

---

# SunFounder RasPad 3

[www.sunfounder.com](http://www.sunfounder.com)

Dec 27, 2023



# Contents

<b>1</b>	<b>SAFETY NOTICES</b>	<b>3</b>
1.1	Precautions about the use of the product . . . . .	3
1.2	Precautions about the maintenance of lithium batteries . . . . .	3
<b>2</b>	<b>COMPONENTS LIST</b>	<b>5</b>
<b>3</b>	<b>ASSEMBLY INSTRUCTIONS</b>	<b>7</b>
<b>4</b>	<b>QUICK USER GUIDE</b>	<b>17</b>
4.1	Install the Raspberry Pi OS . . . . .	17
4.2	Install Virtual Keyboard on RasPad . . . . .	22
4.3	Right Click on RasPad . . . . .	28
4.4	Install Rotating Screen Service . . . . .	29
<b>5</b>	<b>RASPAD LAUNCHER</b>	<b>31</b>
5.1	Introduction . . . . .	31
5.2	Installation Guide . . . . .	32
<b>6</b>	<b>INTERFACES INTRODOCTION</b>	<b>33</b>
6.1	Charge the RasPad . . . . .	34
6.2	Volume and Brightness Settings . . . . .	37
6.3	Connect Headphones or Other Screens . . . . .	40
<b>7</b>	<b>RASPAD PROJECTS AND ACTIVITIES</b>	<b>45</b>
7.1	Multimedia Options . . . . .	45
7.1.1	Watch Video . . . . .	45
7.1.2	Read E-book . . . . .	46
7.1.3	Play Game . . . . .	47
7.2	Retro Games Console . . . . .	48
7.2.1	Description . . . . .	48
7.2.2	Required Components . . . . .	49
7.2.3	Game System Installation . . . . .	51
7.2.4	RetroPie Configuration . . . . .	54
7.3	3D Printer Monitor . . . . .	55
7.3.1	Touch UI . . . . .	56
7.3.2	Make a 3D Model . . . . .	56
7.3.3	Print 3D Model . . . . .	62
7.3.4	Video . . . . .	65
7.4	Home Assistant . . . . .	65
7.4.1	Install Home Assistant . . . . .	66
7.4.2	Install and configure ESPHome . . . . .	67
7.4.3	Log In . . . . .	75
7.4.4	Add Smart Devices . . . . .	80

7.4.5	Add DIY Device with ESP32 . . . . .	85
7.4.6	Start the Home Assistant Service at Boot . . . . .	90
7.4.7	Enter the Home Assistant Page at Boot . . . . .	90
7.4.8	Home Assistant File Structure . . . . .	92
7.5	IoT . . . . .	93
7.5.1	Get Start with Blynk . . . . .	93
7.5.2	Smart Light . . . . .	107
7.5.3	Door Window Sensor . . . . .	112
7.5.4	Smart Fan . . . . .	116
7.5.5	Temperature Recorder . . . . .	120
7.6	GPIO Programming . . . . .	124
7.6.1	Electronic Kit for Raspberry Pi and Arduino . . . . .	125
7.6.2	Da Vinci Kit for Raspberry Pi . . . . .	125
7.7	Scratch Projects . . . . .	125
7.7.1	Animate A Name . . . . .	126
7.7.2	Create a Story . . . . .	135
7.7.3	Make Music . . . . .	145
7.7.4	Clicker Game . . . . .	150
7.7.5	Shooting Game . . . . .	155
7.7.6	Hit the Magic Fish . . . . .	164
7.7.7	Binary Game . . . . .	172
7.7.8	Flappy Parrot . . . . .	180
7.7.9	Paint Box . . . . .	191
7.7.10	Eating Banana Game . . . . .	206
7.7.11	Create Musical Instrument . . . . .	210
7.7.12	DIY RasPad Monitor Device . . . . .	216
7.8	Install Other Systems on RasPad . . . . .	222
<b>8</b>	<b>APPENDIX</b>	<b>223</b>
8.1	Assemble the Camera Module . . . . .	223
8.2	GPIO Extension Board . . . . .	227
8.3	Obtain the IP Address of the Raspberry Pi . . . . .	229
8.3.1	With a Screen . . . . .	229
8.3.2	Without a Screen . . . . .	230
8.4	Use the SSH Remote Control . . . . .	231
<b>9</b>	<b>FAQ</b>	<b>235</b>
9.1	Fix Raspberry Pi boot problems . . . . .	235
9.2	Can't detect Micro SD card? . . . . .	235
9.3	Screen can not be rotated? . . . . .	235
9.4	About sleep problem . . . . .	236
9.5	About the shutdown problem . . . . .	236
9.6	Desktop is not full screen . . . . .	236
9.7	About the touch problem . . . . .	237
9.8	About the Audio problem . . . . .	238
9.9	Fan is too noisy? . . . . .	238
9.10	Power consumption is too fast? . . . . .	238
9.11	Touch function sometimes fails? . . . . .	238
9.12	The WiFi connection is unstable? . . . . .	238
<b>10</b>	<b>Copyright Notice</b>	<b>239</b>

An ideal programming Raspberry Pi tablet perfectly compatible with Raspberry Pi 4B. RasPad 3 with internal 10.1 inch IPS Pi touchscreen LCD monitor, high resolution of 1280×800 pixels, 10-finger touch, 5 hours long battery life. Easily assembled with 2 steps to get started anytime anywhere.

By leveraging the HDMI, you can easily extend RasPad with other monitors for better multitasking or mirror your display. In addition, due to triangular structure design and six-axis acceleration sensor, it can be placed flat or upright on the table, or rotate it 90°/180°/270° to give amazing presentations!

With all the Raspberry Pi ports, including Ethernet, HDMI, Audio, USB, and Power. All the RPi I/O interface, a customized OS, stereo speakers and many more features for developers to have a convenient hardware-hacking experience, specially designed GPIO Extension and camera holes making users build projects or install cameras for RasPad. You can start programming from anywhere and fun daily work.

Compatible with Arduino, Raspbian, Ubuntu Desktop, Chromium OS, Android and more... Realize A Tablet to Program & A Remote Controller & A Spare Display. Ideal for applications like retro gaming, industrial control, as a secondary monitor, in home automation, for 3D printing control, and as a .

### **About the display language**

In addition to English, we are working on other languages for this course. Please contact [service@sunfounder.com](mailto:service@sunfounder.com) if you are interested in helping, and we will give you a free product in return. In the meantime, we recommend using Google Translate to convert English to the language you want to see.

The steps are as follows.

- In this course page, right-click and select **Translate to xx**. If the current language is not what you want, you can change it later.

- There will be a language popup in the upper right corner. Click on the menu button to **choose another language**.
- Select the language from the inverted triangle box, and then click **Done**.

# Chapter 1

## SAFETY NOTICES

### 1.1 Precautions about the use of the product

- Be sure to use the dedicated power adapter for this product for charging and discharging, and never use an overpowered power adapter;
- Do not use the product in a high temperature environment, and the battery power should not be too low. Generally, it is best to start charging at about 30%;
- Avoid plugging in the power adapter for a long time, and stop charging after charging. When the product is not in use, it is recommended to disconnect the power adapter to avoid battery damage caused by long-term charging.

### 1.2 Precautions about the maintenance of lithium batteries

- Lithium batteries can be stored in a clean, dry, and ventilated room with an ambient temperature of -5°C—35°C and a relative humidity of not more than 75%. Avoid contact with corrosive substances and keep away from fire and heat sources.
- Do not store the lithium battery with full charge for a long time. After long-term storage, it will easily cause swelling, which will affect the discharge performance. The best storage voltage is about 3.8V for a single chip. Fully charge before use and then use it, which can effectively avoid battery swelling.







## **Chapter 2**

# **COMPONENTS LIST**

## Chapter 3

# ASSEMBLY INSTRUCTIONS

1. Before inserting the Micro SD card, connect the button-board to the Raspberry Pi using the included FFC cable.
2. Connect all the cables, and then attach the Raspberry Pi with four of the M2.5x4 screws.
3. Orient the fan so the logo is facing the exterior vent, then attach the fan with the four PWA1.7X9 screws.
4. Finally, attach the back cover with five M2.5x4 screws.

You can also follow the video to assemble RasPad 3 step by step.

### **Detailed Assembly Steps**

1. Take out the RasPad and turn it over, unscrew the screws and remove the back cover.

Below is an image of the internal structure of the RasPad.

---

**Note:** Avoid pressing on the two speakers during assembly.

---

2. Insert the Micro SD card board into the Raspberry Pi's Micro SD card slot.

3. Insert the other end of the FFC cable into the RasPad button board.

---

**Note:** This step can be skipped if the Micro SD card board has already been connected to the button board in the RasPad that you received.

---

- Use two fingers to gently pull up the tab on the top of the CSI connector.
- Insert the FFC cable, being sure that it is fully inserted and not crooked.
- Use your fingers to press it firmly from both sides.
- Pull up the tab gently again if it is not attached properly and can't pull out the FFC forcefully.

4. Attach the Raspberry Pi with 4 M2.5x4 screws.

5. Connect the Ethernet, USB to USB, 2 Mrico HDMI and Type-C cables in order.

---

**Note:** It doesn't matter if you find the direction of two Mrico HDMI cables slightly off, you can just rotate them manually.

---

6. Attach the three heat-sinks to the Raspberry Pi, and insert the Accel SHIM module for the RasPad's auto-rotate function.

---

**Note:** Accel SHIM module has no soldered pins so you need to observe whether the 6 holes are in contact with the Raspberry Pi pins. If not, gently break the Raspberry Pi pins.

---



7. To use the GPIO pins of the Raspberry Pi, connect a 40-pin GPIO ribbon cable to the Raspberry Pi, and route the ribbon cable out the top of the case.

---

**Note:** If you don't need to build the circuit with RasPad, you can skip this step.

---

8. Attach the fan onto the back cover with four PWA1.7x9 screws.

---

**Note:** The logo faces out towards the exterior vent.

---

9. Connect the fan to the Main Board.

10. Attach the back cover with five M2.5x4 screws.

---

**Note:** Do not force the back cover in place. If the enclosure is hard to connect, or the screw-holes do not line up correctly, please check that there are no internal components interfering with the fit before trying to reattach the back cover.

---





## Chapter 4

# QUICK USER GUIDE

This chapter includes flashing the Raspberry Pi image, installing the virtual keyboard, setting the right-click function and calibrating the rotating screen function.

### 4.1 Install the Raspberry Pi OS

An operating system needs to be installed onto the Micro SD card before inserting it into the Raspberry Pi.

1. Browse to the Raspberry Pi Imager site and download the version that matches the computer or laptop's operating system ([www.raspberrypi.org/software](http://www.raspberrypi.org/software)). When the download is complete, launch the installer program and follow the prompts to install the Raspberry Pi Imager.
2. If the Windows operating system prevents the application from launching, click on **More info** and then click on **Run anyway** to install the Raspberry Pi Imager.

3. Insert the Micro SD card into the computer or laptop's SD card slot. If the computer or laptop only has a full-size port for SD cards, use a Micro SD card adapter.
4. In the Raspberry Pi Imager, click **CHOOSE OS** -> **Raspberry Pi OS(Legacy)**.

**Warning:**

- Please do not install the **Bookworm** version to avoid the **double-click** function and the **virtual keyboard** not working.
- You need to install the **Raspberry Pi OS (Legacy)** version - **Debian Bullseye**.
- Additionally, for a perfect touch experience, it is recommended to install the **Ubuntu** system.

