

معرفی و راه‌اندازی ماژول نمایشگر LCD با آردوینو



این مطلب بخش هشتم از [آموزش جامع آردوینو \(مبتدی و پیشرفته\)](#) است. در این آموزش نحوه کار با LCD کاراکتری را خواهید آموخت. در مطالب قبل برای نشان دادن یک متن یا عدد (مثلا مقدار خروجی یک سنسور یا زاویه یک موتور) از سریال مانیتور استفاده کردیم که در دسترس‌ترین انتخاب برای نمایش است. اما در پروژه‌های واقعی همیشه دسترسی به رایانه وجود ندارد و معمولا علاقه داریم که سیستم ما به صورت مستقل از رایانه بتواند کار مورد نظر را انجام دهد. در این موارد، اولین انتخاب LCD کاراکتری است. استفاده از LCD از آنجا که امکان اجرای پروژه‌ها را بدون رایانه فراهم می‌کند، یک نقطه مهم در انجام پروژه‌های هوشمند است.

قطعات مورد نیاز:

1 عدد

[برد آردوینو و کابل رابط](#)

1 عدد

[نمایشگر LCD کاراکتری 2x16](#)

1 عدد

[سنسور دما و رطوبت DHT11](#)

2 عدد

[پتانسیومتر 10k](#)

