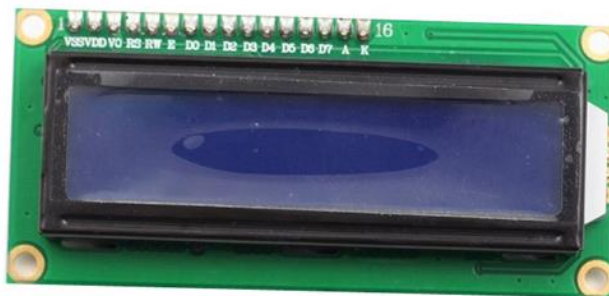


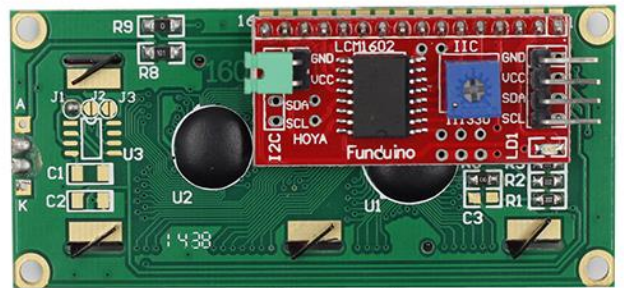
# I2C LCD1602

## Introduction

LCD1602 is a character type liquid crystal display, which can display 32 (16\*2) characters at the same time. It has 16 pins, of which at least 7 would be used each time. You can use a PCF8574 I2C chip to expand I/O ports so only two GPIO ports would be occupied.



Front



Back

## Components

- 1 \* Raspberry Pi
- 1 \* Breadboard
- 4 \* Jumper wires (Male to Male, 2 red and 2 black)
- 1 \* Network cable (or USB wireless network adapter)
- 1 \* I2C LCD1602
- Several jumper wires (Male to Female)

