

校園發明大賽

INVENTION FOR SCHOOLS CONTEST

Basic Raspberry Pi 樹莓派入門

Dr. YIP Chi Lap [Beta], Faculty of Engineering, HKU
香港大學工程學院 葉志立博士

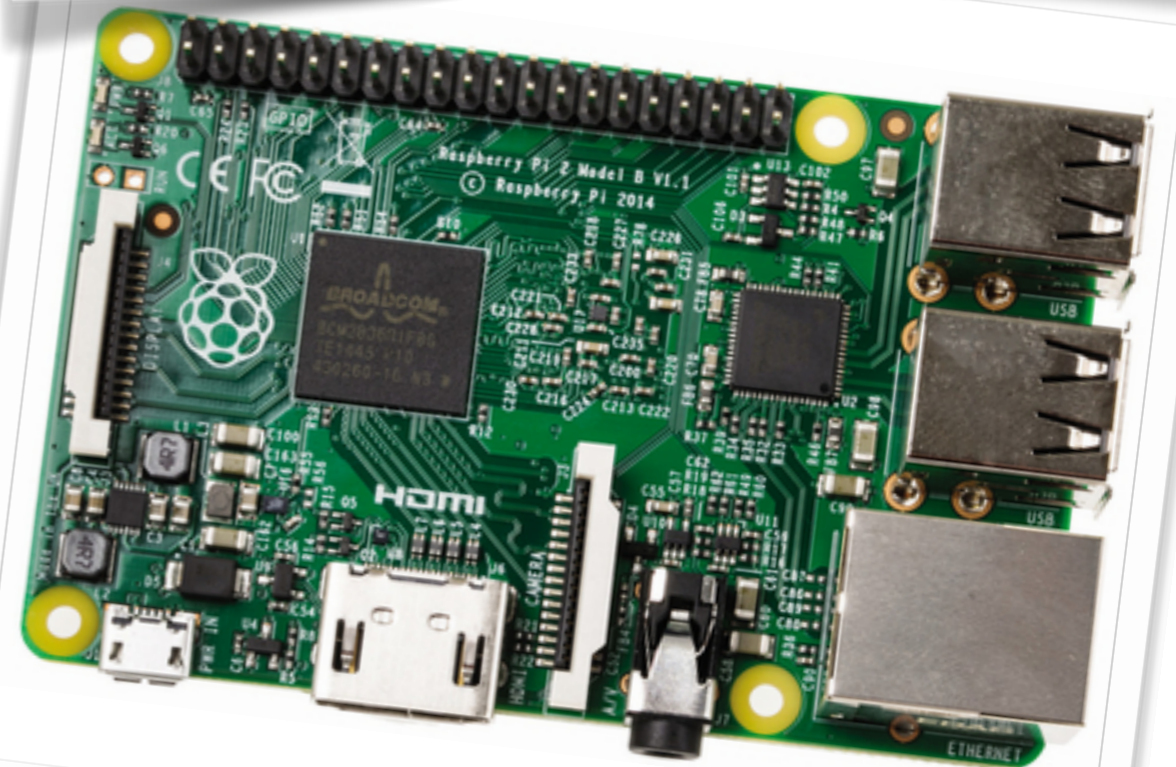
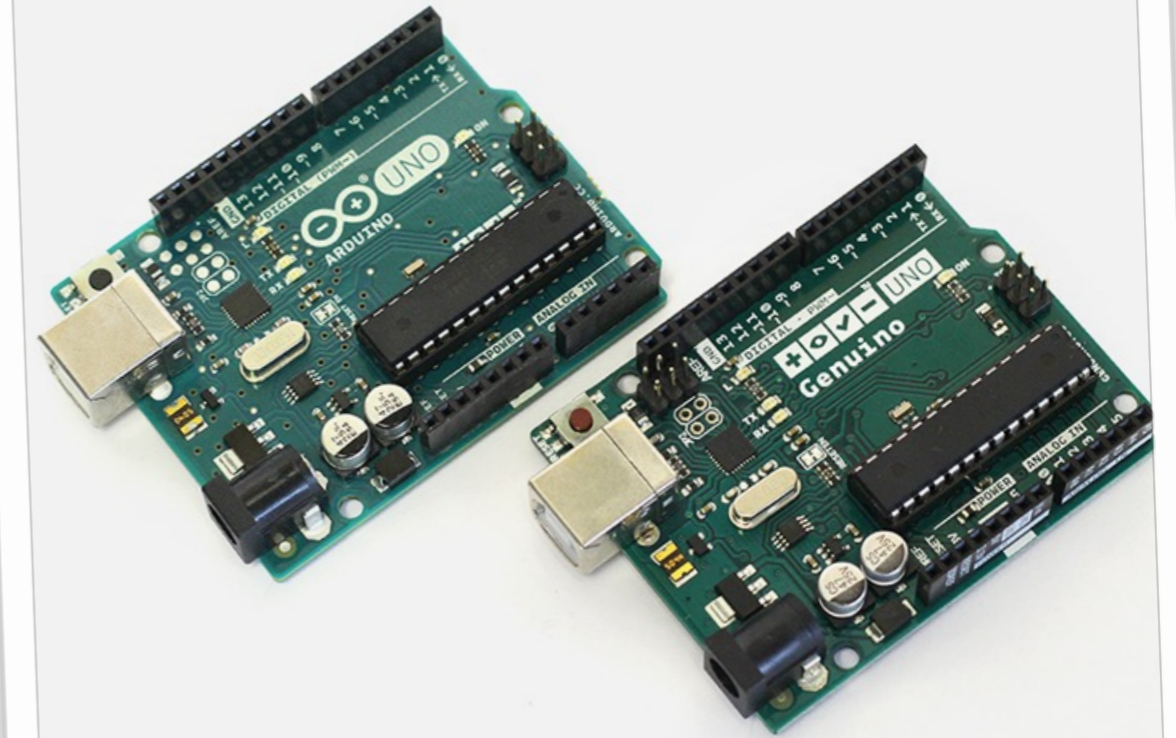
Topics to cover...

內容...

- Credit-card sized computers 卡片式電腦
- Raspberry Pi: overview 樹莓派概探
- Setting up a Raspberry Pi 設置樹莓派
- Getting ready for Raspberry Pi: one-off tasks
準備迎接樹莓派：一次性任務
- Let's program it! 編個程式吧！
- Interfacing with the real world 與世界接軌

Credit-card sized
computers

卡片式電腦



Why credit card-sized computers? 為什麼使用卡片式電腦?

- Low cost 便宜
- Small 小巧
- Portable 易攜
- Programmable 可編寫程式
- Can be used as a desktop computer
可作桌面式電腦使用
- Can be connected to other electronic components to make different devices
可連接到其他電子零件去製作出不同的裝置

Raspberry Pi: overview

樹莓派概探

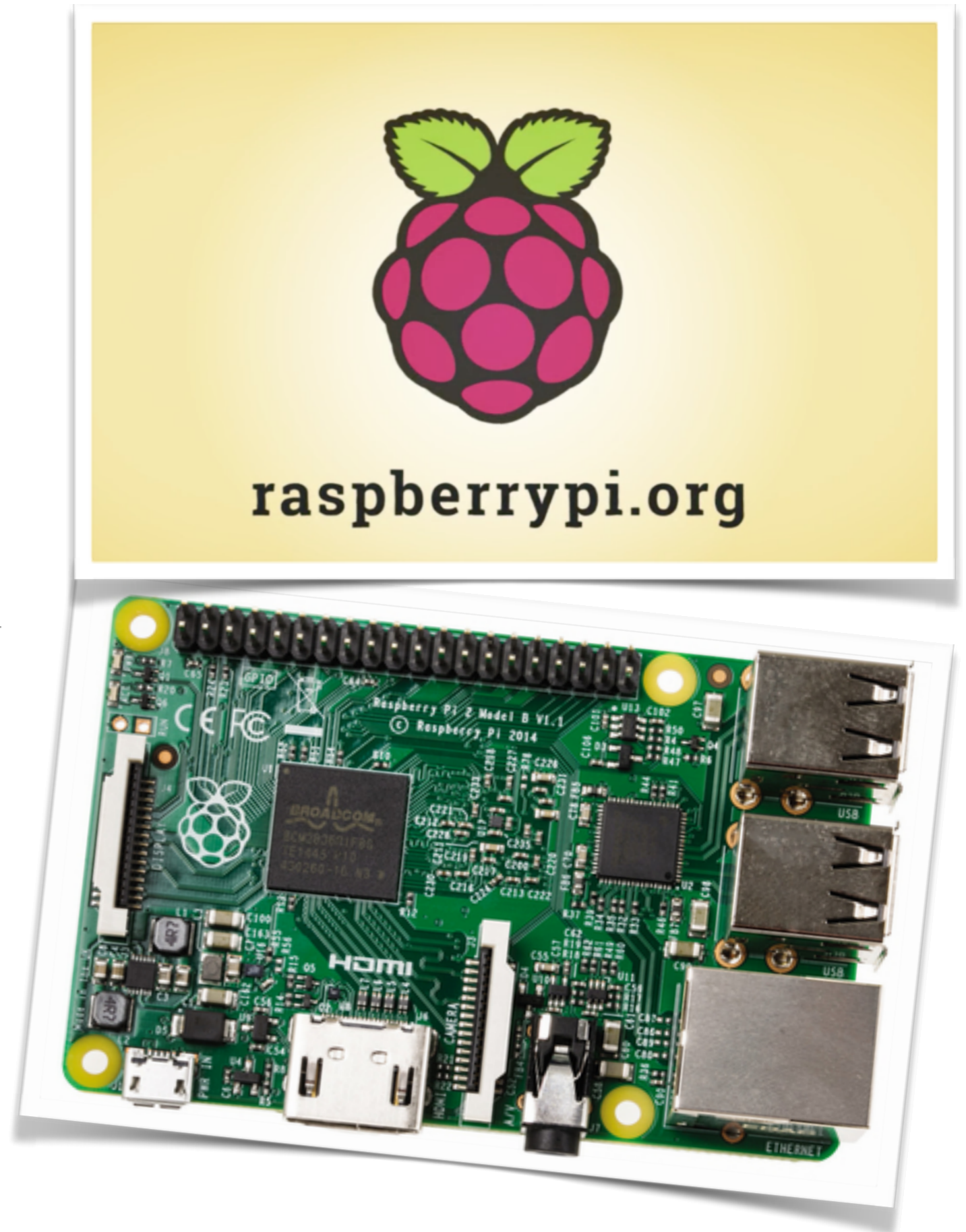
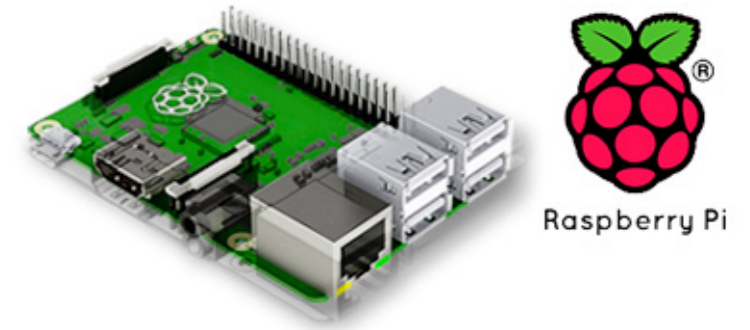


Image source:

- <https://www.raspberrypi.org/>

What is Raspberry Pi?

什麼是樹莓派？



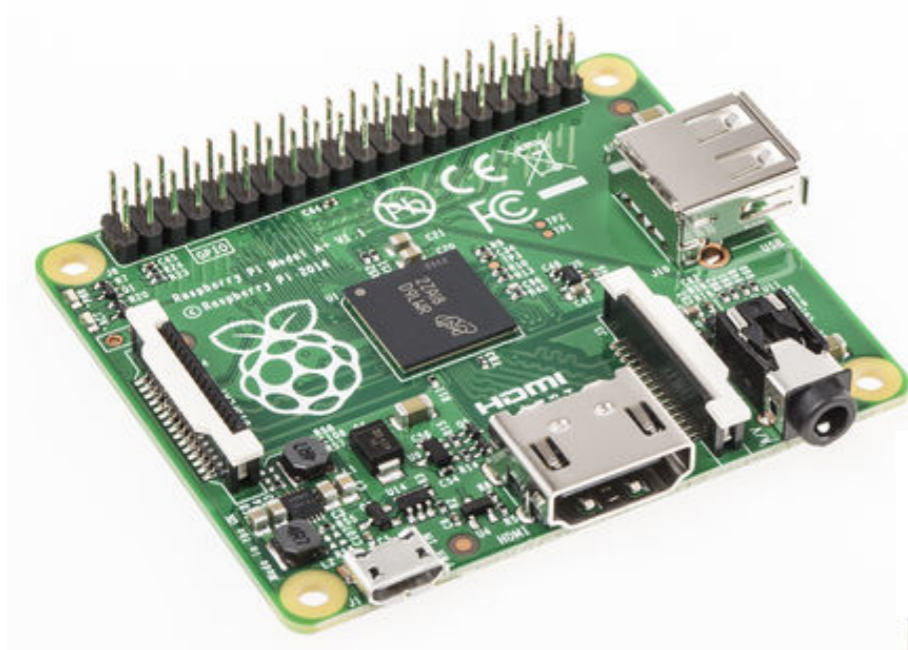
- Low cost, credit card-sized computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse.
便宜的卡片式電腦，可接駁顯示屏、鍵盤及滑鼠使用。
- Capable of doing everything you'd expect a desktop computer to do, from browsing the Internet and playing high-definition video, to making spreadsheets, word-processing, and playing games.
可作桌面式電腦使用，能用以瀏覽網頁、播放影片、製作試算表、文字報告或玩電腦遊戲等。
- Can be connected to other electronic components to make different devices.
可連接到其他電子零件去製作出不同的裝置。
- Can be used to learn how to program in languages like Scratch and Python.
可用以學習程式編寫如 Scratch 及 Python。

Image source:

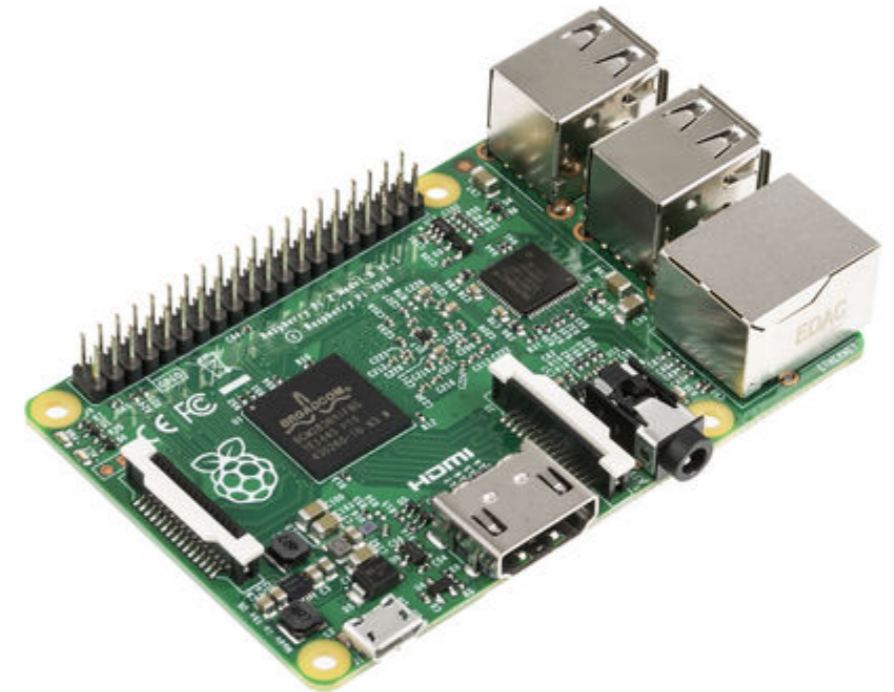
- <http://hken.rs-online.com/>

Raspberry Pi models

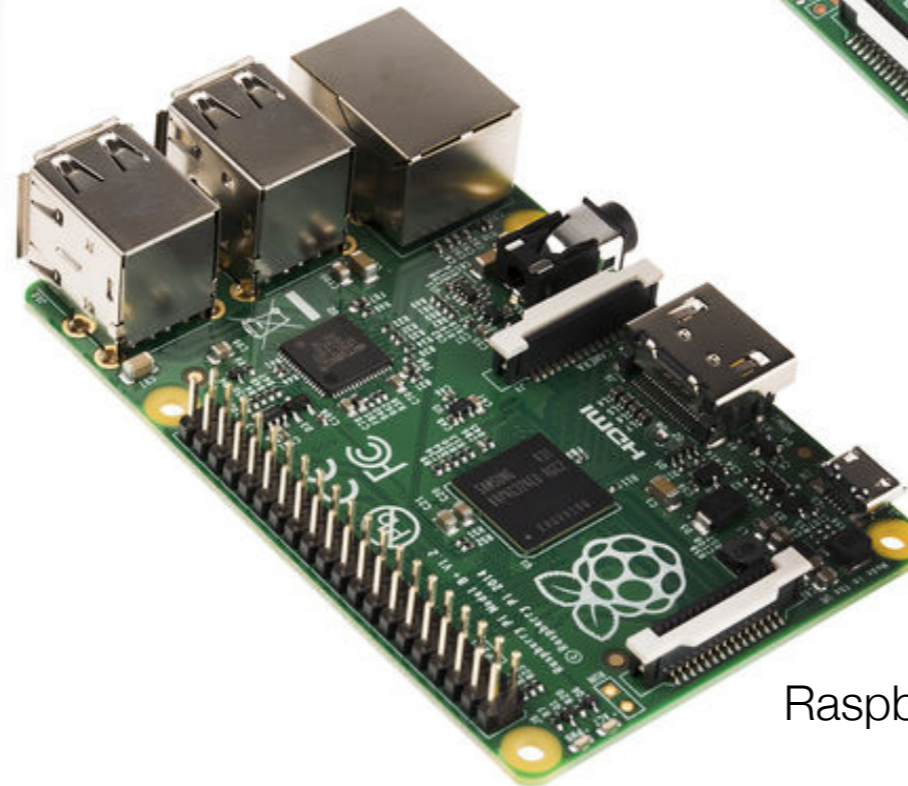
樹莓派類型



Raspberry Pi A+



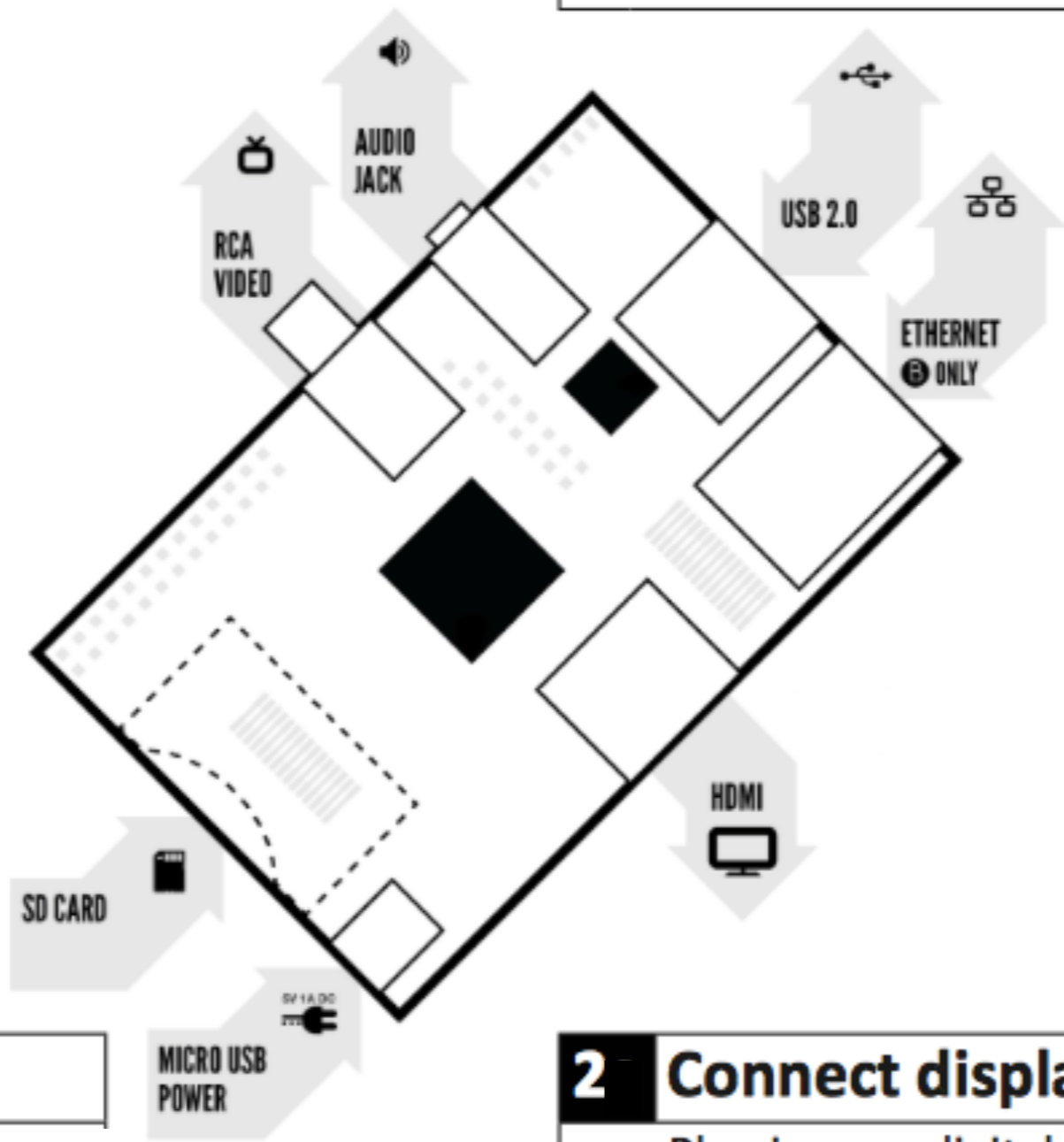
Raspberry Pi 2 B



Raspberry Pi B+

Comparison 比較 :

<https://www.raspberrypi.org/documentation/hardware/raspberrypi/models/specs.md>



1 Insert SD card

插入 SD card

3 Connect input
Plug in a USB keyboard and mouse

接駁輸入裝置

接駁 USB 鍵盤及滑鼠

4 Connect network
Connect to your wired network [optional]

接駁網絡

5 Power up

接駁電源

2 Connect display
Plug in your digital TV or monitor

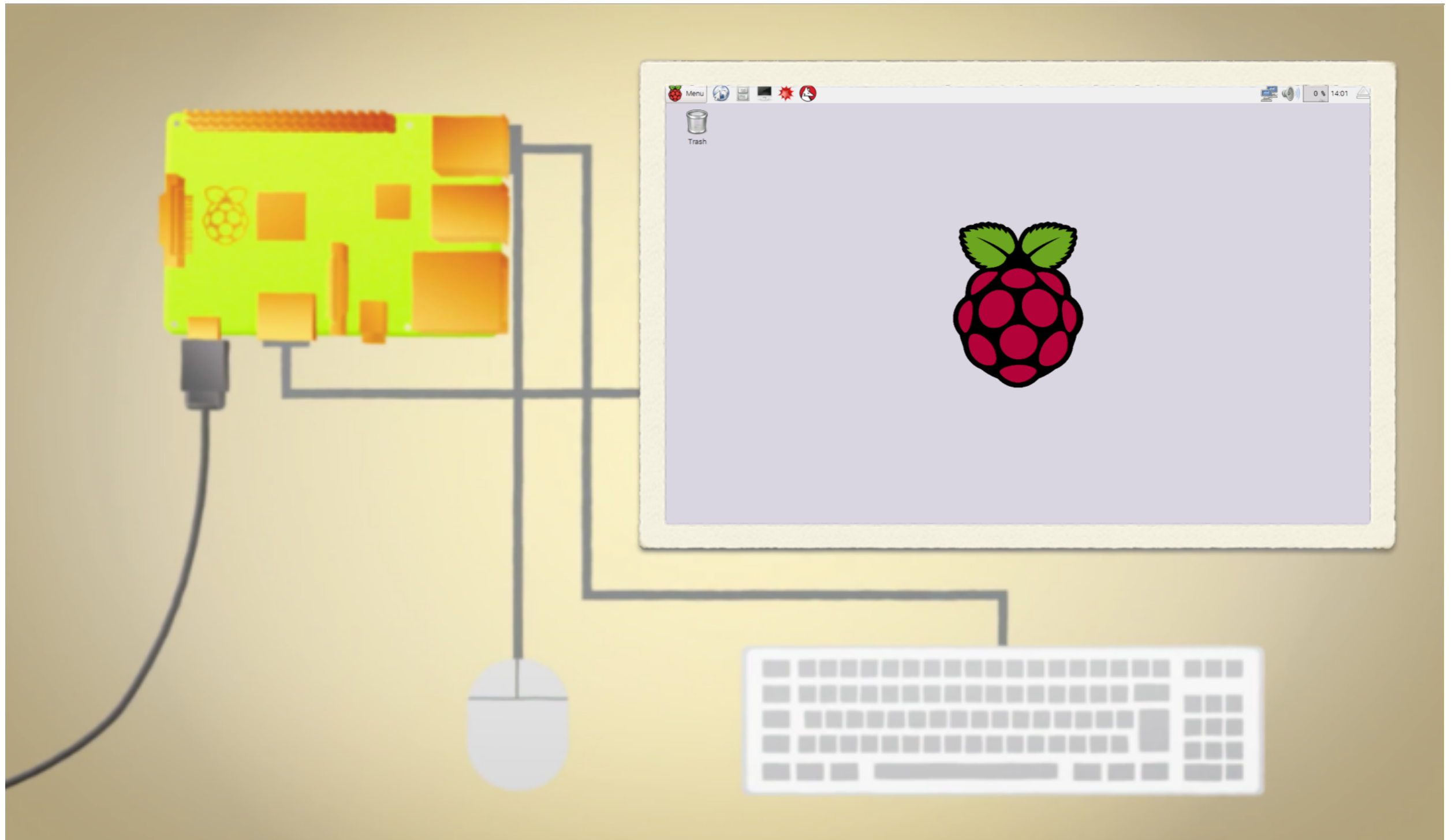
接駁顯示屏

以HDMI或模擬輸入接駁顯示屏

Image source:

- <https://www.raspberrypi.org/wp-content/uploads/2012/12/quick-start-guide-v1.1.pdf>

Connect to a display, keyboard, mouse and power
接駁顯示屏、鍵盤、滑鼠，和電源



Setting up a Raspberry Pi

設置樹莓派

Setting up a Raspberry Pi

設置樹莓派

- What you need 你需要的:
 - SD card.
 - Monitor or TV. 顯示器或電視。
 - HDMI cable. HDMI 連接線。
 - USB keyboard, mouse. USB 鍵盤與滑鼠。
 - Power supply. 電源供應器。

Main reference:

- Setting up your Raspberry Pi. Quick Start Guide. <https://www.raspberrypi.org/help/quick-start-guide/>

SD card: capacity, size, and class

SD card: 容量、尺寸，和速度等級

- Capacity: at least 8GB recommended.
容量：建議最少8GB.
- Size of SD card (Micro SD or Full SD) depends on Raspberry Pi model.
不同的樹莓派用不同尺寸的 SD card.









Model 類型	Model A	Model B	Model A+	Model B+
Size 尺寸	Full SD	Full SD	Micro SD	Micro SD

- The card class determines the sustained write speed for the card; a class 4 card will be able to write at 4MB/s, whereas a class 10 should be able to attain 10 MB/s.
速度等級決定了卡的持續寫入速度。Class 4 的卡每秒可寫入4MB的資料，Class 10則每秒10MB。

Table data source:

- <https://www.raspberrypi.org/documentation/hardware/raspberrypi/models/specs.md>

SD card

	Full SD	miniSD	microSD	SD Card Capacity SD卡容量
SD				up to 上限 2GB
SDHC 高容量				High Capacity 4GB — 32GB
SDXC 超容量				Extended Capacity 32GB — 2TB

- There are products with a Micro SD card and a Micro SD-to-Full SD adaptor in a package.
有些產品具有 Micro SD 卡及 Micro SD-to-Full SD 的適配器。

Table and image sources:

- <https://www.sdcard.org/consumers/choices/index.html>
- <https://www.sdcard.org/cht/consumers/choices/index.html>

Monitor/TV and HDMI Cable

顯示器或電視，與及 HDMI 連接線

- If your monitor or TV has HDMI input:
如你的顯示器或電視有 HDMI 輸入：
 - You just need a HDMI (male–male) cable.
你只需要一條HDMI 公對公連接線。
- Your monitor or TV with DVI input:
如你的顯示器或電視有 DVI 輸入：
 - Use a HDMI to DVI (specifically, DVI-D) male–male cable.
你需要用一條HDMI 至 DVI-D 公對公連接線。
- Sound goes to Monitor/TV directly in both cases.
在兩種情況下，聲音會直接從顯示器或電視輸出。

Monitor/TV and HDMI Cable

顯示器或電視，與及 HDMI 連接線



HDMI male–male cable
HDMI 公對公連接線



HDMI to DVI-D male–male cable
HDMI 至 DVI-D 公對公連接線

Image source:

- <http://hken.rs-online.com/web/p/digital-video-monitor-cable-assemblies/6162362/>
- <http://hken.rs-online.com/web/p/digital-video-monitor-cable-assemblies/7424343/>

Monitor/TV and HDMI Cable

顯示器或電視，與及 HDMI 連接線

- Your monitor or TV only supports VGA input.
如你的顯示器或電視只有 VGA 輸入：
 - Use HDMI (male) to VGA (female) adaptor and a normal VGA (male-female) cable to connect to computer.
你需要用 HDMI (公)至 VGA (母)適配器及VGA 公對母連接線去連接電腦。
- Some models support sound output, some not.
只有部分型號支援聲音輸出。



Image source:

- <http://www.apple.com/shop/product/H9139VC/B/kanex-atv-pro-x-hdmi-to-vga-adapter-with-audio-support>
- <http://hken.rs-online.com/web/p/digital-video-monitor-cable-assemblies/7781882/>

Keyboard, mouse

鍵盤，滑鼠

- Any USB keyboard and mouse would do.
任何USB鍵盤和滑鼠。
- Those with builtin USB hub (e.g., Apple keyboard) are also supported.
亦支援配有 USB 集線器（如：Apple keyboard）的USB 鍵盤和滑鼠。



Image source:

- <http://www.logitech.com/zh-hk/product/desktop-mk120?crd=27>

Power supply 電源

- Use a 5V micro USB power supply.
使用5V micro USB 電源供電。
- Power bank for mobile phone.
可携式電源。
- Computer USB port.
電腦的 USB 端口。
- USB power supply.
USB 電源供應器。
- A male type A USB to male micro USB cable may be needed.
可能需要配備Type-A USB 至 micro USB 公對公連接線



Image sources:

- <http://hken.rs-online.com/web/p/plug-in-power-supply/8226373/>
- <http://hken.rs-online.com/web/p/power-banks/7757508/>
- <https://www.facebook.com/orico.hk/photos/a.644756615659662.1073742025.147921822009813/644756678992989/>

Getting ready for Raspberry Pi: one-off tasks

準備迎接樹莓派：一次性任務

Download files

下載檔案

- Download the NOOBS (New Out Of the Box) zip file from this web site:
從這網站下載 NOOBS (New Out Of the Box) 的 zip 檔:

<https://www.raspberrypi.org/downloads/>

- Download SD Formatter 4.0 software from
https://www.sdcard.org/downloads/formatter_4/
and install it.

