

Hazırlayan: Öğ. Gör. Sabri Uzuner https://www.researchgate.net/profile/Sabri\_Uzuner

# **NE ÖĞRENECEĞİZ?**

- Farklı yollarla 16X2 LCD programlamak.
- Ekrana metin yazdırmak.
- Ekranda metni kaydırmak.
- Özel karakterler kullanmak
- Metni kırpmak ve istenilen yere konumlandırmak.
- Sayıcı devreler tasarlamak

# Arduino LCD Kullanımı

**Arduino LCD** kullanımı için Arduino ile LCD ekranımızın bağlantılarını yaptıktan sonra Arduino kütüphanelerinden biri olan **LiquidCrystal** kütüphanesini kullanarak LCD ekranlarımızı kontrol edebiliriz. Bu kütüphane Arduino kartlarını kullanarak, Hitachi HD44780 sürücüsü ile uyumlu olan LCD'leri kolay bir şekilde kontrol edebilmemizi sağlamaktadır. Dolayısıyla HD44780 sürücüsünü destekleyen çeşitli boyutlardaki LCD ekranların kullanımı için Arduino LiquidCrystal kütüphanesini kullanabiliriz.

Ders boyunca kullanılcak ekran HD44780 sürücüsünü destekleyen ve piyasada çok uygun bir fiyata bulunabilecek 16×2 LCD ekran . Ekranın 16×2 olarak adlandırılması LCD'nin 2 satıra sahip olduğu ve satır başına 16 karakter görüntüleyebileceği anlamına geliyor. Yani ekran aynı anda 32 adet karakter görüntüleyebiliyor. Bunun dışında kaydırma yaparak 32 karakterden fazlasını da görüntüleyebilirsiniz.

Bu yazıdaki kodlar, standart Hitachi HD44780 sürücüsünü kullanan LCD'ler için yazılmıştır. LCD'nizin 16 iğnesi varsa, muhtemelen Hitachi HD44780 sürücüsüne sahiptir.

# Arduino LCD Kullanımı İçin Gerekli Malzemeler

- Arduino Kartı
- LCD Hitachi HD44780 sürücüsü ile uyumlu LCD ekran
- > 10 k ohm potansiyometre
- 220 ohm direnç
- Bağlantılar için breadboard ( ekmek tahtası ) ve kablolar

# Arduino 16×2 LCD Bağlantıları

| VSS<br>VDD<br>VEE | RS<br>V | 00<br>02<br>05<br>05<br>05<br>05<br>05<br>05<br>05<br>05<br>05<br>05<br>05<br>05<br>05 |
|-------------------|---------|--|
| 9 57 7            | 4 G Ø   | 10 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   |

# Arduino 16×2 LCD Bağlantıları

| Pin NO | Pin Adı   | Özelliği   |
|--------|-----------|--|
| 1      | VSS-GND   | Toprak bağlantısı  |
| 2      | VDD-VCC   | Giriş gerilimi ( 5V )  |
| 3      | VEE       | Potansiyometre girişi – LCD'nin kontrast ayarını yapmak için   |
| 4      | RS        | (Register Select) – Komut register'ı ve veri register'ı<br>arasında geçiş yapmak için kullanılır. RS pini HIGH (1)<br>olduğunda, veri register'ı seçilir ve DO-D7 bacaklarında bulunan<br>sinyaller veri register'ına alınır. RS pini LOW (0) olduğunda,<br>komut register'ı seçilir ve DO-D7 bacaklarındaki sinyaller komut<br>register'ına alınır. |
| 5      | R/W       | Okuma / Yazma modları arasında seçim yapmak için kullanılır.   |
| 6      | E         | ( Enabla Pin ) Register'lara yazmayı sağlayan etkinleştirme pini.  |
| 7-14   | D0-D7     | 8 adet pinden oluşan veri yoludur. LCD'ye bu pinlerden veri ve<br>komutlar gönderilir.   |
| 15     | LEDA-LED+ | Arka plan ışığının giriş gerilim ( 3.3 V)  |
| 16     | LEDK-LED- | Arka plan ışığının toprak bağlantısı   |

# HD44780 LCD- REGISTER SELECT (RS) PIN



#### **Instruction Mode**

When the instruction mode is selected, the data that the LCD receives on the 8 data pins (D0 to D7) will be interpreted as instruction data. These can be instructions such as "clear the display", "move the cursor", etc. Depending on which pins are high and low on the 8 data pins determines the instruction that the LCD will be sent and, thus, execute. For example, when in instruction mode, the LCD will clear the entire screen display when fed the voltages 00000001 on its 8 data pins.

