

ELECTRONICS 1

ELECTRONICS FOR INTERACTIVE MEDIA DESIGN  
LES6

EMMA PARESCHI

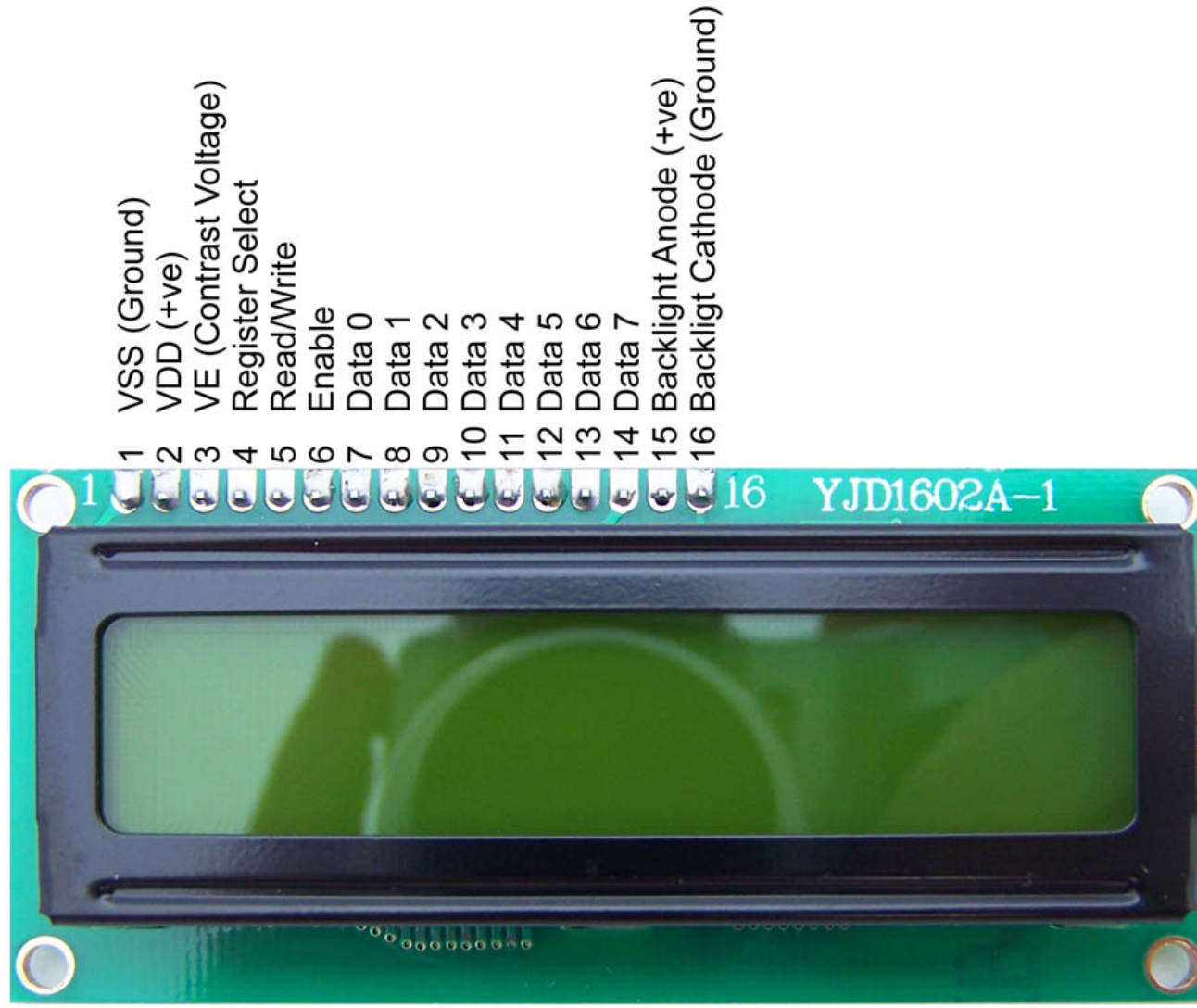
# LCD - CHIP

HITACHI HD44780 CHIP

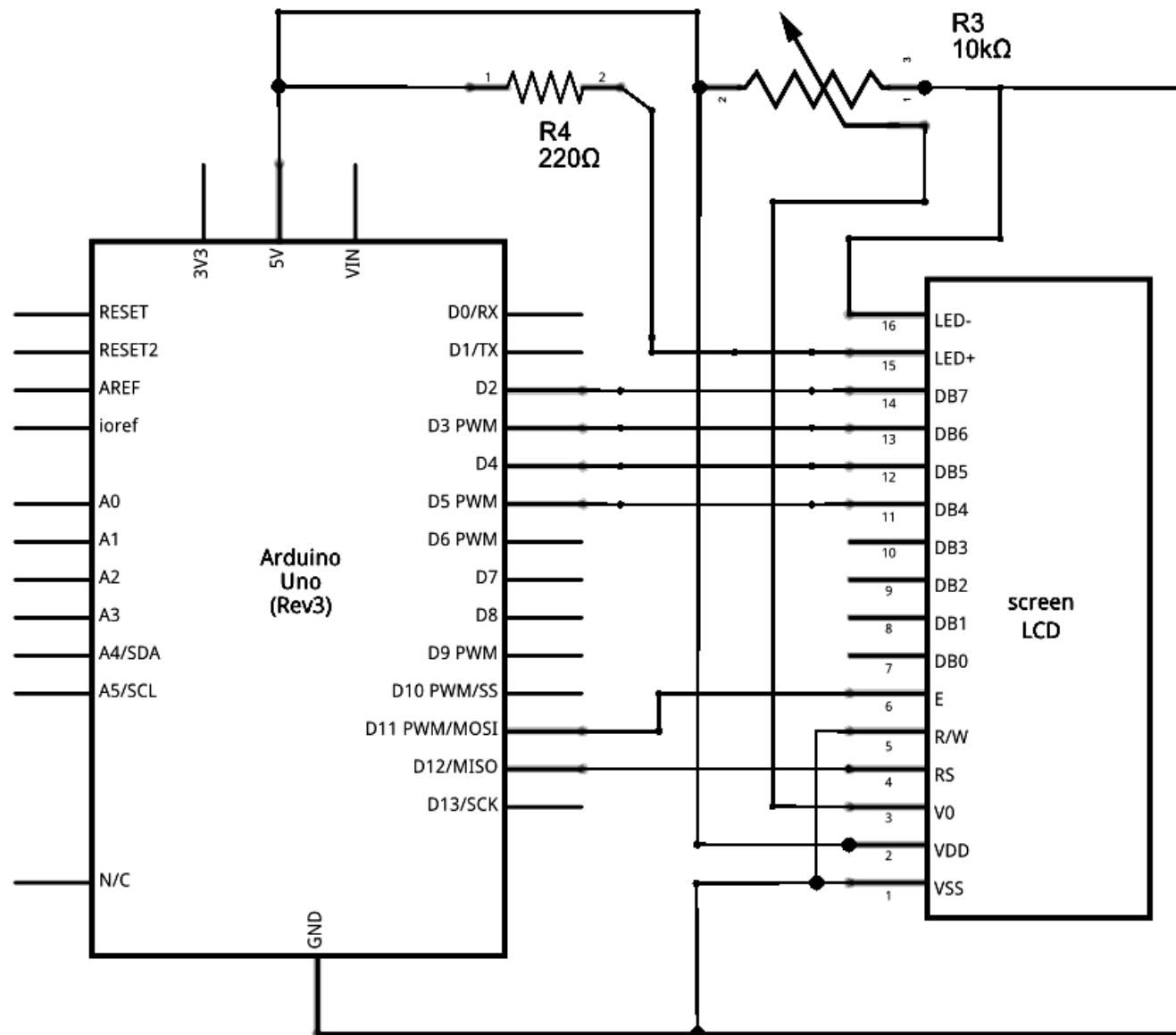
[HTTPS://WWW.SPARKFUN.COM/DATASHEETS/LCD/HD44780.PDF](https://www.sparkfun.com/datasheets/LCD/HD44780.pdf)



# LCD - PINS



# LCD - SCHEMATIC



# LCD - LIBRARY

[HTTPS://WWW.ARDUINO.CC/EN/REFERENCE/LIQUIDCRYSTAL](https://www.arduino.cc/en/Reference/LiquidCrystal)

LiquidCrystal library for driving LCD displays based on the HD44780 chip.

Most text LCDs supplied for use with Arduino will be compatible with the Hitachi HD44780 controller. If you are not sure about your controller, check the data sheet to see if it is a 44780 or compatible.

```
#include "LiquidCrystal.h"
```

```
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
```

# LCD - PINS

A register select (RS) pin that controls where in the LCD's memory you're writing data to. You can select either the data register, which holds what goes on the screen, or an instruction register, which is where the LCD's controller looks for instructions on what to do next.

