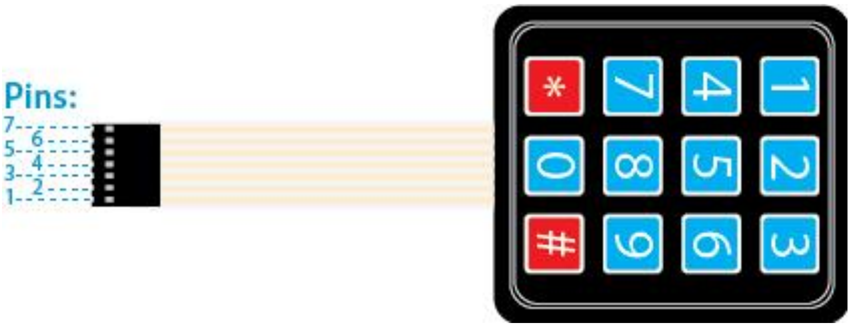
4x3 Matrix 12-Key Keypad Tutorial: Arduino

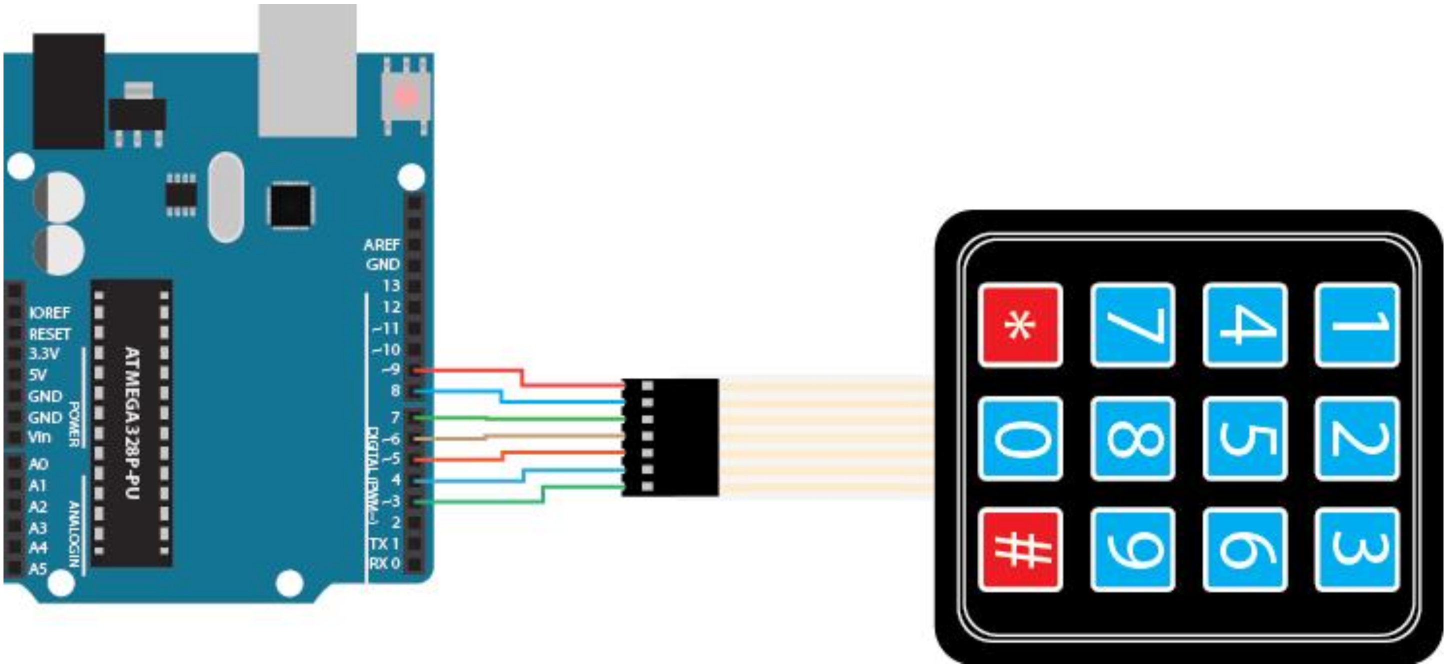
In this tutorial we will be using the 4x3 matrix array 12-key membrane switch keypad, as described at https://www.addicore.com/product-p/142.htm.

Wiring

The following shows the needed connections between the 12-key keypad and the Arduino.

