스마트 모빌리티 프로그래밍

Ch 14. 스마트 모빌리티 플랫폼 – Raspberry Pi, Jetson Developer Kit (Nano/TX2/Xavier)

정보통신공학과

교수 김 영 탁

(Tel : +82-53-810-2497; E-mail : ytkim@yu.ac.kr)

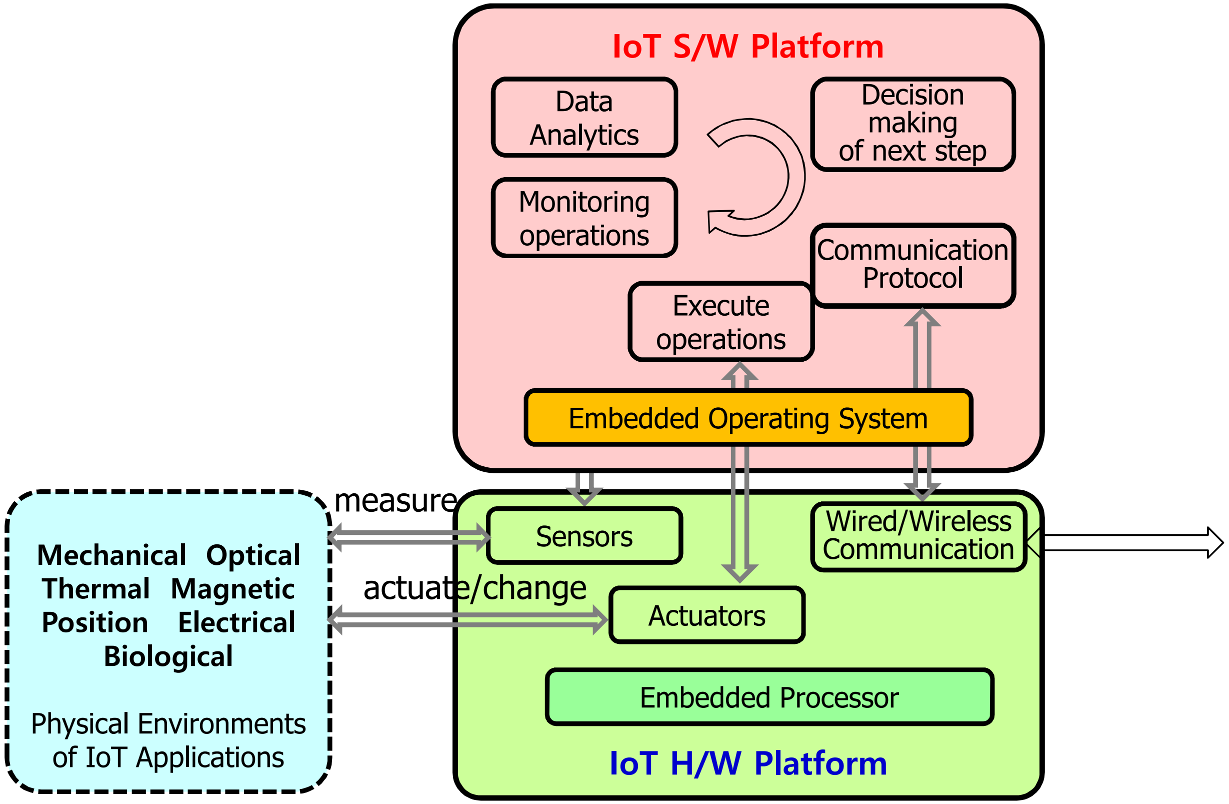
Outline

- ◆ 스마트 모빌리티 플랫폼
- ◆ Raspberry Pi
- ◆ Raspberry Pi OS (Raspbian) 설치 및 초기화
- ◆ mobaxterm 설치 및 원격 접속
- ◆ Raspberry Pi OS 환경에서의 Python 설치 확인
- ◆ Raspberry Pi 환경에서의 Python 프로그래밍 예제
- ◆ Jetson Developer Kit (Nano/TX2/Xavier)

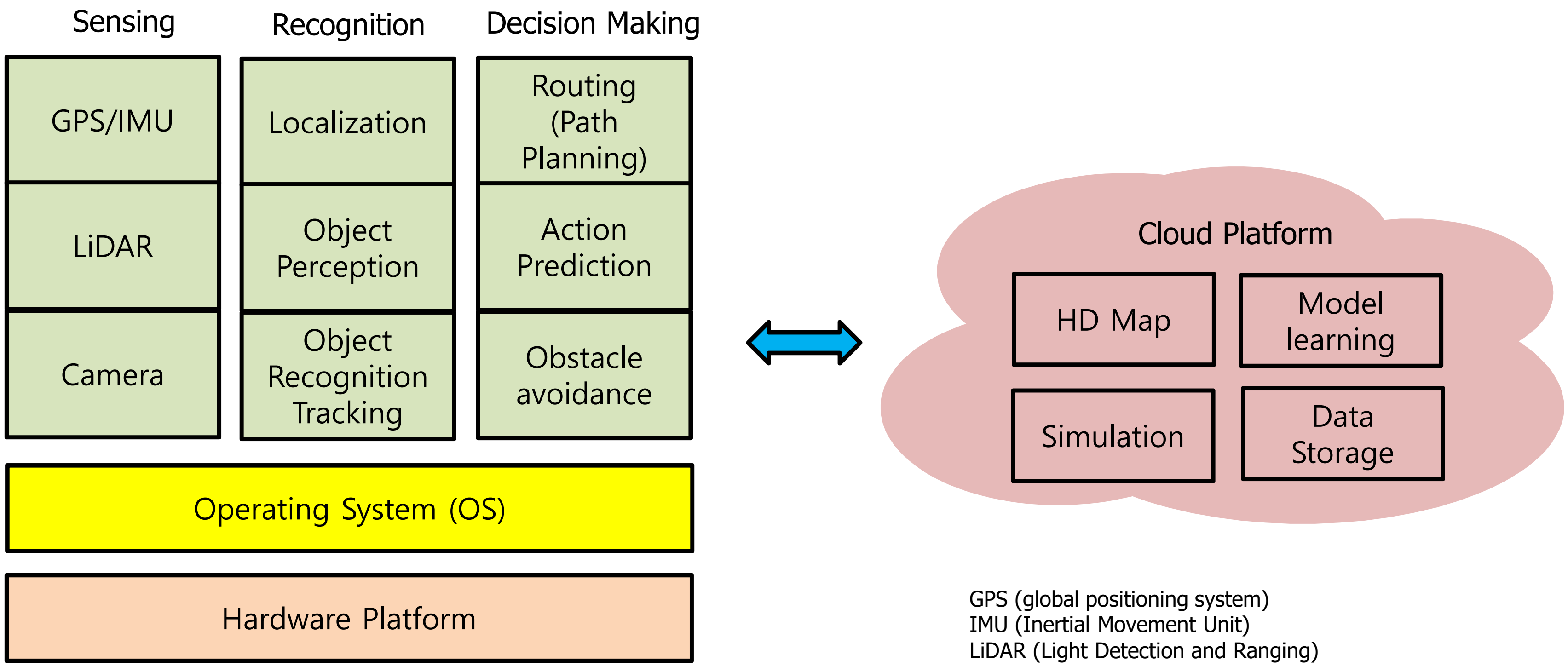
스마트 모빌리티 플랫폼의 기능 구조

스마트 모빌리티 플랫폼 / 사물인터넷 단말장치 기능 구조

◆ 스마트 모빌리티 플랫폼 / 사물인터넷 단말장치 기능 구조



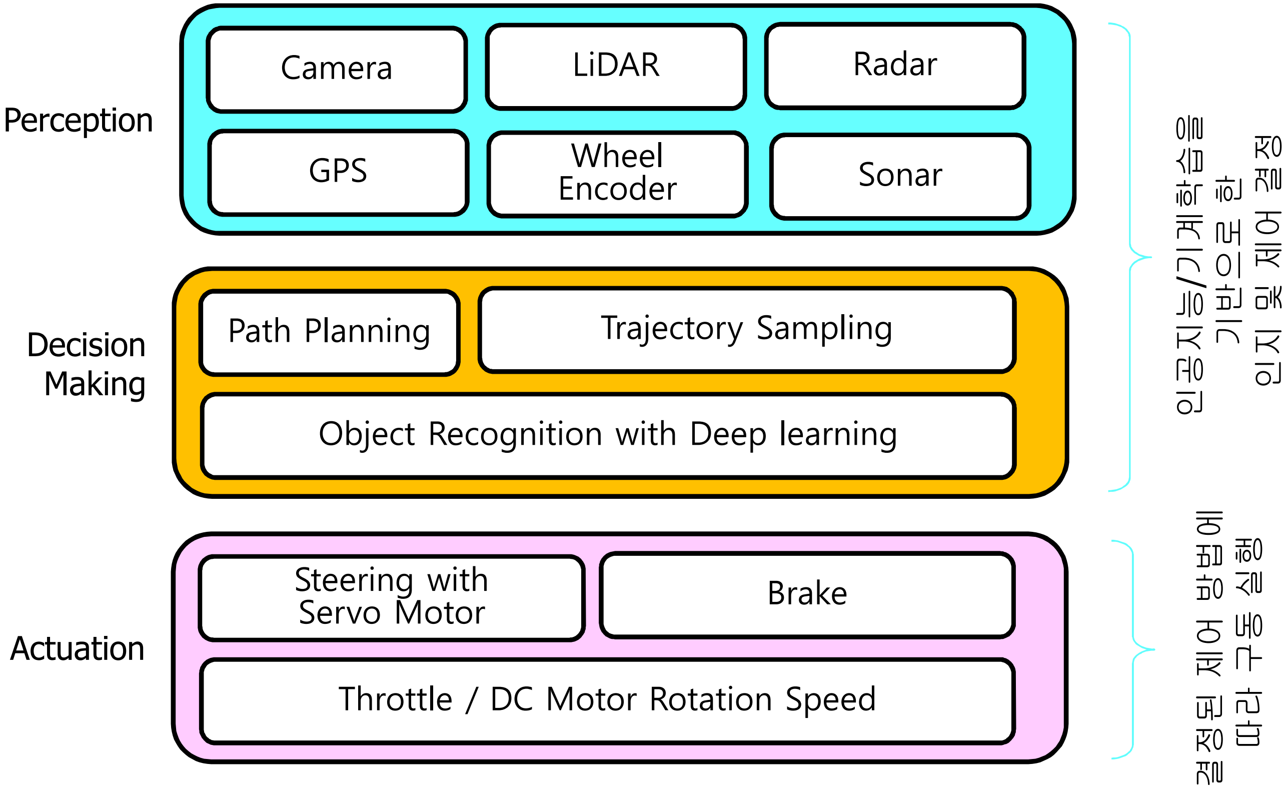
자율주행 자동차의 기본 기능 구조



GPS (global positioning system)
IMU (Inertial Movement Unit)
LiDAR (Light Detection and Ranging)

자율자동차의 인지, 제어 결정, 구동 제어

◆ 자율자동차의 개념적 기능 구분



Perception

Camera

LiDAR

Radar

GPS

Wheel Encoder

Sonar

Decision Making

Path Planning

Trajectory Sampling

Object Recognition with Deep learning

Actuation

Steering with Servo Motor

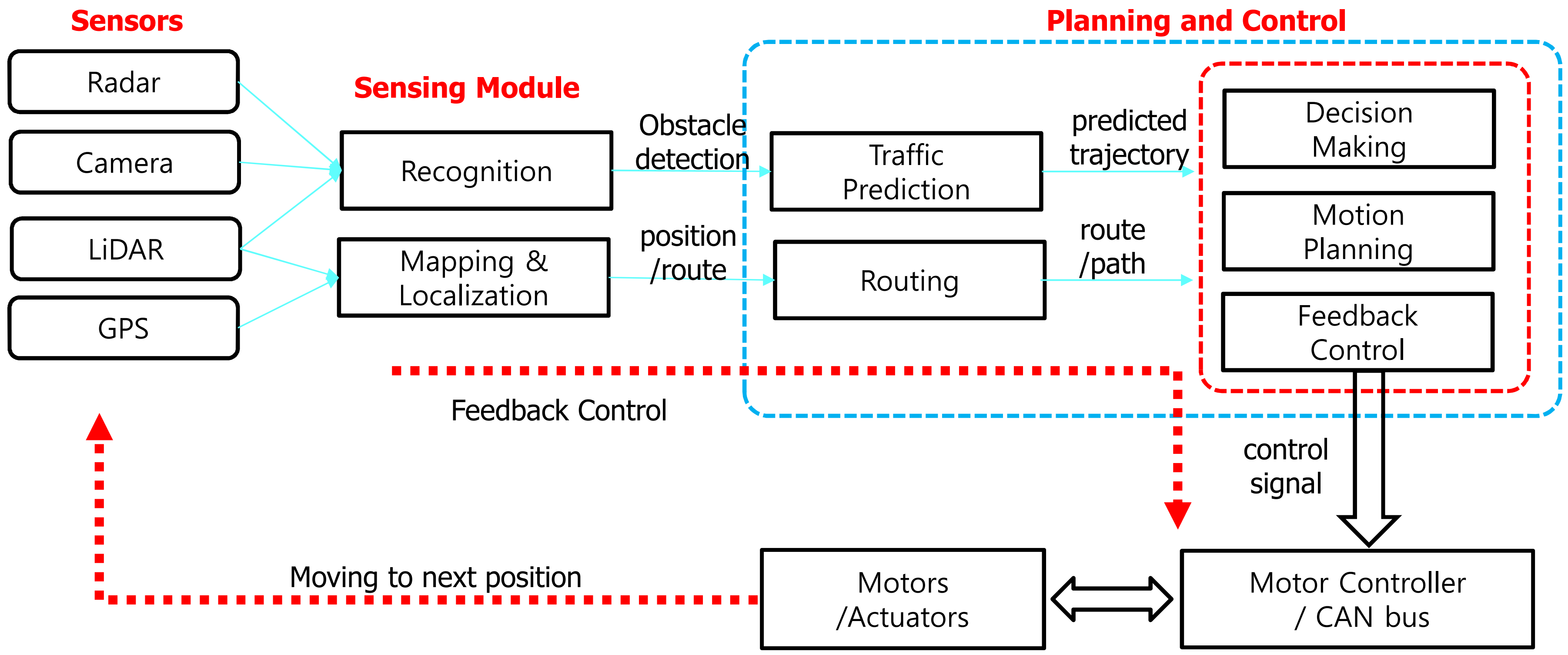
Brake

Throttle / DC Motor Rotation Speed

인공지능/기계학습을 기반으로 한 인지 및 제어 결정

결정된 제어 방법에 따라 구동 실행

Feedback Control in Autonomous Driving



스마트 모빌리티 / 사물인터넷 단말장치 플랫폼

◆ Raspberry Pi

- 컴퓨터 교육용으로 개발된 single board computer
- Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
- 2GB, 4GB or 8GB LPDDR4-3200 SDRAM (depending on model)
- 2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE
- Gigabit Ethernet
- 2 x USB 3.0 ports; 2 x USB 2.0 ports.
- Raspberry Pi standard 40 pin GPIO header (fully backwards compatible with previous boards)
- Raspberry Pi OS (Raspbian)

◆ Jetson Nano / Xavier

- GPU (Graphics Processing Unit)이 포함된 single board computer
- Linux Operating System (Ubuntu)
- 인공지능 / Deep learning이 포함된 기능 구현에 적합

Raspberry Pi

Raspberry Pi

◆ Overview of Raspberry Pi (1)

- https://www.raspberrypi.com/products/raspberry-pi-4-model-b/
- The Raspberry Pi is a series of credit card-sized single-board computers developed in the United Kingdom by the Raspberry Pi Foundation to promote the teaching of basic computer science in schools and developing countries.
- Several generations of Raspberry Pis have been released. The first generation (Raspberry Pi 1 Model B) was released in February 2012. It was followed by a simpler and inexpensive model Model A. In 2014 the foundation released a board with an improved design in Raspberry Pi 1 Model B+. The model laid the current "mainline" form-factor. Improved A+ and B+ models were released a year later. A cut down "compute" model was released in April 2014, and a Raspberry Pi Zero with smaller size and limited input/output (I/O), general-purpose input/output (GPIO), abilities released in November 2015 for US$5.
- The Raspberry Pi 2 which added more RAM was released in February 2015.
- Raspberry Pi 3 Model B released in February 2016 is bundled with on-board WiFi and Bluetooth. As of 2016, Raspberry Pi 3 Model B is the newest mainline Raspberry Pi. These boards are priced between US $20-35.
- Recently, faster and more powerful **Raspberry Pi 4 Model B** has been released with Broadcom BCM2711 quad-core Cortex-A72 64-bit SoC@1.5GHz, 2GB ~ 8GB SDRAM, USB C power supply connection, Gigabit Ethernet, USB 3, and 2 micro HDMI interfaces for easy dual display connection.
  Costs from US $35.

Raspberry Pi 4 Model B

◆ Overview of Raspberry Pi (2)

- All models feature a Broadcom system on a chip (SoC), which includes an ARM compatible central processing unit (CPU) and an on chip graphics processing unit (GPU, a VideoCore IV).
- Raspberry Pi 4 with Broadcom BCM2711 quad-core Cortex-A72 64-bit SoC@1.5GHz, 2GB ~ 8GB SDRAM
- Secure Digital (SD) cards are used to store the operating system and program memory in either the SDHC or MicroSDHC sizes. Most boards have between one and four USB slots, HDMI and composite video output, and a 3.5 mm phone jack for audio. Lower level output is provided by a number of GPIO pins which support common protocols like I^{2}C.
- Raspberry Pi 4 B-models have an Gigabit Ethernet port and on-board Wi-Fi 802.11ac and Bluetooth 5.0.
- Power over Ethernet (PoE) enabled (requires separate PoE HAT)
- The Foundation provides Raspberry Pi OS (previously called Raspbian), a Debian-based Linux distribution for download, as well as third party Ubuntu, Windows 10 IOT Core, RISC OS, and specialized media center distributions.
- Raspberry Pi OS can be easily installed to microSD card with Raspberry Pi Imager
- It promotes Python and Scratch as the main programming language, with support for many other languages. The default firmware is closed source, while an unofficial open source is available.