A Read/Write (R/W) pin that selects reading mode or writing mode

An Enable pin that enables writing to the registers

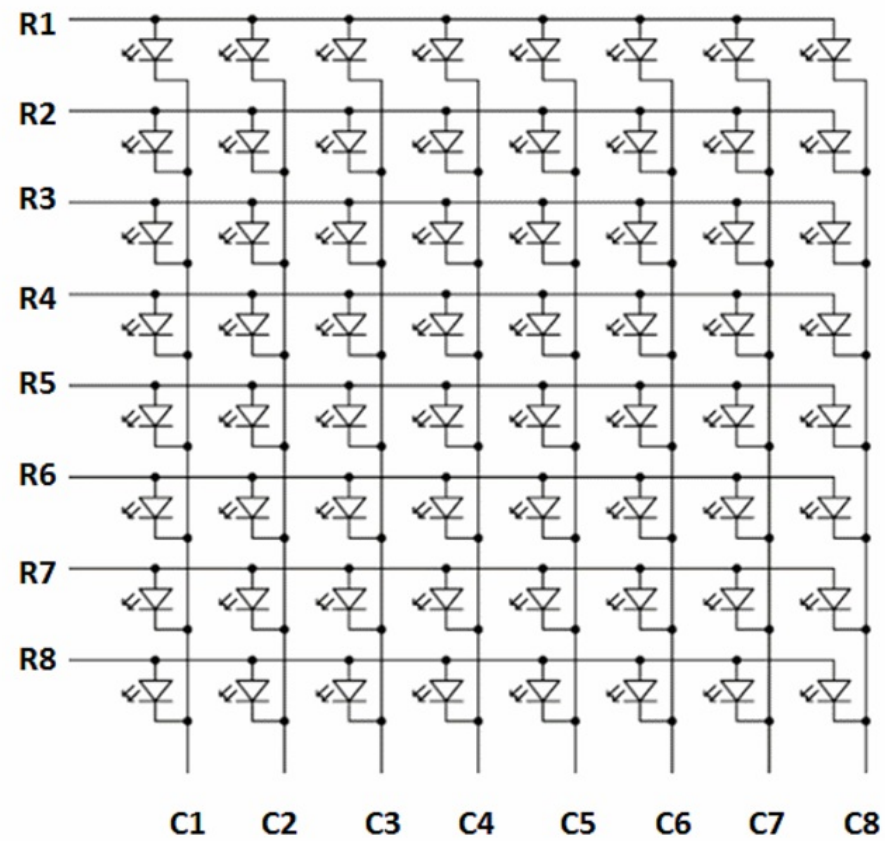
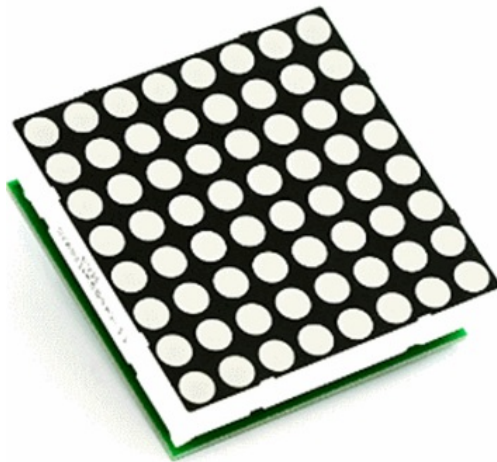
8 data pins (D0 -D7). The states of these pins (high or low) are the bits that you're writing to a register when you write, or the values you're reading when you read.

There's also a display contrast pin (Vo), power supply pins (+5V and Gnd) and LED Backlight (Bklt+ and Bklt-) pins that you can use to power the LCD, control the display contrast, and turn on and off the LED backlight, respectively.