5. Select the SD card you are using.

6. To open the advanced options page, click the **setting** button (appears after selecting operating system) or press **Ctrl+Shift+X**. Now, set hostname, enable ssh and set the username and password.

**Warning:** Make sure to note down the hostname, username, and password; they're crucial for later remote access to the Raspberry Pi.

7. Then scroll down to complete the wifi configuration and click **SAVE**.

---

**Note:** **wifi country** should be set the two-letter ISO/IEC alpha2 code for the country in which you are using your Raspberry Pi, please refer to the following link: [https://en.wikipedia.org/wiki/ISO\\_](https://en.wikipedia.org/wiki/ISO_)

3166-1\_alpha-2#Officially\_assigned\_code\_elements

---

8. Click the **WRITE** button.

9. The Imager will overwrite any files already on the SD card. If the SD card has any files, back up these files before clicking Yes.



10. After the Imager completes installing the Operating System on the SD card, click Continue to complete the installation.

11. Now insert the Micro SD card with the system installed into the Raspberry Pi. The power button is on the right side. Press and hold for over three seconds to power the RasPad on or off. A short press will turn the screen on or off.

---

**Note:** Before use, it is recommended to fully *Charge the RasPad*.

---

**Warning:** The SD card slot is a snap-in style attached to the back cover. Before removing or replacing the back cover, remove the Micro SD card to avoid damaging the Micro SD card and the RasPad's internal button board.

## 4.2 Install Virtual Keyboard on RasPad

When you use a touch panel such as RasPad, you can connect an external keyboard to help you do a text input operation, but it is best to install a virtual keyboard.

Install the required software with the following command.

```
sudo apt install onboard -y  
sudo apt install at-spi2-core
```

In order to make the virtual keyboard have a better effect, you need to do further settings.

Click the Raspberry Pi icon in the upper left corner and select **Preferences -> Onboard Settings**.

In the **General** option, check the following 2 items. When you check **Automatically display when editing text**, you will be prompted to reboot, you can reboot after all settings are complete.

In the **Window** option, check **Dock to screen edge**.

And in **Auto-show** option, check **Auto-show when editing text** again.

The next 2 items are optional, the ones checked in the picture are recommended by us, you can also check other ones.  
In **Layout** option, **Small** is recommended.

In **Theme** option, **DarkRoom** is recommended.

After the setting is complete, restart RasPad. Every time you restart RasPad, you need to click **General Access -> Onboard** to enable the virtual keyboard.

Now you can use this keyboard to edit your files or codes.

## 4.3 Right Click on RasPad

The touchscreen makes it easy to perform simple navigation tasks with your finger or stylus, but you may want to be able to use the context menu (right-click menu).

Then you will need to download a Touchégg app. Enter the following command to install it.

**For 32-bit OS:**

```
wget https://github.com/JoseExposito/toucheegg/releases/download/2.0.14/toucheegg_2.0.14_
↳armhf.deb
sudo apt install ./toucheegg_2.0.14_armhf.deb
```

**For 64-bit OS:**

```
wget https://github.com/JoseExposito/toucheegg/releases/download/2.0.14/toucheegg_2.0.14_
↳arm64.deb
sudo apt install ./toucheegg_2.0.14_arm64.deb
```

After restarting, you can double-tap the screen to bring up the context menu.

```
sudo reboot
```



### Available gestures

#### Swipe

Swipe gestures are executed when three or more fingers are moved synchronously in the same direction. Note that three is the minimum number of fingers that Touchégg allows for swipe gestures on touchpads and two on touchscreens.

#### Pinch

Pinch gestures are executed when two or more fingers are located on the touchpad and are either changing the relative distance to each other (pinching) or are changing the relative angle (rotate).

#### Tap

Tap gestures are executed when two or more fingers “click” on the touchscreen.

For more details please check Touchégg.

## 4.4 Install Rotating Screen Service

The Accel SHIM module was inserted into the Raspberry Pi when the RasPad was assembled, and now we write a script to enable the RasPad to automatically rotate the screen.

This script currently supports the following systems.

#### Currently supports

- Raspberry Pi OS
- Twister OS
- Ubuntu Desktop 21.04

**Run the following commands to install it.**

```
git clone https://github.com/raspad-tablet/raspad-auto-rotator
cd raspad-auto-rotator
sudo python3 install.py
```

After installation, you may be prompted to reboot, if so, the auto-rotate function will work after restarting RasPad. If there is no prompt to reboot, the auto-rotate function will also work.

### Calibration

After the first installation, some angles may not trigger the auto-rotation function, please try to turn it in the opposite direction for one second and then turn it back again, it will automatically calibrate. Or rotate RasPad slowly 720 degrees on three axes and it will also auto-calibrate.

If it still doesn't work at some angles no matter how you rotate and calibrate it, in that case, try to recalibrate it with the command.

```
raspad-auto-rotator reset
```

After running the above command, the auto-rotator will restart. Now recalibrate according to the 2 methods mentioned before.

## Chapter 5

# RASPAD LAUNCHER

### 5.1 Introduction

RasPad Launcher is an open source software that simulates a launcher menu, with a focus on improving the users touchscreen experience with the RasPad or other touchscreens.

---

**Note:** If you want to use the original desktop of the Raspberry Pi, you can skip this chapter.

---

## 5.2 Installation Guide

Download the RasPad Launcher package, and extract it.

```
wget https://github.com/raspad-tablet/raspad-launcher/releases/latest/download/raspad-
↳launcher.zip
unzip raspad-launcher.zip
cd raspad-launcher
```

The script will install the following:

- RasPad launcher with desktop profile.
- RasPad FAQ desktop profile (Just a quick icon to browser RasPad FAQ webpage).
- display auto rotate for Accl SHIM.

Run install script.

```
chmod +x install
sudo ./install
```

---

**Note:** To manually install RasPad Launcher and specific components, please refer to RasPad Launcher.

---

## Chapter 6

# INTERFACES INTRODUCTION

- **Micro SD Card Port:** The SD card slot is a snap-in style attached to the back cover. Before removing or replacing the back cover, remove the micro-SD card to avoid damaging the micro-SD card and the RasPad's internal button board.
- **Power Button:** Press and hold for over three seconds to power the RasPad on or off. A short press will turn the screen on or off.
- **Volume/+:** Press to enter the volume adjustment state. If **Brightness/-** is pressed first, it can be used to increase the brightness when pressed this button.
- **Brightness/-:** Press to enter the brightness adjustment state; If **Volume/+** is pressed first, it can be used to decrease the volume when pressed this button.

- **Battery Indicator:** Lights up in green. When charging, the corresponding LED flashes; when fully charged, all 3 LEDs are on.
- **Power Indicator:** When RasPad is turned on, it lights up as red.
- **Ethernet Port:** Standard Ethernet port.
- **USB 3.0 Port:** Standard USB 3.0 port.
- **HDMI Port:** It can be used as an expansion screen through this port.
- **Headphone Jack:** Standard headphone jack.
- **Power Supply Port:** 15V/2A DC port.

### 6.1 Charge the RasPad

The number of green LEDs indicates battery percentage. When the low battery warning icon appears, it indicates that RasPad needs to be charged.

Take out the power adapter and power cable.

Insert one side of the power cable into the power adapter, and the other side to an outlet.

Plug the power adapter into the RasPad power port.

At this time, one of the green LED might be blinking and the charging icon will appear, indicating that the RasPad is charging.



## **6.2 Volume and Brightness Settings**

Next to the power button, the red LED should now be lit, indicating that the RasPad is powered on.

At this point, the RasPad should be booted up.

If the screen is too bright, give the brightness button a short press to enter the screen-brightness adjustment page.

A brightness adjustment bar will appear on the screen.

The brightness can be adjusted via these two buttons.

After three seconds without operation, the brightness bar on the screen automatically disappears. The brightness should not be too high or too low, otherwise it will be harmful to the eyes.

To adjust the volume, press the volume button once to enter the volume adjustment page.

Then press these two buttons to adjust the volume.

## 6.3 Connect Headphones or Other Screens

Plug in the headphone if you need.

The headphone volume can only be adjusted on the desktop screen. Enter the Raspberry Pi desktop by tapping the top left menu icon of the RasPad Launcher.

Tap the speaker icon and the volume control box will appear. Slide the volume cursor up or down to adjust the headphone's volume.

**Warning:** When wearing headphones, do not adjust the volume too high.

Use the HDMI port to add a second monitor, and the USB ports for a mouse and a keyboard.







## Chapter 7

# RASPAD PROJECTS AND ACTIVITIES

To get started using the RasPad, this chapter includes some interesting and challenging projects to choose from.

### 7.1 Multimedia Options

Watch videos, read e-books, or play games on the RasPad.

#### 7.1.1 Watch Video

The RasPad has a built-in video player: VLC Video Player.

Watch a local video directly by double-clicking on the VLC Video Player icon, or log in to the YouTube website from Google Chrome browser to watch videos.

### 7.1.2 Read E-book

Read online from the Google Chrome browser.

### **7.1.3 Play Game**

Click **Preferences** -> **Recommended Software** -> **Games** to install games on RasPad 3.

For example, install **Minecraft**, and plug in a keyboard and mouse to play.

## 7.2 Retro Games Console

### 7.2.1 Description

You can turn RasPad 3 into a retro games console playing with your friends, let's see how we can do it!

### **7.2.2 Required Components**

- A RasPad 3
- 8G+ MicroSD Card
- Micro-SD Card Reader
- Keyboard
- Gamepad

It is recommended to use a Raspberry Pi 4 as the main control board, with Retro Pie as the Operating System.

That Raspberry Pi uploads or downloads the game system and game ROM needs taking up a large memory, so it is recommended to use a large-capacity SD card to avoid configuration failures.

When playing games, a gamepad and a keyboard are needed.

The RasPad is equipped with a 1280x800 LCD touch screen, a 2-watt stereo speaker, and three USB 3.0 ports, allowing for high resolution and sound quality to provide an excellent gaming experience.

### **7.2.3 Game System Installation**

RetroPie allows you to turn your Raspberry Pi, ODroid C1/C2, or PC into a retro-gaming machine. It builds upon Raspbian OS, Emulation Station, RetroArch and many other projects to enable you to play your favorite Arcade, home-console, and classic PC games with the minimum set-up.

Installing **RetroPie**:

**Step 1:** Download the SD image compatible with the Raspberry Pi 4 on the RetroPie official website.



**Step 2:** After the download is complete, unzip the downloaded package containing the image file.

**Step 3** Then flash the RetroPie image into the micro-SD card.

- For Windows, use: Raspberry Pi Imager, Etcher, or Win32DiskImager.

---

**Note:** Win32DiskImager requires an .img file extracted from the .img.gz image downloaded in step 2. You can use a program like 7zip to do this.

---

- For macOS, use: Raspberry Pi Imager, Etcher, Apple Pi Baker, or the dd command.
- For Linux, use: Raspberry Pi Imager, Etcher, or the dd command

---

**Note:** MacOS/Linux users can optionally extract the .img image from the downloaded .img.gz by using gunzip (macOS users can also simply double-click it).

---

**Step 4:** Insert the micro-SD card into the RasPad, and press the power button to boot up the system.

**Warning:** The SD card slot is a snap-in style attached to the back cover. Before removing or replacing the back cover, remove the micro-SD card to avoid damaging the micro-SD card and the RasPad's internal button board.

### 7.2.4 RetroPie Configuration

After the RasPad boots up, the Controller and WiFi settings should be configured, as well as transferring game ROMs. A keyboard and a gamepad are needed when doing these steps.

