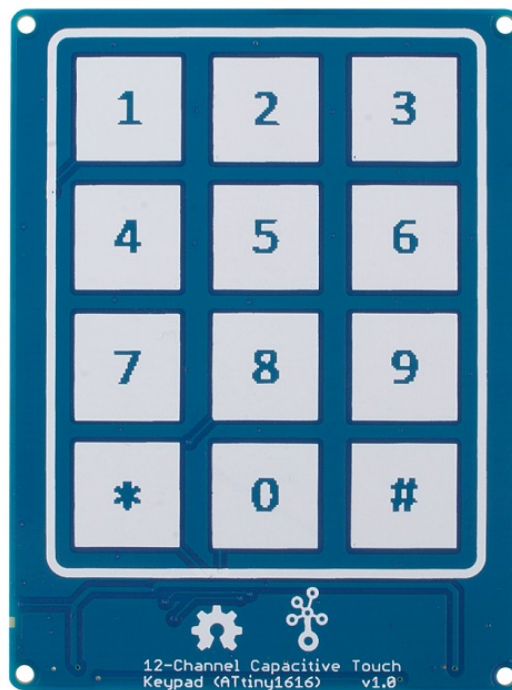


# Grove - 12 Channel Capacitive Touch Keypad (ATtiny1616)



The Grove 12 button Capacitive Touch Keypad is built around the ATtiny1616, an AVR® 8-bit processor running at up to 16 MHz. ATtiny1616 is a low-power, high-performance chip integrated QTouch® peripheral touch controller which supports capacitive touch interfaces with proximity sensing and driven shield. With this

module, you can easily create an arduino password keypad or a DIY phone keypad.

We made this keypad into a 3x4 form, just like the layout of a mobile phone keyboard. The traditional keypad requires 3 vertical lines and 4 horizontal lines to scan, which will occupy 7 I/O pins of the microcontroller. With the help of ATtiny1616 and Grove connector, only RX and TX two pins are enough for Grove - 12-Channel Capacitive Touch Keypad. You can easily use this module with a microcontroller with a hardware UART interface, or you can use the software UART to read the button input with two normal I/O pins.

All in all, the Grove 12 button Capacitive Touch Keypad is an easy-to-use module that requires very little code, especially when you use it with [Grove compatible mainboards](#) [<https://www.seeedstudio.com/seeeduino-boards-c-987.html>], no soldering, just plug and play.

[Get One Now !\[\]\(339a16584d5da0f0a3ca4e9ec17bf6a1\_img.jpg\)](#)

[<https://www.seeedstudio.com/Grove-12-Channel-Capacitive-Touch-Keypad-ATtiny1616-p-4068.html>]

