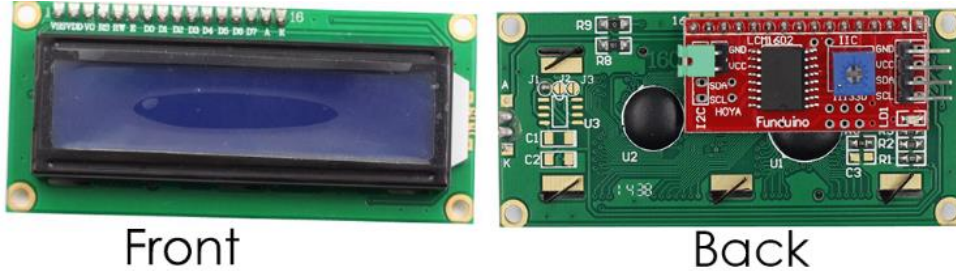# Driving an LCD Display with the
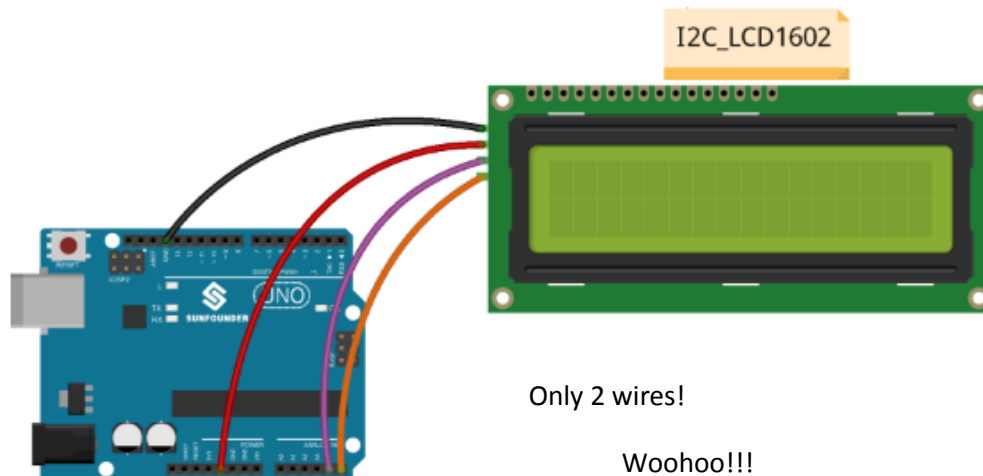# I2C Module

(G Payne – 2017)

Overview:

Connecting an LCD display to an Arduino is fairly straightforward. This tutorial is based on the Hitachi HD44780 type 16 row by 2 colum LCD display and an I2C display driver module.

Front   Back

You can drive a display directly with an Arduino, however the wiring is quite complex and it uses at least 6 pins on your Arduino.

Too many wires!

Yuck!!!
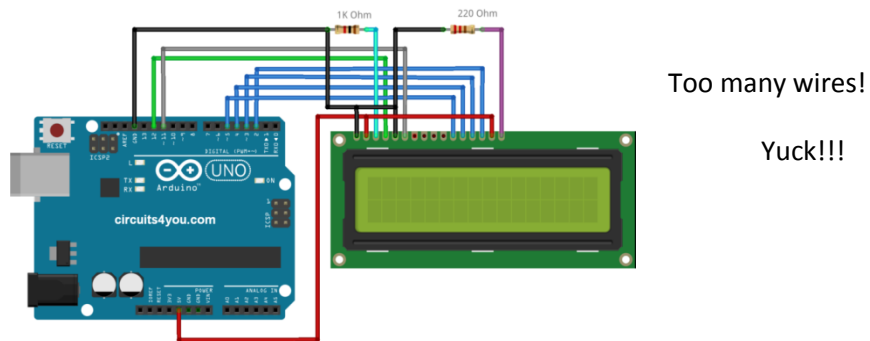
The I2C protocol was developed by Philips. The value of I2C devices is that they only require two (2) wires on your Arduino to control the device; an SDA data pin and an SCL clock pin.

I2C_LCD1602

Only 2 wires!

Woohoo!!!

Connecting the Module to your Arduino

The I2C module has 4 pins to connect to the Arduino. In the case of an Arduino UNO,

      VCC goes to a 5V source   GND goes to Ground

      SDA goes to Analog Pin 4   SCL goes to Analog Pin 5.

Here are the SDA and SCL pins for various Arduino boards:

Board	I2C / TWI pins
Uno, Ethernet	A4 (SDA), A5 (SCL)
Mega2560	20 (SDA), 21 (SCL)
Leonardo	2 (SDA), 3 (SCL)

As a fallback, you can always connect the SDA pin on the I2C module to the SDA pin on the Arduino and the SCL pin on the I2C module to the SCL pin on the Arduino.

The back of the I2C module has two additional pins. These are for the backlight that illuminates the display. If you want the backlight to be ON, you should connect these two pins together with a jumper clip or a short female-female jumper wire ('Dupont' connector). You may require 10K pull-up resistors on the SDA and SCL pins but try without the resistors first.

Writing Your Sketch

To drive the I2C module, you need the *LiquidCrystal_I2C* library for Arduino. There are several different libraries you can use. You can download it from:

https://github.com/fdebrabander/Arduino-LiquidCrystal-I2C-library

Just follow the instructions on the screen for putting this library into your Documents/Arduino/libraries folder.

Now type this code into a fresh Arduino sketch.

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

// Set the LCD address to 0x27 for a 16 chars and 2 line display
LiquidCrystal_I2C lcd(0x27, 16, 2);

void setup()
{
  // initialize the LCD
  lcd.begin();

  // Turn on the blacklight
  lcd.backlight();

}
```

```
void loop()
{
  lcd.setCursor(3, 0); //column 3, row 0
  lcd.print("Hello,");
  delay(500);
  lcd.setCursor(9, 1); //column 9, row 1
  lcd.print("World!");
  delay(500);
  lcd.clear();//clear the display
  delay(500);

}
```

Upload the sketch to your Arduino and you should see the message flashing on both lines of the display.

IT'S NOT WORKING!!!

1. Check to see if you correctly installed the LiquidCrystal_I2C library in your Documents/Arduino/libraries folder.
2. Make sure your Arduino is connected to the correct port and the IDE is set to the correct board you're using. These settings are under the 'Tools' menu.
3. Make sure you've connected the display to the Arduino correctly.
4. If the display is unreadable, make sure you connected the two backlight pins together on the back of the I2C module.
5. If the display is too dim, use the small adjustment screw in the back of the I2C module to adjust the brightness.
6. Perhaps your display is not address 0x27. To find the address of any I2C device, find an I2C Scanner sketch on the web and load/run it on your Arduino. The sketch will show you the address of the I2C device. If it's not 0x27, put the correct address in the 'LiquidCrystal_I2C' command in the setup() block of your sketch.

Now go out and MAKE SOMETHING AMAZING!!!!