The detailed steps are shown in the video:

---

**Note:**

1. You can also go to RetroPie official website to detailed tutorial: [RetroPie Docs](#).
  2. RetroPie allows you to turn your Raspberry Pi or PC into a retro-gaming machine. But because of the nature/complexity of copyright/intellectual property law (country-specific), RetroPie doesn't provide ROMs for games. If you want to get them, you can download from the forum or Google to find the sources, then place one ROM under the directory of RetroPie emulator.
- 

Below is an example of the RasPad using the ROM of Super Mario 3.

## 7.3 3D Printer Monitor

When using a 3D printer, we will need to use OctoPrint. It is an open source 3D printer controller application, which provides a web interface for the connected printers. It displays printers' status and key parameters and allows user to schedule prints and remotely control the printer.

Please refer to the detailed installation tutorial for OctoPrint: <https://community.octoprint.org/t/setting-up-octoprint-on-a-raspberry-pi-running-raspbian-or-raspberry-pi-os/2337>.

This tutorial has written very detailed installation steps, which may take a long time and requires more patience.

---

**Note:**

- The Raspberry Pi Operating System will need to be installed on the micro-SD card before starting the tutorial.

Please refer to *Install the Raspberry Pi OS*.

- **Webcam Option:** A camera will need to be installed on the RasPad for webcam use. Please refer to the tutorial: *Assemble the Camera Module*.
  - **Touch UI Option:** A new Raspberry Pi image might not have an auto start function for Touch UI. Please refer to the tutorial: *Touch UI* to configure the settings for the Touch UI function.
- 

### 7.3.1 Touch UI

If the autostart file is not located in the `~/.config/lxsession/LXDE-pi` path, the file will need to be added manually.

Create the `lxsession` folder and the `LXDE-pi` folder in the `~/config` directory.

```
mkdir ~/.config/lxsession
mkdir ~/.config/lxsession/LXDE-pi
```

Copy the autostart file from the path `/etc/xdg/lxsession/LXDE-pi` to the `~/.config/lxsession/LXDE-pi` folder.

```
cp /etc/xdg/lxsession/LXDE-pi/autostart ~/.config/lxsession/LXDE-pi/autostart
```

Set the permissions of the autostart file to be readable and writable.

```
chmod 644 ~/.config/lxsession/LXDE-pi/autostart
nano .config/lxsession/LXDE-pi/autostart
```

Open the autostart file with a text editor such as Nano, and add the following line to the end of the file to make the RasPad execute the `startTouchUI.sh` script file on boot.

```
@/home/pi/startTouchUI.sh
```

After restarting the RasPad, the OctoPrint's Touch UI will open in full screen mode. Press F11 to exit the full screen mode and enter the desktop.

### 7.3.2 Make a 3D Model

Click this link: <https://projects.raspberrypi.org/en/projects?hardware%5B%5D=3d-printer>, refer to the official Raspberry Pi tutorial, you can get the 3D model file in the format of `.stl`.

Generally, 3D printers cannot directly process `.stl` files. You need to use **Ultimaker Cura** software to slice them, and then upload them to the 3D printer through **OctoPrint** to print the 3D model file.

Download Ultimaker Cura. Since Ultimaker Cura is not available on the Raspberry Pi system, you need to perform the slicing operation on your computer.

Select the version you need.

When installing Ultimaker Cura, please note that in the **choose components** step, **Open STL files with Cura** has been checked by default, so that .stl files can be sliced.

If you want to slice other types of model files, check the corresponding option, otherwise you can install it directly.

When opening Ultimaker Cura for the first time, there will be a series of configuration prompts. At the **Add a Printer** step, select the model of printer used and click **Next**.

After selecting the correct printer model, verify that the parameters provided by Ultimaker Cura in the Machine Settings page are correct, or change the parameters directly.

Follow the onscreen prompts to complete the configuration of Ultimaker Cura.

Once Ultimaker Cura has successfully launched, click the **Folder** icon in the upper left-hand corner and browse to the folder with the .stl 3D model file that needs to be sliced, and click **Open** to add the .stl file to Ultimaker Cura's library.



After the file has been added, click the **Slice** option in the lower right-hand corner, and Ultimaker Cura will automatically perform the slicing operation.

After slicing is complete, click the **Save to Disk** option in the lower right corner to save the sliced file locally.

Select the type of file extension recognized by the 3D printer, then click **Save**.

### 7.3.3 Print 3D Model

After the .stl file has been sliced, the 3D model file can be sent to the 3D printer through OctoPrint to be printed.

Open the RasPad's browser, and enter `http://192.168.18.179/?#temp` to log in to OctoPrint.

---

**Note:** Before logging in to the OctoPrint's web UI, OctoPrint will first need to have been successfully installed on the RasPad.

The IP address 192.168.18.179 will need to be replaced with the local IP address of the RasPad. Hover the cursor over the WiFi icon on the RasPad desktop, and the local IP address will be displayed.

---

Now you have entered OctoPrint.

Click the **Upload** option to select the sliced 3D model file.

Click the **print** icon. The 3D printer will start to print the 3D model file after the slicing process is complete.

If you have transferred the sliced file to the Raspberry Pi, you can also open the OctoPrint UI in RasPad to print.

### 7.3.4 Video

The following video shows that after installing OctoPrint, connect your 3D printer and RasPad 3 through a USB cable-upload the designed 3D file, and then use the camera to monitor the printing process.

The temperature can also be monitored to prevent the 3D printer from getting too hot or cold, which will affect the printed 3D model.

## 7.4 Home Assistant

Home Assistant is a Python-based open source smart home system that supports smart home devices of many brands and can easily implement voice control and automation of the devices. You can run Home-assistant on the Raspberry Pi, without having to send your data out onto the internet. This means that you have total control over your data, limit the amount of internet traffic from your smart devices, and tighten up security.

And RasPad 3 is powered by Raspberry Pi 4 with 10.1" HD touch screen and 5 hours battery life. We also wrote scripts for it to boot up and full screen self-start, so you can use Home Assistant on RasPad more easily and quickly.

### Expand Viewing

Here are some projects made by others that combine Raspberry Pi and Home Assistant.

- My cheap smart home system | HomeAssistant Wall Panel - AGT Designs
- Getting started with Home Assistant - JuanMTech
- DIY Home Automation Intruder Alarm System! || Home Assistant + Raspberry Pi + ESP8266 - GreatScott!

### Quick Guide

## 7.4.1 Install Home Assistant

### Install Docker

Here will introduce how to use the container environment to install and run Home Assistant. We recommend using Docker as the container environment. The way to install Docker is very simple, just run the following commands.

```
curl -fsSL https://get.docker.com -o get-docker.sh
sudo sh get-docker.sh
```

---

**Note:** You can see more details in [Install Docker Engine on Debian](#).

---

### Install Home Assistant

After installing the container environment, if you are using **Raspberry Pi 3**, run the following commands to install the Home Assistant.

```
sudo docker run -d \
--name homeassistant \
--privileged \
--restart=unless-stopped \
-e TZ=MY_TIME_ZONE \
-v /home/pi/homeassistant:/config \
--network=host \
ghcr.io/home-assistant/raspberrypi3-homeassistant:stable
```

If you are using **Raspberry Pi 4**, run the following commands to install the Home Assistant.

```
sudo docker run -d \
--name homeassistant \
--privileged \
--restart=unless-stopped \
-e TZ=MY_TIME_ZONE \
-v /home/pi/homeassistant:/config \
--network=host \
ghcr.io/home-assistant/raspberrypi4-homeassistant:stable
```

---

**Note:** For more details, please see [Install Home Assistant Container](#).

---

### Start the Home Assistant Service

After the installation is complete, you can run the following command to start Home Assistant.

```
sudo docker start homeassistant
```

Use the following command to close Home Assistant.

```
sudo docker stop homeassistant
```

Restart Home Assistant with the following command.

```
sudo docker restart homeassistant
```

### Enter Home Assistant Page

Now you can enter `http://<localhost>:8123` in the browser to enter the Home Assistant operation interface. For example, my Raspberry Pi IP is 192.168.6.136, then visit `http://192.168.6.136:8123`.

---

#### Note:

1. If you need to configure boot-up, please refer to: *Start the Home Assistant Service at Boot*
  2. If you need to configure a full screen boot, please refer to: *Enter the Home Assistant Page at Boot*
- 

## 7.4.2 Install and configure ESPHome

---

**Note:** If you want to add ESPHome in Home Assistant, then you need to go through this section to install and configure it.

If you don't use it, you can skip this section and go directly to the next one: *Log In*.

---

ESPHome is the perfect solution for creating custom firmware for your ESP8266/ESP32 boards.

Here's a very quick introduction to how ESPHome works: ESPHome is a tool which aims to make managing your ESP boards as simple as possible. It reads in a YAML configuration file (just like Home Assistant) and creates custom firmware which it installs on your ESP device. Devices or sensors added in ESPHome's configuration will automatically show up in Home Assistant's UI.

### Installation

1. Installation dependencies.

```
sudo pip3 install cryptography==2.8
```

2. Install ESPHome, you need to install Python and install the console script through pip3.

---

**Note:** Python 3.7 or above is required to install ESPHome 1.18.0 or above.

---

```
sudo pip3 install esphome
```

Alternatively, there's also a Docker image available for easy installation (the Docker hub image is available for AMD64, ARM and ARM64(AARCH64) right now; if you have another architecture, please install ESPHome through pip or use the Home Assistant add-on:

```
docker pull esphome/esphome
```

3. Open the ESPHome Dashboard.

---

**Note:** You need to keep this command running while you are operating in the ESPHome dashboard.

---

`esphome dashboard config/`

### Configure ESPHome

1. Open a browser and enter ESPHome Dashboard through `http://localhost:6052` or `http://X.X.X.X:6052` (replace X.X.X.X with your Raspberry Pi's IP address).
2. Create configuration.

Click on the "+" to start adding configurations.

Enter a name and the WiFi and password that you need to have your device connect to and then click **NEXT**.



Select the device type (ESP32 for example).

Enter

Once added, you will see them in the Dashboard of ESPHome.

---

**Note:** When it is not the first time to add a device, click "+" on the main page, follow the prompts to enter the device name, WIFI and password, select the device type.

---

3. Edit the `.yaml` file.

Click **EDIT** in the bottom right corner to go inside the `.yaml` file, you will see the following default components.

- **esphome:** Contains the name, platform, and board type that you set up.
- **logger:** The logger component automatically logs all log messages through the serial port and through MQTT topics.
- **api:** The ESPHome native API is used to communicate with clients directly, with a highly-optimized network protocol. Currently, only the ESPHome tool and Home Assistant use this native API.
- **ota:** With the OTA (Over The Air) update component you can upload your firmware binaries to your node without having to use a USB cable for uploads.
- **wifi:** This core ESPHome component sets up WiFi connections to access points for you. It needs to be in your configuration or otherwise ESPHome will fail in the config validation stage.
- **captive\_portal:** The captive portal component in ESPHome is a fallback mechanism for when connecting to

the configured WiFi fails. After 1 minute of unsuccessful WiFi connection attempts, the ESP will start a WiFi hotspot (with the credentials from your configuration).

Now start to add other components. There are a lot of components in the ESPHome official website, including **Sensor Components**, **Output Components**, **Light Components**, etc., Here we take **Binary Light** in **Light Components** as an example.

