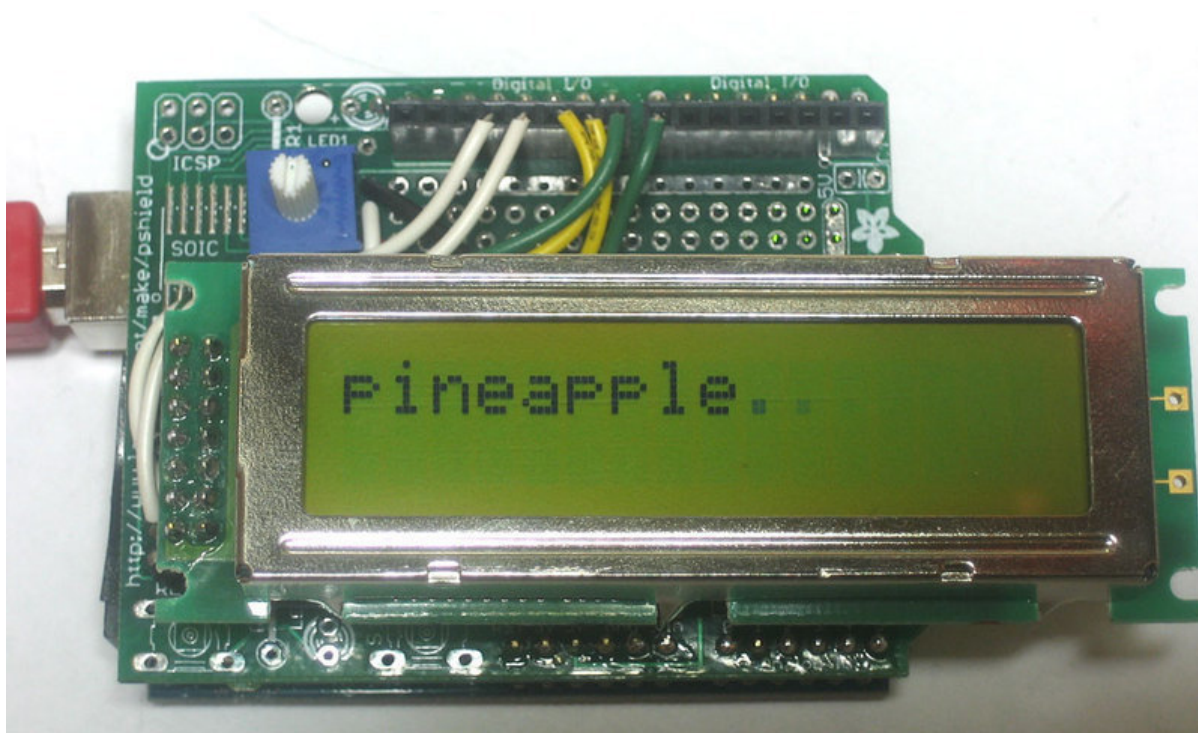




DIY 8x2 LCD Shield

Created by lady ada



<https://learn.adafruit.com/diy-8x2-lcd-shield>

Last updated on 2024-06-03 01:09:27 PM EDT

Table of Contents

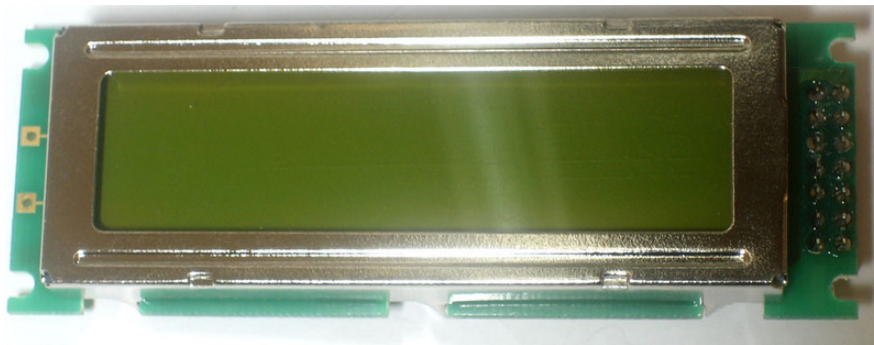
Overview & Parts	3
• Parts	
Check Out the Pins	3
Headers and Wiring	5
• Wiring the LCD	
Using the LCD Shield	8
• Arduino Sketch	

Overview & Parts

This tutorial will show you how to solder together a shield for an LCD with a 8x2 header.