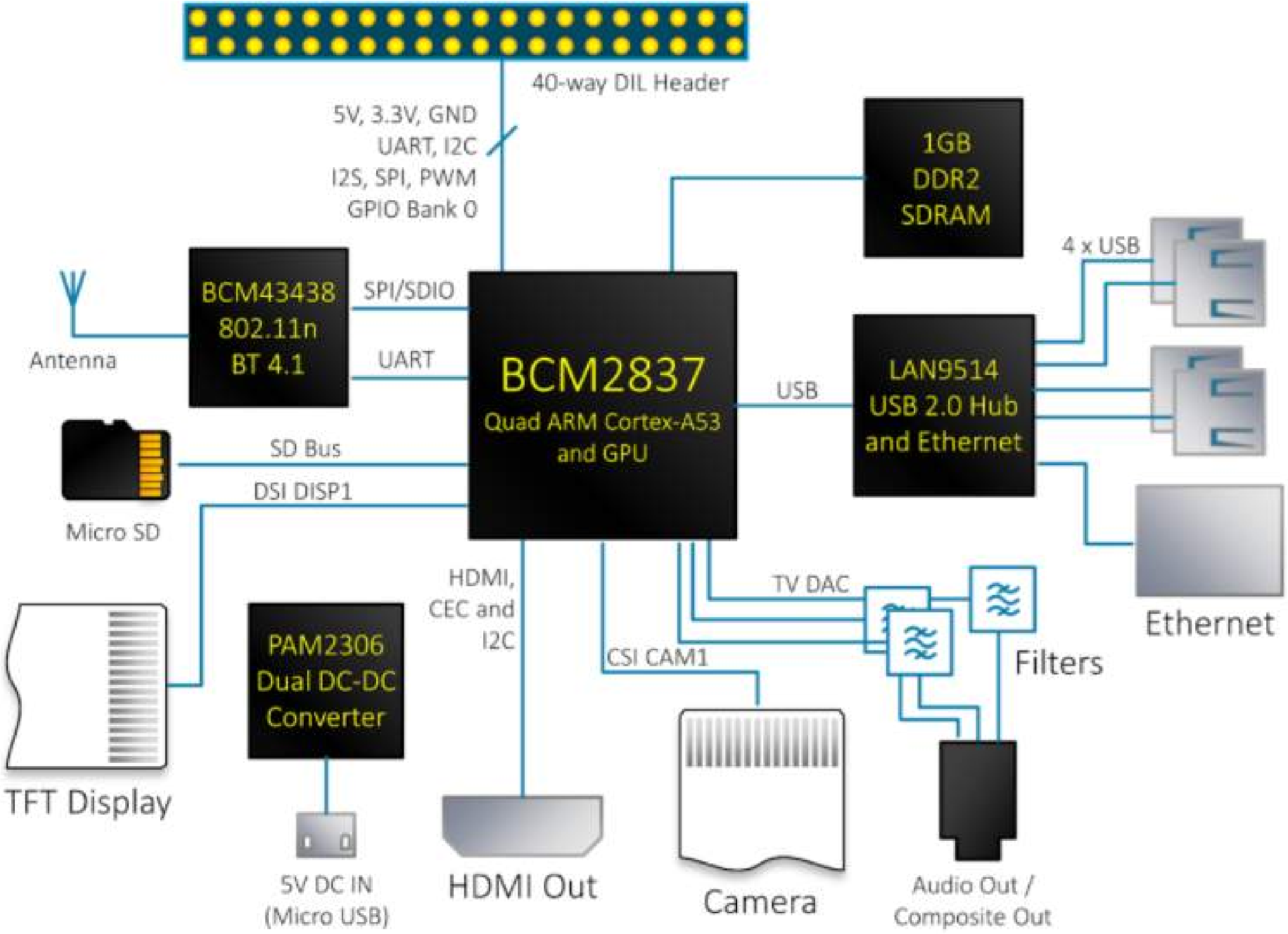
Raspberry Pi 4 Model B

◆ Technical Specifications

- Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
- 2GB, 4GB or 8GB LPDDR4-3200 SDRAM (depending on model)
- 2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE
- Gigabit Ethernet
- 2 USB 3.0 ports; 2 USB 2.0 ports.
- Raspberry Pi standard 40 pin GPIO header (fully backwards compatible with previous boards)
- 2 × micro-HDMI ports (up to 4kp60 supported)
- 2-lane MIPI DSI display port
- 2-lane MIPI CSI camera port
- 4-pole stereo audio and composite video port
- H.265 (4kp60 decode), H264 (1080p60 decode, 1080p30 encode)
- OpenGL ES 3.1, Vulkan 1.0
- Micro-SD card slot for loading operating system and data storage
- 5V DC via USB-C connector (minimum 3A*)
- 5V DC via GPIO header (minimum 3A*)
- Power over Ethernet (PoE) enabled (requires separate PoE HAT)
- Operating temperature: 0 – 50 degrees C ambient

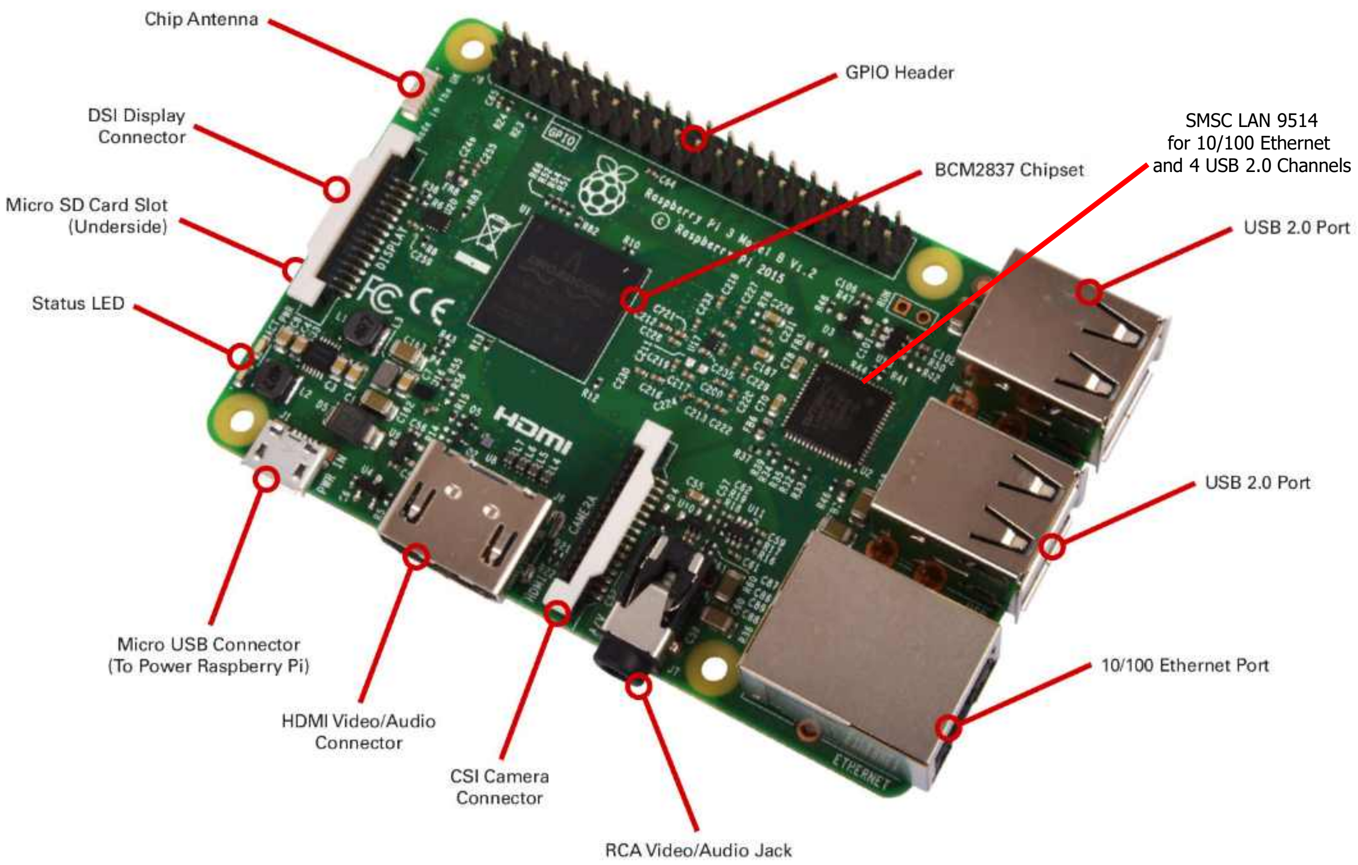
(Source: https://www.raspberrypi.com/products/raspberry-pi-4-model-b/specifications/)

Raspberry Pi 3 Functional Block Diagram



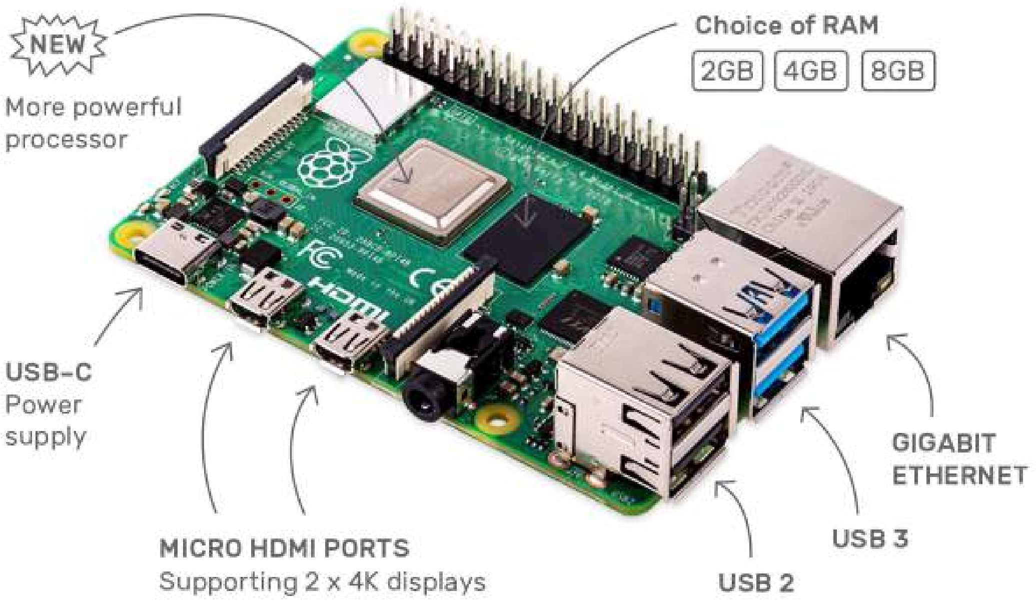
Top View

◆ Top view of Raspberry Pi 3 Model B

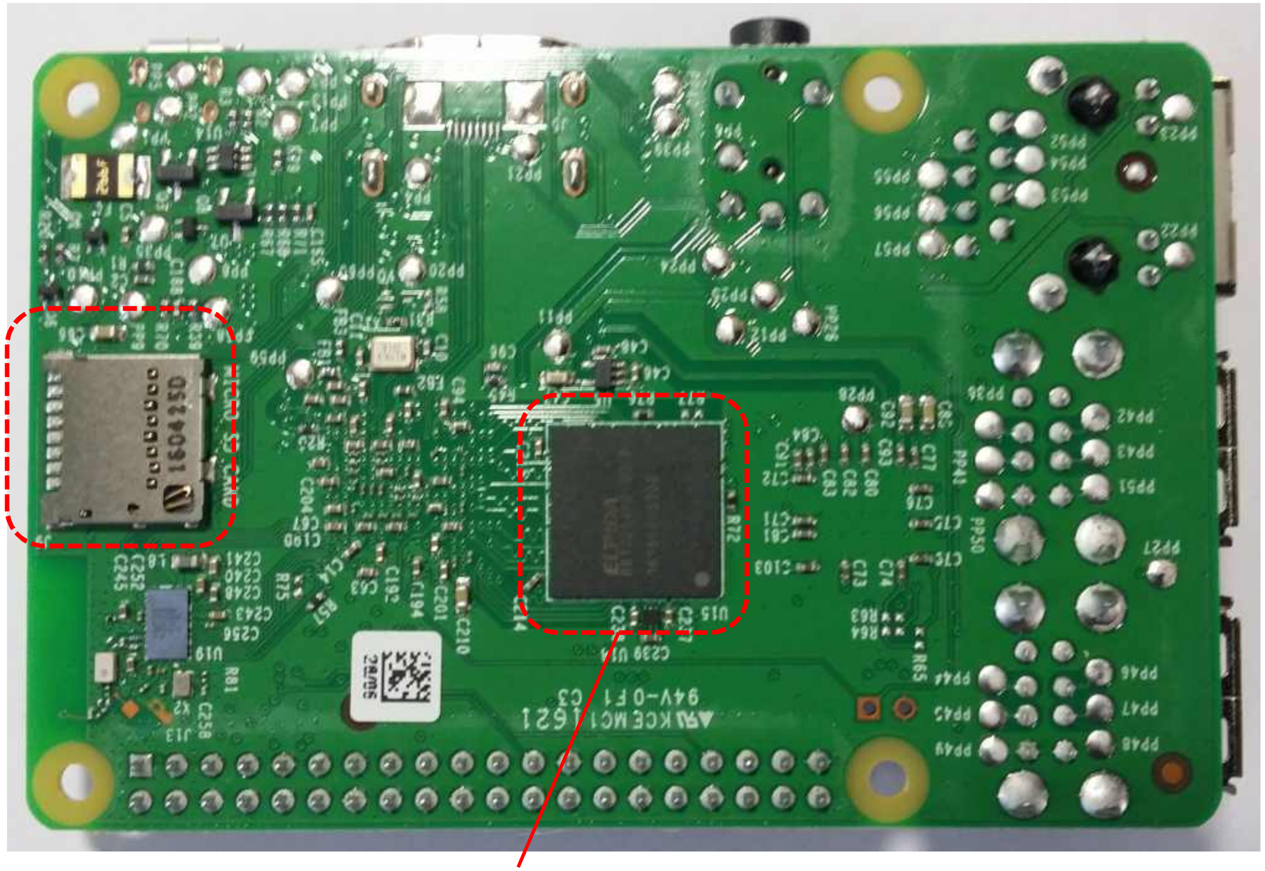

Chip Antenna
GPIO Header
DSI Display Connector
BCM2837 Chipset
SMSC LAN 9514 for 10/100 Ethernet and 4 USB 2.0 Channels
Micro SD Card Slot (Underside)
USB 2.0 Port
Status LED
USB 2.0 Port
Micro USB Connector (To Power Raspberry Pi)
10/100 Ethernet Port
HDMI Video/Audio Connector
CSI Camera Connector
RCA Video/Audio Jack

Raspberry Pi 4

◆ Raspberry Pi 4 Model B



Bottom View



micro
SD Card

1GByte LPDDR2
memory

UART (Universal Asynchronous Receiver/Transmitter)

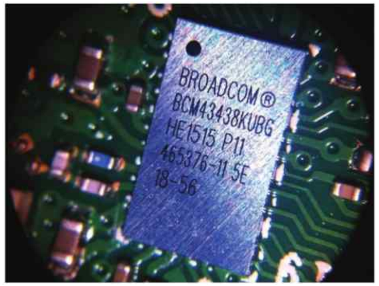
◆ Mini UART

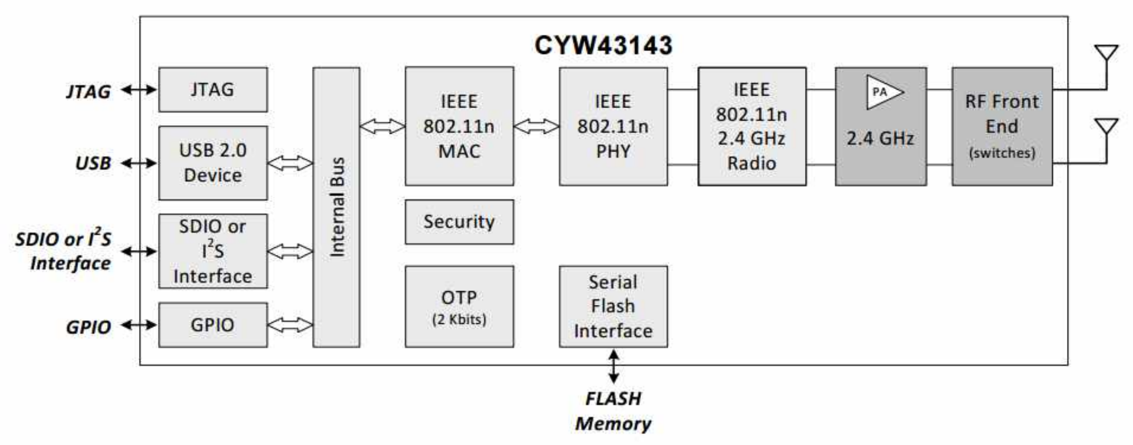
- 7 or 8 bit operation
- 1 start and 1 stop bit
- No parities
- Break generation
- 8 symbols deep FIFOs for receive and transmit
- SW controlled RTS (Request to Send), SW readable CTS (Clear to Send)
- Auto flow control with programmable FIFO level
- 16550 like registers
- Baud rate derived from system clock.

WiFi on Board

◆ BCM43143 (Cypress CYW43143)

- the Broadcom BCM43438 chip provides 2.4GHz 802.11n wireless LAN, Bluetooth Low Energy, and Bluetooth 4.1 Classic radio support
- 2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE





Bluetooth Low Energy (BLE) on board

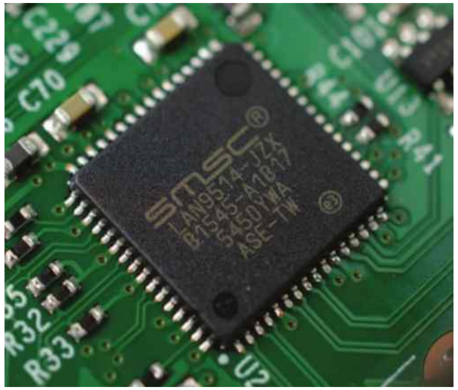
◆ BLE on Raspberry Pi 3

- **Raspberry Pi 3 Model B**, comes with both built-in Wi-Fi and Bluetooth 4.1
- Bluetooth devices that can be connected to Raspberry Pi
  - Keyboard
  - Mouse
  - Audio

USB Ports on Raspberry Pi 3

◆ SMSC LAN9514 USB Hub and Ethernet Controller

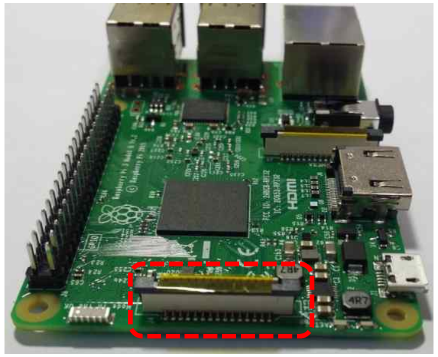
- The Raspberry Pi 3 shares the same SMSC LAN9514 chip as its predecessor, the Raspberry Pi 2, adding 10/100 Ethernet connectivity and four USB 2.0 channels to the board.
- As before, the SMSC chip connects to the SoC via a single USB channel, acting as a USB-to-Ethernet adaptor and USB hub.



Raspberry Pi Display

◆ Raspberry Pi Display

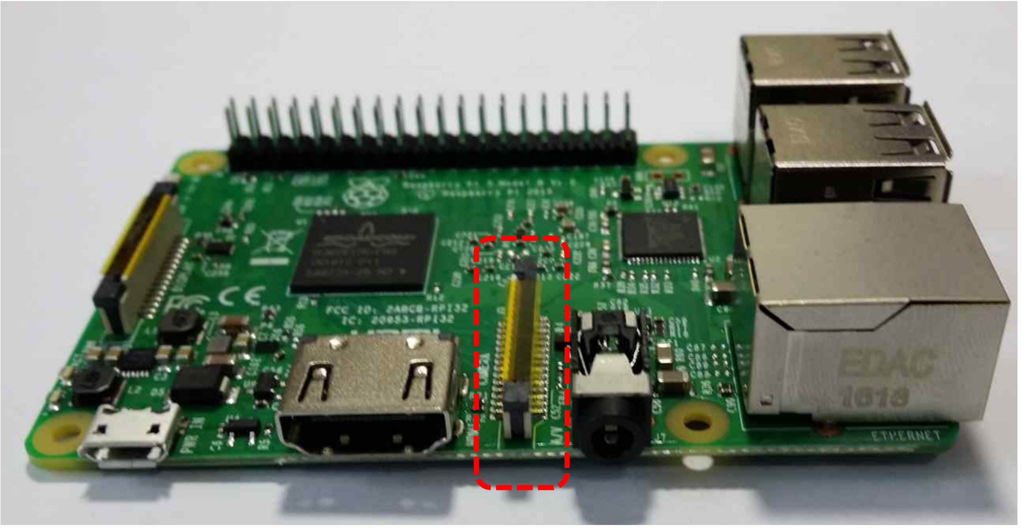
- The 7″ Touchscreen Monitor for Raspberry Pi gives users the ability to create all-in-one, integrated projects such as tablets, infotainment systems and embedded projects.
- The 800 x 480 display connects via an adapter board which handles power and signal conversion.
- Only two connections to the Pi are required; power from the Pi's GPIO port and a ribbon cable that connects to the DSI port present on all Raspberry Pis.
- Touchscreen drivers with support for 10-finger touch and an on-screen keyboard will be integrated into the latest Raspbian OS for full functionality without a physical keyboard or mouse.



