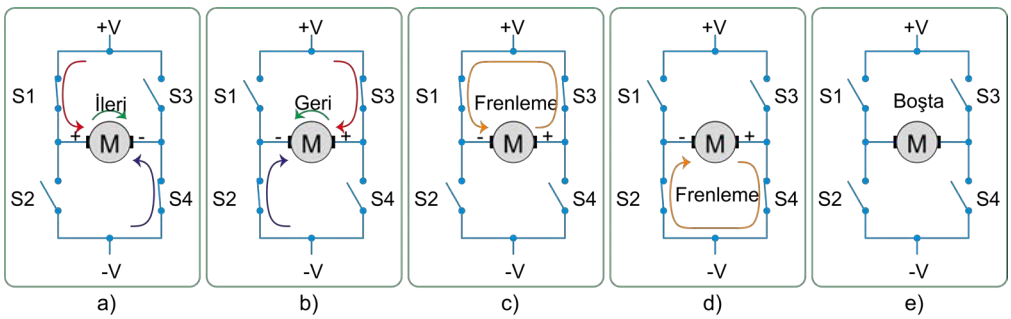
4.1.1. DC Motor ve Sürücüleri

DC motorlar, elektrik enerjisini mekanik enerjiye dönüştüren makinelerdir. Motora elektrik akımı uygulandığında içinde yer alan sargılarda manyetik alan oluşur. Motorun içinde, sabit durumdaki mıknatıslar sayesinde bu manyetik alana zıt bir manyetik kuvvet ortaya çıkar. Bu sayede bir dönme hareketi oluşur. Bu hareket süreklilik gösterir. Mikrodenetleyici uygulamalarında düşük güçlü motorlar tercih edilir (Görsel 4.1).



Görsel 4.1: Arduino projelerinde kullanılan 6V redüktörlü DC motor

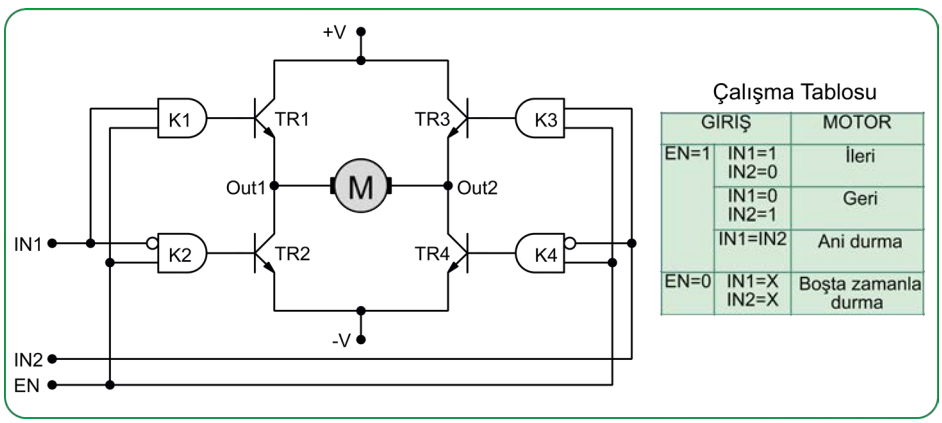
DC motorların devir yönünü değiştirmek için motora uygulanan DC gerilimin yönü değiştirilmelidir. Bunun için H köprüsü bağlantı şekli kullanılır (H harfine benzediği için bu ismi almıştır.). Görsel 4.2:a'da bulunan anahtarlardan S2-S3 açılıp, S1-S4 kapandığında motor bir yönde (ileri) döner. S1-S4 açılıp, S2-S3 kapandığında motor aksi yönde (geri) döner (Görsel 4.2:b). Motor dönerken S2-S4 açılıp S1-S3 kapanırsa (Görsel 4.2:c) veya S1-S3 açılıp S2-S4 kapanırsa (Görsel 4.2:d) motor uçları kısa devre edilir ve frenlemeyle ani durma gerçekleşir. Tüm anahtarlar açılırsa motorun enerjisi kesilir ve frenleme olmaz. Bu durumda motor yavaşlayarak durur (Görsel 4.2:e). Bu bağlantıda hiçbir zaman aynı kolda bulunan



Görsel 4.2: H köprüsü ile DC motor çalıştırılmasına ait devre şeması

S1-S2 veya S3-S4 anahtarları aynı anda kapatılmamalıdır. Aksi durumda besleme kaynağı, anahtarlar üzerinden kısa devre olur ve sisteme zarar verir.

Anahtarlar yerine transistörler kullanılarak DC motor sürücüsü gerçekleştirilebilir. Bunun için hazır entegre devreler ve modüller mevcuttur. L298, düşük güçlü motorları sürmek için kullanılan bir entegre devredir. İçerisinde iki adet H köprüsü ve kontrol devreleri bulunmaktadır. Bu nedenle iki adet DC motor veya bir adet step motor sürmek için kullanılabilir. Transistör kullanıldığı için çok hızlı açma-kapama yapabilir, bu nedenle PWM sürmeye ve frenlemeye olan tepkisi çok hızlıdır. DC motorların devir sayısı, uygulanan DC gerilimle doğru orantılı olacak şekilde artar veya azalır. PWM kullanılarak bu gerilim değeri değiştirilir ve bu nedenle hız kontrol işleminde de kullanılır. Görsel 4.3'te L298 entegre devrenin bir kanalına ait iç devre şeması görülmektedir.



Çalışma Tablosu

GİRİŞ		MOTOR
EN=1	IN1=1 IN2=0	İleri
	IN1=0 IN2=1	Geri
	IN1=IN2	Ani durma
EN=0	IN1=X IN2=X	Boşta zamanla durma

Görsel 4.3: L298 entegre devrenin bir kanalına ait H köprüsü içyapısı ve çalışma prensibi

Görsel 4.3'teki devrenin çalışması aşağıdaki gibidir.

EN girişi lojik 1(+5 V) olduğunda tüm kapıların ortak girişleri lojik 1'dir. Kapı çıkışlarını ve H köprüsünde bulunan transistörlerin sürülmesini IN1 ve IN2 belirler. EN girişi lojik 0 (0 V) olduğunda IN1 ve IN2 girişleri ne olursa olsun kapı çıkışları 0 V olur. Bu durumda Out1, Out2 çıkışları 0 V olur.

EN=lojik 1, IN1=lojik 1, IN2=lojik 0 durumu: K1 kapı çıkışı aktif olur ve TR1 iletime geçer. K2 kapı çıkışı girişinden değillendiği için pasif olur ve TR2 kesimdedir. Bu durumda Out1 çıkışında +V görülür. IN2= 0 olduğu için K3 kapı çıkışı pasif olur ve TR3 kesimdedir. K4 kapı çıkışı girişinden değil aldığı için aktif olur ve TR4 iletime geçer. Bu durumda Out2 çıkışında -V görülür. TR1, TR4 iletimde; TR2, TR3 kesimde olduğu için motor bir yönde döner.

EN=lojik 1, IN1=lojik 0, IN2=lojik 1 durumu: TR1, TR4 kesimde; TR2, TR3 iletimde olduğu için motor bir öncekinin tersi yönde döner.

EN=lojik 1, IN1=lojik 0, IN2=lojik 0 durumu: TR1, TR3 kesimde; TR2, TR4 iletimde olduğu için motor iletimde olan transistör uçlarından kısa devre edilerek frenleme yapılır.

EN=lojik 1, IN1=lojik 1, IN2=lojik 1 durumu: TR2, TR4 kesimde; TR1, TR3 iletimde olduğu için motor iletimde olan transistör uçlarından kısa devre edilerek frenleme yapılır.

EN=lojik 0, IN1=X, IN2=X durumu: Tüm transistörler kesimdedir. Bu durumda motor uçları boştadır. Hız kontrolü için girişlere PWM sinyali verilmelidir.

EN=lojik 1, IN1=PWM1, IN2=lojik 0 durumu: Her zaman TR3 kesimde, TR4 iletimdedir. TR1, PWM sinyalinin yükselen kenarında iletime, düşen kenarında kesime girer. TR2, PWM sinyalinin düşen kenarında iletime, yükselen kenarında kesime girer. Bu durumda PWM sinyalinin duty oranında, bir yönde motor hız ayarı gerçekleşir.

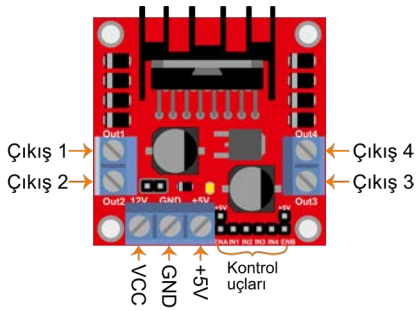
EN=lojik 1, IN1=lojik 0, IN2=PWM2 durumu: Her zaman TR1 kesimde, TR2 iletimdedir. TR3, PWM sinyalinin yükselen kenarında iletime, düşen kenarında kesime girer. TR4, PWM sinyalinin düşen kenarında iletime, yükselen kenarında kesime girer. Bu durumda PWM sinyalinin duty oranında, bir öncekinin aksi yönde motor hız ayarı gerçekleşir.

PWM sinyalinin düşen kenarında TR2 ve TR4 iletimde olduğu için motor uçları bu transistörler üzerinden kısa devre edilerek oluşan zıt EMK hızlıca deşarj olur. Böylelikle PWM daha verimli kullanılmış olur.

EN ucuna tek bir PWM kaynağı bağlanarak hız kontrolü sağlanabilir ancak PWM sinyalinin düşen kenarında motor boşta çalışacağı için verimli bir hız kontrolü gerçekleştirilemez.

L298N Motor Sürücü Kartı

DC motorları sürmek için hazırlanmış iki kanallı motor sürücü kartıdır. 24 V'a kadar olan motorları sürebilir. Her kanal 2 A akım vermektedir (Görsel 4.4).

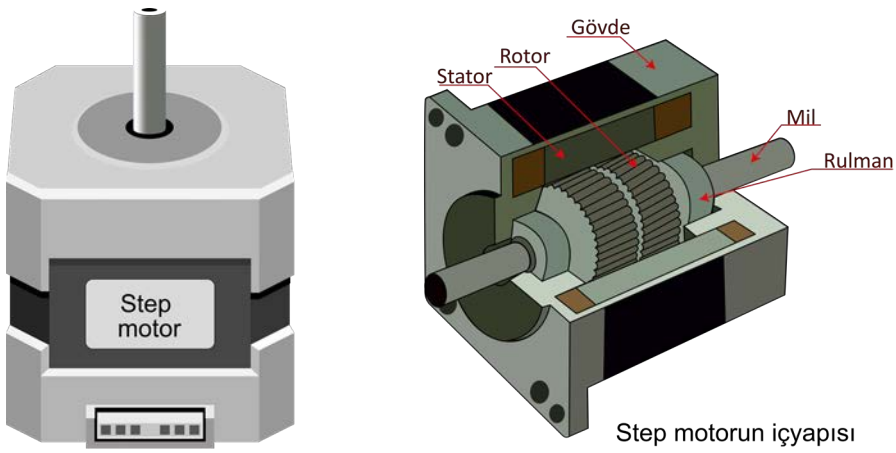


Görsel 4.4: L298N sürücü kartı pin yapısı

4.1.1. Step Motor ve Sürücüleri

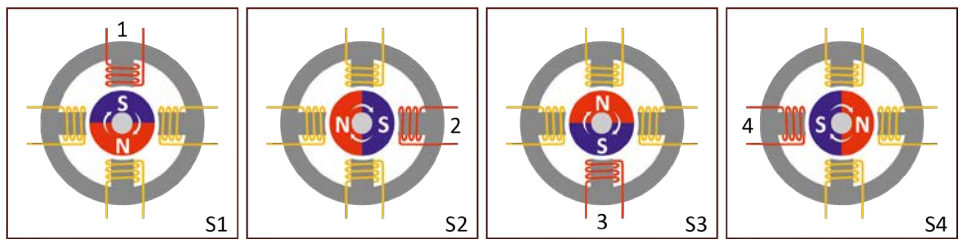
Step motor rotorunda sabit mıknatıs vardır. Statörde bulunan bir bobin sargısı enerjilendiğinde rotorda bulunan mıknatısı kendine doğru sabit bir açıyla çeker. Adım şeklinde hareket elde edilmesini sağlayan motordur.

Görsel 4.5'te step motor ve içyapısı görülmektedir.



Görsel 4.5: Step motor ve içyapısı

Step motor milinden açısal dönme hareketi elde etmek için motor bobin sargılarına sırasıyla pals uygulanır. Enerji uygulanan sargıda manyetik alan oluşur. Oluşan manyetik alan tarafından çekilen rotor, dönerek bir adım atar ve enerjili bobinin karşısına gelerek durur. Böylece istenilen sıralamayla enerjilenen sargılar rotoru döndürür (Görsel 4.6). Sargılara sırasıyla uygulanan pals frekansı motorun hızını, pals sırası ise dönüş yönünü belirler. Her bir palste bir adım attığı için hassas hareket ve kontrollerde sıkça kullanılır.



Görsel 4.6: Step motorun çalışması

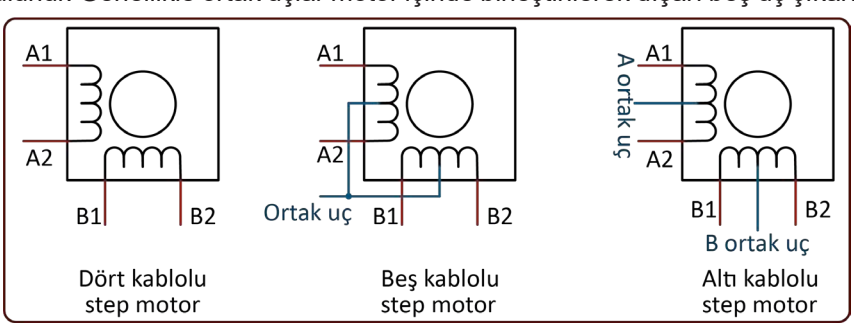
Step Motorların Sargı Bağlantısı

Step motorların iki tip sargı bağlantısı vardır: tek kutuplu (unipolar), çift kutuplu (bipolar).

Tek Kutuplu (Unipolar): Tek yönlü beslenen ve akımın tek yönde iletildiği motorlardır. Motor kablolarından birisi ortak uç olarak kullanılır. Sürülmeleri kolaydır ancak her adımda faz sarımlarının yarısı aktif olduğu için tork düşer.

Çift Kutuplu (Bipolar): Çift yönlü beslenen ve akımın her iki yönde iletildiği motorlardır. Kablo bağlantılarında ortak uç yoktur ve her faz ayrı ayrı sürülür. Sürülmeleri zordur ancak faz sarımlarının tamamı kullanıldığından torkları yüksektir.

Step motorlar 4, 5, 6 veya 8 kablolu olabilir. Bu kablolar stator bobinlerinden gelen kablolardır ve farklı renktedir. Motorun özelliğine göre stator sargıları farklı şekilde bağlanır. Bipolar motorlarda faz başına tek sargı ve iki uç vardır, ortak uç yoktur. Unipolar motorlarda ise faz başına iki sargı vardır. Her faz için üç uç bulunur. Genellikle ortak uçlar motor içinde birleştirilerek dışarı beş uç çıkarılır (Görsel 4.7).



Görsel 4.7: Dört, beş ve altı kablolu step motorlar

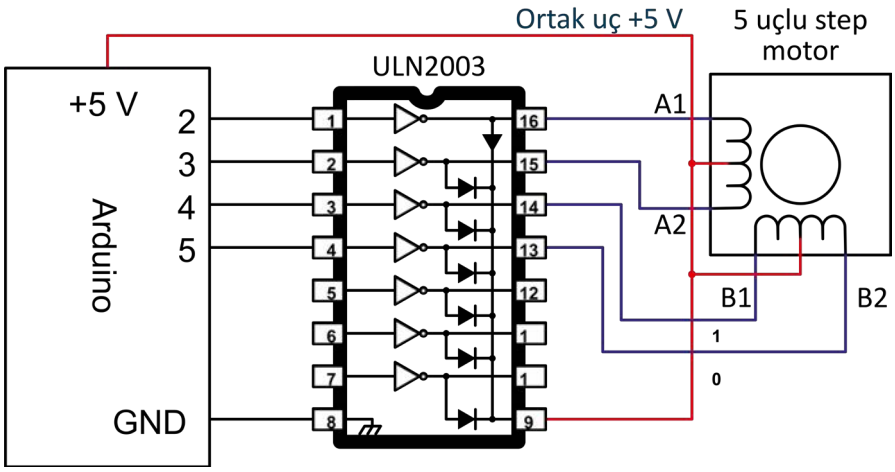
Motor Uyartım Şekilleri

Tek Faz Uyartım: Motor sargılarının sadece birinin uyartıldığı yöntemdir. Uyartım, saat yönü (CW) için 1000-0100-0010-0001 sıralamasında gerçekleşir. Yani ilk adımda sadece faz 1 enerjilenir. Daha sonra sırayla faz 2, faz 3 ve faz 4'ün enerjilenmesiyle motor saat yönünde döner. Bobinlerin enerjilenme sırası değişirse yani 0001-0010-0100-1000 sıralamasında enerji verilirse motor, saat yönünün tersi (CCW) yönde döner.

İki Faz Uyartım: Motor sargılarından ikisinin sırayla aynı anda uyartıldığı yöntemdir. Bu yöntemde rotorun tepkisi tek faz uyartıma göre daha hızlıdır ancak aynı anda iki faz bobini enerjili olduğu için çekilen güç ve buna bağlı olarak tork iki katına çıkar. Uyartım, CW için 1001-1100-0110-0011 sıralamasında gerçekleşir.

Unipolar Step Motor Sürücüsü

Unipolar step motorlarda ortak uç bulunduğu için sürülmeleri kolaydır. Ortak uç artı (+) besleme kaynağına bağlandığında kalan bobin uçlarına sırasıyla eksi (-) besleme veya 0 (GND) verilir ve motorun dönmesi gerçekleşir. Ortak uç eksi (-) besleme kaynağına bağlandığında kalan bobin uçlarına sırasıyla artı (+) besleme verilir ve motorun dönmesi gerçekleşir.

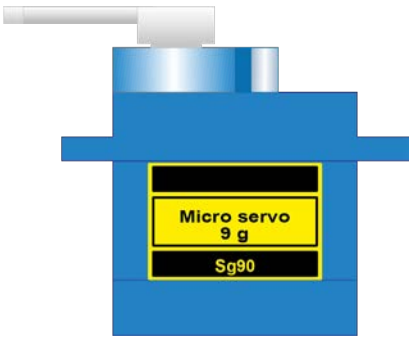


Görsel 4.8: ULN2003 mikrodenetleyici ve step motor bağlantısı

ULN2003, içinde 7 kanal transistör devresi içeren bir sürücü entegresidir. Her bir kanal 500 mA akım kapasitesine sahiptir. Çıkışlarına LED bağlanarak modül hâline getirilmiştir. Görsel 4.8'de görüldüğü üzere, bu entegrenin 1 no.lu giriş ucuna lojik 1 verildiğinde ona bağlı olan 16 no.lu çıkış ucu GND seviyesine iner. Bu durumda step motorun ortak ucundan verilen akım, A1 sargısından geçerek GND ile devresini tamamlar, enerjilenir ve motor bir adım hareket eder. Step motorun hareketine devam edebilmesi için sırasıyla girişlere bu şekilde uyartım palslerinin verilmesi gerekir. Mikrodenetleyici çıkışından bu palsler verilerek, motorun ileri veya geri adım atarak çalışması sağlanır.

4.1.1. Servo Motor ve Sürücüleri

İçerisinde DC motor, dişli mekanizması, potansiyometre, pozisyon kontrol devresi ve motor sürücüsü barındıran, pozisyon kontrolü sağlamak için tasarlanmış motorlardır. Potansiyometre dişli mekanizmasının çıkışına bağlanmıştır. Motor döndükçe potansiyometre de motor yönünde döner ve kontrol devresi motorun bulunduğu pozisyonla istenen pozisyonu karşılaştırarak, motorun o pozisyona geldiğinde durmasını sağlar. Genellikle çalışma açıları 180^0 ile sınırlandırılmıştır. 360^0 dönme açısına sahip bazı özel amaçlı servo motorlar da vardır. Görsel 4.9'da arduino uygulamalarında kullanılan servo motor görülmektedir.



Görsel 4.9: Arduino projelerinde kullanılan servo motor

Servo motorlar PWM sinyaliyle çalışmaktadır. Bu sinyalin periyodu 20 ms olacak şekilde ayarlanmalıdır. Uygulanacak darbe genişlikleri 0,5-2,5 ms (500-2500 µs) dönme açısını belirler (Görsel 4.10).



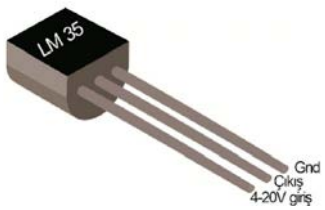
Görsel 4.10: Servo motor PWM sinyal grafiği

180^0 lik dönme açısı 2000 µs darbe genişliği ile kontrol edilir. 2000/180 ≅11 µs'lik adımlar 1^0 dönme açısına denk gelmektedir. Servo motorun 0-10^0 dönmesi için 500 µs + (10 x 11) =610 µs PWM sinyali verilmelidir.

4.2. SENSÖR UYGULAMALARI

Devrelerde herhangi bir anda herhangi bir olasılığı algılayabilmek için sensörler kullanılmaktadır.

4.2.1. LM35 Sıcaklık Sensörü



Görsel 4.11: LM35 sıcaklık sensörü

Küçük değerlerdeki sıcaklığı ölçen analog sıcaklık sensörüdür (Görsel 4.11). Sıcaklıkla orantılı olarak çıkış gerilimi artar veya azalır. LM35 sıcaklık sensörü 4-20 V gerilimle beslenmektedir. -55 ile +150 derece arasındaki sıcaklık ölçümlerini yapabilir. Her bir 1 °C'lik sıcaklık değişiminde 10 mV'luk değişim gösterir. Örneğin çıkış gerilimi 220 mV ölçüldüğünde ortam sıcaklığı=220 / 10 mV=22 ^{0}C'dir.

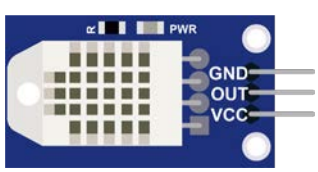
4.2.2. LDR Işık Sensörü



Görsel 4.12: LDR ışık sensörü

Üzerine düşen ışık miktarına göre direnç değişimi gösteren optik sensörlerdir (Görsel 4.12). Foto direnç olarak da adlandırılır. Ortamdaki ışık şiddeti arttığında direnci azalır, ışık şiddeti azaldığındaysa direnci artar. Günlük hayatta sokak lambaları, gece lambaları, kumanda sistemleri vb. yerlerde oldukça yaygın olarak kullanılır.

4.2.3. DHT11 Nem ve Sıcaklık Sensörü



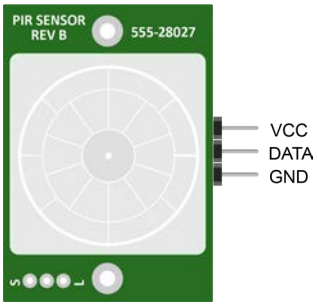
Görsel 4.13: DHT11 nem ve sıcaklık sensör modülü

Ortamın sıcaklık ve nem değerini okumak için kullanılır. DHT11, diğer sensörlere oranla ucuz fakat çözünürlüğü düşüktür. Nem ölçüm aralığı %20-%90 arasında ise doğruluğu ±%5, çözünürlüğü 1'dir. Sıcaklık ölçüm aralığı 0-50 ^{0}C arasında ise doğruluğu ±2 ^{0}C, çözünürlüğü 1 ^{0}C'dir. Görsel 4.13'te DHT11 sensör kullanılarak oluşturulmuş modül görülmektedir. Tek kablo ile haberleşme yapıldığı için sıkça tercih edilir.

4.2.4. HC-SR501 Hareket Sensörü

Ortamdaki hareketleri algılamak için kullanılan sensörlerdir. Her madde sahip olduğu ısıdan dolayı kızılötesi bir ışık yayar. Hareket sensörü ortamda

yayılan kızılötesi ışık dalgalarını içindeki piroelektrik malzemeyle birlikte elektrik enerjisine dönüştürür. Ortamda hareket algılandığında çıkışı lojik 1, hareket olmadığında ise çıkışı lojik 0 vermektedir. Genellikle hareket algıladığında yanan merdiven ve koridor lambalarında, otomatik açılıp kapanan kapılarda, güvenlik sistemleri vb. yerlerde yaygın olarak kullanılır (Görsel 4.14).

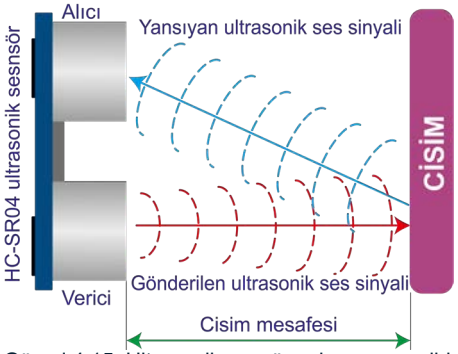


Görsel 4.14: HC-SR501 hareket sensör modülü

4.2.5. HC-SR04 Ultrasonik Sensör

Ultrasonik: İnsan kulağının duyma sınırının (20 kHz) üzerindeki ses dalgalarıdır. Ultrasonik sistemler birçok alanda kullanılmakta olup farklı isimler almaktadır. Tıpta **ultrason** (teşhis amaçlı), denizcilikte **sonar** (su altı cisim algılama), otomotivde araç park sensörleri, endüstriyel kontrol sistemlerinde cisim algılama, temizlik, plastik kaynak vs.

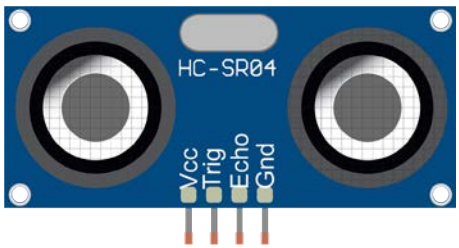
Ultrasonik sensörleri cisim algılama veya mesafe ölçümünde kullanmak için radar prensibine göre çalıştırmak gerekir. Radar çalışma prensibinde elektromanyetik dalga (radar sinyali), mesafesi ölçülecek olan cisme gönderilir. Bu sırada bir zamanlayıcı çalıştırılır ve cisimden geri yansıma (echo) beklenir. Geri yansıma sinyali geldiğinde zamanlayıcı değeri alınarak cismin uzaklığı hesaplanır. Radar sinyali ışık hızında (yaklaşık 300.000 km/s) ilerlediği için geçen süre ile ışık hızı çarpılarak bu sinyalin toplam aldığı yol bulunur (Fizikte yol=hız x zaman). Cisim uzaklığı x olduğunda gönderilen sinyal x, yansıyan sinyal de x kadar yol almış olur. Toplamda sinyal 2x yol katetmiş olur. Cismin radara olan mesafesi hesaplanırken toplam alınan yol (2x) ikiye bölünerek mesafe (x) hesaplanır.



Görsel 4.15: Ultrasonik sensör çalışma prensibi

Ultrasonik sensörlerde ışık yerine ses dalgaları kullanılır (Görsel 4.15). Ses dalgası havada yaklaşık 343 m/s hızla hareket eder. Vericiden gönderilen ses dalgasıyla birlikte bir zamanlayıcı çalıştırılır. Ses dalgalarının önünde bir engel olduğunda dalgaların bir kısmı bu engele çarparak geri yansır. Alıcı tarafından yansıma sinyali alındığında zamanlayıcı değeriyle ses hızı çarpılır ve çıkan sonuç aynı radarda olduğu gibi ikiye bölünerek cismin sensöre olan mesafesi hesaplanır.

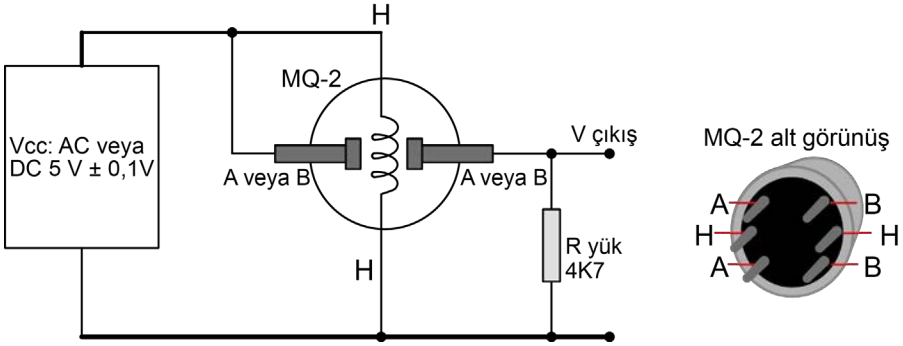
HC-SR04 ultrasonik sensör modülü 2- 400 cm arasında ölçüm yapabilme özelliğine sahiptir. Üzerinde alıcı (ultrasonik mikrofon) ve verici (ultrasonik hoparlör) vardır. Frekansları 40 kHz'de çalışmak için ayarlanmıştır (Görsel 4.16).



Görsel 4.16: HC-SR04 ultrasonik mesafe sensör modülü

4.2.6. MQ-2 Gaz Sensörü

MQ-2 gaz sensörü, ortamda bulunan yanıcı gazları (LPG, propan, hidrojen, metan vs.) algılamak için kullanılır. İçyapısında gazı algılamaya duyarlı bir tel olan SnO2 (kalay (IV) oksit) ve ısıtıcı rezistans vardır. Sensörün çıkışına yük direnci bağlanmalıdır (Görsel 4.17).



Görsel 4.17: MQ2 gaz sensörü ve bağlantı şeması

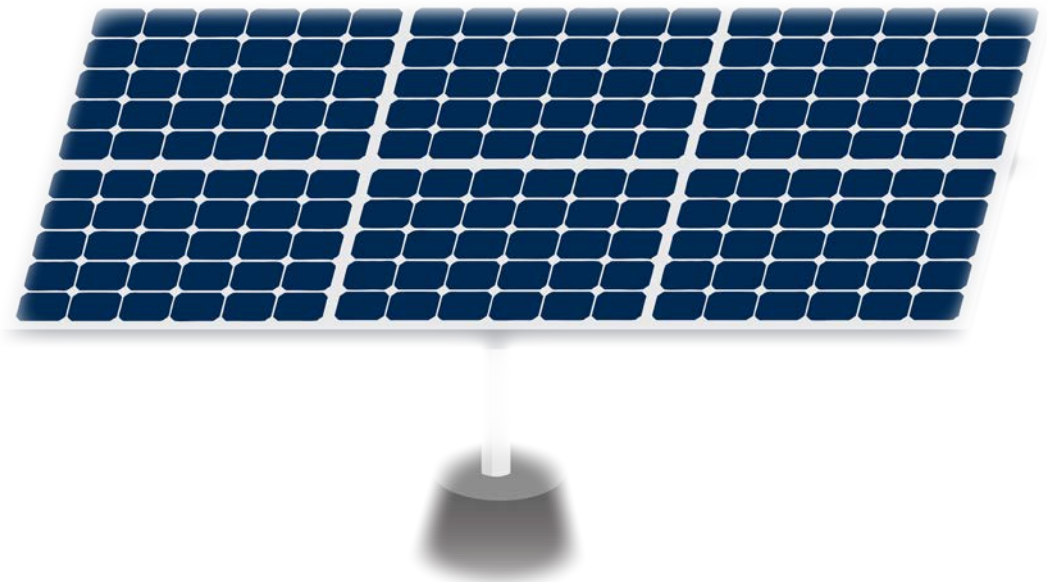
Isıtıcı rezistansın etkisiyle ısınan yanıcı gaz, SnO_2 telinin üzerinden geçerek direncinin değişmesini sağlar. SnO_2'nin direnci, temiz havada yüksek bir değere sahiptir. Ortamda yanıcı gaz yoğunluğu arttığında SnO_2'nin direnç değeri düşer ve üzerinden geçen akım artar. Yük direnci ile bu değişim 0,2-5 V arası gerilime çevrilir. Bu gerilim değeri ortamdaki gaz yoğunluğu ile doğru orantılı şekilde değişir.

4.3. HAREKET KONTROLÜ UYGULAMALARI

Günümüzde makinelerin otomatik çalışması için mikrodenetleyicilerle birçok farklı hareket kontrol işlemi yapmak mümkündür. Yapılacak işleme göre sensörlerden alınan veriler mikrodenetleyicilerle değerlendirilmekte, genellikle motorlar yardımıyla hareket kontrolü yapılmaktadır. Örneğin güneş paneli veya rüzgâr türbini açılarının otomatik değiştirilmesi gibi.

4.3.1. Güneş Panelleri

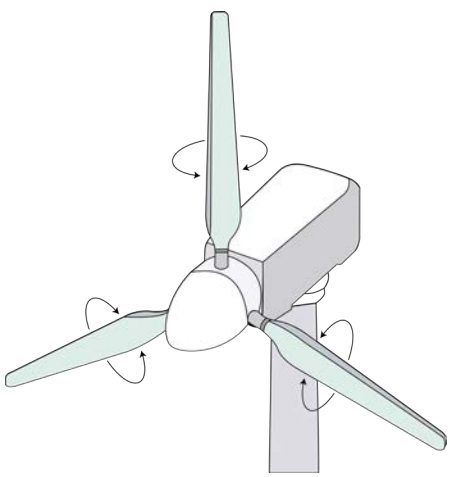
Fotovoltaik panellerin güneş ışınlarını dik olarak alabilmesi için paneller mikrodenetleyiciler ile kontrol edilebilmektedir. Mikrodenetleyiciler, güneş ışığının yönü değiştiğinde panelleri güneşe yönlendirebilmek için motorları çalıştırır ve uygun açıya geldiğinde motorları otomatik olarak durdurur. Böylece paneller günün her saatinde otomatik olarak güneş ışığına yönlenir ve güneş panellerinden alınan verim artar. Fotovoltaik panellerin bu şekilde güneş ışığını yakalamasını sağlayan sistemlere **güneş takip sistemi** denilmektedir (Görsel 4.18).



Görsel 4.18: Fotovoltaik panellerin güneş takip sistemi

4.3.2. Rüzgâr Türbinleri

Rüzgâr türbinlerinin kanat açısını (pitch) değiştirmek için çoğunlukla mikrodenetleyiciler kullanılır. Rüzgâr hızının türbin kanatlarına zarar verecek düzeye gelmesi durumunda veya türbine bakım yapılacağı zaman mikrodenetleyiciye verilen komutlara göre kanatlar kendi etrafında döndürülmektedir. Mikrodenetleyiciler sensörlerden veya kontrol biriminden gelen bu komutları işler, kanatları çevirebilmek için motorları çalıştırır. Kanat açıları yeterli seviyeye geldiğinde motorları durdurur (Görsel 4.19).

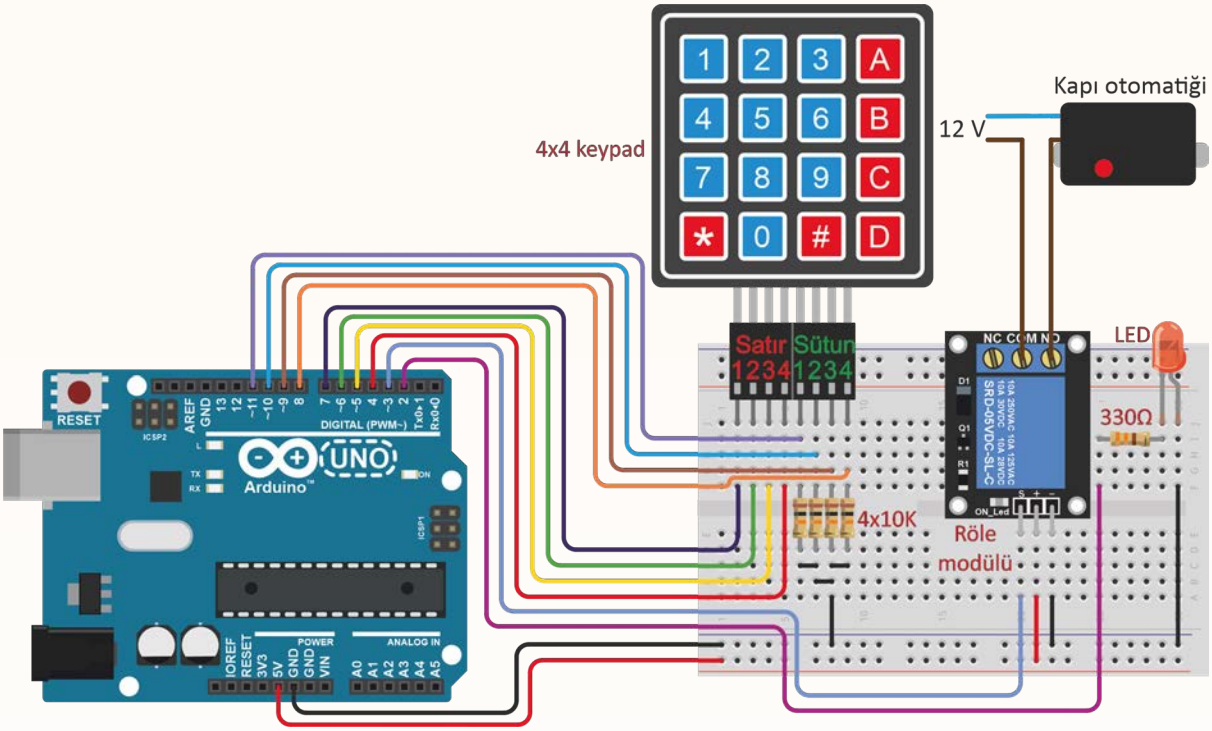


Görsel 4.19: Rüzgâr türbini kanat açısı değiştirme sistemi

1. UYGULAMA

4X4 MATRİX KEYPAD İLE ŞİFRELİ KİLİT UYGULAMASI

AMAÇ: **4x4 keypad ile şifreli kilit yapmak.**

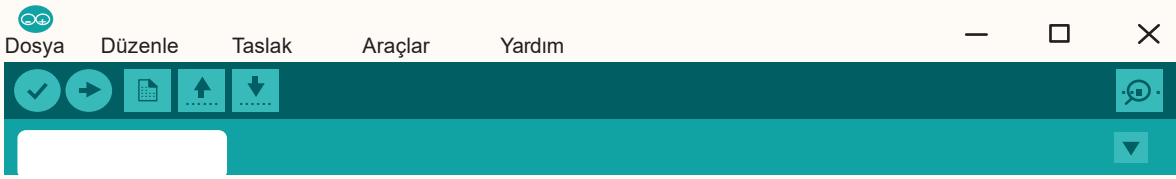


Görsel 4.20: Uygulamaya ait devre bağlantı şeması



Görsel 4.21: Uygulamaya ait seri port ekranı

UYGULAMAYA AIT PROGRAM




```
                                    // 4x4 matrix keypad şifreli kilit uygulaması.
char Sifre[] = {"1234"};            // Şifrenin saklandığı dizi.
char SifKar[4];                     // Karşılaştırma için girilen şifre dizisi.
char KeyPadDizi[] = {"123A456B789C*0#D"}; // Keypad dizisi.
char KeyPadNo;
int Basdeg,Satir,Sutun,Say,Kont,SifSira,SifDogru=0;
void setup()
{
  DDRD = B11111100;                 //7,6,5,4 no.lu pinleri satır, 2 no.lu pini LED, 3 no.lu pini röle için çıkış ayarla.
  DDRB = B00000000;                 // 11,10,9,8 no.lu pinleri sütun için giriş ayarla.
  Serial.begin(9600);               // Seri portu 9600 baud hız için ayarla.
}
void SifKontrol(){                  // Şifre kontrol ve PC'ye gönderilmesi.
   if (Basdeg==1){ return ;}  // Tuşlara art arda basmayı engelle.
   Serial.print(KeyPadNo);          // Basılan tuş bilgisini seri porttan gönder.
   Serial.print(" => Tuşuna basıldı."); Serial.println();
   delay(100);
   Basdeg=1;                        // Tuşlara art arda basmayı engelleme değişkeni.
   if (KeyPadNo=='*'){ SifDogru=0;SifSira=0; return;}          // Sıfırla.
   SifKar[SifSira]=KeyPadNo;        // Basılan tuşu SifKar dizisinin elemanına sırayla aktar.
   SifSira++;
   if (SifSira>3){                  // 4 defa tuşa basıldıysa aşağıdaki kodları uygula.
     SifDogru=0; SifSira=0;
     for (Kont=0;Kont<4;Kont++){
       if (SifKar[Kont]!=Sifre[Kont]){          //SifKar ile Sifre dizisini karşılaştır. Farklıysa aşağıdaki kodları uygula.
       Serial.print(" Şifre yanlış."); Serial.println();
        PORTD = B00000100; delay(1000);       //Röle modülü OFF, kırmızı LED ON
        PORTD = B00000000;                    // Röle modülü ve kırmızı LED OFF
        return; // Yanlış giriş olduğu için fonksiyondan çık.
       }
     }
        // Karşılaştırmalar doğruysa aşağıdaki kodları uygula.
        Serial.print(" => Şifre doğru."); Serial.println();
        PORTD = B00001000; delay(1000);       // Röle modülü 1 sn. ON
        PORTD = B00000000;                    // Röle modülü ve kırmızı LED OFF
   }}
void loop()                                   // Ana program fonksiyonu.
{
  for (Satir=1;Satir<=4;Satir++){
    if (Satir==1){ PORTD = B10000000;Say=0;}          // 1.satır tarama
    if (Satir==2){ PORTD = B01000000;Say=4;}          // 2.satır tarama
    if (Satir==3){ PORTD = B00100000;Say=8;}          // 3.satır tarama
    if (Satir==4){ PORTD = B00010000;Say=12;}         // 4.satır tarama
    delay(1);
   Sutun=PINB & B00001111;
    if (Sutun==B1000){ KeyPadNo=KeyPadDizi[Say] ; SifKontrol();break;}
    if (Sutun==B0100){ KeyPadNo=KeyPadDizi[Say+1];SifKontrol();break;}
    if (Sutun==B0010){ KeyPadNo=KeyPadDizi[Say+2];SifKontrol();break;}
    if (Sutun==B0001){ KeyPadNo=KeyPadDizi[Say+3];SifKontrol();break;}
  }
    if (Sutun==B0000){ Basdeg=0;}   // Tuşlara art arda basma yoksa engelleme değişkenini 0 yap.
}
```

İŞLEM BASAMAKLARI

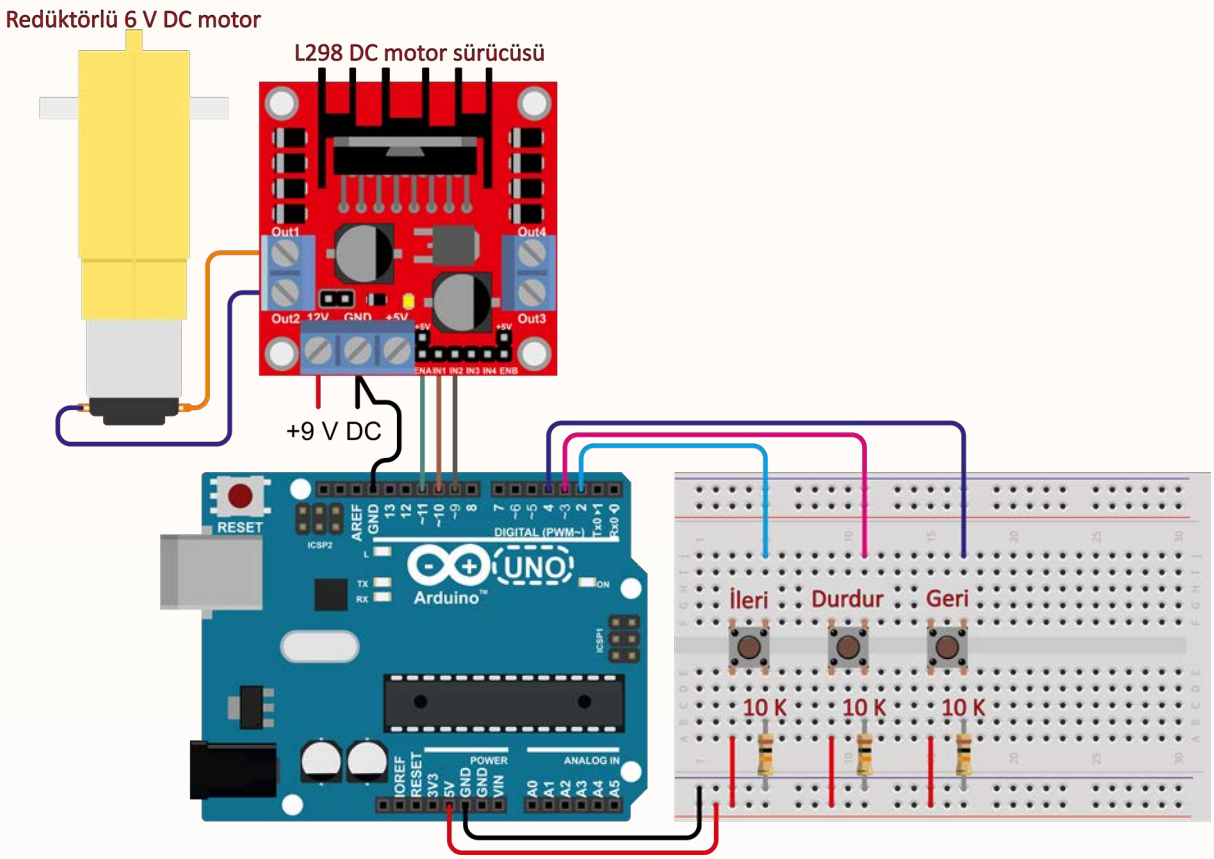
1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 4.20'de verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. Seri port ekranını açınız.
5. Görsel 4.21'de olduğu gibi bastığınız tuşlara ait bilgileri seri port ekranından okuyunuz.
6. Şifreyi yanlış girdiğinizde kırmızı LED'in 1 saniye yandığını gözleyiniz.
7. Şifreyi doğru girdiğinizde röle modülünün 1 saniye çalıştığını ve kapı kilidinin açıldığını gözleyiniz.
8. Yeni şifre için gerekli program değişikliğini yapınız ve programı yeniden yükleyiniz.
9. Yeni şifreyi kullanarak kapı kilidinin açıldığını gözleyiniz.
10. Şifreyi 5 basamaklı yapmak için gerekli program değişikliğini yaparak programı yeniden yükleyiniz.
11. 5 basamaklı yeni şifreyi kullanarak kapı kilidinin açıldığını gözleyiniz.
12. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Donanım ve programda gerekli değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

2. UYGULAMA

DC MOTOR SÜRÜCÜSÜ İLE MOTORUN İLERİ-GERİ ÇALIŞTIRILMASI VE DURDURULMASI

AMAÇ: **DC motor sürücüsünü kullanmak.**

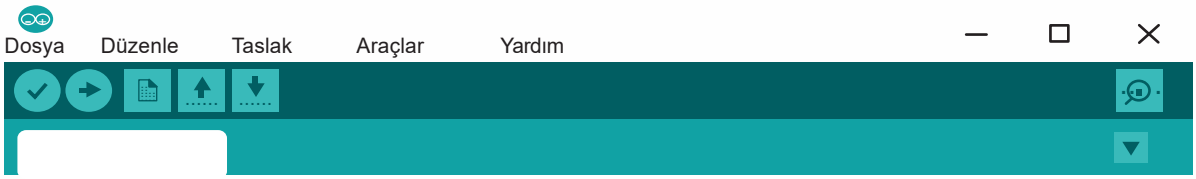


Görsel 4.22: Uygulamaya ait devre bağlantı şeması

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 4.22'de verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. İleri, durdur ve geri butonlarına basarak motorun ileri-geri döndüğünü gözleyiniz.
5. Motor ileri dönerken geri butonuna basınız ve motorun yön değiştirmediğini gözleyiniz.
6. İleri butonuna bastığınızda motorun 1 sn. ileri gitmesini, 1 sn. durmasını, sonra 1 sn. geri dönerek durmasını sağlayan gerekli program değişikliğini yapıp programı yeniden yükleyiniz.
7. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM



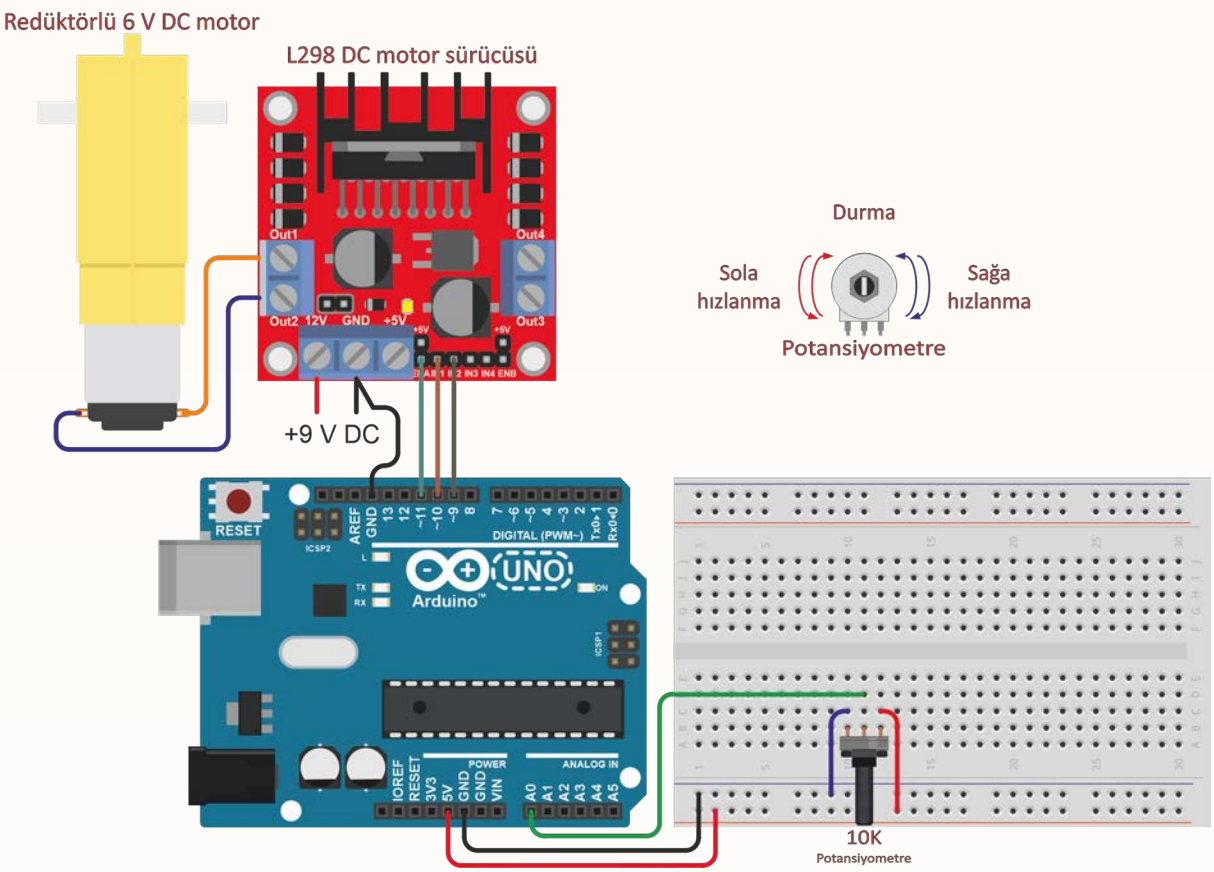

```
// DC motor sürücüsü ile motorun ileri-geri çalıştırılması ve durdurulması.
int Motkoruma=0;                // Motor ters yön koruma değişkeni.
int MotEN1=11;                  // L298 ENA girişi 11 no.lu pine bağlandı.
int MotIN1=10;                  // L298 IN1 girişi 10 no.lu pine bağlandı.
int MotIN2=9;                   // L298 IN2 girişi 9  no.lu pine bağlandı.
int Butileri=2;                 // İleri butonu  2 no.lu pine bağlandı.
int Butdurdur=3;                // Durdur butonu 3 no.lu pine bağlandı.
int Butgeri=4;                  // Geri butonu   4 no.lu pine bağlandı.
 void setup()
{
 pinMode(MotEN1,OUTPUT); pinMode(MotIN1,OUTPUT); pinMode(MotIN2,OUTPUT);
 pinMode(Butileri,INPUT); pinMode(Butgeri,INPUT); pinMode(Butdurdur,INPUT);
 digitalWrite(MotEN1, 0);
}
void loop()
{
  if (Motkoruma==0) {           //Durdur butonuna basılırsa tersine dönüşü aktif yap.
    if (digitalRead(Butileri)==1) {
      digitalWrite(MotEN1,1);digitalWrite(MotIN1,1);digitalWrite(MotIN2,0);
      Motkoruma=1;              // Motor bir yönde dönerken tersine dönüşü engelle.
    }
    if (digitalRead(Butgeri)==1) {
      digitalWrite(MotEN1,1);digitalWrite(MotIN1,0);digitalWrite(MotIN2,1);
      Motkoruma=1;              // Motor bir yönde dönerken tersine dönüşü engelle.
    }
  }
  if (digitalRead(Butdurdur)==1) {
    Motkoruma=0;                // Motor ters yön korumayı kaldır.
    digitalWrite(MotEN1,1);digitalWrite(MotIN1,1);digitalWrite(MotIN2,1);
    delay(100);
    digitalWrite(MotEN1,0);
  }
}
```

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Programda gerekli değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

3. UYGULAMA

DC MOTOR SÜRÜCÜSÜ VE POTANSİYOMETRE KULLANILARAK MOTORUN SAĞA VEYA SOLA HIZ AYARLI ÇALIŞTIRILMASI

AMAÇ: **DC motoru hız ayarlı çalıştırmak.**

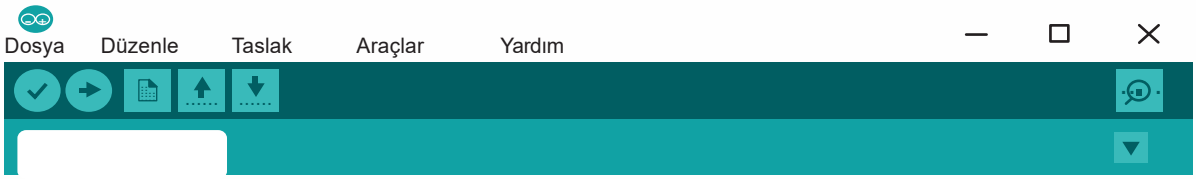


Görsel 4.23: Uygulamaya ait devre bağlantı şeması

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 4.23'te verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. Potansiyometreyi orta konuma alınız ve motorun durduğunu gözleyiniz.
5. Potansiyometreyi sola doğru çeviriniz ve motorun sola doğru hızlandığını gözleyiniz.
6. Potansiyometreyi orta konuma alınız ve motorun durduğunu gözleyiniz.
7. Potansiyometreyi sağa doğru çeviriniz ve motorun sağa doğru hızlandığını gözleyiniz.
8. Potansiyometre yerine iki adet LDR kullanarak ışık yönüne doğru hareket edecek sistem için gerekli donanım ve program değişikliklerini yapınız.
9. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM




```
                                    // DC motor sürücüsü ve potansiyometre kullanılarak
                                    // motorun sağa veya sola hız ayarlı çalıştırılması.
 int Potdeg,MotPWM=0;
 int MotEN1=11;                     // L298 ENA1 girişi 11 no.lu pine bağlandı.
 int MotIN1=10;                     // L298 IN1 girişi  10 no.lu PWM pinine bağlandı.
 int MotIN2=9;                      // L298 IN2 girişi  9  no.lu PWM pinine bağlandı.

  void setup()
{
 pinMode(MotEN1,OUTPUT); pinMode(MotIN1,OUTPUT);
 pinMode(MotIN2,OUTPUT);
 digitalWrite(MotEN1, 0);
}
void loop()
{
                                     // Potansiyometrenin orta konumunda motor duracak (460-563 arası).
                                     // Potansiyometre sola çevrildiğinde motor sola hızlanacak (460-0 arası).
                                    // Potansiyometre sağa çevrildiğinde motor sağa hızlanacak (563-1023 arası).
 Potdeg = analogRead(A0);           // Potdeg = A0. Okunan potansiyometre ADC değeri.

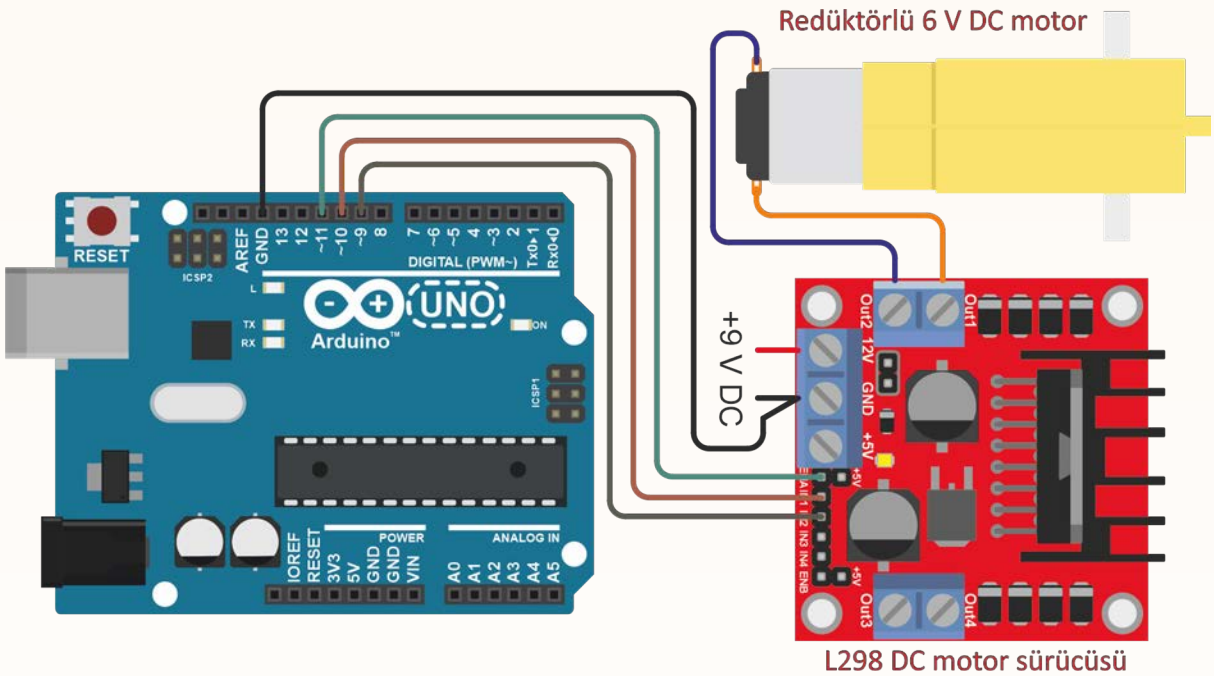
  if (Potdeg>=0 && Potdeg<=460){                                  // 460-0 arası sola doğru hızlanma.
    MotPWM=map(Potdeg,460,0,0,255);                               //460-0 verisini map ile 0-255'e çevir.
    analogWrite(MotIN2,MotPWM); digitalWrite(MotIN1,0);           //Sol PWM hızı.
    digitalWrite(MotEN1,1);
  }
  if (Potdeg>460 && Potdeg<=563){                                  // 460-563 arası durma.
    digitalWrite(MotEN1,1);digitalWrite(MotIN1,1);
    digitalWrite(MotIN2,1);
  }
  if (Potdeg>563 && Potdeg<=1023){                                // 563-1023 arası sağa doğru hızlanma.
    MotPWM=map(Potdeg,563,1023,0,255);//563-1023 verisini 0-255'e çevir.
    analogWrite(MotIN1,MotPWM); digitalWrite(MotIN2,0); //Sağ PWM hızı.
    digitalWrite(MotEN1,1);
  }
}
```

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Donanım ve programda gerekli değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

4. UYGULAMA

DC MOTORUN BİLGİSAYAR KONTROLLÜ İLERİ-GERİ ÇALIŞTIRILIP DURDURULMASI

AMAÇ: DC motoru bilgisayarla kontrol etmek.



Görsel 4.24: Uygulamaya ait devre bağlantı şeması

BİLGİ

Bilgisayarla kontrol işlemi için haberleşme portları kullanılır. Seri port en çok kullanılan haberleşme portudur. Daha önceki uygulamalarda dış ortamdan (mikrodenetleyici) bilgisayara veriler gönderilmiştir. Herhangi bir cihaz kontrolü için bilgisayardan dış ortama veri göndermek gerekir. Bilgisayardan basit kontrolleri yapmak için arduino ide seri port ekranı yeterlidir.

Bilgisayardan kontrol için seri port fonksiyonları kullanılır. Bu fonksiyonlar şunlardır:

Serial.available(); Seri porttan herhangi bir verinin gelip gelmediğini kontrol eder.

Değişken = Serial.read(); Seri porttan okunan veriyi değişkene aktarır.

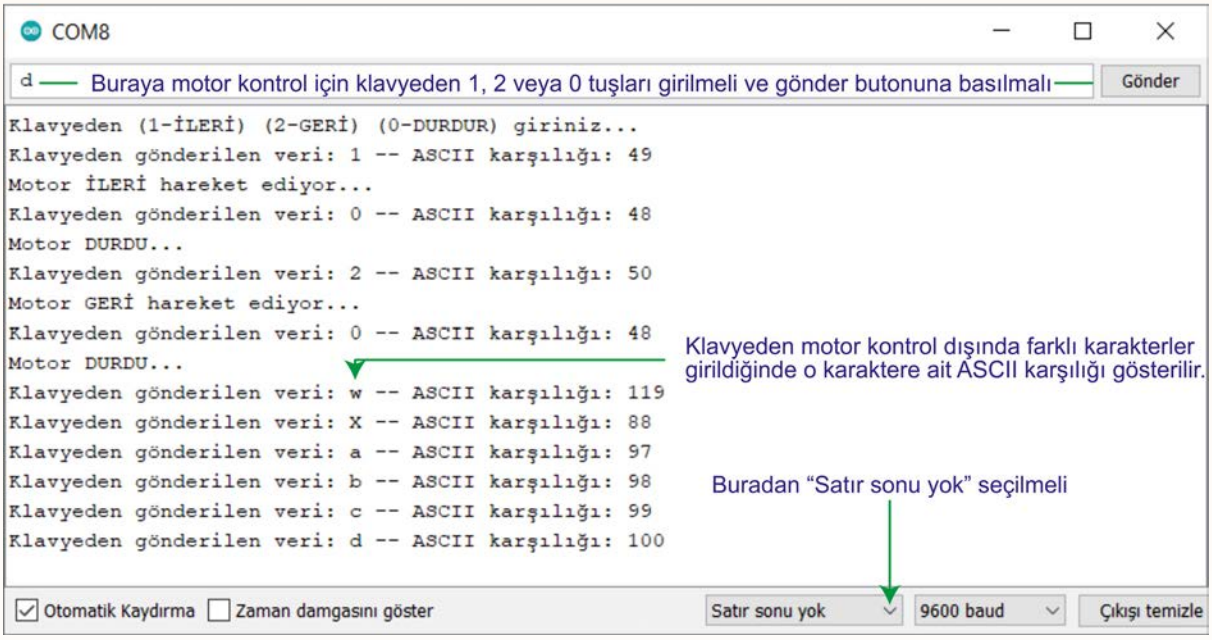
Örnek:

```
if (Serial.available()) {          // Seri port veri kontrolü yap.
  Veri = Serial.read();            // Seri porttan gelen veriyi oku.
  Serial.println((char)Veri);      // Alınan veriyi char'a dönüştür.
}
```

Bu örnekte seri porttan herhangi bir veri alınıp alınmadığı kontrol edilir. Veri alındığında Serial.read() fonksiyonuyla bir değişkene aktarılır ve o değişken mikrodenetleyici tarafından kullanılır.

Bilgisayarda her bir karakterin sayısal kod karşılığı vardır (ASCII tablosu). Örneğin a harfi = 97, 0 rakamı=48, 1 rakamı=49, 2 rakamı=50 gibi. Karakterlerin sayısal karşılığı kelimelerin alfabetik sıralanmasında oldukça yarar sağlar. Örneğin "cab" kelimesinin alfabetik sıralanması için kelimeyi oluşturan harflerin sayısal karşılığı 99,97,98 olacaktır. Rakamlar küçükten büyüğe doğru 97,98,99 olarak sıralandığında yeni kelimenin alfabetik karşılığı "abc"; büyükten küçüğe doğru (99,98,97) sıralamada ise kelimenin alfabetik karşılığı "cba" olacaktır. Seri porttan gelen veriler, karakter karşılığı sayısal kod olarak alındığında (char) **değişken** şeklinde karaktere çevrilmelidir. Kontrol ve karşılaştırma işlemi

yapılırken karakter tipi değişken tek tırnak işaretiyle kullanılır (if (**SeriPortVeri**=='1')). Uygulamada klavyeden girilen her bir karakterin ASCII karşılığı da verilmiştir.



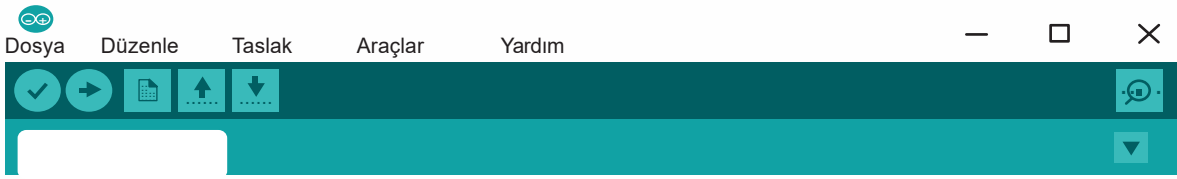
Görsel 4.25: DC motorun bilgisayar kontrollü çalıştırılmasına ait seri port ekranı

Uygulamada seri port ekranı açılmalıdır. Görsel 4.25'te belirtilen ayarlamalar ve motor kontrolü için bilgisayar klavyesinden 1 ileri, 2 geri, 0 durdurma tuşlarının girileceği yer belirtilmiştir. Bu giriş yapıldıktan sonra "gönder" butonuna basılarak bilgisayardan basılan karakter mikrodenetleyiciye gönderilmiş olur. Mikrodenetleyici gelen verileri karşılaştırarak kontrol işlemini gerçekleştirir.

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 4.24'te verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. Motor sürücüsüne enerji veriniz.
5. Görsel 4.25'teki gibi seri port ekranını açınız.
6. Bilgisayar klavyesinden 1 tuşuna basınız ve motorun ileri yönde döndüğünü gözleyiniz.
7. Bilgisayar klavyesinden 0 tuşuna basınız ve motorun durduğunu gözleyiniz.
8. Bilgisayar klavyesinden 2 tuşuna basınız ve motorun geri yönde döndüğünü gözleyiniz.
9. Seri port ekranında bilgisayar klavyesinden bastığınız her tuşun mikrodenetleyici tarafından tekrar bilgisayara gönderildiğini gözleyiniz.
10. Bilgisayar klavyesinden farklı karakterler gönderiniz ve sonucu gözleyiniz.
11. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM



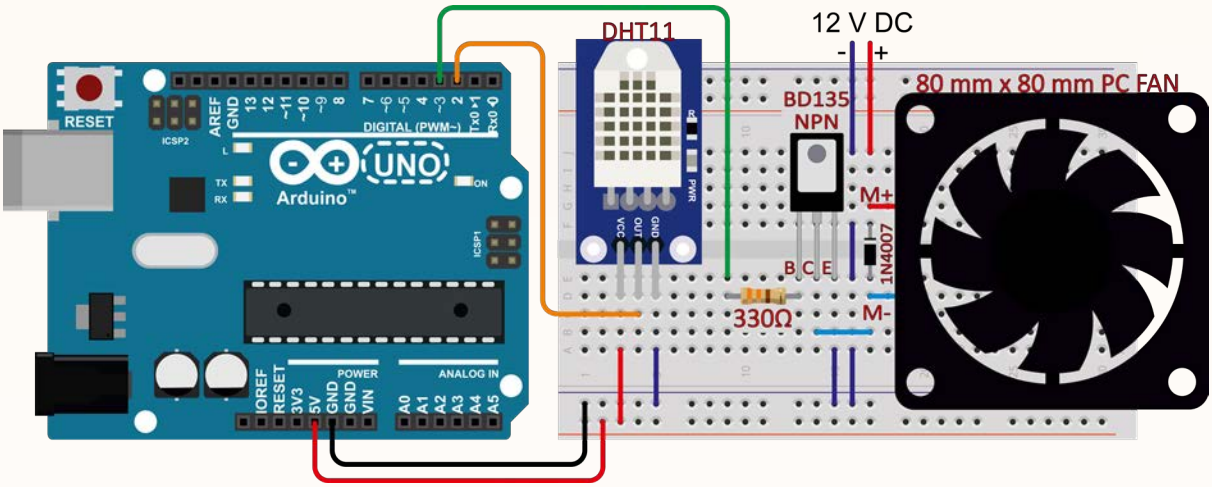

```
int SeriPortVeri,Motkoruma=0;          // DC motorun bilgisayar kontrollü ileri-geri çalıştırılıp durdurulması.
int MotEN1=11;                         // L298 ENA1 girişi 11 no.lu pine bağlandı.
int MotIN1=10;                         // L298 IN1  girişi 10 no.lu PWM pinine bağlandı.
int MotIN2=9;                          // L298 IN2  girişi 9  no.lu PWM pinine bağlandı.
void setup()
{
pinMode(MotEN1,OUTPUT); pinMode(MotIN1,OUTPUT);  pinMode(MotIN2,OUTPUT);
digitalWrite(MotEN1, 0);
Serial.begin(9600);                    // Seri portu 9600 baud hız için ayarla.
Serial.println("Klavyeden (1-İLERİ) (2-GERİ) (0-DURDUR) giriniz...");
}
void loop()
{
 if (Serial.available()) {                    // Seri port veri kontrolü yap.
                                              // herhangi bir veri gelirse aşağıdaki işlemleri yap.
   SeriPortVeri = Serial.read();              // Seri porttan gelen veriyi oku.
   Serial.print("Klavyeden gönderilen veri: ");
   Serial.println((char)SeriPortVeri);        // Alınan veriyi char'a dönüştür.
   Serial.print(" -- ASCII karşılığı:: ");
   Serial.println(SeriPortVeri);              // Alınan veriyi ASCII olarak göster.
 }
 if (Motkoruma==0) {                          //Durdur butonuna basılırsa tersine dönüşü aktif yap.
   if (SeriPortVeri=='1') {                   // Seri porttan 1 verisi gelirse motor ileri.
     digitalWrite(MotIN1,1); digitalWrite(MotIN2,0);
     digitalWrite(MotEN1,1);
     Motkoruma=1;                             // Motor bir yönde dönerken tersine dönüşü engelle.
     Serial.println("Motor İLERİ hareket ediyor... ");
   }
   if (SeriPortVeri=='2') {                   // Seri porttan 2 verisi gelirse motor geri.
     digitalWrite(MotIN1,0);digitalWrite(MotIN2,1);
     digitalWrite(MotEN1,1);
     Motkoruma=1;                             // Motor bir yönde dönerken tersine dönüşü engelle.
     Serial.println("Motor GERİ hareket ediyor... ");
   }
 }
  if (SeriPortVeri=='0') {                    // Seri porttan 0 verisi gelirse motoru durdur.
   Motkoruma=0;                               // Motor ters yön korumayı kaldır.
   digitalWrite(MotIN1,1);digitalWrite(MotIN2,1);digitalWrite(MotEN1,1);
   delay(100);  digitalWrite(MotEN1,0);
   Serial.println("Motor DURDU... ");
   SeriPortVeri=' ';                          // SeriPortVeri değişkenini boşalt.
 }
}
```

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Donanım ve programda gerekli değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

5. UYGULAMA

ORTAM NEM DEĞERİNE GÖRE ÇALIŞAN FAN VE SICAKLIK GÖSTERGE UYGULAMASI

AMAÇ: **Sıcaklık ve nem sensörünü çalıştırmak.**



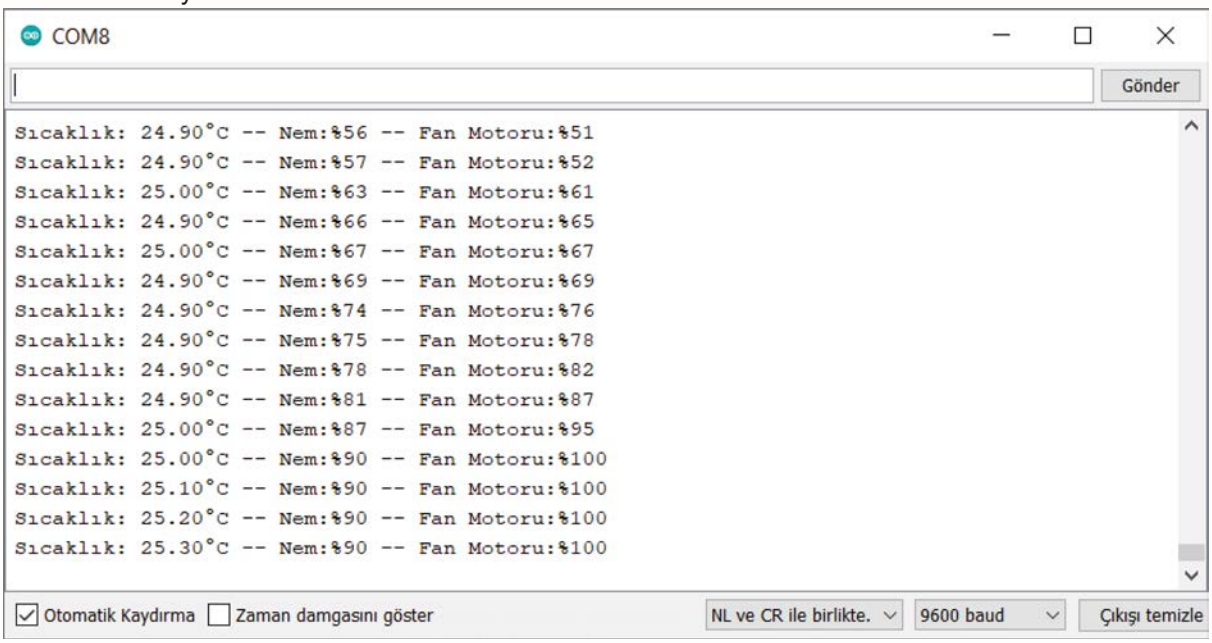
Görsel 4.26: Uygulamaya ait devre bağlantı şeması

BİLGİ

Uygulamada DHT11 nem ve sıcaklık sensörü kullanılmıştır. Mikroişlemci, sensör verilerini okumak istediğinde out pinini çıkış olarak yönlendirir ve lojik 1 yapar. Daha sonra lojik 0, lojik1, lojik 0 yaptıktan sonra out pinini giriş olarak yönlendirir. Bu durumda DHT11 kendisinden veri istendiğini anlayarak 40 bit veri gönderir. Bu verinin içinde 8 bit sıcaklık ve 8 bit nem bilgisi ile kontrol bitleri yer alır. Bu 40 bit verinin içinden nem ve sıcaklık bilgisi alınıp kullanılır. Bu işlem için hazır bulunan kütüphaneler kullanılabilir. DHT sensor library-1.0.0 kullanım kolaylığı açısından tercih edilmiştir.

Uygulamada motor sürücüsü yerine BD135 güç transistörü kullanılmıştır. Nem değerine göre oluşturulan PWM sinyali bu transistörün base ucuna girilerek fanın, nem oranında hız ayarlı çalışması sağlanmıştır.

Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.



Görsel 4.27: Sıcaklık ve nem sensör bilgilerine ait seri port ekranı

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 4.26'da verilen devre kurulumunu gerçekleştiriniz.
3. Sistemi kapalı bir kutu içine kurunuz. Fanı, içerideki havayı dışarıya atacak şekilde yerleştiriniz.
4. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
5. Seri port ekranını açınız.
6. Görsel 4.27'deki ekran görüntüsüne benzer sensör ve motor değerlerini görünüz.
7. Fanı, nemli bir bezle sensör üzerinde nem oluşturacak şekilde yaklaştırınız. Nem oranında fan motorunun hız artışını gözleyiniz.
8. Fan motorunu sıcaklık değerine (örneğin 20-50 ^{0}C) göre çalıştırmak için gerekli program değişikliğini yaparak programı yeniden yükleyiniz ve gözlemleyiniz.
9. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM

Dosya Düzenle Taslak Araçlar Yardım

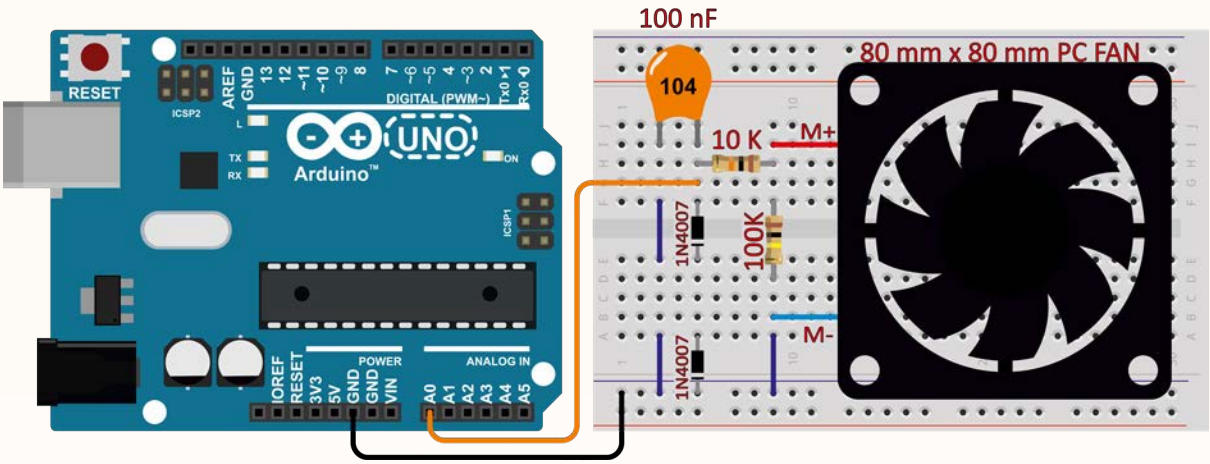
```
// Ortam nem değerine göre çalışan fan ve sıcaklık gösterge uygulaması.
#include "DHT.h"                          // DHT_sensor_library-1.0.0 yüklenecek.
#define DHTPIN 2                          // Dht11 sıcaklık ve nem sensörü pin 2'ye bağlandı.
#define DHTTYPE DHT11                     // DHT11 tip sıcaklık ve nem sensörü seçimi.
DHT dht(DHTPIN, DHTTYPE);                 // Sıcaklık ve nem sensörü fonksiyon tanımı.
int FanMotor=3;                           // Fan motor kontrolü 3 no.lu PWM pinine bağlandı.
int Sicak,Nem,MotPWM,MotPwmYuzde=0;       // Değişken tanımlamaları.
void setup() {
  Serial.begin(9600);
  Serial.println("DHT11 SICAKLIK VE NEM SENSÖR UYGULAMASI");
  dht.begin();                            // DHT11 sensörünü ölçüme hazırla.
}
void loop()
{
  delay(2000); // Dht11 sensörü, ölçümü 2 sn. aralıklarla yapar.
  float Sicak = dht.readTemperature();    // Sıcaklık bilgisini ondalıklı oku.
  int Nem = dht.readHumidity();           // Nem bilgisini tamsayı oku.
  if (Nem>90){Nem=90;}                    // Nem %90'dan büyükse 90'a sabitle.
  MotPWM=map(Nem,20,90,0,255);            // DHT11 %20-90 arası nem ölçümü yapar.
                                          // Nem %20-90 arası değeri map ile 0-255'e çevir.
  MotPwmYuzde=map(MotPWM,0,255,0,100);
                                          // Motor yüzde gösterimi için 0-255 pwm verisini map ile 0-100'e çevir.
  analogWrite(FanMotor,MotPWM);
                                          // Fan motorunu nem değerine göre pwm ile hız ayarlı çalıştır.
  Serial.print("Sıcaklık: "); Serial.print(Sicak); Serial.print("°C");
  Serial.print(" -- Nem:%"); Serial.print(Nem);
  Serial.print(" -- Fan Motoru:%"); Serial.println(MotPwmYuzde);
}
```

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Programda gerekli değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

6. UYGULAMA

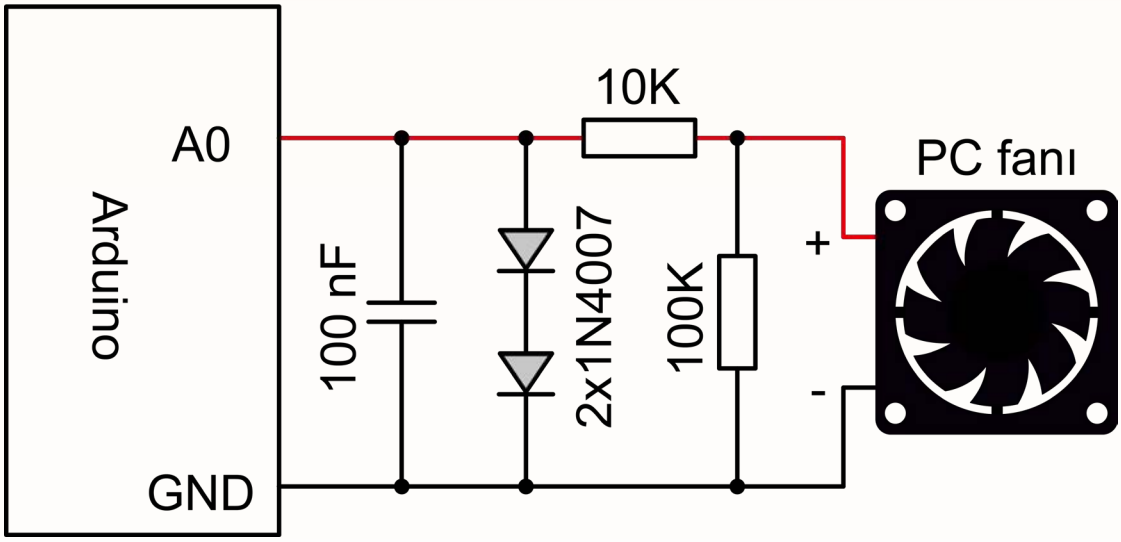
BİLGİSAYAR FANI İLE RÜZGÂRÖLÇER (ANEMOMETRE) YAPILMASI

AMAÇ: **Rüzgârölçer yapmak.**



Görsel 4.28: Uygulamaya ait devre bağlantı şeması

BİLGİ



Görsel 4.29: Rüzgârölçer devre şeması

Rüzgârölçer (anemometre): Rüzgâr / hava akış hızını ölçen cihazlardır. Birimleri m/s, km/s, knots, mph şeklinde olabilir. Genellikle ölçüm için üzerinde küçük bir pervane ve gösterge paneli bulunur. Pervane hızı rüzgâr hızıyla orantılı değiştiği için rüzgâr hızıyla ilişkilendirilerek ölçüm yapılır.

Bilgisayar fanı; üzerinde pervanesi bulunan, sürücü devresi, mıknatıs ve bobinlerden oluşan fırçasız DC motordur. Fan beslemesi verildiğinde pervaneyi çevirerek hava dolaşımını gerçekleştirir. Besleme verilmeden fan pervanesi çevrildiğinde, çevirme hızına göre fan besleme uçlarında dalgalı DC bir gerilim oluşur. Bu özelliğinden faydalanılarak hız ölçümü gerçekleştirilir.

Görsel 4.29'daki şemada kullanılan malzemeler mikrodenetleyici analog girişini korumaya yöneliktir. Direnç, fanın oluşturduğu akımı sınırlar. İki adet diyot, seri bağlanarak fanın oluşturabileceği yüksek gerilimi diyotların iletim gerilimi ile sınırlar (Her bir diyodun iletime geçme gerilimi 0,6-0,7 V civarı olduğu için yüksek gerilim yaklaşık 1,2-1,4 V arası sınırlanır.). Kondansatör, filtre amacıyla temiz bir sinyal almak için kullanılmıştır.

Mikrodenetleyici Analog Referans Değerini Ayarlamak

Referans gerilim değeri, analog kanaldan ölçülecek sinyalin karşılaştırılacağı değerdir. Standart değeri +5 V'tur. Örneğin bir sıcaklık sensörünün 0-100 ^{0}C ölçüme karşılık çıkış gerilim aralığı 0-2,5 V olsun. Eğer standart 5 V referans ile ölçüm alınırsa (0-5 V gerilime karşılık 0-1023 ADC değeri), sensörün son çıkış gerilimi olan 2,5 V değere karşılık ölçülecek ADC değeri 512'de kalacaktır ve 512-1023 arası hiç kullanılamayacaktır. Ayrıca 4,88 mV'luk bir hassasiyete sahip olacaktır. Oysaki referans gerilimi, sensörün son çıkış gerilimi olan 2,5 V'a ayarlandığında 10 bit ADC kanalı tam skala olarak kullanılacak ve hassasiyet artarak 2,44 mV olacaktır.

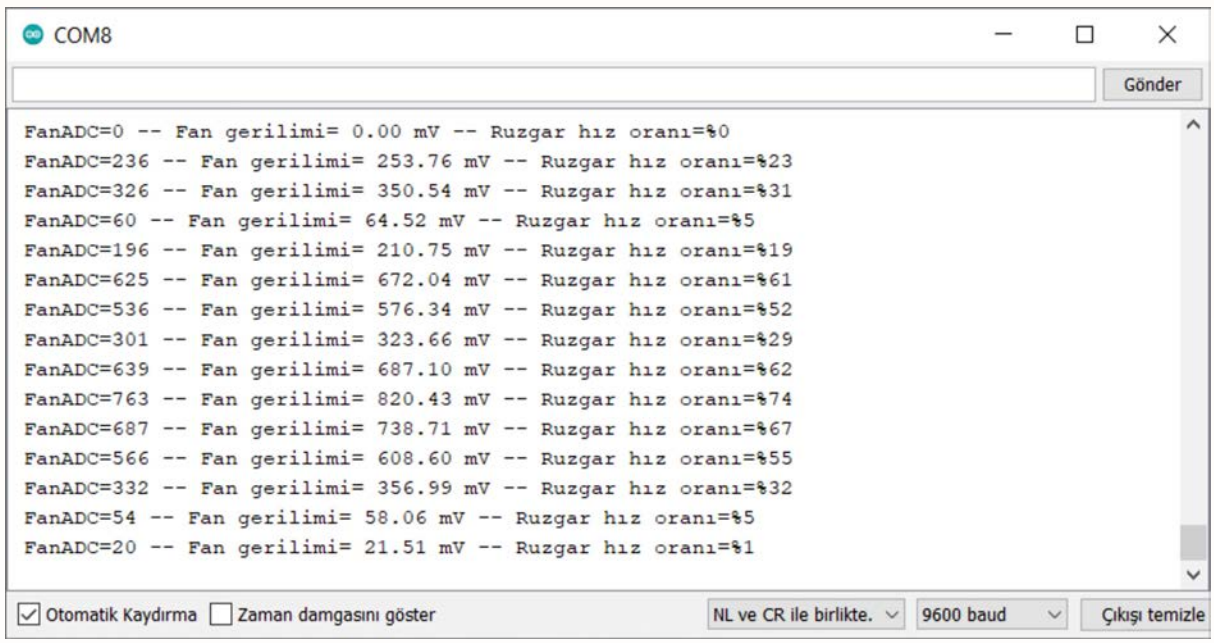
Fanın çıkış gerilimi 1,4 V ile sınırlandırıldığı için analog girişten daha hassas ölçüm almak amacıyla analog ölçüm referans değeri düşürülerek +1,1 V olarak ayarlanmıştır. 0-1,1 V değeri 0-1023 olacak şekilde ADC değerine çevrilir. Arduino Uno kartta dâhilî ve haricî referans gerilimi kullanılabilir. Haricî referans için AREF girişine belirlenen referans gerilimi uygulanıp programda gerekli ayarlamalar yapılır. Dâhilî referans içinse sadece program içinde gerekli ayarlamaların yapılması yeterlidir. Bu ayarlamalar aşağıda belirtildiği gibi yapılır.

analogReference(DEFAULT): Arduino Uno kart için dâhilî +5 V olarak ayarlanır.

analogReference(INTERNAL): Arduino Uno kart için dâhilî AREF 1,1 V olarak ayarlanır.

analogReference(EXTERNAL): Arduino Uno kart için haricî olarak ayarlanır. Mikrodenetleyici AREF girişine referans gerilimi uygulamak gerekir.

Uygulamada rüzgâr hız değeri yerine yüzdesel oran yapılmıştır. Herhangi bir rüzgârölçerle değerler karşılaştırılıp kalibre işlemi gerçekleştirilebilir.



Görsel 4.30: Rüzgâr ölçüm bilgilerine ait seri port ekranı

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 4.28'de verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. Görsel 4.30'daki gibi seri port ekranını açınız.
5. Fanı rüzgâra doğru çeviriniz veya farklı hızlarda dönmesini sağlayacak şekilde saç kurutma makinesi kullanınız.
6. Seri port ekranından, oluşan rüzgâr hızıyla orantılı değerleri gözleyiniz.

7. Analog ölçüm referans değeri standart +5 V olacak şekilde gerekli program değişikliğini yaparak programı yeniden yükleyiniz.
8. Analog referans değeri değişiminin ölçüm sonuçlarını nasıl etkilediğini gözleyiniz.
9. Arduino dijital çıkışlarından birini kullanarak buzzer ekleyiniz.
10. Rüzgâr hız oranı %50'nin üzerine çıktığında alarm verecek şekilde gerekli program değişikliğini yaparak programı yeniden yükleyiniz.
11. Devreye LCD ve batarya bağlayarak taşınabilir bir rüzgârölçer için gerekli program ve donanım değişikliğini yapıp programı yeniden yükleyiniz.
12. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz

UYGULAMAYA AİT PROGRAM

Dosya Düzenle Taslak Araçlar Yardım

```
                                        // Bilgisayar fanı ile rüzgârölçer (anemometre) uygulaması.
int FanPin= A0;                         // A0 analog girişe bilgisayar fanı bağlandı.
int FanADC,RuzgarOrani=0;               // Değişken tanımlamaları.
float FanV;
void setup(){
 analogReference(INTERNAL);             // Analog referans gerilimini 1,1 V ayarla.
 Serial.begin(9600);                    // Seri portu 9600 baud hız için ayarla.
}

void loop() {
 FanADC=analogRead(FanPin);                          // 0 - 1,1 V gerilimi 0-1023 ADC'ye çevir.
 FanV=(1100.0/1023.0)*FanADC;                        // (1,1V=1100 mV) 1100mV/1023 x ADC değeri.
 RuzgarOrani=map(FanADC,0,1023,0,100);               //0-1023 değerini 0-100'e çevir.
 PCgonder();                                          // Rüzgâr ölçüm değerlerini PC'ye gönder.
}

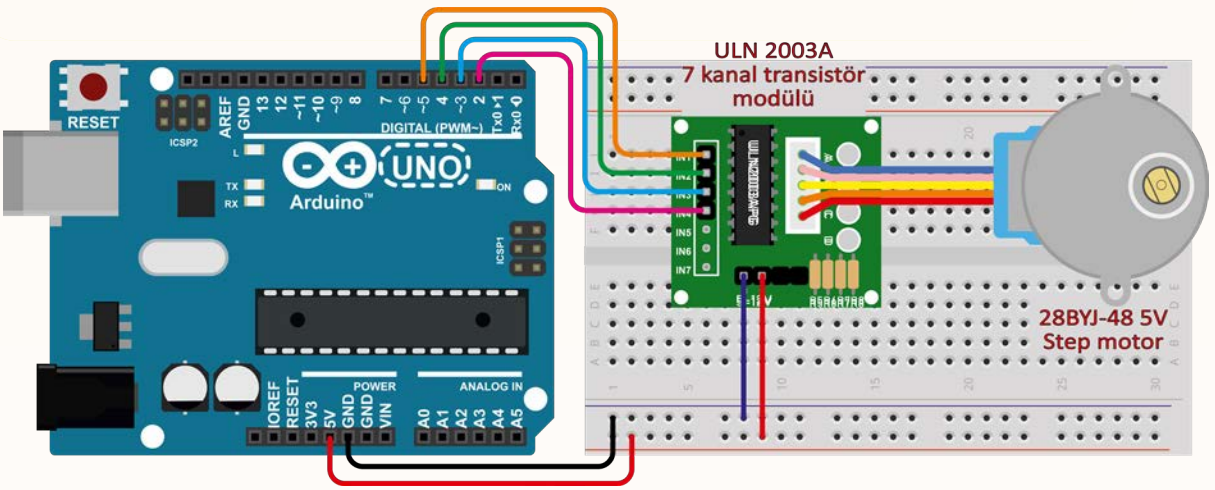
void PCgonder () {
 Serial.print(" FanADC="); Serial.print(FanADC);
 Serial.print(" -- Fan gerilimi= ");
 Serial.print(FanV); Serial.print(" mV");
 Serial.print(" -- Ruzgar hız oranı=%");
 Serial.println(RuzgarOrani);
 delay(500);
}
```

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Donanım ve programda gerekli değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

7. UYGULAMA

360⁰ İLERİ-GERİ HAREKET EDEN STEP MOTOR UYGULAMASI

AMAÇ: **Step motoru çalıştırmak.**



Görsel 4.31: Uygulamaya ait devre bağlantı şeması

BİLGİ

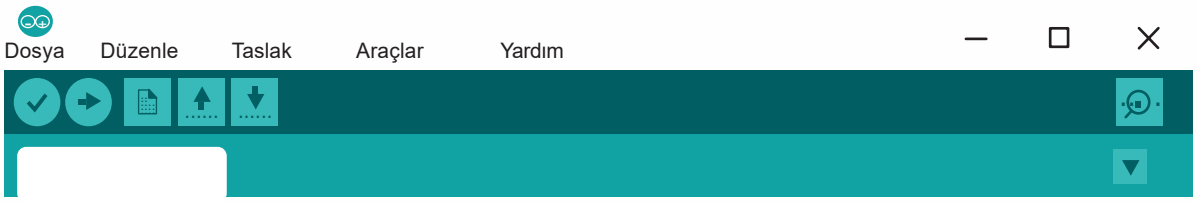
Uygulamada 5 uçlu 28BYJ-48 5 V step motoru kullanılmıştır. Motor, tam bir turunu 32 adımda gerçekleştirir. Ayrıca motor miline bağlı 1/64 oranında redüktör bulunur. Çıkış milinin tam bir tur dönmesi için 32 x 64 = 2048 pals verilmesi gerekir.

Sürücü için ULN2003 entegresi ile yapılmış modül kullanılmıştır.

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 4.31'de verilen devre kurulumunu gerçekleştiriniz.
3. Step motorun durduğu yeri anlayabilmek için step motor mili ve gövde arasına işaret koyunuz.
4. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
5. Step motorun 360^0 tam tur dönüp bekledikten sonra geri hareket ettiğini gözleyiniz.
6. Step motorun işaret koyduğunuz noktalar arasında ileri-geri çalışarak tam noktasında durduğunu gözleyiniz.
7. Yarım tur (180^0) ileri-geri hareket için gerekli program değişikliğini yaparak programı yeniden yükleyiniz ve gözlemleyiniz.
8. Dönüş hızını artırmak için gerekli program değişikliğini yaparak programı yeniden yükleyiniz ve gözlemleyiniz.
9. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM



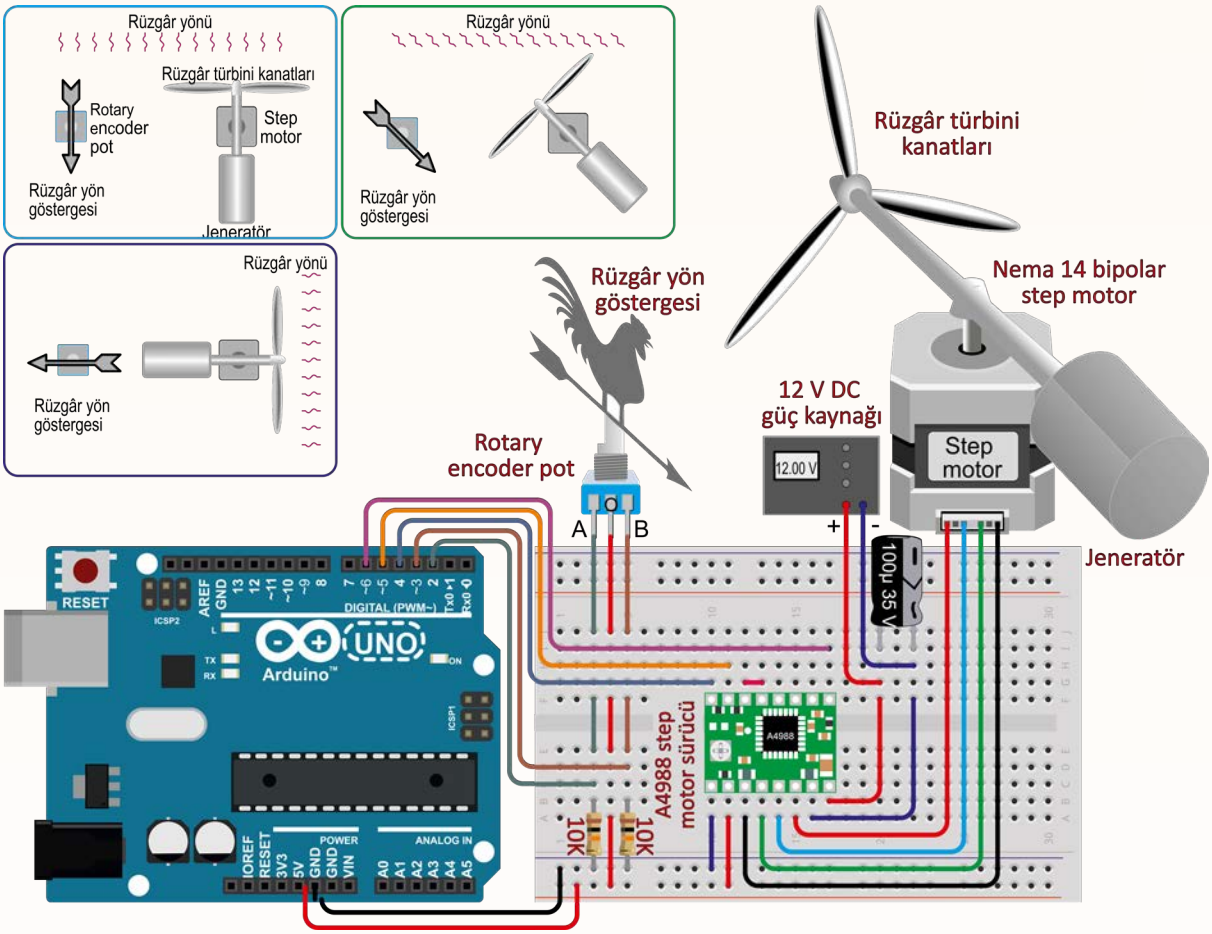

```
                                        // 28BYJ-48 5 V step motorunun tek faz uyartım ile
                                        // 360 derece ileri-geri çalıştırılması.
int Hiz=5;                              // Hız değişkeni adımlar arası 5 ms ayarla.
int Say=0;                              // Değişken tanımlamaları.
void setup() {
  DDRD = B00111100;                     // 5,4,3,2 no.lu pinleri step motor için çıkış ayarla.
}
void loop()
{
                          // 28BYJ-48 step motorunda 1/64 redüktör oranı ve 32 adım/tur olduğu için
                         // 2048 pals verilmesi gerekir. For döngüsünde 512 defa 4 adım gittiğinde
                         // 2048 pals verilir ve 360 derece tamamlanır sonra geri harekete geçer.
  for(Say=0;Say<=512;Say++){            // İleri 360 derece hareket et.
    PORTD = B00000100; delay(Hiz);      // Step motor 1. adım
    PORTD = B00001000; delay(Hiz);      // Step motor 2. adım
    PORTD = B00010000; delay(Hiz);      // Step motor 3. adım
    PORTD = B00100000; delay(Hiz);      // Step motor 4. adım
    PORTD = B00000000;                  // Step motor boşta.
  }
  delay(2000);                          // Dönüşler arasında 2 sn. bekle.
  for(Say=0;Say<=512;Say++){            // Geri 360 derece hareket et.
    PORTD = B00010000; delay(Hiz);      // Step motor 3. adım
    PORTD = B00001000; delay(Hiz);      // Step motor 2. adım
    PORTD = B00000100; delay(Hiz);      // Step motor 1. adım
    PORTD = B00100000; delay(Hiz);      // Step motor 4. adım
    PORTD = B00000000;                  // Step motor boşta.
  }
  delay(2000);                           // Dönüşler arasında 2 sn. bekle.
}
```

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Programda gerekli değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

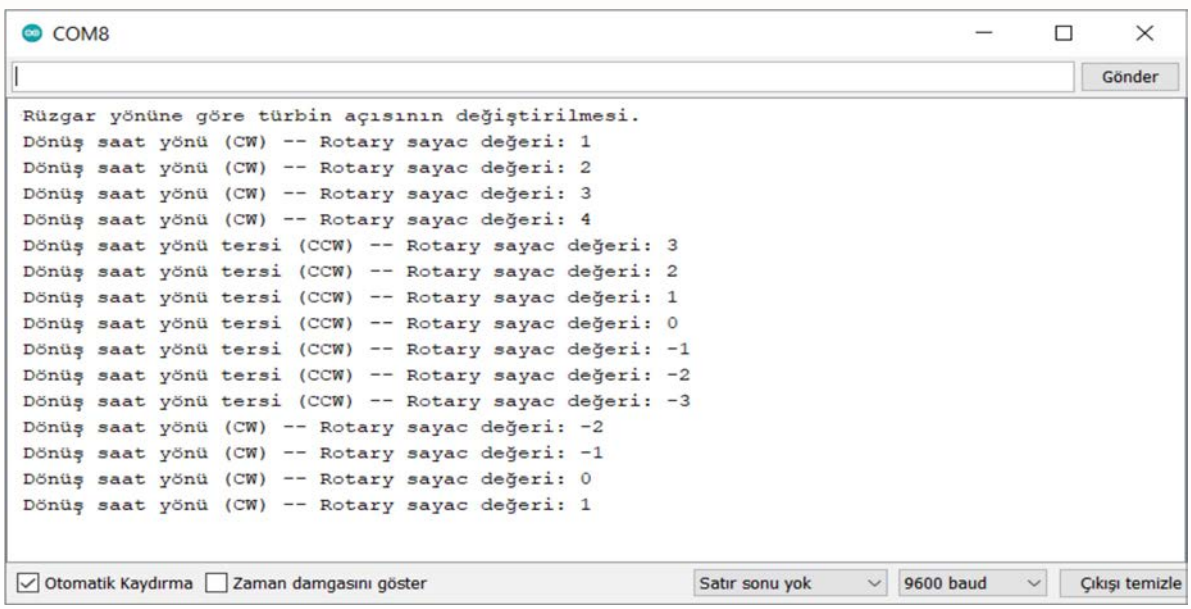
8. UYGULAMA

ROTARY ENCODER VE BİPOLAR STEP MOTORLA RÜZGÂR YÖNÜNE GÖRE TÜRBİN AÇISININ DEĞİŞTİRİLMESİ

AMAÇ: Gelişmiş step motor sürücüsüyle rotary encoder kullanmak.



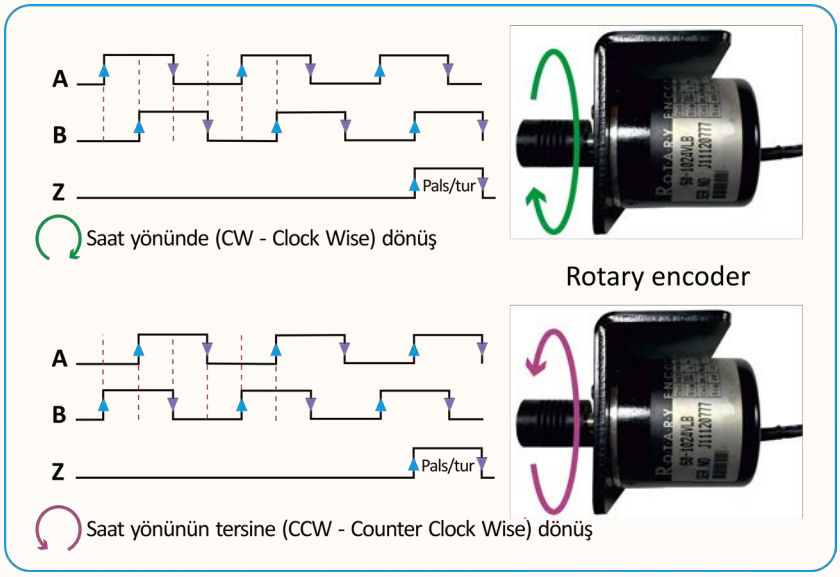
Görsel 4.32: Uygulamaya ait devre bağlantı şeması



Görsel 4.33: Rüzgâr yönüne göre türbin açısının değiştirilmesine ait seri port ekranı

BİLGİ

Bu uygulamada türbin kanatlarını rüzgâr yönüne doğru hareket ettirmek için rotary encoder pot, bipolar step motor ve gelişmiş step motor sürücüsü kullanılmıştır.