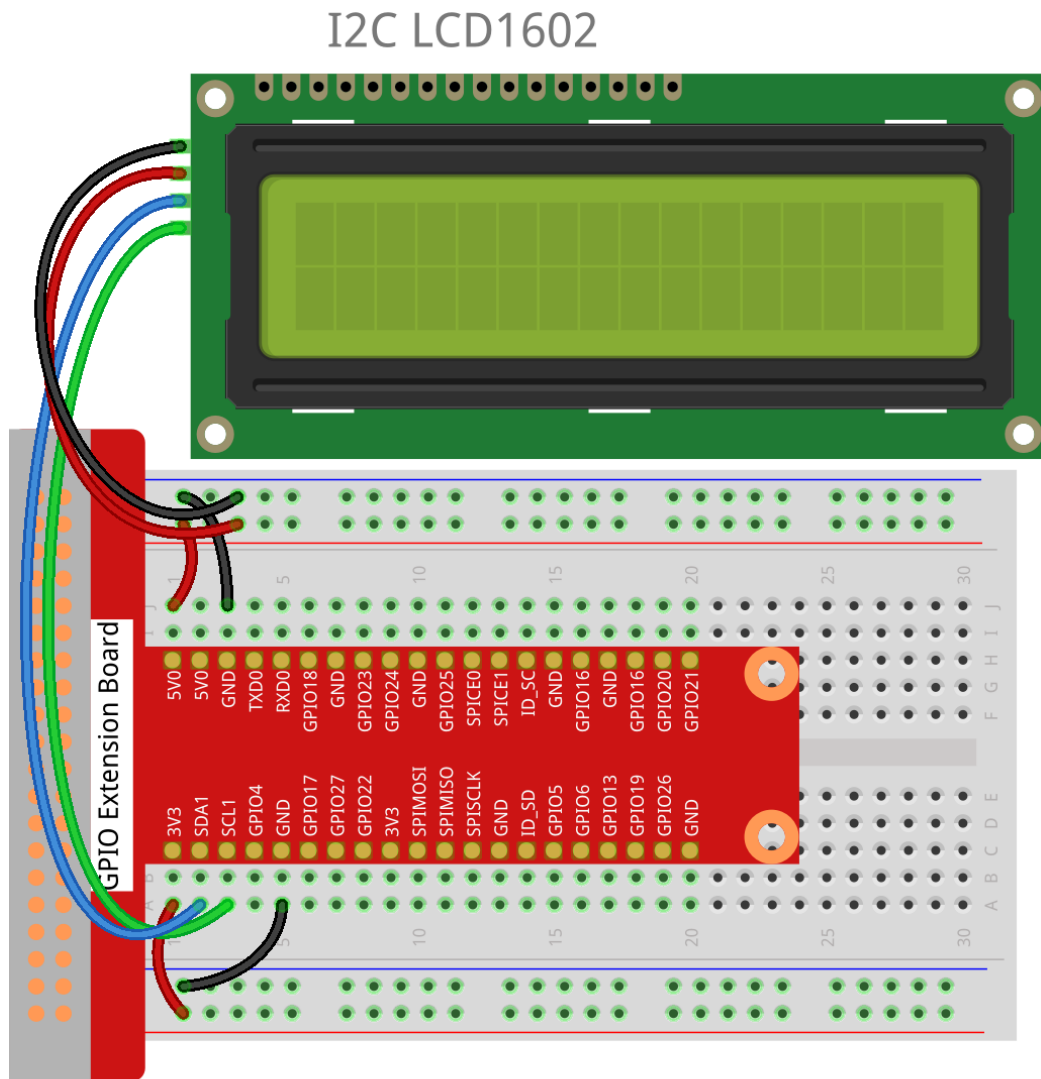
## Experimental Principle

In this experiment, I2C is used to configure LCD so that you can control the LCD1602 to display characters. The I2C slave address of I2C LCD1602 here is 0x27.

## Experimental Procedures

**Step 1:** Connect the circuit

Raspberry Pi	T-Cobbler	I2C LCD1602 Module
SCL	SCL1	SCL
SDA	SDA1	SDA
5V	5V0	VCC
GND	GND	GND



fritzing

**Step2:** Setup I2C (see Appendix 1. If you have set I2C, skip this step.)

**Step 3:** Download the I2C LCD1602 Test Experiment for Raspberry Pi.zip file , then extract it and copy the **Code** folder `/home/pi/` on your Raspberry Pi.

**For C language users:**

**Step 4:** Change directory

```
cd /home/pi/Code/C/i2c_lcd1602/
```

**Step 5:** Compile

```
gcc i2c_lcd1602.c -lwiringPi
```

**Step 6:** Run

```
sudo ./a.out
```

**For Python users:**

**Step 4:** Create a file LCD1602.py

```
Nano LCD1602.py
```

Copy the code in the LCD1602.py file to this blank file and then save and exit.