Keypad 12-Key Pins	Arduino Pins
1	3
2	4
3	5
4	6
5	7
6	8
7	9



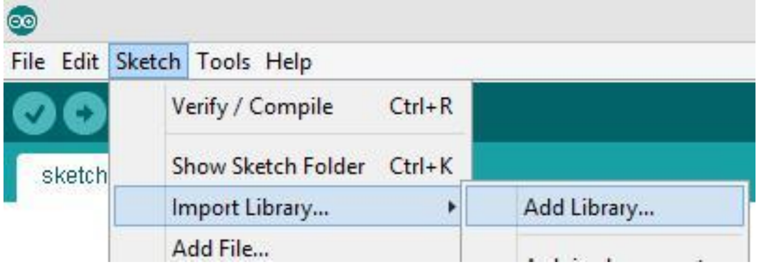


© Addicore LLC 2014 v1.1

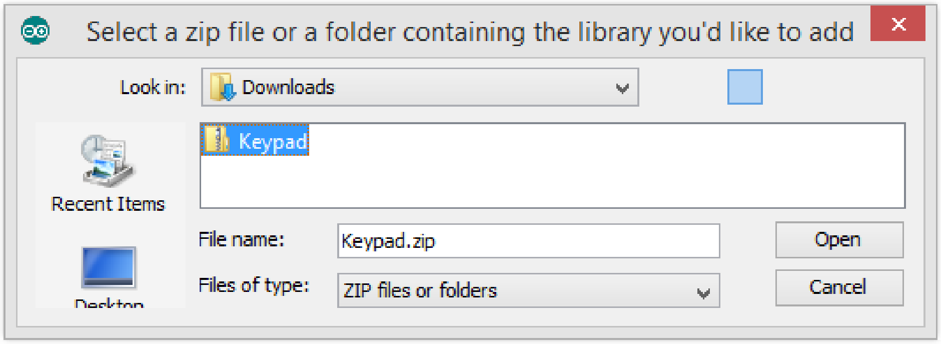
Adding the Library

If you haven’t already done so, the Keypad library needs to be added to your Arduino library depository.

1. Download the Keypad Library
2. Starting with Arduino IDE version 1.0.5 you can automagically ☺ install additional libraries from within the Arduino IDE. If you currently use an Arduino IDE older than version 1.0.5 you can still install the library manually. Arduino provides instructions to do so that can be found at http://arduino.cc/en/Guide/Libraries (Scroll down the page until you find the section on “Manual Installation”).
   Open the Arduino IDE and navigate to *Sketch > Import Library*. A drop down menu will appear. Click the option to “Add Library.”



3. A window will open. Navigate to the location where the Keypad library you downloaded above is currently located and open it.

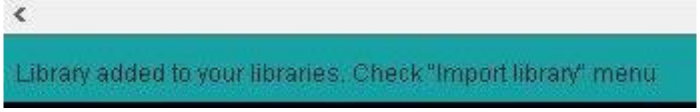

Select a zip file or a folder containing the library you'd like to add

Look in: Downloads

File name: Keypad.zip

Files of type: ZIP files or folders

After the library has been installed the bottom left side of the IDE should show the following message:


Library added to your libraries. Check "Import library" menu

4. You can confirm that the library has been installed by again navigating to *Sketch > Import Library*. The Keypad library should now show in the list of “Contributed” libraries.

Program the Arduino

After installing the Keypad library in the steps above the library will be available to use in sketches but any example sketches included with the library will not be shown in *File > Examples* until after restarting the Arduino IDE.

1. Restart the Arduino IDE by closing all currently open Arduino IDE windows and then opening the Arduino IDE.

2. When the Arduino IDE opens navigate to *File* > *Examples* > *Keypad*. Select the “Addicore_12Key_Keypad” sketch. This will open a sketch which we will use with the Addicore 12-key keypad that we wired to our Arduino earlier.
   A second option instead of opening the example sketch is to copy the code below into a new sketch.

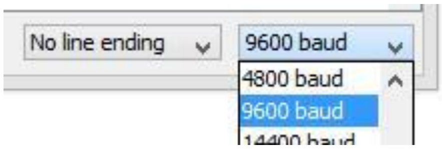
```
/*
|| Example sketch for the Addicore 4x3 matrix array 12-key membrane switch keypad,
|| found at: https://www.addicore.com/product-p/142.htm
|| After connecting your keypad to your Arduino (as detailed below) and uploading
|| this sketch to your Arduino, you can then open the Serial Monitor on your
|| computer and ensure that the baud is set to 9600. Then press any of the buttons
|| on the keypad to see the pressed button show up on the Serial Monitor.
||
|| Wire the keypad to your Arduino using information below:
|| Note: Pins one and seven on the keypad are labeled with little embossed plastic
||      numbers on the black connector
|| Wiring:
||        Keypad ---> Arduino
||          1          3
||          2          4
||          3          5
||          4          6
||          5          7
||          6          8
||          7          9
||
|| Modified by Craig Thompson (craig@addicore.com), December 15, 2014
|| Based on CustomKeypad.pde by Alexander Brevig (alexanderbrevig@gmail.com)
*/
#include <Keypad.h>

const byte ROWS = 4; //four rows
const byte COLS = 3; //four columns
//define the cymbols on the buttons of the keypads
char hexaKeys[ROWS][COLS] = {
  {'1','2','3'},
  {'4','5','6'},
  {'7','8','9'},
  {'*','0','#'}
};
byte rowPins[ROWS] = {9, 8, 7, 6}; //connect to the row pinouts of the keypad
byte colPins[COLS] = {5, 4, 3};    //connect to the column pinouts of the keypad

//initialize an instance of class NewKeypad
Keypad customKeypad = Keypad( makeKeymap(hexaKeys), rowPins, colPins, ROWS, COLS);

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

void loop(){
  char customKey = customKeypad.getKey();
```

