Deney 2- Gösterge Uygulamaları

Deneyin Amacı:

Arduino kartı ile deney tablası kullanarak gösterge bağlantısını yapmayı, uygulama devresini kurmayı, giriş/çıkış portlarının kullanılmasını öğrenmek için gösterge devre ve modüllerinin çalışma yöntemlerinin, donanımlarının incelenmesi ve programlanmasının yapılmasıdır.

13

Deney Öncesi Yapılacak İşlemler:

Bu deneyde kullanılan LED göstergelerin, sıvı kristal göstergelerin (LCD) elektronik devre malzemelerinin işlevleri, kullanım özellikleri, malzemenin fiziksel görünümü, adının ve değerinin üzerindeki verilerden okunarak elde edilmesi öğrenilecektir. Burada verilen uygulama programları incelenecek, her komut satırının işlevi, açıklaması anlaşılacaktır.

Deneyde Yapılacak İşlemler:

Burada verilen 7-parça LED gösterge ve 16x2 LCD ekranın nasıl bağlanacağı ve bir Arduino ile nasıl sürüleceği, örnek uygulama programları ile gösterilecektir.

a) LED Gösterge Uygulaması

Arduino ve 7-parça LED Gösterge Donanımı:

7-parça LED gösterge basit bir cihazdır. Şekil 2-1'de 7-parça LED göstergenin parça tanımları, iç bağlantısı, uç tanımları ve görünümü verilmiştir. Bu şekilde görüldüğü gibi birer uçları bağlanarak ortak yapılmış 8 LED (ondalık nokta, decimal point, dp) ile oluşmuştur. LED'lerin anotları ortak ise ortak anot (Common Anode, CA), katotları ortak ise ortak katot (Common Cathode, CC), 7-parça LED gösterge olarak adlandırılır. Sayıları oluşturmak için Şekil 2-4'te gösterilen şekilde farklı parça kombinasyonlar kullanılarak düzenlenebilir.



Şekil 2-1. 7-parça LED göstergenin görünümü, parça ve uç tanımları, ve iç bağlantısı

Arduino UNO karta 7- parça LED gösterge bağlamak için bu uygulamada kullanılan Tablo 2-1'de verilen eşleşmeleri kullanarak deney tablasında Şekil 2-3a'daki elektrik devresi Şekil 2-3b'deki montaj görüntüsünde verilen şekilde kurulacaktır.

Tablo 2-1. Arduino UNO ile 7	-parça LED	gösterge uç	eşleşmeleri
------------------------------	------------	-------------	-------------

Arduino UNO	2	3	4	5	6	7	8
7-parça LED Gösterge	7 (a)	6 (b)	4 (c)	2 (d)	1 (e)	9 (f)	10 (g)

7-parça LED gösterge Arduino G/Ç uçlarına doğrudan bağlanırsa, mikrodenetleyici, G/Ç uçları veya LED gösterge zarar görebilecektir! Zarar görmemesi, akımın sınırlaması için seri direnç kullanılması gerekir. Bu direnç değerinin Ortak Anot LED için hesaplanması, Şekil 2-2'de verilen eşdeğer devre kullanılarak yapılabilir. LED 'lerin ortak anot ucu VCC'ye bağlanır. Bunun beraber, Arduino kart ile parça uçları arasındaki bağlantılarda her bir G/Ç ucu ile Şekil 2-3'te verilen şekilde bir direnç kullanılır.



Şekil 2-3. LED gösterge uygulaması (a) Elektrik Devresi (b) Montaj Görüntüsü

Ondalık sayıların 7-parça LED göstergedeki şekilleri ve parça tanımları Şekil 2-4'te gösterilmiştir. Bu şekillerin göstergede görülmesi için Arduino kartın program çıkış uçlarına yazılacak parça kodları, ortak anot LED gösterge için 0'dan 9'a kadar sayıların veri dönüşüm kod tablosu aşağıda Tablo 2-2 'de verilmiştir.