Add the following code to the end of the .yaml file.

```
# Example configuration entry
light:
  - platform: binary
    name: "Desk Lamp"
    output: light_output

output:
  - id: light_output
    platform: gpio
    pin: GPIO16
```

4. Install the .yaml file to ESP32 board.

After editing, click **INSTALL** in the lower right corner to compile and install. There are 2 methods for you to choose: Wirelessly and USB Port. But for the first installation you need to connect the ESP32 board to the Raspberry Pi using USB and then choose the second installation method. At a later time you can install and compile via Wirelessly.

It takes about 10 minutes to download the dependencies during the first compilation, so please be patient.

After successful installation, if you connect an LED to GPIO16 in ESP32 board, you will see the LED is lit. Also please record this IP, you will need to fill in this IP when you add ESPHome to Home Assistant.

---

**Note:** If there is no response when clicking INSTALL, please clear the browser cache and INSTALL again.

```
sudo rm -rf ~/.cache/chromium
```

---

ESPHome configuration is complete and you can now add it to Home Assistant.

### 7.4.3 Log In

1. Now you can enter `http://<ip>:8123` in the browser to enter the Home Assistant operation interface. Then set up a username and password for your Home Assistant.

- 
2. Create a name for your home and select your region.



3. Select the content you want to share to Home Assistant.

4. You can choose to configure the device now, or later. It is recommended that beginners configure the device later, as the configuration is described in detail later in the tutorial.

5. After setting up, enter your login username and password in the login screen.

---

**Note:** If the login screen does not jump automatically, please open Home Assistant again via <http://localhost:8123>.

---

After logging in successfully, you will enter your Home.

### 7.4.4 Add Smart Devices

You can add your purchased smart devices to Home Assistant, such as smart sound, smart desk lamp, etc. You can also make your DIY device with ESP32 and add it in.

In this section, you will learn how to add your smart devices.

#### 1. Configure the Smart Device

You need to make sure that your smart device has been assigned an IP. You can complete this step through the corresponding APP of the smart device. For example, the picture below shows a bedside lamp configured with **HomeKit**.

## **2. Add Integration**

Now, visit <http://ip:8123> to access your Home Assistant, then click Configuration in the left column and select Integrations.

If Home Assistant finds a device on your network, it will display the corresponding integration, which can be easily added with just a few clicks.

If your smart device has not been discovered yet, don't worry, click the + **ADD INTEGRATION** button at the bottom right and search for your integration in the list.

Each integration may be different in use, you can visit [Home Assistant Integration](#) for details.

### Edit Dashboard

Now a CARD needs to be added to control this smart device.

Return to the Overview page. If the device you just added does not appear, you need to **Edit Dashboard**.

Click **Overview** → **Edit Dashboard** → **ADD CARD**, you can select the corresponding CARD according to your needs, for example, select Button here, and then select the corresponding Entity.

After adding the corresponding card, click **Done** to exit editing, and then you can control your smart device.



### 7.4.5 Add DIY Device with ESP32

You can add your purchased smart devices to Home Assistant, such as smart sound, smart desk lamp, etc. You can also make your DIY device with ESP32 and add it in.

In this section, you will learn how to add your DIY device.

Before you start the following operations, you need to complete *Install and configure ESPHome*.

#### **Add Integration**

1. Now, visit <http://ip:8123> to access your Home Assistant, click **Settings** in the left column, select **Devices & Services**.

2. If Home Assistant finds a device on your network, it will display the corresponding integration, which can be easily added with just a few clicks. If your smart device is not yet discovered, don't worry, you can add it via the **+ ADD INTERGATION** button at the bottom right.

3. Search for ESPHome in the pop-up window and fill in the Host (the IP recorded after the INSTALL step in chapter *Install and configure ESPHome*), with the option to place your esp-light in the bedroom, kitchen or other area of your choice.

4. After clicking Finish, you will see it in the list.

### **Edit Dashboard**

Now a CARD needs to be added to control this esp-light.

Click **Overview** → **Edit Dashboard** → **ADD CARD**, you can select the corresponding CARD according to your needs, for example, select **Button** here, and then select the corresponding Entity.

After clicking **SAVE**, you will be able to use this **Button** to control the esp-light (if you connect an LED to GPIO16 in ESP32 board, click on the **Button** to make the LED light up or turn off).

For more styles, you can refer to Official Demo.

### 7.4.6 Start the Home Assistant Service at Boot

Normally, Docker is self-starting after booting by default, which also means that after you install Home Assistant, you can use it as long as you start the Raspberry Pi.

You can use the following command to view the boot list. The `docker.service` should be in the enabled state.

```
sudo systemctl list-unit-files | grep enable
```

To turn off Docker's self-startup, please enter the following command:

```
sudo systemctl disable docker.service
```

To enable Docker's self-startup, please enter the following command:

```
sudo systemctl enable docker.service
```

### 7.4.7 Enter the Home Assistant Page at Boot

Experience the full-screen self-boot interface first. Please close the already open Home Assistant URL and run `chromium-browser --start-fullscreen "http://localhost:8123"`.

If you want to automatically display the Home Assistant interface in full screen after booting, then do the following.

1. Go to the autostart folder.

```
cd /home/pi/.config/autostart/
```

---

**Note:** If the autostart folder does not exist, you need to create a new one.

```
sudo mkdir -p /home/pi/.config/autostart/
```

2. Create `chrome_start_fullscreen.desktop` file.

```
sudo nano chrome_start_fullscreen.desktop
```

3. Edit the `chrome_start_fullscreen.desktop` file as follows.

```
[Desktop Entry]
Type = Application
Exec = chromium-browser --start-fullscreen "http://localhost:8123"
```

Save and exit: `Ctrl + X, Y, Enter`.

---

**Note:** If you want to cancel the full-screen self-start, comment out the contents of the `.desktop` file (by `"#"`) and restart the Raspberry Pi.

```
cd /home/pi/.config/autostart/
sudo nano chrome_start_fullscreen.desktop
```

4. Exit full screen.

**computer:**

Method 1: Press F11.

Method 2: Move the mouse to the upper column of the screen and click the exit button that appears.

Method 3: Right-click the pop-up menu, select "Exit full screen".

**Touch screen:**

Long press the blank space to pop up the menu, click the exit button at the top of the screen or select "Exit full screen".

### 7.4.8 Home Assistant File Structure

Homeassistant creates a configuration file by default under the path of the executing user (created homeassistant) at the `~/.homeassistant` path.

The file directory structure is as follows:

- `.storage` directory contains a lot of user-related information, including user login information (username/password, encrypted in `auth_provider.homeassistant` file).
- `configuration.yaml`: User-edited configuration files.
- `home-assistant.log`: Run log (cleared with each reboot).
- `home-assistant_v2.db`: Database..
- `.storage`: Various elements of front-end configuration.



## 7.5 IoT

In this article, we'll show you how to connect your Raspberry Pi to the **Blynk** platform to make some interesting IoT projects.

Blynk is a full suite of software required to prototype, deploy, and remotely manage connected electronic devices at any scale: from personal IoT projects to millions of commercial connected products. With Blynk anyone can connect their hardware to the cloud and build a no-code iOS, Android, and web applications to analyze real-time and historical data coming from devices, control them remotely from anywhere in the world, receive important notifications, and much more...

### 7.5.1 Get Start with Blynk

You will learn how to use Blynk in this project.

In the event that you trigger widgets on Blynk, your Raspberry Pi will print out their values.

Follow the steps below, and note that you must do them in order and not skip any chapters.

## 1. Configuring the Blynk

1. Go to the BLYNK and click **START FREE**.
2. Fill in your email address to register an account.

3. Go to your email address to complete your account registration.

4. Afterwards, **Blynk Tour** will appear and you can read it to learn the basic information about the Blynk.

5. Next we need to create a template and device, click **Cancel**.

6. Go to Template from the navigation bar.

7. Create New Template

8. Fill in **NAME**, feel free to do so; **HARDWARE** should be **Raspberry Pi**. Then **Done**.

9. You will be redirected to the Info page, just click on save in the top right corner.

10. Go to **Search** page from the navigation bar.

11. Create New Device.

12. From template.

13. Select **TEMPLATE** as the one you just set, **DEVICE NAME** can be customized. Then click Create.

14. You should now see a page like this one, which means that your initial Blynk setup is complete.



## 2. Edit Dashboard

1. Click on the menu icon in the upper right corner and select edit dashboard.
2. Then drag any CONTROL Widgets you want onto the Dashboard. I dragged a Switch and a Slider.

3. Tap the setting icon on the widget.

4. Create Datastream, select Virtual Pin

5. Complete the datastream setup. Here is the datastream created for Switch, so **DATA TYPE** select Interger, **MIN** and **MAX** set to 0 and 1. Create and then Save.

6. Use the same steps to create a Datastream for the slider widget, and again, modify **DATA TYPE**, **MIN** and **MAX** to what you need.

7. When finished, click Save And Apply at the top right.

### 3. Install the Blynk library

Run the following command to install.

```
cd /home/pi
git clone https://github.com/vshymanskyi/blynk-library-python.git
cd blynk-library-python
sudo python3 setup.py
```

### 4. Download the Example

We have provided some examples, please run the following command to download them.

```
cd /home/pi
git clone https://github.com/sunfounder/blynk-raspberrypi-python.git
```

### 5. Run the Code

1. Go to Blynk's Device Info page, you will see some information under **FIRMWARE CONFIGURATION**, you need to copy **BLYNK\_AUTH\_TOKEN** down.

2. Edit the code.

```
cd /home/pi/blynk-raspberrypi-python
sudo nano blynk_start.py
```

3. Find the line below and past your BLYNK\_AUTH\_TOKEN.

```
BLYNK_AUTH = 'YourAuthToken'
```

4. Run the code.

```
sudo python3 blynk_start.py
```

5. Go to Blynk, and operate the widget on Dashboard.

6. Now you will be able to see your actions on the terminal.

```
..
  _ _ _ _ _
 / - )/ / - _ _ _ _ / / -
 / - / / // / - \ / ' - /
 / _ _ _ / - \ _ , / _ // - / \ _ \
      / _ _ / for Python v1.0.0 (linux)

Connecting to blynk.cloud:443...
Blynk ready. Ping: 142 ms
V0 value: ['1']
V0 value: ['0']
V1 value: ['3']
V1 value: ['8']
V0 value: ['1']
```

## 7.5.2 Smart Light

In this project, we use Blynk's Slider to control the brightness of the LED, turning it on and off with Switch.

**Note:** Before starting this project, we recommend that you complete *Get Start with Blynk*. The following will give you a clear understanding of Blynk.

### 1. Wiring

## **2. Create Widget and Datastream**

1. Click on the menu icon in the upper right corner and select edit dashboard.

2. Add a Switch widget and a Slider widget to the Dashboard.



3. Create a Datastream for the Switch widget (I used V3). It will be used to control the turning on and off of the LED.

4. Create a Datastream for the Slider widget (I used V2), its value range is 0 to 100, it will be used to control the brightness of the LED.

5. When finished, click Save And Apply at the top right.

### **3. Run the Code**

1. Edit the code

```
cd /home/pi/blynk-raspberrypi-python  
sudo nano blynk_light.py
```

2. Find the line below and past your BLYNK\_AUTH\_TOKEN.

```
BLYNK_AUTH = 'YourAuthToken'
```

3. Run the code.

```
sudo python3 blynk_light.py
```

4. Go to Blynk, operate widget on Dashboard. now you click switch widget will turn on/off LED. slide Slider widget will change LED brightness.

### 7.5.3 Door Window Sensor

When you're outside, you've probably had this confusion. "Are the doors and windows of my house closed?"