## Features

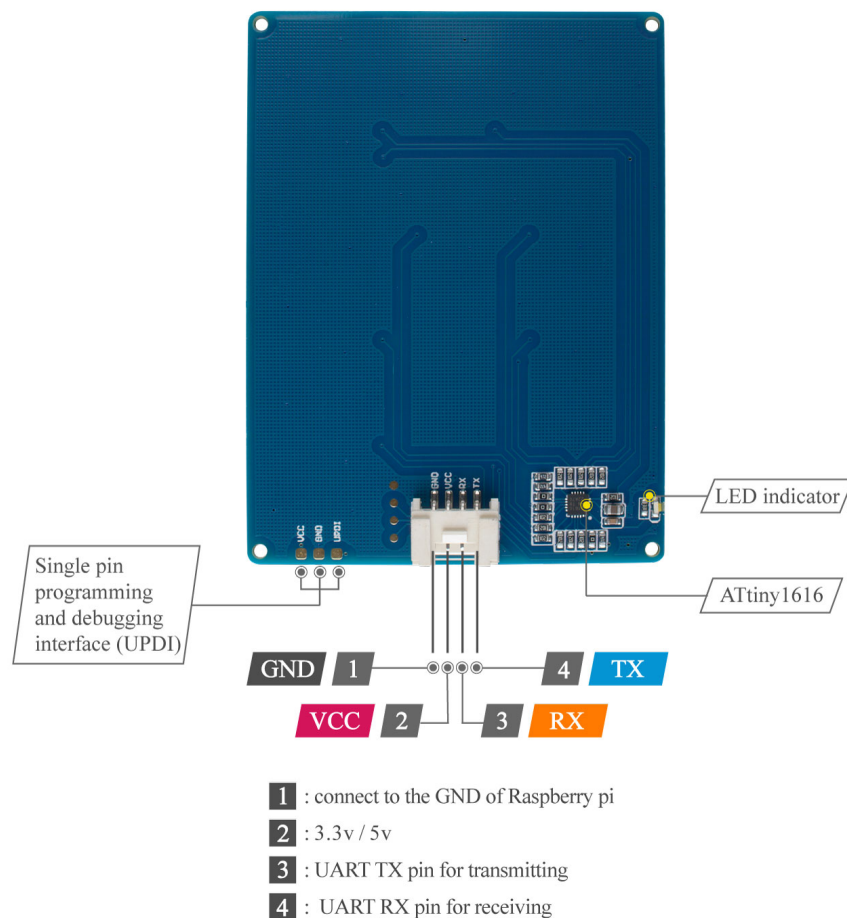
- Low Power ATtiny1616 controller
- 3.3V / 5V compatible
- Capacitive touch, high sensitivity
- 12 button keypad

- 4 pin Grove UART connector
- On-board LED indicator

## Applications

- Phone keypad
- Password access
- Extended input interface

## Pinout



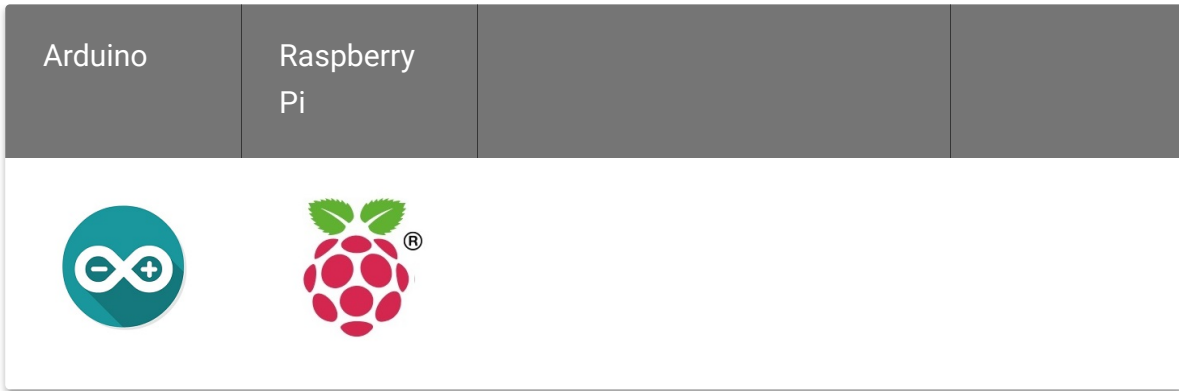
### Figure 1. Hardware overview

[<https://files.seeedstudio.com/wiki/Grove-12-Channel-Capacitive-Touch-Keypad-ATtiny1616/img/pinmap.jpg>]

## Specification

Parameter	Value
Supply voltage	3.3V / 5V
CPU	AVR® 8-bit CPU @ 16MHz
Sensor type	Capacitive Touch Keypad
Button Quantity	12
Operate temperature Range	-40°C to 105°C
Output Interface	UART
Firmware Download Interface	UPDI

