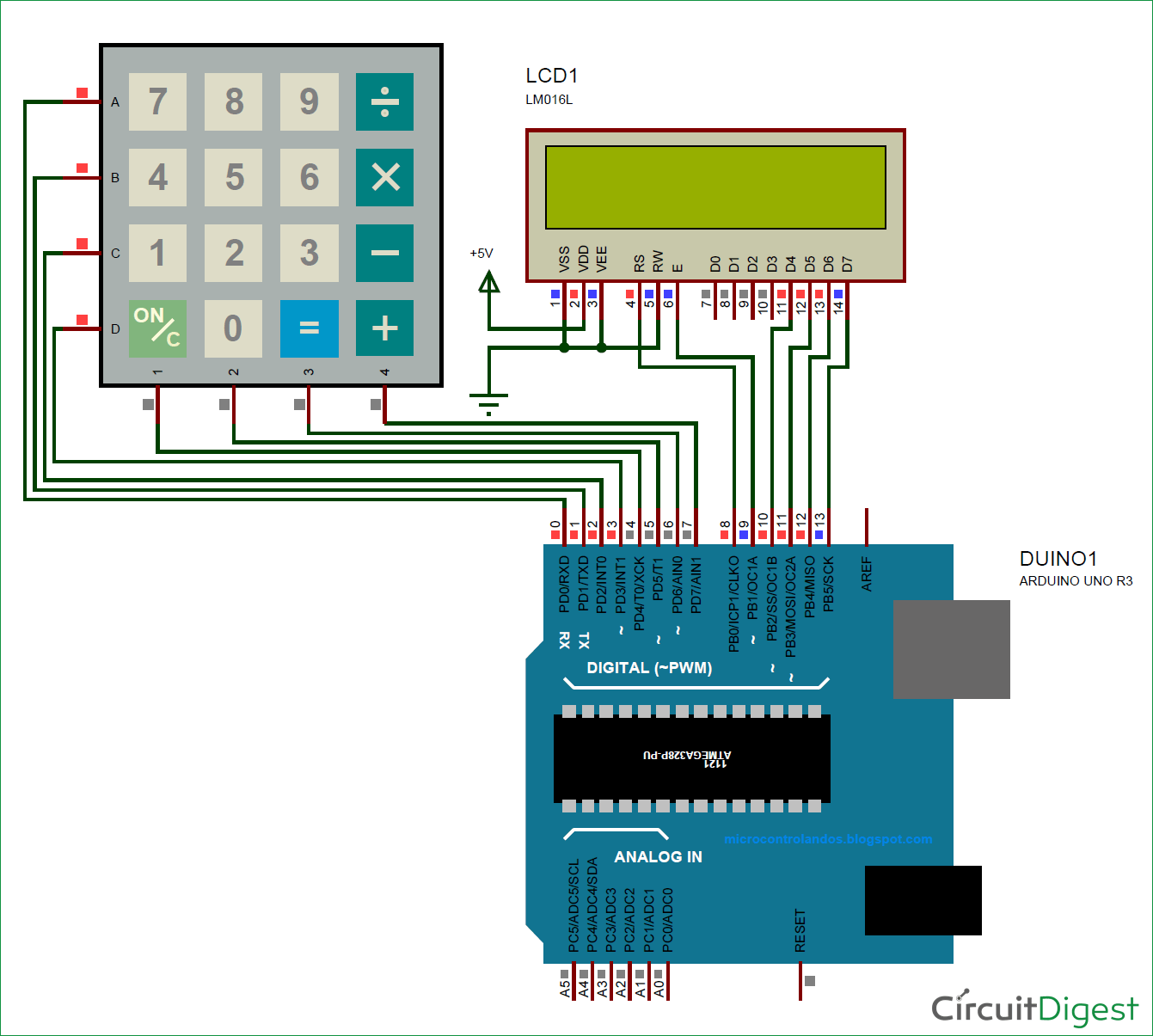
* Arduino Calculator using 4x4 Keypad

Programming is always fun and Arduino is a wonderful platform if you are just getting started with Embedded programming. In this tutorial we will **build our own calculator with Arduino**. The values can be sent in through a **keypad (4×4 keypad)** and result can be viewed on a **LCD screen** (16×2 Dot-matrix). This calculator could perform simple operations like Addition, Subtraction, Multiplication and Division with whole numbers. But once you understand the concept you can implement even scientific functions with Arduino’s built in functions. At the end of this project you will know how to use a [16x2 LCD](https://circuitdigest.com/article/16x2-lcd-display-module-pinout-datasheet) and Keypad with Arduino and also how easy it is to program for them using the readily available libraries. You will also understand how to program your Arduino for accomplishing a particular task.

### ****Materials Required:****

1. Arduino Uno (Any version will work)
2. 16×2 LCD Display
3. 4×4 Keypad
4. 9V Battery
5. Breadboard and Connecting wires

The complete circuit diagram of this **Arduino Calculator Project** is given above. The +5V and ground connection shown in the circuit diagram can be obtained from the 5V and ground pin of the Arduino. The Arduino itself can be powered from your laptop or through the DC jack using a 12V adapter or 9V battery.