Raspberry Pi Camera Interface

◆ Camera Serial Interface (CSI)

- The Raspberry Pi has a *Mobile Industry Processor Interface* (MIPI) *Camera Serial Interface Type 2* (CSI-2), which facilitates the connection of a small camera to the main Broadcom BCM2835 processor. This is a camera port providing an electrical bus connection between the two devices



Interface for External Sensors/Actuators - GPIO (General Purpose Input/Output) of Raspberry Pi

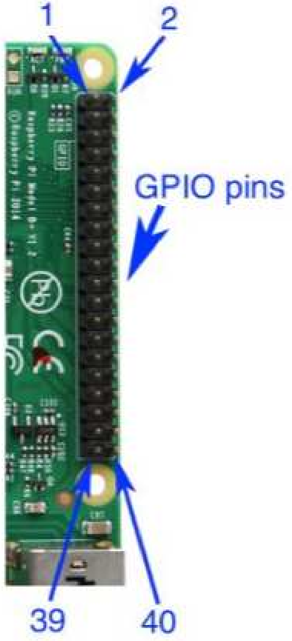
40-pin Extended GPIO

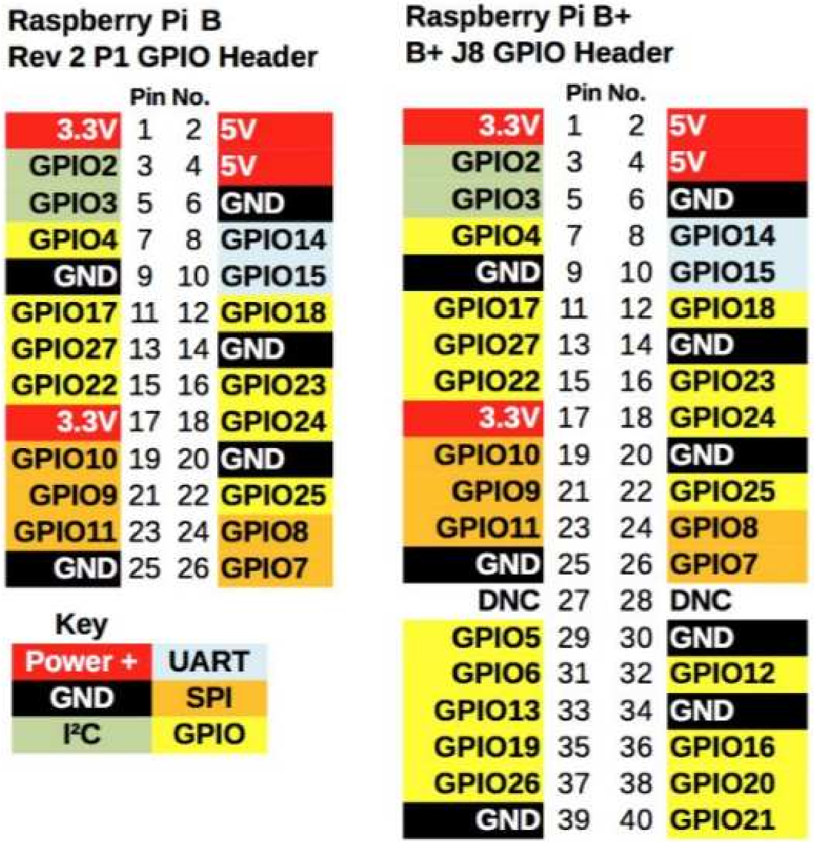
◆ GPIO (General Purpose Input / Output)

- The Raspberry Pi 4 features the same 40-pin general-purpose input-output (GPIO) header as all the Pis going back to the Model B+ and Model A+. Any existing GPIO hardware will work without modification; the only change is a switch to which UART is exposed on the GPIO's pins, but that's handled internally by the operating system.

40-pin Extended Raspberry GPIO

◆ Raspberry Pi 4 General purpose input-output (GPIO) Connector





Raspberry Pi B
Rev 2 P1 GPIO Header

	Pin No.		
3.3V	1	2	5V
GPIO2	3	4	5V
GPIO3	5	6	GND
GPIO4	7	8	GPIO14
GND	9	10	GPIO15
GPIO17	11	12	GPIO18
GPIO27	13	14	GND
GPIO22	15	16	GPIO23
3.3V	17	18	GPIO24
GPIO10	19	20	GND
GPIO9	21	22	GPIO25
GPIO11	23	24	GPIO8
GND	25	26	GPIO7

Key

Power +	UART
GND	SPI
I²C	GPIO

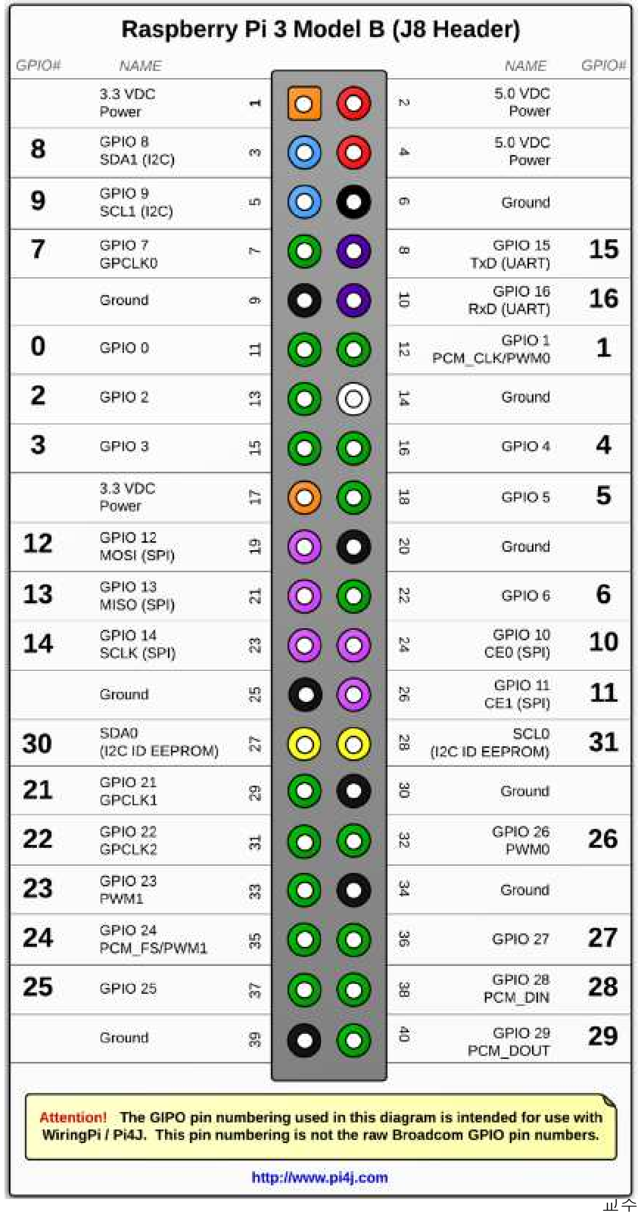
Raspberry Pi B+
B+ J8 GPIO Header

	Pin No.		
3.3V	1	2	5V
GPIO2	3	4	5V
GPIO3	5	6	GND
GPIO4	7	8	GPIO14
GND	9	10	GPIO15
GPIO17	11	12	GPIO18
GPIO27	13	14	GND
GPIO22	15	16	GPIO23
3.3V	17	18	GPIO24
GPIO10	19	20	GND
GPIO9	21	22	GPIO25
GPIO11	23	24	GPIO8
GND	25	26	GPIO7
DNC	27	28	DNC
GPIO5	29	30	GND
GPIO6	31	32	GPIO12
GPIO13	33	34	GND
GPIO19	35	36	GPIO16
GPIO26	37	38	GPIO20
GND	39	40	GPIO21

Raspberry GPIO

- **3.3VDC**
  - pin 1, 17
- **5.0VDC**
  - pin 2, 4
- **Ground**
  - pin 6, 9, 20, 25, 34, 39
- **GPCLK**
  - GPCLK0 (GPIO 7): pin 7
  - GPCLK1 (GPIO21): pin 29
  - GPCLK2 (GPIO22): pin 31
- **GPIO for general application**
  - GPIO0: pin 11; GPIO2: pin 13
  - GPIO3: pin 15; GPIO4: pin 16
  - GPIO5: pin 18; GPIO6: pin 22
  - GPIO25: pin 37; GPIO27: pin 36
- **UART (Universal Asynchronous Receiver / Transmitter)**
  - TxD: pin 8 (GPIO 15)
  - RxD: pin 10 (GPIO 16)

(Source: http://pi4j.com/images/j8header-3b-large.png)



Raspberry Pi 3 Model B (J8 Header)

GPIO#	NAME			NAME	GPIO#
	3.3 VDC Power	1	2	5.0 VDC Power	
8	GPIO 8 SDA1 (I2C)	3	4	5.0 VDC Power	
9	GPIO 9 SCL1 (I2C)	5	6	Ground	
7	GPIO 7 GPCLK0	7	8	GPIO 15 TxD (UART)	15
	Ground	9	10	GPIO 16 RxD (UART)	16
0	GPIO 0	11	12	GPIO 1 PCM_CLK/PWM0	1
2	GPIO 2	13	14	Ground	
3	GPIO 3	15	16	GPIO 4	4
	3.3 VDC Power	17	18	GPIO 5	5
12	GPIO 12 MOSI (SPI)	19	20	Ground	
13	GPIO 13 MISO (SPI)	21	22	GPIO 6	6
14	GPIO 14 SCLK (SPI)	23	24	GPIO 10 CE0 (SPI)	10
	Ground	25	26	GPIO 11 CE1 (SPI)	11
30	SDA0 (I2C ID EEPROM)	27	28	SCL0 (I2C ID EEPROM)	31
21	GPIO 21 GPCLK1	29	30	Ground	
22	GPIO 22 GPCLK2	31	32	GPIO 26 PWM0	26
23	GPIO 23 PWM1	33	34	Ground	
24	GPIO 24 PCM_FS/PWM1	35	36	GPIO 27	27
25	GPIO 25	37	38	GPIO 28 PCM_DIN	28
	Ground	39	40	GPIO 29 PCM_DOUT	29

Attention! The GIPO pin numbering used in this diagram is intended for use with WiringPi / Pi4J. This pin numbering is not the raw Broadcom GPIO pin numbers.

http://www.pi4j.com

◆ SPI (Serial Peripheral Interface)

- MOSI: pin 19 (GPIO 12)
- MISO: pin 21 (GPIO 13)
- SCLK: pin 23 (GPIO 14)
- CE0: pin 24 (GPIO 10)
- CE1: pin 26 (GPIO 11)

◆ I²C (Inter-IC)

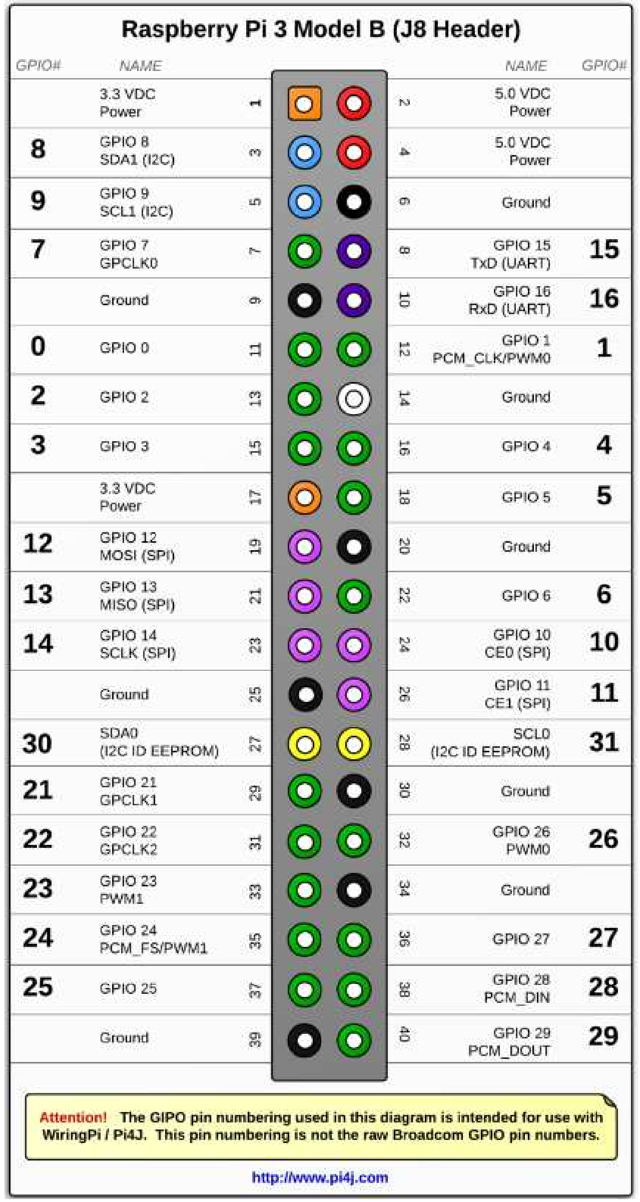
- SDA0(I2C ID EEPROM): pin 27
- SCL0(I2C ID EEPROM): pin 28
- SDA1(I2C): pin 3 (GPIO 8)
- SCL1(I2C): pin 5 (GPIO 9)

◆ PCM (Pulse Coded Modulation)

- PCM_DIN: pin 38 (GPIO 28)
- PCM_DOUT: pin 40 (GPIO 29)
- PCM_FS/PWM1: pin 35 (GPIO 24)
- PCM_CLK/PWM0: pin 12 (GPIO 1)

◆ PWM (Pulse Width Modulation)

- PWM0: pin 32 (GPIO 26)
- PWM1: pin 33 (GPIO 23)



Raspberry Pi 3 Model B (J8 Header)

GPIO#	NAME			NAME	GPIO#
	3.3 VDC Power	1	2	5.0 VDC Power	
8	GPIO 8 SDA1 (I2C)	3	4	5.0 VDC Power	
9	GPIO 9 SCL1 (I2C)	5	6	Ground	
7	GPIO 7 GPCLK0	7	8	GPIO 15 TxD (UART)	15
	Ground	9	10	GPIO 16 RxD (UART)	16
0	GPIO 0	11	12	GPIO 1 PCM_CLK/PWM0	1
2	GPIO 2	13	14	Ground	
3	GPIO 3	15	16	GPIO 4	4
	3.3 VDC Power	17	18	GPIO 5	5
12	GPIO 12 MOSI (SPI)	19	20	Ground	
13	GPIO 13 MISO (SPI)	21	22	GPIO 6	6
14	GPIO 14 SCLK (SPI)	23	24	GPIO 10 CE0 (SPI)	10
	Ground	25	26	GPIO 11 CE1 (SPI)	11
30	SDA0 (I2C ID EEPROM)	27	28	SCL0 (I2C ID EEPROM)	31
21	GPIO 21 GPCLK1	29	30	Ground	
22	GPIO 22 GPCLK2	31	32	GPIO 26 PWM0	26
23	GPIO 23 PWM1	33	34	Ground	
24	GPIO 24 PCM_FS/PWM1	35	36	GPIO 27	27
25	GPIO 25	37	38	GPIO 28 PCM_DIN	28
	Ground	39	40	GPIO 29 PCM_DOUT	29

Attention! The GIPO pin numbering used in this diagram is intended for use with WiringPi / Pi4J. This pin numbering is not the raw Broadcom GPIO pin numbers.

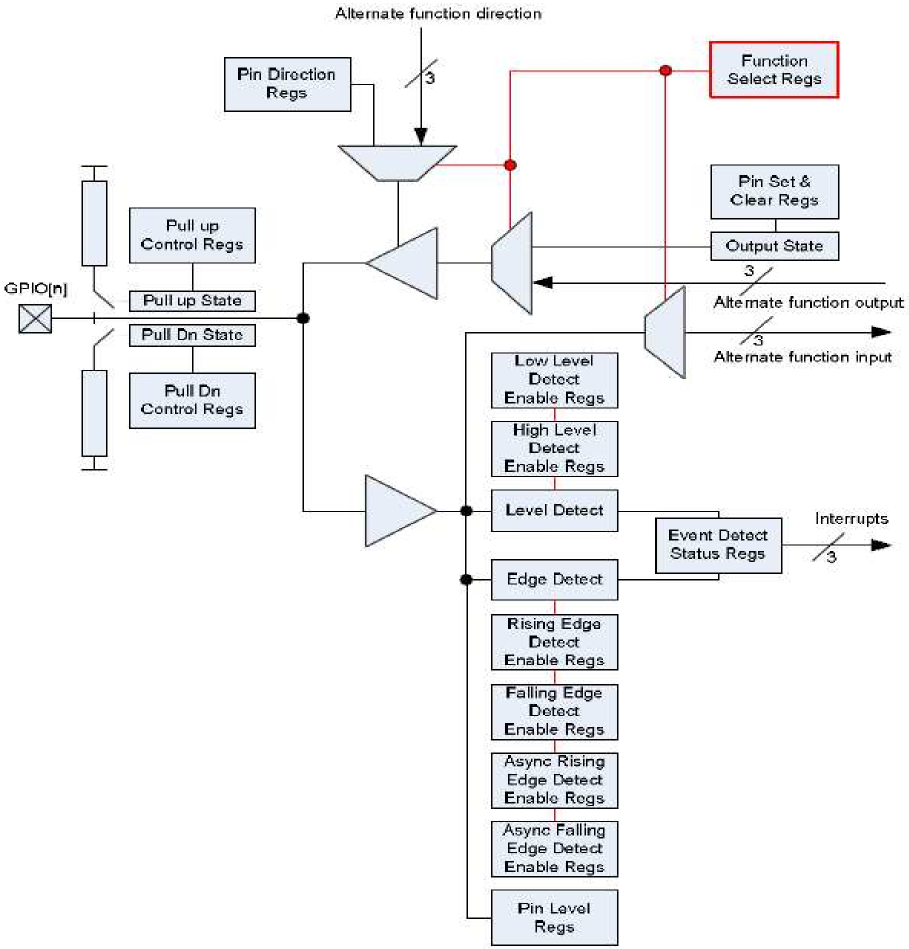
http://www.pi4j.com

(Source: http://pi4j.com/images/j8header-3b-large.png)

General Purpose Input/Output (GPIO)

◆ Programmable GPIO

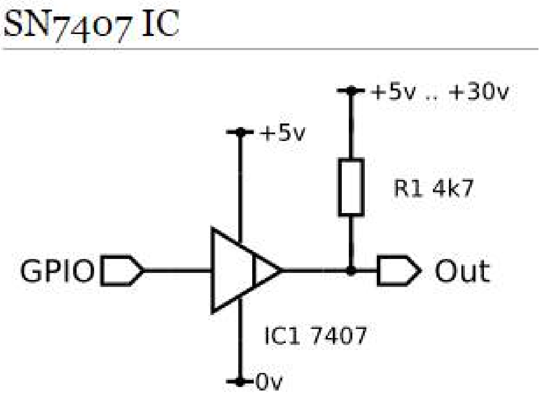
- internal pull-up and pull-down
- programmable input or output functions
- event detect and interrupt


Alternate function direction
Pin Direction Regs
Function Select Regs
3
Pin Set & Clear Regs
Output State
Pull up Control Regs
Pull up State
Pull Dn State
Pull Dn Control Regs
GPIO[n]
Alternate function output
Alternate function input
Low Level Detect Enable Regs
High Level Detect Enable Regs
Level Detect
Event Detect Status Regs
Interrupts
Edge Detect
Rising Edge Detect Enable Regs
Falling Edge Detect Enable Regs
Async Rising Edge Detect Enable Regs
Async Falling Edge Detect Enable Regs
Pin Level Regs

RPi GPIO Open Collector Interface Circuits

◆ SN7407 IC

- The **SN7407 open-collector buffer IC** is useful for a variety of level shifting and output-driver tasks. When the GPIO output is '0' (low), the 7407 output will be low. When the GPIO is '1', the 7407 output is 'not driven', and R1 will pull the circuit output high. The output-high voltage can be as high as 30V.
- The 7407 output can sink up to 30mA when the output is low, so low-current loads (e.g. LEDs) can be connected directly in place of R1. The IC has six independent circuits in a 14-pin DIP package, and is quite cheap (roughly £0.50).



