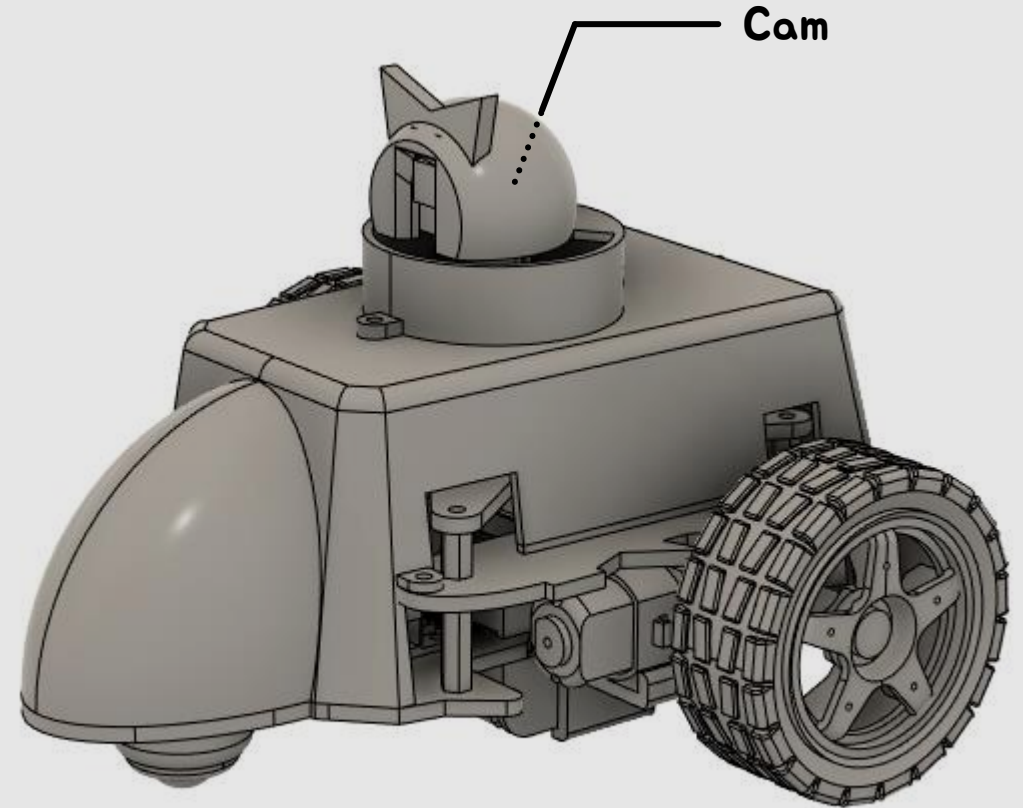


Raspberry Pi

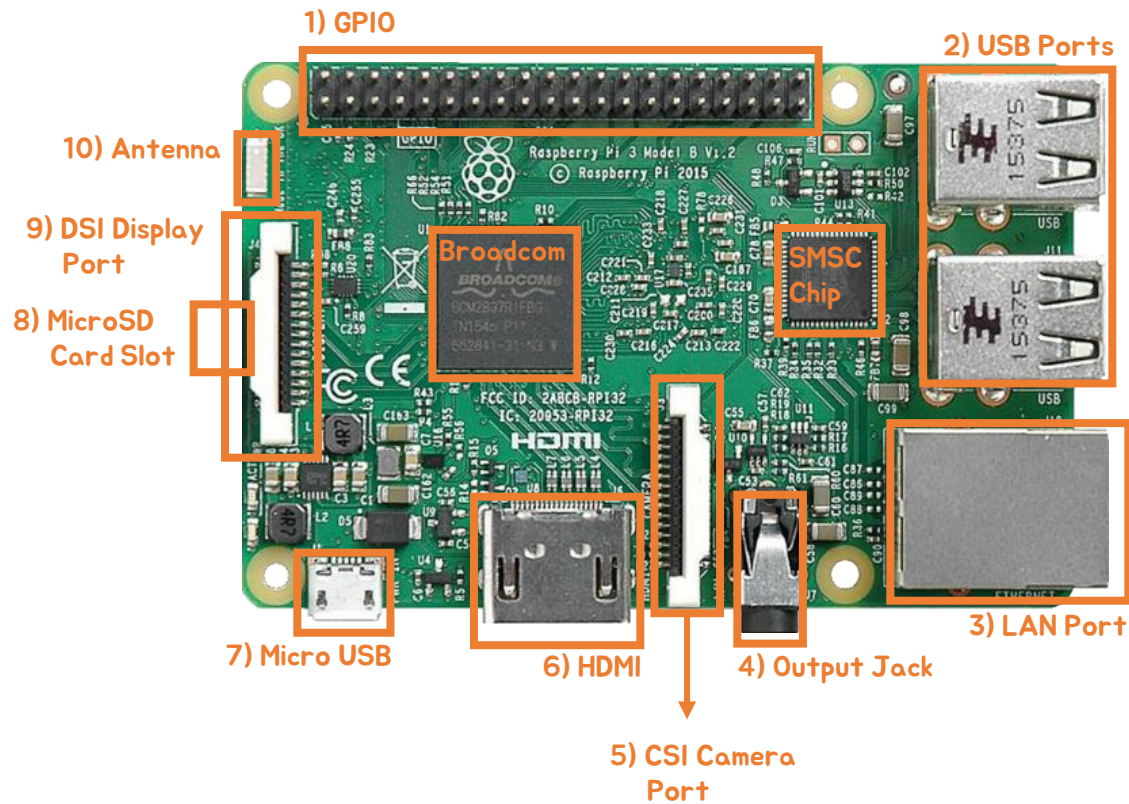
IoT CCTV RC Car



Intro

라즈베리파이란?

2012년 2월 영국의 라즈베리파이 재단에서 교육적 목적으로 만든 싱글 보드 컴퓨터(SBC)



라즈베리파이 3B Diagram

- 1) GPIO: 디지털 값의 입력/출력에 사용
- 2) USB Ports: USB 장치를 연결할 수 있는 포트
- 3) LAN Port: 이더넷 포트
- 4) Output Jack: 오디오 연결 단자
- 5) CSI: 카메라 연결 인터페이스
- 6) HDMI: HDMI 단자
- 7) Micro USB: 전원 단자
- 8) Micro SD Card Slot: 마이크로 SD카드 장착 슬롯
- 9) DSI: 디스플레이 장치 인터페이스
- 10) Antenna: 와이파이 안테나

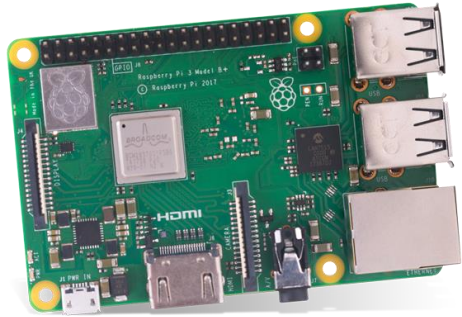
Intro

라즈베리파이 SPEC

모델명	4 MODEL B	3 MODEL B+	3 MODEL B	ZERO
출시일	19.06.24	18.03.14	16.01.29	15.11.25
SOC	Broadcom BCM2711	BCM2837B0	BCM2837	BCM2835
CPU	Quad core Cortex-A72(ARM v8) 64-bit SoC @1.5GHz	Quad Cortex-A53 @1.4GHz	Quad Cortex-A53 @1.2GHz	ARM11 @1GHz
RAM	1GB, 2GB, 4GB, 8GB LPDDR4-3200 SDRAM (depending on model)	1GB SDRAM	1GB SDRAM	512MB SDRAM
Storage	Micro-SD			
Ethernet	Gigabit Ethernet	Gigabit	10/100	None
Wireless	2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0 BLE	WLAN/Bluetooth 4.2	WLAN/Bluetooth 4.0	None ZERO W의 경우 Wireless 지원
GPIO	40	40	40	40

Intro

준비물



라즈베리파이 3B 이상



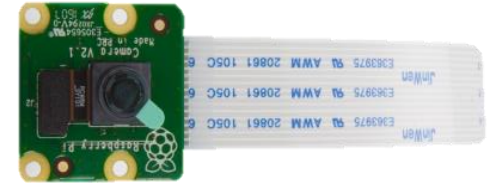
마이크로SD Card
16GB 이상



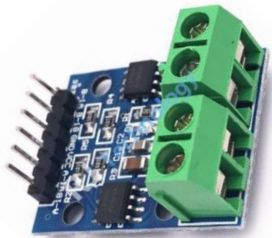
마이크로SD Card
리더기



마이크로 USB



라즈베리파이용 카메라



L9110 모터 드라이버



DC모터 x 2



서보 모터



라커 스위치



배터리 홀더



리튬 배터리



리튬 배터리 충전모듈

+ 조립용 물품

+ WIFI 공유기 관리자 ID/PWD

+ 제어할 PC (라즈베리파이를 PC로 사용시에 필요없음)

Install



Install

Raspberry OS

Install Raspberry Pi OS using Raspberry Pi Imager

Raspberry Pi Imager is the quick and easy way to install Raspberry Pi OS and other operating systems to a microSD card, ready to use with your Raspberry Pi. [Watch our 45-second video](#) to learn how to install an operating system using Raspberry Pi Imager.

Download and install Raspberry Pi Imager to a computer with an SD card reader. Put the SD card you'll use with your Raspberry Pi into the reader and run Raspberry Pi Imager.

Download for Windows

[Download for macOS](#)

[Download for Ubuntu for x86](#)

To install on **Raspberry Pi OS**, type
`sudo apt install rpi-imager`
in a Terminal window.



참고 링크:

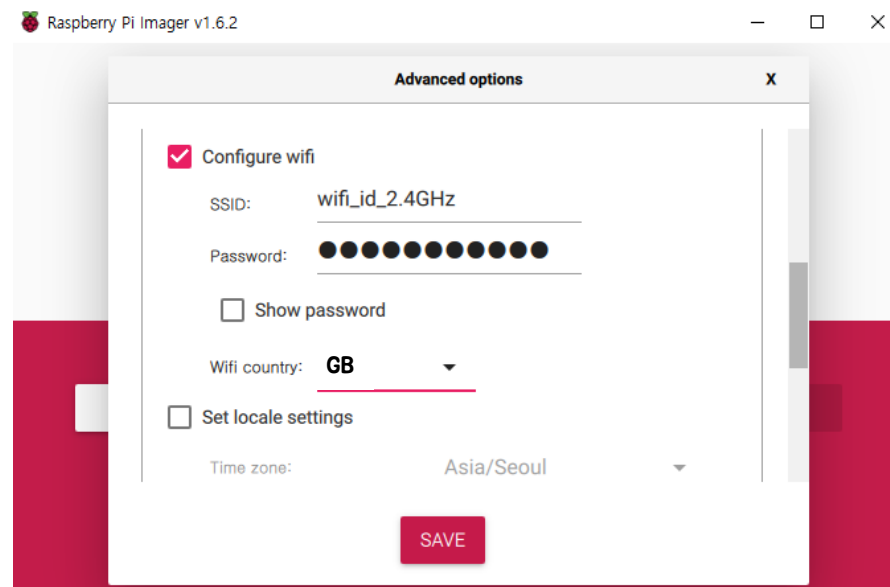
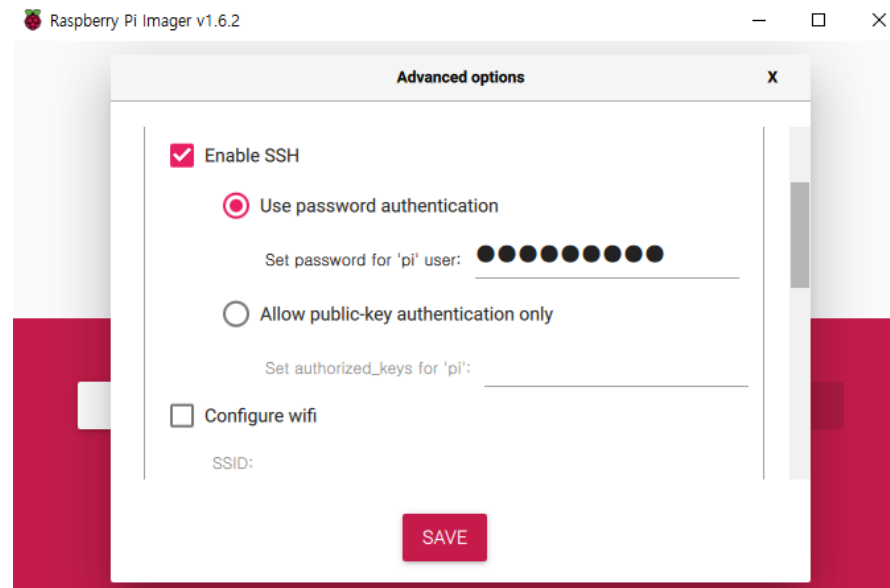
<https://www.youtube.com/watch?v=ntaXWS8Lk34>

Install

Raspberry OS



Ctrl + Shift + x



CHOOSE OS → Use custom

(최신 버전에서 카메라 호환의 문제가 있는 것으로 판단)

구버전: <http://downloads.raspberrypi.org/raspbian/images/raspbian-2020-02-14/>

Install

Raspberry OS



Install

program



Install

Python

<https://www.python.org/>

1991년 귀도 반 로섬이 발표한 고급 프로그래밍 언어

언어 이름은 귀도가 좋아하는 코미디 Monty Python's Flying Circus에서 따옴

인터프리식 언어, 객체지향적, 동적 타이핑 대화형 언어

비영리 파이썬 소프트웨어 재단이 관리하고 있으며, 사실상 표준은 C언어로 구현된 Cython



귀도 반 로섬

***인터프리식 언어:** 코드를 한줄 한줄 읽어가며 명령을 처리

****객체지향적:** 파이썬은 Object(객체)로 이루어져 있으며, 객체 지향 프로그래밍을 추구하고 있다.

OOP(Object Oriented Programming, 객체 지향 프로그래밍): 프로그래밍에 필요한 데이터를 추상화하여 상태와 행위를 가진 객체를 만들고, 객체들 간의 유기적 상호작용을 통해 로직을 구성하는 방법

*****동적 타이핑 대화형 언어:** 파이썬은 변수를 만들 때 변수의 타입을 먼저 지정해주지 않습니다. 그리고, 프로그램 자체가 영어 친화적으로 대화하듯 되어 되어 있습니다.

Install

코드 편집툴: VS Code

<https://code.visualstudio.com/>

The image shows the Visual Studio Code website and a screenshot of the VS Code interface. The website header includes links for Visual Studio Code, Docs, Updates, Blog, API, Extensions, FAQ, and Learn, along with a Search Docs button and a Download button. A banner for Version 1.60 is visible. The main content area features the text "Code editing. Redefined." and "Free. Built on open source. Runs everywhere." with a "Download for Windows" button and a link to "Other platforms and Insiders Edition". A disclaimer states: "By using VS Code, you agree to its license and privacy statement."