1 عدد

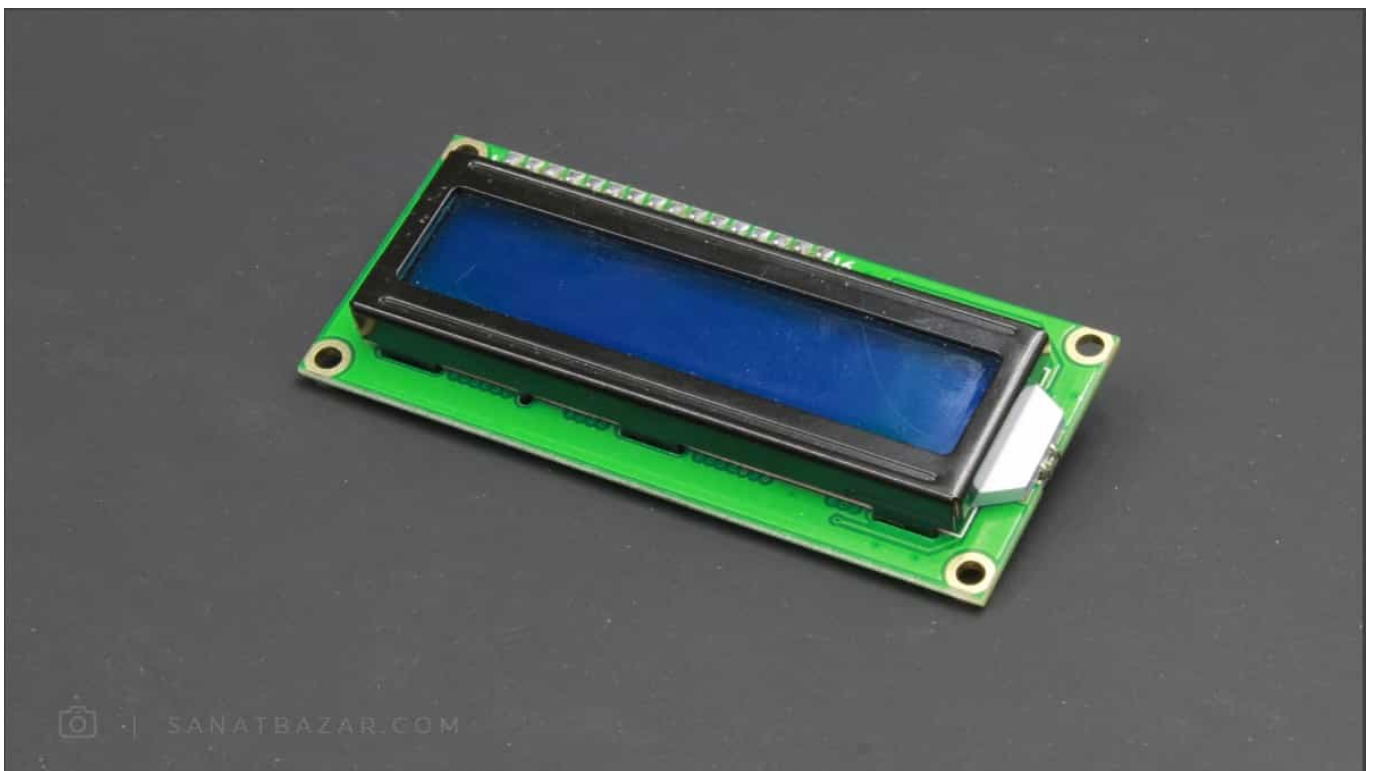
بازدید

1 بسته

سیم جامپر

معرفی LCD

درون LCD کاراکتری یک صفحه شامل کریستال مایع وجود دارد که به ولتاژ الکتریکی حساس بوده و با اعمال ولتاژ از خود نور ساطع می‌کند. صفحه کریستال مایع بین دو صفحه از جنس شیشه قرار دارد که آن را در جای خود نگه داشته و از آن محافظت می‌کنند. دلیل اینکه به این نوع LCD ، کاراکتری می‌گویند این است که صفحه آن به تعدادی بخش‌های مربعی تقسیم شده است و به صورت پیش‌فرض هر کدام از این بخش‌ها یک کاراکتر (شامل عدد، حرف، علامت) را نشان می‌دهد. LCD ها با تعداد کاراکترهایی که نمایش می‌دهند مشخص می‌شوند؛ مثلا نمونه بسیار معروفی که در این آموزش هم ما از آن استفاده کرده‌ایم، مدل 2x16 است به این معنا که صفحه LCD دارای 16 ستون و 2 ردیف است.



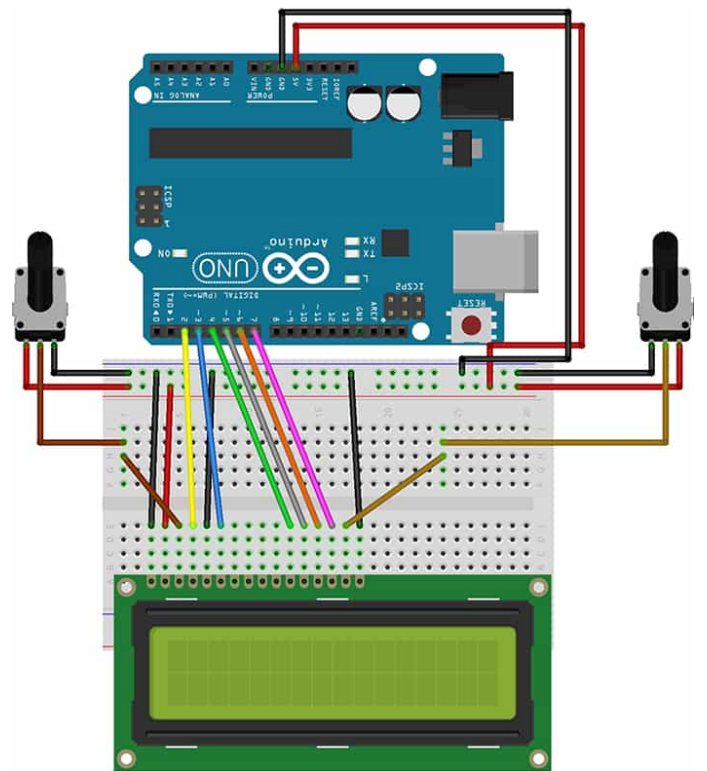
LCD کاراکتری نسبت به اکثر ماژول‌ها دارای تعداد زیادی پایه است (16 تا)؛ اما نگران نباشید چون لازم نیست از همه آنها استفاده کنید! پایه‌های D0 تا D7 (8 بیت) برای انتقال داده کاراکترها به صورت سریال استفاده می‌شود. با این وجود می‌توانید بجای 8 بیت، داده‌ها را به صورت 4 بیتی (با استفاده از 4 پایه D4 تا D7) بفرستید. این کار سرعت انتقال را کمتر می‌کند؛ البته معمولا نه آنقدر که قابل تشخیص باشد، پس با خیال راحت داده‌ها را به صورت 4 بیتی انتقال دهید. پایه‌های شماره 1 (VSS) و 2 (VDD) به ترتیب منفی و مثبت LCD اند. پایه RS وضعیت LCD را بین دو حالت دستور (Instruction) و داده (Data) و پایه RW وضعیت LCD را بین دو حالت خواندن (Read) و نوشتن (Write) انتخاب می‌کند. همچنین پایه E (Enable) برای انتقال اطلاعات از حافظه LCD به نمایشگر استفاده می‌شود. خوشبختانه با استفاده از کتابخانه LiquidCrystal می‌توانید کار با وضعیت پایه‌ها را به نرم‌افزار واگذار کنید. این کتابخانه به صورت پیش‌فرض در نرم‌افزار آردوینو وجود دارد. دو بخش مهم دیگر در پایه‌های LCD وجود دارد: تنظیم نور نوشته‌ها (V0) و تنظیم نور پس‌زمینه (A, K). می‌توانید پایه‌های V0 و A را با یک مقاومت به +5V وصل کنید اما پیشنهاد می‌کنم برای اینکه خیلی راحت بتوانید روشنایی صفحه را به میزان دلخواه تنظیم کنید، از دو پتانسیومتر استفاده کنید؛ یکی برای پایه V0 و دیگری برای پایه A. به این ترتیب می‌توانید با تغییر مقدار مقاومت پتانسیومتر، صفحه نمایشگر را به حالتی که وضوح دلخواهتان را داشته باشد برسانید.