從這網站下載然後安裝SD Formatter 4.0格式化軟體:

https://www.sdcard.org/cht/downloads/formatter_4/index.html

Image sources:

- <https://www.raspberrypi.org/help/noobs-setup/>

Use SD Formatter to format the SD card

利用 SD Formatter 4.0 將 SD card 格式化

- Insert your SD card into a card reader connected to your computer.
將你的 SD card 插入連接到您的電腦的讀卡器。
- Run the software. 執行軟件。
- Choose 選取 "Overwrite Format".
- Click the "Option" button, choose "Yes" in the menu that pops up, then click "OK".
點擊 "Option"，從彈出的菜單中選取 "Yes"，然後點擊 "OK"。
- Specify the name of the SD card, e.g., "RPI".
給SD card 一個名字，如 "RPI"。
- Click "Format" and wait for completion.
點擊 "Format"，等格式化完成。

References:

- <https://www.raspberrypi.org/help/noobs-setup/>
- <https://www.raspberrypi.org/documentation/installation/noobs.md>

Install NOOBS onto the SD card

在 SD card 上安裝 NOOBS

- Expand the zip file NOOBS_v1_4_2.zip
解壓 zip 檔 NOOBS_v1_4_2.zip
- Copy the expanded files and folders to the newly formatted SD card.
將已解壓的文件和文件夾複製到剛格式化的 SD card.
- Eject the card from your computer.
將 SD card 從電腦中退出。

Installing Raspbian on your Raspberry Pi

安裝 Raspbian 至樹莓派

- Put the SD card with NOOBS into the card slot of your Raspberry Pi.
將有 NOOBS 的 SD card 放入樹莓派的卡槽內。
- Connect the video cable, keyboard and mouse to your Raspberry Pi.
接駁顯示屏、鍵盤，和滑鼠至樹莓派。
- Power up the Raspberry Pi by connecting its micro USB connection to a power source (USB power supply, a power bank, or a computer).
利用 micro USB 向樹莓派供應電源。電源可從 USB 電源供應器、可攜式電源、或電腦獲取。
- Choose the keyboard type to be US keyboard if necessary.
如需要，選擇鍵盤類型為美式鍵盤。
- Choose to install Raspbian. Confirm and wait.
選擇安裝 Raspbian，確定，然後等待安裝完成。

Ready to go!

可用了！

Usable now!

即可使用！

- The Raspberry Pi can be used after Raspbian is installed without configuration.
裝了Raspbian後，不必配置，樹莓派已可使用。
- Login name 登錄名：pi
Password 密碼：raspberry
- A program can be used to do system-wide configuration when required.
需要時，可利用一個程式做系統配置。

raspi-config

- Run the terminal program by clicking the icon.

按此圖標啟動終端機程式。



- Type in the terminal the command line, and press Enter.
在終端機程式內輸入此命令，然後按Enter。

```
sudo raspi-config
```

- By using up, down, left and right arrow keys and the Enter key, you can configure various aspects of the system there.
利用上下左右和Enter鍵，你可以利用該程式作各樣配置。

Main reference:

- <https://www.raspberrypi.org/documentation/configuration/raspi-config.md>

raspi-config

- Example: force use of audio jack for audio output.
例：強制使用音頻插孔輸出聲音。

The image displays three sequential screenshots of the Raspberry Pi Software Configuration Tool (raspi-config) interface, illustrating the steps to force audio output through the 3.5mm jack.

Screenshot 1: Main Menu
Raspberry Pi Software Configuration Tool (raspi-config)
1 Expand Filesystem
2 Change User Password
3 Boot Options
4 Wait for Network at Boot
5 Internationalisation Options
6 Enable Camera
7 Add to Rastrack
8 Overclock
9 Advanced Options
0 About raspi-config

Screenshot 2: Advanced Options
Raspberry Pi Software Configuration Tool (raspi-config)
A1 Overscan
A2 Hostname
A3 Memory Split
A4 SSH
A5 Device Tree
A6 SPI
A7 I2C
A8 Serial
A9 Audio
A0 Update
<Select> <Back>

Screenshot 3: Audio Output Selection
Choose the audio output
0 Auto
1 Force 3.5mm ('headphone') jack
2 Force HDMI
<Ok> <Cancel>

Main reference:

- <https://www.raspberrypi.org/documentation/configuration/raspi-config.md>
- <https://www.raspberrypi.org/documentation/configuration/audio-config.md>

Let's program it!

編個程式吧！

Programming the Raspberry Pi

利用樹莓派編寫程式

- Visual programming 可視化編程: Scratch
 - Easy to learn, not all features included.
容易學習，不包所有功能。
- Writing code 寫程式碼: Python
 - A powerful programming language used in many platforms. Used in programs with a few lines to large scale systems.
多種平台可用的強大的編程語言，由幾行的程式至大型項目均會使用。
- Making music 玩音樂: Sonic Pi
 - An open-source programming environment, designed for creating new sounds with code in a live coding environment.
一個開放源碼的編程環境，能實時用代碼創建新的聲音。

Main references:

- <https://www.raspberrypi.org/documentation/usage/>

Programming the Raspberry Pi

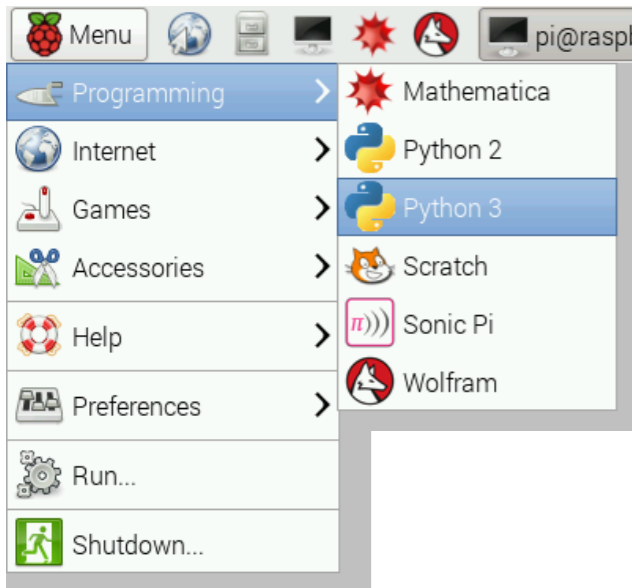
- You can also use C, C++, Java, HTML, etc.
你亦可使用 C, C++, Java, HTML 等。
- Start at the submenu Menu > Programming.
環境在子菜單 Menu > Programming 內。

Main references:

- <https://www.raspberrypi.org/documentation/usage/>

Programming with Raspberry Pi

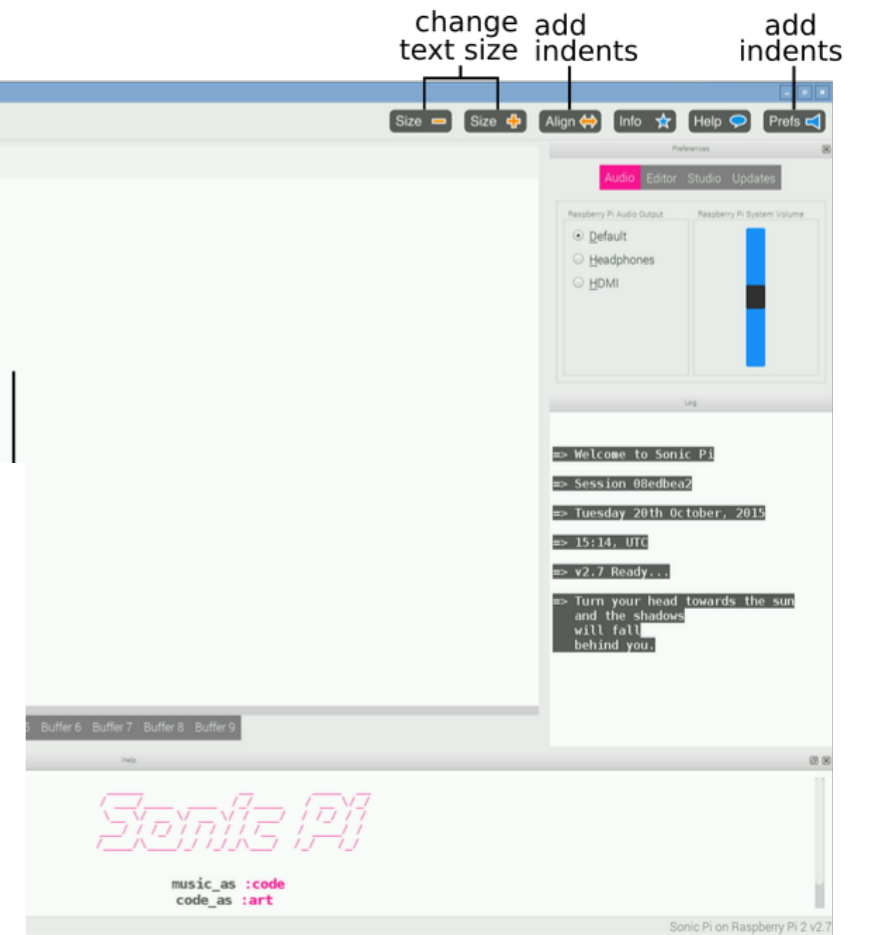
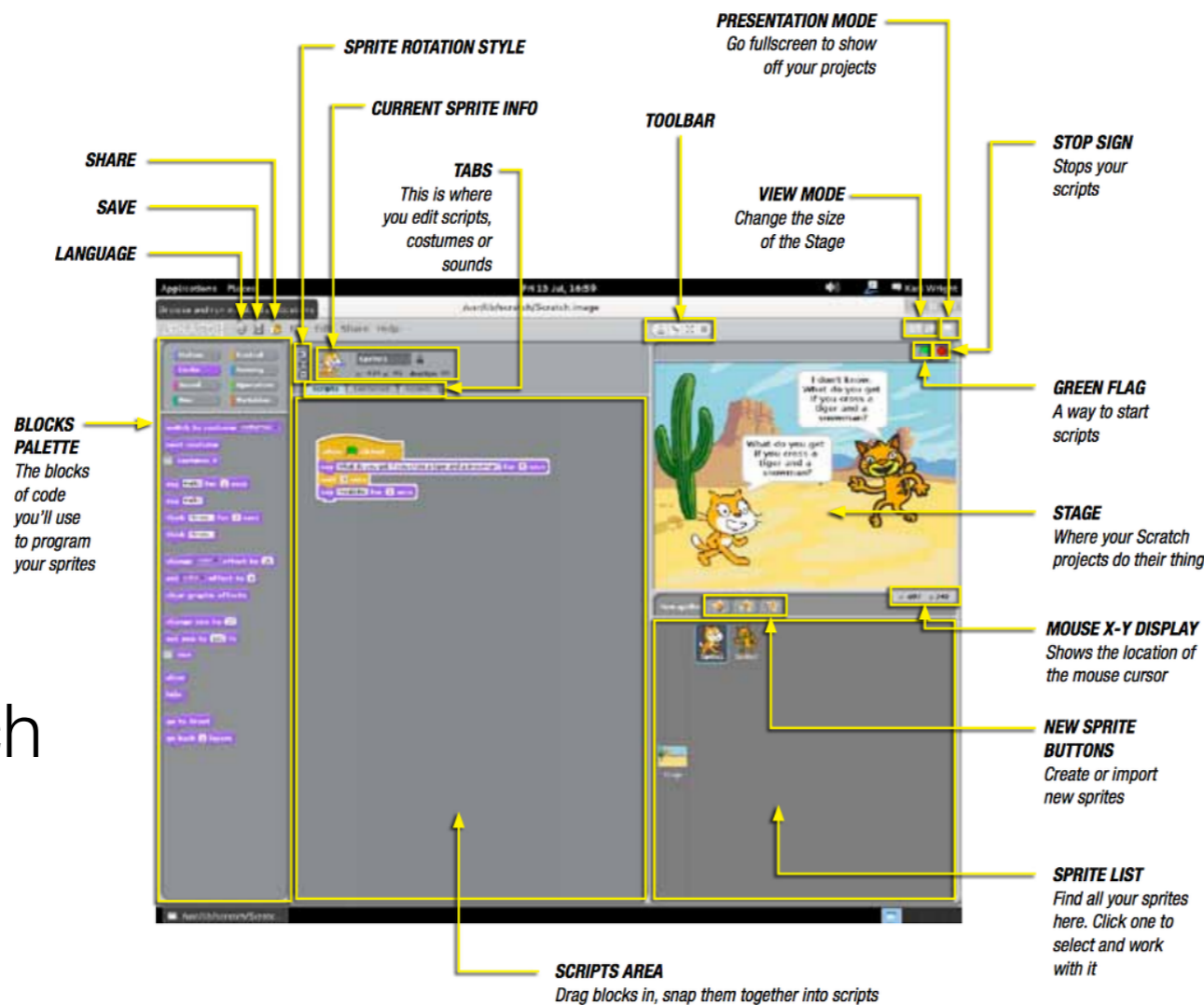
利用樹莓派編寫程式