#### **Character Mode**

When the character mode is selected, the data that the LCD receives on the 8 data pins (D0 to D7) will be interpreted as character data. Depending on which pins are high and low on the 8 data pins determines the character(s) which will be shown on the LCD. For example, when in character mode, the LCD will display the "\$" (dollar sign) when the data pins are fed 01000010.

## **INSTRUCTION MODE**

#### Instruction Set for the HD44780 LCD Module

| COMMAND                       | COMMAND CODE  |     |                                  |            |       |  |                                    |      |   |   | E-CYCLE   |             |
|-------------------------------|---|-----|----------------------------------|------------|-------|--|------------------------------------|------|---|---|---|-------------|
| COMMAND                       | RS  | R/W | DB7                              | DB6        | DB5   | DB4  | DB3                                | DB2  | DB1   | B1 DB0  |   | fosc=250KHz |
| SCREEN<br>CLEAR               | 0   | 0   | 0                                | 0          | 0     | 0  | 0                                  | 0    | 0   | 1   | Screen Clear, Set AC to 0<br>Cursor Reposition      | 1.64ms      |
| CURSOR<br>RETURN              | 0   | 0   | 0                                | 0          | 0     | 0  | 0                                  | 0    | 1   | •   | DDRAM AD=0, Return,<br>Content Changeless           | 1.64ms      |
| NPUT SET                      | 0   | 0   | 0                                | 0          | 0     | 0  | 0                                  | 1    | I/D   | s   | Set moving direction of cursor,<br>Appoint if move  | 40us        |
| DISPLAY<br>SWITCH             | 0   | 0   | 0                                | 0          | 0     | 0  | 1                                  | D    | с   | в   | Set display on/off,cursor on/off,<br>blink on/off   | 40us        |
| SHIFT                         | 0   | 0   | 0                                | 0          | 0     | 1  | S/C                                | R/L  |   | •   | Remove cursor and whole<br>display,DDRAM changeless | 40us        |
| FUNCTION<br>SET               | 0   | 0   | 0                                | 0          | 1     | DL   | N                                  | F    | S• (  | •   | Set DL, display line, font                          | 40us        |
| CGRAM<br>AD SET               | 0   | 0   | 0                                | 1          | 1 ACG |  | Set CGRAM AD,<br>send receive data | 40us |   |   |   |             |
| DDRAM<br>AD SET               | 0   | 0   | 1                                | 1 ADD      |       | Set DDRAM AD,<br>send receive data               | 40us                               |      |   |   |   |             |
| BUSY/AD<br>READ CT            | 0   | 1   | BF                               | F AC       |       | Executing internal function,<br>reading AD of CT | 40us                               |      |   |   |   |             |
| CGRAM/<br>DDRAM<br>DATA WRITE | 1   | 0   |                                  | DATA WRITE |       | Write data from CGRAM or<br>DDRAM                | 40us                               |      |   |   |   |             |
| CGRAM/<br>DDRAM<br>DATA READ  | DRAM 1 1 DATA READ  |     | Read data from CGRAM or<br>DDRAM | 40us       |       |  |                                    |      |   |   |   |             |
|                               | I/D=1: Increment Mode; I/D=0: Decrement Mode<br>S=1: Shift<br>S/C=1: Display Shift; S/C=0: Cursor Shift<br>R/L=1: Right Shift; R/L=0: Left Shift<br>DL=1: 8D DL=0: 4D<br>N=1: 2R N=0: 1R<br>F=1: 5x10 Style; F=0: 5x7 Style<br>BF=1: Execute Internal Function;<br>BF=0: Command Received |     |                                  |            |       |  |                                    |      | DDRAM: Display data RAM<br>CGRAM: Character<br>Generator RAM<br>ACG: CGRAM AD<br>ADD: DDRAM AD & Cursor<br>AD<br>AC: Address counter for<br>DDRAM & CGRAM | E-cycle<br>changing<br>with main<br>frequency.<br>Example:<br>If fcp or<br>f <sub>cec</sub> =270KHz<br>40us x<br>250/270<br>=37us |   |             |