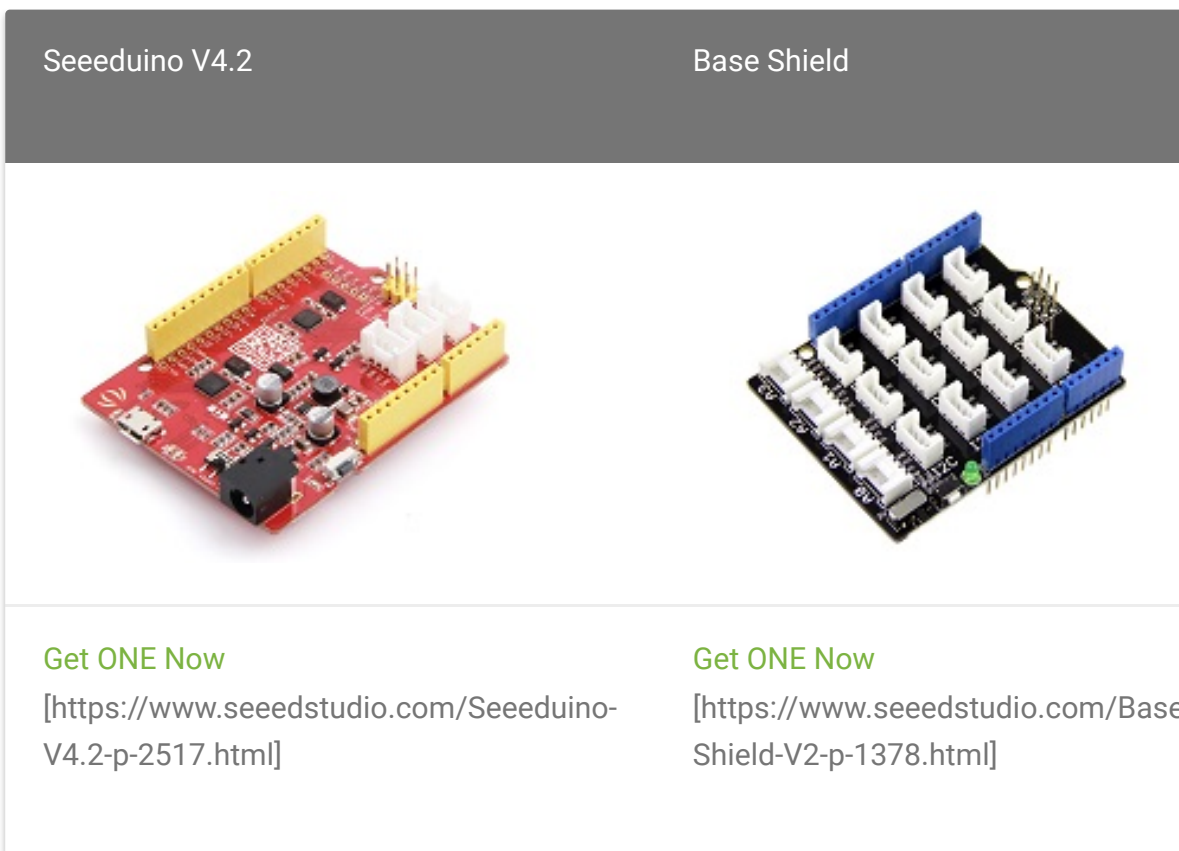
## Platforms Supported



## Getting Started

### Play With Arduino

#### Materials required



In addition, you can consider our new [Seeeduino Lotus M0+](https://www.seeedstudio.com/Seeeduino-Lotus-Cortex-M0-p-2896.html) [https://www.seeedstudio.com/Seeeduino-Lotus-Cortex-M0-p-2896.html], which is equivalent to the combination of Seeeduino V4.2 and Baseshield.



#### Note

**1** Please plug the USB cable gently, otherwise you may damage the port. Please use the USB cable with 4 wires inside, the 2 wires cable can't transfer data. If you are not sure about the wire you have, you can click [here](https://www.seeedstudio.com/Micro-USB-Cable-48cm-p-1475.html) [https://www.seeedstudio.com/Micro-USB-Cable-48cm-p-1475.html] to buy

**2** Each Grove module comes with a Grove cable when you buy. In case you lose the Grove cable, you can click [here](https://www.seeedstudio.com/Grove-Universal-4-Pin-Buckled-20cm-Cable-%285-PCs-pack%29-p-936.html) [https://www.seeedstudio.com/Grove-Universal-4-Pin-Buckled-20cm-Cable-%285-PCs-pack%29-p-936.html] to buy.