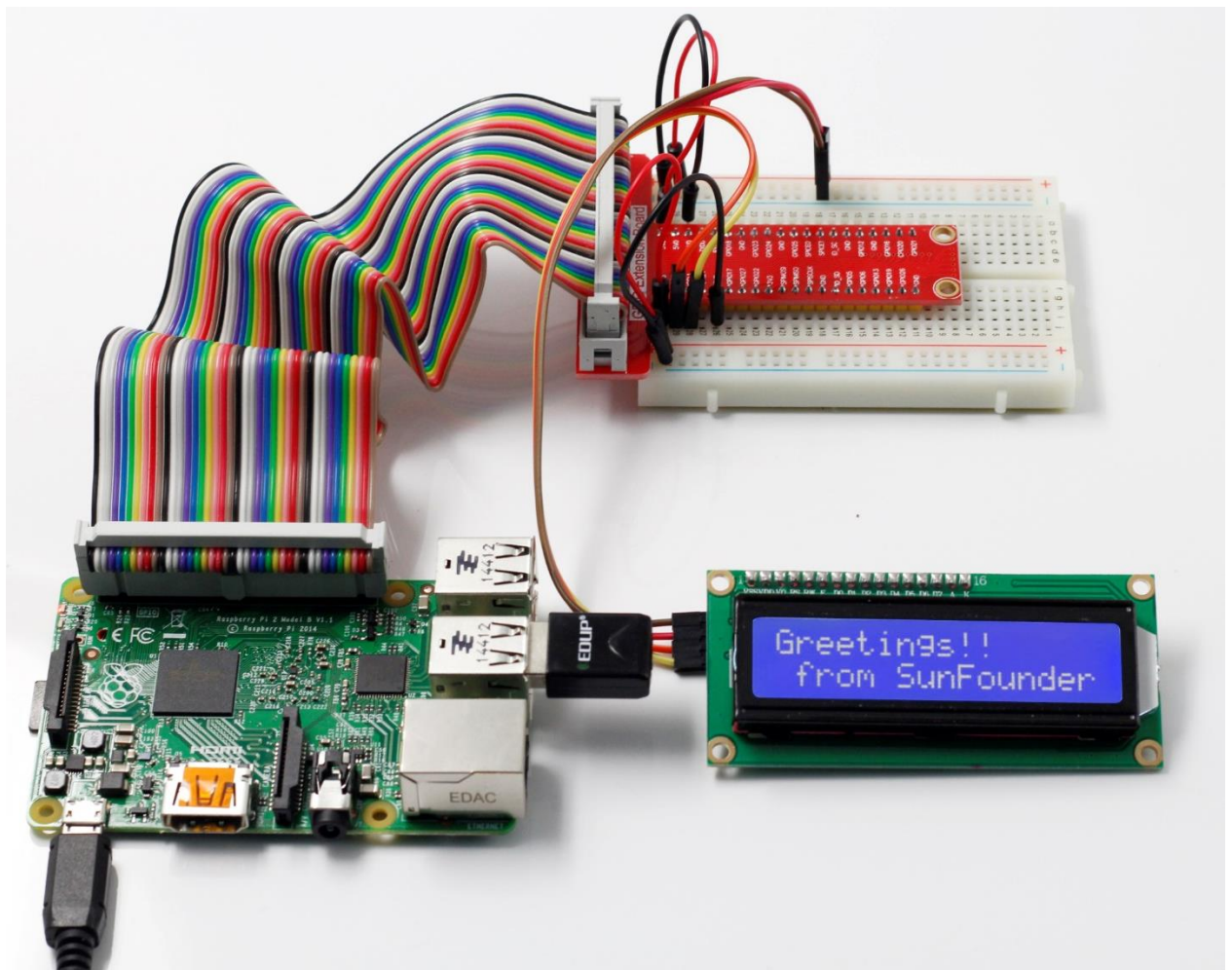
**Step 5:** Change directory

```
cd /home/pi/Code/Python/
```

**Step 6:** Run

```
sudo python i2c_lcd1602.py
```

Now you can see "Greetings! From SunFounder" displayed on the LCD.