# LCD 4bit and 8bit mode



Displaying "Y" on Lcd Binary value= 01011001 Decimal 89

R/W (Read/Write) =0 Write to Lcd Rs (Register Select)=1 Data Register is selected

### Arduino IDE'de Dosya -> Örnekler -> LiquidCrystal

| 👓 sl | ketch_jan21a   A         | rduino 1.8.0     |       |                             |   |                 |
|------|--------------------------|------------------|-------|-----------------------------|---|-----------------|
| Dosy | a Düzenle Tas            | lak Araçlar Ya   | ardım |                             |   |                 |
|      | Yeni                     | Ctrl+N           |       |                             |   |                 |
|      | Aç                       | Ctrl+O           |       |                             |   |                 |
|      | Sonuncuyu aç             |                  | >     |                             |   |                 |
|      | Taslak defteri           |                  | >     |                             |   | 1               |
|      | Örnekler                 |                  | 2     | Δ                           |   |                 |
|      | Kapat                    | Ctrl+W           |       | Dahili Örnekler             |   |                 |
|      | Kaydet                   | Ctrl+S           |       | 01.Basics                   | > |                 |
|      | Farklı Kaydet            | Ctrl+Shift+S     |       | 02.Digital                  | > |                 |
|      | Saufa Avarlari           | Ctrl , Shift , D | -     | 03.Analog                   | > |                 |
|      | Sayra Ayariari<br>Vəzdur | Ctrl i D         |       | 04.Communication            | > |                 |
|      | Yazar                    | Cui+P            | _     | 05.Control                  | > |                 |
|      | Tercihler                | Ctrl+Comma       |       | 06.Sensors                  | > |                 |
|      | Cikis                    | Ctrl+0           |       | 07.Display                  | > |                 |
|      | Υ <sup>IKI</sup> 3       | curre            | -     | 08.Strings                  | > |                 |
|      |                          |                  |       | 09.USB                      | > |                 |
|      |                          |                  |       | 10.StarterKit_BasicKit      | > |                 |
|      |                          |                  |       | 11.ArduinoISP               | > |                 |
|      |                          |                  |       | Examples for any board      |   |                 |
|      |                          |                  |       | Adafruit Circuit Playground | > |                 |
|      |                          |                  |       | Bridge                      | > |                 |
|      |                          |                  |       | Esplora                     | > |                 |
|      |                          |                  |       | Ethernet                    | > |                 |
|      |                          |                  |       | Firmata                     | > |                 |
|      |                          |                  |       | GSM                         | > |                 |
|      |                          |                  |       | LiquidCrystal               | 3 | Autoscroll      |
|      |                          |                  |       | Robot Control               | > | Blink           |
|      |                          |                  |       | Robot Motor                 | > | Cursor          |
|      |                          |                  |       | SD                          | ; | CustomCharacter |
|      |                          |                  |       | Servo                       | , | Display         |
|      |                          |                  |       | SpacebrewYun                | > | HelloWorld      |
|      |                          |                  |       | Stepper                     | > | Scroll          |
|      |                          |                  |       | Temboo                      | 2 | SerialDisplay   |
|      |                          |                  |       | TFT                         | 2 | setCursor       |
|      |                          |                  |       | WiFi                        | 2 | TextDirection   |
|      |                          |                  |       |                             |   |                 |

# LiquidCrystal Kütüphanesi Fonksiyonları

| LiquidCrystal() | begin()             | print()              |
|-----------------|---------------------|----------------------|
| setCursor()     | cursor()            | noCursor()           |
| clear()         | home()              | write()              |
| blink()         | noBlink()           | display()            |
| noDisplay()     | scrollDisplayLeft() | scrollDisplayRight() |
| leftToRight()   | rightToLeft()       | autoScroll()         |
| noAutoScroll()  | createChar()        |                      |

LiquidCrystal kütüphanesinin projeye eklenmesi 📃 < 🏣 🖪 Arduino 1 #include <LiquidCrystal.h>

LiquidCrystal kütüphanesini kullanabilmemiz için yukarıdaki kod satırında olduğu gibi kütüphaneyi projemize eklememiz gerekmektedir. Kütüphanemizi ekledikten sonra yukarıdaki fonksiyonları kullanabiliriz.