Python



Scratch



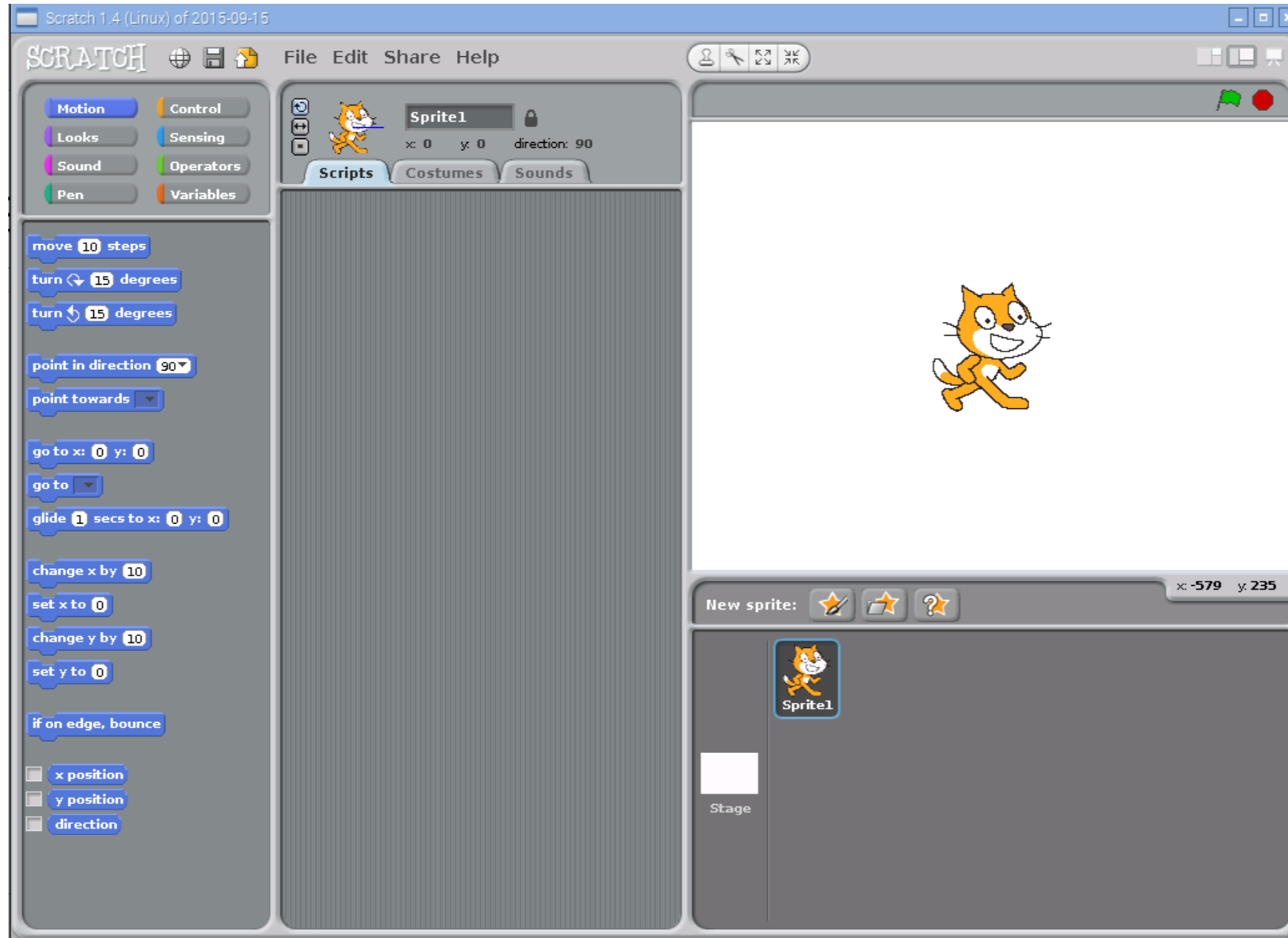
Sonic Pi

Scratch



- A visual programming tool which allows the user to create animations and games with a drag-and-drop interface.
用拖放界面的可視化編程工具，可寫動畫和遊戲。
- Included in Raspbian. Raspbian 內已有。
- An MIT project 由麻省理工開發：<https://scratch.mit.edu/>
- Raspberry Pi page 在樹莓派的網頁：
<https://www.raspberrypi.org/documentation/usage/scratch/>

Scratch



Python



- A powerful programming language that's easy to read and write.
易讀易寫的強大的編程語言。
- Language is used in many platforms and projects of various sizes. 能在不同平台上建立不同大小的程式。
- Some success stories 成功例子：
<https://www.python.org/about/success/>
- Web site 網站：<https://www.python.org/>

Python



```
print("Hello Python")

print(1+2*3/4)

for i in range(4):
    print('Hello Python')

for i in range(10,20,5):
    print('Number', i)

n = 0
for i in range(1, 101):
    n += i
print("The sum of the numbers 1 to 100 is:",n)

for i in range(3):
    print(i," ",end='')

age = 15
print(age)
age += 1
print(age)
```

```
people = [ "Peter", "Paul", "Mary" ]
for person in people:
    print(len(person), person)

for char in "Hello":
    print(char)

for person in people:
    if person[0]=='P':
        print("The word",person, "starts with a P")
    else:
        print("The word",person, "does not start with a P")
```

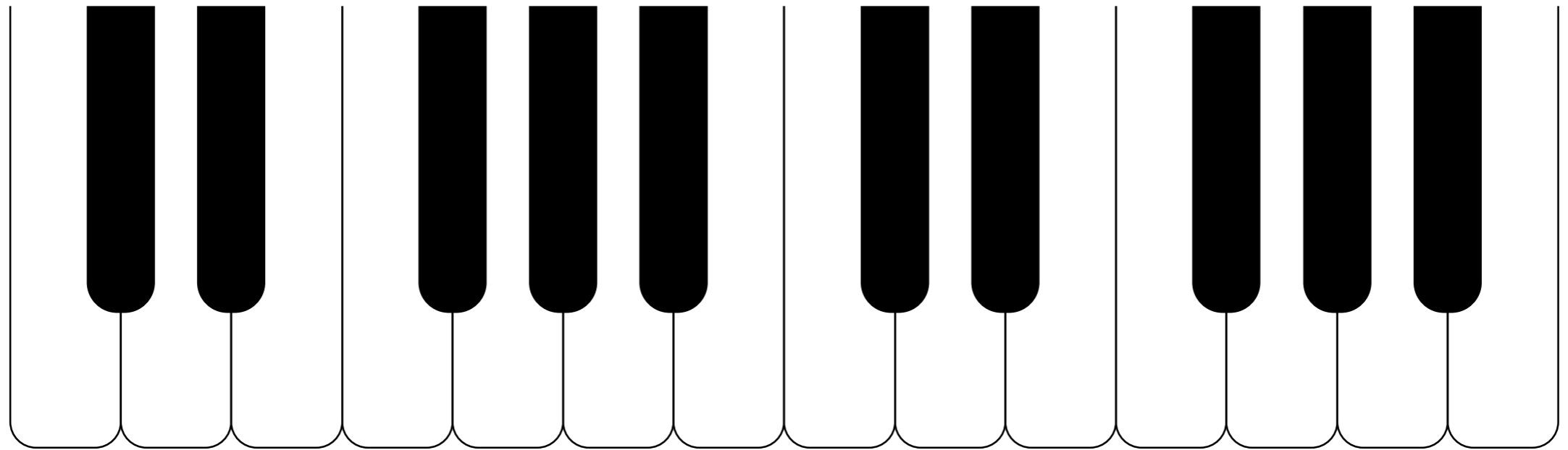
Sonic Pi



- A free live coding synth for everyone originally designed to support computing and music lessons within schools.
可即時編碼的合成器，設計為支援在學校的電腦和音樂課。
- Developed by Sam Aaron in the University of Cambridge Computer Lab.
由劍橋大學電腦中心的 Sam Aaron 及其團隊開發。
- Based on the Ruby programming language.
建基於 Ruby 編程語言。
- Available in Raspberry Pi, OS X, and Windows.
在樹莓派、OS X，和 Windows 均能使用。

Music quiz! 音樂測驗!

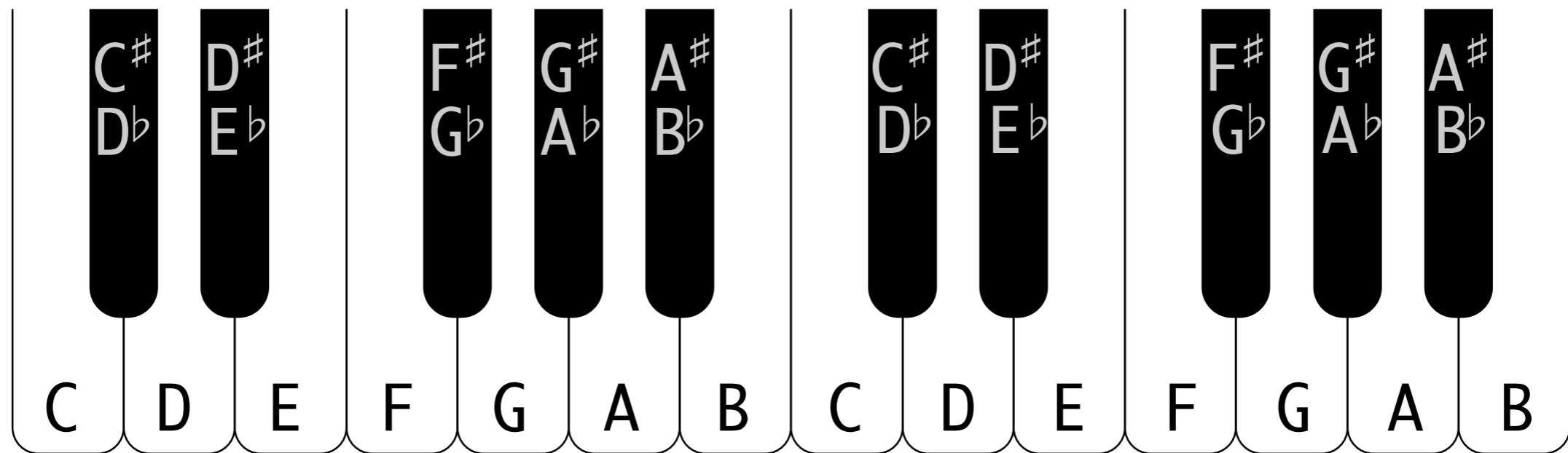
- How to names the notes on a piano?
鋼琴上的音符怎樣叫?



Music quiz! 音樂測驗!



- How to names the notes on a piano?
鋼琴上的音符怎樣叫?



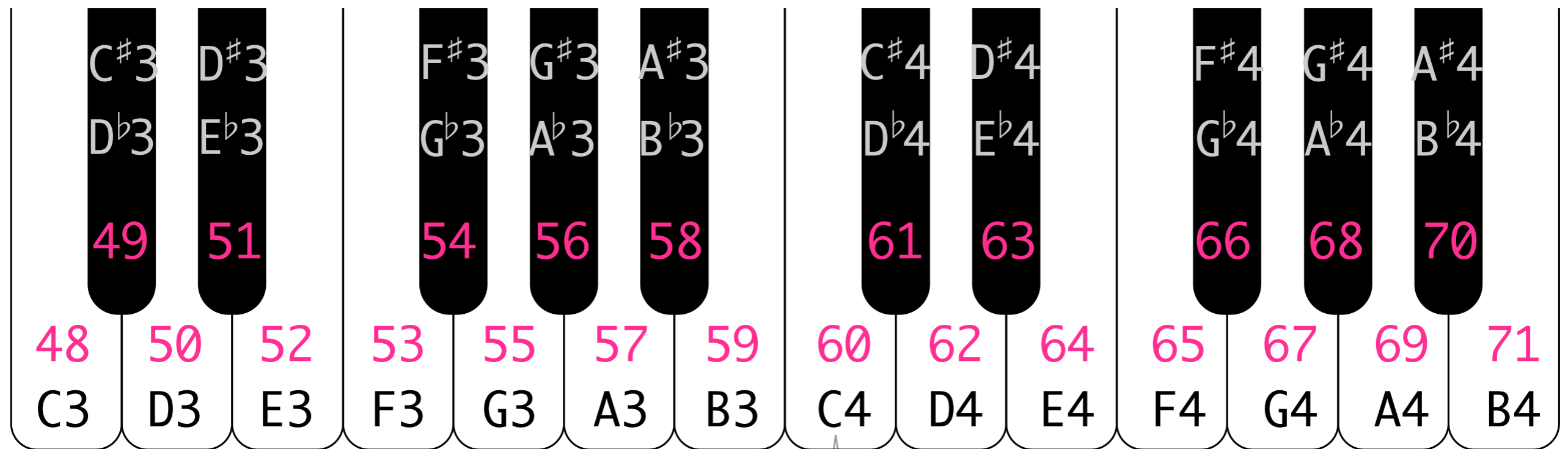
MIDI 樂器數位介面



- To process music, computers can use Musical Instrument Digital Interface (MIDI).
電腦處理音樂，可以用樂器數位介面 (MIDI)。
- Every note has a number after its name to designate the octave it is in, and every note has a unique MIDI note number.
每個音符的音名後，都有個號碼表明它在哪個八度。而每個音符都有個獨特的 MIDI 音符號碼。
- Middle C is named C4, with a MIDI number 60.
中央 C 叫 C4，MIDI 音符號碼 60。
- The MIDI number increases by one by going up a semitone.
向上一個半音，MIDI 音符號碼就會加一。

MIDI note names and numbers

MIDI 音名及音符號碼



Middle C 中

MIDI note numbers

MIDI 音符號碼

- Range of piano marked in yellow. 黃色底為鋼琴音域。

Octave number	C	C [#]	D	D [#]	E	F	F [#]	G	G [#]	A	A [#]	B
		D ^b		E ^b			G ^b		A ^b		B ^b	
-1	0	1	2	3	4	5	6	7	8	9	10	11
0	12	13	14	15	16	17	18	19	20	21	22	23
1	24	25	26	27	28	29	30	31	32	33	34	35
2	36	37	38	39	40	41	42	43	44	45	46	47
3	48	49	50	51	52	53	54	55	56	57	58	59
4	60	61	62	63	64	65	66	67	68	69	70	71
5	72	73	74	75	76	77	78	79	80	81	82	83
6	84	85	86	87	88	89	90	91	92	93	94	95
7	96	97	98	99	100	101	102	103	104	105	106	107
8	108	109	110	111	112	113	114	115	116	117	118	119
9	120	121	122	123	124	125	126	127				

Sonic Pi



- Web site 網站 : <http://sonic-pi.net/>
- Learning resource page 學習資源 : <https://www.raspberrypi.org/learning/getting-started-with-sonic-pi/>

The screenshot shows the Sonic Pi application window. The code editor contains the following code:

```
1 use_synth :piano
2 2.times do
3   play 60
4   sleep 0.5
5   play 64
6   sleep 0.5
7   play 67
8   sleep 0.5
9 end
```

The log window on the right shows the execution output:

```
Log
=> Starting run 5
[Run 5, Time 0.0]
└─ synth :piano, {note: 60.0}
[Run 5, Time 0.5]
└─ synth :piano, {note: 64.0}
[Run 5, Time 1.0]
└─ synth :piano, {note: 67.0}
[Run 5, Time 1.5]
└─ synth :piano, {note: 60.0}
[Run 5, Time 2.0]
└─ synth :piano, {note: 64.0}
[Run 5, Time 2.5]
└─ synth :piano, {note: 67.0}
=> Completed run 5
```




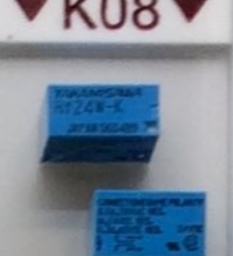
















The interface also includes a toolbar with buttons for Run, Stop, Save, Rec, Size, Align, Info, Help, and Prefs. At the bottom, there is a buffer indicator showing Buffer 0 through Buffer 9, with Buffer 0 highlighted. The version number 'Sonic Pi on Raspberry Pi v2.7' is visible in the bottom right corner.