Görsel 4.34: Rotary encoder ve çıkış dalga şekilleri

Rotary Encoder (Döner Kodlayıcı)

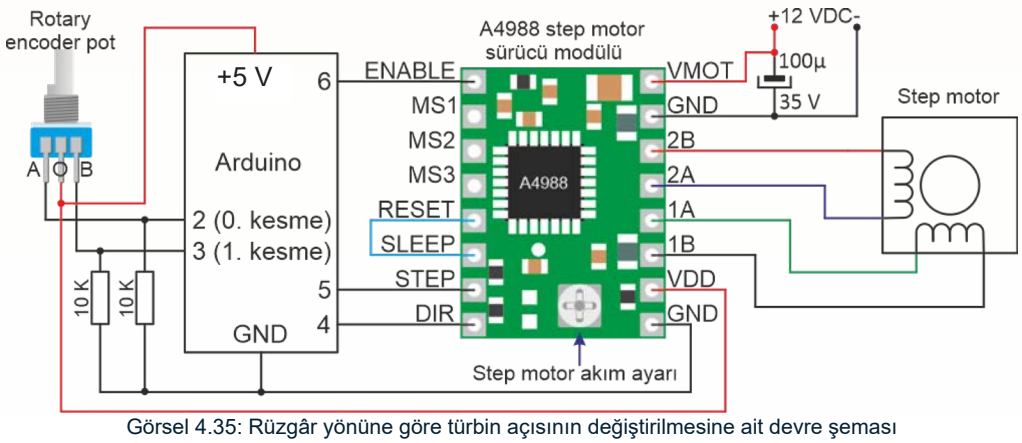
Döner bir cismin (motor vs.) miline bağlanarak bu cismin hareketine karşılık sayısal sinyal üreten elektromekanik bir cihazdır. Döner cismin konum, hız, yön ve hareketlerini belirlemek için kullanılır.

Görsel 4.34'te rotary encoder ve çıkış dalga şekilleri verilmiştir. A, B ve Z faz çıkışları mevcuttur. A ve B faz çıkışları hareket yönü ve döner cismin hareket açısına karşılık pals çıkışı üretir. Z fazı tur başına bir pals üretir. Örneğin saat yönüne dönüşte önce A sonra B fazı, saat yönünün tersindeyse önce B sonra A fazı çıkış verir. Mikrodenetleyiciye hangi fazın önce geldiği algılatılırsa dönüş yönü belirlenmiş olur. A veya B fazlarından birinin sayılmasıyla dönüş açı bilgisi hesaplanır.

Bir tur başına A veya B fazının ürettiği pals sayısı rotary encoderin çözünürlüğünü belirtir. Örneğin bir motora bağlı rotary encoder bir turda 1440 pals üretirse 0,25°lik adımlarla motor hareket bilgisi elde edilir.

Uygulamada kullanılan rotary encoder pot aynı prensibe göre çalışır. Sınırsız sayıda saat yönü veya tersine dönüş yapabildiği için tercih edilmiştir. Çözünürlüğü 20 pals/tur olup Z fazı yoktur. Genellikle elektronik cihazlarda bir fonksiyonu artırma veya azaltma amaçlı kullanılmaktadır (ses+, ses- gibi). A ve B fazları bir direnç üzerinden GND'ye bağlanıp, rotary encoder'ın O ucundan lojik 1 verildiğinde A ve B faz çıkışlarından dönüş yönü ve açısına göre Görsel 4.34'teki gibi çıkış gözlenir.

Rotary encoder pot, A ve B faz çıkışlarında oluşan pals durumlarının kaçırılmadan algılanması için mikrodenetleyici kesme girişlerine bağlanmıştır (Görsel 4.35).



Görsel 4.35: Rüzgâr yönüne göre türbin açısının değiştirilmesine ait devre şeması

Bipolar Step Motor Sürücüleri

Bipolar step motorları sürebilmek için gelişmiş step motor sürücülerine ihtiyaç duyulur. A4988 sürücü entegresi ile yapılmış bu modül, soğutucu olmadan 1 A akım ve 8-35 V gerilime kadar çalışabilmektedir. Bu değerlere uygun küçük tip step motor (nema14 vs.) seçimi yapılmalıdır. Sürücü üzerinde step motor akımını sınırlayan küçük bir trimpot vardır. Bu tip step motor sürücülerinin kullanımı oldukça kolaydır. Kontrol ve çıkış uçları şunlardır:

DIR (direction-yön): Step motorun yön değiştirmesi için lojik 1 veya lojik 0 verilir.

STEP (adım): Verilen her pals, çözünürlük oranında (MS1, MS2, MS3) step motoru bir adım hareket ettirir. Lojik 1, ardından lojik 0 şeklinde bir pals verilir ve step motorun adım atması sağlanır.

SLEEP (uyku): Sürücü kullanılmadığında enerji harcamasını en alt düzeye indirir. Aktifleştirmek için lojik 0 verilmelidir.

RESET: Yapılan ayarların sıfırlanması için kullanılır. Aktifleştirmek için lojik 0 verilmelidir. Uygulamada sleep ve reset kullanılmadığı için birbirine bağlanarak pasif yapılmıştır.

ENABLE: Sürücüyü aktif veya pasif yapmak için kullanılır. Aktifleştirmek için lojik 0 verilmelidir. Bu giriş aktif yapıldığında step motor bobinleri enerji altında bırakılarak konum kilitlemesi yapılabilir ancak sürekli akım çekileceği unutulmamalıdır.

Çözünürlük ayarları (MS1, MS2, MS3): Mikro step olarak tanımlanır. Bu ayarlar step motorun adım açısını belirler. Uygulamada bu uçlar kullanılmadığı için otomatik olarak 1/1 adım seçilmiş olur. Örneğin 1,80 / adım step motorda çözünürlük ayarları aşağıdaki tabloda gösterildiği gibidir.

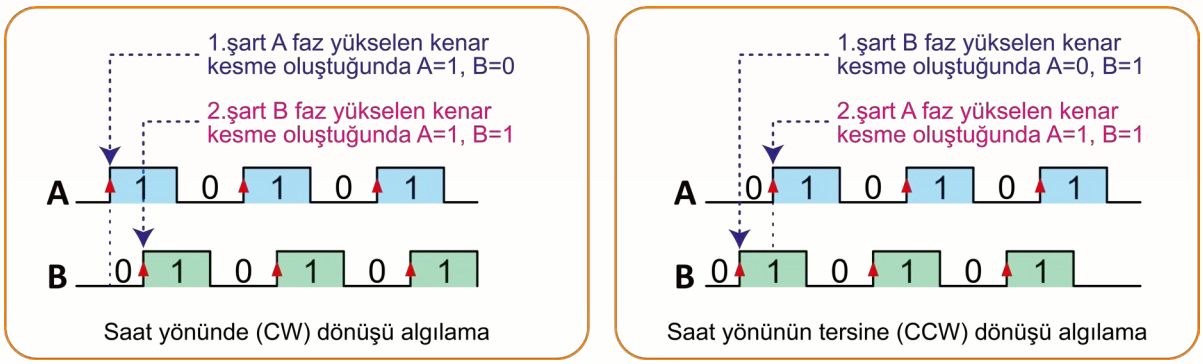
MS1	MS2	MS3	Mikro step çözünürlüğü
Lojik 0	Lojik 0	Lojik 0	1/1 (Tam adım): Step motor her palste 1,80 adım döner.
Lojik 1	Lojik 0	Lojik 0	1/2 (Yarım adım): Step motor her palste 0,90 adım döner.
Lojik 0	Lojik 1	Lojik 0	1/4 (Çeyrek adım): Step motor her palste 0,450 adım döner.
Lojik 1	Lojik 1	Lojik 0	1/8: Step motor her palste 0,2250 adım döner.
Lojik 1	Lojik 1	Lojik 1	1/16: Step motor her palste 0,11250 adım döner.

VMOT-GND: Step motor besleme girişleri, 8-35 V.

VDD-GND: Sürücü modül besleme girişleri, +5 V.

1A-1B: Step motor 1. faz bobin bağlantı çıkışları.

2A-2B: Step motor 2. faz bobin bağlantı çıkışları.



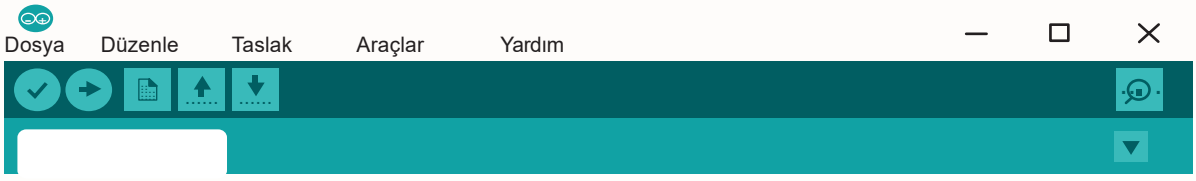
Görsel 4.36: Rotary encoder pot dönüş yönünün algılanmasına ait sinyal grafiği

Uygulama programında 0. ve 1. kesme girişleri haricî kesme ve yükselen kenar tetikleme olarak aktif yapılmıştır. A veya B fazının her yükselen (lojik 0'dan lojik 1'e) kenarında "**EncoderKesme**" ile belirtilen fonksiyona gitmesi sağlanmıştır. Kesme fonksiyonunda Görsel 4.36'daki gibi dönüş yönünün algılanması için şartlar değerlendirilir. Algılanan dönüş yönüne göre sayaç değeri değiştirilir ve step motorun hangi yönde dönmesi gerektiği yön girişine (DIR) verilir. Her bir değişimde "**StepMotorCalistir**" fonksiyonu çalıştırılır ve step motorun dönmesi sağlanır. Rotary encoder pot, bir tur döndüğünde step motorun da bir tur dönmesi için rotary encoder potun her bir pals değerinde step motor 10 adım atmalıdır [Step motor 200 adım/tur (1,8^0/adım) ve rotary encoder 20 pals/tur].

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 4.32'de verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. Görsel 4.33'teki gibi seri port ekranını açınız.
5. Rotary encoder potu önce saat yönünde, sonra ters yönde yavaşça çeviriniz. Step motorun aynı oranda ve yönde döndüğünü seri port ekranından da gözleyiniz.
6. Görsel 4.32'deki gibi step motor miline rüzgâr türbini ve rotary encoder pot miline rüzgâr yön göstergesini yerleştiriniz.
7. Türbin açısının rüzgâr yönüne göre değiştiğini gözleyiniz.
8. Rüzgâr yön göstergesi ve türbini sökünüz.
9. Step motor sürücüsünün çözünürlük ayar girişlerini (MS1, MS2, MS3) mikrodenetleyicide uygun dijital çıkışlara bağlayınız.
10. Farklı çözünürlük ayarları için gerekli program değişikliğini yaparak programı yeniden yükleyiniz.
11. Rotary encoder potu çeviriniz. Çözünürlük ayarına göre step motorun dönme açısını gözleyiniz.
12. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM




```
                    //Rotary enkoder pot ve step motorla rüzgâr yönüne göre türbin açısını değiş.
int FazA=2;         // Rotary encoder pot A faz ucu 2 no.lu (0.kesme) uca bağlandı.
int FazB=3;         // Rotary encoder pot B faz ucu 3 no.lu (1.kesme) uca bağlandı.
int DIR = 4;        // A4988 DIR ucu 4 no.lu uca bağlandı.
int STEP = 5;       // A4988 STEP ucu 5 no.lu uca bağlandı.
int ENABLE = 6;     // A4988 ENABLE ucu 6 no.lu uca bağlandı.
int EnkoderHareket,RotarySayac,Adim = 0;                    // Değişken tanımlamaları.
int SartCW,SartCCW,DurumFazA,DurumFazB =0;
String MevcutYon ="";                                       // MevcutYon değişkenini string olarak ayarla.

void setup() {
 pinMode(DIR, OUTPUT);  pinMode(STEP, OUTPUT);
 pinMode(ENABLE, OUTPUT); digitalWrite(ENABLE,1);
 pinMode(FazA,INPUT); pinMode(FazB,INPUT);
 Serial.begin(9600);
 Serial.println("Rüzgâr yönüne göre türbin açısının değiştirilmesi.");
                    // Rotary encoder pot, A ve B fazlarının her yükselen kenarında kesmeye
                    // gitmesi için haricî kesmeleri RISING olarak ayarla.
 attachInterrupt(0, EncoderKesme, RISING);                  // A fazı kesme için ayarla.
 attachInterrupt(1, EncoderKesme, RISING);                  // B fazı kesme için ayarla.
}
```

```
void loop() {
  if (EnkoderHareket==1){                                  // Rotary encoder hareket algılandıysa
    StepMotorCalistir();                                   // Step motoru çalıştır.
    Serial.print(MevcutYon);
    Serial.print(" -- Rotary sayac değeri: ");
    Serial.println(RotarySayac);
    EnkoderHareket=0;                                      // hareket algılama değişkenini 0 yap.
  }
}
void StepMotorCalistir() {
  digitalWrite(ENABLE,0);                                  //Sürücü ENABLE aktif yap.
                                  // Step motor 200 adım/tur ve rotary encoder pot 20 pals/tur olduğu için
                                  // Rotary encoder potun her bir adımı step motorda 200/20=10 adımda olacaktır.
  for(Adim=1;Adim<=10;Adim++){                             // Step motorun 10 adım atmasını sağla.
    digitalWrite(STEP,1);                                  // Sürücü STEP girişine adım için pals ver.
    digitalWrite(STEP,0);
    delayMicroseconds(3000);                               // Adımlar arası 3 ms bekle.
  }
  digitalWrite(ENABLE,1);                                  //Sürücü ENABLE pasif yap.
}
void EncoderKesme(){                                       // Faz A ve B yükselen kenar kesme fonksiyonu.
  DurumFazA = digitalRead(FazA);                           // Faz A durumunu oku ve değişkene aktar.
  DurumFazB = digitalRead(FazB);                           // Faz B durumunu oku ve değişkene aktar.

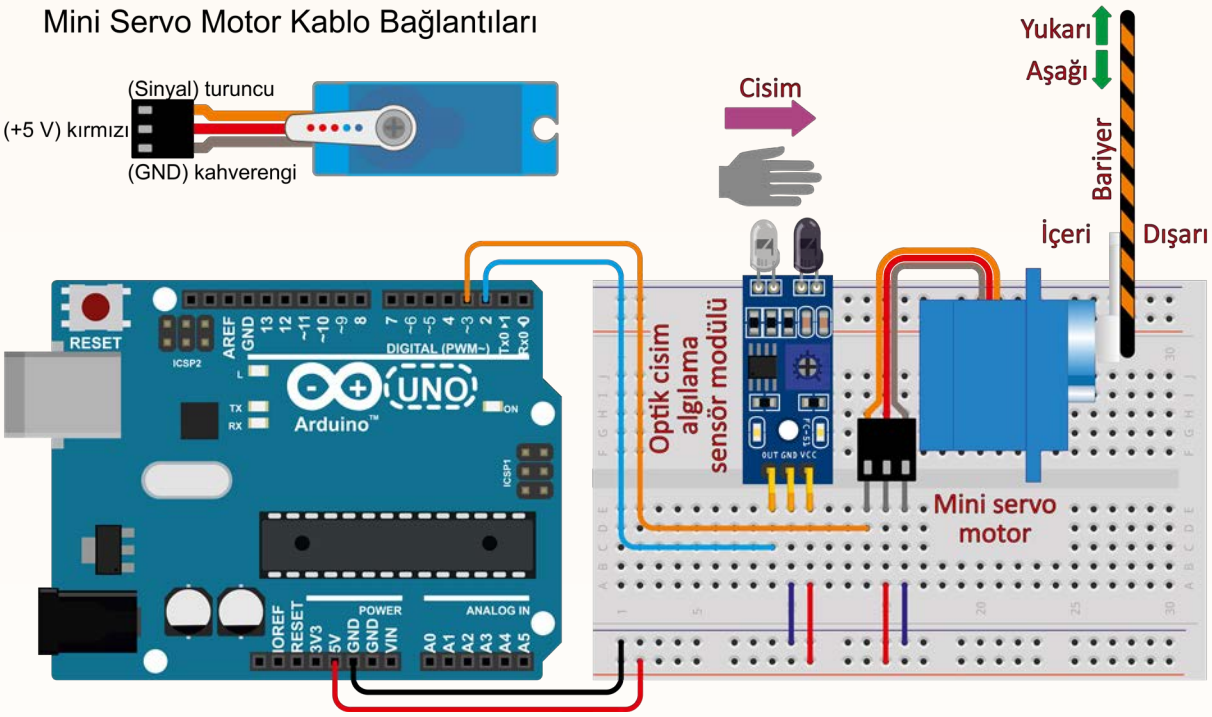
 if(DurumFazA==1 && DurumFazB==1){                         // Her iki durum için 2. şart olduysa
    if(SartCW==1 ){                  // Saat yönü için 1.şart gerçekleştiyse
                                     // Dönüş algılandı. Rotary sayacı artır ve step motoru CW yönüne ayarla.
      EnkoderHareket=1; RotarySayac ++; SartCW=0;
      digitalWrite(DIR,1); MevcutYon =" Dönüş saat yönü (CW)";
    }
    if(SartCCW==1 ){                 // Saat yönü tersi için 1.şart gerçekleştiyse
                                      // Dönüş algılandı. Rotary sayacı azalt ve step motoru CCW yönüne ayarla.
      EnkoderHareket=1; RotarySayac --; SartCCW=0;
      digitalWrite(DIR,0); MevcutYon =" Dönüş saat yönünün tersi (CCW)";
    }
 }
  if(DurumFazA==1 && DurumFazB==0){SartCW=1;SartCCW=0;}          // CW  1.şart.
  if(DurumFazA==0 && DurumFazB==1){SartCW=0;SartCCW=1;}          // CCW 1.şart.
}
```

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Donanım ve programda gerekli değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

9. UYGULAMA

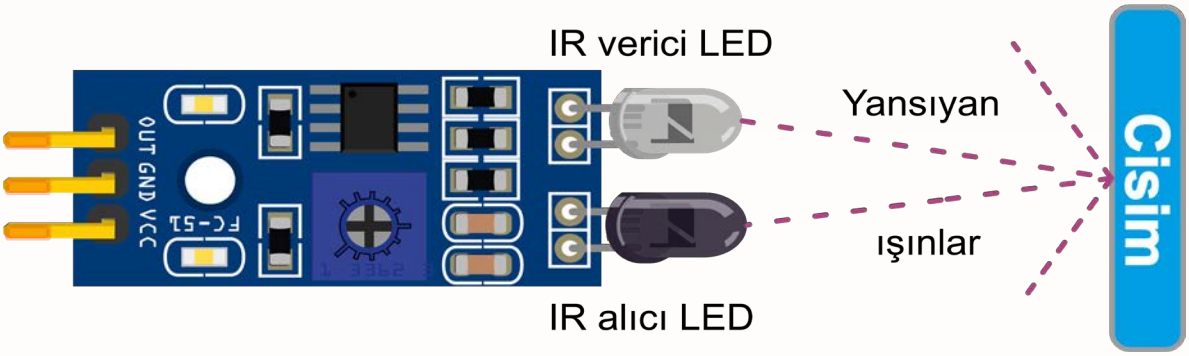
OPTİK CİSİM ALGILAMA SENSÖRÜ VE SERVO MOTORLA BARİYER KAPI UYGULAMASI

AMAÇ: Sensörlü otomatik açılıp kapanan bariyer kapı yapmak.



Görsel 4.37: Uygulamaya ait devre bağlantı şeması

BİLGİ



Görsel 4.38: Optik cisim algılama sensör modülü

Optik Cisim Algılama Sensör Modülü

Üzerinde IR (Infrared / kızılötesi) verici ve alıcı LED bulunan modüldür (Görsel 4.38). Verici LED tarafından gönderilen ışık bir cisme çarptığında, o cisimden etrafa dağılarak yansır. Yansıyan bazı ışınlar alıcı LED'e ulaşır. Alıcı LED çıkışında, cismin uzaklığına göre düşük veya yüksek seviyeli bir sinyal oluşur. Modül üzerinde bulunan yükselteç devresiyle bu sinyal güçlendirilerek uygun lojik seviyeye getirilir. Ayarlı direnç ile algılama mesafesi ayarlanabilir. Uygulamada kullanılan modülün çıkışı, bir cisim yaklaştığında lojik 0, cisim olmadığında ise lojik 1'dir.

Kızılötesi ışık, insan gözünün algılama sınırı dışında olduğu için çıplak gözle görülemez ancak kızılötesi sensör veya standart bir kamerayla çalışması gözlenebilir.

Uygulamada servo motor kontrolünün daha iyi anlaşılması için 1. program kütüphane kullanılmadan verilmiştir. 2. programda servo motor kütüphanesi eklenmiştir. Bu kütüphanede kullanılan tanımlar aşağıdaki gibidir.

Servo **ServoKapi**: Servo adı, ServoKapi olarak kullanılmıştır.

ServoKapi.attach(**ServoMotor**): Servo motor bağlantı ve kontrol pini tanımlanır.

ServoKapi.attach(**ServoMotor**): Servo motor bağlantı ve kontrol pini tanımlanır.

ServoKapi.write(**pos**): Pos değişkeni ile belirtilen açı kadar servo motoru döndürür.

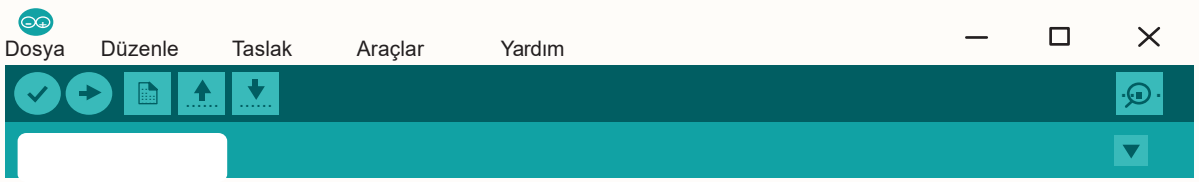
Örneğin **ServoKapi**.write(**149**) yazıldığında servo motor 149^0 döner.

Uygulamada, optik cisim algılama sensör modülü çıkışı, cisim olmadığında lojik 1 seviyesindedir. Modülün önüne bir cisim geldiğinde (araç, insan vs.) modül çıkışı lojik 0'a düşer. Servo motora bağlı bariyer 90^0 açılır (90^0den 180^0ye). Cisim, sensör önünde durmaya devam ettiği sürece bariyer açık kalacaktır. Cisim geçtikten 3 sn. sonra kapı 90^0 kapanır (180^0den 90^0ye).

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 4.37'de verilen devre kurulumunu gerçekleştiriniz.
3. Servo motor miline bariyer görevi görecek bir çubuk bağlayınız ve servo motoru yatay olarak monte ediniz.
4. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
5. Optik cisim algılama sensörü önüne elinizi yaklaştırınız ve bariyerin açıldığını gözleyiniz.
6. Sensör modülü önünden elinizi çekmediğiniz sürece bariyerin açık kaldığını gözleyiniz.
7. Sensör modülü önünden elinizi çektikten 3 sn. sonra bariyerin kapandığını gözleyiniz.
8. Bariyer kapı, sensörle tek taraflı olarak içeriden açılmaktadır. Dışarıdan da bir butonla açmak için gerekli program ve donanım değişikliğini yaparak program ve donanımı yeniden yükleyiniz.
9. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

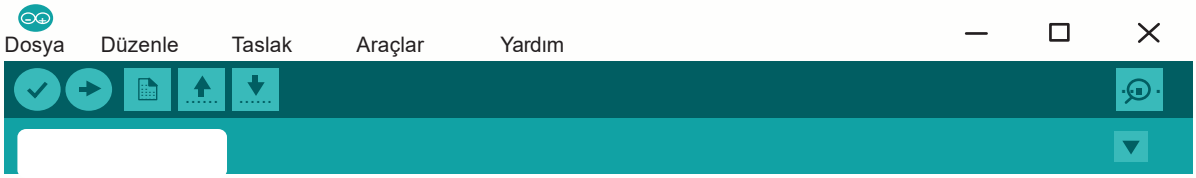
UYGULAMAYA AİT PROGRAM (KÜTÜPHANE KULLANILMADAN)




```
                                    // Servo motorlu optik cisim algılama sensörlü otomatik bariyer kapı uygulaması
int CisimSns= 2;                    // Optik cisim algılama sensör modülü.
int ServoMotor = 3;                 // Servo motor kontrol pini.
int Cisim,Kapideg,pos,PwmSinyal=0;                  // Değişken tanımlamaları.
void setup() {
 pinMode(CisimSns, INPUT);  pinMode(ServoMotor, OUTPUT);
}
void loop() {
 Cisim=!digitalRead(CisimSns);                      // Cisim gördüğünde 0 olduğu için değilini al.
  if (Cisim==1){  KapiAc(); }
  KapiKapat();
}
void KapiAc() {                                     // Kapı açma fonksiyonu.
 if (Kapideg==1){ return; }
   for (pos = 0; pos <= 1000; pos=pos+5) {          // 90°den 180°ye 5° açılarla dön.
      ServoKontrol();  }                            // ServoKontrol fonksiyonuna git.
   Kapideg = 1;
}
void KapiKapat(){                                   // Kapı kapatma fonksiyonu.
                                    // Kapı açılıyorken veya cisim algıladığında işlem yapmadan geri dön.
 if (Kapideg==0 || Cisim==1){return;}
   delay(3000);                                     // Kapı açıldıktan sonra 3 sn. bekle.
   for (pos = 1000; pos >= 0; pos=pos-5){           // 180°den 90°ye 5° açılarla dön.
      ServoKontrol(); }                             // ServoKontrol fonksiyonuna git.
  Kapideg=0; }
void ServoKontrol(){                                // Servo motor pwm sinyali oluştur.
    PwmSinyal=1500+pos; // 1500 us – 2500 us (90°ile 180°) arası
```

```
    digitalWrite(ServoMotor,1);delayMicroseconds(PwmSinyal);          // PwmSinyal değişkeni kadar lojik 1 yap.
    digitalWrite(ServoMotor,0);delayMicroseconds(20000-PwmSinyal);  // 20 ms (20000 us) periyot oluştur.
}
```

UYGULAMAYA AİT PROGRAM (KÜTÜPHANE KULLANILARAK)



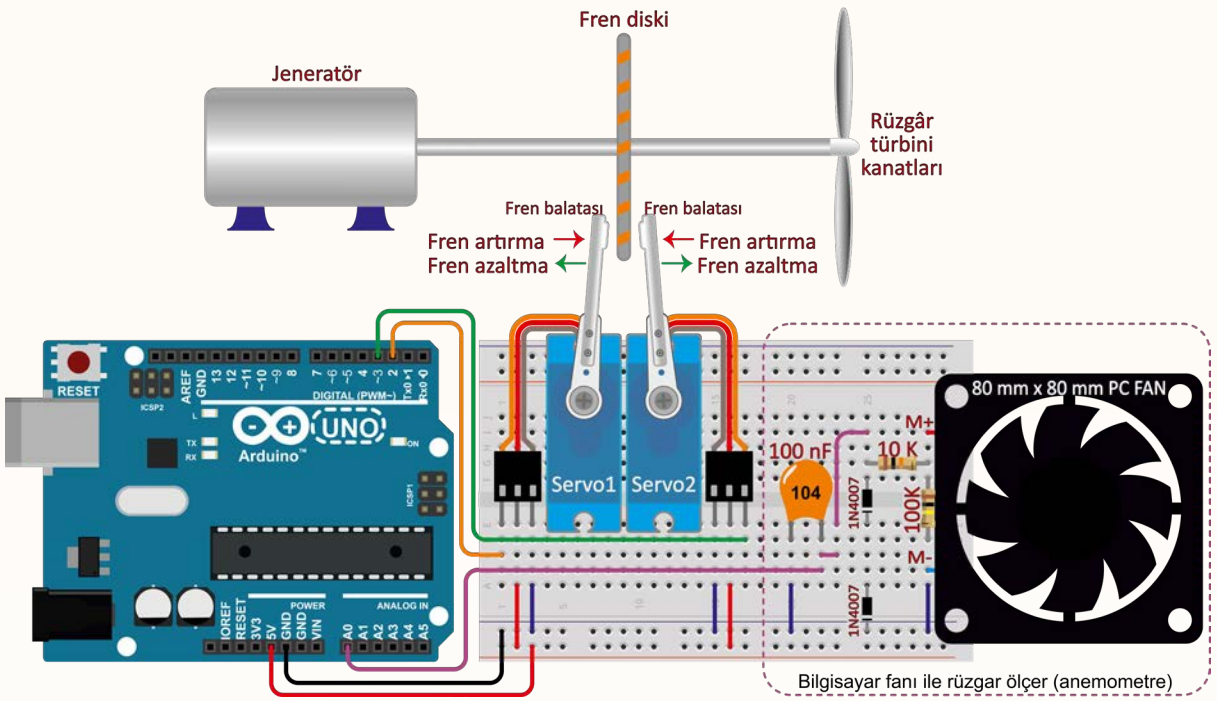

```
                                    // Servo motorlu optik cisim algılama sensörlü otomatik bariyer kapı uygulaması
#include <Servo.h>                  // Servo motor kontrol kütüphanesini ekle.
int CisimSn
s= 2;                                       // Optik cisim algılama sensör modülü.
int ServoMotor = 3;                         // Servo motor kontrol pini.
int Cisim,Kapideg,pos=0;                    // Değişken tanımlamaları.
Servo ServoKapi;                            // Servo adını ServoKapi olarak kullan.
void setup() {
 ServoKapi.attach(ServoMotor);              // 3 no.lu çıkışı servo motor için ayarla.
 pinMode(CisimSns, INPUT);
 pinMode(ServoMotor, OUTPUT);
}
void loop() {
 Cisim=!digitalRead(CisimSns);              // Cisim gördüğünde 0 olduğu için değilini al.
  if (Cisim==1){  KapiAc(); }
  KapiKapat();
}
void KapiAc() {                             // Kapı açma fonksiyonu.
 if (Kapideg==1){ return; }                 // Kapideg değişkeni 1 ise işlem yapma geri dön.
   for (pos = 90; pos <= 180; pos ++) {     // 90°den 180°ye 1°lik açılarla dön.
      ServoKapi.write(pos); delay(10);      // Açı değerini servo motora gönder.
   }
  Kapideg = 1;
}
void KapiKapat(){                           // Kapı kapatma fonksiyonu.
                                    // Kapı açılıyorken veya cisim algıladığında işlem yapmadan geri dön.
 if (Kapideg==0 || Cisim==1){return;}
   delay(3000);                             // Kapı açıldıktan sonra 3 sn. bekle.
   for (pos = 180; pos >= 90; pos--){       // 180°den 90°ye 1°lik açılarla dön.
      ServoKapi.write(pos); delay(10);      // Açı değerini servo motora gönder.
   }
  Kapideg=0;
}
```

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Donanım ve programda gerekli değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

10. UYGULAMA

RÜZGÂR HIZINA GÖRE TÜRBİN FREN KONTROLÜNÜN SERVO MOTORLARLA YAPILMASI

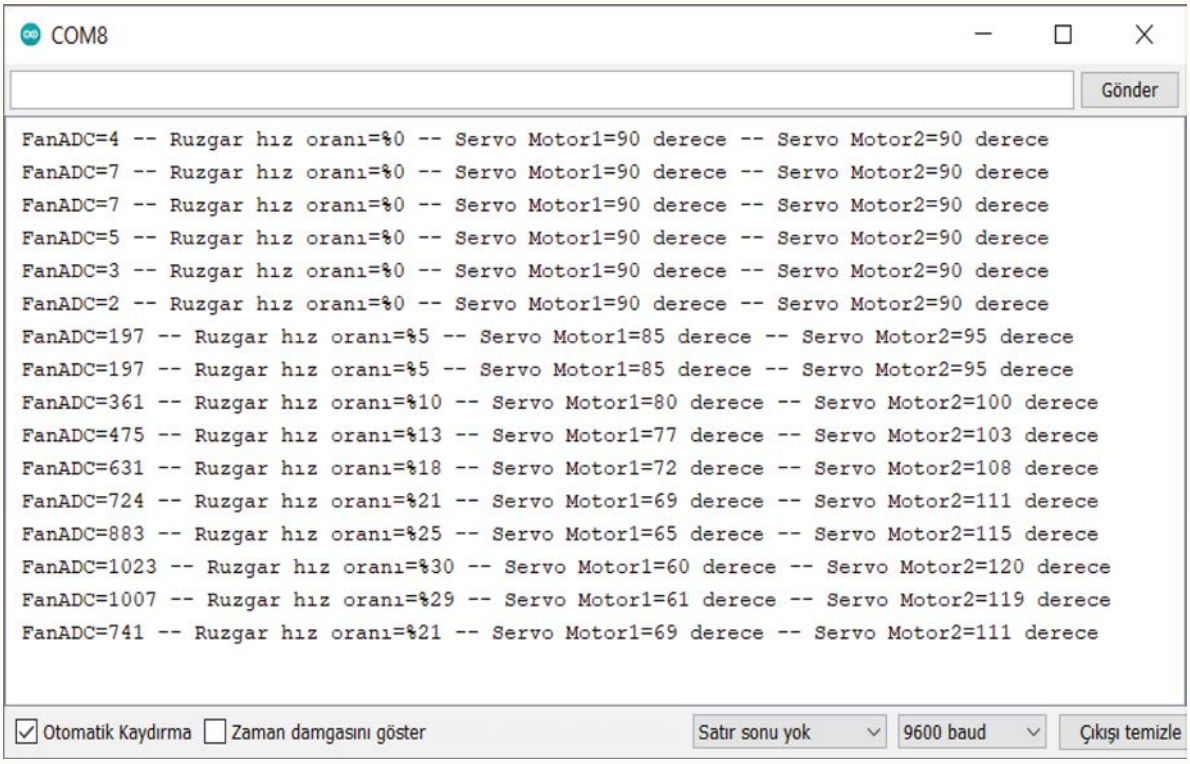
AMAÇ: Servo motorları anemometreyle kontrol etmek.



Görsel 4.39: Uygulamaya ait devre bağlantı şeması

BİLGİ

Bu uygulama, bilgisayar fanı kullanılarak rüzgârölçer ve servo motor uygulamalarının birleştirilmiş hâlidir. Rüzgâr hızı arttıkça türbin kanatlarının aşırı hızla dönmesini engellemek için fren diski ve balata sistemi kullanılmıştır. Servo motorlar rüzgâr hızına göre birbirine ters yönde hareket eder ve fren diskini sıkıştırıp gevşetir. Anemometre çıkışından rüzgâr hızıyla orantılı gerilim alınır. Bu gerilim değerine göre servo motorların dönme açısı servo1 için $90\text{-}60^0$ ve servo 2 için $90\text{-}120^0$ olacak şekilde $0\text{-}30^0$ ayarlanmıştır.

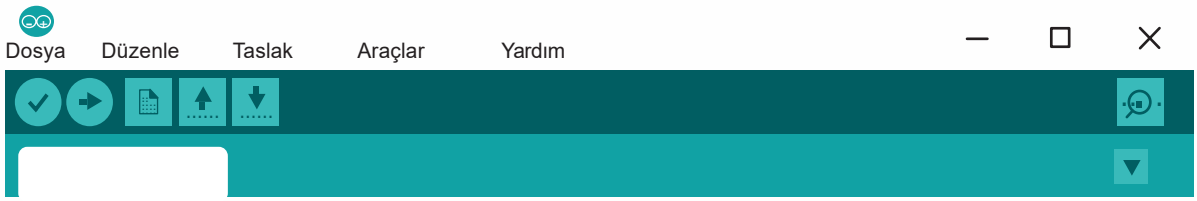


Görsel 4.40: Rüzgâr türbininin fren kontrolüne ait seri port ekranı

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 4.39'da verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. Görsel 4.40'taki gibi seri port ekranını açınız.
5. Fanı rüzgâra doğru çeviriniz veya farklı hızlarda dönmesini sağlayacak şekilde saç kurutma makinesi kullanınız.
6. Seri port ekranından, oluşan rüzgâr hızıyla orantılı değerleri gözleyiniz.
7. Rüzgâr hızıyla orantılı şekilde servo motorların birbirine ters şekilde maksimum 30^0 döndüğünü gözleyiniz.
8. Görsel 4.39'da verilen mekanik kısımların (fren diski, balata, rüzgâr türbini kanatları vs.) kurulumunu yapınız.
9. Rüzgâr türbiniyle fanı rüzgâra doğru çeviriniz veya dönmesini sağlayacak şekilde kuvvetlice üfleyiniz.
10. Rüzgâr hızıyla orantılı şekilde servo motorların fren diskini sıkıştırdığını veya gevşettiğini gözleyiniz.
11. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM




```
                                        // Rüzgâr hızına göre türbin fren kontrolünün yapılması

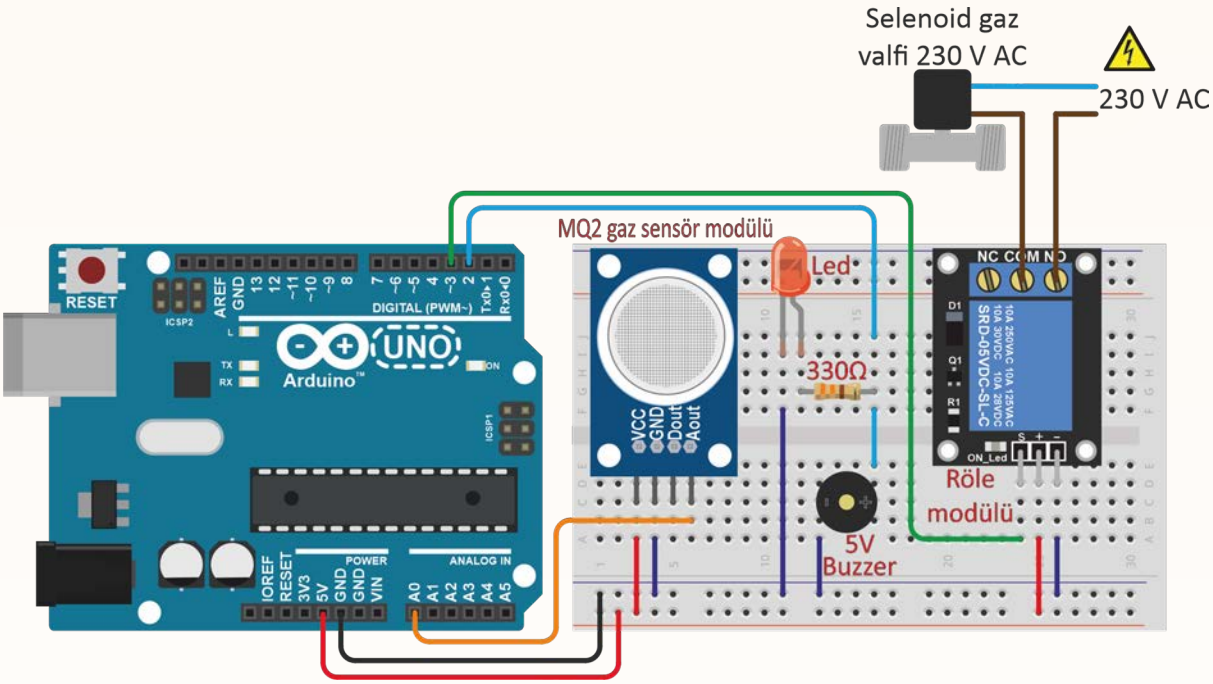
#include <Servo.h>                      // Servo motor kontrol kütüphanesini ekle.
int FanPin= A0;                         // A0 analog girişe bilgisayar fanı bağlandı.
int Servo1 = 2;                         // Servo motor1 kontrol pini 2 no.lu uca bağlandı.
int Servo2 = 3;                         // Servo motor2 kontrol pini 3 no.lu uca bağlandı.
int Pos1,Pos2,FanADC=0;                 // Değişken tanımlamaları.
int RuzgarOrani,PcGosterZaman=0;
Servo ServoMotor1;                      // 1.servo için ServoMotor1 adını kullan.
Servo ServoMotor2;                      // 2.servo için ServoMotor2 adını kullan.
void setup() {
 Serial.begin(9600);                    // Seri portu 9600 baud hız için ayarla.
 analogReference(INTERNAL);             // Analog referans gerilimini 1,1 V olarak ayarla.
 ServoMotor1.attach(Servo1);            // 2 no.lu çıkışı 1.servo motor için ayarla.
 ServoMotor2.attach(Servo2);            // 3 no.lu çıkışı 2.servo motor için ayarla.
}
void loop() {
 FanADC=analogRead(FanPin);             // 0 - 1,1 V gerilimi 0-1023 ADC'ye çevir.
 RuzgarOrani=map(FanADC,0,1023,0,30);   // 0-1023 değeri 0-30 dereceye çevir.
 Pos1=90-RuzgarOrani;                   // Servo motorları 90 dereceden başlatarak rüzgâr
 Pos2=90+RuzgarOrani;           // oranına göre birbirine ters olarak maksimum 30 derece döndür.
 ServoMotor1.write(Pos1);               // Açı değerini 1.servo motora gönder.
 ServoMotor2.write(Pos2);               // Açı değerini 2.servo motora gönder.
 PCgonder();                            // Rüzgâr ölçüm değerlerini PC'ye gönder.
}
void PCgonder () {
  delay(10);
  PcGosterZaman++;                      // Her 10 ms'de PcGosterZaman değişkenini 1 arttır.
if (PcGosterZaman>=100){                // Saniyede bir ekrana değerleri gönder.
   PcGosterZaman=0;
   Serial.print(" FanADC=");
   Serial.print(FanADC);
   Serial.print(" -- Ruzgar hız oranı=%");
   Serial.print(RuzgarOrani);
   Serial.print(" -- Servo Motor1=");
   Serial.print(Pos1);
   Serial.print(" derece");
   Serial.print(" -- Servo Motor2=");
   Serial.print(Pos2);
   Serial.println(" derece");
 }}
```

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Mekanik kısımların kurulumunu yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

11. UYGULAMA

ORTAMDAKİ YANICI GAZLARIN ALGILANMASI VE SELENOİD VALF İLE GAZIN KESİLMESİ

AMAÇ: **Gaz sensörünü kullanmak.**



Görsel 4.41: Uygulamaya ait devre bağlantı şeması

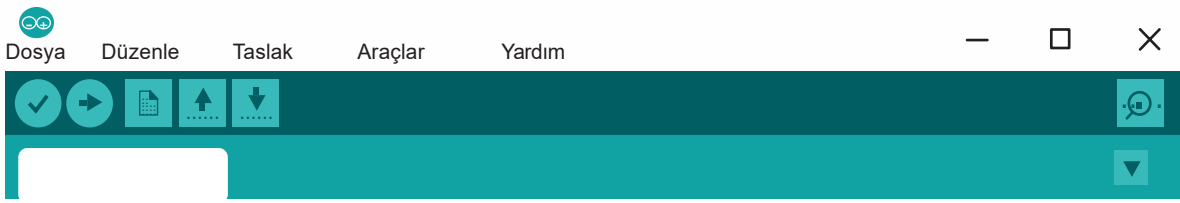
BİLGİ

Uygulamada MQ2 gaz sensör modülü kullanılmıştır. Aout (analog çıkış) gerilimi gaz yoğunluğuna göre değişim gösterir. Yazılımla belirlenen bir değerin üzerinde gaz algılandığında röle modülü enerjilenir ve ona bağlı gaz kapatma selenoid valfi devreye alınarak gaz girişi kesilir. Ayrıca LED ile görsel, buzzer ile sesli uyarı verilmektedir. Ortamdaki hava saf olmadığı için devre enerjilendiğinde düşük miktarda bir çıkış verecektir. Bunun için analog değer okunurken belli bir değer üzeri işleme alınmıştır.

İŞLEM BASAMAKLARI

1. İş sağlığı ve güvenliği tedbirlerini alarak çalışma ortamını hazırlayınız.
2. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
3. Görsel 4.41'de verilen devre kurulumunu gerçekleştiriniz.
4. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
5. Gaz sensörüne LPG (çakmak gazı) uygulayınız.
6. Seri port ekranını açınız ve değerleri gözleyiniz.
7. Gaz oranı belirlediğiniz değerin üstüne çıktığında selenoid valfin çalıştığını gözleyiniz.
8. Gaz oranı normal seviyeye düştüğünde selenoid valfin çalışmaya devam ettiğini gözleyiniz.
9. Selenoid valfi bir buton bağlayarak kapatmak için gerekli program ve donanım değişikliğini yapıp program ve donanımı yeniden yükleyiniz.
10. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM




```
                                    // Ortamdaki yanıcı gazların algılanması ve selenoid valf ile gazın kesilmesi.

int MQ2GazSns = A0;                 // MQ-2 gaz sensörü A0 analog pine bağlandı.
int LEDBuzzer = 2;                  // LED ve buzzer 2 no.lu pine bağlandı.
int Selenoid =  3;                  // Röle modülü 3 no.lu pine bağlandı.
int GazADC,GazOrani = 0;            // Değişken tanımlamaları.


void setup() {
  Serial.begin (9600);
  pinMode(MQ2GazSns, INPUT);
  pinMode(LEDBuzzer, OUTPUT);
  pinMode(Selenoid, OUTPUT);
  delay(3000);                      // Sensör rezistansının ısınması için 3 sn. bekle.
}

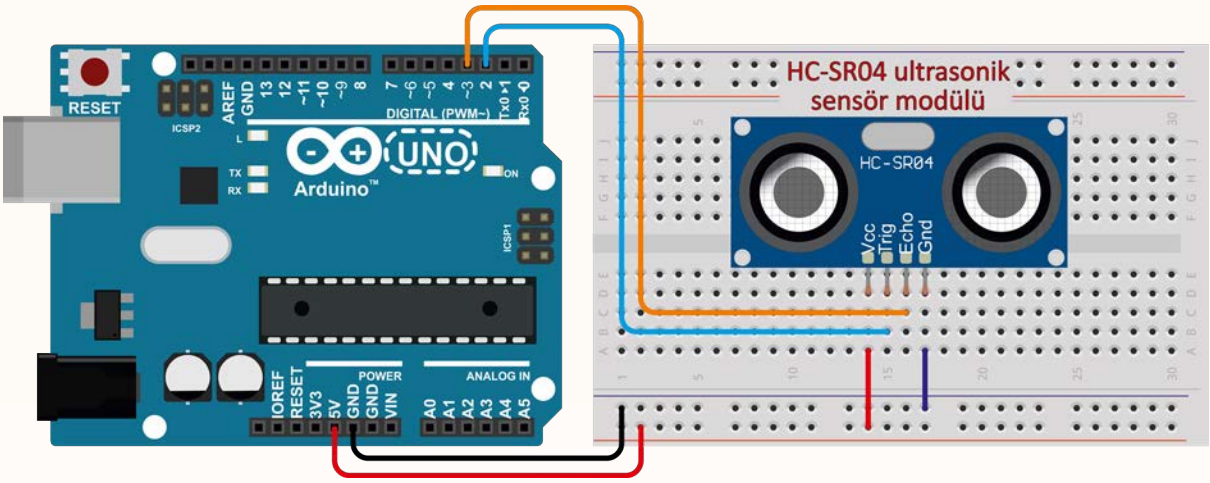
void loop() {
  GazADC=analogRead (MQ2GazSns);            // A0 analog kanaldan gaz değerini oku.
                                            // Ortamdaki hava saf olmadığı için 300 üzerini işleme al.
  GazOrani=map(GazADC,300,1023,0,100);      //300-1023 değerini 0-100'e çevir.
  if (GazOrani<0){GazOrani=0;}              // - değerleri gösterme.
  Serial.print ("GazADC:"); Serial.print (GazADC);
  Serial.print (" -- GazOranı:%"); Serial.println (GazOrani);
  if (GazOrani>50){                         // Gaz değeri %50'nin üzerine çıkarsa
     Serial.println (" !! Dikkat gaz kaçağı var.");
     digitalWrite(Selenoid, 1);             // Röleyi ve ona bağlı selenoidi çalıştır.
     digitalWrite(LEDBuzzer, 1); delay(250);  //Sesli ve ışıklı uyarı ver.
     digitalWrite(LEDBuzzer, 0); delay(250);
   }
   delay(500);
 }
```

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Donanım ve programda gerekli değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

12. UYGULAMA

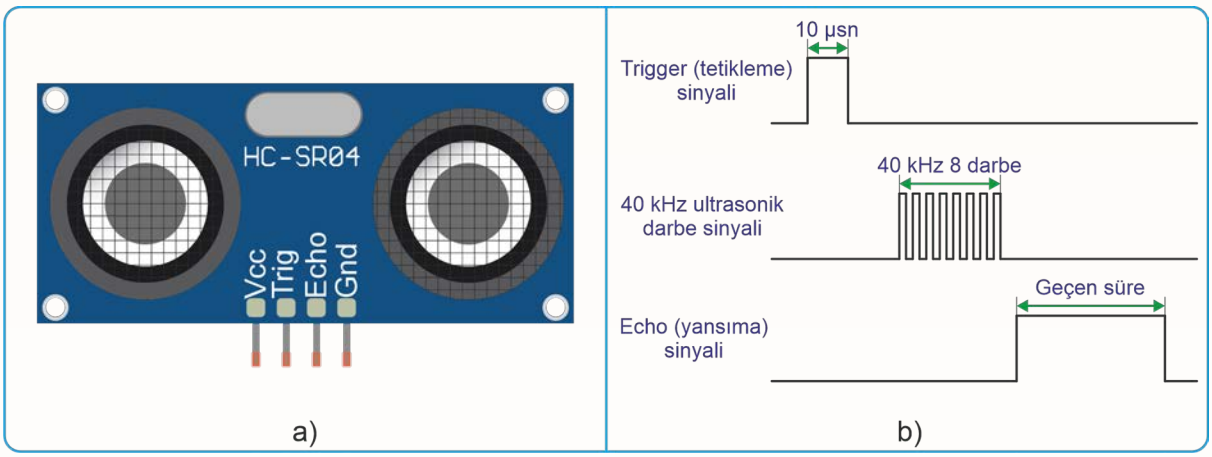
ULTRASONİK SENSÖR İLE ÖLÇÜLEN MESAFENİN BİLGİSAYARA GÖNDERİLMESİ

AMAÇ: Ultrasonik sensörü kullanmak.



Görsel 4.42: Uygulamaya ait devre bağlantı şeması

BİLGİ



Görsel 4.43: Ultrasonik sensör modülü ve çalışma prensibi

Uygulamada kullanılan HC-SR04 ultrasonik sensör modülü (Görsel 4.43:a), trigger girişinden 10 µs lojik 1 sinyali uygulandığında, modül içindeki kontrol devreleriyle 40 kHz'lik 8 pals oluşturarak vericiden gönderir ve zamanlayıcısını çalıştırır. Yansıma sinyali alındığında zamanlayıcıdan alınan toplam geçen süre kadar lojik 1 değerini echo (yankı) ucundan pals şeklinde verir (Görsel 4.43:b).

Mesafe hesaplaması şu şekildedir:

Mesafe (m) = [Geçen süre (µs) x ses hızı (0,000343)] / 2

Mesafe (cm) = [Geçen süre (µs) x ses hızı (0,0343)] / 2

Mesafe (mm) = [Geçen süre (µs) x ses hızı (0,343)] / 2

Programda geçen sürenin alınması için pulseIn (palsin) fonksiyonu kullanılmıştır. Bu fonksiyonun yazımı şu şekildedir:

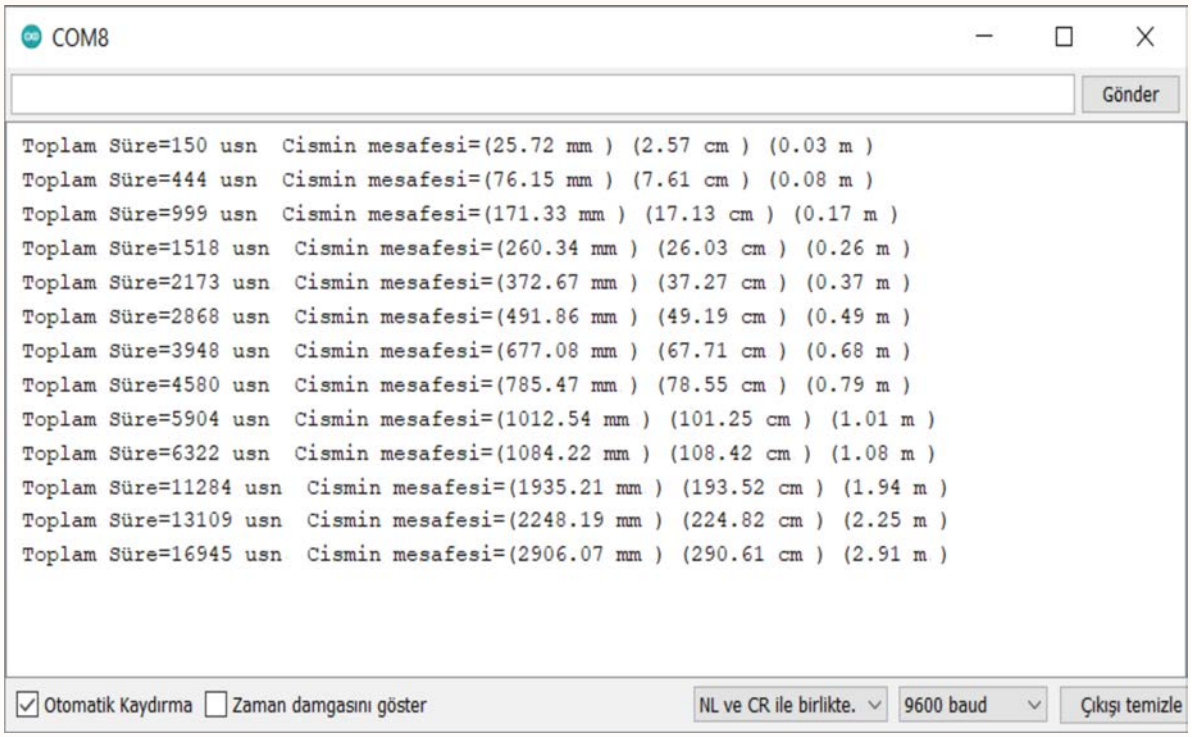
Değişken=pulseIn **(pin, değer, zaman aşımı)**

Değişken: Ölçülen sinyal süresinin atanacağı değişkendir. µs cinsinden büyük bir pozitif değer üretebileceği için bu değişkenin unsigned long tipi olması faydalıdır.

Pin: Lojik sinyalin okunacağı ucu gösterir.

Değer: Okunacak sinyal lojik 1 ise HIGH veya 1, lojik 0 ise LOW veya 0 yazılır.

Zaman aşımı: µs cinsinden belirtilen zaman aralığı kadar sinyalin okunması beklenir. Bu sürenin dışına çıkılırsa fonksiyondan çıkar. Zaman aşımı değeri yazılmazsa 1.000.000 µs (1 s) olarak kabul edilir.



Görsel 4.44: Ultrasonik sensöre ait seri port ekranı.

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 4.42'de verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. Görsel 4.44'teki gibi seri port ekranını açınız.
5. Ultrasonik sensör modülü önüne sabit bir cisim koyunuz.
6. Seri port ekranından cismin uzaklığını gözleyiniz.
7. Cisimle sensör modülü arasına cetvel koyarak ölçümün doğruluğunu gözleyiniz.
8. Ölçümde hatalar varsa formülde uygun düzeltmeleri yapınız ve gerekli program değişikliğini yaparak programı yeniden yükleyiniz.
9. Sensör modülünün minimum ve maksimum ölçüm aralığını büyük bir cisimle test ediniz.
10. Arduino dijital çıkışlarından birini kullanarak buzzer ekleyiniz.
11. Cismin, sensör modülüne 50 cm'den daha yakın olduğu durumda alarm verecek şekilde gerekli program değişikliğini yaparak programı yeniden yükleyiniz.
12. Araç park sensörlerinde olduğu gibi mesafeye göre buzzer alarm sıklığını artıracak şekilde gerekli program değişikliğini yaparak programı yeniden yükleyiniz.
13. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz

UYGULAMAYA AİT PROGRAM

Dosya Düzenle Taslak Araçlar Yardım

```
                                        // Ultrasonik sensör ile ölçülen mesafenin bilgisayara gönderilmesi
int TrigPin= 2;                         // Trigger 2 no.lu çıkışa bağlandı.
int EchoPin= 3;                         // Echo 3 no.lu girişe bağlandı.
unsigned long TopGecenSure;             // Değişken tanımlamaları.
float TopMesafe,Mesafemm=0;

void setup(){
 pinMode(TrigPin, OUTPUT); pinMode(EchoPin, INPUT);
 Serial.begin(9600);                                  // Seri portu 9600 baud hız için ayarla.
}
void loop() {
                                                      // Sensör trigger ucundan 10 usn'lik pals ver.
 digitalWrite(TrigPin,1);
 delayMicroseconds(10);digitalWrite(TrigPin, 0);
                                        // EchoPin lojik 1 süresini ölç. Zaman aşımı yazılmadığı için 1 sn. olur.
 TopGecenSure = pulseIn(EchoPin, 1);    // Toplam geçen süreyi hesapla.
 TopMesafe = TopGecenSure * 0.343;      // Toplam alınan mesafeyi (mm) hesapla.
 Mesafemm = TopMesafe/2;                //Toplam mesafeyi gidiş dönüş olduğu için 2'ye böl.
 PCgonder();                            // Ölçüm değerlerini PC'ye gönder.
}

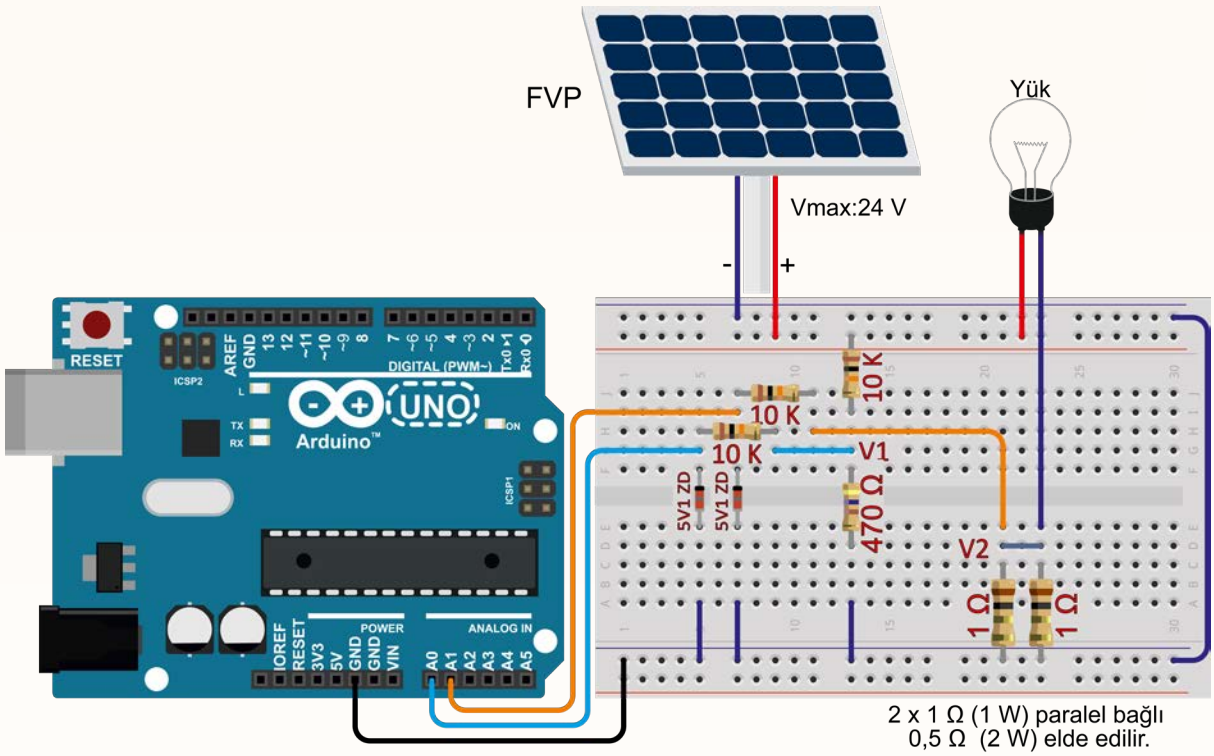
void PCgonder () {
 Serial.print(" Toplam Süre="); Serial.print(TopGecenSure);
 Serial.print(" usn");
 Serial.print(" Cismin mesafesi=("); Serial.print(Mesafemm);
 Serial.print(" mm)");
 Serial.print(" ("); Serial.print(Mesafemm/10); Serial.print(" cm)");
 Serial.print(" ("); Serial.print(Mesafemm/1000);
 Serial.println(" m)");
 delay(500); // Ölçümler arasında 500 ms bekle.
}
```

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Donanım ve programda gerekli değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

13. UYGULAMA

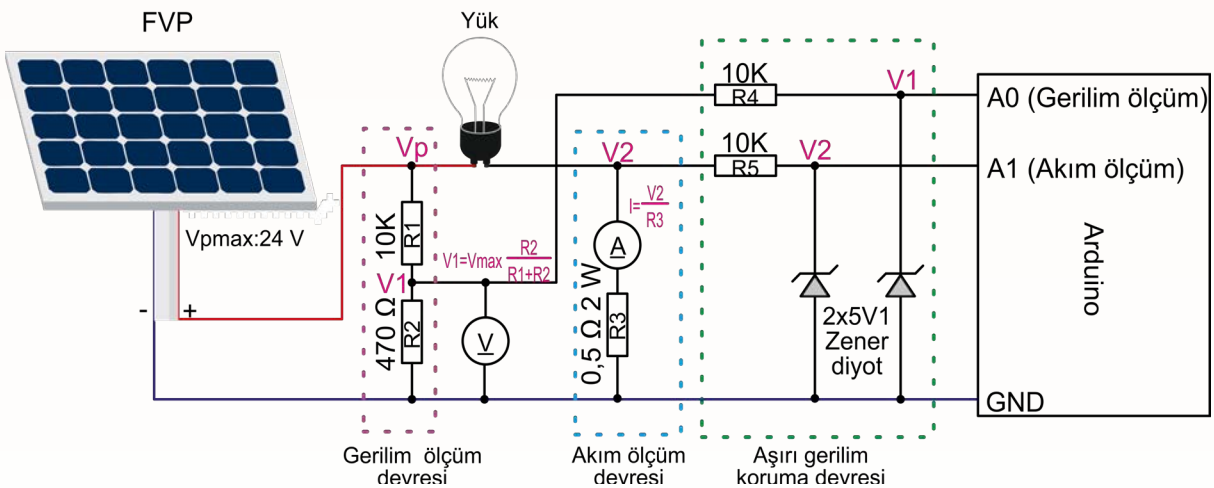
GÜNEŞ PANELİNİN AKIM, GERİLİM VE GÜÇ DEĞERLERİNİN BİLGİSAYARA AKTARILMASI

AMAÇ: Güneş panelinin elektriksel değerlerini ölçmek.



Görsel 4.45: Uygulamaya ait devre bağlantı şeması

BİLGİ



Görsel 4.46: Güneş panelinin elektriksel değerlerinin ölçümüne ait devre şeması

Görsel 4.46'da güneş panelinin (FVP) elektriksel değerlerinin ölçümüne (akım, gerilim ve güç ölçümü) ait devre şeması verilmiştir. R1 ve R2 dirençleriyle gerilim bölücü olarak gerilim değeri ölçümü, R3 direnciyle akım ölçümü sağlanmıştır. Aşırı gerilim koruma devresi, mikrodenetleyici analog girişlerini aşırı gerilimden korumak için eklenmiştir. A0 ve A1 girişlerine 1,5-5 V arası zener diyot kullanılabilir.

Gerilim ölçümü: Ölçülecek gerilim, mikrodenetleyicinin AREF ile belirlenen ölçme sınırından büyükse (örneğin +5 V veya dâhilî +1,1 V) gerilim bölücü dirençlerle değeri düşürülerek ölçüm yapılır. Vpmax: +24 V olan bir güneş panelinde V1=(Vp) x R2 / (R1+R2) formülü ile V1 gerilimi dolayısıyla

mikrodenetleyici A0 analog kanalına gelen gerilim değeri hesaplanır.

Özellikle akım ölçümü düşük değerli direnç üzerinden alınacağı için gerilim düşümü az olacaktır. Daha küçük değerleri hassas olarak ölçebilmek için mikrodenetleyici dâhilî AREF 1,1 V olarak ayarlanmıştır. Bu nedenle ölçülecek gerilim aralığı 0-1,1 V arasında olmalıdır. Bu değerleri sağlayacak standart direnç değerleri seçilerek istenen gerilim değeri 1,1 V'a en yakın olacak şekilde hesaplanmıştır. Vp=24 V, R1=10 K, R2=470 Ω Bu değerlere göre V1=24 x 470 (10000+470) =1,077 V bulunur. Böylelikle güneş panelinin 0-24 V arasındaki değerleri V1 noktasında 0-1,077 V (1077 mV) arasına düşürülmüş olur. 0-1,1 V (1100 mV) değere karşılık ADC değeri 0-1023 olacaktır.

Mikrodenetleyici bilgi sayfasında dâhilî AREF (1,1 V) 1-1,2 V arasında toleransa sahiptir. Bu durum standart 1,1 V alındığında hatalara neden olabilir. Hataları en aza indirerek doğru gerilim değerinin ölçülmesi için güneş paneli bağlanmadan önce Vp gerilimi sabit bir gerilim kaynağıyla (+10 V gibi) beslenmeli ve multimetre ile kontrol edilmelidir. Gerilim kaynağı sabitlendikten sonra VADC değeri seri port ekranından alınmalıdır. Bu değerler oranlanarak bir çarpan değeri elde edilir ve çıkış gerilimi daha doğru ölçülür. Yapılan işlem mikrodenetleyici analog ölçümünü multimetreye göre kalibre etmektir, bu nedenle ölçüm alınacak multimetrenin doğru ölçümü göstermesi gerekir.

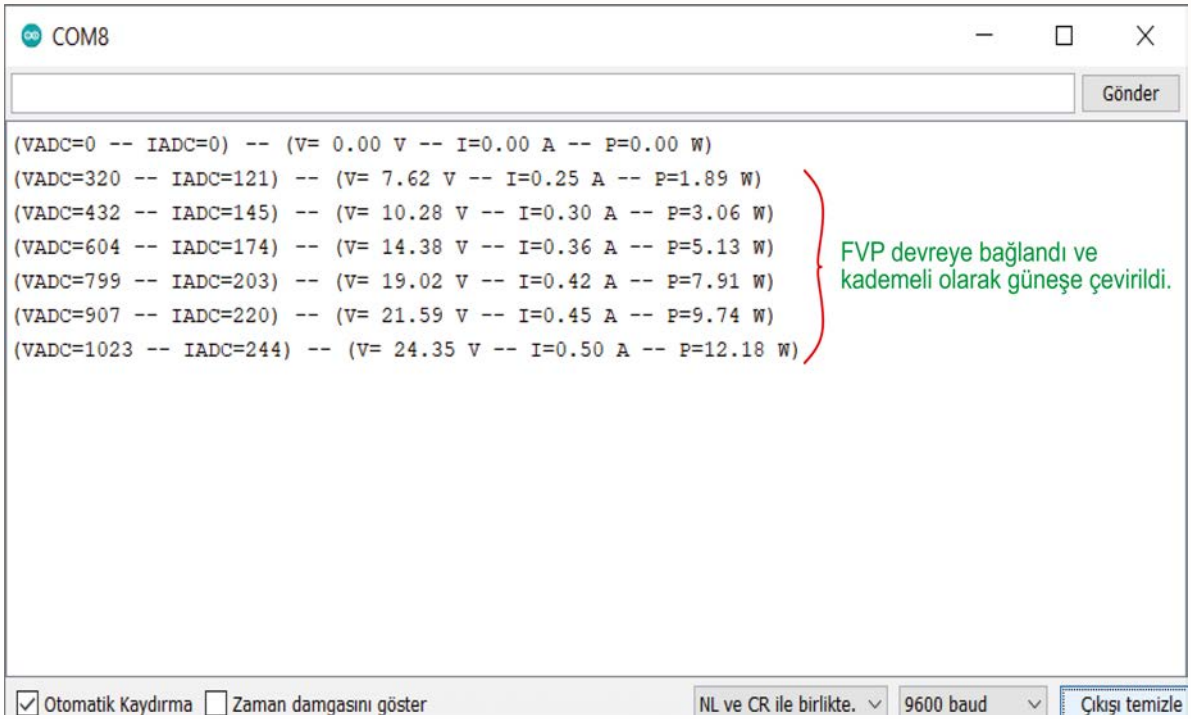
Uygulamada sabit gerilim +10 V ayarlandı ve VADC değeri seri port ekranından 420 olarak görüldü. Çarpan değeri (V)=10/420 ≅ 0,0238 olarak bulundu. Programda gerilim hesaplaması;

Gerilim=çarpan değeri (V) x VADC (Gerilim=0,0238 x VADC) değeri olarak basitleştirilmiş olur.

Akım ölçümü: Akım ölçümünde birçok elektronik devre ve hazır entegre vardır. En basit yöntem, yüke seri bağlanan düşük değerli bir direnç üzerinde düşen gerilimi ölçmektir. Görsel 4.46'da bulunan R3 direnci bu amaçla kullanılmıştır. Yük akımı arttıkça R3 direnci üzerindeki gerilim de artar. Ohm kanunu hesaplamasıyla akım bulunabilir ancak AREF kaynaklı hataları en aza indirmek için R3 direncine seri bağlı bir ampermetreyle değer okunur ve IADC değeriyle oranlanarak bir çarpan değeri elde edilir. Uygulamada ampermetre 0,284 A ve IADC değeri 138 olarak görüldü. Çarpan değeri=0,284/138≅0,00205 olarak bulundu. Programda akım hesaplaması;

Akım= çarpan değeri (I) x IADC (Akım=0,00205 x IADC) değeri olarak bulunur.

Güç ölçümü (DC): Yük üzerindeki gerilim ve çektiği akım bilinirse güç hesaplaması P=V x I formülüyle hesaplanır.

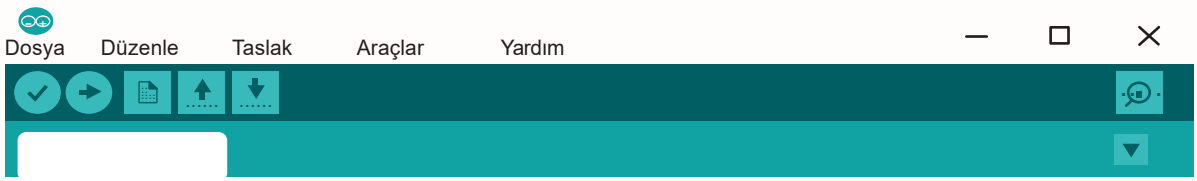


Görsel 4.47: Güneş panelinin elektriksel değerlerine ait seri port ekranı

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 4.45'te verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. Görsel 4.47'deki gibi seri port ekranını açınız.
5. Güneş paneli yerine sabit bir gerilim kaynağı bağlayınız ve Vp gerilimini multimetreyle ölçünüz, seri port ekranından VADC değerini alınız ve çarpan değeri (V) hesaplayınız.
6. Devreye yük bağlayınız ve multimetreyle akımını ölçünüz, seri port ekranından IADC değerini alınız ve çarpan değeri (I) hesaplayınız.
7. Hesapladığınız çarpan değerlerini program değişikliğini yaparak yeniden yükleyiniz.
8. Multimetreyle okuduğunuz gerilim ve akım değerlerinin seri port ekranındaki değerlere çok yakın olduğunu gözleyiniz.
9. Sabit gerilim kaynağını devreden çıkarıp yerine güneş panelini bağlayınız. Güneş panelini en yüksek güç değerlerini verecek şekilde ayarlayınız.
10. Röle modülü kullanarak yükün aşırı güç harcaması (örneğin 10 W üstü) durumunda, güneş panelini devreden çıkaracak gerekli donanım ve program değişikliğini yaparak donanım ve programı yeniden yükleyiniz.
11. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM



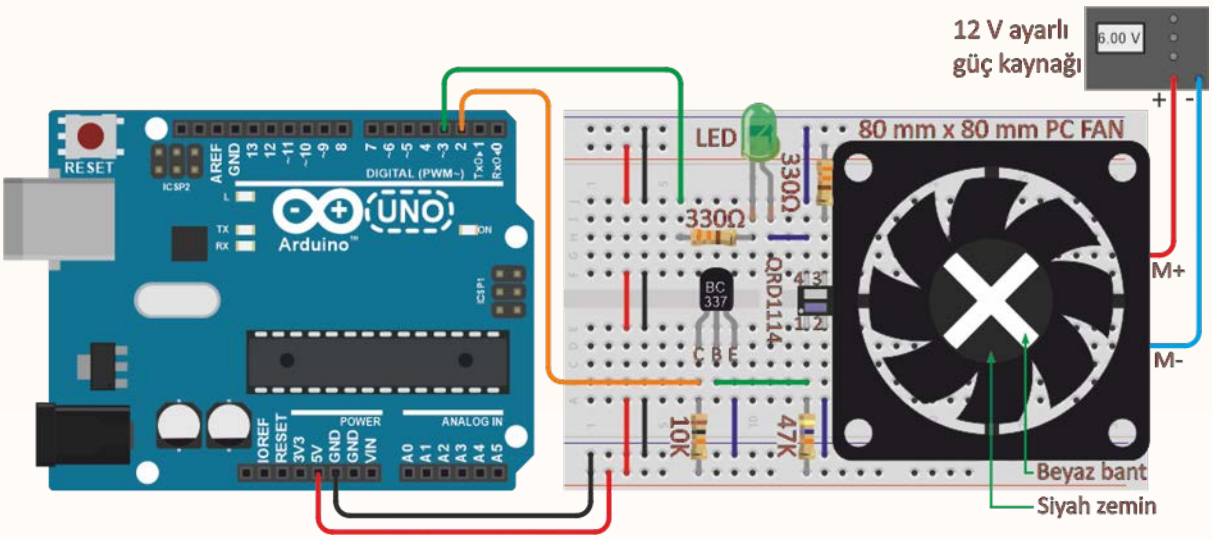

```
                                    // Güneş panelinin akım, gerilim ve güç değerlerini PC'ye göndermek
int VPin= A0;                       // Gerilim ölçme A0 analog girişe bağlandı.
int IPin= A1;                       // Akım ölçme A1 analog girişe bağlandı.
int VADC,IADC=0; float V,I,P=0;     // Değişken tanımlamaları.
void setup(){
 analogReference(INTERNAL);         // Analog referans gerilimini dâhilî 1,1 V olarak ayarla.
 Serial.begin(9600);}               // Seri portu 9600 baud hız için ayarla.
void loop() {
 VADC=analogRead(VPin);             // 0 - 1,1 V gerilimi 0-1023 ADC'ye çevir.
 IADC=analogRead(IPin);             // 0 - 1,1 V gerilimi 0-1023 ADC'ye çevir.
 V=0.0238 * VADC;                   // Gerilim= Çarpan değeri(V) x VADC değeri.
 I=0.00205 * IADC;                  // Akım=Çarpan değeri(I) x IADC değeri.
 P=V * I;                           // Güç=Gerilim x akım.
 PCgonder();                        // Ölçüm değerlerini PC'ye gönder.
}
void PCgonder () {
 Serial.print("(VADC=");    Serial.print(VADC);
 Serial.print(" -- IADC="); Serial.print(IADC);
 Serial.print(") -- (V= "); Serial.print(V);
 Serial.print(" V -- I=");  Serial.print(I);
 Serial.print(" A -- P=");  Serial.print(P); Serial.println(" W)");
 delay(500);                        // 500 ms aralıklarla ölçümü tekrarla.
}
```

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Donanım ve programda gerekli değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

14. UYGULAMA

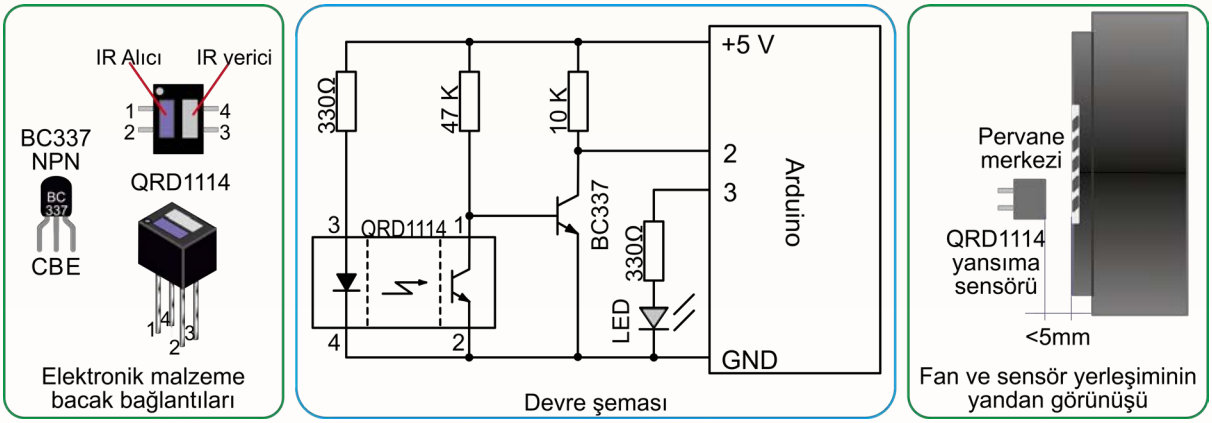
TÜRBİN DEVİR SAYISININ CİSİMDEN YANSIMALI SENSÖRLE ÖLÇÜLMESİ

AMAÇ: **Türbin devir sayısını ölçmek.**



Görsel 4.48: Uygulamaya ait devre bağlantı şeması

BİLGİ



Görsel 4.49: Türbin devir sayısının cisimden yansımalı sensörle ölçülmesine ait devre şeması

QRD1114, içinde bir adet IR alıcı (fototransistör) ve verici LED bulunduran, cisimden yansımalı sensördür. Yansıtıcı yüzeye yakınlığı veya bu yüzeyin yansıtma oranında analog çıkış verir. Genellikle çıkışı elektronik devrelerle lojik 1 veya lojik 0 seviyesine çevrilir.

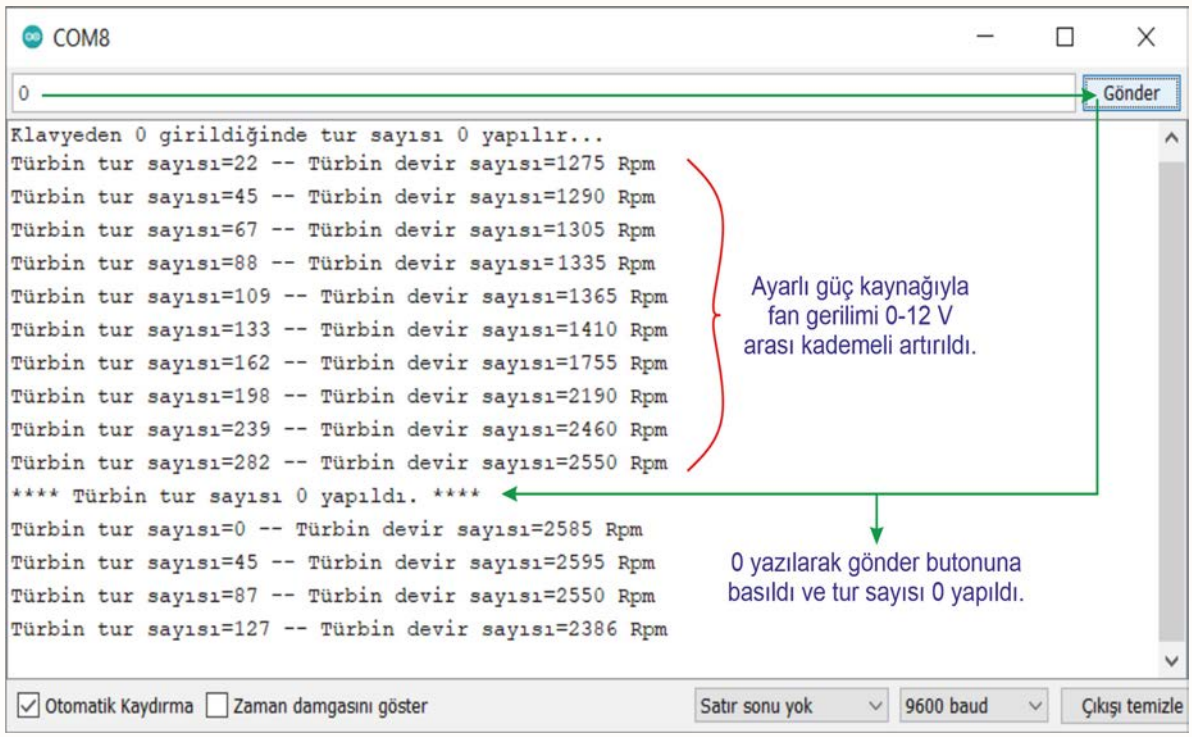
Görsel 4.49'daki devreye enerji verildiğinde IR verici LED ışık yaymaya başlar. Eğer yansıtıcı bir yüzey yoksa fototransistör kollektörü (1 no.lu uç) +5 V'a yakın analog değer verir. Bu durumda BC337 transistör iletime geçerek kollektörü ve mikrodenetleyicinin 2 no.lu ucunu lojik 0 seviyesine çeker. Eğer yansıtıcı bir yüzey olursa mikrodenetleyicinin 2 no.lu ucu lojik 1 seviyesinde olacaktır. Kısacası sensör devresi, beyaz gördüğünde lojik 1, siyah gördüğünde lojik 0 olur.

Görsel 4.48'deki uygulamada türbini temsilen fan kullanılmıştır. Fanın merkezinde bulunan dairesel alan siyah renkte (yansıtıcı olmayan yüzey) olduğu için bu alana artı (+) şeklinde beyaz bant (yansıtıcı yüzey) yapıştırılmıştır. QRD1114 sensörü ve fan, Görsel 4.49'daki gibi yandan yerleşim görüntüsüne göre yerleştirilmelidir. Sensörün, beyaz ve siyah kısımları tam olarak algılayabilmesi için fana olan mesafesi 5 mm'den küçük olmalıdır. Yerleşim ve devre kurulumu doğru yapıldığında devre çıkışı pals şeklinde lojik 1(beyaz) ve lojik 0 (siyah) olacaktır. Fan bir tur döndüğünde üzerindeki beyaz bant artı (+) şeklinde

yerleştirildiği için 4 adet pals çıkışı verecektir. Mikrodenetleyiciyle bu palsler sayılarak fanın tur sayısı ve dakikadaki devir sayısı (Rpm- Revolutions per minute) ölçülebilir.

Uygulama programı, kesme işlemine güzel bir örnek olması açısından önemlidir. Tur sayısını ölçmek için kesme pals sinyalinin yükselen kenarında (sensör beyaz bant üzerinde lojik 1'e yükseldiği için) pals sayılır. Pals sayısı 4 olduğunda 1 tur atılmış olur. Tur değişkeni 1 artar ve LED yanıp söner. Seri port ekranında tur sayısını 0 yapmak için 0 yazılarak "gönder" butonuna basılmalıdır. Devir sayısı ölçümü için 1 dk. beklemek yerine 1 sn.deki tur sayısı 15 ile çarpılır (60 sn/4 pals). Devir sayısı 15'in katları şeklinde hesaplanacaktır. Eğer fan merkezi, beyaz bant ile 30 eşit parçaya bölünseydi devir sayısı hesabı için 1 sn.deki tur sayısı 2 ile çarpılacaktı. Bölme sayısı arttıkça ölçüm hassasiyeti de artacaktır.

Uygulamada türbini temsilen kullanılan fan, ayarlı bir güç kaynağı ile beslenmelidir. Böylelikle fan gerilimiyle orantılı olarak devir sayısı değişimi gözlenebilir. Ayarlı güç kaynağında ayarlanan gerilim değeri fanın normal besleme gerilimini aşmamalıdır. Ayrıca yansıma sensörü dış ortamdaki ışıktan etkilenebileceği için ölçüm yerinin karanlık bir ortamda olmasına dikkat edilmelidir.

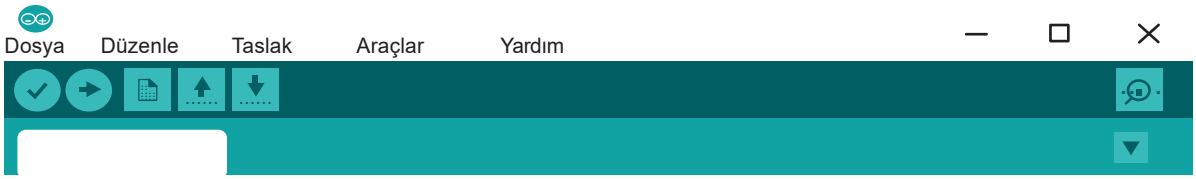


Görsel 4.50: Türbin devir sayısının ölçümüne ait seri port ekranı

İŞLEM BASAMAKLARI

1. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
2. Görsel 4.48'de verilen devre kurulumunu gerçekleştiriniz.
3. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
4. Görsel 4.50'deki gibi seri port ekranını açınız.
5. Görsel 4.48'deki gibi fan merkezi üzerindeki siyah dairesel alanı beyaz bantla + şeklinde dört eşit parçaya bölünüz.
6. Görsel 4.49'daki yandan yerleşim görüntüsüne göre fan ve yansıma sensörünü mesafeyi de dikkate alarak yerleştiriniz.
7. Devreye buzzer ekleyiniz.
8. Devir sayısı 1500'ü geçtiğinde alarm verecek şekilde gerekli donanım ve program değişikliğini yaparak donanım ve programı yeniden yükleyiniz.
9. Ayarlı güç kaynağını devreden çıkarınız.
10. Bu uygulamayı devir ölçme şeklinde rüzgârölçere (anemometre) çevirmek için gerekli donanım ve program değişikliğini yaparak donanım ve programı yeniden yükleyiniz.
11. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

UYGULAMAYA AİT PROGRAM




```
                                  // Türbin devir ve tur sayısının cisimden yansımalı sensörle ölçülmesi.
 int YansimaSens = 2;             // Yansıma sensörü 2 no.lu uca (0.kesme girişi) bağlandı.
 int LEDYesil = 3;                                    // Yeşil LED 3 no.lu uca bağlandı.
 int TurbinDevir,TurbinTur,PalsSay,SeriPortVeri=0;    // Değişken tanımı.

void setup() {
 pinMode(LEDYesil, OUTPUT);
 pinMode(YansimaSens,INPUT);
 Serial.begin(9600);                                   // Seri portu 9600 baud hız için ayarla.
 Serial.println("Klavyeden 0 girildiğinde tur sayısı 0 yapılır...");
 attachInterrupt(0, YansimaKesmesi, RISING);
                                  // Uno 0.dış kesme, Yansıma sensörü beyaz noktaya geldiğinde
                                  // Kesme oluştur ve YansimaKesmesi fonksiyonuna çağır.
 }
void loop() {
                                  // Fan merkezi beyaz bant ile 4'e bölündüğü için bir turda 4 sayım yapılır.
                                  // Dakikadaki devir hesabı için 1 sn. içinde sayımı 15 ile çarp.
 TurbinDevir=TurbinDevir*15;                          // Dakikadaki dönüş sayısını (Rpm) hesapla.

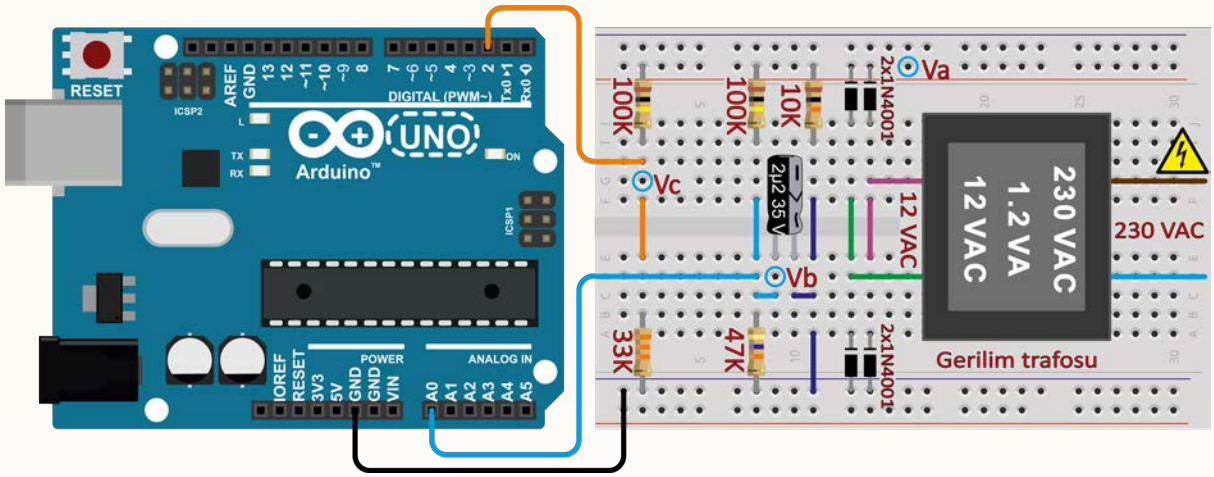
  if (Serial.available()) {                           // Seri porttan gelen veri kontrolünü yap.
    SeriPortVeri = Serial.read();                     // Seri porttan gelen veriyi oku.
  // Seri porttan 0 verisi gelirse PalsSay ve TurbinTur değişkenlerini 0 yap.
      if (SeriPortVeri=='0') {
        TurbinTur=0;PalsSay=0;
        Serial.println("**** Türbin tur sayısı 0 yapıldı. ****");
      }
   }
 Serial.print("Türbin tur sayısı=");   Serial.print(TurbinTur);
 Serial.print(" -- Türbin devir sayısı="); Serial.print(TurbinDevir);
 Serial.println(" Rpm");
 TurbinDevir=0;
 delay(1000);                                         // 1 sn. aralıklarla ölçüm almak için 1 sn. bekle.
}
void YansimaKesmesi () {                              // Kesme oluştuğunda çağrılacak fonksiyon.
  digitalWrite(LEDYesil, 0);                          // Yeşil LEDi söndür.
  TurbinDevir++;
  PalsSay++;
    if (PalsSay>=4){                                  // 4 pals sayımı bir tam tur olur.
     TurbinTur++; PalsSay=0;                          // TurbinTur 1 artır. PalsSay 0 yap.
     digitalWrite(LEDYesil, 1);                       // Yeşil LEDi 1 tur olduğu için yak.
    }
}
```

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Donanım ve programda gerekli değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

15. UYGULAMA

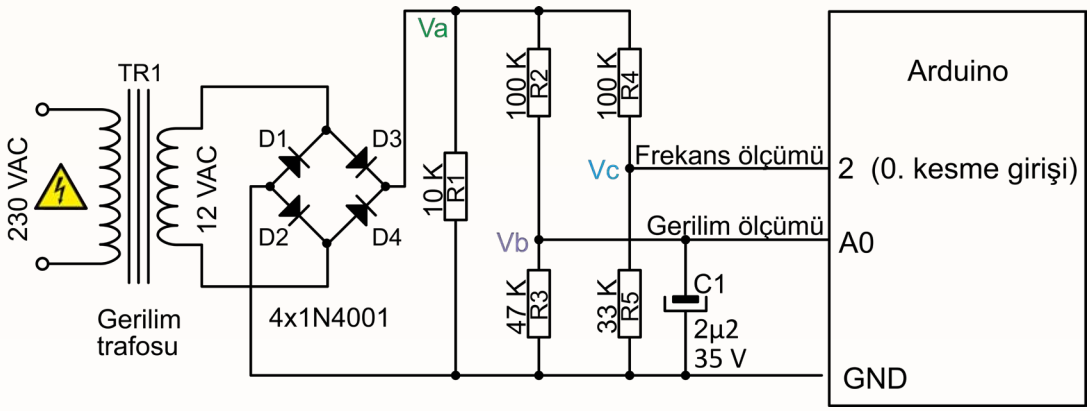
GERİLİM TRAFOSUYLA ŞEBEKE GERİLİM VE FREKANSININ İZOLELİ ÖLÇÜMÜ

AMAÇ: Şebeke gerilim ve frekansını ölçmek.



Görsel 4.51: Uygulamaya ait devre bağlantı şeması

BİLGİ



Görsel 4.52: Gerilim trafosuyla şebeke gerilim ve frekansının ölçümüne ait devre şeması

Yüksek gerilim değerini ölçmek için gerilim seviyesi, gerilim bölücü dirençlerle ölçülebilecek düzeye indirilmelidir. Gerilim bölücü dirençlerle şebekeden direkt AC gerilimi ölçmek mümkündür ancak yüksek gerilimden izole edilmediği için risk taşır. Bu nedenle gerilim trafosuyla ölçülecek AC gerilim, şebekeden izole olarak risk oluşturmayacak seviyeye düşürülür.

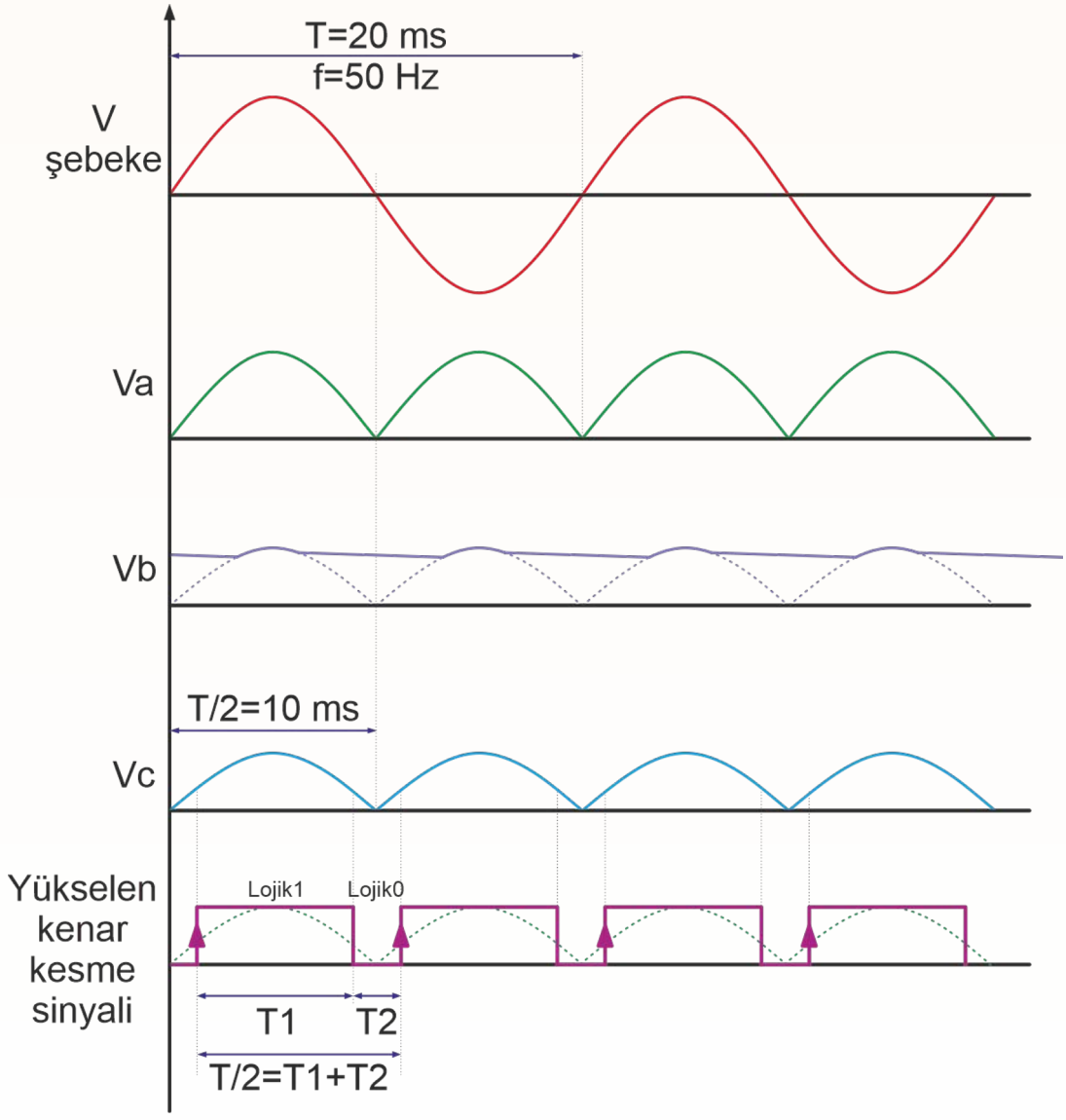
Görsel 4.52'de şeması verilen devrede, şebeke gerilimi TR1 trafosuyla 12 VAC seviyesine düşürülür. Diyotlardan oluşan köprü tipi tam dalga doğrultma devresiyle negatif alternanslar da pozitife çekilir. Görsel 4.53'te Va dalga şekli görülür. Devrede kullanılan direnç değerleri büyük olduğu için Va gerilimi 0 V'a tam olarak düşemez. R1 direnci bu sorunu gidermek için kullanılmıştır.

Gerilim ölçme: Va noktasındaki gerilim, R2 ve R3 gerilim bölücü dirençlerle mikrodenetleyicinin ölçebileceği gerilim seviyesine indirilir. C1 kondansatörüyle filtre edilerek DC'ye çok yakın dalga şekli elde edilir. Görsel 4.53'te Vb dalga şekli görülmektedir. Vb gerilimi, şebeke gerilimiyle doğru orantılı olarak değişim gösterir. Mikrodenetleyici analog girişinden bu değişim değeri okunur (ADC). Şebeke gerilimi multimetreyle ölçülüp oranlanarak çarpan değeri bulunur ve sonuç hesaplanır.

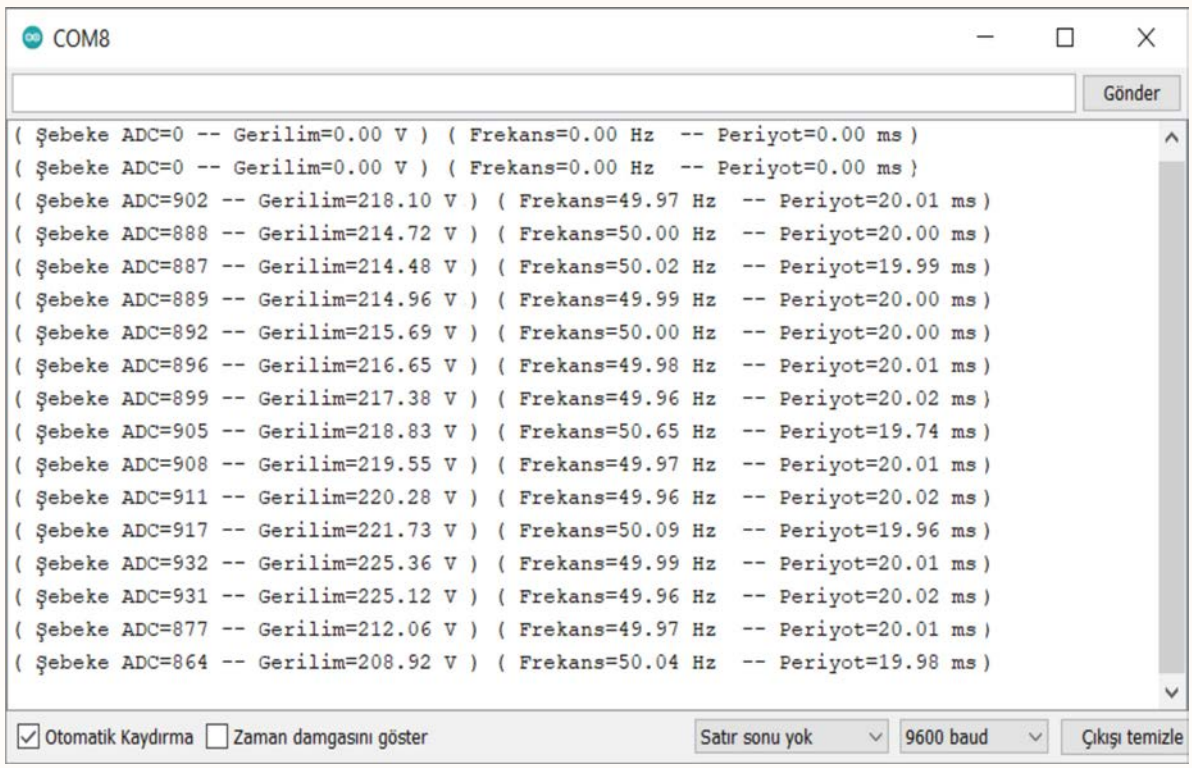
Frekans ölçme: Va noktasındaki frekansı ölçülecek gerilim, R4 ve R5 gerilim bölücü dirençlerle mikrodenetleyicinin ölçebileceği gerilim seviyesine indirilir. Görsel 4.53'te Vc dalga şekli görülür.

Frekans (f) saniyede oluşan periyot (T) sayısıdır. f=1/T formülüyle bulunur. Bu hesaplamanın yapılabilmesi için periyot süresinin ölçülmesi gerekir. Örneğin 50 Hz şebeke frekansının periyodu T=1/50 = 20 ms'dir.

Vc giriş sinyali mikrodenetleyicinin 2 no.lu ucuna (0. kesme girişi) bağlanmış ve programda yükselen kenar kesme fonksiyonu oluşturulmuştur. Vc geriliminin belli bir noktasından sonra kesme oluşur (Mikrodenetleyicilerde lojik 1 veya lojik 0 seviyesinin algılanması için giriş geriliminin belli bir değere çıkması veya düşmesi gerekir). Kesme, Vc geriliminin belli bir noktasında devreye gireceğinden doğru ölçüm için önemlidir. Kesme fonksiyonunda palsin fonksiyonuyla ilk önce Vc geriliminin oluşturacağı lojik 1 süresi (T1) ve sonra lojik 0 süresi (T2) ölçülür. Bu iki süre toplanır. Toplam süre gerçek şebeke periyodunun yarısı kadardır (T/2=T1+T2). Gerekli hesaplamalarla frekans değeri bulunarak seri porttan gönderilir. Palsin fonksiyonu lojik 1 veya lojik 0 süresini µs cinsinden vereceği için hesaplamalarda buna dikkat edilmelidir.



Görsel 4.53: Gerilim dalga şekilleri



Görsel 4.54: Trafoyla şebeke gerilim ve frekans ölçümüne ait seri port ekranı

İŞLEM BASAMAKLARI

1. İş sağlığı ve güvenliği tedbirlerini alarak çalışma ortamını hazırlayınız.
2. Uygulamada kullanılacak malzemelerin teknik bilgilerini üretici firmanın bilgi sayfasından alınız.
3. Görsel 4.51'de verilen devre kurulumunu gerçekleştiriniz.
4. Uygulamaya ait programı bilgisayarda yazınız ve mikrodenetleyici karta yükleyiniz.
5. Görsel 4.54'teki gibi seri port ekranını açınız.
6. Multimetreyle şebeke gerilimini ölçünüz, seri port ekranından şebeke ADC değerini alınız ve çarpan değerini hesaplayınız.
7. Hesapladığınız çarpan değerini program değişikliğini yaparak yeniden yükleyiniz.
8. Multimetreyle okuduğunuz şebeke gerilim değerlerinin seri port ekranındaki gerilim değerlerine çok yakın olduğunu gözleyiniz.
9. Çalışma sonrası devre elemanlarını dikkatlice sökünüz, öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz

UYGULAMAYA AİT PROGRAM

Dosya Düzenle Taslak Araçlar Yardım

```
                                        // Trafoyla şebeke gerilim ve frekans değerlerinin ölçülmesi.
 int FrekansVb=2;                       // Şebeke frekans 2 no.lu uca (0.kesme girişi) bağlandı.
 int GerilimVc=A0;                      // Şebeke gerilim ölçüm ADC pin A0'a bağlandı.
 int GerilimADC=0;                      // Değişken tanımlamaları.
 long T1,T2;
 float Gerilim,Frekans,Periyot=0;
 void setup() {
  pinMode(FrekansVb,INPUT);
  Serial.begin(9600);                   // Seri portu 9600 baud hız için ayarla.
  KesmeAktif();delay(50);               // Kesme fonksiyonunu yükselen kenar olarak
                                        // aktif yap ve şebekeVc belli bir değerin üzerindeyse kesme oluştur.
 }
void KesmeAktif(){attachInterrupt(0, FrekansKesmesi, RISING);}
void KesmePasif(){detachInterrupt(0);}          // Kesme fonksiyonunu pasif yap.

void loop() {
 GerilimADC=analogRead(GerilimVc);              // Ölçülen 223 V ADC karşılığı 922
 Gerilim=GerilimADC*0.2418;                     // 223/922=0,2418 çarpan değerinin hesabı.
   Serial.print("( Şebeke ADC=");  Serial.print(GerilimADC);
   Serial.print(" -- Gerilim=");   Serial.print(Gerilim);
   Serial.print(" V) ( Frekans="); Serial.print(Frekans);
   Serial.print(" Hz");
   Serial.print(" -- Periyot=");    Serial.print(Periyot);
   Serial.println(" ms)");
   delay(1000);                                 // Saniyede bir ölçüm al.
   KesmeAktif();                                // Yeni ölçüm için kesmeyi tekrar aktif yap.
}
void FrekansKesmesi () {                        // Kesme fonksiyonu.
  KesmePasif();                                 // Hesaplama ve ölçüm bitmeden yeni bir kesmeyi engelle.
  T1=pulseIn(FrekansVb,1);  T2=pulseIn(FrekansVb,0);    // T1,T2 süre ölçümü.
  Periyot=(T1+T2)/500.0;  Frekans=(1000.0/Periyot);     // Frekans hesabı.
  if (Periyot==0){Frekans=0;}                           // Sonuç 0 çıkarsa 0'a bölme hatasını engelle.
}
```

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Çalışma ortamını temizleyerek düzenledi.	20	
Sınıfı :	2	Uygulamayı verilen süre içinde yaptı.	20	
Numarası :	3	Devre elemanlarını doğru bağladı.	20	
ÖĞRETMEN	4	Uygulama programını doğru yazdı.	20	
Adı Soyadı :	5	Donanım ve programda gerekli değişiklikleri yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki cümlelerin başındaki boşluğa cümleler doğru ise (D), yanlış ise (Y) yazınız.