# LiquidCrystal ()

Kütüphanemizin yapıcı fonksiyondur. Bu fonksiyon aracılığıyla bir LiquidCrystal nesne oluşturabilir ve bu nesnenin özelliklerine kullanarak bağlantılarını yaptığımız LCD ekranımızın kontrol edebiliriz. Bu yapıcı fonksiyonun 4 farklı kullanım şekli bulunmaktadır.

Kullanım Şekli:

LiquidCrystal(RS, E, D4, D5, D6, D7) LiquidCrystal(RS, RW, E, D4, D5, D6, D7) LiquidCrystal(RS, E, D0, D1, D2, D3, D4, D5, D6, D7) LiquidCrystal(RS, RW, E, D0, D1, D2, D3, D4, D5, D6, D7)

```
LiquidCrystal () - Yapıcı fonksiyon kullanım şekli

1 #include <LiquidCrystal.h>

3 LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
```

# Malzeme Listesi

- Arduino (Arduino Uno)
- Breadboard
- 16×2 LCD Ekran
- > 10 k $\Omega$  potansiyometre
- Erkek-erkek jumper kablo

## 9. UYGULAMA 16X2 LCD (Hello world!)



```
#include <LiquidCrystal.h>
```

```
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
```

```
void setup() {
lcd.begin(16, 2);
lcd.print("hello, world!");
}
void loop() {
```

#### LCD Ekran Görüntüsü



#### Icd.begin()

}

Bu işlev LCD'nin boyutlarını ayarlar. Programın void setup () bölümünde herhangi bir LiquidCrystal işlevinin önüne yerleştirilmelidir. Satır ve sütunların sayısı lcd.begin (sütunlar, satırlar) olarak belirtilir. 16×2 LCD için, lcd.begin (16, 2) kullanmanız gerekiyor ve 20×4 LCD için lcd.begin (20, 4) kullanmanız gerekiyor.

# **Malzeme Listesi**

## 10. Yanıp sönen metin (Hello world!)

 9. Uygulama ile aynı eleman listesine ve devresine sahip

# LCD Ekran Görüntüsü



```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
void setup() {
lcd.begin(16, 2);
}
void loop() {
lcd.print("hello, world!");
delay(500);
lcd.clear();
delay(500);
}
```

#### Icd.clear()

Bu işlev, LCD'de halihazırda görüntülenen herhangi bir metni veya veriyi siler. Icd.print () ile lcd.clear () öğesini ve void loop () bölümünde delay () işlevini kullanırsanız, basit bir yanıp sönen metin programı oluşturabilirsiniz:



## 11. lcd.home ( ) Fonksiyonu

 9. Uygulama ile aynı eleman listesine ve devresine sahip

# LCD Ekran Görüntüsü

| • | •••••• |        | • |
|---|--------|--------|---|
|   | XXXlo, | world! |   |
| • |        |        | • |

### Icd.home()

Bu işlev, imleci ekranın sol üst köşesine yerleştirir ve sonraki metni bu konuma yazdırır. Örneğin, bu uygulamada bu kod "hello, world!" metninin ilk üç harfini X ile değiştirir:

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
void setup() {
lcd.begin(16, 2);
lcd.print("hello, world!");
}
void loop() {
lcd.home();
lcd.print("XXX");
}
```



## 12. lcd.setCursor ( ) Fonksiyonu

 9. Uygulama ile aynı eleman listesine ve devresine sahip

# LCD Ekran Görüntüsü



#### lcd.setCursor()

Benzer, ancak lcd.home () 'dan daha kullanışlıdır. Bu işlev, imleci (ve herhangi bir yazılı metni) ekranın herhangi bir yerine yerleştirir. Programınızın void setup () veya void loop () bölümünde kullanılabilir.

İmleç konumu, lcd.setCursor (sütun, satır) ile tanımlanır. Sütun ve satır koordinatları sıfırdan başlar (sırasıyla 0-15 ve 0-1). Örneğin, yukarıdaki "hello, world!" programının void setup () bölümünde lcd.setCursor (2, 1) kullanılarak, "hello, world!" alt satıra yazdırılır ve sağa iki alan kaydırılır.

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
void setup() {
lcd.begin(16, 2);
lcd.setCursor(2, 1);
lcd.print("hello, world!");
}
void loop() {
}
```