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

3. Now connect your Arduino to your computer and upload the code.

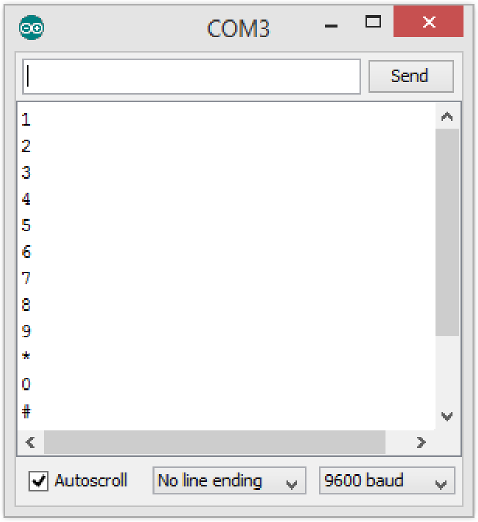
Typing on the keypad and the results

With your Arduino programmed with the “Addicore_12Key_Keypad” sketch and with it still plugged into your computer open the serial monitor by clicking the Serial Monitor icon or pressing *CTRL+SHIFT+M*.
If it isn’t already selected set the serial baud rate to 9600 baud by clicking on the drop down selection in the lower right corner of the Serial Monitor window as shown below and selecting 9600 baud.



Now press any of the keys on the keypad and see what happens on the serial monitor.

Your Serial Monitor should show something similar to the following:



```
1
2
3
4
5
6
7
8
9
*
0
#
```

Changing what the keys represent

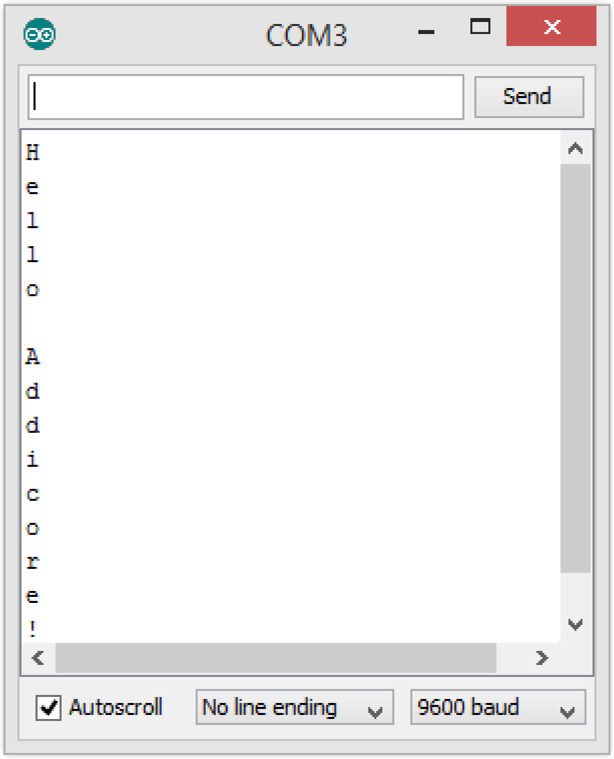
You can change which characters each key on your keypad will represent by changing the following body of code in the “Addicore_12Key_Keypad” sketch:

```
//define the symbols on the buttons of the keypads
char hexaKeys[ROWS][COLS] = {
  {'1','2','3'},
  {'4','5','6'},
  {'7','8','9'},
  {'*','0','#'}
};
```

For example let’s change it to the following:

```
//define the symbols on the buttons of the keypads
char hexaKeys[ROWS][COLS] = {
  {'A','d','i'},
  {'c','o','r'},
  {'e','!','H'},
  {' ','l','#'}
};
```

Now we can type the following sequence of keys on the keypad: 9 7 0 0 5 * 1 2 2 3 4 5 6 7 8 which will create the message “*Hello Addicore!*” on the Serial Monitor as shown below:



```
H
e
l
l
o

A
d
d
i
c
o
r
e
!
```

☑ Autoscroll   No line ending   9600 baud

*Don’t forget to set the correct baud rate on the serial monitor.

Notes & Comments by the Author

This guide has been as simple as I could possibly make it and still have a working setup. This is by no means an exhaustive explanation or use of the module; rather, this is meant to get you started.
Any errors found in this document can be sent to craig@addicore.com.

Reference

The following resources are ones that we found useful.
http://playground.arduino.cc/code/Keypad

Copyright