Interfacing with the real world 與世界接軌

You need some electronic components...
你要些電子零件……

	
07-10-1224 Dae Yang DYR-MY-4C 4P2T 3A/DC 24V Relay	07-10-1224 Dae Yang DYR-MY-4C 4P2T 3A/DC 24V Relay
▼ J11 ▼	▼ J12 ▼
	
07-10-1822 Dae Yang DYR-LY-2CMA 2P2T 10A/ AC220V Relay	07-10-1822 Dae Yang DYR-LY-2CMA 2P2T 10A/ AC220V Relay
▼ J15 ▼	▼ J16 ▼
	
07-08-3024 MASSUSE ME-13F-012-2Z2 DC12V 2P2T 8 Pin Relay	07-08-3024 MASSUSE ME-13F-02-4 DC24V 2P1T 8 Pin Relay 650Ω
▼ J19 ▼	▼ J20 ▼
	
07-08-3024 MASSUSE ME-13F-012-1Z1 DC12V 1P2T 8 Pin Relay, 160Ω	07-08-3024 MASSUSE ME-13F-012-1Z1 DC24V 1P2T 8 Pin Relay, 160Ω
▼ J23 ▼	▼ J24 ▼
	
07-08-9003 MASSUSE ME-13F-A240-2Z2 AC240V 8 Pin Relay	07-08-9003 MASSUSE ME-28A-012-1Z1 DC12V 1P1T 4 Pin Relay

			
07-01-3005 TAKAMISAWA RY-5W-K DC5V/1A 2P2T 8 Pin Relay, 167Ω	07-01-3006 TAKAMISAWA RY-6WK DC6V/1A 2P2T 8 Pin Relay	07-01-3012 TAKAMISAWA RY-12WK DC12V/1A 2P2T 8 Pin Relay	07-01-3024 TAKAMISAWA RY-24W-K DC24V/1A 2P2T 8 Pin Relay
▼ K09 ▼	▼ K10 ▼	▼ K11 ▼	▼ K12 ▼
			
07-01-3205 TAKAMISAWA JY-5H-K DC5V/5A 1P1T 4 Pin Relay	07-01-3212 TAKAMISAWA JY-12H-K DC12V/5A 1P1T 4 Pin Relay	07-01-3224 TAKAMISAWA JY-24H-K DC24V/5A 1P1T 4 Pin Relay	07-01-3103 TAKAMISAWA SY-3K DC 3V/2A 1P2T 6 Pin Relay, 15Ω
▼ K13 ▼	▼ K14 ▼	▼ K15 ▼	▼ K16 ▼
			
07-01-3105 TAKAMISAWA SY-5-K DC 5V/2A 1P2T 6 Pin Relay, 167Ω	07-01-3109 TAKAMISAWA SY-9-K DC 9V/2A 1P2T 6 Pin Relay, 540Ω	07-01-3112 TAKAMISAWA SY-12K DC12V/2A 1P2T 6 Pin Relay	07-01-3124 TAKAMISAWA SY-24K DC24V/2A 1P2T 6 Pin Relay
▼ K17 ▼	▼ K18 ▼	▼ K19 ▼	▼ K20 ▼
			
07-01-4005 TAKAMISAWA A-5W-K DC 5V/2A 2P2T 10 Pin Relay, 178Ω	07-01-4012 TAKAMISAWA A-12W-K DC12V/2A 2P2T 10Pin Relay	07-01-3305 TAKAMISAWA NY-5W-K DC5V/5A 1P1T 4 Pin Relay	07-01-3312 TAKAMISAWA NY-12W-K DC12V/5A 1P1T 4 Pin Relay
▼ K21 ▼	▼ K22 ▼	▼ K23 ▼	▼ K24 ▼
			
07-01-3324 TAKAMISAWA NY-24W-K DC24V/5A 1P1T 4 Pin Relay	07-08-5303 MASSUSE ME-4-L1-3 DC3V 1 Coil Latch 2P2T 8 Pin Relay	07-08-5305 MASSUSE ME-4-L1-5B DC5V 1 Coil Latch 2P2T 8 Pin Relay, 1440Ω	07-08-5312 MASSUSE ME-4-L1-12B DC12V 1 Coil Latch 2P2T 8 Pin Relay, 1440Ω

		
07-08-1412 MASSUSE ME-15M-012-HTS DC12V 12A 1P1T 4 Pin Relay, 400Ω	07-08-9017 MASSUSE ME-25-012-HS DC12V (617Ω/ 8A 250V)	
▼ L09 ▼	▼ L10 ▼	▼ L11 ▼
		
07-08-1506 MASSUSE ME-15M-006-TS DC6V 1P2T 5 Pin Relay, 300Ω	07-08-1509 MASSUSE ME-15M-009-TS DC9V 12A 1P2T 5 Pin Relay, 300Ω	07-08-1512 MASSUSE ME-15M-012-TS DC12V 12A 1P2T 5 Pin Relay, 300Ω
▼ L13 ▼	▼ L14 ▼	▼ L15 ▼
		
07-08-5005 MASSUSE ME-1A-05B DC12V 1P2T 5 Pin Relay, 700Ω	07-08-5012 MASSUSE ME-1A-12B DC12V 2A 1P2T 5 Pin Relay, 400Ω	07-08-5024 MASSUSE ME-1A-24B DC24V 2A 1P2T 5 Pin Relay, 1600Ω
▼ L17 ▼	▼ L18 ▼	▼ L19 ▼
		
07-08-6005 MASSUSE ME-3-005-1Z5 DC12V 1P2T 6 Pin Relay, 135Ω	07-08-6012 MASSUSE ME-3-012-1Z5 DC12V 1A 1P2T 6 Pin Relay, 720Ω	07-08-9007 MASSUSE ME-12-12 12V DC 2P2T 10 Pin Relay, 1028Ω
▼ L21 ▼	▼ L22 ▼	▼ L23 ▼
		
07-08-1100 MASSUSE ME-11-000-1Z1 DC12V 1P2T 5 Pin Relay	07-08-1100 MASSUSE ME-11-012-1Z1 DC12V 12A 1P2T 5 Pin Relay, 360Ω	07-08-1100 MASSUSE ME-11-024-1Z1 DC24V 12A 1P2T 5 Pin Relay, 1440Ω

Know the components:

Light Emitting Diodes (LEDs)

認識元件：發光二極管



Light Emitting Diode

發光二極管

- An LED is a semiconductor component.
發光二極管是半導體元件。
- LED is a special kind of diode, which is a component that allows one-way flow of current only.
發光二極管是一種特別的二極管。在二極管內，電流只能單向流動。
- What makes LED different from a normal diode is that when there is current flow, it lights up.
發光二極管特別的是，電流通過時，它會發光。
- Usually, a resistor is needed to limit the current flow through an LED so it would not be burnt.
通常，發光二極管要用電阻以限流，以免被燒壞。



Light Emitting Diode

發光二極管

- Since an LED allows unidirectional current flow only, it is polarized.
正因發光二極管只能讓電流單向流動，它是有極性的元件。
- The direction of current flow is along the triangle in the circuit diagram.
電流由符號中三角形那邊的正極流往粗線那邊的負極。
- The side current enters is marked as positive using the + symbol, and the other side negative using the – symbol.
正極可用正號 + 表示，負極則用負號 –。
- The lead on the positive side of a new LED is often longer.
新的發光二極管正極腳會長些。



Light Emitting Diode 發光二極管

- The polarity of an LED can also be determined by looking at the transparent or translucent LED itself.
其實，我們可以從透明或半透明的發光二極管的結構，得知其引線的極性。
- Inside an LED, one side is larger in size than the other. The larger side connects to the negative (–) terminal.
發光二極管中，金屬大些的那邊為負極。
- Which is the positive terminal in this LED?
這隻發光二極管哪邊是正極？



LED quiz! LED問答!

- What are the polarities of the LEDs Cony is holding?
兔兔拿着兩個發光二極管，極性是甚麼？

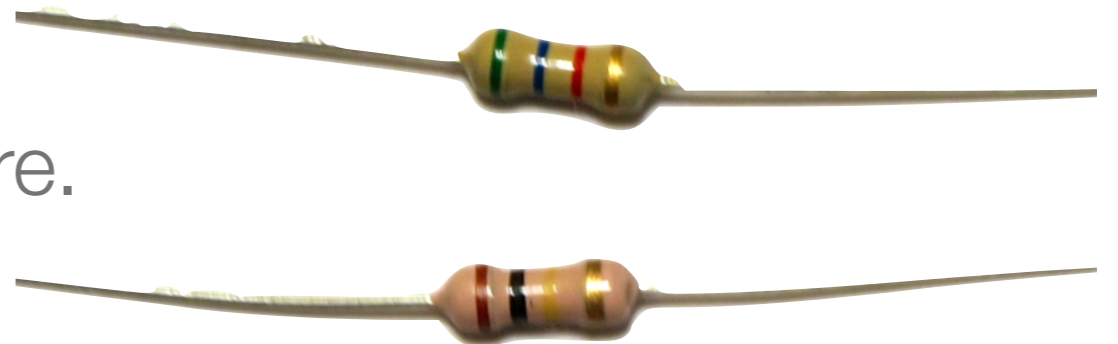


Know the components: Resistors

認識元件：電阻

Resistors 電阻

- In a circuit diagram, the zig-zag symbol correspond to resistors.
電路圖上鋸齒形的符號代表電阻。
- Resistors limit current flow and its values are measures in ohms, with the greek letter Ω (Omega) as symbol.
電阻減低電流，單位為「歐姆」 Ohm，利用希臘字母 Ω 代表。
- Resistors are non-polar. That is, unlike batteries, installing a resistor one way or another does not matter.
電阻是無極性元件。即是說，不像電池，正反來安裝也沒所謂。
- Resistors look like those in the picture.
電阻的外觀如圖。



Ohm's law 歐姆定律

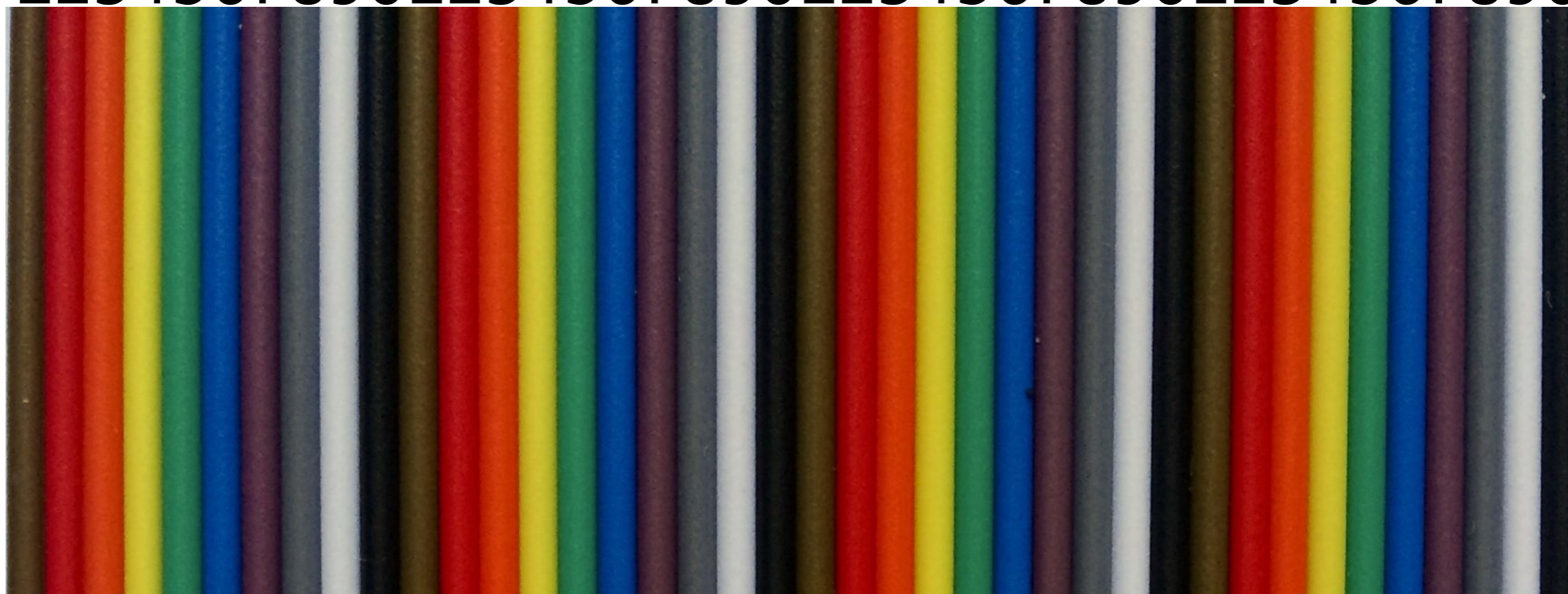
- The larger the value the resistor, the smaller will be the current flow given the same voltage.
電阻值越大，施加同樣電壓時電流越小。
- With a resistor value of R and a voltage of V , the current $I=V/R$. This is called the Ohm's law and can be remembered as $V=IR$.
若電壓為 V ，電阻值為 R ，電流 I 就等於 V/R 。此公式為歐姆定律，可以 $V=IR$ 記之。
- Many components like light emitting diodes (LEDs) cannot bear too much current, and resistors are used in series with them to limit the current so as not to damage them.
很多電子元件如發光二極管不能承受太多電流，故會串連電阻限流，以免燒壞。