Parts

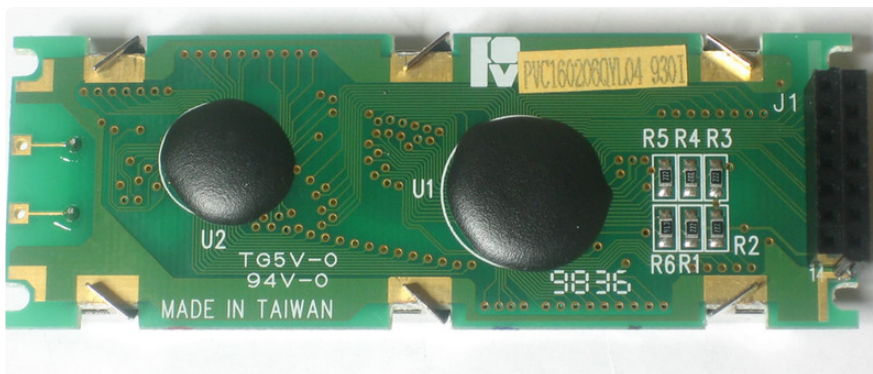
Aside from the parts offered available from Adafruit, you will need to source the LCD yourself. Parallel LCDs come in a couple different sizes, from 16 characters, 1 line (16x1) to 24 characters, 4 lines (24x4).



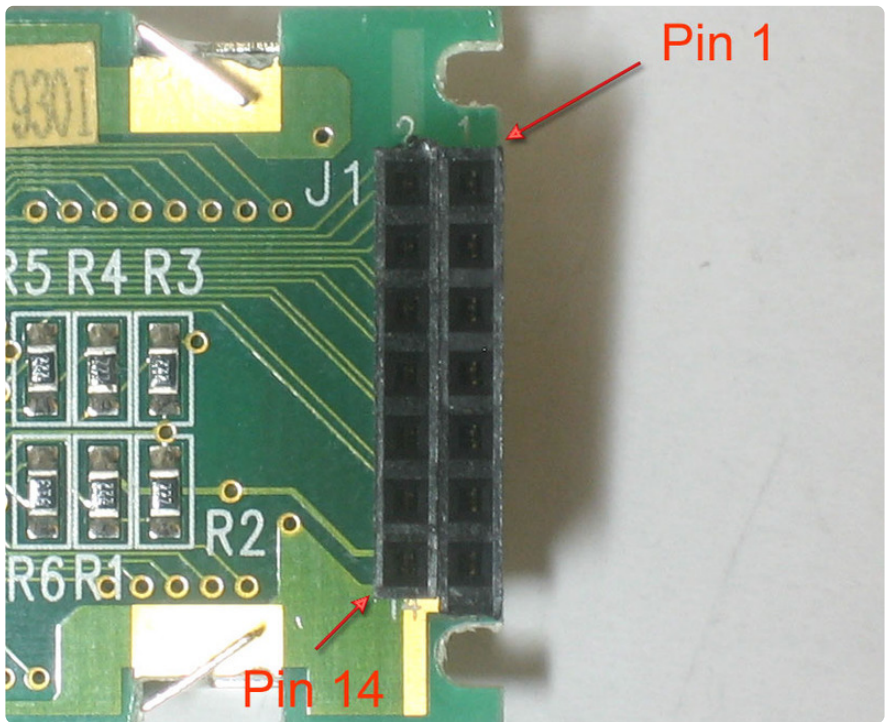
In this tutorial I use the [16x2 Picvue from Jameco \(https://adafru.it/aIJ\)](https://adafru.it/aIJ) because that's what I had in my box of stuff, but you can easily adapt it for any other LCD. The wiring may differ a little bit but that will be covered.