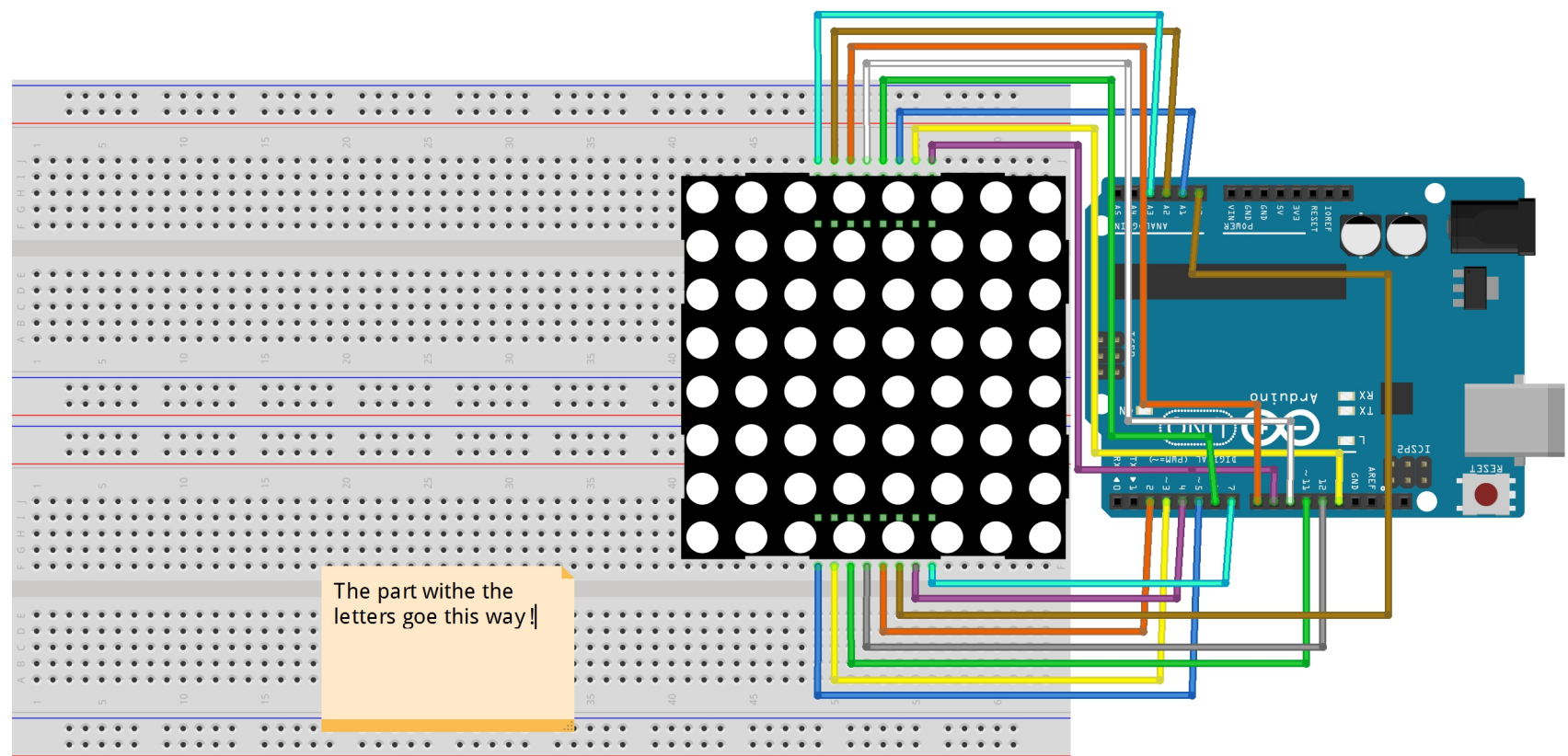
The Hitachi-compatible LCDs can be controlled in two modes: 4-bit or 8-bit. The 4-bit mode requires seven I/O pins from the Arduino, while the 8-bit mode requires 11 pins. For displaying text on the screen, you can do most everything in 4-bit mode.

# LED MATRIX 8x8

[HTTPS://WWW.SPARKFUN.COM/DATASHEETS/LCD/HD44780.PDF](https://www.sparkfun.com/datasheets/LCD/HD44780.pdf)

