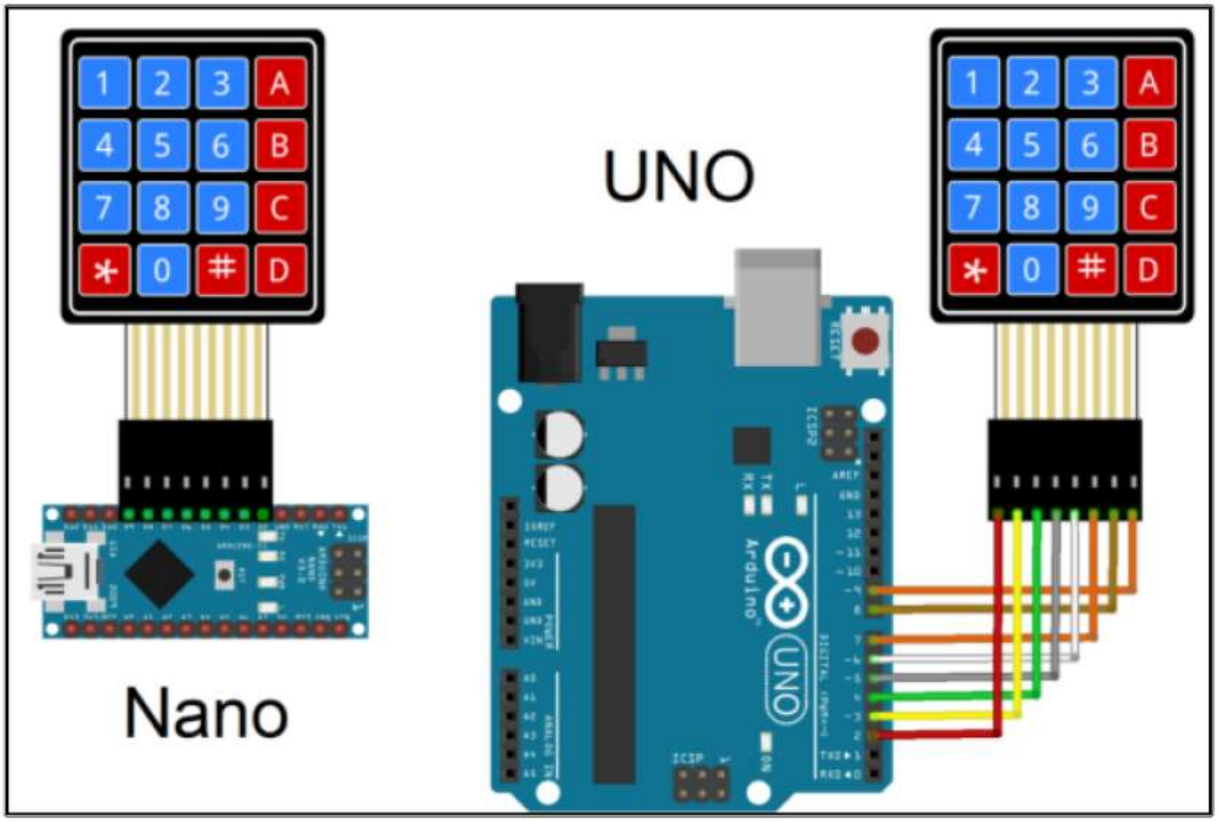
Al-Mustaqbal University
Department of Medical Instrumentation Techniques Engineering
Class: four
Subject: Advanced logic design
Lecturer: Dr. Zahraa hashim kareem
Lecture- 5: Matrix keypad interface with Arduino

Matrix keypad interface with Arduino



Reading a Keypad

Problem

You have a matrix keypad and want to read the key presses in your sketch. For example, you have a telephone-style keypad similar to the SparkFun 12-button keypad.