Colour coding 色碼

- Resistors and some other components are colour coded.
電阻和一些電子零件都會用色碼記錄數值。

1234567890123456789012345678901234567890



Colour code table 色碼表

- Four colour bands are used for resistors with $\pm 5\%$ or $\pm 10\%$ tolerance.
誤差值為 $\pm 5\%$ 和 $\pm 10\%$ 的電阻通常有四個色環。
- Five colour bands are used for resistors with $\pm 1\%$ tolerance.
誤差值為 $\pm 1\%$ 的電阻通常有五個色環。

Colour 顏色	Value 數值	Multiplier 倍數	Tolerance 誤差
Black 黑	0	$\times 10^0$	
Brown 啡	1	$\times 10^1$	$\pm 1\%$
Red 紅	2	$\times 10^2$	
Orange 橙	3	$\times 10^3$	
Yellow 黃	4	$\times 10^4$	
Green 綠	5	$\times 10^5$	
Blue 藍	6	$\times 10^6$	
Violet 紫	7	$\times 10^7$	
Grey 灰	8	$\times 10^8$	
White 白	9	$\times 10^9$	
Gold 金		$\times 10^{-1}$	$\pm 5\%$
Silver 銀		$\times 10^{-2}$	$\pm 10\%$

To read resistor values 讀電阻數值

- First, identify the tolerance band. If there is a gold or silver band at the end of the resistor, it must be the tolerance band. Otherwise, probably your resistor has five colour bands, and a brown band at one end, which could be a bit separated from other bands or very much near the end of the resistor, is the tolerance band.
首先要確定哪個是誤差色環。如其一尾端之色環是金色或銀色，那肯定是誤差色環。如沒有，那電阻很可能是有五個色環，而其中一尾環會是啡色的誤差色環。
- The colour band besides the tolerance band is the multiplier band. Other colour bands are value bands.
它通常會距離其他色環較遠，或該環較近尾部。誤差色環旁的，就是倍數色環。其他的，就是代表數值的色環。

Reading resistor colour code 讀電阻色碼

- From the opposite side of the tolerance band, start reading the value bands. This would form a two- or three-digit number. The value of the resistor is the number, multiplied by the multiplier indicated by the multiplier band, in ohms. 要讀電阻值，先由不是誤差色環的那面開始，讀數值色環的數值。這會得出一個兩位或三位數。將此數乘以倍數色環所代表的值，便會得出電阻值，單位為歐姆。

$$5 \ 6 \ 2 \ (5\%) \Rightarrow 56 \times 10^2 = 5600\Omega = 5.6\text{k}\Omega$$



$$1 \ 0 \ 4 \ (5\%) \Rightarrow 10 \times 10^4 = 100000\Omega = 100\text{k}\Omega$$



What are their resistance values?

它們的阻值是甚麼？

$$4715\% \Rightarrow 47 \times 10^1 = 470\Omega$$

$$30101\% \Rightarrow 301 \times 10^0 = 301\Omega$$

$$47011\% \Rightarrow 470 \times 10^1 = 4.7\text{k}\Omega$$

$$1045\% \Rightarrow 10 \times 10^4 = 100\text{k}\Omega$$

Image source:

- Yet another picture of resistors. By Omegatron. <https://commons.wikimedia.org/wiki/File:Resistors.jpg>

Preferred values

標準阻值

- Resistor values may have errors, and tolerance values tell their range.
電阻值會有誤差，誤差值告訴用者阻值範圍。
- Preferred values, derived from tolerance values, are used to make mass production of resistors easier.
為使更易大量生產電阻，有利用誤差值計出來的標準阻值。
- E-12 series, most commonly used, designed for resistors with 10% tolerance:
最常用的 E-12 系列，設計給誤差值 10% 的電阻：
10, 12, 15, 18, 22, 27, 33, 39, 47, 56, 68, 82
- E-24 series, often available in resistors with 5% and 1% tolerance:
E-24 系列，誤差值 5% 和 1% 的電阻常用：
10, 11, 12, 13, 15, 16, 18, 20, 22, 24, 27, 30, 33, 36, 39, 43, 47, 51, 56, 62, 68, 75, 82, 91

References:

- <https://www.youtube.com/watch?v=jSPNvJ0XYCQ>
- <http://www.resistorguide.com/resistor-values/>

Resistor quiz!

電阻問答！

- What is the resistance of the resistors Cony is holding?
兔兔拿着兩支電阻，阻值是多少？

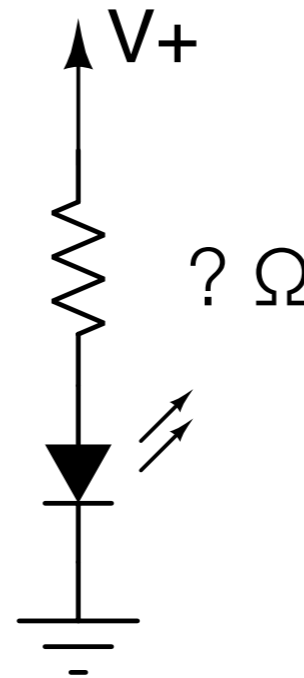


Ohm's law: example

歐姆定律：例子

- An LED operates at 1.5V and can take up at most 5mA of current. What should the series resistor value be when the power supply is 5V?

一工作電壓為1.5V 的發光二極管能承受不多於 5mA 的電流。當電源電壓為 5V，串連的限流電阻阻值應是多少？



Ohm's law: example

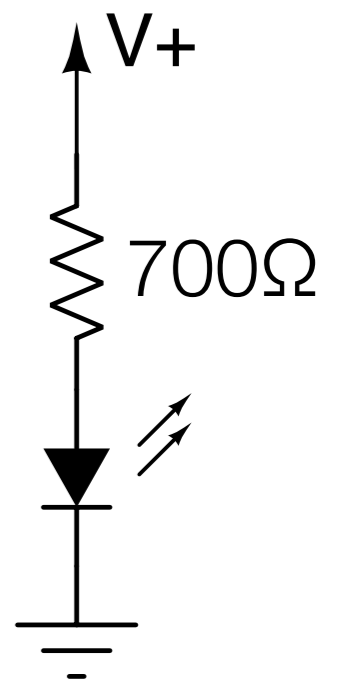
歐姆定律：例子

- An LED operates at 1.5V and can take up at most 5mA of current. What should the series resistor value be when the power supply is 5V?

一工作電壓為1.5V 的發光二極管能承受不多於 5mA 的電流。當電源電壓為 5V，串連的限流電阻阻值應是多少？

- Answer 答案: $(5V - 1.5V) / 5mA = 3.5V / 0.005A = 700\Omega$
- Higher resistance limits the current more, so a rule of thumb is to use a value slightly more than 700Ω , e.g., 820Ω .

電阻阻值大，限流更多，故可用經驗法則，用稍大於 700Ω 的阻值，例如 820Ω 。



Connecting to the real world

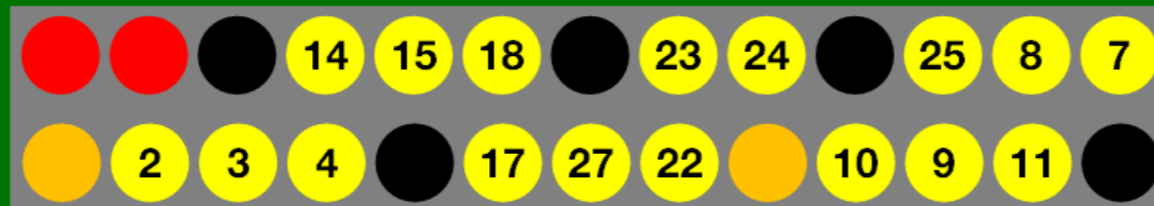
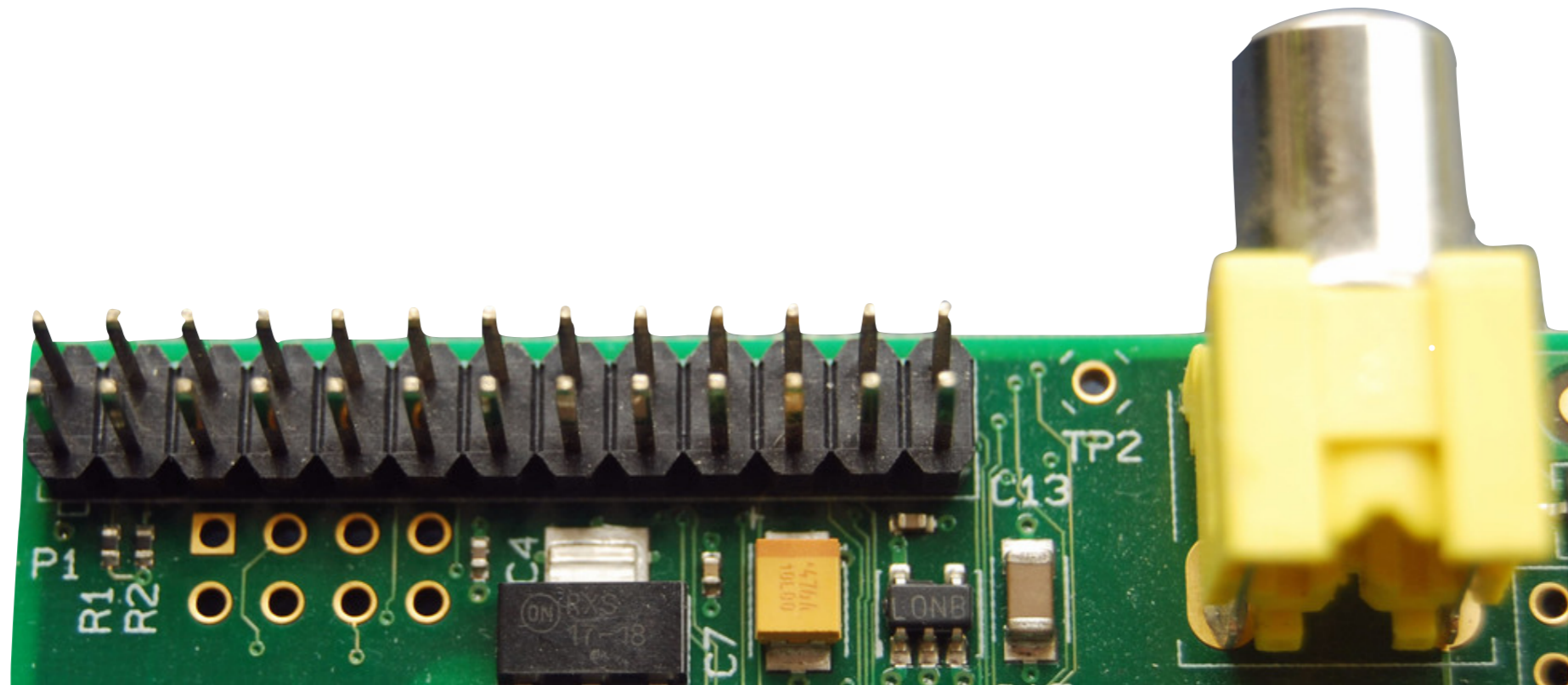
連接現實世界

GPIO 通用輸入輸出

- The General Purpose Input/Output (GPIO) pins on a Raspberry Pi connects to the world.
通用輸入輸出（GPIO）引腳可將樹莓派與外界連接。
- You can use various hardware components and modules for input/output.
可用各種零件和模塊作輸入/輸出。
- e.g., buttons, sensors, LEDs, motors, ...
例：按鈕、傳感器、發光二極管、摩打……

Raspberry Pi Model A and B GPIO pinout

Raspberry Pi Model A and B GPIO 腳位圖

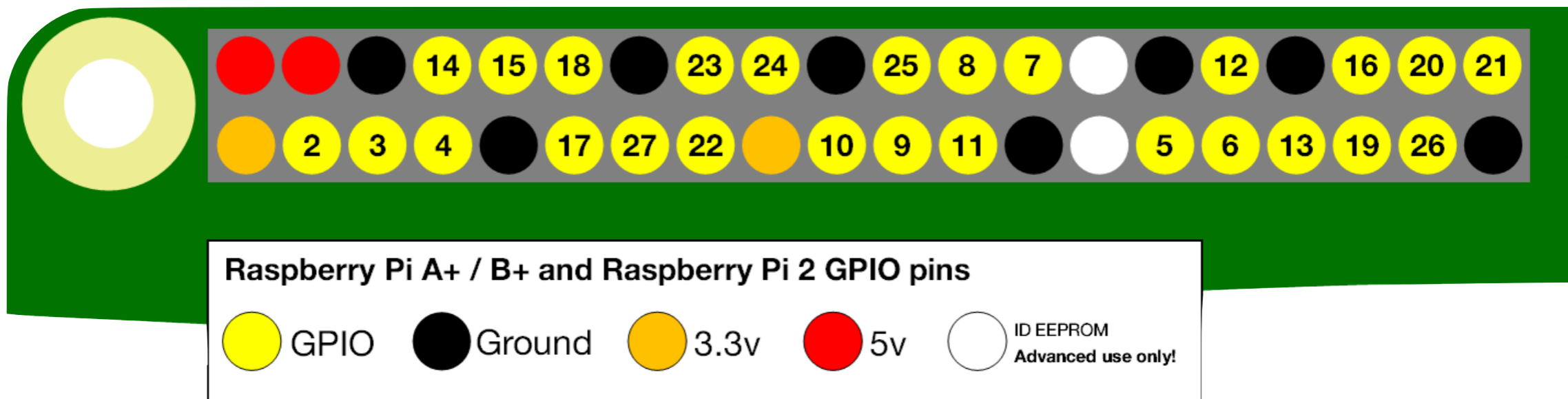
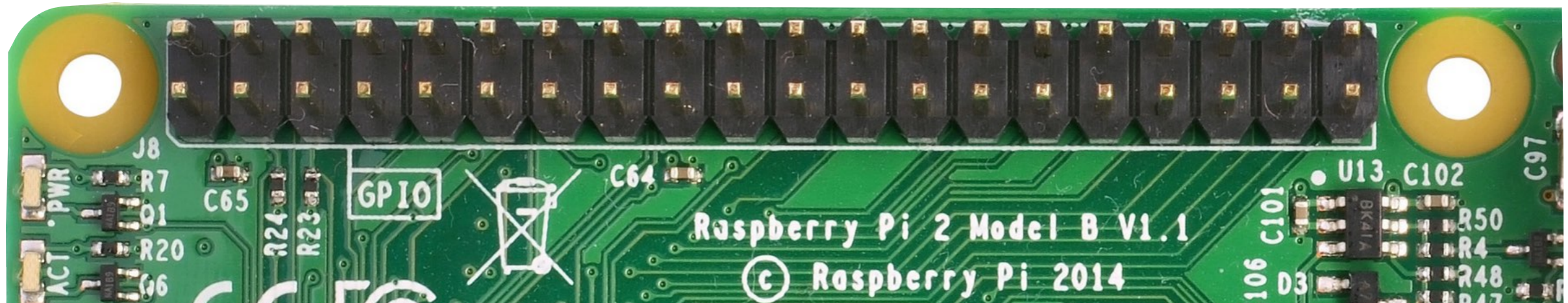


