

# Arduino - Display

# Keypad

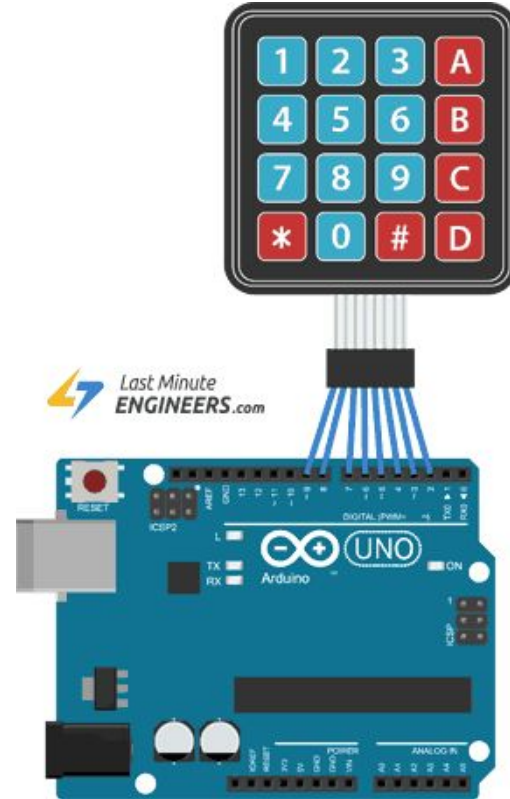
Membrane keypads come in a variety of sizes, the most common of which are the 4×3 keypad (12 keys) and the 4×4 keypad (16 keys). They have a layout similar to that of a standard telephone keypad, making them easy to use for anyone.



The keypad has a female Dupont connector. When looking at the front of the keypad, the row pins are on the left, and they usually have a dark strip near the connector to help identify them. The pinouts are as follows:



# Wiring Arduino



# Installing Keypad Library

To determine which key was pressed, we must continuously scan rows and columns. Fortunately, Keypad.h was written to abstract away this unnecessary complexity.

To install the library, navigate to Sketch > Include Library > Manage Libraries.