Check Out the Pins

The first step is to turn over the LCD and check out the pins. Parallel LCDs almost always have 14 or 16 pins. In this model they are all the way to the left, near the label J1. I soldered 2 7-pin female headers for e-z plugging. Sometimes the pins are along the bottom all in a row.



Look closely at the header for the numbers that show which pins are which. In this model, the first pin is at the top right, the second pin is top left, etc till pin 14 in the bottom left.

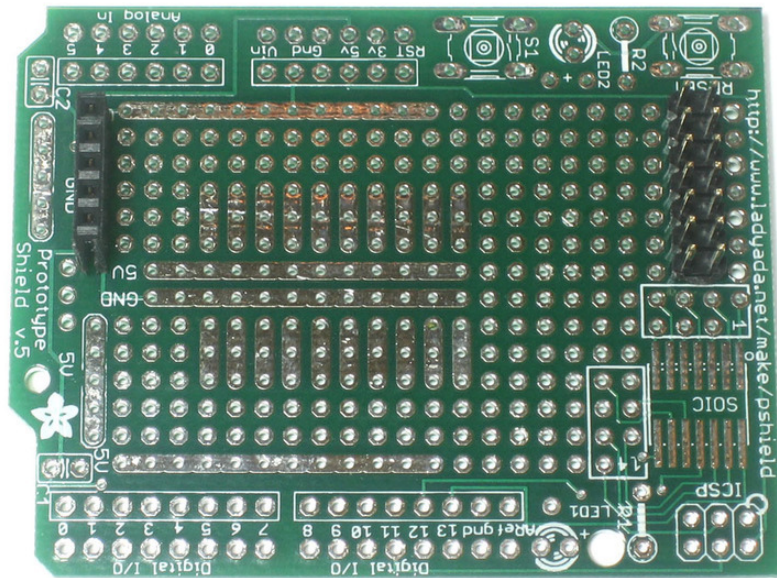


When the LCD is flipped over, the pinout will be mirrored, to help me keep track of it, I made a diagram:

1	2
3	4
5	6
7	8
9	10
11	12
13	14

Whatever your pinout is, make sure you have a diagram written out since you will need to refer to it many times!

Headers and Wiring



OK now that we are diagrammed out, I took the protoshield and soldered 2 7-pin headers at one end of the pcb. Note that they are not all the way to the edge, I left one row of holes so I could easily solder some wires. I also soldered a short piece of header (that comes with the shield kit) at the left so that the LCD will be propped up.



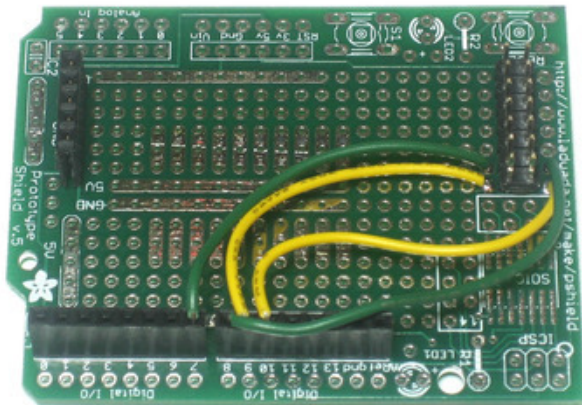
You can now do a test fit to verify how it will look. it hangs over a bit but thats OK by me.