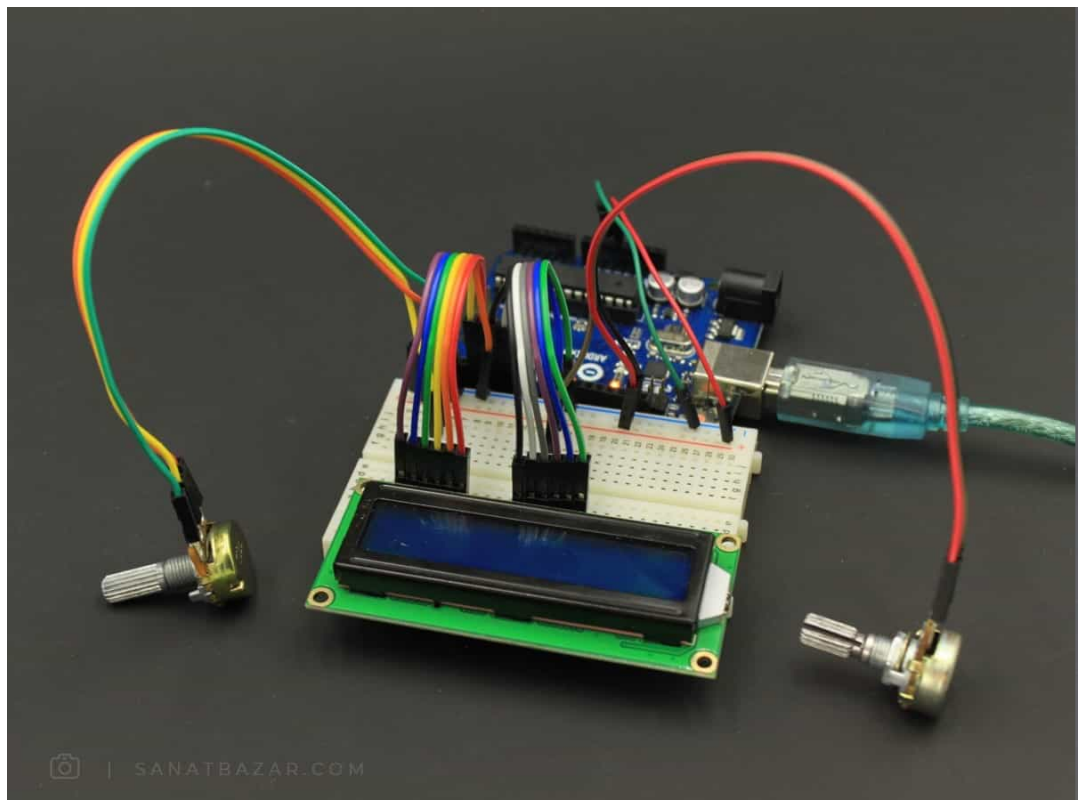
راه اندازی LCD

برای شروع باید پایه‌های LCD را به برد وصل کنید. برای راحتی کار می‌توانید ابتدا آنرا به یک برد متصل کرده و سپس از برد آردوینو سیم‌کشی کنید. پایه‌ها را مطابق جدول زیر به آردوینو وصل کنید.