## Hardware Connection

- **Step 1.** Connect the Grove 12 Channel Capacitive Touch Keypad to the **D2** port of the Base Shield.

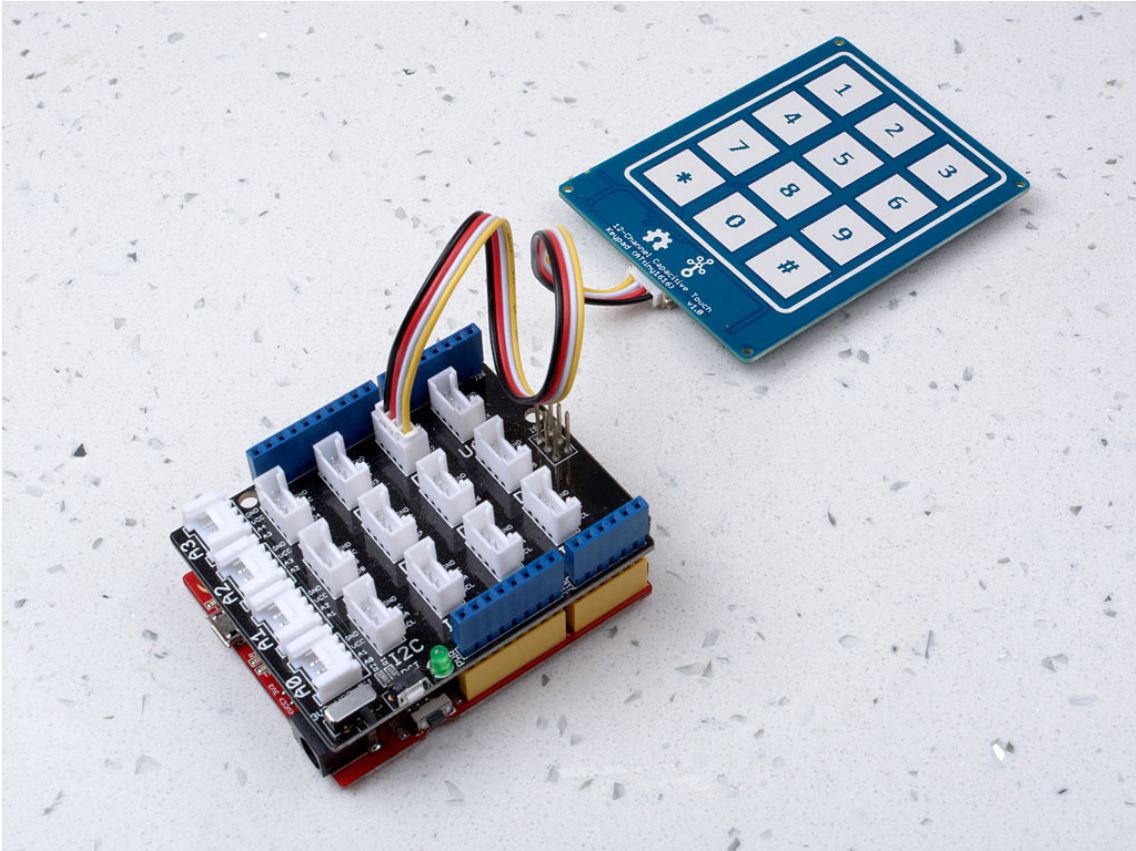
If you are using a SAM board, then you should connect to the **UART** port. Please refer to table 1 for more detail

Board	UART Type	Connect Port
AVR Board(like Seeeduino V4.2	Soft UART	D2,D3
SAM Board(like Seeeduino Lotus M0+	Hardware UART	UART

**Table 1.**UART Port Selection

- **Step 2.** Plug Grove - Base Shield into Seeeduino.

- **Step 3.** Connect Seeduino to PC via a USB cable.



## Software



### Attention

If this is the first time you work with Arduino, we strongly recommend you to see [Getting Started with Arduino](https://wiki.seeedstudio.com/Getting_Started_with_Arduino/) [https://wiki.seeedstudio.com/Getting\_Started\_with\_Arduino/] before the start.