The screenshot of the VS Code interface shows the Extensions Marketplace on the left with a list of extensions including Python, GitLens, C/C++, ESLint, Debugger for Chrome, Language Support for Java, vscode-icons, Vetur, and C#. The main editor area displays a JavaScript file named "serviceWorker.js" with a dropdown menu showing various browser APIs. The terminal at the bottom shows the output of a command, indicating that the create-react-app can be viewed in the browser at http://localhost:3000/.

Install

Raspberry Pi, IP 확인

ipTIME A3008-MU

192.168.0.1/session-bin/timepro.cgi?tmnu=main_frame&smnu=main_frame

ipTIME A3008-MU

메뉴탐색기

- 기본 설정
 - 시스템 요약 정보
 - 인터넷 설정 정보
 - 무선 설정/보안
 - 펌웨어 업그레이드
 - Easy Mesh
- 고급 설정
 - 네트워크 관리
 - 인터넷 설정 정보
 - 내부 네트워크 설정
 - DHCP 서버 설정
 - 무선랜 관리
 - NAT/라우터 관리
 - 보안 기능
 - 특수기능
 - 트래픽 관리
 - 시스템 관리

내부 네트워크 설정

내부 IP주소 192.168.0.1 (70-5D-CC-01-28-9B)

서브넷 마스크 255.255.255.0

☐ 허브/AP모드 내부 게이트웨이

내부 게이트웨이 ☐ 수동 지정

내부 기본 DNS

내부 보조 DNS ☐ 수동 지정

적용

사용중인 IP 주소 정보 21개사용중

192.168.0.53	B8-27-EB-BF-98-4D	raspberrypi	무선:자동할당
--------------	-------------------	-------------	---------

Mobile UI

Raspberrypi ip주소 확인

Install

VS Code SSH Extension

VS Code Extension에 'Remote -SSH'를 설치하여 라즈베리파이와 연결

The screenshot displays the Visual Studio Code interface with the 'Remote - SSH' extension page open. The left sidebar shows the 'EXTENSIONS: MARKET...' view with a search for 'ssh'. The main panel shows the 'Remote - SSH' extension by Microsoft, version 0.65.8, with a 4.5-star rating and 4,825,099 downloads. The extension description states: 'Open any folder on a remote machine using SSH and take advantage of VS Code's full feature set.' Below the description, there are tabs for 'Details', 'Feature Contributions', and 'Extension Pack'. The 'Details' tab is active, showing a description of the extension and a list of features. The 'Features' section lists:

- Develop on the same operating system you deploy to or use larger, faster, or more specialized hardware than your local machine.
- Quickly swap between different, remote development environments and safely make updates without worrying about impacting your local machine.
- Access an existing development environment from multiple machines or locations.
- Debug an application running somewhere else such as a customer site or in the cloud.

Below the features, a note states: 'No source code needs to be on your local machine to gain these benefits since the extension runs commands and other extensions directly on the remote machine. You can open any folder on the remote machine and work with it just as you would if the folder were on your own machine.'

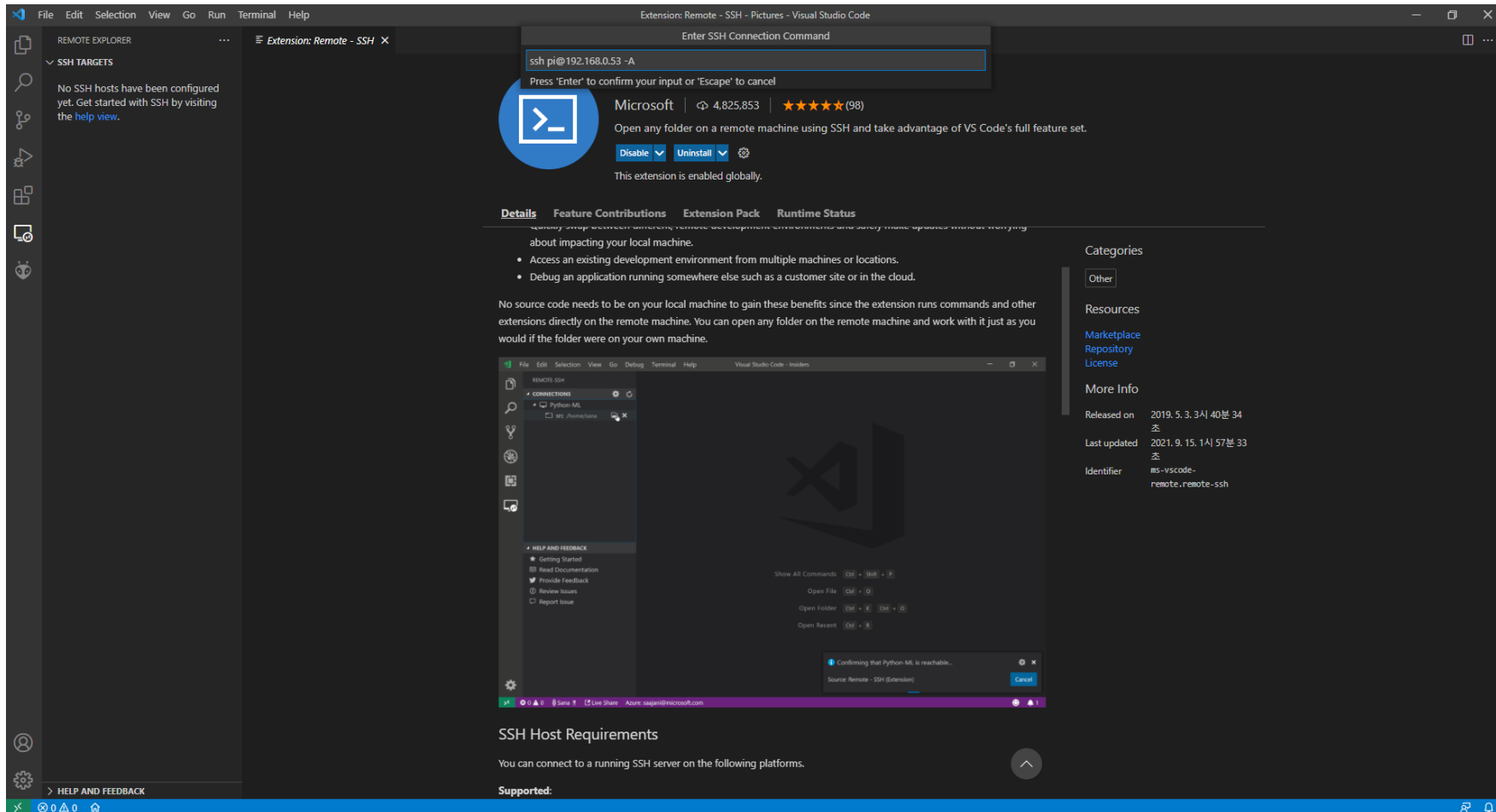
On the right side of the extension page, there are sections for 'Categories' (Other), 'Resources' (Marketplace, Repository, License), and 'More Info' (Released on 2019.5.3 3시 40분 34 초, Last updated 2021.9.15 1시 57분 33 초, Identifier ms-vscode-remote.remote-ssh).

At the bottom of the screenshot, a preview window shows a Python script named 'main.py' running on a remote machine. The script imports 'time', 'sklearn', 'sklearn.datasets', 'sklearn.model_selection', and 'sklearn.metrics'. It defines a function 'fetch_opensml' to fetch data from the opensml dataset repository. The script then uses 'cross_val_score' to evaluate a model on the fetched data. The output of the script is shown in the terminal window.

Install

VS Code SSH Extension

ssh id@ip주소 -A

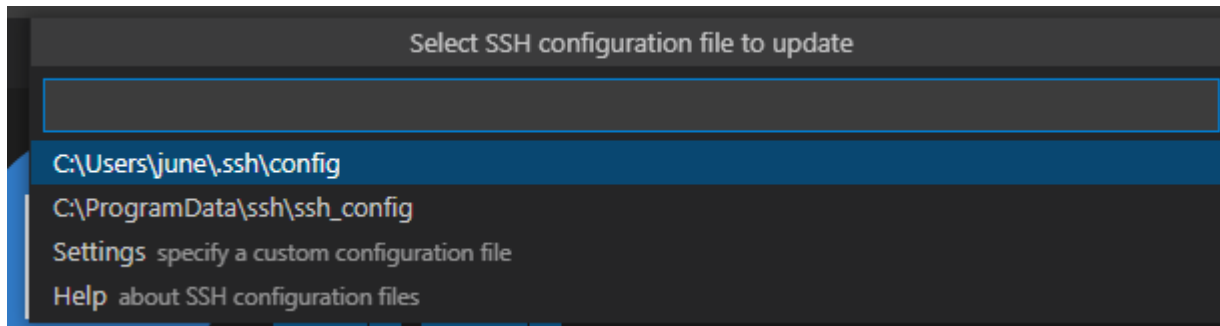


Install

VS Code SSH Extension

설정할 파일 위치 저장

기본 처음으로 설정해서 진행함



Install

VS Code SSH Extension

The screenshot shows the Visual Studio Code interface with the Remote - SSH extension page open. The extension is by Microsoft, version v0.65.8, and is marked as a Preview. It has 4.825,853 downloads and a 5-star rating from 98 reviews. The extension is enabled globally. The page includes tabs for Details, Feature Contributions, Extension Pack, and Runtime Status. The Details tab is active, showing a description of the extension and a list of features. A terminal window is open in the background, showing the command prompt and the extension's output. The terminal window has a title bar that says "Visual Studio Code - Insiders". The terminal content includes the command "ssh" and the output "Confirming that Python ML is reachable...". The terminal window also shows the "Remote Explorer" sidebar with a list of SSH targets, including "192.168.0.53". The terminal window has a status bar at the bottom that says "Source Remote - SSH (Extension)".

해당 부분을 클릭하여 연결

Remote - SSH v0.65.8 Preview
Microsoft | 4,825,853 | ★★★★★ (98)
Open any folder on a remote machine using SSH and take advantage of VS Code's full feature set.
[Disable](#) [Uninstall](#) [Settings](#)
This extension is enabled globally.

Details Feature Contributions Extension Pack Runtime Status

Many developers use remote development environments and they more often are working about impacting your local machine.

- Access an existing development environment from multiple machines or locations.
- Debug an application running somewhere else such as a customer site or in the cloud.

No source code needs to be on your local machine to gain these benefits since the extension runs commands and other extensions directly on the remote machine. You can open any folder on the remote machine and work with it just as you would if the folder were on your own machine.

Categories
Other

Resources
[Marketplace](#)
[Repository](#)
[License](#)

More Info
Released on 2019. 5. 3. 3시 40분 34초
Last updated 2021. 9. 15. 1시 57분 33초
Identifier ms-vscode-remote.remote-ssh

SSH Host Requirements
You can connect to a running SSH server on the following platforms.
Supported:

Host added!
Source: Remote - SSH (Extension) [Open Config](#) [Connect](#)

Install

VS Code SSH Extension

Linux로 설정

Select the platform of the remote host "192.168.0.53"

Linux

Windows

macOS



Continue

Are you sure you want to continue?

