LAB MANUAL 4

Getting Started with GrovePI & Raspberry PI

What is GrovePi+

GrovePi+ is add-on board with 15 Grove 4-pin interfaces that brings Grove sensors to the Raspberry Pi. It is the newest version compatible with Raspberry Pi model B/B+/A+/2/3/4 perfectly.

GrovePi+ is an easy-to-use and modular system for hardware hacking with the Raspberry Pi, no need for soldering or breadboards: plug in your Grove sensors and start programming directly. Grove is an easy to use collection of more than 100 inexpensive plug-and-play modules that sense and control the physical world. By connecting Grove Sensors to Raspberry Pi, it empowers your Pi in the physical world. With hundreds of sensors to choose from Grove families, the possibilities for interaction are endless.

1. Compatible with Raspberry Pi model B/B+/A+/2/3/4
2. Faster SPI and higher reliability UART connections
3. Easier to assemble camera cables and LCD cables
4. Simplified procedures of firmware update

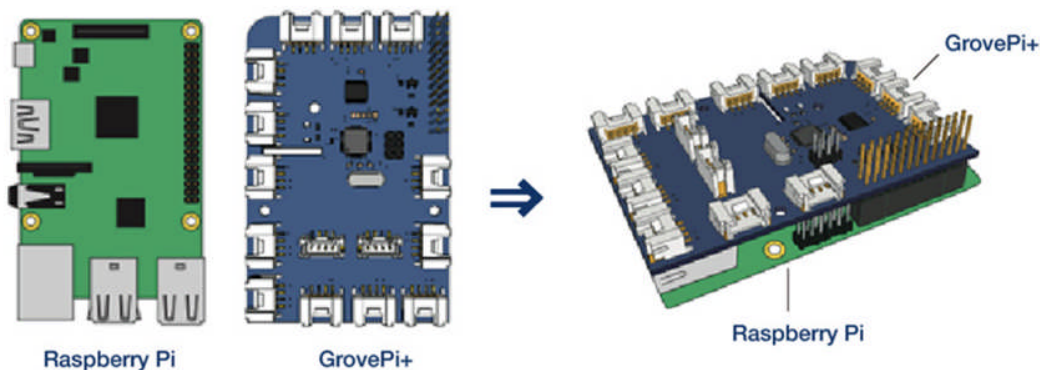Reference: GrovePi+ add-on for Raspberry Pi – Seeed Studio

Set-up in 4 simple steps
Step 1: Slip the GrovePi+ board over your Raspberry Pi
Step 2: Connect the Grove modules to the GrovePi+ board
Step 3: Upload your program to Raspberry Pi
Step 4: Begin taking in the world data