(Source: http://elinux.org/ RPi_GPIO_Interface_Circuits)

Interface for External Sensors/Actuators
- PWM (Pulse Width Modulation)
- SPI (Serial Peripheral Interface)

Raspberry PI Hardware PWM

◆ Features

- Two independent output bit-streams, clocked at a fixed frequency.
- Bit-streams configured individually to output either PWM or a serialised version of a 32-bit word.
- PWM outputs have variable input and output resolutions.
- Serialize mode configured to load data to and/or read data from a FIFO storage block, which can store up to eight 32-bit words.
- Both modes clocked by *clk_pwm* which is nominally 100MHz, but can be varied by the clock manager.

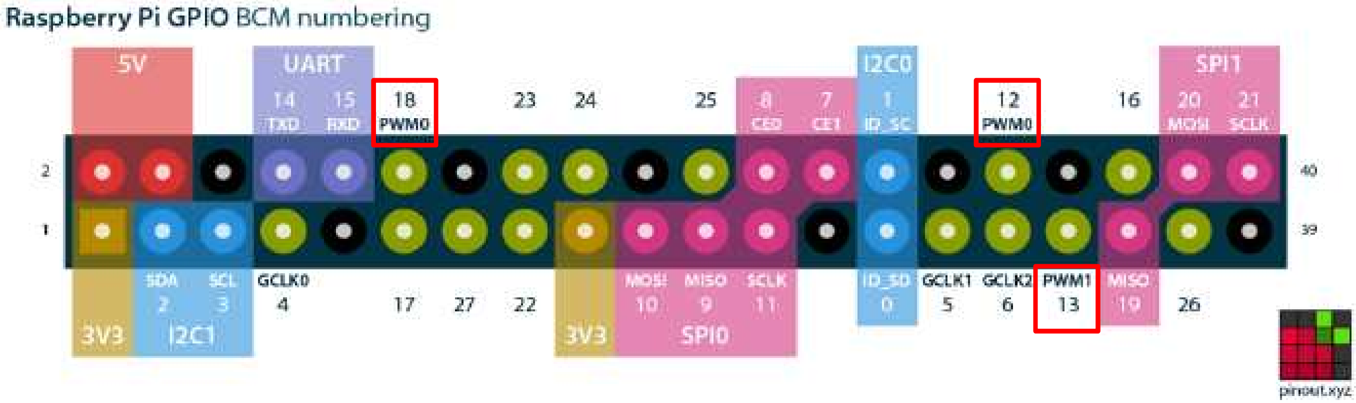
Hardware PWM

Hardware PWM API at WiringPi

- pinMode (int pin, int mode)
  - This sets the mode of a pin **PWM_OUTPUT**.
    Note that only ***wiringPi*** pin 1 (BCM_GPIO 18) supports PWM output.
- pwmSetMode (int mode)
  - The PWM generator can run in 2 modes – "balanced" and "mark:space". The mark:space mode is traditional, however the default mode in the Pi is "balanced". You can switch modes by supplying the parameter: **PWM_MODE_BAL** or **PWM_MODE_MS**.
- pwmSetRange (unsigned int range)
  - This sets the range register in the PWM generator. The default is 1024.
- pwmSetClock (int divisor)
  - This sets the divisor for the PWM clock.
  - Parameter divisor range: 2 ~ 4095
- pwmWrite (int pin, int value)
  - Writes the value to the PWM register for the given pin.
    (Again, note that only ***wiringPi*** pin 1 (BCM_GPIO 18) supports PWM)

Hardware PWM

- **Hardware PWM API at WiringPi**



(src: https://pinout.xyz/)

Serial Peripheral Interface (SPI)

◆ What is SPI?

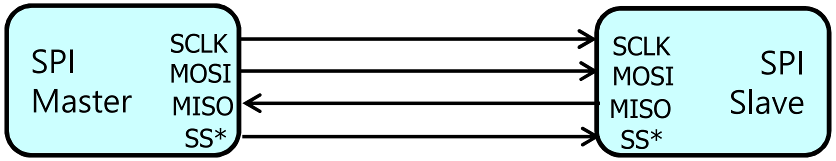
- serial peripheral interface (SPI)
- a synchronous serial communication interface specification used for short distance communication
- SPI bus can operate with a single master device and with one or more slave devices

◆ SPI Connection

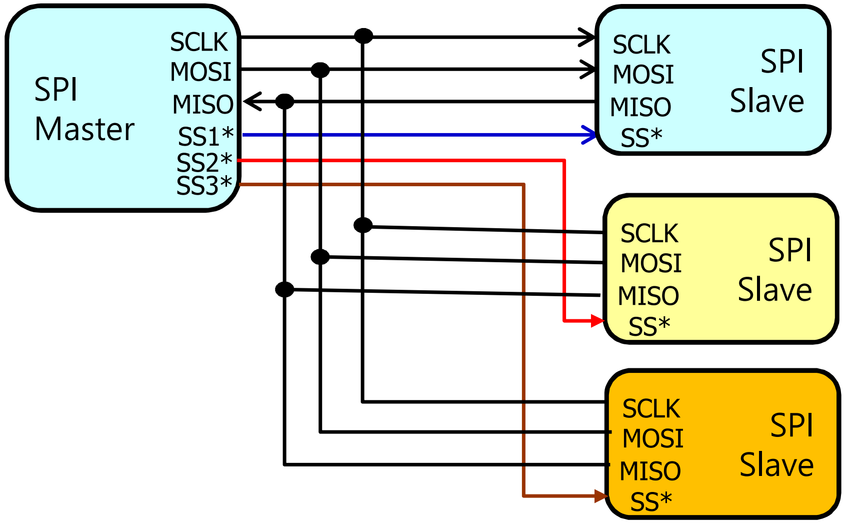
- SCLK – Serial Clock (output from the master)
- MOSI – Master Output, Slave Input
- MISO – Master Input, Slave Output
- SS – Slave Select (active low, output from the master)

Serial Peripheral Interface (SPI)

◆ SPI Connection



(a) Master – Slave 1:1 Connection



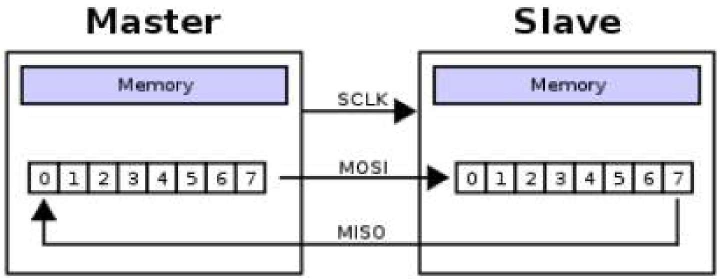
(b) Master – Slave 1:N Connection

Serial Peripheral Interface (SPI)

◆ Features of SPI

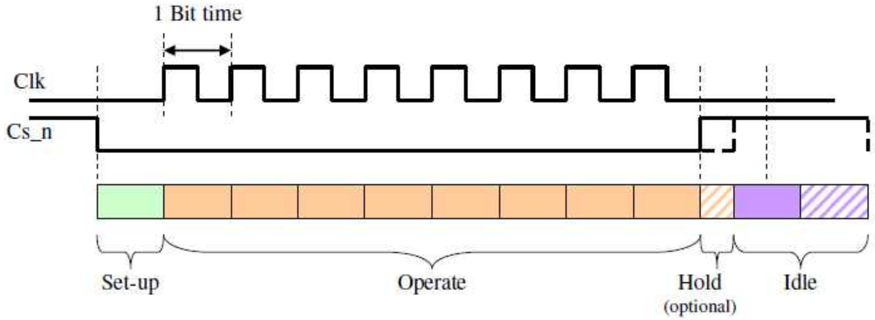
- SPI is a full duplex synchronous communication protocol
- SPI is single master – multiple slaves communication protocol
- SPI protocol has complete flexibility for the bits transferred as there is no limit for 8-bit word, message size, or purpose

◆ A typical hardware setup using two shift registers to form an inter-chip circular buffer



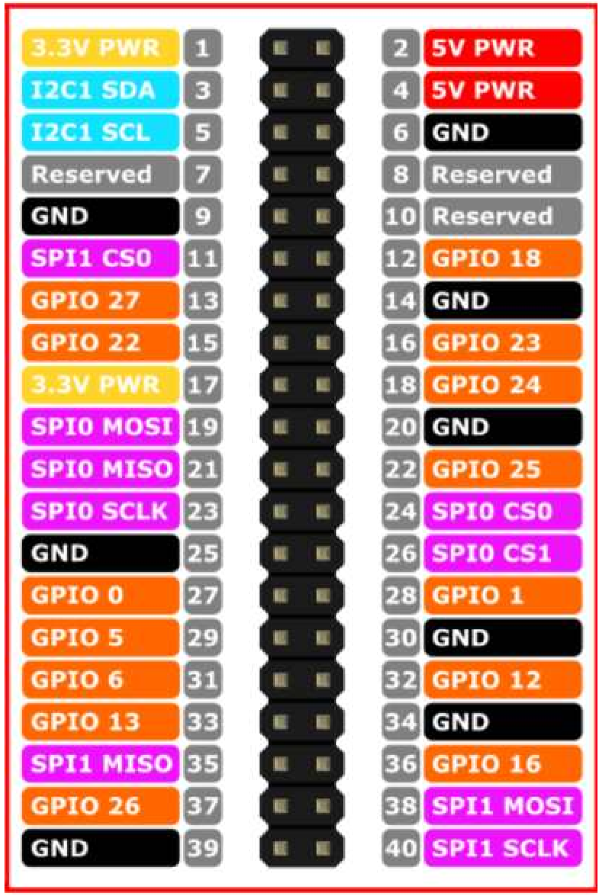
RaspberryPI SPI

◆ typical SPI access cycle



◆ Raspberry PI Pin Layout

- 2 SPI channels: SPI0, SPI1
- cs0 and cs1 for SPI0
- cs0 for SPI1



3.3V PWR	1	2	5V PWR
I2C1 SDA	3	4	5V PWR
I2C1 SCL	5	6	GND
Reserved	7	8	Reserved
GND	9	10	Reserved
SPI1 CS0	11	12	GPIO 18
GPIO 27	13	14	GND
GPIO 22	15	16	GPIO 23
3.3V PWR	17	18	GPIO 24
SPI0 MOSI	19	20	GND
SPI0 MISO	21	22	GPIO 25
SPI0 SCLK	23	24	SPI0 CS0
GND	25	26	SPI0 CS1
GPIO 0	27	28	GPIO 1
GPIO 5	29	30	GND
GPIO 6	31	32	GPIO 12
GPIO 13	33	34	GND
SPI1 MISO	35	36	GPIO 16
GPIO 26	37	38	SPI1 MOSI
GND	39	40	SPI1 SCLK

RaspberryPI SPI

◆ RaspberryPI SPI

- Hardware
  - Standard mode
    - In Standard SPI master mode the peripheral implements the standard 3wire serial protocol.
    - Signal names: Clock=CLK, ChipSelect=CE, SerialOut=MOSI (Master Out Slave In), SerialIn=MISO (Master In Slave Out)
  - Bidirectional mode
  - LoSSI(Low Speed Serial Interface) mode
- Transfer modes
  - Polled
  - Interrupt
  - DMA (Direct Memory Access)
- Speed
  - SCLK = Core Clock() / CDIV
  - The lowest SPI clock frequency with a 250 MHz system clock is 30.5 KHz.
  - 250 MHz system clock will add hold times in units of 4 ns.

Raspberry Pi와
Raspberry Pi OS (Raspbian)
준비/설치

Setting up Raspberry Pi

◆ Power Supply

- 5V, 3A, 15W
- micro-USB lead

◆ Keyboard and Mouse

- USB keyboard
- USB mouse

◆ Display

- HDMI display

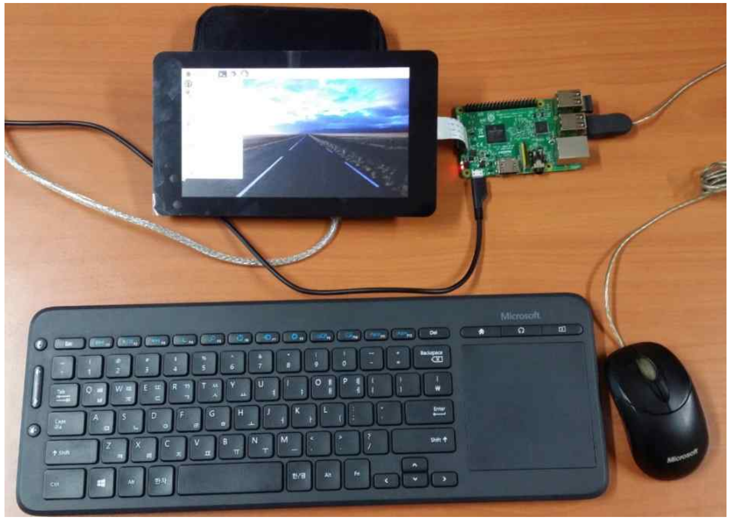
◆ Micro-SD Card

- prepared with Raspberry Pi Imager

Connecting Everything Together

◆ Raspberry Pi System

- Raspberry Pi 3
  - with micro-SD card of Raspbian OS
- Monitor
- USB Keyboard
- USB Mouse
- USB power adapter:
  - 1 for Raspberry Pi
  - 1 for Display (touch screen)
- Ethernet



Raspberry Pi OS (Raspbian)

◆ Linux

- open source operating system based on UNIX operating system concept
- various Linux distributions (or distros)

◆ Linux distributions (or distros)

- an operating system made from a software collection, which is based upon the Linux kernel and, often, a package management system
  - Debian: a non-commercial distribution and one of the earliest, maintained by a volunteer developer community
  - **Raspbian**: Debian-based computer operating system for Raspberry Pi
  - Ubuntu: desktop and server distribution derived from Debian, maintained by British company Canonical Ltd
  - Fedora: a community distribution sponsored by American company Red Hat
  - Centos: a distribution derived from the same sources used by Red Hat

Raspbian

◆ Raspberry Pi OS

- https://www.raspberrypi.com/software/

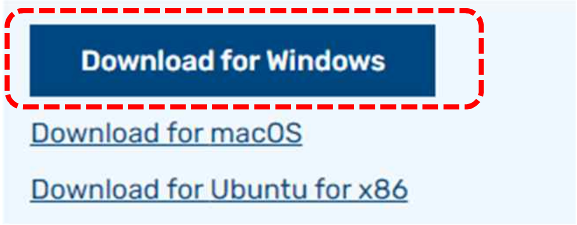
◆ Raspbian

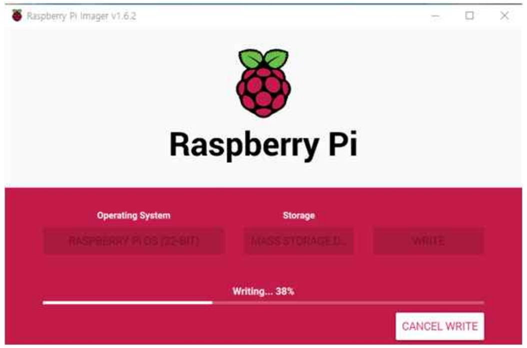
- (Source: https://en.wikipedia.org/wiki/Raspbian)
- **Raspbian** is a Debian-based computer operating system for Raspberry Pi, developed by a small team of developers.
- Raspberry Pi Foundation provides a Raspbian image which is listed as an officially supported operating system.

MicroSD 카드에 Raspbian 설치

◆ Raspberry Pi Imager

- https://www.raspberrypi.com/software/
- https://www.youtube.com/watch?v=ntaXWS8Lk34
- Raspberry Pi Imager “Download for Windows”
- Select operating system: Raspberry Pi OS (32-bit)
- Storage: Mass Storage (USB)
- Write

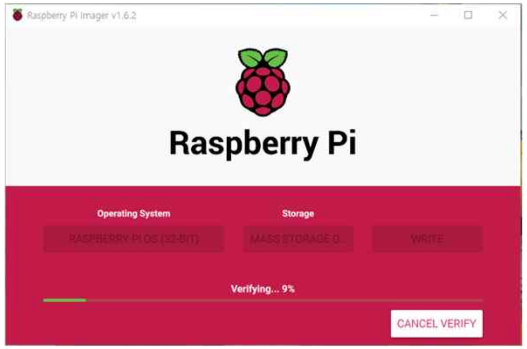



Raspberry Pi Imager v1.6.2

Operating System: RASPBERRY PI OS (32-BIT)

Storage: MASS STORAGE D...

Writing... 38%


Raspberry Pi Imager v1.6.2

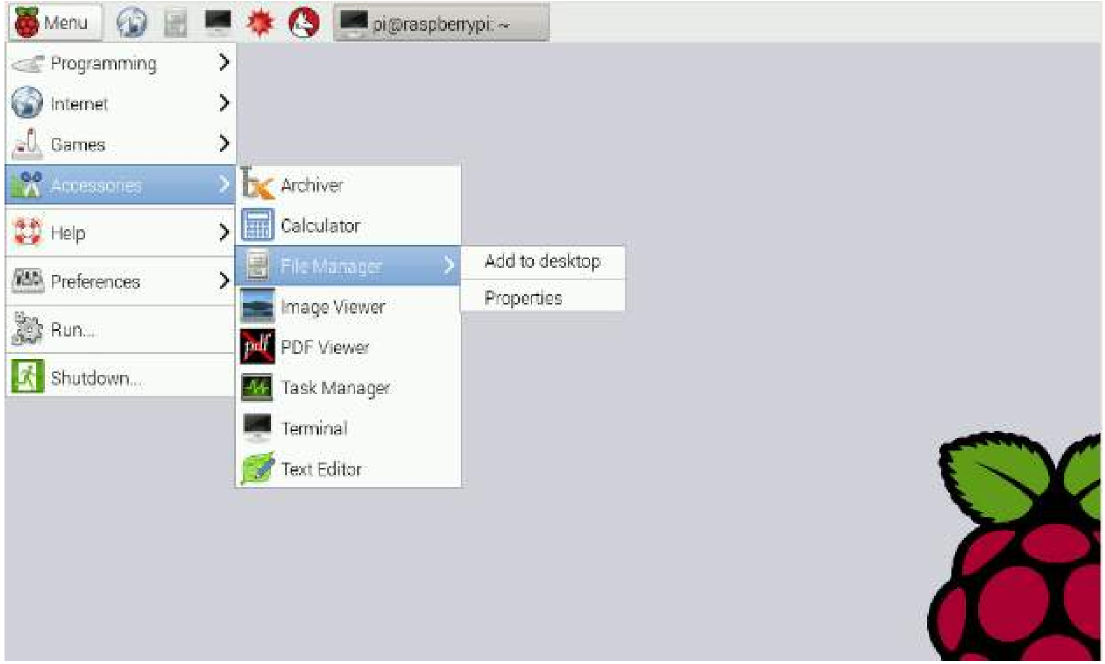
Operating System: RASPBERRY PI OS (32-BIT)

Storage: MASS STORAGE D...