محل اتصال	وظیفه	نام پین	شماره پین LCD
GND	قطب منفی	VSS	1
5V+	قطب مثبت	VCC	2
پتانسیومتر شماره 1	تنظیم میزان روشنایی کاراکترها	VEE	3
پایه 2 دیجیتال	Register select : Instruction register = 0 Data register = 1	RS	4
GND	Read / Write Write mode = 0 Read = 1	RW	5
پایه 3 دیجیتال	Enable Latch data to LCD = 0 Disable = 1	E	6
-	Data bit 0	D0	7
-	Data bit 1	D1	8
-	Data bit 2	D2	9
-	Data bit 3	D3	10
پایه 4 دیجیتال	Data bit 4	D4	11
پایه 5 دیجیتال	Data bit 5	D5	12
پایه 6 دیجیتال	Data bit 6	D6	13
پایه 7 دیجیتال	Data bit 7	D7	14
پتانسیومتر شماره 2	تنظیم میزان روشنایی نور پس‌زمینه	A	15
GND	قطب منفی روشنایی پس‌زمینه	K	16



fritzing



حالا LCD آماده انجام کارهای جذاب است. از اینجا به بعد همه چیز به خلاقیت شما وابسته است. در این قسمت قصد داریم برنامه‌ای بنویسیم که در سطر اول LCD عبارت Hello world! را نوشته و در سطر دوم زمانی که از ابتدای شروع فرمان گذشته را بر حسب دقیقه و ثانیه مانند یک ساعت دیجیتال نشان دهد. قبل از اینکه نرم‌افزار را اجرا کنید، بد نیست با چند دستور و تابع مهم در آردوینو آشنا شوید. برای استفاده از LCD باید از کتابخانه LiquidCrystal استفاده کنید. این کتابخانه به صورت پیش‌فرض در نرم‌افزار آردوینو موجود است. این کتابخانه را در ابتدای کدتان به صورت زیر فراخوانی کنید:

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

همچنین پس از تعریف کتابخانه باید یک شیء از جنس LCD تعریف شود. مثلا:

```
LiquidCrystal lcd(RS, E, D4, D5, D6, D7)
```

که ورودی‌های آن عدد پین‌هایی است که به پایه‌های LCD که در دستور فوق مشخص شده است وصل شده‌اند. همان طور که می‌دانید پس از تعریف شیء، توابع داخلی آنرا می‌توانید به صورت زیر فراخوانی کنید:

```
lcd.function()
```

در ادامه برای سادگی، از دستورات داخلی LCD بدون پیش‌وند lcd نام می‌بریم.

دستور millis() مثل یک ساعت است که از زمانی که فراخوانی می‌شود شروع به اندازه‌گیری زمان بر حسب میلی‌ثانیه می‌کند. دستور مشابه دیگری نیز وجود دارد به نام micros() که زمان را بر حسب میکروثانیه اندازه می‌گیرد. همچنین عملگر % باقیمانده تقسیم یک عدد بر عدد دیگر را می‌دهد. مثلا اگر زمان را بر حسب ثانیه داشته باشید و آنرا با n نشان دهید، عبارت n/60 تعداد دقیقه و n%60 تعداد ثانیه‌های ساعت دیجیتال ما را محاسبه خواهد کرد. ابتدا سعی کنید خودتان برنامه این پروژه را بنویسید و بعد برنامه زیر را اجرا کنید. برنامه پیشنهادی ما به صورت زیر است. برنامه را بارگذاری کنید و از آن لذت ببرید.

```
/*
SanatBazar
Arduino Tutorial Series
Author: Davood Dorostkar
Website: www.sanatbazar.com

LCD Connections:
RS --> 2          RW --> GND
E --> 3          VSS --> GND
D4 --> 4         VDD --> +5V
D5 --> 5         V0 --> Potentiometer #1
D6 --> 6         A --> Potentiometer #2
D7 --> 7         K --> GND

*/

#include <LiquidCrystal.h>
#define RS 2
#define E 3
#define D4 4
#define D5 5
#define D6 6
#define D7 7
LiquidCrystal lcd(RS, E, D4, D5, D6, D7);

void setup()
{
  lcd.begin(16, 2);
}

void loop()
{
  lcd.clear();
  lcd.print("Hello world!");
  int minute = millis() / 1000 / 60;
  lcd.setCursor(0, 1);
  if (minute < 10)
    lcd.print("0");
  lcd.print(minute);
  lcd.setCursor(2, 1);
  lcd.print(":");
  int second = (millis() / 1000) % 60;
  lcd.setCursor(3, 1);
  if (second < 10)
    lcd.print("0");
```

