

ENGINE



Erasmus+

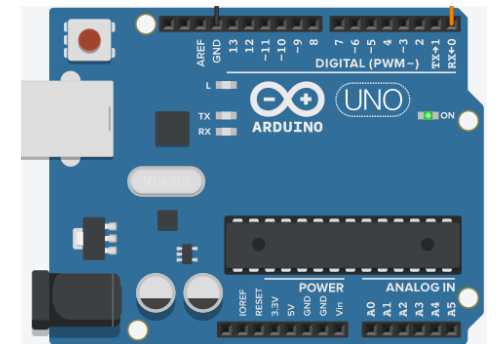
TEACHING ONLINE ELECTRONICS, MICROCONTROLLERS AND PROGRAMMING
IN HIGHER EDUCATION

Module_1-5. Keypad 4x4

Arduino Uno with Tinkercad

Contents

- Keypad 4x4 layout
- Programming functions for the Arduino Uno
- Example



Module_1-5. Keypad 4x4

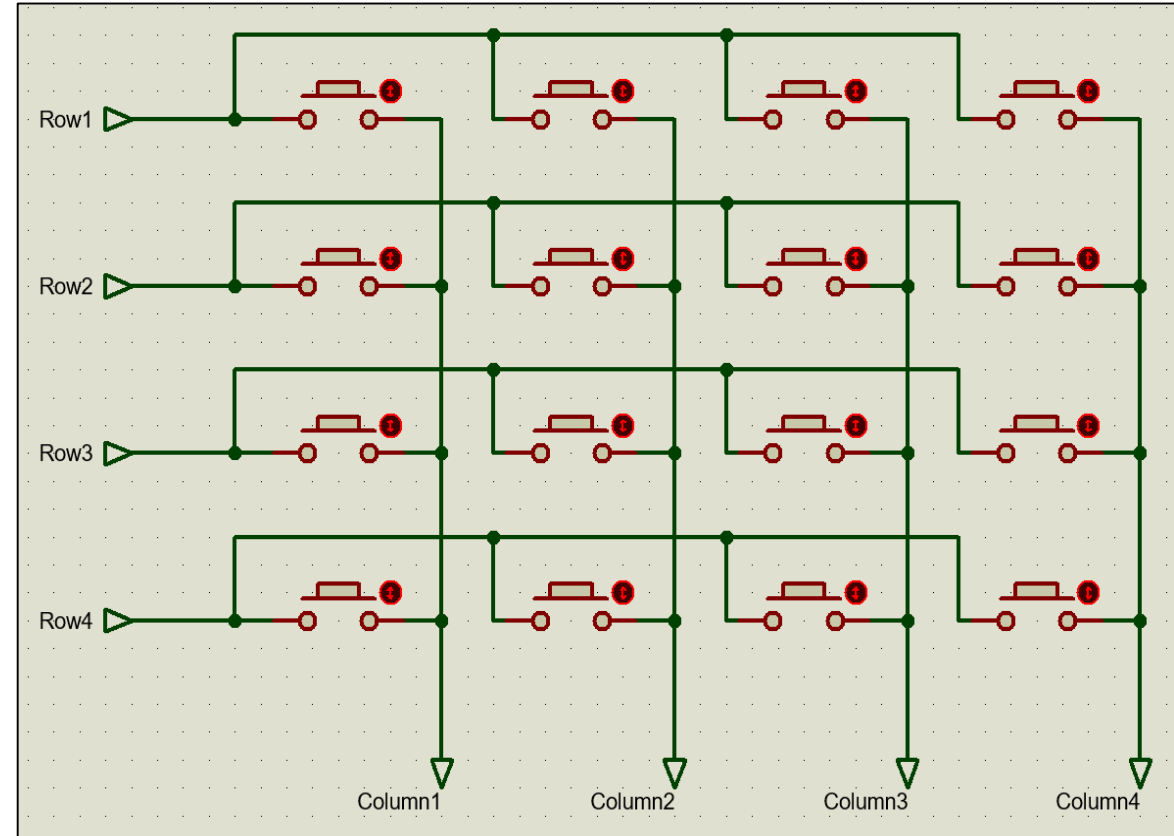
Keypad layout

A keypad 4x4, consists of:

- 4 columns
- 4 rows
- 16 keys



Keypad 4x4



Pinout of Keypad 4x4

1. <https://www.electronicwings.com/sensors-modules/4x4-keypad-module>
2. <https://www.circuitbasics.com/how-to-set-up-a-keypad-on-an-arduino/>

Module_1-5. Keypad 4x4

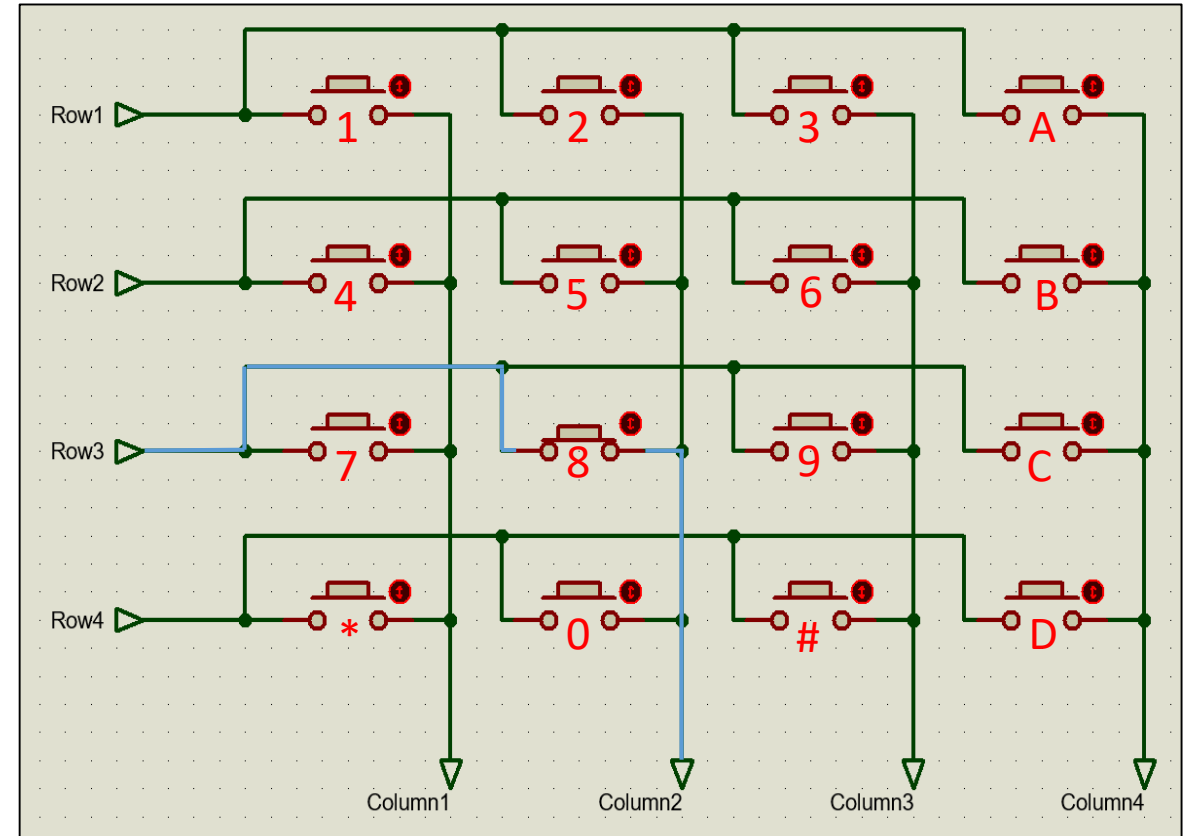
Keypad layout

The keypad rows go to the input pins of the Arduino Uno with the pull-up resistors activated.

Keypad columns go to Arduino Uno output pins.

The Arduino Uno sends sequential signals to the columns, and reads the rows.

For example: if 8 is pressed, row 3 will be connected with column 2.