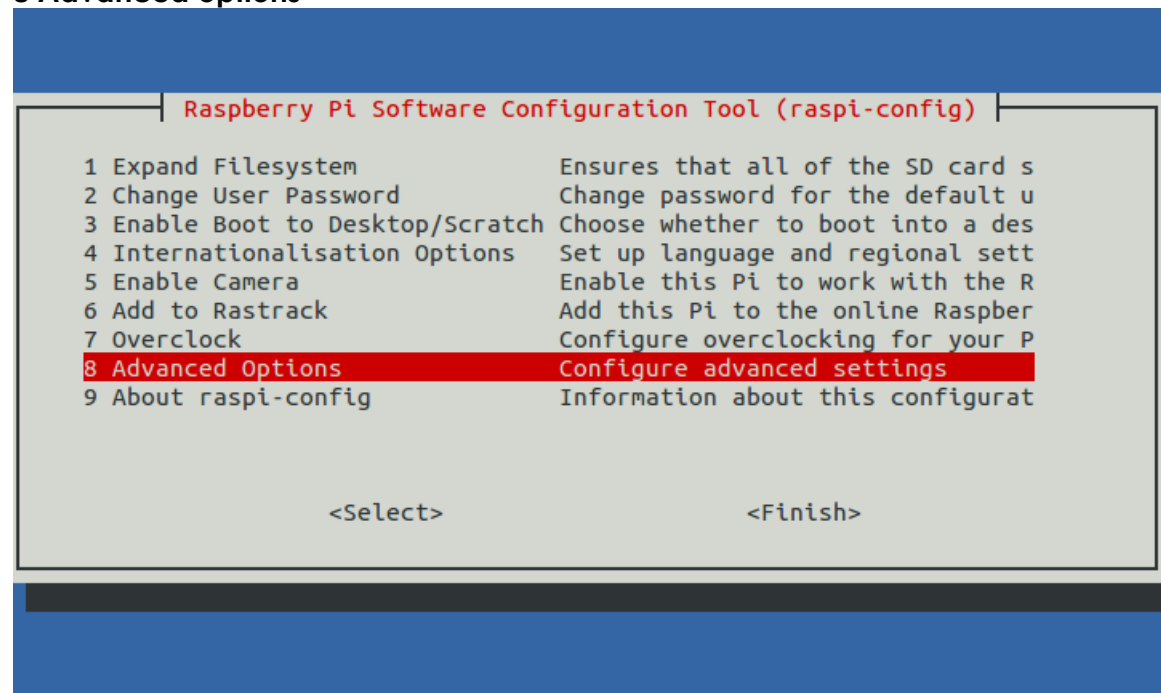


## Appendix 1: I2C Configuration

**Step 1:** enable the I2C port of your Raspberry Pi (If you have enabled it, skip it. If you do not know whether you have enabled it or not, please continue):

```
sudo raspi-config
```

### 8 Advanced options



A7 I2C

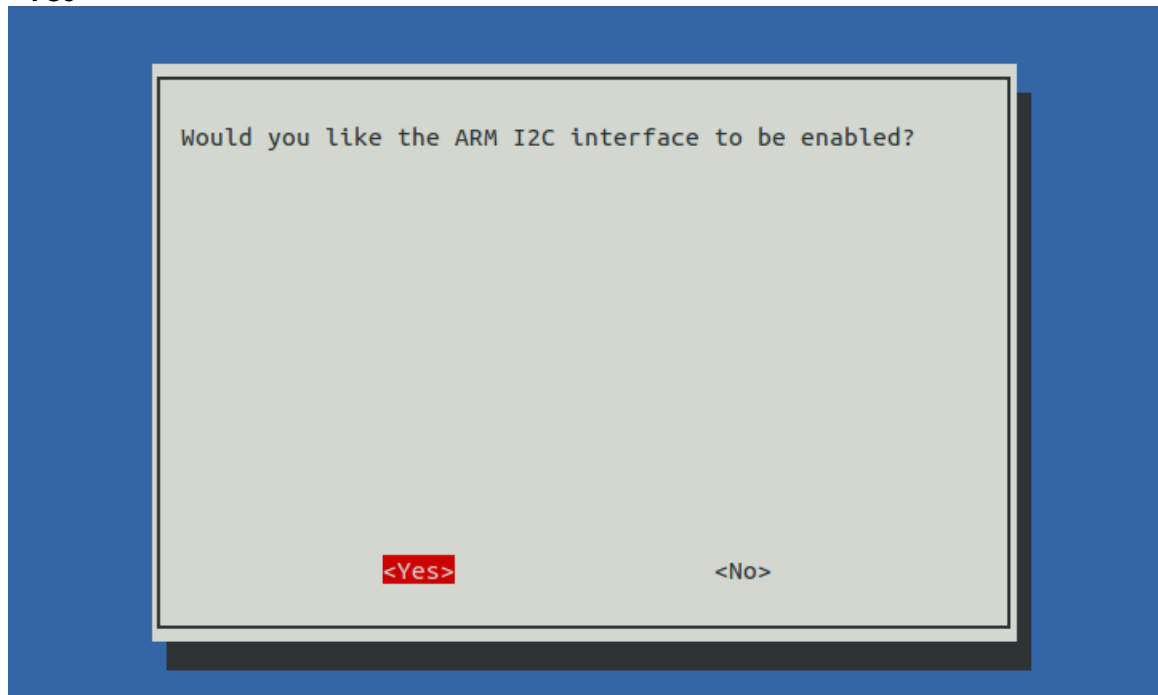
## Raspberry Pi Software Configuration Tool (raspi-config)