Tablo 2-2. Ortak Anot 7-parça LED gösterge için veri dönüşüm kod tablosu



Şekil 2-4. Ondalık sayıların 7-parça LED göstergedeki şekilleri ve parça tanımları.

Arduino ve 7-parça LED Gösterge Yazılımı:

Bir basamak 7-parça LED göstergede, sürekli olarak 0'dan 9'a kadar sayan bir program. **1.Uygulama programı:**

Program uzun, ayrıntılı bir yöntem kullanan ancak basit bir programdır.

```
// İSTÜN Bilgisayar/Yazılım/Mekatronik Müh.
// Arduino ile ortak anot LED gösterge "0-9" sayan basit bir program
// BL306 Mikroişlemciler ve Gömülü Sist. Lab.
// Deney 2: Gösterge Uygulamaları, a) Arduino ile 7-parça LED Gösterge
Uygulaması
// (c)Y.Doc.Dr.Tuncay UZUN
// Önce 7-parça LED Göstergeye bağlı Arduino uçları çıkış yapılır
void setup() {
Serial.begin(9600);
 pinMode(2, OUTPUT); // (a) parçası
 pinMode(2, OUTPUT); // (d) parças1
pinMode(3, OUTPUT); // (b) parças1
pinMode(4, OUTPUT); // (c) parças1
pinMode(5, OUTPUT); // (d) parças1
pinMode(6, OUTPUT); // (e) parças1
                       // (f) parçası
 pinMode(7, OUTPUT);
 pinMode(8, OUTPUT);
                       // (g) parçası
}
// Ortak anot için 0 = LED ON, 1 = LED OFF:
void loop() {
// Göstergeye '0' yaz
digitalWrite(2,0);digitalWrite(3,0);digitalWrite(4,0);digitalWrite(5,0);
digitalWrite(6,0);digitalWrite(7,0);digitalWrite(8,1);
delay(1000); //yazılan sayının görülmesi için 1 saniye bekle
// Göstergeve '1' vaz
digitalWrite(2,1);digitalWrite(3,0);digitalWrite(4,0);digitalWrite(5,1);
digitalWrite(6,1);digitalWrite(7,1);digitalWrite(8,1);
delay(1000); //yazılan sayının görülmesi için 1 saniye bekle
// Göstergeye '2' yaz
digitalWrite(2,0);digitalWrite(3,0);digitalWrite(4,1);digitalWrite(5,0);
 digitalWrite(6,0);digitalWrite(7,1);digitalWrite(8,0);
delay(1000); //yazılan sayının görülmesi için 1 saniye bekle
// Göstergeye '3' yaz
digitalWrite(2,0);digitalWrite(3,0);digitalWrite(4,0);digitalWrite(5,0);
 digitalWrite(6,1);digitalWrite(7,1);digitalWrite(8,0);
delay(1000); //yazılan sayının görülmesi için 1 saniye bekle
// Göstergeye '4' yaz
 digitalWrite(2,1);digitalWrite(3,0);digitalWrite(4,0);digitalWrite(5,1);
 digitalWrite(6,1);digitalWrite(7,0);digitalWrite(8,0);
delay(1000); //yazılan sayının görülmesi için 1 saniye bekle
// Göstergeye '5' yaz
digitalWrite(2,0);digitalWrite(3,1);digitalWrite(4,0);digitalWrite(5,0);
 digitalWrite(6,1);digitalWrite(7,0);digitalWrite(8,0);
delay(1000); //yazılan sayının görülmesi için 1 saniye bekle
// Göstergeye '6' yaz
 digitalWrite(2,0);digitalWrite(3,1);digitalWrite(4,0);digitalWrite(5,0);
 digitalWrite(6,0);digitalWrite(7,0);digitalWrite(8,0);
delay(1000); //yazılan sayının görülmesi için 1 saniye bekle
// Göstergeye '7' yaz
digitalWrite(2,0);digitalWrite(3,0);digitalWrite(4,0);digitalWrite(5,1);
digitalWrite(6,1);digitalWrite(7,1);digitalWrite(8,1);
delay(1000); //yazılan sayının görülmesi için 1 saniye bekle
// Göstergeye '8' yaz
digitalWrite(2,0);digitalWrite(3,0);digitalWrite(4,0);digitalWrite(5,0);
digitalWrite(6,0);digitalWrite(7,0);digitalWrite(8,0);
delay(1000); //yazılan sayının görülmesi için 1 saniye bekle
```