### lcd.write ()

LCD'ye farklı tipte veri yazmak için bu işlevi kullanabilirsiniz, örneğin bir sıcaklık sensöründen okuma veya bir GPS modülünden gelen koordinatlar. Kendiniz oluşturduğunuz özel karakterleri yazdırmak için de kullanabilirsiniz. Programınızın void setup () veya void loop () bölümünde lcd.write () öğesini kullanabilirsiniz. **Icd.print ()** 

Bu işlev, LCD'ye metin yazdırmak için kullanılır. Programın void setup () bölümünde veya void loop () bölümünde kullanılabilir.

Harfleri ve sözcükleri yazdırmak için, metnin etrafına tırnak işaretleri ("") yerleştirin. Örneğin; hello, world!'ü yazdırmak için yukarıda lcd.print ("hello,world!") kullanmıştık.

Sayıları yazdırmak için tırnak işaretleri gerekmez. Örneğin; 123456789'u yazdırmak için lcd.print (123456789) yazabilirsiniz.

lcd.print (); sayıları ondalık, ikili, onaltılık ve sekizlik bazlarda yazdırabilir. Örneğin:

Icd.print (100, DEC) "100" yazar. Icd.print (100, BIN) "1100100" yazar. Icd.print (100, HEX) "64" yazar. Icd.print (100, OCT) "144" yazar.

# **Malzeme Listesi**

 9. Uygulama ile aynı eleman listesine ve devresine sahip

## 13. Farklı tipte sayı gösterimi

## LCD Ekran Görüntüsü



```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
void setup() {
lcd.begin(16, 2);
}
void loop() {
lcd.home();
lcd.print ("DECIMAL=");
lcd.print (204, DEC); // "204" yazar.
lcd.setCursor (0,1);
lcd.print ("BINARY=");
lcd.print (204, BIN); // "11001100" yazar.
delay(2000);
lcd.clear();
lcd.setCursor (0,0);
lcd.print ("HEX=");
lcd.print (204, HEX); // "CC" yazar.
lcd.setCursor (0,1);
lcd.print ("OCT=");
lcd.print (204, OCT); // "314" yazar.
delay(2000);
lcd.clear();
}
```



 9. Uygulama ile aynı eleman listesine ve devresine sahip

## 14. lcd.Cursor () Ve lcd.noCursor () Fonksiyonu

# LCD Ekran Görüntüsü



#### Icd.Cursor( )

Bu işlev görünür bir imleç oluşturur. İmleç, LCD'ye yazdırılacak bir sonraki karakterin altına yerleştirilmiş yatay bir çizgidir.

**Icd.noCursor ()** işlevi imleci kapatır. Icd.cursor () ve Icd.noCursor () birçok metin giriş alanında gördüğünüze benzer yanıp sönen imleci yazdırmak için void loop () bölümünde birlikte kullanılabilir.

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
void setup() {
lcd.begin(16, 2);
lcd.print("hello, world!");
}
void loop() {
lcd.cursor();
delay(500);
lcd.noCursor();
delay(500);
}
```

# **Malzeme Listesi**

 9. Uygulama ile aynı eleman listesine ve devresine sahip 15. Lcd cursor'ü ekrandan herhangi bir yere yerleştirmek

# LCD Ekran Görüntüsü



```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
void setup() {
lcd.begin(16, 2);
lcd.print("hello, world!");
}
void loop() {
lcd.setCursor(12, 1);
lcd.cursor();
delay(500);
lcd.setCursor(12, 1);
lcd.noCursor();
delay(500);
}
```



# Malzeme Listesi

 9. Uygulama ile aynı eleman listesine ve devresine sahip

## o16. lcd.blink () ve lcd.noBlink () Fonksiyonu

## Temsili LCD Ekran Görüntüsü



### lcd.blink()

Bu işlev, döngü başına yaklaşık 500 milisaniyede bir yanıp sönen blok stili bir imleç oluşturur. Bunu void loop () bölümünde kullanın.

Icd.noBlink () işlevi yanıp sönen blok imlecini devre dışı bırakır.