To solve this problem, in this project, we will build a door and window sensor with Reed Switch and magnets.

Install this sensor and magnet on both sides of the door or window. You will be able to check whether your doors and windows are closed or not from the Blynk APP on your phone.

---

**Note:** Before starting this project, we recommend that you complete *Get Start with Blynk*. The following will give you a clear understanding of Blynk.

---

#### 1. Wiring

## **2. Create Datastream**

1. Click on the menu icon in the upper right corner and select edit dashboard.

2. Go to the Datastreams page and create a New Datastream.

3. Create a Virtual Pin V4.

4. When finished, click Save And Apply at the top right.

### 3. Run the Code

1. Edit the code

```
cd /home/pi/blynk-raspberrypi-python  
sudo nano blynk_reed.py
```

2. Find the line below and past your BLYNK\_AUTH\_TOKEN.

```
BLYNK_AUTH = 'YourAuthToken'
```

3. Run the code.

```
sudo python3 blynk_reed.py
```

### 4. Open Blynk APP

---

**Note:** As datastreams can only be created in Blynk on the web, you will need to reference different projects to create datastreams on the web, then follow the tutorial below to create widgets in Blynk on your mobile device.

---

1. Open Google Play or APP Store on your mobile device and search for “Blynk IoT” (not Blynk(legacy)) to download.

2. After opening the APP, login in, this account should be the same as the account used on the web client.
3. Then go to **Dashboard** (if you don't have one, create one) and you will see that the **Dashboard** for mobile and web are independent of each other.

4. Click **Edit** Icon.
5. Click on the blank area.
6. Choose **LED** widget.

7. Now you will see a **LED** widget appear in the blank area, even if it looks like a blank grid, click on it.
8. **LED** Settings will appear, select the **V4** datastreams you just set in the web page. Note that each widget corresponds to a different datastream in each project.
9. Go back to the **Dashboard** page. Now if you see that the **LED** widget is filled with color, your door or window is open; if the **LED** widget is not filled with color, the door or window is closed.

### 7.5.4 Smart Fan

In this project, you can see the temperature from Blynk and turn on the fan remotely.

---

**Note:** Before starting this project, we recommend that you complete *Get Start with Blynk*. The following will give you a clear understanding of Blynk.

---

#### 1. Wiring



## **2. Create Widget and Datastream**

1. Click on the menu icon in the upper right corner and select edit dashboard.
2. Add a Switch widget and a Label widget to the Dashboard.
3. Create a Datastream (I used V3) for the Switch widget. It will be used to turn on the Motor.

4. Create a Datastream for the Label widget(I used V0). It will be used to display the temperature. Set **DATA TYPE** to String.

5. When finished, click Save And Apply at the top right.

### 3. Run the Code

1. Edit the code

```
cd /home/pi/blynk-raspberrypi-python  
sudo nano blynk_motor.py
```

2. Find the line below and past your BLYNK\_AUTH\_TOKEN.

```
BLYNK_AUTH = 'YourAuthToken'
```

3. Run the code.

```
sudo python3 blynk_motor.py
```

4. Go to Blynk, on the Dashboard you can check the temperature via Label widget; you can turn on/off the fan via Switch widget.

### 7.5.5 Temperature Recorder

In this project, you can see the current temperature and the temperature change line graph from Blynk.

---

**Note:** Before starting this project, we recommend that you complete *Get Start with Blynk*. The following will give you a clear understanding of Blynk.

---

#### 1. Wiring

#### 2. Create Widget and Datastream

1. Click on the menu icon in the upper right corner and select edit dashboard.

2. Add a Gauge widget and a Chart widget to the Dashboard.

3. Create a Datastream for the Gauge widget (I used V5). It will be used to display the temperature. Set **DATA TYPE** to Double, **DECIMALS** to #. # (two valid decimal places).

4. Add the V5 Datastream you just created to the Chart widget.

5. When finished, click Save And Apply at the top right.

### 3. Run the Code

1. Edit the code

```
cd /home/pi/blynk-raspberrypi-python  
sudo nano blynk_temp.py
```

2. Find the line below and past your BLYNK\_AUTH\_TOKEN.

```
BLYNK_AUTH = 'YourAuthToken'
```

3. Run the code.

```
sudo python3 blynk_temp.py
```

4. Go to Blynk. Now you can view the temperature and temperature change line graph on the Dashboard.

## 7.6 GPIO Programming

RasPad 3 can connect the 40 I/O pins of the Raspberry Pi to the breadboard, so you can use these pins to do some interesting projects.

---

**Note:** Refer to *GPIO Extension Board* instructions for extending the 40 pins of the Raspberry Pi inside the RasPad to the external breadboard.

---

SunFounder has a very complete course system about Raspberry Pi GPIO learning kits, such as the following two learning kits.

- *Electronic Kit for Raspberry Pi and Arduino*

This basic learning kit is suitable for the Raspberry Pi and Arduino novice. This kit teaches basic circuit design and introduces Python programming for beginners.

- *Da Vinci Kit for Raspberry Pi*

This is an advanced Raspberry Pi teaching kit that contains a robust array of components. This kit teaches physical computing and Python programming principles for beginners and advanced users.



## 7.6.1 Electronic Kit for Raspberry Pi and Arduino

### Tutorial Link

<https://docs.sunfounder.com/projects/electronic-kit/en/latest/index.html>

**Product Link** Global: <https://www.sunfounder.com/products/electronics-fun-kit>

United States: <https://www.amazon.com/dp/product/B07QT78FXF>

Germany: <https://www.amazon.de/dp/B07QT78FXF>

Canada: <https://www.amazon.ca/dp/product/B07RQ7Z6G4>

## 7.6.2 Da Vinci Kit for Raspberry Pi

### Tutorial Link

<https://docs.sunfounder.com/projects/davinci-kit-for-raspberry-pi/en/latest/>

### Product Link

Global: <https://www.sunfounder.com/products/sunfounder-davinci-kit>

United States: <https://www.amazon.com/dp/product/B07WV2HYC6>

Japan: <https://www.amazon.co.jp/dp/product/B083SJQXCL>

Canada: <https://www.amazon.de/dp/B07WYX8M76>

Germany: <https://www.amazon.ca/dp/product/B07KH2C215>

## 7.7 Scratch Projects

In this chapter, you will learn to use Scratch on RasPad 3, which includes 10 examples.

If you are a user who has just used Scratch, we recommend that you try the teaching examples in order so that you can quickly get started with Scratch.

If the Raspberry Pi system you downloaded comes with recommended software, you can find Scratch 3 in **Programming**.

If you download a system with only a desktop, you can click **Preferences -> Recommended Software -> Programming** to install the Scratch 3 on RasPad 3.

---

**Note:** Before trying the teaching examples, you should have downloaded the relevant materials and code files. Open a Terminal and enter the following command to download them from github.

```
git clone https://github.com/sunfounder/raspad3.git
```

---

### Projects

#### 7.7.1 Animate A Name

##### Description

Today we will learn to write our name on Scratch. When we click on the letters in the name, they will become active.

Click the green flag to start.

Or click Animate A Name, and then learn online tutorial on the Scratch official website.

## **Required Components**

- A RasPad 3
- Scratch 3 (either online or offline)

## **You Will Learn**

- Choose the sprite you want.
- Program different sprites separately.
- Make the sprite change color.
- Let the sprite spin.
- Let the sprite zoom.
- Let the sprite flash.

## **Lesson Guide**

### **My Name is Abby.**

In the lower right corner of Scratch, there are two areas. We can add the sprite and backdrop we want here.

First delete the original sprite, we don't need it.

Then choose a sprite.

Find the Letter sprite.

Repeat several times to spell out your name, like me, “Abby”.



**A can change its color.**

Click on the Glow-A sprite and let's write a bit of code for it.

When **A** is clicked, it changes its color.

Click on the sprite, find **when this sprite clicked** in Events, and drag it to the right coding area.

Drag **change color effect by...** from **Looks** to the bottom of when this sprite clicked.

At this time, we click on the letter **A** on the stage and it will change its color.

**B can rotate 360°.**

Click on the Glow-B sprite, let's write a code for the Glow-B wizard. When the letter **B** on the stage is clicked, it will rotate in a circle.



Do you know why B is rotated 15 degrees 24 times? Because  $24 \times 15 = 360$ , it turns 360 degrees!

**B2 can zoom in and out.**

Click on the Glow-B2 sprite, and let us write a code for the Glow-B2 sprite.

When the second **B** on the stage is clicked, it will zoom in and then zoom out.

**Y can show and disappear.**

Click on the Glow-Y sprite and let us write a piece of code for the Glow-Y sprite. When the **Y** on the stage is clicked, it will repeatedly disappear and appear.

### Challenge

I believe that you will be smart enough to program and implement this game soon. Next, we will add some challenges to enrich our game content.

- Add two new sprites, one of which will blink and change color when clicked; the other will zoom and rotate when clicked.

### 7.7.2 Create a Story

#### Description

Do you like to hear stories? For example, before going to bed, you can have a sweet dream at night by listening to a little story from your parents. Today we are going to “tell” the story of a little monkey looking for a banana on Scratch.

Click the green flag to start.

Or click Create a Story, and then learn online tutorial on the Scratch official website.

### Required Components

- A RasPad 3
- Scratch 3 (either online or offline)

### You Will Learn

- Choose the sprite you want.
- Let the sprites move around.
- Let the sprites talk to each other.
- Switch scenes.

### Lesson Guide

**This is a happy Monkey named Harley.**

In the lower right corner of the Scratch, there are two areas-the character area and the stage area, where we can add the sprites and stage we want.

First the current sprite, and then click the **Choose a Sprite** button to add the sprite.

Here we need to find Monkey, click on it! You can use the search box above to find it. Here we give it a name **Harley**.

**Harley is hungry. He asked his friend Perry where is the banana?**

Click on your Harley sprite, and then click on the Event category. Drag **when clicked** to the coding area on the right.

Click on the Looks category and drag **say “Hello!” for 2 second** to the bottom of “when clicked”.

Change what it says and how long it says in the input box.

**Perry had seen bananas in the park, so he told Harley.**

Perry is a Frog, add Frog sprite and Forest backdrop.

Click on the Frog sprite, add a 3 second **delay** block (wait for Harley to finish speaking), and then it speaks.



Now, click on the green flag, we can see Harley and Perry chat.

**Harley went to the park and found bananas on the table.**

Now, click on the Backdrop and let it switch after 6 seconds (after the conversation ends).

Now, by clicking on the green flag, we can see that the Perry and Harley “come” to the park together after the conversation.

But Harley should come to the park alone, and there should be bananas on the table. So, we have to make Perry “disappear”.

Click on the Frog sprite, add an event, **when backdrop switches to...**, and find “hide” and “show” in Looks. Let the Perry appear at the right time!

Add the Banana sprite and adjust the angle.

Also let it appear at the right time.

Now, click on the green flag and you can see a complete story.

### **Challenge**

Believe that you who are smart will be able to program and implement this game soon, then we will add some challenges to enrich our story content.

- Harley will take the bananas back to the Forest, share it with Perry.

## **7.7.3 Make Music**

### **Description**

Everyone has been to the concert hall. It is filled with a dazzling array of musical instruments. These musical instruments make wonderful sounds under the performance of musicians.

Today we will also be a musician and add some musical instruments on the stage of Scratch. When you click on different instruments, they will emit corresponding instrument sounds.

Click the green flag to start.

Or click Make Music, and then learn online tutorial on the Scratch official website.