Continue

Cancel



비밀번호 작성: 초기 비밀번호: raspberry

Enter password for pi@192.168.0.53

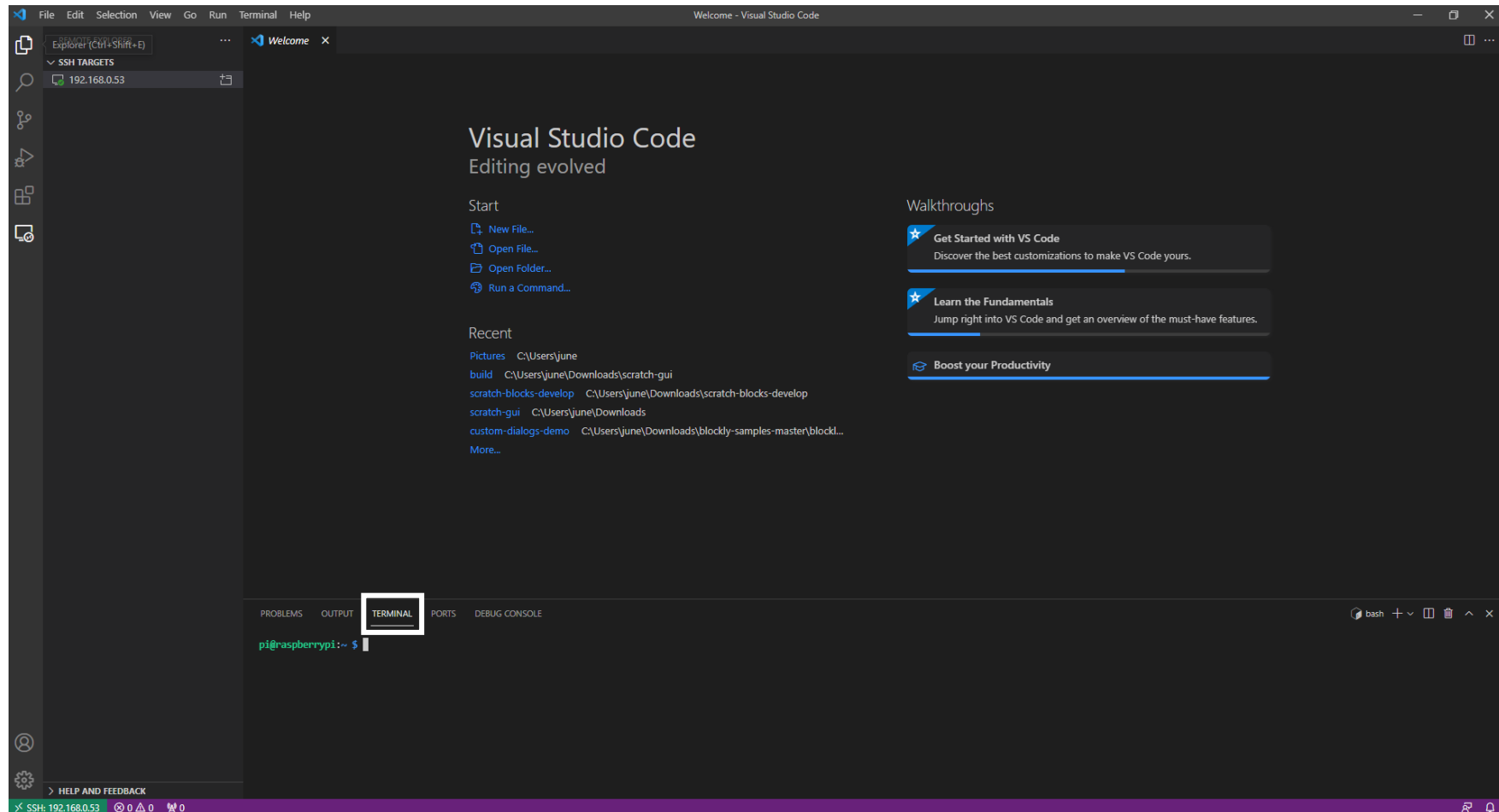
|

Press 'Enter' to confirm your input or 'Escape' to cancel

Install

VS Code SSH Extension

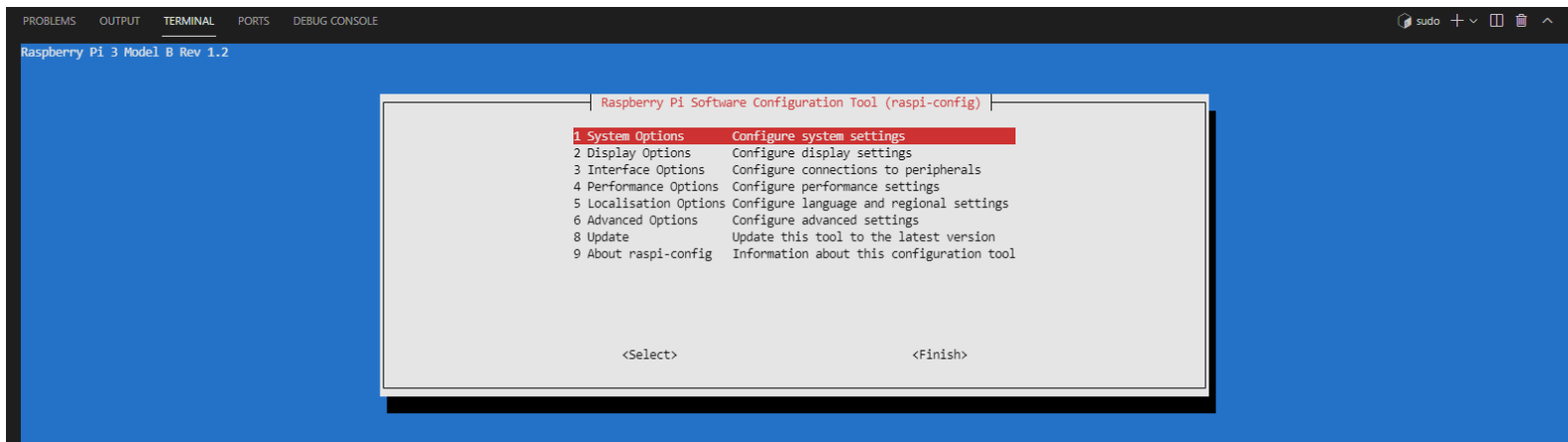
터미널을 열면 SSH로 제어 가능



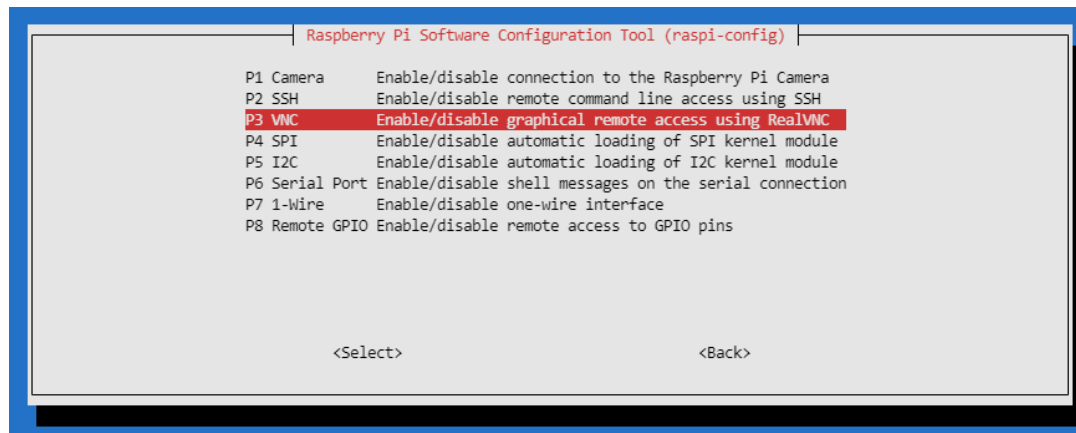
Install

Raspberry Pi VNC ON

터미널에 'sudo raspi-config' 작성시에 해당 설정이 나옴



해당 부분에서 '3 Interface Options' 클릭시에 아래와 같이 나오며



VNC를 눌러서 'YES'를 눌러서 사용 가능하게 설정 → 이와 같이 하면 VNC Viewer에서 접근 가능

Install

VNC Viewer

<https://www.realvnc.com/en/connect/download/viewer/>

VNC® Connect consists of VNC® Viewer and VNC® Server

Download VNC® Viewer to the device you want to control from, below. Make sure you've [installed VNC® Server](#) on the computer you want to control.



Windows



macOS



Linux



Raspberry Pi

iOS

iOS



Android

solaris

Solaris

hp ux

HP-UX



AIX

Download VNC Viewer

SHA-256:

EXE x86/x64

[Looking for VNC® Server?](#)

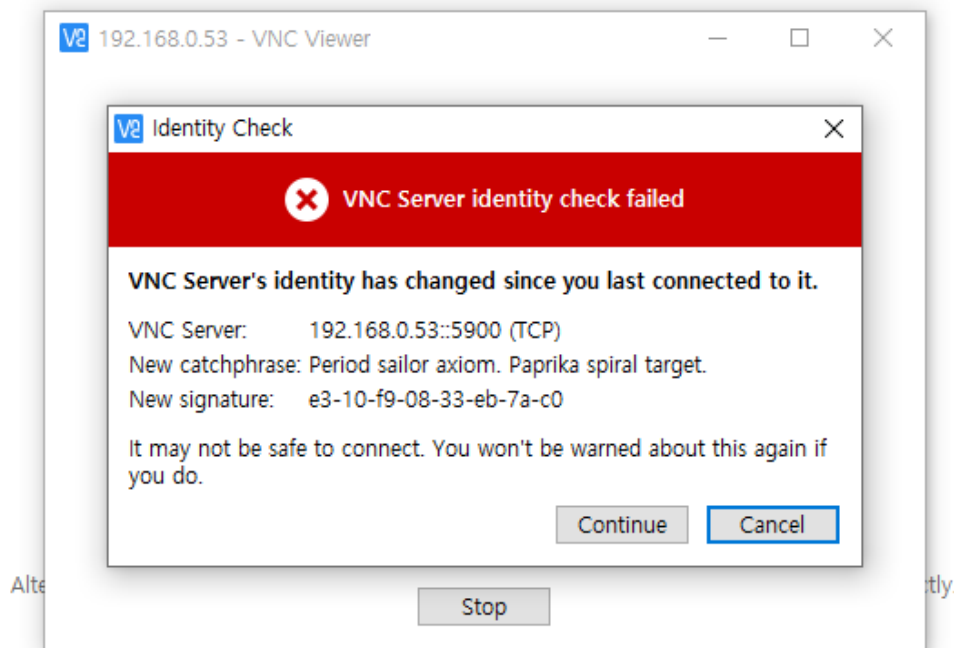
VS Code SSH로 연결하여 진행할 시에 VNC Viewer 필요없이 텍스트 상으로만 제어하실 수도 있습니다.

Install

VNC Viewer



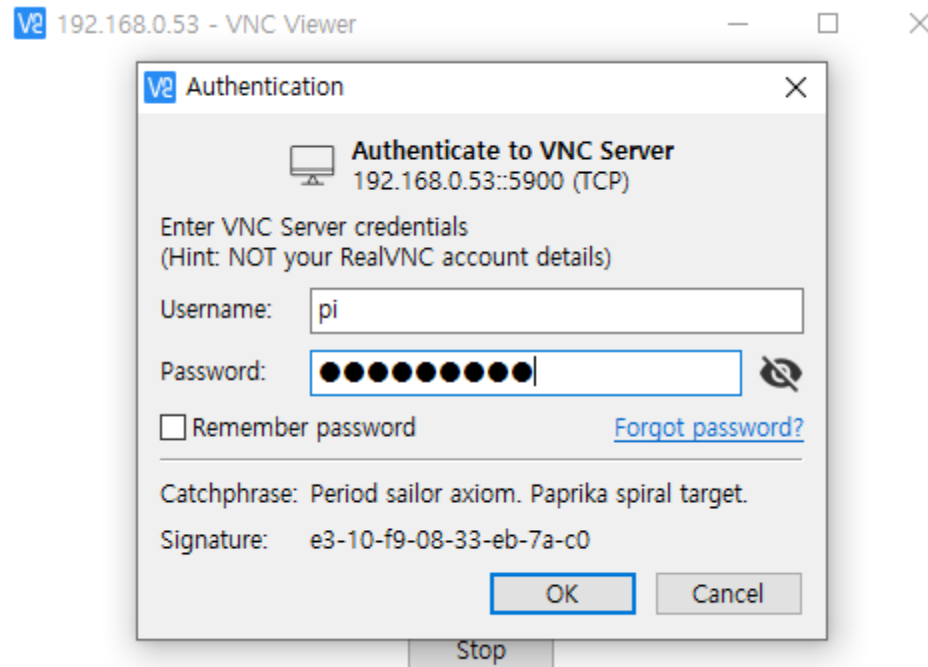
라즈베리파이 ip 주소 작성을 하고 엔터를 누르면



해당 창이 나오고, continue를 눌러서 진행하면 됩니다.