```
#include <LiquidCrystal.h>
```

```
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
```

```
void setup() {
lcd.begin(16, 2);
lcd.print("hello, world!");
}
```

```
void loop() {
 lcd.blink();
 delay(500);
 lcd.noBlink();
 delay(500);
}
```



# <u>Malzeme Listesi</u>

 9. Uygulama ile aynı eleman listesine ve devresine sahip

## o17. lcd.display () ve lcd.noDisplay () Fonksiyonu

#### lcd.display()

Bu işlev, LCD ekrana yazdırılmış herhangi bir metin veya imleci açar. **Icd.noDisplay** () işlevi, LCD'nin belleğinden silmeden LCD'ye yazdırılan herhangi bir metni veya imleci kapatır.

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
void setup() {
lcd.begin(16, 2);
lcd.print("hello, world!");
}
void loop() {
lcd.display();
delay(500);
lcd.noDisplay();
delay(500);
}
```



### lcd.scrollDisplayLeft()

Bu işlev, LCD'ye yazdırılan her şeyi alır ve sola taşır. Bunu takiben bir gecikme komutu ile void loop () bölümünde kullanılmalıdır. İşlev, ilk karaktere geri dönmeden önce metni 40 alan sola taşır. Bu kod, "hello, world!" metnini, karakter başına bir saniyede olmak üzere sola taşır:

### lcd.scrollDisplayRight( )

Bu işlev, lcd.scrollDisplayLeft () gibi davranır ancak metni sağa taşır.

```
#include <LiquidCrystal.h>
```

```
void setup() {
lcd.begin(16, 2);
lcd.print("hello, world!");
}
```

```
void loop() {
lcd.scrollDisplayLeft();
delay(1000);
}
```

```
Sola kaydırma
```

```
#include <LiquidCrystal.h>
```

```
LiquidCrystal lcd(12, 11, 5, 4, 3, LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
```

```
void setup() {
lcd.begin(16, 2);
lcd.print("hello, world!");
}
```

```
void loop() {
lcd.scrollDisplayRight();
delay(1000);
}
```

### Sağa kaydırma



#### lcd.autoscroll( )

Bu işlev bir metin dizisi alır ve dizinin karakter sayısının artışlarıyla sağdan sola kaydırır. Örneğin; 3 karakter uzunluğunda bir metin diziniz varsa, her adımda metni 3 alan sola kaydırır. Icd.scrollDisplay () işlevleri gibi; metin, tekrarlanmadan önce en fazla 40 karakter uzunluğunda olabilir.

### lcd.noAutoscroll( )

Icd.noAutoscroll (), Icd.autoscroll () işlevini kapatır. Kayan metin veya animasyon dizileri oluşturmak için void loop () bölümünde Icd.autoscroll () işlevinden önce veya sonra bu işlevi kullanabilirsiniz.

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
void setup() {
lcd.begin(16, 2);
}
void loop() {
lcd.setCursor(0, 0);
lcd.autoscroll();
lcd.print("ABC");
delay(500);
}
```





## 20. lcd.rightToLeft( ) Fonksiyonu

 9. Uygulama ile aynı eleman listesine ve devresine sahip

# LCD Ekran Görüntüsü



### lcd.rightToLeft()

Bu işlev, metnin ekrana yazdırıldığı yönü ayarlar. Varsayılan mod, lcd.leftToRight () komutunu kullanarak soldan sağa doğrudur, ancak metnin ters yönde yazılışının yararlı olduğu bazı durumlar bulabilirsiniz:

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
void setup() {
lcd.begin(16, 2);
lcd.setCursor(12, 0);
lcd.rightToLeft();
lcd.print("hello, world!");
}
void loop() {
}
```