- **Step 1.** Download the [12\\_Channel\\_Keypad.ino](https://github.com/Seeed-Studio/Seeed_ATtiny1616/blob/master/12_Channel_Keypad.ino) [https://github.com/Seeed-Studio/Seeed\_ATtiny1616/blob/master/12\_Channel\_Keypad.ino] from Github.
- **Step 2.** Then open `12_Channel_Keypad.ino`, or you can just copy the following code into a new Arduino sketch.

```
1  #include "SoftwareSerial.h"
2
3  #if defined(ARDUINO_ARCH_AVR)
4  #define SERIAL Serial
5  SoftwareSerial mySerial(2,3);
6  #define TRANS_SERIAL mySerial
7  #elif defined(ARDUINO_ARCH_SAMD)
8  #define SERIAL SerialUSB
9  #define TRANS_SERIAL Serial
10 #else
11
12 #endif
13
14
15 //RX=2,TX=3(D2) Software Serial Port
16
17 void setup() {
18     TRANS_SERIAL.begin(9600);
19
20     SERIAL.begin(9600); // start serial for output
21     SERIAL.println("Version:v1.0");
22 }
23
24 void loop() {
25     printData();
26 }
27
28 /*
29 * data mapping:E1---1; E2---2; E3---3; E4---4; E5---5; E6
30 *               E7---7; E8---8; E9---9; EA---*; EB---0; EC
31 */
32 void printData() {
33     while(TRANS_SERIAL.available()) {
34         uint8_t data = TRANS_SERIAL.read();
35         switch(data) {
36             case 0xE1 :
37                 SERIAL.println("1");
38                 break;
39             case 0xE2 :
40                 SERIAL.println("2");
```





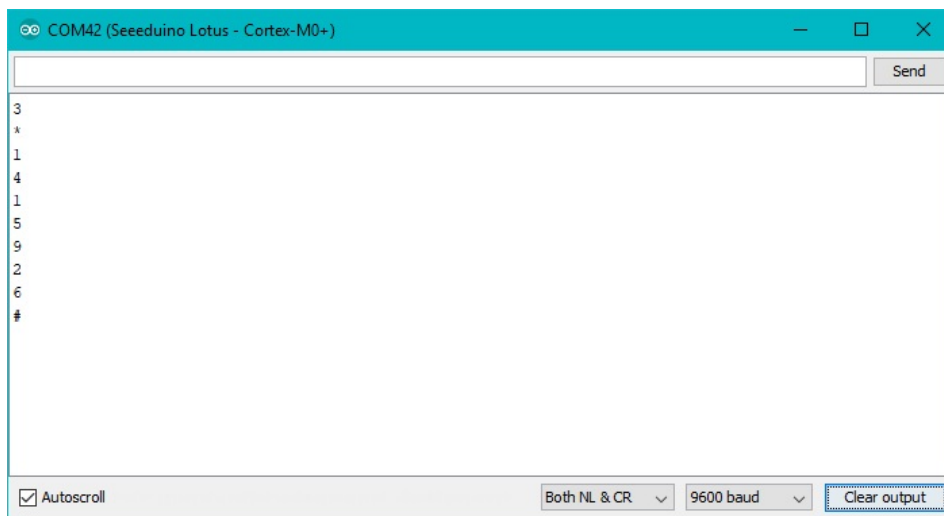
```
41         break;
42     case 0xE3 :
43         SERIAL.println("3");
44         break;
45     case 0xE4 :
46         SERIAL.println("4");
47         break;
48     case 0xE5 :
49         SERIAL.println("5");
50         break;
51     case 0xE6 :
52         SERIAL.println("6");
53         break;
54     case 0xE7 :
55         SERIAL.println("7");
56         break;
57     case 0xE8 :
58         SERIAL.println("8");
59         break;
60     case 0xE9 :
61         SERIAL.println("9");
62         break;
63     case 0xEA :
64         SERIAL.println("*");
65         break;
66     case 0xEB :
67         SERIAL.println("0");
68         break;
69     case 0xEC :
70         SERIAL.println("#");
71         break;
72     default:
73         break;
74     }
75 }
76
77 }
```