1. (…) Frekans (f) saniyede oluşan periyot (T) sayısıdır.
2. (…) DC devrede güç ölçümü; gerilim, akım ve direncin çarpımına eşittir.
3. (…) Herhangi bir yük, mikrodenetleyici portlarına doğrudan bağlanabilir.

B) Aşağıdaki cümlelerde boş bırakılan yerlere uygun sözcüğü yazınız.

4. DC motorların devir yönünü değiştirmek için uygulanan DC gerilimin …… değiştirilmelidir.
5. Servo motorlar…… sinyaliyle kontrol edilir.
6. analogReference(INTERNAL); fonksiyonuyla arduino için dâhilî AREF …… olarak ayarlanır.

C) Aşağıda verilen ifadeler ve bu ifadelere karşılık gelen kavramları, ifadelerin önündeki parantezlere kavramların önündeki harflerden uygun olanları yazarak eşleştiriniz.

İFADELER		
7.	(…)	Önceliği yüksek sinyal
8.	(…)	Rotary encoder
9.	(…)	Haberleşme portu
10.	(…)	Yüksek gerilim değerini ölçmek
11.	(…)	Rüzgârölçer

KAVRAMLAR	
A	Anemometre
B	Seri port
C	Gerilim bölücü direnç
D	Kesme
E	Döner kodlayıcı
F	Solarimetre

Ç) Aşağıdaki soruların doğru seçeneğini işaretleyiniz.

12. Bobin uçlarına verilen pals ile açısal dönme gerçekleştiren motor aşağıdakilerden hangisidir?
A) Asenkron B) Dahlender C) Senkron
D) Step E) Servo

13. Döner bir cismin dakikadaki devir sayısı birimi aşağıdakilerden hangisidir?
A) Pcm B) Ppm C) Rpm
D) Pwm E) Rst

14. Servo motorun 0^0 den 30^0 ye dönmesi için uygulanacak pwm sinyalinin uzunluğu aşağıdakilerden hangisidir?
A) 2150 ms B) 830 µs C) 330 ms
D) 2150 µs E) 830 ms

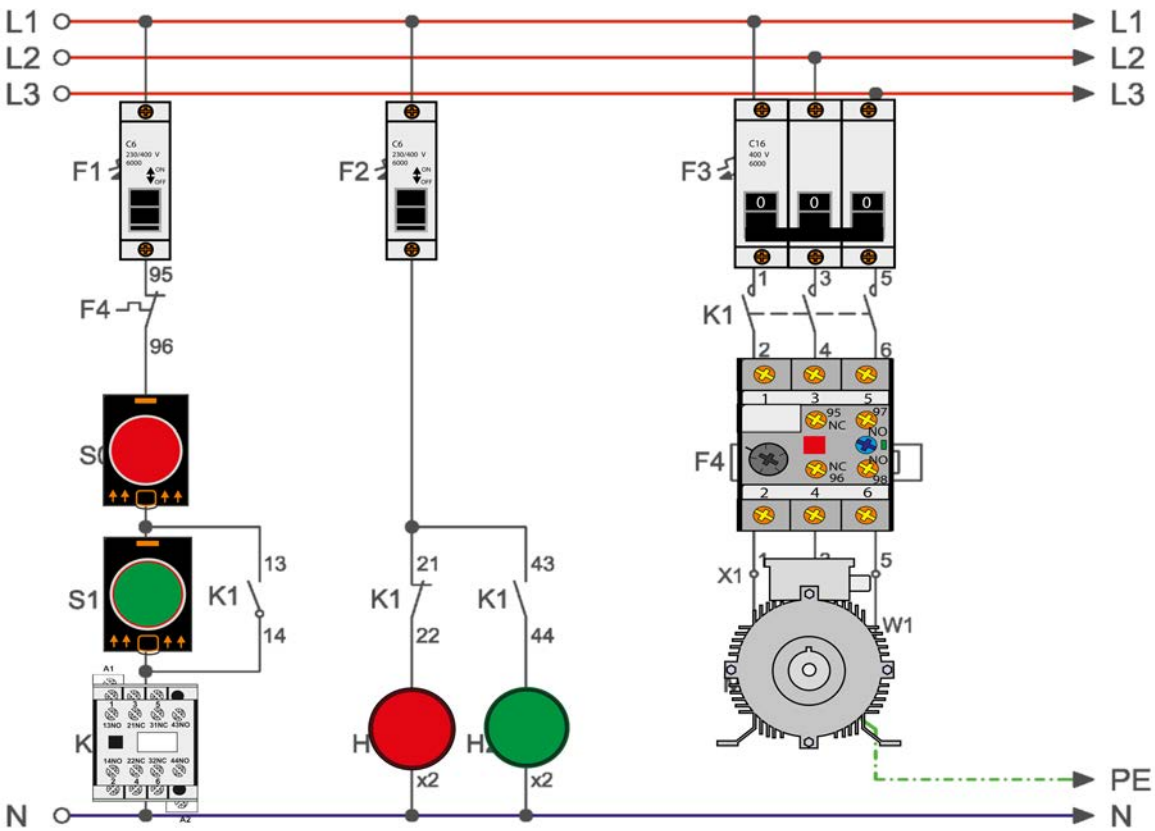
15. Bir palsin lojik 0 veya lojik 1 süresini ölçen fonksiyon aşağıdakilerden hangisidir?
A) analogRead B) analogReference C) digitalRead
D) pinMode E) pulseIn

5. ÖĞRENME BİRİMİ

TEMEL KUMANDA VE GÜÇ DEVRELERİ

KONULAR

5.1. KUMANDA VE GÜÇ DEVRE ELEMANLARI
5.2. KUMANDA VE GÜÇ DEVRE UYGULAMALARI



TEMEL KAVRAMLAR

Asenkron motor, buton, kablo, kontaktör, koruma röleleri, paket şalter, röle, sınır anahtarı, sigorta, zaman rölesi

HAZIRLIK ÇALIŞMALARI

1. Kumanda ve güç devre elemanları ile ilgili bildiklerinizi arkadaşlarınızla paylaşınız.
2. Üç fazlı asenkron motorun kumandasıyla ilgili düşüncelerinizi ifade ediniz.

5.1. KUMANDA VE GÜÇ DEVRE ELEMANLARI

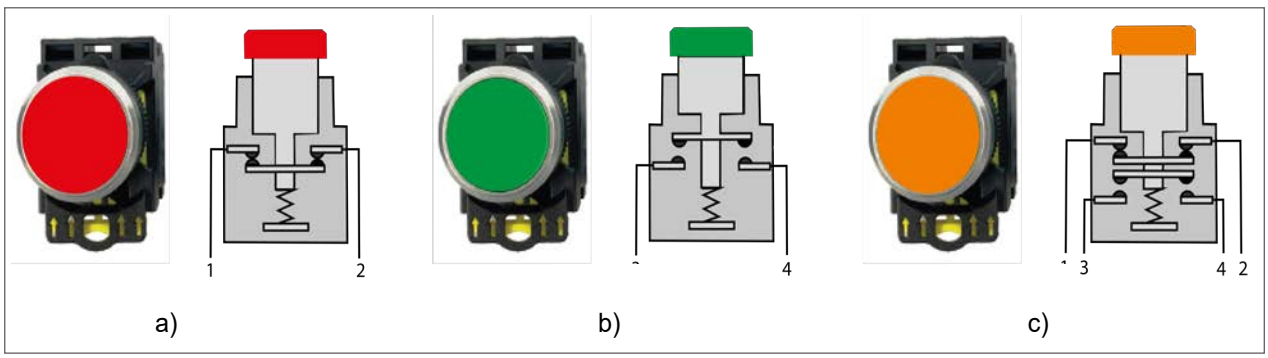
Elektrik motorlarını istenilen şekilde çalıştırmak amacıyla devre elemanları kullanılarak yapılan kontrol işlemlerine **kumanda** denir. Bir dizi kumanda işleminin art arda uygulanmasına da **otomatik kumanda** denir. Motorların kumandası kumanda devreleriyle gerçekleştirilir. Kumanda devrelerinde ana elemanlar motor ve motoru kontrol etmeye yarayan kumanda elemanlarıdır.

5.1.1. Kumanda Butonları

Kumanda devrelerini başlatmak veya durdurmak için kullanılan devre elemanlarına **kumanda butonu** denir. Ani temaslı, kalıcı tip ve acil yardım butonu olmak üzere üç çeşit buton vardır.

Ani Temaslı Butonlar

Yapısında yay bulunan butonlardır. Bu sebeple yay geri dönüşlü buton olarak da isimlendirilir. Butona basıldığında kontakları konum değiştirir ve baskı kaldırıldığında da eski konumuna geri döner. Start, stop ve jog butonu olmak üzere üç çeşittir (Görsel 5.1).



Görsel 5.1: Ani temaslı butonlar

Stop (Durdurma) Butonu: Kumanda devresinin çalışmasını durduran butondur. Normalde kapalı kontağa sahiptir. Butona basıldığı sürece akım geçişine izin vermez (Görsel 5.1:a).

Start (Başlatma) Butonu: Kumanda devresinin çalışmasını başlatan butondur. Normalde açık kontağa sahiptir. Butona basıldığı sürece akım geçişine izin verir (Görsel 5.1:b).

Jog (Çift Yollu) Buton: Start ve stop butonlarının birleştirilmesiyle oluşan butondur. Hem normalde açık hem de normalde kapalı kontağa sahiptir. Butona basıldığında, kapalı kontağı açılıp açık kontağı kapanır (Görsel5.1:c).

Kalıcı Tip Butonlar

Butona basıldığında, kontakları konum değiştiren, baskı kalktığında ise kontaklarının konumunu koruduğu butonlardır. Kontakların konumunun değişmesi için butona tekrar basılması gerekir. Kalıcı tip butonların basmalı, çevirmeli, ışıklı ve anahtarlı tipte olanları mevcuttur.

Acil Durdurma Butonu

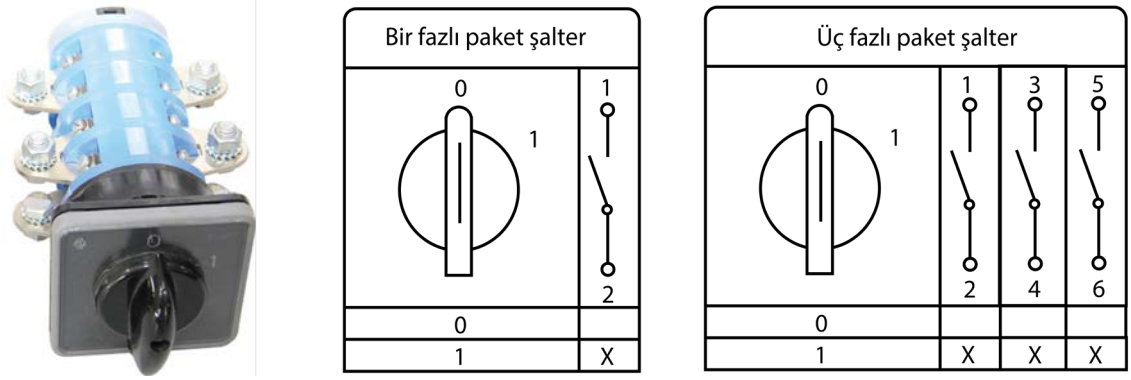
Butona basıldığında, bağlı bulunduğu devrenin enerjisini keserek çalışmasını durduran ve tüm potansiyel tehlikeleri engelleyen butondur. Kumanda devresinde tehlike oluşması hâlinde sistemi kolayca durdurabilecek şekilde tasarlanmıştır.

5.1.2. Paket Şalterler

Bir eksen etrafında dönen mil üzerine dizilip paketlenmiş kontaklardan oluşan çok konumlu şaltere **paket (pako) şalter** denir. Şalterin her bir diliminde iki, üç ya da dört kontak bulunur. Kontakların açılıp kapanması dilimler üzerindeki çıkıntılar sayesinde olur. Şalter kolunun çevrilmesi ile bakalit dilimler döner ve üzerindeki kontaklar konum değiştirir.

Paket şalterler, küçük güçlü alıcıların kumandasında (açma-kapama, devir yönü değiştirme ve yol verme) kullanılır. Üzerlerinde koruma fonksiyonu yoktur. Termik ve manyetik akımlara karşı koruma yapmaz.

Paket şalter şemaları ve çalışma diyagramları kontaklarının konumları hakkında bilgi verir (Görsel 5.2).



a) b) c)

Görsel 5.2: Paket şalter ve paket şalter diyagramları

5.1.3. Sınır Anahtarları

Hareketli aygıtlarda bir hareketi durdurup başka bir hareketi başlatan ve aygıtın hareket eden parçası tarafından kumanda edilen elemanlara **sınır anahtarı** denir (Görsel 5.3:a). Sınır anahtarının normalde açık ve normalde kapalı iki kontağı mevcuttur. Bant sistemlerinde, kapı sistemlerinde, takım tezgâhları gibi hareketli sistemlerde kullanılır. Makaralı ve pimli olmak üzere iki çeşittir.

5.1.4. Sinyal Lambaları

Bir kumanda devresinde devrenin çalışma durumunu gösteren ışıklı bildirim elemanına **sinyal lambası** denir (Görsel 5.3:b). Sinyal lambaları 12, 24, 220 V gibi çeşitli gerilim değerlerinde ve AC/DC çalışmaya uygun farklı gerilim tiplerinde üretilir.



a) b)

Görsel 5.3: Sınır anahtarı ve sinyal lambaları

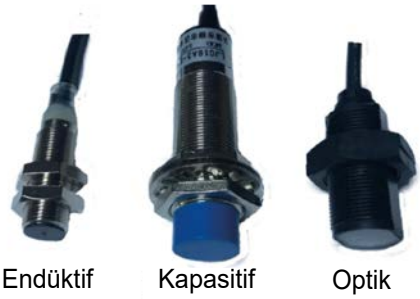
5.1.5. Sensörler

Isı, ışık, hareket, basınç gibi fiziksel büyüklükleri algılayıp elektriksel işarete dönüştüren devre elemanlarına **sensör** denir. Kumanda devrelerinde temassız algılama ve kontrol için kullanılır. Kumanda gerilimleri 10-30 VDC veya 220 V AC/DC'dir. Uygulamada genellikle 24 VDC tercih edilir.

En çok kullanılan sensörler endüktif, kapasitif ve optik sensörlerdir.

Endüktif Sensörler: Metal cisimleri algılayan sensörlerdir. Endüktif sensörler bir bobin yardımıyla sürekli bir manyetik alan oluşturur. Bu manyetik alana metal bir cisim yaklaştığında oluşan manyetik etkileşim nedeniyle endüktif sensörün çıkış sinyali değişir. Bu sensörler genellikle silindirik yapıda olup çapları **M5, M8, M12, M18 ve M30**'dur (Görsel 5.4). Sensör büyüdükçe algılama mesafesi de artar. **1 ile 50 mm** arası algılama yapabilir.

Kapasitif Sensörler: Metal olan ve olmayan cisimleri algılayan sensördür. Yapısı endüktif sensöre benzer. Genellikle silindirik yapıda olup büyüklükleri **8-30 mm** arasındadır (Görsel 5.4).



Görsel 5.4: Endüktif, kapasitif ve optik sensör

Sensör büyüdükçe algılama mesafesi de artar. **2 ile 40 mm arası** algılama yapabilir.

Optik Sensörler: Ortamdaki ışık miktarının değişimini algılayarak elde edilen verileri elektriksel sinyale çeviren sensörlerdir. Optik sensörlerin algılama mesafesi, algılanacak cismin boyutları ile orantılı olarak değişir. Algılama sistemine göre 0,3 ile 10 m arasında algılama yapabilir. Genellikle ölçüm ve analiz, temassız algılama, parçaların sayılması veya konumlandırılması için kullanılır (Görsel 5.4).

5.1.6. Sigortalar

Elektrik hattını ve hatta bağlı cihazları aşırı akımın etkilerine karşı koruyan devre elemanına **sigorta** denir. Sigorta, devreye daima seri bağlanır ve hattın güvenliğini sağlar. Standart akım değerlerinde üretilen sigortalar bu akım değerleri aşıldığında hattın enerjisini keserek koruma sağlar. Sigorta seçiminde bu akım değerlerine dikkat edilmelidir.



Görsel 5.5: Otomatik sigortalar

Otomatik Sigortalar

Kumanda devrelerinde en çok kullanılan sigorta çeşididir. **0,5 A–1 A–1,6 A–2 A–3 A–4 A–6 A–10 A–16 A–20 A–25 A–32 A–40 A–50 A–63 A–80 A–100 A–125 A** gibi standart akım değerlerinde üretilir. Otomatik sigortalara **W Otomat** da denir (Görsel 5.5).

Sigortalar B ve C tipi olmak üzere iki çeşittir.

B Tipi Sigortalar: Aşırı akımda hemen atar. Aydınlatma ve priz tesislerinde kullanılır.

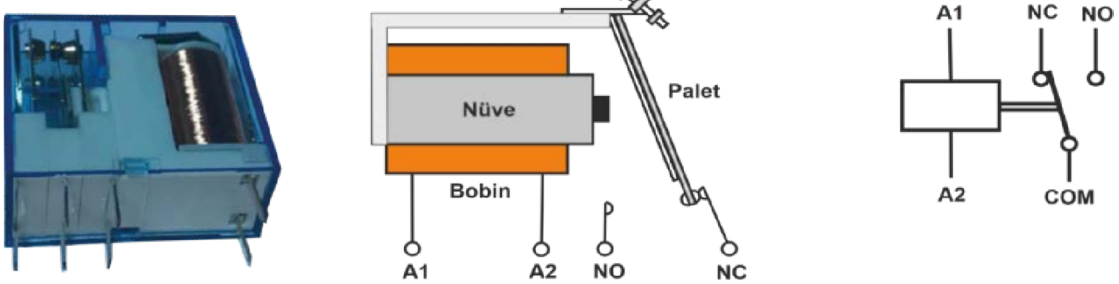
C Tipi Sigortalar: Aşırı akımda gecikmeli atar. Motor koruma devrelerinde kullanılır. Gecikmeli çalışması sayesinde motorların ilk kalkınma anında devreye girmez.

NH (Bıçaklı) Sigortalar

Sanayi tesislerindeki yüksek akımlı alıcıların korunmasında kullanılan sigortalardır. Altlık (gövde) ve buşon olmak üzere iki parçadan oluşur. Akım taşıma kapasitesine göre içinde kullanılan bakır telin erimesiyle bağlı bulunduğu devrenin akımını keser. Dış yüzeyi porselen malzemedir.

5.1.7. Röleler

Küçük güçlü elektromanyetik anahtarlara **röle** denir. Devrelerde küçük bir akımla büyük güçlü alıcıları kontrol etmede kullanılır. Temel olarak bobin ve buna bağlı çalışan kontaklardan oluşur. Bobin uçları **A1** ve **A2** harfleri ile gösterilir. Kontaklar **NC** (normalde kapalı) ve **NO** (normalde açık) olmak üzere iki tiptir (Görsel 5.6).



Görsel 5.6: Röle yapısı ve sembolü

Rölenin bobin gerilimi 5-48 V aralığındadır. Bobini, doğru ya da alternatif gerilimle çalışan tipleri mevcuttur. Röleye enerji uygulandığında bobin elektromıknatıs hâline gelir ve palete bağlı kontakları konum değiştirir. Akım kesilince elektromıknatıslık ortadan kalkar ve esnek gergi yayı, paleti geri çekerek kontakları ilk konumuna getirir. Kumanda devrelerinde özellikle zamanlama ve koruma uygulamalarında kullanılan pek çok röle çeşidi vardır.

5.1.7.1. Zaman Röleleri

Kumanda devrelerinin zamana bağlı olarak çalışıp durmalarını sağlayan zaman geciktirici devre elemanına **zaman rölesi** denir (Görsel 5.7). Zaman röleleri çoğunlukla zaman ayarını sağlayan elektronik devreden ve gecikme ile konum değiştiren kontaklardan oluşur.



Görsel 5.7: Çeşitli zaman röleleri

Düz Zaman Rölesi (Çekmede Gecikmeli Tip): Besleme uçlarına enerji uygulandıktan sonra gecikme yapan zaman röleleridir. Normalde açık, zaman gecikmeli kapanan ve normalde kapalı, zaman gecikmeli açılan kontak olmak üzere zamana bağlı olarak konum değiştiren iki tip kontağı vardır. Enerjilendikten sonra gecikme yapar ve ayarlanan süre sonunda kontakları konum değiştirir. Enerjisi kesildiğinde gecikmeli çalışan kontakları ani olarak normal konumlarına döner.

Ters Zaman Rölesi (Bırakmada Gecikmeli Tip): Enerjisi kesildikten sonra gecikme yapan zaman röleleridir. Normalde açık, zaman gecikmeli açılan ve normalde kapalı, zaman gecikmeli kapanan olmak üzere iki tip kontağı vardır. Enerji verildiğinde kontaklar ani olarak konum değiştirir. Enerji kesildikten sonra gecikme başlar ve süre sonunda kontaklar tekrar konum değiştirir.

Yıldız-Üçgen Zaman Rölesi: Yıldız-üçgen yol verme işleminde kullanılan zaman röleleridir. Röle içerisinde yıldız ve üçgen kontağı olmak üzere iki çeşit kontak bulunur. Enerjilendiğinde normalde açık olan yıldız kontağı anında kapanarak yıldız kontaktörünü enerjilendirir ve motorun yıldız bağlantılı çalışmasını sağlar. Ayarlanan süre sonunda yıldız kontağı açılır, üçgen kontağı kapanır. Motor, üçgen bağlantılı çalışır. Devrede bağlantı kolaylığı sağlar.

Sağ-Sol Zaman Rölesi: Motoru ileri-geri yönde belirli sürede çalıştırmak amacıyla kullanılan zaman röleleridir. Çalışma mantığı olarak düz zaman rölesi ile benzer özelliktedir.

5.1.7.2. Koruma Röleleri

Aşırı Akım Röleleri: Elektrik motorlarını aşırı akımların zararlarından koruyan rölelerdir. Sigortalar çalışma karakteristikleri nedeniyle elektrik motorlarını koruyamadıkları için koruma röleleri kullanılır. AA röleleri, küçük hata akımlarında gecikmeli olarak, büyük hata akımlarında ise çok kısa bir sürede devreyi açarak koruma sağlar.

Güç devresine seri bağlanan üç adet kontağı (1-2, 3-4, 5-6), kumanda devresine seri bağlanan bir adet kapalı kontağı (95-96) ve genellikle aşırı akım etkisini ışıklı bildirimde kullanmak için bir adet açık kontağı (97-98) vardır. Termik ve manyetik aşırı akım rölesi olmak üzere iki tiptir.

Kaçak Akım Koruma Röleleri: Elektrik devrelerinde oluşan hata akımlarını algılayarak devreyi açan koruma elemanlarıdır. Elektrikli cihazların iletkenlerinin veya enerji altındaki parçalarının izolasyonunun bozulması sonucunda cihaz gövdesinde oluşan istenmeyen akımlar, cihaz gövdesine dokunan kişi için tehlike oluşturur. Bu röleler bu akımları algılar ve enerjiyi keser.

Motor Koruma Röleleri: Üç fazlı asenkron motorlarda L1-L2-L3 fazlarının varlık-yokluk kontrolünü yapan ve fazlardan birinin kesilmesi durumunda devreyi açan koruma elemanlarıdır. Bu röleye **faz koruma rölesi** de denir. Üç fazlı devrelerde fazlardan biri kesilse de motor çalışmaya devam eder. Ancak iki faza kalan motor, şebekeden yüksek akım çekerek kısa süre içinde yanar. Bu durumu önlemek için diğer koruma elemanlarıyla birlikte bu röleler de kullanılır (Görsel 5.8).



Görsel 5.8: Çeşitli koruma röleleri

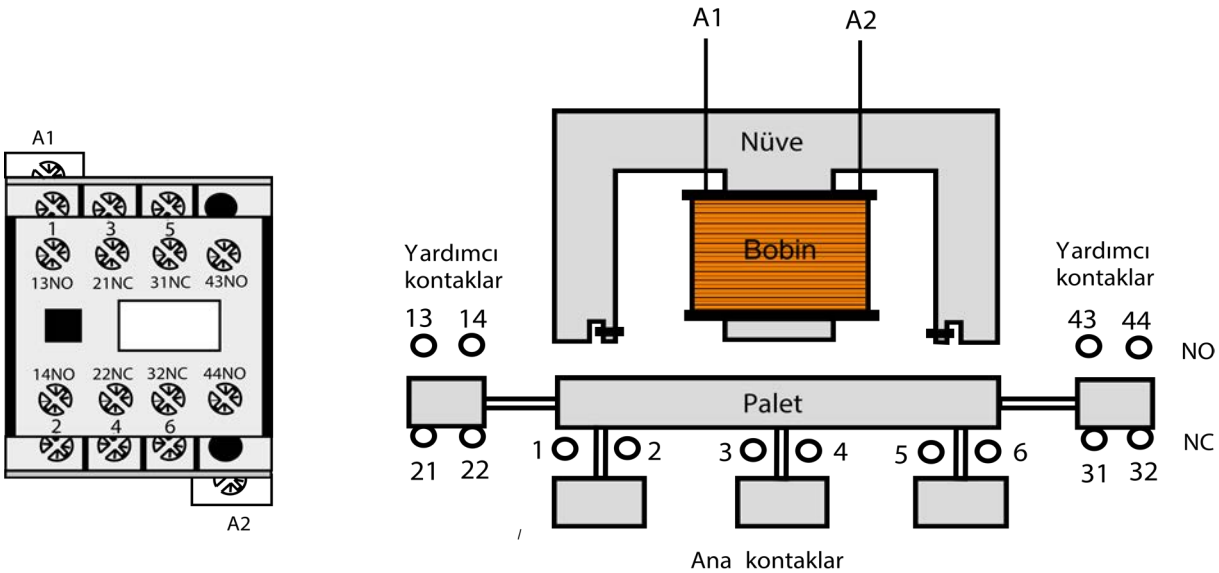
Faz Sırası Röleleri: Üç fazlı asenkron motorlarda L1-L2-L3 fazlarının sıra kontrolünü yapan ve faz yerlerinin değişmesi hâlinde devreyi açan koruma röleleridir. Faz sıralarının değişimi üç fazlı asenkron motorların dönüş yönünü de değiştirmektedir. Motor dönüş yönünün istem dışı olarak değişmesinin istenmediği yerlerde (asansör, kompresör vb.) kullanılır.

Gerilim Koruma Röleleri: Motorları ve sistemleri aşırı veya düşük gerilime karşı koruyan koruma elemanlarıdır. Alt ve üst gerilim sınırları belirlenerek gerilim dalgalanmalarını önler.

Frekans Koruma Röleleri: Şebeke frekansının değişimlerine karşı koruma sağlayan rölelerdir. Alt ve üst limit sınırlarının dışına çıkılması hâlinde motor devreden çıkarılır.

5.1.8. Kontaktörler

Büyük güçlü elektromanyetik anahtarlara **kontaktör** denir. Kontaktörler; elektromıknatıs, palet ve kontaklar olmak üzere üç kısımdan oluşur (Görsel 5.9).



Görsel 5.9: Kontaktör ve yapısı

Kontaktör bobini enerjilendiğinde, bobin üzerinde bir manyetik alan meydana gelerek nüvenin mıknatıslanmasını sağlar. Elektromıknatıs özelliği kazanan sabit nüve, karşısında bulunan hareketli paleti kendine çekerek kontakların konum değiştirmesini sağlar. Elektrik motoru ve aydınlatma gruplarının kumandasında kullanılır.

Kontaktör seçiminde dikkat edilecek hususlar şunlardır:

Anma Gerilimi: Kontaktörün kumanda edeceği gerilim değeridir (230 V veya 400 V).

Bobin Gerilimi: Bobinin çalışma gerilimidir. Bu gerilim AC veya DC olabilmektedir.

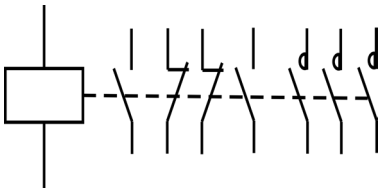
Anma Akımı: Kontaktörün güç kontaklarının akım değeridir.

Anma Gücü: Kumanda edilecek alıcının gücüdür.

Kontak Yapısı ve Sayısı: Kontaktörlerde güç (ana) ve kumanda (yardımcı) kontakları olmak üzere iki tip kontak mevcuttur. Güç kontakları, yüksek akıma dayanıklı olup motor gibi alıcıları çalıştırmak için kullanılır. Kumanda kontakları ise kumanda devre elemanlarının kontrolünde görev yapar. Güç kontakları üç adettir ve tümü normalde açık (NO) kontaklardır. Kumanda kontakları ise üreticiye göre çeşitli kombinasyonlarda ve farklı sayıda NO ve NC kontaklardan oluşur (Tablo 5.1).

Kullanma Sınıfı: Kontaktörler; çalışma gerilimine, işletme ve kullanma şekillerine göre sınıflara ayrılarak standartlaştırılmıştır. Sınıflarına göre farklı yükleri kumanda eder. Kullanma sınıfının doğru tespit edilmesi ve uygun olarak seçim yapılması, kontaktörün sağlıklı çalışması açısından önemlidir. Kontaktörlerin üretim sınıfları ve kullanım alanları Tablo 5.2 ve Tablo 5.3'te verilmiştir.

Tablo 5.1: Kontaktör Güç ve Kumanda Kontakları



A1-A2	Bobin uçları	**13-14**	1.kontak, NO yardımcı kontak
1-2	L1 fazına ait güç kontağı	**21-22**	2.kontak, NC yardımcı kontak
3-4	L2 fazına ait güç kontağı	**31-32**	3.kontak, NC yardımcı kontak
5-6	L3 fazına ait güç kontağı	**43-44**	4.kontak, NO yardımcı kontak

Tablo 5.2: Alternatif Akım Kontaktörlerinin Üretim Sınıfları ve Kullanım Alanları

Sınıfı	Kullanım Alanları
AC-1	Omik yüklerde, özellikle ısıtma uygulamalarında kullanılır.
AC-2	Bilezikli asenkron motorlara yol vermede kullanılır.
AC-3	Sincap kafesli asenkron motorlarda kullanılır. En yaygın uygulama sınıfıdır.
AC-4	Kesik çalışma ve ters akımla frenleme uygulamalarında kullanılır.
AC-5b	Akkor flamanlı lambaların kumandasında kullanılır.
AC-6a	Transformatörlerin kumandasında kullanılır.
AC-7a/b	Az endüktif yüklerde, özellikle ev cihazlarında vb. uygulamalarda kullanılır.

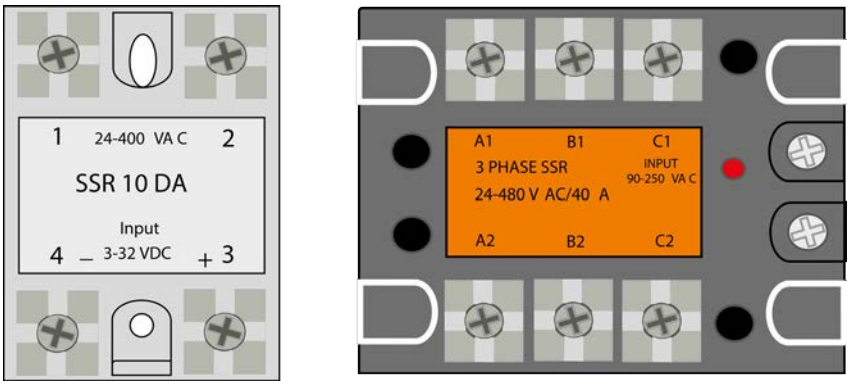
Tablo 5.3: Doğru Akım Kontaktörlerinin Üretim Sınıfları ve Kullanım Alanları

DC-1	Omik DC yüklerin beslenmesinde kullanılır.
DC-3	Şönt motorların çalıştırılması ve frenlenmesiyle ilgili devrelerde kullanılır.
DC-5	Seri motorların çalıştırılması ve frenlenmesiyle ilgili devrelerde kullanılır.
DC-6	Akkor flamanlı lambaların kumandasında kullanılır.

5.1.9. Katı Hal Röleleri

Kumanda ve güç devrelerinde anahtarlama yapabilen elektronik anahtarlara **katı hal rölesi** denir. Bu röleler **solid state röle** veya kısaca **SSR** diye de isimlendirilir. Klasik röle ve kontaktör yerine kullanılabilir. Bu elemanlarda olduğu gibi düşük akımlar kullanılarak yüksek akımlar kontrol edilebilir. Bir ve üç fazlı çeşitleri vardır (Görsel 5.10). Bu rölelerin özellikleri şunlardır:

- Hareketli mekanik parçası olmayıp tamamen elektronik devre elemanlarından oluşur.
- 1-2 A'den 1000-2000 A'e kadar farklı akımlarda anahtarlama yapabilir.
- AC veya DC gerilimle kontrol edilebilir.
- Elektromekanik rölelere göre daha hızlı anahtarlama yapar.



Görsel 5.10: Bir ve üç fazlı katı hal röleleri

5.1.10. Kumanda ve Güç Devrelerinde Kullanılan Kablolar

Kumanda devrelerinde özel bir durum belirtilmemişse NYAF tipi kablolar kullanılır. Bu kablolar çok damarlı olup rahatça bükülebilir ve şekillendirilebilir. Genellikle 1-1,5-2,5 mm^2 kesitli kablolar kullanılsa da kesitler kumanda ettikleri elemanın gücüne göre değişim gösterebilir.

Kumanda ve güç devrelerinde genel olarak kullanılan kablolar şunlardır:

NYA Kablolar: Tek damarlı, tek telli bakır kablolardır. Küçük güçlü devrelerde kullanılır.

NYAF Kablolar: Tek damarlı, çok telli bakır kablolardır. Kumanda devrelerinde genellikle bu kablolar kullanılır.

TTR Kablolar: Çok damarlı, çok telli bakır ve PVC dış kılıflı kablolardır. Mekanik zorlamanın az olduğu uygulamalarda kullanılır. TTR kablolar; 2, 3, 4 veya 5 damarlı olabilir.

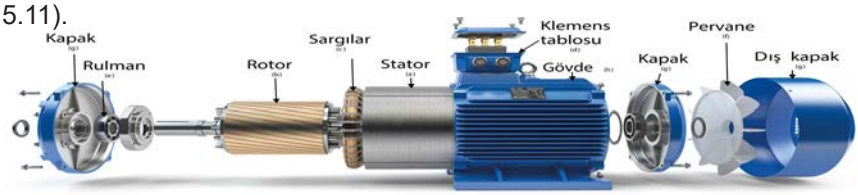
Halojen Free Kablolar: Halojen içermeyen kablolardır. PVC kabloların yangın esnasında çıkarmış olduğu yoğun duman ve zehirli gazları çıkarmaz. Alev geciktirici özelliğe sahiptir. Bu özellikleri sayesinde yangın sırasında alev alsa bile kendiliğinden söner ve yangını büyütmez. Genellikle kapalı alanlarda tercih edilir.

5.1.11. Asenkron Motorlar

Elektrik enerjisini mekanik enerjiye çeviren elektrik makinelerine **elektrik motoru** denir. Stator döner alan devri ile rotor devri farklı olan motorlara da **asenkron motor** denir. Endüstriyel fanlar, havalandırma sistemleri, kompresörler, konveyör sistemleri, vinçler asenkron motorların kullanım alanlarından bazılarıdır.

5.1.11.1. Asenkron Motorların Yapısı

Asenkron motorlar; stator, sargılar, rotor, rulman, pervane, kapak, klemens tablosu ve gövdeden oluşur (Görsel 5.11).



Görsel 5.11: Asenkron motorun yapısı

Stator: Asenkron motorun duran kısmıdır. 0,35-0,8 mm'lik silisyum katkılı, birer tarafı yalıtılmış ve iç yüzeyine oluklar açılmış sacların preslenerek paketlenmesiyle elde edilir. Motor sargılarını üzerinde taşır (Görsel 5.11:a).

Rotor: Asenkron motorun dönen kısmıdır. Sincap kafesli ve sargılı (bilezikli) olmak üzere iki çeşittir. Sincap kafesli rotor, sac paketi ve bunu çevreleyen kısa devre çubuklarından oluşur (Görsel 5.11:b). Bilezikli rotor üzerinde ise statorda olduğu gibi sargılar bulunur.

Sargılar: Emaye izoleli bakır iletkenlerin motor özelliğine göre bir araya getirilerek statora yerleştirilmesiyle oluşturulan kablo grubudur (Görsel 5.11:c). Bilezikli asenkron motorlarda rotora da sargı yerleştirilir.

Klemens Tablosu: Statora yerleştirilen sargı uçlarının bağlandığı tablodur. Stator sargıları bu tabloya çıkarılarak motorun özelliğine göre uygun şekilde bağlantısı yapılır (Görsel 5.11:d).

Rulmanlar: Rotorun kolayca dönmesini sağlayan mekanik yapılı parçalardır (Görsel 5.11:e).

Soğutma Pervanesi: Motorun dönen miline bağlanan plastik ya da metal pervanedir (Görsel 5.11:f). 20 kW güce kadar motorlar soğutma pervanesi yardımıyla soğutulur.

Kapaklar: Dış etkilere karşı gövde ile birlikte motoru koruyan parçalardır. Aynı zamanda rotorun stator içinde merkezî olarak yataklanması görevini de yapar (Görsel 5.11:g).

Gövde: Motor elemanlarını üzerinde taşıyan ve dış etkilere karşı motoru koruyan kısımdır (Görsel 5.11:h). Alüminyum, demir ya da demir alaşımından üretilir.

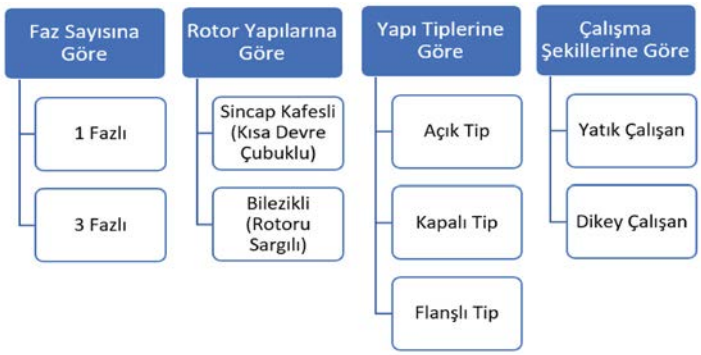
5.1.11.2. Asenkron Motorların Çalışma Prensibi

Asenkron motorlar, indüksiyon prensibine göre çalışır. Motora alternatif gerilim uygulandığında **içinden akım geçen bir iletkenin etrafında manyetik alan oluşur** prensibine göre stator sargıları etrafında değişken bir manyetik alan oluşur. Bu manyetik alana **stator manyetik alanı** denir. Bu alan sargıların yerleşim şeklinden dolayı döner manyetik alandır.

Döner alan kuvvet çizgileri rotoru keserek rotor üzerinde gerilim indükler. Rotorun içinden bir kısa devre akımı geçer. Bu akım rotor etrafında ikinci bir manyetik alan meydana getirir. Buna **rotor manyetik alanı** denir. Stator döner alan kutupları ile rotor manyetik alan kutupları birbirini etkiler. **Aynı kutuplar birbirini iter, zıt kutuplar birbirini çeker** prensibine göre iki alan kutupları arasındaki etkileşim sonucu rotor döner.

5.1.11.3. Asenkron Motor Çeşitleri

Asenkron motorlar genel anlamda bir ve üç fazlı olmak üzere iki çeşittir. Ancak yapı ve çalışma şekillerine göre de farklı çeşitleri bulunur (Görsel 5.12).



Görsel 5.12: Asenkron motor çeşitleri

Faz Sayısına Göre Asenkron Motorlar

Bir Fazlı: Bir fazlı şebeke gerilimiyle (230 V) çalışan motorlardır. Çamaşır makinesi, buzdolabı gibi ev aletlerinde kullanılır. Güçleri genellikle 1,5 kW'a kadardır. Çeşitli özelliklerde bir fazlı motorlar olmakla birlikte en çok kullanılan türü yardımcı sargılı motorlardır. Yardımcı sargılı motorların statorunda **ana sargı** ve **yardımcı sargı** olmak üzere iki ayrı sargı bulunur. Ana sargı kalın telden az sipirlidir ve uçları U1-U2 (U-V) harfleri ile gösterilir. Yardımcı sargısı ise ince telden çok sipirlidir ve sargı uçları Z1-Z2 (W-Z) harfleri ile gösterilir.

Üç Fazlı: Üç fazlı şebeke gerilimiyle çalışan motorlardır. En çok kullanılan motor çeşididir.

Rotor Yapılarına Göre Asenkron Motorlar

Sincap Kafesli (Kısa Devre Çubuklu): Rotorunda sargı bulunmayan asenkron motorlardır. Rotor, silisyumlu sacların paketlenmesinden sonra açılan kanallara alüminyum veya bakır çubuklar yerleştirilmesiyle oluşturulur. Bu çubuklar sargı görevi görür.

Bilezikli (Rotoru Sargılı): Rotorunda sargı bulunan asenkron motorlardır. Rotor üzerinde açılan oluklara alternatif akım sargıları yerleştirilir.

Yapı Tiplerine Göre Asenkron Motorlar

Açık Tip: Kapaklarında ve gövdesinde açıklıklar bulunan motor tipidir. Koruma bakımından zayıf olan motorlardır.

Kapalı Tip: Tamamen kapalı olan motorlardır. Koruma bakımından en uygun yapı tipidir. Bu sebeple çok kullanılır.

Flanşlı Tip: Motor milinde dairesel biçimde metal kapak bulunan motorlardır. Bu kapağa **flanş** denir. Aynı kapak, motorun bağlanacağı sistemde de bulunur. Motor, bu kapaklar birbirine monte edilerek mekanik enerjiyi aktaracağı sisteme bağlanmış olur.

Çalışma Şekillerine Göre Asenkron Motorlar

Yatık Çalışan: Motor milinin yatay olarak durduğu çalışma şeklidir.

Dikey Çalışan: Motor milinin dikey olarak durduğu çalışma şeklidir.

5.1.11.4. Üç Fazlı Asenkron Motorların Avantaj ve Dezavantajları

Avantajları: Yük altında devir sayıları değişmez. Momentleri yüksektir. Frekans invertörleri ile devir sayıları değiştirilebilir. Az arıza yapar ve az bakım gerektirir. Yaygın kullanım alanına sahiptir.

Dezavantajları: Kalkınma anında normal çalışma akımının 3-6 katı fazla akım çeker. Devir sayılarını değiştirmek için ilave donanıma ihtiyaç duyar ve gürültülü çalışır.

1. UYGULAMA

KUMANDA DEVRE ELEMANLARININ KONTAK TESPİTİ

AMAÇ : Ölçü aletiyle buton, kontaktör, aşırı akım rölesi ve zaman rölesinin açık kapalı kontaklarını tespit etmek.

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Buton	Start, stop ve jog butonu	3 adet
Kontaktör	-	1 adet
Aşırı akım rölesi	-	1 adet
Zaman rölesi	Ortak uçlu, düz veya ters zaman rölesi	1 adet
Multimetre	Dijital	1 adet

İŞLEM BASAMAKLARI

1. Multimetreyi diyot veya buzzer (bazır) kademesine alınız.
2. Start butonunun kontak uçlarına ölçü aletinin problarını dokundurarak sonucu gözlemleyiniz.
3. Probları kaldırmadan start butonuna basınız ve sonucu gözlemleyiniz.
4. Aynı işlemi stop butonu için gerçekleştiriniz ve sonucu gözlemleyiniz.
5. Jog butonunun herhangi iki kontak ucuna ölçü aletinin problarını dokundurarak sonucu gözlemleyiniz. Probları kaldırmadan butona basınız ve sonucu gözlemleyiniz.
6. Kontaktörün A1-A2 uçlarına ölçü aletinin problarını dokundurunuz ve sonucu gözlemleyiniz.
7. Ölçü aleti problarını kontaktörün ana kontaklarına dokundurunuz. Kontaktör pimine basmadan ve bastıktan sonra sonucu gözlemleyiniz.
8. Ölçü aleti problarını kontaktörün yardımcı kontaklarına dokundurunuz. NO ve NC kontaklarında kontaktör pimine basmadan ve bastıktan sonra ölçü aletindeki sonucu gözlemleyiniz.
9. Aşırı akım rölesinin 95-96 no.lu kapalı kontaklarına ölçü aletinin problarını dokundurarak sonucu gözlemleyiniz.
10. Aşırı akım rölesinin 97-98 no.lu açık kontaklarına ölçü aletinin problarını dokundurarak sonucu gözlemleyiniz.
11. Zaman rölesinin 1-2 ve 3-4 no.lu kontaklarına sırayla ölçü aletinin problarını bağlayarak sonucu gözlemleyiniz.

SORULAR

1. Normalde açık ve normalde kapalı kontak ne demektir? Açıklayınız.
2. Stop ve start butonlarının kontakları arasındaki farkı açıklayınız.
3. Ölçümünü yaptığınız kontaktörün ana ve yardımcı kontak sayısını belirterek kontakların görevini açıklayınız.
4. Aşırı akım rölesinin kontakları hangi amaçla kullanılır? Belirtiniz.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Ölçü aleti kademe seçimini yaptı.	20	
Sınıfı :	2	Buton kontaklarının tespitini yaptı.	20	
Numarası :	3	Kontaktör kontaklarının tespitini yaptı.	20	
ÖĞRETMEN	4	Aşırı akım rölesi kontaklarının tespitini yaptı.	20	
Adı Soyadı :	5	Zaman rölesi kontaklarının tespitini yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

5.1.12. Kumanda ve Güç Devre Sembolleri

Kumanda ve güç devre şemalarında elemanlar sembollerle gösterilir. Farklı semboller olmakla birlikte dünya genelinde IEC [AYİSİ (International Electrotechnical Commission / Uluslararası Elektroteknik Komisyonu)] ve ANSI [ENSİ (The American National Standards Institute / Amerikan Ulusal Standartlar Enstitüsü)] normlarının kullanımı ağırlıktadır (Görsel 5.13).

SEMBOLÜN ANLAMI	IEC	ANSI	SEMBOLÜN ANLAMI	IEC	ANSI
1 fazlı AC gerilim kaynağı	L N	R Mp	Düz zaman rölesi bobini (çekme gecikmeli)	TR A1 A2	TR
3 fazlı AC gerilim kaynağı	L1 L2 L3 N	R S T Mp	Düz zaman rölesi gecikmeli açılan kontak (TO)	TR 11 12 TO	TR TO
1 fazlı sigorta	F 1 2		Düz zaman rölesi gecikmeli kapanan kontak (TC)	TR 13 14 TC	TR TC
3 fazlı sigorta	F 1 3 5 2 4 6		Ters zaman rölesi bobini (bırakma gecikmeli)	TR A1 A2	TR
1 fazlı otomatik sigorta	F 1 2	e	Ters zaman rölesi gecikmeli kapanan kontak (TC)	TR 11 12 TC	TR TC
3 fazlı otomatik sigorta	F 1 3 5 2 4 6	e	Ters zaman rölesi gecikmeli açılan kontak (TO)	TR 13 14 TO	TR TO
Termik aşırı akım rölesi bobinleri	F 1 3 5 2 4 6	OL	1 fazlı paket şalter	S 1 2	
Termik aşırı akım rölesi (kapalı-açık kontak)	F 95 96 F 97 98	OL OL	3 fazlı paket şalter	S 1 2 3 4 5 6	
Stop (durdurma) butonu	SE 1 2	b	Sınır anahtar kontağı (açık/kapalı)	LS 11 12 LS 11 12	LS LS
Start (başlatma) butonu	SE 3 4	b	Sinyal lambası	H1 x1 x2	
Çift yollu (jog) butonu	SE 1 3 2 4	b	Üç fazlı asenkron motor (yıldız bağlı)	M1 M 3~ U1 V1 W1	U V W
Kontaktör bobini	K A1 A2	M	Üç fazlı asenkron motor (yıldız-üçgen)	M1 M 3~ U1 V1 W1 U2 V2 W2	U V W X Y Z
Kontaktör güç bobini kontakları (açık)	K 1 3 5 2 4 6				
Kontaktör açık kontağı (NO)	K 13 14		Bir fazlı asenkron motor	M1 M 1~ U1 U2 Z1 Z2	M1 M 1~ U V W Z
Kontaktör kapalı kontağı (NC)	K 21 22				

Görsel 5.13: Kumanda ve güç devrelerinde kullanılan semboller (IEC ve ANSI normlarında)

2. UYGULAMA

KUMANDA VE GÜÇ DEVRE ELEMANI SEMBOLLERİNİN ÇİZİMİ

AMAÇ: Kumanda ve güç devresinde kullanılan sembolleri uygun norma ve teknik resim kurallarına göre çizmek.

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Antetli kâğıt	A4	1 adet
Gönyeler	30-60-90 ve 45-45-90 derece	2 adet
Daire şablonu	-	1 adet
Kurşun kalem	B veya 2B	1 adet
Silgi	Yumuşak	1 adet

İŞLEM BASAMAKLARI

1. Görsel 5.13'te verilen kumanda sembollerini inceleyiniz.
2. Antetli A4 kâğıdına tabloyu çiziniz.
3. Teknik resim kurallarına uyarak sembolleri sırasıyla belirlenen alanlara çiziniz.
4. Tablodaki açıklamaları norm yazı ile uygun alanlara yazınız.
5. Çizimleri öğretmeninize teslim ediniz.

SORULAR

1. Devre şemalarında sembol kullanımına niçin ihtiyaç duyulur? Belirtiniz.
2. IEC ve ANSI normlarını karşılaştırarak benzer sembolleri tespit ediniz.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Sembolleri norma uygun çizdi.	20	
Sınıfı :	2	Sembolleri çizim alanına ortaladı.	20	
Numarası :	3	Sembol yazılarını norm yazı ile yazdı.	20	
ÖĞRETMEN	4	Sembol yazılarını hizaladı.	20	
Adı Soyadı :	5	Çizimde temizlik ve düzene dikkat etti.	20	
İmza :		**TOPLAM PUAN**	100	

5.1.13. Kumanda ve Güç Devre Şeması Çizimi

Kumanda tekniklerine ait devre şemaları, kumanda ve güç devre şemalarından oluşur. Motorun kontrolünü sağlayan elemanlar kumanda devresinde bulunur. Motor ve motoru koruma elemanlarının güç kontakları ise güç devresinde bulunur.

5.1.13.1. Kumanda Devrelerinin Çizimi

- İlk olarak çizimin hangi norm ile yapılacağı belirlenir (IEC, ANSI).
- Norma göre çizim yönü belirlenir. Çizim yönü IEC normunda yukarıdan aşağıya doğru, ANSI normunda ise soldan sağa doğrudur.
- Kumanda devre elemanları fazdan nötr hattına doğru belirli sırada yerleştirilir. Yerleştirme yapılırken elemanların hiza ve mesafelerine dikkat edilmelidir.
- Elemanlar arası hatlar çizilir.
- Hatlarda bulunan ek yerleri işaretlenir.
- Her elemanın bobin ve kontakları isimlendirilir (Tablo 5.4). IEC normunda kontaklara numara verilirken ANSI normunda genellikle verilmez.

Tablo 5.4: IEC ve ANSI Normlarına Göre Elemanların İsimlendirilmesi

Elemanlar	IEC	ANSI
Kontaktör	K1, K2 …	M1, M2 …
Yardımcı kontaktör	K1, K2 …	A, B, C ...
Zaman rölesi	TR1,TR2 …	TR1, TR2 …
Aşırı akım rölesi	F1, F2, F3 …	OL1, OL2, OL3
Sigorta	F1, F2, F3 …	e1, e2, e3 …
Buton	S0, S1, S2 …	İsim (START…)
Sinyal lambası	H1, H2 …	İsim (ÇIKIŞ…)
Asenkron motor	M1, M2 …	3~ ASM Motor
Sınır anahtarı	LS1, LS2 …	S1, S2 (SW1, SW2)
Kondansatör	C1, C2 …	c1 (C1), c2 (C2) …
Bobin	L1, L2 …	L1, L2 …
Klemens	X1, X2…	X1, X2 …

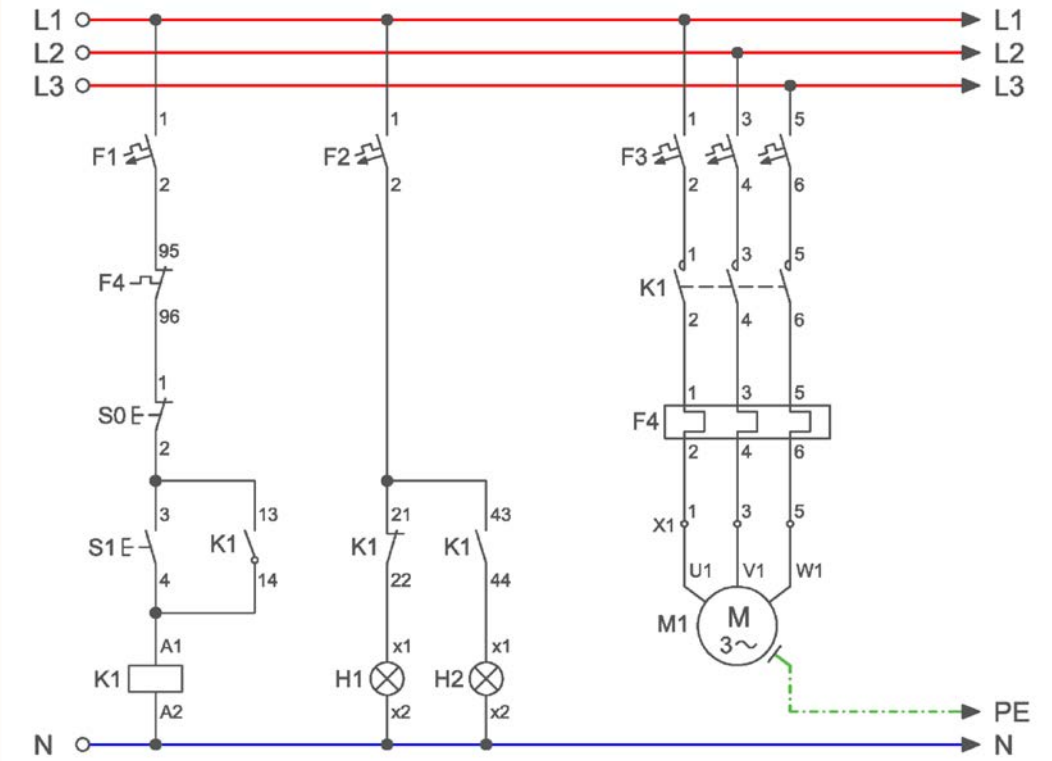
5.1.13.2. Güç Devrelerinin Çizimi

- Güç devresinin çizimi her zaman yukarıdan aşağıya doğrudur.
- IEC normunda enerji hattı ortak, diğer normlarda ayrı olarak gösterilir.
- Güç devresinin girişine üç fazlı sigorta bağlanır.
- Çizilecek her bir eleman arasında eşit aralıklar bırakılmalıdır.
- Kumanda devresi bağlantılarına göre daha kalın hatlar çizilir.
- Hatlar arasında eşit mesafeler bırakılır (Nötr ve toprak hattı da dâhildir.).

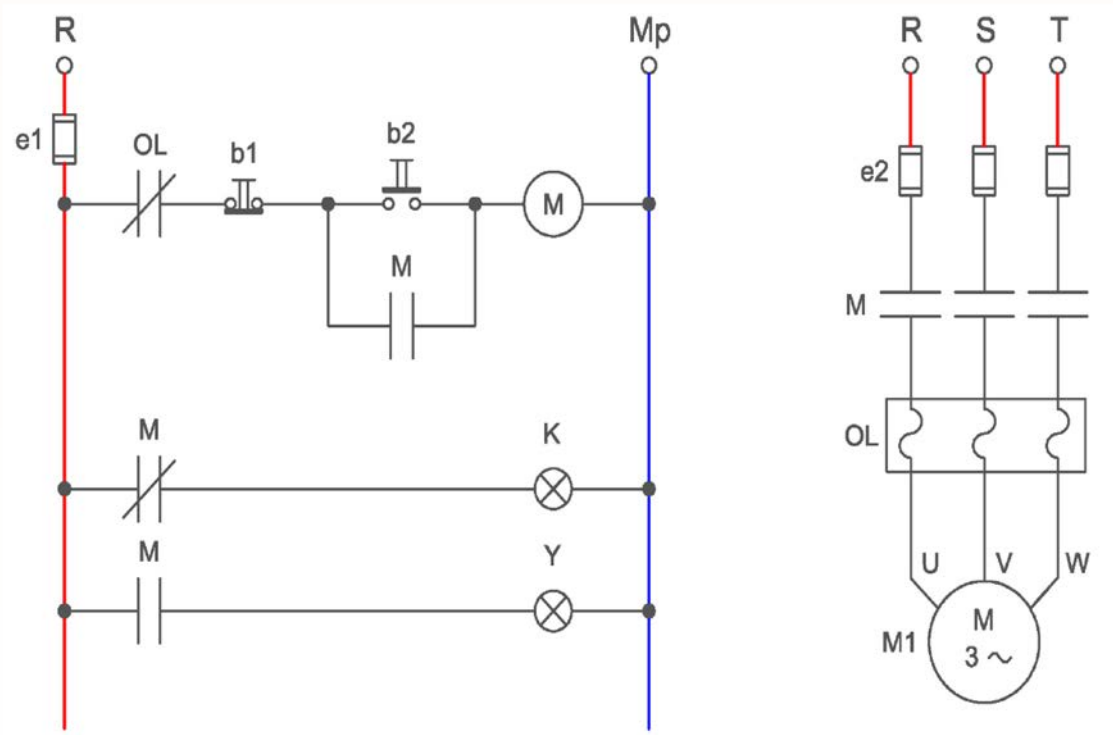
3. UYGULAMA

KUMANDA VE GÜÇ DEVRE ŞEMASI ÇİZİMİ

AMAÇ: Üç fazlı asenkron motorun sürekli çalışmasına ait kumanda ve güç devresini IEC ve ANSI normlarına göre teknik resim kurallarına uygun olarak çizmek.



Görsel 5.14: Kumanda ve güç devresinin IEC normuna göre çizimi



Görsel 5.15: Kumanda ve güç devresinin ANSI normuna göre çizimi

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Antetli kâğıt	A4	2 adet
Gönyeler	30-60-90 ve 45-45-90 derece	2 adet
Daire şablonu	-	1 adet
Kurşun kalem	B veya 2B	1 adet
Silgi	Yumuşak	1 adet

İŞLEM BASAMAKLARI

1. Görsel 5.14'te IEC normuna uygun olarak verilen kumanda ve güç devresini antetli kâğıda teknik resim kurallarına uygun olarak çiziniz ve gerekli isimlendirmeleri yapınız.
2. Görsel 5.15'te ANSI normuna uygun olarak verilen kumanda ve güç devresini antetli kâğıda teknik resim kurallarına uygun olarak çiziniz ve gerekli isimlendirmeleri yapınız.
3. Çizimleri öğretmeninize teslim ediniz.

SORULAR

1. IEC ve ANSI normları arasında ne gibi farklılıklar vardır? Belirtiniz.
2. Çizdiğiniz devrelerde elemanlara verdiğiniz isim listesini çıkararak Tablo 5.5'e yazınız.

Tablo 5.5: Üç Fazlı Asenkron Motorun Etiket Değerleri

Elemanlar	IEC	ANSI
Faz, nötr ve topraklama		
Kumanda devresi sigortaları		
Güç devresi sigortaları		
Aşırı akım rölesi bobin ve kontağı		
Stop butonu		
Start butonu		
Devre kontaktörü ve kontakları		
Sinyal lambası		
Klemens		
Asenkron motor		

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Devreyi IEC normuna göre çizdi.	20	
Sınıfı :	2	Devreyi ANSI normuna göre çizdi.	20	
Numarası :	3	Elemanları isimlendirdi.	20	
ÖĞRETMEN	4	Sembolleri normlara uygun olarak kullandı.	20	
Adı Soyadı :	5	Çizimde temizlik ve düzene dikkat etti.	20	
İmza :		**TOPLAM PUAN**	100	

5.2. KUMANDA VE GÜÇ DEVRE UYGULAMALARI

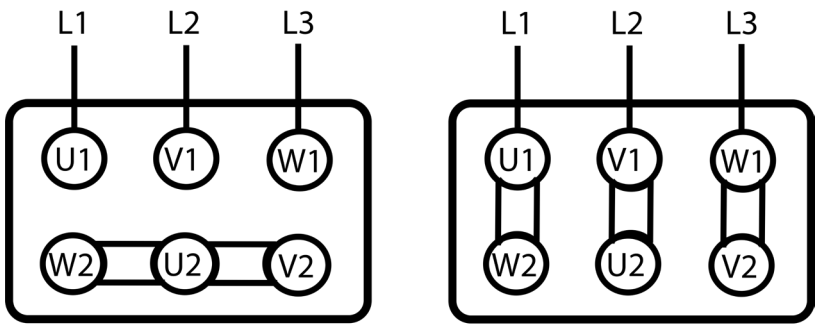
Kumanda devrelerinde alıcı olarak genellikle üç fazlı asenkron motorlar kullanılır. Bu motorlar uygulama alanlarında da en çok kullanılan motorlardır.

5.2.1. Üç Fazlı Asenkron Motor Bağlantıları

Üç fazlı asenkron motorların stator sargı uçları klemens tablosuna altı adet olarak çıkarılır. Bu uçlardan üçü giriş, üçü de çıkış ucudur.

1. **L1 (R) fazı için** sargı giriş ucu **U1 (U)**, çıkış ucu **U2 (X)**
2. **L2 (S) fazı için** sargı giriş ucu **V1 (V)**, çıkış ucu **V2 (Y)**
3. **L3 (T) fazı için** sargı giriş ucu **W1 (W)**, çıkış ucu **W2 (Z)** harfleri ile ifade edilir.

Üç fazlı asenkron motorların sargıları, yıldız bağlantı ve üçgen bağlantı olarak iki şekilde bağlanır. Bağlantılar klemens tablosundaki köprülerle yapılır. Yıldız bağlantıda W2-U2-V2 uçları (Görsel 5.16:a), üçgen bağlantıda ise U1-W2, V1-U2, W1-V2 uçları köprülenir (Görsel 5.16:b).



a) Yıldız bağlantı b) Üçgen bağlantı

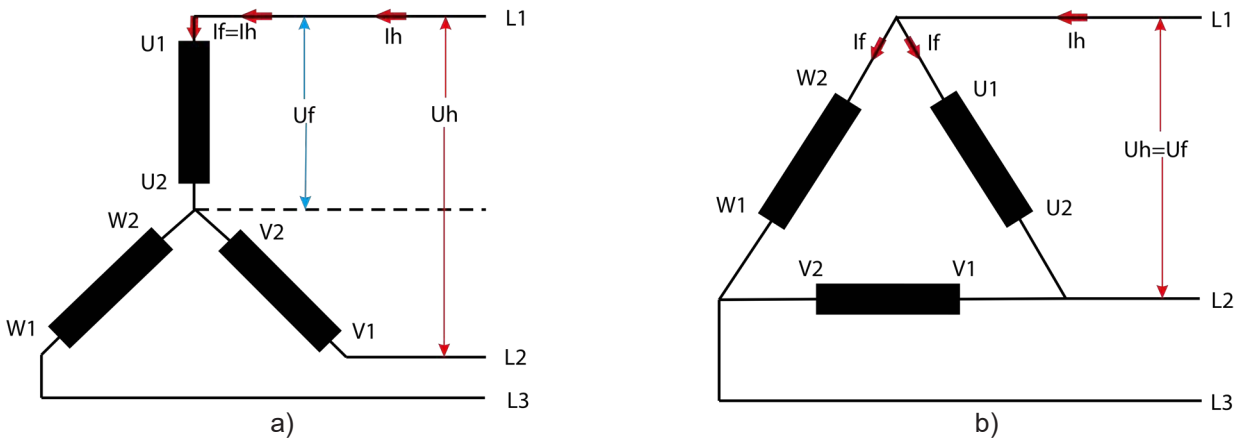
Görsel 5.16: Asenkron motorun klemens tablosuna yıldız ve üçgen köprü bağlantısı

Üç Fazlı Asenkron Motorun Yıldız Bağlantısı

Stator sargı giriş uçlarına üç fazlı gerilim uygulanıp çıkış uçlarının kısa devre edilmesiyle oluşturulan bağlantıya **yıldız bağlantı** denir (Görsel 5.17:a). Yıldız bağlantı λ şeklinde gösterilir. Yıldız bağlantıda hat akımları, faz akımlarına eşittir (Ih=If). Hat gerilimleri ise faz gerilimlerinin √3 (1,73) katıdır (Uh=1,73.Uf).

Üç Fazlı Asenkron Motorun Üçgen Bağlantısı

Stator sargılarından birinci fazın çıkış ucunun ikinci fazın giriş ucuna, ikinci fazın çıkış ucunun üçüncü fazın giriş ucuna ve üçüncü fazın çıkış ucunun birinci fazın giriş ucuna bağlanmasıyla oluşturulan bağlantıya **üçgen bağlantı** denir (Görsel 5.17:b). Üçgen bağlantı △ şeklinde gösterilir. Üçgen bağlantıda hat gerilimleri, faz gerilimlerine eşittir (Uh=Uf). Hat akımları ise faz akımlarının √3 katıdır (Ih=1,73.If).



Görsel 5.17: Üç fazlı asenkron motorun yıldız ve üçgen bağlantısı

BİLGİ NOTU

Yıldız veya üçgen bağlantıya motor etiketine bakılarak karar verilir. Yıldız bağlanması gereken bir motor yanlışlıkla üçgen bağlanırsa sargılar zarar görür.

5.2.2. Üç Fazlı Asenkron Motorların Çalışma Özellikleri

Üç Fazlı Asenkron Motorlara Yol Verme

Asenkron motorların çalışmaya başladıkları ilk anda şebekeden çektiği kalkış akımı yaklaşık olarak normal çalışma akımının **3-6** katı arasındadır. Bu akım kısa sürelidir ve küçük güçlü motorlarda şebeke üzerinde etkili değildir. Büyük güçlü motorlarda ise şebekeye ve motora olumsuz etkileri vardır. Örneğin 100 kW gücündeki bir asenkron motor kalkınırken çalıştığı atölyedeki lambaların aydınlatma şiddeti yaklaşık 3-4 saniye boyunca azalır ve bu durum her çalışmada tekrarlanır. Aynı zamanda motor sargıları ısınır. Bu ısı, makinenin yalıtım malzemelerine zarar verecek kadar artabilir. Bu nedenle gücü **4 kW**'tan (yaklaşık **5 Hp**) büyük motorlara aşağıda verilen yol verme yöntemleri uygulanır.

Direkt Yol Verme: Motorun herhangi bir yöntem kullanılmadan direkt çalıştırılmasıdır. Gücü 4 kW'ın altında olan motorlarda uygulanabilir. En ekonomik ve basit yol verme yöntemidir.

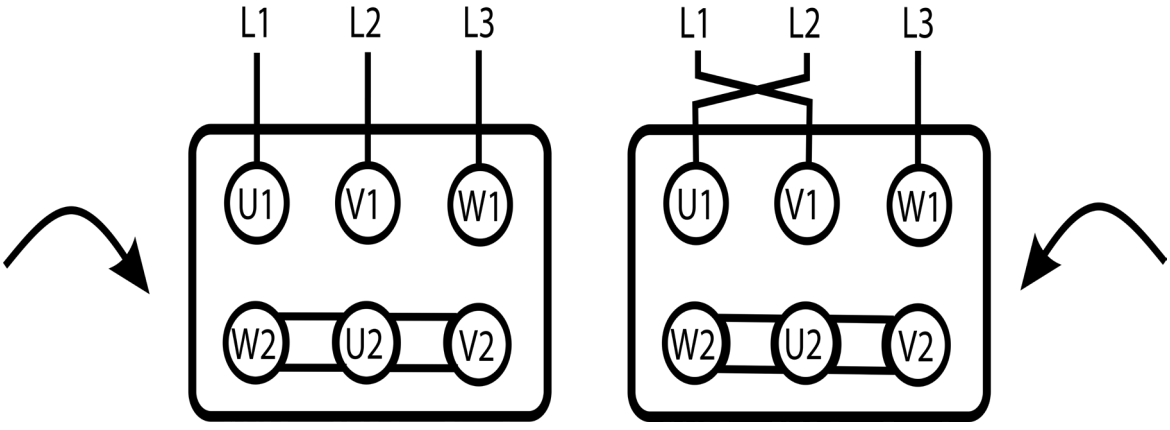
Yıldız-Üçgen Yol Verme: Üç fazlı asenkron motorun önce yıldız sonra üçgen çalıştırılarak yol verilmesidir. Bu yöntemle yol verilebilmesi için motorun üçgen bağlı çalışma gerilimi şebeke gerilimine eşit olmalıdır. Bu yol verme yönteminde, kalkış sırasında yıldız çalışan motor sargılarına uygulanan gerilim **Uh/1,73** değerine, motorun şebekeden çektiği akım ise 1/3 değerine düşer. Kalkınma tamamlandıktan sonra motor üçgen bağlantıya geçerek çalışmasına devam eder.

Yumuşak Yol Verici İle Yol Verme: Yumuşak yol vericiler, asenkron motorlara uygulanan gerilimi kademeli olarak artıran mikro işlemci ve tristör tabanlı cihazlardır. Cihaz, gerilimi sürekli kontrol eder ve başlangıçta yavaşça arttırır, duruşta ise yavaşça azaltır. Böylece yumuşak ve kararlı bir hızlanma elde edilir.

Motor Sürücüsü İle Yol Verme: Motor sürücüleri, şebeke frekansını kontrol ederek motorun hız kontrolünü yapan elektronik cihazlardır. Frekans konvertörü, hız kontrol cihazı, driver gibi farklı isimlerle de anılır. Daha çok hız kontrolü için kullanılır. Yumuşak yol vericilerde olduğu gibi yumuşak bir kalkış ve duruş sağlar.

Üç Fazlı Asenkron Motorun Devir Yönünün Değiştirilmesi

Üç fazlı asenkron motorların çalışma durumuna göre devir yönlerinin değiştirilmesi gerekebilir. Bunun için motor giriş fazlarından (L1-L2-L3) herhangi ikisinin yeri değiştirilir ve biri sabit bırakılır. Bu durum hem yıldız hem de üçgen bağlantı için geçerlidir (Görsel 5.18).



Görsel 5.18: Üç fazlı asenkron motorun devir yönünün değiştirilmesi

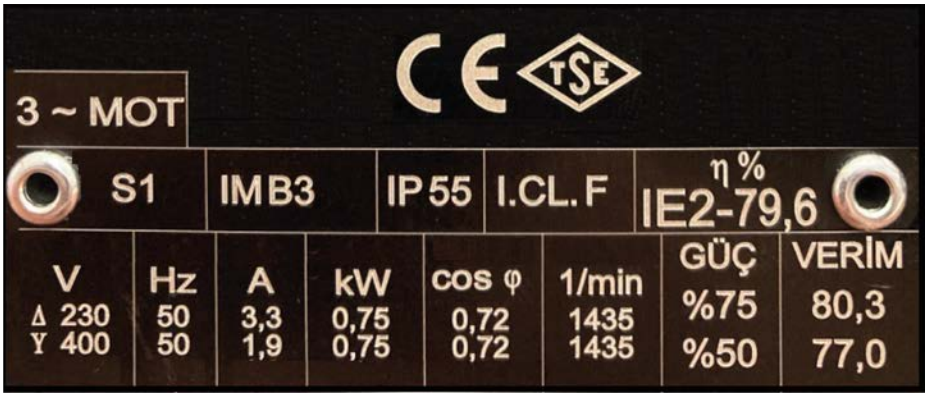
Üç Fazlı Asenkron Motorlarda Kayma ve Devir Ayarı

Stator döner manyetik alan devri ile rotor devri arasındaki devir farkına **kayma** denir. Asenkron motorlar kayma prensibine göre çalışır. Motorun yapısı gereği rotor üzerinde mekanik (sürtünme ve vantilasyon) ve elektriksel kayıplar meydana gelir. Bu sebeple de rotor, stator döner alanından daha az devirle döner. Yani iki devir hiçbir zaman birbirine eşit olmaz.

Üç fazlı asenkron motorların devir sayısı ayarı motor sürücüleri (frekans invertörleri) ile yapılır.

Üç Fazlı Asenkron Motorun Etiket Bilgileri

Motorların yapı ve çalışma özellikleri motor etiketinde belirtilir. Etiket değerleri, tam yük değerleridir (Görsel 5.19).



Görsel 5.19: Üç fazlı asenkron motor etiketi

Tablo 5.6'da üç fazlı asenkron motorun etiket değerleri, motor özellikleri ve açıklamalarıyla gösterilmiştir.

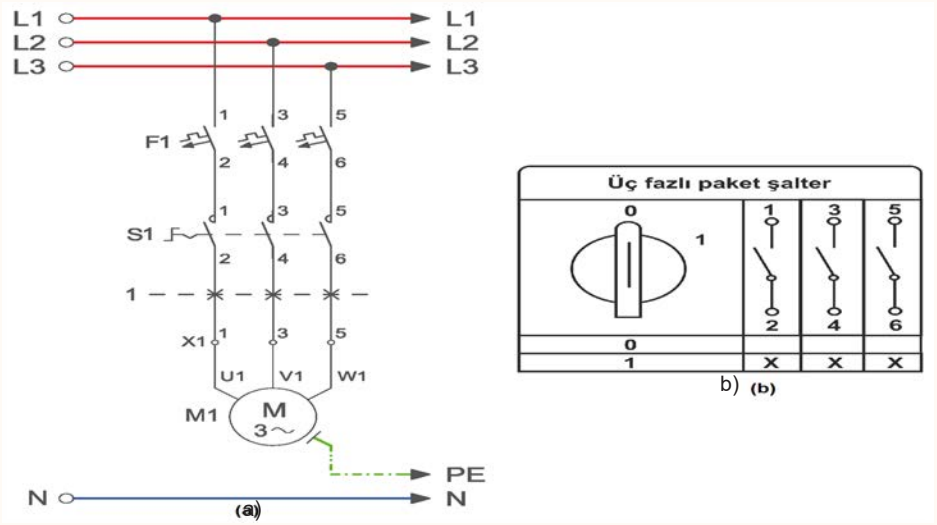
Tablo 5.6: Görsel 5.19'daki Üç Fazlı Asenkron Motorun Etiket Değerleri

Motor Özellikleri	Açıklama
Faz Sayısı	Üç fazlı
Çalışma Sınıfı	S1 tip (sürekli çalışan)
Üretim Tipi	IM B3 (ayaklı motor)
Koruma Sınıfı	IP 55 (toza ve tazyikli suya karşı koruma)
Yalıtım Sınıfı	I.CL.F (Sargılar 155 °C'ye kadar dayanabilir.)
Verimlilik Sınıfı	IE2 (yüksek verimli) - %79,6
Motor Verimi	Üçgen bağlantıda % 80,3; yıldız bağlantıda % 77
Sargı Bağlantısı	230 V şebeke geriliminde üçgen 400 V şebeke geriliminde yıldız
Şebeke Frekansı	50 Hz
Motor Anma Akımı	Üçgen bağlantıda 3,3 A; yıldız bağlantıda 1,9 A
Motor Gücü	0,75 kW
Motor Güç Katsayısı (Cosφ)	0,72
Motor Devir Sayısı	1435 devir/dk.

4. UYGULAMA

ÜÇ FAZLI ASENKRON MOTORA PAKET ŞALTERLE DİREKT YOL VERME

AMAÇ: Üç fazlı asenkron motora aç-kapa paket şalterle direkt yol vermek.



Görsel 5.20: Üç fazlı asenkron motorun aç-kapa paket şalterle çalıştırılması

DEVRENİN ÇALIŞMASI

Devrede asenkron motor sargılarına enerji, paket şalter kontaklarından verilmektedir. Paket şalter **0** konumundayken kontaklar açıktır. Bu durum şalter diyagramında 0 konumunun işaretsiz olmasıyla belirtilir. Motor sargılarına enerji uygulanmadığından asenkron motor çalışmaz.

Şalter 1 konumuna alındığında, motora seri bağlı şalter kontakları kapanarak üç fazın motor sargılarına uygulanmasını sağlar ve motor çalışır. Bu durum, şalter diyagramında 1 konumunun X ile işaretlenmesiyle belirtilir (Görsel 5.20:b).

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Üç fazlı paket şalter	Aç-kapa (0-1)	1 adet
Asenkron motor	Üç fazlı	1 adet
Kumanda kabloları	Jaklı kumanda kablosu veya 1-2,5 mm^2 NYAF	-
Ölçü aleti ve el aletleri	Multimetre, pense, yan keski, kontrol kalemi vb.	-
Silgi	Yumuşak	1 adet

İŞLEM BASAMAKLARI

1. Ölçü aleti ile devre eleman ve kablolarının sağlamlık kontrollerini yapınız.
2. Görsel 5.20'de verilen devre şemasına göre bağlantıları yapınız.
3. Öğretmeninizin kontrolünde devreye enerji veriniz.
4. Üç fazlı aç-kapa paket şalteri 1 konumuna alınız ve motorun çalıştığını gözlemleyiniz.
5. Üç fazlı aç-kapa paket şalteri 0 konumuna alınız ve motorun durduğunu gözlemleyiniz.
6. Enerjiyi kesiniz, malzemeleri sökerek öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

SORULAR

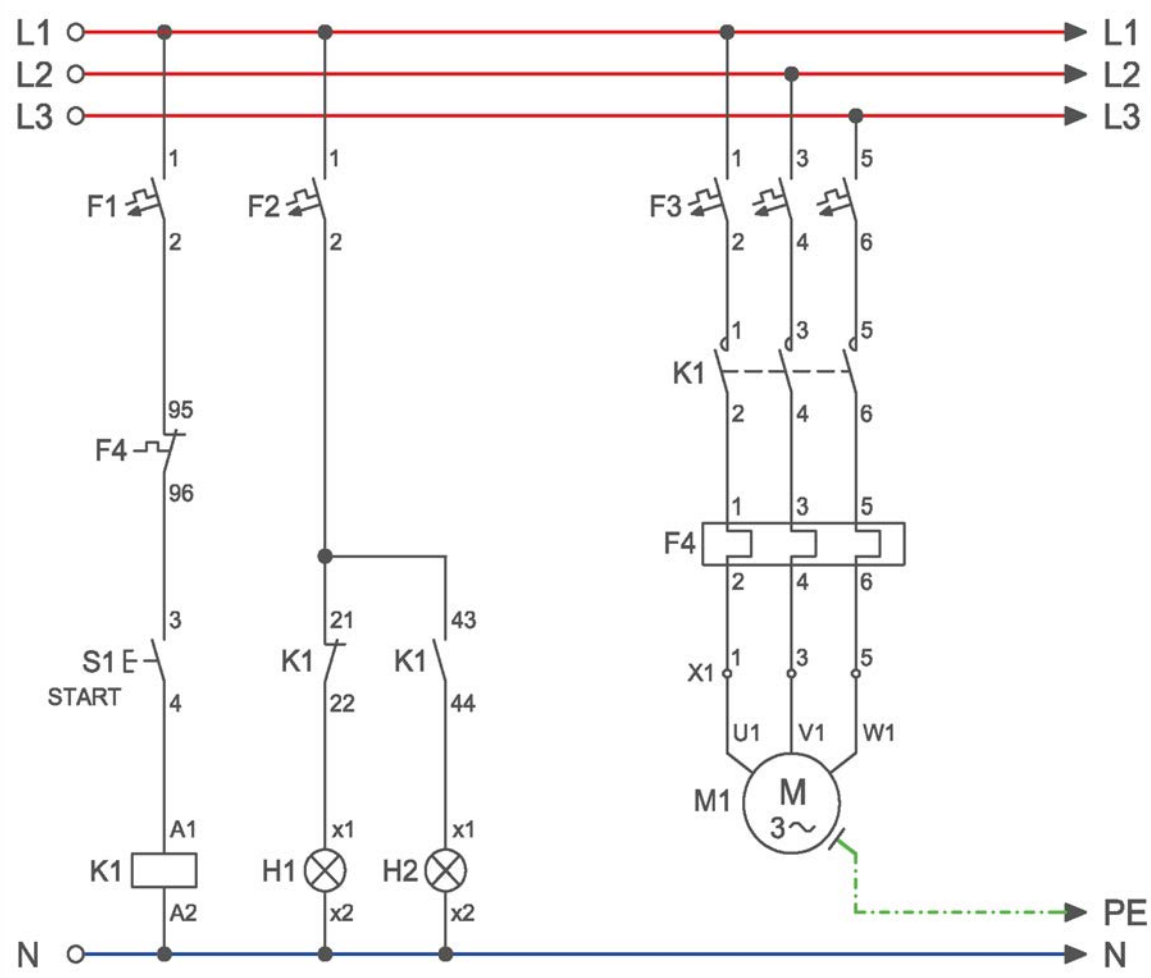
1. Üç fazlı paket şalterin bağlantısı sırasında dikkat edilecek hususlar nelerdir? Açıklayınız.
2. Kullandığınız paket şalter diyagramını çiziniz.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Devre elemanlarının sağlamlık kontrolünü yaptı.	20	
Sınıfı :	2	Devre kablolarının sağlamlık kontrollerini yaptı.	20	
Numarası :	3	Üç fazlı paket şalteri devreye doğru bağladı.	20	
ÖĞRETMEN	4	Asenkron motoru çalıştırdı.	20	
Adı Soyadı :	5	Çalışmada temizlik ve düzene dikkat etti.	20	
İmza :		**TOPLAM PUAN**	100	

5. UYGULAMA

ÜÇ FAZLI ASENKRON MOTORUN BASİT KESİK ÇALIŞTIRILMASI

AMAÇ: **Üç fazlı asenkron motoru bir yönde kesik (aralıklı) çalıştırmak.**



Görsel 5.21: Üç fazlı asenkron motorun bir yönde kesik çalıştırılması devresi

DEVRENİN ÇALIŞMASI

Kumanda devresinde S1 (start) butonuna basıldığında K1 kontaktörü enerjilenir ve kontakları konum değiştirir. K1 kontaktörünün güç devresindeki açık kontaklarının kapanmasıyla motor çalışır. S1 butonuna basıldığı müddetçe motor çalışmaya devam eder. Butondan elin çekilmesiyle kontaktör bobininin enerjisi kesilir ve güç kontakları açılarak motoru durdurur. Motor dururken kırmızı, çalışırken yeşil sinyal lambası yanar.

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Buton	Start ani temaslı	1 adet
Kontaktör	4 kW	1 adet
Sinyal lambası	Kırmızı ve yeşil	2 adet
Aşırı akım rölesi	2,5-4 A	1 adet
Asenkron motor	Üç fazlı	1 adet
Kumanda kabloları	2,5 mm^2 NYAF	-
Ölçü aleti ve el aletleri	Multimetre, pense, yan keski, kontrol kalemi	-

İŞLEM BASAMAKLARI

1. Ölçü aleti ile devre eleman ve kablolarının sağlamlık kontrollerini yapınız.
2. Görsel 5.21'de verilen şemaya göre kumanda devre bağlantılarını yapınız.
3. Öğretmeninizin kontrolünde kumanda devresine enerji veriniz ve kırmızı sinyal lambasının yandığını gözlemleyiniz.
4. S1 butonuna basarak butona basılı olduğu müddetçe kontaktörün kontaklarını çektiğini ve yeşil sinyal lambasının yandığını gözlemleyiniz.
5. Enerjiyi kesiniz.
6. Görsel 5.21'de verilen şemaya göre güç devre bağlantılarını yapınız.
7. Öğretmeninizin kontrolünde devreye enerji veriniz ve kırmızı sinyal lambasının yandığını gözlemleyiniz.
8. S1 butonuna basınız ve motorun çalıştığını, motor çalışırken yeşil sinyal lambasının yandığını gözlemleyiniz.
9. S1 butonundan elinizi çekiniz ve çalışan motorun durduğunu, motor durunca yeşil sinyal lambasının sönerek kırmızı sinyal lambasının yandığını gözlemleyiniz.
10. Enerjiyi kesiniz.
11. Malzemeleri sökerek öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

SORULAR

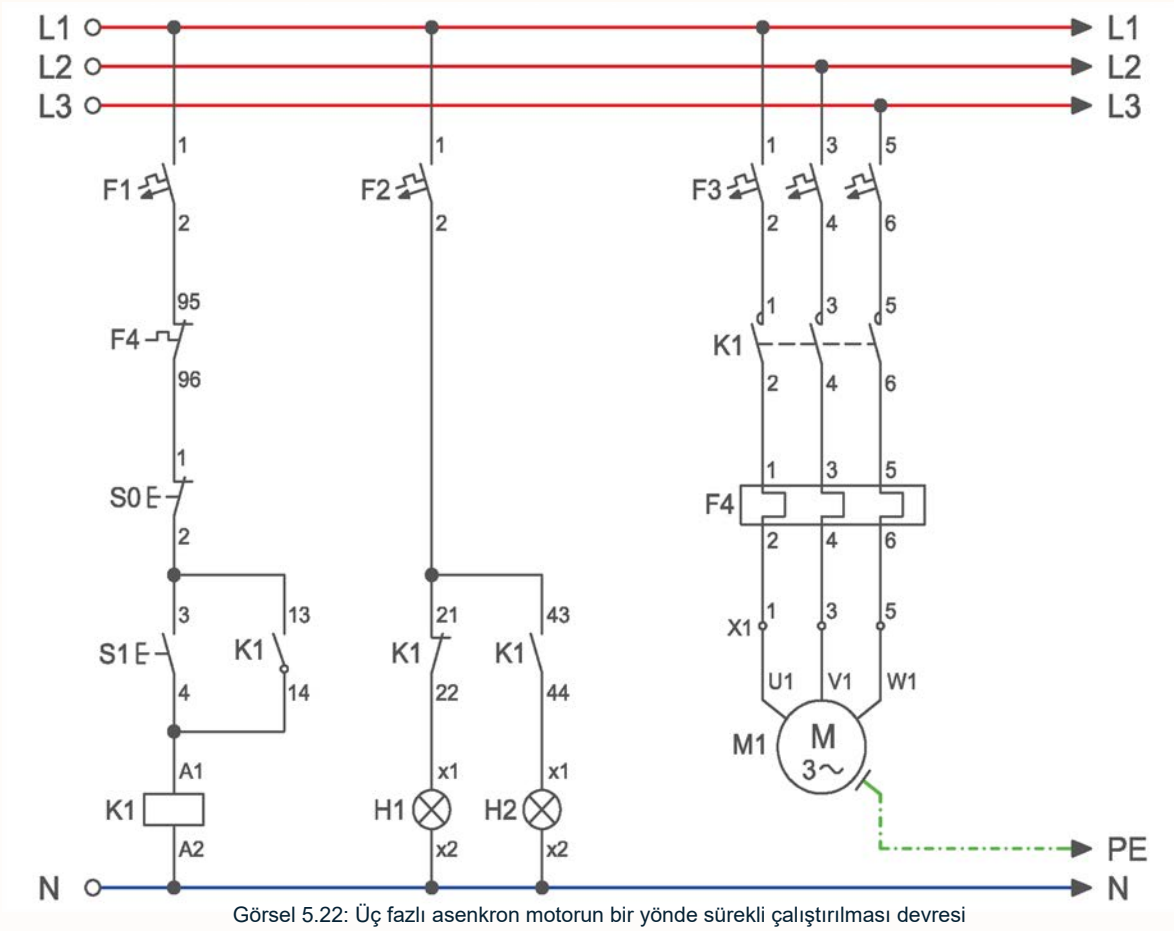
1. Kesik çalışma ne demektir? Kesik çalışma devresinin kullanım alanına örnekler veriniz.
2. Devrede start butonu yerine stop butonu kullanılırsa nasıl bir çalışma ortaya çıkar? Açıklayınız.
3. Devrede sinyal lambaları niçin kullanılmıştır? Açıklayınız.
4. Sinyal lambalarının önünde bulunan NO ve NC kontakları yer değiştirirse devre nasıl çalışır? Açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Kumanda devre bağlantılarını yaptı.	20	
Sınıfı :	2	Güç devre bağlantılarını yaptı.	20	
Numarası :	3	Kumanda devresini doğru çalıştırdı.	20	
ÖĞRETMEN	4	Güç devresini doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışmada temizlik ve düzene dikkat etti.	20	
İmza :		**TOPLAM PUAN**	100	

6. UYGULAMA

ÜÇ FAZLI ASENKRON MOTORUN SÜREKLİ ÇALIŞTIRILMASI

AMAÇ: **Üç fazlı asenkron motoru bir yönde sürekli olarak çalıştırmak.**



Görsel 5.22: Üç fazlı asenkron motorun bir yönde sürekli çalıştırılması devresi

DEVRENİN ÇALIŞMASI

Kumanda devresinde S1 butonuna basıldığında K1 kontaktörü enerjilenir ve kontakları konum değiştirir. Kumanda devresindeki S1 butonuna paralel bağlı NO kontağı kapanarak K1 kontaktörünün kalıcı olarak çalışmasını sağlar. Buton bırakılsa bile devre çalışmaya devam eder. Start butonuna paralel bağlanan ve sürekli çalışmayı sağlayan bu kontağa **mühürleme kontağı** denir.

K1 kontaktörünün kontaklarının konum değiştirmesiyle güç devresinde motora seri bağlı güç kontakları da kapanarak motoru çalıştırır. Motorun çalışması S0 butonuna basıncaya dek devam eder. Motor dururken kırmızı, çalışırken yeşil sinyal lambası yanar.

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Buton	Start ve stop ani temaslı	Birer adet
Kontaktör	4 kW	1 adet
Sinyal lambası	Kırmızı ve yeşil	2 adet
Aşırı akım rölesi	2,5-4 A	1 adet
Asenkron motor	Üç fazlı	1 adet
Kumanda kabloları	2,5 mm^2 NYAF	-
Ölçü aleti ve el aletleri	Multimetre, pense, yan keski, kontrol kalemi	-

İŞLEM BASAMAKLARI

1. Ölçü aleti ile devre eleman ve kablolarının sağlamlık kontrollerini yapınız.
2. Görsel 5.22'de verilen şemaya göre kumanda devre bağlantılarını yapınız.
3. Öğretmeninizin kontrolünde kumanda devresine enerji veriniz ve kırmızı sinyal lambasının yandığını gözlemleyiniz.
4. S1 butonuna basarak kontaktörün kontaklarını çektiğini ve yeşil sinyal lambasının yandığını gözlemleyiniz.
5. Enerjiyi kesiniz.
6. Görsel 5.22'de verilen şemaya göre güç devre bağlantılarını yapınız.
7. Öğretmeninizin kontrolünde devreye enerji veriniz ve kırmızı sinyal lambasının yandığını gözlemleyiniz.
8. S1 butonuna basınız ve motorun çalıştığını, motor çalışırken yeşil sinyal lambasının yandığını gözlemleyiniz.
9. S1 butonundan elinizi çekiniz ve sonucu gözlemleyiniz.
10. Enerjiyi kesiniz.
11. Malzemeleri sökerek öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

SORULAR

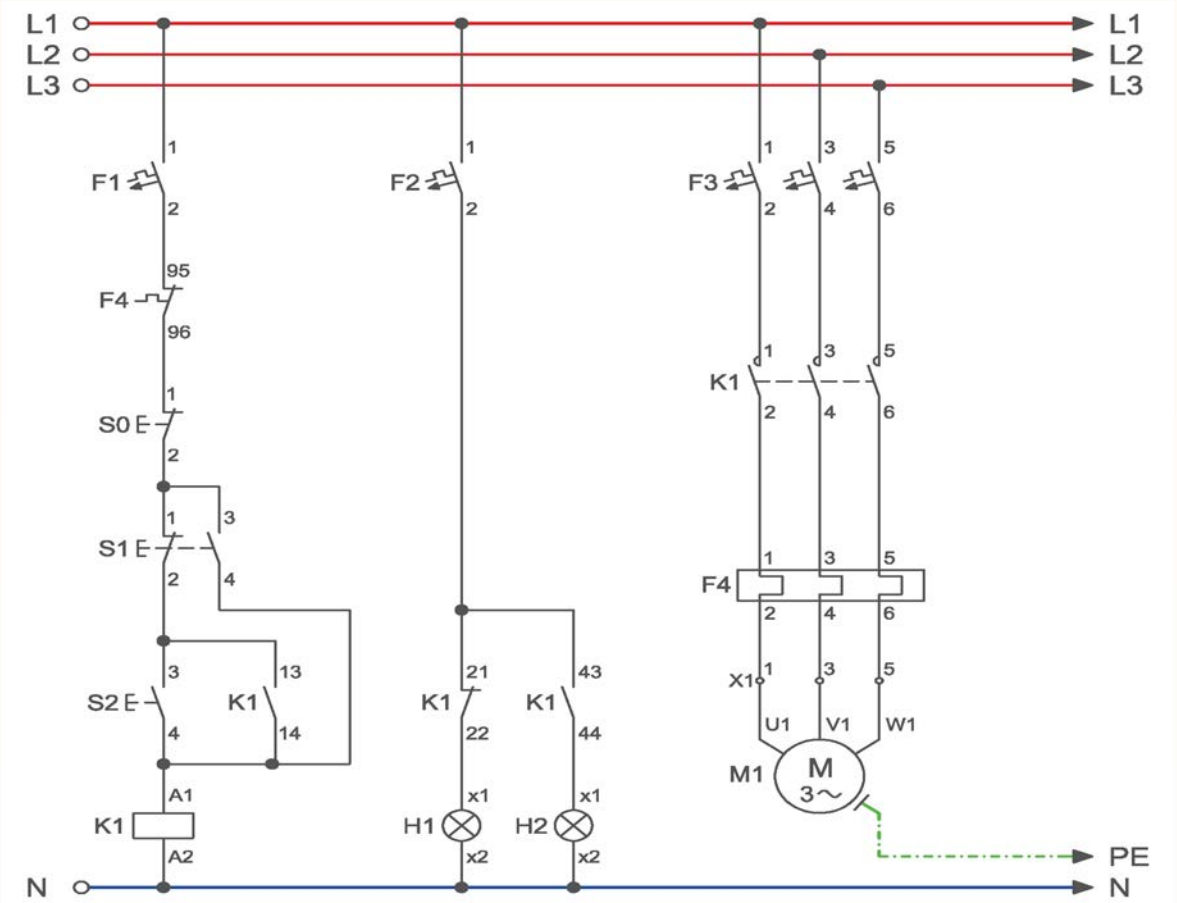
1. Devrenin kesik çalıştırma devresinden farkı nedir? Açıklayınız.
2. Mühürleme kontağının görevini açıklayınız.
3. Mühürleme kontağı normalde kapalı kontak olsaydı devre nasıl çalışırdı? Açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Kumanda devre bağlantılarını yaptı.	20	
Sınıfı :	2	Güç devre bağlantılarını yaptı.	20	
Numarası :	3	Kumanda devresini doğru çalıştırdı.	20	
ÖĞRETMEN	4	Güç devresini doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışmada temizlik ve düzene dikkat etti.	20	
İmza :		**TOPLAM PUAN**	100	

7. UYGULAMA

ÜÇ FAZLI ASENKRON MOTORUN KESİK VE SÜREKLİ ÇALIŞTIRILMASI

AMAÇ: **Üç fazlı asenkron motoru bir yönde kesik ve sürekli olarak çalıştırmak.**



Görsel 5.23: Üç fazlı asenkron motorun bir yönde kesik ve sürekli çalıştırılması devresi

DEVRENİN ÇALIŞMASI

Kumanda devresinde S2 start butonuna basıldığında K1 kontaktörü enerjilenerek motoru çalıştırır. Kumanda devresindeki mühürleme kontağı kapanarak sürekli çalışmayı sağlar. Motorun durdurulması için S0 stop butonuna basılması yeterlidir. S1 jog butonu kesik çalışma için kullanılır. Motor dururken kırmızı, çalışırken yeşil sinyal lambası yanar.

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Start, stop ve jog butonu	Ani temaslı	Birer adet
Kontaktör	4 kW	1 adet
Sinyal lambası	Kırmızı ve yeşil	Birer adet
Aşırı akım rölesi	2,5-4 A	1 adet
Asenkron motor	Üç fazlı	1 adet
Kumanda kabloları	2,5 mm^2 NYAF	-
Ölçü aleti ve el aletleri	Multimetre, pense, yan keski, kontrol kalemi	-

İŞLEM BASAMAKLARI

1. Ölçü aleti ile devre eleman ve kablolarının sağlamlık kontrollerini yapınız.
2. Görsel 5.23'te verilen şemaya göre kumanda devre bağlantılarını yapınız.
3. Öğretmeninizin kontrolünde kumanda devresine enerji veriniz ve kırmızı sinyal lambasının yandığını gözlemleyiniz.
4. S1 butonuna basarak kontaktörün kontaklarını çektiğini ve kırmızı sinyal lambasının sönerek yeşil sinyal lambasının yandığını gözlemleyiniz.
5. S1 butonunu bırakarak sonucu gözlemleyiniz.
6. S2 butonuna basıp bırakarak kontaktör ve sinyal lambalarının durumunu gözlemleyiniz.
7. Enerjiyi kesiniz.
8. Görsel 5.23'te verilen şemaya göre güç devre bağlantılarını yapınız.
9. Öğretmeninizin kontrolünde devreye enerji veriniz ve kırmızı sinyal lambasının yandığını gözlemleyiniz.
10. S1 butonuna basıp bırakarak motoru kesik çalıştırınız. Bu arada sinyal lambalarının durumunu gözlemleyiniz.
11. S2 butonuna basarak motoru sürekli çalıştırınız ve sinyal lambalarının durumunu gözlemleyiniz.
12. S0 butonuna basarak motoru durdurunuz.
13. Enerjiyi kesiniz.
14. Malzemeleri sökerek öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

SORU

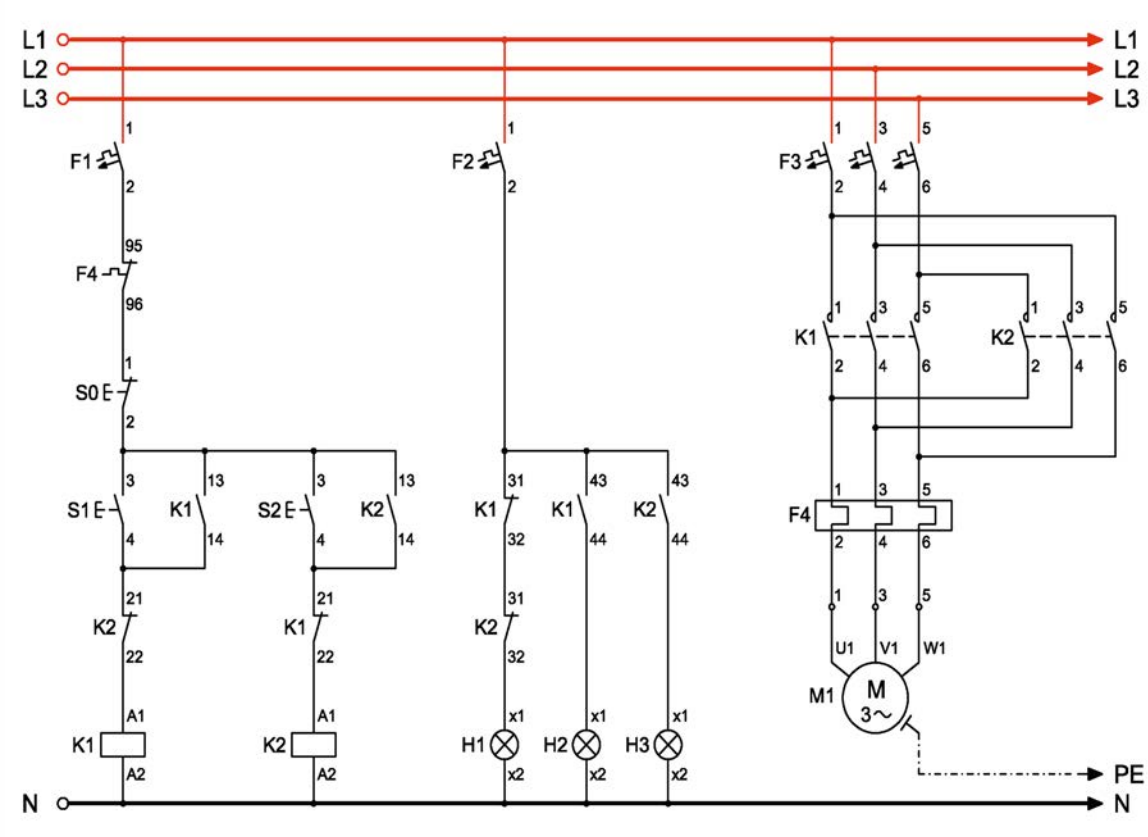
- Devrede kullanılan butonların görevini açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Kumanda devre bağlantılarını yaptı.	20	
Sınıfı :	2	Güç devre bağlantılarını yaptı.	20	
Numarası :	3	Kumanda devresini doğru çalıştırdı.	20	
ÖĞRETMEN	4	Güç devresini doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışmada temizlik ve düzene dikkat etti.	20	
İmza :		**TOPLAM PUAN**	100	

8. UYGULAMA

ÜÇ FAZLI ASENKRON MOTORUN DEVİR YÖNÜNÜN DEĞİŞTİRİLMESİ (ELEKTRİKSEL KİLİTLEMELİ)

AMAÇ: **Üç fazlı asenkron motorun devir yönünü elektriksel kilitlemeli olarak değiştirmek.**



Görsel 5.24: Üç fazlı asenkron motorun elektriksel kilitlemeli devir yönünün değiştirilmesi devresi

DEVRENİN ÇALIŞMASI

İki kontaktörün normalde kapalı kontaklarının karşılıklı olarak birbirinin önüne bağlanmasıyla yapılan korumaya **elektriksel kilitleme** denir. Bu bağlantı iki kontaktörün aynı anda çalışmaması istenen yerlerde, özellikle devir yönü değiştirme uygulamalarında kullanılır. Bu kilitleme sayesinde motor bir yöne dönerken diğer yöne doğru dönmesi engellenmiş olur.

Devrede S1 butonuna basıldığında K1 kontaktörü enerjilenir ve motor ileri yönde dönmeye başlar. K1 kontaktörüne ait K1 normalde kapalı kontağı açıldığı için S2 start butonuna basılsa bile K2 kontaktörü enerjilenemeyeceğinden motor geri yönde dönemez. S0 stop butonuna basılarak motor durdurulur.

S2 butonuna basıldığında K2 kontaktörü enerjilenir ve motor geri yönde döner. K2 kontaktörü çalışırken, K1 kontaktör bobinine seri bağlı K2 açık kontağı açılacağından S1 butonuna basılsa dahi motor ileri yönde çalışmaz. S0 stop butonuna basılarak motor durdurulur.