Wiring the LCD

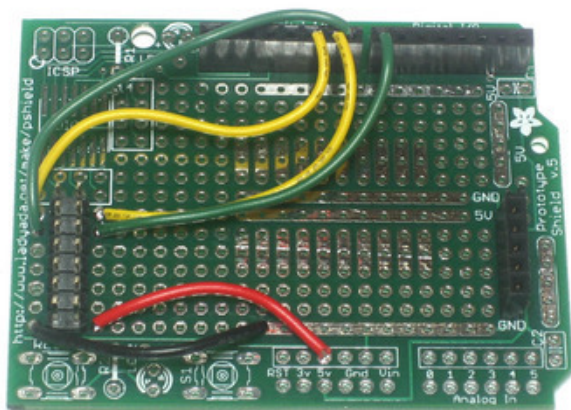
Since this is a parallel LCD, data will be sent to it over a parallel interface. That is, multiple bits at a time. These LCDs are designed for either a 8-bit or 4-bit interface. Since we'd like to save pins, lets go with the 4-bit interface! The data pins are name **D4**, **D5**, **D6**, and **D7**. Double-check your datasheet but almost all parallel LCDs have these pins numbered 4, 3, 2, and 1 respectively.

1 (D7)	2 (D6)
3 (D5)	4 (D4)
5	6
7	8
9 (ENABLE)	10 (R/W)
11 (RS)	12 (CONTRAST)
13 (GND/VSS)	14 (+5V/VDD)

There's a lot of wiring to be done but we're going to go thru it very slowly so it shouldn't be too bad.

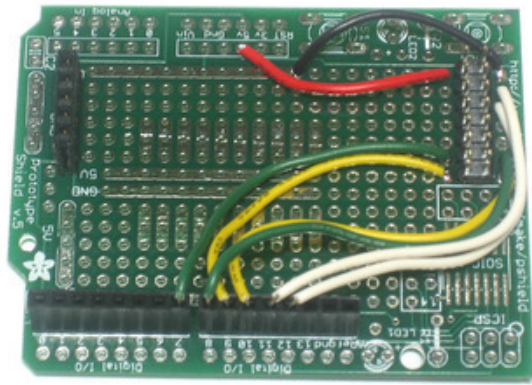


Lets connect these to the arduino thusly:
D4 -> Arduino pin #7, D5 -> Arduino pin #8,
D6 -> Arduino pin #9, D7 -> Arduino pin #10.

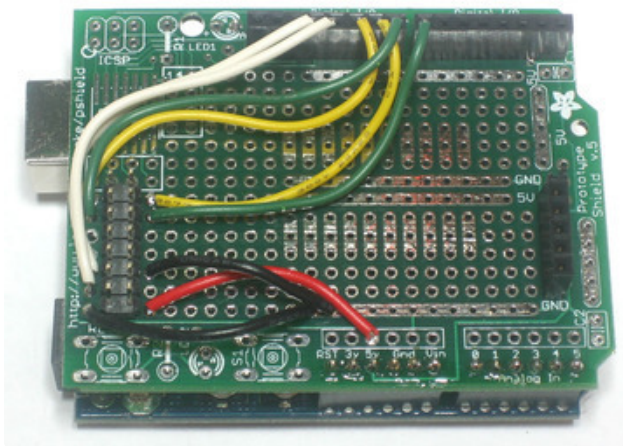


Since I wasnt sure of the wiring, I used the sockets on the protoshield. Once I test and verify they are correct, I'll solder them in!

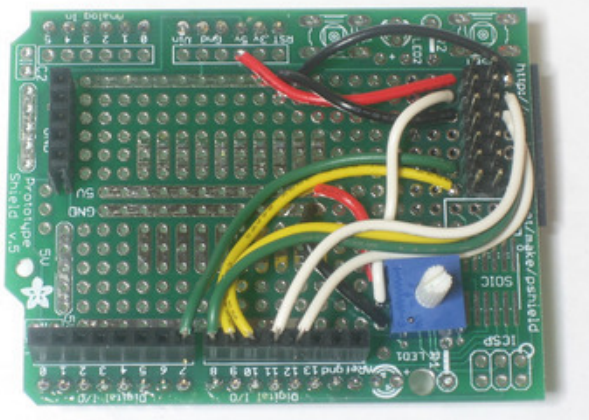
Next are the two power wires. Parallel LCDs run off of +5V so you can just solder the Vcc wire to 5V and the ground wire to GND.



Next are the 2 control wires, **ENABLE** and **RS** which we connect to pins 12 and 11 respectively.