# Code example

```
#include <Keypad.h>

const int ROW_NUM = 4; //four rows
const int COLUMN_NUM = 4; //three columns

char keys[ROW_NUM][COLUMN_NUM] = {
    {'1','2','3', 'A'},
    {'4','5','6', 'B'},
    {'7','8','9', 'C'},
    {'*','0','#', 'D'}
};

byte pin_rows[ROW_NUM] = {13, 12, 11, 10}; //connect to the row pinouts (R1,R2,R3,R4) of the keypad
byte pin_column[COLUMN_NUM] = {7, 6, 5, 4}; //connect to the column pinouts (C1,C2,C3,C4) of the keypad

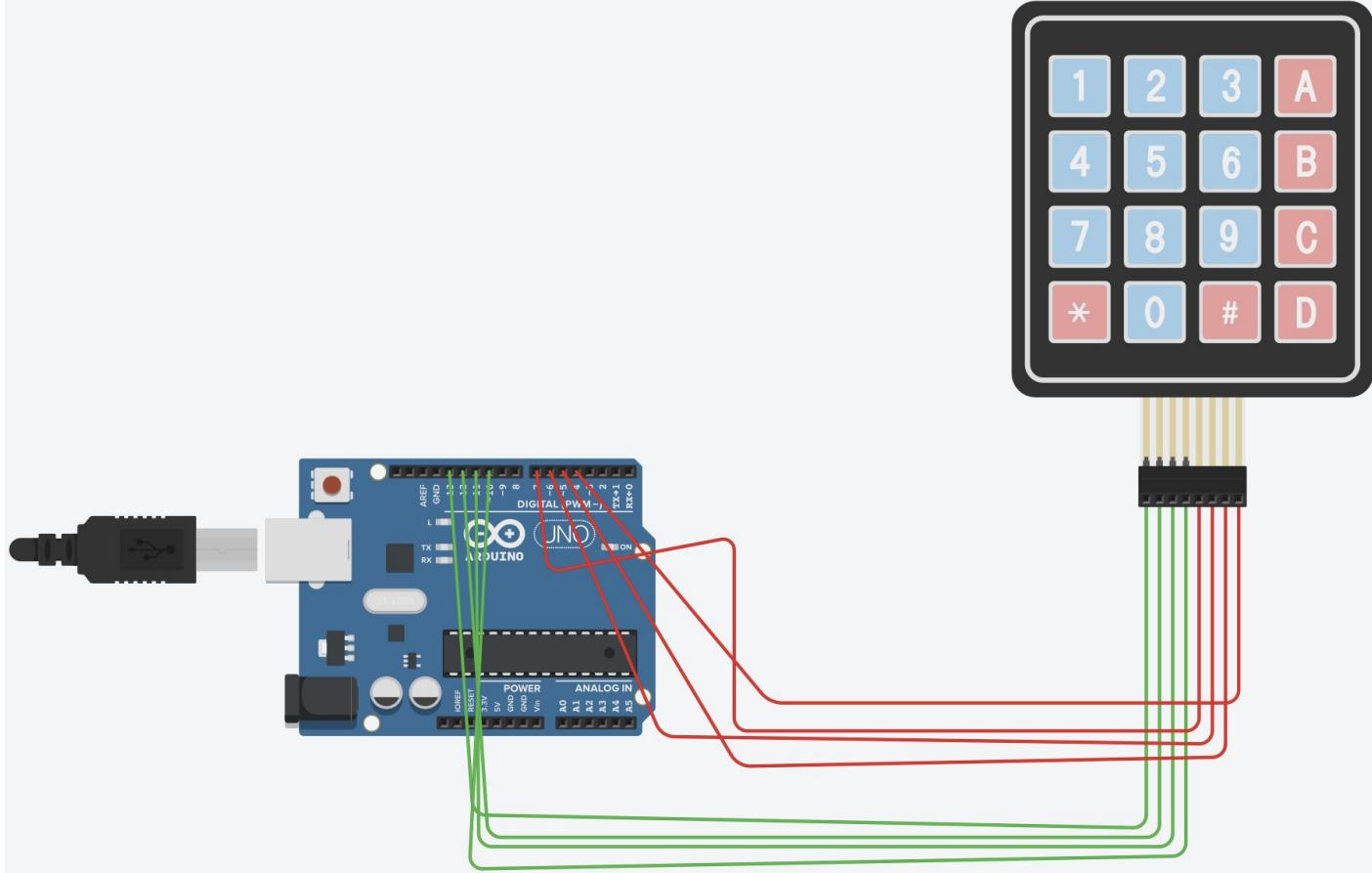
//any arduino pin is alright

Keypad keypad = Keypad( makeKeymap(keys), pin_rows, pin_column, ROW_NUM, COLUMN_NUM );

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

void loop(){
    char key = keypad.getKey();

    if (key){
        Serial.println(key);
    }
}
```



# Arduino-Keyboard

<https://arduinogetstarted.com/tutorials/arduino-keypad>

<https://lastminuteengineers.com/arduino-keypad-tutorial/>

[https://www.tinkercad.com/things/bCnP9fiNvMp-funky-bojo/editel?sharecode=wylHqLDhZLqrNsLfUgkHX\\_t591RZhDJhx4no\\_lhNVuM](https://www.tinkercad.com/things/bCnP9fiNvMp-funky-bojo/editel?sharecode=wylHqLDhZLqrNsLfUgkHX_t591RZhDJhx4no_lhNVuM)