MALZEME LİSTESİ

Adı	Özelliği	Miktarı
Start ve stop butonu	Ani temaslı	2+1 adet
Kontaktör	4 kW	2 adet
Sinyal lambası	Kırmızı, yeşil ve sarı	Birer adet
Aşırı akım rölesi	2,5-4 A	1 adet
Asenkron motor	Üç fazlı	1 adet
Kumanda kabloları	2,5 mm^2 NYAF	-
Ölçü aleti ve el aletleri	Multimetre, pense, yan keski, kontrol kalemi	-

İŞLEM BASAMAKLARI

1. Ölçü aleti ile devre eleman ve kablolarının sağlamlık kontrollerini yapınız.
2. Görsel 5.24'te verilen şemaya göre kumanda devre bağlantılarını yapınız.
3. Öğretmeninizin kontrolünde kumanda devresine enerji veriniz.
4. S1 butonuna basarak kontaktörün kontaklarını çektiğini gözlemleyiniz.
5. Bu arada S2 butonuna basarak sonucu gözlemleyiniz.
6. S0 butonuna basınız ve K1 kontaktörünün kontaklarını bıraktığını gözlemleyiniz.
7. S2 butonuna basarak K2 kontaktörünün kontaklarını çektiğini gözlemleyiniz.
8. Bu arada S1 butonuna basarak sonucu gözlemleyiniz.
9. Enerjiyi kesiniz.
10. Görsel 5.24'te verilen şemaya göre güç devre bağlantılarını yapınız.
11. Öğretmeninizin kontrolünde devreye enerji veriniz.
12. S1 butonuna basınız ve motorun ileri yönde çalıştığını gözlemleyiniz.
13. Bu arada S2 butonuna basarak sonucu gözlemleyiniz.
14. S0 butonuna basınız ve motorun durduğunu gözlemleyiniz.
15. S2 butonuna basarak motorun geri yönde çalıştığını gözlemleyiniz.
16. Bu arada S1 butonuna basarak sonucu gözlemleyiniz.
17. S0 butonuna basarak motoru durdurunuz ve enerjiyi kesiniz.
18. Malzemeleri sökerek öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

SORU

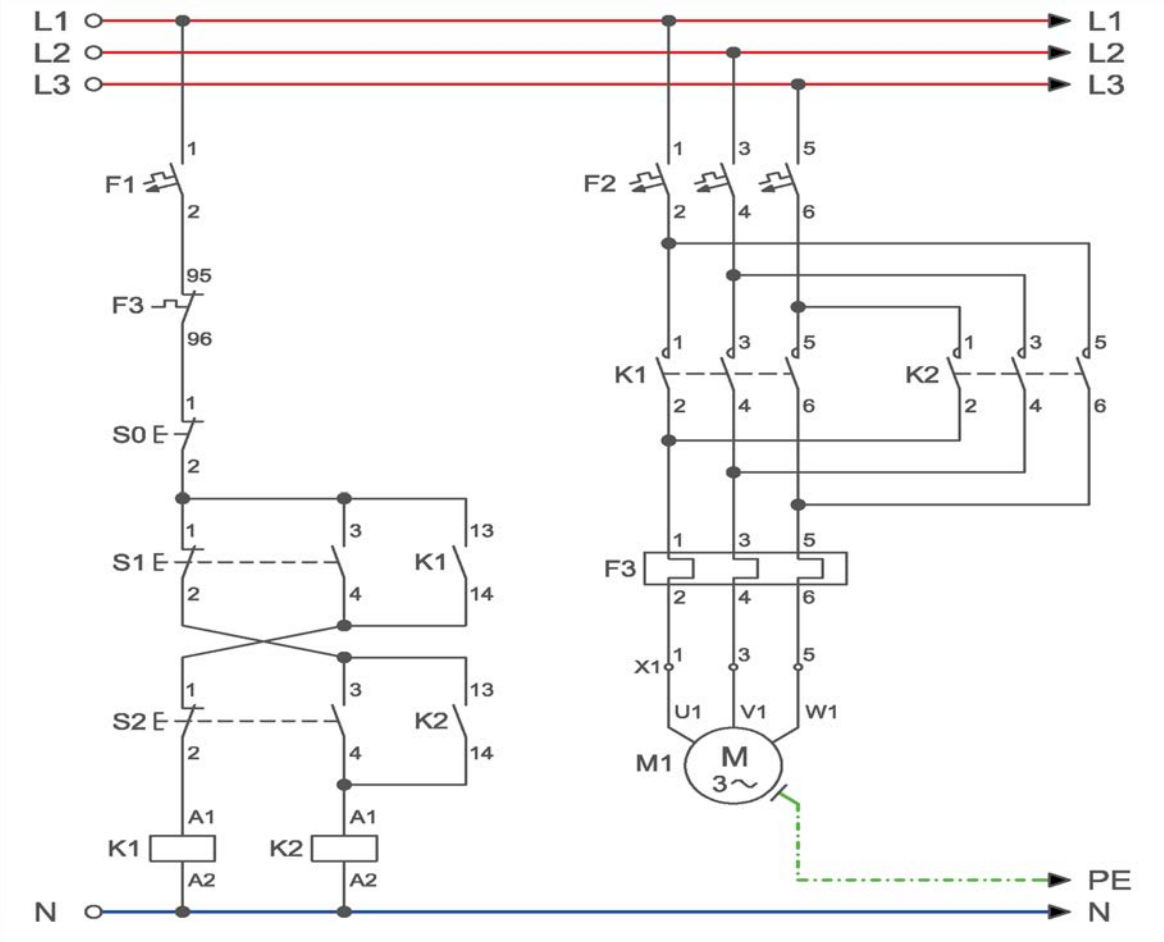
- Asenkron motorlarda devir yönü nasıl değiştirilir? Açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Kumanda devre bağlantılarını yaptı.	20	
Sınıfı :	2	Güç devre bağlantılarını yaptı.	20	
Numarası :	3	Kumanda devresini doğru çalıştırdı.	20	
ÖĞRETMEN	4	Güç devresini doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışmada temizlik ve düzene dikkat etti.	20	
İmza :		**TOPLAM PUAN**	100	

9. UYGULAMA

ÜÇ FAZLI ASENKRON MOTORUN DEVİR YÖNÜNÜN DEĞİŞTİRİLMESİ (BUTONSAL KİLİTLEMELİ)

AMAÇ: **Üç fazlı asenkron motorun butonsal kilitlemeli olarak devir yönünü değiştirmek.**



Görsel 5.25: Üç fazlı asenkron motorun butonsal kilitlemeli devir yönünün değiştirilmesi devresi

DEVRENİN ÇALIŞMASI

Jog butonu kontakları ile yapılan kilitlemeye **butonsal kilitleme** denir. K1 ve K2 kontaktör bobinleri enerjilerini jog butonlarının üst kontaklarından alır. Devrede S1 butonuna basıldığında S2 jog butonu üzerinden K1 kontaktörü enerjilenir ve motor ileri yönde çalışmaya başlar. Motoru geri yönde çalıştırmak için S2 butonuna basmak yeterlidir.

S2 butonuna basıldığında K1 kontaktörünün enerjisi kesilir ve ileri yönde çalışma sonlanarak motor geri yönde çalışmaya başlar. Motoru durdurmak için S0 butonuna basmak yeterlidir.

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Stop ve jog butonu	Ani temaslı	1+2 adet
Kontaktör	4 kW	2 adet
Sinyal lambası	Kırmızı ve yeşil	Birer adet
Aşırı akım rölesi	2,5-4 A	1 adet
Asenkron motor	Üç fazlı	1 adet
Kumanda kabloları	2,5 mm^2 NYAF	-
Ölçü aleti ve el aletleri	Multimetre, pense, yan keski, kontrol kalemi	-

İŞLEM BASAMAKLARI

1. Ölçü aleti ile devre eleman ve kablolarının sağlamlık kontrollerini yapınız.
2. Görsel 5.25'te verilen şemaya göre kumanda devre bağlantılarını yapınız.
3. Öğretmeninizin kontrolünde kumanda devresine enerji veriniz.
4. S1 butonuna basarak K1 kontaktörünün kontaklarını çektiğini gözlemleyiniz.
5. S2 butonuna basarak, K1 kontaktörünün kontaklarını bırakıp K2 kontaktörünün kontaklarını çektiğini gözlemleyiniz.
6. S0 butonuna basarak çekili kontaktörün kontaklarını bıraktığını gözlemleyiniz.
7. Enerjiyi kesiniz.
8. Görsel 5.25'te verilen şemaya göre güç devre bağlantılarını yapınız.
9. Öğretmeninizin kontrolünde devreye enerji veriniz.
10. S1 butonuna basınız ve motorun ileri yönde çalıştığını gözlemleyiniz.
11. S2 butonuna basınız ve motorun geri yönde çalıştığını gözlemleyiniz.
12. S0 butonuna basarak motoru durdurunuz.
13. Enerjiyi kesiniz.
14. Malzemeleri sökerek öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

SORU

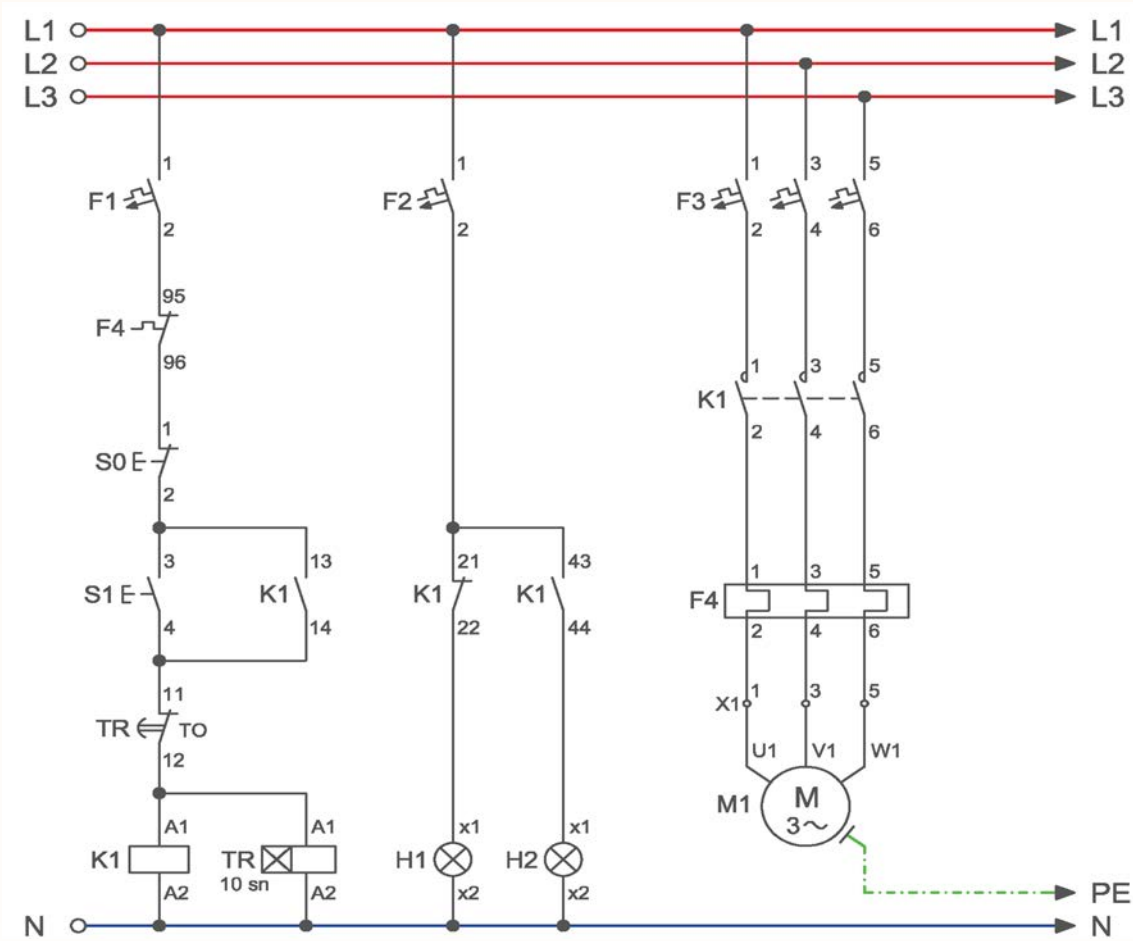
- Elektriksel ve butonsal kilitlemeli devir yönü değiştirme devrelerinin farklarını açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Kumanda devre bağlantılarını yaptı.	20	
Sınıfı :	2	Güç devre bağlantılarını yaptı.	20	
Numarası :	3	Kumanda devresini doğru çalıştırdı.	20	
ÖĞRETMEN	4	Güç devresini doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışmada temizlik ve düzene dikkat etti.	20	
İmza :		**TOPLAM PUAN**	100	

10. UYGULAMA

ÜÇ FAZLI ASENKRON MOTORUN ZAMAN AYARLI DURDURULMASI

AMAÇ: **Üç fazlı asenkron motoru düz zaman rölesiyle zaman ayarlı olarak durdurmak.**



Görsel 5.26: Üç fazlı asenkron motorun düz zaman rölesiyle durdurulması devresi

DEVRENİN ÇALIŞMASI

Bazı hâllerde asenkron motorların belirli bir sürenin sonunda durması istenir. Bu işlem zaman röleleri ile gerçekleştirilir. Bu devrede düz zaman rölesi kullanılmıştır.

S1 butonuna basıldığında K1 kontaktörü enerjilenerek motoru çalıştırır. Bu esnada TR düz zaman rölesi de enerjilenerek belirlenen süreyi saymaya başlar. Ayarlanan süre sonunda zaman rölesinin kontakları konum değiştirir ve K1 kontaktör bobinine seri bağlı, normalde kapalı, gecikmeli açılan kontak açılır. Bobin enerjisi kesilen K1 kontaktörü güç kontaklarını açarak motoru durdurur. Ayarlanan süre beklenmeden motor durdurulmak istenirse S0 stop butonuna basılır. Motor dururken kırmızı, çalışırken yeşil sinyal lambası yanar.

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Start ve stop	Ani temaslı	Birer adet
Kontaktör	4 kW	1 adet
Sinyal lambası	Kırmızı ve yeşil	Birer adet
Aşırı akım rölesi	2,5-4 A	1 adet
Asenkron motor	Üç fazlı	1 adet
Kumanda kabloları	2,5 mm^2 NYAF	-
Ölçü aleti ve el aletleri	Multimetre, pense, yan keski, kontrol kalemi	-

İŞLEM BASAMAKLARI

1. Ölçü aleti ile devre eleman ve kablolarının sağlamlık kontrollerini yapınız.
2. Görsel 5.26'da verilen şemaya göre kumanda devre bağlantılarını yapınız.
3. Zaman rölesini 10 sn.ye ayarlayınız.
4. Öğretmeninizin kontrolünde kumanda devresine enerji veriniz ve kırmızı sinyal lambasının yandığını gözlemleyiniz.
5. S1 butonuna basarak kontaktörün kontaklarını çektiğini ve yeşil sinyal lambasının yandığını gözlemleyiniz.
6. Ayarlanan süre sonunda K1 kontaktörünün enerjisinin kesildiğini ve kontaktörün kontaklarını bıraktığını gözlemleyiniz.
7. Enerjiyi kesiniz.
8. Görsel 5.26'da verilen şemaya göre güç devre bağlantılarını yapınız.
9. Öğretmeninizin kontrolünde devreye enerji veriniz ve kırmızı sinyal lambasının yandığını gözlemleyiniz.
10. S1 butonuna basınız ve motorun çalıştığını, motor çalışırken yeşil sinyal lambasının yandığını gözlemleyiniz.
11. 10 sn. bekleyiniz ve süre sonunda motorun kendiliğinden durduğunu gözlemleyiniz.
12. Enerjiyi kesiniz.
13. Malzemeleri sökerek öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

SORULAR

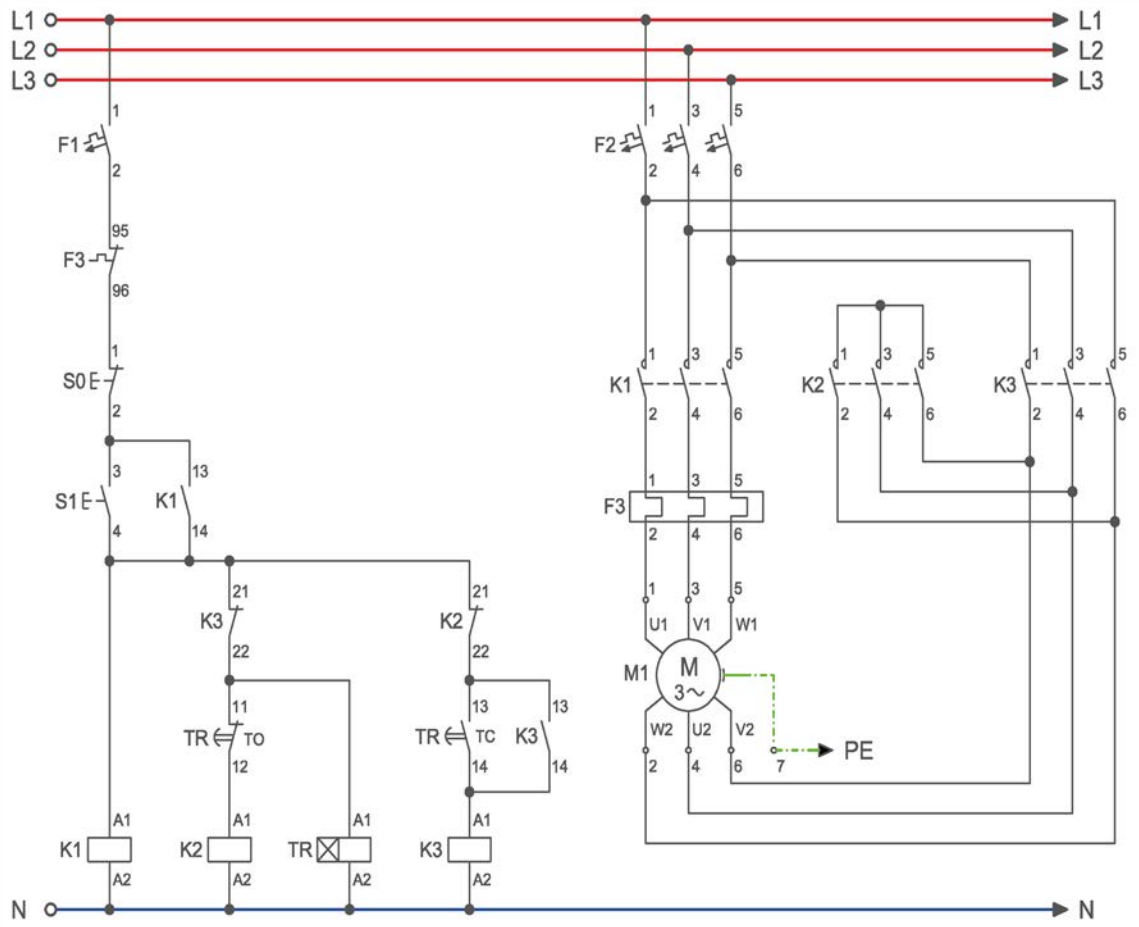
1. Motorun niçin 10 sn. çalıştığını açıklayınız.
2. Devrede ters zaman rölesi kullanılsaydı devre nasıl çalışırdı? Açıklayınız.
3. K1 kontaktörünün önüne zaman rölesinin normalde açık kontağı bağlansaydı devre nasıl çalışırdı? Açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Kumanda devre bağlantılarını yaptı.	20	
Sınıfı :	2	Güç devre bağlantılarını yaptı.	20	
Numarası :	3	Kumanda devresini doğru çalıştırdı.	20	
ÖĞRETMEN	4	Güç devresini doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışmada temizlik ve düzene dikkat etti.	20	
İmza :		**TOPLAM PUAN**	100	

11. UYGULAMA

ÜÇ FAZLI ASENKRON MOTORA OTOMATİK YILDIZ-ÜÇGEN YOL VERME

AMAÇ: **Üç fazlı asenkron motora otomatik olarak yıldız-üçgen yol vermek.**



Görsel 5.27: Üç fazlı asenkron motora yıldız-üçgen yol verme

DEVRENİN ÇALIŞMASI

Devrede S1 butonuna basıldığında K1, K2 (λ) kontaktörleri ve TR zaman rölesi enerjilenir. Devre kendini mühürler ve motor yıldız çalışmaya başlar. Aynı anda TR zaman rölesi de ayarlanan süreyi saymaya başlar. Süre sonunda zaman rölesinin kontakları konum değiştirir. Zaman rölesinin normalde kapalı kontağı açılarak K2 kontaktörünün enerjisini keser. Motorun yıldız çalışması sona erer. Aynı anda K3 (Δ) kontaktörü enerjileneceğinden motor üçgen çalışmaya geçer. Devreyi durdurmak için S0 stop butonuna basmak yeterlidir.

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Start ve stop butonu	Ani temaslı	Birer adet
Kontaktör	4 kW	3 adet
Zaman rölesi	Düz	1 adet
Aşırı akım rölesi	2,5-4 A	1 adet
Asenkron motor	Üç fazlı	1 adet
Kumanda kabloları	2,5 mm^2 NYAF	-
Ölçü aleti ve el aletleri	Multimetre, pense, yan keski, kontrol kalemi	-

İŞLEM BASAMAKLARI

1. Ölçü aleti ile devre eleman ve kablolarının sağlamlık kontrollerini yapınız.
2. Görsel 5.27'de verilen şemaya göre kumanda devre bağlantılarını yapınız.
3. Zaman rölesini 5 sn.ye ayarlayınız.
4. Öğretmeninizin kontrolünde kumanda devresine enerji veriniz.
5. S1 butonuna basarak K1 ve K2 kontaktörlerinin kontaklarını çektiğini gözlemleyiniz.
6. 5 sn. bekleyiniz ve sürenin sonunda K2 kontaktörünün kontaklarını bırakıp K3 kontaktörünün kontaklarını çektiğini gözlemleyiniz.
7. Stop butonuna basarak sonucu gözlemleyiniz.
8. Enerjiyi kesiniz.
9. Görsel 5.27'de verilen şemaya göre güç devre bağlantılarını yapınız.
10. Öğretmeninizin kontrolünde devreye enerji veriniz.
11. S1 butonuna basınız ve motorun önce yıldız sonra üçgen çalıştığını gözlemleyiniz.
12. S0 butonuna basarak motoru durdurunuz.
13. Enerjiyi kesiniz.
14. Malzemeleri sökerek öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

SORULAR

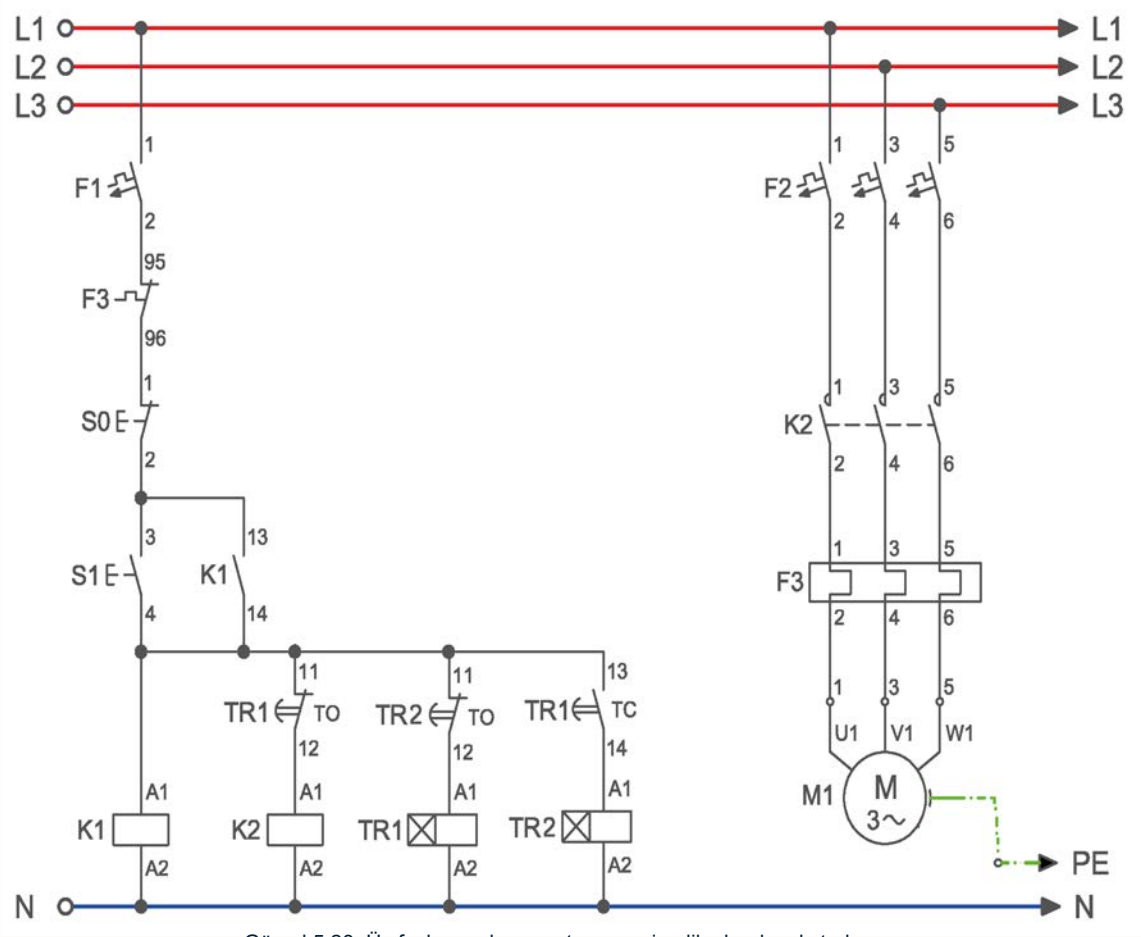
1. Devrede kullanılan K1, K2 ve K3 kontaktörlerinin görevleri nelerdir? Belirtiniz.
2. Devrede normalde kapalı K2 ve K3 kontaklarının görevini açıklayınız.
3. Yıldız çalışmadan üçgen çalışmaya geçiş süresinin normalden az veya çok olmasının sakıncaları nelerdir? Açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Kumanda devre bağlantılarını yaptı.	20	
Sınıfı :	2	Güç devre bağlantılarını yaptı.	20	
Numarası :	3	Kumanda devresini doğru çalıştırdı.	20	
ÖĞRETMEN	4	Güç devresini doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışmada temizlik ve düzene dikkat etti.	20	
İmza :		**TOPLAM PUAN**	100	

12. UYGULAMA

ÜÇ FAZLI ASENKRON MOTORUN PERİYODİK ÇALIŞTIRILMASI

AMAÇ: **Üç fazlı asenkron motoru belirli zaman aralıklarıyla periyodik olarak çalıştırmak.**



Görsel 5.28: Üç fazlı asenkron motorun periyodik olarak çalıştırılması

DEVRENİN ÇALIŞMASI

Devrede TR1 ve TR2 zaman röleleri 5 sn.ye ayarlanır. S1 butonuna basıldığında K1, K2 kontaktörü ile TR1 zaman rölesi aynı anda enerjilenir. K2 kontaktörü motoru çalıştırırken TR1 düz zaman rölesi de belirlenen süreyi saymaya başlar. Ayarlanan süre sonunda TR1 zaman rölesinin kontakları konum değiştirir ve K2 kontaktör bobinine seri bağlı, normalde kapalı, gecikmeli açılan kontak açılır. Bobin enerjisi kesilen K2 kontaktörü güç kontaklarını açarak motoru durdurur. Bu arada TR2 zaman rölesi enerjilenir ve süreyi saymaya başlar. Ayarlanan süre sonunda TR1 zaman rölesinin enerjisi kesilir ve kontakları normal konumuna döner. K2 kontaktörü tekrar enerjilenir ve motor çalışır. Motor periyodik olarak 5 sn. çalışıp 5 sn. durur.

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Start ve stop butonu	Ani temaslı	Birer adet
Kontaktör	4 kW	3 adet
Zaman rölesi	Düz	2 adet
Aşırı akım rölesi	2,5-4 A	1 adet
Asenkron motor	Üç fazlı	1 adet
Kumanda kabloları	2,5 mm^2 NYAF	-
Ölçü aleti ve el aletleri	Multimetre, pense, yan keski, kontrol kalemi	-

İŞLEM BASAMAKLARI

1. Ölçü aleti ile devre eleman ve kablolarının sağlamlık kontrollerini yapınız.
2. Görsel 5.28'de verilen şemaya göre kumanda devre bağlantılarını yapınız.
3. TR1 ve TR2 zaman rölelerini 5 sn.ye ayarlayınız.
4. Öğretmeninizin kontrolünde kumanda devresine enerji veriniz.
5. S1 butonuna basarak K1 ve K2 kontaktörünün kontaklarını çektiğini gözlemleyiniz.
6. 5 sn. sonra K2 kontaktörünün kontaklarını bıraktığını ve yine 5 sn. sonra tekrar çektiğini gözlemleyiniz.
7. Stop butonuna basarak çalışmayı sonlandırınız.
8. Enerjiyi kesiniz.
9. Görsel 5.28'de verilen şemaya göre güç devre bağlantılarını yapınız.
10. Öğretmeninizin kontrolünde devreye enerji veriniz.
11. S1 butonuna basınız ve motorun 5 sn. çalışıp 5 sn. durduğunu gözlemleyiniz.
12. S0 butonuna basıp devrenin çalışmasını sonlandırınız.
13. Enerjiyi kesiniz.
14. Malzemeleri sökerek öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

SORULAR

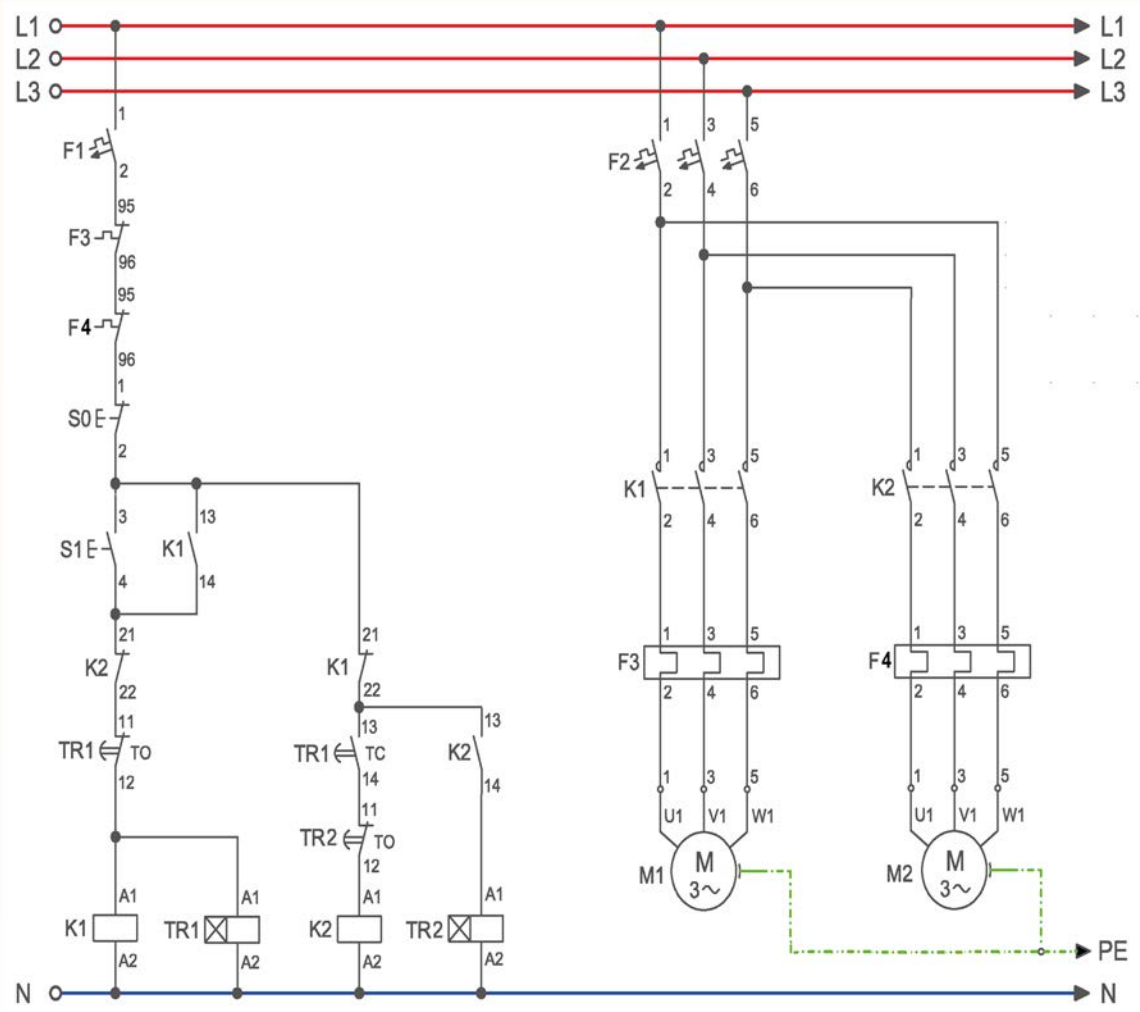
1. TR1 zaman rölesi 10 sn. olarak ayarlanırsa devre nasıl çalışır? Açıklayınız.
2. TR2 zaman rölesi arızalanırsa devrenin çalışması nasıl etkilenir? Açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Kumanda devre bağlantılarını yaptı.	20	
Sınıfı :	2	Güç devre bağlantılarını yaptı.	20	
Numarası :	3	Kumanda devresini doğru çalıştırdı.	20	
ÖĞRETMEN	4	Güç devresini doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışmada temizlik ve düzene dikkat etti.	20	
İmza :		**TOPLAM PUAN**	100	

13. UYGULAMA

İKİ ADET ÜÇ FAZLI ASENKRON MOTORUN BELİRLENEN ŞARTLARDA ÇALIŞTIRILMASI

AMAÇ: **İki adet üç fazlı asenkron motoru istenen şartlarda çalıştırmak.**



Görsel 5.29: İki adet üç fazlı asenkron motorun istenen şartlarda çalıştırılması

DEVRENİN ÇALIŞMASI

Devrede TR1 zaman rölesi 5 sn. ve TR2 zaman rölesi de 10 sn.ye ayarlanır. S1 butonuna basıldığında K1 kontaktörü ve TR1 zaman rölesi enerjilenir. Birinci motor (M1) çalışmaya başlar. 5 sn. sonra TR1 düz zaman rölesi kontakları konum değiştirir ve K2 kontaktörü ile TR2 zaman rölesi enerjilenir. Birinci motor dururken, ikinci motor (M2) ters yönde dönerek çalışmaya başlar. 10 sn. sonra ikinci motor da durur.

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Start ve stop butonu	Ani temaslı	Birer adet
Kontaktör	4 kW	2 adet
Zaman rölesi	Düz	2 adet
Aşırı akım rölesi	2,5-4 A	2 adet
Asenkron motor	Üç fazlı	2 adet
Kumanda kabloları	2,5 mm^2 NYAF	-
Ölçü aleti ve el aletleri	Multimetre, pense, yan keski, kontrol kalemi	-

İŞLEM BASAMAKLARI

1. Ölçü aleti ile devre eleman ve kablolarının sağlamlık kontrollerini yapınız.
2. Görsel 5.29'da verilen şemaya göre kumanda devre bağlantılarını yapınız.
3. TR1 zaman rölesini 5 sn. ve TR2 zaman rölesini de 10 sn.ye ayarlayınız.
4. Öğretmeninizin kontrolünde kumanda devresine enerji veriniz.
5. S1 butonuna basarak K1 kontaktörünün kontaklarını çektiğini gözlemleyiniz.
6. 5 sn. sonra K1 kontaktörünün kontaklarını bırakarak K2 kontaktörünün çektiğini ve 10 sn. sonra K2 kontaktörünün de kontaklarını bıraktığını gözlemleyiniz.
7. Enerjiyi kesiniz.
8. Görsel 5.29'da verilen şemaya göre güç devre bağlantılarını yapınız.
9. Öğretmeninizin kontrolünde devreye enerji veriniz.
10. S1 butonuna basınız ve birinci motorun çalıştığını gözlemleyiniz.
11. 5 sn. sonra birinci motorun durup ikinci motorun çalıştığını ve ters yönde döndüğünü gözlemleyiniz.
12. 10 sn. sonra ikinci motorun da durduğunu gözlemleyiniz.
13. Enerjiyi kesiniz.
14. Malzemeleri sökerek öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

SORULAR

1. Devrede K1 ve K2 normalde kapalı kontakları hangi amaçla kullanılmıştır? Belirtiniz.
2. İkinci motorun ters yönde dönmesinin sebebi nedir? Açıklayınız.
3. Devrede stop butonuna ne zaman ihtiyaç duyulur? Belirtiniz.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Kumanda devre bağlantılarını yaptı.	20	
Sınıfı :	2	Güç devre bağlantılarını yaptı.	20	
Numarası :	3	Kumanda devresini doğru çalıştırdı.	20	
ÖĞRETMEN	4	Güç devresini doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışmada temizlik ve düzene dikkat etti.	20	
İmza :		**TOPLAM PUAN**	100	



ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki cümlelerin başındaki boşluğa cümleler doğru ise (D), yanlış ise (Y) yazınız.

1. (…) Motorların yapı tiplerine göre en güvenilir yapı, kapalı tip asenkron motordur.
2. (…) Otomatik sigortalar, B ve C olmak üzere iki tiptir.
3. (…) Kontaktör, IEC normuna göre M ile isimlendirilir.
4. (…) Enerji verildikten sonra gecikme yapan zaman rölesi düz zaman rölesidir.
5. (…) NO normalde kapalı kontağı, NC normalde açık kontağı ifade eder.

B) Aşağıdaki cümlelerde boş bırakılan yerlere uygun sözcüğü yazınız.

6. Üç fazlı asenkron motorun devir yönünü değiştirmek için …… yeri değiştirilir.
7. Stator devri ile rotor devri arasındaki devir farkına …… denir.
8. İki kontaktörün normalde kapalı kontaklarının karşılıklı olarak birbirinin önüne bağlanmasıyla yapılan korumaya …… denir.
9. Büyük güçlü elektromanyetik anahtarlara …… denir.
10. IEC normuna göre üç fazlı asenkron motor sargı giriş uçları …… şeklindedir.

C) Aşağıdaki soruların doğru seçeneğini işaretleyiniz.

11. Asenkron motorun dönen parçası aşağıdakilerden hangisidir?
A) Gövde B) Kapak C) Rotor
D) Sargı E) Stator

12. Aşağıdakilerden hangisi yapı tipine göre asenkron motor çeşididir?
A) Dik B) Flanşlı tip C) Sincap kafesli
D) Üç fazlı E) Yatık

13. Biri normalde kapalı, diğeri ise normalde açık iki kontağa sahip olan ve üzerine basıldığında kontakların konum değiştirdiği butona ne ad verilir?
A) Jog B) Paket şalter C) Start
D) Stop E) Tek yollu

14. Küçük değerli bir akım ile yüksek güçlü bir alıcıyı çalıştırabilmek için kullanılan elemanlara ne ad verilir?
A) Asenkron motor B) Röle C) Sınır anahtarı
D) Sigorta E) Sinyal lambası

15. Aşağıdakilerden hangisi start butonuna paralel bağlanan ve devrenin sürekli çalışmasını sağlayan kontağa verilen isimdir?
A) Düz B) Faz koruma C) Mühürleme
D) Ters E) Yıldız–üçgen

6. ÖĞRENME BİRİMİ

PLC SEÇİMİ

KONULAR

6.1. YAPI VE ÇALIŞMA PRENSİBİNE GÖRE PLC SEÇİMİ
6.2. PLC'NİN YAPI VE ÇALIŞMA PRENSİBİNE GÖRE GİRİŞ-ÇIKIŞ VE ÇEVRE BİRİMİ BAĞLANTILARI
6.3. PLC PROGRAMLAMA EDİTÖRÜNÜN KURULUMU



TEMEL KAVRAMLAR

CPU, dijital ve analog port, giriş-çıkış ve besleme birimi, PLC, PLC ek donanımları

HAZIRLIK ÇALIŞMALARI

1. PLC ile ilgili bildikleriniz nelerdir? Bildiklerinizi arkadaşlarınızla paylaşınız.
2. PLC kullanım alanlarıyla ilgili neler biliyorsunuz?
3. PLC giriş ve çıkış elemanları neler olabilir?

6.1. YAPI VE ÇALIŞMA PRENSİBİNE GÖRE PLC SEÇİMİ

Girişine uygulanan sinyalleri algılayıp, yazılımıyla işleyerek istenen çalışma şartlarını gerçekleştirmek için çıkışlarını yönlendiren süreç kontrol cihazlarına **PLC (Programlanabilir Lojik Kontrolör)** denir (Görsel 6.1).



Görsel 6.1: PLC ve ek donanımları

PLC'ler endüstriyel ortamlarda (nemli, tozlu, gürültülü, titreşimli vb.) güvenli çalışmak amacıyla tasarlanmıştır. Klasik kumanda devrelerinin yaptığı kontrol işlemini yapabilir. Bu kumanda devrelerine göre daha az yer kaplar. Bilgisayar ile programlanabilir. Bu sayede program değiştirilerek tek bir elemanla birden fazla kumanda işlemi yapılabilir.

6.1.1. PLC'nin Yapı Özellikleri

PLC'ler yapı itibarıyla bilgisayarlara benzer. Mini bir bilgisayar olarak kabul edilebilir. CPU, hafıza, giriş-çıkış ve haberleşme birimlerinden oluşur. PLC'nin yapı tasarımı üretici firmalara göre farklılık gösterir.

CPU ve Bellek (Hafıza) Birimi

CPU; PLC üniteleri arasında iletişimi sağlayan, kontrol eden, girişten gelen verileri işleyerek çıkışa aktaran, oluşturulan programa göre işlemleri yürüten ve değerlendiren birimdir. PLC'nin genel özellikleri işlemciye göre belirlenir.

Hafıza alanı, PLC içindeki verilerin kalıcı veya geçici olarak saklandığı bölümdür. Program bellek, geçici bellek, kalıcı bellek alanı gibi bölümlere ayrılır. CPU ve bellek alanı PLC'nin içyapısında bulunur (Görsel 6.2).



Görsel 6.2: PLC'nin içyapısı

PLC Besleme

PLC beslemesi DC 24 V veya AC 230 V ile yapılabilir. Besleme için genellikle DC 24 V kullanılır.

Giriş uçları 0 V ve 24 V şeklinde gösterilir (Görsel 6.3:a).

Giriş (Input) Birimi

Giriş elemanlarından gelen elektriksel sinyalleri CPU'ya aktaran birimdir.

Giriş elemanları; buton, anahtar, paket şalter,

sınır anahtarı veya sensör olabilir. Bu elemanlardan alınan sinyaller, giriş biriminde CPU tarafından işlenebilecek hâle getirilir. Girişler genellikle **I** harfiyle isimlendirilir. PLC'lerde birden fazla giriş bulunduğu için sırasıyla I1, I2 … şeklinde isimlendirme yapılır (Görsel 6.3:b). İsimlendirme PLC markalarına göre değişiklik gösterebilir.

PLC'lerde dijital ve analog olmak üzere iki farklı giriş bulunur.

- **Dijital Girişler:** Dijital giriş sinyallerinin uygulandığı girişlerdir. Bazı dijital girişler yüksek hızlı sinyalleri algılayabilecek özelliktedir. Girişlerin yavaş veya hızlı algılanması yazılımla ayarlanır.
- **Analog Girişler:** Analog giriş sinyallerinin uygulandığı girişlerdir. Analog girişlerde 0-5 ve 0-10 V seviyelerindeki gerilim ya da 0-20 ve 4-20 mA seviyelerindeki akım sinyalleri kullanılır.

Çıkış (Output) Birimi

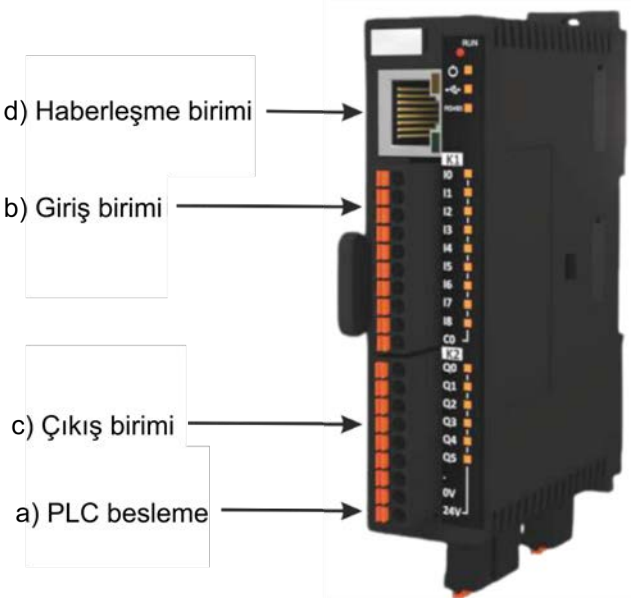
CPU tarafından işlenen bilgiyi, çıkış elemanlarının çalışabileceği gerilim seviyelerine dönüştüren birimdir. Çıkış elemanları; röle, kontaktör, sinyal lambası, sesli uyarı cihazları, selenoid valf veya motor olabilir. Çıkışlar genellikle **Q** harfiyle isimlendirilir. Aynı şekilde birden fazla çıkış olduğu için Q1, Q2 … şeklinde isimlendirme yapılır (Görsel 6.3:c). Çıkış isimlendirmeleri de markalara göre farklılık gösterir.

Dijital ve analog çıkış olmak üzere iki tip çıkış vardır. Aynı zamanda farklı çalışma şartları için dijital çıkışlar transistörlü ya da röleli olabilir.

- **Transistörlü Çıkışlar:** Anahtarlama hızları yüksektir. Bu sebeple devreye sık girip çıkan çalışma alanlarında tercih edilir. Mekanik aşınmaları olmadığından ömürleri uzundur ve 0,5 A gibi DC akım taşıma değerine sahiptir. Transistörlü çıkışlar NPN veya PNP özellikli olabilir.
- **Röleli Çıkışlar:** Anahtarlama hızları düşüktür. Çıkış röleleri 2 A gibi akım taşıma değerine sahiptir. Çıkışlarına DC ya da AC ile çalışan son kumanda elemanları bağlanabilir. Bu sebeple farklı gerilimlerde çalışan elemanları kontrol etmek kolaydır.
- **Analog Çıkışlar:** Analog çıkış sinyallerinin alındığı kısımdır. Analog çıkışlarda 0-10 V seviyelerindeki gerilim ya da 0-20 mA ve 4-20 mA seviyelerindeki akım sinyalleri kullanılır.

Haberleşme Birimi

Haberleşme birimi, PLC'lerle giriş-çıkış birimleri arasındaki ya da PLC ile bilgisayar arasındaki veri alışverişini sağlar. Bağlantı işlemi doğrudan yapılabileceği gibi bir network üzerinden de sağlanabilir. Haberleşme için RS 232, RS 485 ve Ethernet gibi ağ yapıları; haberleşme dili olarak Modbus, PROFIBUS, PROFINET gibi protokoller kullanılır (Görsel 6.3:d).



Görsel 6.3: PLC'nin genel yapısı

PLC'nin Ek Donanımları

PLC donanım özelliklerinin yetersiz kalması durumunda ek donanımlar kullanılır. Ek donanımlar genellikle giriş-çıkış sayısı ve haberleşme özelliklerini arttırma amaçlıdır. İhtiyaç duyulan donanım, üretici kataloglarına bakılarak seçilir.

6.1.2. PLC'nin Çalışma Prensibi

PLC'de programın yürütülmesi belirli işlemlerin periyodik çevrimiyle gerçekleştirilir (Görsel 6.4). PLC'nin girişten algılanan sinyalleri değerlendirip ara işlemlerden sonra tekrar girişe aktarmasına kadar geçen süreye **tarama süresi** ya da **saykıl** denir. Tüm işlemlerin gerçekleşmesi için geçen zamana da **tarama çevrimi** adı verilir. Tarama çevrimi, PLC'nin enerjisi kesilinceye ya da PLC stop konumuna alınıncaya kadar tekrar eder. Bir tarama çevriminin süresi PLC'nin çalışma hızına, kullanılan komutların niteliğine ve kontrol programının uzunluğuna göre değişir.



Görsel 6.4: PLC'nin tarama çevrimi

PLC'ye enerji verildiğinde PLC işletim sistemi programını çalıştırır. Dâhilî kontrollerin ardından hata yoksa ve PLC çalış (RUN) konumundaysa aşağıdaki sıralı işlemleri gerçekleştirir.

- Girişlere bağlı buton, sensör gibi elemanlardan alınan sinyaller okunarak belleğe kaydedilir.
- İkinci aşamada PLC işlemcisi bellekte bulunan verilere ve kullanıcı programındaki komutlara göre verileri işler. İşlem sonucunda oluşan yeni veriler belleğe yazılır.
- Üçüncü aşamada ise çıkış birimi, bellekten aldığı yeni verilere göre kendisine bağlı olan röle, kontaktör gibi elemanları çalıştırır. Böylece bir tarama çevrimi tamamlanır. Çevrim, PLC stop konumuna alınıncaya kadar devam eder.

6.1.3. PLC'nin Kullanım Alanları

PLC'ler otomasyon sistemlerinin her alanında kullanılmaktadır. Aydınlatma sistemleri, klimalar, havalandırma ve soğutma tesisleri, imalat, tarım ve tekstil, paketleme ve ambalaj tesisleri, otomobil endüstrisi PLC'nin kullanım alanlarından bazılarıdır. PLC ile kontrolü yapılacak sistemler büyüklük açısından farklılık gösterebilir. Sadece bir makine kontrolü yapılabileceği gibi bir fabrikanın komple kumandası da gerçekleştirilebilir. PLC'lerin genel uygulama alanları şöyledir:

- **Sıra Kontrolü:** PLC'lerin sıralı çalışma mantığıyla kullanıldığı ve röleli sistemlere en yakın uygulama alanıdır. Bağımsız makine ya da makine hatlarında, paketleme makinelerinde ve modern asansör denetim sistemlerinde kullanılmaktadır. PLC'lerin en çok kullanıldığı çalışma alanıdır.

- **Hareket Kontrolü:** Doğrusal ve döner hareket sistemlerinin kontrol edildiği çalışma alanıdır. Robotik sistemler bu alana en güzel örnektir.
- **Süreç Kontrolü:** PLC'nin hız, basınç, sıcaklık, ağırlık, debi gibi fiziksel parametreleri kontrol ettiği çalışma alanıdır. Buna tipik örnek olarak plastik enjeksiyon makineleri ve ısıtma fırınları verilebilir.
- **Veri Yönetimi:** PLC'ler sadece kontrol amacıyla değil, kontrol ettiği sistem hakkında veri toplayıcı olarak da kullanılabilir. Toplanan bu veri başka bir cihaza aktarılarak sistem içinde değerlendirilir. Bu özellik; kâğıt, metal, yiyecek işleme gibi birçok alanda kullanılır.

6.1.4. PLC Seçimi

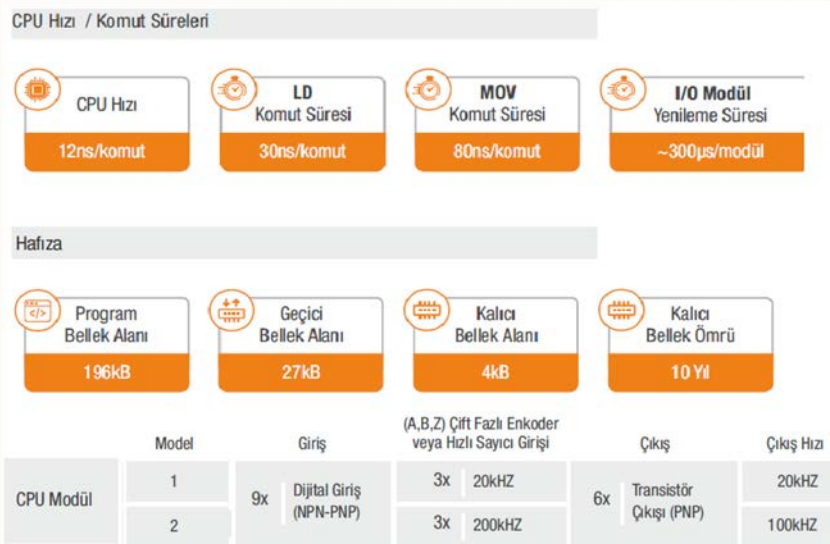
PLC, yapılacak işe uygun olmalıdır. Bu sebeple kullanılacak PLC'nin seçiminde aşağıda verilen yapı özelliklerine dikkat edilmelidir.

- **Besleme Gerilimi:** Öncelikle kullanılacak PLC besleme geriliminin tür (AC, DC) ve değeri (230 V ve 24 V) belirlenmelidir. 230 VAC beslemeli bir PLC doğrudan şebekeye bağlanabilir ancak 24 VDC beslemeli PLC için ayrı bir güç kaynağı gerekir.
- **Giriş Özellikleri:** PLC'nin sahip olduğu girişlerin tipi (analog ve dijital) ve giriş sayısı PLC seçiminde en önemli ölçütlerden birisidir. Bazı PLC'lerde analog giriş bulunmamaktadır. PLC'ye analog giriş elemanı bağlanacaksa ek donanım kullanma gerekliliği dikkate alınmalıdır. Aynı durum giriş sayısının yetersiz kalması durumunda da geçerlidir. Dijital PLC girişlerinin desteklediği besleme tipine (PNP, NPN) de dikkat edilmelidir.
- **Çıkış Özellikleri:** PLC çıkışlarında dikkat edilecek en önemli husus çıkış tipidir. Yüksek anahtarlama hızı istenen ve sık devreye girip çıkan çalışma alanlarında transistör çıkışlı PLC'ler tercih edilmelidir, ancak bu çıkışların 0,5 A'den fazla akım taşıyamayacağı unutulmamalıdır. Ayrıca bu çıkışların desteklediği çıkış tipine de (NPN veya PNP) dikkat edilmelidir.
- Yüksek akımlı (2 A) alıcıların kontrol edileceği durumlarda röle çıkışlı PLC'ler tercih edilebilir, ancak bu PLC'lerin dâhilî rölelerinin arızalanarak çıkışın iptal olması riski vardır. Bu sebeple transistör çıkışlı PLC kullanılıp çıkışına röle bağlanması daha iyi bir seçenektir.
- Aynı zamanda sistemde analog çıkış elemanları kullanılacaksa PLC'nin analog çıkışlara sahip olması gerekir. Analog çıkış yoksa ek donanım kullanılmalıdır.
- **Haberleşme Özellikleri:** PLC'nin kullanılacağı sistemde hangi haberleşme yöntem ve protokollerinin desteklendiği önceden tespit edilmelidir. PLC'de sisteme uygun haberleşme donanımlarının bulunup bulunmadığı kontrol edilmelidir.
- **İşlemci Hızı ve Bellek Kapasitesi:** PLC'nin işlemci hızı ve bellek kapasitesi, yapılan iş için yeterli olmalıdır. PLC'ler sürekli gelişmekte ve modelleri yenilenmektedir.
- **Genişleme İmkânı ve Diğer Sistemlerle Uyumluluğu:** Özellikle büyük otomasyon sistemlerinde kullanılacak PLC'lerin genişleme imkânı ve SCADA gibi sistemlerle uyumluluğu bulunmalıdır.
- **Kullanım Kolaylığı:** PLC'nin gerek donanım bağlantılarının kolay yapılması gerekse yazılım ve programlanmasının kolay olması istenen özelliklerdendir.
- **Servis Ağı ve Maliyeti:** PLC ile ilgili yaşanan problemleri çözecek servis ve hizmet desteğinin olması, olası arızalarda hızlı çözüm sağlayacak, emek ve zaman kaybını önleyecektir. Aynı zamanda uygun maliyet PLC tercihinde önemli rol oynayacaktır.

1. UYGULAMA

PLC KATALOG OKUMA

AMAÇ : PLC kataloglarını okuyabilmek.



Görsel 6.5: PLC katalog değerleri

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Bilgisayar	Giriş seviyesi	1 adet
PLC kataloğu	Farklı firmalara ait kataloglar	3 adet

İŞLEM BASAMAKLARI

1. Üreticilerinin bilgi sayfalarından PLC kataloglarını indiriniz.
2. Katalogları inceleyiniz.
3. Belirlenen üç ayrı PLC modelinin özelliklerini Tablo 6.1'e kaydediniz

Tablo 6.1: Katalog Değerlerine Göre PLC Özellikleri

PLC	Besleme	Giriş Sayıları		Çıkış Sayıları ve Çıkış Özellikleri				Haberleşme
	Besleme gerilimi	Dijital giriş	Analog giriş	Dijital çıkış sayısı	Analog çıkış sayısı	Çıkış özelliği (Transis-tör-Röle)	Çıkış tipi (NPN/PNP)	Haberleşme özelliği
1								
2								
3								

SORU

- PLC kataloglarının kullanım amacını açıklayınız.

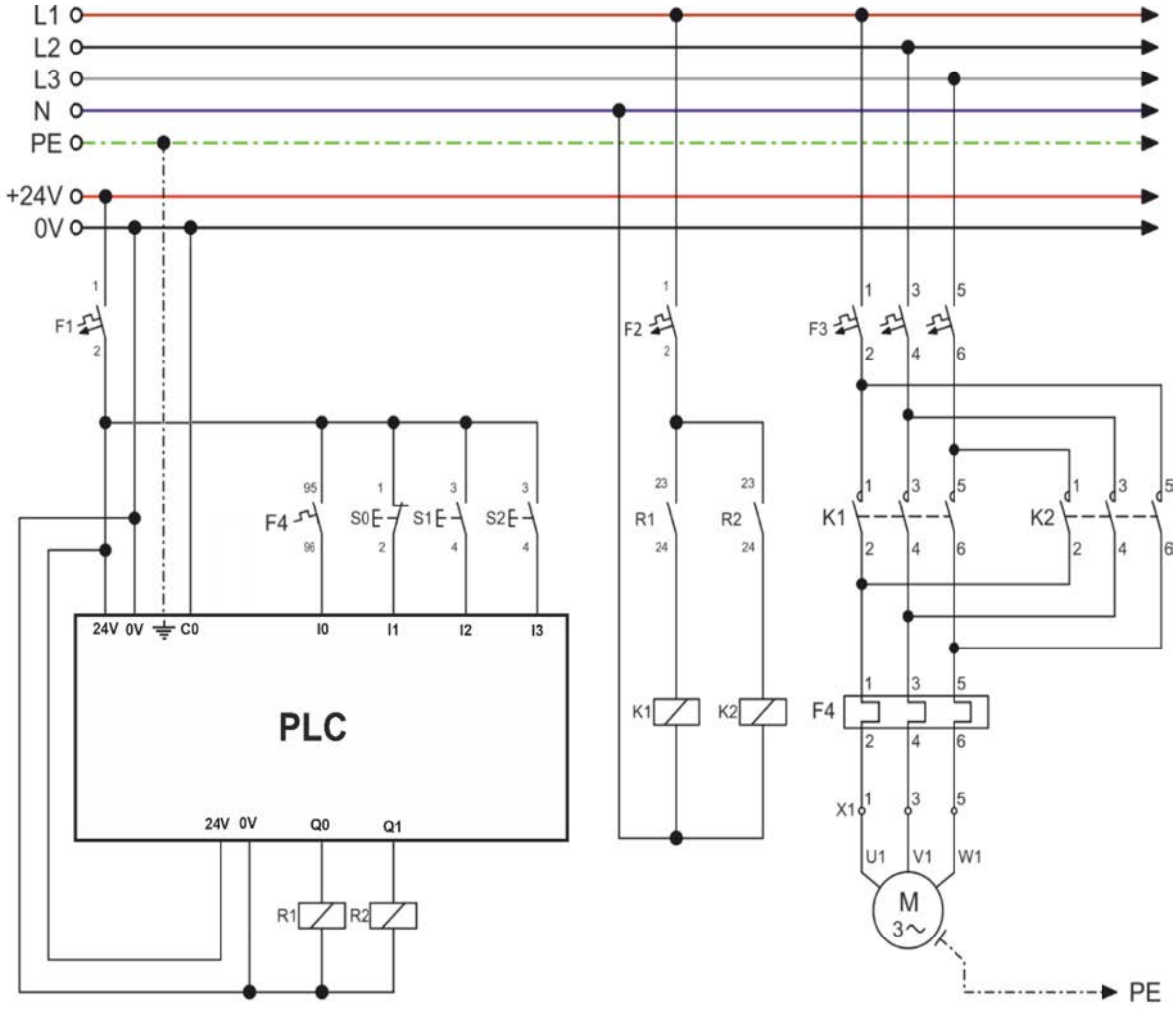
ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	PLC besleme değerlerini doğru kaydetti.	20	
Sınıfı :	2	PLC giriş değerlerini doğru kaydetti.	20	
Numarası :	3	PLC çıkış değerlerini doğru kaydetti.	20	
ÖĞRETMEN	4	PLC haberleşme değerlerini doğru kaydetti.	20	
Adı Soyadı :	5	Katalogları doğru yorumladı.	20	
İmza :		**TOPLAM PUAN**	100	

6.2. PLC'NİN GİRİŞ-ÇIKIŞ VE ÇEVRE BİRİMİ BAĞLANTILARI

PLC'lerde temel olarak besleme, haberleşme ve giriş-çıkış elemanlarının bağlantısı olmak üzere üç farklı bağlantı gerçekleştirilir. Bu bağlantılar PLC'nin özelliği dikkate alınarak yapılmalıdır. Bu sebeple PLC tipinin iyi bilinmesi gerekir. Bunun için de kullanım kılavuzu veya kataloglardan faydalanılır.

PLC bağlantılarında dikkat edilmesi gereken hususlar aşağıda verilmiştir.

- PLC besleme bağlantılarında, besleme gerilim tip (AC, DC) ve değerine (24 V, 230 V vb.) dikkat edilmeli ve PLC'ye uygun gerilim uygulanmalıdır.
- Buton, anahtar, sınır anahtarı gibi iki uçlu ve kutupsuz elemanların bir ucu güç kaynağının pozitif girişine, diğer ucu ise programda tanımlanan giriş adresine bağlanmalıdır.
- Sensör gibi kutuplu elemanlar bağlanırken pozitif ve negatif uçları ilgili besleme uçlarına, sinyal çıkış ucu ise programda tanımlanan giriş adresine bağlanmalıdır. Aynı zamanda sensör tipi, giriş bağlantı tipine uygun (PNP/NPN) olmalıdır.
- Genel olarak giriş sinyali pozitif olmakla birlikte girişe negatif sinyal de uygulanabilmektedir. Bu durumda kutupsuz elemanların bir ucu güç kaynağının negatif girişine, diğer ucu ise programda tanımlı giriş adresine bağlanmalıdır.
- PLC çıkış akım değerine dikkat edilmeli ve çıkışa uygun akımlı alıcı bağlanmalıdır. Çıkış akımı üzerinde akım çekebilecek elemanlar röle üzerinden sürülmelidir.
- Eleman bağlantıları yapılırken giriş-çıkışların dijital ve analog özelliğine dikkat edilmelidir.
- Transistör çıkışlı PLC çıkışlarına gerektiğinde röle bağlanmalı ve alıcılar bu röleler üzerinden sürülmelidir (Görsel 6.6).
- Haberleşme bağlantıları için uygun kablo kullanılmalıdır.

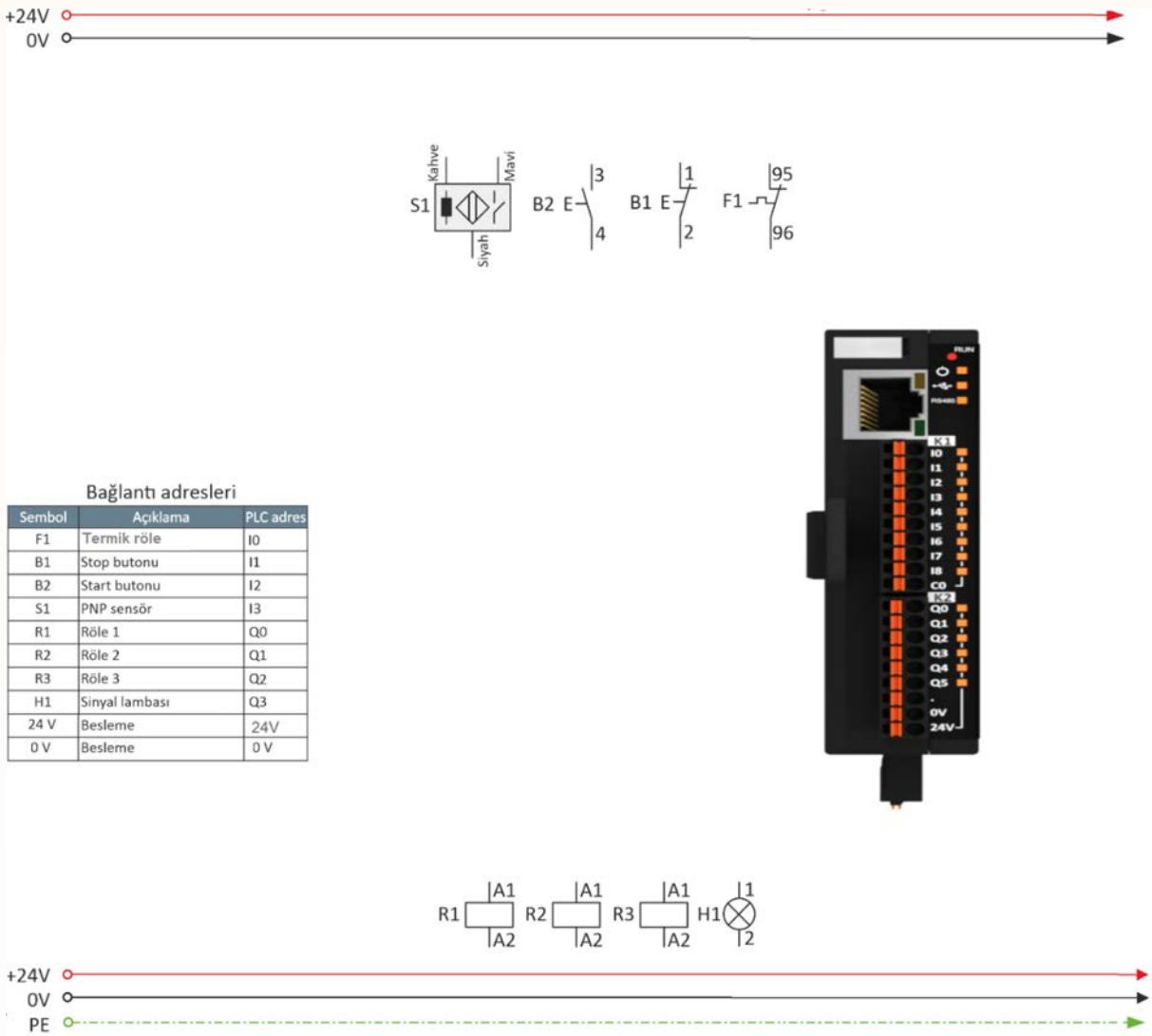


Görsel 6.6: PLC'nin giriş-çıkış ve çevre birimi bağlantıları

2. UYGULAMA

TRANSİSTÖR ÇIKIŞLI PLC'NİN DİJİTAL GİRİŞ-ÇIKIŞ BAĞLANTILARI

AMAÇ: **Transistör çıkışlı PLC'nin dijital giriş-çıkış bağlantılarını yapmak.**



Bağlantı adresleri

Sembol	Açıklama	PLC adres
F1	Termik röle	I0
B1	Stop butonu	I1
B2	Start butonu	I2
S1	PNP sensör	I3
R1	Röle 1	Q0
R2	Röle 2	Q1
R3	Röle 3	Q2
H1	Sinyal lambası	Q3
24 V	Besleme	24V
0 V	Besleme	0 V

Görsel 6.7: Transistör çıkışlı PLC'nin giriş-çıkış bağlantı şeması

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
PLC	Transistör çıkışlı	1 adet
Güç kaynağı	24 VDC	1 adet
F1	Termik kontağı 95-96	1 adet
B1	Stop butonu 1 NC	1 adet
B2	Start butonu 1 NO	1 adet
S1	S1 PNP manyetik yaklaşım sensörü	1 adet
R1	Röle 24 VDC bobin	1 adet
R2	Röle 24 VDC bobin	1 adet
R3	Röle 24 VDC bobin	1 adet
H1	Sinyal lambası 24 VDC	1 adet
El aletleri	Yan keski, tornavida, avometre, kargaburnu	-

İŞLEM BASAMAKLARI

1. Kullanılacak elemanlar için üreticilerinin bilgi sayfasından gerekli bağlantı bilgilerini ve teknik bilgileri alınız.
2. Görsel 6.7'de verilen bağlantı adresleri tablosunu kullanarak devre şeması bağlantılarını çiziniz.
3. Çizdiğiniz şemaya göre devre elemanlarının bağlantısını yapınız.
4. Bağlantıların sağlamlığını ve doğruluğunu öğretmeniniz denetiminde kontrol ediniz.
5. Devreyi dikkatlice sökerek elemanları öğretmeninize teslim ediniz.

SORULAR

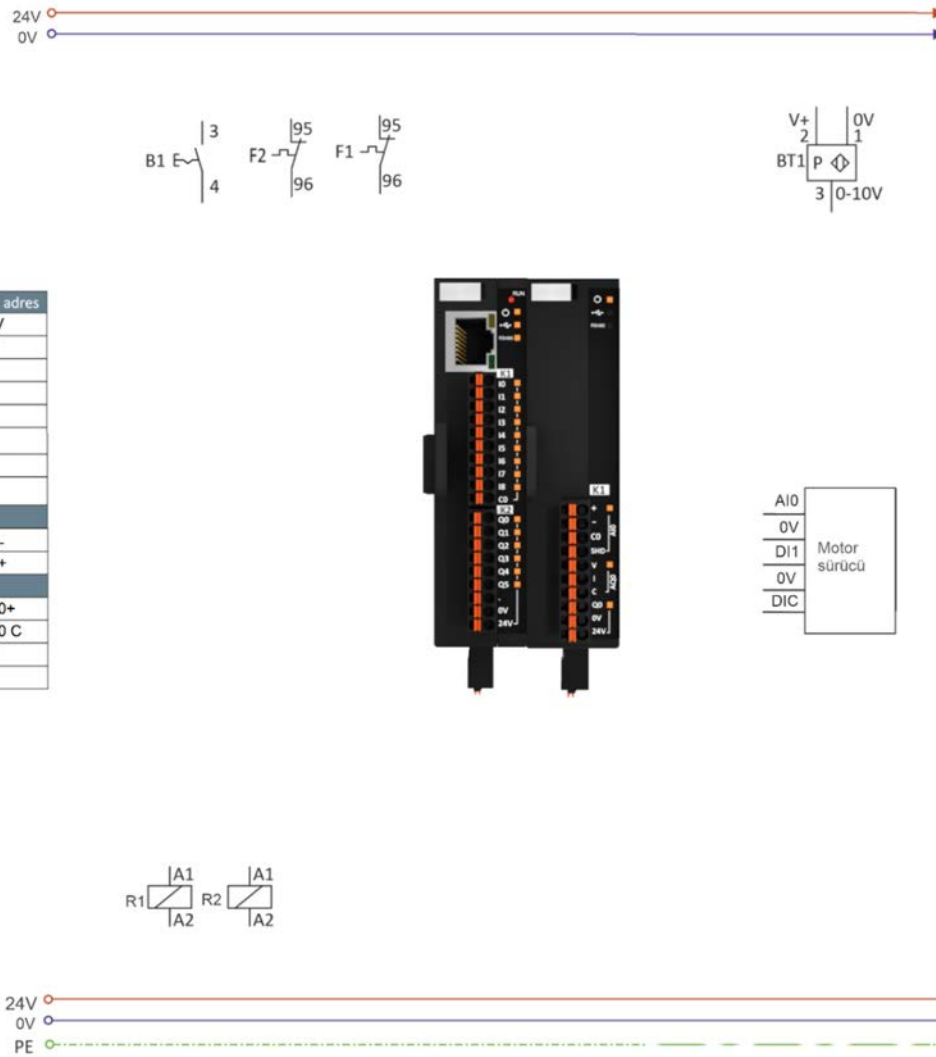
1. Transistör ve röle çıkışlı PLC arasındaki farklar nelerdir? Yazınız
2. Kullandığınız PLC'nin çıkış transistörleri ne tiptir? Belirtiniz.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Devre şemasını çizdi.	20	
Sınıfı :	2	Devrenin montajını gerçekleştirdi.	20	
Numarası :	3	Bağlantıyı doğru ve sağlam bir şekilde gerçekleştirdi.	20	
ÖĞRETMEN	4	Çalışma ortamını temizleyip düzenledi.	20	
Adı Soyadı :	5	İşi, verilen süre içerisinde tamamladı.	20	
İmza :		**TOPLAM PUAN**	100	

3. UYGULAMA

PLC'NİN ANALOG GİRİŞ-ÇIKIŞ BAĞLANTILARI

AMAÇ: PLC'nin analog giriş-çıkış bağlantılarını yapmak.



Bağlantı adresleri

Sembol	Açıklama	PLC adres
24V	Pozitif besleme	24V
0V	Negatif besleme	0V
F1	Termik kontağı 1	I0
F2	Termik kontağı 2	I1
B1	Start butonu	I2
R1	Röle 1	Q0
R2	Röle 2	Q1
0V	Ortak uç	C0
Bt1	Basınç transmitteri	
0V	Ortak uç	AI0-
0-10 V	Analog çıkış	AI0+
	Sürücü	
AI0	Analog çıkış	AQ0+
0V	0V çıkış	AQ0 C
DI1	Dijital giriş	Q0
0V	Ortak uç	0V

Görsel 6.8: PLC'nin analog giriş-çıkış bağlantı şeması

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
PLC	Transistör çıkışlı	1 adet
Ek donanım	Analog çıkış modülü	1 adet
Güç kaynağı	24 VDC	1 adet
F1	Termik kontağı 95-96	1 adet
F2	Termik kontağı 95-96	1 adet
B1	Start butonu 1 NO	1 adet
BT1	Basınç transmiteri 0-10 bar 0-10 VDC	1 adet
R1	Röle1 24 VDC bobin	1 adet
R2	Röle2 24 VDC bobin	1 adet
Sürücü	Motor sürücüsü	1 adet
El aletleri	Yan keski, tornavida, avometre, kargaburnu	-

İŞLEM BASAMAKLARI

1. Kullanılacak elemanlar için üreticilerinin bilgi sayfasından gerekli bağlantı bilgilerini ve teknik bilgileri alınız.
2. Görsel 6.8'de verilen bağlantı adresleri tablosunu kullanarak devre şeması bağlantılarını çiziniz.
3. Çizdiğiniz şemaya göre devre elemanlarının bağlantısını yapınız.
4. Bağlantıların sağlamlığını ve doğruluğunu öğretmeniniz denetiminde kontrol ediniz.
5. Devreyi dikkatlice sökerek elemanları teslim ediniz.

SORU

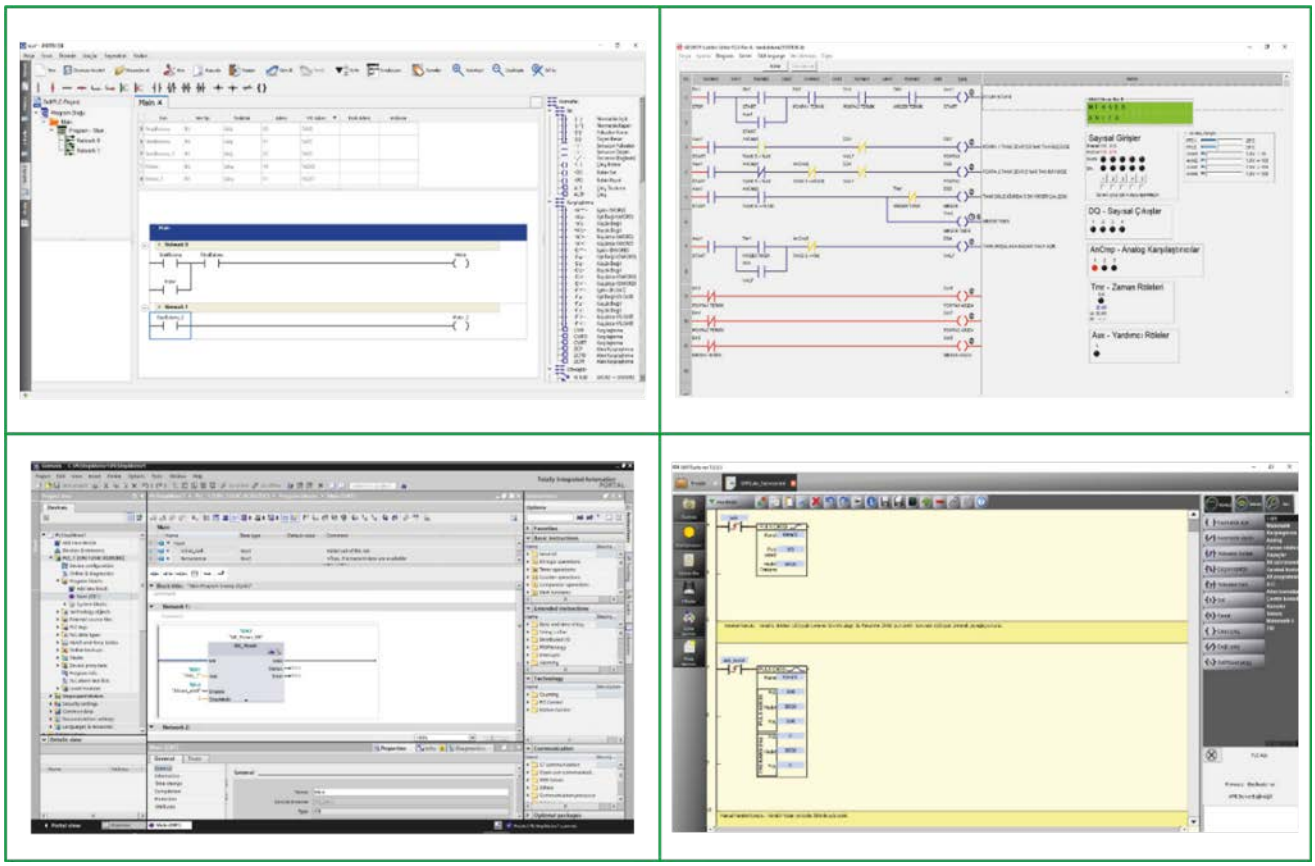
- Ek donanımların kullanım amacını açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Devre şemasını çizdi.	20	
Sınıfı :	2	Devrenin montajını gerçekleştirdi.	20	
Numarası :	3	Bağlantıyı doğru ve sağlam bir şekilde gerçekleştirdi.	20	
ÖĞRETMEN	4	Çalışma ortamını temizleyip düzenledi.	20	
Adı Soyadı :	5	İşi, verilen süre içerisinde tamamladı.	20	
İmza :		**TOPLAM PUAN**	100	

6.3 PLC PROGRAMLAMA EDİTÖRÜNÜN KURULUMU

PLC cihazlarının programlanması için kullanılan yazılımlara **programlama editörü** denir.

Programlama editörü; programın yazılması, bilgisayara kaydedilmesi, PLC için derlenmesi, PLC'ye yüklenmesi ve programın çalışmasının incelenmesi işlemlerini yapar. Bu editörler, üretici firmaya özel olmakla birlikte benzerlik gösterir (Görsel 6.9). PLC programlamada genellikle IEC 61131-3 standardına uygun PLC programlama editörleri kullanılır.



Görsel 6.9: Çeşitli PLC editörleri

PLC'lere ait yazılımlar farklı olmakla birlikte kurulumları hemen hemen aynıdır. Öncelikle yazılımın kurulacağı bilgisayarın istenen özellikleri karşılayıp karşılamadığı kontrol edilmelidir. Daha sonra ilgili yazılım çalıştırılarak kurulum başlatılmalıdır. Yazılım yönergeleri kurulum sırasında kullanıcıyı yönlendirecektir. Bu yönergeler takip edilerek PLC'ye ait editör yazılımı bilgisayara kurulur.

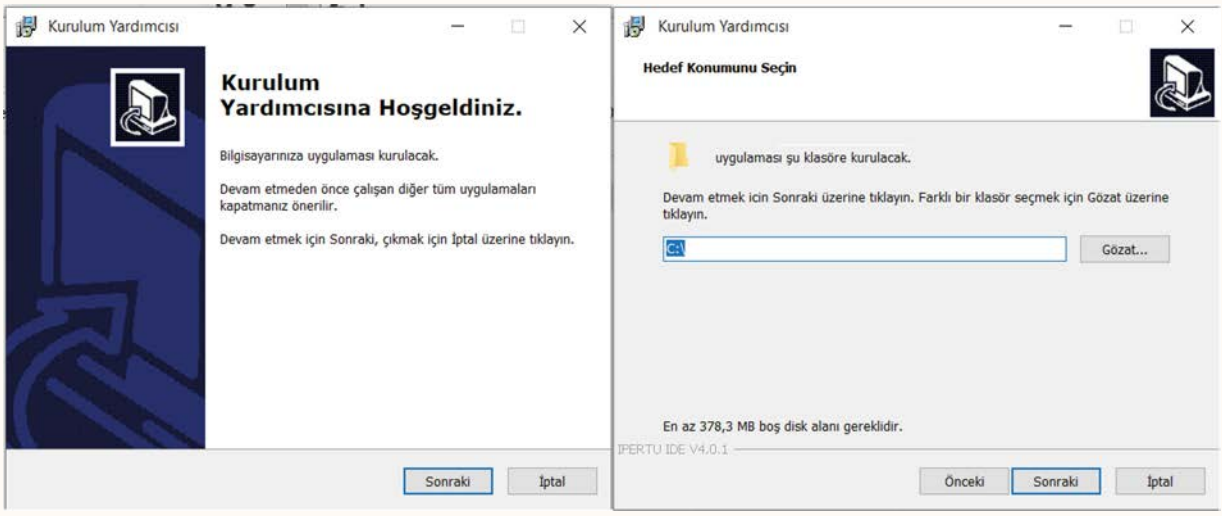
Yazılım menüleri de birbirine yakın seçeneklere sahiptir. Programlama için yeni bir proje oluşturulur ve cihaz yapılandırılır. PLC modeli seçilir. Editör ana penceresinde değişken listesi ve sırasıyla networkler oluşturulur. Network üzerinde kütüphane objeleri varsa yapılandırılır.

Program hazırlandıktan sonra programın bilgisayara aktarılması için gerekli haberleşme ayarları yapılır. Uygun kablo kullanılarak hazırlanan program PLC'ye yüklenir. Yazılan programın gereksinimleri karşılayıp karşılamadığı gerçek zamanlı olarak veya PLC simülatöründen test edilir.

4. UYGULAMA

PLC PROGRAMLAMA EDİTÖRÜNÜN KURULUMU

AMAÇ : PLC programlama editörünün kurulumunu yapmak.



Görsel 6.10: PLC programlama editörünün kurulumu

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Bilgisayar	PLC editörü yazılımı gereksinimlerine uygun	1 adet
PLC editörü yazılımı	Bilgisayar işletim sistemine uygun	1 adet

İŞLEM BASAMAKLARI

1. Programlama editörünün kurulum dosyasına çift tıklayarak kurulumu başlatınız.
2. Kurulum yapılacak klasörü seçiniz.
3. İlgili seçeneğe tıklayarak kurulumun tamamlanmasını bekleyiniz.
4. İlgili seçeneğe tıklayarak kurulumu tamamlayınız.
5. İlgili simgeye tıklayarak editör yazılımını çalıştırınız.

SORU

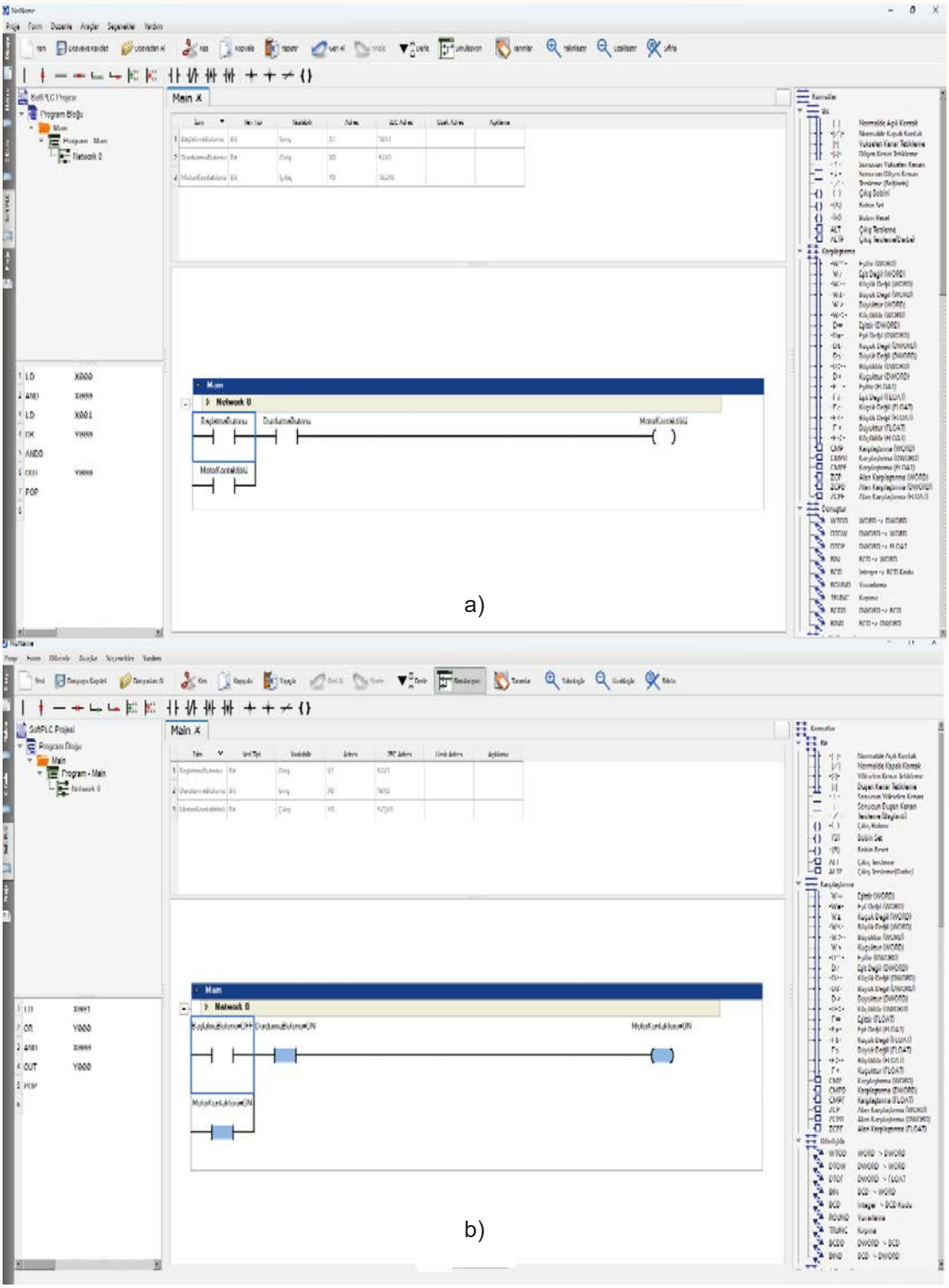
- Editör kurulumu sırasında dikkat edilmesi gereken hususları yazınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Editör yazılımını doğru klasöre kurdu.	20	
Sınıfı :	2	Yazılım kurulumunda işlem basamaklarına uydu.	20	
Numarası :	3	Yazılımı doğru çalıştırdı.	20	
ÖĞRETMEN	4	Çalışma ortamını temiz ve düzenli bıraktı.	20	
Adı Soyadı :	5	İşi, verilen süre içinde tamamlandı.	20	
İmza :		**TOPLAM PUAN**	100	

5. UYGULAMA

PLC PROGRAMLAMA EDİTÖRÜNÜN UYGULAMASI

AMAÇ: PLC programlama editörünün uygulamasını yapmak



Görsel 6.11: PLC programlama editörünün uygulaması

ETİKET TABLOSU					
Eleman İsmi	**Veri Tipi**	**Bağlantı Tipi**	**Bağlantı Adresi**	**Adres**	**IEC Adresi**
Stop butonu	Bit	Giriş	I0	X0	%IX0
Start butonu	Bit	Giriş	I1	X1	%IX1
Kontaktör	Bit	Çıkış	Q0	Y0	%QX0

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Bilgisayar	PLC editörü yazılımı gereksinimlerine uygun	1 adet
PLC editörü yazılımı	Bilgisayar işletim sistemine uygun	1 adet

İŞLEM BASAMAKLARI

1. Bilgisayar ve programı çalışma kurallarına uygun olarak açınız.
2. PLC programlama editörünü çalıştırınız.
3. Yeni bir proje oluşturunuz.
4. Çalışma alanına gerekli elemanları alarak Görsel 6.11:a'da verilen networkü oluşturunuz.
5. Network üzerinde gerekli düzenlemeleri yapınız.
6. Simülasyonu başlatınız.
7. Devrenin çalışması anında oluşan değişimleri gözlemleyiniz (Görsel 6.11:b).
8. Projeye isim vererek kaydediniz.
9. Program ve bilgisayarı çalışma kurallarına uygun olarak kapatınız.

SORU

- Programlama editörünün kullanım amacını açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Editörde yeni bir proje oluşturdu.	20	
Sınıfı :	2	Editör çalışma alanına semboller aldı.	20	
Numarası :	3	Semboller üzerinde düzenleme işlemlerini yaptı.	20	
ÖĞRETMEN	4	Kayıtlı bir projeyi açtı.	20	
Adı Soyadı :	5	İşi, verilen süre içinde yaptı.	20	
İmza :		**TOPLAM PUAN**	100	

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki cümlelerin başındaki boşluğa cümleler doğru ise (D), yanlış ise (Y) yazınız.

1. (…) PLC'ler endüstriyel ortamlarda güvenli çalışmak amacıyla tasarlanmıştır.
2. (…) PLC'ler klasik kumanda sistemlerine göre daha fazla yer kaplar.
3. (…) PLC'ler bilgisayar ile programlanabilir.
4. (…) PLC girişleri genellikle Q harfiyle isimlendirilir.
5. (…) Analog girişlerde 0-5 ve 0-10 V seviyelerindeki gerilim ya da 0-20 ve 4-20 mA seviyelerindeki akım sinyalleri kullanılır.

B) Aşağıdaki cümlelerde boş bırakılan yerlere uygun sözcüğü yazınız.

6. PLC besleme gerilimleri genellikle DC …… V'tur.
7. PLC genel olarak CPU, bellek birimi, giriş-çıkış birimi ve …… biriminden oluşur.
8. Transistör çıkışlı PLC'lerin anahtarlama hızları ……
9. PLC'nin hız, basınç ve sıcaklık gibi fiziksel parametreleri kontrolüne …… kontrolü denir.
10. PLC donanım özelliklerinin yetersiz kalması durumunda …… donanımlar kullanılır.

C) Aşağıdaki soruların doğru seçeneğini işaretleyiniz.

11. Aşağıdakilerden hangisi PLC cihazlarının programlanması için kullanılan yazılımdır?

A) Giriş ve çıkış editörü B) İletişim protokol yazılımı C) CPU ve bellek birimi yazılımı
D) Programlama editörü E) Süreç kontrol yazılımı

12. Aşağıdakilerden hangisi PLC bağlantılarında dikkat edilmesi gereken hususlar konusunda yanlış bilgi içerir?

A) PLC besleme bağlantılarında, besleme gerilim tip ve değerine dikkat edilmelidir.
B) Sensör gibi kutuplu elemanlar bağlanırken pozitif ve negatif uçları ilgili besleme uçlarına, sinyal çıkış ucu ise pozitif uca bağlanır.
C) PLC çıkış akım değerine dikkat edilmeli ve çıkışa uygun akımlı alıcı bağlanmalıdır.
D) Eleman bağlantıları yapılırken giriş-çıkışların dijital ve analog özelliğine dikkat edilmelidir.
E) Yüksek akımlı alıcılar sürülmek istendiğinde dâhilî rölesiz PLC çıkışlarına röle bağlanmalı ve alıcılar bu röleler üzerinden sürülmelidir.

13. Aşağıdakilerden hangisi giriş elemanı değildir?

A) Anahtar B) Buton C) Motor
D) Sensör E) Sınır anahtarı

14. Aşağıdakilerden hangisi PLC üniteleri arasında iletişimi sağlayan, kontrol eden, girişten gelen verileri işleyerek çıkışa aktaran, oluşturulan programa göre işlemleri yürüten ve değerlendiren birimdir?

A) CPU B) Bellek C) Çıkış birimi
D) Giriş birimi E) Röle

15. Aşağıdakilerden hangisi PLC'nin girişten algılanan sinyalleri değerlendirip ara işlemlerden sonra tekrar çıkışa aktarmasına kadar geçen süreye verilen isimdir?

A) Çevrim süresi B) İşlem süresi C) Periyot
D) Sinyal süresi E) Tarama süresi

7. ÖĞRENME BİRİMİ

PLC PROGRAMLAMA UYGULAMALARI

KONULAR

7.1. TEMEL SEVİYE PLC KOMUTLARI KULLANILARAK PROGRAMLAMA YAPMA
7.2. DOKUNMATİK EKRAN (HMI) İLE UYGULAMA YAPMA
7.3. KONTAKTÖR VE İNVERTÖR KULLANILARAK PLC İLE ASENKRON MOTOR KONTROLÜ
7.4. PLC İLE SERVO MOTOR KONTROLÜ











TEMEL KAVRAMLAR

İnvertör, dokunmatik panel, kontrol paneli, HMI, ladder diyagramı, network, parametre, PLC programlama, programlama editörü, program komutları, simülasyon, servo motor, sürücü

HAZIRLIK ÇALIŞMALARI

1. PLC ile mikroişlemciler arasındaki temel farkları sınıfta arkadaşlarınızla değerlendiriniz.
2. PLC'lerin endüstriyel sistemlerdeki önemi hakkında düşünceleriniz nelerdir?

7.1. TEMEL SEVİYE PLC KOMUTLARI KULLANILARAK PROGRAMLAMA YAPMA

Kumanda sisteminde sözel / sayısal kuralların uygun dil kullanılarak PLC'ye aktarımı için düzenlenmesi işlemine **programlama** denir. Programlama, PLC'nin isteğe göre çalışmasını sağlar. PLC'ler arasında çalışma birliği sağlamak için IEC 61131-3 programlama standardı kabul edilmiştir.

7.1.1. PLC Programlama Mantığı ve Yazılım Dilleri

IEC 61131-3 standardına göre beş adet PLC programlama dili mevcuttur. PLC programı yazılırken ihtiyaca göre bu dillerden bir veya birkaç tanesi kullanılabilir.

FBD (Fonksiyon Blok Diyagramı) Programlama Dili: Lojik kapılar ve blok diyagram temeli üzerine kurulan grafik tabanlı bir programlama dilidir. Bu programlama dilinde her fonksiyonun bir sembolü bulunur. Sembolün sol tarafından giriş sinyalleri, sağ tarafındansa çıkış sinyalleri tanımlanır. Programcı, uygulamasında sık kullandığı işlemleri fonksiyon hâline getirebilir. Programlama editörlerinde sık kullanılan zamanlama, sayma gibi işlemleri hazır fonksiyon olarak sunar.

SFC (Sıralı Fonksiyon Dizisi) Programlama Dili: Yüksek seviyeli ve grafik tabanlı bir dildir. Program yapısı akış diyagramlarına benzer. Sıralı işlem gerektiren uygulamalarda kolaylık sağlar.

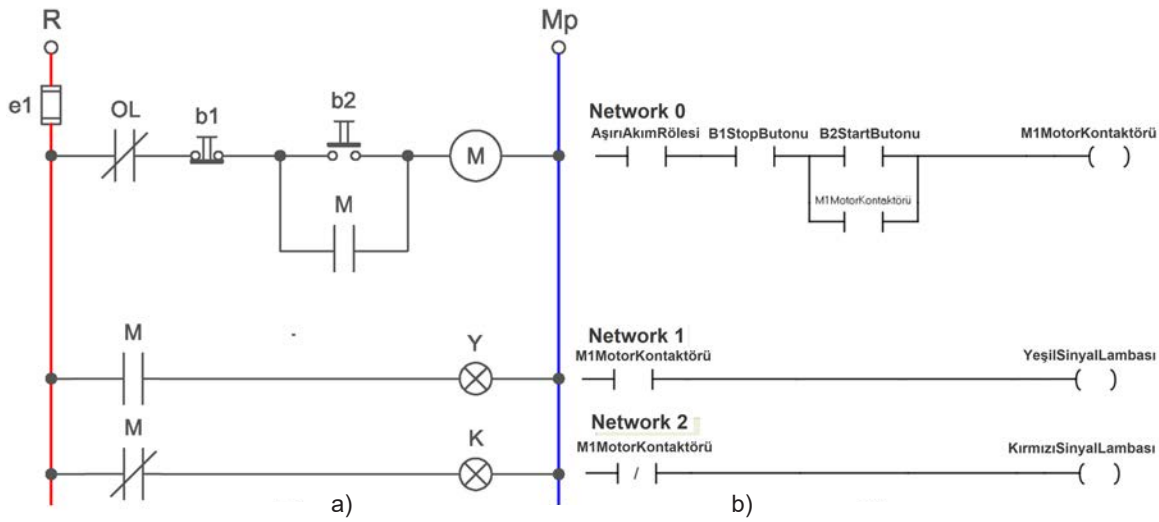
ST (Yapısal Metin) Programlama Dili: Söz dizilimini temel alan metin tabanlı bir programlama dilidir. Matematiksel işlemler ve karşılaştırma işlemleri daha kolay gerçekleştirilir. Ayrıca kapladığı alan, görsel programlama dillerine göre daha azdır. Fonksiyon bloklarını programlamak için uygundur.

IL (Komut Dizisi) Programlama Dili: Makine diline benzer yapıda metin tabanlı bir dildir. Az bellek kullanır ve hızlı çalışır. Ancak programlama işleminin zorluğundan dolayı kullanımı azalmıştır.

LD/LAD (Ladder Diyagramı) Programlama Dili: ANSI standardı kumanda devre şema yapısına benzeyen, grafik tabanlı programlama dilidir. Yatay çizgilerin üzerine lojik giriş-çıkış elemanları eklenerek oluşturulur. PLC, soldan sağa ve yukarıdan aşağıya tarama yaparak program içerisinde belirlenen döngüleri çalıştırır. Devrenin çalışması ve takibi kolay olduğundan en çok kullanılan dillerdendir.

7.1.2. Klasik Kumanda Devrelerinin Ladder Diyagramına Dönüşümü

Klasik kumanda devrelerinin ladder diyagramına dönüşümü Görsel 7.1'de verilmiştir. Diyagramdaki yatay çizgilerin her birine **network** denir. Networklerin üzerine kontak, kontaktör gibi mekanik elemanların lojik karşılıkları yerleştirilir ve her işlem için basamak oluşturulur.



Görsel 7.1: ANSI kumanda devresi ve ladder diyagramı

Ladder programı ile kumanda devresinin çalışması arasında farklar vardır. Bu farklar şunlardır:

- İşlem süresi, ladder programlamada birkaç ms iken kumanda devresinde röle ve kontaktörlerin çekme-bırakma süreleri (2-20 ms) ile röle sayısına bağlı olarak daha uzundur.
- Bir çıkış ladder bir kez tanımlanır. Girişler ise kumanda devrelerinde kontak sayısıyla sınırlı iken ladder programlamada program kapasitesiyle belirlenir ve oldukça fazladır.

- Ladder programlamada giriş açık kontağı olduğu gibi işlenirken kapalı kontak ise girişin tersinin alınacağını ifade eder.
- Ladder programlamada girişler istendiği gibi işlenebilir. Durdurma ve güvenlik kontakları içinse kapalı kontak kullanılır. Bu durum güvenlik gereği tercih edilir.

7.1.3. PLC Programlamada Hafıza Alanları ve Veri Tipleri

PLC'lerde farklı kullanım amaçları için ayrılmış hafıza alanları bulunur. PLC verileri bu bellek alanlarında saklanır. En küçük veri birimi **bit**'tir. 1 bit, 1 veya 0 bilgisine karşılık gelir. Bitlerin bir araya gelmesiyle diğer veriler oluşturulur. PLC veri tipleri, hafıza elemanının boyutunu ve hafızada ne kadar yer kapladığını belirler. IEC 61131-3'e göre veri tipleri Tablo 7.1'de verilmiştir.

Tablo 7.1: IEC 61131-3 Standardı Veri Tipleri

Veri	Veri Tipi	Veri Özelliği	Veri Uzunluğu	Sayı Aralıkları
Bit Tipi Dizeler (Açık ve Kapalı Değerler)	BOOL	1 bit	1 bit = {0,1}	1: True 0: False
	BYTE	8 bit	1 bayt = 8 bit	0 → 255
	WORD	16 bit	1 word=16 bit=2 bayt	0 → 65535
	DWORD	32 bit	1 dword=2 word=4 bayt=32 bit	0 → 4294967295
	LWORD	64 bit	1 lword=2 dword=4 word=64 bit	
İşaretli ve İşaretsiz Tam Sayılar	SINT	İşaretli kısa tam sayı	8 bit (1 bayt)	-128 → +127
	USINT	İşaretsiz kısa tam sayı	8 bit (1 bayt)	0 → 255
	INT	İşaretli tam sayı	16 bit (2 bayt)	-32768 → +32767
	UINT	İşaretsiz tam sayı	16 bit (2 bayt)	0 → 65535
	DINT	İşaretli çift tam sayı	32 bit (4 bayt)	-2147483648 → +2147483647
	LINT	İşaretli uzun çift tam sayı	64 bit (8 bayt)	
	UDINT	İşaretsiz çift tam sayı	32 bit (4 bayt)	0 → 4294967295
	ULINT	İşaretsiz uzun çift tam sayı	64 bit (8 bayt)	
Reel (Kayan Noktalı) Sayılar	REAL	Virgüllü gerçel sayı	32 bit (4 bayt)	-3.402823e+38 → -1.175495e-38
	LREAL	Virgüllü uzun gerçel sayılar	64 bit (8 bayt)	
Süreler, Tarihler, Günler ve Saatler	TIME	Ön işaretli kısa saat	32 bit (4 bayt)	T#5m90s15ms T#5m90s15m-s542us15ns (ns)
	LTIME	Ön işaretli uzun saat	64 bit (8 bayt)	
	DATE	Takvim / tarih (kısa format)	16 bit (2 bayt)	D#2021-12-13 (13.12.2021)
	LDATE	Takvim / tarih (uzun format)	32 bit (4 bayt)	
	TIME_OF_DAY (TOD)	Saat / zaman (kısa format)	32 bit (4 bayt)	TOD#00:00:00. 00TOD#23:59:59.99
	TIME_OF_DAY(LTOD)	Saat / zaman (uzun format)	64 bit (8 bayt)	
	DATE_AND_TIME (DTL)	Tarih ve saat [kısa format (ms)]	32 bit (4 bayt)	DTL#2021-12-13-11:50:45.000 (13.12.2021, 11.50.45)
	DATE_AND_TIME (LDTL)	Tarih ve saat [uzun format (ns)]	64 bit (8 bayt)	

Tablo 7.1: IEC 61131-3 Standardı Veri Tipleri

Veri	Veri Tipi	Veri Özelliği	Veri Uzunluğu	Sayı Aralıkları
Karakter Dizileri ve Birleştirilmiş Datalar	CHAR	Tek baytlık karakter dizisi	8 bit (1 bayt)	(- ? - / - A - %
	WCHAR	Çift baytlık karakter dizisi	16 bit (2 bayt)	() – {} - ||
	STRING	Değişken uzunlukta tek baytlık karakter dizisi	8 bit (254+2) değişken	ASCII kodlu işaretler dizisi 'Otomasyon()'
	WSTRING	Değişken uzunlukta çift baytlık karakter dizisi	16 bit (2 bayt)	ASCII kodlu işaretler dizileri
	ARRAY	Aynı veri tipleri ile oluşturulmuş alanlardır.	Değişken	
	STRUCT	Değişken veri tipleri ile oluşturulmuş alanlardır.	Değişken	

7.1.4. PLC Programlamada Giriş ve Çıkış Adreslemeleri

PLC giriş-çıkışlarının bağlantı ve programlamada kullanılmak üzere isimlendirilmesine **giriş-çıkış adreslemesi** denir. Üretici firmalar giriş-çıkış adreslemeleri için farklı harf ve sayı biçimleri kullanır. Genellikle bağlantı adreslemesinde girişler için **I**, çıkışlar içinse **Q** harfi tercih edilir.

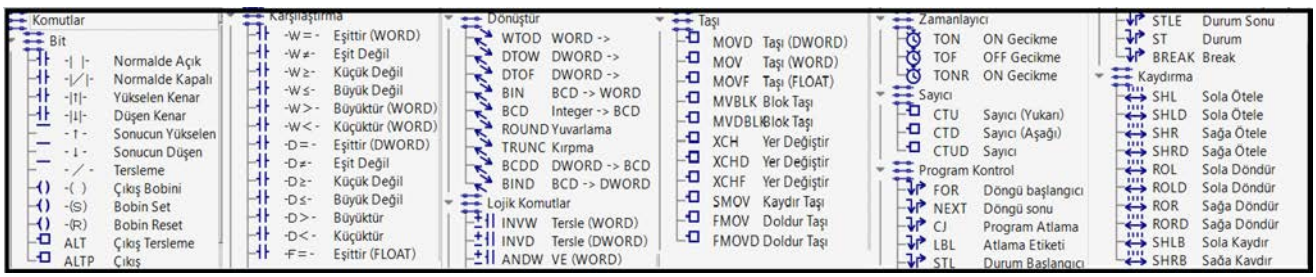
Program adreslemesinde IEC formatı Tablo 7.2'de görüldüğü gibidir. Bazı ülkelerde girişler **X**, çıkışlar **Y** harfiyle gösterilir. Herhangi bir hafıza alanındaki veriye erişmek için veri adresinin kullanılması gerekir.

Tablo 7.2: PLC Giriş ve Çıkış Adreslemeleri

Eleman İsmi	Veri Tipi	Bağlantı Tipi	Bağlantı Adresi	Adres	IEC Adresi
Stop butonu	Bit	Giriş	I0	X0	%IX0
Start butonu	Bit	Giriş	I1	X1	%IX1
Kontaktör	Bit	Çıkış	Q0	Y0	%QX0

7.1.5. Temel Programlama Komutları

PLC ile dijital veya analog işlemler yapmak mümkündür. Her iki işlem için de programlama editörü içinde verilen komutlar kullanılır. PLC programlama komutları, markalara göre değişiklik gösterse de benzer yapıdadır. Komutlar; kullanım kolaylığı açısından bit işlem komutları, karşılaştırma komutları, dönüştürme komutları, lojik komutlar, hesaplama komutları, zamanlayıcı, sayıcı, program kontrol ve kaydırma komutları gibi belirli gruplara ayrılarak verilir (Görsel 7.2).



Görsel 7.2: Genel komut grupları

7.1.5.1. Bit Lojik Komutları

LD, AND ve OR Komutları: LD, her bir satırın başında kullanılması gereken bir yükleme komutudur. Bu işlemden sonra AND ya da OR işlemi yapılır. Kontaklar paralel bağlandığında OR (VEYA); seri bağlandığında AND (VE) komutu ile ifade edilir.