Install

VNC Viewer



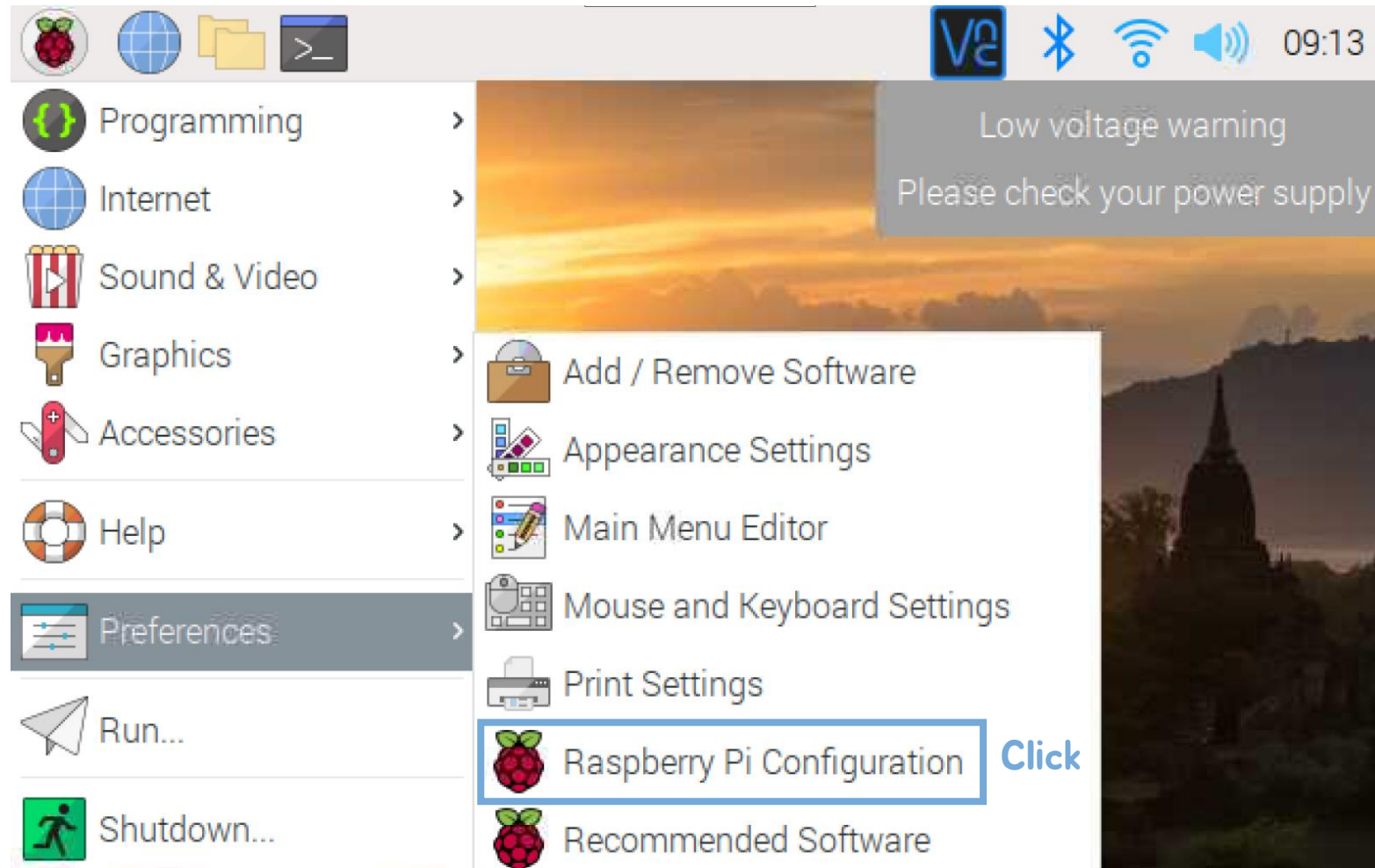
Username과 password를 작성하고 'OK'을 클릭하면 됩니다.

기본 Username: pi

기본 Password: raspberry

Install

VNC Viewer

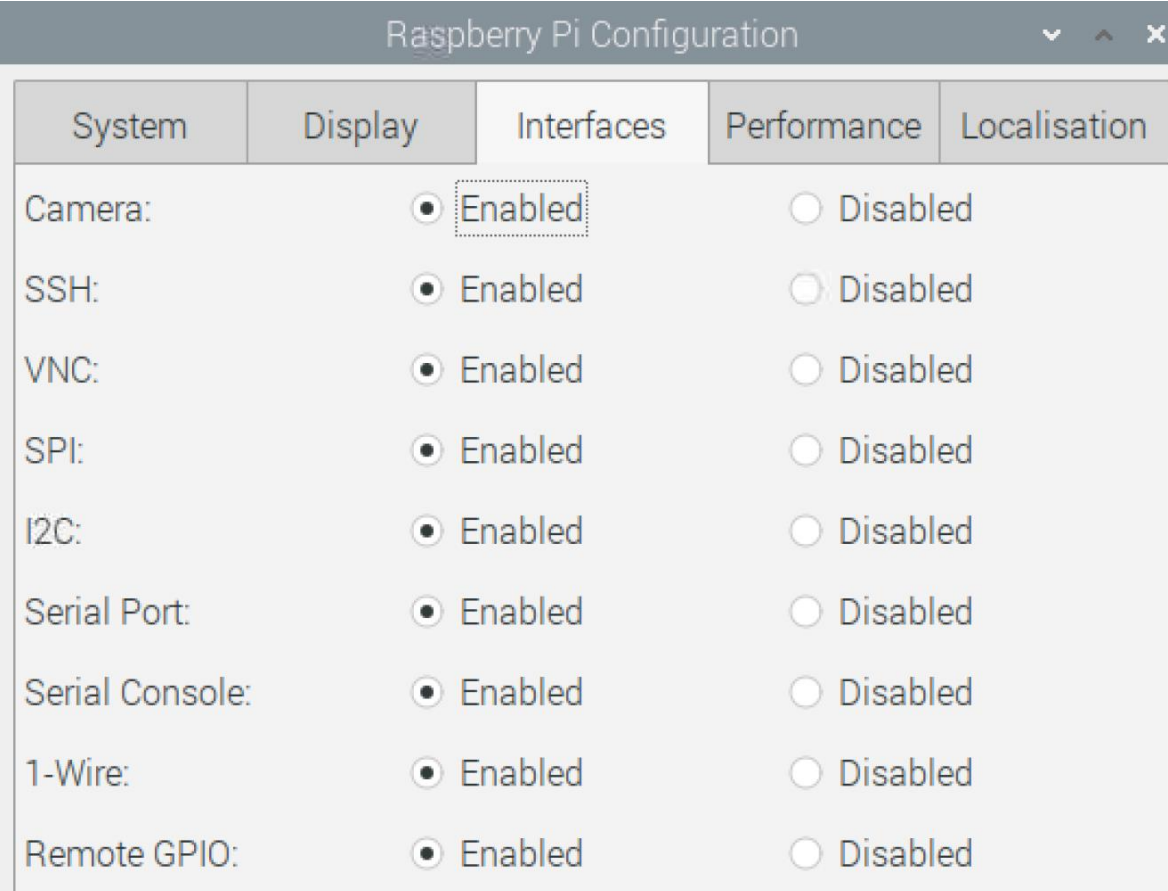


Install

VNC Viewer

Interfaces 부분에 Enabled로 다 두어서 사용 가능하게 하겠습니다.

본래, 사용 목적에 따라 설정해야 하지만, 편의를 위해 다 가능하게 설정



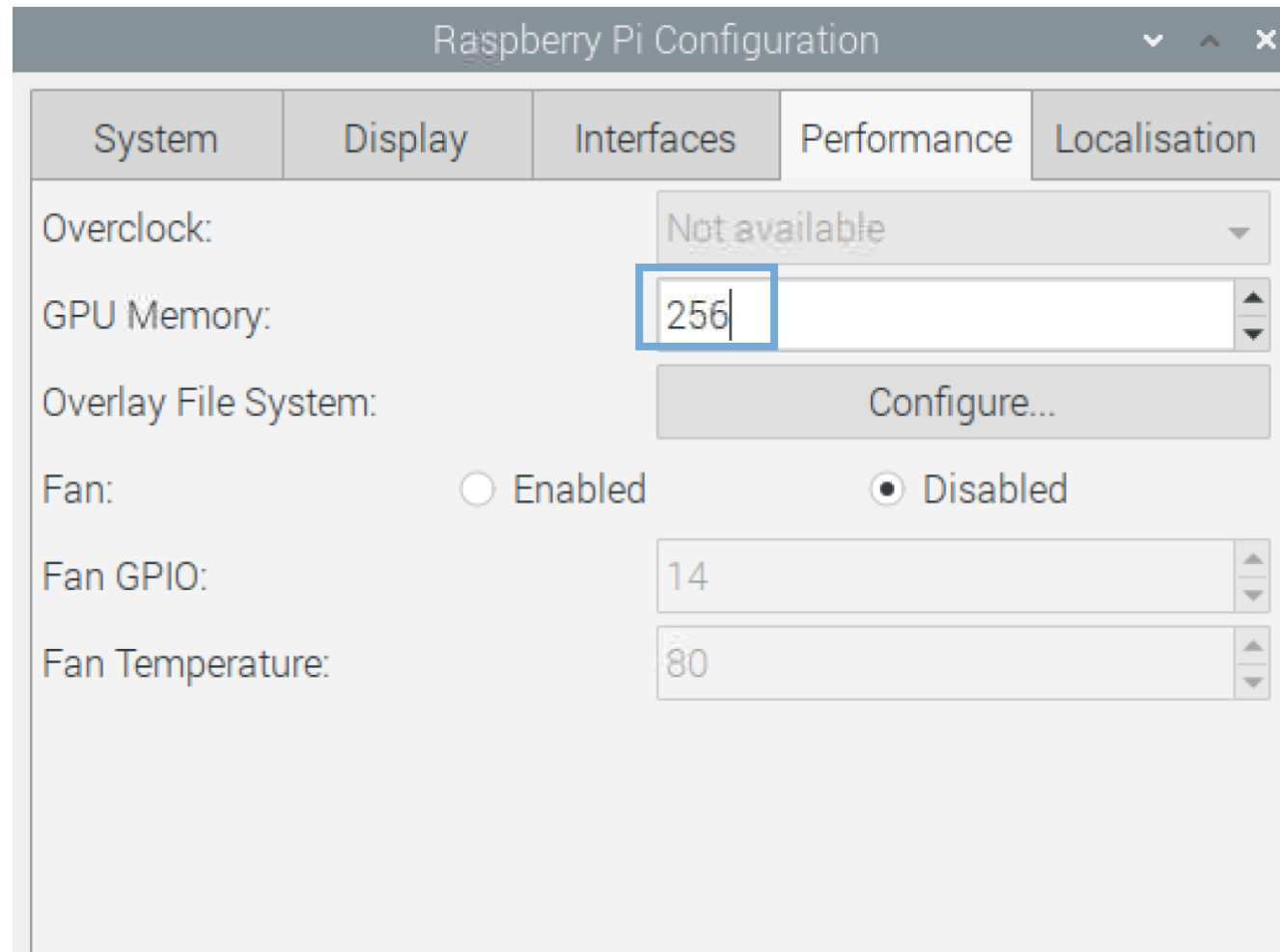
The screenshot shows the 'Raspberry Pi Configuration' window with the 'Interfaces' tab selected. The window has a title bar with a close button (x) and a maximize button (up arrow). The interface is divided into five tabs: System, Display, Interfaces, Performance, and Localisation. The 'Interfaces' tab is active, showing a list of interfaces with radio buttons for 'Enabled' and 'Disabled'. All 'Enabled' options are selected, and the 'Enabled' text is highlighted with a dashed box for the Camera option.

System	Display	Interfaces	Performance	Localisation
Camera:	<input checked="" type="radio"/> Enabled	<input type="radio"/> Disabled		
SSH:	<input checked="" type="radio"/> Enabled	<input type="radio"/> Disabled		
VNC:	<input checked="" type="radio"/> Enabled	<input type="radio"/> Disabled		
SPI:	<input checked="" type="radio"/> Enabled	<input type="radio"/> Disabled		
I2C:	<input checked="" type="radio"/> Enabled	<input type="radio"/> Disabled		
Serial Port:	<input checked="" type="radio"/> Enabled	<input type="radio"/> Disabled		
Serial Console:	<input checked="" type="radio"/> Enabled	<input type="radio"/> Disabled		
1-Wire:	<input checked="" type="radio"/> Enabled	<input type="radio"/> Disabled		
Remote GPIO:	<input checked="" type="radio"/> Enabled	<input type="radio"/> Disabled		

Install

VNC Viewer

GPU Memory를 256 정도로 올려주도록 하겠습니다.