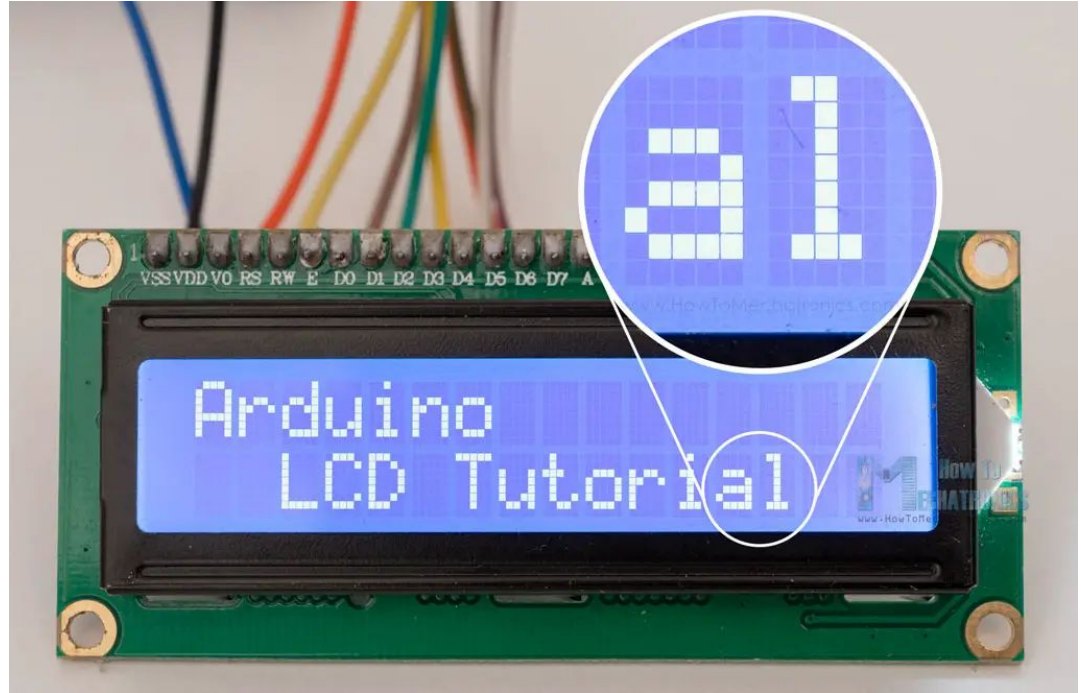
# LCD

Outputs individual ASCII characters with fixed size.

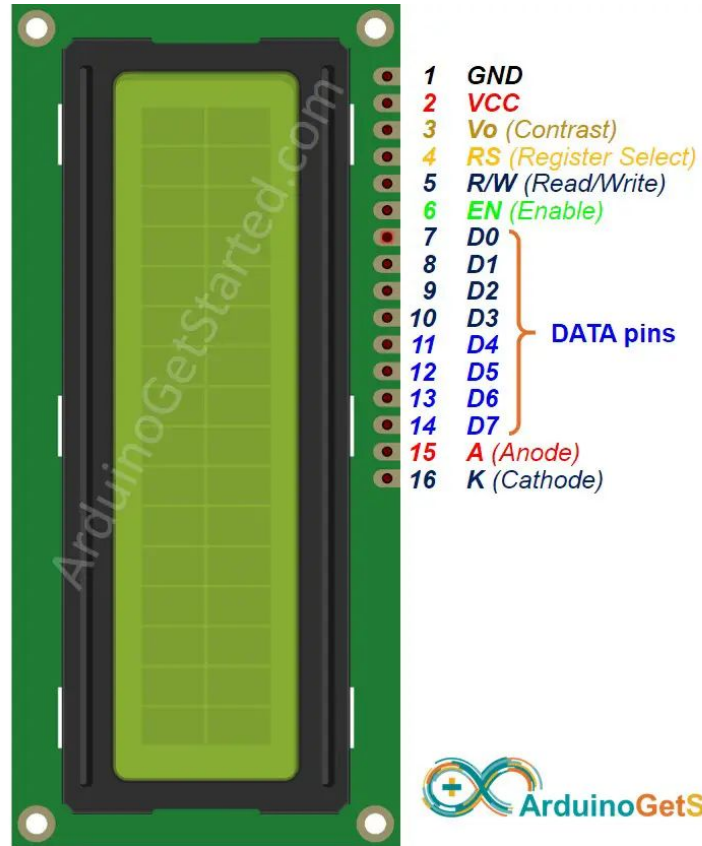
We can notice that there are small rectangular areas composed of 5×8 pixels grid