- **Step 3.** Upload the demo. If you do not know how to upload the code, please check [How to upload code](https://wiki.seeedstudio.com/Upload_Code/) [https://wiki.seeedstudio.com/Upload\_Code/].
- **Step 4.** Open the **Serial Monitor** of Arduino IDE by click **Tool->Serial Monitor**. Or tap the **Ctrl + Shift + M** key at the same time. Set the baud rate to **9600**.



### Success

Now, touch the keypad, then the monitor will output the corresponding key.



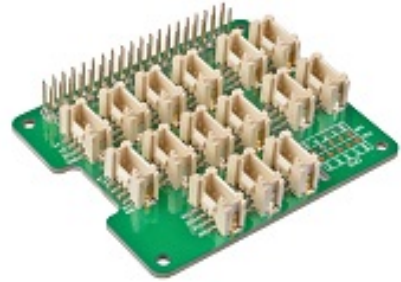
## Play with Raspberry pi

### Materials required

Raspberry pi



Grove Base Hat for RasPi



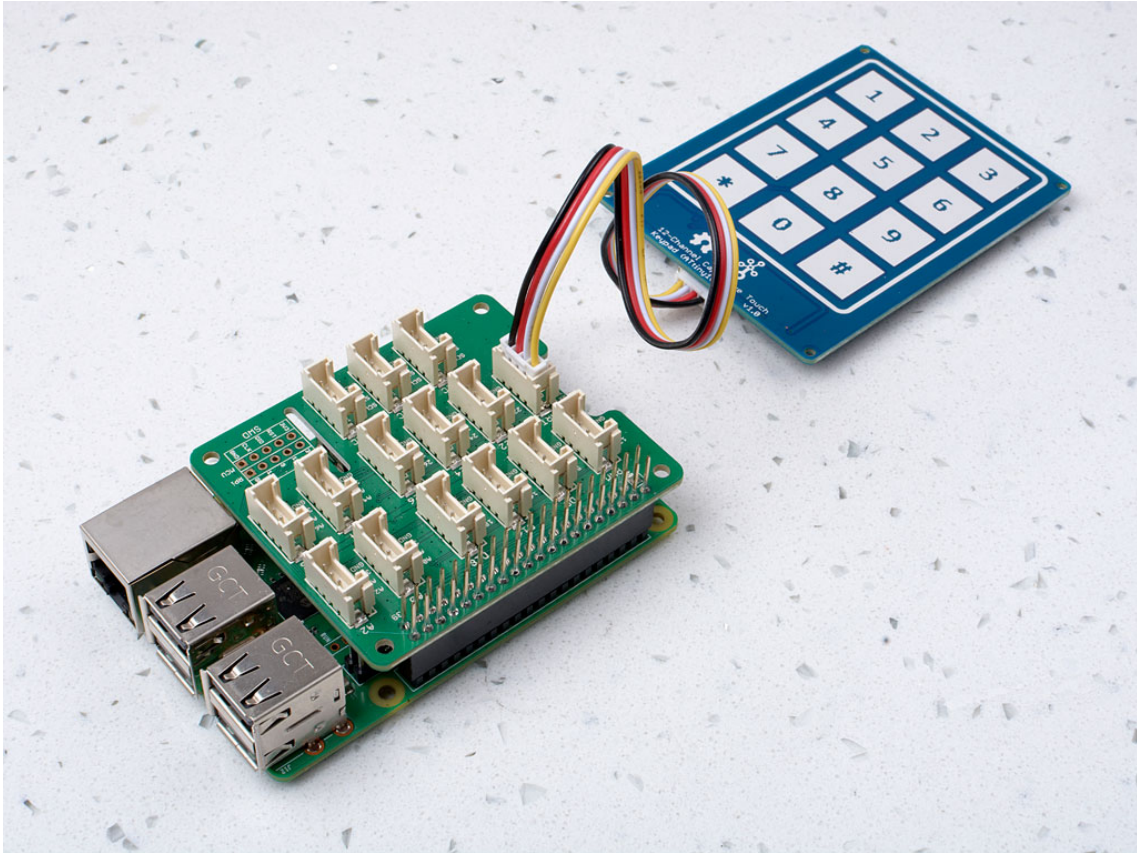
Get ONE Now

[<https://www.seeedstudio.com/Raspberry-Pi-3-Model-B-p-2625.html>]