Verifying... 9%

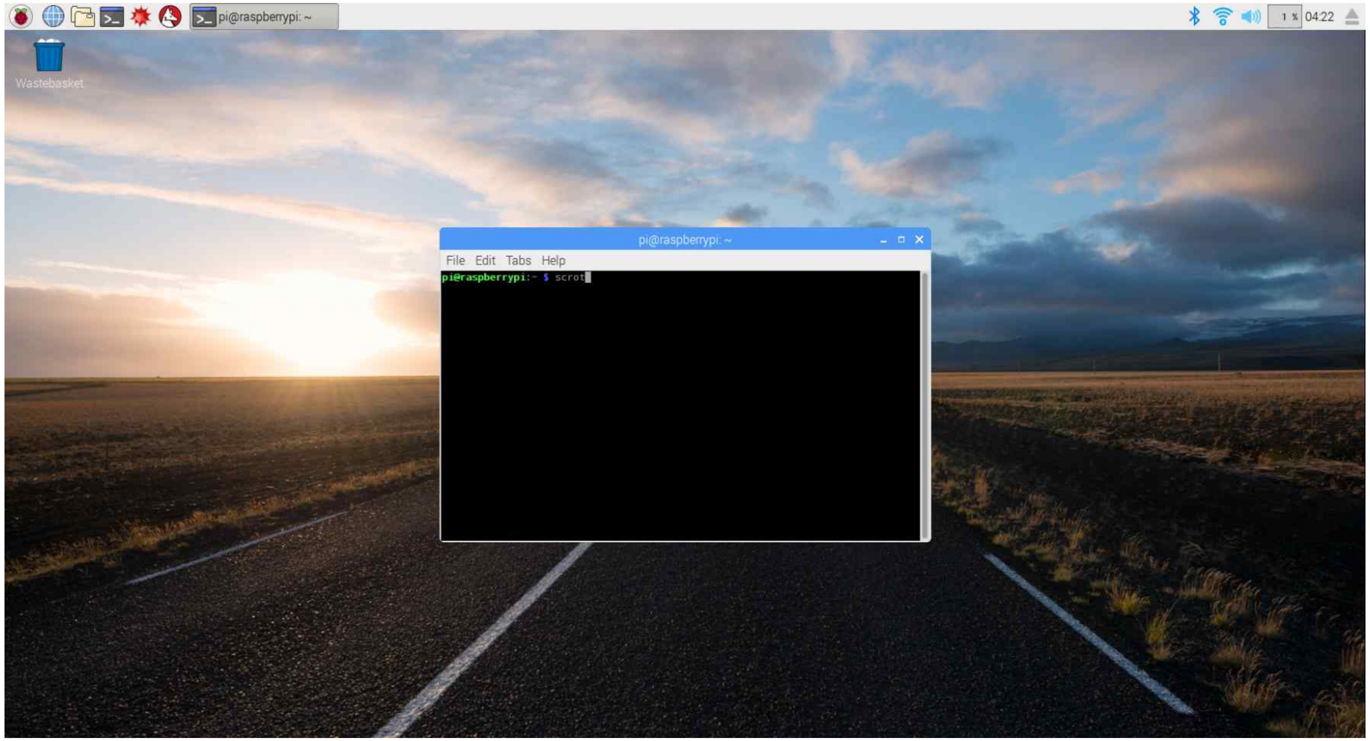
Raspberry Pi Desktop

◆ Raspbian Screen Shot



Default Terminal

◆ LXTerminal



```
pi@raspberrypi:~ $ scrot
```

Command Lines

◆ Basic Command Lines of Raspberry Pi OS

- pwd (print working directory)
- ls (list up files/directories in current directory)
- cd (change directory)
- sudo (super-user do)
- mv (move file)
- cp (copy file)
- rm (remove file)
- mkdir (make directory)
- rmdir (remove directory)
- vi (visual editor)
- apt (advanced packaging tool), apt-get
  apt-get update, apt-get upgrade, apt-get install, apt-get remove, apt-get purge, apt-get autoremove
  (https://codechacha.com/ko/linux-apt-commands/)

Visual Editor (vi)

vi editor

- Start with vi
  - To start using vi, at the Linux/Unix prompt,
    type vi followed by a file name.
  - If you want to edit an existing file, type in its name;
    if you are creating a new file, type in the name of the new file.
  - **% vi filename**

vi Editor command

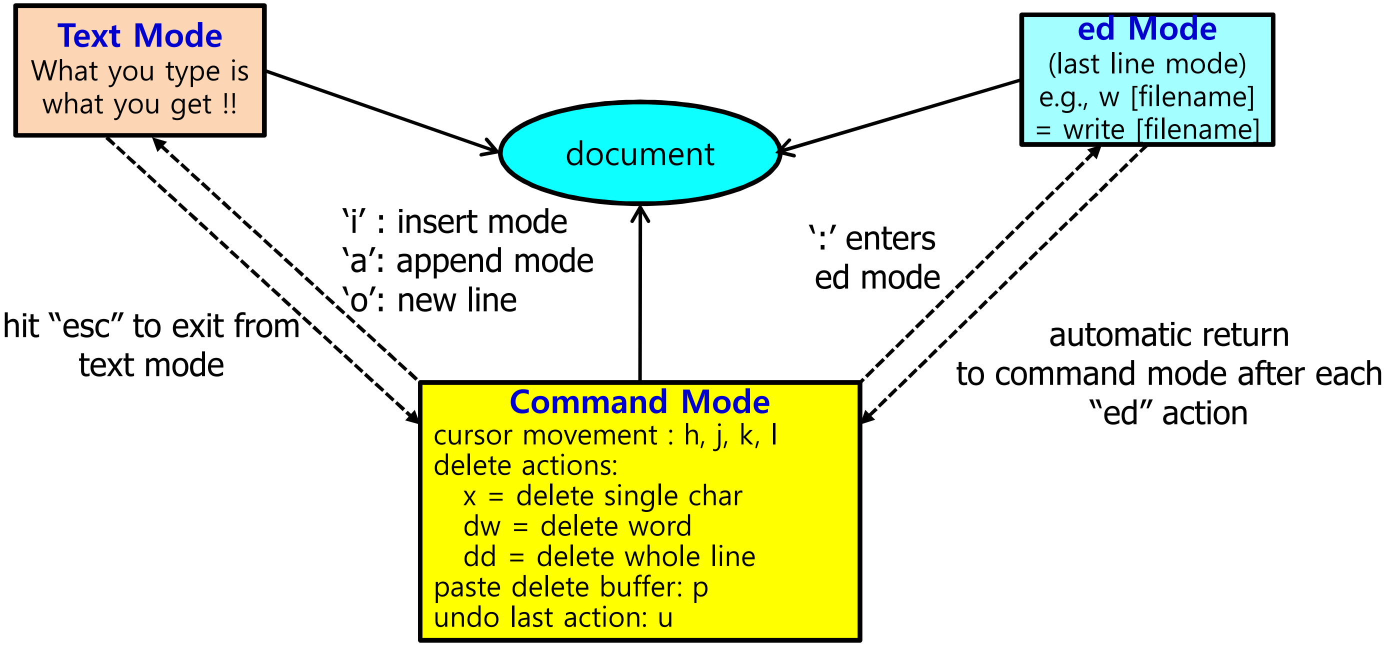
◆ Three modes

- vi has three modes: command mode, insert mode, and sophisticated edit mode (e.g., by using “:”).
- Command mode
  - When you are in command mode, letters of the keyboard will be interpreted as commands. vi always starts out in command mode.
- Insert mode
  - When you are in insert mode the same letters of the keyboard will type or edit text.
- Sophisticated edit mode
  - e.g.) :s/old/new/g (replace all occurrence of “old” with “new” on current line only)
- How to change among the three modes
  - When you wish to move between the three modes, keep these things in mind.
  - You can type “i” to enter the insert mode. If you wish to leave insert mode and return to the command mode, hit the [ESC] key.
  - If you're not sure where you are, hit [ESC] a couple of times and that should put you back in command mode.

Editing text files with vi

◆ Moving between vi modes

- text insertion (that is. input mode);
- cursor movement and simple editing (command mode);
- sophisticated editing operations (edit or ed mode)



Text Mode
What you type is
what you get !!

document

ed Mode
(last line mode)
e.g., w [filename]
= write [filename]

'i' : insert mode
'a': append mode
'o': new line

hit "esc" to exit from
text mode

':' enters
ed mode

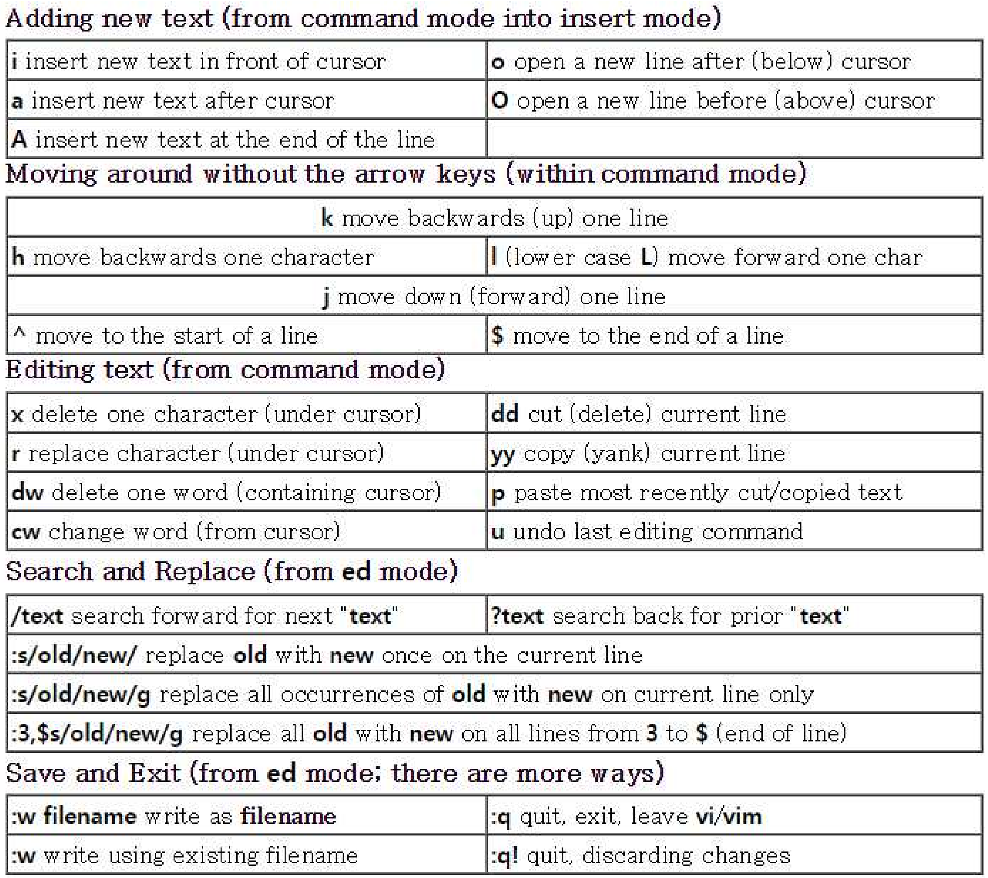
automatic return
to command mode after each
"ed" action

Command Mode
cursor movement : h, j, k, l
delete actions:
  x = delete single char
  dw = delete word
  dd = delete whole line
paste delete buffer: p
undo last action: u

(Source: http://elearning.algonquincollege.com/coursemat/allisor/cst8177/vi.html)

Editing text files with vi

◆ Some Basic Commands for vi



Adding new text (from command mode into insert mode)

i insert new text in front of cursor	**o** open a new line after (below) cursor
a insert new text after cursor	**O** open a new line before (above) cursor
A insert new text at the end of the line	

Moving around without the arrow keys (within command mode)

k move backwards (up) one line	
h move backwards one character	**l** (lower case **L**) move forward one char
j move down (forward) one line	
^ move to the start of a line	**$** move to the end of a line

Editing text (from command mode)

x delete one character (under cursor)	**dd** cut (delete) current line
r replace character (under cursor)	**yy** copy (yank) current line
dw delete one word (containing cursor)	**p** paste most recently cut/copied text
cw change word (from cursor)	**u** undo last editing command

Search and Replace (from ed mode)

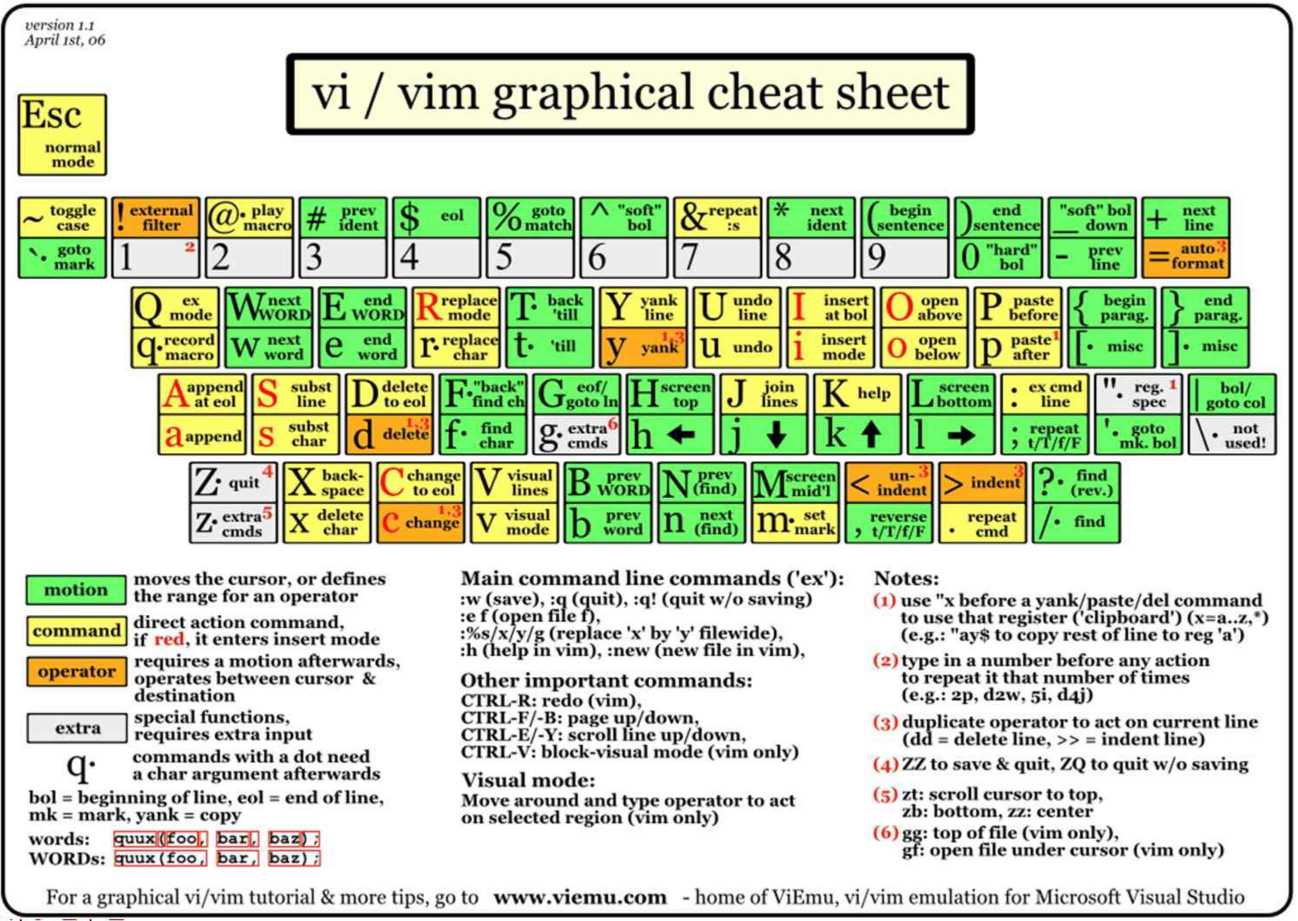
/text search forward for next "**text**"	**?text** search back for prior "**text**"
:s/old/new/ replace **old** with **new** once on the current line	
:s/old/new/g replace all occurrences of **old** with **new** on current line only	
:3,$s/old/new/g replace all **old** with **new** on all lines from **3** to **$** (end of line)	

Save and Exit (from ed mode; there are more ways)

:w filename write as **filename**	**:q** quit, exit, leave **vi/vim**
:w write using existing filename	**:q!** quit, discarding changes

VI Editor

◆ vi Editor Cheat sheet (http://www.viemu.com/vi-vim-cheat-sheet.gif)



🟩 motion: moves the cursor, or defines the range for an operator

🟨 command: direct action command, if red, it enters insert mode

🟧 operator: requires a motion afterwards, operates between cursor & destination

⬜ extra: special functions, requires extra input

q· commands with a dot need a char argument afterwards

bol = beginning of line, eol = end of line,
mk = mark, yank = copy

words: quux(foo, bar, baz);
WORDs: quux(foo, bar, baz);

Main command line commands ('ex'):
:w (save), :q (quit), :q! (quit w/o saving)
:e f (open file f),
:%s/x/y/g (replace 'x' by 'y' filewide),
:h (help in vim), :new (new file in vim),

Other important commands:
CTRL-R: redo (vim),
CTRL-F/-B: page up/down,
CTRL-E/-Y: scroll line up/down,
CTRL-V: block-visual mode (vim only)

Visual mode:
Move around and type operator to act on selected region (vim only)

Notes:

(1) use "x before a yank/paste/del command to use that register ('clipboard') (x=a..z,*) (e.g.: "ay$ to copy rest of line to reg 'a')

(2) type in a number before any action to repeat it that number of times (e.g.: 2p, d2w, 5i, d4j)

(3) duplicate operator to act on current line (dd = delete line, >> = indent line)

(4) ZZ to save & quit, ZQ to quit w/o saving

(5) zt: scroll cursor to top, zb: bottom, zz: center

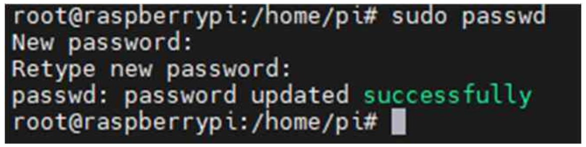
(6) gg: top of file (vim only), gf: open file under cursor (vim only)

For a graphical vi/vim tutorial & more tips, go to **www.viemu.com** - home of ViEmu, vi/vim emulation for Microsoft Visual Studio

Create Root Account registration

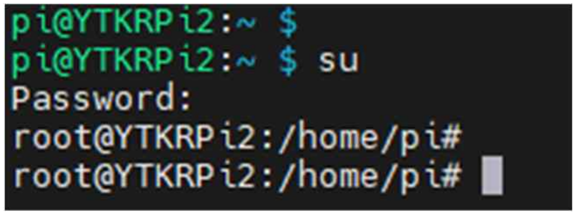
◆ Super-user (root)의 password 등록

```
%sudo passwd
New password: ****   // new super-user password 등록
Retype  new password
```



```
root@raspberrypi:/home/pi# sudo passwd
New password:
Retype new password:
passwd: password updated successfully
root@raspberrypi:/home/pi# 
```