Pinout of Keypad 4x4

1. <https://www.electronicwings.com/sensors-modules/4x4-keypad-module>
2. <https://www.circuitbasics.com/how-to-set-up-a-keypad-on-an-arduino/>

Module_1-5. Keypad 4x4 Programming functions

Functions that can be used on the Arduino Uno as we have seen :

- `pinMode(pin, value), delay(value), millis()`
- `digitalRead(pin), digitalWrite(pin, value)`
- `lcd.begin(cols, rows), lcd.print()`

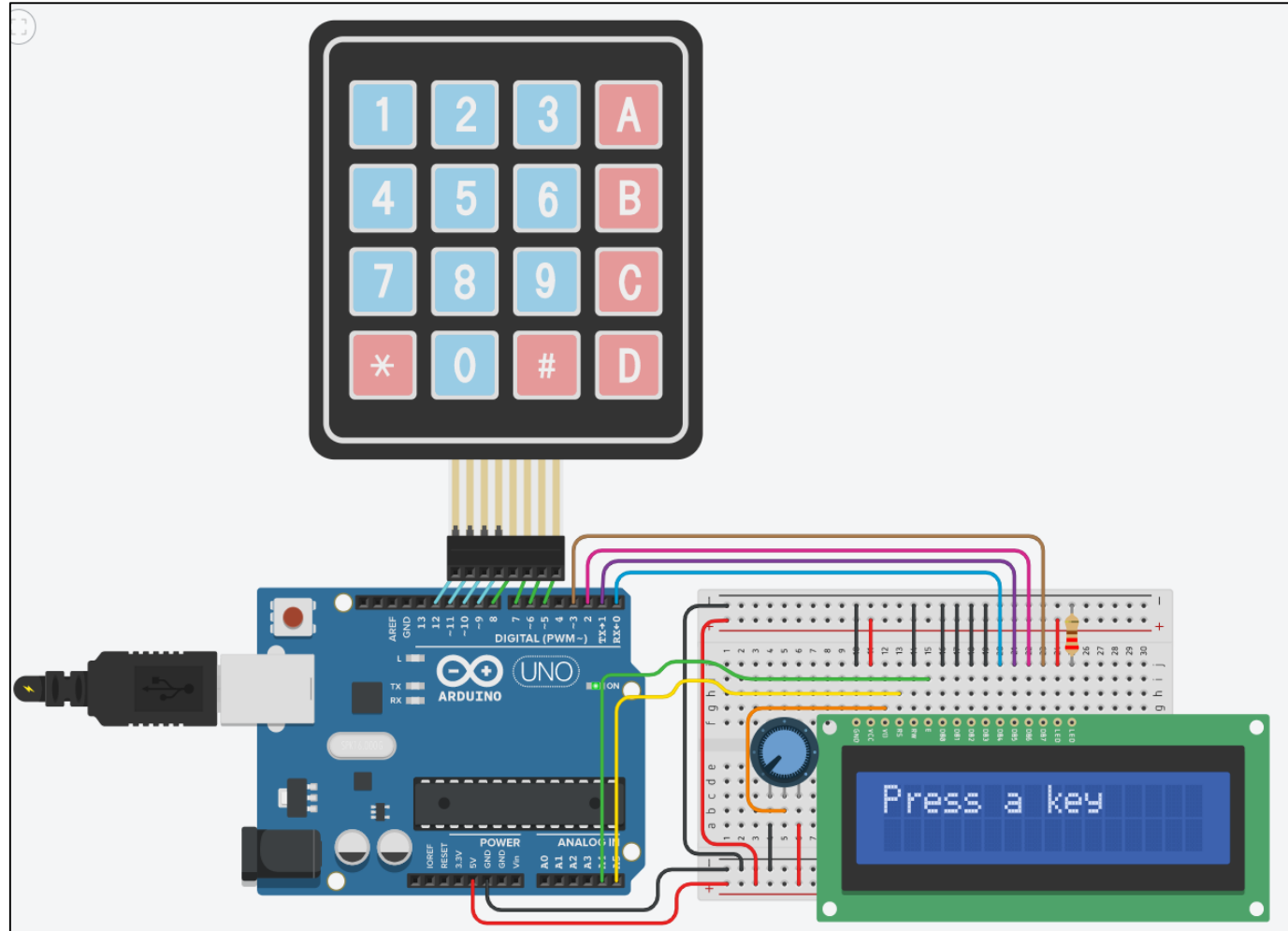
New functions :