<https://howtomechatronics.com/tutorials/arduino/lcd-tutorial/>

↵



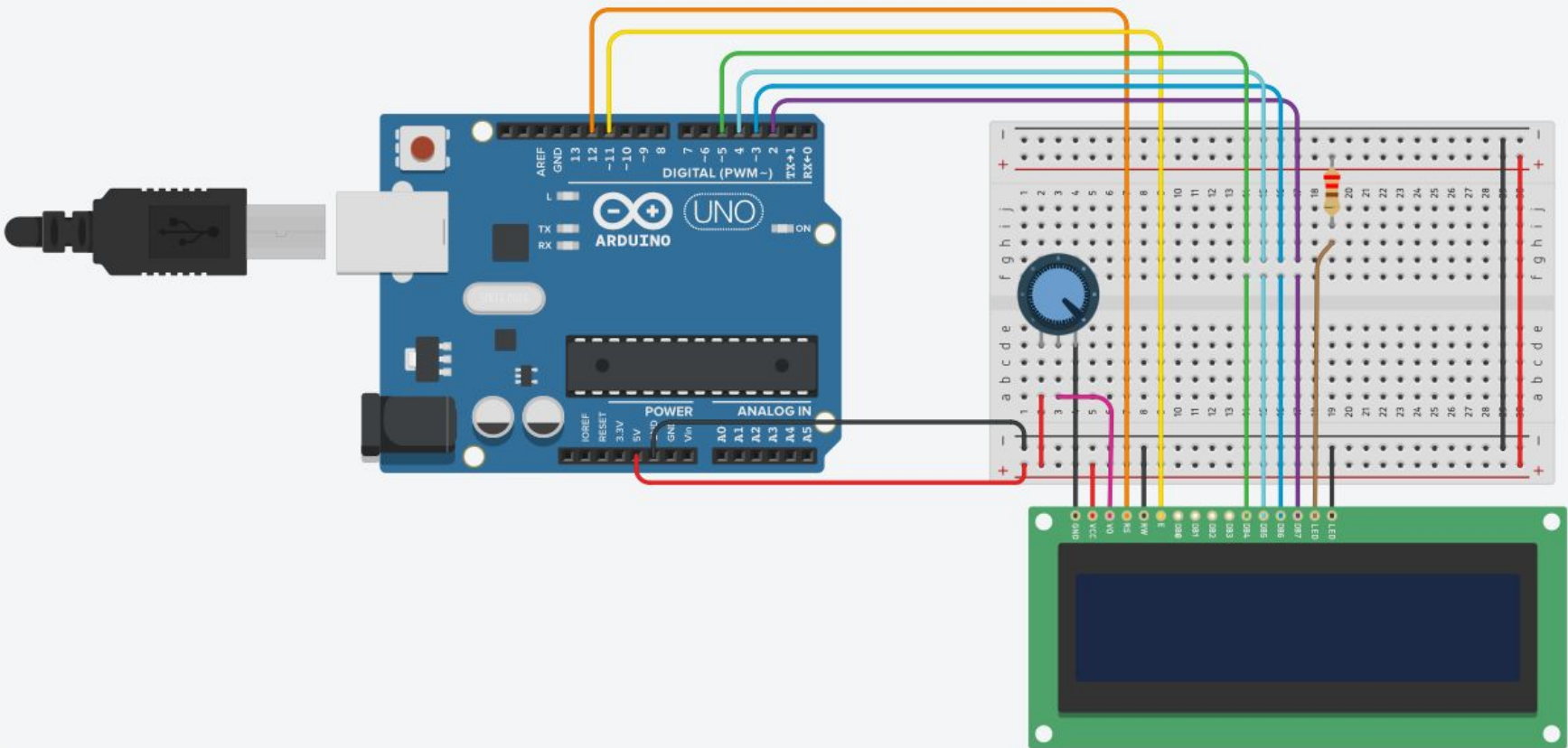
# LCD 16X2 display



## LCD pin table in 4-bit mode

	LCD PIN	CONNECTED TO
01	GND	GND
02	VCC	5V
03	Vo	5V or potentiometer's pin
04	RS	An Arduino's pin
05	R/W	GND
06	EN	An Arduino's pin
07	D0	NOT connected
08	D1	NOT connected
09	D2	NOT connected
10	D3	NOT connected
11	D4	An Arduino's pin
12	D5	An Arduino's pin
13	D6	An Arduino's pin
14	D7	An Arduino's pin
15	A	5V
16	K	GND