We are operating the [LCD in 4-bit mode with Arduino](https://circuitdigest.com/microcontroller-projects/arduino-lcd-interfacing-tutorial) so only the last four data bits of the LCD is connected to Arduino. The Keyboard will have 8 output pins which have to be connected from pin 0 to pin 7 as shown above. You can use the following connection table to verify your connection with Arduino, you can also check [4x4 Keypad interfacing with Arduino](https://circuitdigest.com/microcontroller-projects/keypad-interfacing-with-arduino-uno).

|  |  |
| --- | --- |
| Arduino Pin Name: | Connected to: |
| D0 | 1st pin of the keyboard |
| D1 | 2nd pin of the keyboard |
| D2 | 3rd pin of the keyboard |
| D3 | 4th pin of the keyboard |
| D4 | 5th pin of the keyboard |
| D5 | 6th pin of the keyboard |
| D6 | 7th pin of the keyboard |
| D7 | 8th pin of the keyboard |
| D8 | Register select pin of LCD (pin 4) |
| D9 | Enable pin of LCD (pin 6) |
| D10 | Data pin 4 (pin 11) |
| D11 | Data pin 4 (pin 11) |
| D12 | Data pin 4 (pin 11) |
| D13 | Data pin 4 (pin 11) |
| +5V | Connected to Vdd pin of LCD (pin 2) |
| Ground | Connected to Vss,Vee and RW pin of LCD (pin 1,3 and 5) |

Some Arduino boards might show an error while uploading program if there are anything connected to pin 0 and pin1, so if you experience any just remove the keypad while uploading the program.

Once your connections are done your hardware will look something like this below

### Arduino Calculator using 4x4 Keypad in action hardware implementation****Arduino Calculator Program:****

The **complete Arduino program** for this project is given at the end of this project. The code is split into small meaningful chunks and explained below.

As told earlier we are going to **interface a LCD and keypad with Arduino using libraries**. So let’s add them to our Arduino IDE first. The [library for LCD](https://www.arduino.cc/en/Reference/LiquidCrystal) is already included in your Arduino by default so we need not worry about it. For [Keypad library](https://github.com/Chris--A/Keypad/archive/master.zip) click on the link to download it from Github. You will get a ZIP file, then add this lib to Arduino by Sketch -> Include Library -> Add .ZIP file and point the location to this downloaded file. Once done we are all set for programming.

Even though we have used a library for using a keypad we have to mention few details (shown below) about the keypad to the Arduino.  The variable ROWS and COLS will tell how many rows and columns our keypad has and the keymap shows the order in which the keys are present on the keyboard. The keypad that i am using in this project looks like this below to the key map also represents the same.

Further below we have mentioned to which **pins the Keypad is connected** using the variable array rowPins and colPins.

const byte ROWS = 4; // Four rows

const byte COLS = 4; // Three columns

// Define the Keymap

char keys[ROWS][COLS] = {

{'1','2','3','A'},

{'4','5','6','B'},

{'7','8','9','C'},

{'\*','0','#','D'}

};

byte rowPins[ROWS] = { 0, 1, 2, 3 };// Connect keypad ROW0, ROW1, ROW2 and ROW3 to these Arduino pins.

byte colPins[COLS] = { 4, 5, 6, 7 }; // Connect keypad COL0, COL1 and COL2 to these Arduino pins.

Once we have mentioned what type of keypad we are using and how it is connected, we can **create the keypad** using those details using the line below

Keypad kpd = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS ); // Create the Keypad

Similarly we also have to tell to **which pins of the Arduino the LCD is connected** to. According to our circuit diagram the definitions would be like below

const int rs = 8, en = 9, d4 = 10, d5 = 11, d6 = 12, d7 = 13; //Pins to which LCD is connected

LiquidCrystal lcd(rs, en, d4, d5, d6, d7); //create the LCD

**Inside the setup function**, we just display the name of the project and then proceed to while loop where the main project lies.

Basically, we have to check if anything is being typed on the keypad, if typed we have to recognize what is being typed and then convert it to a variable when the “=” is pressed we have to calculate the result and then finally display it on the LCD. This is exactly what is done inside the loop function as shown below

key = kpd.getKey(); //storing pressed key value in a char

if (key!=NO\_KEY)

DetectButtons();

if (result==true)

CalculateResult();

DisplayResult();

What happens inside each function is explained using the comment lines, go through the complete code below, fiddle around with it to understand how it actually works. If you have any doubt on a specific line, feel free to use the comment section or the forums.

### ****Simulation of Arduino Calculator:****

We can also try simulating the project using Proteus software. Proteus does not have an Arduino component on it’s own, but can be easily downloaded and added to its library.  Once you have the Arduino component on Proteus, just add Alphanumeric LCD and Keypad to make the connection as shown in the circuit diagram.

Then download the hex file from here and add it to the Arduino by double clicking on board in Proteus and point the “program file” to this downloaded hex file. A snap shot of the simulation is shown below, the complete working is shown in the **video**below.

https://circuitdigest.com/microcontroller-projects/arduino-calculator-using-4x4-keypad