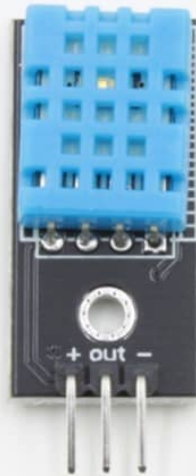
```
lcd.print(second);  
delay(500);  
}
```



نمایش دما و رطوبت در LCD

معمولا هدف از استفاده از LCD ترکیب شدن با اجزای دیگر یک سیستم است تا وضعیت سنسورها، محل موتورها و غیره را نشان دهد. در این قسمت قصد داریم که یک نمونه کوچک از این کار را انجام دهیم. می‌خواهیم سنسور دما و رطوبت را مانند آنچه که در قسمت سوم آموزش آردوینو توضیح دادیم راهاندازی کرده و این بار دما و رطوبت را بجای رایانه در LCD نمایش دهیم. با این کار می‌توانید دستگاه هواشناسی‌تان را ساخته و در محلی که دوست دارید، بدون اینکه نیازی به رایانه باشد نصب کنید.

در اینجا ما برای اندازه‌گیری دما و رطوبت از ماژول DHT11 استفاده کرده‌ایم. این ماژول سه پایه دارد: پایه‌های مثبت و منفی که سمت چپ و راست آن قرار دارند و به +5V و GND آردوینو وصل می‌شوند و پایه وسط که پایه داده سنسور است. این پایه را به پین ۸ دیجیتال آردوینو وصل کنید.



 | SANATBAZAR.COM

اتصالات LCD نیازی به تغییر ندارد. حالا می‌توانید برنامه زیر را در آردوینو بارگذاری کرده و نتیجه را در نمایشگر ببینید.

```

/*
SanatBazar
Arduino Tutorial Series
Author: Davood Dorostkar
Website: www.sanatbazar.com

LCD Connections:
RS --> 2           RW --> GND
E --> 3           VSS --> GND
D4 --> 4          VDD --> +5V
D5 --> 5          V0 --> Potentiometer #1
D6 --> 6          A --> Potentiometer #2
D7 --> 7          K --> GND

*/

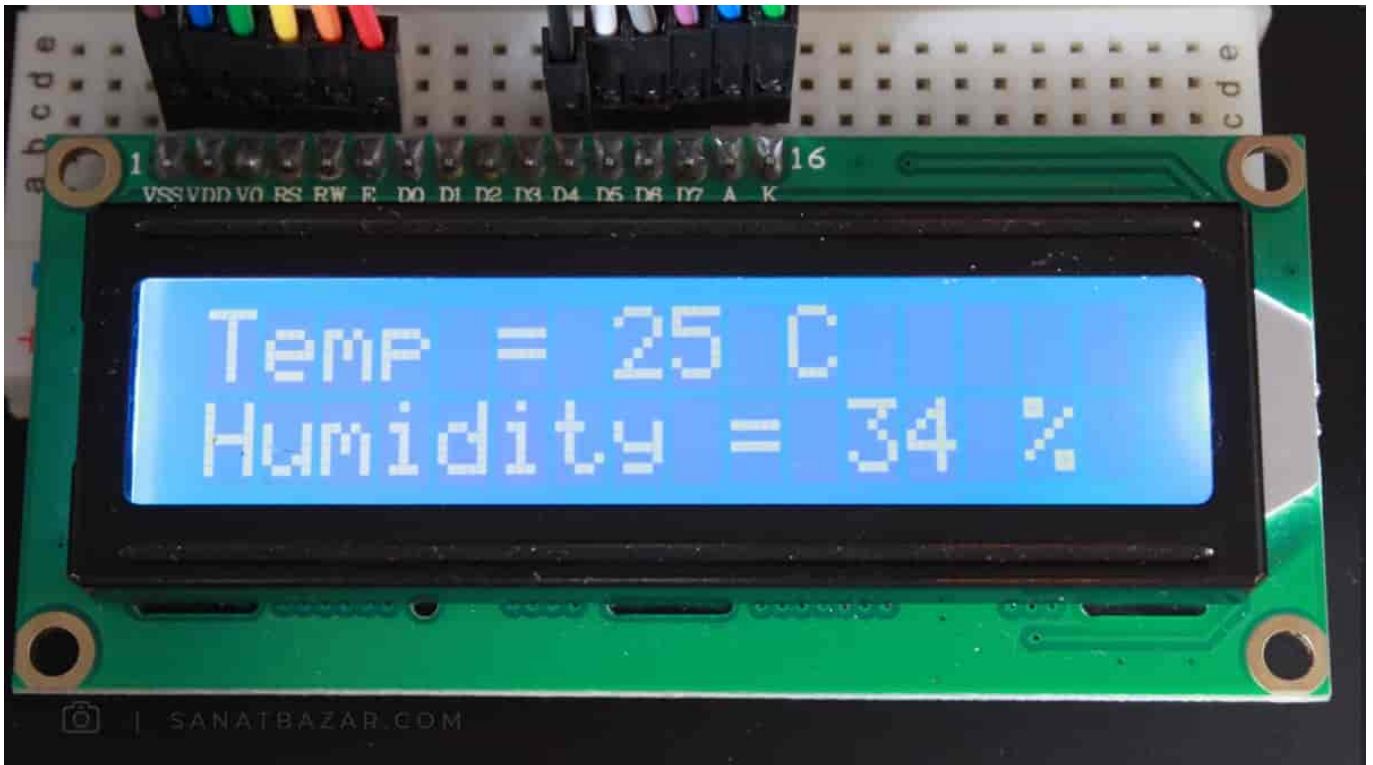
#include <dht.h>
dht DHT;
#define DHT11_PIN 8

#include <LiquidCrystal.h>
#define RS 2
#define E 3
#define D4 4
#define D5 5
#define D6 6
#define D7 7
LiquidCrystal lcd(RS,E,D4,D5,D6,D7);

void setup(){
  lcd.begin(16, 2);
}
void loop()
{
  int t = DHT.read11(DHT11_PIN);
  lcd.setCursor(0,0);
  lcd.print("Temp = ");
  lcd.print((int)DHT.temperature);
  lcd.print(" C");
  lcd.setCursor(0,1);
  lcd.print("Humidity = ");
  lcd.print((int)DHT.humidity);
  lcd.print(" %");
  delay(2000);
}