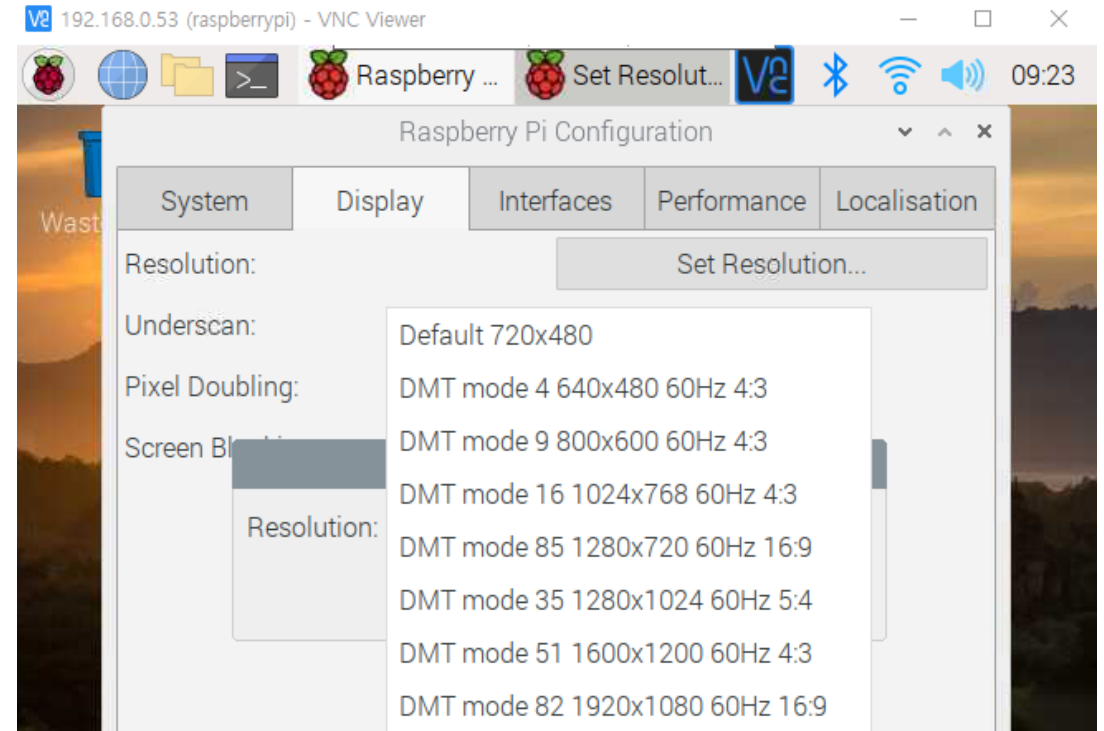
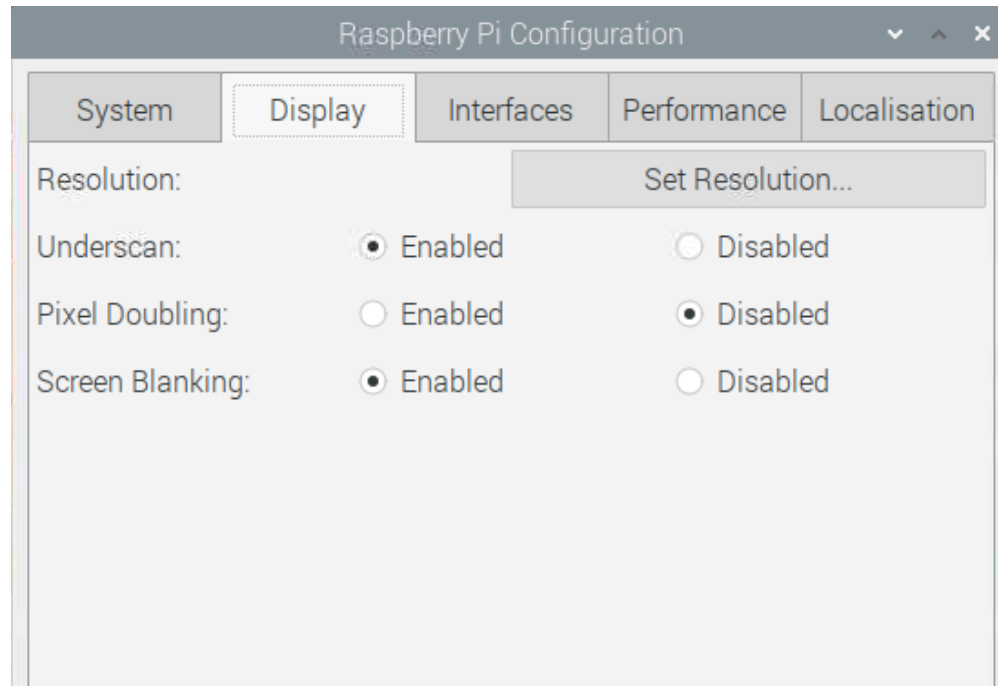


The screenshot shows the 'Raspberry Pi Configuration' window with the 'Performance' tab selected. The 'GPU Memory' field is highlighted with a blue box and contains the value '256'. Other settings visible include 'Overclock' (Not available), 'Overlay File System' (Configure...), 'Fan' (Disabled), 'Fan GPIO' (14), and 'Fan Temperature' (80).

System	Display	Interfaces	Performance	Localisation
Overclock:		Not available		
GPU Memory:		256		
Overlay File System:		Configure...		
Fan:		<input type="radio"/> Enabled <input checked="" type="radio"/> Disabled		
Fan GPIO:		14		
Fan Temperature:		80		

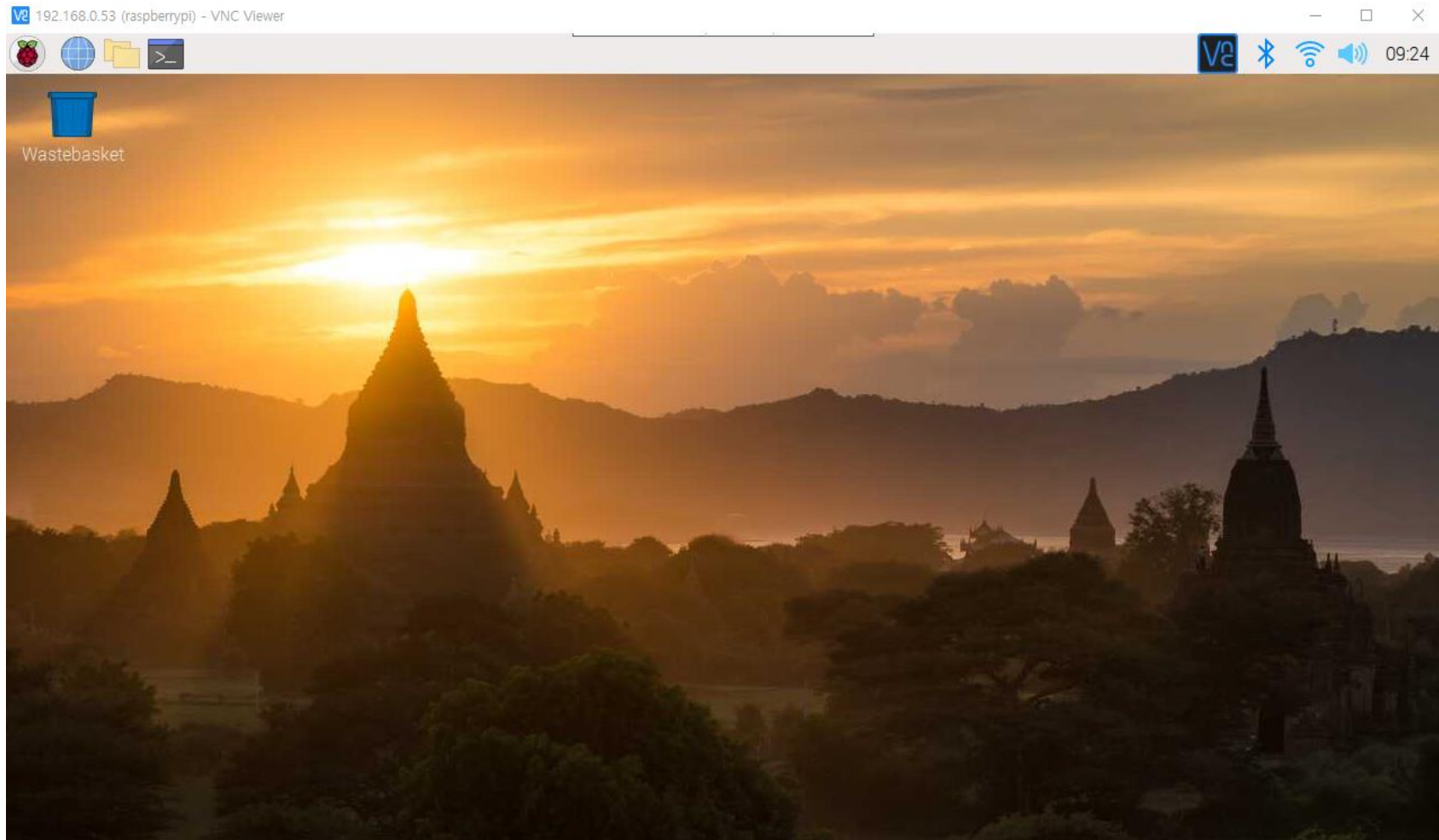
Install

VNC Viewer



Install

VNC Viewer



참고) 라즈베리파이 VNC Port: 5900

해당 프로젝트를 진행하면서 사용할 기본적인 리눅스 명령어만 알아보겠습니다.