A1 Overscan	You may need to configure oversca
A2 Hostname	Set the visible name for this Pi
A3 Memory Split	Change the amount of memory made
A4 SSH	Enable/Disable remote command lin
A5 Device Tree	Enable/Disable the use of Device
A6 SPI	Enable/Disable automatic loading
<b>A7 I2C</b>	<b>Enable/Disable automatic loading</b>
A8 Serial	Enable/Disable shell and kernel m
A9 Audio	Force audio out through HDMI or 3
A0 Update	Update this tool to the latest ve

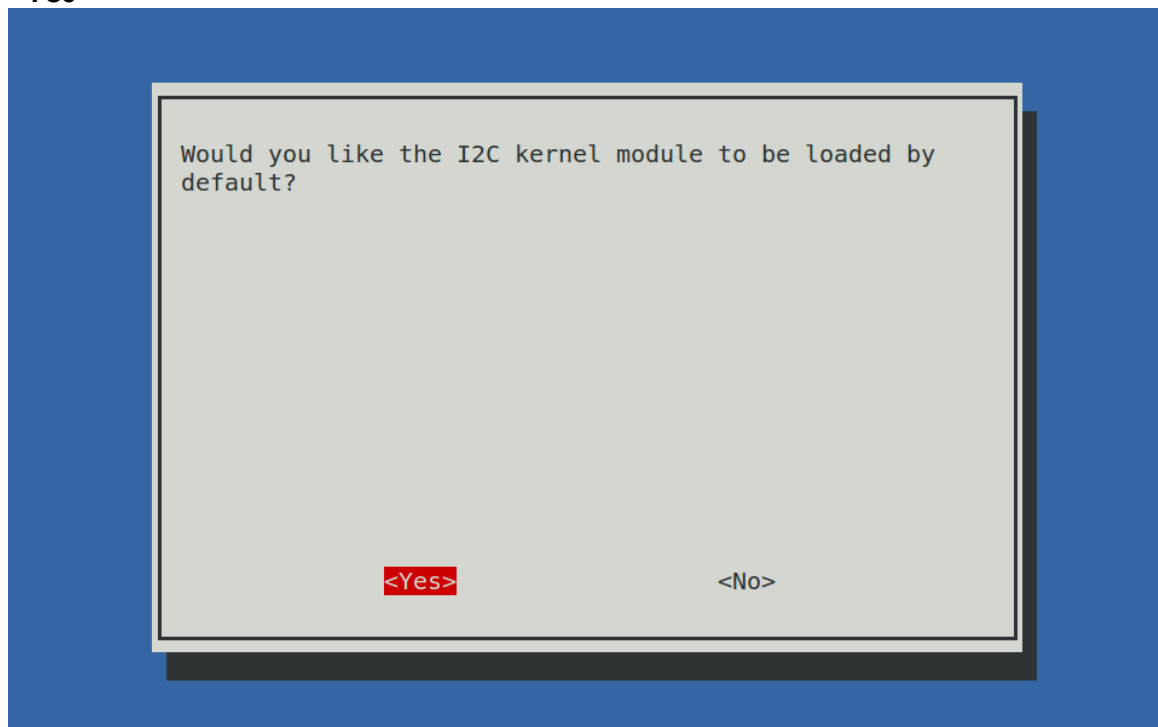
