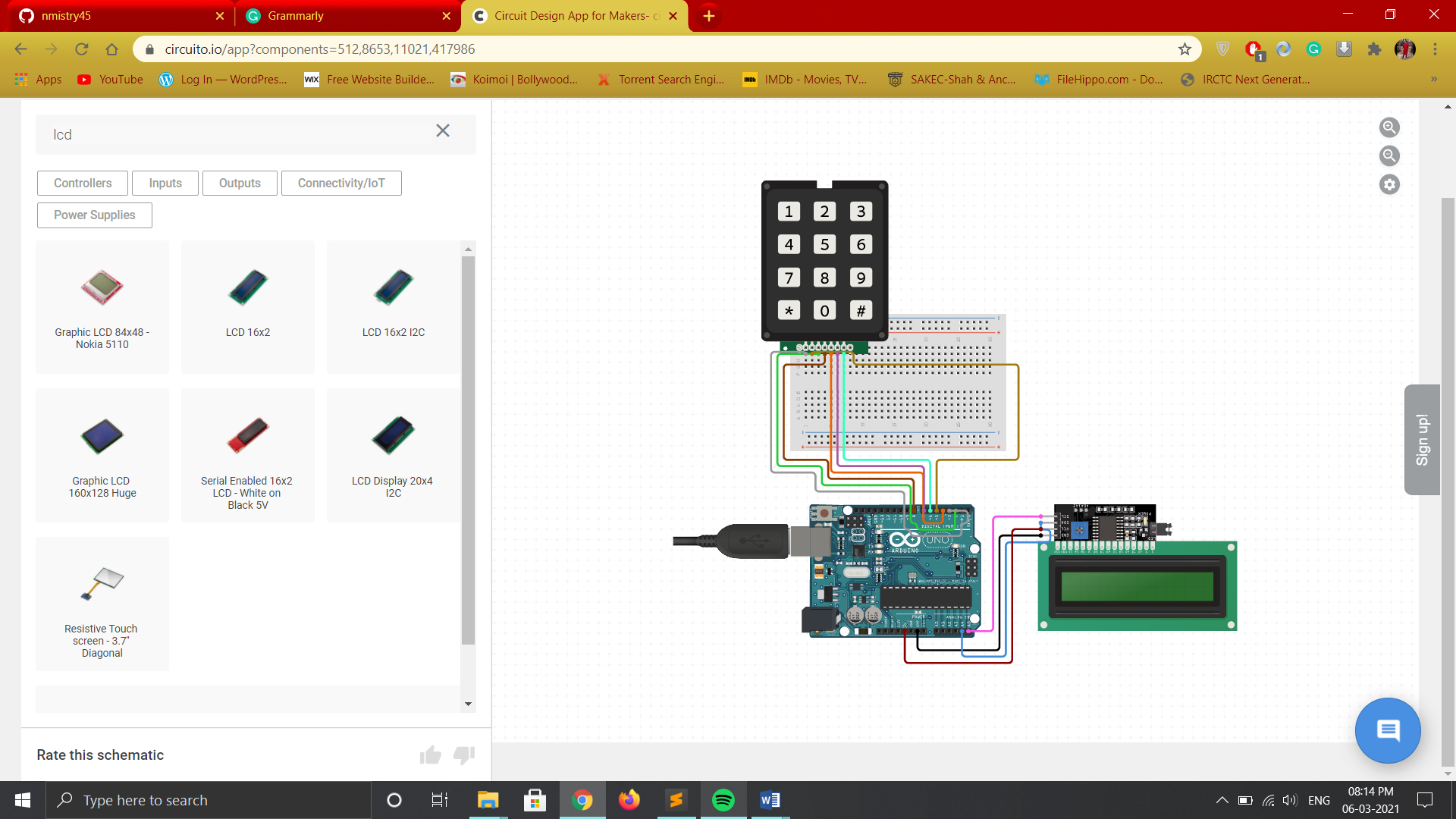
**CALCULATOR & SCIENTIFIC CALCULATOR USING ARDUINO**

Calculators or Scientific Calculators have become an integral part of our daily lives. For small calculations or big ones, to cross-check values, we use them regularly.

We are going to use the 16\*2 LCD to display the calculations from the 4\*4 Keypad Matrix. Mainly, the keypad is the input and it has all 10 numbers, \* and # symbols which can be mapped to anything, including 4 alphabets - A, B, C, & D that can also be used as anything. Arduino IDE is an Embedded C software.

**CIRCUIT DIAGRAM**



**EXPLANATION:-**

Connect the 4\*4 Keypad in the 8 digital pins of the board and LCD Display according to the respective pin configurations on the board. This circuit diagram is containing a 4\*3 Keypad as 4\*4 Keypad was not available while preparing this circuit online.

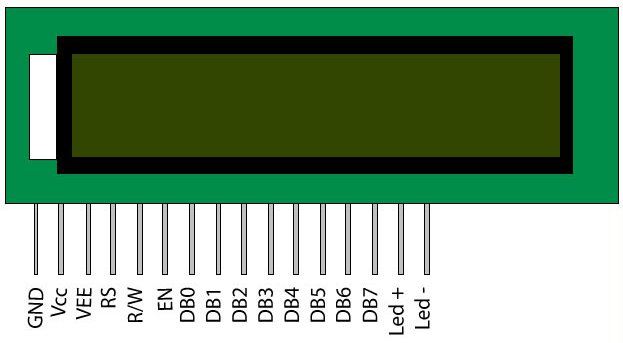
**REQUIREMENT OF HARDWARE & SOFTWARE**

**1. LCD JHD 162A -**

Liquid crystal displays (LCDs) are a commonly used to display data in devices such as calculators, microwave ovens, and many other electronic devices.

