

# 4x3 matrix keypad

**Produktkode:** 278aa

**Tilgjengelighet:** 3

**Pris:** kr. 20,00



## Short Description

Matrix Array 12 Key Membrane Switch Keypad Keyboard For Arduino AVR 4 x 3 CC

## Beskrivelse

Features:

Contact resistance of 500 ( $\Omega$ )

Insulation resistance 100M ( $\Omega$ )

Key Operating Force 150-200N

Rebound time 1 (ms)

Life of 100 million (times)

Operating Temperature 60 degree

The electronic characteristics

Circuit Rating: 35V (DC), 100mA, 1W

Contact resistance: 10 $\Omega$  ~ 500 $\Omega$  (Varies according to the lead lengths and different from those of the material used)

Insulation resistance: 100M $\Omega$  100V

Dielectric Strength: 250VRms (50 ~ 60Hz 1min)

<http://playground.arduino.cc/Main/KeypadTutorial>

[Bibliotek](#)

## Skisse 1

```
#include
```

```
const byte ROWS = 4; //four rows  
const byte COLS = 3; //three columns
```

```
char keys[ROWS][COLS] = {  
  {'1','2','3'},  
  {'4','5','6'},  
  {'7','8','9'},  
  {'#','0','*'}  
};
```

```
byte rowPins[ROWS] = {5, 4, 3, 2}; //connect to the row pinouts of the keypad  
byte colPins[COLS] = {8, 7, 6}; //connect to the column pinouts of the keypad
```

```
Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );
```

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

```
void loop(){  
  char key = keypad.getKey();  
  
  if (key != NO_KEY){  
    Serial.println(key);  
  }  
}
```

## Skisse 2

```
/*  
|| Written by Dougie Kinnear November 2012  
|| building on Nathan Sobieck's|| Simple Password Entry Using Matrix Keypad  
|| 4/5/2012 Updates Nathan Sobieck: Nathan@Sobisource.com|| I used a Textstar LCD  
for visul output. The code may need changed  
|| for other LCD's. Control codes may also appear in the  
|| serial monitor. See the Textstar data sheet for control codes|| This sketch is discussed on  
my blog  
|| http://www.digitalmaker.co.uk/86/ ?  
|| look under the tags PIN, Textstar, Keypad  
|| Please get in touch with any comments, suggestions etc.|| Many thanks to Nathan for  
sharing his work
```

```

*/
// # is to validate password////////////////////////////////////#include
//http://www.arduino.cc/playground/uploads/Code/Password.zip
#include //http://www.arduino.cc/playground/uploads/Code/Keypad.zipint greenled(11);
int redled(12);//----- Keypad set up/mapping -----Password passwo
Password("1234"); // Set the password hereconst byte ROWS = 4; // Four rows
const byte COLS = 3; // columns
char keys[ROWS][COLS] = {
  { '1?', '2?', '3' },
  { '4?', '5?', '6' },
  { '7?', '8?', '9' },
  { '*', '0?', '#' }
};byte rowPins[ROWS] = { 8,7,6,5}; // Assign board pins to keypad rows
byte colPins[COLS] = { 4,3,2}; // Assign board pins to keypad columnsKeypad keypad =
Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );
//-----void setup(){
{
pinMode(11, OUTPUT); //green light
pinMode(12, OUTPUT); //red light
Serial.begin(9600);
}

keypad.addEventListener(keypadEvent); //add an event listener for this
keypaddelay(100); // A further delay to allow the LCD to initialiseSerial.write(254); //
Tell LCD multiple command characters are coming
Serial.write(67); // Send a 'C' = Cursor set up command
Serial.write(4); // Set cursor to flashing underlinedigitalWrite(12, HIGH); // red LED on
delay(2000); // The LCD should be ready now – these delays can be
alteredSerial.write(12); // This is a command to clear the screen and set the cursor to Line
1 column 1
Serial.println("Enter a PIN then"); // Print the text to the screen
Serial.println("press '#'"); // Print the text to the screen}void loop(){
keypad.getKey();
} void keypadEvent(KeypadEvent eKey){ // This is what happens during a keypad
eventswitch (keypad.getState()){
case PRESSED:

Serial.print(eKey); // When a key is pressed * is printedswitch (eKey){

case '#': checkPassword(); break; // press # to check entry
default: password.append(eKey);
}
}

```

```
}  
}void checkPassword(){  
  
if (password.evaluate()){ // If it's correct do this  
  
Serial.write(12);  
Serial.println("CORRECT");  
digitalWrite(12, LOW);  
digitalWrite(11, HIGH);  
  
delay(3000);  
  
digitalWrite(11, LOW);  
digitalWrite(12, HIGH);  
password.reset(); // reset for next event  
  
Serial.println("Enter a PIN then"); // Print the text to the screen  
Serial.println("press '#"); // Print the text to the screen  
  
}  
  
else{  
  
Serial.write(12); // If it's wrong do this  
  
Serial.println("INCORRECT");  
Serial.println("TRY AGAIN");  
  
delay(50); // flash the red LED  
digitalWrite(12,LOW);  
delay(50);  
digitalWrite(12, HIGH);  
delay(50);  
digitalWrite(12,LOW);  
delay(50);  
digitalWrite(12, HIGH);  
delay(50);  
digitalWrite(12,LOW);  
delay(50);  
digitalWrite(12, HIGH);  
delay(50);  
digitalWrite(12,LOW);
```

```
delay(50);
digitalWrite(12, HIGH);
delay(50);
digitalWrite(12,LOW);
delay(50);
digitalWrite(12, HIGH);
delay(50);
digitalWrite(12,LOW);
delay(50);
digitalWrite(12, HIGH);
delay(50);
digitalWrite(12,LOW);
delay(50);
digitalWrite(12, HIGH);
delay(50);
digitalWrite(12,LOW);
delay(50);
digitalWrite(12, HIGH);
delay(50);
digitalWrite(12,LOW);
delay(50);
digitalWrite(12, HIGH);
delay(50);
digitalWrite(12,LOW);
delay(50);
digitalWrite(12, HIGH);
delay(50);
digitalWrite(12,LOW);
delay(50);
digitalWrite(12, HIGH);
delay(50);
digitalWrite(12,LOW);
delay(50); // stop flashing the red LED

digitalWrite(12, HIGH);
password.reset(); // reset for next event

delay(2500);
```

```
Serial.println("Enter a PIN then"); // Print the text to the screen
Serial.println("press '#'"); // Print the text to the screen
}
```

## Product Gallery