Get ONE Now

[<https://www.seeedstudio.com/Grove-Base-Hat-for-Raspberry-Pi-p-3186.html>]

- **Step 1.** Plug the Grove Base Hat into Raspberry.
- **Step 2.** Connect the Grove 12 button Capacitive Touch Keypad to **UART** port of the Base Hat.
- **Step 3.** Power on the Raspberry Pi.



## Software



### Attention

If you are using **Raspberry Pi with Raspberrypi OS >= Bullseye**, you have to use this command line **only with Python3**.

## UART SETTING

Before start, we need to configure the Raspberry Pi UART.

- Step 1. Enable Raspberry Pi3 **UART0**.

```
sudo nano /boot/config.txt
```



Then add the content `dtoverlay=pi3-disable-bt` to the end of the **config.txt**

Tap `Ctrl` + `X` to quit nano, and tap `Y` to save the modification.

- Step 2. Disable the system service to use the UART0.

```
sudo systemctl disable hciuart
```

**Note**

Pi3-disable-bt disables the Bluetooth device and restores UART0/ttyAMA0 to GPIOs 14 and 15. It is also necessary to disable the system service that initialises the modem so it doesn't use the UART: `sudo systemctl disable hciuart`.

- Step 3. Delete the `console=serial0,115200` in `cmdline.txt`.

```
sudo nano /boot/cmdline.txt
```

Then delete `console=serial0,115200` in this file.

**Note**

If you can not find `console=serial0,115200` in this txt file, just skip this step.

- Step 4. Reboot the Raspberry Pi

```
sudo reboot
```

For more detail, please check the official [Raspberry Pi UART Config](https://www.raspberrypi.org/documentation/configuration/uart.md) [https://www.raspberrypi.org/documentation/configuration/uart.md]

## RASPBERRY PI DEMO

- **Step 1.** Follow [Setting Software](#)

[[https://wiki.seeedstudio.com/Grove\\_Base\\_Hat\\_for\\_Raspberry\\_Pi/#installation](https://wiki.seeedstudio.com/Grove_Base_Hat_for_Raspberry_Pi/#installation)] to configure the development environment.

After the system environment is successfully configured, you can see a prompt like this:

```

1 Running setup.py install for grove.py ... done
2 Successfully installed grove.py-0.6
3 #####
4 Lastest Grove.py from github install complete !!!!!
5 #####
  
```

Now, tap 'ls', you can find the **grove.py** folder under the root directory.

```

1 pi@raspberrypi:~ $ ls
2 01_HelloRPi Desktop MagPi rpi_apa10
3 01_HelloRPi.cpp Documents Music Templates
4 4mics_hat Downloads ofxGPIO Videos
5 apa102_led.c env Pictures wiringpi_
6 bcm2835-1.50 grove.py Public wiringpi_
7 bcm2835-1.50.tar.gz led python_games
8 bcm2835-1.50.tar.gz.1 led1 respeaker
  
```

- **Step 3.** Excute the following commands to run the demo.