// Göstergeye '9' yaz digitalWrite(2,0);digitalWrite(3,0);digitalWrite(4,0);digitalWrite(5,0); digitalWrite(6,1);digitalWrite(7,0);digitalWrite(8,0); delay(1000); //yazılan sayının görülmesi için 1 saniye bekle

2.Uygulama programi:

Yukarıdaki programda 7-parça LED gösterge bit desenleri için tekrarlanan yapıda satırlar içermektedir. Bu tekrarlayan satırları oluşturmak için 2 boyutlu bir dizi ve sayma işleminde "for" döngüsü kullanılarak daha kısa ve güzel bir programdır.

```
// İSTÜN Bilgisayar/Yazılım/Mekatronik Müh.
// Arduino ile ortak anot LED gösterge "0-9" sayan iyi bir program
// BL306 Mikroişlemciler ve Gömülü Sist. Lab.
// Deney 2 Gösterge Uygulamaları, a) Arduino ile 7-parça LED Gösterge Uygulaması
// (c)Y.Doc.Dr.Tuncay UZUN
// Önce 7-parça LED Göstergeye bağlı Arduino uçları çıkış yapılır
void setup() {
Serial.begin(9600);
  pinMode(2, OUTPUT); // (a) parçası
  pinMode(3, OUTPUT); // (b) parçası
  pinMode(4, OUTPUT); // (c) parçası
                      // (d) parçası
  pinMode(5, OUTPUT);
  pinMode(6, OUTPUT);
                      // (e) parçası
                      // (f) parçası
  pinMode(7, OUTPUT);
  pinMode(8, OUTPUT);
                      // (g) parçası
}
// 0'dan 9'a kadar LED gösterge kod desenlerinin tablosu
// Ortak anot için 0 = \text{LED ON}, 1 = \text{LED OFF}:
byte seven seg digits[10][7] = {
// a b c d e f g parça kodu
  { 0, 0, 0, 0, 0, 0, 1 }, // "0" için parça deseni
  { 1, 0, 0, 1, 1, 1, 1 }, // "1" için parça deseni
  { 0, 0, 1, 0, 0, 1, 0 }, // "2" için parça deseni
  { 0, 0, 0, 0, 1, 1, 0 }, // "3" için parça deseni
  { 1, 0, 0, 1, 1, 0, 0 }, // "4" için parça deseni
  { 0, 1, 0, 0, 1, 0, 0 }, // "5" için parça deseni
  { 0, 1, 0, 0, 0, 0, 0 }, // "6" için parça deseni
  { 0, 0, 0, 1, 1, 1, 1 }, // "7" için parça deseni
  { 0, 0, 0, 0, 0, 0, 0 }, // "8" için parça deseni
  { 0, 0, 0, 0, 1, 0, 0 } // "9" için parça deseni
// 2 3 4 5 6 7 8 Arduino UNO uç numaraları
};
// Tablodan bakma yöntemiyle sayılan değerin, 7-parça desen tanımını
// matristen alarak porta yazan altprogram
void sevenSegWrite(byte digit) {
byte pin = 2;
  for (byte segCount = 0; segCount < 7; ++segCount) {</pre>
    digitalWrite(pin, seven seg digits[digit][segCount]);
    ++pin;
  }
}
// 7-parca LED göstergede 1 saniye aralıkla "0-9" sayılması
void loop() {
  for (byte count = 0; count < 10; ++count) { // 0-9 Saymaya başla
    sevenSegWrite(count); // Sayılan değeri göstergeye yaz
    Serial.println(count);
    delay(1000); // Sayılan değerin görülebilmesi için 1 saniye bekle
  }
}
```

b) LCD Ekran Uygulaması

Arduino ve LCD Ekran Donanımı:

LCD ekran karmaşık, programlanabilir bir elektronik cihazdır. Şekil 2-5'de 16x2, 2 satırda 16 karakter yazılabilen bir LCD göstergenin uç numaraları ve görünümü verilmiştir. Tablo 2-3'te uç tanımları ve Tablo 2-4'te ise Arduino UNO ile gösterge uç eşleşmeleri verilmiştir.