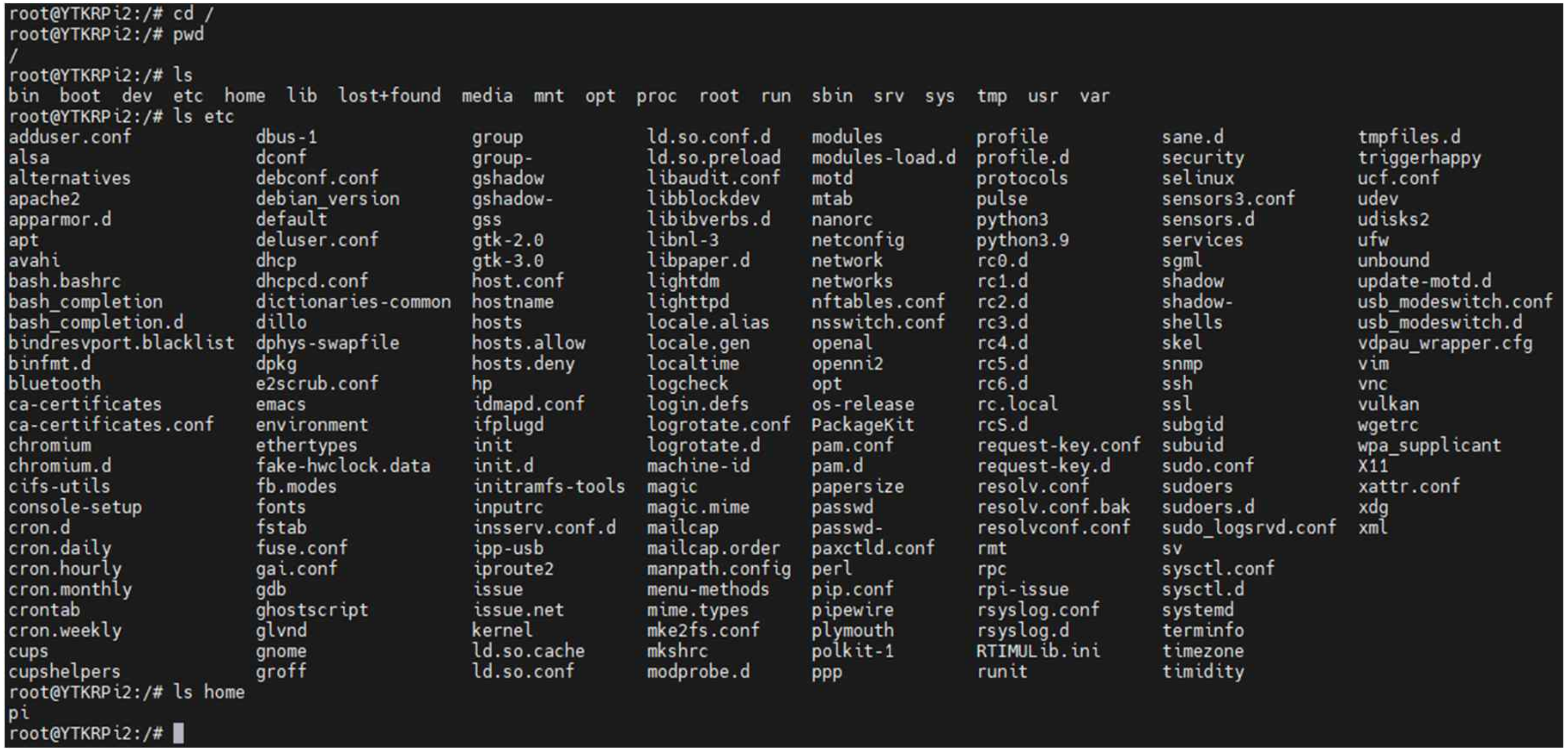
◆ "%su" 명령으로 Super-user (root)로 전환



```
pi@YTKRPi2:~ $
pi@YTKRPi2:~ $ su
Password:
root@YTKRPi2:/home/pi#
root@YTKRPi2:/home/pi# 
```

Raspberry Pi OS의 파일/디렉토리 구조

◆ "#cd /" 명령어로 root directory로 이동하여 ls 명령어로 파일 확인



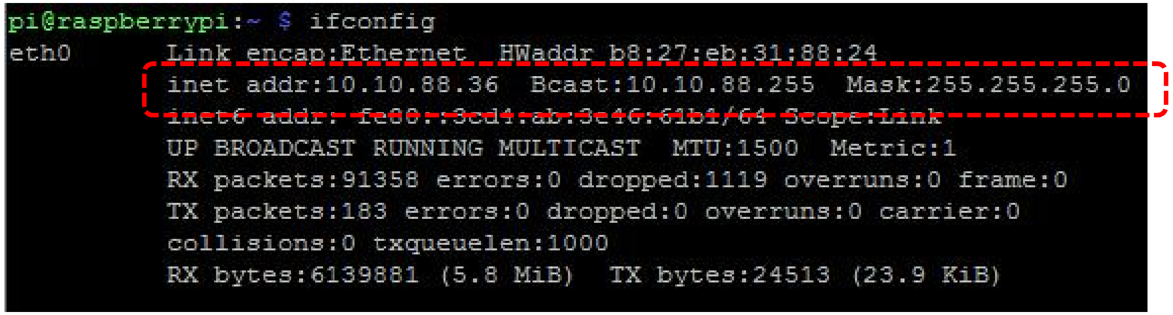
```
root@YTKRPi2:/# cd /
root@YTKRPi2:/# pwd
/
root@YTKRPi2:/# ls
bin  boot  dev  etc  home  lib  lost+found  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@YTKRPi2:/# ls etc
adduser.conf            dbus-1               group            ld.so.conf.d    modules         profile           sane.d             tmpfiles.d
alsa                    dconf                group-           ld.so.preload   modules-load.d  profile.d         security           triggerhappy
alternatives            debconf.conf         gshadow          libaudit.conf   motd            protocols         selinux            ucf.conf
apache2                 debian_version       gshadow-         libblockdev     mtab            pulse             sensors3.conf      udev
apparmor.d              default              gss              libibverbs.d    nanorc          python3           sensors.d          udisks2
apt                     deluser.conf         gtk-2.0          libnl-3         netconfig       python3.9         services           ufw
avahi                   dhcp                 gtk-3.0          libpaper.d      network         rc0.d             sgml               unbound
bash.bashrc             dhcpcd.conf          host.conf        lightdm         networks        rc1.d             shadow             update-motd.d
bash_completion         dictionaries-common  hostname         lighttpd        nftables.conf   rc2.d             shadow-            usb_modeswitch.conf
bash_completion.d       dillo                hosts            locale.alias    nsswitch.conf   rc3.d             shells             usb_modeswitch.d
bindresvport.blacklist  dphys-swapfile       hosts.allow      locale.gen      openal          rc4.d             skel               vdpau_wrapper.cfg
binfmt.d                dpkg                 hosts.deny       localtime       openni2         rc5.d             snmp               vim
bluetooth               e2scrub.conf         hp               logcheck        opt             rc6.d             ssh                vnc
ca-certificates         emacs                idmapd.conf      login.defs      os-release      rc.local          ssl                vulkan
ca-certificates.conf    environment          ifplugd          logrotate.conf  PackageKit      rcS.d             subgid             wgetrc
chromium                ethertypes           init             logrotate.d     pam.conf        request-key.conf  subuid             wpa_supplicant
chromium.d              fake-hwclock.data    init.d           machine-id      pam.d           request-key.d     sudo.conf          X11
cifs-utils              fb.modes             initramfs-tools  magic           papersize       resolv.conf       sudoers            xattr.conf
console-setup           fonts                inputrc          magic.mime      passwd          resolv.conf.bak   sudoers.d          xdg
cron.d                  fstab                insserv.conf.d   mailcap         passwd-         resolvconf.conf   sudo_logsrvd.conf  xml
cron.daily              fuse.conf            ipp-usb          mailcap.order   paxctld.conf    rmt               sv
cron.hourly             gai.conf             iproute2         manpath.config  perl            rpc               sysctl.conf
cron.monthly            gdb                  issue            menu-methods    pip.conf        rpi-issue         sysctl.d
crontab                 ghostscript          issue.net        mime.types      pipewire        rsyslog.conf      systemd
cron.weekly             glvnd                kernel           mke2fs.conf     plymouth        rsyslog.d         terminfo
cups                    gnome                ld.so.cache      mkshrc          polkit-1        RTIMULib.ini      timezone
cupshelpers             groff                ld.so.conf       modprobe.d      ppp             runit             timidity
root@YTKRPi2:/# ls home
pi
root@YTKRPi2:/# █
```

Raspberry Pi에 IP 주소 설정 확인

◆ Check IP address of Ethernet to connect Internet

- Connect Ethernet cable at Raspberry PI
- Check IP Address

  % ifconfig



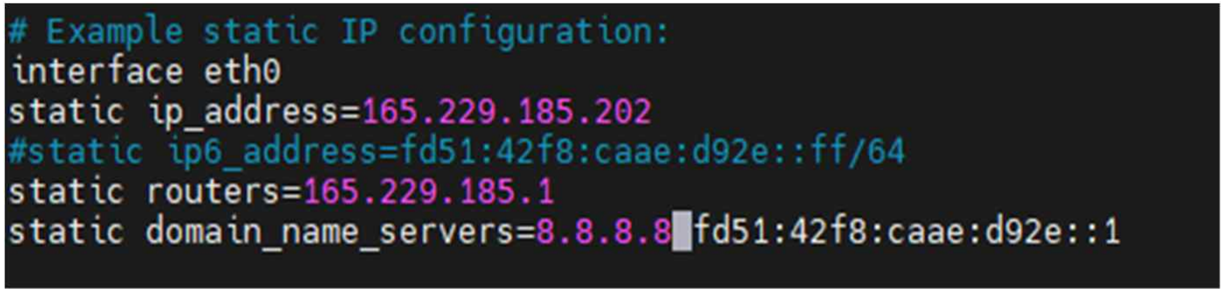
```
pi@raspberrypi:~ $ ifconfig
eth0      Link encap:Ethernet  HWaddr b8:27:eb:31:88:24
          inet addr:10.10.88.36  Bcast:10.10.88.255  Mask:255.255.255.0
          inet6 addr: fe80::3cd1:ab:3e46:61b1/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:91358 errors:0 dropped:1119 overruns:0 frame:0
          TX packets:183 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:6139881 (5.8 MiB)  TX bytes:24513 (23.9 KiB)
```

Raspberry Pi에 고정식 (static) IP 주소 할당

◆ Raspberry allocate static IP

$ sudo vi /etc/dhcpcd.conf

```
interface eth0
static ip_address=165.229.185.202
static routers=165.229.185.1
static domain_name_server = 165.229.11.5
```

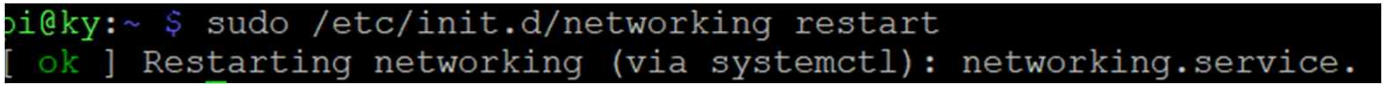


```
# Example static IP configuration:
interface eth0
static ip_address=165.229.185.202
#static ip6_address=fd51:42f8:caae:d92e::ff/64
static routers=165.229.185.1
static domain_name_servers=8.8.8.8 fd51:42f8:caae:d92e::1
```

고정식 (static) IP 주소 할당 결과 확인

- **Raspberry allocate static IP**

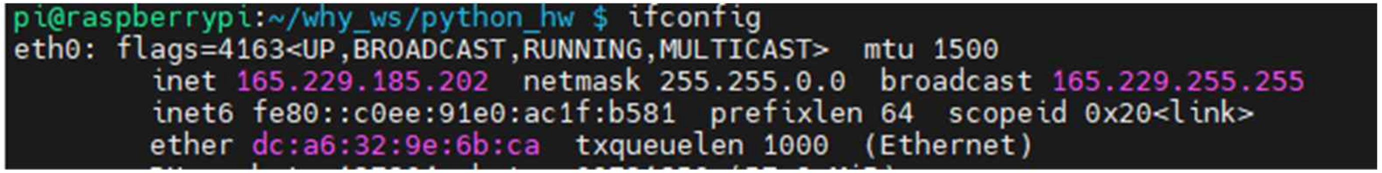
  $ sudo /etc/init.d/networking restart



```
pi@ky:~ $ sudo /etc/init.d/networking restart
[ ok ] Restarting networking (via systemctl): networking.service.
```

- **Raspberry allocate static IP**

  $ ifconfig   // IP 주소 설정이 정상적으로 완료된 것을 확인



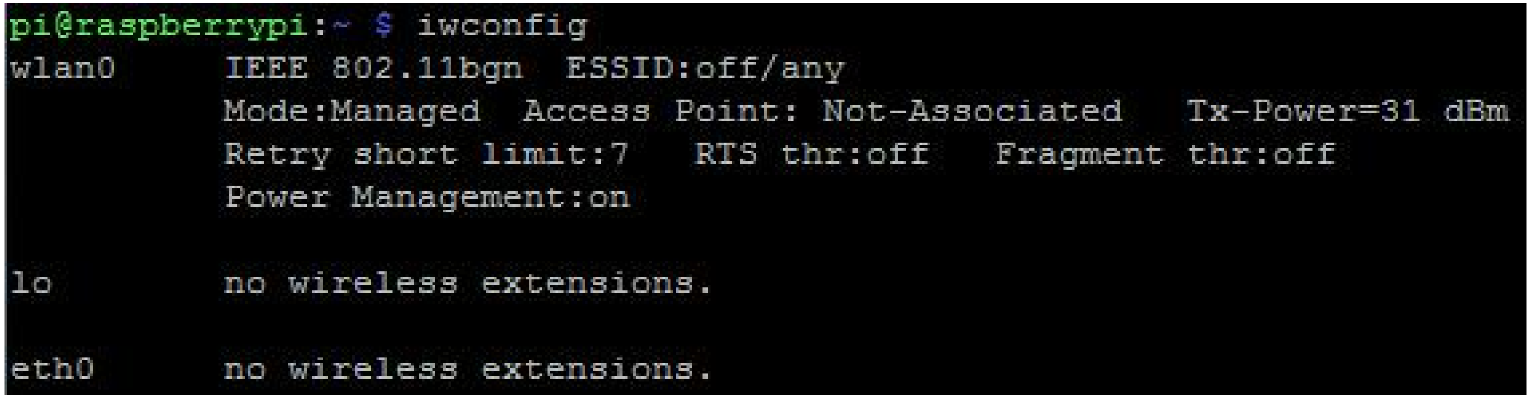
```
pi@raspberrypi:~/why_ws/python_hw $ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 165.229.185.202  netmask 255.255.0.0  broadcast 165.229.255.255
        inet6 fe80::c0ee:91e0:ac1f:b581  prefixlen 64  scopeid 0x20<link>
        ether dc:a6:32:9e:6b:ca  txqueuelen 1000  (Ethernet)
```

무선 LAN (WiFi) 설정 (1)

◆ Setting WiFi to connect Internet (1)

- 참고자료: https://webnautes.tistory.com/903
- Raspberry Pi 4 has WiFi chip, so you can connect WiFi without any other additional modules.
- Using '**iwconfig**' command, check wlan functions at your device.

  % iwconfig



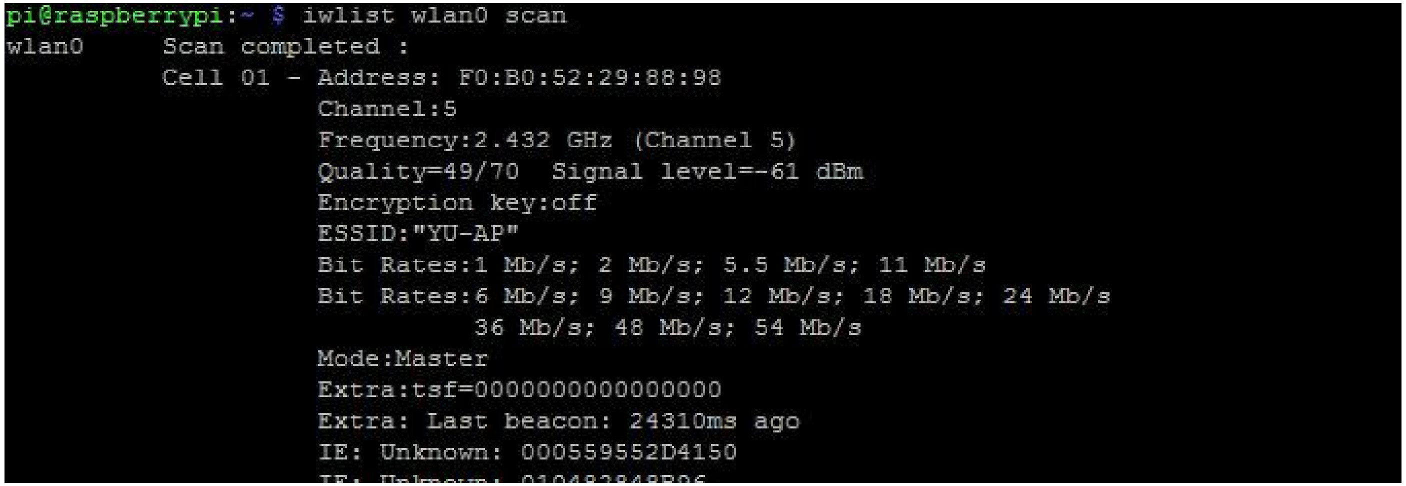
```
pi@raspberrypi:~ $ iwconfig
wlan0     IEEE 802.11bgn  ESSID:off/any
          Mode:Managed  Access Point: Not-Associated   Tx-Power=31 dBm
          Retry short limit:7   RTS thr:off   Fragment thr:off
          Power Management:on

lo        no wireless extensions.

eth0      no wireless extensions.
```

무선 LAN (WiFi) 설정 (2)

◆ Setting WiFi to connect Internet (2)

- If your WiFi is activating, scan WiFi access points (APs) to be used.

% iwlist wlan0 scan

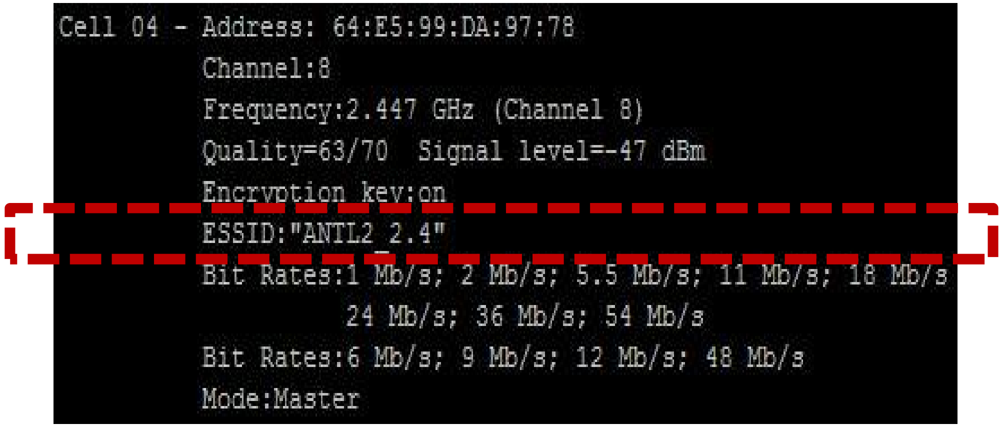


```
pi@raspberrypi:~ $ iwlist wlan0 scan
wlan0     Scan completed :
          Cell 01 - Address: F0:B0:52:29:88:98
                    Channel:5
                    Frequency:2.432 GHz (Channel 5)
                    Quality=49/70  Signal level=-61 dBm
                    Encryption key:off
                    ESSID:"YU-AP"
                    Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s
                    Bit Rates:6 Mb/s; 9 Mb/s; 12 Mb/s; 18 Mb/s; 24 Mb/s
                              36 Mb/s; 48 Mb/s; 54 Mb/s
                    Mode:Master
                    Extra:tsf=0000000000000000
                    Extra: Last beacon: 24310ms ago
                    IE: Unknown: 000559552D4150
                    IE: Unknown: 010482848B96
```

무선 LAN (WiFi) 설정 (3)

◆ Setting WiFi to connect Internet(3)

- from the scanned result, find an AP to use.