- `keypad.waitForKey()`: the program stays here until a key is pressed. The function then returns the key character
- `keypad.getKey()`: if a key is pressed, it returns its character

1. <https://playground.arduino.cc/Code/Keypad/>
2. <https://www.arduino.cc/en/Reference/LiquidCrystal>

Module_1-5. Keypad 4x4 Example

The example uses a keypad 4x4 and a LCD 16x2 to display the key that pressed



Circuit connections

Module_1-5. Keypad 4x4

Example

The code:

```
/* Keypad and LCD

Circuit Connections:
** LCD
Ground      => Gnd
Power       => Vcc
Contrast    => Potentiometer
RS          => PIN_0
RW         => Gnd
E          => PIN_1
DB0        => Gnd
DB1        => Gnd
DB2        => Gnd
DB3        => Gnd
DB4        => PIN_2
DB5        => PIN_3
DB6        => PIN_4
DB7        => PIN_5
LED Anode  => Vcc
LED Cathode => Resistor 220Ω => Gnd
** Potentiometer1
Terminal 1  => Gnd
Wiper      => LCD_Contrast
Terminal 2  => Vcc
**Keypad
PIN_5      => Column4
PIN_6      => Column3
PIN_7      => Column2
PIN_8      => Column1
PIN_9      => Row4
PIN_10     => Row3
PIN_11     => Row2
PIN_12     => Row1
*/

//include the library
#include <LiquidCrystal.h>
#define RS A5 //give the name "RS" to PIN_A5
#define EN A4 //give the name "EN" to PIN_A4
#define DB4 0 //give the name "DB4" to PIN_0
#define DB5 1 //give the name "DB5" to PIN_1
#define DB6 2 //give the name "DB6" to PIN_2
#define DB7 3 //give the name "DB7" to PIN_3

//configure the Arduino Uno - LCD interface
LiquidCrystal lcd(RS, EN, DB4, DB5, DB6, DB7);

//include the library
#include <Keypad.h>

const byte cols = 4; //four columns
const byte rows = 4; //four rows

//keypad output
char keys[rows][cols] = {
  {'1','2','3','A'},
  {'4','5','6','B'},
  {'7','8','9','C'},
  {'*','0','#','D'}
};

//configure the Arduino Uno - Keypad interface
byte row_pins[] = {12,11,10,9}; //pins connect to the rows
byte col_pins[] = {8, 7, 6, 5}; //pins connect to the columns
Keypad keypad = Keypad(makeKeymap(keys), row_pins,
col_pins, rows, cols);

//variable to save keypad's characters
char key;

//The setup() function initializes and sets the initial values
//It will only run once after each power up or reset
void setup(){
  //configure the LCD's columns and rows
  lcd.begin(16, 2);
  //print a message
  lcd.print("Press a key");
}

//loops consecutively
void loop(){
  //wait until a key is pressed
  key=keypad.waitForKey();
  //clear the LCD and print the key
  lcd.clear();
  lcd.print(key);
}
```

ENGINE Partnership

- Warsaw University of Technology (PL) - *coordinator*
- IHU - International Hellenic University (GR)
- EDUMOTIVA - European Lab for Educational Technology (GR)
- University of Padova (IT)
- University of Applied Sciences in Tarnow (PL)



INTERNATIONAL
HELLENIC
UNIVERSITY



UNIVERSITÀ
DEGLI STUDI
DI PADOVA



CONTACT:



www.engined.eu



angelika.tefelska@pw.edu.pl



[@projectENGINE1](https://twitter.com/projectENGINE1)



[@EUprojectEngine](https://www.facebook.com/EUprojectEngine)



Erasmus+

This project has been funded with support from the European Commission. This publication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.