<https://www.tinkercad.com/things/8nzsT9fJLhe-super-elzing/editel?sharecode=JHlBY9FwvxUgWKJQ77UNgurVJR4-IzFLAYgKIBuZI3A>

```
#include <LiquidCrystal.h>

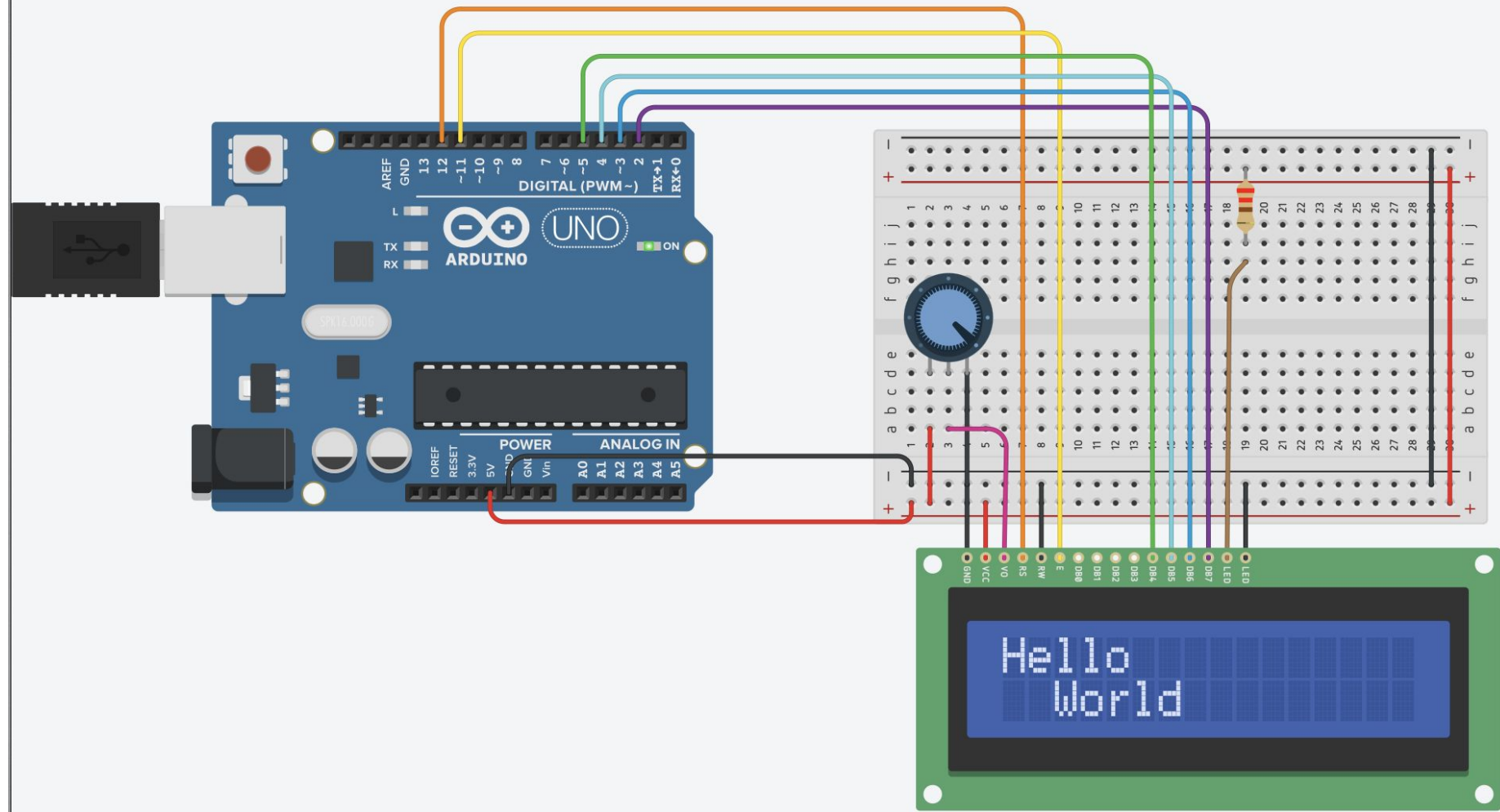
// LCD pins <--> Arduino pins
const int RS = 12, EN = 11, D4 = 5, D5 = 4, D6 = 3, D7 = 2;
LiquidCrystal lcd(RS, EN, D4, D5, D6, D7);

void setup()
{
    lcd.begin(16, 2); // set up number of columns and rows

    lcd.setCursor(0, 0);          // move cursor to (0, 0)
    lcd.print("Hello");           // print message at (0, 0)
    lcd.setCursor(2, 1);          // move cursor to (2, 1)
    lcd.print("World");           // print message at (2, 1)
}

void loop()
{

}
```





```

#include <LiquidCrystal.h>
#include <Keypad.h>

const int ROW_NUM = 4; //four rows
const int COLUMN_NUM = 4; //four columns

char keys[ROW_NUM][COLUMN_NUM] = {
  {'1','2','3', 'A'},
  {'4','5','6', 'B'},
  {'7','8','9', 'C'},
  {'*','0','#', 'D'}
};

byte pin_rows[ROW_NUM] = {48, 46, 44, 42}; //connect to the
row pinouts of the keypad
byte pin_column[COLUMN_NUM] = {40, 38, 36, 34}; //connect to
the column pinouts of the keypad

Keypad keypad = Keypad( makeKeymap(keys), pin_rows,
pin_column, ROW_NUM, COLUMN_NUM );

// LCD pins <--> Arduino pins
const int RS = 11, EN = 12, D4 = 2, D5 = 3, D6 = 4, D7 = 5;

LiquidCrystal lcd(RS, EN, D4, D5, D6, D7);

