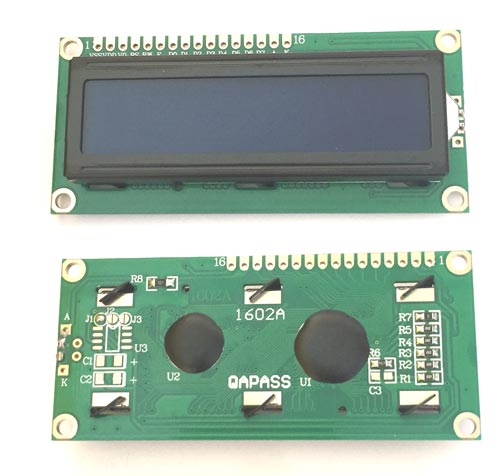
**16X2 LCD**

**Introduction**

The term LCD stands for liquid crystal display. It is one kind of electronic display module used in an extensive range of applications like various circuits & devices.

An LCD screen is an electronic display module that uses liquid crystal to produce a visible image. The 16×2 LCD display is a very basic module commonly used in [DIYs](https://www.electronicsforu.com/category/electronics-projects/hardware-diy) and circuits. The 16×2 translates a display 16 characters per line in 2 such lines. In this LCD each character is displayed in a 5×7-pixel matrix.



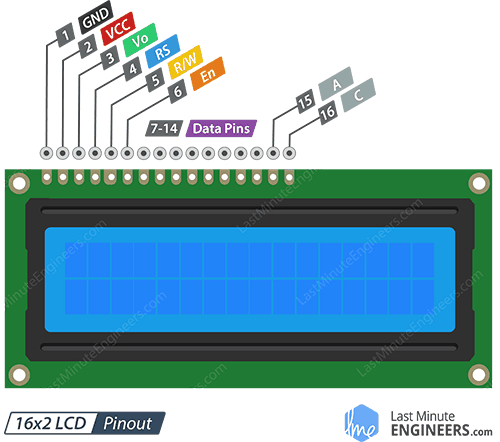
**Hardware Overview of 16X2 LCD**

A 16×2-character LCD, for example, has an LED backlight and can display 32 ASCII characters in two rows of 16 characters each. Each rectangle contains grid of 5×8 pixels.

If you look closely, you can see tiny rectangles for each character on the display and the pixels that make up a character. Each of these rectangles is a grid of 5×8 pixels.

They come in many sizes and colours: for example, 16×1, 16×4, 20×4, white text on a blue background or black text on a green and many others.

**Pinout of 16X2 LCD**



GND is connected to the ground of the Arduino.

VCC is the power supply for the LCD which we connect to the 5V pin on the Arduino.

Vo (LCD Contrast) controls the contrast and brightness of the LCD. Using a simple voltage divider with a potentiometer, we can make fine adjustments to the contrast.

RS (Register Select) pin is set to LOW when sending commands to the LCD (such as setting the cursor to a specific location, clearing the display, etc.) and HIGH when sending data to the LCD. Basically, this pin is used to separate the command from the data.

R/W (Read/Write) pin allows you to read data from the LCD or write data to the LCD. Since we are only using this LCD as an output device, we are going to set this pin LOW. This forces it into WRITE mode.

E (Enable) pin is used to enable the display. When this pin is set to LOW, the LCD does not care what is happening on the R/W, RS, and data bus lines. When this pin is set to HIGH, the LCD processes the incoming data.

D0-D7 (Data Bus) pins carry the 8 bit data we send to the display. For example, if we want to see an uppercase ‘A’ character on the display, we set these pins to 0100 0001 (as per the ASCII table).

A-K (Anode & Cathode) pins are used to control the backlight of the LCD.

**Testing a Character LCD**

First, connect the 5V and GND pins from the Arduino to the breadboard power rail and plug your LCD into the breadboard.

Now we will power the LCD. The LCD has two separate power connections; One for the LCD (pin 1 and pin 2) and the other for the LCD backlight (pin 15 and pin 16). Connect pins 1 and 16 of the LCD to GND and 2 and 15 to 5V.

Most LCDs have a built-in series resistor for the LED backlight. You’ll find this near pin 15 on the back of the LCD. If your LCD does not include such a resistor or you are not sure if your LCD has one, you will need to add one between 5V and pin 15. It is safe to use a 220-ohm resistor, although a value this high may make the backlight a bit dim. For better results you can check the datasheet for maximum backlight current and select a suitable resistor value.

Next, we will make the connection for pin 3 on the LCD which controls the contrast and brightness of the display. To adjust the contrast, we will connect a 10K potentiometer between 5V and GND and connect the potentiometer’s centre pin (wiper) to pin 3 on the LCD.

That’s it. Now turn on the Arduino. You will see the backlight lit up. Now as you turn the knob on the potentiometer, you will start to see the first row of rectangles. If that happens, Congratulations! Your LCD is working fine.

**Test Code for 16x2 LCD:** [16X2\_LCD\16X2\_LCD.ino](16X2_LCD/16X2_LCD.ino)