### Required Components

- A RasPad 3
- Scratch 3 (either online or offline)

### You Will Learn

- Use the Scratch **Add Extension** function to add music extension.
- Modify the initial size of the sprite.
- Let the sprites do some actions.

### Lesson Guide

#### We Need the Drums.

Click on the “Add Extension” icon at the bottom left of Scratch, select **Music**, and then you will find a new extension-Music on the left side of Scratch.

Delete the original sprite, add Drum-highhat sprite and Drum-snare sprite.

Adjust the Drum-highhat sprite and Drum-snare sprite to the appropriate size.



### Play Drum-highhat.

Click on the Drum-highhat sprite and drag out the **play drum...** block in the Music expansion module.

Change the option to **(5) Open Hi-Hat**, then use the “next costume” block to switch the appearance of Drum-highhat.

Now, you can play the Drum-highhat.



### **Play Drum-snare.**

Click on the Drum-snare sprite and drag out the **play drum...** block in the Music expansion module.

Change the option to **(1) Snare Drum**, then use the “next costume” block to switch the appearance of Drum-snare.

Now, you can play the Drum-snare.

## Challenge

I believe that you will be smart enough to program and implement this game soon. Next, we will add some challenges to enrich our game content.

- Add Drum sprite, Drum Kit sprite and Drum-cymbal sprite, modify their size, and choose suitable sound effects. In this way we have made a drum set.

Sprite	Size	Instrument tone options
Drum	200	(3)Side Stick
Drum Kit	150	(2)Bass Drum
Drum-cymbal	200	(4)Crash Cymbal

## 7.7.4 Clicker Game

### Description

Today we will play a game of poking balloons.

Click the green flag to start.

Or click Clicker Game, and then learn online tutorial on the Scratch official website.

### Required Components

- A RasPad 3
- Scratch 3 (either online or offline)

### You Will Learn

- Pick random number.
- Change the position, size and color of the sprite.
- Make a variable.

### Lesson Guide

#### Let the balloons appear randomly.

First, add Balloon1 sprite and Colorful City backdrop.

Set the balloon to a random position.

Let it change the color and size randomly.

Now, run the green flag and you can see the balloons randomly change in the stage area.

**Click on the balloon to add score.**

Now, we need to establish a scoring mechanism. Whenever the balloon is clicked, the score will increase by one.

Create a variable (it can help us keep track of and store the score). In Variables, select **Make a Variable** and named **score**.

We will see it in the upper left corner of the stage.

When the balloon is clicked, the score is increased by one. When you click the green flag (the game starts), the score will be reset to zero.

Now, click the green flag above the stage area to start the game, and click the balloon to increase your score!

### **Challenge**

I believe that you will be smart enough to program and implement this game soon. Next, we will add some challenges to enrich our game content.

- When the balloon is clicked, the sound effect of “POP” is emitted.
- Let the Pufferfish sprite appear randomly on the stage as well, if you click on it, you will lose score!

## **7.7.5 Shooting Game**

### **Description**

Have you seen those shooting games on TV? The closer a contestant shoots a bullet on the target to the bullseye, the higher his score.

Today we are also doing a shooting game in Scratch. In the game, let the Crosshair shoot as far as possible to the bullseye to get a higher score.

Click on the green flag to start. Use the space bar to shoot an bullet.

Or click Shooting Game, and then learn online tutorial on the Scratch official website.

### **Required Components**

- A RasPad 3
- Scratch 3 (either online or offline)
- Crosshair Sprite
- Target Backdrop

### You Will Learn

- Upload new sprites and background images to Scratch.
- The coordinate range of the Scratch stage.
- Stop scripts run by other sprites and get the result of the game.
- Use of Scratch color picker.

### Lesson Guide

#### Arrange the Crosshair and targets.

First, delete the previous sprite, upload the Crosshair sprite and the target background, you can find them under the path `./raspad3/scratch/materials` (You have downloaded from `raspad3.git` before.).

Click the icon below the sprite list, select “Upload Sprite Option”, find the **Cross Hair.svg** file, and upload it. Similarly, upload the backdrop, **Target.jpg**.







**Let's take aim.**

When aiming, the Crosshair will slightly shake, we are now going to use the **glide...** block to achieve this effect.

Set the random position and size of the Crosshair sprite.

Now, click on the green flag to see the effect of this aiming...very good. Let's guess, change the time in the **glide...** block, will the front sight become more stable? Or is it more swaying?

### **Let's shoot!**

We are now going to express the effect of shooting bullets.

how is it? Stop the crosshair moving, and then let it shrink quickly to create a shooting effect.

This means that when the space bar is pressed, we need to stop shaking the code of the Crosshair and keep the size of the Crosshair sprite shrinking.

Click the green flag to try the effect!

### **Calculate the score!**

Then we need to establish a scoring mechanism. Shooting different color circles will get different scores. The closer to the bullseye, the higher the score.

For example, a shot in the red area can score 200 points. Let's use the **Touching Color** block to judge. This block has a color picker, and we use it to pick up the red color on the target.

Use the same method to set corresponding scores for different color circles.



### Challenge

I believe that you will be smart enough to program and implement this game soon. Next, we will add some challenges to enrich our game content.

- In a game, you can shoot three rounds and then calculate the total score.

### 7.7.6 Hit the Magic Fish

#### Description

Before the project Starts, I will tell you the story of a magic fish. In the vast underwater world, there is a magical magic fish. When you hit it with a Starfish, it will become another fish.

Today we use the scratch stage to simulate an underwater world, and then add the Star sprite and fish sprite to make a “hit the magic fish” mini game.

Click on the green flag to Start. Press the space bar to launch the Starfish.

Or click Hit the magic fish, and then learn online tutorial on the Scratch official website.

#### Required Components

- A RasPad 3
- Scratch 3 (either online or offline)

#### You Will Learn

- Let the sprite bounce when it hits the edge of the stage.
- Change the coordinates to make the sprite move.
- Make a timer.
- Use the broadcast to tell the sprite what happened.

#### Lesson Guide

##### This fish is swimming.

Added **Fish** sprite, **Star** sprite and **Underwater1** backdrop image.



Let the Fish appear at the coordinates (-240,120), and then let it swim left and right.

Use the **if on edge, bounce** block to make the Fish sprite fold back from the edge. When it hits the wall, it will fold back in the opposite direction to the original.

### **This Star is rising.**

Let the Star appear at (0,-120). If we press the ← or → key on the keyboard, it will go left and right.

We use the **key ... pressed** block to determine the keyboard, and use **change x by ...** to move the sprite.

Next, launch the Star. If the space bar is pressed, the y coordinate value of the Star sprite will increase, and the Star sprite will rise.

Something is wrong. The Star sprite should continue to rise until it reaches the height of the Fish sprite (remember?

The y coordinate we set for Fish is 120) and then return.

NowIt looks better.

### **Fish hit by a Star.**

After the Star rises, before returning to the bottom, you need to use a **touch...** block to determine whether it hits the Fish.

When the Star sprite hits the Fish sprite, we need to tell the Fish sprite about this and let it do the corresponding behavior-change its shape.

We set up a “Hit” broadcast. This broadcast will tell the entire program that something called “Hit” has happened. All sprite and even the backdrop can hear this broadcast and respond accordingly.

Back to the programming interface of the Fish sprite, use a **when I receive...** block to receive the “Hit” broadcast, and then use **Next costume** to change your form.

### **Calculate the score!**

To make the game more interesting! Add a scoring mechanism. When you click the green flag, the score will be reset to zero. When a “hit” event occurs, the score will increase by 1.

---

**Note:** This does not involve the calculation of sprites and backdrops. We tend to put such codes in backdrops.

---

### **Countdown!**

Add a time mechanism. When the green flag is clicked, the timer is set to 30s, and then decreases by 1 every second until the event is reset to zero to end the game.

But if you hear the “Hit” broadcast, let the timer increase by one to increase the game time.

### **Challenge**

I believe that you will be smart enough to program and implement this game soon. Next, we will add some challenges to enrich our game content.

- Added several different magic fish.
- One will get double points
- One will reduce game time
- One will change the background

### **7.7.7 Binary Game**

#### **Description**

Do you understand binary numbers? binary is a number system widely used in computing technology, so understanding binary numbers is an important step for understanding the computer world.

Click on the green flag to start. A random number will appear in the middle of the screen, and you need to know the corresponding binary number. For example, 5, the corresponding Binary number should be 0101, press the bear's paw on the right to confirm. If it is correct, the score will be increased by 1.



Or click Binary Game, and then learn online tutorial on the Scratch official website.

### Required Components

- A RasPad 3
- Scratch 3 (either online or offline)

### You Will Learn

- Understand binary numbers.
- Convert binary numbers to decimal numbers.

### Lesson Guide

#### Arrange a small binary abacus at the bottom of the screen.


First, we delete the previous sprite. Upload the **Zero.png** file and **Enter.png** file in the sprite list, you have downloaded them from raspad3.git before.

Then we upload the **One.png** file in the **Costumes** option of the **Zero** sprite, so that the **Zero** sprite has **Zero** costume and **One** costume.

Then we copy the **Zero** sprite with the right mouse button, and there will be 3 new **Zero** sprites in the stage area.

Set the default costume of the four sprites to Zero, record with the variables “num1”, “num2”, “num3”, and “num4” respectively, and initialize the positions of the four sprites.

The program in the figure below is the program of the Zero sprite. The other three Zero sprites modify the variable names and increase the X coordinate by 80 to make the effect arranged in order from left to right.

When we click the  sign in front of the variable, the variable will disappear from the stage.



**Change the shape by clicking and record the binary value.**

If you click the **Zero** sprite, it will switch its costume to the next costume, and switch the value of the variable num1, if num1 = 0, change num1 to 1.

In this way, the num1 variable can record the value on the blackboard in real time every time the Zero sprite switches costume.

Copy the program for the remaining three **Zero** sprites and modify the variable names.

**Conversion of binary numbers and decimal numbers.**

Before programming the Enter sprite, let's first understand the Binary number.

A binary number is a number expressed in the base-2 numeral system or binary numeral system, which uses only two symbols: typically "0" (zero) and "1" (one).

How to convert binary to decimal?

For binary number with n digits:

$d_{n-1} \dots d_3 d_2 d_1 d_0$
---------------------------------

The decimal number is equal to the sum of binary digits ( $d_n$ ) times their power of 2 ( $2^n$ ):

$$\text{decimal} = d_0 \times 2^0 + d_1 \times 2^1 + d_2 \times 2^2 + \dots$$

If binary is 1000, the corresponding decimal should be  $1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 = 8$

So, what is the decimal number of 1010?  $1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = 10$ .

### Programming the Backdrop

When the game starts, a random integer between 1-16 appears in the middle of our screen as a decimal number that needs to be converted by binary, and it is stored in the variable random.

Whether we receive a “right” result or a “wrong” result, we have to regenerate the random number and proceed to the next round of the game.

### **Programming the Enter Sprite**

After you understand the conversion of binary numbers and decimal numbers, let's program the Enter sprite.

First, we create a result variable, when the Enter sprite is clicked, We convert the binary number composed of the num1 variable, num2 variable, num3 variable and num4 variable into a decimal number, and save the result in the result variable.

Then use the result variable to compare the value of the random variable. If they are equal, we broadcast a "right" message, otherwise we broadcast a "wrong" message.

Now, click on the green flag above the stage area to start the game and explore the magical binary!

### **Challenge**

I believe that you will be smart enough to program and implement this game soon. Next, we will add some challenges to enrich our game content.