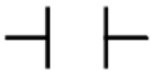
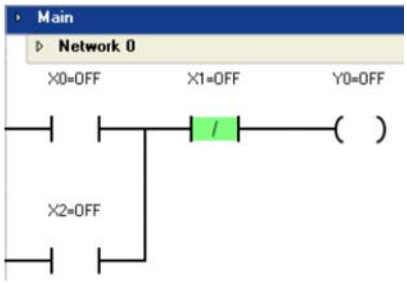
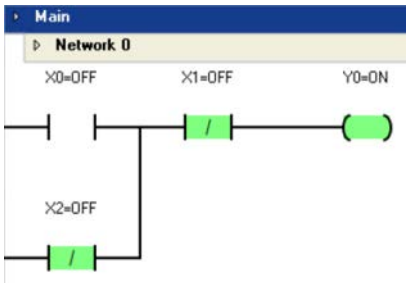
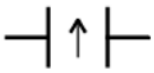
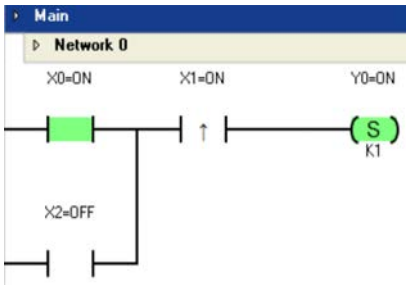
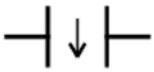
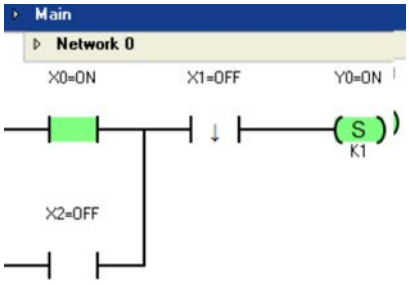
Normalde açık kontak OFF durumundadır, kendinden sonraki işlemler için lojik 0 bilgisi üretilir. Kontak kapalı hâle geçtiğinde ON durumundadır, kendinden sonraki işlemler için lojik 1 bilgisi üretilir.

LDI, ANDI ve ORI Komutları: Normalde kapalı kontak ON durumundadır, terslenerek OFF durumunda kabul edilir, kendinden sonraki işlemler için lojik 0 bilgisi üretilir. Kontak açık hâle geldiğinde OFF durumu terslenerek ON durumunda kabul edilir, kendinden sonraki işlemler için lojik 1 bilgisi üretilir.

LDP, ANDP, ORP: Belirtilen bit için kod işletilmeden önce yapılan tarama ile bir önceki tarama karşılaştırılır, bir önceki tarama ile kod işletilmeden önce yapılan son taramada 0->1 (Açık->Kapalı) geçişi gözlemlendiyse yükselen kenar kabul edilir, kendinden sonraki işlemler yürütülür.

LDF, ANDF, ORF: Belirtilen bit için kod işletilmeden önce yapılan tarama ile bir önceki tarama karşılaştırılır, bir önceki tarama ile kod işletilmeden önce yapılan son taramada 1->0 (Kapalı->Açık) geçişi gözlemlendiyse düşen kenar kabul edilir, kendinden sonraki işlemler yürütülür.

Tablo 7.3: Bit lojik Giriş Komutları

Ladder Sembolü	Açıklama	Veri Tipi	Ladder Diyagramı
LD, AND, OR 	Kontağın elektriksel bilgisini alır. Açık devre->OFF Kapalı devre->ON	BOOL	
LDI, ANDI, ORI 	Kontağın elektriksel bilgisinin tersini alır. Açık devre->ON Kapalı devre->OFF	BOOL	
LDP, ANDP, ORP 	Yükselen kenar darbesi (0->1)	BOOL	
LDF, ANDF, ORF 	Düşen kenar darbesi (1->0) bilgisinin tersini alır.	BOOL	

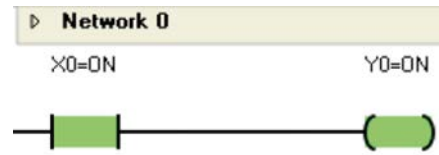
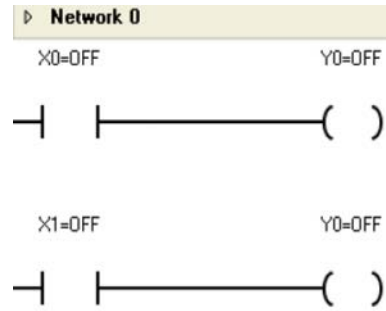
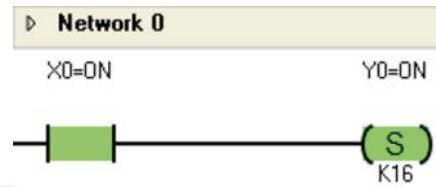
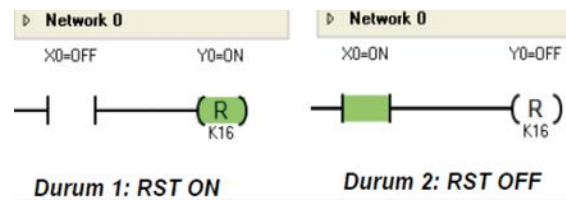
OUT: Kendinden önceki işlem sonucunu, bildirilen bit adresine veya çıkışa yazar.

Not: Çıkış adreslerini tekrarlamak hata durumudur. Örneğin X0 kontağı açık, X1 kontağı kapalı ise X0 kontağı açılır ve çıkış adresi Y0 kontağı aktif hâle geçer. Yardımcı röle M kontağı bu sorunu çözmek için kullanılır.

SET: Kendinden önceki işlemlerin sonucu 1 ise belirtilen adet kadar biti set eder.

RST: Kendinden önceki işlemlerin sonucu 1 ise belirtilen adet kadar biti reset eder.

Tablo 7.4: Bit Lojik Çıkış Komutları

Ladder Sembolü	Açıklama	Veri Tipi	Ladder Diyagramı
OUT 	Çıkış	BOOL	
	Çıkış hatası (OUT error)		
SET 	Çıkış set	BOOL	
RST 	Çıkış reset	BOOL	

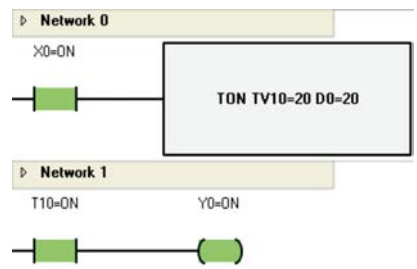
7.1.5.2. Zamanlayıcı (Timer) Komutları

Zamanlayıcılar, ayarlanan süre kadar zaman gecikmesi sağlayan fonksiyonlardır. TON, TOF, TONR gibi zaman fonksiyonları kullanılmaktadır. Her zamanlayıcı için bir veri bloğu oluşturulur. Bu blok içerisinde zamanlayıcı ile ilgili parametreler bulunur. Bir zamanlayıcının çalıştırılabilmesi için girişine uygulanan sinyalin pozitif bir değişim göstermesi şarttır.

TON: Giriş kontağı aktif olduğunda çekmede gecikmeli zaman rölesi sayma işlemine başlar ve set değerine (SV) ulaştığında zaman rölesi biti (OUT) 1 olur, giriş kontağı pasif hâle geldiğinde sayılan zaman sıfırlanır (Tablo 7.5).

Sayma işlemi devam ederken veya sayma değeri 0 iken zaman rölesi biti (OUT) 0'dır.

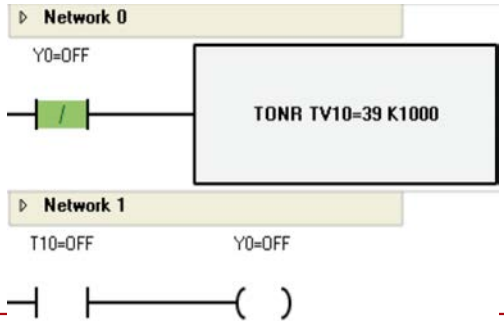
Tablo 7.5: TON Komutu

Komut	Açıklama	Veri Tipi	Ladder Diyagramı
TON	Çekmede gecikmeli zaman rölesi T: Anlık değer SV: Set değeri	WORD	

TONR: TONR komutu, giriş kontağı aktif olduğunda çekmede gecikmeli zaman rölesi sayma işlemine başlar ve set değerine (SV) ulaştığında zaman rölesi biti (OUT) 1 olur; giriş kontağı pasif hâle geldiğinde sayılan zaman sıfırlanmaz.

Sayma işlemi devam ederken veya sayma değeri 0 iken zaman rölesi biti (OUT) 0'dır.

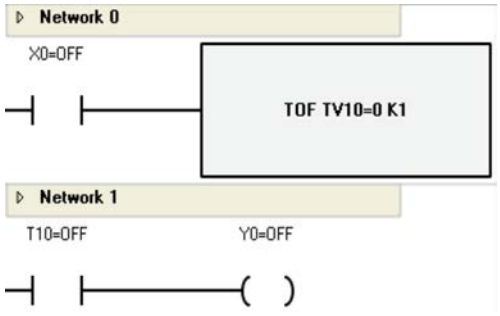
Tablo 7.6: TONR Komutu

Komut	Açıklama	Veri Tipi	Ladder Diyagramı
TONR	Kalıcı çekmede gecikmeli zaman rölesi T: Anlık değer SV: Set değeri	DOUBLE WORD	

TOF: TOF komutu, giriş kontağı pasif olduğunda bırakmada gecikmeli zaman rölesi sayma işlemine başlar ve set değerine (SV) ulaştığında zaman rölesi biti (OUT) 1 olur; giriş kontağı aktif hâle geldiğinde sayılan zaman sıfırlanır.

Sayma işlemi devam ederken veya sayma değeri 0 iken zaman rölesi biti (OUT) 0'dır.

Tablo 7.7: TOF Komutu

Komut	Açıklama	Veri Tipi	Ladder Diyagramı
TOF	Bırakmada gecikmeli zaman rölesi T: Anlık değer SV: Set değeri	WORD	

7.1.5.3. Sayıcı Komutları

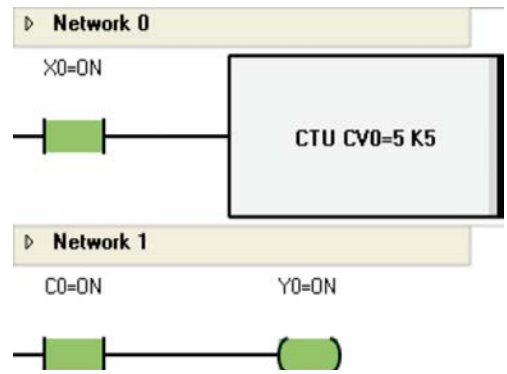
Sayıcılar sayma yapan fonksiyonlardır. Girişine verilen sinyaller ve istenilen sayma değerine bağlı olarak çıkışlarını aktif eder. Sayıcılarda CTU, CTD ve CTUD olmak üzere üç tip fonksiyon kullanılmaktadır. Her sayıcı için bir veri bloğu oluşturulur. Bu blok içinde sayıcı ile ilgili parametreler bulunur.

CTU (Yukarı Sayıcı): CTU komutu, girişin her yükselen kenarında (0->1 geçişinde) anlık sayma değerini (C) 0 değerinden başlayarak 1 arttırır. Sayma değeri (C) set değerine ulaştığında sayıcı rölesi biti (OUT) 1 olur, sayma işlemi durur.

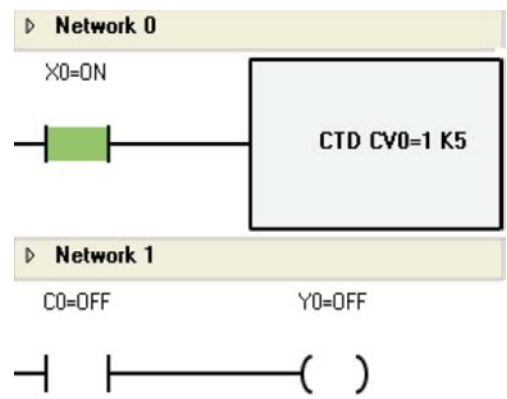
CTD (Aşağı Sayıcı): CTD komutu, girişin her yükselen kenarında (0->1 geçişinde) anlık sayma değerini (C) set değerinden (SV) başlayarak 1 azaltır. 0 değerine ulaştığında sayıcı rölesi biti (OUT) 1 olur, sayma işlemi durur.

CTUD (Yukarı / Aşağı Sayıcı): CTUD komutu, anlık sayma değeri (C) set değerinden küçük ise girişin her yükselen kenarında (0->1 geçişinde) anlık sayma değerini (C) 1 arttırır; büyük ise anlık sayma değeri (C) set değerine eşit olana kadar değeri 1 azaltır. Sayma değeri (C) set değerine ulaştığında sayıcı rölesi biti (OUT) 1 olur, sayma işlemi durur.

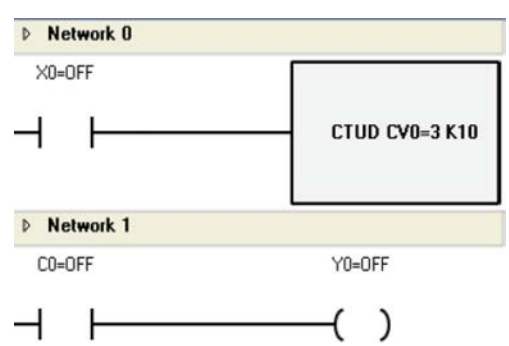
Tablo 7.8: CTU Komutu

Komut	Açıklama	Veri Tipi	Ladder Diyagramı
CTU	Yukarı sayıcı C: Anlık sayma değeri SV: Set değeri	WORD DOUBLE WORD	

Tablo 7.9: CTD Komutu

Komut	Açıklama	Veri Tipi	Ladder Diyagramı
CTD	Aşağı sayıcı C: Anlık sayma değeri SV: Set değeri	WORD DOUBLE WORD	

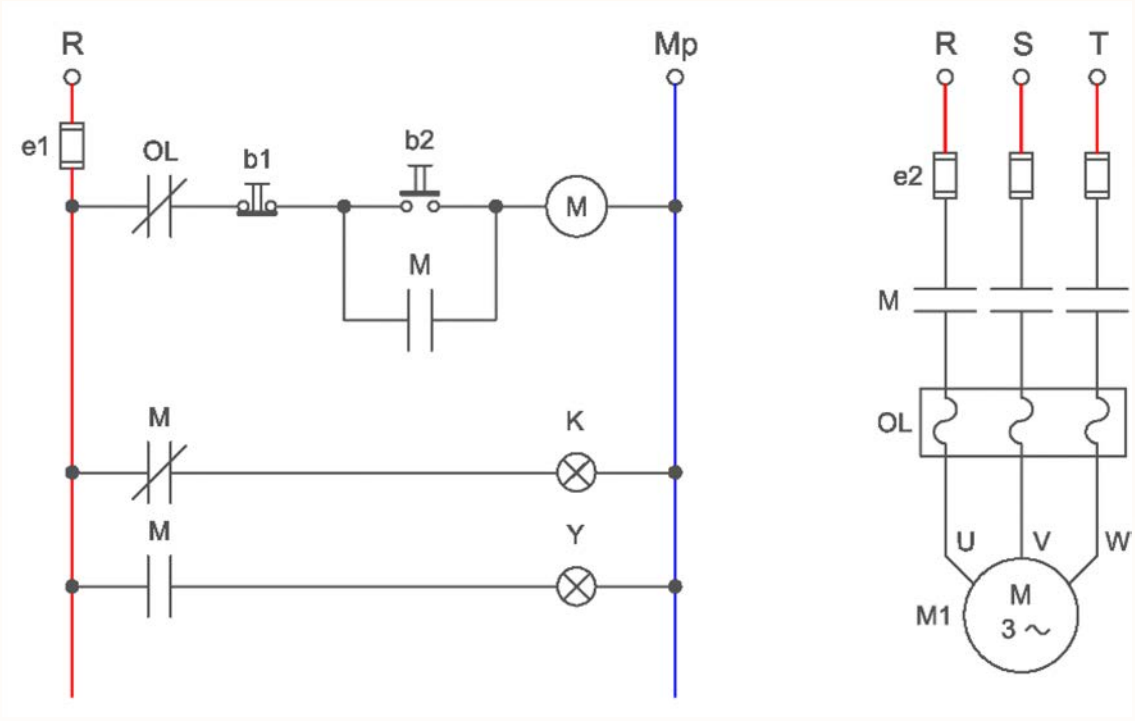
Tablo 7.10: CTUD Komutu

Komut	Açıklama	Veri Tipi	Ladder Diyagramı
CTUD	Yukarı / Aşağı sayıcı EN: Etkin giriş açıldığında, adımlar yürütülür ve çıkış aktif olur, aksi hâlde adımlar yürütülmez. C: Anlık sayma değeri SV: Set değeri	BOOL WORD DOUBLE WORD	

1. UYGULAMA

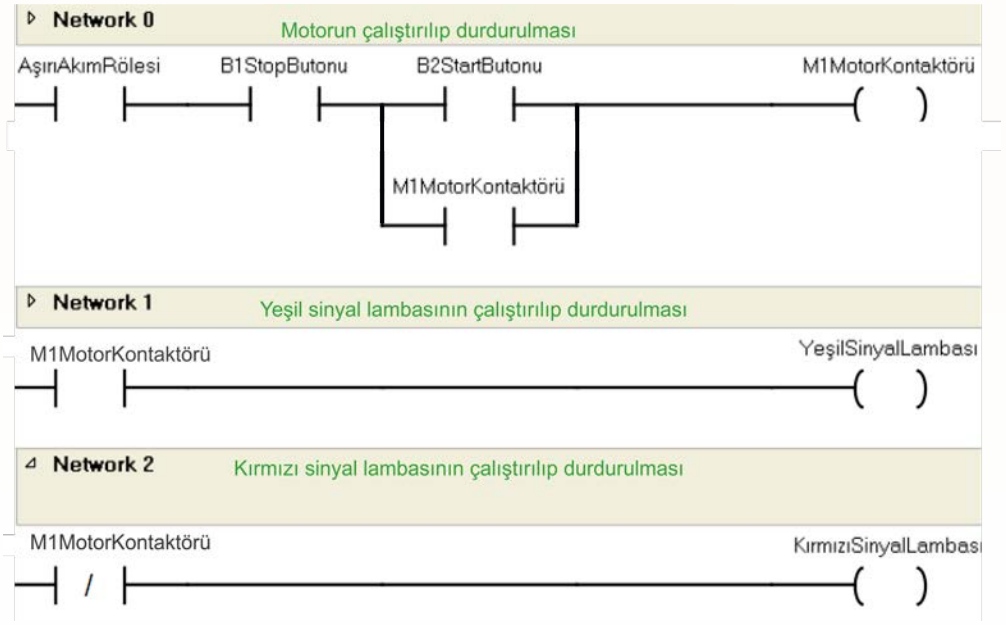
LADDER DİYAGRAMININ OLUŞTURULMASI

AMAÇ: Üç fazlı asenkron motorun sürekli çalışmasına ait kumanda devresinin ladder diyagramını oluşturmak.



Görsel 7.3: Üç fazlı asenkron motorun sürekli çalışmasına ait devre şeması (ANSI normu)

LADDER DİYAGRAMI



Görsel 7.4: Devrenin ladder diyagramı

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Bilgisayar	Editör programına uygun	1 adet
Programlama editörü	Bilgisayar işletim sistemine uygun	1 adet

ETİKET TABLOSU

Eleman İsmi	Veri Tipi	Bağlantı Tipi	Bağlantı Adresi	Adres	IEC Adresi	Açıklama
AşırıAkımRölesi	Bit	Giriş	I0	X0	%IX0	NK kontak
B1StopButonu	Bit	Giriş	I1	X1	%IX1	NK kontak
B2StartButonu	Bit	Giriş	I2	X2	%IX2	NA kontak
M1MotorKontaktörü	Bit	Çıkış	Q0	Y0	%QX0	Motor kontaktörü
YeşilSinyalLambası	Bit	Çıkış	Q1	Y1	%QX1	Yeşil sinyal lambası
KırmızıSinyalLambası	Bit	Çıkış	Q2	Y2	%QX2	Kırmızı sinyal lambası

NOT: PLC programlamada değişken isimlerinin Türkçe karakter olarak kullanılması bazı PLC'lerde problem oluşturabilmektedir.

İŞLEM BASAMAKLARI

1. Bilgisayarı açınız ve PLC editör programını çalıştırınız.
2. Networkleri oluşturunuz.
3. Eleman isimlendirmelerini yapınız.
4. Ladder diyagramını kaydediniz.
5. PLC editör programını ve bilgisayarı kapatınız.

SORULAR

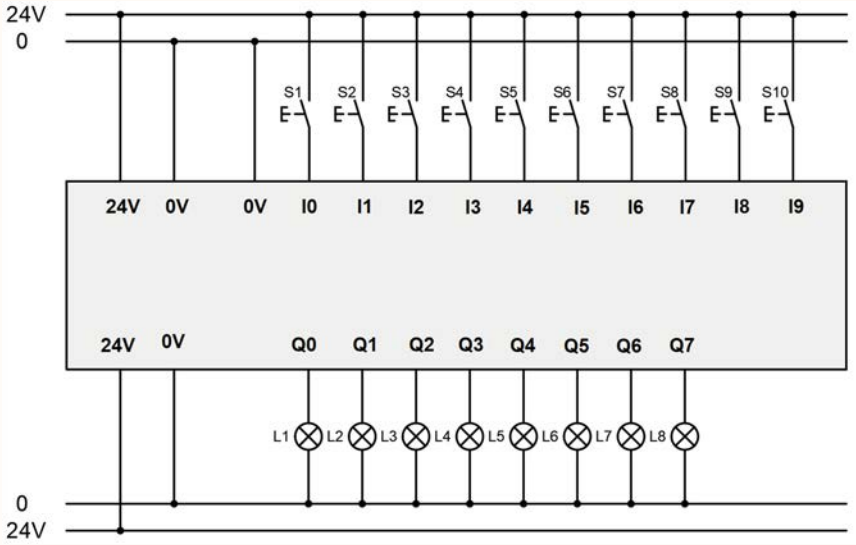
1. Programlama nedir? Açıklayınız.
2. Programlama çeşitlerini kısaca açıklayınız.
3. ANSI normuna göre ladder diyagramının nasıl oluşturulduğunu açıklayınız.
4. Üç fazlı asenkron motorun kesik çalıştırma devresine ait ladder diyagramı oluşturunuz.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Editörü doğru kullandı.	20	
Sınıfı :	2	Networkleri oluşturdu.	20	
Numarası :	3	İsimlendirmeleri yaptı.	20	
ÖĞRETMEN	4	Dosyayı isimlendirerek kaydetti.	20	
Adı Soyadı :	5	Etiket tablosunu doğru yorumladı.	20	
İmza :		**TOPLAM PUAN**	100	

2. UYGULAMA

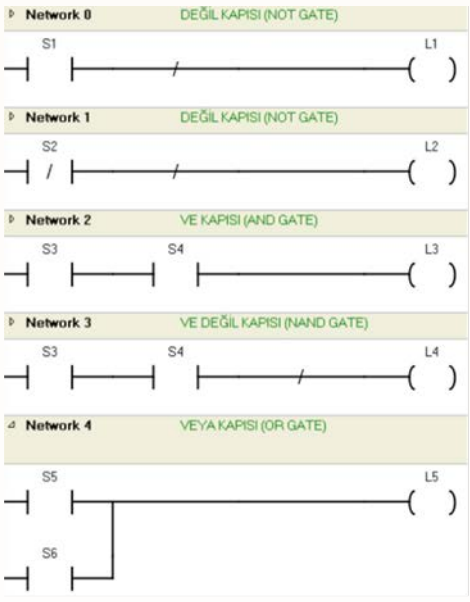
TEMEL MANTIK FONKSİYON UYGULAMALARI

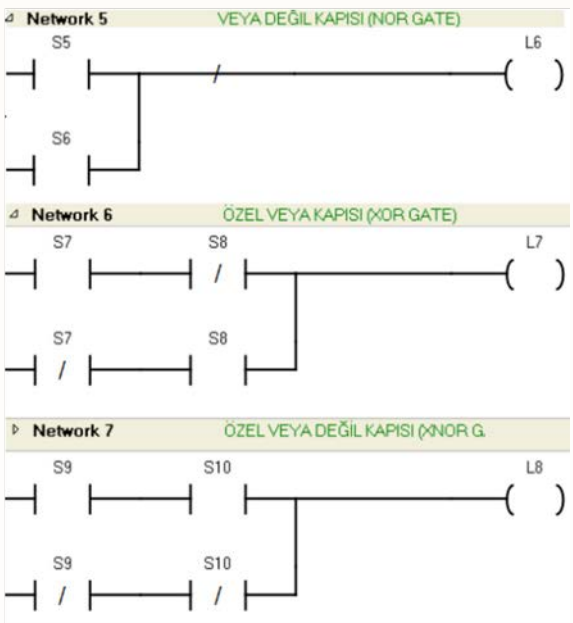
AMAÇ: **Temel mantık fonksiyonları ile oluşturulan bağlantıların uygulamalarını yapmak.**



Görsel 7.5: PLC bağlantı şeması

LADDER DİYAGRAMI





MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
PLC	Transistör çıkışlı	1 adet
Start butonu	Ani temaslı	10 adet
Sinyal lambası	24 VDC	8 adet
PC ve editör programı	Yapılacak uygulamaya uygun	1 adet

ETİKET TABLOSU						
Eleman İsmi	**Veri Tipi**	**Bağlantı Tipi**	**Bağlantı Adresi**	**Adres**	**IEC Adresi**	**Açıklama**
S1	Bit	Giriş	I0	X0	%IX0	NA kontak
S2	Bit	Giriş	I1	X1	%IX1	NA kontak
S3	Bit	Giriş	I2	X2	%IX2	NA kontak
S4	Bit	Giriş	I3	X3	%IX3	NA kontak
S5	Bit	Giriş	I4	X4	%IX4	NA kontak
S6	Bit	Giriş	I5	X5	%IX5	NA kontak
S7	Bit	Giriş	I6	X6	%IX6	NA kontak
S8	Bit	Giriş	I7	X7	%IX7	NA kontak
S9	Bit	Giriş	I8	X8	%IX8	NA kontak
S10	Bit	Giriş	I9	X9	%IX9	NA kontak
L1	Bit	Çıkış	Q0	Y0	%QX0	Lamba 1
L2	Bit	Çıkış	Q1	Y1	%QX1	Lamba 2
L3	Bit	Çıkış	Q2	Y2	%QX2	Lamba 3
L4	Bit	Çıkış	Q3	Y3	%QX3	Lamba 4
L5	Bit	Çıkış	Q4	Y4	%QX4	Lamba 5
L6	Bit	Çıkış	Q5	Y5	%QX5	Lamba 6
L7	Bit	Çıkış	Q6	Y6	%QX6	Lamba 7
L8	Bit	Çıkış	Q7	Y7	%QX7	Lamba 8

İŞLEM BASAMAKLARI

1. PLC projesini oluşturunuz (Görsel 7.5) ve programı bilgisayarda simüle ediniz.
2. PLC giriş ve çıkış elemanlarının bağlantılarını yapınız.
3. Öğretmeninizin kontrolünde PLC deney setine enerji veriniz.
4. I0 ve I1 girişlerine bağlı butonları sırasıyla aktif ederek Q0 ve Q1 çıkışlarına bağlı sinyal lambalarının durumunu gözlemleyiniz (Network 1).
5. I2 ve I3 girişlerine bağlı butonları sırasıyla aktif ya da pasif ederek Q2 çıkışına bağlı sinyal lambasının durumunu gözlemleyiniz (Network 2).
6. Diğer networklerde bulunan çalışmaları da gerçekleştirip PLC çıkışlarını gözlemleyiniz.
7. Deney setinin enerjisini kesip, elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

SORU

- Temel mantık fonksiyonlarını açıklayınız.

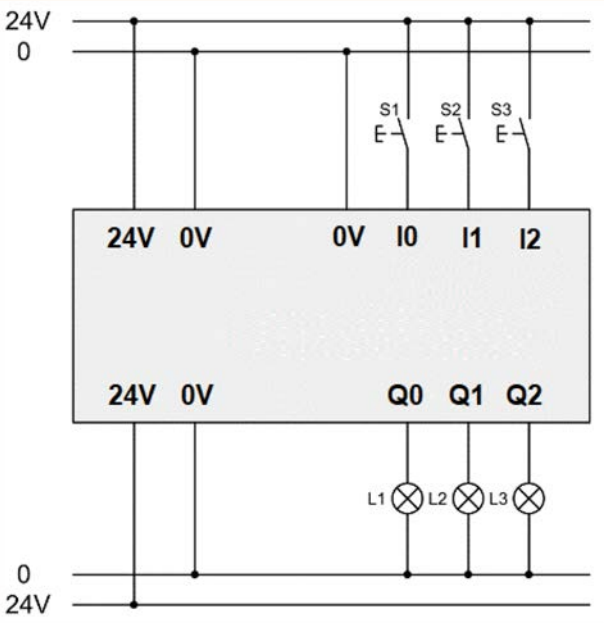
ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	PLC programını oluşturdu ve PLC'ye yükledi.	20	
Sınıfı :	2	Giriş elemanlarını bağladı.	20	
Numarası :	3	Çıkış elemanlarını bağladı.	20	
ÖĞRETMEN	4	Devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

3. UYGULAMA

BUTON UYGULAMALARI

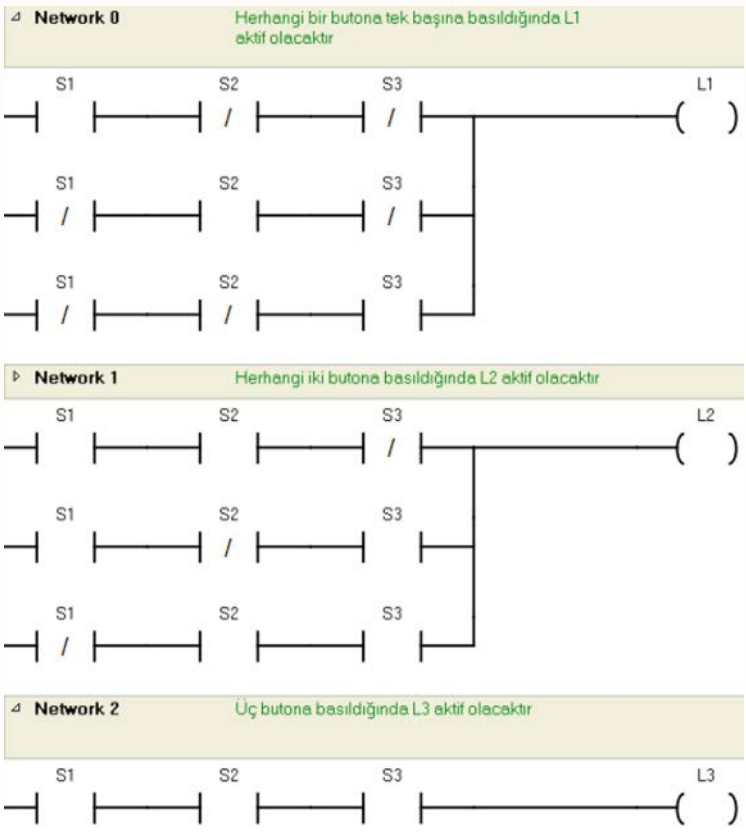
AMAÇ: **Farklı buton bağlantıları ile açık ve kapalı kontak uygulamalarını yapmak.**

YÖNERGE: **Herhangi bir butona basıldığında Q0 çıkışı aktif olacaktır. Herhangi iki butona basıldığında Q1 çıkışı aktif olacak ve üç butona basıldığında ise Q2 çıkışı aktif olacaktır.**



Görsel 7.6: PLC bağlantı şeması

LADDER DİYAGRAMI



MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
PLC	Transistör çıkışlı	1 adet
Start butonu	Ani temaslı	3 adet
Sinyal lambası	24 VDC	3 adet
PC ve editör programı	Yapılacak uygulamaya uygun	1 adet

ETİKET TABLOSU						
Eleman İsmi	**Veri Tipi**	**Bağlantı Tipi**	**Bağlantı Adresi**	**Adres**	**IEC Adresi**	**Açıklama**
S1	Bit	Giriş	I0	X0	%IX0	NA kalıcı kontak
S2	Bit	Giriş	I1	X1	%IX1	NA kalıcı kontak
S3	Bit	Giriş	I2	X2	%IX2	NA kalıcı kontak
L1	Bit	Çıkış	Q0	Y0	%QX0	Lamba 1
L2	Bit	Çıkış	Q1	Y1	%QX1	Lamba 2
L3	Bit	Çıkış	Q2	Y2	%QX2	Lamba 3

İŞLEM BASAMAKLARI

1. Yapılacak işe uygun PLC seçimini yaparak PLC projesini oluşturunuz (Görsel 7.6).
2. Programı bilgisayarda simüle ediniz.
3. PLC giriş ve çıkış elemanlarının bağlantılarını yapınız.
4. Öğretmeninizin kontrolünde PLC deney setine enerji veriniz.
5. S1, S2, S3 butonlarını ayrı ayrı aktif ederek her bir durum için L1 sinyal lambasının yandığını, diğer durumlardaysa yanmadığını deney seti üzerinde gözlemleyiniz (Network 1).
6. S1 ve S2, S1 ve S3, S2 ve S3 butonlarını ikişerli gruplar hâlinde aktif ederek her bir durum için L2 sinyal lambasının yandığını, diğer durumlardaysa yanmadığını deney seti üzerinde gözlemleyiniz (Network 2).
7. S1, S2 ve S3 butonlarının hepsini grup hâlinde aktif ederek L3 sinyal lambasının yandığını, diğer durumlardaysa yanmadığını deney seti üzerinde gözlemleyiniz (Network 3).
8. Deney setinin enerjisini kesip, elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

SORULAR

1. Etiket tablosunun kullanım amacını açıklayınız.
2. Devrenin çalışmasını ladder diyagramı üzerinden açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	PLC programını oluşturdu ve PLC'ye yükledi.	20	
Sınıfı :	2	Giriş elemanlarını bağladı.	20	
Numarası :	3	Çıkış elemanlarını bağladı.	20	
ÖĞRETMEN	4	Devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

4. UYGULAMA

BUTONLARLA ARA VAVİYEN UYGULAMASI

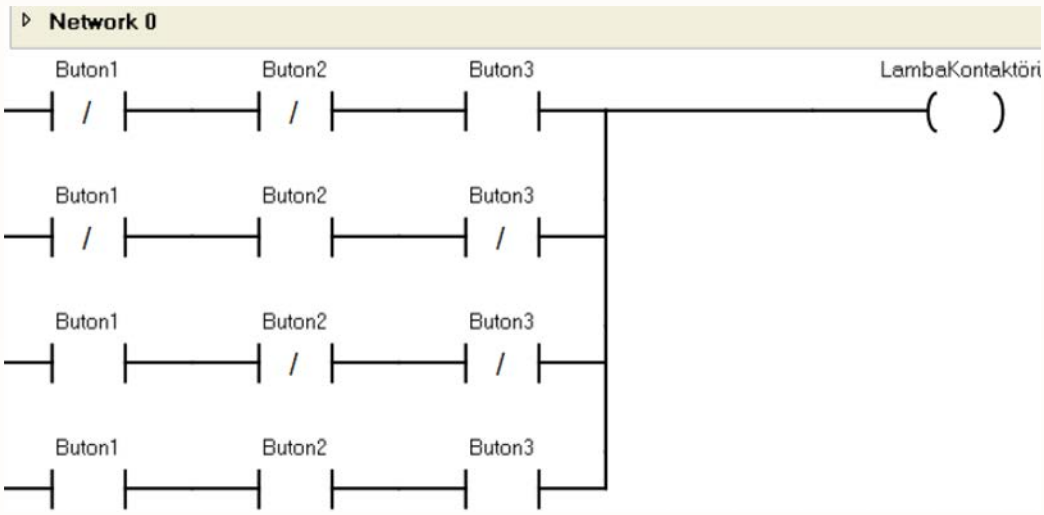
AMAÇ: **Doğruluk tablosu kullanarak buton uygulamalarını gerçekleştirmek.**

YÖNERGE: **Bir aydınlatma sistemi, üç ayrı kalıcı tip buton ile kontaktör üzerinden kontrol edilecektir. Herhangi bir butona basıldığında aydınlatma sistemi devrede ise devre dışı kalacak, devrede değilse devreye girerek ortam aydınlatmasını sağlayacaktır.**

DOĞRULUK TABLOSU

Girişler	**I0**	0	0	0	0	1	1	1	1
	I1	0	0	1	1	0	0	1	1
	I2	0	1	0	1	0	1	0	1
Çıkış	**Q0**	0	1	1	0	1	0	0	1

LADDER DİYAGRAMI



MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
PLC	Transistör çıkışlı	1 adet
Start butonu	Ani temaslı	3 adet
Kontaktör	24 VDC	1 adet
PC ve editör programı	Yapılacak uygulamaya uygun	1 adet

ETİKET TABLOSU						
Eleman İsmi	**Veri Tipi**	**Bağlantı Tipi**	**Bağlantı Adresi**	**Adres**	**IEC Adresi**	**Açıklama**
Buton1	Bit	Giriş	I0	X0	%IX0	NA kalıcı kontak
Buton2	Bit	Giriş	I1	X1	%IX1	NA kalıcı kontak
Buton3	Bit	Giriş	I2	X2	%IX2	NA kalıcı kontak
LambaKontaktörü	Bit	Çıkış	Q0	Y0	%QX0	Lamba kontaktörü

İŞLEM BASAMAKLARI

1. Uygulama projesini oluşturunuz ve bilgisayarda simüle ediniz.
2. Programı kaydederek PLC'ye yükleyiniz.
3. PLC giriş ve çıkış elemanlarının bağlantılarını yapınız.
4. Öğretmeninizin kontrolünde PLC deney setine enerji veriniz.
5. Doğruluk tablosuna uygun olarak, girişleri verilen sırada aktif edip çıkışı gözlemleyiniz.
6. Deney setinin enerjisini kesip, elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

SORU

- Devrenin çalışmasını ladder diyagramı üzerinden açıklayınız.

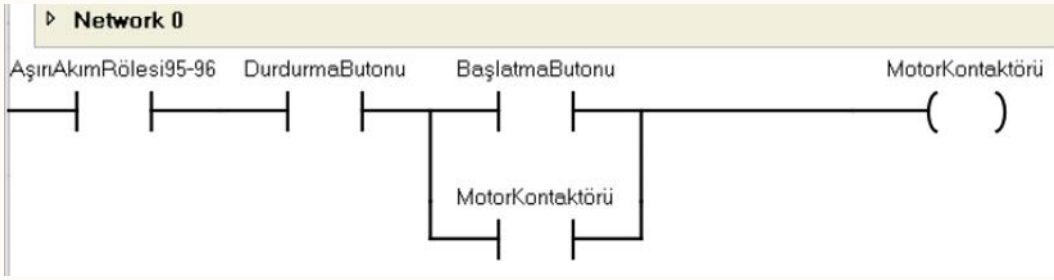
ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	PLC programını oluşturdu ve PLC'ye yükledi.	20	
Sınıfı :	2	Giriş elemanlarını bağladı.	20	
Numarası :	3	Çıkış elemanlarını bağladı.	20	
ÖĞRETMEN	4	Devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

5. UYGULAMA

ÜÇ FAZLI ASENKRON MOTORA DİREKT YOL VERME

AMAÇ: **PLC kullanarak üç fazlı asenkron motora direkt yol vermek.**

LADDER DİYAGRAMI



MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
PLC	Transistör çıkışlı	1 adet
Start butonu	Ani temaslı	2 adet
Kontaktör	4 kW	1 adet
Aşırı akım rölesi	2,5 - 4 A	1 adet
Asenkron motor	Üç fazlı	1 adet
PC ve editör programı	Yapılacak uygulamaya uygun	1 adet

ETİKET TABLOSU						
Eleman İsmi	**Veri Tipi**	**Bağlantı Tipi**	**Bağlantı Adresi**	**Adres**	**IEC Adresi**	**Açıklama**
AşırıAkımRölesi95-96	Bit	Giriş	I0	X0	%IX0	NK kontak
DurdurmaButonu	Bit	Giriş	I1	X1	%IX1	NK kontak
BaşlatmaButonu	Bit	Giriş	I2	X2	%IX2	NA kontak
MotorKontaktörü	Bit	Çıkış	Q0	Y0	%QX0	Motor kontaktörü

İŞLEM BASAMAKLARI

1. Uygulama projesini oluşturunuz ve bilgisayarda simüle ediniz.
2. Programı kaydederek PLC'ye yükleyiniz.
3. PLC giriş ve çıkış elemanlarının bağlantılarını yapınız.
4. Öğretmeninizin kontrolünde PLC deney setine enerji veriniz.
5. Başlatma butonuna basarak motoru çalıştırınız.
6. Durdurma butonuna basarak motoru durdurunuz.
7. Deney setinin enerjisini kesip, elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

SORU

- Devrede mühürleme işleminin nasıl yapıldığını diyagram üzerinden açıklayınız.

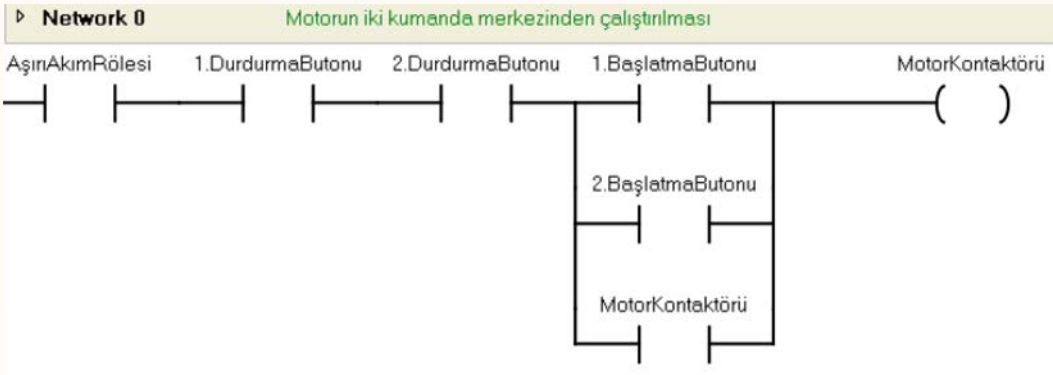
ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	PLC programını oluşturdu ve PLC'ye yükledi.	20	
Sınıfı :	2	Giriş elemanlarını bağladı.	20	
Numarası :	3	Çıkış elemanlarını bağladı.	20	
ÖĞRETMEN	4	Devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

6. UYGULAMA

ÜÇ FAZLI ASENKRON MOTORUN FARKLI MERKEZLERDEN KONTROLÜ

AMAÇ: **PLC kullanarak, üç fazlı asenkron motoru iki farklı yerden çalıştırıp durdurmak.**

LADDER DİYAGRAMI



MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
PLC	Transistör çıkışlı	1 adet
Buton	Ani temaslı	4 adet
Kontaktör	4 kW	1 adet
Aşırı akım rölesi	2,5 - 4 A	1 adet
Asenkron motor	Üç fazlı	1 adet
PC ve editör programı	Yapılacak uygulamaya uygun	1 adet

ETİKET TABLOSU						
Eleman İsmi	**Veri Tipi**	**Bağlantı Tipi**	**Bağlantı Adresi**	**Adres**	**IEC Adresi**	**Açıklama**
AşırıAkımRölesi	Bit	Giriş	I0	X0	%IX0	NK kontak
1.BaşlatmaButonu	Bit	Giriş	I1	X1	%IX1	NA kontak
2.BaşlatmaButonu	Bit	Giriş	I2	X2	%IX2	NA kontak
1.DurdurmaButonu	Bit	Giriş	I3	X3	%IX3	NK kontak
2.DurdurmaButonu	Bit	Giriş	I4	X4	%IX4	NK kontak
MotorKontaktörü	Bit	Çıkış	Q0	Y0	%QX0	Motor kontaktörü

İŞLEM BASAMAKLARI

1. Uygulama projesini oluşturup bilgisayarda simüle ediniz ve PLC'ye yükleyiniz.
2. PLC giriş-çıkış elemanlarının bağlantılarını yapınız.
3. Öğretmeninizin kontrolünde PLC deney setine enerji veriniz.
4. Aynı ve farklı merkezlerden başlatma ve durdurma işlemleri yapınız.
5. Deney setinin enerjisini kesip, elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

SORU

- Farklı merkezlerden kontrolü diyagram üzerinden açıklayınız.

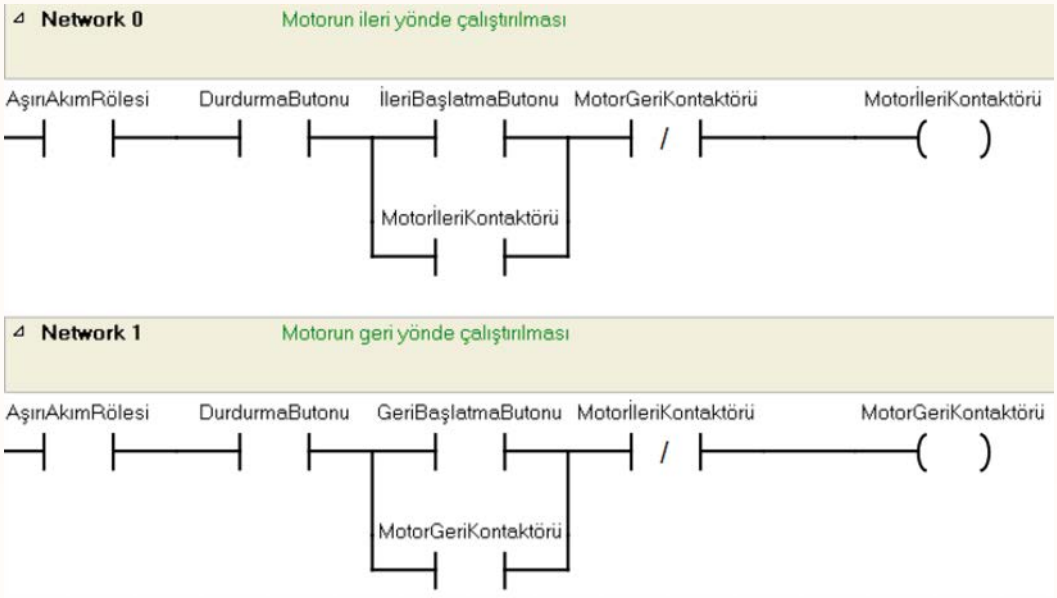
ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	PLC programını oluşturdu ve PLC'ye yükledi.	20	
Sınıfı :	2	Giriş elemanlarını bağladı.	20	
Numarası :	3	Çıkış elemanlarını bağladı.	20	
ÖĞRETMEN	4	Devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

7. UYGULAMA

ÜÇ FAZLI ASENKRON MOTORUN DEVİR YÖNÜNÜ DEĞİŞTİRME (ELEKTRİKSEL KİLİTLEMELİ)

AMAÇ: **PLC kullanarak, üç fazlı asenkron motorun elektriksel kilitlemeli olarak devir yönünü değiştirmek.**

LADDER DİYAGRAMI



MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
PLC	Transistör çıkışlı	1 adet
Buton	Ani temaslı	3 adet
Kontaktör	4 kW	2 adet
Aşırı akım rölesi	2,5 - 4 A	2 adet
Asenkron motor	Üç fazlı	1 adet
PC ve editör programı	Yapılacak uygulamaya uygun	1 adet

ETİKET TABLOSU						
Eleman İsmi	**Veri Tipi**	**Bağlantı Tipi**	**Bağlantı Adresi**	**Adres**	**IEC Adresi**	**Açıklama**
AşırıAkımRölesi	Bit	Giriş	I0	X0	%IX0	NK kontak
DurdurmaButonu	Bit	Giriş	I1	X1	%IX1	NK kontak
GeriBaşlatmaButonu	Bit	Giriş	I2	X2	%IX2	NA kontak
İleriBaşlatmaButonu	Bit	Giriş	I3	X3	%IX3	NA kontak
MotorİleriKontaktörü	Bit	Çıkış	Q0	Y0	%QX0	Motor ileri yön kontaktörü
MotorGeriKontaktörü	Bit	Çıkış	Q1	Y1	%QX1	Motor geri yön kontaktörü

İŞLEM BASAMAKLARI

1. Uygulama projesini oluşturunuz ve bilgisayarda simüle ediniz.
2. Programı kaydederek PLC'ye yükleyiniz.
3. PLC giriş-çıkış elemanlarının bağlantılarını yapınız.
4. Öğretmeninizin kontrolünde PLC deney setine enerji veriniz.
5. İleri yön butonuna basarak motoru ileri yönde çalıştırınız.
6. Geri yön butonuna basarak devir yönünün değişmediğini gözlemleyiniz.
7. Stop butonuna basarak motoru durdurunuz.
8. Geri yön butonuna basarak motoru geri yönde çalıştırınız.
9. İleri yön butonuna basarak motorun ileri yönde çalışmadığını gözlemleyiniz.
10. Stop butonuna basarak motoru durdurunuz.
11. Deney setinin enerjisini kesip, elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

SORU

- Ladder diyagramı üzerinden sistemin çalışmasını açıklayınız.

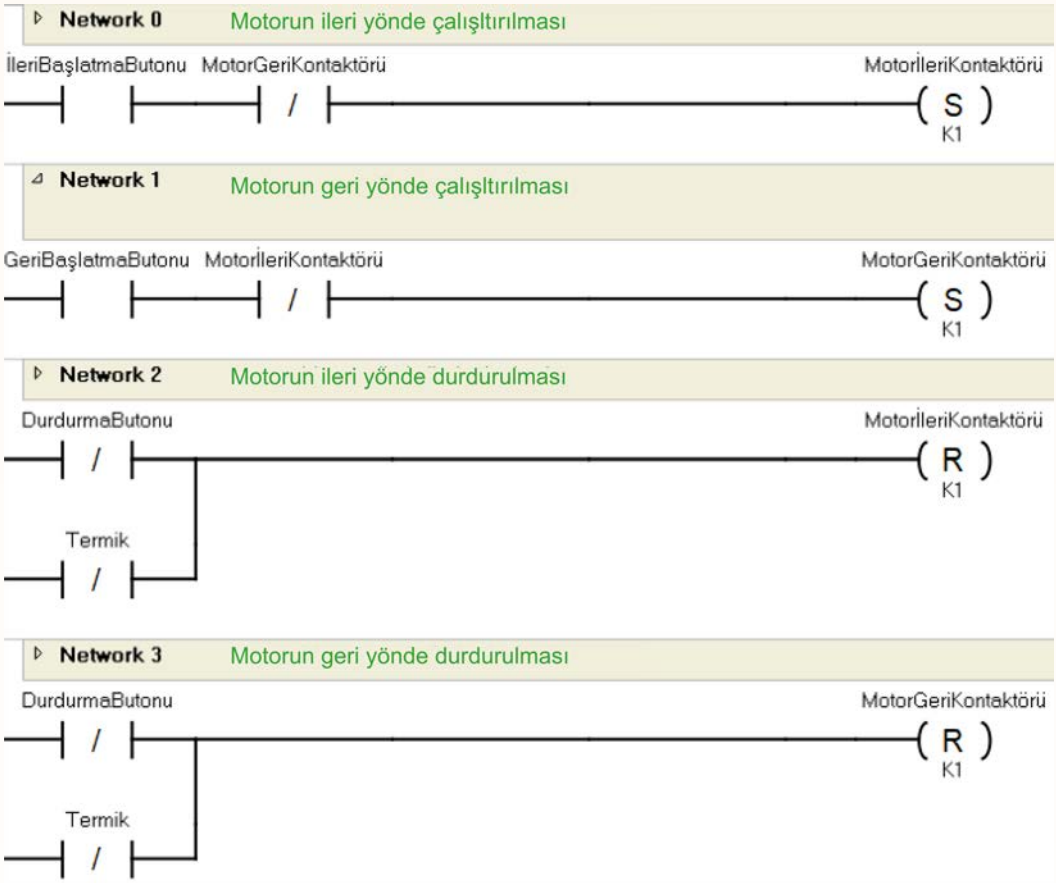
ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	PLC programını oluşturdu ve PLC'ye yükledi.	20	
Sınıfı :	2	Giriş elemanlarını bağladı.	20	
Numarası :	3	Çıkış elemanlarını bağladı.	20	
ÖĞRETMEN	4	Devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

8. UYGULAMA

ÜÇ FAZLI ASENKRON MOTORUN DEVİR YÖNÜNÜ DEĞİŞTİRME (SET RESETLİ)

AMAÇ: **Set reset komutlarını kullanarak üç fazlı asenkron motorun devir yönünü değiştirmek.**

LADDER DİYAGRAMI



MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
PLC	Transistör çıkışlı	1 adet
Buton	Ani temaslı	3 adet
Kontaktör	4 kW	2 adet
Aşırı akım rölesi	2,5 - 4 A	2 adet
Asenkron motor	Üç fazlı	1 adet
PC ve editör programı	Yapılacak uygulamaya uygun	1 adet

ETİKET TABLOSU						
Eleman İsmi	**Veri Tipi**	**Bağlantı Tipi**	**Bağlantı Adresi**	**Adres**	**IEC Adresi**	**Açıklama**
Termik	Bit	Giriş	I0	X0	%IX0	NK kontak
DurdurmaButonu	Bit	Giriş	I1	X1	%IX1	NK kontak
İleriBaşlatmaButonu	Bit	Giriş	I2	X2	%IX2	NA kontak
GeriBaşlatmaButonu	Bit	Giriş	I3	X3	%IX3	NA kontak
MotorİleriKontaktörü	Bit	Çıkış	Q0	Y0	%QX0	Motor ileri yön kontaktörü
MotorGeriKontaktörü	Bit	Çıkış	Q1	Y1	%QX1	Motor geri yön kontaktörü

İŞLEM BASAMAKLARI

1. Uygulama projesini oluşturunuz ve bilgisayarda simüle ediniz.
2. Programı kaydederek PLC'ye yükleyiniz.
3. PLC giriş-çıkış elemanlarının bağlantılarını yapınız.
4. Öğretmeninizin kontrolünde PLC deney setine enerji veriniz.
5. Herhangi bir yönde motorun çalışmasını başlatınız.
6. Farklı yönde çalıştırma butonuna basarak sistemin çalışmasında herhangi bir yön değişimi olmadığını gözlemleyiniz.
7. Durdurma butonuna ve ardından farklı yön butonuna basarak motor devir yönünün değişimini gözlemleyiniz.
8. Deney setinin enerjisini kesip, elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

SORULAR

1. Ladder diyagramı üzerinden sistemin çalışmasını açıklayınız.
2. Set ve Reset komutlarını kısaca açıklayınız.

ÖĞRENCİ	**DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA**			
Adı Soyadı :	1	PLC programını oluşturdu ve PLC'ye yükledi.	20	
Sınıfı :	2	Giriş elemanlarını bağladı.	20	
Numarası :	3	Çıkış elemanlarını bağladı.	20	
ÖĞRETMEN	4	Devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

9. UYGULAMA

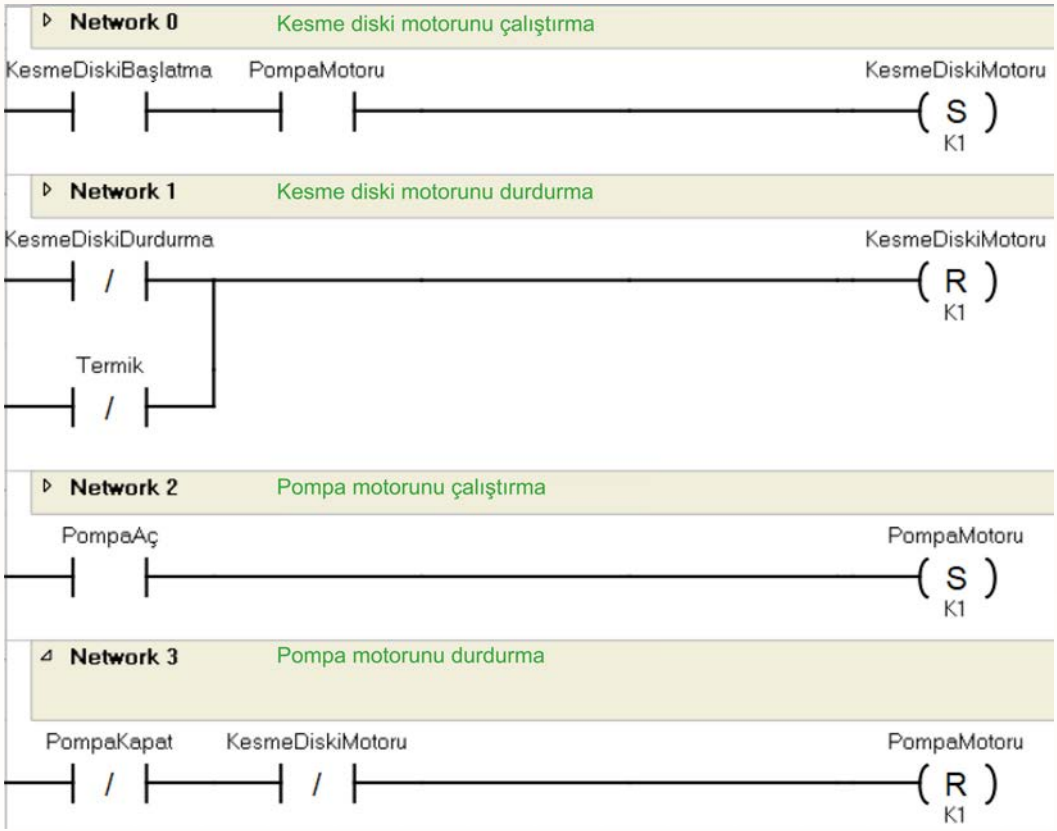
SET RESET KOMUTLU UYGULAMA DEVRESİ 1

AMAÇ: Set reset komutlarıyla pompa uygulama devresi yapmak.

YÖNERGE

- Bir metal kesme tezgâhı, aşağıdaki şartlar altında çalıştırılacaktır.
- Kesme diski ve soğutma sıvı pompasını çalıştıran iki ayrı motor bulunacaktır.
- Kesme diskinin çalışması için soğutma sıvısı pompasının çalışması gerekmektedir.
- Soğutma sıvısı pompasının durdurulabilmesi için kesme diski durdurulmuş olmalıdır.
- Her motor kendi başlatma ve durdurma butonları ile kontrol edilebilecektir.
- Kesme disk motoru termik röle ile korunacaktır.
- Sistemin çalıştırılması için gerekli PLC projesini oluşturarak uygulamasını gerçekleştiriniz.

LADDER DİYAGRAMI



MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
PLC	Transistör çıkışlı	1 adet
Buton	Ani temaslı	2 adet
Kontaktör	4 kW	2 adet
Aşırı akım rölesi	2,5 - 4 A	1 adet
Asenkron motor	Üç fazlı	1 adet
PC ve editör programı	Yapılacak uygulamaya uygun	1 adet

ETİKET TABLOSU						
Eleman İsmi	**Veri Tipi**	**Bağlantı Tipi**	**Bağlantı Adresi**	**Adres**	**IEC Adresi**	**Açıklama**
Termik	Bit	Giriş	I0	X0	%IX0	NK kontak
KesmeDiskiBaşlatma	Bit	Giriş	I1	X1	%IX1	NA kontak
KesmeDiskiDurdurma	Bit	Giriş	I2	X2	%IX2	NK kontak
PompaAç	Bit	Giriş	I3	X3	%IX3	NA kontak
PompaKapat	Bit	Giriş	I4	X4	%IX4	NK kontak
KesmeDiskiMotoru	Bit	Çıkış	Q0	Y0	%QX0	Kesme diski motor kontaktörü
PompaMotoru	Bit	Çıkış	Q1	Y1	%QX1	Pompa motor kontaktörü

İŞLEM BASAMAKLARI

1. Uygulama projesini oluşturunuz ve bilgisayarda simüle ediniz.
2. Programı kaydederek PLC'ye yükleyiniz.
3. PLC giriş-çıkış elemanlarının bağlantılarını yapınız.
4. Öğretmeninizin kontrolünde PLC deney setine enerji veriniz.
5. PompaAç butonuna basarak pompa motorunun çalışmasını gözlemleyiniz (Network 2).
6. Pompa motoru çalıştıktan sonra I1 butonuna basarak kesme diski motorunun çalışmasını gözlemleyiniz (Network 1).
7. Çalışma senaryosuna uygun olarak, farklı çalıştırma ve durdurma kombinasyonları deneyerek sonuçları gözlemleyiniz.
8. Deney setinin enerjisini kesip, elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

SORULAR

1. Sistem klasik röleler kullanılarak gerçekleştirilebilir mi? Açıklayınız.
2. Sistem set ve reset röleleri kullanılarak nasıl gerçekleştirilebilir? Açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	PLC programını oluşturdu ve PLC'ye yükledi.	20	
Sınıfı :	2	Giriş elemanlarını bağladı.	20	
Numarası :	3	Çıkış elemanlarını bağladı.	20	
ÖĞRETMEN	4	Devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

10. UYGULAMA

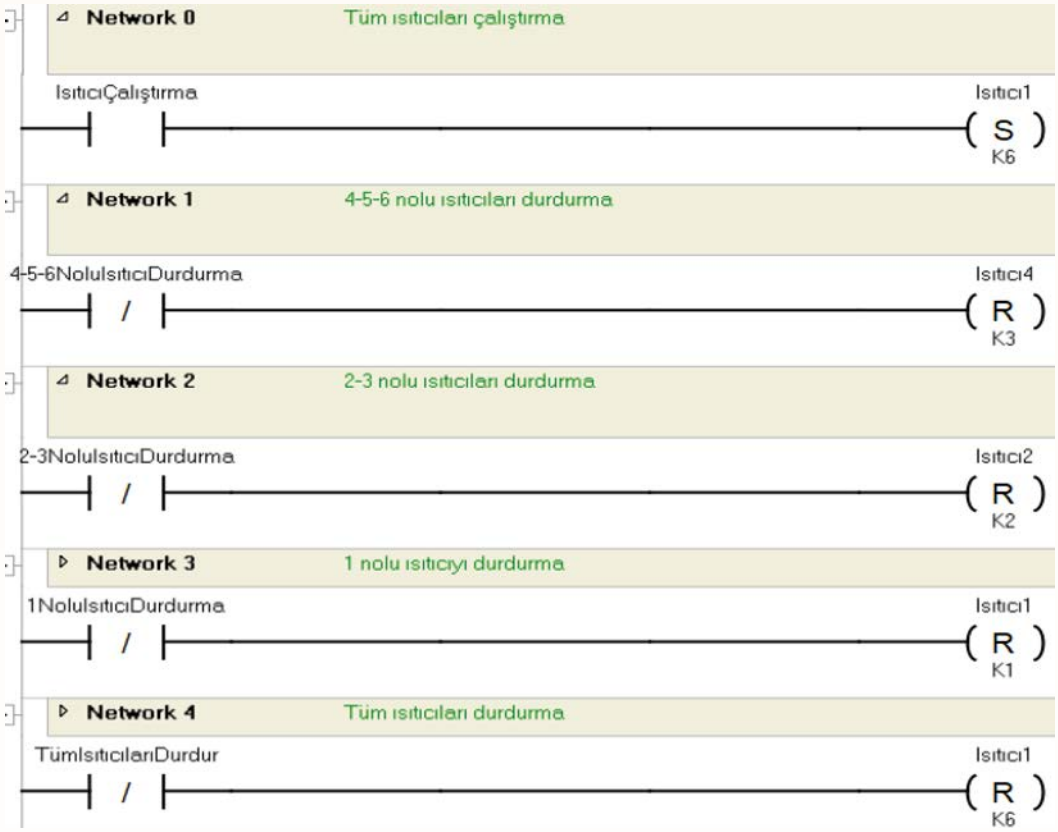
SET RESET KOMUTLU UYGULAMA DEVRESİ 2

AMAÇ: Set reset komutlarını kullanarak ısıtıcı uygulaması yapmak.

YÖNERGE

- Bir boya kurutma ünitesinde 6 adet ısıtıcı manuel olarak kontrol edilmektedir. Sistemin kumandası 1 adet başlatma ve 4 adet durdurma butonu ile sağlanacaktır.
- S1 NA kontak başlatma butonuna basıldığında tüm ısıtıcılar devreye girecektir.
- S2 butonuna basıldığında 4,5 ve 6 numaralı ısıtıcı devre dışı kalacaktır.
- S3 durdurma butonuna basıldığında 2 ve 3 numaralı ısıtıcı devre dışı kalacaktır.
- S4 kapalı kontak durdurma butonuna basıldığında 1 numaralı ısıtıcı devre dışı kalacaktır.
- S5 NK kontak durdurma butonuna basıldığında tüm ısıtıcılar devre dışı kalacaktır.
- Sistemin çalıştırılması için gerekli PLC projesini oluşturarak uygulamasını gerçekleştiriniz.

LADDER DİYAGRAMI



MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
PLC	Transistör çıkışlı	1 adet
Buton	Ani temaslı	5 adet
Kontaktör	4 kW	6 adet
PC ve editör programı	Yapılacak uygulamaya uygun	1 adet

ETİKET TABLOSU						
Eleman İsmi	**Veri Tipi**	**Bağlantı Tipi**	**Bağlantı Adresi**	**Adres**	**IEC Adresi**	**Açıklama**
IsıtıcıÇalıştırma	Bit	Giriş	I0	X0	%IX0	NA kontak
4-5-6No.lulsıtıcıdurdurma	Bit	Giriş	I1	X1	%IX1	NK kontak
2-3No.lulsıtıcıdurdurma	Bit	Giriş	I2	X2	%IX2	NK kontak
1No.lulsıtıcıdurdurma	Bit	Giriş	I3	X3	%IX3	NK kontak
TümIsıtıcılarıdurdur	Bit	Giriş	I4	X4	%IX4	NK kontak
Isıtıcı1	Bit	Çıkış	Q0	Y0	%QX0	Isıtıcı1 kontaktörü
Isıtıcı2	Bit	Çıkış	Q1	Y1	%QX1	Isıtıcı2 kontaktörü
Isıtıcı3	Bit	Çıkış	Q2	Y2	%QX2	Isıtıcı3 kontaktörü
Isıtıcı4	Bit	Çıkış	Q3	Y3	%QX3	Isıtıcı4 kontaktörü
Isıtıcı5	Bit	Çıkış	Q4	Y4	%QX4	Isıtıcı5 kontaktörü
Isıtıcı6	Bit	Çıkış	Q5	Y5	%QX5	Isıtıcı6 kontaktörü

İŞLEM BASAMAKLARI

1. Uygulama projesini oluşturunuz ve bilgisayarda simüle ediniz.
2. Programı kaydederek PLC'ye yükleyiniz.
3. PLC giriş-çıkış elemanlarının bağlantılarını yapınız.
4. Öğretmeninizin kontrolünde PLC deney setine enerji veriniz.
5. IsıtıcıÇalıştırma butonuna basarak tüm ısıtıcıları çalıştırınız.
6. 4-5-6No.lulsıtıcıdurdurma butonuna basarak bu ısıtıcıların devre dışı kaldığını gözlemleyiniz.
7. 2-3No.lulsıtıcıdurdurma butonuna basarak bu ısıtıcıları devre dışı bırakınız.
8. 1No.lulsıtıcıdurdurma butonuna basarak ilk ısıtıcıyı devre dışı bırakınız.
9. Herhangi bir çalışma anında TümIsıtıcılarıdurdurmaa butonuna basarak çıkışların durumunu gözlemleyiniz.
10. Deney setinin enerjisini kesip, elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

SORU

- Devrenin çalışmasını ladder diyagramı üzerinden anlatınız.

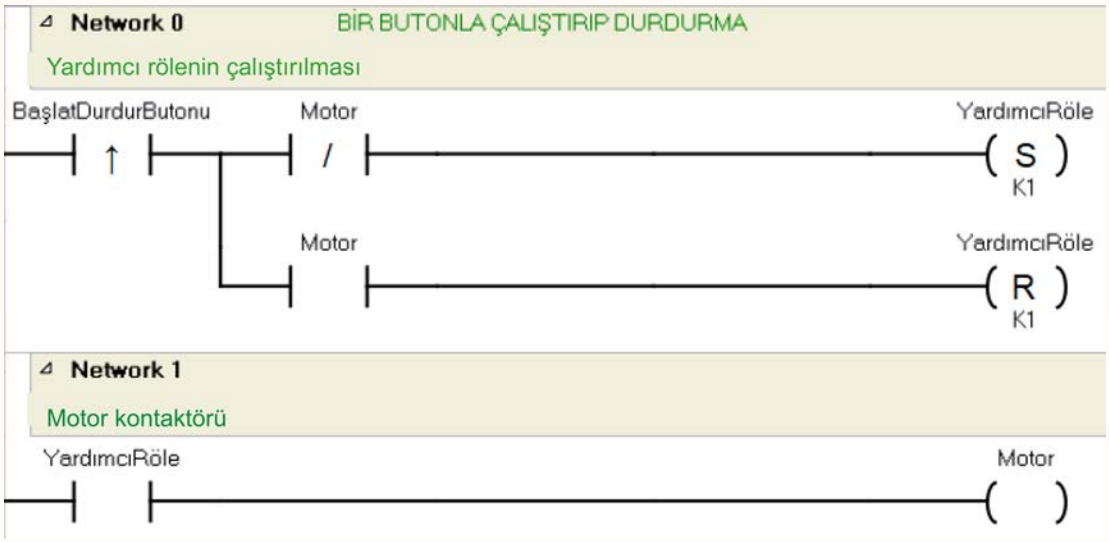
ÖĞRENCİ	**DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA**			
Adı Soyadı :	1	PLC programını oluşturdu ve PLC'ye yükledi.	20	
Sınıfı :	2	Giriş elemanlarını bağladı.	20	
Numarası :	3	Çıkış elemanlarını bağladı.	20	
ÖĞRETMEN	4	Devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

11. UYGULAMA

ÜÇ FAZLI ASENKRON MOTORUN BİR BUTONLA KONTROLÜ

AMAÇ: **Bir butonla üç fazlı asenkron motoru çalıştırıp durdurmak.**

LADDER DİYAGRAMI



MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
PLC	Transistör çıkışlı	1 adet
Buton	Ani temaslı	1 adet
Kontaktör ve röle	4 kW / 24 VDC	Birer adet
Motor	Üç fazlı asenkron	1 adet
PC ve editör programı	Yapılacak uygulamaya uygun	1 adet

ETİKET TABLOSU						
Eleman İsmi	**Veri Tipi**	**Bağlantı Tipi**	**Bağlantı Adresi**	**Adres**	**IEC Adresi**	**Açıklama**
BaşlatDurdurButonu	Bit	Giriş	I0	X0	%IX0	NA kontak
YardımcıRöle	Bit	G/Ç	M0	M0	%MX0.0	Yardımcı röle
Motor	Bit	Çıkış	Q0	Y0	%QX0	Motor kontaktörü

İŞLEM BASAMAKLARI

1. Uygulama projesini oluşturarak bilgisayarda simüle ediniz ve PLC'ye yükleyiniz.
2. PLC giriş-çıkış elemanlarının bağlantılarını yapınız.
3. Öğretmeninizin kontrolünde PLC deney setine enerji veriniz.
4. Butona basılınca motorun çalıştığını, tekrar basılınca durduğunu gözlemleyiniz.
5. Deney setinin enerjisini kesip, elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	PLC programını oluşturdu ve PLC'ye yükledi.	20	
Sınıfı :	2	Giriş elemanlarını bağladı.	20	
Numarası :	3	Çıkış elemanlarını bağladı.	20	
ÖĞRETMEN	4	Devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

12. UYGULAMA

KENAR TETİKLEME UYGULAMASI

AMAÇ: **Kenar tetikleme komutları kullanarak motor kontrolünü yapmak.**

YÖNERGE: **Bir makine üzerinde iki adet motor bulunmaktadır. Başlatma butonuna basıldığında birinci motor, butondan el çekildiğinde ise ikinci motor çalışacaktır. Durdurma butonuna basıldığında ise her iki motor da duracaktır. Sistemin çalıştırılması için gerekli PLC projesini oluşturarak uygulamasını gerçekleştiriniz.**

LADDER DİYAGRAMI



MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
PLC	Transistör çıkışlı	1 adet
Buton	Ani temaslı	2 adet
Kontaktör	4 kW	2 adet
Motor	Üç fazlı asenkron	2 adet
PC ve editör programı	Yapılacak uygulamaya uygun	1 adet

ETİKET TABLOSU						
Eleman İsmi	**Veri Tipi**	**Bağlantı Tipi**	**Bağlantı Adresi**	**Adres**	**IEC Adresi**	**Açıklama**
DurdurmaButonu	Bit	Giriş	I0	X0	%IX0	NK kontak
BaşlatmaButonu	Bit	Giriş	I1	X1	%IX1	NA kontak
Motor1	Bit	Çıkış	Q0	Y0	%QX0	Motor1 kontaktörü
Motor2	Bit	Çıkış	Q1	Y1	%QX1	Motor2 kontaktörü

İŞLEM BASAMAKLARI

1. Uygulama projesini oluşturunuz ve bilgisayarda simüle ediniz.
2. Programı kaydederek PLC'ye yükleyiniz.
3. PLC giriş-çıkış elemanlarının bağlantılarını yapınız.
4. Öğretmeninizin kontrolünde PLC deney setine enerji veriniz.
5. I1 başlatma butonu basılı tutulduğunda birinci motorun çalıştığını, buton bırakıldığında ikinci motorun da çalışmaya başladığını gözlemleyiniz.
6. I0 durdurma butonuna basarak her iki motorun da durduğunu gözlemleyiniz.
7. Deney setinin enerjisini kesip, elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

SORULAR

1. Sistemin çalışmasını ladder diyagramı üzerinden açıklayınız.
2. Kenar tetikleme nedir? Kısaca açıklayınız.

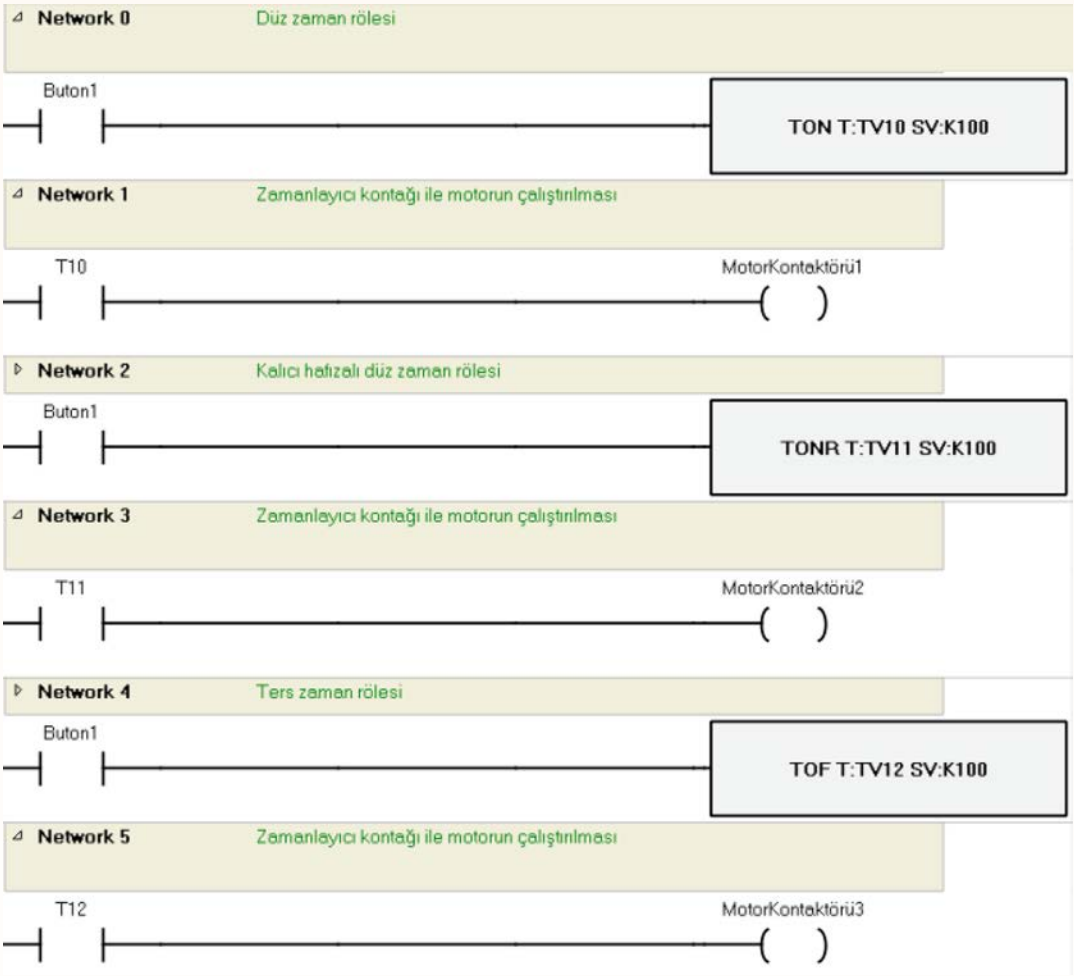
ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	PLC programını oluşturdu ve PLC'ye yükledi.	20	
Sınıfı :	2	Giriş elemanlarını bağladı.	20	
Numarası :	3	Çıkış elemanlarını bağladı.	20	
ÖĞRETMEN	4	Devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

13. UYGULAMA

ZAMANLAYICI UYGULAMALARI

AMAÇ: **Zamanlayıcı komutlarını kullanarak zaman gecikme döngüleri oluşturmak.**

LADDER DİYAGRAMI



MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
PLC	Transistör çıkışlı	1 adet
Buton	Ani temaslı	1 adet
Kontaktör	4 kW	3 adet
Motor	Üç fazlı asenkron	1 adet
PC ve editör programı	Yapılacak uygulamaya uygun	1 adet

ETİKET TABLOSU						
Eleman İsmi	**Veri Tipi**	**Bağlantı Tipi**	**Bağlantı Adresi**	**Adres**	**IEC Adresi**	**Açıklama**
Buton1	Bit	Giriş	I0	X0	%IX0	NA kontak
MotorKontaktörü1	Bit	Çıkış	Q0	Y0	%QX0	Motor1 kontaktörü
MotorKontaktörü2	Bit	Çıkış	Q1	Y1	%QX1	Motor2 kontaktörü
MotorKontaktörü3	Bit	Çıkış	Q2	Y2	%QX2	Motor3 kontaktörü

İŞLEM BASAMAKLARI

1. Uygulama projesini oluşturunuz ve bilgisayarda simüle ediniz.
2. Programı kaydederek PLC'ye yükleyiniz.
3. PLC giriş-çıkış elemanlarının bağlantılarını yapınız.
4. Öğretmeninizin kontrolünde PLC deney setine enerji veriniz.
5. Buton1'i aktif ederek TV10 değerini ve Q0 çıkışını gözlemleyiniz.
6. Buton1'i aktif ederek TV11 değerini ve Q1 çıkışını gözlemleyiniz.
7. Süre tamamlanmadan I0 girişini pasif ve tekrar aktif ederek TV11'in TV10'a göre nasıl değiştiğini gözlemleyiniz.
8. Buton1'i aktif ederek TV12 değerini ve Q2 çıkışını gözlemleyiniz.
9. Buton1'i aktif ederek Q0, Q1 ve Q2 çıkışlarının çalışma farkını gözlemleyiniz.
10. Zamanlayıcıların SV K sabitindeki değerlerini değiştirerek çıkışlarını gözlemleyiniz.
11. Deney setinin enerjisini kesip, elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

SORULAR

1. Zamanlayıcılardaki SV değişkeninin saniye olarak karşılığını belirtiniz.
2. TON, TONR ve TOF komutları arasındaki farkları belirtiniz.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	PLC programını oluşturdu ve PLC'ye yükledi.	20	
Sınıfı :	2	Giriş elemanlarını bağladı.	20	
Numarası :	3	Çıkış elemanlarını bağladı.	20	
ÖĞRETMEN	4	Devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

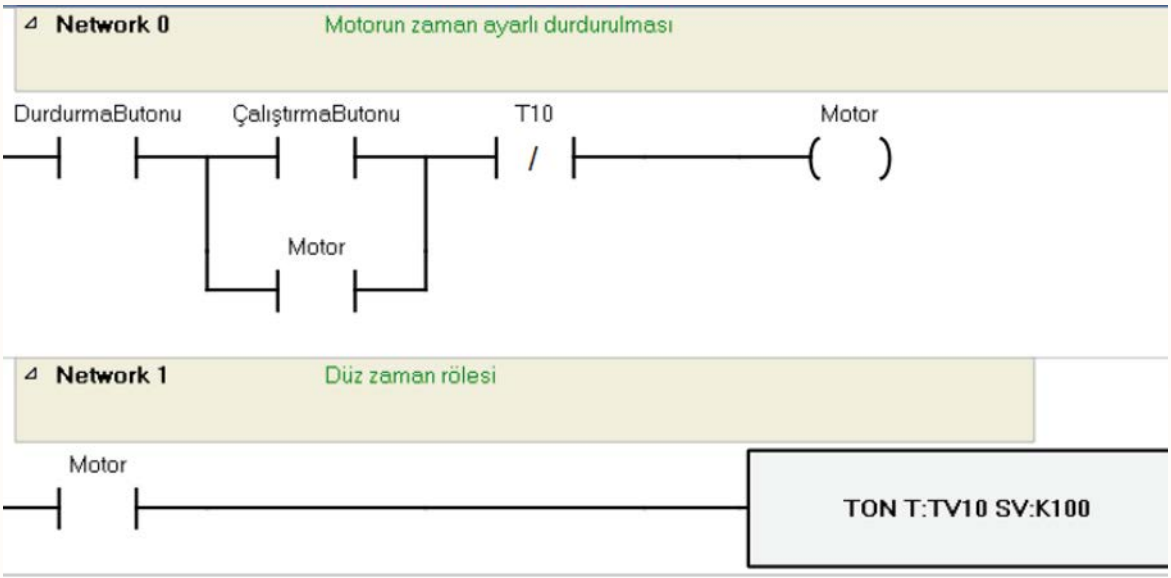
14. UYGULAMA

ÜÇ FAZLI ASENKRON MOTORUN ZAMAN AYARLI DURDURULMASI

AMAÇ: **Üç fazlı asenkron motoru zaman ayarlı durdurmak.**

YÖNERGE: **Üç fazlı asenkron motor, başlatma butonuna basıldığında çalışacak, 10 sn. sonra kendiliğinden duracaktır. Gerekli PLC projesini oluşturarak uygulamasını gerçekleştiriniz.**

LADDER DİYAGRAMI



MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
PLC	Transistör çıkışlı	1 adet
Buton	Ani temaslı	2 adet
Kontaktör	4 kW	1 adet
Motor	Üç fazlı asenkron	1 adet
PC ve editör programı	Yapılacak uygulamaya uygun	1 adet

ETİKET TABLOSU						
Eleman İsmi	**Veri Tipi**	**Bağlantı Tipi**	**Bağlantı Adresi**	**Adres**	**IEC Adresi**	**Açıklama**
ÇalıştırmaButonu	Bit	Giriş	I0	X0	%IX0	NA kontak
DurdurmaButonu	Bit	Giriş	I1	X1	%IX1	NK kontak
Motor	Bit	Çıkış	Q0	Y0	%QX0	Motor kontaktörü

İŞLEM BASAMAKLARI

1. Uygulama projesini oluşturunuz ve bilgisayarda simüle ediniz.
2. Programı kaydederek PLC'ye yükleyiniz.
3. PLC giriş-çıkış elemanlarının bağlantılarını yapınız.
4. Öğretmeninizin kontrolünde PLC deney setine enerji veriniz.
5. I0 butonuna basarak motorun çalışmasını gözlemleyiniz.
6. TON zamanlayıcısı ile ayarlanan süre sonunda motorun kendiliğinden durduğunu gözlemleyiniz.
7. Motoru tekrar çalıştırınız.
8. Ayarlanan süre dolmadan I1 butonuna basarak motorun durduğunu gözlemleyiniz.
9. Deney setinin enerjisini kesip, elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

SORU

- PLC'li zaman ayarlı durdurma devresi ile klasik zaman ayarlı durdurma kumanda devresini karşılaştırarak donanım ve çalışma yönünden farklarını belirtiniz.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	PLC programını oluşturdu ve PLC'ye yükledi.	20	
Sınıfı :	2	Giriş elemanlarını bağladı.	20	
Numarası :	3	Çıkış elemanlarını bağladı.	20	
ÖĞRETMEN	4	Devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

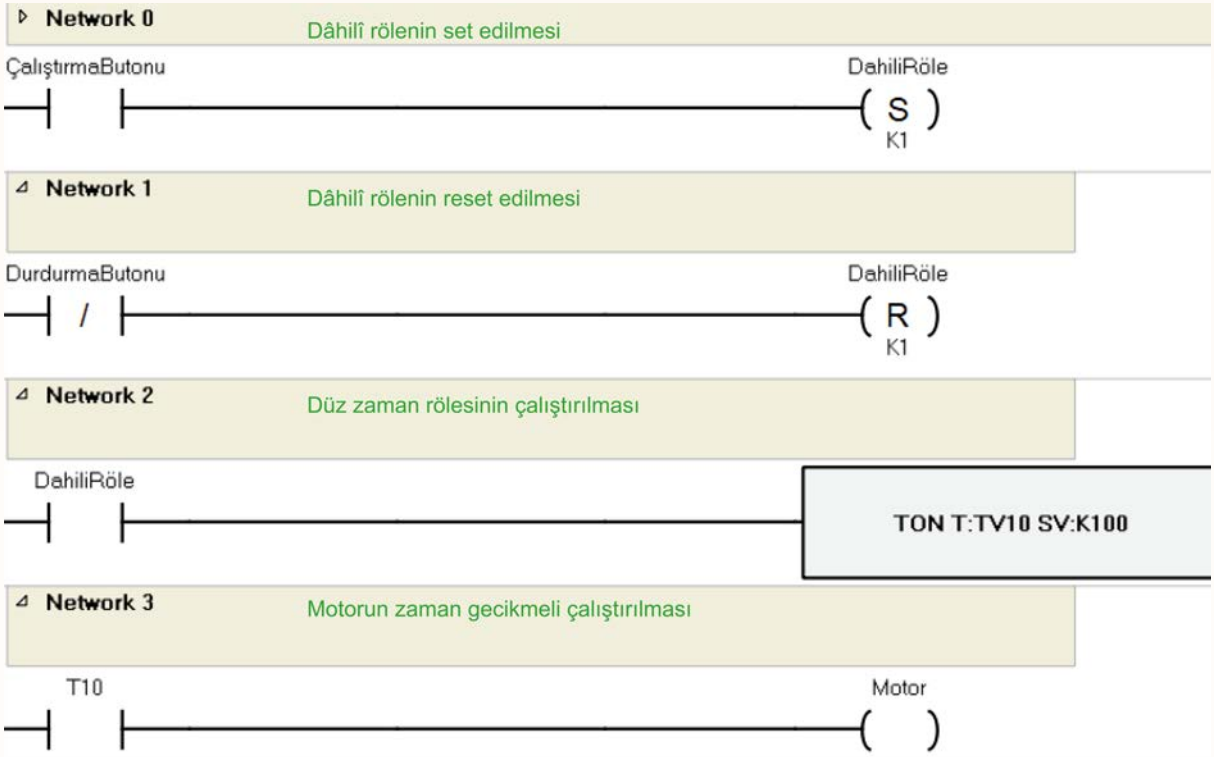
15. UYGULAMA

ÜÇ FAZLI ASENKRON MOTORUN ZAMAN AYARLI ÇALIŞTIRILMASI

AMAÇ: **Üç fazlı asenkron motoru zaman ayarlı çalıştırmak.**

YÖNERGE: **Üç fazlı asenkron motor, başlatma butonuna basıldıktan 5 sn. sonra çalışacaktır. Sistemin çalıştırılması için gerekli PLC projesini oluşturup çalıştırınız.**

LADDER DİYAGRAMI



MALZEME LİSTESİ

Adı	Özelliği	Miktarı
PLC	Transistör çıkışlı	1 adet
Buton	Ani temaslı	2 adet
Kontaktör	4 kW	1 adet
Röle	24 VDC	1 adet
Motor	Üç fazlı asenkron	1 adet
PC ve editör programı	Yapılacak uygulamaya uygun	1 adet

ETİKET TABLOSU

Eleman İsmi	Veri Tipi	Bağlantı Tipi	Bağlantı Adresi	Adres	IEC Adresi	Açıklama
ÇalıştırmaButonu	Bit	Giriş	I0	X0	%IX0	NA kontak
DurdurmaButonu	Bit	Giriş	I1	X1	%IX1	NK kontak
Motor	Bit	Çıkış	Q0	Y0	%QX0	Motor kontaktörü
DahiliRöle	Bit	G/Ç	M0	M0	%MX0.0	Dâhilî röle

İŞLEM BASAMAKLARI

1. Uygulama projesini oluşturunuz ve bilgisayarda simüle ediniz.
2. Programı kaydederek PLC'ye yükleyiniz.
3. PLC giriş-çıkış elemanlarının bağlantılarını yapınız.
4. Öğretmeninizin kontrolünde PLC deney setine enerji veriniz.
5. ÇalıştırmaButonu'na (I0) butonuna basarak ayarlanan süre kadar bekleyiniz ve sürenin sonunda motorun kendiliğinden çalıştığını gözlemleyiniz.
6. I1 butonuna basarak motoru durdurunuz.
7. Deney setinin enerjisini kesip, elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

SORU

- Sistemin çalışmasını ladder diyagramı üzerinden açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	PLC programını oluşturdu ve PLC'ye yükledi.	20	
Sınıfı :	2	Giriş elemanlarını bağladı.	20	
Numarası :	3	Çıkış elemanlarını bağladı.	20	
ÖĞRETMEN	4	Devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

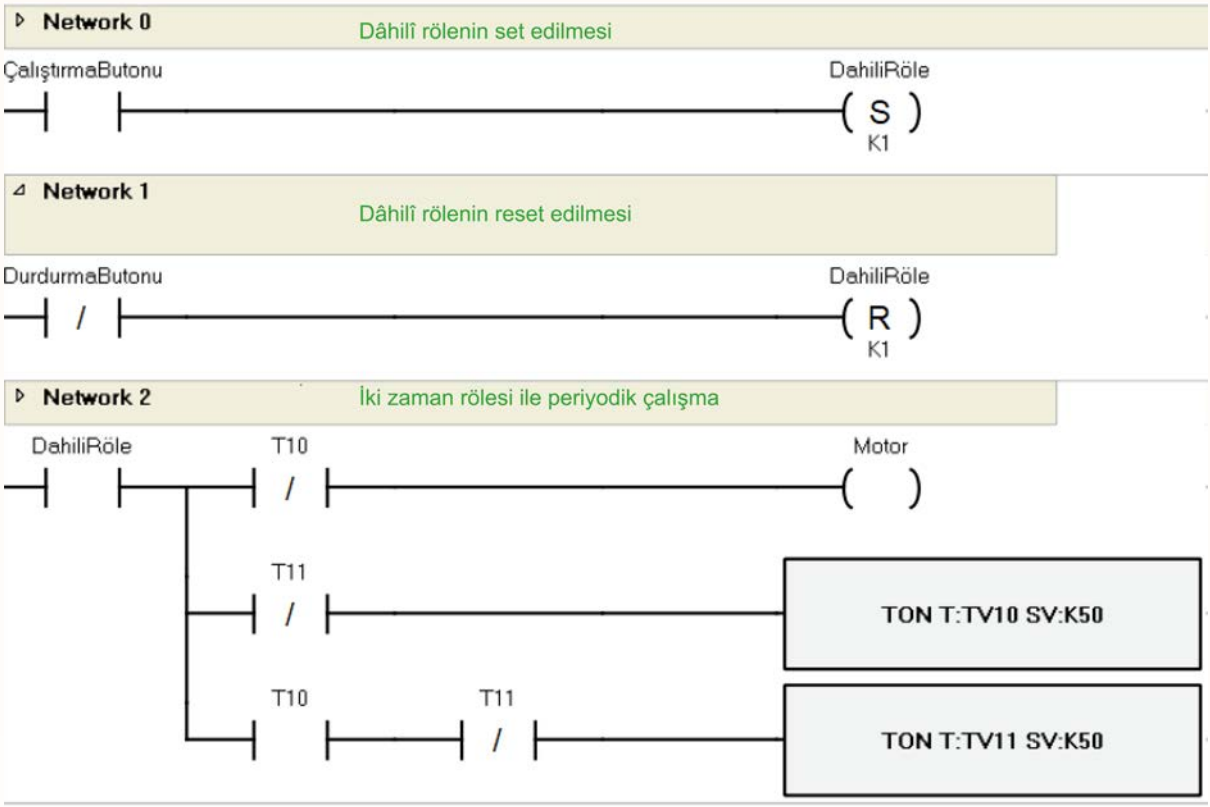
16. UYGULAMA

ÜÇ FAZLI ASENKRON MOTORUN PERİYODİK ÇALIŞTIRILMASI

AMAÇ: **Üç fazlı asenkron motoru periyodik olarak çalıştırmak.**

YÖNERGE: **Üç fazlı bir asenkron motor 5 sn. çalışacak ve 5 sn. duracaktır. Sistemin çalışması durdurma butonuna basılıncaya kadar bu şekilde devam edecektir. Sistemin çalıştırılması için gerekli PLC projesini oluşturarak uygulamasını gerçekleştiriniz.**

LADDER DİYAGRAMI



MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
PLC	Transistör çıkışlı	1 adet
Buton	Ani temaslı	2 adet
Kontaktör	4 kW	1 adet
Röle	24 VDC	1 adet
Motor	Üç fazlı asenkron	1 adet
PC ve editör programı	Yapılacak uygulamaya uygun	1 adet

ETİKET TABLOSU						
Eleman İsmi	**Veri Tipi**	**Bağlantı Tipi**	**Bağlantı Adresi**	**Adres**	**IEC Adresi**	**Açıklama**
ÇalıştırmaButonu	Bit	Giriş	I0	X0	%IX0	NA kontak
DurdurmaButonu	Bit	Giriş	I1	X1	%IX1	NK kontak
Motor	Bit	Çıkış	Q0	Y0	%QX0	Motor kontaktörü
DahiliRöle	Bit	G/Ç	M0	M0	%MX0.0	Dâhilî röle

İŞLEM BASAMAKLARI

1. Uygulama projesini oluşturunuz ve bilgisayarda simüle ediniz.
2. Programı kaydederek PLC'ye yükleyiniz.
3. PLC giriş-çıkış elemanlarının bağlantılarını yapınız.
4. Öğretmeninizin kontrolünde PLC deney setine enerji veriniz.
5. I0 butonuna basarak motorun ayarlanan süreler kadar çalışıp durduğunu gözlemleyiniz.
6. I1 durdurma butonuna basarak periyodik çalışmayı sona erdiriniz.
7. Deney setinin enerjisini kesip, elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

SORULAR

1. Sistemin çalışmasını ladder diyagramı üzerinden açıklayınız.
2. Periyodik çalışma uygulama alanlarına örnekler veriniz.

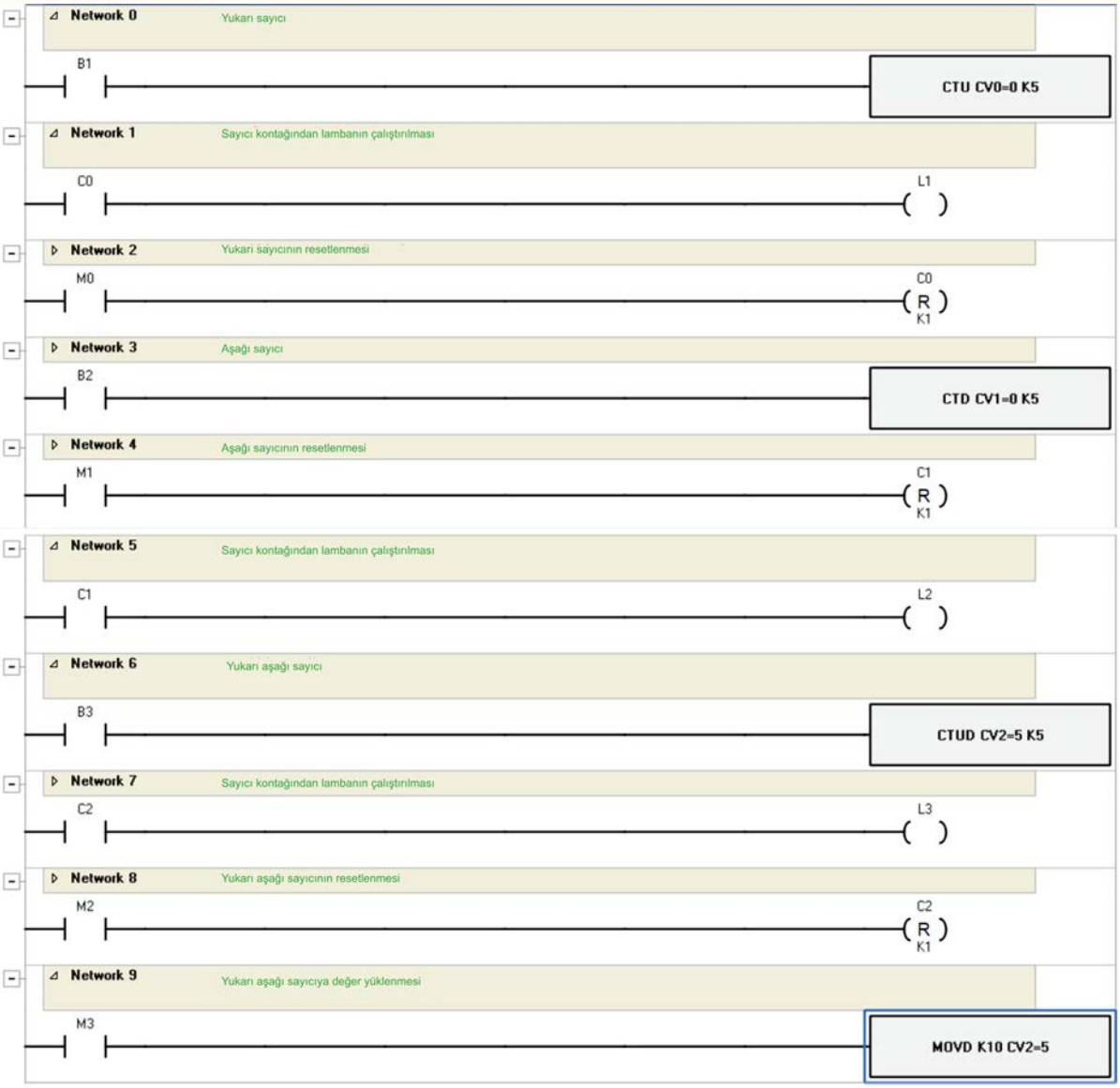
ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	PLC programını oluşturdu ve PLC'ye yükledi.	20	
Sınıfı :	2	Giriş elemanlarını bağladı.	20	
Numarası :	3	Çıkış elemanlarını bağladı.	20	
ÖĞRETMEN	4	Devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

17. UYGULAMA

SAYICI UYGULAMALARI

AMAÇ: **Sayıcı komutlarını kullanarak sayma işlemleri gerçekleştirmek.**

LADDER DİYAGRAMI



MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
PLC	Transistör çıkışlı	1 adet
Buton	Ani temaslı	3 adet
Lamba	24 VDC	3 adet
PC ve editör programı	Yapılacak uygulamaya uygun	1 adet

ETİKET TABLOSU						
Eleman İsmi	**Veri Tipi**	**Bağlantı Tipi**	**Bağlantı Adresi**	**Adres**	**IEC Adresi**	**Açıklama**
B1	Bit	Giriş	I0	X0	%IX0	NA kontak
B2	Bit	Giriş	I1	X1	%IX1	NA kontak
B3	Bit	Giriş	I2	X2	%IX2	NA kontak
L1	Bit	Çıkış	Q0	Y0	%QX0	Lamba1
L2	Bit	Çıkış	Q1	Y1	%QX1	Lamba2
L3	Bit	Çıkış	Q2	Y2	%QX2	Lamba3
DahiliRöle	Bit	G/Ç	M0	M0	%MX0.0	Dâhilî röle
DahiliRöle	Bit	G/Ç	M1	M1	%MX0.1	Dâhilî röle
DahiliRöle	Bit	G/Ç	M2	M2	%MX0.2	Dâhilî röle
DahiliRöle	Bit	G/Ç	M3	M3	%MX0.3	Dâhilî röle

İŞLEM BASAMAKLARI

1. Uygulama projesini oluşturunuz ve bilgisayarda simüle ediniz.
2. Programı kaydederek PLC'ye yükleyiniz.
3. PLC giriş-çıkış elemanlarının bağlantılarını yapınız.
4. Öğretmeninizin kontrolünde PLC deney setine enerji veriniz.
5. B1 butonuna basıp bırakarak yukarı sayma değerini arttırınız. Bu değişimi CV0 üzerinden gözlemleyiniz.
6. Yukarı sayıcı, sayma değerine ulaştığında C0 ve Q0 çıkışının aktif olduğunu gözlemleyiniz.
7. M0 aktif edildiğinde CV0 sayıcısının sıfırlandığını gözlemleyiniz.
8. M1 aktif edildiğinde CV1 sayıcısına sabit değerinin yüklendiğini gözlemleyiniz.
9. B2 butonuyla aşağı sayma değerini azaltınız. Bu değişimi CV1 üzerinden gözlemleyiniz.
10. Aşağı sayıcı değeri sıfıra ulaştığında C1 ve Q1 çıkışının aktif olduğunu gözlemleyiniz.
11. B3 butonuyla yukarı sayma değerini artırınız. Bu değişimi CV2 üzerinden gözlemleyiniz.
12. Aşağı yukarı sayıcı, sayma değerine ulaştığında C2 ve Q2 çıkışının aktif olduğunu gözlemleyiniz.
13. M3 aktif edildiğinde CV2 sayıcısına sabit değerinin yüklendiğini gözlemleyiniz.
14. B3 butonuna basıp bırakarak CV2 değerinin azaldığını gözlemleyiniz.
15. Aşağı yukarı sayıcı, sayma değerine ulaştığında C2 ve Q2 çıkışının aktif olduğunu gözlemleyiniz.
16. M2 aktif edildiğinde CV2 sayıcısının sıfırlandığını gözlemleyiniz.
17. PC ile PLC arasında yapılan çevrim içi bağlantıyı sonlandırarak çalışmalarınızı kaydediniz.
18. Deney setinin enerjisini kesip, elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

SORULAR

1. Reset girişleri hangi amaçla kullanılır?
2. Her bir sayıcının çalışmasını ladder diyagramı üzerinden ayrı ayrı açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	PLC programını oluşturdu ve PLC'ye yükledi.	20	
Sınıfı :	2	Giriş elemanlarını bağladı.	20	
Numarası :	3	Çıkış elemanlarını bağladı.	20	
ÖĞRETMEN	4	Devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

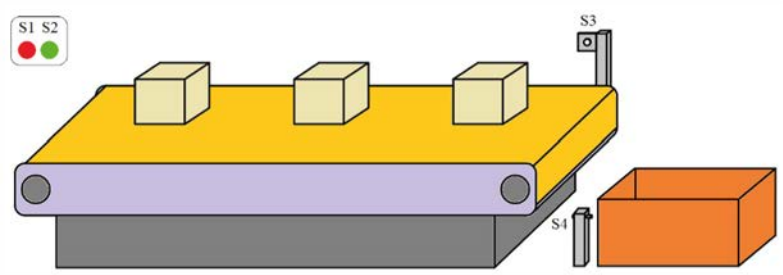
18. UYGULAMA

KONVEYÖR BANT UYGULAMASI

AMAÇ: Sayma işlemlerine bağlı olarak konveyör bant motorunu çalıştırmak.

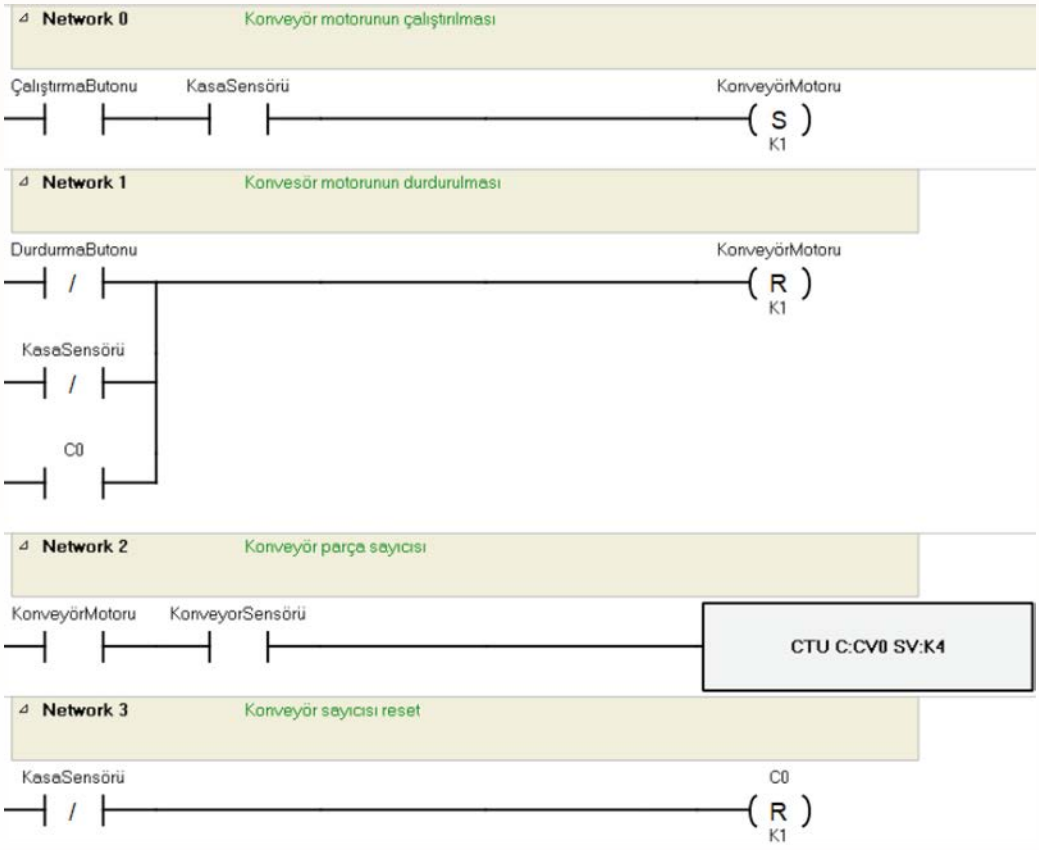
YÖNERGE: Bir konveyör sistemi aşağıdaki şartlarda çalıştırılmak istenmektedir.