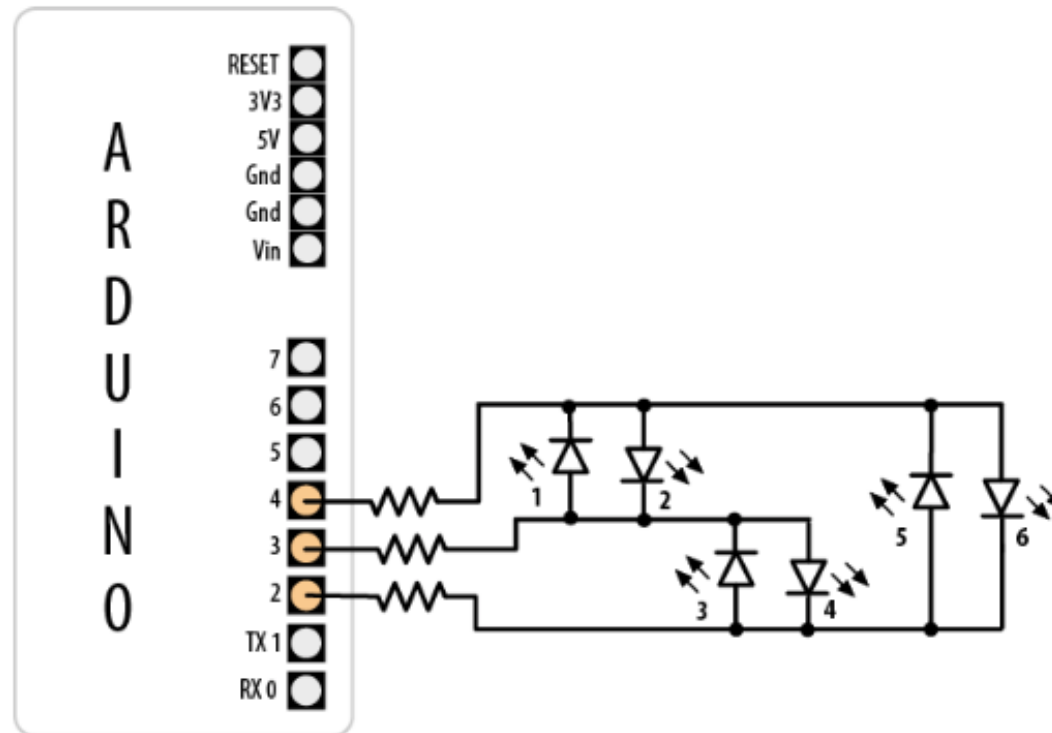


# LED MATRIX 8x8 - SCHEMATIC





# CHARLINPLEXING



# MATRIX

[HTTPS://CREATE.ARDUINO.CC/PROJECTHUB/IGORF2/CONTROLLING-AN-LED-MATRIX-WITH-ARDUINO-UNO-0A9E94](https://create.arduino.cc/projecthub/igorF2/controlling-an-led-matrix-with-arduino-uno-0a9e94)

[HTTPS://CREATE.ARDUINO.CC/PROJECTHUB/SANWANDTER1/PROGRAMMING-8X8-LED-MATRIX-23475A](https://create.arduino.cc/projecthub/sanwandter1/programming-8x8-led-matrix-23475a)

[HTTPS://CREATE.ARDUINO.CC/PROJECTHUB/SEARCH?Q=LED+MATRIX+8X8&TYPE=BASEARTICLE](https://create.arduino.cc/projecthub/search?q=led+matrix+8x8&type=basearticle)

# M I L L I S ( )

Syntax:

`time = millis()`

Returns the number of milliseconds since the Arduino board began running the current program. This number will overflow (go back to zero), after approximately 50 days.

A screenshot of the Arduino IDE interface. The window title is 'millis | Arduino 1.6.9'. The toolbar shows icons for checking, running, saving, and uploading. A dropdown menu is open showing 'millis'. The code editor contains the following code:

```
int time;

void setup(){
  Serial.begin(9600);
}

void loop(){
  time = millis();

  Serial.println(time); //prints time since program started
  delay(1000);          // wait a second so as not to send massive amounts of data
}
```

The status bar at the bottom says 'Done Saving.'

# ASSIGNMENT

- Dec. 12th: think a possible semester project and describe it on your blog.
- Assignment:
  - connect one of the sensor (from the kit) and visualise the value on the Lcd screen
  - Use the matrix with another sensor

# SOURCES AND LICENCE

Crystal Liquid Display

<https://www.sparkfun.com/tutorials/251>

<https://www.arduino.cc/en/Tutorial/HelloWorld>

Arduino Cookbook, O'Reilly, Michael Margolis

Chapter 11 Using Displays, pag. 333

## LICENCE

EXCEPT WHERE OTHERWISE NOTED, THIS WORK IS LICENSED UNDER:

[HTTPS://CREATIVECOMMONS.ORG/LICENSES/BY/4.0/](https://creativecommons.org/licenses/by/4.0/)