<Select>

<Back>

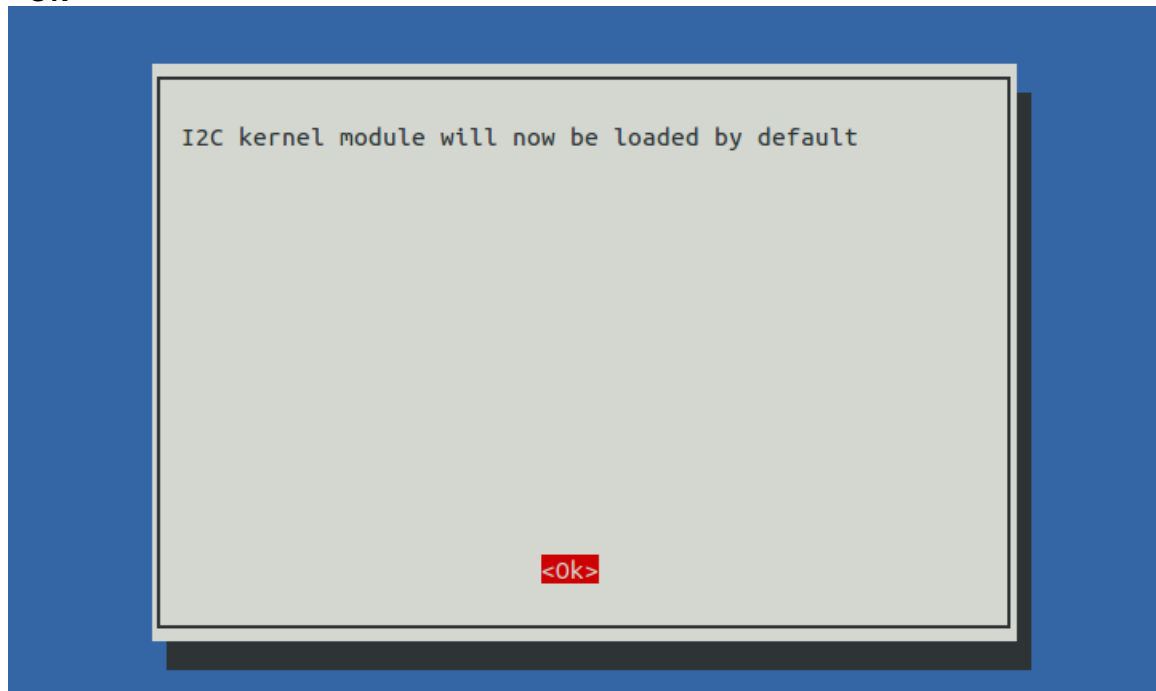
<Yes>



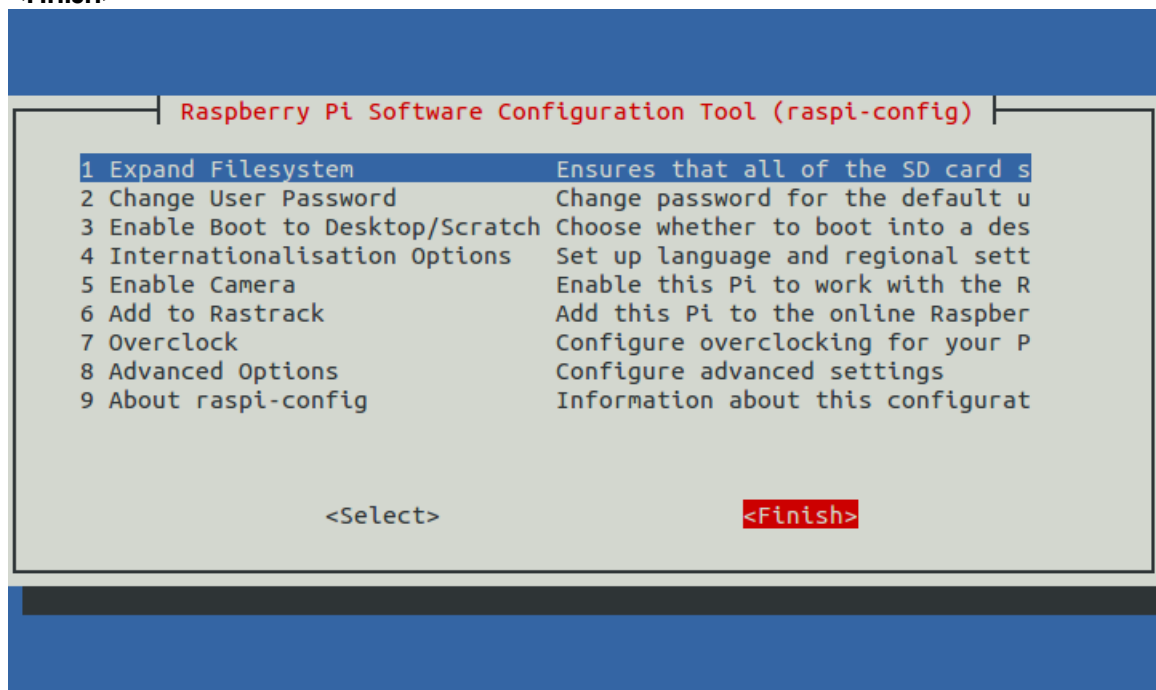
<Yes>



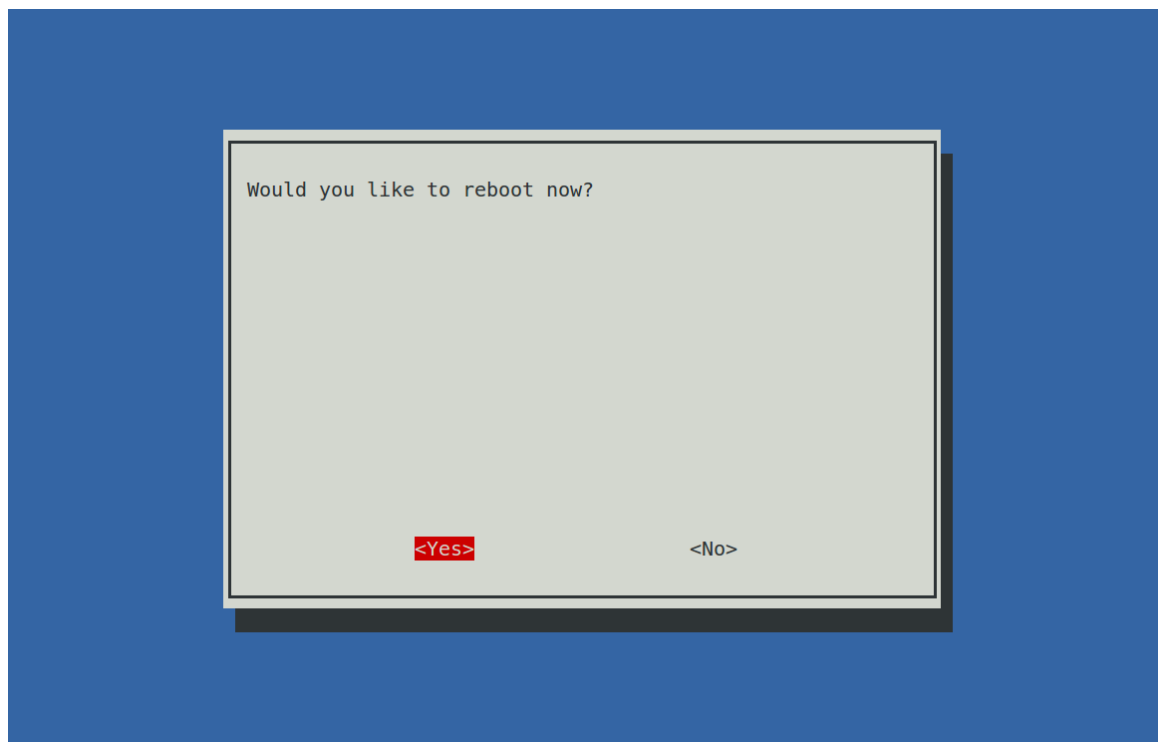
<Ok>



<Finish>



**<Yes>** (If you do not see this page, continue to the next step)



**Step2:** Comment out the i2c-bcm2708 line from the raspi-blacklist.conf file:

Open `/etc/modprobe.d/raspi-blacklist.conf` with *nano*

```
sudo nano /etc/modprobe.d/raspi-blacklist.conf
```

Add the code below.

```
#blacklist i2c-bcm2708
```

**Step3:** From the prompt, run:

```
sudo modprobe i2c_bcm2708
sudo modprobe i2c-dev
```

**Step4:** Check that the i2c modules are loaded and active:

```
lsmod | grep i2c
```

Then the following code will appear (the number may be different)

```
i2c_dev          6276  0
i2c_bcm2708      4121  0
```

**Step5:**

**For C language users:** Install libi2c-dev

```
sudo apt-get install libi2c-dev
```

**For Python users:** Install smbus for I2C

```
sudo apt-get install python-smbus i2c-tools
```