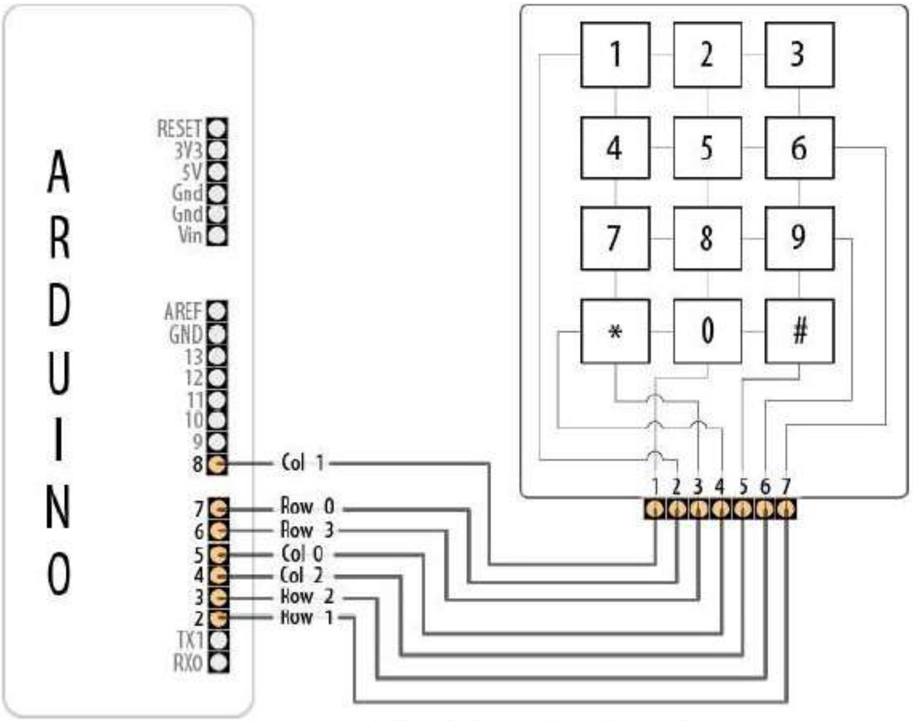
Connecting the SparkFun keyboard matrix

If you ve wired your Arduino and keypad as shown in the above figure, the following sketch will print key presses to the Serial Monitor:

```
/*
Keypad sketch
prints the key pressed on a keypad to the serial port
*/
const int numRows = 4; // number of rows in the keypad
const int numCols = 3; // number of columns
```

Solution

Wire the rows and columns from the keypad connector to the Arduino, as shown in figure below.



Connecting the SparkFun keyboard matrix

If you ve wired your Arduino and keypad as shown in the above figure, the following sketch will print key presses to the Serial Monitor:

```
/*
Keypad sketch
prints the key pressed on a keypad to the serial port
*/
const int numRows = 4; // number of rows in the keypad
const int numCols = 3; // number of columns
```

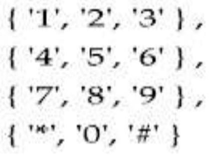
Solution

Wire the rows and columns from the keypad connector to the Arduino, as shown in figure below.

```
const int debounceTime = 20; // number of milliseconds for switch to be stable

// keymap defines the character returned when the corresponding key is pressed

const char keymap[numRows][numCols] = {
```

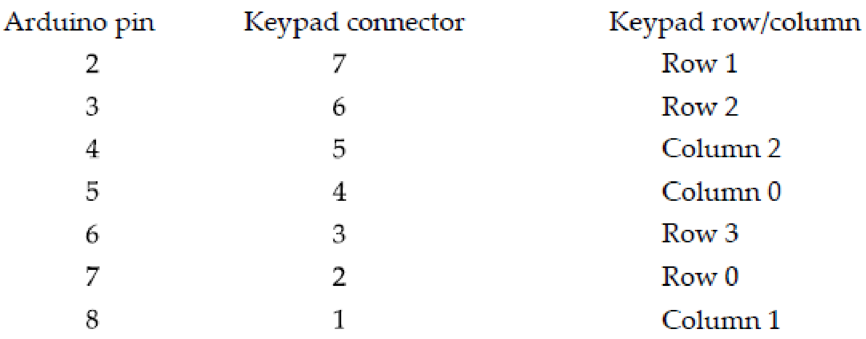


```
{ '1', '2', '3' } ,
{ '4', '5', '6' } ,
{ '7', '8', '9' } ,
{ '*', '0', '#' }
```

```
};
// this array determines the pins used for rows and columns
const int rowPins[numRows] = { 7, 2, 3, 6 }; // Rows 0 through 3
const int colPins[numCols] = { 5, 8, 4 }; // Columns 0 through 2
void setup()
{
Serial.begin(9600);
for (int row = 0; row < numRows; row++)
{
pinMode(rowPins[row],INPUT); // Set row pins as input
digitalWrite(rowPins[row],HIGH); // turn on Pull-ups
}
for (int column = 0; column < numCols; column++)
{
pinMode(colPins[column],OUTPUT); // Set column pins as outputs for writing
digitalWrite(colPins[column],HIGH); // Make all columns inactive
}
}
void loop()
{
char key = getKey();
if( key != 0) { // if the character is not 0 then it's a valid key press
Serial.print("Got key ");
Serial.println(key);
}
}
// returns with the key pressed, or 0 if no key is pressed
char getKey()
{
char key = 0; // 0 indicates no key pressed
for(int column = 0; column < numCols; column++)
{
digitalWrite(colPins[column],LOW); // Activate the current column.
for(int row = 0; row < numRows; row++) // Scan all rows for a key press.
{
if(digitalRead(rowPins[row]) == LOW) // Is a key pressed?
{
delay(debounceTime); // debounce
while(digitalRead(rowPins[row]) == LOW)
; // wait for key to be released
key = keymap[row][column]; // Remember which key was pressed.
}
}
digitalWrite(colPins[column],HIGH); // De-activate the current column.
}
return key; // returns the key pressed or 0 if none
}
```

This sketch will only work correctly if the wiring agrees with the code. Table below shows how
the rows and columns should be connected to Arduino pins. If you are using a different keypad,
check your data sheet to determine the row and column connections. Check carefully, as incorrect wiring can short out the pins, and that could damage your controller chip.

Mapping Table of Arduino pins to SparkFun connector and keypad rows and columns.



Arduino pin	Keypad connector	Keypad row/column
2	7	Row 1
3	6	Row 2
4	5	Column 2
5	4	Column 0
6	3	Row 3
7	2	Row 0
8	1	Column 1