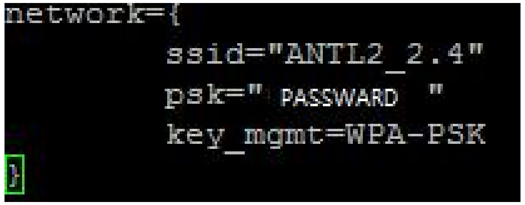
```
Cell 04 - Address: 64:E5:99:DA:97:78
          Channel:8
          Frequency:2.447 GHz (Channel 8)
          Quality=63/70  Signal level=-47 dBm
          Encryption key:on
          ESSID:"ANTL2_2.4"
          Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s; 18 Mb/s
                    24 Mb/s; 36 Mb/s; 54 Mb/s
          Bit Rates:6 Mb/s; 9 Mb/s; 12 Mb/s; 48 Mb/s
          Mode:Master
```

- Remember the ESSID (extended service set identifier) name, and your password for accessing the AP for Internet connectivity.

무선 LAN (WiFi) 설정 (4)

◆ Setting WiFi to connect Internet(4)

% sudo vi /etc/wpa_supplicant/wpa_supplicant.conf

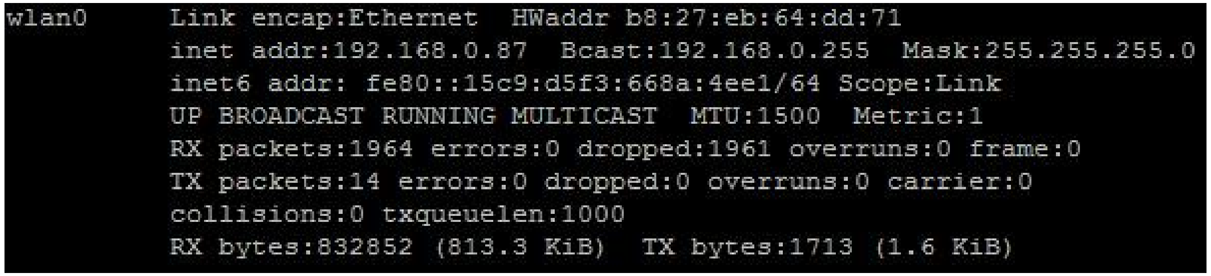


```
network={
        ssid="ANTL2_2.4"
        psk=" PASSWARD  "
        key_mgmt=WPA-PSK
}
```

- write your AP information, restart wlan0 device

% sudo ifdown wlan0

% sudo ifup wlan0

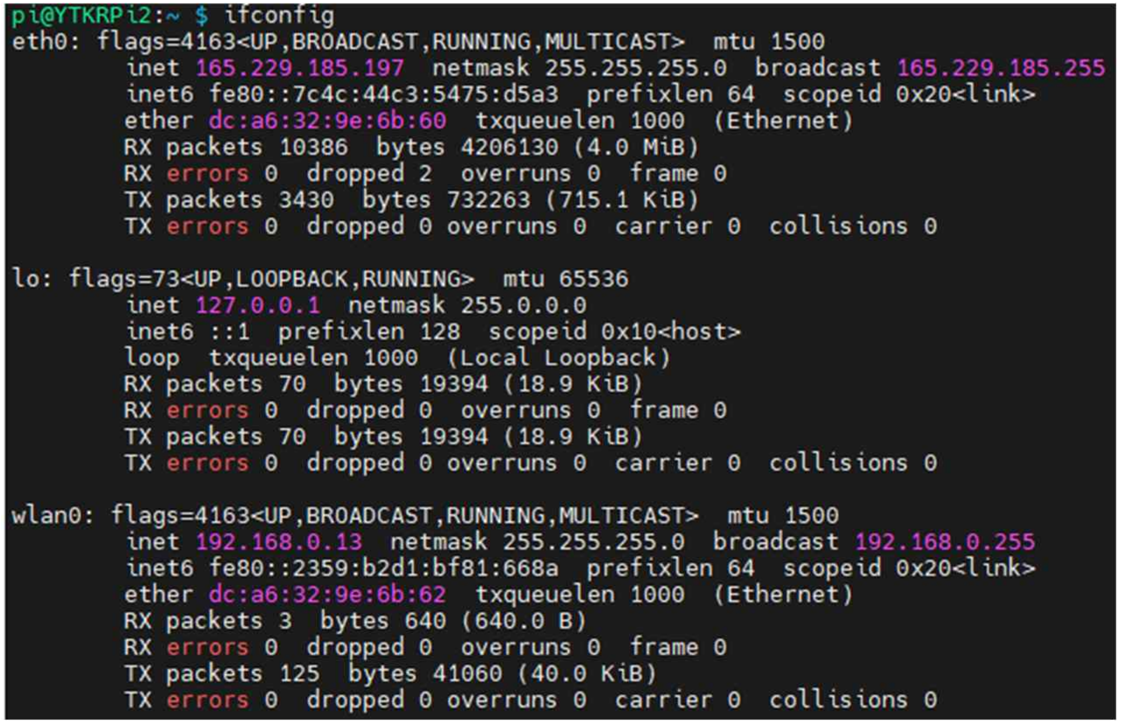


```
wlan0     Link encap:Ethernet  HWaddr b8:27:eb:64:dd:71
          inet addr:192.168.0.87  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::15c9:d5f3:668a:4ee1/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1964 errors:0 dropped:1961 overruns:0 frame:0
          TX packets:14 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:832852 (813.3 KiB)  TX bytes:1713 (1.6 KiB)
```

네트워크 접속 및 인터넷 통신 기능 확인

◆ **%ifconfig 명령어를 사용하여 네트워크 인터페이스 설정 확인**

% ifconfig



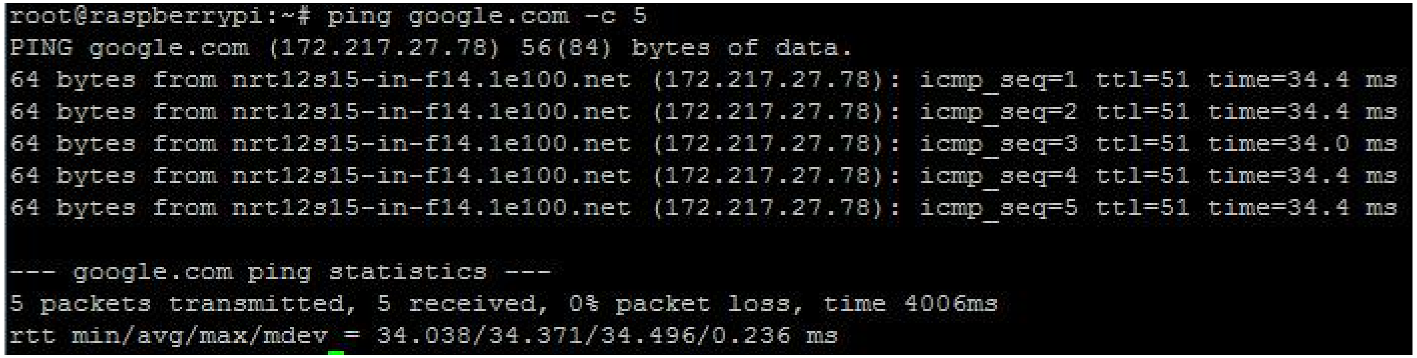
```
pi@YTKRPi2:~ $ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 165.229.185.197  netmask 255.255.255.0  broadcast 165.229.185.255
        inet6 fe80::7c4c:44c3:5475:d5a3  prefixlen 64  scopeid 0x20<link>
        ether dc:a6:32:9e:6b:60  txqueuelen 1000  (Ethernet)
        RX packets 10386  bytes 4206130 (4.0 MiB)
        RX errors 0  dropped 2  overruns 0  frame 0
        TX packets 3430  bytes 732263 (715.1 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 70  bytes 19394 (18.9 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 70  bytes 19394 (18.9 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.13  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::2359:b2d1:bf81:668a  prefixlen 64  scopeid 0x20<link>
        ether dc:a6:32:9e:6b:62  txqueuelen 1000  (Ethernet)
        RX packets 3  bytes 640 (640.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 125  bytes 41060 (40.0 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

네트워크 접속 및 인터넷 통신 기능 확인

◆ Ping

- Ping is a computer network administration software utility used to test the reachability of a host on an Internet Protocol (IP) network.
- It measures the round-trip time for messages sent from the originating host to a destination computer that are echoed back to the source.
- Test Ping
  - send ICMP message to google.com 5 times.

% ping google.com –c 5



```
root@raspberrypi:~# ping google.com -c 5
PING google.com (172.217.27.78) 56(84) bytes of data.
64 bytes from nrt12s15-in-f14.1e100.net (172.217.27.78): icmp_seq=1 ttl=51 time=34.4 ms
64 bytes from nrt12s15-in-f14.1e100.net (172.217.27.78): icmp_seq=2 ttl=51 time=34.4 ms
64 bytes from nrt12s15-in-f14.1e100.net (172.217.27.78): icmp_seq=3 ttl=51 time=34.0 ms
64 bytes from nrt12s15-in-f14.1e100.net (172.217.27.78): icmp_seq=4 ttl=51 time=34.4 ms
64 bytes from nrt12s15-in-f14.1e100.net (172.217.27.78): icmp_seq=5 ttl=51 time=34.4 ms

--- google.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 34.038/34.371/34.496/0.236 ms
```

한글 패키지 설치

- **Raspberry Pi에서 한글 표시를 위한 패키지 설치**

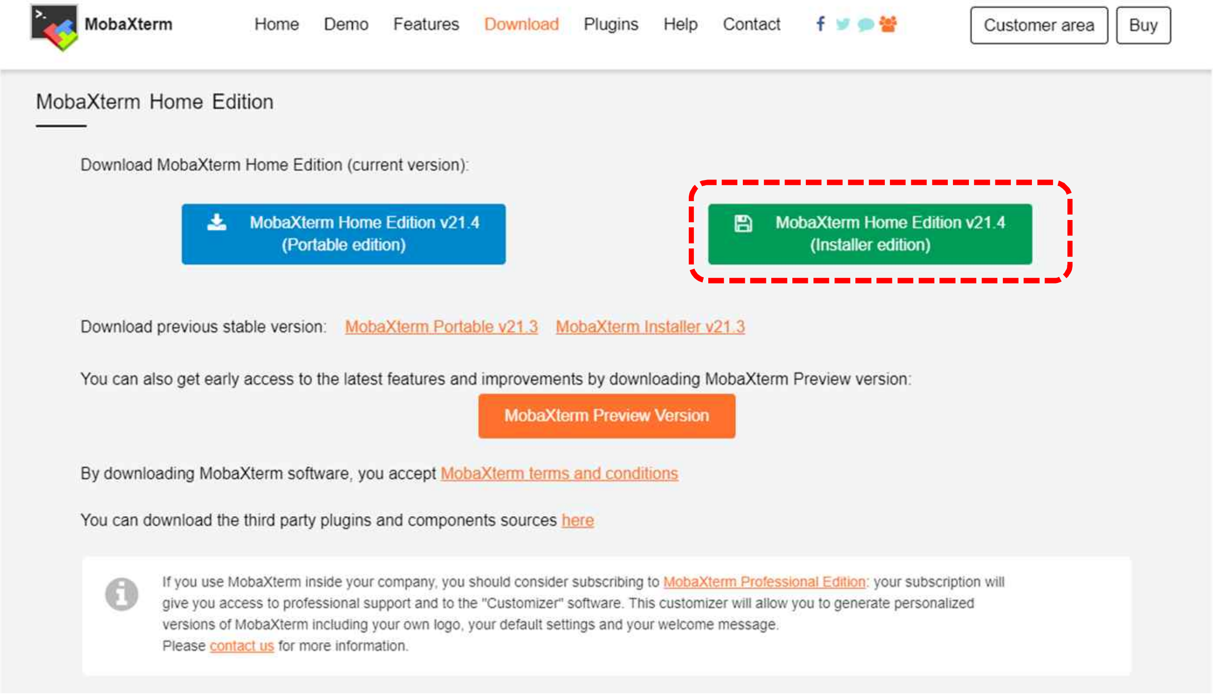
```
% sudo apt install –y fonts-unfonts-core
% apt-get install ibus-hangul
```

Raspberry Pi의 원격접속

원격 접속을 위하여 Desktop에 MobaXterm 설치

◆ MobaXterm

- https://mobaxterm.mobatek.net/download-home-edition.html
- installer edition을 설치



MobaXterm Home Edition

Download MobaXterm Home Edition (current version):

MobaXterm Home Edition v21.4 (Portable edition)

MobaXterm Home Edition v21.4 (Installer edition)

Download previous stable version: MobaXterm Portable v21.3 MobaXterm Installer v21.3

You can also get early access to the latest features and improvements by downloading MobaXterm Preview version:

MobaXterm Preview Version

By downloading MobaXterm software, you accept MobaXterm terms and conditions

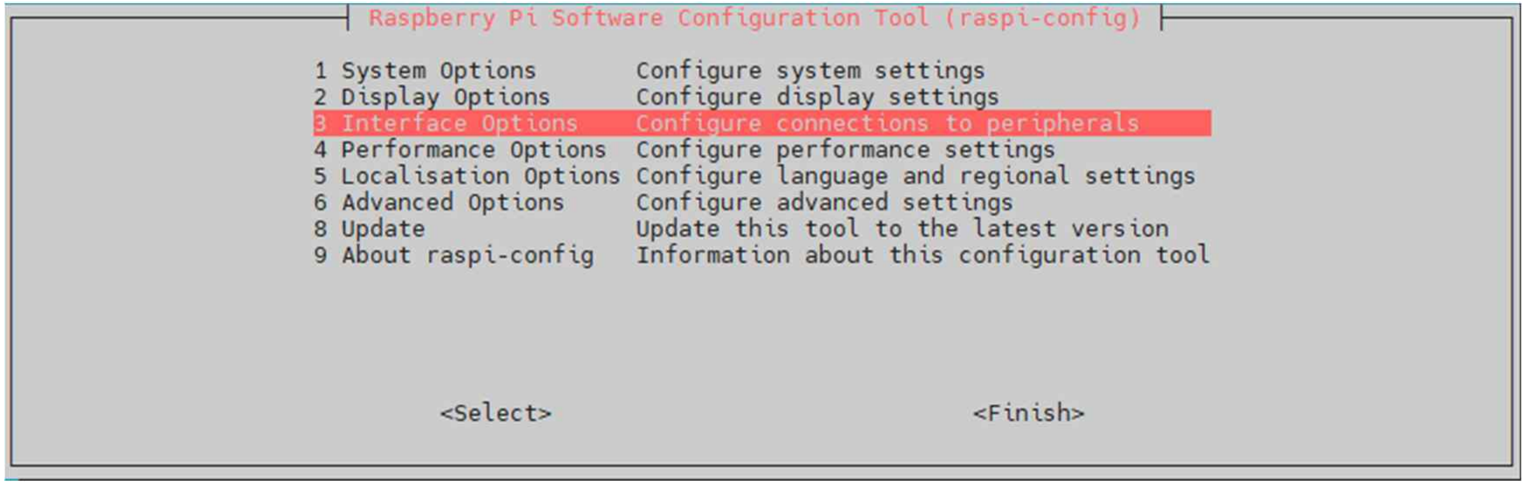
You can download the third party plugins and components sources here

If you use MobaXterm inside your company, you should consider subscribing to MobaXterm Professional Edition: your subscription will give you access to professional support and to the "Customizer" software. This customizer will allow you to generate personalized versions of MobaXterm including your own logo, your default settings and your welcome message. Please contact us for more information.

Checking Remote Connection

◆ Enable ssh on Raspberry Pi

% sudo raspi-config

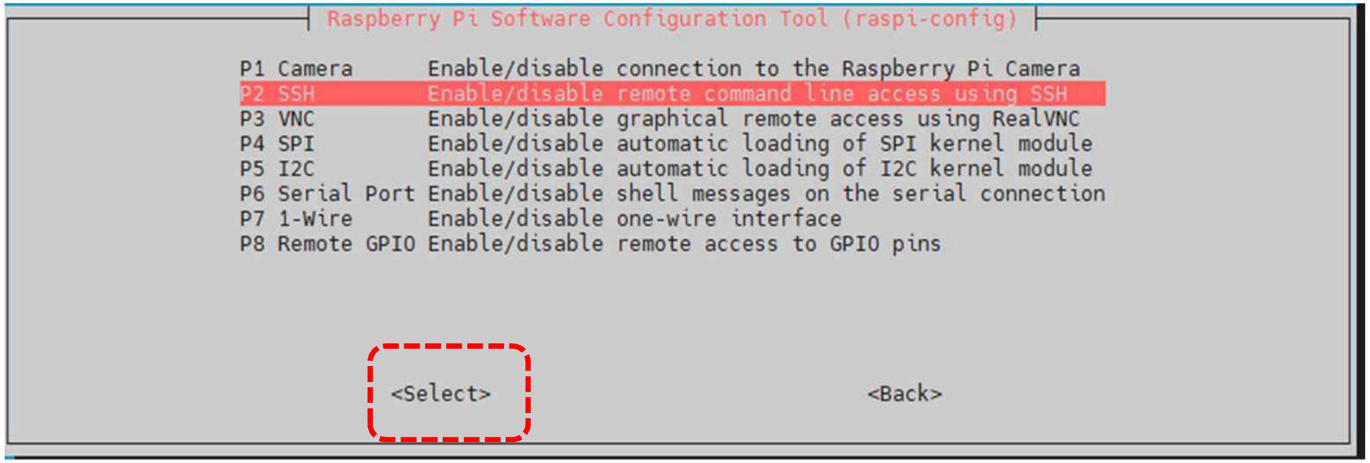


```
        Raspberry Pi Software Configuration Tool (raspi-config)

 1 System Options       Configure system settings
 2 Display Options      Configure display settings
 3 Interface Options    Configure connections to peripherals
 4 Performance Options  Configure performance settings
 5 Localisation Options Configure language and regional settings
 6 Advanced Options     Configure advanced settings
 8 Update               Update this tool to the latest version
 9 About raspi-config   Information about this configuration tool

           <Select>                             <Finish>
```

- Using Keyboard, select "3 Interface Options"

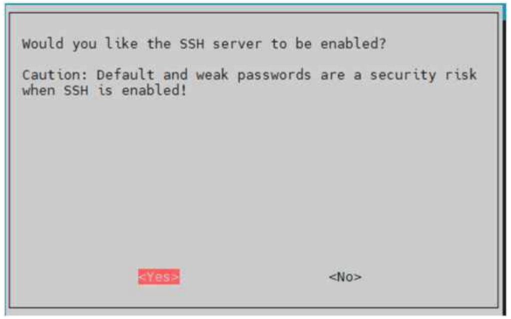
Checking Remote Connection

- Select P2 SSH



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

P1 Camera      Enable/disable connection to the Raspberry Pi Camera
P2 SSH         Enable/disable remote command line access using SSH
P3 VNC         Enable/disable graphical remote access using RealVNC
P4 SPI         Enable/disable automatic loading of SPI kernel module
P5 I2C         Enable/disable automatic loading of I2C kernel module
P6 Serial Port Enable/disable shell messages on the serial connection
P7 1-Wire      Enable/disable one-wire interface
P8 Remote GPIO Enable/disable remote access to GPIO pins
```