- Konveyör, başlatma butonu ile çalışmaya başlayacaktır.
- Konveyörün çalışması başlatma butonuna basılması ve istifleme kasasının yerinde olmasına bağlıdır.
- Paket sayısı 4 olduğunda konveyör kendiliğinden duracaktır.
- Yeniden başlatma işlemi için yeni bir istifleme kasası yerine konularak başlatma butonuna basılmalıdır.
- Sistemin durdurulması stop butonu ile gerçekleştirilecektir.
- Sistemin çalıştırılması için gerekli PLC projesini oluşturarak uygulamasını gerçekleştiriniz.



Görsel 7.7: Sistem çalışma şeması

LADDER DİYAGRAMI



MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
PLC	Transistör çıkışlı	1 adet
Buton	Ani temaslı	2 adet
Sensör	PNP NA	2 adet
Kontaktör	4 kW	1 adet
Motor	Üç fazlı asenkron	1 adet
PC ve editör programı	Yapılacak uygulamaya uygun	1 adet

ETİKET TABLOSU						
Eleman İsmi	**Veri Tipi**	**Bağlantı Tipi**	**Bağlantı Adresi**	**Adres**	**IEC Adresi**	**Açıklama**
ÇalıştırmaButonu	Bit	Giriş	I0	X0	%IX0	NA kontak
DurdurmaButonu	Bit	Giriş	I1	X1	%IX1	NK kontak
KonveyörSensörü	Bit	Giriş	I2	X2	%IX2	PNP NA sensör
KasaSensörü	Bit	Giriş	I3	X3	%IX3	PNP NA sensör
KonveyörMotoru	Bit	Çıkış	Q0	Y0	%QX0	Motor kontaktörü

İŞLEM BASAMAKLARI

1. Uygulama projesini oluşturunuz ve bilgisayarda simüle ediniz.
2. Programı kaydederek PLC'ye yükleyiniz.
3. PLC giriş-çıkış elemanlarının bağlantılarını yapınız.
4. Öğretmeninizin kontrolünde PLC deney setine enerji veriniz.
5. I3 kasa sensörüne algılama yaptırarak I0 çalıştırma butonuna basınız.
6. Bant motorunun çalışmasını gözlemleyiniz.
7. Konveyör sensörüne malzeme algılaması yaptırarak sayıcının sayma değerinin arttığını gözlemleyiniz.
8. Sayıcı, sayma değerine ulaştığında ya da kasa yerinden alındığında bant motorunun durduğunu gözlemleyiniz.
9. Deney setinin enerjisini kesip, elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

SORU

- Sistemin çalışmasını ladder diyagramı üzerinden açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	PLC programını oluşturdu ve PLC'ye yükledi.	20	
Sınıfı :	2	Giriş elemanlarını bağladı.	20	
Numarası :	3	Çıkış elemanlarını bağladı.	20	
ÖĞRETMEN	4	Devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

7.2. DOKUNMATİK EKRAN UYGULAMALARI

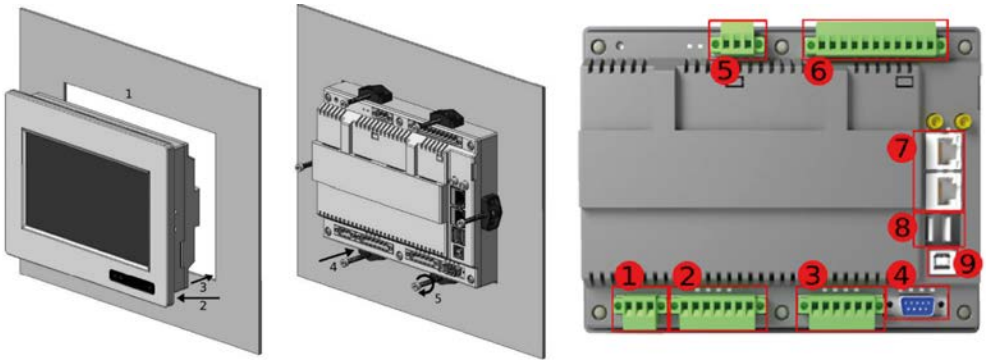
Otomasyon sistemlerinde, sistem ve kullanıcı arasındaki iletişimi sağlayan donanıma **operatör paneli** denir. Operatör paneli, kontrol alanından alınan verilerin bir ekran üzerinden takibini ve değiştirilmesini sağlar. Operatör panelinin diğer bir adı da **HMI**'dir (Human Machine Interface).

Kumanda projeleri büyüdükçe PLC ile beraber kullanılan buton, sinyal lambası, analog ve dijital gösterge gibi elemanların sayısı artar. Bunların yerine PLC ve sürücüler ile beraber operatör panelleri kullanılır.

Operatör panelleri; otomotiv, makine, endüstriyel robotlar, petrokimya, plastik, ahşap, tekstil gibi endüstriyel sistemlerde kullanılır. Bunun yanı sıra bina ısıtma, soğutma, havalandırma, aydınlatma ve tarımsal otomasyon alanlarında da kullanılır.

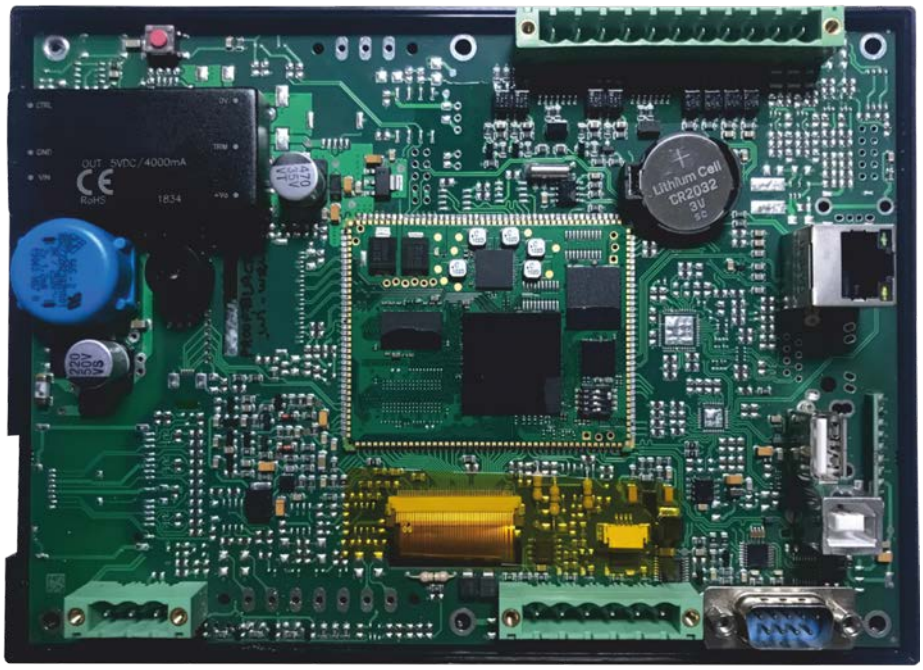
7.2.1.Operatör Panelinin Yapısı

Küçük bir endüstriyel bilgisayar olan operatör paneli; LCD panel, mikroişlemci, hafıza elemanları, haberleşme yongaları ve bağlantı konnektörlerinden oluşur. LCD panel, cihazla ilgili operasyonel işlemlerin yapıldığı önemli bir bileşendir ve dokunmatik özelliğe sahiptir (Görsel 7.8).



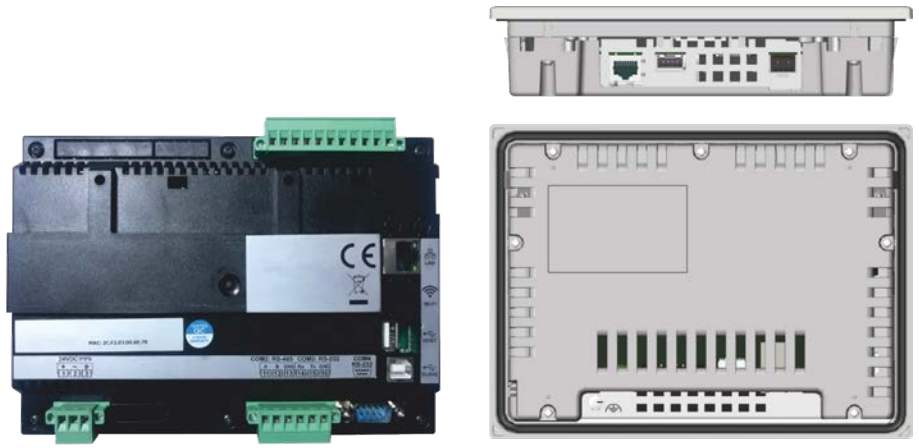
Görsel 7.8: Çeşitli operatör panelleri

Panel içyapısında mikroişlemci, hafıza elemanları ve haberleşme yongaları bulunur (Görsel 7.9).



Görsel 7.9: Operatör panelinin içyapısı

Operatör panellerinde pek çok haberleşme portu bulunur. Normal panellerde RS 232, RS 485, USB gibi haberleşme portları bulunurken, gelişmiş panellerde bu portlara ek olarak ETHERNET, WIFI, CANBUS gibi portlar da bulunur. Paneller bu portlarda MODBUS, MODBUS TCP, CANBUS, PROFINET, ETHERCAT, OPC UA, MQTT, KNX gibi haberleşme protokollerinden bir veya birden fazla haberleşme protokolünü çalıştırabilir. Bu portlar genelde panelin arka kısmındadır (Görsel 7.10).



Görsel 7.10: Operatör panelinin haberleşme portları ve bağlantı klemensleri

7.2.2. Operatör Panelinin Çalışması

Operatör panellerinin işletim sistemi ROM hafızada yüklüdür. Kullanıcı, işletim sitemine müdahale etmez ancak ekran parlaklığı, dokunma hassasiyeti, sistemin tarih ve saati, ekran koruyucusu gibi ayarları yapabilir.

Operatör panelleri PLC'ler gibi programlanabilir. Bilgisayarda hazırlanan program, izin verilen haberleşme portu üzerinden panele yüklenir. Panele enerji verildiğinde sırası ile işletim sistemi, arabirim yazılımı ve hazırlanan program çalıştırılır. Programa göre otomasyon sistemi cihazları (giriş-çıkış modülleri, PLC, motor sürücüsü vb.) istenen işlemi gerçekleştirir.

7.2.3. Operatör Paneli Çeşitleri

Operatör panelleri genelde ekran boyutuna göre çeşitlendirilir. Ekran boyutları arttıkça çözünürlük de artar. Ekranlarda dokunmatik algılama için kapasitif algılamadan çok, dirençli algılama tercih edilir. Bu sebeple ekrana az da olsa kuvvet uygulamak gerekir. En çok kullanılan boyutlar ve çözünürlükleri şunlardır:

- 4.3" (480x272)
- 7" (800x480)
- 10.1" (1024x600)
- 12.1" (1024x768)
- 15.6" (1366x768)
- 18.5" (1920x1080)
- 21.5" (1920x1080)

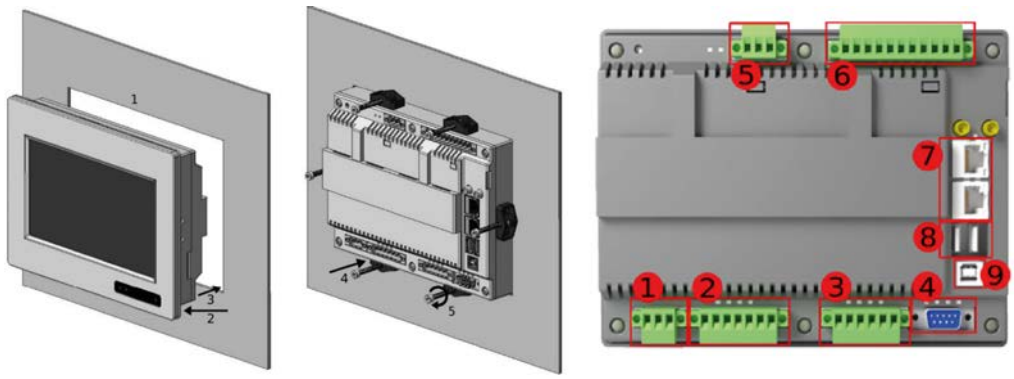
HMI'ler çoğunlukla dokunmatik olmakla birlikte tuşlu modelleri de vardır. Bu modeller CNC gibi uygulamalarda tercih edilir. Çok çekirdekli CPU'ların operatör panellerinde kullanılmasıyla birlikte PLC işlemlerini de yapabilen operatör paneli çeşitleri geliştirilmiştir.

7.2.4. Operatör Panelinin Montajı ve Panel Bağlantıları

Operatör panellerinin mobil modelleri olduğu gibi pano kapağına monte edilen sabit modelleri de vardır. Sabit paneller daha çok kullanılır. Bu modellerin montaj işlem basamakları şöyledir:

1. Montajın yapılacağı panel kesitini uygun ölçülerde hazırlayınız.
2. Cihazın ön paneli üzerinde bulunan sızdırmazlık contasının takılı olduğundan emin olunuz.
3. Cihazı panel üzerindeki kesite yerleştiriniz.
4. Montaj aparatlarını yuvalarına yerleştiriniz.
5. Cihazın aparat vidalarını sıkarak cihazı panele sabitleyiniz.

Operatör paneli; besleme, haberleşme ve kontrol amacıyla kullanılan çeşitli bağlantı birimlerine sahiptir (Görsel 7.11). Bu birimlerin kullanım amaçları aşağıda verilmiştir.



Görsel 7.11: Operatör panelinin montajı ve panelin bağlantısı

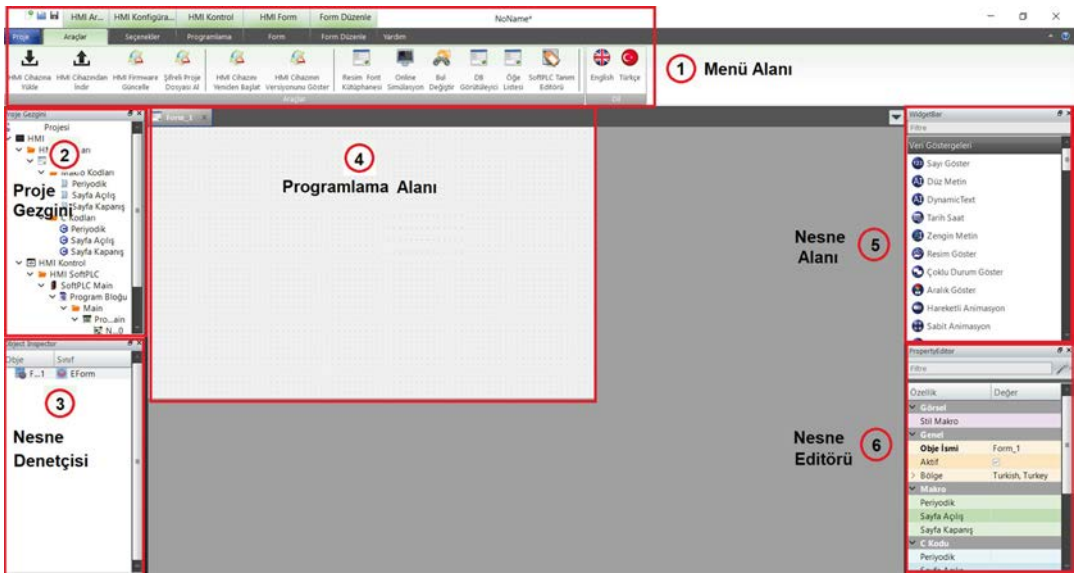
1	Besleme ucu ve toprak hattı bağlantı klemensi. Panellerde besleme gerilimi olarak 24 VDC kullanılır.
2	PLC, bilgisayar, sürücü gibi cihazlarla haberleşme için kullanılan COM1 portu
3	COM2 ve COM3 portu
4	COM4 portu
5	Kullanım dışı
6	Giriş-çıkış bağlantı klemensleri
7	Haberleşme ve internet için kullanılan ethernet portları
8	Klavye, mouse, barkod okuyucu, yazıcı gibi cihaz bağlantıları için USB host portu
9	Bilgisayar bağlantısı için USB aygıt portu

7.2.5. Operatör Panelinin Programlama Editörü Menüleri

Her programlama editörü menüsü kendine özel olmakla birlikte benzer isim ve işlevlere sahiptir. Bu menüler yardımıyla yapılandırma ve ayarlar yapılır. Menüler, örnek bir editör üzerinden anlatılacaktır.

7.2.5.1. Ana Ekran

Programlama için gerekli bölümlerin bulunduğu editör karşılama ekranıdır. Bu ekranda Menü Alanı, Proje Gezgini, Nesne Denetçisi, Programlama Alanı, Nesne Alanı ve Nesne Editörü yer alır (Görsel 7.12).



Görsel 7.12: Programlama editörü ana ekranı

7.2.5.2. Menü Alanı

Programlama için gerekli menü ve menü seçeneklerinin bulunduğu alandır.

Proje Menüsü: Oluşturulacak proje ile ilgili işlemlerin yapıldığı menüdür. Yeni, Aç, Kaydet, Farklı Kaydet, Kapat, Proje Ayarları, Çıkış gibi seçeneklerden oluşur (Görsel 7.13).



Yeni proje açar.
Kayıtlı projeyi açar.
Projeyi kaydeder.
Projeyi farklı kaydeder.
Projeyi kapatır.
Geçmiş projeleri açar.
Editörü kapatır.
Proje ayarlarını yapar.

Görsel 7.13: Proje menüsü

Araçlar Menüsü: Araçlar menüsü, HMI kontrolü ile ilgili işlemlerin yapıldığı menüdür. HMI'ye program yükleme, HMI'den program alma, HMI'yi yeniden başlatma, online simülasyon gibi işlemler bu menü ile yapılabilir (Görsel 7.14).



Görsel 7.14: Araçlar menüsü

Seçenekler Menüsü: Seçenekler menüsü, HMI kontrolü ile ilgili işlemlerin yapıldığı menüdür. HMI'ye program yükleme, HMI'den program alma, HMI'yi yeniden başlatma, online simülasyon gibi işlemler bu menü ile yapılabilir (Görsel 7.15).



Görsel 7.15: Seçenekler menüsü

Programlama Menüsü: Programlama menüsü, makrolar ve programlama ile ilgili işlemlerin yapıldığı menüdür (Görsel 7.16).



Görsel 7.16: Programlama menüsü

Form Menüsü: Form menüsü, çalışma sayfası ile ilgili işlemlerin yapıldığı menüdür. Sayfa ekleme-silme, ön izleme gibi işlemler bu menü ile yapılabilir (Görsel 7.17).



Görsel 7.17: Form menüsü

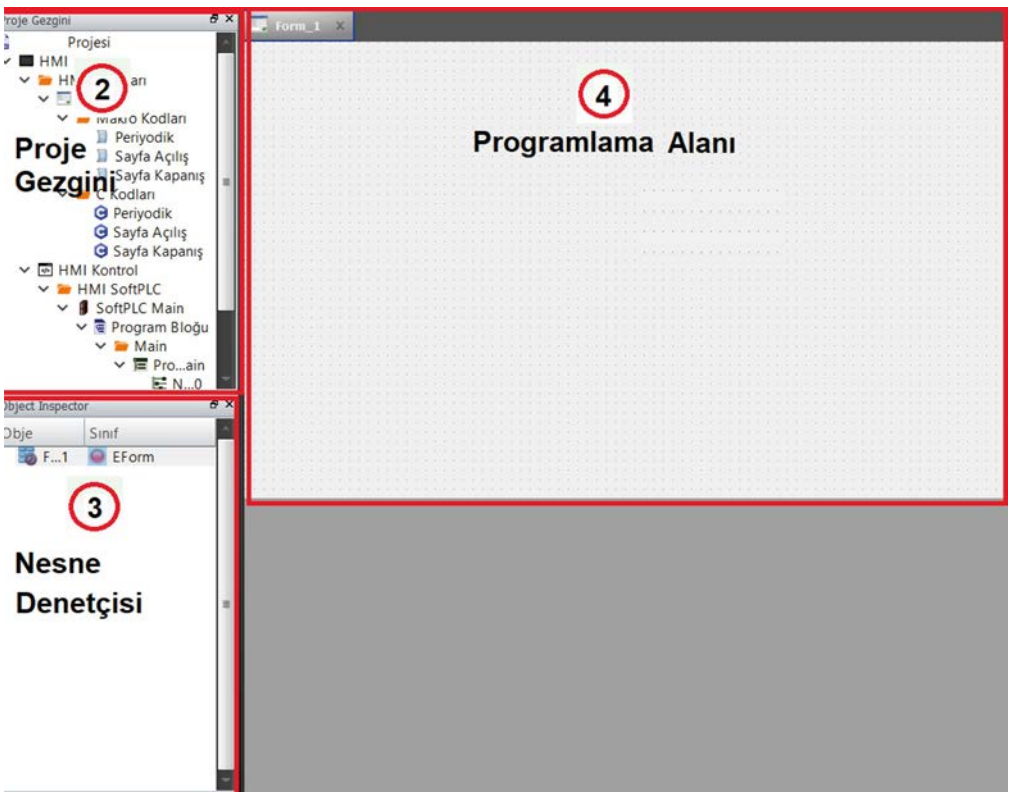
Form Düzenle Menüsü: Form Düzenle menüsü, çalışma sayfalarının düzenlenmesi işlemlerinin yapıldığı menüdür. Kes, kopyala, yapıştır, ileri al, geri al gibi düzenleme işlemleri bu menü ile yapılabilir (Görsel 7.18).



Görsel 7.18: Form Düzenle menüsü

7.2.5.3. Proje Gezgini, Nesne Denetçisi ve Programlama Alanı

Oluşturulan proje ile ilgili aşamaların gösterildiği alandır (Görsel 7.19).

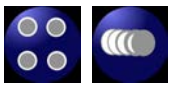
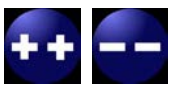
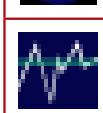
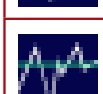


Görsel 7.19: Proje Gezgini, Nesne Denetleyicisi ve Programlama Alanı

7.2.5.4. Nesne Alanı

Form sayfasında programlama için kullanılacak olan nesnelerin listelendiği alandır. Veri Göstergeleri, Butonlar, SVG Butonlar, Veri Giriş, Göstergeler, Diğer ve Şekiller olmak üzere yedi bölümden oluşur. Program için gerekli nesne, bu bölümden sürükle-bırak yöntemiyle programlama alanına alınır. **Filtre** kısmından nesne araması yapılabilir. En çok kullanılan nesneler Tablo 7.11'de verilmiştir.

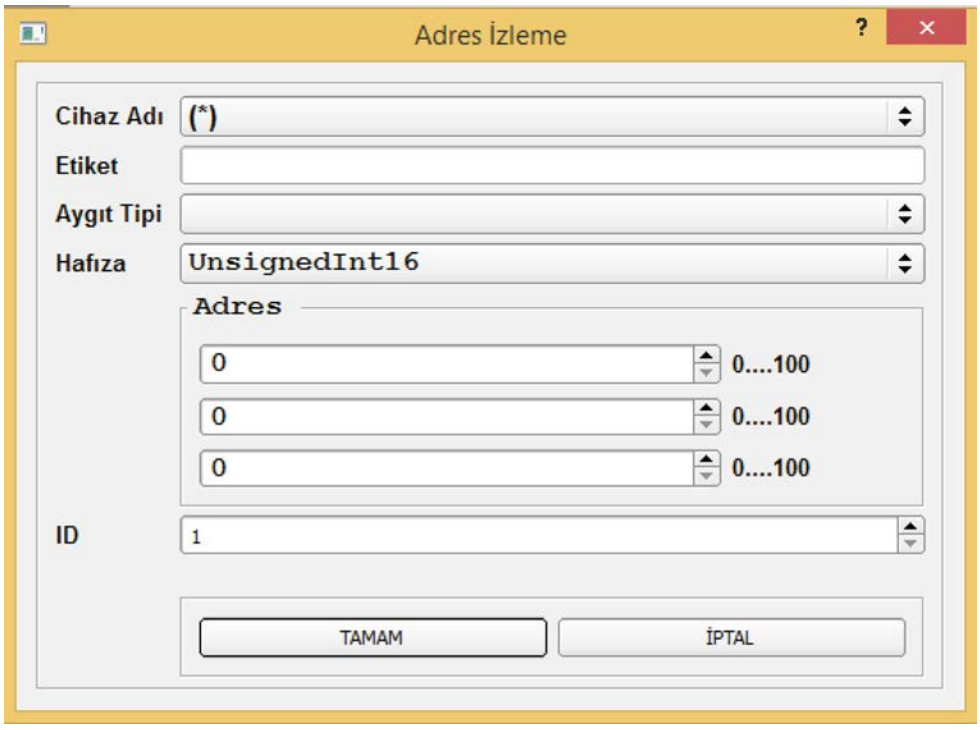
Tablo 7.11: Çok Kullanılan Nesne Alanı Elemanları

Komut	Açıklama	Ladder Diyagramı
	Sayı göster	Belirtilen adresi okuyup rakam olarak görüntüler.
	Düz metin	Formda bir metin değerini görüntüler.
	Tarih/saat	Formda tarih ve saati görüntüler.
	Hareketli ve sabit animasyon	Atanan hareketli ve sabit animasyon görüntülenir.
	LED	Okuma adresindeki değerin durumuna göre renk değişimi görüntülenir.
	Sayıcı	Minimum ve maksimum değerler arasında istenilen miktarda artış veya azalış değeri görüntülenir.
	Push ve kalıcı buton	Butona basıldığında ayarlanan adresin durumu ON, buton bırakıldığında OFF'tur.
	Set ve reset buton	Set butona basıldığında ayarlanan adresin durumu ON, Reset butona basıldığında OFF'tur.
	Değer arttır ve değer azalt	Butona basıldığında belirtilen adresteki değere sabit sayı eklenir veya çıkarılır. Sabit sayı, ayarlar bölümünde düzenlenir.
	Analogmetre, Analogmetre 2, Sirküler bar	Minimum ve maksimum değerler arasında istenilen miktarda artış ve azalış görüntülenir. Ayarlar bölümünde değerin üst ve alt limiti, skala ve ibrenin rengi ayarlanır.
	İlerleme çubuğu, tank	Okuma adresindeki değer değişimi görüntülenir. Ayarlar bölümünde üst ve alt limit renklendirilebilir.
	Termometre, Ampermetre, Takometre, Manometre	Analogmetre ile aynı işlevi görür.
	Datalog alanı	Kaydı tutulan verilerin gösterimi için kullanılan liste ve grafik alanıdır. Anlık ve geçmiş verilerin grafik ve listeleri alınabilir.
	Alarm alanı	Alarm tanımlarının verileri ve gösterimleri için kullanılan alandır. Anlık veya geçmiş alarm durum ve listeleri alınabilir.

7.2.5.5. Nesne Editörü

Programlama alanına alınan nesnelerin özelliklerinin düzenlendiği alandır. Her nesnenin kendine has özellikleri vardır. Bu sebeple nesne editörü seçenekleri, seçilen nesneye göre değişiklik gösterir. Uygun düzenlemeyle nesnelere işlevsellik ve görsellik kazandırılır.

Adres: Veri Göstergeleri, Butonlar, Veri Girişleri ve Göstergeler nesnelerinde adresleme özelliği kullanılır. Bu nesnelerden biri programlama alanına alındıktan sonra **Okuma/Yazma/Gizleme Adresi** bölümlerinden uygun olan bölümdeki üç noktaya tıklanarak açılan **Adres İzleme** penceresinden gerekli tanımlamalar yapılır (Görsel 7.20).



Görsel 7.20: Adres İzleme penceresi

Cihaz Adı bölümündeki **Internal_io** seçeneği dâhilî giriş-çıkış cihazlarını içerir ve hafıza alanı varsayılan değeri bit'tir. **Internal_Memory** seçeneği ise dâhilî hafıza cihazlarını içerir ve hafıza alanı varsayılan değeri unsignedInt16'dır.

Genel: Genel özellikler tüm araçlarda kullanılır. Aktif, Aktif Durum ve Konumlandırma gibi seçeneklerle ilgili ayarlamalar yapılır.

Veri: Veri Göstergeleri, Veri Girişleri, Göstergeler ve Diğer Araçlar bölümündeki araçlarda veri özelliği kullanılır. Okuma adresi alanındaki değerle ilgili ayarlamalar yapılır.

Giriş: Veri girişi araçlarında giriş özelliği kullanılır. Buton adı, buton sayısı gibi değerler ayarlanır.

Buton: Buton ve SVG araçlarında buton özelliği kullanılır. Buton Tipi, Sayfa Fonksiyonu, Adres Fonksiyonu, Durum Tipi gibi seçeneklerle butonların özellikleri ayarlanır.

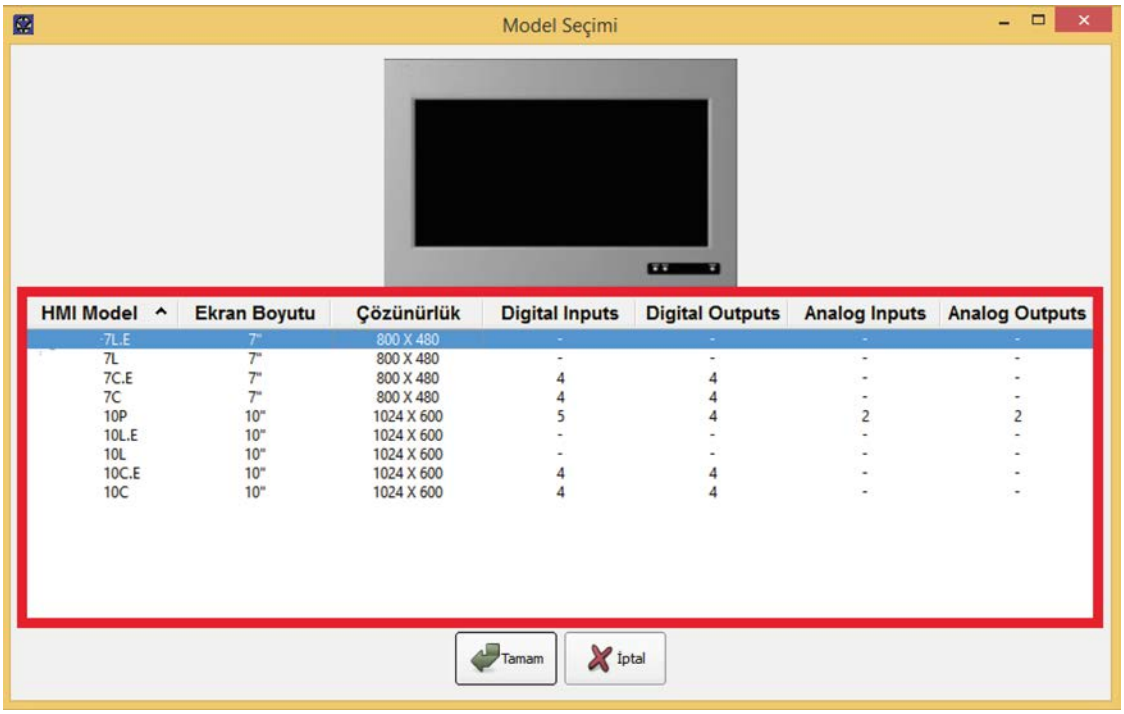
Özel: Veri Girişi, Göstergeler ve Diğer Araçlarda bu özellik kullanılır. Görüntüleme Türü, Yön ve Hız gibi seçeneklerle ilgili nesnelerin özellikleri ayarlanır.

Görsel: Tüm nesnelerin görsel özelliklerinin ayarlanması için kullanılır.

Geometri: Göstergeler hariç diğer nesnelerde koordinat ve boyut belirlemede kullanılır.

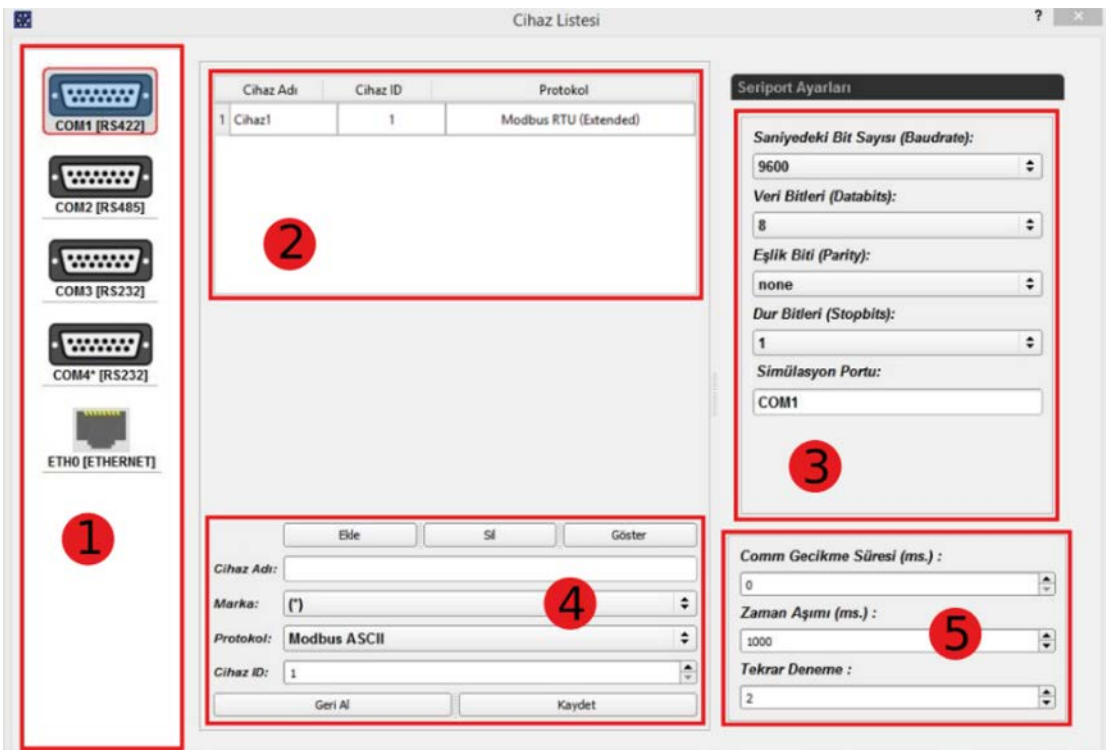
7.2.6. Projeye Panel Eklenmesi ve İletişim Ayarları

Proje oluşturmak için Proje – Yeni menü seçenekleri kullanılarak yeni bir proje oluşturulur. Yeni projenin adı Dosya Adı alanına yazılır ve kaydedilir. Proje kaydından sonra açılan penceredeki listeden uygulamada kullanılacak olan model seçimi yapılır ve panel projeye eklenir (Görsel 7.21).



Görsel 7.21: Projeye panel eklenmesi

Projeye cihaz eklemek için Seçenekler - Haberleşme ayarları seçenekleri kullanılır. Açılan pencerede eklenecek veya ekli olan cihazlarla ilgili ayar bilgileri yer alır (Görsel 7.22).



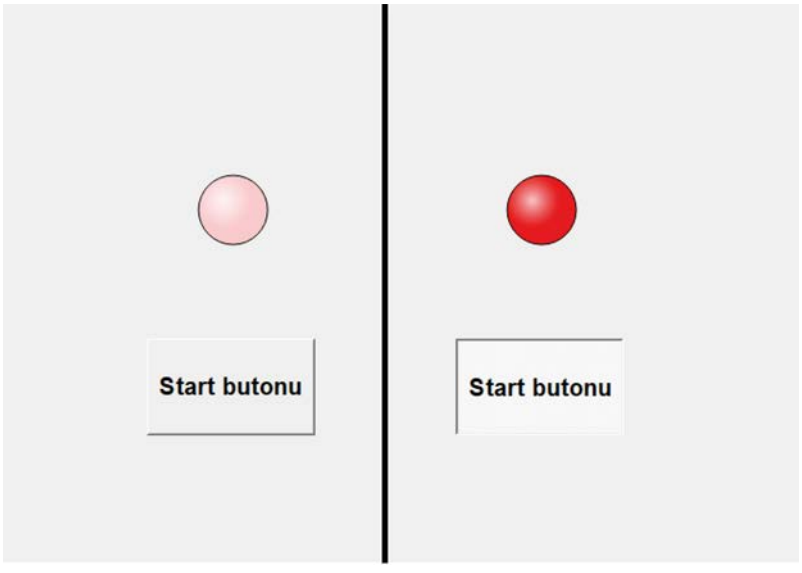
Görsel 7.22: Projeye cihaz eklenmesi ve iletişim ayarları

- 1 numaralı alandan eklenecek cihazın bağlantı noktası seçilir. Bağlantı noktası seçenekleri COM1 (RS 232), COM2 (RS 485), COM3 (RS 232), COM4 (RS 232) ve ETH0'dur (ETHERNET).
- Bağlantı noktası seçildikten sonra 4 numaralı alandan cihaz adı, marka, protokol ve cihaz ID'si bilgileri girilerek **Ekle** seçeneğine tıklanır.
- Eklenen cihaz 2 numaralı alanda listelenir ve 3 numaralı alandan seri port ayarları düzenlenir. Seri port ayarlarındaki simülasyon port alanı, online simülasyon sırasında kullanılacak bilgisayar COM portunu belirtir.
- Son olarak 5 numaralı alandan bağlantı ile ilgili seçenekler düzenlenerek **Kaydet** seçeneğine tıklanır ve cihaz bilgileri güncellenir.

7.2.7. Operatör Paneli Proje Uygulaması

Operatör panelinin programlanması için öncelikle bir proje oluşturulur. Editör menüleri yardımıyla cihaz yapılandırılır ve ilgili program hazırlanır. Hazırlanan program, seçilen haberleşme portu üzerinden operatör paneline yüklenir.

- **Proje – Yeni** menüsünden yeni bir proje açılır.
- **Dosya Adı** alanına projenin adı yazılarak proje kaydedilir.
- Proje kaydından sonra açılan pencereden HMI modeli seçilir.
- Açılan pencereden cihaz ve haberleşme ayarları yapılır.
- Programlama için gerekli formlar (sayfalar) açılır ve isimlendirilir.
- **Nesne Alanı – Butonlar** bölümünden sürükle-bırak yöntemiyle sayfaya **Push Buton** alınır.
- **Nesne Editörü - Görsel** bölümünden butonla ilgili görsel ayarlar yapılır.
- **Nesne Editörü – Adres - Yazma Adresi** bölümünden üç noktaya tıklanır.
- Açılan pencereden **Cihaz Adı – internal_memory** (intırnıl memıri), **Aygıt Tipi – Volatile Memory Bit** (volıtıl memıri bit), **Hafıza – Bit** seçilerek seçenekler onaylanır.
- **Nesne Alanı – Veri Göstergeleri** bölümünden sürükle-bırak yöntemiyle sayfaya **LED** alınır.
- **Nesne Editörü - Görsel** bölümünden LED ile ilgili görsel ayarlar yapılır.
- **Nesne Editörü – Adres - Okuma Adresi** bölümünden üç noktaya tıklanır.
- Açılan pencereden **Cihaz Adı – internal_memory, Aygıt Tipi – Volatile Memory Bit, Hafıza – Bit** seçilerek seçenekler onaylanır.
- **Araçlar – Online Simülasyon** menülerinden simülasyon penceresi açılır (Görsel 7.23).
- Butona basılı tutulduğu müddetçe LED lamba yanacaktır.
- Simülasyon penceresi kapatılır.
- İstenirse **Nesne Editörü – Buton** bölümünden buton tipi kalıcı yapılarak sürekli çalışma sağlanabilir.



Görsel 7.23: Örnek proje uygulaması

19. UYGULAMA

HMI PROGRAMI SİMÜLASYON UYGULAMASI

AMAÇ: HMI programı ile simülasyon uygulaması yapmak.



Görsel 7.24: HMI ekranı

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Bilgisayar ve editör	-	1 adet

OKUMA YAZMA ADRES TABLOSU						
Eleman İsmi	**Adres Tipi**	**Hafıza**	**Aygıt Tipi**	**Alan**	**Adres**	**Açıklama**
ÇALIŞTIR	Okuma/ Yazma	Bit	Volatile Memory	$0	0	Kalıcı tip buton
LED	Okuma	Bit	Volatile Memory	$0	0	LED
Sirküler Bar	Okuma	UnsignedInt16	Non-Volatile Memory	$M0	0	Sirküler bar
Sayıcı	Okuma/ Yazma	UnsignedInt16	Non-Volatile Memory	$M0	0	ecounter
Analogmetre	Okuma	UnsignedInt16	Non-Volatile Memory	$M0	0	Analogmetre 2
Sayıcı	Okuma	UnsignedInt16	Non-Volatile Memory	$M0	0	counter

İŞLEM BASAMAKLARI

1. Yeni bir proje oluşturarak projeyi isimlendirip kaydediniz.
2. Programda, uygun HMI modelini seçiniz.
3. Oluşan Form_1 sayfasına kalıcı bir buton alınız.
4. Buton için okuma ve yazma adreslerini tablodaki gibi veriniz.
5. Butona ÇALIŞTIR etiketini veriniz.
6. Form_1 sayfasına LED alarak okuma yazma adreslerini tablodaki gibi veriniz.
7. Form_1 sayfasına Sirküler Bar alarak okuma yazma adreslerini tablodaki gibi veriniz.
8. Form_1 sayfasına Düz Metin alarak etiketini ANA SAYFA veriniz.
9. Online Simülasyon menü seçeneğine tıklayınız.
10. Simülasyon ekranında ÇALIŞTIR butonuna basıldığında LED'in yandığını, butona tekrar basıldığında ise LED'in söndüğünü gözlemleyiniz.
11. Simülasyon penceresini kapatınız.
12. Form – Yeni Form menü seçeneğinden yeni bir sayfa oluşturunuz.
13. Form_2 sayfasına Sayıcı (ecounter) alarak okuma yazma adreslerini tablodaki gibi veriniz.
14. Form_2 sayfasına Analogmetre alarak okuma adreslerini tablodaki gibi veriniz.
15. Form_2 sayfasına Sayıcı (counter) alarak okuma adreslerini tablodaki gibi veriniz.
16. Form_2 sayfasına düz metin alarak ilgili etiketleri veriniz (Görsel 7.24).
17. Form_2 sayfasına Push Buton alarak etiketini ANA SAYFA veriniz (Görsel 7.24).
18. ANA SAYFA butonunun Sayfa Fonksiyonunu **Önceki Sayfa** olarak ayarlayınız.
19. Form_1 sayfasına Push Buton alarak etiketini AYARLAR veriniz (Görsel 7.24).
20. AYARLAR butonunun Sayfa Fonksiyonunu **Sonraki Sayfa** olarak ayarlayınız.
21. Oluşturduğunuz programı kaydediniz.
22. Online Simülasyon menü seçeneğine tıklayınız.
23. ÇALIŞTIR butonuna basarak LED'in yandığını gözlemleyiniz.
24. Butona tekrar basarak LED'in söndüğünü gözlemleyiniz.
25. AYARLAR butonuna basarak Form_2 sayfasına geçiniz.
26. Sayıcı üzerindeki oklar yardımıyla hız değerini 50 olarak ayarlayınız.
27. Hız değeri değiştikçe Analogmetre ve Sayıcı değerinin değiştiğini gözlemleyiniz.
28. ANA SAYFA butonuna basarak Form_1 sayfasına geliniz.
29. Sirküler Bar'da da değerin 50 olduğunu gözlemleyiniz.
30. Simülasyon ekranını kapatınız.
31. Uygulamayı kaydederek çalışmayı sonlandırınız.

SORULAR

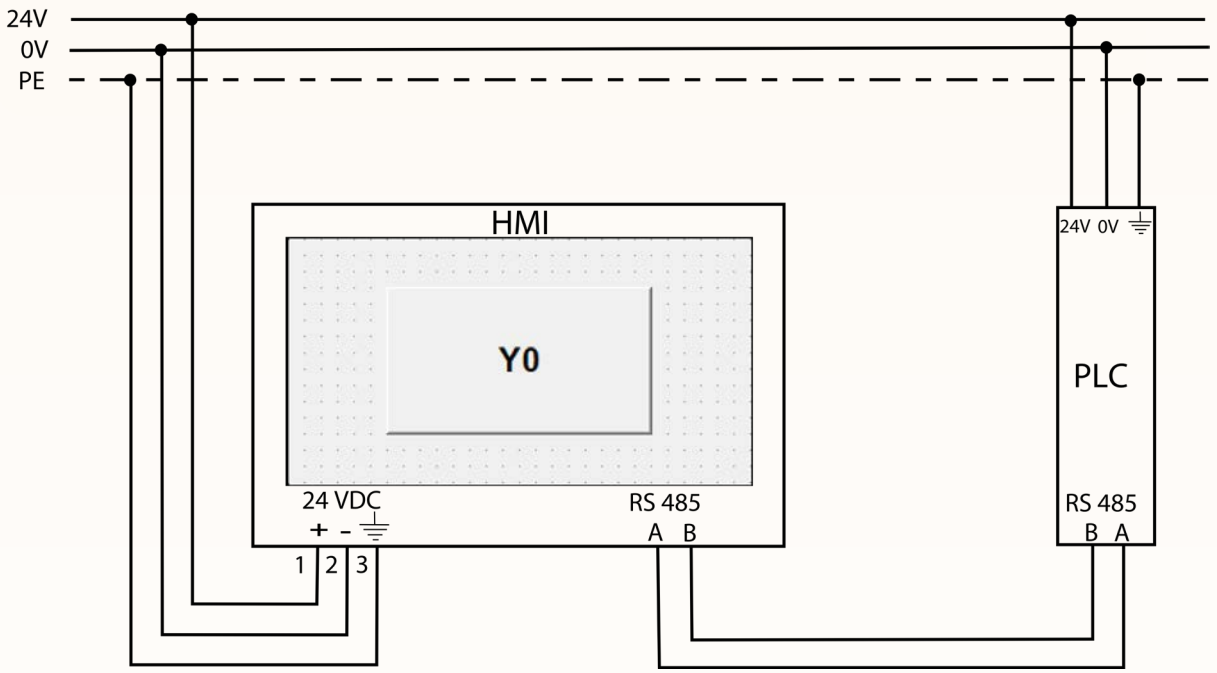
1. Devrede simülasyon yapmanın önemini açıklayınız.
2. Sayfalar arasında gezinmek için Sayfa Fonksiyonu farklı şekilde ayarlanabilir miydi? Program üzerinde test ediniz.
3. Programda gösterge olarak başka gösterge kullanılabilir miydi? Program üzerinde test ediniz.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Sayfalara doğru elemanları aldı.	20	
Sınıfı :	2	Etiketleri doğru verdi.	20	
Numarası :	3	Adresleri doğru verdi.	20	
ÖĞRETMEN	4	Sayfalar arası geçişi yaptı.	20	
Adı Soyadı :	5	Uygulamayı çalıştırdı.	20	
İmza :		**TOPLAM PUAN**	100	

20. UYGULAMA

HMI VE PLC İLE BUTON UYGULAMASI

AMAÇ: **Operatör paneli ve PLC ile buton uygulaması yapmak.**



Görsel 7.25: Devre şeması

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Bilgisayar ve editör	PLC ve HMI'ye uygun	1 adet
PLC	Transistör çıkışlı	1 adet
Operatör paneli	7 inç	1 adet
Kablo	RS 485	1 adet
Güç kaynağı	24 VDC, 2 A	1 adet

OKUMA YAZMA ADRES TABLOSU						
Eleman İsmi	**Adres Tipi**	**Hafıza**	**Aygıt Tipi**	**Alan**	**Adres**	**Açıklama**
Y0	Okuma/ Yazma	Bit	Digital Outputs	Y0	0	Kalıcı tip buton

İŞLEM BASAMAKLARI

1. PLC ve HMI arasında önce haberleşme, sonra besleme bağlantılarını yapınız.
2. HMI programında yeni bir proje oluşturunuz.
3. Uygulamada kullandığınız HMI modelini programda seçiniz.
4. Cihaz listesi penceresinden HMI'ye bağlanacak PLC'ye isim veriniz.
5. PLC portunu ve port ayarlarını seçerek kaydediniz.
6. Oluşan Form_1 sayfasına kalıcı bir buton alınız.
7. Buton için okuma ve yazma adreslerini tablodaki gibi veriniz.
8. Butona Y0 etiketini veriniz.
9. Oluşturduğunuz programı HMI'ye yükleyiniz.
10. PLC projeleri kısmında yeni bir proje oluşturunuz.
11. PLC'nin port ve bağlantı ayarlarını yaparak PLC'ye yükleyiniz.
12. HMI ekranından Y0 butonuna basarak PLC üzerindeki LED'in yandığını gözlemleyiniz.
13. Butona tekrar basarak LED'in söndüğünü gözlemleyiniz.
14. Deney setinin enerjisini kesip, elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

SORULAR

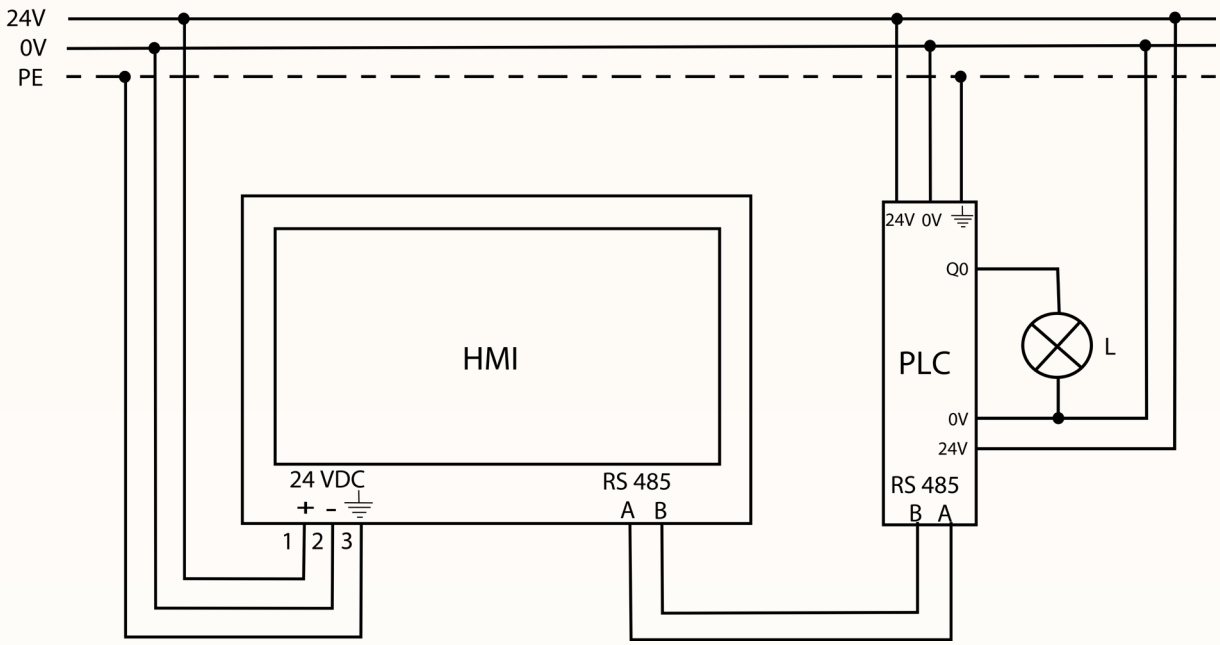
1. Uygulamada hangi haberleşme yöntemini kullandınız? Sebebini açıklayınız.
2. PLC ve HMI bağlantıları yapılırken dikkat edilecek hususlar nelerdir? Açıklayınız.
3. Devrede ani temaslı buton kullanılsaydı devre nasıl çalışırdı? Açıklayınız.

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Besleme ve haberleşme bağlantılarını yaptı.	20	
Sınıfı :	2	HMI programını hazırladı ve HMI'ye yükledi.	20	
Numarası :	3	PLC ayarlarını düzgün yaptı.	20	
ÖĞRETMEN	4	Uygulamayı çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

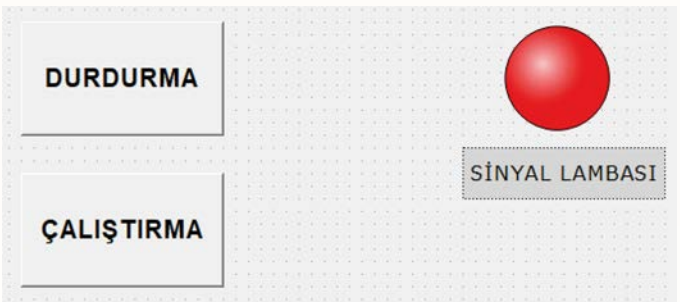
21. UYGULAMA

HMI VE PLC İLE SİNYAL LAMBASI KONTROLÜ

AMAÇ: **HMI ve PLC kullanarak sinyal lambasını kontrol etmek.**

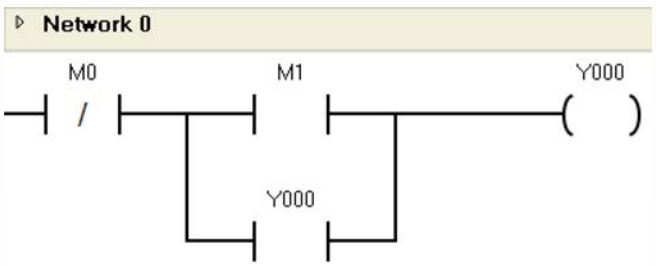


Görsel 7.26: Bağlantı şeması



Görsel 7.27: HMI ekranı

LADDER DİYAGRAMI



MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Bilgisayar ve editör	PLC ve HMI'ye uygun	1 adet
PLC	Transistör çıkışlı	1 adet
Operatör paneli	7 inç	1 adet
Kablo	RS 485	1 adet
Güç kaynağı	24 VDC, 2 A	1 adet

OKUMA YAZMA ADRES TABLOSU						
Eleman İsmi	**Adres Tipi**	**Hafıza**	**Aygıt Tipi**	**Alan**	**Adres**	**Açıklama**
DURDURMA	Okuma/ Yazma	Bit	Memory Bits	M0	0	Ani temaslı buton
ÇALIŞTIRMA	Okuma/ Yazma	Bit	Memory Bits	M1	1	Ani temaslı buton
SİNYAL LAM-BASI	Okuma	Bit	Digital Output	Y0	0	LED

PLC ETİKET TABLOSU						
Eleman İsmi	**Veri Tipi**	**Bağlantı Tipi**	**Bağlantı Adresi**	**Adres**	**IEC Adresi**	**Açıklama**
M0	Bit	Giriş	I0	X0	%IX0	NK kontak
M1	Bit	Giriş	I1	X1	%IX1	NA kontak
Y000	Bit	Çıkış	Q0	Y0	%QX0	Sinyal lambası

İŞLEM BASAMAKLARI

1. PLC ve HMI arasında önce haberleşme, sonra besleme bağlantılarını yapınız.
2. HMI programında yeni bir proje oluşturunuz.
3. Uygulamada kullandığınız HMI modelini programda seçiniz.
4. Cihaz listesi penceresinden HMI'ye bağlanacak PLC'ye isim veriniz.
5. PLC portunu ve port ayarlarını seçerek kaydediniz.
6. Oluşan Form_1 sayfasına ani temaslı iki buton ve bir LED alınız.
7. Buton ve LED için okuma yazma adreslerini tablodaki gibi veriniz.
8. Elemanların etiketlerini veriniz.
9. Oluşturduğunuz programı HMI'ye yükleyiniz.
10. PLC projeleri kısmında yeni bir proje oluşturunuz.
11. PLC'nin port ve bağlantı ayarlarını yapınız.
12. Ladder diyagramı oluşturarak PLC'ye yükleyiniz.
13. Operatör paneli ekranından ÇALIŞTIRMA butonuna basarak sinyal lambasının yandığını gözlemleyiniz.
14. DURDURMA butonuna basarak sinyal lambasının söndüğünü gözlemleyiniz.
15. Enerjiyi kesip elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

SORULAR

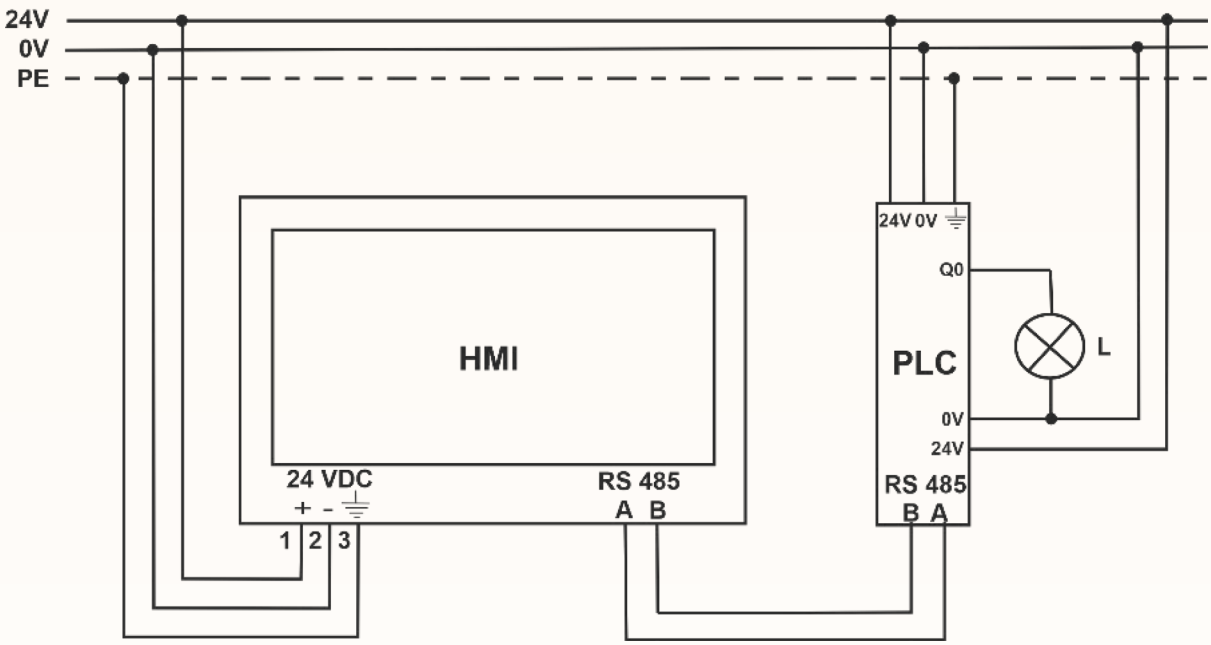
1. Okuma yazma adreslerinin kullanım amacını açıklayınız.
2. Uygulamada çıkışa kontaktör bağlanırsa devre nasıl çalışır? Açıklayınız.

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA		
Adı Soyadı :	1	Besleme ve haberleşme bağlantılarını yaptı.	20	
Sınıfı :	2	HMI programını hazırladı ve HMI'ye yükledi.	20	
Numarası :	3	PLC programını hazırladı ve PLC'ye yükledi.	20	
ÖĞRETMEN	4	Uygulamayı çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

22. UYGULAMA

HMI VE PLC İLE ZAMANLAYICI UYGULAMASI

AMAÇ: **HMI ve PLC kullanarak sinyal lambasının zamana bağlı kontrolünü yapmak.**



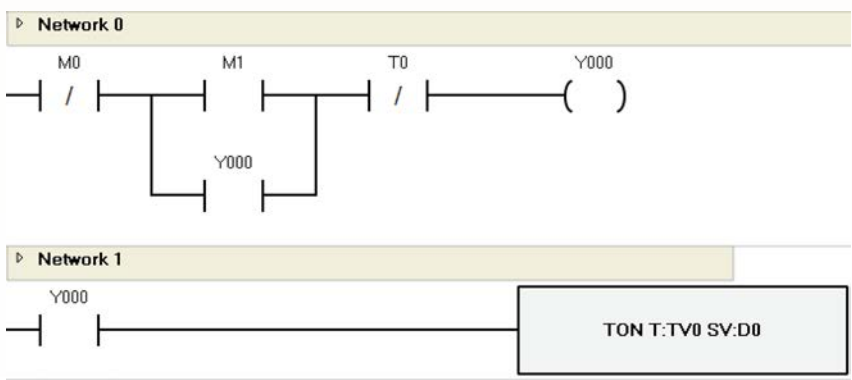
Görsel 7.28: Bağlantı şeması

LADDER DİYAGRAMI



Görsel 7.29: HMI ekranı

LADDER DİYAGRAMI



MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Bilgisayar ve editör	PLC ve HMI'ye uygun	1 adet
PLC	Transistör çıkışlı	1 adet
Operatör paneli	7 inç	1 adet
Kablo	RS 485	1 adet
Güç kaynağı	24 VDC, 2 A	1 adet

OKUMA YAZMA ADRES TABLOSU						
Eleman İsmi	**Adres Tipi**	**Hafıza**	**Aygıt Tipi**	**Alan**	**Adres**	**Açıklama**
DURDURMA	Okuma/ Yazma	Bit	Memory Bits	M0	0	Ani temaslı buton
ÇALIŞTIRMA	Okuma/ Yazma	Bit	Memory Bits	M1	1	Ani temaslı buton
SİNYAL LAMBASI	Okuma	Bit	Digital Output	Y0	0	LED
AYAR	Okuma	UnsignedInt16	Data Memory	D0	0	Sayıcı (ecounter)
	Yazma	UnsignedInt32				

PLC ETİKET TABLOSU						
Eleman İsmi	**Veri Tipi**	**Bağlantı Tipi**	**Bağlantı Adresi**	**Adres**	**IEC Adresi**	**Açıklama**
M0	Bit	Giriş	I0	X0	%IX0	NK kontak
M1	Bit	Giriş	I1	X1	%IX1	NA kontak
Y000	Bit	Çıkış	Q0	Y0	%QX0	Sinyal lambası

İŞLEM BASAMAKLARI

1. PLC ve HMI arasında önce haberleşme, sonra besleme bağlantılarını yapınız.
2. HMI programında yeni bir proje oluşturarak gerekli HMI ve PLC ayarlarını yapınız.
3. Oluşan Form_1 sayfasına gerekli elemanları alınız.
4. Butonlar, LED ve Sayıcı için okuma yazma adreslerini tablodaki gibi veriniz.
5. Elemanların etiketlerini oluşturunuz.
6. Oluşturduğunuz programı HMI'ye yükleyiniz.
7. PLC projeleri kısmında yeni bir proje oluşturarak gerekli ayarları yapınız.
8. Ladder diyagramı oluşturarak PLC'ye yükleyiniz.
9. HMI ekranından çalıştırma butonuna basarak sinyal lambasının yandığını ve ayarlanan süre sonunda söndüğünü gözlemleyiniz.
10. Çalıştırma butonuna basarak lambayı tekrar yakınız ve süre dolmadan durdurma butonuna basarak lambanın söndüğünü gözlemleyiniz.
11. Deney setinin enerjisini kesip, elemanları dikkatlice sökerek öğretmeninize teslim ediniz.

SORU

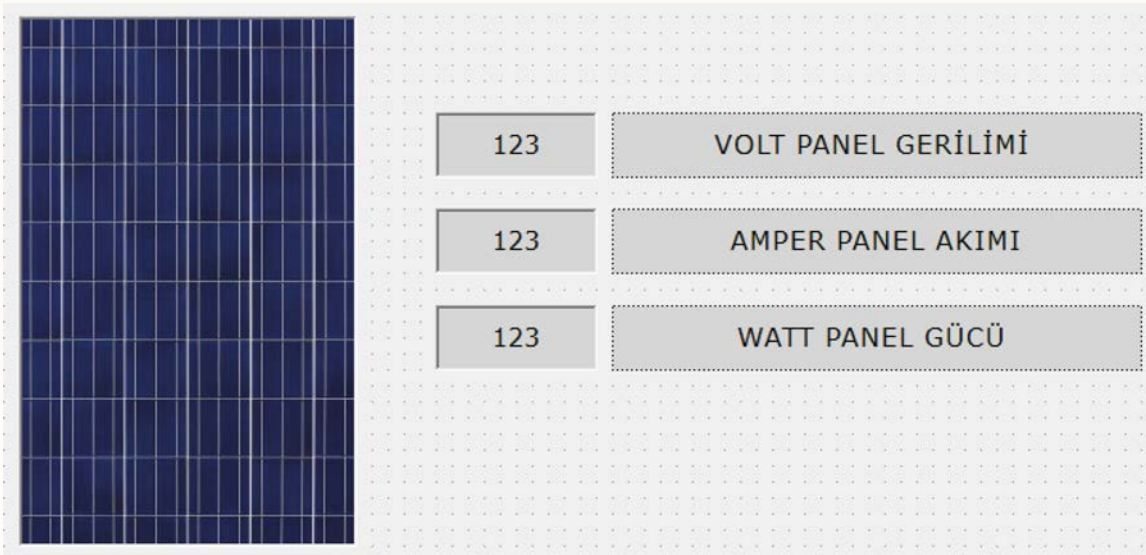
- Uygulamada üç fazlı asenkron motor kumandası için ne gibi düzenlemeler yapılmalıdır? Açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Besleme ve haberleşme bağlantılarını yaptı.	20	
Sınıfı :	2	HMI programını hazırladı ve HMI'ye yükledi.	20	
Numarası :	3	PLC programını hazırladı ve PLC'ye yükledi.	20	
ÖĞRETMEN	4	Uygulamayı çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

23. UYGULAMA

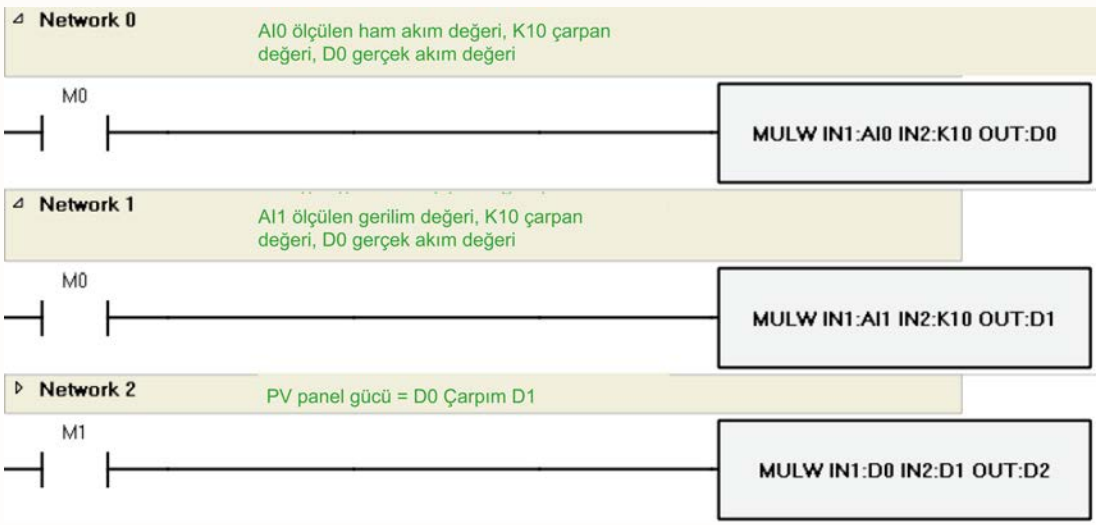
HMI EDİTÖRÜ İLE PV PANEL SİMÜLASYONU

AMAÇ: **HMI editörü ile PV panel simülasyonu yapmak.**



Görsel 7.30: Operatör paneli ekranı

LADDER DİYAGRAMI



MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Bilgisayar ve editör	PLC ve HMI'ye uygun	1 adet

OKUMA YAZMA ADRES TABLOSU						
Eleman İsmi	**Adres Tipi**	**Hafıza**	**Aygıt Tipi**	**Alan**	**Adres**	**Açıklama**
GERİLİM	Okuma	UnsignedInt16	Data Memory	D0	0	Sayı göster
AKIM	Okuma	UnsignedInt16	Data Memory	D1	1	Sayı göster
GÜÇ	Okuma	UnsignedInt16	Data Memory	D2	2	Sayı göster

İŞLEM BASAMAKLARI

1. HMI programında yeni bir proje oluşturarak gerekli ayarları yapınız.
2. PV panel için kullanacağınız resim dosyasını Resim ve Font kütüphanesine ekleyiniz.
3. Form_1 sayfasına gerekli elemanları alınız.
4. Elemanlar için okuma adreslerini tablodaki gibi veriniz.
5. Elemanların etiketlerini oluşturunuz.
6. PLC projeleri kısmında yeni bir proje oluşturarak gerekli ayarları yapınız.
7. Ladder diyagramı oluşturunuz.
8. Gerilim ve akım için değerler vererek GÜÇ alanından sonucu gözlemleyiniz.
9. Program ve bilgisayarı kurallarına uygun olarak kapatınız.

SORU

- Uygulama panel ekranı, görsel değişiklikler yapılarak nasıl farklılaştırılabilir? Editör programını inceleyerek programın imkânları dâhilinde ekranı yeniden düzenleyiniz.

ÖĞRENCİ		**DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA**		
Adı Soyadı :	1	PV panel resmini kütüphaneye ekledi.	20	
Sınıfı :	2	HMI programını hazırladı.	20	
Numarası :	3	PLC programını hazırladı.	20	
ÖĞRETMEN	4	Uygulamayı çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

7.3. FREKANS İNVERTÖRLERİ İLE ASENKRON MOTOR KONTROLÜ

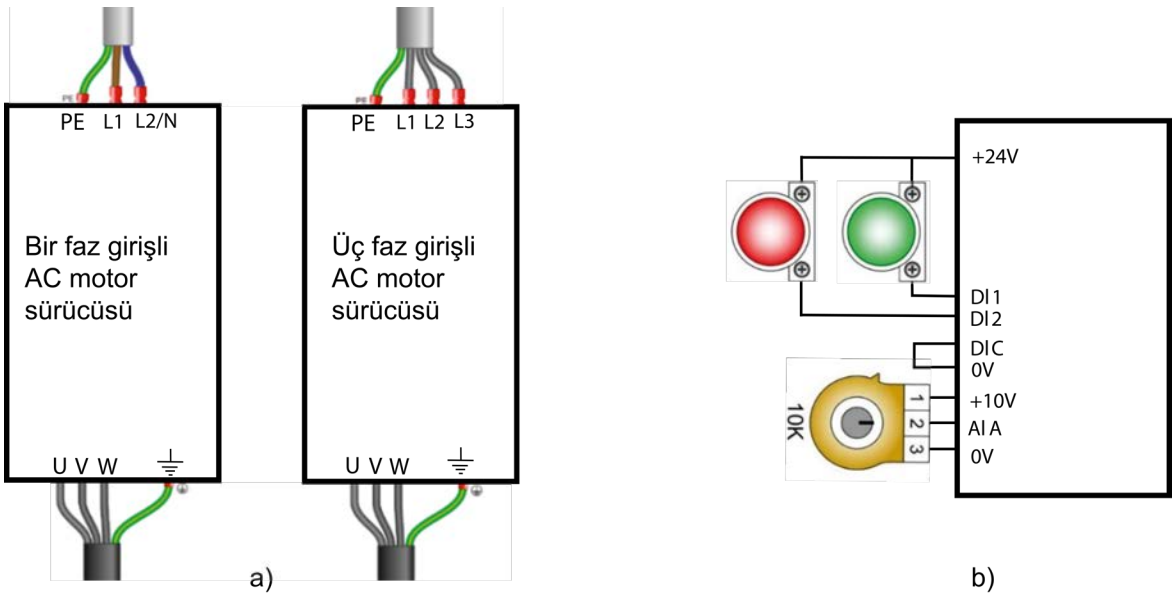
Asenkron motorların frekans ve gerilimini değiştirerek devir sayısının kontrol edilmesini sağlayan elektronik cihazlara **frekans invertörü** denir. Frekans invertörlerine; **sürücü, konvertör, driver** gibi farklı isimler de verilmektedir.

Sürücüler, kontrol paneli ve bağlantı terminallerinden oluşur. Kontrol paneli gerekli parametre ayarlarının yapılmasını, terminallerse şebeke ve diğer cihazlara olan bağlantıyı sağlar. Sürücülerin yapısında; şebeke gerilimini doğrultmak için tek veya üç fazlı doğrultma diyotları, filtre için bobin ve kondansatörler, değişken frekans ve gerilim elde etmek için transistörler, kontrol algoritmalarını çalıştırmak için mikroişlemciler bulunur.

7.3.1. Sürücülerin Güç ve Kontrol Bağlantıları

Sürücüler bir ve üç faz girişli olarak üretilir. Terminaller şebekeye giriş özelliğine göre bağlanır. Çıkış bağlantıları her iki sürücüde aynı olup motor bağlantı kabloları invertör üzerinde belirtilen çıkış uçlarına bağlanır. Sürücü gövdesi topraklama iletkeni (PE) ile topraklanmalıdır (Görsel 7.31:a).

Bir faz beslemeli sürücülerde, kullanılacak motor yıldız bağlı ise üçgen bağlantıya dönüştürülmelidir.



Görsel 7.31: Sürücülerin güç ve kontrol bağlantıları

Sürücü terminallerinde giriş ve çıkışlar analog veya dijital özelliğe sahiptir. Sürücünün dijital girişlerine çeşitli tiplerde buton ve dijital sensörler, analog girişlere haricî potansiyometre ya da analog sensör, dijital çıkışlara da ihtiyaca göre uygun devre elemanları bağlanır (Görsel 7.31:b).

7.3.2. Sürücü Parametreleri ve Parametre Giriş İşlemleri

Sürücü kontrolünün temeli parametre ayarıdır. Parametre ayarları çoğunlukla sürücü üzerindeki sabit veya panel kapağındaki taşınabilir panel aracılığıyla yapılır. Aynı zamanda sürücüye bilgisayar, tablet gibi cihazlarla kablolu veya kablosuz bağlantı kurularak da yapılabilmektedir.

Operatör panelinin üzerindeki ayar tuşları; başlatma, durdurma, çeşitli modlarda seçim yapma gibi işlemlerin yapılmasını sağlar. Ayar işlemleri sürücünün marka ve modeline göre değişiklik gösterebilir. Örneğin bir sürücüde menüler arasında geçişler **M** tuşuyla, onaylama **OK** tuşuyla gerçekleştirilir. Tuşların üzerine 2 sn.den az basmak **kısa basma**, 2 sn.den uzun basmak ise **uzun basma** olarak adlandırılır. Menüye ait parametreler arasında yukarı (↑) ve aşağı (↓) tuşları ile seçim yapılabilir.

Her üretici firmanın kendi ürününe göre hazırlamış olduğu parametreler kullanıcı kılavuzlarında yer alır. Uygulamalarda yer alan motor sürücüsünde de sıklıkla kullanılan parametreler Tablo 7.12' de verilmiştir.

Tablo 7.12: Sürücülerde Sıklıkla Kullanılan Parametreler

Parametre	Parametre Açıklaması	Değerler
P0003	Parametre erişim seviyesi	0: Kullanıcı tanımlı 1: Standart 2: Genişletilmiş 3: Uzman 4: Bakım
P0010	Devreye alma	0: Hazır 1: Hızlı devreye alma 2: Konverter 29: Download 30: Fabrika ayarı
P0100	Güç ve frekans ayarları	0: Avrupa (kW), motor baz frekansı 50 Hz
P0304	Motor anma gerilimi (V)	λ / Δ bağlantıya göre motor etiket gerilim değeri
P0305	Motor anma akımı (A)	λ / Δ bağlantıya göre motor etiket akım değeri
P0307	Motor anma gücü (kW/hp)	P0100=0 ya da 2 girilmişse (kW), 1 girilmişse (hp)
P0308	Motor anma güç katsayısı	P0100=0 ya da 2 girilmişse (kW)
P0309	Motor anma verimi (%)	P0100=1 girilmişse motor etiketindeki değer
P0310	Motor anma frekansı (Hz)	50 / 60 Hz
P0311	Motor anma devir sayısı	Motor etiketindeki devir sayısı değeri
P0700	Başlangıç komut kaynağı seçimi	0: Fabrika varsayılan ayarı 1: Operatör paneli (Fabrika varsayılanı) 2: Klemens terminali 5: RS 485 üzerinden USS / MODBUS
P0701	Dijital giriş 1 (DI1) işlevi	0: (start/dir) 1: ON / OFF1 2: ON ters yön/OFF1 5: ON / OFF2 9: Hata onayı 10: JOG sağa 11: JOG sol 12: Ters yön 13: Frekans arttırma 14: Frekans azaltma
P0702	Dijital giriş 2 (DI2) işlevi	
P0703	Dijital giriş 3 (DI3) işlevi	
P0704	Dijital giriş 4 (DI4) işlevi	
P1001	Sabit frekans 1	Fabrika ayarı: 10 Hz
P1002	Sabit frekans 2	Fabrika ayarı: 15 Hz
P1003	Sabit frekans 3	Fabrika ayarı: 25Hz
P1004	Sabit frekans 4	Fabrika ayarı: 50 Hz
P1032	Ters yön engelleme	0: Engellenmez 1: Engellenir

7.3.3. Sürücü Kontrol Yöntemleri

Asenkron motorların devir sayısı, **ns=(60xf)/p** formülüne göre frekans ve kutup sayısına bağlıdır. Sürücü, motorun frekansını değiştirerek devir sayısını değiştirir. Motorun frekansını değiştirirken gerilimini de değiştirir. Frekans daha çok istenilen devire göre seçilir. Motor geriliminde ise devir sayısının yüke göre sabit kalabilmesi için aşağıda verilen kontrol algoritmaları kullanılır.

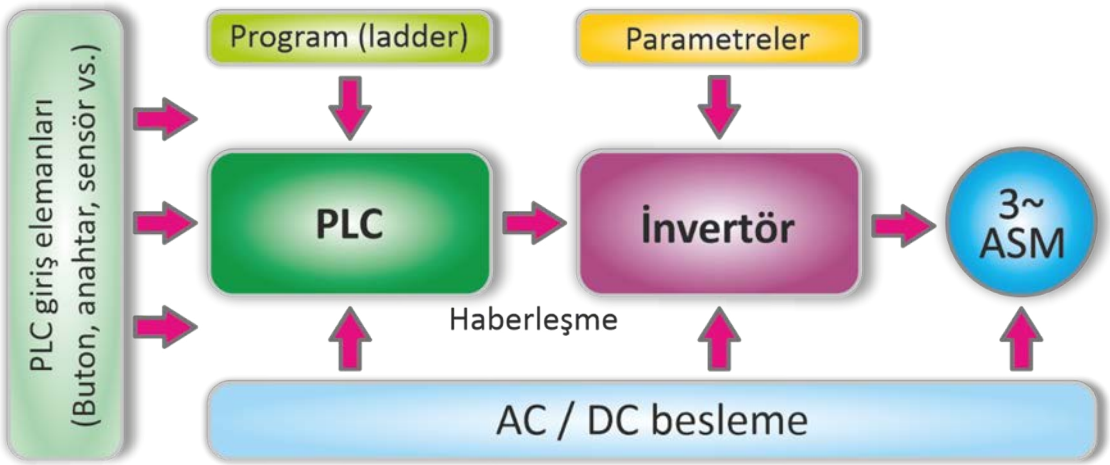
V/f Kontrol: Bu kontrol yönteminde temel değişkenler gerilim ve frekanstır. Frekansı değiştirerek motor hızı ayarlanırken birim frekans başına düşen gerilim oranının da (V/f oranı) sabit tutulması gerekir. Uygun V/f oranı, motorun nominal geriliminin nominal frekansına bölünmesine eşittir. Bu kontrol modu, hassas kontrol gerektirmeyen uygulamalarda (pompa, fan vb.) kullanılır.

Vektör Kontrol: Bu kontrol yönteminin amacı düşük hızda torku arttırmaktır. Vektör kontrollü sürücüler maksimum tork için gerilim ve frekansı ayrı olarak değiştirebilir. Böylece daha fazla başlangıç torku üretilir. Temel olarak kapalı çevrim ve açık çevrim (sensörsüz) olmak üzere iki tip vektör kontrol yöntemi vardır. Bu kontrol modu, yüksek hız kontrol hassasiyeti gerektiren uygulamalarda (vinç vb.) kullanılır.

Bu kontrol algoritmaları ile beraber, uygulamalara göre farklı algoritmalar da kullanılabilir. Örneğin güneş panellerinde değişken DC gerilimle beslenen invertörlerde, gerilime göre frekans değiştirilerek maksimum güç takip etme algoritması kullanılır.

7.3.4. PLC İle Sürücü Kontrolü

Sürücüler, küçük otomasyon sistemlerinde bağımsız; yapısında PLC ve HMI bulunduran otomasyon sistemlerindeyse bu elemanlarla birlikte çalışır. Birlikte çalışma uygulamasında PLC çıkışları invertör girişine; buton, anahtar, sensör gibi giriş elemanları da PLC girişlerine bağlanır. (Görsel 7.32).



Görsel 7.32: PLC ve sürücü bağlantısı prensip şeması

Çalışma, durma, devir yönü, devir sayısı gibi verilerin sürücüye girişi aşağıda verilen yöntemlerden herhangi biriyle yapılabilir.

- Sürücü üzerindeki tuş takımından
- Sürücü dijital ve analog terminallerinden
- Sürücü haberleşme portundan (Modbus, Canbus, Profinet, Ethercat vb.)
- Karma yöntem (Bu yöntemlerden herhangi ikisinin birlikte kullanılması)

Seçilecek yönteme göre PLC ve sürücü için ilave modüller (analog, haberleşme vb.) kullanmak gerekebilir. PLC'nin programlanması kullanılacak yönteme göre yapılır. Sürücünün çalışması sırasında hangi verilerin kullanılacağı parametre ayarları ile belirlenir.

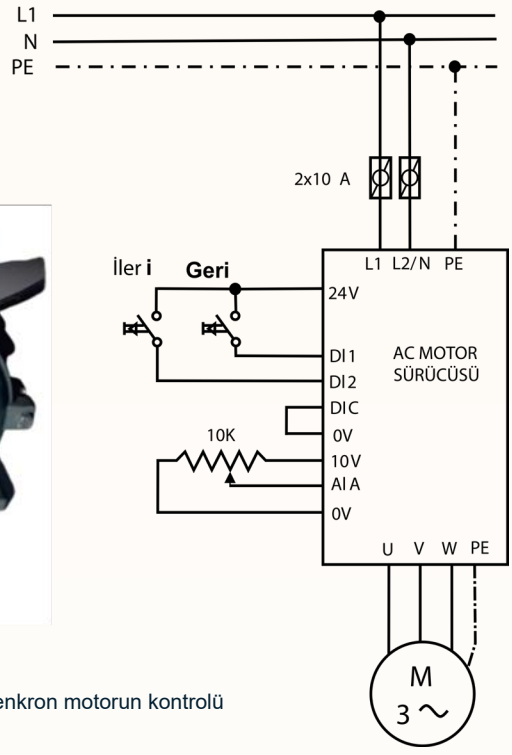
Asenkron motor devrinin hesaplanarak bulunması algoritması, PLC veya sürücüde çalıştırılarak bulunabilir. Motor devrine PLC karar veriyorsa motorun çalışma, durma, devir yönü ve devir sayısı verileri PLC tarafından sürücüye bildirilir. PLC; sürücüye dijital, analog ve haberleşme terminallerinden verileri iletir.

24. UYGULAMA

FREKANS İNVERTÖRÜNÜN HARİCÎ ELEMANLARLA ÇALIŞTIRILMASI

AMAÇ: **Frekans invertörüne haricî elemanlar bağlayarak üç fazlı asenkron motoru kontrol etmek.**





Görsel 7.33: Frekans invertörüyle üç fazlı asenkron motorun kontrolü

PARAMETRE LİSTESİ

Parametre	Parametre Açıklaması
P0003 = 3	Tüm parametrelere erişim imkânı verir.
P0010 = 30 P0970 = 1	P0010 parametresine 30 ve P0970 parametresine 1 girilirse tüm parametreler fabrika varsayılan değerleri olarak ayarlanır.
P0304 = 220	Motor anma gerilimi (V). Motor etiketine göre girilir (220 V).
P0305 = 1,9	Motor anma akımı (A). Motor etiketine göre girilir (1,9 A).
P0307 = 0,37	Motor anma gücü (kW/Hp). Motor etiketine göre girilir (0,37 kW).
P0308 = 0,75	Motor anma güç katsayısı (cosφ). Motor etiketine göre girilir (0,75).
P0310 = 50	Motor anma frekansı (Hz). Motor etiketine göre girilir (50 Hz).
P0311 = 2800	Motor anma devir sayısı (RPM). Motor etiketine göre girilir (2800 RPM).
P1120 = 2	Motor kalkış hızlanma süresi (sn.). Ayarlanan devire 2 sn. içinde ulaşır.
P1121 = 3	Motor duruş yavaşlama süresi (sn.). 3 sn. sonunda motor durur.
P0700 = 2	Başlangıç komut kaynağı seçimi. 2: Haricî klemens seçilir.
P0701 = 1	Dijital giriş 1 (DI1) işlev seçimi. ON/OFF1 geri butonu bağlanmıştır.
P0702 = 2	Dijital giriş 2 (DI2) işlev seçimi. ON ters yön/OFF1 İleri butonu bağlanmıştır.
P1000 = 2	Frekans (hız) bilgisi ayar seçimi. 2: Potansiyometre için analog ayar noktası 1
P1032 = 0	Ters yön engelleme. İleri-geri çalışma için ters yön engeli kaldırılmıştır.

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Asenkron motor	Üç fazlı (Δ 220 V), 0,37 kW, 2800 RPM	1 adet
Motor sürücüsü	Bir fazlı (motor gücüne uygun)	1 adet
Kumanda butonu	Ani temaslı	2 adet
Potansiyometre	5 K veya 10 K	1 adet

İŞLEM BASAMAKLARI

1. Kullanılacak malzemelerin bağlantı ve teknik bilgilerini üretici firmaların bilgi sayfasından alınız.
2. Görsel 7.33'te şeması verilen elemanların güç ve kontrol devre bağlantılarını yapınız.
3. Öğretmeninizin kontrolünde devreye enerji veriniz.
4. Parametre listesinden faydalanarak sürücüyü fabrika değerlerine ayarlayınız.
5. Motor anma değerlerini ve sürücü kontrol parametre bilgilerini giriniz.
6. Potansiyometreyi orta seviyeye ayarlayınız.
7. İleri butonuna basarak motorun dönüş yönü ve hızını gözlemleyiniz.
8. Motor durduğunda geri butonuna basarak motorun dönüş yönü ve hızını gözlemleyiniz.
9. Potansiyometre ayarını değiştirerek motorun hızını gözlemleyiniz.
10. Enerjiyi kesip devreyi dikkatlice sökünüz ve elemanları öğretmeninize teslim ediniz.

SORULAR

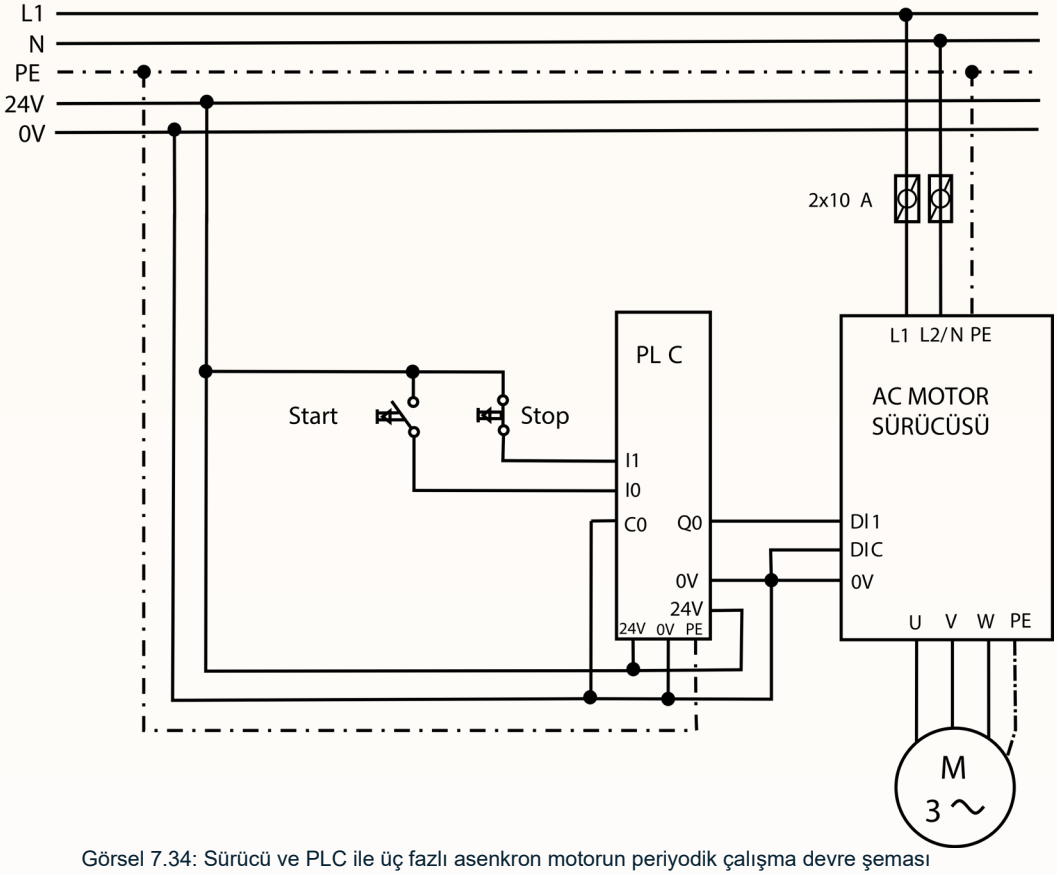
1. Sürücülerde kullanılan kontrol yöntemlerini açıklayınız.
2. Bu uygulamada motor hız bilgisinin sürücüye nasıl girildiğini açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Sürücü ve motor bağlantılarını yaptı.	20	
Sınıfı :	2	Sürücü parametre ayarlarını yaptı.	20	
Numarası :	3	Motorun hız kontrolünü doğru yaptı.	20	
ÖĞRETMEN	4	Motorun yön kontrolünü doğru yaptı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

25. UYGULAMA

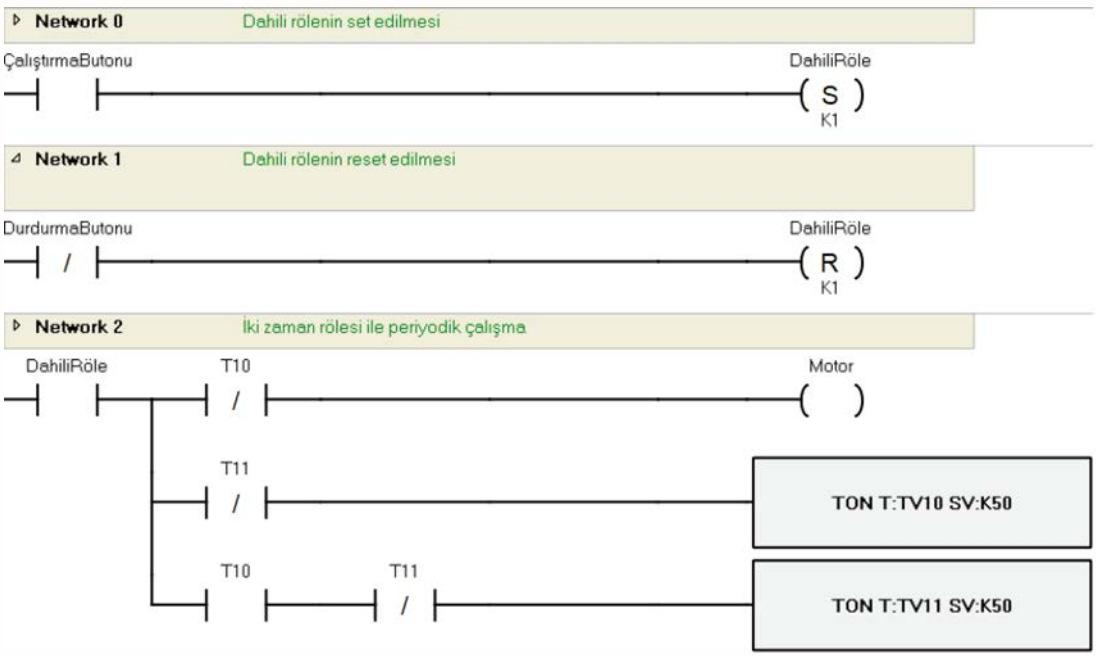
SÜRÜCÜ VE PLC İLE ÜÇ FAZLI ASENKRON MOTORUN PERİYODİK OLARAK ÇALIŞTIRILMASI

AMAÇ: **Sürücü ve PLC ile üç fazlı asenkron motoru periyodik olarak çalıştırmak.**



Görsel 7.34: Sürücü ve PLC ile üç fazlı asenkron motorun periyodik çalışma devre şeması

LADDER DİYAGRAMI



PARAMETRE LİSTESİ

Parametre	Parametre Açıklaması
P0003 = 3	Tüm parametrelere erişim imkânı verir.
P0010 = 30 P0970 = 1	P0010 parametresine 30 ve P0970 parametresine 1 girilirse tüm parametreler fabrika varsayılan değerleri olarak ayarlanır.
P0304 = 220	Motor anma gerilimi (V). Motor etiketine göre girilir (220 V).
P0305 = 1,9	Motor anma akımı (A). Motor etiketine göre girilir (1,9 A).
P0307 = 0,37	Motor anma gücü (kW/Hp). Motor etiketine göre girilir (0,37 kW).
P0308 = 0,75	Motor anma güç katsayısı (cosφ). Motor etiketine göre girilir (0,75).
P0310 = 50	Motor anma frekansı (Hz). Motor etiketine göre girilir (50 Hz).
P0311 = 2800	Motor anma devir sayısı (RPM). Motor etiketine göre girilir (2800 RPM).
P1120 = 2	Motor kalkış hızlanma süresi (sn.). Ayarlanan devire 2 sn. içinde ulaşır.
P1121 = 5	Motor duruş yavaşlama süresi (sn.). 5 sn. sonunda motor durur.
P0700 = 2	Başlangıç komut kaynağı seçimi. 2: Haricî klemens terminali seçilir.
P0701 = 1	Dijital giriş 1 (DI1) işlev seçimi. 1: ON/OFF1 ileri start butonu bağlanmıştır.
P1000 = 1	Frekans (hız) bilgisi ayar seçimi. 1: MOP (keypad) ayar noktası.

MALZEME LİSTESİ

Adı	Özelliği	Miktarı
Motor sürücüsü	Bir fazlı (motor gücüne uygun seçilmeli)	1 adet
PLC	Transistör çıkışlı	1 adet
Asenkron motor	Üç fazlı (Δ 220 V), 0.37 kW, 2800 RPM	1 adet
Kumanda butonları	Ani temaslı NO ve NC	2 adet

PLC ETİKET TABLOSU

Eleman İsmi	Veri Tipi	Bağlantı Tipi	Bağlantı Adresi	Adres	IEC Adresi	Açıklama
ÇalıştırmaButonu	Bit	Giriş	I0	X0	%IX0	NA kontak
DurdurmaButonu	Bit	Giriş	I1	X1	%IX1	NK kontak
DahiliRöle	Bit	G/Ç	M0	M0	%MX0.0	Dâhilî röle
Motor	Bit	Çıkış	Q0	Y0	%QX0	Motor kontaktörü

İŞLEM BASAMAKLARI

1. Görsel 7.34'te şeması verilen cihazların güç ve kontrol devre bağlantılarını yapınız.
2. Öğretmeninizin kontrolünde devreye enerji veriniz.
3. Devrenin ladder diyagramını hazırlayarak PLC'ye yükleyiniz.
4. Sürücüyü, parametre listesinden faydalanarak fabrika değerlerine ayarlayınız.
5. Motor anma değerlerini ve kontrol bilgilerini ilgili parametrelere giriniz.
6. Start butonuna basınız ve motorun PLC programında ayarlanan süre kadar bir yönde periyodik olarak çalışıp durduğunu gözlemleyiniz.
7. Çalışma sırasında stop butonuna basınız ve motorun durumunu gözlemleyiniz.
8. Parametre ayarlarından motor kalkış ve yavaşlama değerlerini değiştirip, start butonuna tekrar basarak motor hareketini gözlemleyiniz.
9. Stop butonuna basınız. PLC programından zaman ayarlarını değiştirip, start butonuna tekrar basarak motor hareketini gözlemleyiniz.
10. Enerjiyi kesip devreyi dikkatlice sökünüz ve elemanları öğretmeninize teslim ediniz.

SORU

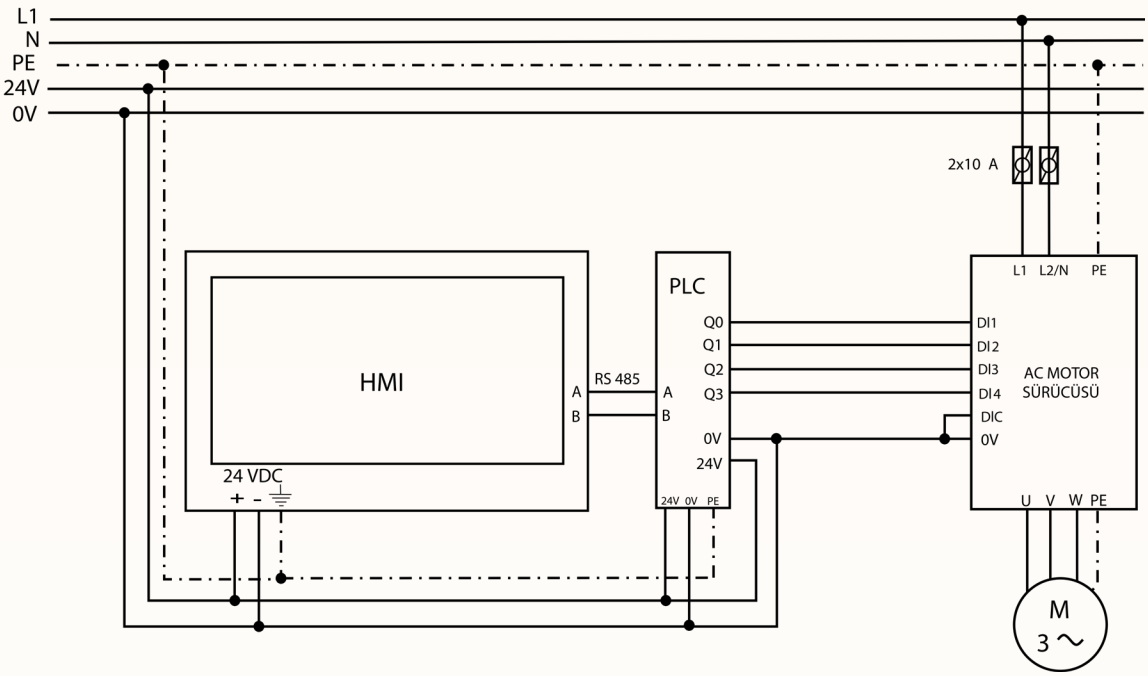
- Bu uygulamada motorun periyodik çalışmasının sebebi nedir? Açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Sürücü motor ve PLC bağlantılarını yaptı.	20	
Sınıfı :	2	Sürücü parametre ayarlarını yaptı.	20	
Numarası :	3	PLC programını PLC'ye yükledi.	20	
ÖĞRETMEN	4	Motoru doğru çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

26. UYGULAMA

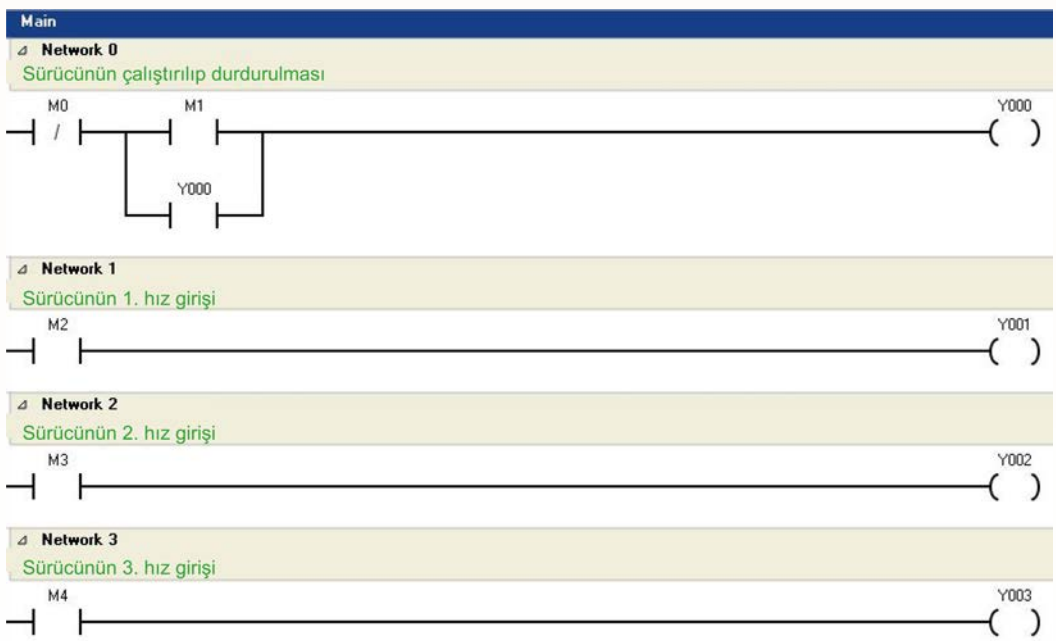
ÜÇ FAZLI ASENKRON MOTORUN PLC VE HMI KULLANILARAK HIZ KONTROLÜ

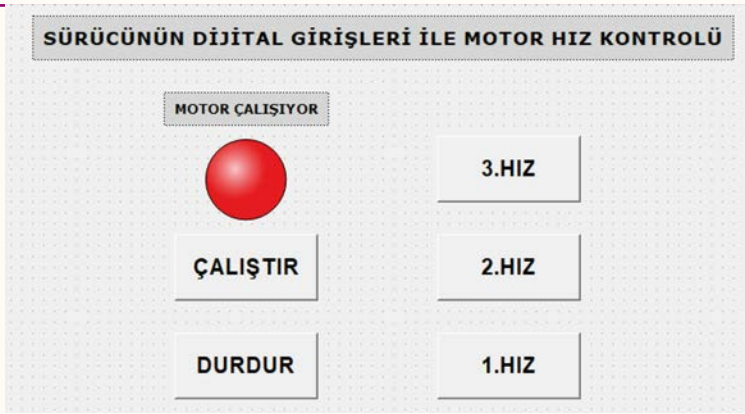
AMAÇ: Üç fazlı asenkron motoru PLC ve HMI ile sürücünün dijital girişleri kullanılarak farklı hızlarda çalıştırmak.



Görsel 7.35: Üç fazlı asenkron motorun PLC ve HMI ile hız kontrolü devre şeması

LADDER DİYAGRAMI





Görsel 7.36: HMI ekranı

PARAMETRE LİSTESİ

Parametre	Parametre Açıklaması			
P0003 = 3	Tüm parametrelere erişim imkânı verir.			
P0010 = 30 P0970 = 1	P0010 parametresine 30 ve P0970 parametresine 1 girilirse tüm parametreler fabrika varsayılan değerleri olarak ayarlanır.			
P0304 = 220	Motor anma gerilimi (V). Motor etiketine göre girilir (220 V).			
P0305 = 1,9	Motor anma akımı (A). Motor etiketine göre girilir (1,9 A).			
P0307 = 0,37	Motor anma gücü (kW/Hp). Motor etiketine göre girilir (0,37 kW).			
P0308 = 0,75	Motor anma güç katsayısı (cosφ). Motor etiketine göre girilir (0,75).			
P0310 = 50	Motor anma frekansı (Hz). Motor etiketine göre girilir (50 Hz).			
P0311 = 2800	Motor anma devir sayısı (RPM). Motor etiketine göre girilir (2800 RPM).			
P1120 = 2	Motor kalkış hızlanma süresi (sn.). Ayarlanan devire 2 sn. içinde ulaşır.			
P1121 = 3	Motor duruş yavaşlama süresi (sn.). 3 sn. sonunda motor durur.			
P0700 = 2	Başlangıç komut kaynağı seçimi. 2: Haricî klemens terminali seçilir.			
P1000 = 3	Hız bilgisi kaynağı dijital giriş sabit frekanslar.			
P0701 = 1	1. Dijital giriş fonksiyonu ON/OFF.			
P0702 = 15	2. Dijital giriş fonksiyonu sabit hız biti 0.			
P0703 = 16	3. Dijital giriş fonksiyonu sabit hız biti 1.			
P0704 = 17	4. Dijital giriş fonksiyonu sabit hız biti 2.			
P1016 = 1	Sabit frekans modu.			
P1020 = 722.1	Sabit frekans seçim biti 0 DI2.		P1021 = 722.2	Sabit frekans seçim biti 1 DI3.
P1022 = 722.3	Sabit frekans seçim biti 2 DI4.			
P1001 = 10	Sabit frekans 1.	P1002 = 15	Sabit frekans 2.	
P1003 = 25	Sabit frekans 3.	P0771 = 21	Analog çıkış anlık frekans.	
P0731 = 52.2	1.Dijital çıkış fonksiyonu sürücü çalışıyor.		P0732 = 52.3	2.Dijital çıkış fonksiyonu sürücü çalışıyor.