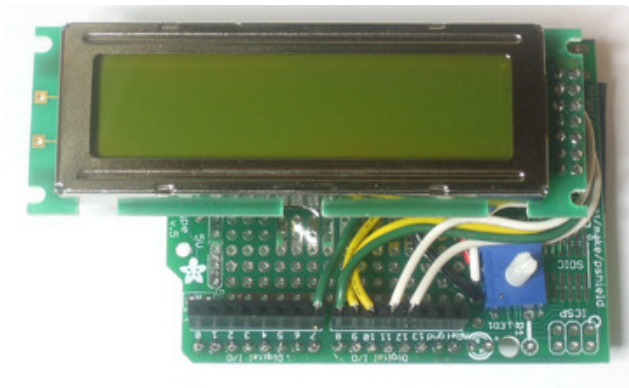


There's another control line called **R/W** that you can use to control whether you're reading or writing to the LCD. Since we'll just be writing, that pin can be connected to ground, saving another Arduino pin.



The last wire is the contrast control, we need to connect a potentiometer to this to make the display visible. I don't know the specifics of the input current but I used a 10K potentiometer and it worked great.

One pin is connected to +5V, the pin on the other side is connected to ground and the middle pin is connected to the contrast line.

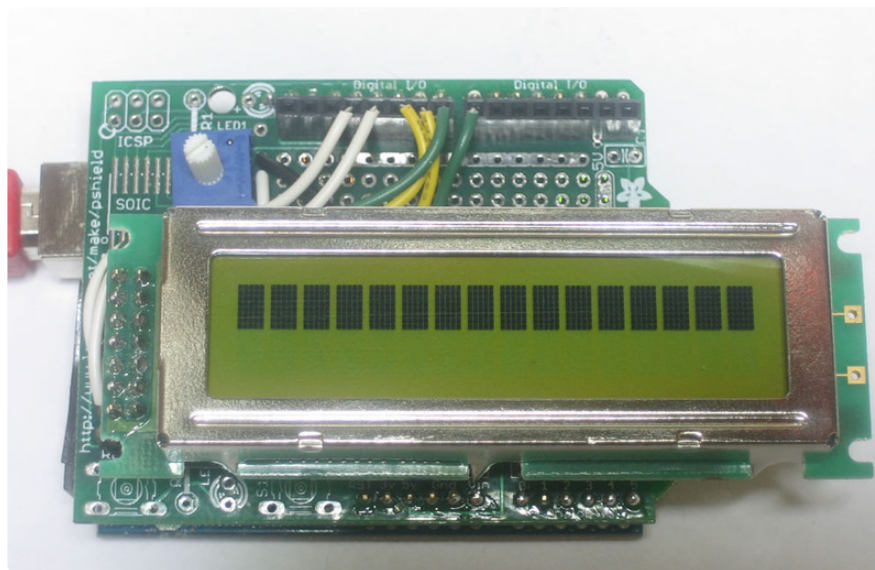


Now place the LCD on top. Looks good!

Make sure you finish up the rest of the shield so you can plug it into an arduino. At least solder in the male headers.

Using the LCD Shield

Our first test will be just to connect it up to power and see what happens. Plug it into an Arduino and power it up. You should see the following:



Make sure you tweak the contrast potentiometer, if the contrast is all the way down you may not see anything.

Arduino Sketch

Now we must upload some sketch to the Arduino to talk to the LCD. Luckily the **LiquidCrystal** library is already built in. So we just need to load one of the examples and modify it for the pins we used.

If you've changed the pins, you'll want to make a handy table so you can update the sketch properly.

LCD pin name	RS	EN	DB4	DB5	DB6	DB7
Arduino pin #	11	12	7	8	9	10

Open up the **File->Examples->LiquidCrystal->HelloWorld** example sketch

Now we'll need to update the pins. Look for this line:

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

And change it to:

```
LiquidCrystal lcd(11, 12, 7, 8, 9, 10);
```

To match the pin table we just made.

Now you can compile and upload the sketch, adjusting the contrast if necessary. (The image below is from a different fruity sketch but it will look similar).