sudo: Super User Do

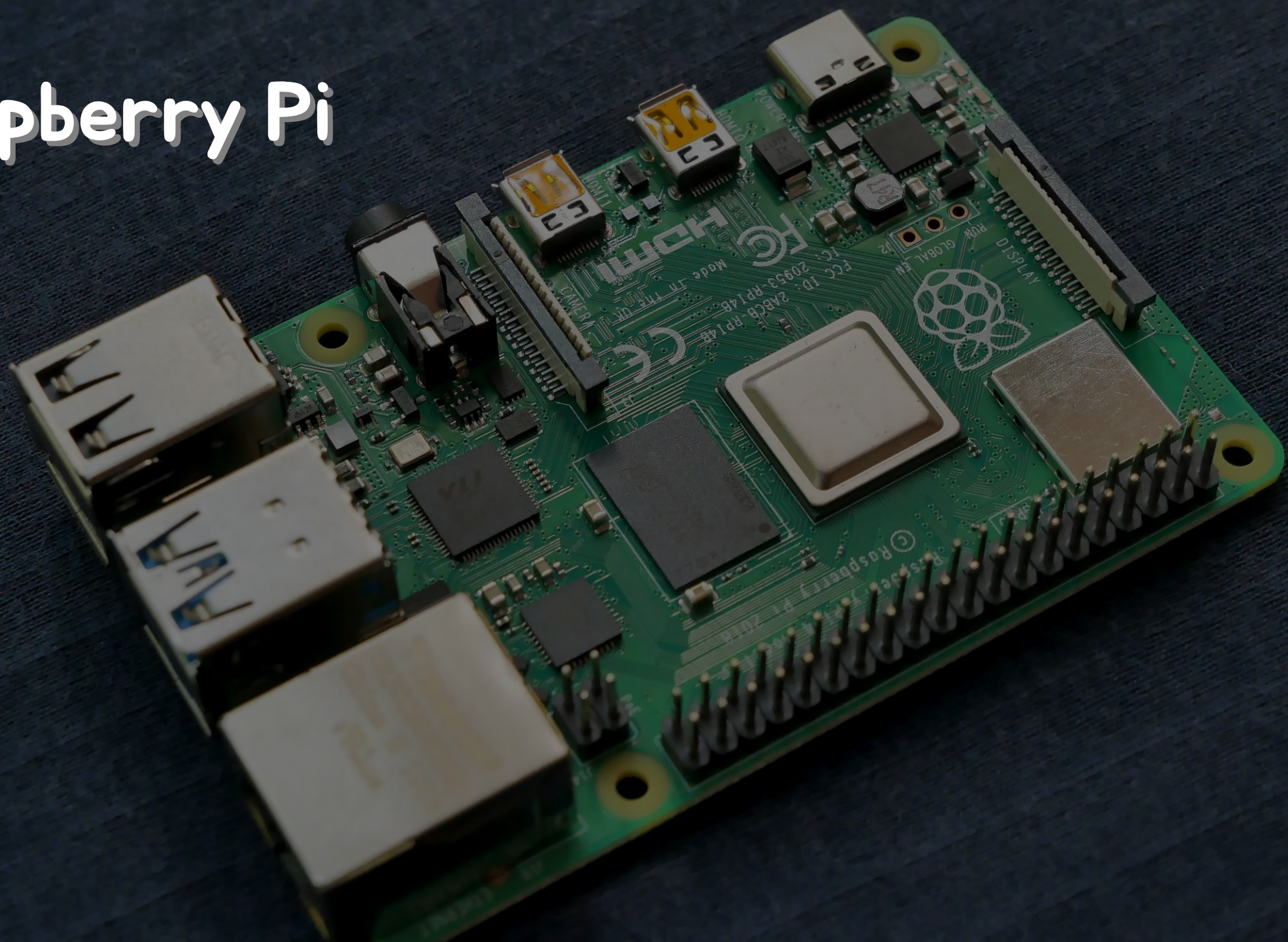
ls : List

cd : Change Directory

mkdir: Make Directory

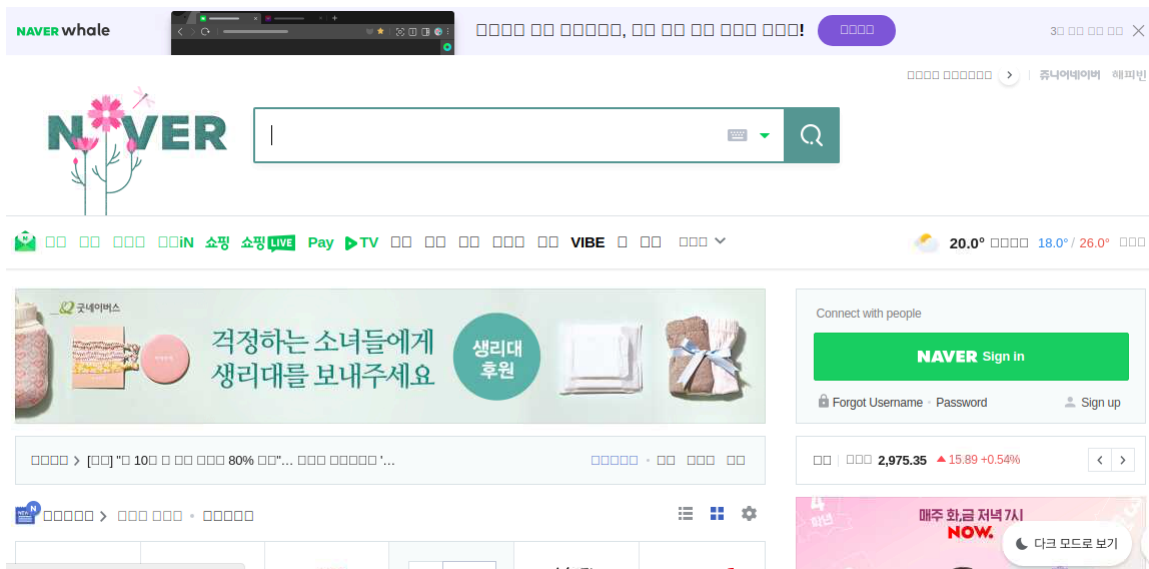
touch: Create File

Raspberry Pi



Raspberry Pi 3B +

한글화



초기에는 한글이 깨져서 나오는 것을 확인 가능

먼저, 터미널을 열고

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

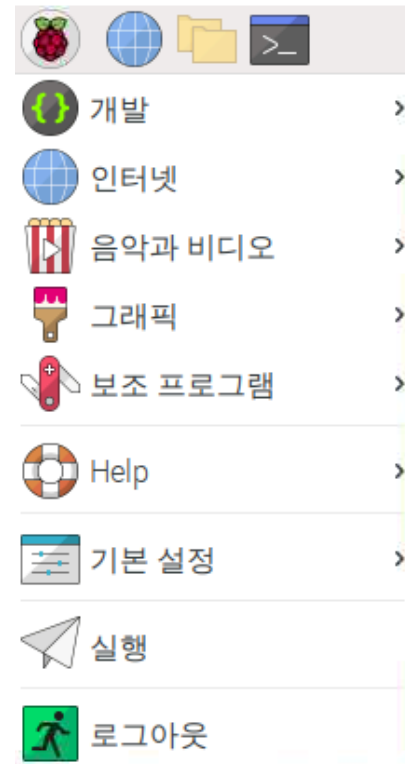
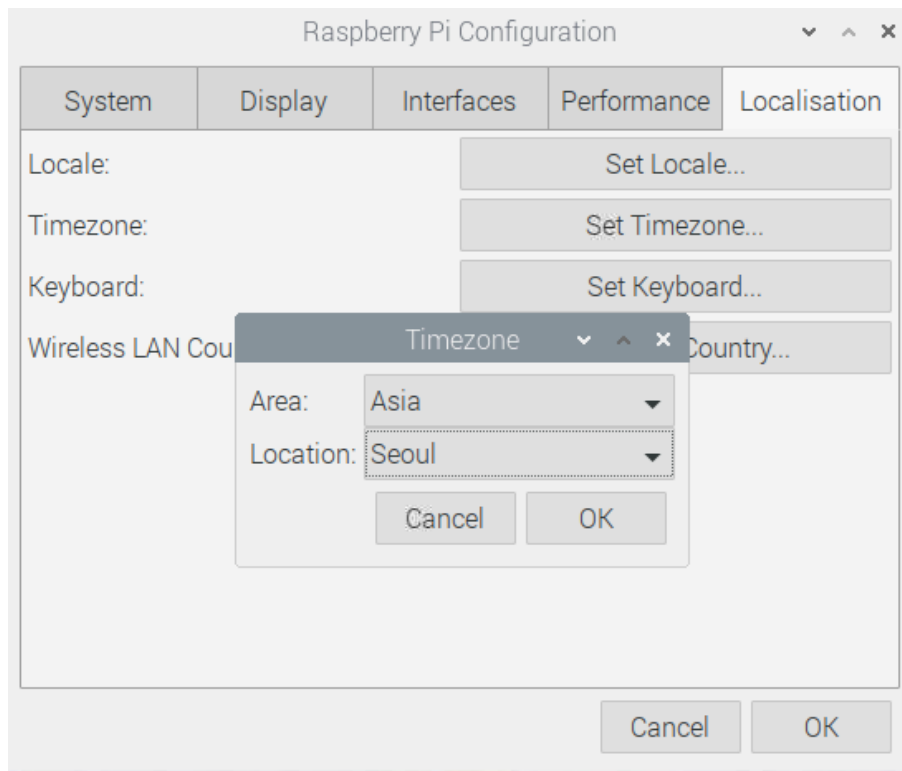
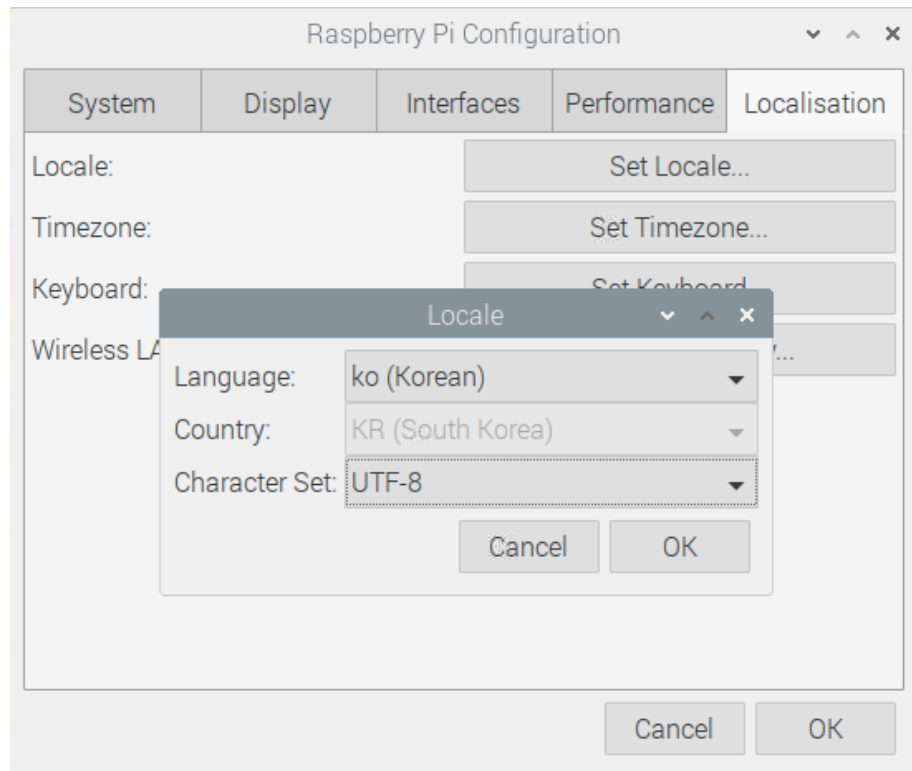
Apt 패키지를 업데이트 해줍니다.

Advanced Packaging Tool(apt)은 리눅스에서 쓰이는
패키지 관리 명령어 도구입니다.

```
sudo apt install fonts-unfonts-core -y
```

Raspberry Pi 3B +

한글화



이와 같이 한글로 나오는 것을 확인할 수 있습니다.

Raspberry Pi 3B +

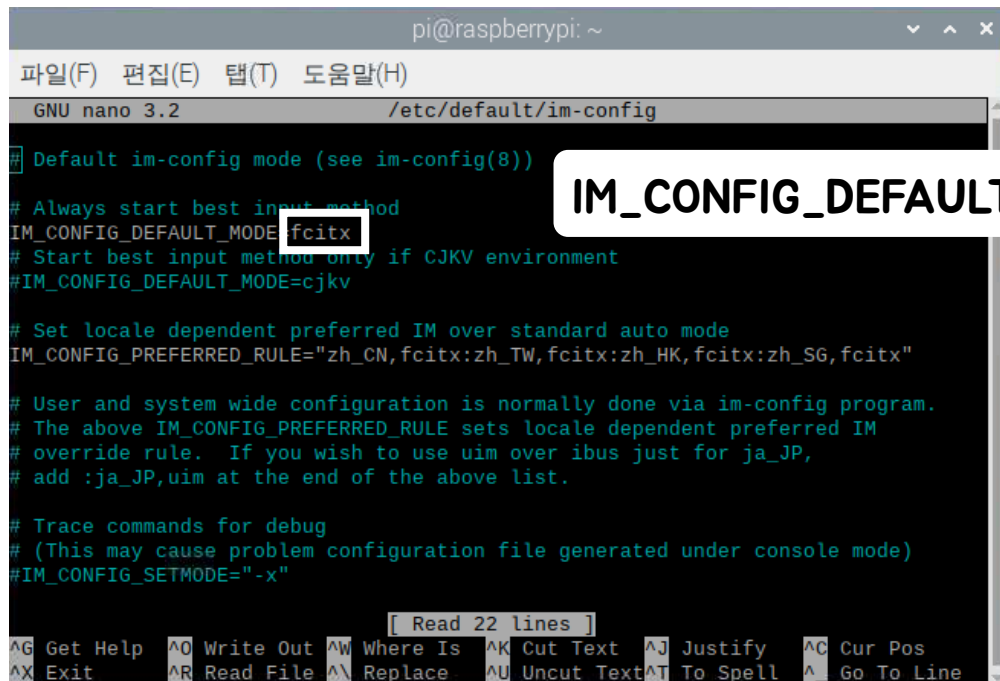
한글화

```
sudo apt remove ibus ibus-hangul  
sudo apt install fcitx fcitx-hangul -y  
sudo nano /etc/default/im-config
```

기존에 패키지를 삭제하고

필요한 패키지 설치

환경 설정을 위해 파일 OPEN



```
pi@raspberrypi: ~  
파일(F) 편집(E) 탭(T) 도움말(H)  
GNU nano 3.2 /etc/default/im-config  
# Default im-config mode (see im-config(8))  
# Always start best input method  
IM_CONFIG_DEFAULT_MODE=fcitx  
# Start best input method only if CJKV environment  
#IM_CONFIG_DEFAULT_MODE=cjvk  
# Set locale dependent preferred IM over standard auto mode  
IM_CONFIG_PREFERRED_RULE="zh_CN,fcitx:zh_TW,fcitx:zh_HK,fcitx:zh_SG,fcitx"  
# User and system wide configuration is normally done via im-config program.  
# The above IM_CONFIG_PREFERRED_RULE sets locale dependent preferred IM  
# override rule. If you wish to use uim over ibus just for ja_JP,  
# add :ja_JP,uim at the end of the above list.  
# Trace commands for debug  
# (This may cause problem configuration file generated under console mode)  
#IM_CONFIG_SETMODE="-x"  
[ Read 22 lines ]  
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos  
^X Exit ^R Read File ^S Replace ^U Uncut Text ^T To Spell ^_ Go To Line
```

IM_CONFIG_DEFAULT_MODE=fcitx

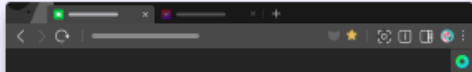
해당 부분 수정 후, `ctrl+x` → `Y` 엔터

터미널에 sudo reboot로 재실행

Raspberry Pi 3B +

한글화

NAVER whale




여러분의 눈은 소중한니까, 지금 바로 다크 모드를 켜세요!

다운로드


3일 동안 보지 않기 ✕

네이버를 시작페이지로 > | 즐겨찾기 | 네이버 | 해피빈



Q

메일 카페 블로그 지식iN 쇼핑 쇼핑LIVE Pay TV 사전 뉴스 증권 부동산 지도 VIBE 책 웹툰 더보기 ▾

 21.0° 흐림 18.0° / 26.0° 부승동

미디어오늘 > 방통위가 잡아낸 건강프로-흡소핑 연계편성, 3월에...