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Asenkron motor	Üç fazlı (Δ 220 V), 0,37 kW, 2800 RPM	1 adet
Motor sürücüsü	Bir fazlı (motor gücüne uygun)	1 adet
HMI	7 inç	1 adet
PLC	Transistör çıkışlı	1 adet
Kablo	RS 485 haberleşme, 0,75 mm^2 bağlantı	---
Güç kaynağı	24 VDC, 2 A	1 adet

OKUMA YAZMA ADRES TABLOSU						
Eleman İsmi	**Adres Tipi**	**Hafıza**	**Aygıt Tipi**	**Alan**	**Adres**	**Açıklama**
DURDUR	Okuma/Yazma	Bit	Memory Bits	M0	0	Push Button
ÇALIŞTIR	Okuma/Yazma	Bit	Memory Bits	M1	1	Push Button
1.HIZ	Okuma/Yazma	Bit	Memory Bits	M2	2	Kalıcı tip buton
2.HIZ	Okuma/Yazma	Bit	Memory Bits	M3	3	Kalıcı tip buton
3.HIZ	Okuma/Yazma	Bit	Memory Bits	M4	4	Kalıcı tip buton
MOTOR ÇALIŞIYOR	Okuma	Bit	Memory Bits	Y0	0	LED

İŞLEM BASAMAKLARI

1. Görsel 7.35'te şeması verilen cihazların güç, kontrol ve haberleşme bağlantılarını yapınız.
2. Öğretmeninizin kontrolünde devreye enerji veriniz.
3. HMI programında yeni bir proje oluşturarak gerekli HMI ve PLC ayarlarını yapınız.
4. Oluşan Form_1 sayfasına gerekli elemanları alınız.
5. Elemanlar için okuma yazma adreslerini tablodaki gibi veriniz.
6. Elemanların etiketlerini oluşturunuz.
7. Oluşturduğunuz programı HMI'ye yükleyiniz.
8. PLC projeleri kısmında yeni bir proje oluşturarak gerekli ayarları yapınız.
9. Ladder diyagramı oluşturarak PLC'ye yükleyiniz.
10. Parametre listesinden faydalanarak sürücüyü fabrika değerlerine ayarlayınız.
11. Motor anma değerlerini ve kontrol bilgilerini ilgili parametrelere giriniz.
12. HMI ekranından 1.HIZ bilgisini seçerek, ÇALIŞTIR butonuna basıp motorun çalışmasını gözlemleyiniz.
13. DURDUR butonuna basarak motoru durdurunuz.
14. HMI ekranından sırasıyla 2.HIZ ve 3.HIZ bilgisini seçerek, ÇALIŞTIR butonuna basıp motorun çalışmasını gözlemleyiniz.
15. DURDUR butonuna basarak motoru durdurunuz.
16. Enerjiyi kesip devreyi dikkatlice sökünüz ve elemanları öğretmeninize teslim ediniz.

ÖĞRENCİ	**DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA**			
Adı Soyadı :	1	Cihaz bağlantılarını yaptı.	20	
Sınıfı :	2	Sürücü parametre ayarlarını yaptı.	20	
Numarası :	3	HMI ve PLC programlarını yapıp HMI ve PLC'ye yükledi.	20	
ÖĞRETMEN	4	Motoru farklı hızlarda çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

7.4. PLC İLE SERVO MOTORLARIN KONTROLÜ

Servo, hatayı algılayarak düzeltmeye çalışan kapalı çevrim kontrol sistemine verilen genel addır. Kapalı çevrim kontrol; motorun hızını, konumunu ve yönünü ayarlamak için geri besleme sinyalini kullanan bir sistemdir. Bu sistemde bir motor ve geri besleme ünitesi (enkoder) bulunur. Servo sisteme göre çalışan motorlara **servo motor** denir (Görsel 7.37).



Görsel 7.37: Çeşitli servo motorlar

Asenkron, DC ve senkron motorlar, pozisyon kontrolü amacıyla kullanıldığında servo motor olarak isimlendirilir. Rotoru sabit mıknatıslı senkron motor, daha iyi pozisyonlama performansı gösterdiği için servo motor olarak tercih edilir. Bu motorda, rotor yapısındaki sabit mıknatısların manyetik alanı ile statorda oluşturulan döner manyetik alanın etkisinde dönme hareketi oluşturulur. Rotor devri, stator döner alan devrine eşittir.

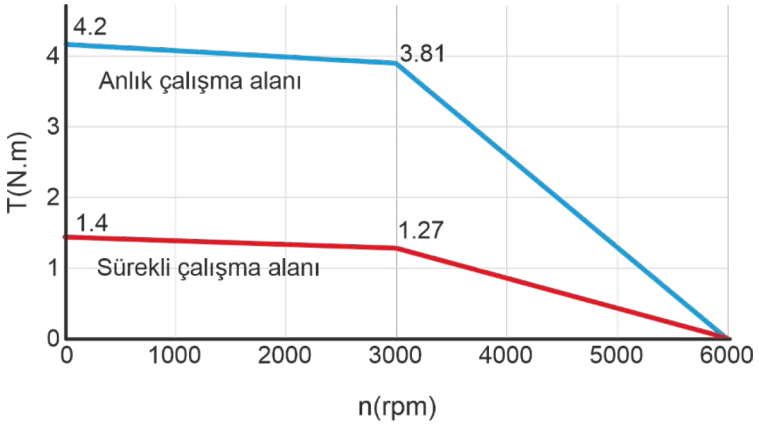
Servo motor, frenli veya frensiz olabilir. Frenli motorlarda fren bobini bulunur ve freni açmak için bobin enerjilenir. Motor hareketi durduğunda yük, yer çekimi gibi etkiler ile motorun hareket etmesi istenmiyorsa frenli servo motor tercih edilir.

Uygulamalarda yükün türüne ve büyüklüğüne göre redüktör (mekanik yük dönüştürücü) kullanılır. **Redüktör**, dişlilerden oluşan mekanik sistemdir ve devir sayısını yükün istediği duruma getirme amacıyla kullanılır.

Servo sistemin hızını, motor mili konumunu ve yükün bulunduğu konumu ölçmek için kullanılan elemanlarına **geri besleme elemanı** denir. Servo motorlarda geri besleme elemanı olarak enkoder kullanılır. Enkoderler optik veya manyetik yapıdadır ve bir turu 16-22 bit çözünürlüğünde bölebilir.

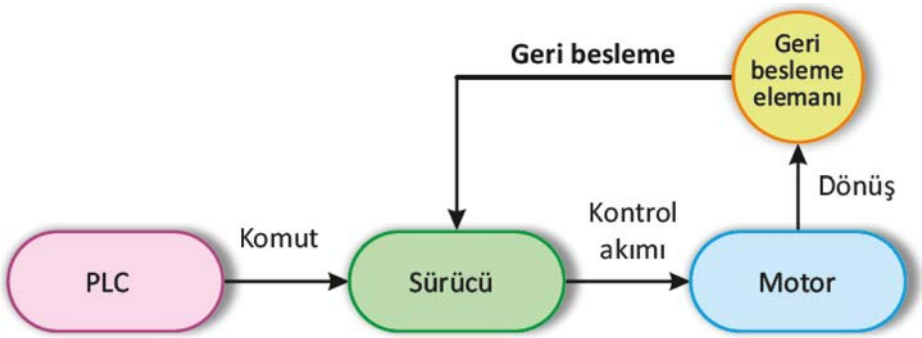
Enkoderler, mutlak ve artımsal olarak iki farklı prensip ile çalışır. Mutlak enkoderli servo motorların sürücülerine pil takılıdır. Bu sayede motorun konum bilgisi, elektrik kesilse bile silinmez. Artımsal enkoderli motorlarda ise elektrik kesildiğinde konum bilgisi silinir. Bu durumda başlangıç konumu, home sensörleri ile bulunur.

Servo motorlarda tork, nominal (belirlenmiş) hıza kadar çok az düşüş gösterir ve hız sınırının aşılmasıyla aniden düşer. Anlık çalışmada, sürekli çalışmaya göre tork daha yüksektir (Görsel 7.38).



Görsel 7.38: Servo motor tork-hız eğrisi

Servo motorlar farklı hızlarda kararlı olarak çalışır, hassas devir ayarı yapabilir ve boyutu küçük olsa bile büyük moment elde edilebilir. Bu motorlar kendilerine özel sürücüyle çalışır ve PLC ile kontrol edilebilir (Görsel 7.39).



Görsel 7.39: Servo motor ve kontrol sistemi

Servo motorlar; CNC tezgâhları, robotlar, askerî ekipmanlar, tıbbi cihazlar, laboratuvar ekipmanları, elektrikli otomobiller, endüstriyel ve ev tipi makineler gibi birçok alanda kullanılır.

7.4.1. Servo Motor Sürücüleri

Servo motor sürücüleri, enkoderden gelen geri besleme değerine göre motorun hız, konum ve torkunu kontrol eden cihazlardır. Güç ve kontrol devresinden oluşur. Güç devresinde besleme doğrultma diyotları, filtre kondansatörleri ve invertör transistörleri bulunur. Kontrol devresinde ise dijital sinyal işlemci ve programlanabilir kapı dizisi devresi vardır.



Görsel 7.40: Servo motor sürücüsü

Parametre girişleri için sürücü üzerinde kontrol paneli ve haricî bağlantıların yapıldığı bağlantı bölümleri mevcuttur. Bağlantı bölümünde güç, motor, frenleme direnci, PLC, enkoder ve PC bağlantıları için konnektörler bulunur. Konnektör girişlerine haricî cihazların bağlantısı, uygun kablolar yardımıyla bu kısımdan yapılır (Görsel 7.40).

Servo motorlarda güç ve enkoder bağlantılarının sağlandığı iki adet giriş soketi veya soketli kablo bulunur. Giriş soketli sistemlerde güç soketi pinleri kalın, enkoder soket pinleri incedir. Özel bir kabloyla bu soketlerden motor sürücüsüne bağlantı yapılır. Soket kablolu motorlar doğrudan uygun sürücüye bağlanır (Görsel 7.41).



Görsel 7.41: Soket kablolu servo motor

7.4.2. Sürücü Bağlantıları

Servo sürücüler, motorları dört farklı modda çalıştırabilir. Bu modlar şunlardır:

Pozisyon Modu: Sürücü, motoru istenilen pozisyonu referans alarak çalıştırır. Varsayılan moddur.

Hız Modu: Sürücü, motoru istenilen hızı referans alarak çalıştırır.

Tork Modu: Sürücü, motoru istenilen torku referans alarak çalıştırır.

Karma Modlar: İstenildiği anda mod değişimi yapılarak pozisyon, hız ve tork modlarından birisinde çalışma yapılır.

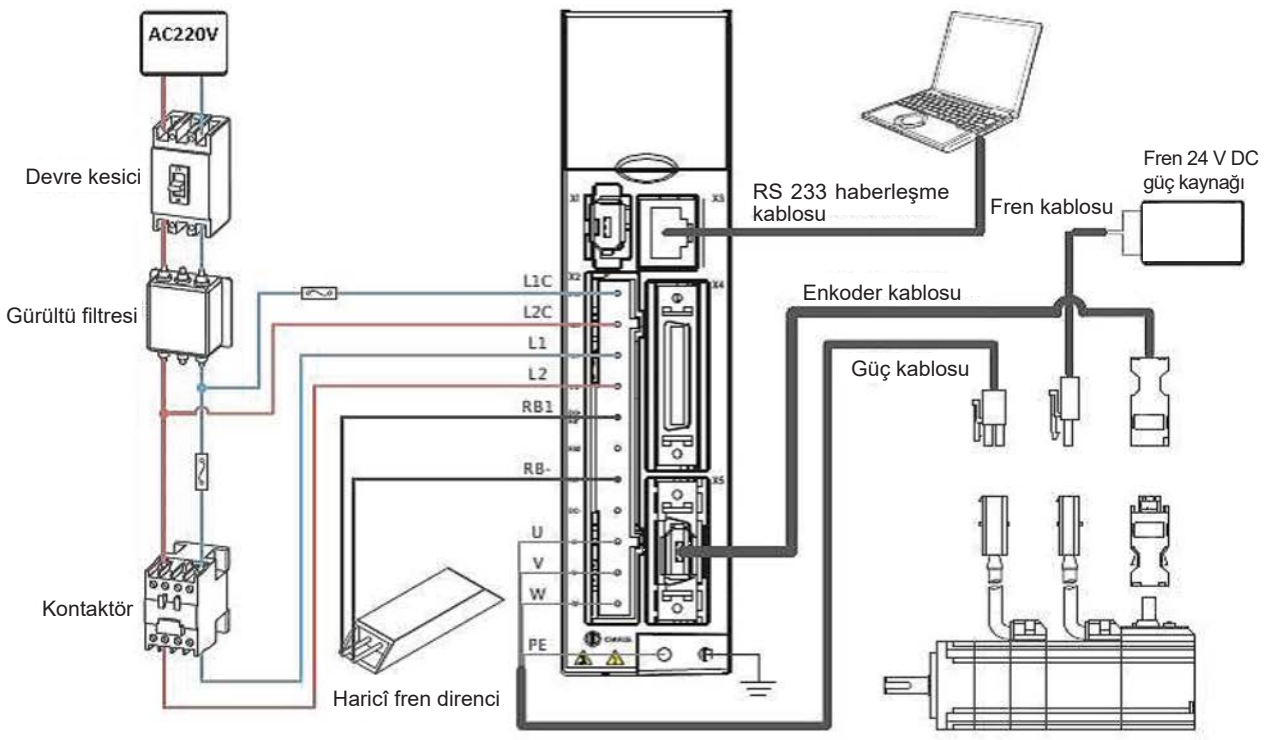
Servo sürücülerde kontrol ve güç bağlantıları, frenleme direnci bağlantıları, haberleşme bağlantıları ve topraklama bağlantısı bulunur (Tablo 7.13). Haberleşme bağlantıları sürücü modeline göre farklılık gösterir. Bazı model sürücülerin dâhilî hafızalarına konum bilgileri saklanarak bu konumlara gitmesi sağlanabilir.

Tablo 7.13: Servo Motor Bağlantı Uçları

<table>
<tr><th>X1</th><th colspan="3">X2</th><th>X3</th><th>X4</th><th>X5</th></tr>
<tr><td rowspan="9">Ayrılmış konnektör</td><td rowspan="9">L1C
L2C
L1
L2
DC+/RB1
RB2
RB-
DC-
U
V
W</td><td>Pin</td><td>Fonksiyon</td><td rowspan="9">RS 232 haberleşme konnektörü</td><td rowspan="9">O (giriş-çıkış) kontrol konnektörü</td><td rowspan="9">Enkoder konnektörü</td></tr>
<tr><td>L1C – L2C</td><td>Kontrol güç girişi</td></tr>
<tr><td>L1 – L2</td><td>Sürücü güç girişi</td></tr>
<tr><td>DC+ – DC-</td><td>DC bus + ve DC bus -</td></tr>
<tr><td>RB1 – RB-</td><td>Haricî frenleme direnci girişi</td></tr>
<tr><td>RB2</td><td>Dâhilî frenleme direnci girişi</td></tr>
<tr><td>U – V – W</td><td>Motor güç çıkış uçları</td></tr>
</table>

Görsel 7.42'de AC servo motorun devreye bağlantısı görülmektedir. AC şebeke üzerinden sürücü güç ve motor konnektörüne enerji girişi yapılmıştır. İstenirse bu kısımdan haricî fren direnci de bağlanabilir. Aynı şekilde servo motor güç girişi de buradan yapılır.

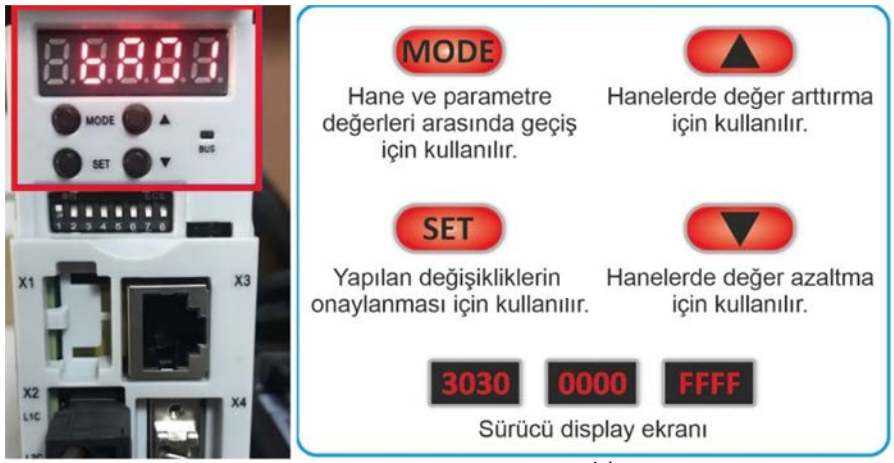
Enkoder kablosu ile sürücü ve motor enkoderi arasında bağlantı sağlanır. Sürücünün sahip olduğu haberleşme portuna göre uygun kablo kullanılarak haberleşme bağlantısı yapılır. PLC ise X4 konnektörüne bağlanır.



Görsel 7.42: Servo motorun sürücü ile devreye bağlanması

7.4.3.Servo Motor Sürücüsünün Kontrol Paneli

Kontrol paneli motorun güç, hız, konum, tork gibi ayarlarının yapılmasını sağlayan paneldir. Ayar tuşları ve göstergeden oluşur (Görsel 7.43:a). Panel üzerindeki tuşlar sayesinde motor parametreleri bu kısımdan ayarlanır (Görsel 7.43:b).



Görsel 7.43: Sürücü kontrol paneli ve panel tuşları

7.4.4. Motor Parametreleri ve Parametre Ayarları

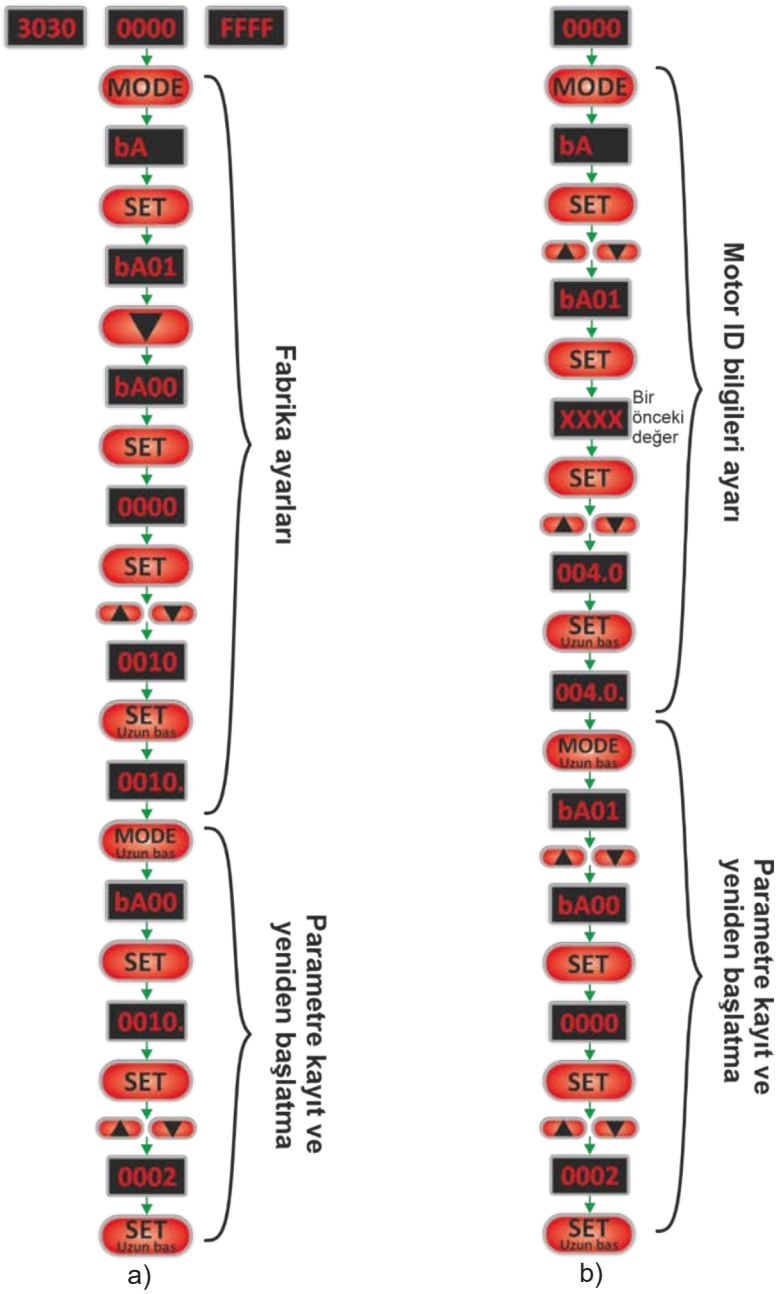
Servo motorun kontrolü için sürücüde tanımlı pek çok parametre bulunur. Parametrelerin kullanım şekli ve açıklamaları sürücü teknik belgelerinde belirtilir. Tek tek ayarlamanın yanı sıra kolay kullanım parametreleri ile hızlıca tanımlama da yapılabilir. Örnek bir sürücü için kolay kullanım parametreleri Tablo 7.14'te verilmiştir.

Tablo 7.14: Servo Motor Sürücüsünün Kolay Kullanım Parametreleri

Parametre	Açıklama	Fabrika Değeri
bA00	**Parametre Kayıt** **1**: Kontrol ve motor parametrelerini kaydeder. **2**: Parametreleri kaydeder ve sürücüyü yeniden başlatır. **3**: Servo sürücüyü yeniden başlatır. **10**: Kontrol parametrelerini fabrika değerlerine döndürür.	0000
bA01	**Motor Tipi Seçimi** Kullanılan motor ID'si girildikten sonra BA00=2 yapılarak kaydedilir.	404b
bA02	**Komut Tipi Seçimi** **0**: CW/CCW pulse train modu **1**: P/D pulse yön modu **2**: A/B faz kontrol master/slave modu **6**: Analog hız modu AIN1 **7**: Analog hız modu AIN2 **8**: Haberleşme modu **9**: Pozisyon tablo modu	1
bA03	**Elektronik Dişli Çarpan Değeri** BA02=0-1-2 modları için geçerlidir.	1000
bA04	**Elektronik Dişli Bölen Değeri** BA02=0-1-2 modları için geçerlidir.	

Kullanılacak servo motor ve sürücü için basit parametre ayar sıralaması Tablo 7.14'ten faydalanılarak aşağıdaki gibi yapılır.

- bA00 parametresine 10 değeri girilerek fabrika değerleri ayarlanır (Görsel 7.44:a).
- bA01 parametresiyle servo motor ID değerleri ayarlanır. Motor tanıtımı için üreticinin bilgi sayfasından kullanılacak motor ID bilgisi alınmalıdır. 400 W servo motor için ID 4.0 olarak verilmiş ve işlem sırasına göre bA01 parametresine girilmiştir (Görsel 7.44:b).
- bA02 parametresiyle komut tipi ayarlanır. PLC ile kontrol için (P/D) pals/yön değeri ayarlanır (bA02=1).
- bA03 parametresiyle 10000 pals/tur için elektronik dişli çarpan değeri ayarlanır (bA03=8192).
- bA04 parametresiyle 10000 pals/tur için elektronik dişli bölen değeri ayarlanır (bA04=1250).
- bA00 parametresiyle yapılan ayarlar kaydedilir ve sürücünün yeniden başlatılması sağlanır (bA00=2).



Görsel 7.44: Sürücü parametre ayarları

7.4.5. Servo Motorların Tur/Pals Hesabı

Servo motorun PLC veya haricî pals üreteci için tur/pals hesabı aşağıdaki gibi yapılır.

$$\textbf{Elektronik dişli oranı} = \frac{\text{Elektronik dişli çarpan değeri (bA03)}}{\text{Elektronik dişli bölen değeri (bA04)}}$$

bA03 parametre değeri +32768 ile -32767 arasında tam sayı ve bA04 parametre değeri 1 ile +32767 arasında tam sayı olmalıdır. Bu parametre değerleri girilirken 9999 değerine kadar desimal, bu değerden sonra ise heksadesimal şeklinde gösterilir.

Servo motor enkoderi bir tam turda 16 bit yani 65536 pals üretir.

$$\textbf{Tur/pals} = \frac{\text{Servo motor enkoder değeri (65536)}}{\text{Elektronik dişli değeri}}$$

ÖRNEK

Servo motora 10000 pals verildiğinde bir tam tur dönmesi için gerekli olan sürücü parametre ayarları: bA03=8192 bA04=1250

$$\textbf{Elektronik dişli oranı} = \frac{8192}{1250} = \textbf{6,5536} \qquad \textbf{Tur/pals} = \frac{65536}{6,5536} = \textbf{10000}$$

ÖRNEK

Servo motora 2000 pals verildiğinde bir tam tur dönmesi için gerekli olan bA03 ve bA04 parametre ayarları:

$$\textbf{Elektronik dişli oranı} = \frac{\text{Servo motor enkoder değeri (65536)}}{\text{Tur/pals}} = \frac{65536}{2000} = 32,768 = \frac{\text{bA03}}{\text{bA04}}$$

Bu oranı sağlayan değerler bA03=4096 ve bA04=125 parametre olarak girilir.

7.4.6. Servo Motorların PLC İle Kontrolü

Servo motorlar, PLC ile kontrol edilebilir. Bunun için önce PLC'nin sürücüye bağlanması, sonra da programlama editörleriyle uygun programın hazırlanması gerekir.

Servo motor hareketinin başlangıç konumuna **home pozisyon** denir. Motor, hareketlerini home pozisyonuna göre belirler. Motor hareketlerinin sınırlandırılması için limit anahtarları (limit switch) kullanılır. Bu anahtarlar, referans noktası sağlamak amacıyla çift olarak hareketin başlangıç ve bitiş noktalarına yerleştirilir. Limit anahtarı olarak mekanik, endüktif, optik ve hall sensörleri gibi sensörler kullanılır.

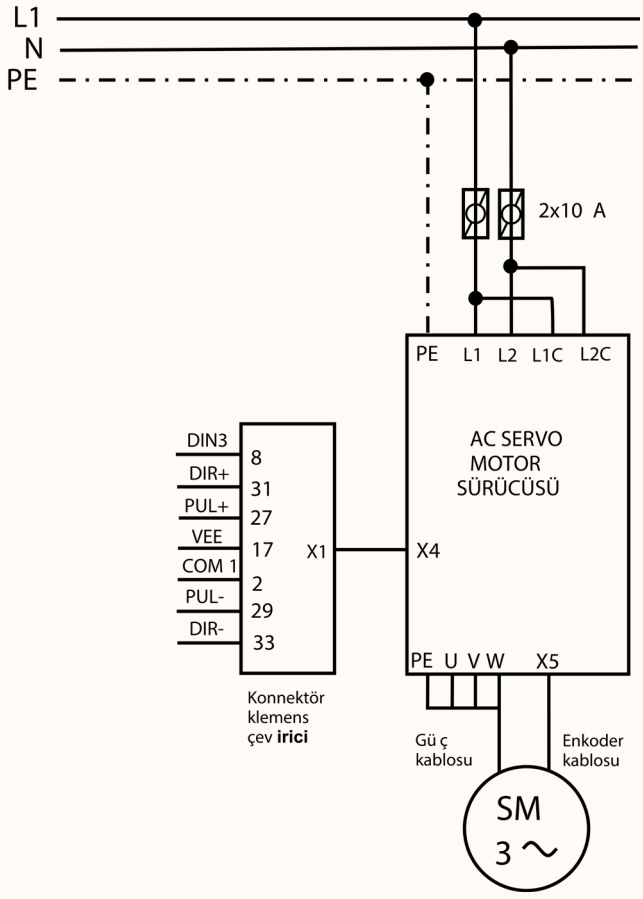
PLSY Komutu: Servo motorun hız ve konumunu ayarlamak için kullanılan komuttur. D10 servo motorun hız değerini, D20 motorun gideceği konum bilgisini pals cinsinden belirtir. Üçüncü değer de kullanılan çıkışı ifade eder.

27. UYGULAMA

SERVO MOTOR SÜRÜCÜSÜ VE GÜÇ BAĞLANTILARI

AMAÇ: **Servo motor sürücüsünün güç bağlantılarını yapmak.**

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Servo motor	400 W, 1,27 Nm, 3000 RPM	1 adet
Servo motor sürücüsü	200-240 VAC, 200-750 W	1 adet
Konnektör klemens çevirici	Sürücü konnek-tör girişine uygun	1 adet



Görsel 7.45: Servo motor sürücüsü ve güç bağlantıları

İŞLEM BASAMAKLARI

1. Kullanılacak malzemelerin bağlantı ve teknik bilgilerini üretici firmaların bilgi sayfasından alınız.
2. Görsel 7.45'teki bağlantı şemasını inceleyiniz.
3. Servo motor sürücüsünün güç bağlantısı için topraklama ve sigorta bağlantılarını yapınız.
4. Servo motorun enkoder kablosunu sürücü üzerinde bulunan enkoder soket girişine (X5), güç kablosunu da güç girişine bağlayınız.
5. Servo motoru haricî kumanda elemanları ile kontrol etmek için kullanılacak olan konnektör klemens çeviriciyi sürücü üzerindeki X4 girişine bağlayınız.

SORU

- Sürücü bağlantılarında dikkat edilecek hususlar nelerdir?

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Sürücü güç besleme bağlantılarını yaptı.	20	
Sınıfı :	2	Motor enkoder bağlantılarını yaptı.	20	
Numarası :	3	Motor güç bağlantılarını yaptı.	20	
ÖĞRETMEN	4	Konnektör klemens çevirici bağlantısını yaptı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

28. UYGULAMA

SERVO MOTOR SÜRÜCÜSÜNÜN TEMEL PARAMETRE AYARLARI

AMAÇ: Servo motor sürücüsünün kontrol paneliyle temel parametre ayarlarını yapmak.

SÜRÜCÜ PARAMETRE LİSTESİ

Parametre	Parametre Açıklaması
bA00=10	Tüm parametreler fabrika varsayılan değerleri olarak ayarlanır.
bA01=4.0	400 W servo motor için ID bilgisi **4.0** olarak girilir.
bA02=1	Kontrol tipi için **1:P/D** pals yön modu seçilir.
bA03=8192	Elektronik dişli çarpan değeri **8192** olarak girilir.
bA04=1250	Elektronik dişli bölen değeri **1250** olarak girilir.
bA00=2	Parametreler kaydedilerek sürücü yeniden başlatılır.

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Servo motor	400 W, 1,27 Nm, 3000 RPM	1 adet
Servo motor sürücüsü	200-240 VAC, 200-750 W	1 adet
Konnektör klemens çevirici	Sürücü konnektör girişine uygun.	1 adet

İŞLEM BASAMAKLARI

1. Kullanılacak malzemelerin bağlantı ve teknik bilgilerini üretici firmaların bilgi sayfasından alınız.
2. Servo motor ve sürücü bağlantılarını yaparak öğretmeninizin kontrolünde sürücüye enerji veriniz.
3. İlk defa enerji verilen sürücünün kontrol paneli ekran değerlerini gözlemleyiniz.
4. Sürücü parametre listesinden faydalanarak sürücüyü fabrika değerlerine ayarlayınız.
5. Servo motor ID bilgisini tespit ederek parametre değerini sürücüye giriniz.
6. Komut tipi için P/D seçimini yaparak parametre değerini sürücüye giriniz.
7. Servo motorun 10000 pals/tur için hesaplamalarını yapınız.
8. Hesaplama sonucu bulduğunuz parametre değerlerini sürücüye giriniz.
9. Yaptığınız tüm parametre değişimlerini kaydederek sürücüyü yeniden başlatınız.
10. Enerjiyi keserek elemanları öğretmeninize teslim ediniz.

SORU

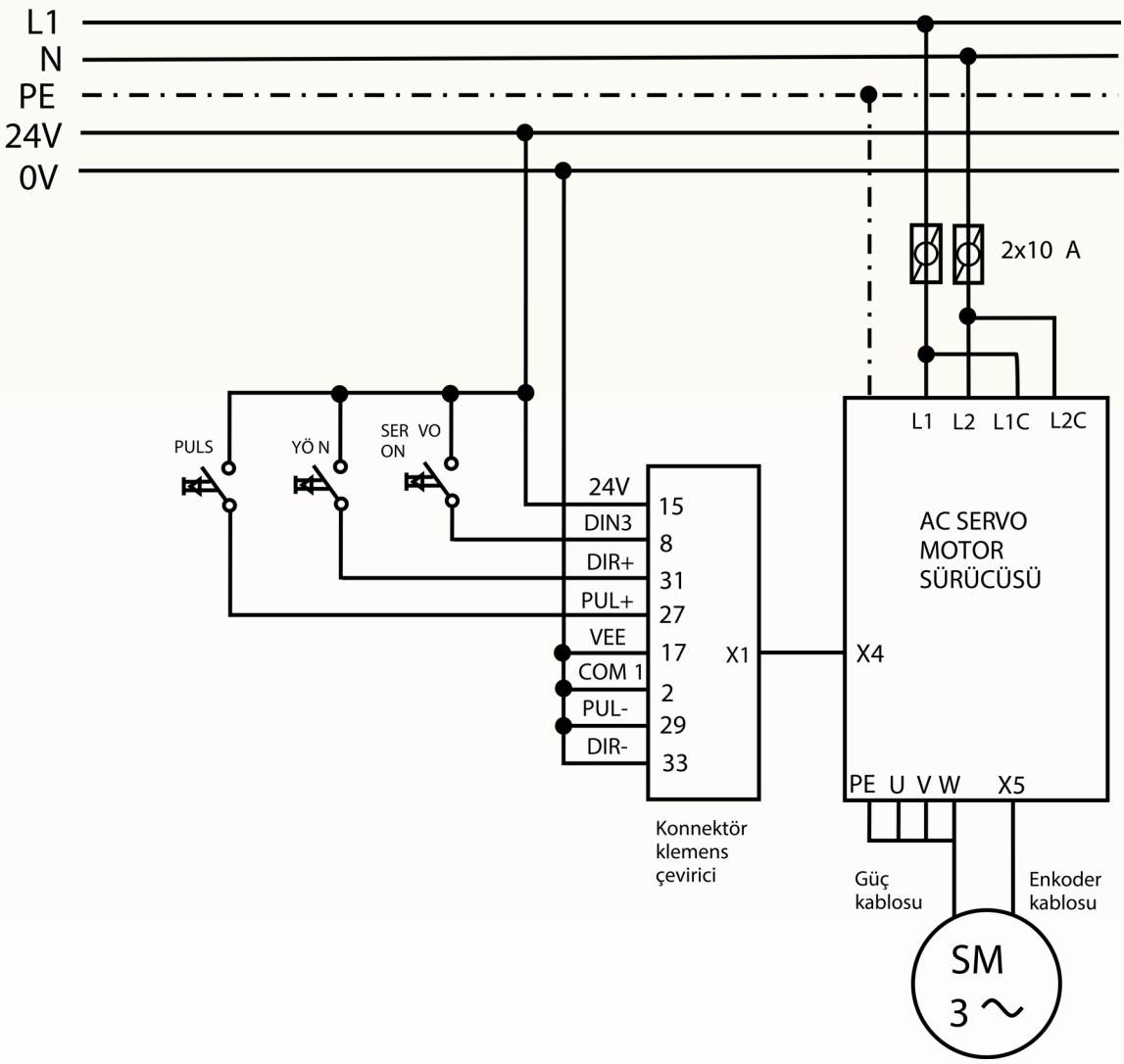
- Pals/tur parametre değerinin yanlış girilmesi ne gibi sorunlar oluşturur? Açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Sürücü bağlantılarını yaptı.	20	
Sınıfı :	2	Güç bağlantılarını yaptı.	20	
Numarası :	3	Pals/tur hesaplamalarını yaptı.	20	
ÖĞRETMEN	4	Parametreleri girdi.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

29. UYGULAMA

SERVO MOTOR SÜRÜCÜSÜNÜN BUTONLARLA ÇALIŞTIRILMASI

AMAÇ: **Servo motorun sürücü bağlantısını yaparak butonlarla çalıştırmak.**



Görsel 7.46: Servo motor sürücüsü ve buton bağlantıları

SÜRÜCÜ PARAMETRE LİSTESİ

Parametre	Parametre Açıklaması
bA00=10	Tüm parametreler fabrika varsayılan değerleri olarak ayarlanır.
bA01=4.0	400 W servo motor için ID bilgisi **4.0** olarak girilir.
bA02=1	Kontrol tipi için **1:P/D** pals yön modu seçilir.
bA03=16384	Elektronik dişli çarpan değeri **16384** olarak girilir.
bA04=50	Elektronik dişli bölen değeri **50** olarak girilir.
bA00=2	Parametreler kaydedilerek sürücü yeniden başlatılır.

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
AC servo motor	400 W, 1,27 Nm, 3000 RPM	1 adet
Servo motor sürücüsü	200-240 VAC, 200-750 W	1 adet
Konnektör klemens çevirici	Sürücü konnektör girişine uygun.	1 adet
Kontrol butonları	Tact switch	3 adet

İŞLEM BASAMAKLARI

1. Kullanılacak malzemelerin bağlantı ve teknik bilgilerini üretici firmaların bilgi sayfasından alınız.
2. Görsel 7.46'da şeması verilen devre bağlantılarını yapınız.
3. Öğretmeninizin kontrolünde devreye enerji veriniz.
4. Sürücü parametre listesinden faydalanarak sürücüyü fabrika değerlerine ayarlayınız.
5. Temel parametreleri ayarlayınız.
6. Servo motorun 200 pals/tur için hesaplamalarını yapınız.
7. Hesaplama sonucu bulduğunuz parametre değerlerini sürücüye giriniz.
8. Yaptığınız tüm parametre değişimlerini kaydederek sürücüyü yeniden başlatınız.
9. Pals butonuna basıp bırakarak servo motorun dönüp dönmediğini gözlemleyiniz.
10. Servo motor dönmüyorsa aktif/pasif butonunu sürekli basılı tutunuz.
11. Pals butonuna tekrar basıp bırakarak servo motorun hangi yöne döndüğünü gözlemleyiniz.
12. Yön butonuna basılı tutunuz ve bu esnada pals butonuna tekrar basıp bırakarak servo motorun bir önceki yönün tersine döndüğünü gözlemleyiniz.
13. Servo motorun mevcut konumuna işaret koyunuz (başlangıç noktası).
14. Yön butonu boşta olacak şekilde pals butonuna yirmi defa basıp bırakarak servo motorun durduğu yeri işaretleyiniz (son nokta).
15. Yön butonuna basınız ve bu esnada pals butonuna yirmi defa basıp bırakarak servo motorun işaret koyduğunuz başlangıç noktasında durduğunu gözlemleyiniz.
16. Devreyi dikkatlice sökerek elemanları öğretmeninize teslim ediniz.

SORULAR

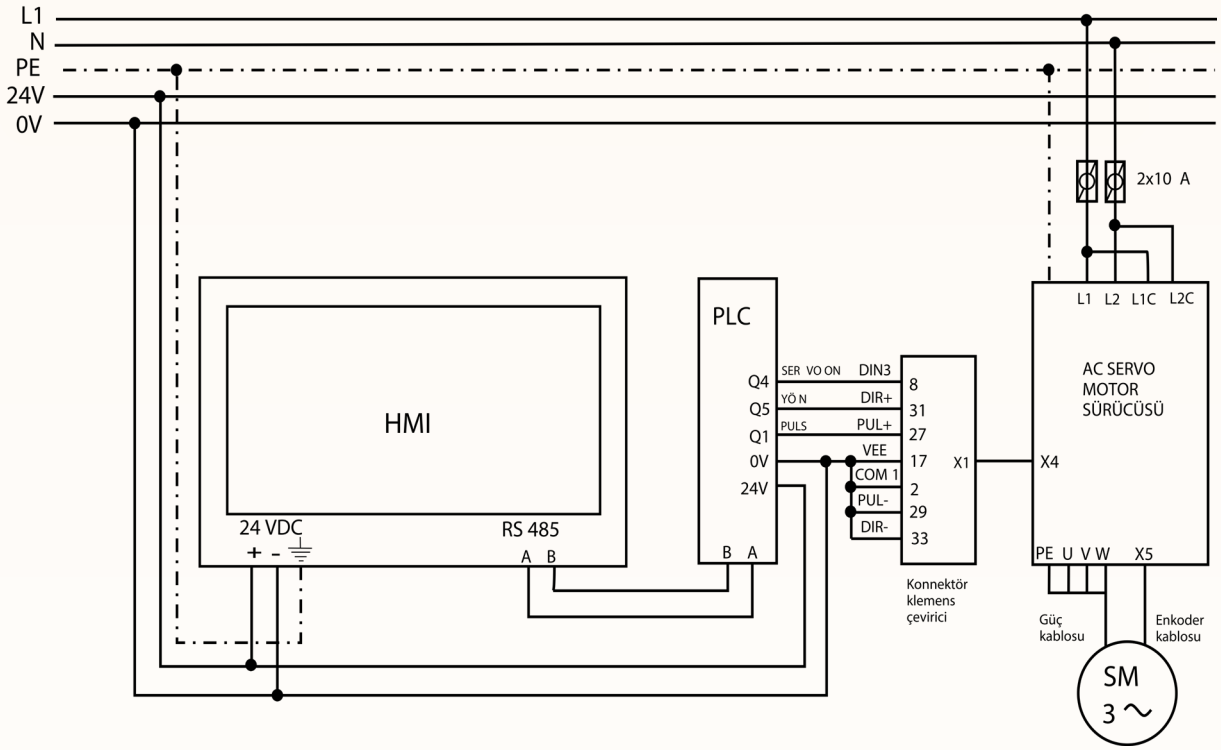
1. Servo motorların tercih edilme nedenlerini açıklayınız.
2. Elektronik dişli çarpan ve bölen parametre ayarlarının faydasını açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Sürücü bağlantılarını yaptı.	20	
Sınıfı :	2	Güç bağlantılarını yaptı.	20	
Numarası :	3	Parametre ayarlarını yaptı.	20	
ÖĞRETMEN	4	Servo motor sürücüsünü butonlarla çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	

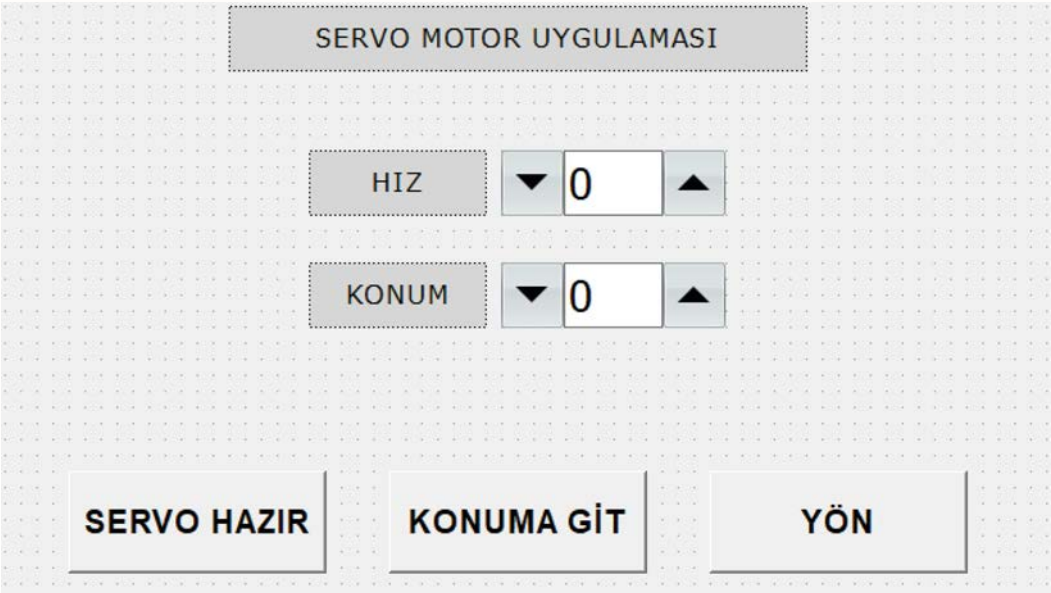
30. UYGULAMA

SERVO MOTORUN HMI VE PLC İLE ÇALIŞTIRILMASI

AMAÇ: **Servo motoru, sürücü bağlantısını yaparak HMI ve PLC ile çalıştırmak.**

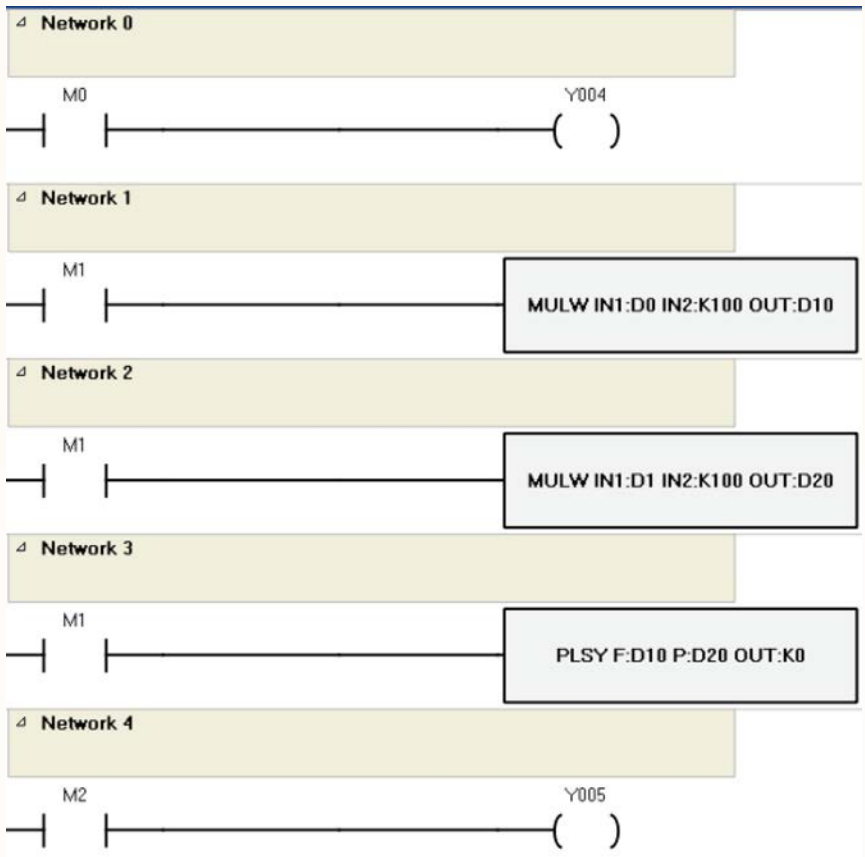


Görsel 7.47: Servo motor sürücüsü, HMI ve PLC devre bağlantıları



Görsel 7.48: Servo motor sürücüsü, HMI ve PLC devre bağlantıları

LADDER DİYAGRAMI



SÜRÜCÜ PARAMETRE LİSTESİ

Parametre	Parametre Açıklaması
bA00=10	Tüm parametreler fabrika varsayılan değerleri olarak ayarlanır.
bA01=4.0	400 W servo motor için ID bilgisi **4.0** olarak girilir.
bA02=1	Kontrol tipi için **1:P/D** pals yön modu seçilir.
bA03=4096	Elektronik dişli çarpan değeri **4096** olarak girilir.
bA04=125	Elektronik dişli bölen değeri **125** olarak girilir.
bA00=2	Parametreler kaydedilerek sürücü yeniden başlatılır.

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Servo motor	400 W 1,27 Nm 3000 RPM	1 adet
Servo motor sürücüsü	200-240 VAC, 200-750 W	1 adet
PLC	Transistör çıkışlı	1 adet
Konnektör klemens çevirici	Sürücü konnektör girişine uygun	1 adet
Kumanda butonları	Ani temaslı	3 adet
Güç kaynağı	24 VDC, 2 A	1 adet

HMI OKUMA YAZMA ADRES TABLOSU						
Eleman İsmi	**Adres Tipi**	**Hafıza**	**Aygıt Tipi**	**Alan**	**Adres**	**Açıklama**
SERVO HAZIR	Okuma/Yazma	Bit	Memory Bits	M0	0	Kalıcı tip buton
KONUMA GİT	Okuma/Yazma	Bit	Memory Bits	M1	1	Kalıcı tip buton
YÖN	Okuma/Yazma	Bit	Memory Bits	M2	2	Kalıcı tip buton
HIZ	Okuma/Yazma	UnsignedInt16	Data Memory	D0	0	Sayıcı
KONUM	Okuma/Yazma	UnsignedInt16	Data Memory	D1	1	Sayıcı
MOTOR ÇALIŞIYOR	Okuma	Bit	Memory Bits	Y0	0	LED

İŞLEM BASAMAKLARI

1. HMI programını hazırlayarak HMI'ye yükleyiniz.
2. Ladder diyagramı hazırlayarak PLC'ye yükleyiniz.
3. Görsel 7.47'de şeması verilen devre bağlantılarını yapınız.
4. Öğretmeninizin kontrolünde devreye enerji veriniz.
5. Sürücüyü fabrika değerlerine getirip temel parametre ayarlarını yapınız.
6. Servo motorun 2000 pals/tur için hesaplamalarını yapıp bulunan değerleri sürücüye giriniz.
7. Yaptığınız tüm parametre değişimlerini kaydederek sürücüyü yeniden başlatınız.
8. HMI üzerinden hız ve konum bilgilerini giriniz.
9. HMI ekranındaki butonları kullanarak servo motoru kontrol ediniz.
10. Devreyi dikkatlice sökerek elemanları öğretmeninize teslim ediniz.

SORU

- Bu devre HMI kullanılmadan yapılabilir mi? Açıklayınız.

ÖĞRENCİ	**DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA**			
Adı Soyadı :	1	HMI programını yazarak HMI'ye yükledi.	20	
Sınıfı :	2	Ladder programını yazarak PLC'ye yükledi.	20	
Numarası :	3	Devre bağlantıları ve sürücü parametre ayarlarını yaptı.	20	
ÖĞRETMEN	4	Devreyi çalıştırdı.	20	
Adı Soyadı :	5	Çalışma ortamını temiz ve düzenli bıraktı.	20	
İmza :		**TOPLAM PUAN**	100	



ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki cümlelerin başındaki boşluğa cümleler doğru ise (D), yanlış ise (Y) yazınız.

1. (…) Ladder diyagramı, yatay çizgilerin üzerine lojik giriş-çıkış elemanları eklenerek oluşturulur.
2. (…) Servo motor sürücüleri motorun hız, konum ve torkunu kontrol eden cihazlardır.
3. (…) HMI, kontrol alanından alınan verilerin bir ekran üzerinden takibini ve değiştirilmesini sağlar.
4. (…) PLC bağlantı adreslemesinde girişler için Q, çıkışlar içinse I harfleri tercih edilir.
5. (…) AC motor sürücüleri iki faz girişli olarak üretilir.

B) Aşağıdaki cümlelerde boş bırakılan yerlere uygun sözcüğü yazınız.

6. Ladder diyagramındaki yatay çizgilerin her birine …… denir.
7. Bit, PLC programlamada en …… veri birimidir.
8. HMI editöründe çalışma sayfası ile ilgili işlemlerin yapıldığı menü …… menüsüdür.
9. HMI programının operatör paneline yüklenmeden önce editör üzerinde çalıştırılmasına …… denir.
10. AC motor sürücü ayarlarına …… ayarları denir.

C) Aşağıdaki soruların doğru seçeneğini işaretleyiniz.

11. Aşağıdakilerden hangisi operatör paneli çeşitlerinden biri değildir?
A) 4.3” B) 7” C) 8.1”
D) 12.1” E) 15.6”

12. Aşağıdakilerden hangisi klasik kumanda devre yapısına benzeyen programlama dilidir?
A) FBD B) GRAPH C) LAD
D) SCL E) STL

13. Aşağıdakilerden hangisi klasik kumandada kullanılan düz zaman rölesi gibi görev yapar?
A) ET B) IN C) R
D) TOF E) TON

14. Aşağıdakilerden hangisi AC motor sürücülerine verilen isimlerden biri değildir?
A) Driver B) İnvertör C) Konvertör
D) Servo E) Sürücü

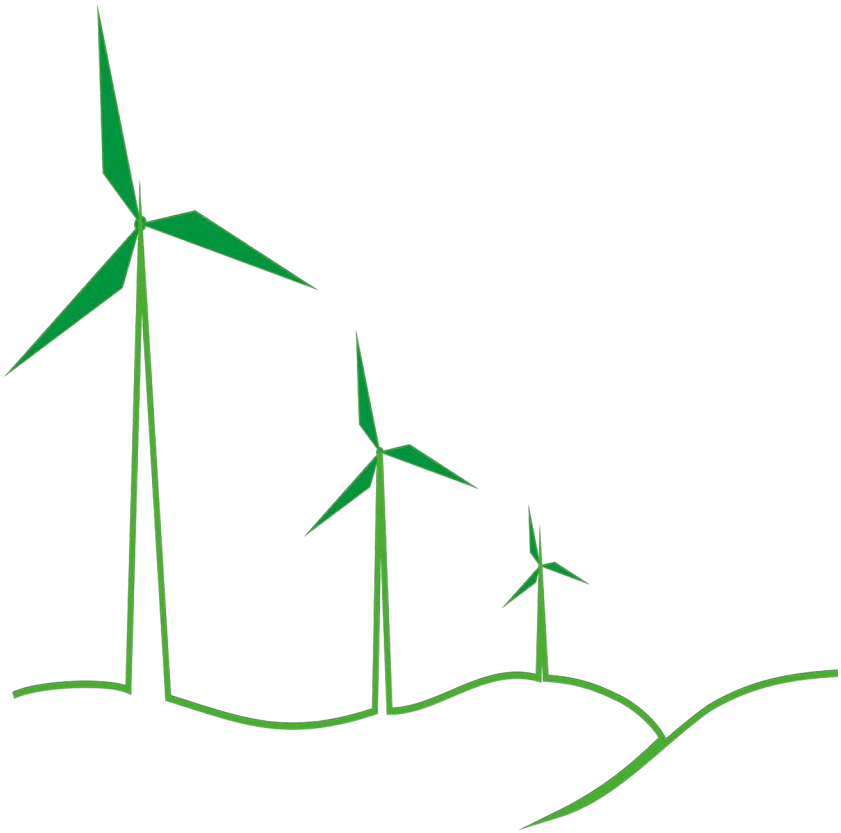
15. Aşağıdakilerden hangisi servo motorlarda kullanılan geri besleme elemanıdır?
A) Enkoder B) Konnektör C) Potansiyometre
D) Redüktör E) Tork

8. ÖĞRENME BİRİMİ

ENERJİ SANTRALLERİNİN ELEKTRİK VE HAREKET SİSTEMLERİ

KONULAR

8.1. ENERJİ SANTRALİNİN ELEKTRİK VE ŞALT SAHASI SİSTEMİ
8.2. GÜNEŞ ROTASI VE IŞINIMI, MERKEZİ HAREKET SİSTEMİ
8.3. TÜRBİN KANAT VE YÖN KONTROLÜ



TEMEL KAVRAMLAR

Enerji santralleri, şalt sahaları, güneş panelleri, güneş panelleri yerleştirme açısı, rüzgâr türbinleri, rüzgâr türbinlerinin kanat kontrolü, yön kontrolü

HAZIRLIK ÇALIŞMALARI

1. Enerji santralleri ve şalt sahaları hakkında bildiklerinizi arkadaşlarınızla paylaşınız.
2. Yenilenebilir enerji kaynakları hakkında bildiklerinizi arkadaşlarınızla paylaşınız.

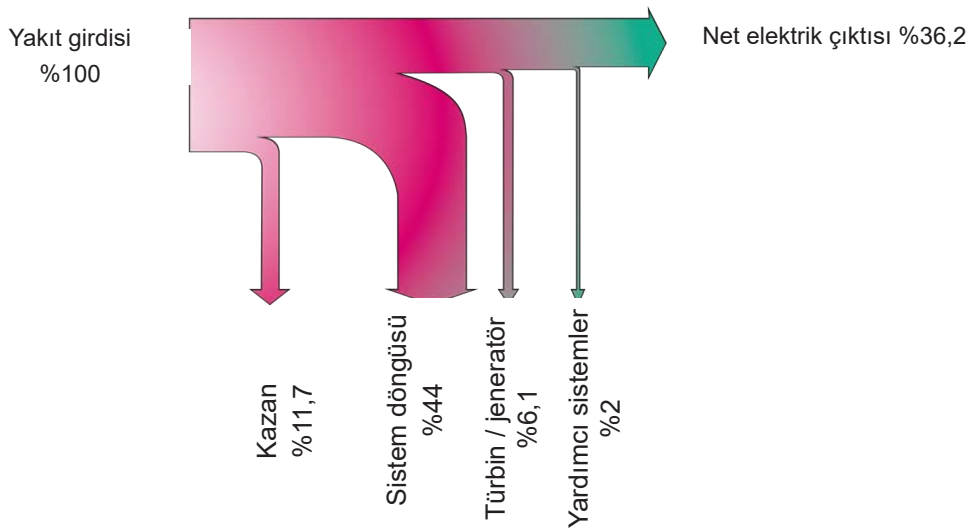
8.1. ENERJİ SANTRALİNİN ELEKTRİK VE ŞALT SAHASI SİSTEMİ

Elektrik santrallerinde üretilen elektrik enerjisinin tüketiciye ulaştırılmadan önce şalt sahası olarak isimlendiren tesislerde bir dizi işlemden geçirilmesi gerekir. Şalt sahaları elektrik enerjisinin yükseltilmesi, düşürülmesi ve enerji iletim güvenliği açısından koruma işlemlerinin yapıldığı yerlerdir.

8.1.1. Elektrik Santrali Sistemi

Enerji santralleri genel olarak hareket ve güneş enerjisini elektrik enerjisine dönüştüren tesisler olarak ifade edilebilir. Hidroelektrik santralleri, termik santraller, rüzgâr ve güneş santralleri bu amaca hizmet eden tesislerdir. Hidroelektrik santralleri suyun potansiyel enerjisini elektrik enerjisine dönüştürürken termik santraller kömür, doğal gaz veya petrol ürünlerinden elde edilen ısı enerjisini hareket ve elektrik enerjisine dönüştürür.

Enerji santrallerinde bu dönüşümün gerçekleşebilmesi için giriş enerjisi ile dönüşümden sonra elde edilen çıktı enerjisi arasında bir dizi enerji dönüştürücü eleman kullanılmaktadır. Kullanılan bu sistem elemanları da enerji santrali giriş ve çıkış enerjileri arasında kayıplara sebep olmaktadır. Aşağıdaki görselde (Görsel 8.1) bir termik santrale ait kayıplar görülmektedir.



Görsel 8.1: Termik enerji santrali kayıpları

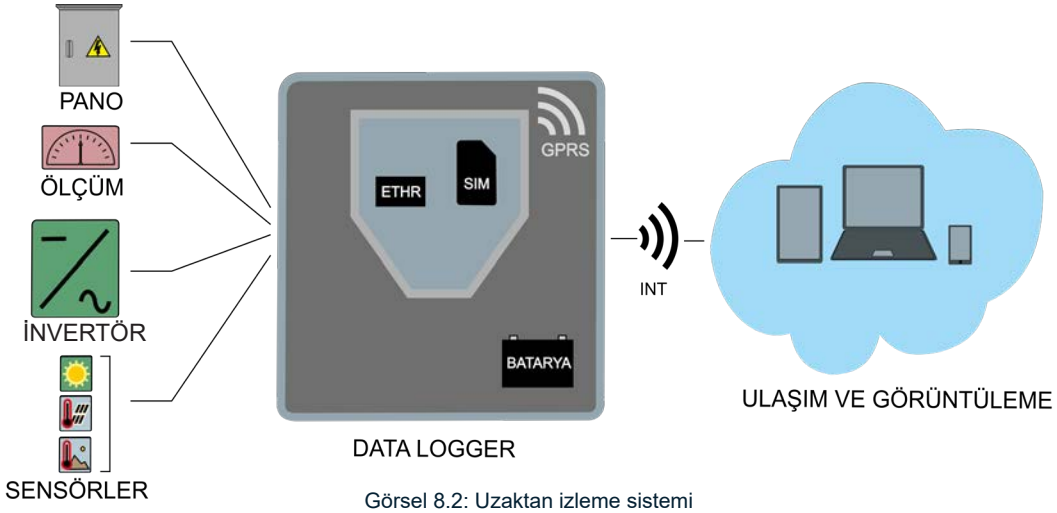
Yukarıdaki görselde termik santral veriminin %36,2 olduğu görülmektedir. Bu değer, termik santrale verilen 100 birimin sadece %36,2'sinin elektrik enerjisine çevrilebildiğini göstermektedir. Düşük olan bu değer, enerji santralinin işletilmesi sürecinde boruların kireçlenmesi, ekipmanların aşınması, kalibrasyonun bozulması veya santral elemanlarından birinin arızalanması sonucunda daha da düşmektedir. Bu nedenle enerji santrallerinin sürekli izlenmesi sistemde oluşacak olumsuzluklara erken müdahale etmeye, önlem almaya ve sistemin verimli kullanılmasına yardımcı olacaktır.

Enerji santrallerinin izlenmesine olanak sağlayan bu sistemler genel olarak "uzaktan izleme sistemleri" olarak isimlendirilmektedir. Bu sistemler enerji santralinin anlık olarak izlenmesinin yanında, müdahalelerin kayıt altına alınması, tarihsel veri tabanı oluşturulması, ayrıntılı analiz ve olası arıza kestirimi yapılmasına olanak sağlamaktadır.

Uzaktan izleme sistemleri özellikle veri depolama, izleme, değerlendirme ve raporlama olanağı sağlamaktadır. Veri depolama, enerji santralinde enerji üretimi için kullanılan sistem elemanlarından alınan bilgilerin belirlenen zaman aralığında kayıt altına alınması olarak ifade edilebilir. Kaydedilen bu bilgilerin web tabanlı bir uygulama yardımı ile izlenebilmesi sağlanmaktadır. İzlemedeki amaç, veri alınan elemanlardan gelen bilgilerin doğru olup olmadığının karşılaştırılmasıdır. Verilerdeki en küçük bir sapma erken müdahale açısından önem arz etmektedir.

Değerlendirme aşaması ise teknik özellikleri bilinen elemanlardan gelen bilgilerin karşılaştırılmalı olarak kontrol altında tutulması şeklinde ifade edilebilir. Uzaktan izleme sistemlerinin gerekli durumlarda uyarı veya hata mesajları gönderebilme özelliği sayesinde sistem arızası ve bakım kaydı otomatik olarak ilgili birime yönlendirilebilmektedir.

Uzaktan izleme sistemlerinin bir diğer özelliği de raporlama aşamasıdır. Raporlama aşaması; enerji santralinin günlük, aylık ve yıllık üretim parametrelerinin değişimlerini, çalışma grafiklerini ve benzeri bilgileri çıktı olarak almayı sağlamaktadır (Görsel 8.2).



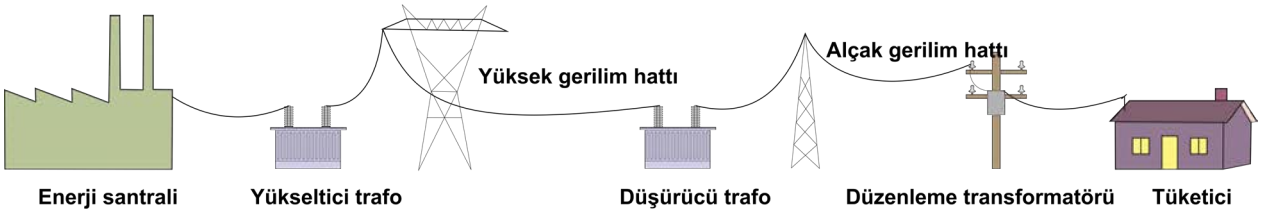
Görsel 8.2: Uzaktan izleme sistemi

Yukarıdaki görselde uzaktan izleme sistemlerinin mimarisi görülmektedir. Sistemin en önemli bileşeni "Datalogger" olarak bilinen veri toplama birimidir. Bu birim, sahadan alınan bilgileri internet bağlantısı ile merkezî veri tabanına ulaştırmaktadır. Herhangi bir nedenle bilgilerin merkezî veri tabanına ulaşamaması ihtimaline karşın, datalogger topladığı verileri belirli bir süre için hafızada tutabilmeli ve internet bağlantısı sağlandığında hafızasındaki bilgileri merkezî veri tabanına aktarabilmelidir.

Uzaktan izleme sistemlerinin önemli bileşenlerinden biri de yazılımdır. Yazılım, bir taraftan belirlenen zaman aralığında hangi verilerin okunacağını, hangi sistem elemanlarının takip edileceğini, elde edilen bilgilerin nereye ve nasıl yazılacağını yönetirken, diğer taraftan elde edilen verileri anlamlı bilgilere çevirerek bilgisayar, cep telefonu veya tabletlerden bu bilgilere ulaşılabilirliği sağlamaktadır.

8.1.2. Enerji Santralinin Şalt Sahası Sistemi

Enerji santrallerinde üretilen elektrik enerjisinin iletim hatları yardımı ile tüketiciye ulaştırılması sağlanır. Enerji santralinde üretilen elektrik enerjisinin gerilim değeri, iletim hattının başında kurulan trafo merkezlerinde yükseltilir ve yüksek gerilimde düşük akımla iletim sağlanır. İletim kayıplarını azaltmak için yüksek gerilim olarak iletilen elektrik enerjisi, tüketiciye ulaştırılmadan önce tekrar orta ve alçak gerilim değerine düşürülerek dağıtılır. İletim hattının başında ve sonunda, elektrik enerjisinin gerilim değerini yükseltme ve alçaltma işleminin yapıldığı bu tesislere **şalt sahası** denir.



Görsel 8.3: Enerji iletim hattı

Şalt sahalarının enerji iletim hatlarının başında ve sonunda olması sebebiyle bu alanlarda meydana gelebilecek bir arıza, elektrik enerjisi aktarımının kesintiye uğramasına neden olacaktır. Bu nedenle şalt sahalarında meydana gelecek arızalara erken müdahale edilmesi kesintisiz enerji iletimi açısından büyük önem teşkil etmektedir. Bu nedenle şalt sahası elemanları düzenli olarak izlenmeli, teknik verilerinden sapma gerçekleşen devre elemanlarına daha arıza gerçekleşmeden müdahale edilmelidir. Kesintisiz enerji iletimi için alınabilecek bir diğer önlem ise en sık karşılaşılan arızalara sebep olan devre elemanlarının yedeğinin anlık olarak devreye alınmasını sağlayan sistemlerin geliştirilmesidir. İletim hatlarının can damarı olan şalt sahalarında en sık karşılaşılan arızaları şöyle sıralayabiliriz:

- Transformatör sargılarında aşırı yükleme sonucu aşırı akım oluşması
- Transformatörlerde soğutma amaçlı kullanılan yağ seviyesinin düşmesi
- Aşırı akım sonucunda transformatör sargılarında meydana gelen sıcaklık artışı
- İç ve dış etmenlerle oluşan ark sonucu malzemelerin zarar görmesi

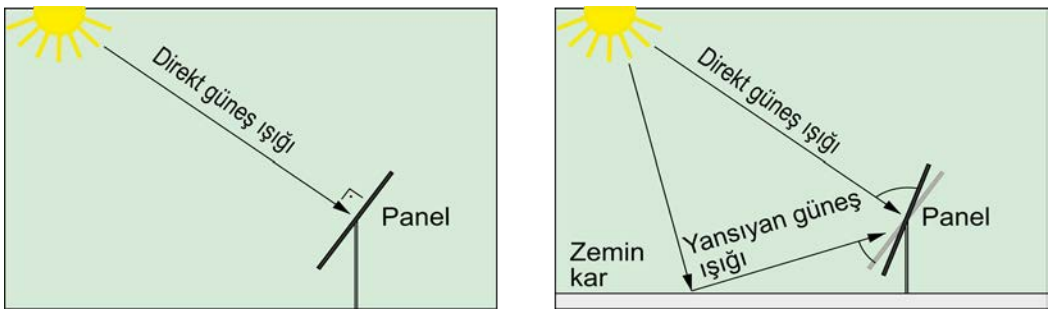
8.2. GÜNEŞ ROTASI VE IŞINIMI, MERKEZÎ HAREKET SİSTEMİ

Günümüz dünyasının en önemli enerji türü olan elektrik enerjisi, büyük oranda fosil yakıtların yakılması ile elde edilmektedir. Bir taraftan sanayileşmeye bağlı olarak enerji ihtiyacı artarken diğer taraftan kısıtlı olan fosil yakıt kaynaklarının ilerleyen süreçte enerji ihtiyacını karşılayamayacağı görülmektedir. Bunun dışında fosil yakıtların çevreye olan olumsuz etkileri neticesinde ortaya çıkan küresel ısınma ve iklim değişiklikleri temiz enerji kaynaklarına olan ihtiyacı artırmaktadır. Bu açıdan bakıldığında güneş pilleri (PV paneller-Güneş pilleri) çevre kirletici etkisi olmayan alternatif enerji kaynaklarının başında gelmektedir. Güneş pilleri, güneş radyasyonunu hareketli parça kullanmadan, gürültüsüz bir şekilde elektrik enerjisine çevirmektedir.

Güneş pillerinin bu işlemi gerçekleştirebilmesi için en önemli nokta güneş ışığının güneş piline 90°lik bir açı ile gelmesini sağlamaktır. Açı 90°den saptıkça güneş pilinden alınan güç azalacaktır.

Ülkemizde son dönemlerde sıkça karşımıza çıkan güneş pillerinin genellikle sabit bir açı ile yerleştirildiğini görmekteyiz. Güneşin gün içerisindeki hareketi dikkate alındığında sabit bir güneş pili panelinin günün her saatinde güneş ışığını 90°lik bir açı ile alması mümkün değildir. İşte tam bu noktada güneş takip sistemleri devreye girmektedir. Güneş takip sistemleri, güneşin hareketini sürekli takip ederek elde ettiği değerlere göre panel açısını otomatik olarak değiştirmektedir; ancak panelden maksimum verimi almak için bu takip sistemi bile yeterli değildir.

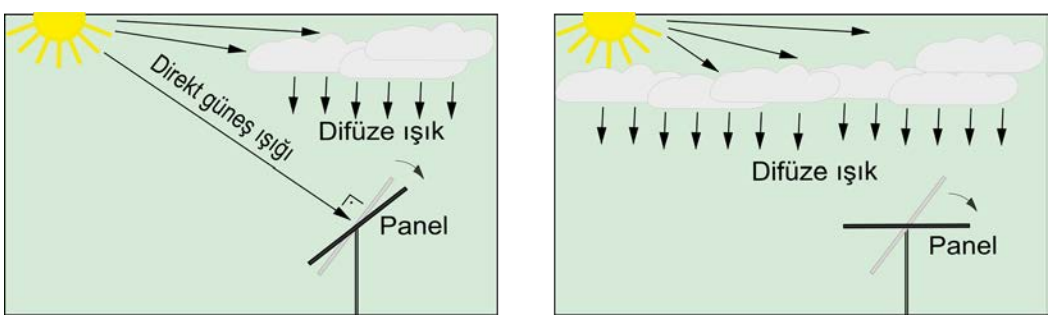
Güneş takip sistemlerinin farklı hava koşullarını dikkate alarak panel açısını belirlemesi büyük önem taşımaktadır.



Görsel 8.4: Farklı atmosfer koşullarında güneş ışığı açısı

Yukarıdaki 1. görselde (Görsel 8.4) açık gökyüzü koşullarında güneş ışığının direkt güneş pili paneline gelmesiyle, 2. Görselde ise panele direkt gelen ve karlı zeminden yansıyan güneş ışığı ile elektrik enerjisi üretimi gerçekleştirilmektedir.

Güneş paneli, aynı şekilde az bulutlu hava koşullarında hem direkt hem de bulutlardan difüzyon yoluyla gelen güneş ışınları ile elektrik enerjisi üretimini gerçekleştirmektedir. Tamamen kapalı hava koşullarında güneş ışığı bulutlar tarafından kesileceğinden sadece difüzyona uğrayan elektrik enerjisi üretecektir. Bu durumda güneş panelinin yatay pozisyona geçirilmesi gerekmektedir (Görsel 8.5).



Görsel 8.5: Kapalı atmosfer şartlarında güneş ışığı açısı

Değişik hava koşulları dikkate alındığında güneş takip sistemlerinin sadece güneşin hareketlerini takip etmesi yeterli olmamaktadır. Maksimum verim elde edebilmek için güneş pili panelinin güneşin en parlak olduğu noktaya çevrilmesi büyük önem taşımaktadır.

8.2.1. Güneş Pili Panelinin Açısını Hesaplama

Dünya'nın yuvarlak olması nedeniyle güneş ışınlarının Dünya'nın her bölgesine aynı açı ile gelmesi mümkün değildir. Bu nedenle dünyanın farklı bölgelerine yerleştirilecek güneş pillerinin güneş ışınlarını dik alabilmesi için farklı açılarda yerleştirilmesi gerekmektedir. Aynı durum ülkemiz için de geçerlidir. İstanbul'da yerleştirilen bir panel ile Hatay'da yerleştirilen bir panelin aynı açıda olması mümkün değildir. Güneş ışınlarından daha iyi yararlanabilmek için panelin yerleştirileceği yerin enlem değeri dikkate alınarak bir hesap yapılması gerekmektedir. Ülkemizin Kuzey Yarım Küre'de olduğu dikkate alınarak basit bir yöntemle bu hesaplama için aşağıdaki formül kullanılabilir.

ÖRNEK

Panel açısı= Enlem değeri x 0,87+3,1

- Örneğin Ankara'nın enlem değeri 39,92'dir.

Panel açısı=39,92 x 0,87+3,1

Panel açısı=37,8° olarak hesaplanır.

- Aynı işlemi Hatay için yapalım.

Hatay'ın enlem değeri 36,40'tır.

Panel açısı=36,40 x 0,87+3,1

Panel açısı=34,8° olarak hesaplanır.

Yukarıdaki hesaplamalara göre güneş ışınlarından maksimum verim elde edebilmek için güneş panelleri Ankara'da 37,8° , Hatay'da ise 34,8° açı ile yerleştirilmelidir.

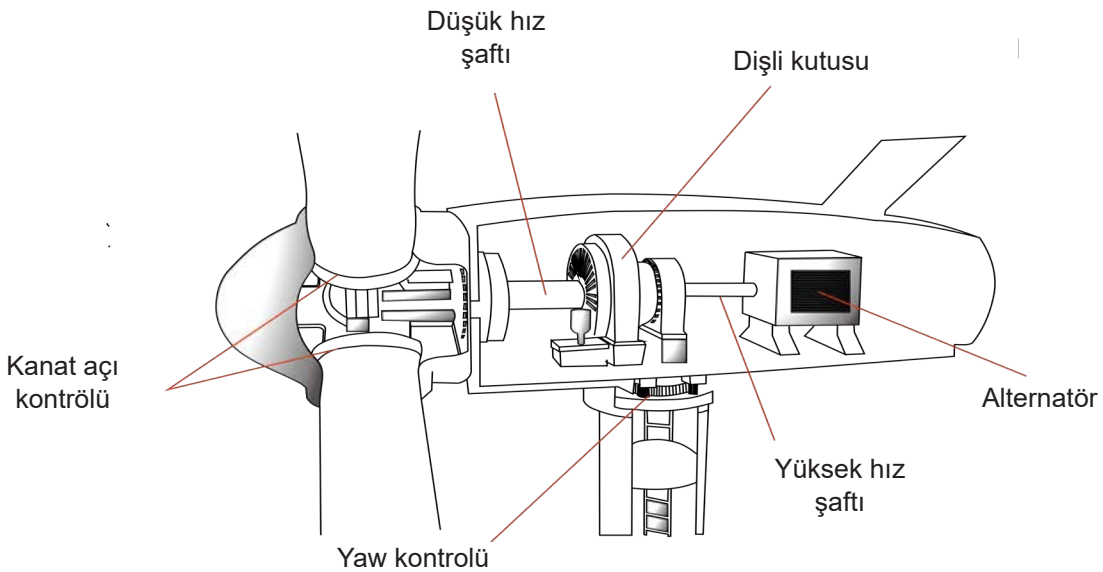
SORU

Artvin 41,18, İzmir 38,43 enlem değerine sahiptir. Burada kurulacak bir güneş panelleri için, panel açı değerini hesaplayınız. Hesapladığınız açı değerleri arasındaki farkın sebebini arkadaşlarınızla tartışınız.

*Örneğe uygun olarak soruyu çözünüz.

8.3. RÜZGÂR SANTRALLERİ

Rüzgâr santralleri yenilenebilir enerji kaynaklarından biridir. Bu santraller elektrik enerjisi üretebilmek için rüzgâr türbinlerine ihtiyaç duyar. Rüzgâr türbinleri rüzgârın kinetik enerjisini mekanik enerjiye dönüştürür. Elde edilen mekanik enerji ise rüzgâr türbini içerisinde bulunan bir alternatör yardımı ile elektrik enerjisine dönüştürülür.



Görsel 8.6: Rüzgâr türbini

Yukarıdaki görselde (Görsel 8.6) rüzgâr türbininin genel yapısı görülmektedir. Bir rüzgâr türbini; kule, kanatlar, düşük hız şaftı, dişli kutusu, yüksek hız şaftı, frenleme sistemi, pitch sistemi, yaw sistemi, alternatör gibi bileşenlerden meydana gelir. Rüzgâr türbini üreticilerinin özel uygulamaları nedeniyle küçük farklılıklar olsa da genel yapı bileşenleri bunlardır.

Kule, rüzgâr türbinini tüm bileşenleri ile taşıyan kısımdır. Çelik kafes veya silindirik yapıda olabilir. Kule yükseklikleri 20 metreden başlayıp kanat çapına göre 100 metreye kadar çıkabilmektedir.

Kanatlar, rüzgârın kinetik enerjisini mekanik enerjiye çevirerek önce dişli kutusuna, oradan da alternatöre gönderen birimdir. Kanatlar, hafif olmaları gerektiğinden genel olarak polimerden üretilse de küçük güçlü türbinlerde çelik veya alüminyum malzemeler de kullanılmaktadır. Kanatlarda en önemli noktalardan biri de kanat sayısıdır. Türbinlerde kanat sayısı dizayn durumuna göre 2-40 arasında değişebilmektedir. Yüksek kalkınma momenti gerektiren yerlerde kanat sayısının fazla olması gerekirken daha düşük kalkınma momenti gerektiren yerlerde kanat sayısı daha azdır. Örneğin su pompalama uygulamalarında kullanılan türbinlerde kanat sayısı fazla iken elektrik üretiminde kullanılan türbinlerde kanat sayısı azdır. Günümüzde elektrik enerjisi üretiminde daha çok üç kanatlı rüzgâr türbinleri kullanılmaktadır.

Düşük hız şaftı, kanatlar ile dişli kutusunu birbirine bağlayan bileşendir. Rüzgârın etkisi ile hareket eden kanadın hareketini aynı orandaki hızla dişli kutusuna aktarma görevini yapar.

Dişli kutusu, kanatların düşük dönme hızını yüksek dönme hızına dönüştüren bileşendir. Orta güçteki bir rüzgâr türbininde kanatlar yaklaşık olarak dakikada 18-35 devir hızla döner. Bu hız, alternatörün elektrik enerjisi üretebilmesi için yeterli değildir. Alternatörün elektrik enerjisi üretebilmesi için en az 1200 devir/dakika hızla dönmesi gerekir. Dişli kutusunun sistemdeki görevi, içerisinde bulunan değişik çaplardaki dişliler yardımı ile kanatlardan gelen düşük dönme hızını, alternatörün ihtiyacı olan en az 1200 devir/dakika hızına çıkarmaktır. Arabalarda kullanılan vites kutusuna benzer.

Yüksek hız şaftı, dişli kutusu ile alternatörü birbirine bağlayan bileşendir. Dişli kutusu çıkışından elde edilen yüksek hızı alternatöre aktarır.

Alternatör, yüksek hız şaftından gelen mekanik hareketi elektrik enerjisine çeviren sistem bileşenidir. Rüzgâr türbini içerisindeki alternatörde üretilecek elektrik enerjisini etkileyecek tek değer alternatörün dönüş hızıdır. Hız ne kadar artarsa üretilecek elektrik enerjisi de o oranda artacaktır. Bu nedenle üretilecek elektrik enerjisinin çok yüksek hızdan dolayı tehlikeli değerlere ulaşmasına, kontrol sistemleri yardımı ile engel olunması ve devir sayısının sürekli kontrol altında bulundurulması, sabit tutulması gerekmektedir.

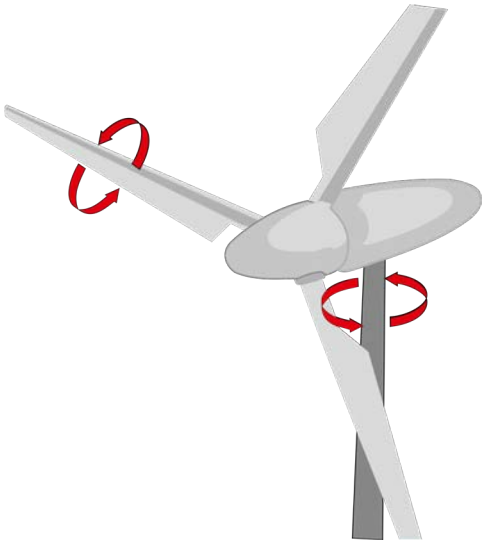
Frenleme sistemi, kanat hızının azaltılması için kullanılır. Atmosferik etkiler nedeniyle türbin kanatlarının aşırı yüksek hızda dönmesi, elektrik üretiminin kontrolsüz artışına neden olmasının yanında türbinin mekanik sisteminin zarar görmesine de neden olacaktır. Bu nedenle bazen kanat hızlarının azaltılmasına ihtiyaç duyulmaktadır. Frenleme sistemleri ihtiyaç duyulması hâlinde türbin kanat hızının düşürülmesini sağlamaktadır.

Mekanik ve aerodinamik olmak üzere iki tip fren sistemi vardır. Mekanik fren sistemi, dişli kutusuna yerleştirilen bir diske baskı yapan balata sisteminden oluşmaktadır. Aerodinamik fren sistemi ise kanat açısını değiştirmek sureti ile frenleme (pitch kontrol) yapmaktadır.

8.3.1. Rüzgâr Santrallerinin Kontrol Sistemi

Rüzgâr türbinlerinde farklı kanat tipleri kullanılsa da enerji kaynağı rüzgârın kinetik enerjisi olduğundan maksimum güç için rüzgâr hızının önemi büyüktür. Rüzgâr hızının çok az olması yeterli dönüş hızı sağlamazken çok yüksek rüzgâr hızı da türbin mekanik sisteminin zarar görmesine neden olacaktır. Bu nedenle rüzgâr türbinlerinin kanat hızının ideal aralıkta sabit tutulması ve rüzgâra dik konumda olması büyük önem taşımaktadır. Bu amaçla çeşitli kontrol yöntemleri geliştirilmiştir.

Bu kontrol yöntemlerinin en önemlileri, rüzgâr hızına bağlı olarak kanat açısının değiştirilmesi ve türbin kanatlarının sürekli olarak rüzgâra dik pozisyonda kalmasını sağlamaktır. Görsel 8.7'de bu kontroller görülmektedir.

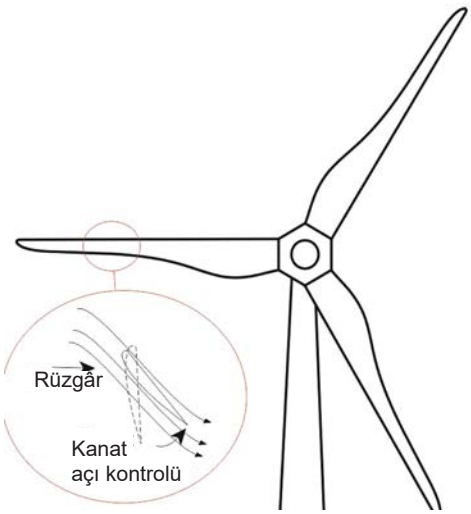


Görsel 8.7: Rüzgâr türbininin kanat ve yön kontrolü

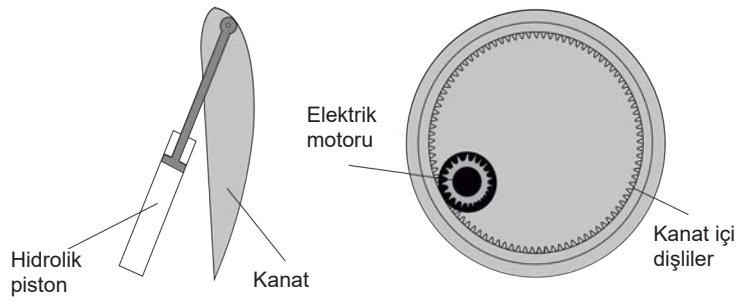
8.3.2. Rüzgâr Türbininin Kanat Açı Kontrolü

Kanat açısının rüzgâr hızına bağlı olarak kontrol edilmesi **pitch kontrolü** olarak isimlendirilir. Pitch kontrolü türbin kanatlarının aynı anda 90^0ye kadar hareket ettirilmesini sağlayan sistemdir. Bu sistem, rüzgâr hızına bağlı olarak kanat açısını değiştirmek sureti ile kanat dönme hızının sürekli ideal sınırlar içerisinde kalmasını sağlar (Görsel 8.8).

Rüzgâr türbininin kanat açı kontrolü için genel olarak elektrik motoru veya hidrolik sistem kullanılmaktadır. Görsel 8.9'da bu sistemler görülmektedir.



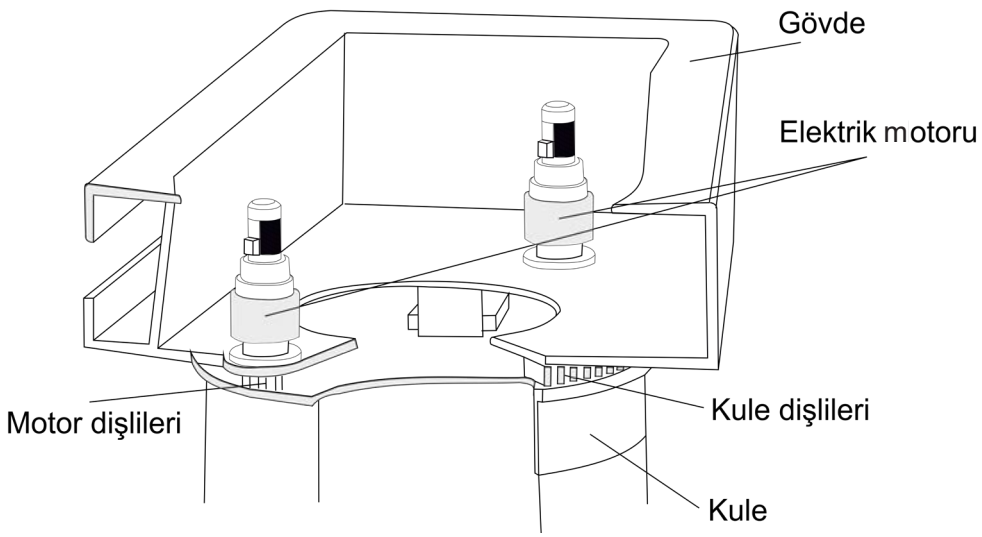
Görsel 8.8: Rüzgâr türbininin pitch (kanat açı) kontrolü



Görsel 8.9: Rüzgâr türbininin hidrolik ve elektrik motorlu kanat açı kontrolü

8.3.3. Rüzgâr Türbininin Yön Kontrolü

Rüzgâr enerjisinden maksimum verim elde edebilmek için türbin kanatlarının sürekli olarak rüzgâra dik pozisyonda kalması gerekmektedir. Bunu sağlayan sistem yön kontrol, saptırma veya yaw sistemi olarak isimlendirilir. Yön kontrol sistemi, bir dişli kutusu ve yön kontrol motoru yardımı ile türbini döndürerek, kanatların sürekli olarak rüzgâra dik pozisyonda kalmasını sağlar. Bu sistem genel olarak anemometreden alınan sensör bilgisine bağlı olarak bir elektrik motoru yardımı ile türbin kanatlarının rüzgâra dik pozisyonda kalmasını sağlar. Aşağıdaki görselde (Görsel 8.10) bu kontrol sistemi görülmektedir.

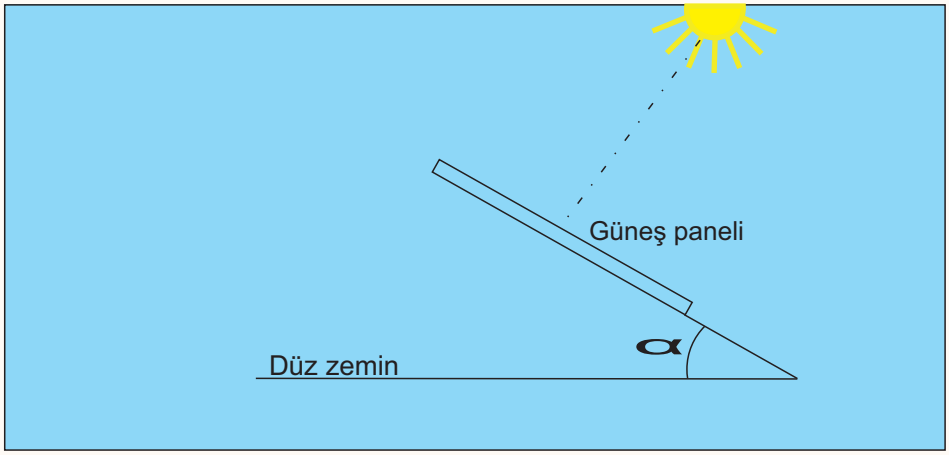


Görsel 8.10: Rüzgâr türbininin yön kontrol sistemi

1. UYGULAMA

GÜNEŞ PANELİ AÇI HESAPLAMA UYGULAMASI

AMAÇ : Güneş ışınlarının, güneş paneline doğru açı ile gelmesini sağlayacak şekilde kurulması ve bölgeye göre açı hesaplaması.



Görsel 8.11: Güneş panelinin yerleştirilmesi

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Güneş paneli	Herhangi bir güçte	1 adet
Metal iskelet	Açı ayarına uygun yapıda	1 adet
Açı ölçer	-	1 adet

İŞLEM BASAMAKLARI

1. Görsel 8.11'de verilen yapıya göre metal iskeleti yerleştiriniz.
2. Güneş panelini iskelet üzerine oturtunuz.
3. Bulunduğunuz bölgenin enlem değerini internet yolu ile öğreniniz.
4. Konu içerisinde anlatılan yöntemle gerekli hesaplamayı yapınız.
5. Hesaplama sonucunda elde ettiğiniz panel açı değerine göre güneş panelini açısını ayarlayınız.

SORULAR

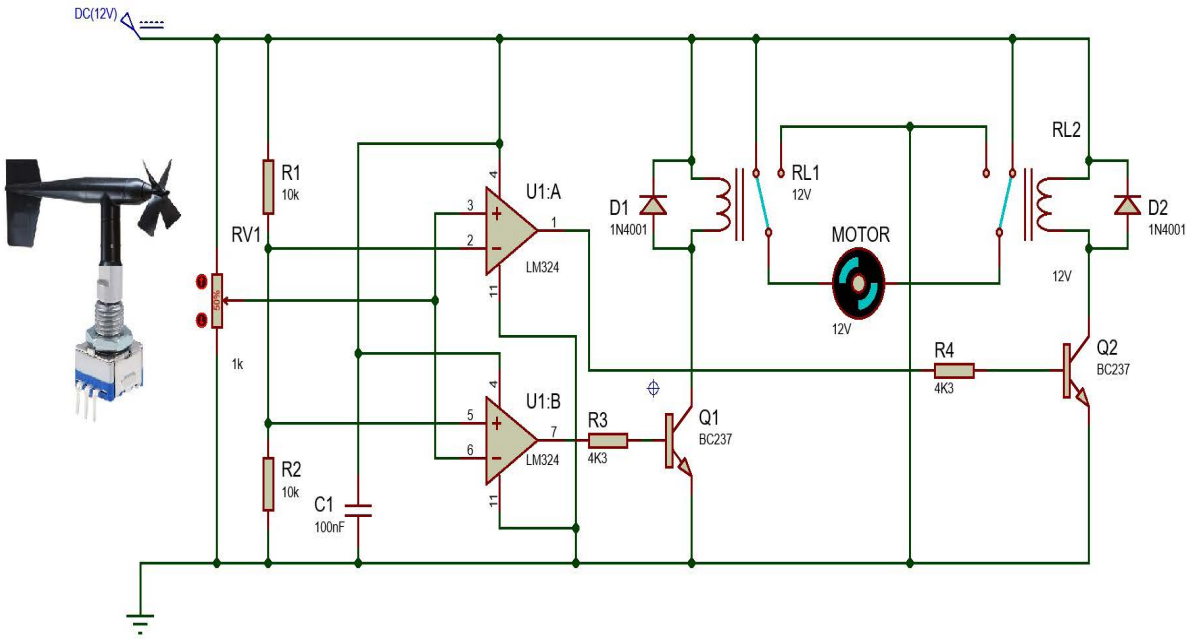
1. Güneş paneli açı değerinin her bölgeye göre değişmesinin sebebini açıklayınız.
2. Uygun açı ile yerleştirilmeyen güneş panellerinin ne tür sorun çıkarabileceğini arkadaşlarınızla tartışınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Elemanları inceledi.	20	
Sınıfı :	2	Elemanlarının seçimini doğru yaptı.	20	
Numarası :	3	Sistemi hesaplamasını doğru yaptı.	20	
ÖĞRETMEN	4	Sistemi doğru kurdu.	20	
Adı Soyadı :	5	Malzemeleri teslim etti.	10	
	6	Çalışmada temizlik ve düzene dikkat etti.	10	
İmza :		**TOPLAM PUAN**	100	

2. UYGULAMA

RÖLELİ DC MOTOR YÖN KONTROL SİSTEMİ

AMAÇ: **Potansiyometre yardımı ile DC motorun yön kontrolünü sağlamak.**



Görsel 8.12: Potansiyometre yardımı ile DC motor yön kontrolü

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
DC güç kaynağı	12 volt-3 amper	1 adet
LM324 entegre	-	1 adet
10K direnç	¼ Watt	2 adet
4K3 direnç	¼ Watt	2 adet
Potansiyometre	1-5K	1 adet
1N4001	-	2 adet
Röle	12 V tek kontak röle (1NC, 1 NO)	2 adet
BC 237 transistör	PNP	2 adet
DC motor	12 volt	1 adet

İŞLEM BASAMAKLARI

1. Görsel 8.12'de verilen devre şemasına göre bağlantıları yapınız.
2. Öğretmeninizin kontrolünde devreye enerji veriniz.
3. LDR'lerden birini elinizle kapatarak motorun hangi yöne hareket ettiğini gözlemleyiniz.
4. Diğer LDR'yi kapatarak motorun hangi yöne hareket ettiğini gözlemleyiniz.
5. Enerjiyi kesiniz, malzemeleri sökerek öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

SORULAR

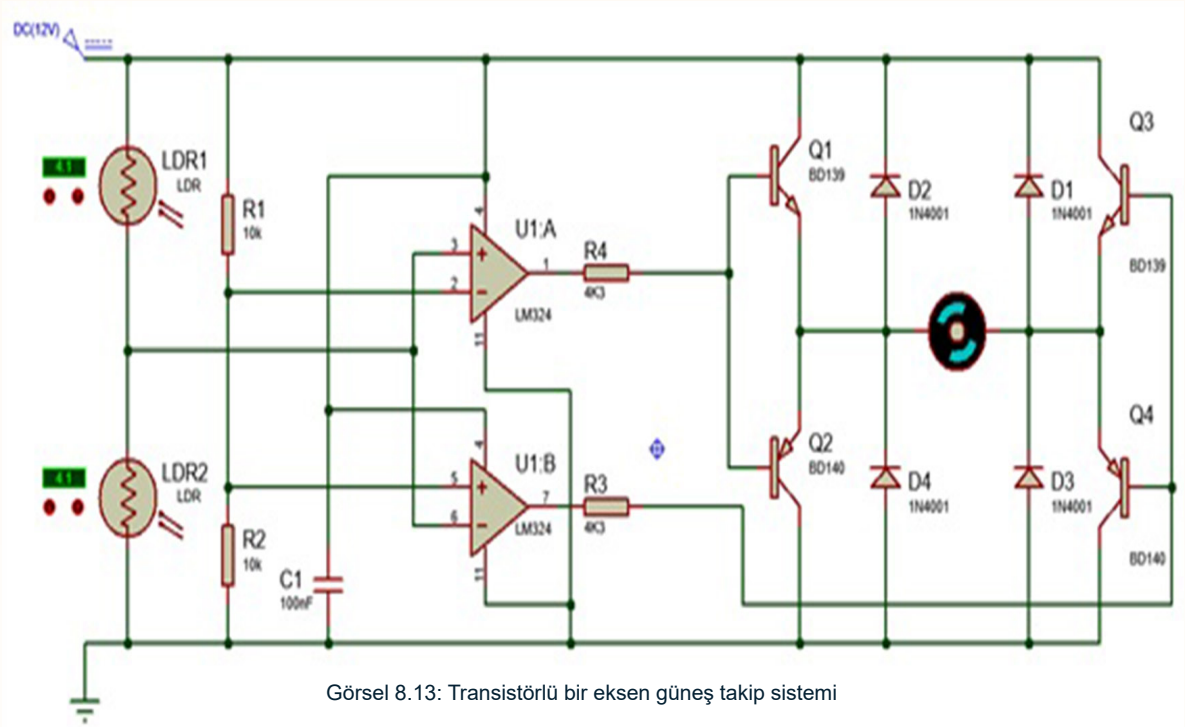
1. Güneş takip sistemine neden ihtiyaç duyulduğunu açıklayınız.
2. Yaptığınız devrede foto direncin (LDR) görevi nedir? Foto direnç yerine farklı bir malzeme ile bu devre kurulabilir mi? Açıklayınız.
3. Yaptığınız devrede kullandığınız 4 adet transistörün DC motorun yön değiştirmesine olan etkisini açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Devre elemanlarını inceledi.	20	
Sınıfı :	2	Devre elemanlarının yerleşimini doğru yaptı.	20	
Numarası :	3	Devre elemanlarını düzgün lehimledi.	20	
ÖĞRETMEN	4	Elektronik devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Malzemeleri teslim etti.	10	
İmza :	6	Çalışmada temizlik ve düzene dikkat etti.	10	
İmza :		**TOPLAM PUAN**	100	

3. UYGULAMA

TRANSİSTÖRLÜ BİR EKSEN GÜNEŞ TAKİP SİSTEMİ

AMAÇ: **Güneş ışınlarının güneş paneline sürekli dik açıda gelmesini sağlayacak sistemi kurmak.**



Görsel 8.13: Transistörlü bir eksen güneş takip sistemi

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
DC güç kaynağı	12 volt-3 amper	1 adet
LM324 entegre	-	1 adet
10K direnç	¼ Watt	2 adet
4K3 direnç	¼ Watt	2 adet
LDR – foto direnç	2-5K	2 adet
1N4001	-	4 adet
BD139	NPN transistör	2 adet
BD140	PNP transistör	2 adet
DC motor	12 volt	1 adet

İŞLEM BASAMAKLARI

1. Görsel 8.13'de verilen devre şemasına göre bağlantıları yapınız.
2. Öğretmeninizin kontrolünde devreye enerji veriniz.
3. LDR'lerden birini elinizle kapatarak motorun hangi yöne hareket ettiğini gözlemleyiniz.
4. Diğer LDR'yi kapatarak motorun hangi yöne hareket ettiğini gözlemleyiniz.
5. Enerjiyi kesiniz, malzemeleri sökerek öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

SORULAR

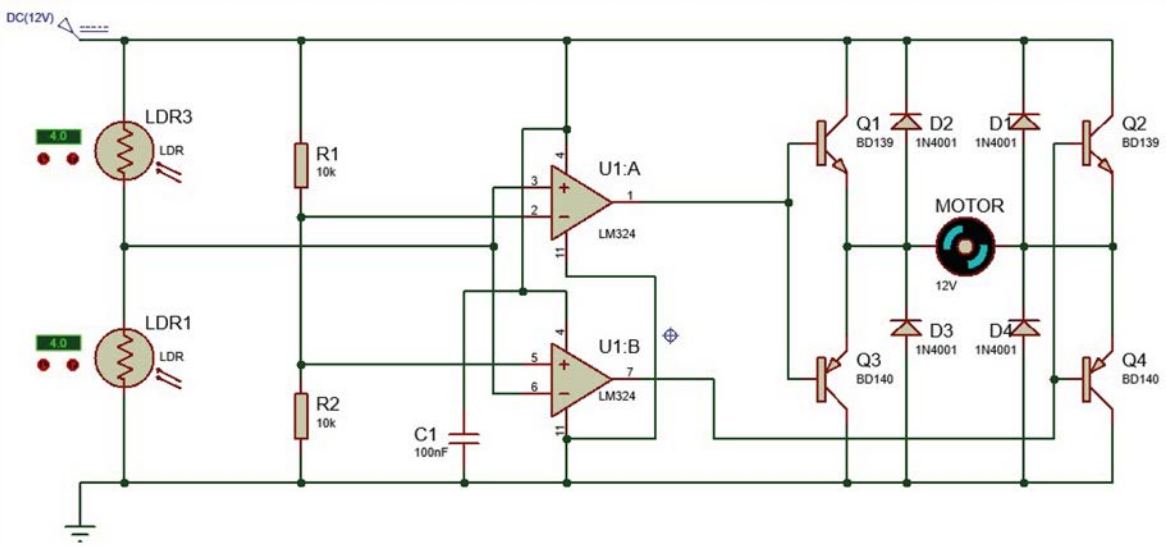
1. Güneş takip sistemine neden ihtiyaç duyulduğunu açıklayınız.
2. Yaptığınız devrede foto direncin (LDR) görevi nedir? Foto direnç yerine farklı bir malzeme ile bu devre kurulabilir mi? Açıklayınız.
3. Yaptığınız devrede kullandığınız 4 adet transistörün DC motorun yön değiştirmesine olan etkisini açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Devre elemanlarını inceledi.	20	
Sınıfı :	2	Devre elemanlarının yerleşimini doğru yaptı.	20	
Numarası :	3	Devre elemanlarını düzgün lehimledi.	20	
ÖĞRETMEN	4	Elektronik devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Malzemeleri teslim etti.	10	
İmza :	6	Çalışmada temizlik ve düzene dikkat etti.	10	
İmza :		**TOPLAM PUAN**	100	

4. UYGULAMA

RÖLELİ BİR EKSEN GÜNEŞ TAKİP SİSTEMİ

AMAÇ: Güneş ışınlarının güneş paneline sürekli dik açıda gelmesini sağlayacak sistemi kurmak.



Görsel 8.14: Röleli bir eksen güneş takip sistemi

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
DC güç kaynağı	12 volt-3 amper	1 adet
LM324 entegre	-	1 adet
10K direnç	¼ Watt	2 adet
4K3 direnç	¼ Watt	2 adet
LDR – foto direnç	2-5K	2 adet
1N4001	-	2 adet
Röle	12 V, tek kontak röle (1NC, 1 NO)	2 adet
BC237	PNP transistör	2 adet
DC motor	12 volt	1 adet

İŞLEM BASAMAKLARI

1. Görsel 8.14'te verilen devre şemasına göre bağlantıları yapınız.
2. Öğretmeninizin kontrolünde devreye enerji veriniz.
3. LDR'lerden birini elinizle kapatarak motorun hangi yöne hareket ettiğini gözlemleyiniz.
4. Diğer LDR'yi kapatarak motorun hangi yöne hareket ettiğini gözlemleyiniz.
5. Enerjiyi kesiniz, malzemeleri sökerek öğretmeninize teslim ediniz ve çalışma alanını temizleyiniz.

SORULAR

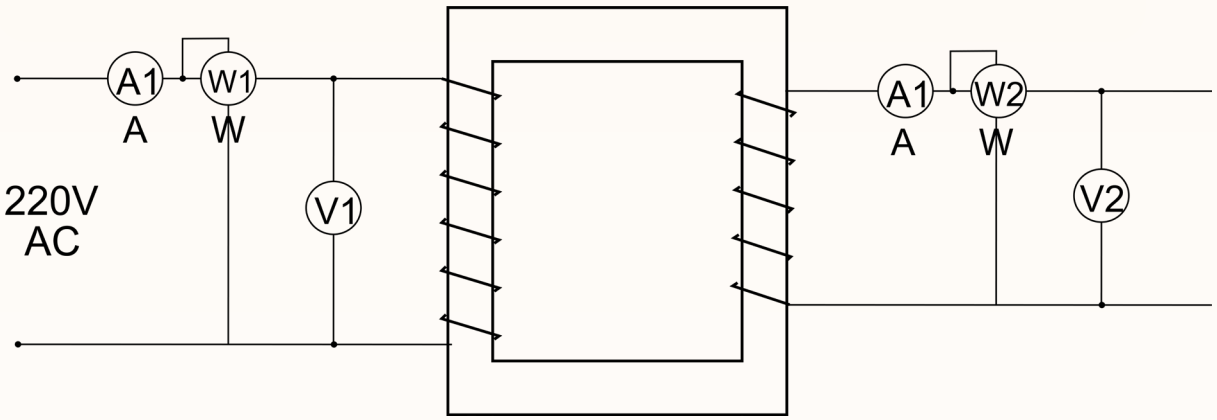
1. Yaptığınız devrede foto direncin (LDR) görevi nedir? Foto direnç yerine farklı bir malzeme ile bu devre kurulabilir mi? Açıklayınız.
2. Yaptığınız devrede kullandığınız 2 adet rölenin DC motorun yön değiştirmesine olan etkisini açıklayınız.
3. Yaptınız devrede kullandığınız diyotların kullanım amacını açıklayınız.