The 16x2 LCD used in this experiment has a total of 16 pins. As shown in the table below, eight of the pins are data lines (pins 7-14), two are for power and ground (pins 1 and 16), three are used to control the operation of LCD (pins 4-6), and one is used to adjust the LCD screen brightness (pin 3). The remaining two pins (15 and 16) power the backlight. The details of the LCD terminals are as follows:

|  |  |
| --- | --- |
| Terminal 1 | GND |
| Terminal 2 | +5V |
| Terminal 3 | Mid terminal of potentiometer (for brightness control) |
| Terminal 4 | Register Select (RS) |
| Terminal 5 | Read/Write (RW) |
| Terminal 6 | Enable (EN) |
| Terminal 7 | DB0 |
| Terminal 8 | DB1 |
| Terminal 9 | DB2 |
| Terminal 10 | DB3 |
| Terminal 11 | DB4 |
| Terminal 12 | DB5 |
| Terminal 13 | DB6 |
| Terminal 14 | DB7 |
| Terminal 15 | +4.2-5V |
| Terminal 16 | GND |



**2. 4\*4 Keypad Matrix -**

Keypads are used in many devices, including cell phones, microwaves, door locks etc. They’re practically everywhere. Tons of electronic devices use them for user input. When a key is pressed, it shows up at the Serial Monitor on your computer.

The type of keypad we will use is a matrix keypad. This is a keypad that follows an encoding scheme that allows it to have much less output pins than there are keys. For example, the matrix keypad we are using has 16 keys (0-9, A-D, \*, #), yet only 8 output pins.With a linear keypad, there would have to be 17 output pins (one for each key and a ground pin) in order to work. The matrix encoding scheme allows for less output pins and thus much less connections that have to be made for the keypad to work. In this way, they are more efficient than linear keypads, being that they have less wiring.

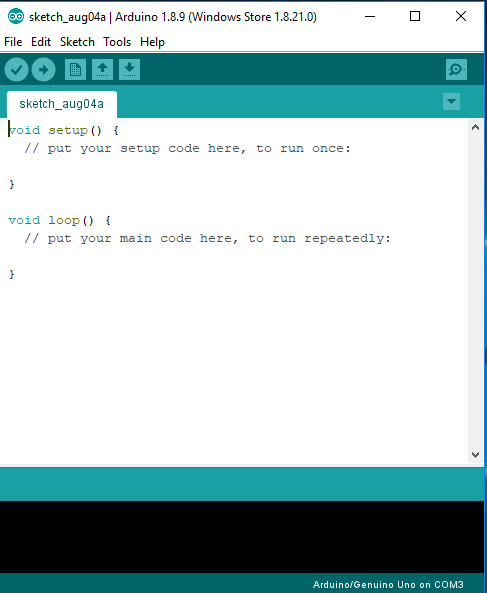


The four row pins are connected to D4, D5, D6, and D7 of the board.

The four column pins are connected to D8, D9, D10, and D11 of the board.

**3. Arduino Software (Version 1.8.9) –**

**Arduino software** is used to put the instructions of the code written to the microcontroller. The IDE is written in ‘Embedded C’ language for communicating with micro-controllers. When signal data arrives the Arduino the pin which corresponds to the particular input is set to high. We have used Arduino IDE version 1.8.9 for writing program for Arduino. It is recommended to keep on checking their official website for any updates in the software as it will help in further projects with less errors and bugs present in the software. There are two functions pre-defined in the software, which are setup () and loop () respectively. The program or condition or any line of code written in setup () section will be only executed once in the entire run of the code. Whereas, the code written in the loop section will run continuously till you remove the power supply or stop the program manually. It is very helpful for real time data applications.



**EXPLANATION OF CODE: -**

1. There is one important issue you may face during the project. If you observe unusual symbols on the LCD screen, then don’t worry about it. There is a simple issue with this problem. If your Serial communication is on, eg. Serial.begin() or Serial.print(), then your sensor reading is getting confused for the output. It will be shown on LCD as well as Serial Monitor. Hence, we get these unusual symbols on the LCD screen.

The simple solution to this problem is to not use Serial communication and display the output only on the LCD Screen. Remove any Serial communication code lines if you used any, and then you will find the perfect working of your code.

2. When you turn on the LCD Screen and upload the code and even afterward when you start up the code, you will get an intro screen and then there will be one unusual symbol between two 0’s on-screen. Don’t worry about that unusual symbol as it will not affect our calculations or calculator.

It is present there since in the very beginning there is no operation specified to be performed, hence, there is an unusual symbol there. Once you input your numbers and your specific operation, that symbol disappears up and until you use the code for your calculations.

3. Another important thing is that you must connect your wires perfectly and tightly. LCD Screen wires can be inserted properly, but it may be difficult with the keypad.

You may even have to push those wires inside the keypad for the entire course of your calculations. You can think about your ways to make your work easier, but just be careful and be precise in the connections!

**RESULT**