네이버뉴스 · 연예 스포츠 경제

뉴스스탠드 > 구독한 언론사 · 전체언론사

네이버

뉴스스탠드

구독한 언론사

전체언론사

☰

☐

⚙

데일리안	전자신문	스포츠동아	디지털타임스	일간스포츠	스포츠조선
뉴스리퍼	MBC	SBS	한국경제TV	KBS	한국일보
<div>></div>					

네이버를 더 안전하고 편리하게 이용하세요

NAVER 로그인

아이디 · 비밀번호찾기

회원가입

이슈

코로나바이러스감염증-19 현황

<

>

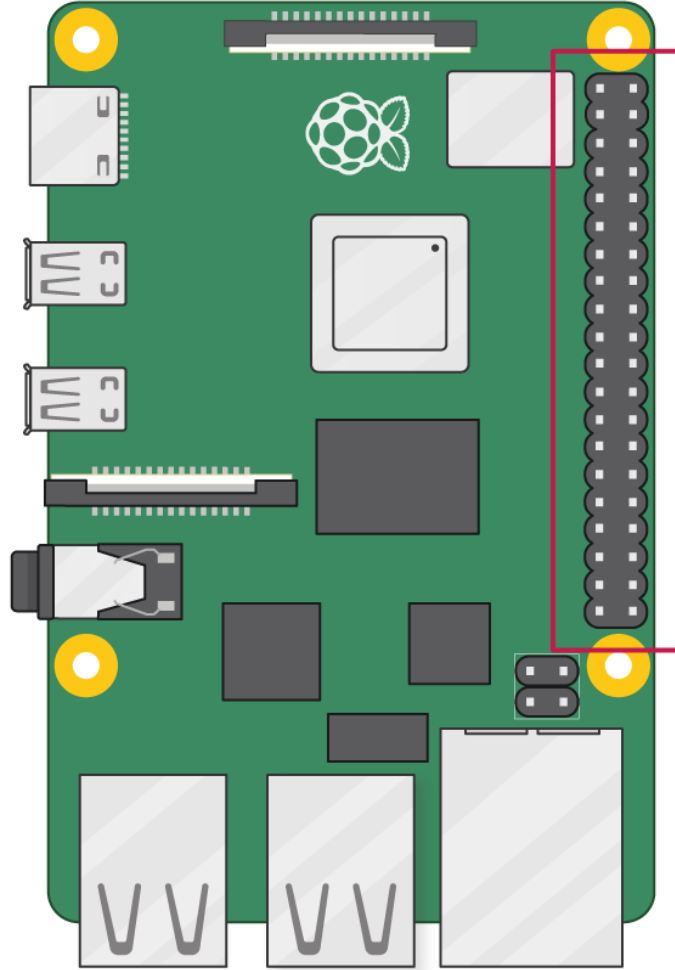
다크 모드로 보기

⌵

`ctrl + space`를 누르면 한/영키 변환이 가능

Raspberry Pi 3B +

GPIO(BCM) 핀맵



3V3 power	1	2	5V power
GPIO 2 (SDA)	3	4	5V power
GPIO 3 (SCL)	5	6	Ground
GPIO 4 (GPCLK0)	7	8	GPIO 14 (TXD)
Ground	9	10	GPIO 15 (RXD)
GPIO 17	11	12	GPIO 18 (PCM_CLK)
GPIO 27	13	14	Ground
GPIO 22	15	16	GPIO 23
3V3 power	17	18	GPIO 24
GPIO 10 (MOSI)	19	20	Ground
GPIO 9 (MISO)	21	22	GPIO 25
GPIO 11 (SCLK)	23	24	GPIO 8 (CE0)
Ground	25	26	GPIO 7 (CE1)
GPIO 0 (ID_SD)	27	28	GPIO 1 (ID_SC)
GPIO 5	29	30	Ground
GPIO 6	31	32	GPIO 12 (PWM0)
GPIO 13 (PWM1)	33	34	Ground
GPIO 19 (PCM_FS)	35	36	GPIO 16
GPIO 26	37	38	GPIO 20 (PCM_DIN)
Ground	39	40	GPIO 21 (PCM_DOUT)

GPIO 제어 with Python

LED ON / OFF

```
1  import RPi.GPIO as GPIO
2  import time
3
4  GPIO.setmode(GPIO.BCM)
5
6  PIN_LED = 26
7
8  GPIO.setup(PIN_LED, GPIO.OUT)
9
10 GPIO.output(PIN_LED, True)
11 time.sleep(5)
12 GPIO.output(PIN_LED, False)
13
14 GPIO.cleanup()
```

LED를 켜고 5초 후에 끄는 CODE

Import로 외부 라이브러리 사용 설정

- Rpi.GPIO 라즈베리 파이 GPIO제어 라이브러리 **as** → GPIO 별칭 설정
- time: 시간 관련 라이브러리

GPIO.setMode → BOARD와 BCM 설정

- BOARD: 1~40번
- BCM: GPIO 번호

LED 핀번호 GPIO 26 설정

GPIO.setup → 해당 핀 번호, 출력/입력 설정

GPIO.output → 해당 핀, 전기 + / - 설정

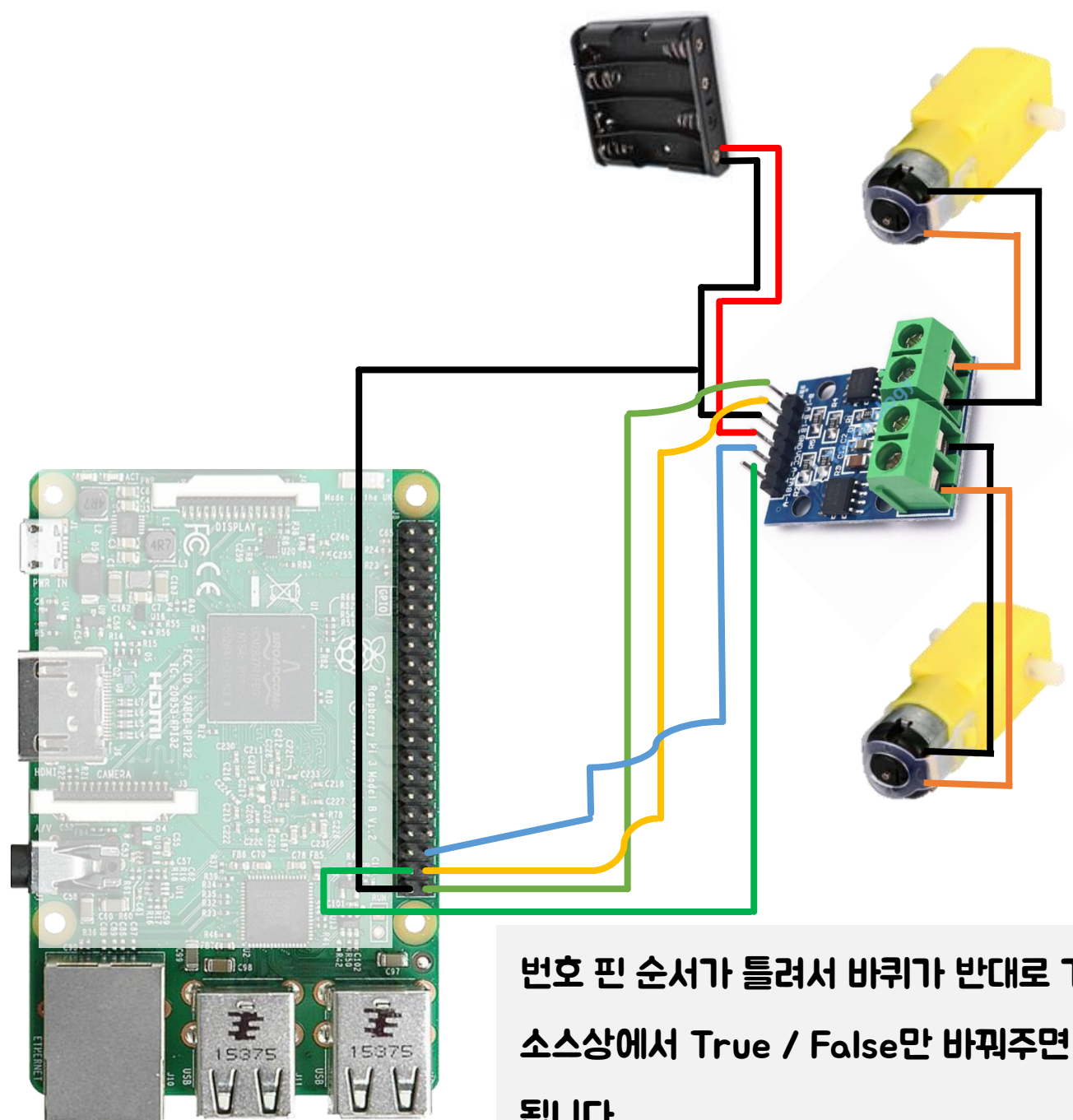
time.sleep(숫자) : 숫자 `초`만큼 멈추기

GPIO.cleanup() → GPIO 설정 핀 clean

GPIO 제어 with Python

DC Motor

```
1 import RPi.GPIO as GPIO
2 import time
3
4 GPIO.setmode(GPIO.BCM)
5
6 PIN_LEFT_MOTOR_FW = 20
7 PIN_LEFT_MOTOR_BW = 21
8 PIN_RIGHT_MOTOR_FW = 16
9 PIN_RIGHT_MOTOR_BW = 26
10
11 GPIO.setup(PIN_LEFT_MOTOR_FW, GPIO.OUT)
12 GPIO.setup(PIN_LEFT_MOTOR_BW, GPIO.OUT)
13 GPIO.setup(PIN_RIGHT_MOTOR_FW, GPIO.OUT)
14 GPIO.setup(PIN_RIGHT_MOTOR_BW, GPIO.OUT)
15
16 GPIO.output(PIN_LEFT_MOTOR_FW, True)
17 GPIO.output(PIN_LEFT_MOTOR_BW, False)
18 time.sleep(3)
19 GPIO.output(PIN_LEFT_MOTOR_FW, False)
20 GPIO.output(PIN_LEFT_MOTOR_BW, True)
21 time.sleep(3)
22 GPIO.output(PIN_LEFT_MOTOR_FW, False)
23 GPIO.output(PIN_LEFT_MOTOR_BW, False)
24 time.sleep(1)
25 GPIO.output(PIN_RIGHT_MOTOR_FW, True)
26 GPIO.output(PIN_RIGHT_MOTOR_BW, False)
27 time.sleep(3)
28 GPIO.output(PIN_RIGHT_MOTOR_FW, False)
29 GPIO.output(PIN_RIGHT_MOTOR_BW, True)
30 time.sleep(3)
31 GPIO.output(PIN_RIGHT_MOTOR_FW, False)
32 GPIO.output(PIN_RIGHT_MOTOR_BW, False)
33 time.sleep(1)
34
35 GPIO.cleanup()
```



번호 핀 순서가 틀려서 바퀴가 반대로 가면
소스상에서 True / False만 바꿔주면
됩니다.

GPIO 제어 with Python

DC Motor

```
1 import RPi.GPIO as GPIO
2 import time
3
4 GPIO.setmode(GPIO.BCM)
5
6 PIN_LEFT_MOTOR_FW = 5
7 PIN_LEFT_MOTOR_BW = 6
8 PIN_RIGHT_MOTOR_FW = 22
9 PIN_RIGHT_MOTOR_BW = 23
10
11 GPIO.setup(PIN_LEFT_MOTOR_FW, GPIO.OUT)
12 GPIO.setup(PIN_LEFT_MOTOR_BW, GPIO.OUT)
13 GPIO.setup(PIN_RIGHT_MOTOR_FW, GPIO.OUT)
14 GPIO.setup(PIN_RIGHT_MOTOR_BW, GPIO.OUT)
15
16 def motor_ctrl(is_left_front, is_right_front, is_left_stop, is_right_stop):
17     if is_left_stop:
18         GPIO.output(PIN_LEFT_MOTOR_FW, False)
19         GPIO.output(PIN_LEFT_MOTOR_BW, False)
20     else:
21         GPIO.output(PIN_LEFT_MOTOR_FW, is_left_front)
22         GPIO.output(PIN_LEFT_MOTOR_BW, not is_left_front)
23
24     if is_right_stop:
25         GPIO.output(PIN_RIGHT_MOTOR_FW, False)
26         GPIO.output(PIN_RIGHT_MOTOR_BW, False)
27     else:
28         GPIO.output(PIN_RIGHT_MOTOR_FW, is_right_front)
29         GPIO.output(PIN_RIGHT_MOTOR_BW, not is_right_front)
30
31 motor_ctrl(True, False, False, True)
32 time.sleep(3)
33 motor_ctrl(False, False, False, True)
34 time.sleep(3)
35 motor_ctrl(False, False, True, True)
36 time.sleep(1)
37 motor_ctrl(False, True, True, False)
38 time.sleep(3)
39 motor_ctrl(False, False, True, False)
40 time.sleep(3)
41 motor_ctrl(False, False, True, True)
42 time.sleep(1)
43
44 GPIO.cleanup()
```

Python에 함수 문법을 이용하여
제어를 편하게 사용하기 위해 변경된 CODE

Python에서 함수는 `def` 로 표현

함수 참조 링크

- <https://github.com/EduProgramming/Python/blob/develop/15def.ipynb>

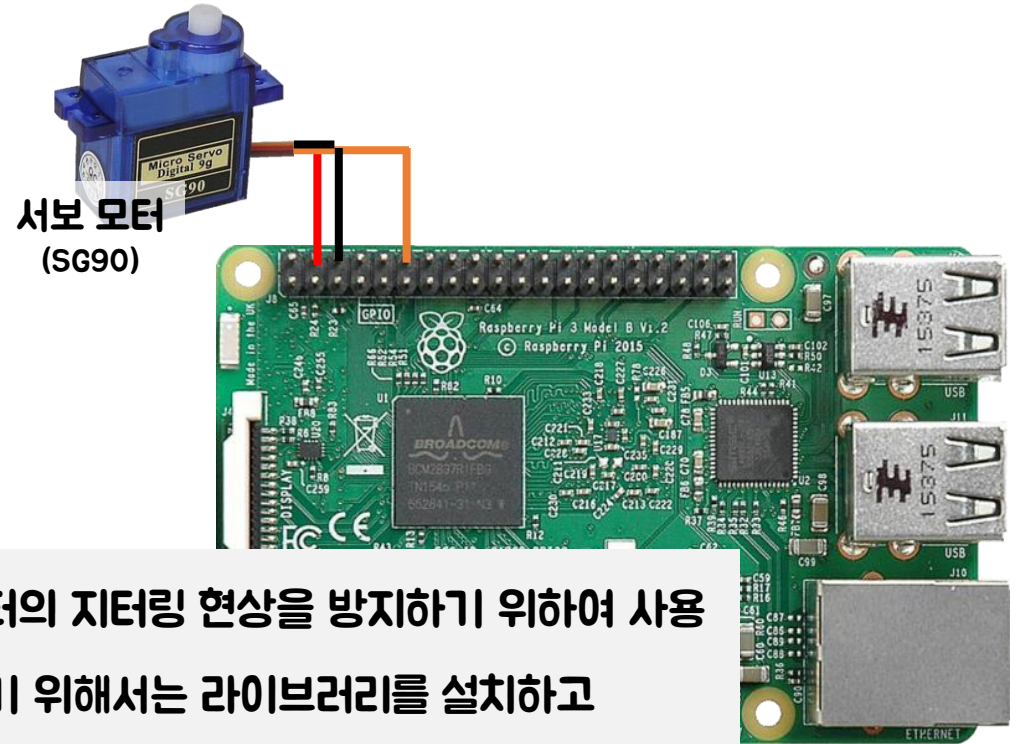
- <https://wikidocs.net/24>

GPIO 제어 with Python

Servo

```
sudo apt-get install pigpio python-pigpio python3-pigpio  
sudo pigpiod
```

```
1 import pigpio  
2 import time  
3  
4 PIN_SERVO = 18  
5  
6 servo = pigpio.pi()  
7 servo.set_servo_pulsewidth(PIN_SERVO, 1500)  
8 time.sleep(1)  
9 servo.set_servo_pulsewidth(PIN_SERVO, 2400)  
10 time.sleep(1)  
11 servo.set_servo_pulsewidth(PIN_SERVO, 600)  
12 time.sleep(1)  
13 servo.set_servo_pulsewidth(PIN_SERVO, 1500)  
14 time.sleep(1)
```