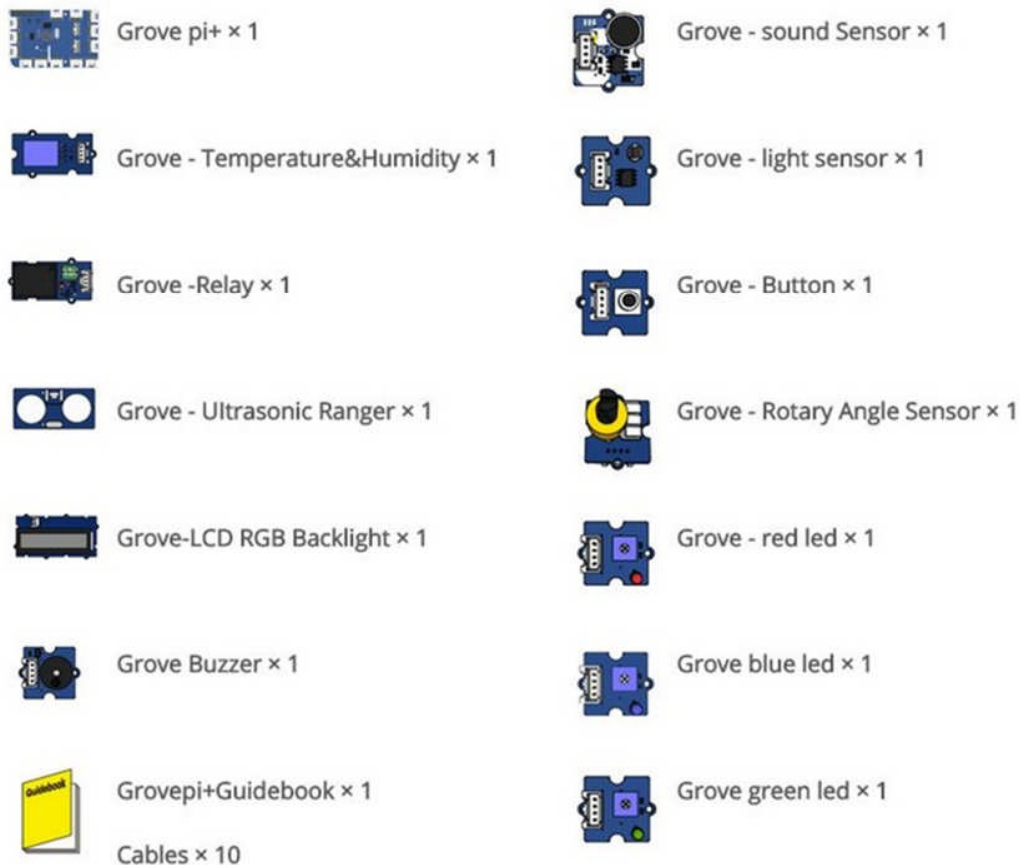Reference: GrovePi+ add-on for Raspberry Pi – Seeed Studio

Features:
- 7 digital Ports
- 3 analoge Ports
- 3 I2C ports
- 1 Serial port connect to GrovePi
- 1 Serial port connect to Raspberry Pi
- Grove header Vcc output Voltage: 5Vdc

The GrovePi Starter Kit gets you up and running with the GrovePi quickly. The starter kit bundles the most popular sensors for education and hobbyists, and lets you start playing and prototyping hardware with Raspberry Pi. No soldering required!

The GrovePi Starter Kit package includes:
1. GrovePi+ Board
2. 12 different Grove sensors and modules
3. Grove cables for connecting the sensors to the GrovePi board.

Grove pi+ × 1		Grove - sound Sensor × 1	
Grove - Temperature&Humidity × 1		Grove - light sensor × 1	
Grove -Relay × 1		Grove - Button × 1	
Grove - Ultrasonic Ranger × 1		Grove - Rotary Angle Sensor × 1	
Grove-LCD RGB Backlight × 1		Grove - red led × 1	
Grove Buzzer × 1		Grove blue led × 1	
Grovepi+Guidebook × 1 Cables × 10		Grove green led × 1	

Reference: starter-Kit-content.jpg (758×647) (dexterindustries.com)