- Add a Zero sprite to convert random integers between 1-32.
- When you click on the Zero sprite, a “Pop” sound effect will be emitted.
- 1 point will be deducted for incorrect answers!

## **7.7.8 Flappy Parrot**

### **Description**

Today we made the parrot Devin avoid the trees for food.

Click on the green flag to start.

Or click Flappy Parrot, and then learn online tutorial on the Scratch official website.

### **Required Components**

- A RasPad 3
- Scratch 3 (either online or offline)



## **You Will Learn**

- Add more costumes to sprites and let them change their appearance
- Logical Operation
- Determine if the sprites are touching each other

## **Lesson Guide**

### **Arrange the parrot and background image**

First, we delete the default sprites and add Parrot sprites and Blue Sky backdrop. Change the size of the Parrot sprite to 20.

Upload the Trees sprite, select the Upload Sprite option, find the “Trees.png” file, and upload it.

Click Costumes, and then click the “Upload Costume” option to upload Trees2.png, Trees3.png, Trees4.png, and Trees5.png.

Copy out the second Trees.



### Let the Parrot sprite fly on the stage

Parrot stayed still, and the Trees and Trees2 sprites kept moving to the left, so it looked like the Parrot sprites were flying on the stage!

Set the initial position and movement of the Trees sprite so that it can reach the far left from the far right of the stage. Then, randomly changing costumes reappear on the far right side of the stage.

The code of the Trees2 sprite is almost the same as the Trees sprite, the only difference is that it hides for 1.5 seconds at the beginning.

Now, click on the green flag to see the effect of this flight... Very good, the Parrot sprite seems to be flying to the right side of the stage.

### Let the space button control Parrot to fly

When the game starts, Parrot will continue to drop down, that is, the y coordinate will continue to decrease. When we press the space button, the y coordinate of Parrot will increase.

### **Constantly increase the speed of Parrot falling through variables**

Parrot drop is a bit slow, we add flight variables to make Parrot drop more and more!

### **Change the costume so that Parrot has flight animation**

When we press the Space button, Parrot's clothing switches to Parrot-b.





### Calculate the score!

Create a new score variable. When the Trees and Trees2 sprites move to the right side of the Parrot sprite, it means that Parrot has crossed the Trees and Trees2. At this time, we will add score+1. In order to avoid always adding points, we need to add it after score+1 Wait 1 second for the code. (This code is required in both Trees and Trees2 sprites!)

## **Game failure mechanism**

When the Parrot sprite encounters a tree or falls to the ground during the flight, we judge the game is over, and then we stop all scripts.

## **Challenge**

I believe that you will be smart enough to program and implement this game soon. Next, we will add some challenges to enrich our game content.

- When the game fails, it switches to a cursing background screen.
- Add a countdown screen to the game start screen, let us prepare for the game.

## **7.7.9 Paint Box**

### **Description**

Have you ever had a Paint Box? You can use the colorful brushes in the Paint Box to paint a colorful world. Today we are also making an exclusive Paint Box on scratch. Let's take action.

Click on the green flag to start.

Or click Paint Box, and then learn online tutorial on the Scratch official website.

### Required Components

First, you need to prepare the following components:

- A RasPad 3
- Scratch 3 (either online or offline)

### You Will Learn

- Use the Pen extension function to draw on scratch.
- Let the sprite move with the mouse.
- Modify the size of the costume and the location of the cursor.
- Use the slider to modify the value of the variable.

### Lesson Guide

#### Lay out the drawing board, add Pen expansion function

First, we delete the previous sprite, and then add the Pencil sprite and Block-X sprite to Scratch.

Upload **Eraser.svg**, **Red.png**, **Blue.png**, **Green.png**, **Yellow.png** and other files in the Sprite list (You have downloaded from raspad3.git before.), and finally set the size of the Pencil sprite and Block-X sprite Change to 50.

Then, we add the **Pen** extension function.

### Add drawing function to Pencil sprite

Click on the Pencil sprite. First, we need to make the Pencil sprite move with the mouse. If the mouse button is pressed, we call the **pen down** block in the Pen extension function, which can draw along the track of the cursor. Otherwise, call the **pen up** block to stop drawing.

Click on the green flag and try to control the Pencil sprite with the mouse to draw a smiley face. You will find that the image is not drawn from the tip of the Pencil.

---

**Note:** Since the cursor is not on the Pencil tip, the line drawn is drawn from the middle of the Pencil sprite.

---

We click on the **Costumes**, select the entire Pencil sprite and move it until the Pencil tip is aligned with the cross cursor.

### Let the Pencil sprite change the painting color.

How to achieve that when we click on the color option below the stage, the Pencil sprite will switch the corresponding color?

First we write code for the Red sprite, when it is clicked, we broadcast a new “red” message.



Then we return the code of the Pencil sprite, and when it receives the “red” message, we change the color of the Pencil to red.

In the same way, we also write code for the Blue sprite, and when it is clicked, we broadcast a new “blue” message.

Return the code of the Pencil sprite. When it receives the “blue” message, we change the color of the Pencil to blue. Before painting, we clear the scene on the stage and set the Pencil color to red. Zoom in on the stage area and try your brush again.

### Add the function of clear drawing image.

Sometimes accidents happen when we draw, so we need to add a clear function. First select the Block-X sprite, when it is clicked, we clear all the screens.

Next, we set a new costume for the Pencil sprite, click the “Costumes” tab, and then select the **Upload Costume** option from the icon on the bottom left to upload the **Eraser.svg** file.

Modify the size of the Eraser garment appropriately, and then place it on the upper right of the cursor.

We started programming the Eraser sprite, and when it was clicked, we broadcast a new “eraser” message.

Then continue to add code for the Pencil sprite. When it receives the “eraser” message, it will change to the Eraser costume,

so that when we click on the Eraser sprite, we can change the Pencil sprite’s costume to look like an Eraser sprite.

Then change the color to white that is consistent with the background color of the stage, so as to achieve the function of partial removal.

**Toggle clothing and brush width tools.**

When the costume of the Pencil sprite is switched to Eraser, we need to switch the costume of the Pencil sprite to the Pencil-a costume by clicking the red and blue on the drawing board, so when we receive the “red” message and the “blue” message, We want to switch back to the original Pencil-a costume.

In order to make our painting function more fun, we need to be able to change the width of the Pencil at any time.

Create a new variable named “width”, then right-click the “width” variable on the stage, select slider, and then you will see a slider on the stage that can change the value of the “width” variable.

Use the value of the variable “width” as the size of the Pencil.



### Limit the painting range of the Pencil.

Only when the Pencil enters the drawing area, can painting be allowed.

When the pen enters the toolbar, that is, when the y coordinate is less than or equal to -120, we need to call the **pen up** function to stop painting.





### Challenge

I believe that you will be smart enough to program and implement this game soon. Next, we will add some challenges to enrich our game content.

- Add more color brushes to make a Paint Box that belongs to you.

### 7.7.10 Eating Banana Game

#### Description

Scratch has a Video Sensing expansion module, which can turn on the camera in Scratch and detect the movement of objects on the camera screen.

Today, we will use the camera to make a eating banana game. In the stipulated time, help the Monkey eat more bananas.

To play the game against a white background, click on the green flag to start. Move colored objects in front of the camera to control the Monkey sprite.

Or click Eating Banana Game, and then learn online tutorial on the Scratch official website.

#### Required Components

- A Raspad
- Scratch 3 (either online or offline)

#### You Will Learn

- Install camera on RasPad.
- Add Video Sensing expansion module and use camera function
- Let the camera detect our movements

#### Lesson Guide

##### Install and use the camera.

For the camera installation tutorial, please refer to *Assemble the Camera Module*.

Click the “Add Extension” icon at the bottom left of scratch and select the **Video Sensing** extension.

At this point, you can see the camera shot on the Scratch stage.

### Arrange monkeys and bananas

First, we delete the original sprite, then add Monkey sprite and Bananas sprite, and change their sizes to 50.

Let Bananas appear randomly.

Bananas disappears after encountering the Monkey, which means it was eaten by the Monkey and reappears randomly.

Let the Monkey appear in the center of the stage and initialize the camera data (transparency is set to 20).

If the camera detects an object moving, let the Monkey move towards the object.

Now, click on the green flag at the top of the stage area to start the game.

Let the Monkey eat bananas, it is very hungry! Try to play this game on a white background to prevent interference from other objects.

### Challenge

I believe that you will be smart enough to program and implement this game soon. Next, we will add some challenges to enrich our game content.

- When Monkey eats banana, we add 1 to the score. Within 30s, see who has the highest score!
- When Monkey eats a banana, it emits a suitable sound effect.

### 7.7.11 Create Musical Instrument

#### Description

You can turn RasPad 3 into a game screen playing with your friends, a smart alarm showing the weather and time, a display monitoring your robot's action and many other things.

This article will show you how to DIY a Musical Instrument with your RasPad 3. Let's take a look!

### **Required Components**

- A RasPad 3
- 8G+ SD Card
- Scratch 3 (either online or offline)
- Micro SD Card Reader
- 40 Pin Ribbon Cable
- T-Type GPIO Extension Board
- Breadboard
- Button

- 10k Ohm Resistor
- Several Jumper Wires

### You Will Learn

- Use note functions on Scratch.
- Input the button value from the GPIO pins.

### Lesson Guide

#### Build the Circuit

First connect the GPIO Extension Board, please read *GPIO Extension Board* for specific steps.

Insert the T-type GPIO extension board into the breadboard and build the circuit as shown below.



### Programming with Scratch 3

Insert the SD card into the slot of RasPad 3 and then long press the power button to boot the RasPad 3.

**Warning:** The card slot is set on the back cover and it's in snap style. Therefore before opening or closing the back cover, you need to take out the microSD card to avoid damaging your microSD card and the internal button board.

On the left of Scratch 3 main page are some neatly arranged blocks with which you can drag to program. In this project, we need to add in other two functions: Music and Raspberry Pi GPIO. The Music function to play instruments and

drums, and the Raspberry Pi GPIO function can be used to control the whole pins of Raspberry Pi.

Tap Add icon at lower left corner and choose Music and Raspberry Pi GPIO to add two functions on the left of the main page of the Scratch 3.

Finished the program.

Press these three buttons on the breadboard and there emits the tones “Do, Re, Mi”.

If you hate reading, see this video.

In this article, we introduce how to use RasPad 3 to create musical instrument. Surely, you can exert all your imagination and inspiration to upgrade your musical instrument, for example you can add more buttons, notes and an LED to make a cool music box.

### **7.7.12 DIY RasPad Monitor Device**

#### **Description**

You can turn RasPad3 into a game screen playing with your friends, a smart alarm showing the weather and time, a display monitoring your robot's action and many other things.

This article will show you how to DIY a RasPad Monitor Device on your RasPad 3. Let's take a look!

### **Required Components**

- A RasPad 3
- 8G+ SD Card
- Scratch 3 (either online or offline)
- Micro SD Card Reader
- 40P Ribbon Cable
- T-Type GPIO Extension Board
- Breadboard
- PIR Module
- Camera Module
- FFC Cable
- Jump Wire F/M

### You Will Learn

- Use Raspberry Pi extensions on Scratch.
- Use audio functions on Scratch.
- Use PIR module.

### Lesson Guide

#### Build the Circuit

First connect the GPIO Extension Board, please read *GPIO Extension Board* for specific steps.

Plug the T-type GPIO extension board into the breadboard and build the circuit.