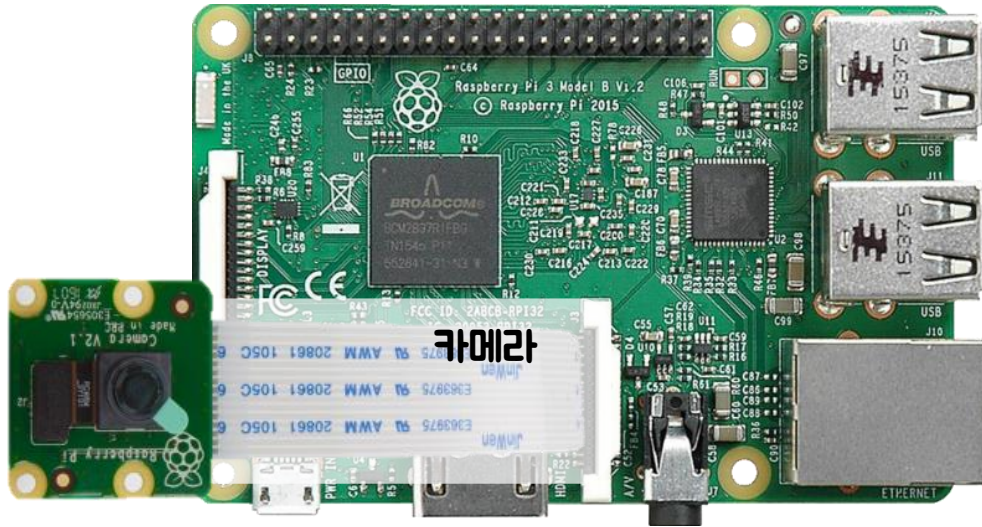
pigpiod는 서보모터의 지터링 현상을 방지하기 위하여 사용
pigpiod를 사용하기 위해서는 라이브러리를 설치하고
pigpiod 서버를 실행하면 됩니다.

Pulse 범위: 500 ~ 2500

참고 링크: <http://abyz.me.uk/rpi/pigpio/python.html>

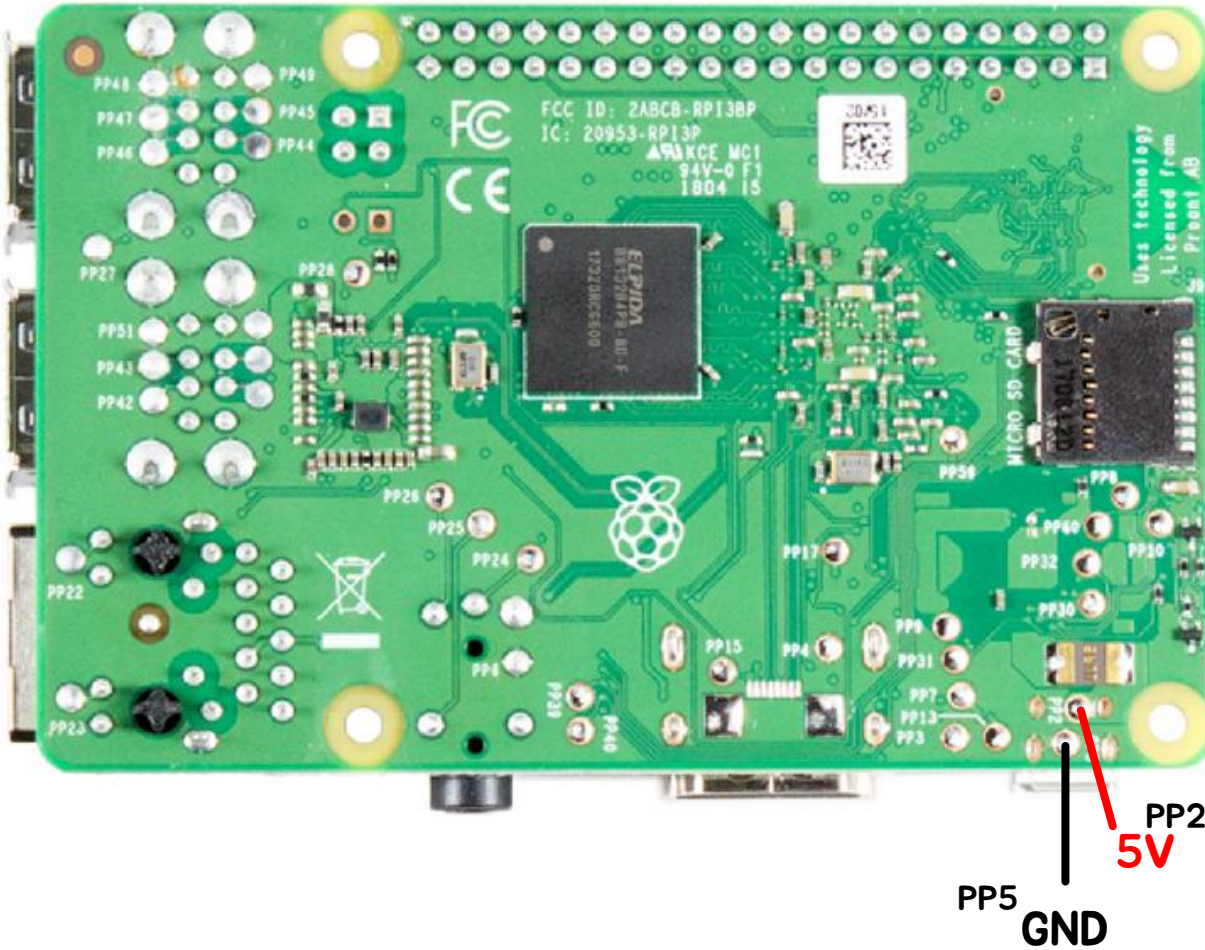
Raspberry Pi 3B +

Camera Module



외부 전력 공급 방법

PP2 / PP5

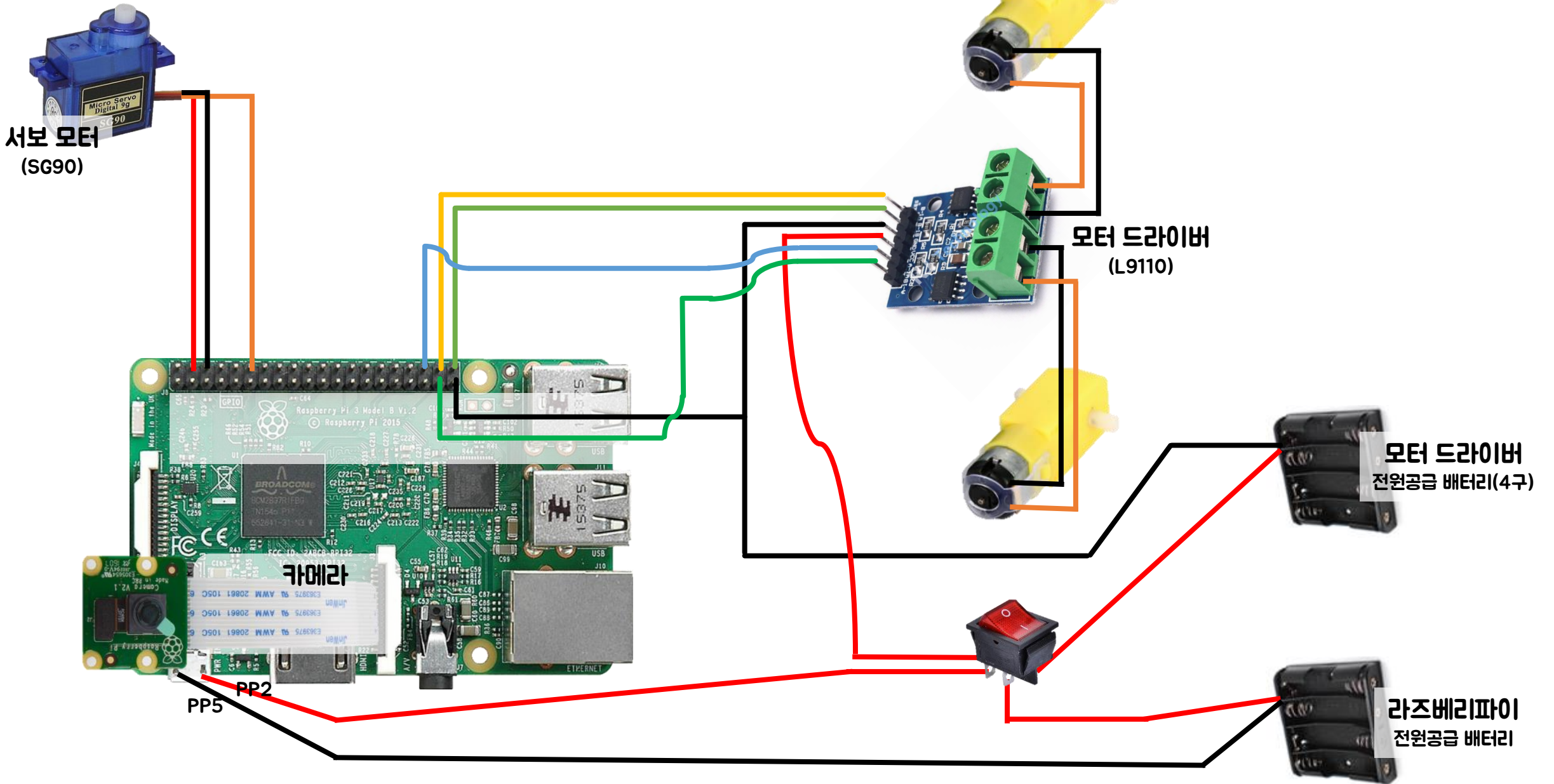


PP2에 **5V** 전원단

PP5에 GND

연결하여 외부 전원으로 작동 시킬 수 있습니다.

전체 회로도



CODE

Flask

파이썬 웹 프레임워크 중 하나로서, 웹서버 역할을 합니다.

프레임워크: 뼈대 역할을 하는 것(간단히, 클래스 + 라이브러리 형태)

```
from flask import Flask

app = Flask(__name__)

@app.route("/")
def hello():
    return "Hello, World!"

if __name__ == '__main__':
    app.run(host='0.0.0.0', threaded=True)
```

라즈베리파이 IP:5000 브라우저에 주소를 치면

localhost:5000도 VS Code 진행시 가능했음

Hello, World!

이렇게 나오는 웹 페이지가 나옵니다.

참고링크: <https://flask.palletsprojects.com/en/2.0.x/>

Flask

flask의 render_template을 이용하여 다른 HTML을 열게 설정

```
from flask import Flask, render_template

app = Flask(__name__)

@app.route('/')
def index():
    return render_template('index.html')

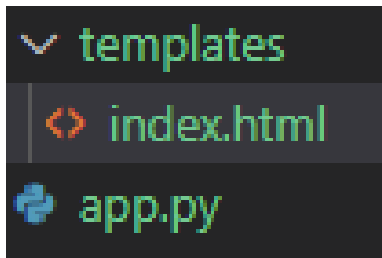
if __name__ == '__main__':
    app.run()
```

HTML CODE

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible"
content="IE=edge">
    <meta name="viewport"
content="width=device-width, initial-
scale=1.0">
    <title>Document</title>
</head>
<body>
    Hello, RaspberryPi
</body>
</html>
```

Flask

여기서 주의할 점은 HTML 파일의 위치인데 Flask는 기본적으로
`templates` 폴더를 기반으로 html파일을 찾습니다.



App.py 파일이 있는 곳에 `templates` 폴더가 있으며,
해당 폴더 안에는 index.html 파일이 있습니다.

`sudo python app.py` 터미널에 작성시에 서버가 실행됩니다.

127.0.0.1:5000으로 이동하게 되면 아래와 같이 나오게 됩니다.

← → ↻ ⓘ 127.0.0.1:5000

Hello, RaspberryPi

해당 주소는 서버를 켜진 local에서만 사용 가능
이외의 요소에서는 접근이 불가능합니다.