```

```

void setup()

{
  Serial.begin(9600);
  lcd.begin(16, 2); // set up number of columns and rows
  lcd.setCursor(0, 0); // move cursor to (0, 0)
  lcd.print("Hello"); // print message at (0, 0)
  lcd.setCursor(2, 1); // move cursor to (2, 1)
  lcd.print("World"); // print message at (2, 1)
}

void loop()
{
  char key = keypad.getKey();
  if(key){
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print(key);
  }
}

```

# Custom Character

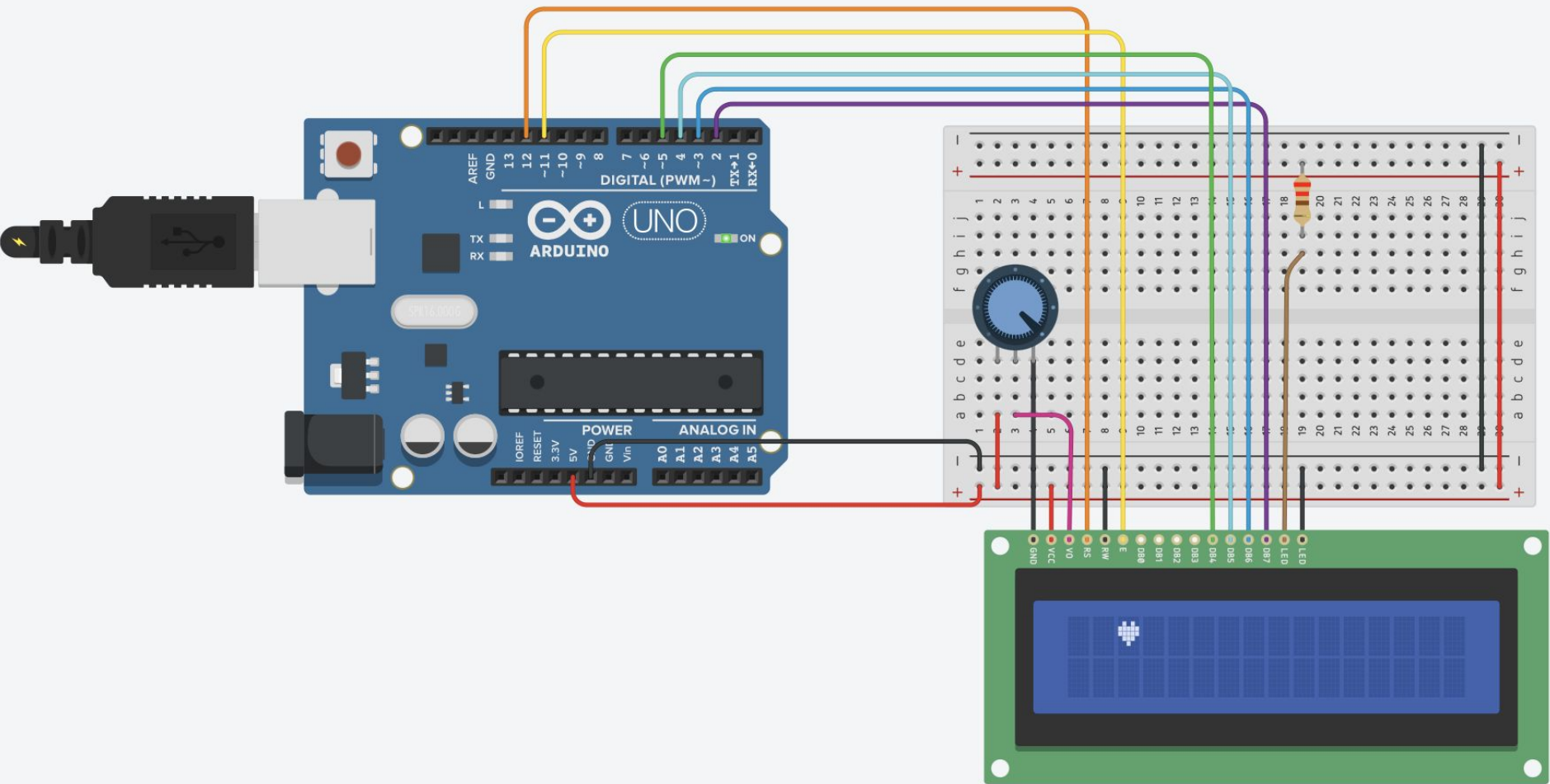
```
#include <LiquidCrystal.h>

// LCD pins <--> Arduino pins
const int RS = 12, EN = 11, D4 = 5, D5 = 4, D6 = 3, D7 = 2;
LiquidCrystal lcd(RS, EN, D4, D5, D6, D7);

byte customChar[8] = {
    0b00000,
    0b01010,
    0b11111,
    0b11111,
    0b01110,
    0b00100,
    0b00000,
    0b00000
};

void setup(){
    lcd.begin(16, 2); // set up number of columns and rows
    lcd.createChar(0, customChar); // create a new custom character
    lcd.setCursor(2, 0); // move cursor to (2, 0)
    lcd.write((byte)0); // print the custom char at (2, 0)
}

void loop()
{
}
```



```

#include <LiquidCrystal.h>

// LCD pins <--> Arduino pins
const int RS = 12, EN = 11, D4 = 5, D5 = 4,
D6 = 3, D7 = 2;
LiquidCrystal lcd(RS, EN, D4, D5, D6, D7);

byte customChar[8] = {
    0b00000,
    0b01010,
    0b11111,
    0b11111,
    0b01110,
    0b00100,
    0b00000,
    0b00000
};

byte customChar2[8] = {
    0b00000,
    0b00000,
    0b01110,
    0b01110,
    0b01110,
    0b00000,
    0b00000,
    0b00000
};