Raspberry Pi Model A and B GPIO pins

GPIO Ground 3.3v 5v

Raspberry Pi Model A+, B+ and 2 GPIO pinout

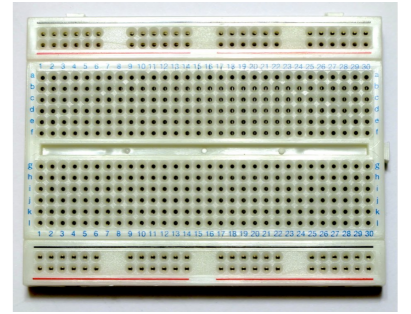
Raspberry Pi Model A+, B+ and 2 GPIO 腳位圖



- The leftmost 26 pins have pinout the same as that for Model A and B.
最左二十六腳腳位與 Model A 和 B 的腳位一樣。

Breadboard

麵包版

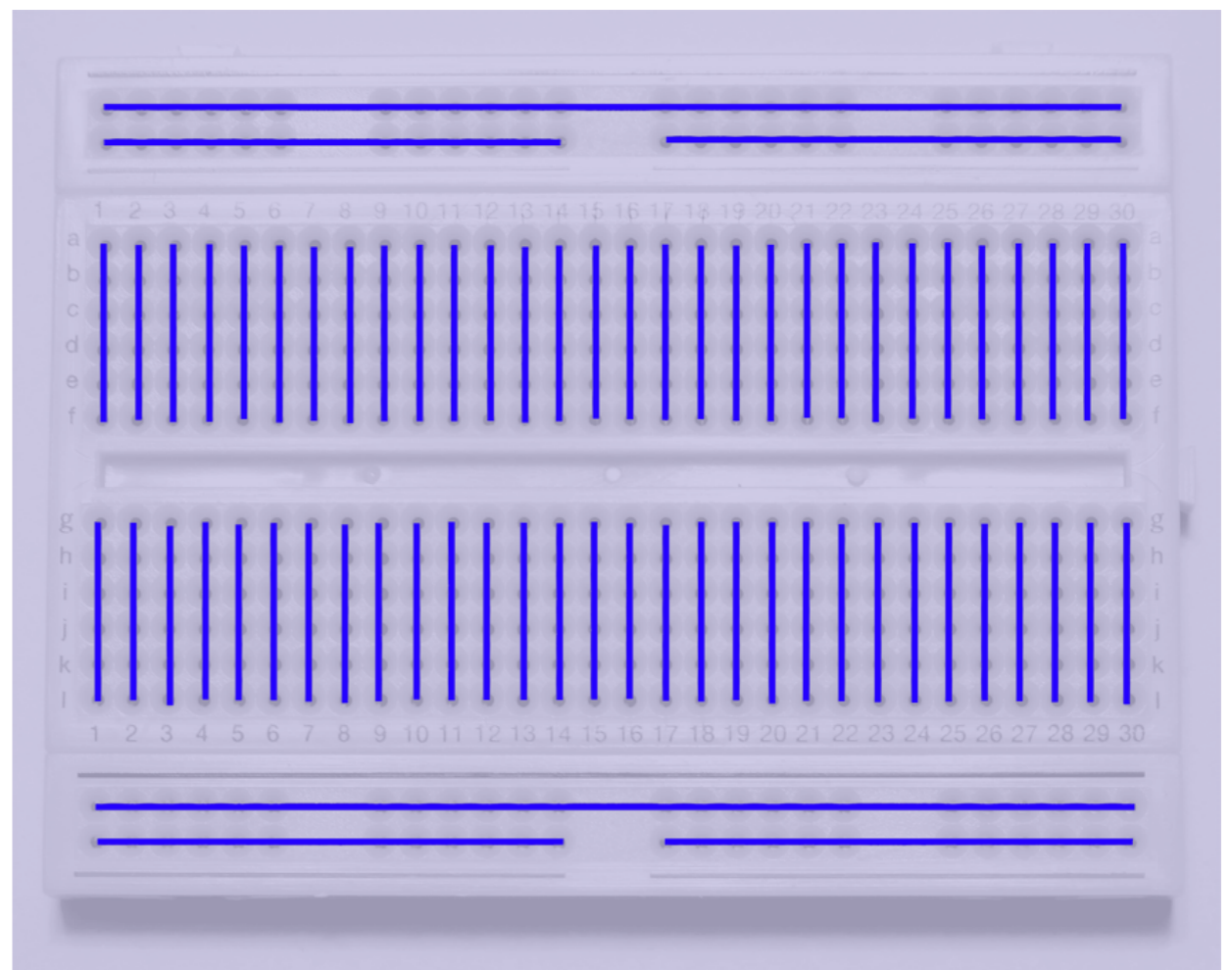
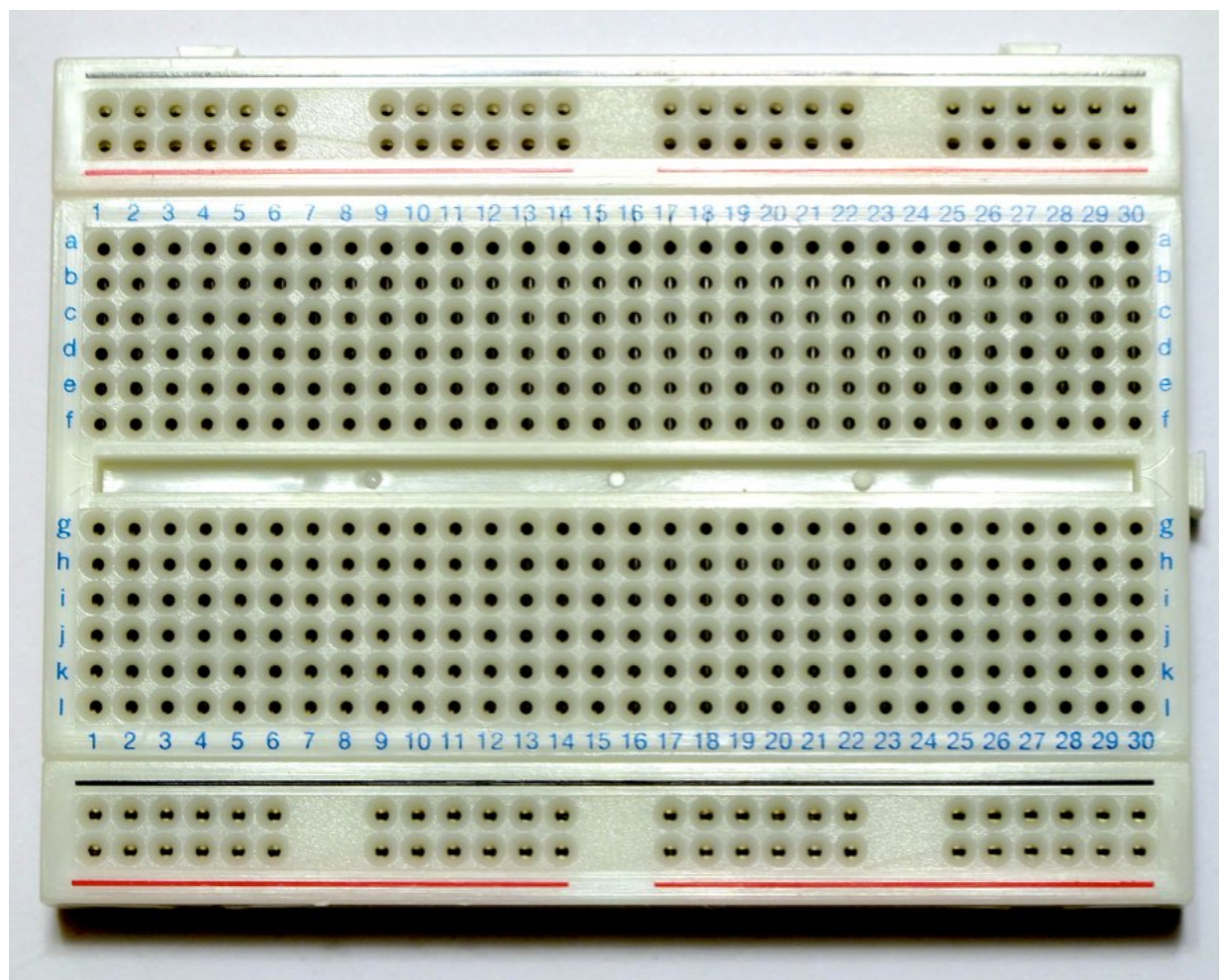


- A breadboard is a board with internal connections and holes for placing components.
麵包版上有孔洞以安插電子電件，而內部有金屬連接各洞。
- Usually, there are connected holes on the two sides for power supply connections (black for ground, red for positive voltage), and those in the middle are connected every row, though rows on the two sides are not connected.
通常，麵包版兩邊有兩行洞用來連接電源（紅正黑負），而中間的洞就每行相接，但兩邊的洞並不相連。

Breadboard: internal connections

麵包版：內裏連接例

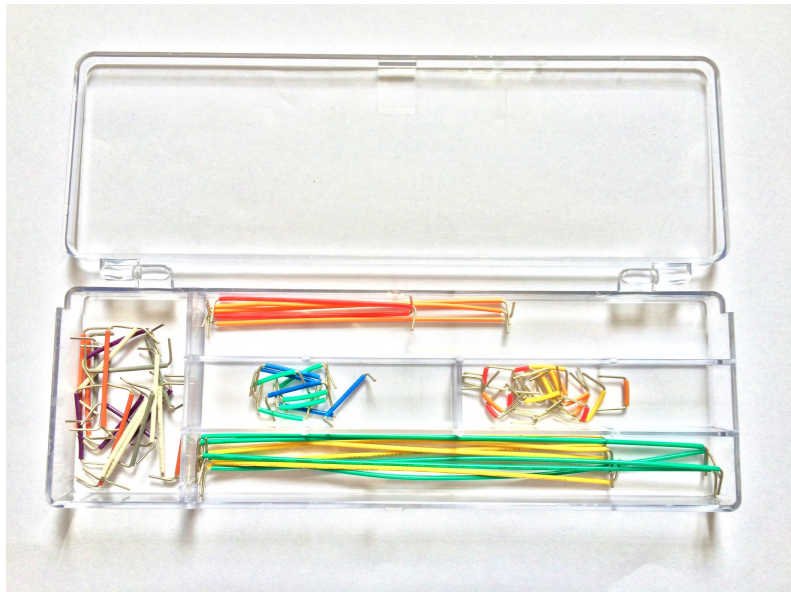
- The power connections are marked on the surface of this board. 這麵包版上有電源行的連接標記。



Connecting components

連接零件

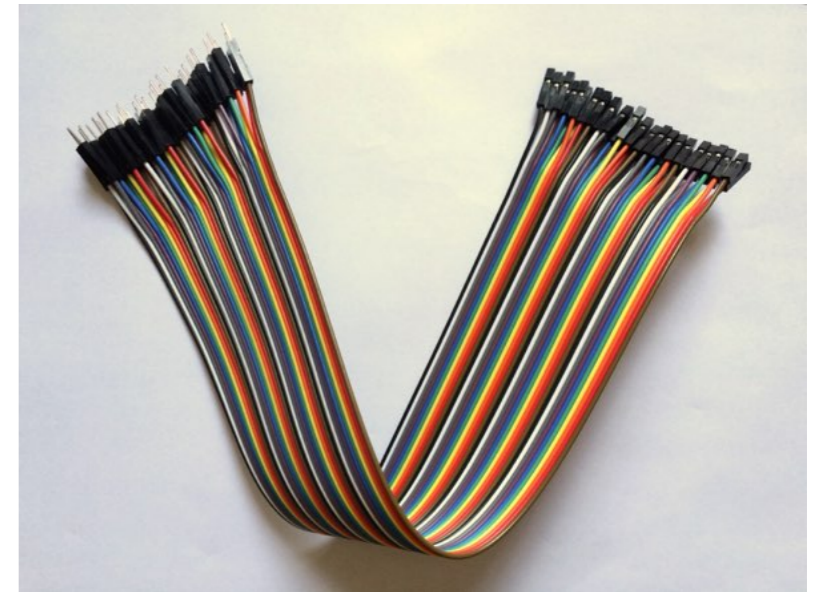
- Electronic components are connected using wires on a breadboard.
電子零件用電線在麵包版上連接。



Jumper cables
跳線



Jumper wire roll
單枝電線卷



Flat cable with connectors
杜邦彩排線

Scratch GPIO

- To use GPIO of Raspberry Pi, the GPIO server needs to be started. Use one of the following ways.

用樹莓派的 GPIO，須啟動 GPIO 伺服器，可用以下任一種方法。

- Choose Edit > Start GPIO server manually.
人手選 Edit > Start GPIO server.

- Broadcast `gpioserveron` within Scratch.
在 Scratch 程式內廣播 `gpioserveron` 訊息。



- Configure GPIO 4 to be an output pin by broadcasting the `config4out` message.
要將 GPIO 4 作輸出端子，廣播 `config4out` 訊息。
- Broadcasting the `gpio4on` and `gpio4off` message turns the GPIO on and off respectively.
廣播訊息 `gpio4on` 和 `gpio4off` 分別將 GPIO 4 開和關。

Reference:

- <https://www.raspberrypi.org/documentation/usage/scratch/gpio/README.md>

A Scratch program 一個 Scratch 程式

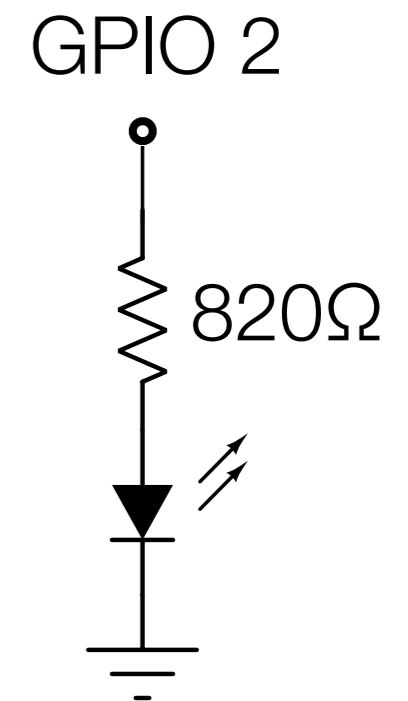
- Which GPIO pin does the program use?
這程式用哪個輸出端子？
- GPIO 2.
- What does it do? 這程式做甚麼？
- Repeatedly turn on GPIO 2 for 0.2 seconds, then turn it off for 0.2 seconds.
不斷將 GPIO 2 開 0.2 秒關 0.2 秒。



Scratch Blinking light

Scratch 閃燈

- What should the circuit be if the program is to control an LED?
如要用這程式控制一顆發光二極管，電路應是怎樣？
- Remember how the circuit to light up an LED looks like?
記得點亮發光二極管的電路是怎樣的嗎？
- It's like this. 是這樣的。
- How to use GPIO 2 to control the LED?
如何用 GPIO 2 控制發光二極管？
- This works. 這就行了。



A new program..... 一個新的程式.....

