

# Datenblatt RBS11088

## 4x4 Keypad Modul



### Inhaltsverzeichnis

Technische Daten .....	2
Funktionsweise.....	2
Schaltplan .....	2
Pinbelegung.....	3
Anschlussplan.....	3
Code für Arduino.....	4

## Technische Daten

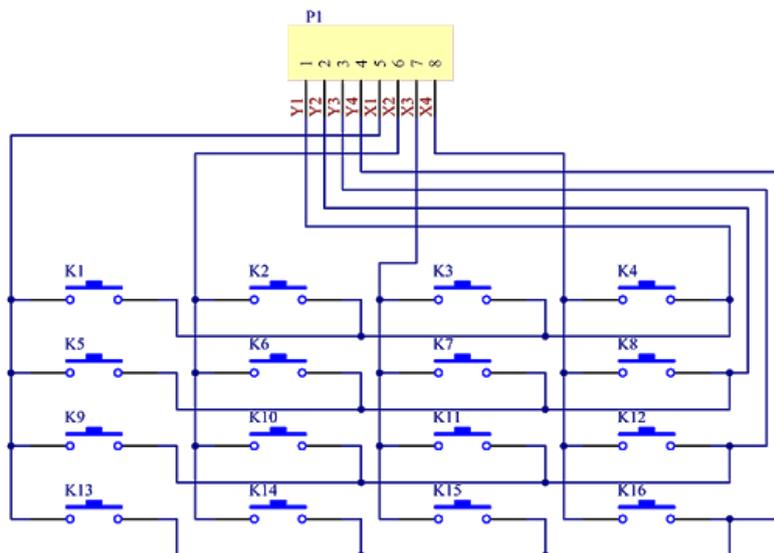
- Maximale Betriebsspannung: DC 24V
- Maximaler Betriebsstrom: 20mA
- Kontaktwiderstand: max 200 Ohm
- Druckwiderstand: ca. 180g
- Lebensdauer: > 1 Millionen Betätigungen
- Abmessungen: 65 x 63 x 10 mm (ohne Pins)
- Montagebohrungen: 61x61mm

## Funktionsweise

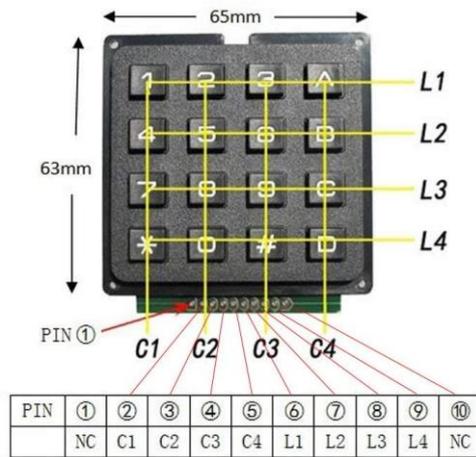
Das numerische Tastenfeld besteht aus 16 Tasten, die in einer Matrix angeordnet sind, d. h. alle Tasten einer Spalte sind mit einem Eingang und alle Tasten einer Zeile mit einem anderen verbunden.

Wenn eine Taste gedrückt wird, wird der entsprechende Eingang der Zeile mit dem der Spalte verbunden, wodurch der Stromkreis geschlossen wird. Der Vorteil dieses Aufbaus ist, dass 16 Tasten mit nur 8 Eingängen des Mikrocontrollers angesteuert werden können.

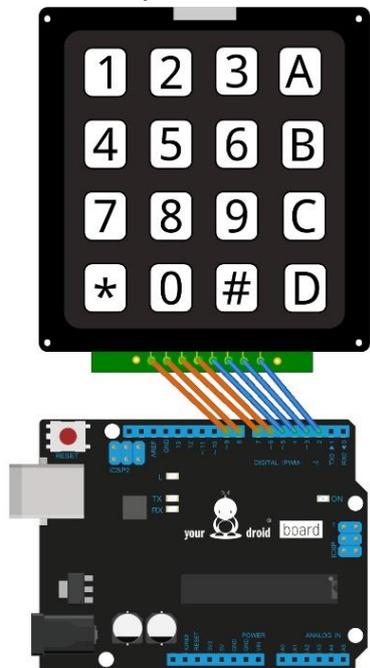
## Schaltplan



## Pinbelegung



## Anschlussplan



fritzing

Keypad	Arduino
C1	Pin 9
C2	Pin 8
C3	Pin 7
C4	Pin 6
L1	Pin 5
L2	Pin 4
L3	Pin 3
L4	Pin 2

## Code für Arduino

Zur Verwendung von diesem Beispielcode wird die Keypad.h Bibliothek benötigt.

Benötigte Komponenten:

- Arduino Board
- USB-Kabel
- 4x4 Keypad
- 8 Jumperkabel

Mit diesem Beispiel können Sie eine Taste drücken und die gedrückte Taste im seriellen Monitor der Arduino IDE auslesen.

```
#include "Arduino.h"
#include "Keypad.h"
const byte ROWS = 4; // vier Reihen
const byte COLS = 4; // vier Spalten
char keys[ROWS][COLS] = {
  {'1','2','3','A'},
  {'4','5','6','B'},
  {'7','8','9','C'},
  {'*','0','#','D'}
};
byte rowPins[ROWS] = {9,8,7,6}; // Pins fuer die Reihen
byte colPins[COLS] = {5,4,3,2}; // Pins fuer die Spalten
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);
  }
}
```