```

```

void setup()
{
    lcd.begin(16, 2); // set up number of columns and rows

    lcd.createChar(1, customChar); // create a new custom character, first
    //parameter can be from 0 to 7. Later the same number needs to be used to write
    //on LCD

    lcd.setCursor(0, 0); // move cursor to (0, 0)
    lcd.write((byte)1); // print the custom char at (0, 0)

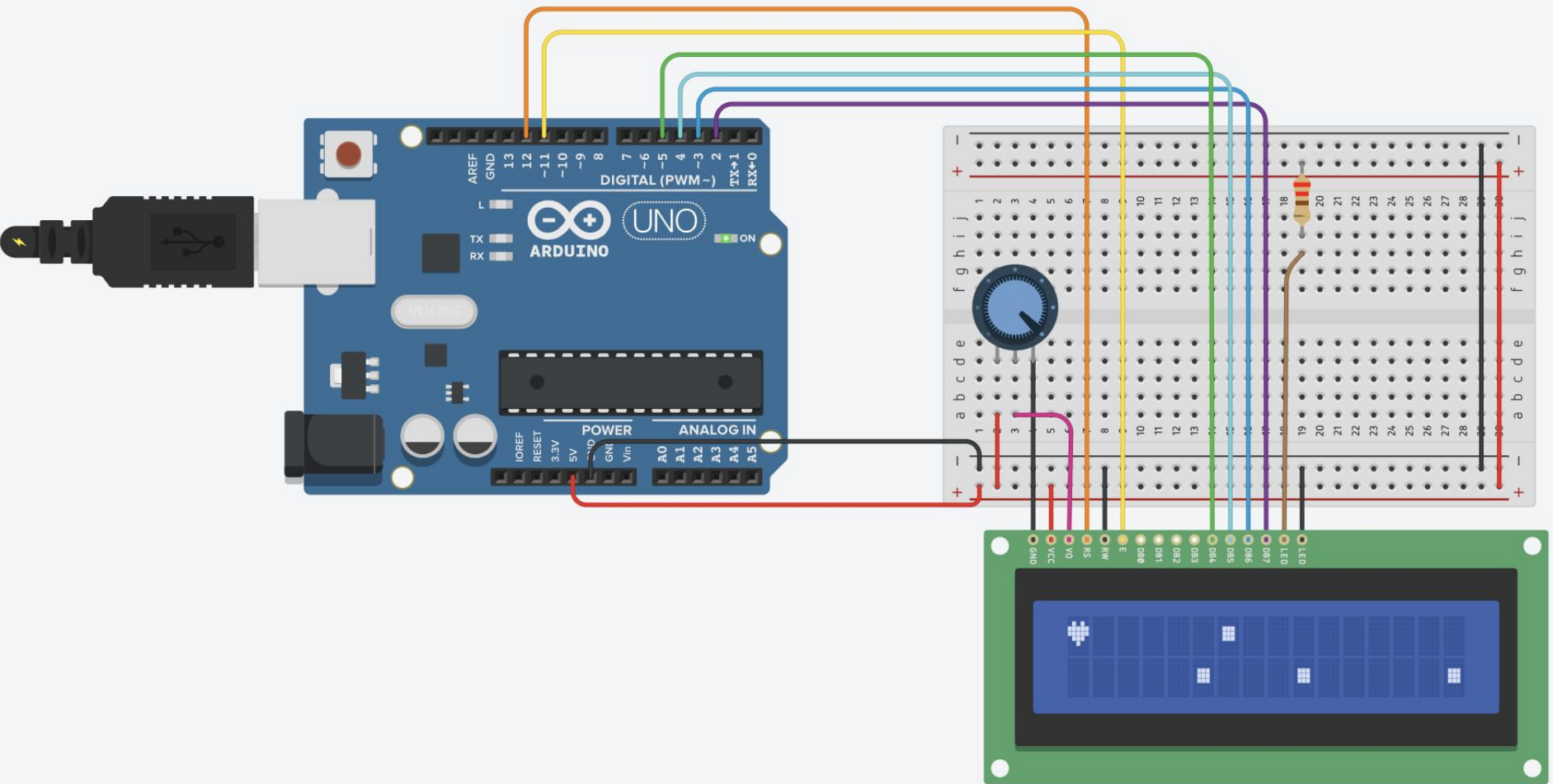
    lcd.createChar(2, customChar2); // create a new custom character
    lcd.setCursor(5, 1); // move cursor to (5, 1)
    lcd.write((byte)2); // print the custom char at (5, 1)

    lcd.setCursor(9, 1); // move cursor to (9, 1)
    lcd.write((byte)2); // print the custom char at (9, 1)

    lcd.setCursor(6, 0); // move cursor to (6, 0)
    lcd.write((byte)2); // print the custom char at (6, 0)
    lcd.setCursor(15, 1); // move cursor to (15, 1)
    lcd.write((byte)2); // print the custom char at (15, 1)
}

void loop()
{
}

```



<https://arduinogetstarted.com/tutorials/arduino-lcd>

<https://docs.arduino.cc/learn/electronics/lcd-displays>

<https://howtomechatronics.com/tutorials/arduino/lcd-tutorial/>