ÖĞRENCİ		DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Devre elemanlarını inceledi.	20	
Sınıfı :	2	Devre elemanlarının yerleşimini doğru yaptı.	20	
Numarası :	3	Devre elemanlarını düzgün lehimledi.	20	
ÖĞRETMEN	4	Elektronik devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Malzemeleri teslim etti.	10	
İmza :	6	Çalışmada temizlik ve düzene dikkat etti.	10	
İmza :		**TOPLAM PUAN**	100	

5. UYGULAMA

TRANSFORMATÖRÜN AŞIRI YÜKLEME UYGULAMASI

AMAÇ: Transformatörün sabit primer geriliminde, sekonder sargısına bağlanan omik yükteki değişime bağlı olarak transformatörün aşırı yüklenmesinin incelenmesi.



Görsel 8.15: Transformatörün aşırı yüklenmesi deneyi bağlantısı

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Transformatör	220/55V 100W	1 Adet
Ampermetre	0-5 Amper	2 Adet
Voltmetre	0-250V	2 Adet
Watmetre	5A-250V	2 Adet
Reosta (Ayarlı direnç)	100Ohm -100Watt	1 Adet

	R 100 OHM	R 75 OHM	R 50 OHM	R 15 OHM
A1				
A2				
V1				
V2				
W1				
W2				

İŞLEM BASAMAKLARI

1. Görsel 8.15'de verilen devreyi kurunuz.
2. Reostayı (Ayarlı direnç) en yüksek direnç değerine alınız ve devreye enerji uygulayınız.
3. Tabloda verilen direnç değerlerinde ölçü aletlerinin gösterdiği değerleri kaydediniz.
4. Watmetrelerin gösterdiği değer, trafo gücünün üzerine çıktığında trafoda meydana gelen ses ve ısı değişimini gözleyiniz.
5. Enerjiyi keserek deneyi sonlandırınız.

SORULAR

1. Primer wattmetresi ile, sekonder wattmetresinin aynı değeri göstermesinin nedenini açıklayınız.
2. Watmetrelerin gösterdiği değer, trafo gücünün üzerine çıktığında trafoda meydana gelen ses ve ısı değişiminin nedenini arkadaşlarınızla tartışınız.
3. Transformatörün aşırı yük ile uzun süre çalıştırılmasının sakıncalarını arkadaşlarınızla tartışınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Devre elemanlarını inceledi.	20	
Sınıfı :	2	Devre elemanlarının yerleşimini doğru yaptı.	20	
Numarası :	3	Devre elemanlarını düzgün lehimledi.	20	
ÖĞRETMEN	4	Elektronik devreyi doğru çalıştırdı.	20	
Adı Soyadı :	5	Malzemeleri teslim etti.	10	
İmza :	6	Çalışmada temizlik ve düzene dikkat etti.	10	
İmza :		**TOPLAM PUAN**	100	

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki cümlelerin başındaki boşluğa cümleler doğru ise (D), yanlış ise (Y) yazınız.

1. (…) Hidroelektrik santralleri suyun potansiyel enerjisini elektrik enerjisine çevirir.
2. (…) Enerji santrallerinin izlenmesine olanak sağlayan sistemler genel olarak "uzaktan izleme sistemleri" olarak isimlendirilmektedir.
3. (…) Enerji iletim hattı başında ve sonunda elektrik enerjisinin gerilimini sadece alçaltma işleminin yapıldığı tesisler şalt sahası olarak isimlendirilir.
4. (…) Güneş ışığı açısı 90o ye yaklaştıkça güneş pilinden alınan güç azalacaktır.
5. (…) Rüzgâr türbinleri rüzgârın kinetik enerjisini mekanik enerjiye dönüştürür.

B) Aşağıdaki soruların doğru seçeneğini işaretleyiniz.

6. Uzaktan izleme sistemlerinin veri toplama birimi aşağıdakilerden hangisidir?
A) Datalogger B) İnvertör C) Ölçüm aletleri
D) Pano E) Sensör

7. Enlem değeri 41° olan bir bölgeye yerleştirilecek güneş panelinin eğim açısı aşağıdakilerden hangisidir?
A) 37,77° B) 38,77° C) 39,77°
D) 40,77° E) 41,77°

8. Rüzgâr türbinlerinde kanatlardan iletilen mekanik enerjiyi elektrik enerjisine çeviren aygıt aşağıdakilerden hangisidir?
A) Alternatör B) Dişli kutusu C) Pitch sistemi
D) Şaft E) Yaw sistemi

9. Rüzgâr hızına bağlı olarak kanatların açısını ayarlayan sistem aşağıdakilerden hangisidir?
A) Dişli kutusu B) Kule C) Pitch sistemi
D) Şaft E) Yaw sistemi

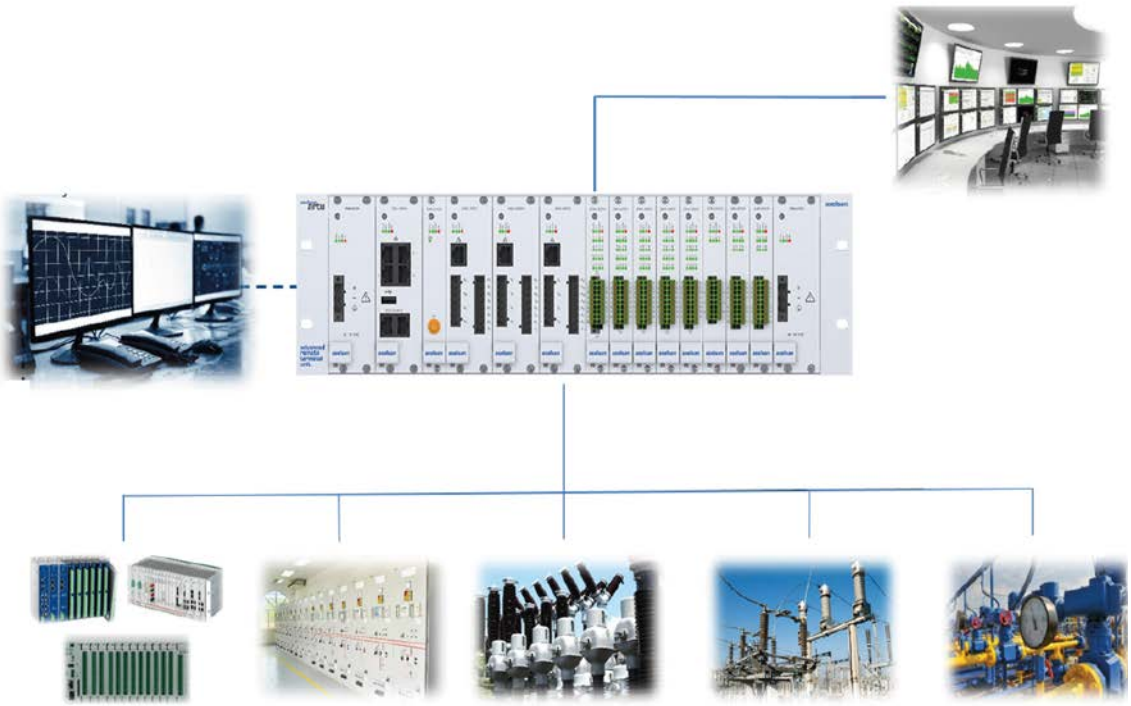
10. Rüzgâr yönüne bağlı olarak rüzgâr türbininin yönünü ayarlayan sistem aşağıdakilerden hangisidir?
A) Dişli kutusu B) Kule C) Pitch sistemi
D) Şaft E) Yaw sistemi

9. ÖĞRENME BİRİMİ

SCADA

KONULAR

9.1. FOTOVOLTAİK (GÜNEŞ) PANELİN HAREKET KONTROLÜ
9.2. TÜRBİN HAREKET KONTROLÜ
9.3. ENERJİ SANTRALİNİN GÜVENLİK KONTROLÜ
9.4. ENERJİ SANTRALİNİN ŞALT SAHASI MANEVRA KONTROLÜ



TEMEL KAVRAMLAR

Fotovoltaik panel, MODBUS, MODBUS TCP, PLC, RTU, rüzgâr türbini, SCADA, şalt sahası

HAZIRLIK ÇALIŞMALARI

1. SCADA, PLC, RTU ile ilgili bildiklerinizi arkadaşlarınızla paylaşınız.
2. SCADA ile yapılan kontrol sistemleriyle ilgili düşüncelerinizi ifade ediniz.

9.1. FOTOVOLTAİK (GÜNEŞ) PANELİN HAREKET KONTROLÜ

Ayçiçekleri, güneşten en iyi şekilde faydalanabilmek için güneşi takip eder. Güneş panellerinin de güneşi bu şekilde takip etmesi hâlinde elektrik enerjisi üretim verimleri %30-40 dolayında artar. Hareket esnasında güneş ışınlarının güneş paneline 90° açıyla gelmesi hedeflenir. Güneş panelinin hareketi doğu-batı, kuzey-güney yönleri olarak iki eksenlidir (Görsel 9.1).

Güneş panellerinin hareketinin kontrolünde ısı ve ışık algılayıcıları kullanılabileceği gibi algılayıcı kullanılmadan coğrafi konum bilgisine göre de hareket sağlanabilir. Güneş panellerinin kW başına maliyetinin hareket kontrol sistemlerine göre daha az olmasından dolayı hareket kontrollü güneş panellerinin kullanımı azalmıştır.



Görsel 9.1: Hareket kontrollü güneş panelleri

SCADA, ingilizce "Supervisory Control And Data Acquisition" (Supırvizory kontrol end deda aküzişın) kelimelerinin kısaltmasıdır. Merkezî denetim ve veri toplama anlamına gelir. SCADA, bilgisayarlarda çalışan bir yazılım olup temelde veri izleme, veri kontrol, veri toplama ve veri kaydetme işlemlerini gerçekleştirir.

Sahada kontrol işlemleri için PLC, RTU [Remote Terminal Unit (Rimot terminıl yunit / Uzak Terminal Birimi) gibi kontrol devre elemanları kullanılır. SCADA yazılımı bu elemanlara bir ağ üzerinden bağlanır. SCADA, otomasyon katmanında PLC ve RTU'ların üzerinde yer alır. Kontrol devre elemanlarındaki değerler SCADA bilgisayarı ekranından görülebilir, değiştirilebilir ve kayıt altına alınabilir. Toplanan ve kaydedilen verilere göre grafikler, raporlar, alarm bilgileri oluşturulabilir.

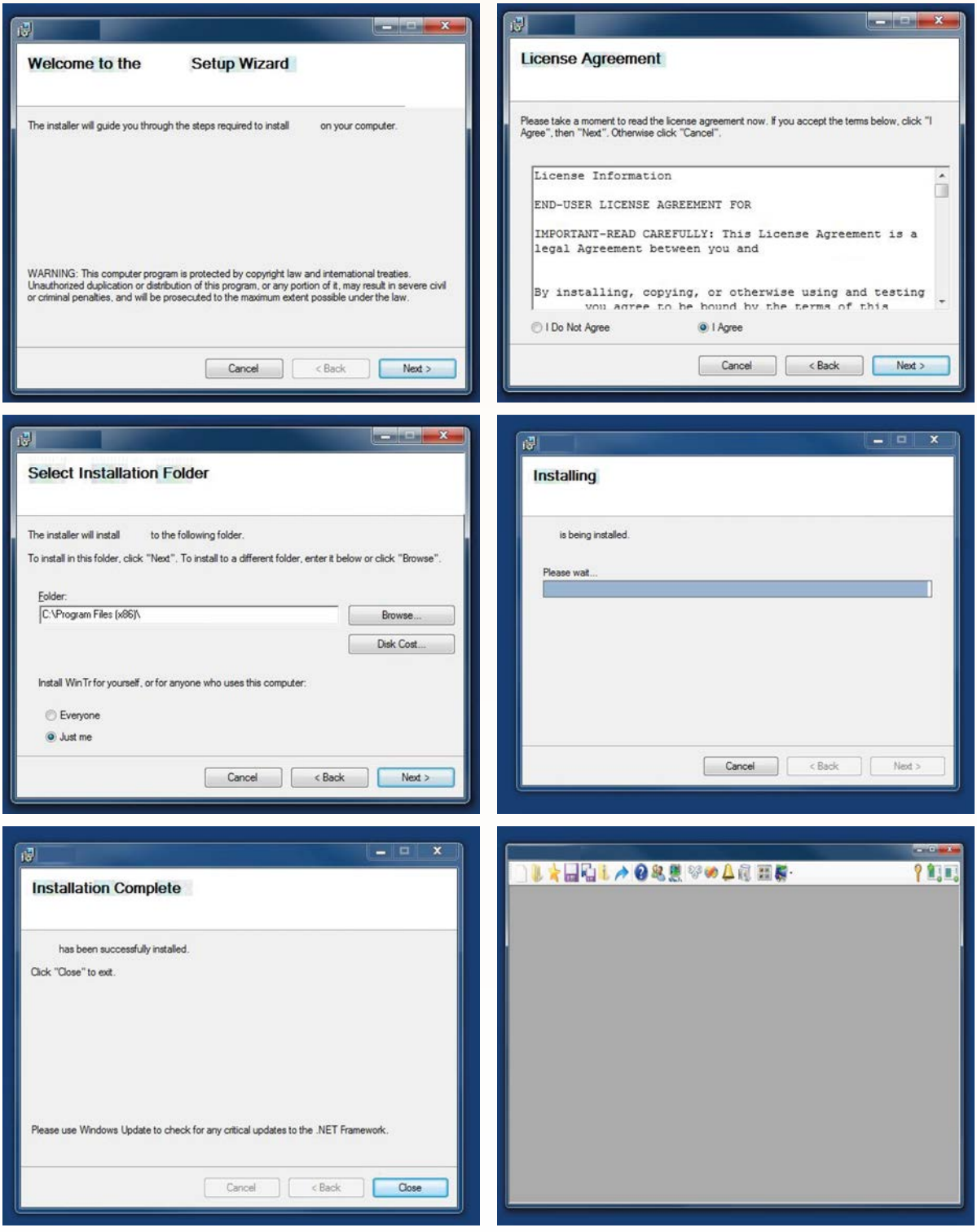
SCADA; elektrik enerjisi üretimi ve iletimi, su üretimi ve iletimi, petrol ve gaz hatları, raylı sistemler trafik denetimi, ısıtma, havalandırma, bina yönetimi gibi süreçlerin denetiminde yaygın olarak kullanılır. SCADA programları, çalışacağı bilgisayarın işletim sistemi, donanımı, haberleşme yöntemleri gibi özellikleri göz önünde bulundurularak seçilir.

1. UYGULAMA

SCADA PROGRAMININ KURULUMU

AMAÇ: SCADA programının kurulumunu yapmak.

YÖNERGE: SCADA programları kurulurken çalışacağı bilgisayar donanımı, işletim sistemi, veri tabanı gibi ek yazılımlar dikkate alınır.



Görsel 9.2: SCADA programının kurulumu

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Bilgisayar	SCADA yazılımı sistem gereksinimlerine uygun	1 adet
SCADA yazılımı	Gereksinimleri karşılayacak özelliklerde	1 adet
Ek yazılımlar	İşletim sistemi, veri tabanı yazılımı	1 adet
İnternet bağlantısı	Uygun hızda	-

İŞLEM BASAMAKLARI

1. Sisteminize uygun SCADA yazılım ve veri tabanı kurulum dosyalarını üreticinin internet sayfasından bilgisayarınıza indiriniz.
2. SCADA yazılımı kurulum dosyasını çalıştırınız (Görsel 9.2).
3. Kurulum esnasında istenen seçimleri yaparak yazılımı bilgisayarınıza kurunuz (Görsel 9.2).
4. SCADA yazılımının kurulumundan sonra veri tabanı yazılım kurulumu otomatik başlamıyorsa veri tabanı yazılımı kurulum dosyasını çalıştırınız (Görsel 9.2).
5. SCADA yazılımının ve veri tabanı programının çalışmasını kontrol ediniz (Görsel 9.2).

SORULAR

1. Kurmuş olduğunuz SCADA yazılımı hangi haberleşme protokollerini desteklemektedir? Belirtiniz.
2. SCADA yazılımları, veri tabanı programına neden ihtiyaç duyar? Açıklayınız.

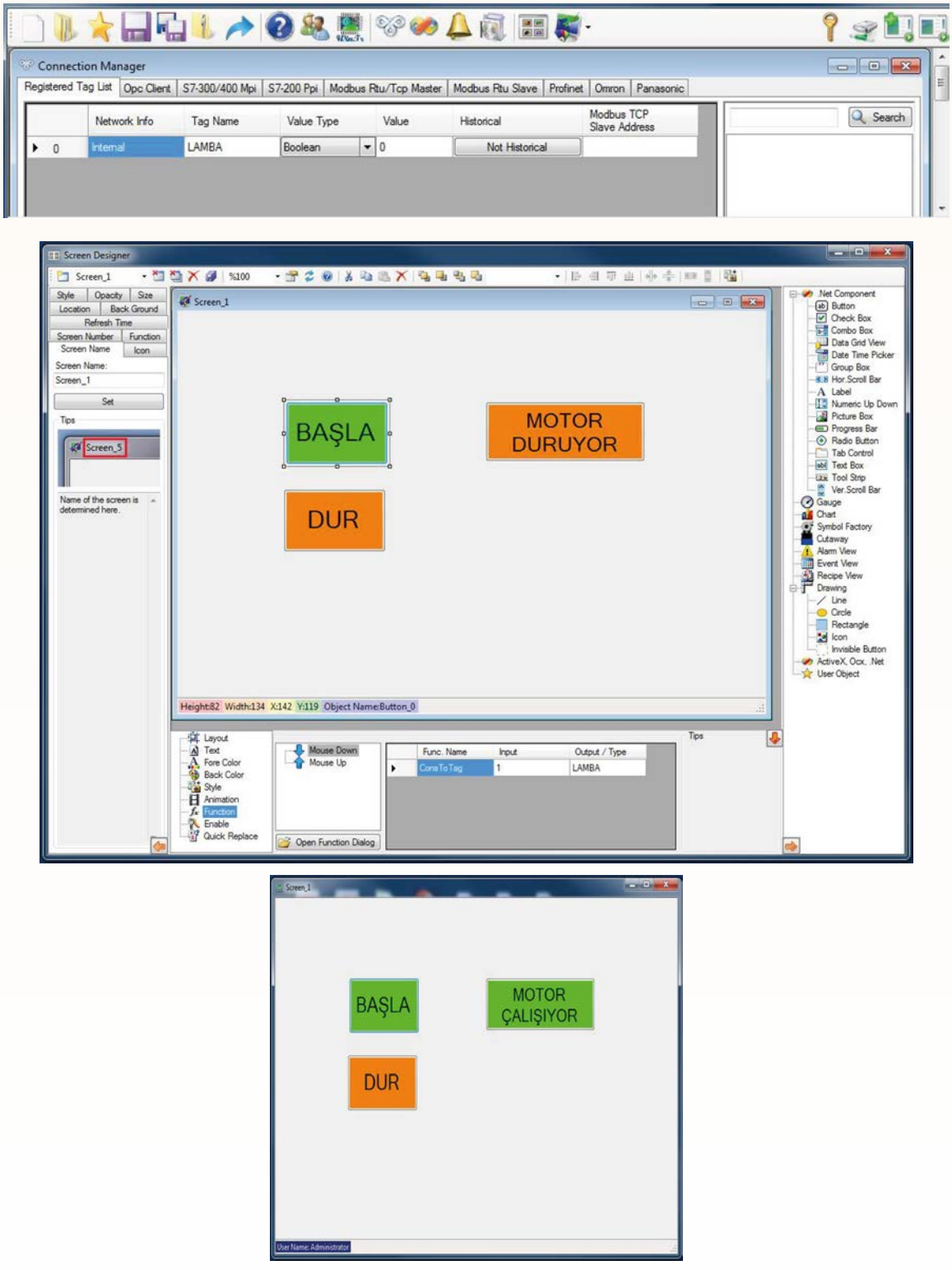
ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Uygun SCADA sürümünü seçti.	20	
Sınıfı :	2	İşletim sistemini güncelledi.	20	
Numarası :	3	SCADA programını kurdu.	20	
ÖĞRETMEN	4	Veri tabanı programını kurdu.	20	
Adı Soyadı :	5	Soruları cevaplandırdı.	10	
	6	Verilen sürede işi tamamladı.	10	
İmza :		**TOPLAM PUAN**	100	

2. UYGULAMA

SCADA BUTON VE SİNYAL LAMBASI BİLEŞENLERİNİN KULLANIMI

AMAÇ: **SCADA bileşenlerini kullanabilmek.**

YÖNERGE: **SCADA sayfalarında kullanılabilecek farklı görsel ve işlevsel bileşenler bulunmaktadır. SCADA programında değişkenler etiket (tag) ismi verilerek kaydedilir.**



Görsel 9.3: SCADA yazılımında buton ve sinyal lambası oluşturma

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Bilgisayar	SCADA yazılımı sistem gereksinimlerine uygun	1 adet
SCADA yazılımı	Gereksinimleri karşılayacak özelliklerde	1 adet
Ek yazılımlar	İşletim sistemi, veri tabanı yazılımı	1 adet

İŞLEM BASAMAKLARI

1. SCADA programını çalıştırınız ve projenize yeni bir isim vererek dosyaları kaydediniz.
2. Bağlantı yöneticisi (Connection Manager / Konnekşın menecır) penceresindeki Etiket (Tag) listesinden Lamba isimli 0,1 BOOLEAN (boolien) tipinde dâhilî bir etiket oluşturunuz.
3. Ekran Tasarımcısı (Screen Designer / sikriin dizaynır) penceresinden yeni pencere oluşturunuz.
4. Düğme (Button / Batin) bileşeninden oluşturduğunuz pencere üzerinde üç adet düğme oluşturunuz (Görsel 9.3).
5. BAŞLA düğmesinin boyutunu, rengini ve fare tıklamasında LAMBA değişkenini 1 yapacak şekilde düzenleyiniz (Görsel 9.3).
6. DUR düğmesinin boyutunu, rengini ve fare tıklamasında LAMBA değişkenini 0 yapacak şekilde düzenleyiniz.
7. MOTOR ÇALIŞIYOR düğmesinin boyutunu, rengini ve LAMBA değişkenine göre görünümünü düzenleyiniz (Görsel 9.3).
8. Dosyanızı kaydederek Geliştirici Çalıştırma (Development Runtime / Divalapmınt rantiym) modunda çalıştırınız (Görsel 9.3).
9. Başla düğmesine tıkladığınızda motor çalışıyor, Dur düğmesine tıkladığınızda motor duruyor şeklinde lamba değişimini gözlemleyiniz.

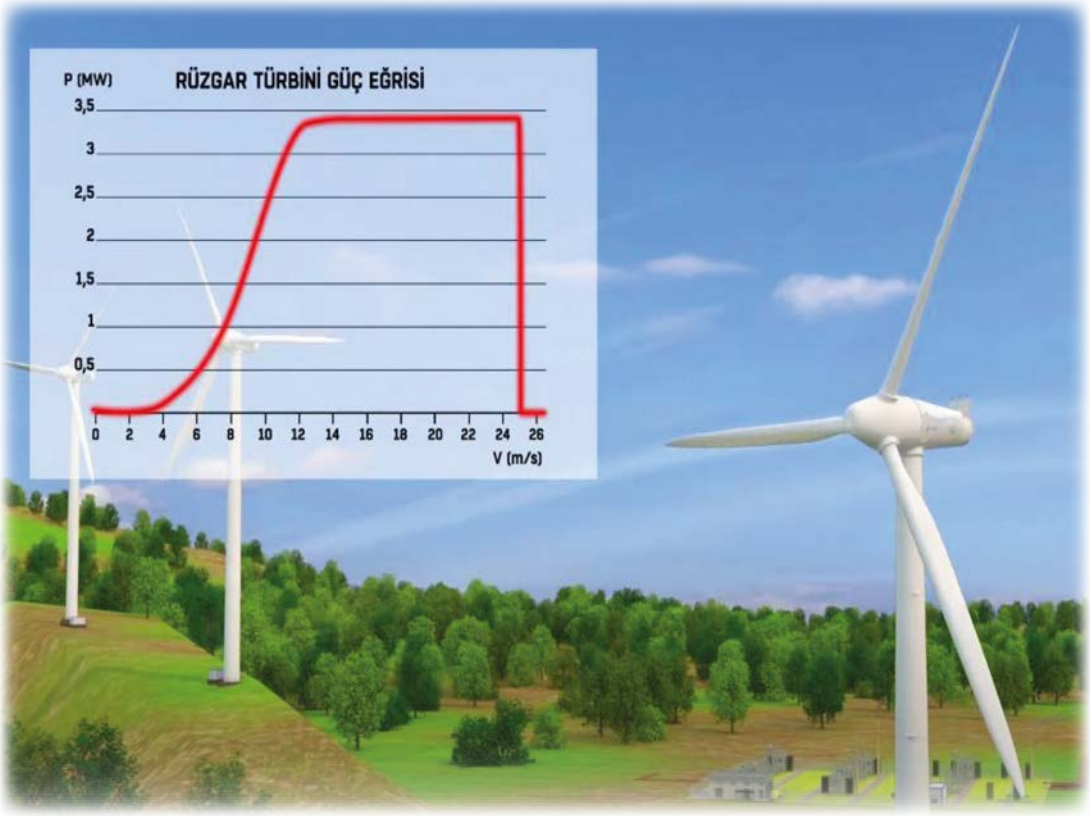
SORULAR

1. SCADA yazılımınızda Etiketlerin alabileceği değişken tipleri nelerdir? Belirtiniz.
2. SCADA yazılımınızda ne gibi ekran bileşenleri bulunur? Açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Etiketi tanımladı.	20	
Sınıfı :	2	Sayfayı oluşturdu.	20	
Numarası :	3	Sayfadaki düğmeleri oluşturdu.	20	
ÖĞRETMEN	4	Projeyi çalıştırdı.	20	
Adı Soyadı :	5	Soruları cevaplandırdı.	10	
	6	Verilen sürede işi tamamladı.	10	
İmza :		**TOPLAM PUAN**	100	

9.2. TÜRBİN HAREKET KONTROLÜ

Rüzgâr türbini, rüzgârdan gelen kinetik enerjiyi mekanik enerjiye dönüştüren döner bir makinedir (Görsel 9.4). Bu mekanik enerji daha sonra elektriğe dönüştürülür. Bu enerji dönüşümlerinden sorumlu türbin bileşenleri rotor ve jeneratördür.



Görsel 9.4: Rüzgâr türbini

Rüzgâr türbininin güç çıkışını optimize etmek veya sınırlamak için farklı kontrol yöntemleri kullanılır. Jeneratör hızı, kanat açısı ayarı ve türbininin gövde açı dönüşü kontrol edilerek türbinin kontrolü sağlanır (Görsel 9.5). Kanat ve gövde açısının değerleri, türbinin kontrol işlemcisinde matematik modelin hesaplanması ile bulunur.



Görsel 9.5: Rüzgâr türbininin kanat ve gövde açı kontrolü

3. UYGULAMA

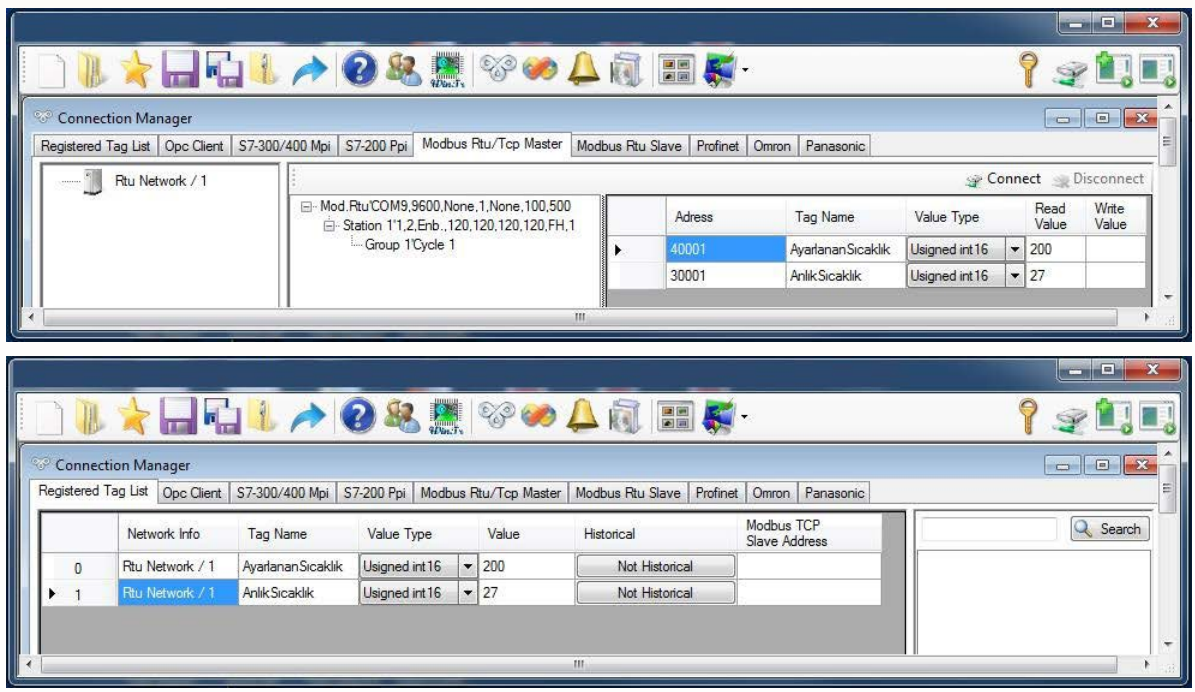
SCADA MODBUS RTU HABERLEŞMESİ

AMAÇ: SCADA ile Modbus haberleşmesi yapmak.

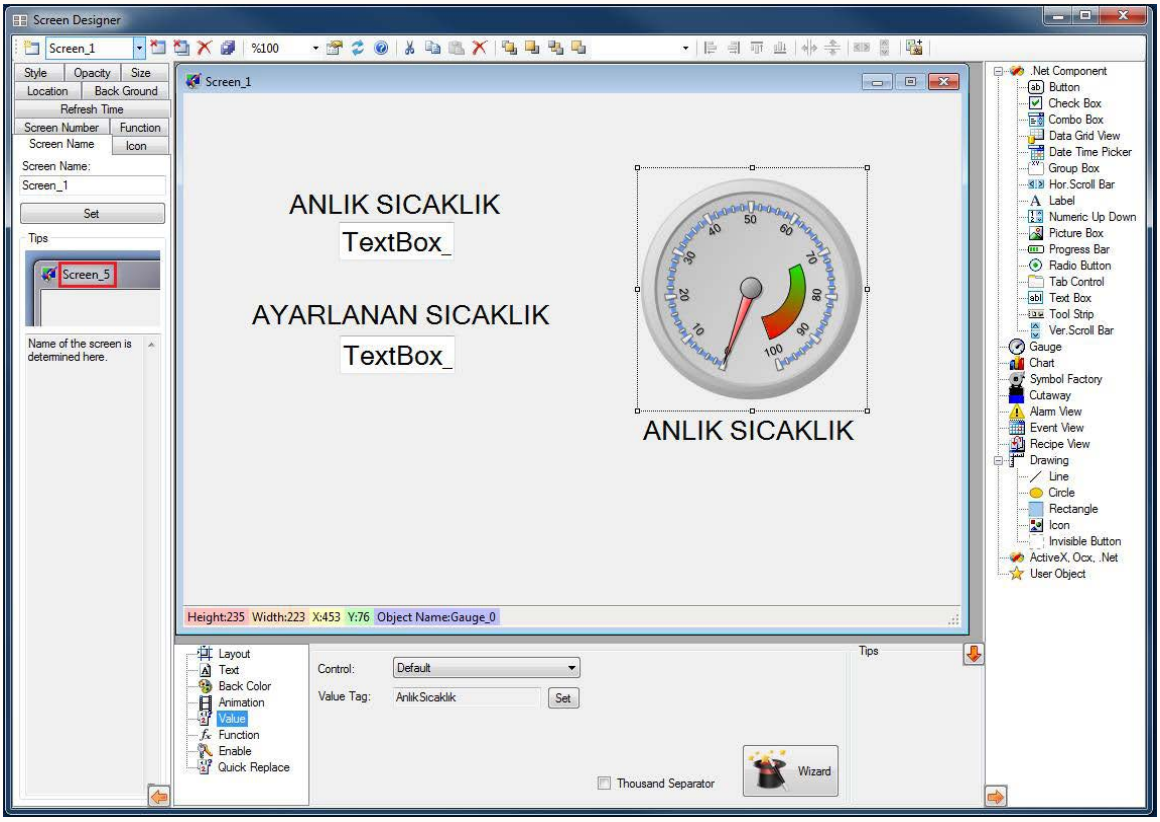
YÖNERGE: Modbus, endüstriyel cihazlar arasında haberleşmek için kullanılan haberleşme protokolüdür. İlk geliştirilen protokollerden olması, basit, ücretsiz ve açık olmasından dolayı yaygın olarak kullanılır.



Görsel 9.6: Devre bağlantısı



Görsel 9.7: SCADA haberleşme ekranı



Görsel 9.8: SCADA tasarım ekranı



Görsel 9.9: Geliştirici modunda çalıştırma

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Bilgisayar	SCADA yazılımı sistem gereksinimlerine uygun	1 adet
SCADA yazılımı	Gereksinimleri karşılayacak özelliklerde	1 adet
Ek yazılımlar	İşletim sistemi, veri tabanı yazılımı	1 adet
Sıcaklık kontrol rölesi	Dijital, Modbus RTU heberleşme özellikli	1 adet
Termokupl	J tipi	1 adet
RS 485-USB dönüştürücü	İzoleli veya izolesiz	1 adet
El aletleri ve kablo	Pense, yan keski, tornavida, 0,75 mm² kablo	-

İŞLEM BASAMAKLARI

1. Görsel 9.6'daki gibi bağlantıyı kurunuz ve RS 485-USB seri dönüştürücüyü bilgisayarınızın boş USB portuna takınız (Dönüştürücünün işletim sistemi tarafından tanınıp tanınmadığını kontrol ediniz. Dönüştürücü işletim sistemi tarafından tanınmazsa üreticinin sağladığı sürücü dosyalarını bularak işletim sistemine yükleyiniz.).
2. Bilgisayarınızın aygıt yöneticisinden RS 485-USB dönüştürücünün hangi COM port adresini aldığını tespit ediniz.
3. SCADA programını çalıştırınız ve projenize yeni bir isim vererek dosyaları kaydediniz.
4. Görsel 9.7'deki gibi Bağlantı yöneticisi (Connection Manager) penceresinden Modbus RTU bağlantısı oluşturunuz (COM port adresi dönüştürücünün adresi ile aynı olmalıdır. Bağlantı hızı ve özelikleri sıcaklık kontrol cihazındaki değerler ile aynı olmalıdır.).
5. Oluşturulan bağlantıda Ayarlanan Sıcaklık isimli değişkeni oluşturunuz (Görsel 9.7). [Değişkenin özellikleri, okuma ve yazma yapılabilen 40001 adresinde, 16 bitlik işaretsiz tam sayı (Usigned int 16) olmalıdır.]
6. Oluşturulan bağlantıda Anlık Sıcaklık isimli değişkeni oluşturunuz (Görsel 9.7). [Değişkenin özellikleri, sadece okunabilen 30001 adresinde, 16 bitlik işaretsiz tam sayı (Usigned int 16) olmalıdır. Modbus adresleri sıcaklık kontrol rölesi markasına göre farklılık gösterebilir.]
7. Kayıtlı etiket listesinde değişkenlerinizi görünüz (Görsel 9.7).
8. Ekran Tasarımcısı ekranında yeni sayfa oluşturunuz.
9. Sayfa üzerinde Etiket (Label / Lebıl) olarak Anlık Sıcaklık ve Ayarlanan Sıcaklık etiketleri oluşturunuz (Görsel 9.8).
10. Sayfa üzerinde iki adet Yazı Kutusu (Text Box / Tekst baks) oluşturunuz. Yazı kutularının Değer (Value / Valiu) kısımlarına, bağlantı yöneticisinde oluşturmuş olduğunuz değişkenleri seçiniz (Görsel 9.8).
11. Sayfa üzerinde Ölçme Aleti (Gauge / Geeç) oluşturunuz. Değer (Value) kısmına bağlantı yöneticisinde oluşturmuş olduğunuz Anlık Sıcaklık değişkenini seçiniz (Görsel 9.8).
12. Dosyanızı kaydederek Geliştirici Çalıştırma (Development Runtime) modunda çalıştırınız ve hataları ayıklayınız (Görsel 9.9).

SORULAR

1. Ölçme Aletini (Gauge) farklı şekillerde görselleştirebilir misiniz? Açıklayınız.
2. SCADA yazılımınızın desteklemiş olduğu haberleşme protokolleri hangileridir? Belirtiniz.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Bağlantıyı oluşturdu.	20	
Sınıfı :	2	Değişkenleri tanımladı.	20	
Numarası :	3	Sayfayı oluşturdu.	20	
ÖĞRETMEN	4	Uygulamayı çalıştırdı.	20	
Adı Soyadı :	5	Soruları cevaplandırdı.	10	
	6	Verilen sürede işi tamamladı.	10	
İmza :		**TOPLAM PUAN**	100	

9.3. ENERJİ SANTRALİNİN GÜVENLİK KONTROLÜ

6331 sayılı İş Sağlığı ve Güvenliği Kanununa göre enerji santralleri çok tehlikeli işyeri sınıfı kapsamındadır. Bu sebeple enerji santrallerinde güvenli bir çalışma ortamı oluşturmak ve sürdürülebilirliği sağlamak gerekir. Bunun için dâhilî ve haricî kaza kaynakları belirlenmeli, gerekli önlemler alınmalı ve bu süreç etkili bir şekilde yönetilmelidir. Bu amaçla iş güvenliği açısından yapılan risk analizleri sonucunda, riskleri yok etmek veya kabul edilebilir seviyeye indirmek için uygun kontrol tedbirleri alınmalıdır. Aksi hâlde istenmeyen durumların ortaya çıkması kaçınılmazdır (Görsel 9.10).



Görsel 9.10: Rüzgâr türbini yangını

Uluslararası Elektroteknik Komisyonu tarafından yayınlanan IEC 61508 standardı, güvenlikle ilgili sistemlerin nasıl uygulanacağını, tasarlanacağını ve sürdürüleceğini belirten uluslararası bir standarttır.

İş güvenliği açısından takip edilmesi gereken sıcaklık, basınç, titreşim gibi değişkenler SCADA da tarihsel etiket (Historical tag / historigıl teg) olarak seçilir. SCADA, veri tabanı uygulamasında oluşturduğu sayfaya istenilen verileri kaydeder. Kaydedilen bu veriler SCADA uygulamasında grafik ve rapor olarak gözlemlenebilir, dış ortama dosya olarak aktarılabilir.

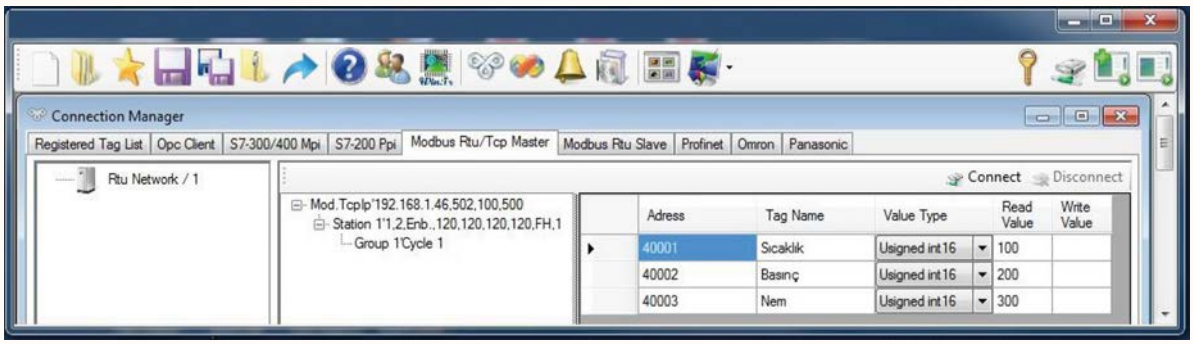
SCADA yazılımlarında ayrıca Alarm sayfası seçeneği vardır. Alarm sayfalarına Alarm olarak seçilen değişkenlerin alt ve üst sınır değerleri tanımlanır. Bu sayfada takip edilen değişkenler tanımlanan alt veya üst sınırlara ulaştığında Alarm olarak veri tabanı sayfasında kayıt altına alınır. Sınır değerler aşıldığında kullanıcıya anlık alarm bilgisi gösterilir. İstenirse kısa mesaj veya e-posta gönderilebilir. Kullanıcı, verilen alarmın gereklerini kendisi yapabileceği gibi yazılıma talimat da verilebilir. Örneğin duman alarmı geldiğinde "yangın vanalarını aç" gibi kontrol işlemleri yaptırılabilir.

4. UYGULAMA

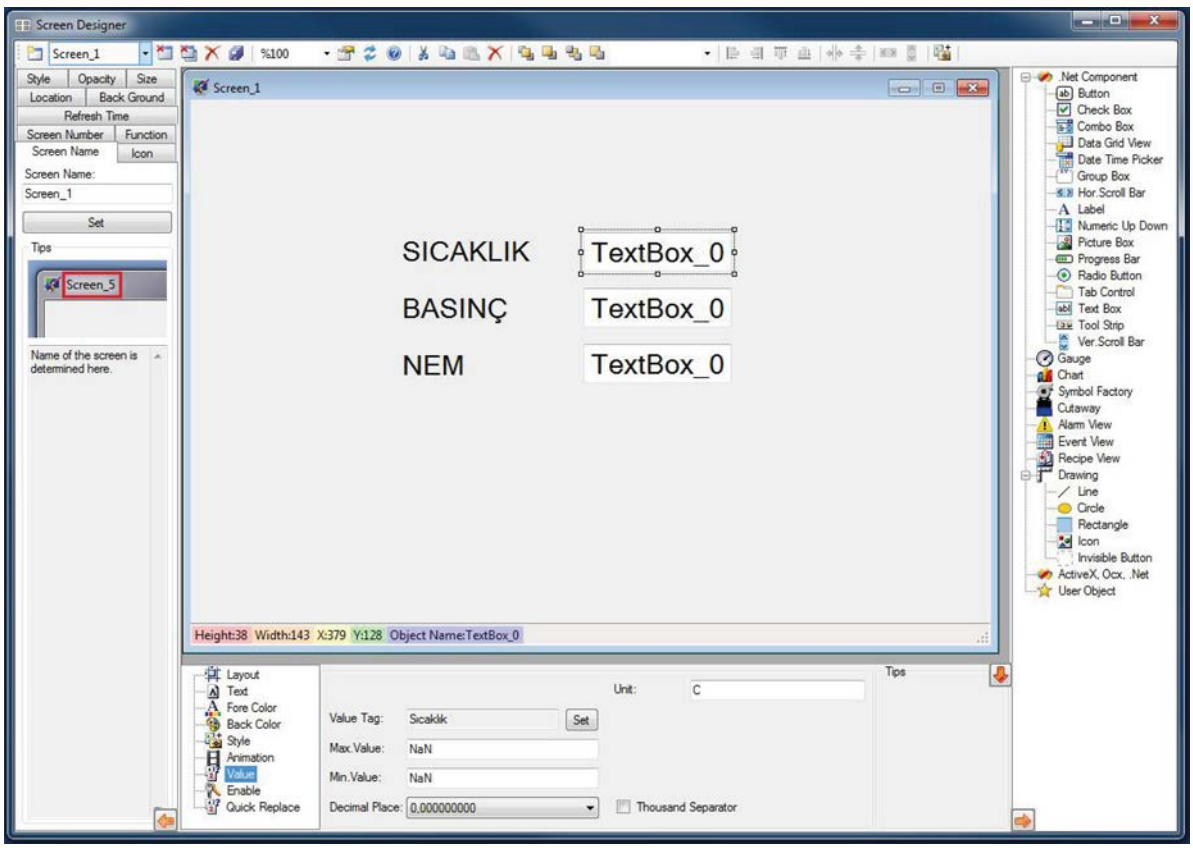
SCADA MODBUS TCP/IP HABERLEŞMESİ

AMAÇ: SCADA ile Modbus TCP/IP haberleşmesi yapmak..

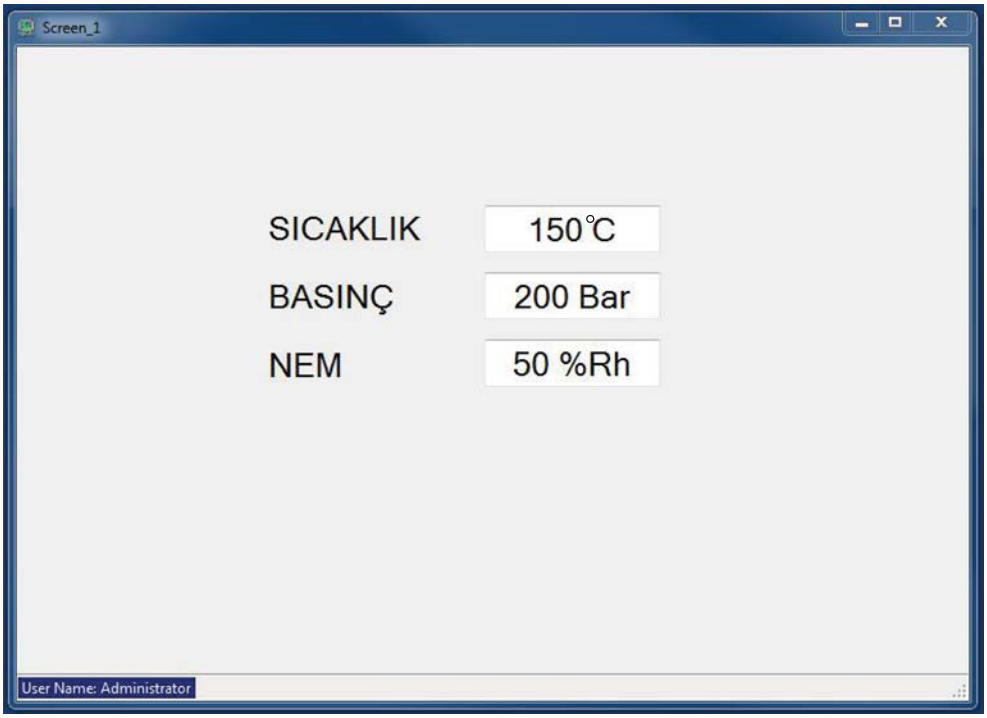
YÖNERGE: Modbus RTU ve Modbus TCP/IP, Modbus protokolünün en çok kullanılan sürümleridir. Modbus RTU, RS 232 ve RS 485 ağında çalışır. Modbus TCP/IP ise ethernet ağında çalışır.



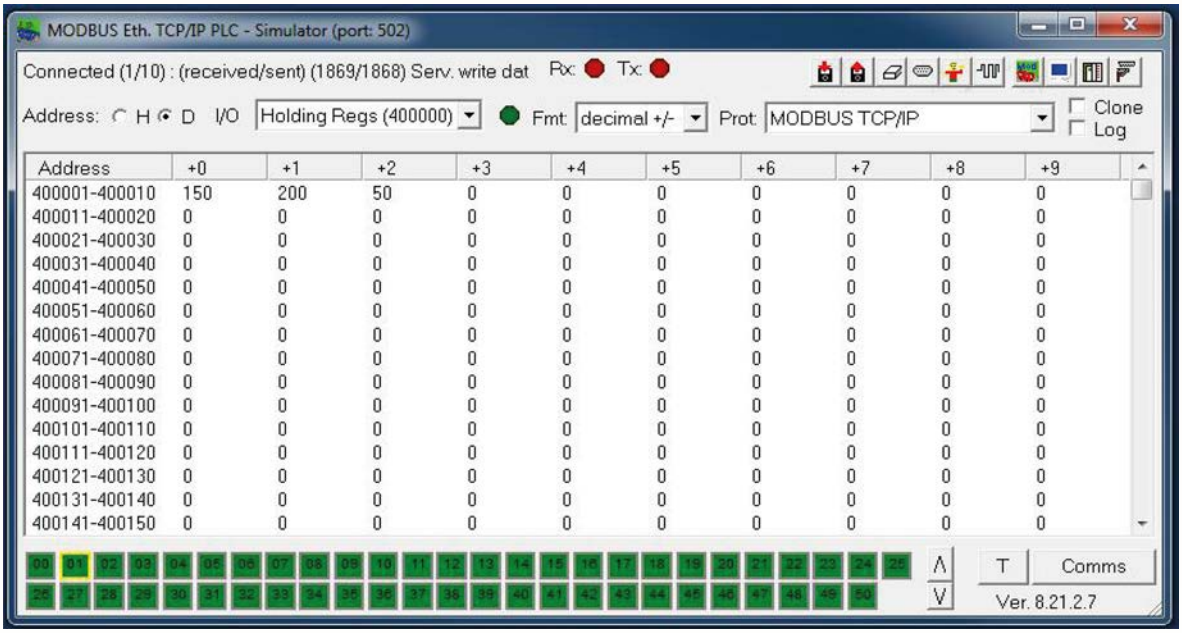
Görsel 9.11: Modbus TCP/IP bağlantısı



Görsel 9.12: Sıcaklık, basınç ve nem etiketlerinin oluşturulması



Görsel 9.13: Geliştirici modunda çalıştırma



Görsel 9.14: Modbus Eth. TCP/IP PLC Simülatör programının çalıştırılması

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Bilgisayar	SCADA yazılımı sistem gereksinimlerine uygun	1 adet
SCADA yazılımı	Gereksinimleri karşılayacak özelliklerde	1 adet
Ek yazılımlar	İşletim sistemi, veri tabanı, modbus simülatör yazılımı	-

İŞLEM BASAMAKLARI

1. SCADA programını çalıştırıp, projenize yeni bir isim vererek dosyaları kaydediniz.
2. Görsel 9.11'deki gibi Bağlantı Yöneticisi penceresinden Modbus TCP/IP bağlantısı oluşturunuz.
3. Modbus TCP/IP kısmına bilgisayarınızın IP'sini öğrenerek yazınız (IP için bilgisayar komut satırına CMD yazıp açılan pencerede ipconfig komutunu veriniz. Modbus TCP/IP portu 502'dir.).
4. İlgili bağlantıda Ayarlanan Sıcaklık isimli değişkeni oluşturunuz (Görsel 9.11). [Değişkenin özellikleri, okuma ve yazma yazılabilen 40001 adresinde 16 bitlik işaretsiz tam sayı (Usigned int 16) olmalıdır.]
5. Aynı şekilde ilgili bağlantıda Ayarlanan Basınç isimli değişkeni oluşturunuz (Görsel 9.11). [Değişkenin özellikleri, okuma ve yazma yazılabilen 40002 adresinde 16 bitlik işaretsiz tam sayı (Usigned int 16) olmalıdır.]
6. Son olarak ilgili bağlantıda Ayarlanan Nem isimli değişkeni oluşturunuz (Görsel 9.11). [Değişkenin özellikleri, okuma ve yazma yazılabilen 40003 adresinde 16 bitlik işaretsiz tam sayı (Usigned int 16) olmalıdır.]
7. Kayıtlı etiket listesinde değişkenlerinizi görünüz.
8. Ekran Tasarımcısı ekranında yeni sayfa oluşturunuz.
9. Sayfa üzerinde Sıcaklık, Basınç ve Nem etiketlerini oluşturunuz (Görsel 9.12).
10. Sayfa üzerinde üç adet Yazı Kutusu (Text Box) oluşturunuz (Görsel 9.12).
11. Yazı kutularının Değer (Value) kısımlarına, bağlantı yöneticisinde oluşturmuş olduğunuz değişkenleri seçiniz (Görsel 9.12).
12. Dosyanızı kaydederek Geliştirici Çalıştırma (Development Runtime) modunda çalıştırınız (Görsel 9.13).
13. Modbus Eth. TCP/IP PLC Simülatör programını çalıştırıp 40001, 40002, 40003 adreslerine değerler girerek SCADA sayfasındaki değişimleri gözlemleyiniz (Görsel 9.14).

SORULAR

1. Modbus RTU ve Modbus TCP/IP haberleşme arasında hız olarak nasıl bir fark vardır? Açıklayınız.
2. Bu uygulamada Modbus Eth. TCP/IP PLC Simülatör programı yerine ne kullanılabilir? Belirtiniz.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Bağlantıyı oluşturdu.	20	
Sınıfı :	2	Değişkenleri tanımladı.	20	
Numarası :	3	Sayfayı oluşturdu.	20	
ÖĞRETMEN	4	Uygulamayı çalıştırdı.	20	
Adı Soyadı :	5	Soruları cevaplandırdı.	10	
	6	Verilen sürede işi tamamladı.	10	
İmza :		**TOPLAM PUAN**	100	

9.4. ENERJİ SANTRALİNİN ŞALT SAHASI MANEVRA KONTROLÜ

Şalt sahası; transformatör, bara, kesici, ayırıcı, ölçü transformatörleri, ölçme kontrol röleleri gibi elemanların bir araya getirilmesi ile oluşturulan ayrılmış alandır. Şalt sahalarında gerilim seviyesi değişimi, elektrik enerjisinin akış kontrolü, koruma, ölçme gibi işlemler yapılır.

İşletme koşullarında elektrik üretim, iletim ve dağıtım şirketleri sorumlu oldukları enerji altyapısını sürekli izleyerek sisteme verdikleri veya aldıkları enerjinin sürekliliğini ve kalitesini kontrol etmek zorundadır. SCADA sistemi şalt sahalarına yalınlık, esneklik, haberleşme gibi imkânlar sağlar. Şalt sahası SCADA sisteminde kullanıcı, SCADA sayfasında sahayı izler ve işlemleri yapar (Görsel 9.15).



Görsel 9. 15: SCADA ile şalt sahasının izlenmesi

Enerji üretim ve iletiminde süreklilik açısından SCADA bilgisayarı çift paralel (Redundant / ridandıt) olarak yapılandırılır. Bu yapılandırmada SCADA bilgisayarının birisi arıza yaparsa diğeri ile kesintisiz çalışma sürdürülür. SCADA bilgisayarı sahadaki PLC, RTU, enerji analizörü, koruma röleleri gibi cihazlar ile haberleşirken internet ağı üzerinden de diğer SCADA bilgisayarları ile haberleşebilir. Bu haberleşme yetenekleri SCADA ile çok büyük ağlar oluşturmaya olanak sağlar.

İlk zamanlarda haberleşme protokollerinde Modbus TCP/IP gibi haberleşme protokolleri kullanılmış olsa da zamanla bu ve bunun gibi protokoller santraller arası haberleşmede yetersiz kalmıştır. Bu sebeple IEC (Uluslararası Elektroteknik Komisyonu) tarafından elektrik güç sistemleri için IEC 61850 ve IEC 60870 haberleşme protokolleri geliştirilmiştir, yeni sürümleri geliştirilmeye devam etmektedir.

Elektrik enerjisi santralleri ve şalt sahalarındaki işlemler sırasında (yük altında kesiciden önce ayırıcının açılmaması gibi) uyulması gereken bazı kurallar vardır. Bu kurallar, SCADA yazılımında program parçaları (script / sıkript) şeklinde oluşturulur. Bu program parçaları ile kullanıcının yanlış işlem yapması engellenir. Oluşabilecek arıza durumları program parçalarında tanımlanarak arıza görselleştirilebilir, alarm olarak kayıt altına alınabilir.

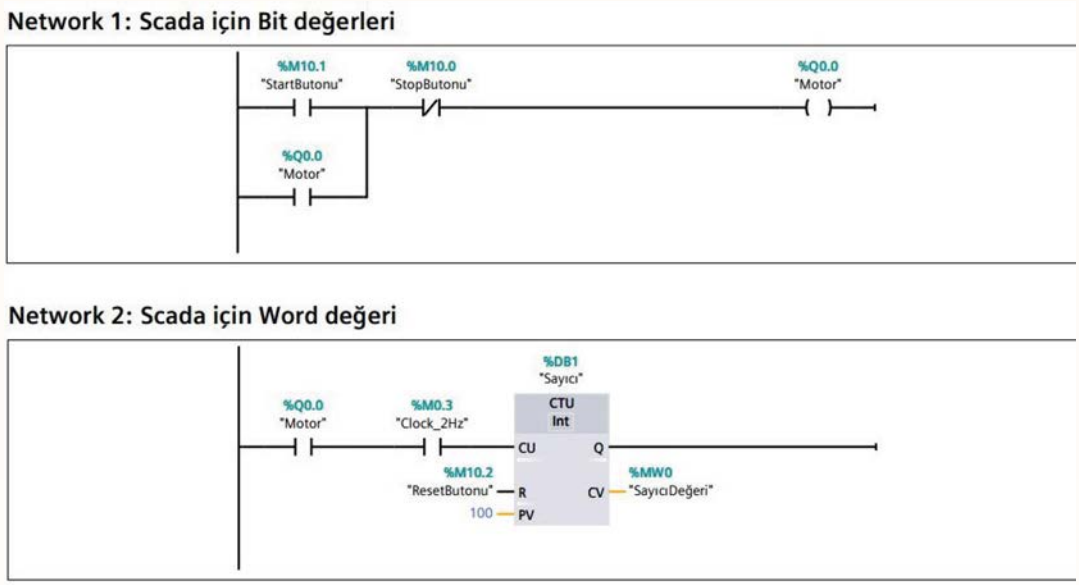
SCADA ile yapılan kontrol işlemleri 100-1000 milisaniye arasında gerçekleşir. Ancak bu sürenin altında gerçekleşmesi istenen işlemler de vardır. Örneğin kısa devre ve yük atma işlemlerinin en fazla 100 milisaniyede yapılması istenir. Bu süre SCADA işlem süresinin altında kalacağından hızlı işlemlerde SCADA yerine özel koruma röleleri ve RTU' lar kullanılır. Bu özel röle ve RTU 'lar işlemi kendisi yapmakla birlikte SCADA ile de haberleşir.

5. UYGULAMA

SCADA İLE PLC HABERLEŞMESİ

AMAÇ: SCADA ile PLC haberleşmesi yapmak.

YÖNERGE: SCADA ile PLC arasındaki haberleşmede farklı protokoller kullanılır. PROFINET bu protokollerden biridir. PROFINET Ethernet ağı üzerinde çalışır, Modbus TCP/IP protokolüne göre geliştirilmiş özellikleri vardır.

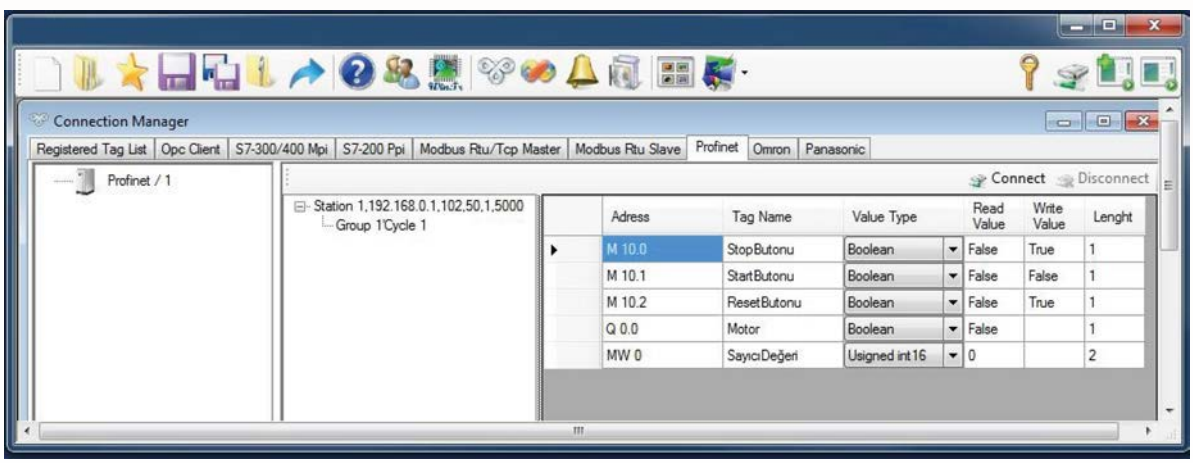


Görsel 9.16: PLC ladder diyagramı

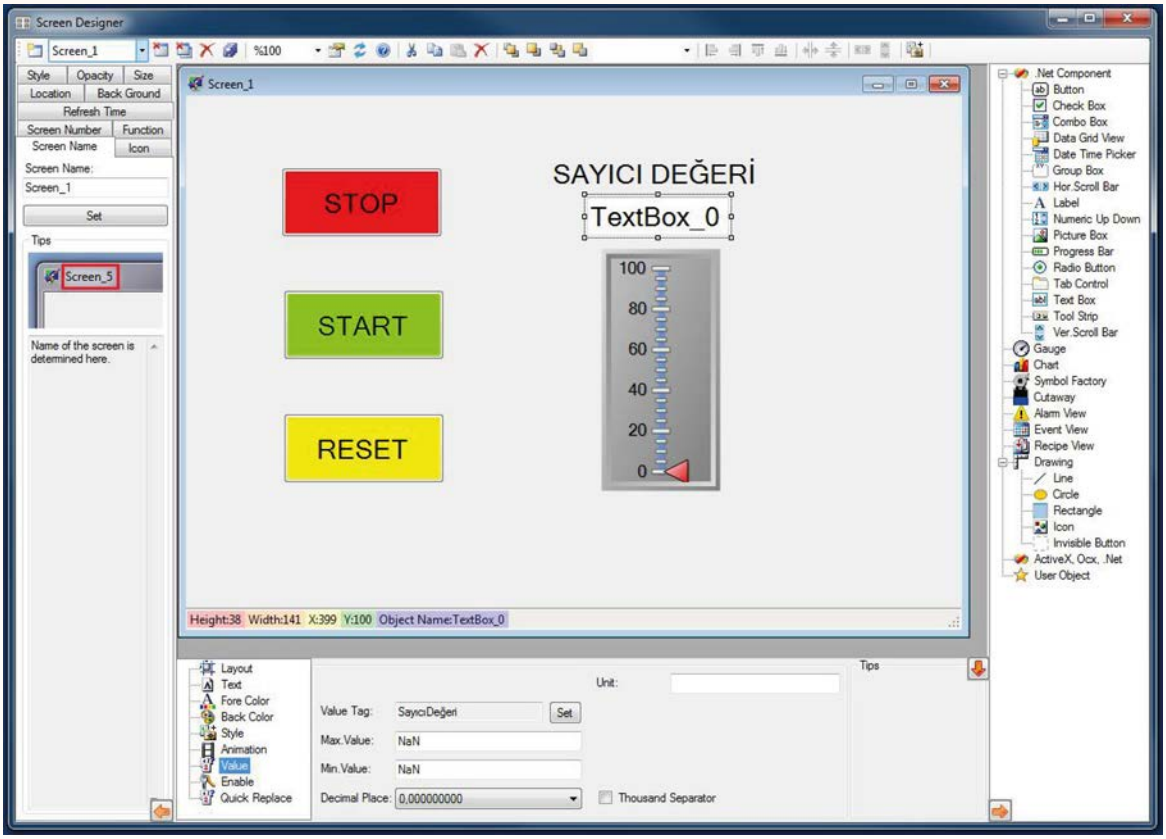
PLC tags

	Name	Data type	Address	Retain
	Motor	Bool	%Q0.0	False
	StopButonu	Bool	%M10.0	False
	StartButonu	Bool	%M10.1	False
	ResetButonu	Bool	%M10.2	False
	SayıcıDeğeri	Int	%MW0	False
	Clock_2Hz	Bool	%M0.3	False

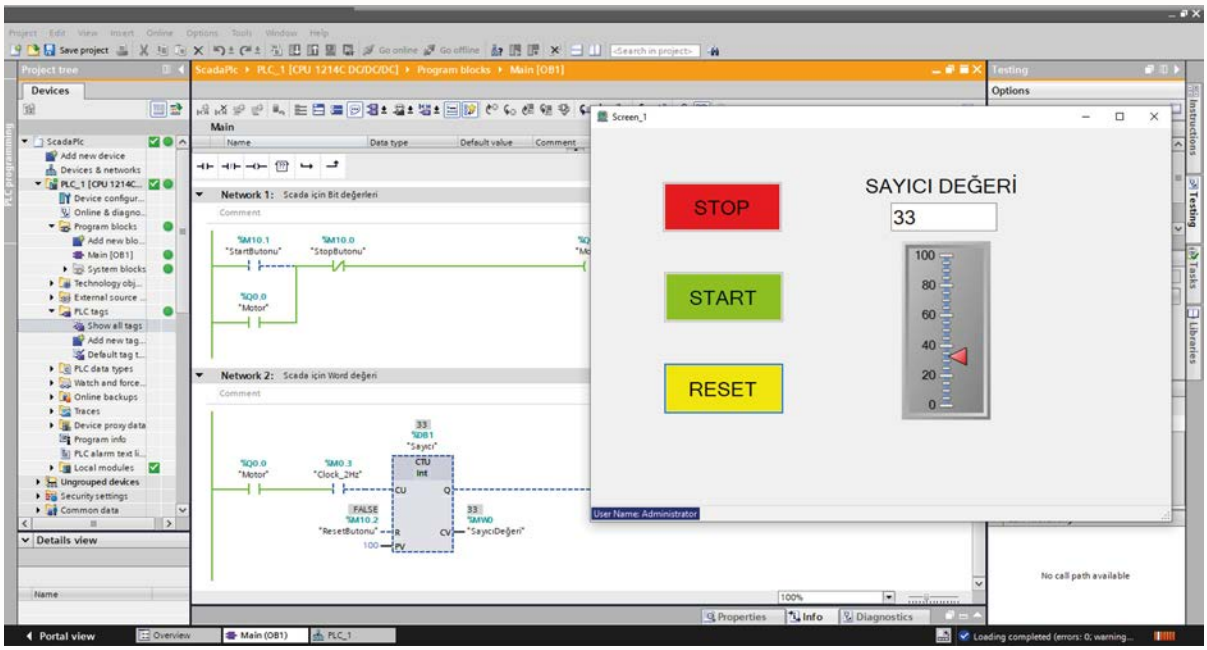
Görsel 9.17: PLC program etiketleri



Görsel 9.18: Bağlantı yöneticisi ekranı



Görsel 9.19: Bileşenlerin oluşturulması



Görsel 9.20: PLC ile bağlantının kurulması

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Bilgisayar	SCADA yazılımı sistem gereksinimlerine uygun	1 adet
SCADA yazılımı	Gereksinimleri karşılayacak özelliklerde	1 adet
Ek yazılımlar	İşletim sistemi, veri tabanı, PLC editör yazılımı	1 adet
PLC	Profinet ağ özellikli	1 adet
El aletleri, kablo	Pense, yan keski, tornavida, ağ kablosu, 0,75 mm^2 kablo	-

İŞLEM BASAMAKLARI

1. PLC ve bilgisayarı ağ kablosu ile birbirine bağlayınız.
2. PLC editör programı ile PLC LADDER diyagramını oluşturunuz (Görsel 9.16).
3. PLC'ye ağ üzerinden tam erişim izni vererek programı yükleyiniz.
4. SCADA programını çalıştırıp, projenize yeni bir isim vererek dosyaları kaydediniz.
5. Bağlantı yöneticisi penceresinden (PLC IP'sini dikkate alıp PROFINET bağlantısı oluşturarak) ilgili değişken tanımlamalarını yapınız (Görsel 9.18).
6. Ekran Tasarımcısı ekranında yeni sayfa oluşturunuz.
7. Sayfa üzerinde Görsel 9.19'da verilen bileşenleri oluşturunuz.
8. Bileşenlerin Değer (Value) kısımlarına, bağlantı yöneticisinde oluşturmuş olduğunuz değişkenleri atayınız (Görsel 9.19).
9. Dosyanızı kaydederek Geliştirici Çalıştırma (Development Runtime) modunda çalıştırınız.
10. PLC ile bağlantı kurunuz (Görsel 9.20).
11. SCADA sayfası ve PLC Editör programını aynı anda görecek şekilde ekranınızı düzenleyiniz.
12. SCADA sayfasından düğmelere tıklayarak SCADA sayfasında ve PLC Editör sayfasında meydana gelen değişimleri gözlemleyiniz.

SORULAR

1. SCADA programında dâhilî ve haricî değişken arasındaki farkları belirtiniz.
2. SCADA ekranından düğmelere tıklandığında, PLC editör ekranındaki ilgili değişkenlerin değişiminde zaman farkı meydana gelmesinin sebebini açıklayınız.

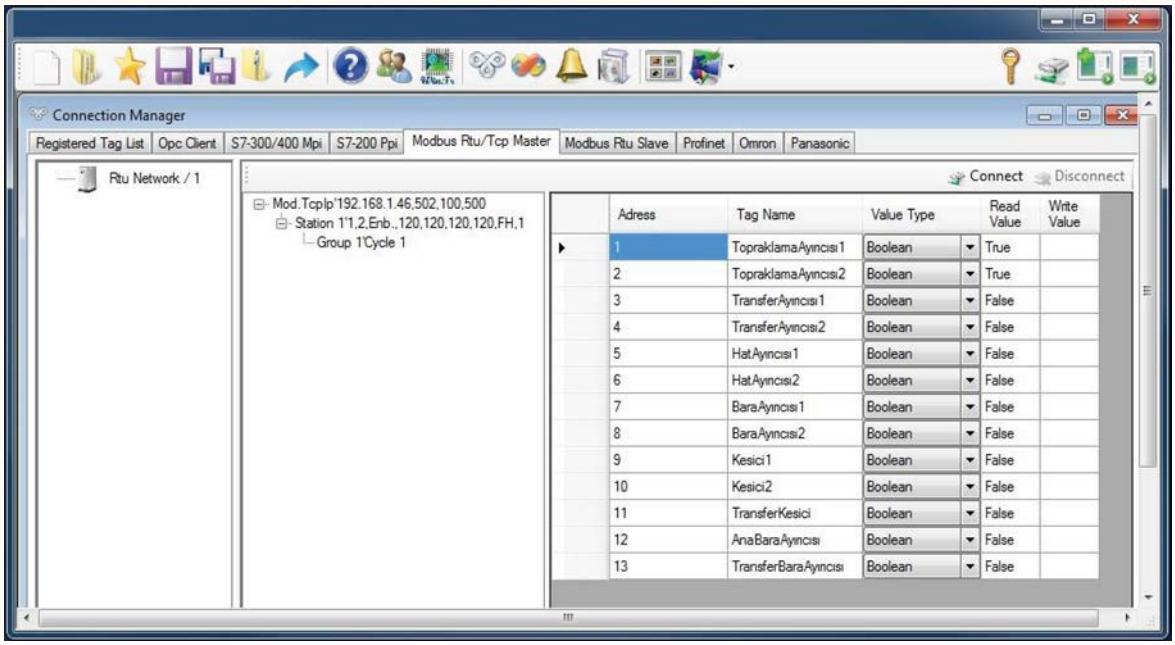
ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	PLC programını oluşturdu.	20	
Sınıfı :	2	SCADA bağlantısını oluşturdu.	20	
Numarası :	3	SCADA programını oluşturdu.	20	
ÖĞRETMEN	4	Uygulamayı çalıştırdı.	20	
Adı Soyadı :	5	Soruların cevaplandırdı.	10	
	6	Verilen sürede işini tamamladı.	10	
İmza :		**TOPLAM PUAN**	100	

6. UYGULAMA

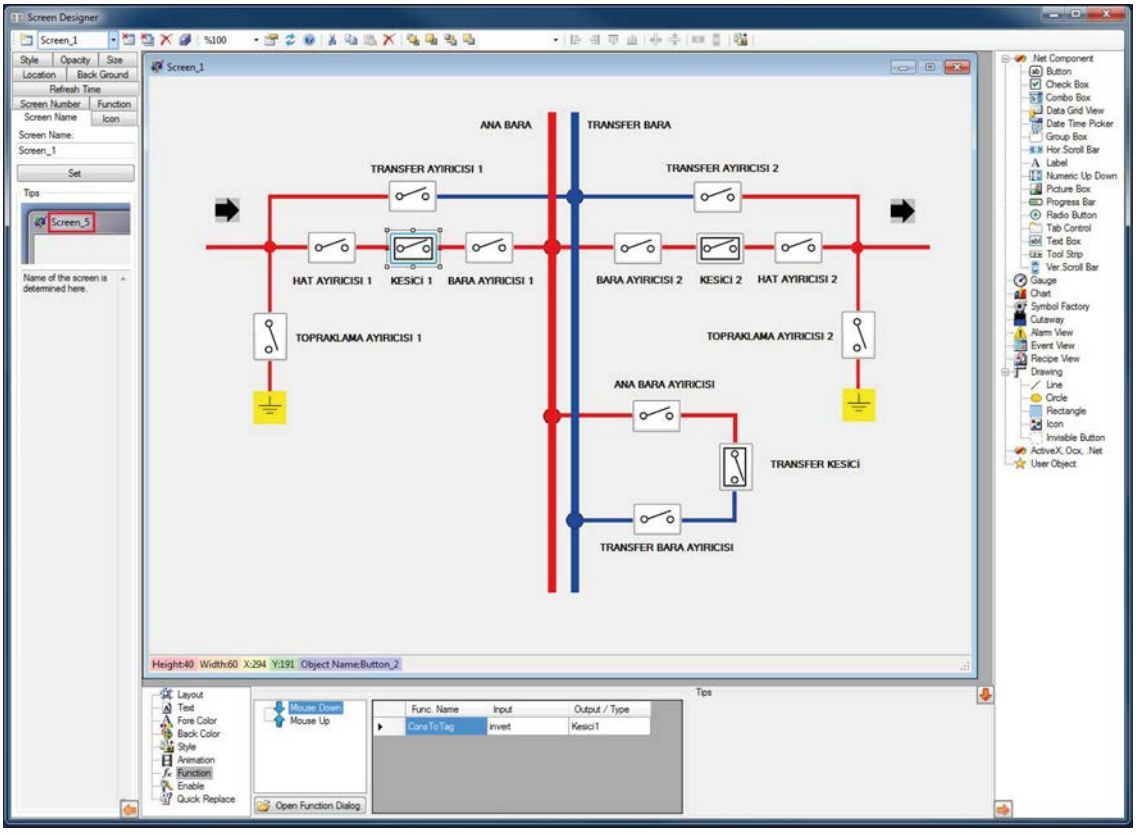
SCADA İLE ŞALT SAHASI KONTROLÜ

AMAÇ: SCADA ile şalt sahası kontrolü yapmak.

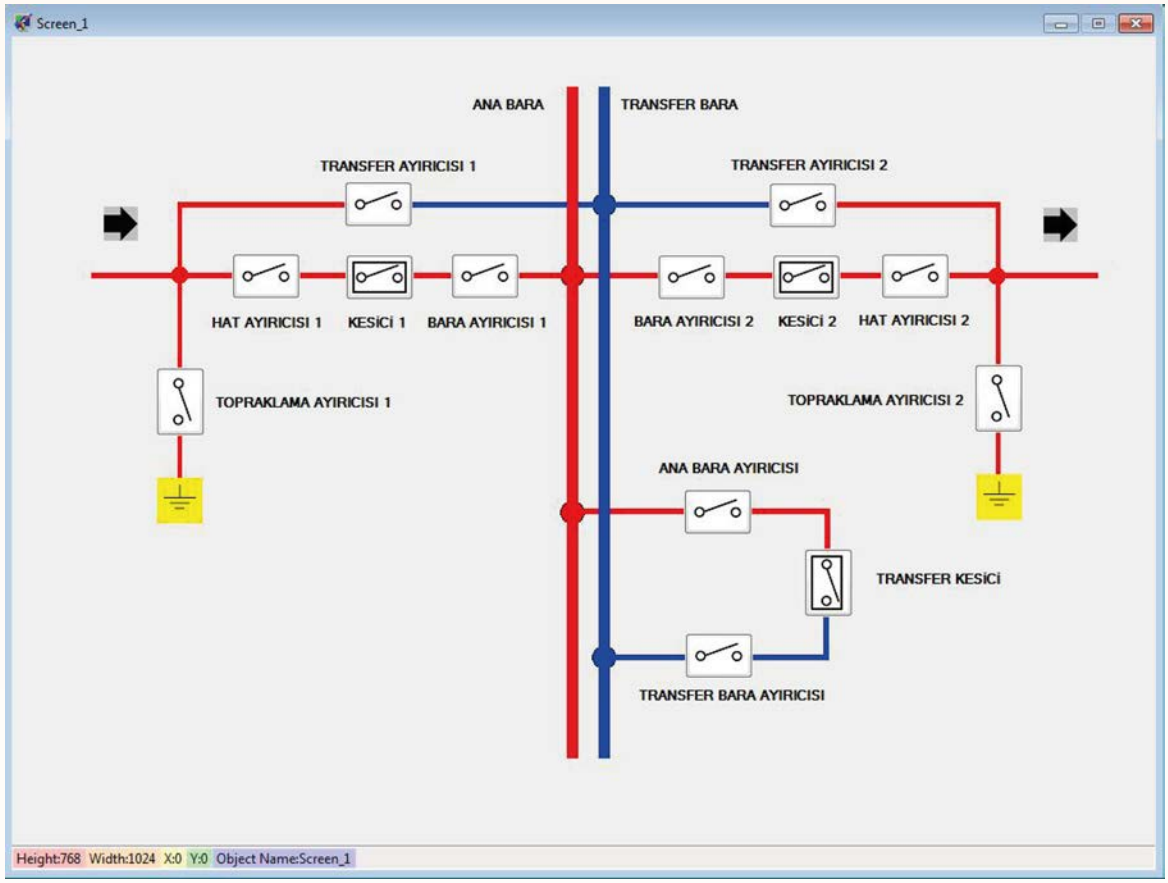
YÖNERGE: SCADA, şalt sahalarında PLC veya RTU'lar ile haberleşir. Uygulamada protokol olarak Modbus TPC/IP kullanılacaktır.



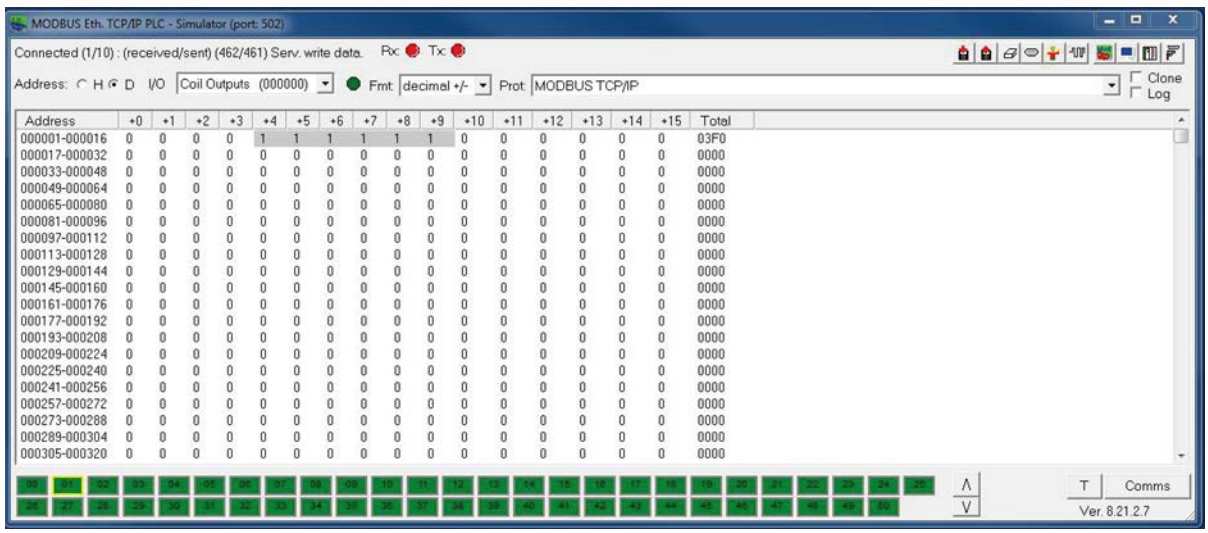
Görsel 9.21: Bağlantı yöneticisi penceresinde Modbus TCP/IP bağlantısı ve değişkenlerin oluşturulması



Görsel 9. 22: Sayfaya bileşenlerin eklenmesi



Görsel 9.23: Geliştirici modunda çalıştırma



Görsel 9.24: 00000-00013 adreslerindeki değişimlerin gözlemlenmesi

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Bilgisayar	SCADA yazılımı sistem gereksinimlerine uygun	1 adet
SCADA yazılımı	Gereksinimleri karşılayacak özelliklerde	1 adet
Ek yazılımlar	İşletim sistemi, veri tabanı, modbus simülatör yazılımı	1 adet

İŞLEM BASAMAKLARI

1. SCADA programını çalıştırıp, projenize yeni bir isim vererek dosyaları kaydediniz.
2. Görsel 9.21'de görüldüğü gibi Bağlantı yöneticisi penceresinde Modbus TCP/IP bağlantısını oluşturunuz.
3. Modbus TCP/IP kısmına bilgisayarınızın IP'sini öğrenerek yazınız (IP için bilgisayar komut satırına CMD yazıp açılan pencerede ipconfig komutunu veriniz. Modbus TCP/IP portu 502'dir.).
4. İlgili bağlantıda okuma ve yazma yapılabilen 1 bitlik Boolean adreslerindeki on üç adet değişkeni oluşturunuz (Görsel 9.21).
5. Yeni bir sayfa oluşturarak sayfaya bileşenleri ekleyiniz (Görsel 9.22).
6. Bara resimlerini ilgili seçeneği kullanarak (Drawing, Rectangle) çiziniz.
7. Düğme resimlerini (kesici açık, kesici kapalı, ayırıcı açık, ayırıcı kapalı, toprak), 60x40 çözünürlüğünde olmak üzere bir grafik programında oluşturunuz (İlgili resimler jpeg, png veya bmp formatlarında olabilir.).
8. Düğme özelliklerinde Back Color Set-Background Off-On Image seçeneklerini kullanarak oluşturduğunuz grafik dosyalarını seçiniz.
9. Düğme özeliklerinde Back Color-Trigger Tag seçeneğini kullanarak temsil ettiği değişkeni seçiniz.
10. Düğme özelliklerinde Function seçeneğini kullanarak Cons To Tag-Invert değişken ismi yoluyla fonksiyonu belirleyiniz.
11. Dosyanızı kaydederek Geliştirici Çalıştırma (Development Runtime) modunda çalıştırınız (Görsel 9.23).
12. Modbus Eth. TCP/IP PLC Simülatör programını çalıştırınız.
13. SCADA sayfasından düğmelere basarak simülatör programındaki 00000-00013 adreslerindeki değişimleri gözlemleyiniz (Görsel 9.24).

SORULAR

1. Şalt sahalarında iki baralı sistem kullanılmasının amacı nedir? Açıklayınız.
2. Modbus Eth. TCP/IP PLC Simülatör programında değişken değerleri için hangi adresler kullanıldı? Açıklayınız.

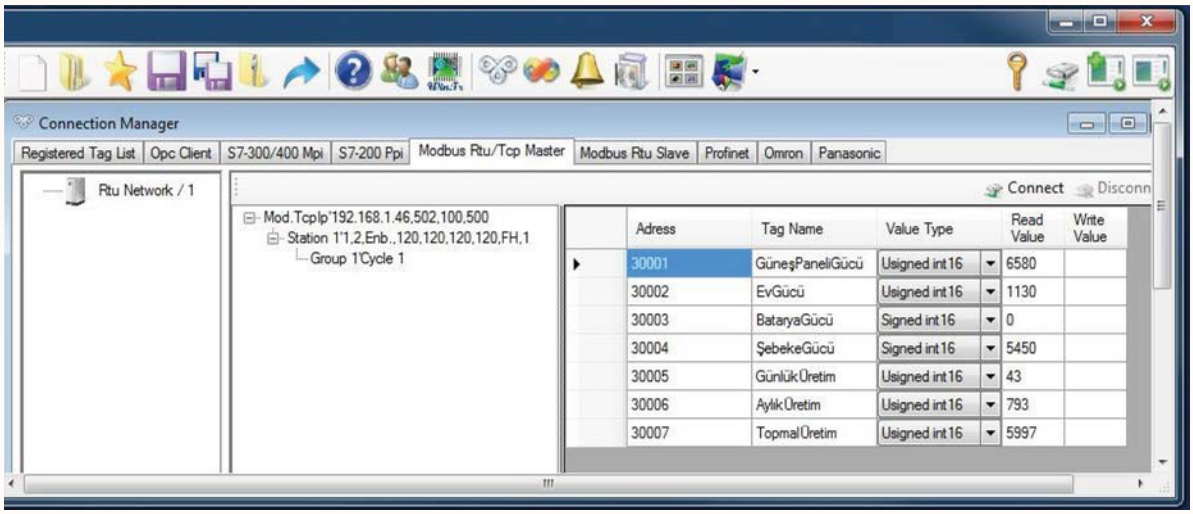
ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Bağlantıyı oluşturdu.	20	
Sınıfı :	2	Değişkenleri tanımladı.	20	
Numarası :	3	Sayfayı oluşturdu.	20	
ÖĞRETMEN	4	Uygulamayı çalıştırdı.	20	
Adı Soyadı :	5	Soruları cevaplandırdı.	10	
	6	Verilen sürede işi tamamladı.	10	
İmza :		**TOPLAM PUAN**	100	

7. UYGULAMA

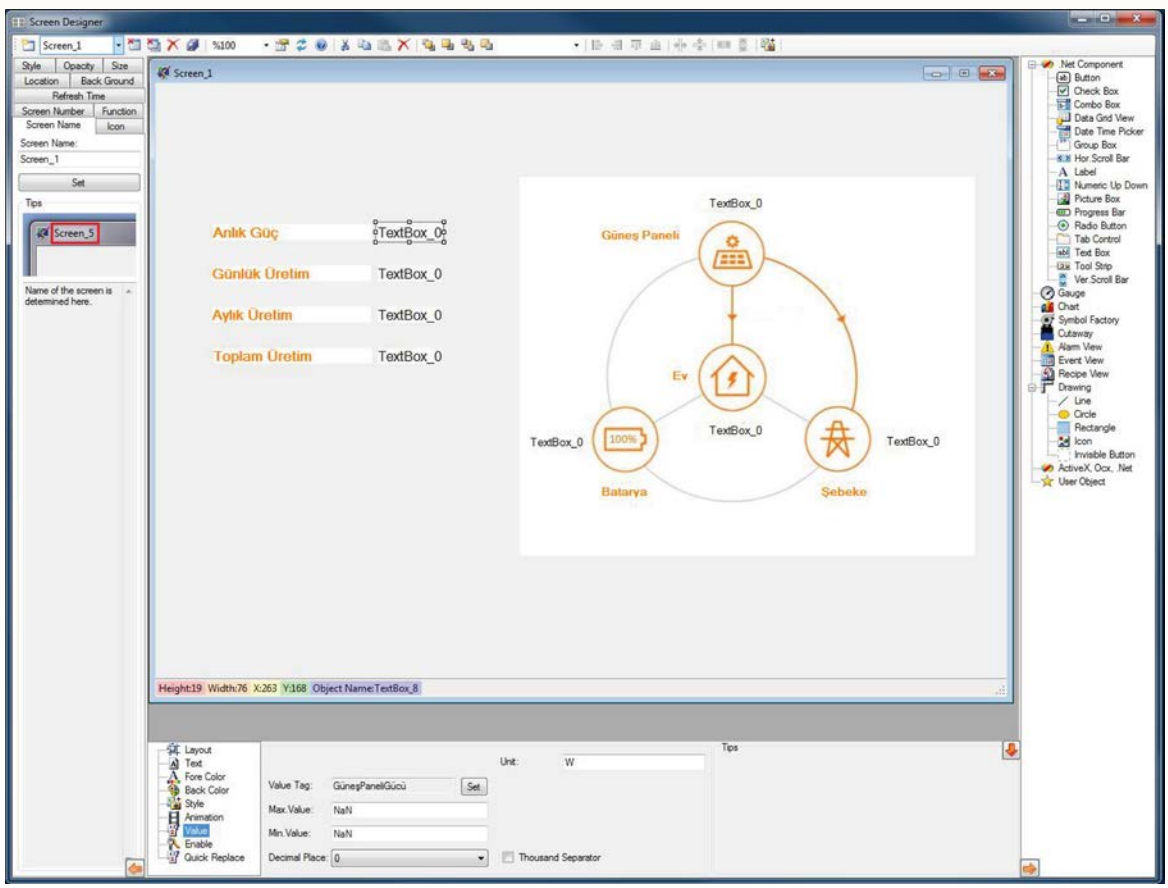
SCADA İLE GÜNEŞ ENERJİSİ KONTROLÜ

AMAÇ: **SCADA ile güneş enerjisi kontrolü yapmak.**

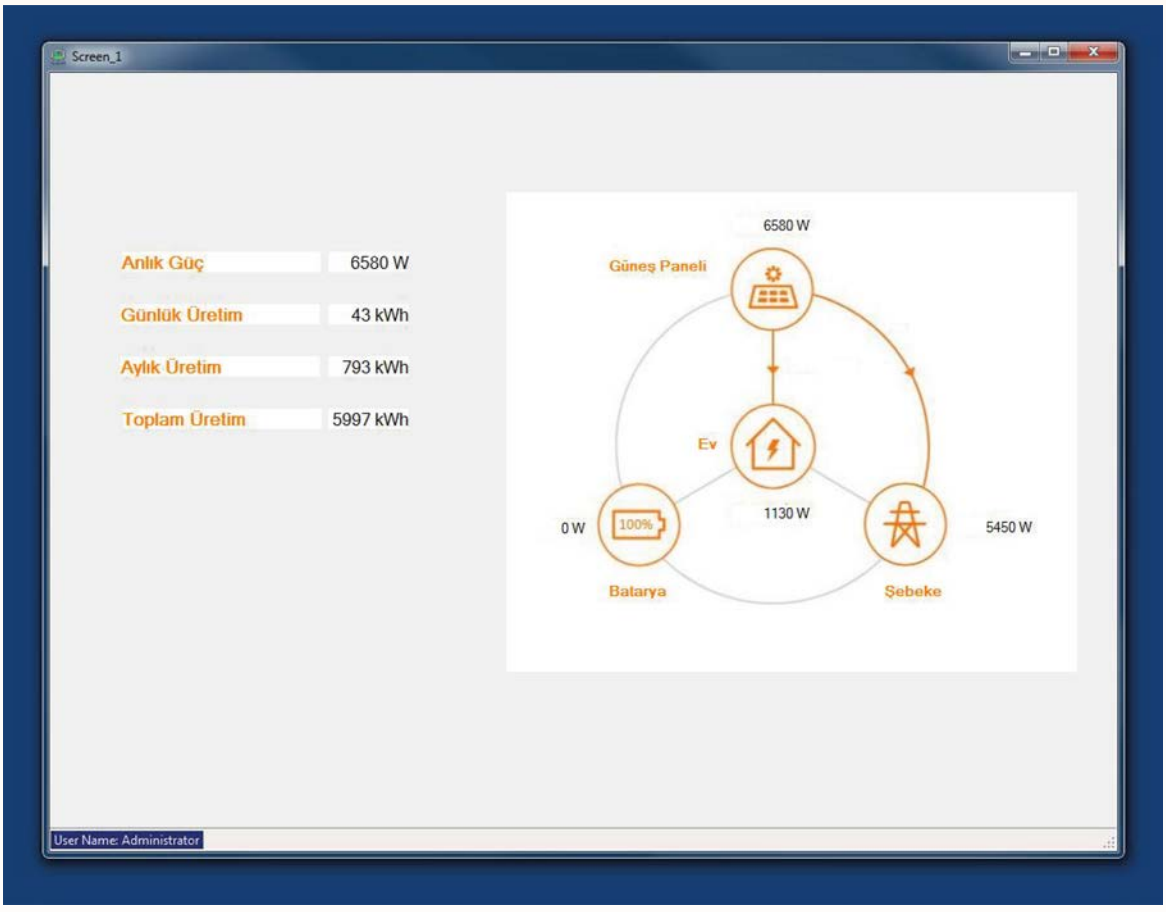
YÖNERGE: **Güneş enerjisi invertörlerinden Modbus protokolü ile veri okumak mümkündür.**



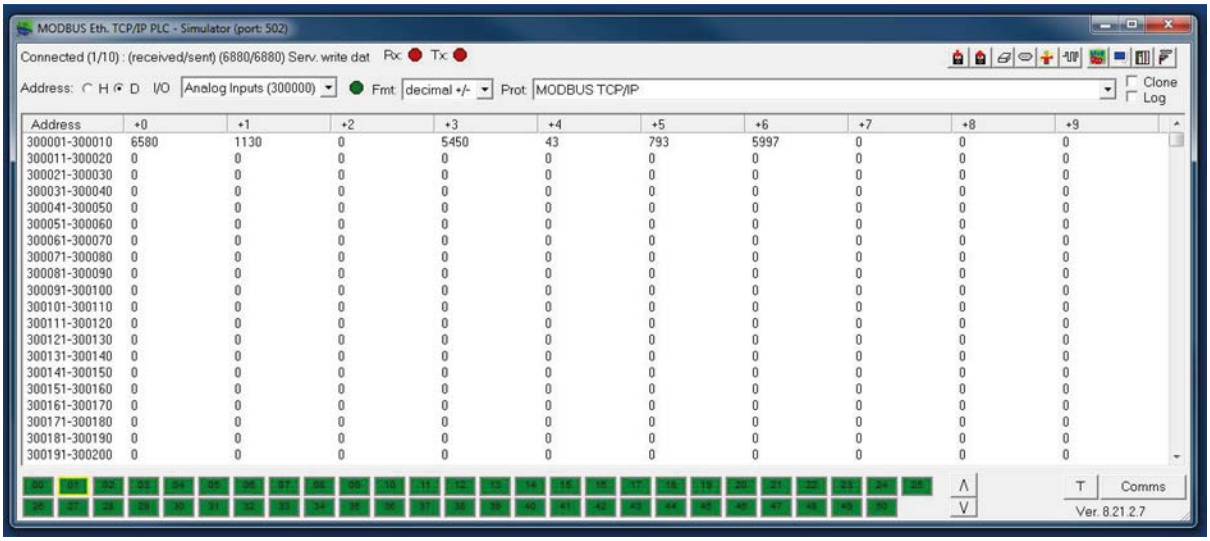
Görsel 9.25: Modbus TCP/IP bağlantısının oluşturulması



Görsel 9.26: Bileşenlerin eklenmesi



Görsel 9.27: Bileşenlerin oluşturulması



Görsel 9.28: Değişimlerin gözlemlenmesi

MALZEME LİSTESİ		
Adı	**Özelliği**	**Miktarı**
Bilgisayar	SCADA yazılımı sistem gereksinimlerine uygun	1 adet
SCADA yazılımı	Gereksinimleri karşılayacak özelliklerde	1 adet
Ek yazılımlar	İşletim sistemi, veri tabanı, modbus simülatör yazılımı	1 adet

İŞLEM BASAMAKLARI

1. SCADA programını çalıştırıp, projenize yeni bir isim vererek dosyaları kaydediniz.
2. Görsel 9.25'te görüldüğü gibi Bağlantı yöneticisi penceresinden Modbus TCP/IP bağlantısını oluşturunuz.
3. Modbus TCP/IP kısmına bilgisayarınızın IP'sini yazınız.
4. İlgili bağlantıda okuma yapılabilen 16 bitlik işaretli ve işaretsiz tam sayı olarak Modbus 30001-30007 adresleri aralığında yedi adet değişkeni oluşturunuz (Görsel 9.25).
5. Yeni sayfa oluşturarak sayfaya bileşenleri ekleyiniz (Görsel 9.26).
6. Ev, batarya gibi resimleri SCADA'nın grafik nesnelerini kullanarak veya herhangi bir grafik programında çiziniz.
7. Oluşturulan dosyayı ilgili formatlardan herhangi biriyle SCADA sayfasına ilave ediniz (Resmi ilave etmek için Resim Kutusu seçeneğini kullanabilirsiniz.).
8. Resim üzerinde ve liste olarak değişkenleri göstermek için diğer bileşenleri oluşturunuz (Görsel 9.27).
9. Dosyanızı kaydederek Geliştirici Çalıştırma modunda çalıştırınız.
10. Modbus Eth. TCP/IP PLC Simülatör programını çalıştırınız.
11. Simülatör programında 30001-30007 adreslerinden değerler girerek SCADA sayfasındaki değişimleri gözlemleyiniz (Görsel 9.28).

SORU

- Uygulamada oluşturulan Modbus Eth. TCP/IP bağlantısı adresine niçin kullanılan bilgisayarın IP adresi yazılır? Açıklayınız.

ÖĞRENCİ	DEĞERLENDİRME ÖLÇÜTLERİ VE PUANLAMA			
Adı Soyadı :	1	Bağlantıyı oluşturdu.	20	
Sınıfı :	2	Değişkenleri tanımladı.	20	
Numarası :	3	Sayfayı oluşturdu.	20	
ÖĞRETMEN	4	Uygulamayı çalıştırdı.	20	
Adı Soyadı :	5	Soruları cevaplandırdı.	10	
	6	Verilen sürede işi tamamladı.	10	
İmza :		**TOPLAM PUAN**	100	

ÖLÇME VE DEĞERLENDİRME

A) Aşağıdaki cümlelerin başındaki boşluğa cümleler doğru ise (D), yanlış ise (Y) yazınız.

1. (…) Scada yazılımı işletim sistemi yazılımına ihtiyaç duymadan çalışır.
2. (…) Modbus RTU protokolü ethernet ağında kullanılır.
3. (…) Profinet protokolü ethernet ağında kullanılır.
4. (…) Modbus protokolünde 16 bitlik okunup yazılabilen değişkenler 40001 adresi ile başlar.
5. (…) IEC 61508 güvenlik standardıdır.

B) Aşağıdaki cümlelerde boş bırakılan yerlere uygun sözcüğü yazınız.

6. Güneş panelinin hareket kontrolünde güneş ışınlarının …… derece gelmesi hedeflenir.
7. Temelde veri izleme, veri kontrol, veri toplama ve veri kaydetme işlemlerini …… yazılımı gerçekleştirir.
8. RS 232 ve RS 485 haberleşme ağında …… protokolü çalışır.
9. SCADA programları verileri kaydetmek için …… programlarına ihtiyaç duyar.
10. SCADA programında bağlantılar …… sayfasında tanımlanır.

C) Aşağıdaki soruların doğru seçeneğini işaretleyiniz.

11. Rüzgâr enerjisinden en iyi faydalanabilmek için rüzgâr türbininde aşağıdakilerden hangisi değiştirilir?
A) Gövde açısı B) Kanat açısı C) Gövde ve kanat açısı
D) Rotor hızı E) Jeneratör hızı

12. SCADA progamında aşağıdakilerin hangisinde sayfa tasarımı yapılır?
A) Drawing B) Picture Box C) Screen Designer
D) Text Box E) User Object

13. Aşağıdakilerden hangisi uzak bağlantı ünitesinin kısaltmasıdır?
A) CTU B) IEC C) PLC
D) PSU E) RTU

14. Şalt sahalarında SCADA daha çok aşağıdakilerden hangisi ile haberleşir?
A) Ayırıcı B) IOT C) Kesici
D) PSU E) RTU

15. SCADA programında programı deneme çalıştırması yapmak için aşağıdakilerden hangisi kullanılır?"
A) Block Runtime B) Bug Runtime C) Development Runtime
D) Test Runtime E) Runtime

CEVAP ANAHTARI

1. ÖĞRENME BİRİMİ

1. D
2. Y
3. D
4. D
5. D
6. D
7. Y
8. 10
9. Binary
10. Harflerde
11. 2
12. 8
13. 7 - 14
14. Transistörler
15. C
16. D
17. E
18. A
19. B
20. F
21. G
22. E
23. C
24. E
25. A
26. B
27. C

2. ÖĞRENME BİRİMİ

1. D
2. Y
3. Y
4. D
5. Y
6. CPU
7. çıkış
8. mikrodenetleyici
9. 
10. A
11. E
12. D
13. A
14. E
15. B

3. ÖĞRENME BİRİMİ

1. D
2. Y
3. Y
4. pinMode
5. analog
6. NTC
7. D
8. A
9. E
10. B
11. C
12. E
13. D
14. A
15. E

4. ÖĞRENME BİRİMİ

1. D
2. Y
3. Y
4. yönü
5. PWM
6. 1,1 V
7. D
8. E
9. B
10. C
11. A
12. D
13. A
14. C
15. B

5. ÖĞRENME BİRİMİ

1. D
2. D
3. Y
4. D
5. Y
6. iki fazın
7. kayma
8. elektriksel kilitleme
9. kontaktör
10. U1-V1-W1
11. C
12. B
13. A
14. B
15. C

6. ÖĞRENME BİRİMİ

1. D
2. Y
3. D
4. Y
5. D
6. 24 V
7. haberleşme
8. yüksektir
9. süreç
10. ek
11. D
12. B
13. C
14. A
15. E

7. ÖĞRENME BİRİMİ

1. D
2. D
3. D
4. Y
5. Y
6. network
7. küçük
8. form
9. simülasyon
10. parametre
11. C
12. C
13. E
14. D
15. A

8. ÖĞRENME BİRİMİ

1. D
2. D
3. Y
4. Y
5. D
6. A
7. B
8. A
9. C
10. E

9. ÖĞRENME BİRİMİ

1. Y
2. D
3. D
4. D
5. Y
6. Dik
7. Scada
8. Modbus
9. Veritabanı
10. Bağlantı yöneticisi
11. C
12. C
13. E
14. E
15. C

KAYNAKÇA

Altun, C. (2006). Yenilenebilir Enerji Eğitim Seti. Ankara: Yıldırım Elektronik.
Çalışkan, N., Kekik, A., Korateş, M.H., Özdemir, H., Öztürk, Z., Yenice, H., Yıldız, A. Otomasyon Atölyesi
Kabalcı, E. Rüzgâr türbinleri kontrol sistemleri
Rüzgâr Türbinleri İnternet Notları, Aselsan.
Yenilenebilir Enerji Teknolojileri Alanı Atölye 11 Çerçeve Öğretim Programı, Ankara, 2020.

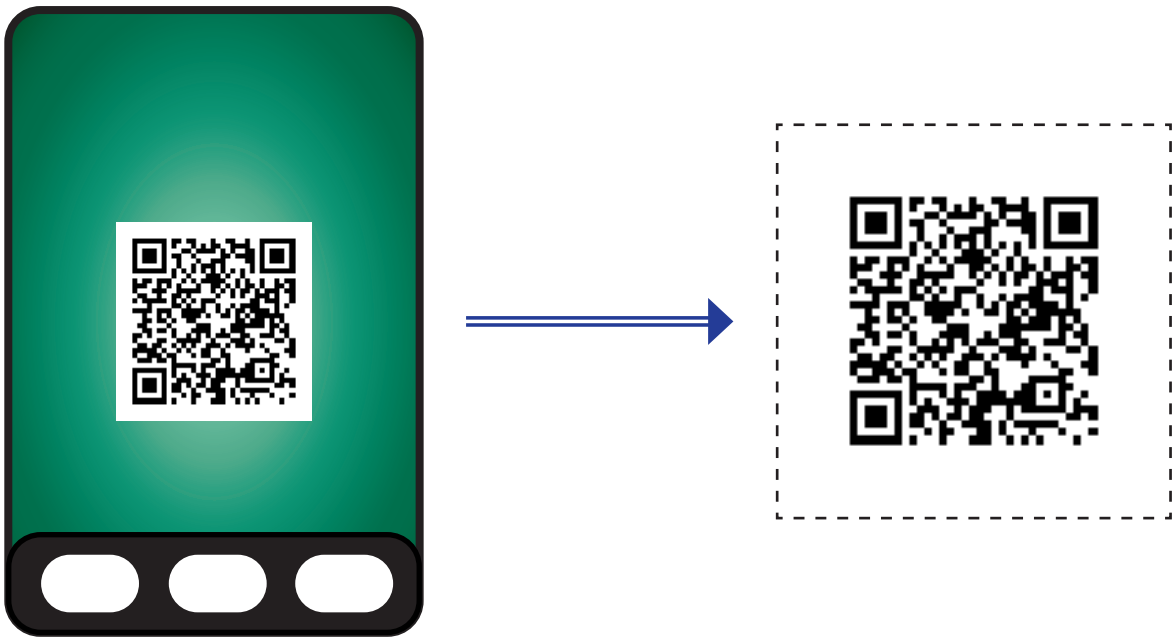
Not: Kaynakça atıf sistemi, APA 6.0 yazım kuralları ve kaynak gösterme biçimine göre düzenlenmiştir.

GENEL AĞ KAYNAKÇASI

https://www.liebherr.com/ (08.10.2022, 15.50)
www.emkoelektronik.com.tr (24.11.2022, 09.10)
www.gmtcontrol.com (11.11.2022, 13.20)
https://www.fultek.com.tr/programlar/scada/ (05.01.2023, 10.00)

GÖRSEL KAYNAKÇA

Görsellerin kaynakçasına ulaşmak için karekodu okutunuz.



http://kitap.eba.gov.tr/karekod/Kaynak.php?KOD=1992