```

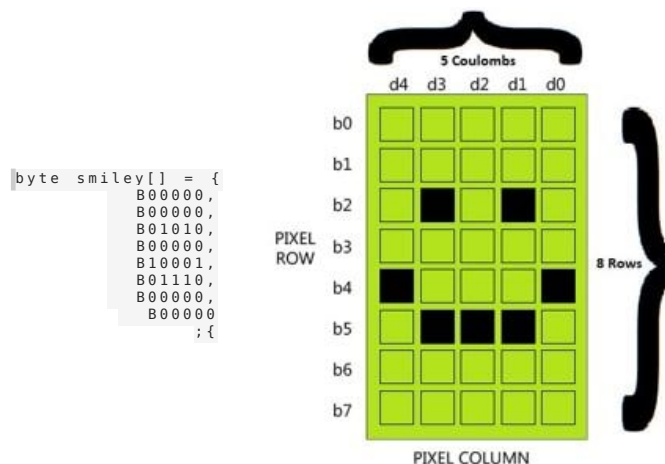
}



آردوینو را از رایانه جدا کرده و با آداپتور به برق شهری یا باتری وصل کنید. در صورت نیاز می‌توانید با کلید ریست آردوینو برنامه را مجدداً فراخوانی کنید.

نمایش کاراکتر شخصی با LCD

اگر از نزدیک به LCD نگاه کنید متوجه می‌شوید که هر کدام از کاراکترهای آن به صورت پیوسته نمایش داده نمی‌شود و خود به تعدادی پیکسل جدا از هم تقسیم شده است. به طور معمول هر کاراکتر با استفاده از یک ماتریس از پیکسل‌ها شامل ۸ سطر و ۵ ستون ساخته می‌شود. برد الکترونیکی که در قسمت پشت LCD قرار دارد وظیفه دارد تا دستوراتی که به صورت کاراکتر به آن ارسال می‌شود را به ماتریسی از پیکسل‌ها تبدیل کند و بر روی صفحه نمایش دهد. شاید تا اینجا حدس زده باشید که احتمالاً بتوانید دستور نمایش کاراکترها را شخصی‌سازی کنید. حدستان درست است! این یکی از جذابیت‌های دنیای الکترونیک است که تقریباً هر چیزی در دسترس و قابل برنامه‌ریزی است. شما می‌توانید کاراکتر جدیدی تعریف کرده و در LCD نمایش دهید. برای تولید هر کاراکتر باید ابتدا یک آرایه ۸ تایی (به تعداد سطرهای پیکسلی) ایجاد کنید که هر درایه آن می‌تواند یک عدد از ۰ تا ۳۲ باشد (تعداد ستون‌های کاراکتر ۵ تا است به این معنی که هر ستون شامل ۲۵ یا ۳۲ حالت است). تعریف این عدد می‌تواند در مبنای ۱۰ (مثلاً ۱۳)، در مبنای ۲ یا Binary (مثلاً B01101) یا در مبنای ۱۶ یا hexadecimal (مثلاً 0x0D) نوشته شود. برای مثال تعریف شکلک لبخند در یک کاراکتر می‌تواند به صورت زیر باشد:



به شکل بالا نگاه کنید. در این شکل سطر اول و دوم همه پیکسل‌ها خاموش‌اند، پس دو درایه اول کاراکتر به صورت B00000 هستند. همچنین در سطر سوم پیکسل دوم و چهارم روشن‌اند پس درایه سوم کاراکتر به صورت B01010 نوشته شده که معادل عدد ۱۰ است.

پس از تعریف کاراکترهای شخصی باید با استفاده از دستور `createChar(n,name)` آنرا برای LCD تعریف کنید. ورودی اول شماره کاراکتر و ورودی دوم نام کاراکتری است که قبلاً ساخته‌اید. پس از این کار در طول برنامه، کاراکتر با شماره آن فراخوانی می‌شود. با استفاده از کتابخانه `LiquidCrystal` می‌توانید حداکثر تا ۸ کاراکتر شخصی بسازید. برای نمایش کاراکترهای خاص از دستور `write(byte(n))` استفاده می‌شود که در آن `n` شماره کاراکتر تعریف شده است.

برنامه زیر را بر روی آردوینو بارگذاری کنید. در این برنامه از چند دستور خاص در کتابخانه `LiquidCrystal` استفاده شده است: دستور `setCursor(n,m)` که نشانگر را برای نوشتن کاراکتر جدید به محل مورد نظر شما می‌برد و دستور `clear()` که کل صفحه نمایشگر را پاک می‌کند. برای اینکه روش ایجاد کاراکتر را بهتر متوجه شوید، شکل‌های لبخند، ساعت شنی و چند شکل دیگر (برنامه را اجرا کنید تا شکل آنها را ببینید!) در میناهای ۲، ۱۰ و ۱۶ تعریف شده‌اند. با اجرای برنامه، کاراکترها به صورت پشت سر هم ظاهر می‌شوند تا اینکه صفحه پر شود.

```

/*
SanatBazar
Arduino Tutorial Series
Author: Davood Dorostkar
Website: www.sanatbazar.com

LCD Connections:
RS --> 2           RW --> GND
E --> 3           VSS --> GND
D4 --> 4          VDD --> +5V
D5 --> 5          V0 --> Potentiometer #1
D6 --> 6          A --> Potentiometer #2
D7 --> 7          K --> GND

*/

#include <LiquidCrystal.h>
#define RS 2
#define E 3
#define D4 4
#define D5 5
#define D6 6
#define D7 7
LiquidCrystal lcd(RS,E,D4,D5,D6,D7);

// Definitions of characters:

byte smiley[] = {
  B00000,
  B00000,
  B01010,
  B00000,
  B10001,
  B01110,
  B00000,
  B00000
};
byte diamond[] = {
  0,
  4,
  10,
  17,
  10,
  4,
  0,
  0
};
byte arrow[] = {
  B00100,
  B01110,
  B11111,
  B00100,
  B00100,
  B00100,
  B00100,
  B00100
};
byte newShape[] = {
  B11111,
  B10001,
  B10001,
  B00100,
  B00100,