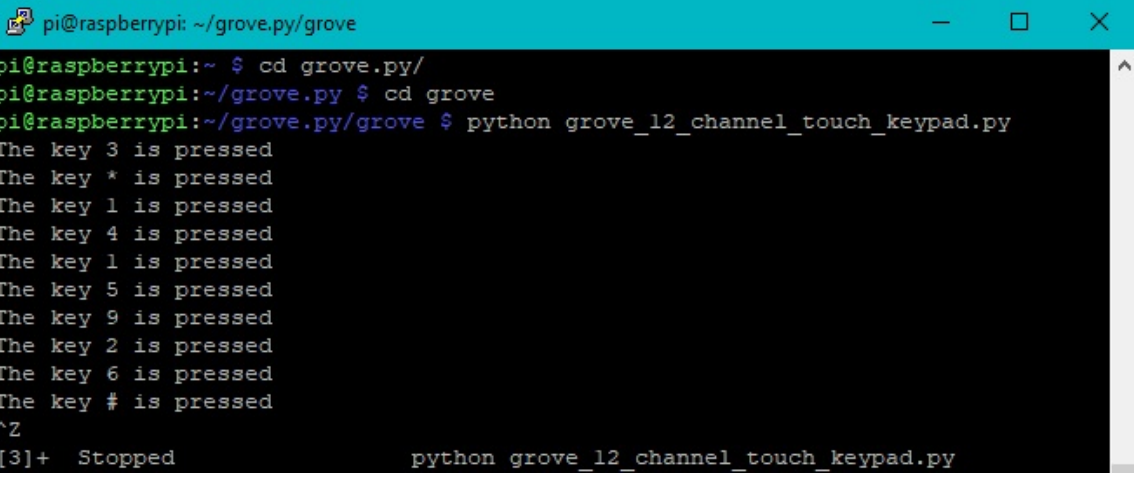
```

1 cd grove.py/grove
2 python grove_12_channel_touch_keypad.py
  
```



**Success**

Then touch the keycap, the terminal will output the corresponding key.



```
pi@raspberrypi: ~/grove.py/grove
pi@raspberrypi:~ $ cd grove.py/
pi@raspberrypi:~/grove.py $ cd grove
pi@raspberrypi:~/grove.py/grove $ python grove_12_channel_touch_keypad.py
The key 3 is pressed
The key * is pressed
The key 1 is pressed
The key 4 is pressed
The key 1 is pressed
The key 5 is pressed
The key 9 is pressed
The key 2 is pressed
The key 6 is pressed
The key # is pressed
^Z
[3]+ Stopped python grove_12_channel_touch_keypad.py
```

## Schematic Online Viewer



## Resources

- **[ZIP]** [Grove 12 Channel Capacitive Touch Keypad \(ATtiny1616\) Schematic file](https://files.seeedstudio.com/wiki/Grove-12-Channel-Capacitive-Touch-Keypad-ATtiny1616/res/Grove%20-%2012-Channel%20Capacitive%20Touch%20Keypad%20(ATtiny1616).zip) [[https://files.seeedstudio.com/wiki/Grove-12-Channel-Capacitive-Touch-Keypad-ATtiny1616/res/Grove%20-%2012-Channel%20Capacitive%20Touch%20Keypad%20\(ATtiny1616\).zip](https://files.seeedstudio.com/wiki/Grove-12-Channel-Capacitive-Touch-Keypad-ATtiny1616/res/Grove%20-%2012-Channel%20Capacitive%20Touch%20Keypad%20(ATtiny1616).zip)]



- **[PDF]** [ATtiny1616 Datasheet](#)

[[https://files.seeedstudio.com/wiki/Grove-12-Channel-Capacitive-Touch-Keypad-ATtiny1616/res/ATtiny1616-1617\\_Datasheet.pdf](https://files.seeedstudio.com/wiki/Grove-12-Channel-Capacitive-Touch-Keypad-ATtiny1616/res/ATtiny1616-1617_Datasheet.pdf)]

## Tech Support

Please submit any technical issue into our [forum](#)

[<https://forum.seeedstudio.com/>]



[[https://www.seeedstudio.com/act-4.html?](https://www.seeedstudio.com/act-4.html?utm_source=wiki&utm_medium=wikibanner&utm_campaign=newproducts)

[utm\\_source=wiki&utm\\_medium=wikibanner&utm\\_campaign=newproducts](https://www.seeedstudio.com/act-4.html?utm_source=wiki&utm_medium=wikibanner&utm_campaign=newproducts)]  
oducts]