Hardware connection for GrovePi+ to raspberry Pi
First, mount your GrovePi on the Raspberry Pi. The GrovePi slides over top of the Raspberry Pi as shown in the picture below.
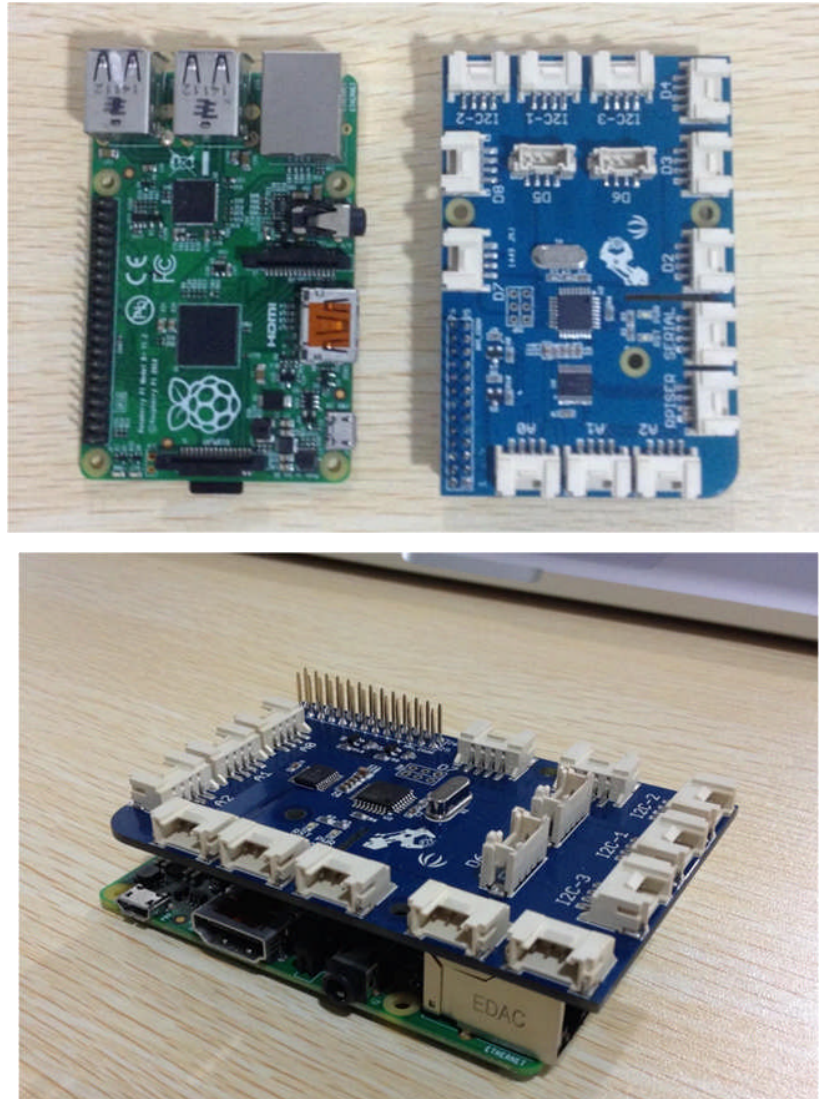
Ensure that the pins are properly aligned when stacking the GrovePi.

Powering up the Raspberry Pi

To power the GrovePi+ and the Raspberry Pi, you can use the micro-USB power port on the Raspberry Pi. Remember to use a good power adapter capable of supplying 1A at 5V and you should be fine with the power. If you want to run the GrovePi+ in a standalone configuration, then you should use a USB power bank

Setup the Software on the Raspberry Pi

Next we will install the software on the Raspberry Pi.

Clone and install GrovePI repository

- curl -kL dexterindustries.com/update_grovepi | bash

- sudo pip install cffi && sudo pip3 install cffi
- curl -kL dexterindustries.com/update_grovepi | bash

NOTE : Do the next command only after connecting GrovePi Shield on Raspberry Pi

Update Firmware
- cd ~/Dexter/GrovePi/Firmware
- bash firmware_update.sh

Running Tests (Optional)
- Check version
- python
- >>import grovepi
- >>grovepi.version()

Python code for Check all the digital port working fine-

```python
import time
from grovepi import *
digital_ports = [2, 3, 4, 5, 6, 7, 8]
def check_digital_ports():
    for port in digital_ports:
        try:
            pinMode(port, "OUTPUT")
            digitalWrite(port, 1)
            time.sleep(1)
            state = digitalRead(port)
            print(f"Digital Port {port} - HIGH state: {state}")
            digitalWrite(port, 0)
            time.sleep(1)
            state = digitalRead(port)
            print(f"Digital Port {port} - LOW state: {state}")
        except IOError:
            print(f"Error reading/writing Digital Port {port}")
```