Şekil 2-5. 16x2 LCD göstergenin uç numaraları ve görünümü

Tablo 2-3. TOX2 LOD gosterge uç tarinman					
16x2 LCD	Uç Adı	Açıklama			
1	VSS	GND			
2	VDD	5V			
3	VO	LCD için Güç Kaynağı			
4	RS	Yazmaç Seçimi (H:Veri / L:Komut			
5	R/W	Okuma/Yazma (H:Read / L:Write)			
6	E	İzin (Enable)			
7-14	D0-D7	Veri Yolu uçları			
15 / 16	A/K	LCD Arka ışık Anot / Katot			

Tablo 2-3. 16x2 LCD gösterge uç tanımları

Tablo 2-4. Arduino UNO ile LCD ekran uç eşleşmeleri

Arduino UNO	2	3	4	5	11	12
16x2 LCD	14 (D7)	13 (D6)	12 (D5)	11 (D4)	6 (E)	4 (RS)

Arduino UNO karta LCD ekran bağlamak için bu uygulamada kullanılan Tablo 2-4'te verilen eşleşmeleri kullanarak deney tablasında Şekil 2-6a'daki elektrik devresi Şekil 2-6b'deki montaj görüntüsünde verilen şekilde kurulacaktır.





Arduino ve LCD ekran Yazılımı:

16 karakter 2 satır LCD ekranda ilk olarak "Merhaba Dunya" yazan, sonra geçen süreyi saniye olarak ekrana ve seri porta yazan bir programdır. Bu program, Arduino IDE içinde hazır bulunan, yükleme gerektirmeyen "LiquidCrystal" kitaplığını kullanan LCD ekran

kullanımı için bir başlangıç programıdır. Bu program gibi değişik örnek programlar Arduino IDE ana menüde File>>Examples>> LiquidCrystal alt menülerinde bulunmaktadır. Ayrıca <u>https://docs.arduino.cc/learn/electronics/lcd-displays/</u> adresinde LCD ile Arduino donanımı ve yazılımı hakkında ek bilgileri de görebilirsiniz. Buradaki "LiquidCrystal" kitaplığı, Hitachi HD44780 yonga setini kullanan metin tabanlı LCD ekranları kontrol etmenizi sağlar.

1.Uygulama programı:

Aşağıda, Tablo 2-5'te "LiquidCrystal" kitaplığında bulunan komutlar verilmiştir. Bu komutların işlevlerine deney öncesinde kısaca bakınız. Programda ilk olarak LCD ekranın çalışma şekli ve LCD ekranın Arduino bağlantısı için uç tanımları yapılır. Sonra kütüphane komutları kullanılarak LCD ekranın bir satırındaki karakter sayısı ve satır sayısı ayarı 16x1 olarak yapılır ve ekrana "Merhaba, Dunya!" yazılır.