- Which part do you recognize? 你認得哪部分呢？

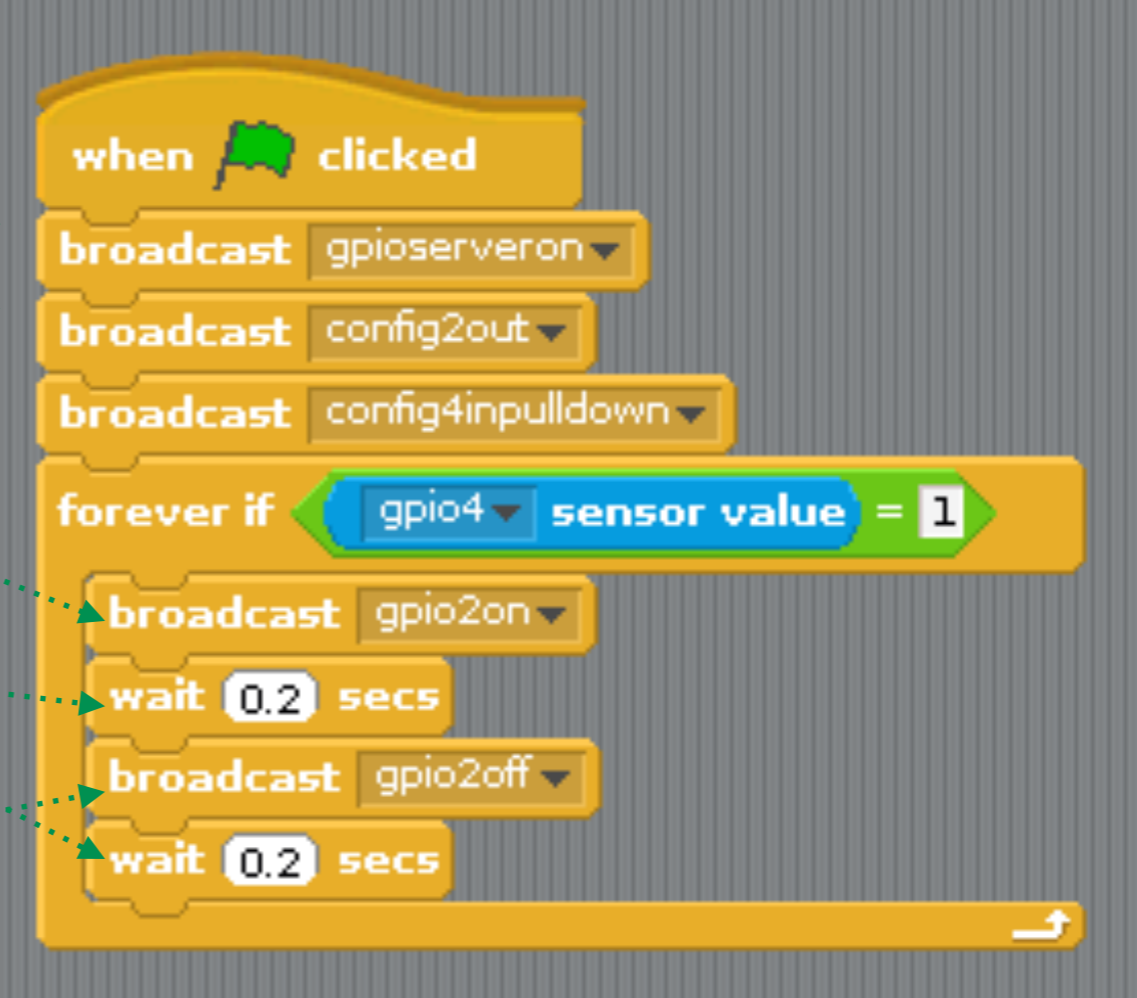
Start the GPIO server
啟動 GPIO 伺服器

Configure GPIO 2 as output
將 GPIO 2 用作輸出端子

Turn on GPIO 2
開 GPIO 2

等 0.2 秒

Turn on GPIO 2
開 GPIO 2



The image shows a Scratch script with the following blocks: a 'when clicked' block, three 'broadcast' blocks (gpioserveron, config2out, config4inpulldown), a 'forever if' loop with the condition 'gpio4 sensor value = 1'. Inside the loop are four blocks: 'broadcast gpio2on', 'wait 0.2 secs', 'broadcast gpio2off', and 'wait 0.2 secs'. Green dotted arrows point from the Chinese text on the left to the corresponding blocks in the script.

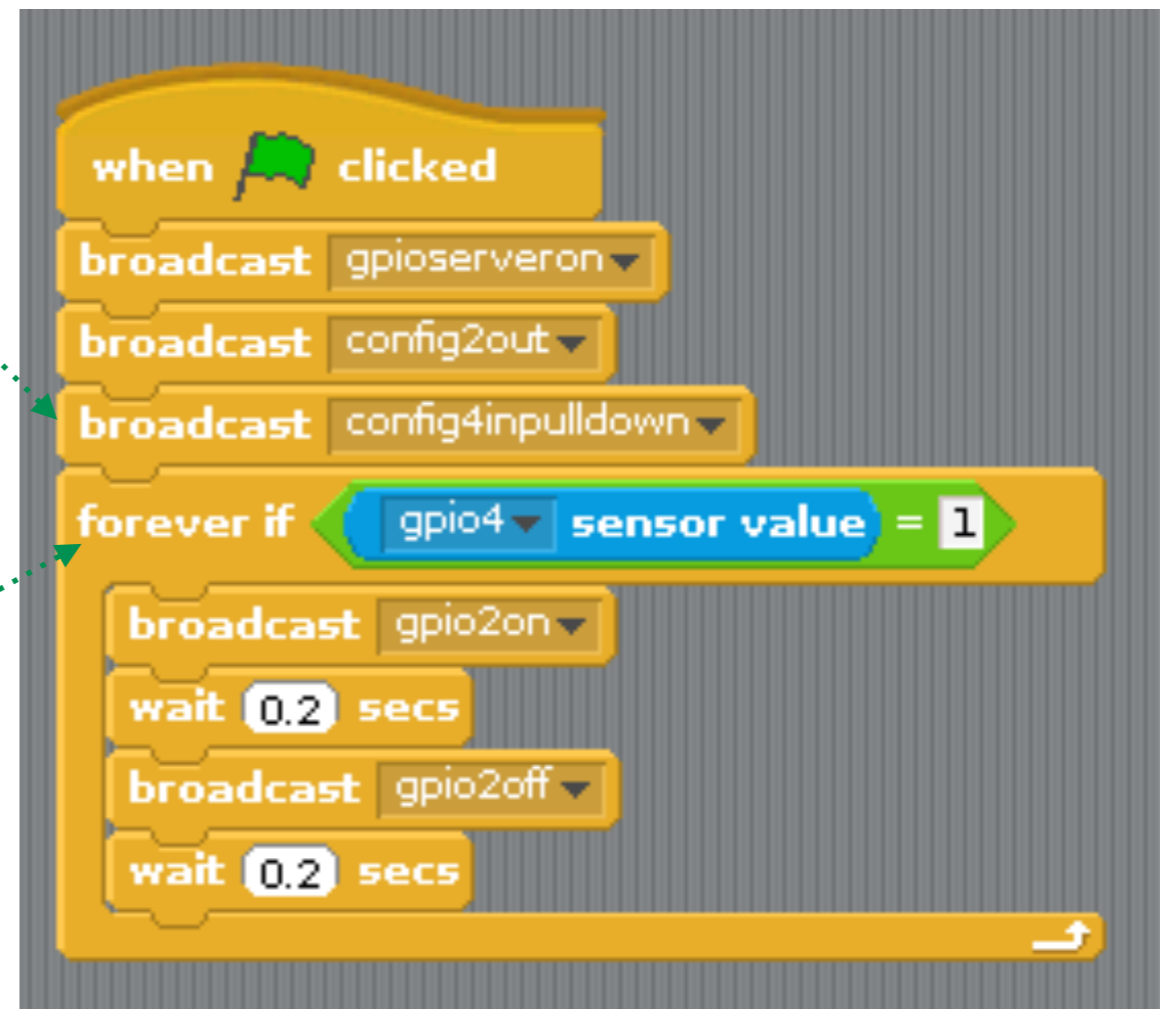
What does this program do? 這程式做甚麼？

Configure GPIO 4 as input,
with pull down resistor.
i.e., no input is taken as low (0).

將 GPIO 4 用作輸入端子，
內置下拉電阻。

即是沒有電壓輸入時當作低 (0) 輸入。

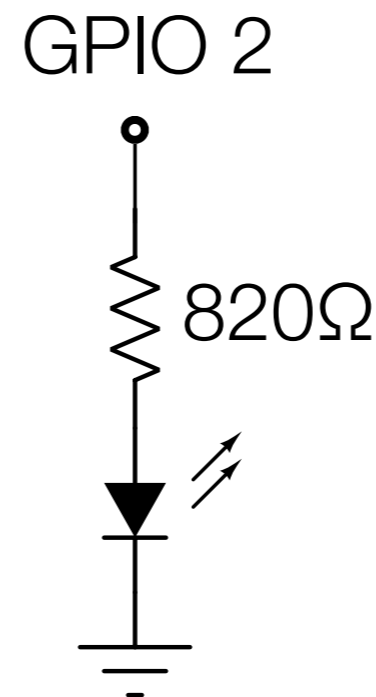
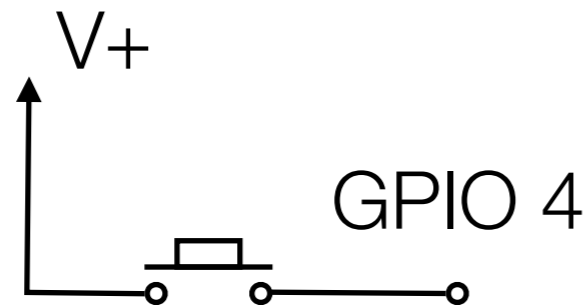
Loop only when GPIO 4 is 1.
迴圈只在 GPIO 4 為 1 時進行。



And the circuit is... 電路是.....

```
when clicked
broadcast gpioserveron
broadcast config2out
broadcast config4inpulldown
forever if gpio4 sensor value = 1
  broadcast gpio2on
  wait 0.2 secs
  broadcast gpio2off
  wait 0.2 secs
```

- The power can be connected to GPIO 4 via a push button as GPIO 4 is configured to have been pulled down. 因為 GPIO 4 有下拉電阻，它可以經按鈕直接電源。



Python GPIO: blinking light 閃燈

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
led = 2
GPIO.setup(led, GPIO.OUT)
while True:
    GPIO.output(led, 1)
    time.sleep(0.2)
    GPIO.output(led, 0)
    time.sleep(0.2)
```

- The RPi.GPIO package should be used to access GPIO pins in Python. 在 Python 用 GPIO，要用 RPi.GPIO 軟件包。
`import RPi.GPIO as GPIO`
- Use time package for generating a delay. 利用 time 軟件包使程式等待。
`import time`
- Specify that the GPIO numbers (as opposed to the physical pin numbers) are used to name the GPIO pins: 告訴系統你會用 GPIO 號碼（而不是引腳位置號碼）命名 GPIO 的引腳：
`GPIO.setmode(GPIO.BCM)`
- Suppress the warning that the GPIO pin is in use: 抑制 GPIO 引腳在使用的警告：
`GPIO.setwarnings(False)`
- Configure GPIO 2 to be an output pin by using: 要將 GPIO 2 作輸出端子，要用：
`GPIO.setup(2, GPIO.OUT)`
- Set the output of GPIO 2 to high (1) or low (0): 將 GPIO 2 輸出變成高 (1) 或低 (0):
`GPIO.output(2, 1)`
- `while True:` makes the program loop forever. `while True:` 使程式無限迴圈。

References:

- <https://www.raspberrypi.org/documentation/usage/python/more.md>
- <https://pypi.python.org/pypi/RPi.GPIO>
- <http://sourceforge.net/p/raspberry-gpio-python/wiki/Home/>

Python GPIO: blinking light with push button

Python GPIO: 按鈕閃燈

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
led = 2
button = 4
GPIO.setup(led, GPIO.OUT)
GPIO.setup(button, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)
while True:
    if GPIO.input(button):
        GPIO.output(led, 1)
        time.sleep(0.2)
        GPIO.output(led, 0)
        time.sleep(0.2)
```

- Configure GPIO 4 to be an input pin:
將 GPIO 3 用作輸入端子，要用：
`GPIO.setup(3, GPIO.IN)`

Configure an internal pull down resistor for the pin:
`GPIO.setup(3, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)`

Python programs finish when all the statements finish, so the input GPIO condition is checked within the loop to determine if the LED should be made to blink.

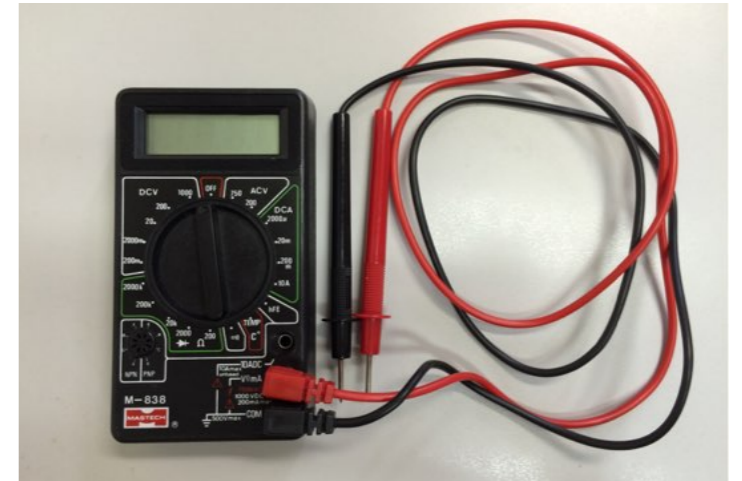
Python 程式會在所有指令完成後終結，故需在無限迴環內測試 GPIO 輸入端子的狀態，以決定是否閃動發光二極管。

References:

- <https://www.raspberrypi.org/documentation/usage/python/more.md>
- <https://pypi.python.org/pypi/RPi.GPIO>
- <http://sourceforge.net/p/raspberry-gpio-python/wiki/Home/>

Tools 工具

- Multimeter 萬用表
 - For testing components
用來測試零件
- Cutter pliers 剪鉗
 - For cutting component leads short
用來剪短零件腳



Thank you 謝謝

Questions? 問題?