```

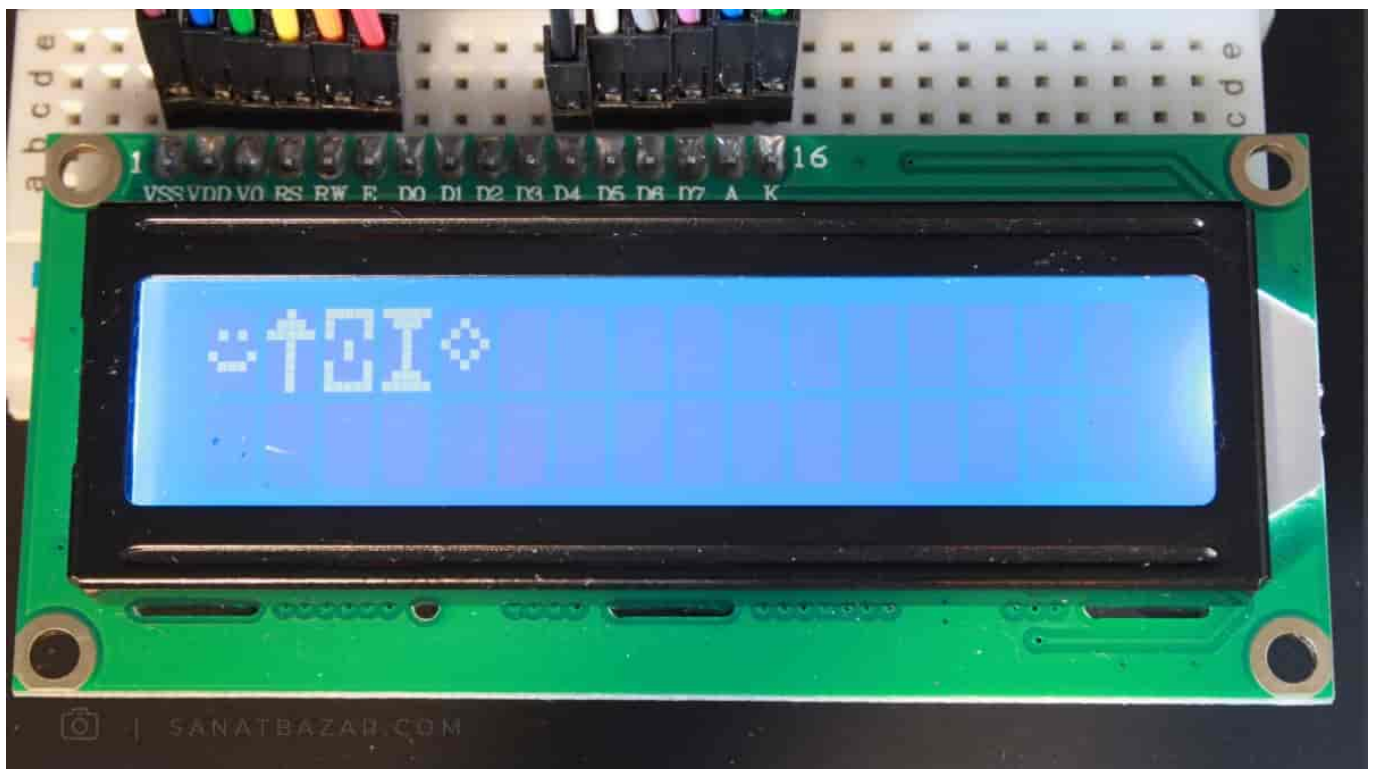
```

    B10001,
    B10001,
    B11111
};
byte sandglass[] = {
    0x1F,
    0x0E,
    0x04,
    0x04,
    0x04,
    0x04,
    0x04,
    0x0E,
    0x1F
};

void setup() {
    lcd.createChar(0, smiley);
    lcd.createChar(1, arrow);
    lcd.createChar(2, newShape);
    lcd.createChar(3, sandglass);
    lcd.createChar(4, diamond);
    lcd.begin(16, 2);
}

void loop() {
    int numberOfCharacters=0;
    lcd.clear();
    lcd.setCursor(0,0);
    for (int i=0; i<32; i++){
        if (numberOfCharacters==16) {
            lcd.setCursor(0,1);
        }
        lcd.write(byte(i%5));
        numberOfCharacters++;
        delay(1000);
    }
}

```



نتیجه‌گیری

در این آموزش نحوه کار با LCD کاراکتری را یاد گرفتید و به اهمیت آن پی بردید. از LCD کاراکتری در ساخت دستگاه‌هایی مانند پرینتر سه‌بعدی هایپرکیوب استفاده شده است.

در آموزش بعدی، [نحوه راه‌اندازی و کار با انواع LED](#) را خواهید آموخت. نظرات شما باعث بهبود محتوای آموزشی ما می‌شود. اگر این آموزش را دوست داشتید، همین‌طور اگر سوالی در مورد آن دارید، از شنیدن نظراتتان خوشحال خواهیم شد.