For the camera installation tutorial, please refer to *Assemble the Camera Module*.

**Programming with Scratch 3**

In this step you will learn how to upload the prepared music to the Scratch. Tap the “Sounds” option on the left upper corner then tap the “speaker” icon and choose “Upload Sound” icon to upload the prepared music file - hello, finally tap “Open” to confirm.

Tap Add icon at lower left corner and choose “Video Sensing” and “Raspberry Pi GPIO” to add two functions.



Back to the main page, drag a “when gpio 0 is high” from Raspberry Pi GPIO function and a “play sound (hello) until done”to the coding area.

Stick the pir module and camera to the wall outside the door, And stick the RasPad to the wall inside the door or anywhere. When the door is opened, you will hear music and then see who is there.

If you hate reading, see this video.

In this article, we introduce how to use RasPad 3 to DIY a RasPad Monitor Device. Surely, you can also add a relay and a stepper motor to open the door when someone is detected at the door.

## 7.8 Install Other Systems on RasPad

In addition to the **Raspberry Pi OS**, and RetroPie Operating system that we used before, RasPad 3 also supports other operating systems.

Here we only provide a link for learning reference. You can try it again according to the operation steps above the link.

---

**Note:** We have only tested Raspberry Pi OS, Ubuntu MATE , RetroPie on RasPad. If you have other systems installed, but some of them may have some bugs that make some of the functions unusable, we can only suggest you to consult the developer of that system.

---

23 Operating Systems That Run on Your Raspberry Pi.

## Chapter 8

# APPENDIX

For new RasPad users, the following sections are essential operations that will need to be used for doing certain projects. Please read these sections carefully.

### 8.1 Assemble the Camera Module

Open the back cover of the RasPad. Connect the FFC cable to the CSI interface of the Camera Module. Carefully pass the CSI interface cable through the slot on the back cover of the RasPad. Connect the CSI interface of the Raspberry Pi. Then carefully close the back cover of the RasPad.

<p><b>Warning:</b> The SD card slot is a snap-in style attached to the back cover. Before removing or replacing the back cover, remove the micro-SD card to avoid damaging the micro-SD card and the RasPad's internal button board.</p>
--

Insert the micro-SD card with the Raspberry Pi OS image into the slot, and long-press the power button to boot the RasPad.

Open **Raspberry Pi Configuration**.

In the **Interfaces** option, **Enable** the Camera, and then click **OK**.

In the pop-up prompt box, choose to restart now.

After the restart is complete, use the following command line to check whether the camera is available.

If the camera screen appears it means that the camera is installed successfully. Otherwise the FFC cable needs to be unplugged and plugged in again.

```
raspivid -o vid.h264
```

## 8.2 GPIO Extension Board

Before starting the project, you first need to know more about the pins of the Raspberry Pi, which is key to the build circuit.

The pins of Raspberry Pi have three ways to name them. They are wiringPi, BCM and Board. Among these naming conventions, the 40-pin GPIO Extension board uses the naming convention BCM.

The following table shows the naming convention for WiringPi, Board and the intrinsic Name of each pin on GPIO Extension board.

For example, for the GPIO17, the Board naming method of it is 11, the wiringPi naming method is 0, and the intrinsic naming method of it is GPIO0.

Open the back cover of the RasPad and insert the 40 pin ribbon cable into the Raspberry Pi. Pass the ribbon cable through the available slot on the base of the RasPad, and connect the other end of the 40 pin ribbon cable to the GPIO extension board. Then replace the back cover of the RasPad.

<b>Warning:</b> The SD card slot is a snap-in style attached to the back cover. Before removing or replacing the back
---



cover, remove the micro-SD card to avoid damaging the micro-SD card and the RasPad's internal button board.

## 8.3 Obtain the IP Address of the Raspberry Pi

### 8.3.1 With a Screen

If you have a screen, it will be easy for you to get the IP address of Raspberry Pi.

Required Components	
Any Raspberry Pi	1 * Power Adapter
1 * Micro SD card	1 * Screen Power Adapter
1 * HDMI cable	1 * Screen
1 * Mouse	1 * Keyboard

1. Insert the micro-SD card with the Raspberry Pi OS image into the micro SD card slot on the underside of the Raspberry Pi.
2. Plug in the Mouse and Keyboard.
3. Connect the screen to the Raspberry Pi's HDMI port. Make sure the screen is plugged into a wall socket and turned on.

**Note:** For the Raspberry Pi 4 models, connect the screen to the HDMI0 port, nearest to the power-in socket.

4. Use the power adapter to power the Raspberry Pi. After a few seconds, the Raspberry Pi OS desktop will be displayed.
5. Hover the cursor over the WiFi icon, and the IP address of the Raspberry Pi will be displayed.

### **8.3.2 Without a Screen**

After the Raspberry Pi is connected to WIFI, we need to get the IP address of it. There are many ways to know the IP address, and two of them are listed as follows.

#### **1. Checking via the router**

Check the addresses assigned to Raspberry Pi on the administration interface of the router.

The default hostname of the Raspberry Pi OS is raspberrypi. If you are using an ArchLinuxARM system, please find alarmpi.

#### **2. Network Segment Scanning**

Network scanning applications can be used to look up the IP address of Raspberry Pi, such as Advanced IP Scanner.

Scan the IP range set, and the names of all connected devices will be displayed. The default hostname of the Raspberry Pi OS is raspberrypi.

## 8.4 Use the SSH Remote Control

Bash is the standard default shell of Linux, and can be opened on the Raspberry Pi by using SSH. The Shell itself is a program written in C programming language that acts as a bridge linking the customers and Unix/Linux systems.

### For Linux or/Mac OS X Users

Go to **Applications -> Utilities**, find the **Terminal**, and open it.

Type in `ssh pi@ip_address` where `pi` is the username and `ip_address` is the IP address of the Raspberry Pi. For example:

```
ssh pi@192.168.18.197
```

At the prompt, type yes in lowercase.

The default password is raspberry.

---

**Note:** The characters do not display when typing in the password. Make sure to input the correct password.

---

We now get the Raspberry Pi connected and are ready to go to the next step.

### **For Windows Users**

Windows users can use SSH with the PuTTY application.

Download PuTTY, open the application, and click Session on the left of the tree-like structure. Enter the IP address of the Raspberry Pi in the text box under Host Name, and enter 22 under Port.

Click Open.

---

**Note:** When first logging in to the Raspberry Pi with the IP address, there will be a security prompt. Click Yes to continue.

---

When the PuTTY window prompts **login as:**, type in `pi` for the user name of the Raspberry Pi. The default password is `raspberrypi`.

---

**Note:** The characters do not display when typing in the password. Make sure to input the correct password.

---

Here, we get the Raspberry Pi connected and it is time to conduct the next steps.

# Chapter 9

## FAQ

### 9.1 Fix Raspberry Pi boot problems

1. Check whether the power indicator is on, if it does not, you need to charge the RasPad or try a different power source. It is best to use the power supply provided by SunFounder.
2. You might be booting but not getting video output. First, check that your cables are connected properly. Take them all out and reconnect them. Try a different HDMI cable if you have one.
3. Take out the microSD card and use Raspberry Pi Imager to flash a new installation of Raspberry Pi OS to it (see *Install the Raspberry Pi OS*).
4. Try a different microSD card.

### 9.2 Can't detect Micro SD card?

1. Whether the FFC on the Micro SD card board is connected correctly, refer to *ASSEMBLY INSTRUCTIONS*.
2. Check whether the golden finger (transmission signal) on the SD card or the Micro SD card board turn black. If it is, it means that the gold finger has been oxidized and you need to erase the oxide layer with an eraser.
3. Check whether the SD card is damaged. If the plug of the SD card is unplugged while reading the SD card, causing the power supply to the card to be interrupted, the card may be damaged.

### 9.3 Screen can not be rotated?

The reason is that the built-in Accel SHIM module has not been calibrated.

Please refer to: *Install Rotating Screen Service*.

## 9.4 About sleep problem

- If your RasPad screen is on but black, this is the Raspberry Pi going into sleep mode. You can wake up by tapping the touch screen, or by pressing the external keyboard or mouse.
- If the RasPad's screen is not lit or you short press the power button, this is the RasPad entering sleep mode. You can wake up by pressing the power button shortly.

## 9.5 About the shutdown problem

- When you click shutdown button on the Raspberry Pi desktop or shutdown the Raspberry Pi at the command line, the Raspberry Pi will shut down, but the RasPad is not shut down, it will automatically go to sleep after a period of time because it does not detect the HDMI signal. You need to long press the RasPad's power button to shut down.
- If you directly press and hold the power button of RasPad, both Raspberry Pi and RasPad will be turned off at the same time. But it should be noted that this is to turn off the power of the Raspberry Pi directly, which may damage the Micro SD card. It is recommended to turn off the Raspberry Pi first, and then turn off RasPad.

## 9.6 Desktop is not full screen

If the desktop does not display in full screen after the Raspberry Pi boots, you need to leave **Underscan** enabled.

Click on the menu icon in the top left corner and select **Preferences -> Raspberry Pi Configuration**.



In the **Display** page, make Underscan enabled and then restart the Raspberry Pi.

## 9.7 About the touch problem

### A. Touch does not work

First of all, please check whether the internal cables connection are stable.

For the old version of RasPad 3, because the touch will become very strange after an external display, so turn off the touch function.

### B. Touch inaccurate

Because the Raspberry Pi system is based on the PC mouse control, the human hand can not be as accurate as the mouse, if the touch accuracy problems cause inconvenience, we recommend that you increase the size of the icon, or use the *RASPAD LAUNCHER* to install RasPad theme.

## 9.8 About the Audio problem

### A. There is no sound.

If the volume of either RasPad and Raspberry Pi is set to 0, it will result in no sound.

### B. headphone jack does not work properly

Since the 3.5mm headphone jack is inside the RasPad 3 case, please check if the headphone is fully inserted.

## 9.9 Fan is too noisy?

This is because the fan speed is too high, which was originally done to enhance the cooling capacity of RasPad 3. You can reduce the noise by setting the fan switch on the upper left corner of the mainboard to 's', but this will reduce the fan speed as well as the cooling capacity.

## 9.10 Power consumption is too fast?

RasPad's battery capacity is  $3 \times 3.7V \times 3Ah = 33Wh$ , which is a bit larger than iPad pro 2020's 28.65Wh. But because the Raspberry Pi itself is relatively large power consumption, standby about 3W, and the Raspberry Pi system is not similar to the power-saving mechanism of the tablet or even iOS. Compared to the previous generation, we have reduced the original 7w screen drive power consumption to 3W or less, and the use of time has been improved.

We will continue to optimize this use of time.

## 9.11 Touch function sometimes fails?

When opening some software of Raspberry Pi, you will find some places where touch does not work, this is caused by Raspberry Pi OS not doing adaptation for touch screen. We can't solve it and need to wait for the Raspberry Pi OS to optimize the solution.

## 9.12 The WiFi connection is unstable?

Because we found that USB 3.0 high-speed transmission will interfere with the Raspberry Pi's 2.4G wifi signal. So if you find the 2.4G signal is unstable, please try to use WIFI 5G AP.

## **Chapter 10**

# **Copyright Notice**

All contents including but not limited to texts, images, and code in this manual are owned by the SunFounder Company. You should only use it for personal study, investigation, enjoyment, or other non-commercial or nonprofit purposes, under the related regulations and copyrights laws, without infringing the legal rights of the author and relevant right holders. For any individual or organization that uses these for commercial profit without permission, the Company reserves the right to take legal action.