## 20. Özel karakter Kullanımı

 9. Uygulama ile aynı eleman listesine ve devresine sahip



### lcd.createChar()

Bu komut kendi özel karakterlerinizi oluşturmanıza izin verir. 16×2 LCD'nin her bir karakterinin 5 piksel genişliği ve 8 piksel yüksekliği vardır. Tek bir programda 8 farklı özel karakter tanımlanabilir. Kendi karakterlerinizi tasarlamak için, özel karakterinizin ikili karakter matrisini <u>LCD karakter jeneratörü</u>nden yapmanız veya kendiniz haritalamanız gerekir. LCD hafızasında ASCII karakterler tanımlı olduğu için Türkçe karakterler bu listede bulunmamaktadır. Türkçe karakter yazdırmaya çalışırsanız yanlış şekiller ekrana gelir. Türkçe karakterleri özel olarak ayrılmış LCD cgram'e kaydedip, gerektiğinde oradan çekebiliriz. Ancak özel karakter hafızası <u>8</u> karakter ile sınırlıdır. 9. karakteri kaydetmeye çalışırsanız 1. karakterin üzerine yazacaktır.

| 5x7 matis | ikilik değer (binary) | onluk (decimail) |
|-----------|-----------------------|------------------|
|           | 00100                 | 4                |
|           | 00000                 | 0                |
|           | 01110                 | 14               |
|           | 00100                 | 4                |
|           | 00100                 | 4                |
|           | 00100                 | 4                |
|           | 01110                 | 14               |

LCD de her karakterin yerleşebileceği 5×8 birimlik toplam 40 adet nokta vardır. Resmi inceleyecek olursak "i" harfinin yerleşmesi için hangi noktaların açık, hangi noktaların kapalı olduğu belirtilmiştir.

| 1  | <pre>#include <liquidcrystal.h></liquidcrystal.h></pre>                                   |
|----|---|
| 2  | LiquidCrystal lcd(9, 8, 3, 2, 1, 0);  |
| 3  | byte karakter0[8] = { 0x00, 0x0E, 0x11, 0x10, 0x11, 0x0E, 0x04, 0x00 }; // c              |
| 4  | byte karakter1[8] = { 0x0C, 0x12, 0x12, 0x0C, 0x00, 0x00, 0x00, 0x00 }; // Derece         |
| 5  | byte karakter2[8] = { 0x0A, 0x00, 0x0E, 0x11, 0x11, 0x11, 0x0E, 0x00 }; // ö              |
| 6  | byte karakter3[8] = { 0x1F, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x1F }; // kare           |
| 7  | byte karakter4[8] = { 0x00, 0x00, 0x0A, 0x1F, 0x1F, 0x0E, 0x04, 0x00 }; // kalp           |
| 8  | byte karakter5[8] = { 0x04, 0x0E, 0x15, 0x04, 0x04, 0x04, 0x04, 0x00 }; // yukarı ok      |
| 9  | byte karakter6[8] = { 0x00, 0x00, 0x0A, 0x00, 0x11, 0x0E, 0x00, 0x00 }; // gülen surat    |
| 10 | byte karakter7[8] = { 0x11, 0x0E, 0x0A, 0x0A, 0x0A, 0x0A, 0x0E, 0x11 }; // anlamsız şekil |
| 11 | void setup()  |
| 12 | {   |
| 13 | lcd.begin(16, 2);   |
| 14 | lcd.createChar(0, karakter0);   |
| 15 | lcd.createChar(1, karakter1);   |
| 16 | lcd.createChar(2, karakter2);   |
| 17 | lcd.createChar( <mark>3</mark> , karakter3);  |
| 18 | lcd.createChar(4, karakter4);   |
| 19 | lcd.createChar( <mark>5</mark> , karakter5);  |
| 20 | lcd.createChar( <mark>6</mark> , karakter6);  |
| 21 | lcd.createChar(7, karakter7);   |
| 22 | <pre>lcd.clear();</pre>   |
| 23 | }   |
| 24 | void loop()   |
| 25 | {   |
| 26 | for (int i = 0; i < 8; i++)   |
| 27 | {   |
| 28 | <pre>lcd.write(byte(i));</pre>  |
| 29 | }   |
| 30 | while (true)  |
| 31 | {   |
| 32 | }   |
| 33 | }   |

# Thank you for listening