LiquidCrystal()	LiquidCrystal(rs, enable, d4, d5, d6, d7)							
	LiquidCrystal(rs, rw, enable, d4, d5, d6, d7)							
	LiquidCrystal(rs, enable, d0, d1, d2, d3, d4, d5, d6, d7)							
	LiquidCrystal(rs, rw, enable, d0, d1, d2, d3, d4, d5, d6, d7)							
begin()	<pre>lcd.begin(cols, rows, charsize)</pre>							
clear()	<pre>lcd.clear()</pre>							
home()	lcd.home()							
setCursor()	<pre>lcd.setCursor(col, row)</pre>							
write()	<pre>lcd.write(data)</pre>							
print()	lcd.print(data)							
	lcd.print(data, BASE)							
cursor()	lcd.cursor()							
noCursor()	lcd.noCursor()							
blink()	<pre>lcd.blink()</pre>							
noBlink()	<pre>lcd.noBlink()</pre>							
display()	<pre>lcd.display()</pre>							
noDisplay()	<pre>lcd.noDisplay()</pre>							
scrollDisplayLeft()	<pre>lcd.scrollDisplayLeft()</pre>							
scrollDisplayRight()	<pre>lcd.scrollDisplayRight()</pre>							
autoscroll()	lcd.autoscroll()							
noAutoscroll()	lcd.noAutoscroll()							
leftToRight()	<pre>lcd.leftToRight()</pre>							
rightToLeft()	<pre>lcd.RightToleft()</pre>							
createChar()	<pre>lcd.createChar(num, data)</pre>							

Гаblo 2-5. L	.iauidCrv	stal Arduino	kitaplığında	bulunan	komutlar

Örnek program:

```
#include <LiquidCrystal.h>
// LDC ekran çalışma şeklinin tanımlanması
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
void setup()
{
    lcd.begin(16,1);
    lcd.print("Merhaba, Dunya!");
}
void lcop() {}
```

2.Uygulama programı:

Aşağıdaki programda ilk olarak "LiquidCrystal" kütüphanesi ile Arduino uç tanımlarını yaparak bağlantısını sağlar. Sonra kütüphane komutlarını kullanarak ekran boyut ayarını yapar ve ekrana "Merhaba Dunya" yazar. Ana programda ise ekranın alt satır başına saniye olarak geçen zamanı yazar.

```
// İSTÜN Bilgisayar/Yazılım/Mekatronik Müh.
// Arduino ile LCD ekranda "Merhaba Dunya" yazan uygulama programı
// BL306 Mikroişlemciler ve Gömülü Sist. Lab.
// Deney 2: Gösterge Uygulamaları, b) Arduino ile LCD Uygulaması
// (c)Y.Doc.Dr.Tuncay UZUN
#include <LiquidCrystal.h>
// ilk koşullandırmada kütüphane LCD arayüzü tanımlar
// Aynı zamanda Arduino uç bağlantılarını da belirler
const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
void setup() {
  // LCD ekranın sütun ve satır sayısı:
  lcd.begin(16, 2);
  // LCD ekrana ilk mesajın ilk satıra yazılması:
  lcd.print("Merhaba, Dunya!");
}
void loop() {
  // imleci sütun 0, satır 1'e alt satırın başına ayarla
  lcd.setCursor(0, 1); lcd.print("t=");
  lcd.setCursor(2, 1); delay(100);
  // saniye olarak gecen zamanın ekrana yazılması
  lcd.print(millis() / 1000); lcd.print(" saniye");
}
```

Sorular:

- **1.** 7-parça LED göstergelerin türler, çalışma yöntemleri ve özellikleri nelerdir?
- 2. Çok basamaklı 7-parça LED gösterge Arduino karta nasıl bağlanır? Nasıl Çalışır?
- **3.** LCD ekran türleri, çalışma yöntemleri ve özellikleri nelerdir?
- 4. Uygulama programları çalışırken hangi uçtan hangi işaret gönderilir?
- 5. Uygulama programının başlangıcında bulunan "setup" fonksiyonu ne için kullanılır?
- 6. Uygulama programının son kısmında bulunan "loop" fonksiyonu ne için kullanılır?
- 7. "LiquidCrystal" komutunun özelliği ve çalışma şekli nasıldır?
- 8. "lcd.begin" komutunun özelliği ve çalışma şekli nasıldır?
- 9. "lcd.print()"komutunun özelliği ve çalışma şekli nasıldır?
- 10."lcd.setCursor(0, 1)" komutunun özelliği ve çalışma şekli nasıldır?