- Select Yes



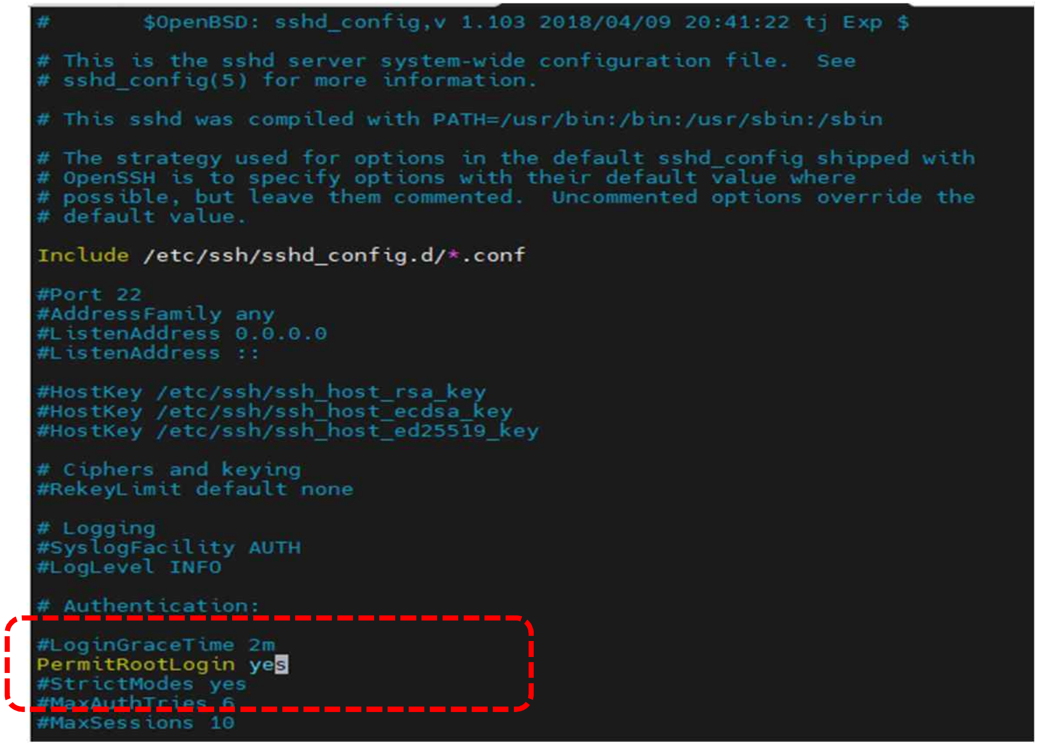
```
Would you like the SSH server to be enabled?

Caution: Default and weak passwords are a security risk
when SSH is enabled!
```

SSH를 Root 권한으로 접속할 수 있도록 설정

◆ /etc/ssh/sshd_config파일에서 다음 내용 수정

- PermitRootLogin **yes**

# service sshd restart

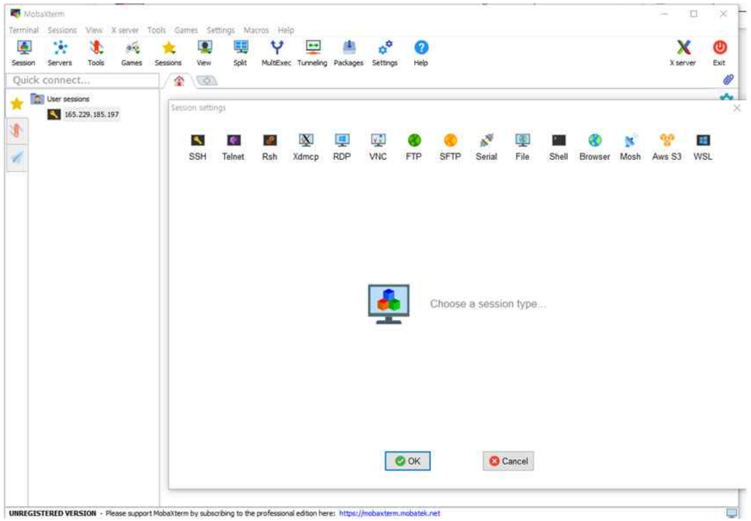


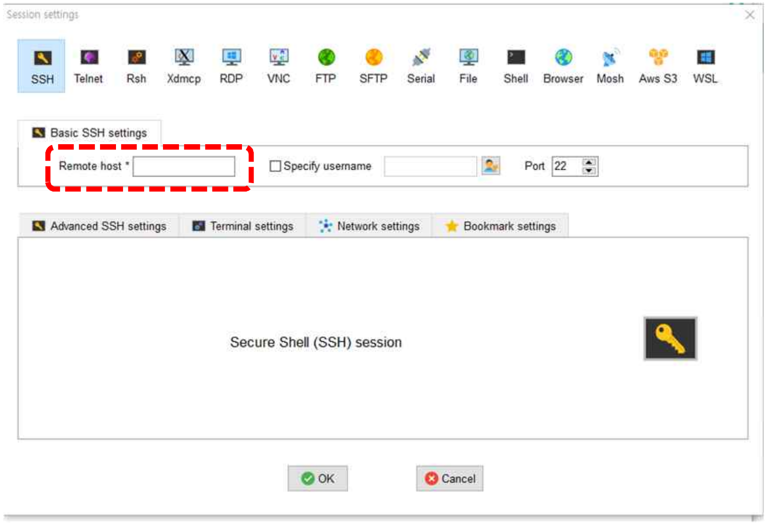
```
#       $OpenBSD: sshd_config,v 1.103 2018/04/09 20:41:22 tj Exp $

# This is the sshd server system-wide configuration file.  See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented.  Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
PermitRootLogin yes
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10
```

mobaxterm을 사용하여 원격 접속

◆ Remote Connection to Raspberry Pi

- execute mobaxterm
- create a new session -> SSH -> insert IP address of Remote host, and press OK button


MobaXterm
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
User sessions
165.229.185.197
Session settings
SSH Telnet Rsh Xdmcp RDP VNC FTP SFTP Serial File Shell Browser Mosh Aws S3 WSL
Choose a session type...
OK
Cancel
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net



Remote host *

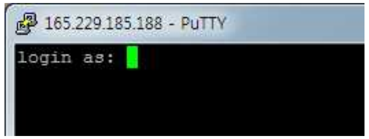
□ Specify username

Port 22

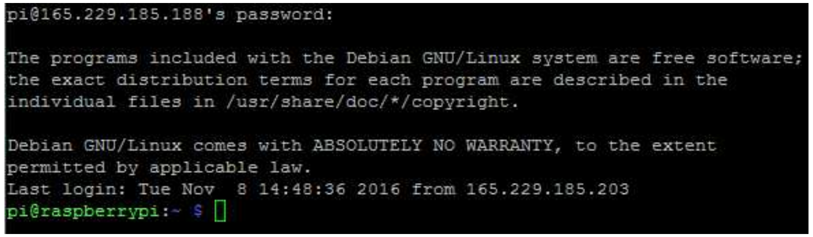
Root 권한으로 원격 접속 허용 설정

◆ Login

- login as : pi
- pi's password (예시): 1234



```
login as:
```



```
pi@165.229.185.188's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Nov  8 14:48:36 2016 from 165.229.185.203
pi@raspberrypi:~ $
```

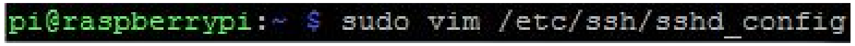
◆ Login as root

- set root password

  % sudo passwd root

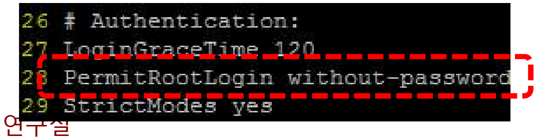
- edit sshd configuration

  % sudo vi(or vim) /etc/ssh/sshd_config

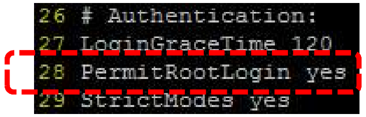


```
pi@raspberrypi:~ $ sudo vim /etc/ssh/sshd_config
```

- "PermitRootLogin without-password" 항을 "PermitRootLogin yes" 로 변경



```
# Authentication:
LoginGraceTime 120
PermitRootLogin without-password
StrictModes yes
```

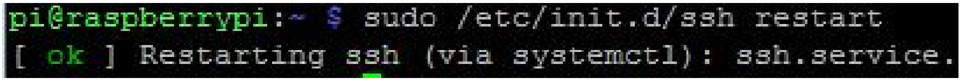


```
# Authentication:
LoginGraceTime 120
PermitRootLogin yes
StrictModes yes
```

Checking Remote Connection

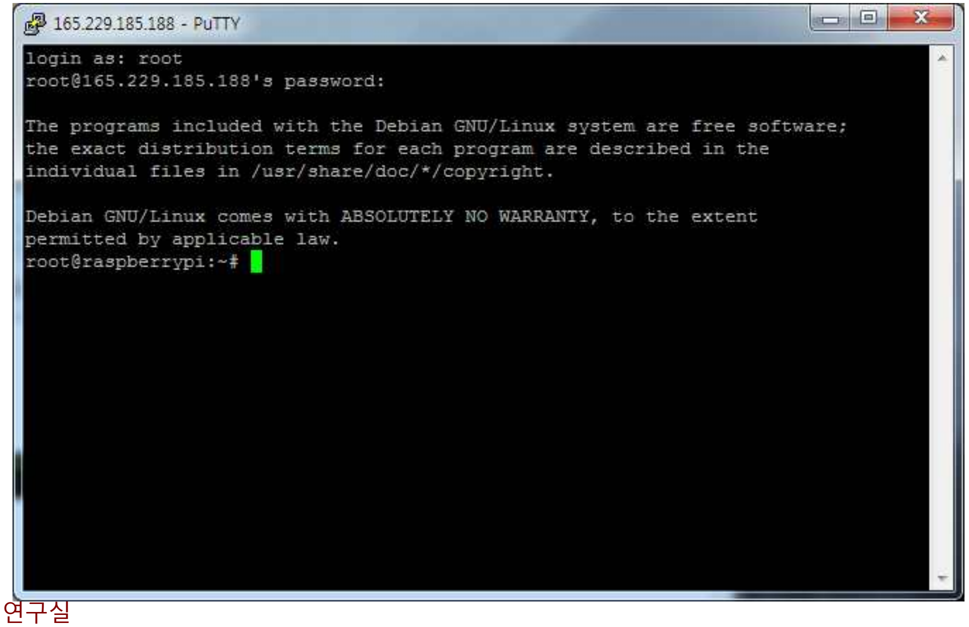
- ssh daemon restart

% sudo /etc/init.d/ssh restart



```
pi@raspberrypi:~ $ sudo /etc/init.d/ssh restart
[ ok ] Restarting ssh (via systemctl): ssh.service.
```

- mobaxterm을 사용하여 원격 접속, root로 login



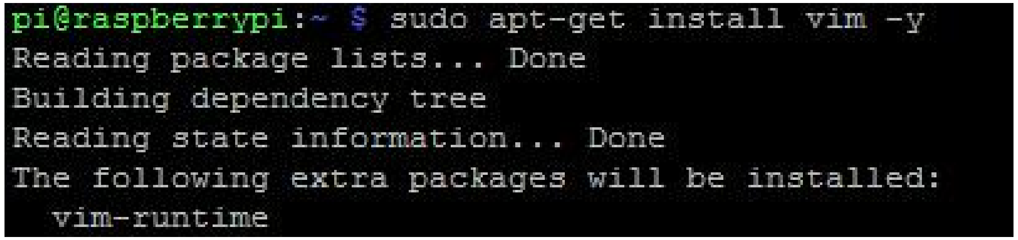
```
login as: root
root@165.229.185.188's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@raspberrypi:~#
```

Installation of Visual Editor (vim)

◆vi(m) editor

- The vi editor (short for visual editor) is a screen editor which is available on almost all Unix systems.
- vim editor an improved version of vi editor.
  It is more user friendly tool but still strong.
- Install vim

  % sudo apt-get install vim



```
pi@raspberrypi:~ $ sudo apt-get install vim -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  vim-runtime
```

gcc Compiler

◆gcc(GNU Compiler Collection)

- The GNU Compiler Collection (GCC) is a compiler system produced by the GNU Project supporting various programming languages.
- Originally named the GNU C Compiler, when it only handled the C programming language, GCC 1.0 was released in 1987.
- It was extended to compile C++ in December of that year. Front ends were later developed for Objective-C, Objective-C++, Fortran, Java, Ada, and Go among others.s

gcc Compiler

◆Compile options

- If you only want some of the stages of compilation, you can use -x (or filename suffixes) to tell gcc where to start, and one of the options -c, -S, or -E to say where gcc is to stop. Note that some combinations (for example, '-x cpp-output -E') instruct gcc to do nothing at all.
- -c
  - Compile or assemble the source files, but do not link. The linking stage simply is not done. The ultimate output is in the form of an object file for each source file.
  - By default, the object file name for a source file is made by replacing the suffix '.c', '.i', '.s', etc., with '.o'.
  - Unrecognized input files, not requiring compilation or assembly, are ignored.
- -S
  - Stop after the stage of compilation proper; do not assemble. The output is in the form of an assembler code file for each non-assembler input file specified.
  - By default, the assembler file name for a source file is made by replacing the suffix '.c', '.i', etc., with '.s'.
  - Input files that don't require compilation are ignored.
- -E
  - Stop after the preprocessing stage; do not run the compiler proper. The output is in the form of preprocessed source code, which is sent to the standard output.
  - Input files that don't require preprocessing are ignored.

gcc Compiler

◆ Compile options(cont.)

- -o file
  - Place output in file file. This applies to whatever sort of output is being produced, whether it be an executable file, an object file, an assembler file or preprocessed C code.
  - If -o is not specified, the default is to put an executable file in a.out, the object file for source.suffix in source.o, its assembler file in source.s, a precompiled header file in source.suffix.gch, and all preprocessed C source on standard output.
- -v
  - Print (on standard error output) the commands executed to run the stages of compilation, as verbose mode. Also print the version number of the compiler driver program and of the preprocessor and the compiler proper.
- -###
  - Like -v except the commands are not executed and arguments are quoted unless they contain only alphanumeric characters or ./-_. This is useful for shell scripts to capture the driver-generated command lines.
- --help
  - Print (on the standard output) a description of the command-line options understood by gcc. If the -v option is also specified then --help is also passed on to the various processes invoked by gcc, so that they can display the command-line options they accept. If the -Wextra option has also been specified (prior to the --help option), then command-line options that have no documentation associated with them are also displayed.

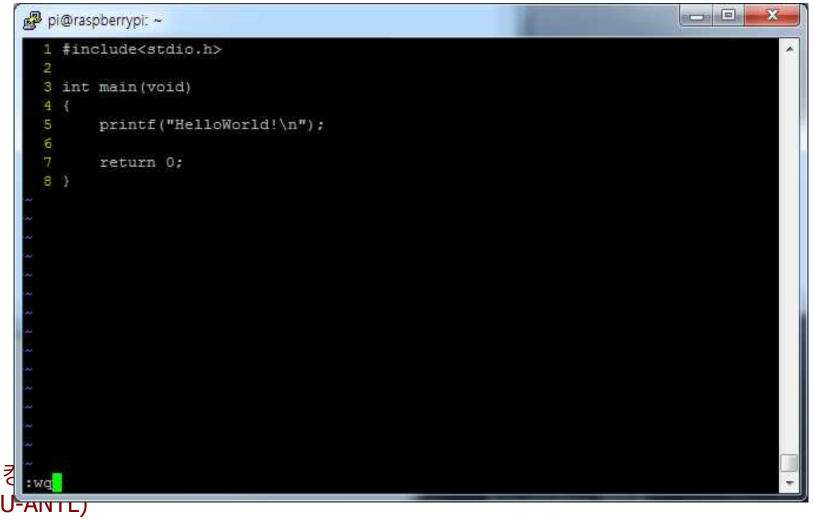
Hello World C program on Raspberry Pi

◆ Hello World C program (1)

- Using vi(m) editor and gcc compiler, let make simple program that print strings "Hello World" on the screen.
- Write C Program Source File

  % vi HelloWorld.c

  - type '**i**' to be insert mode, and write this codes
  - When you finished the writing codes, press **:wq** and Enter
  - 'w' means write, and 'q' means quit editor.



```
#include<stdio.h>

int main(void)
{
    printf("HelloWorld!\n");

    return 0;
}
~
:wq
```

Hello World C program on Raspberry Pi

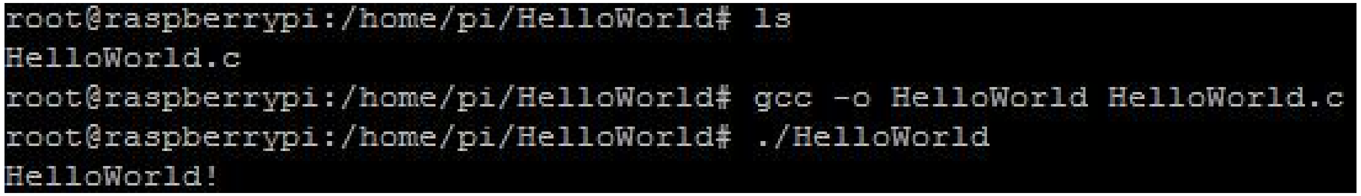
◆ Hello World C program (2)

- Compile

  % gcc –o HelloWorld HelloWorld.c

  - You can see HelloWorld binary
    File using ls command

  % ./HelloWorld



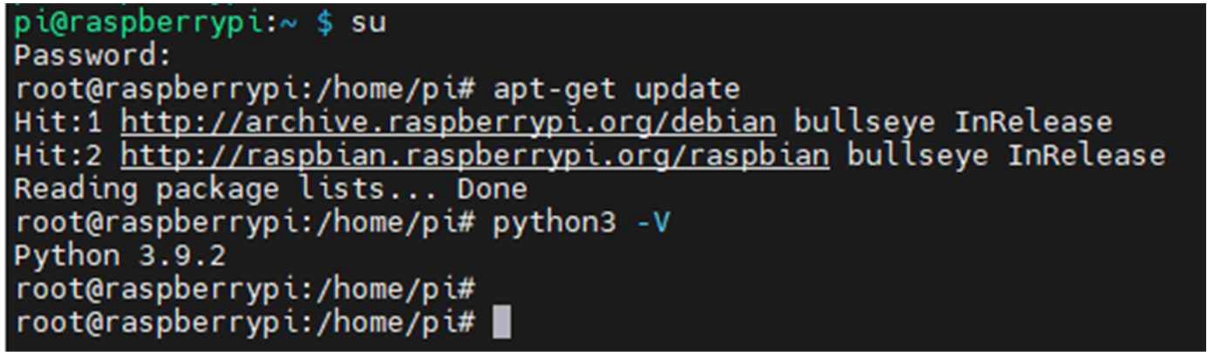
```
root@raspberrypi:/home/pi/HelloWorld# ls
HelloWorld.c
root@raspberrypi:/home/pi/HelloWorld# gcc -o HelloWorld HelloWorld.c
root@raspberrypi:/home/pi/HelloWorld# ./HelloWorld
HelloWorld!
```

Raspberry Pi에 설치된
Python 확인 및 프로그래밍

Raspberry Pi에 Python 설치 확인

◆ Raspbian 설치에서 기본적으로 Python 포함

- %su 명령어를 사용하여 super-user mode로 실행
- #python3 –V 명령어로 파이썬 설치 상황 확인 (V는 대문자 !!)

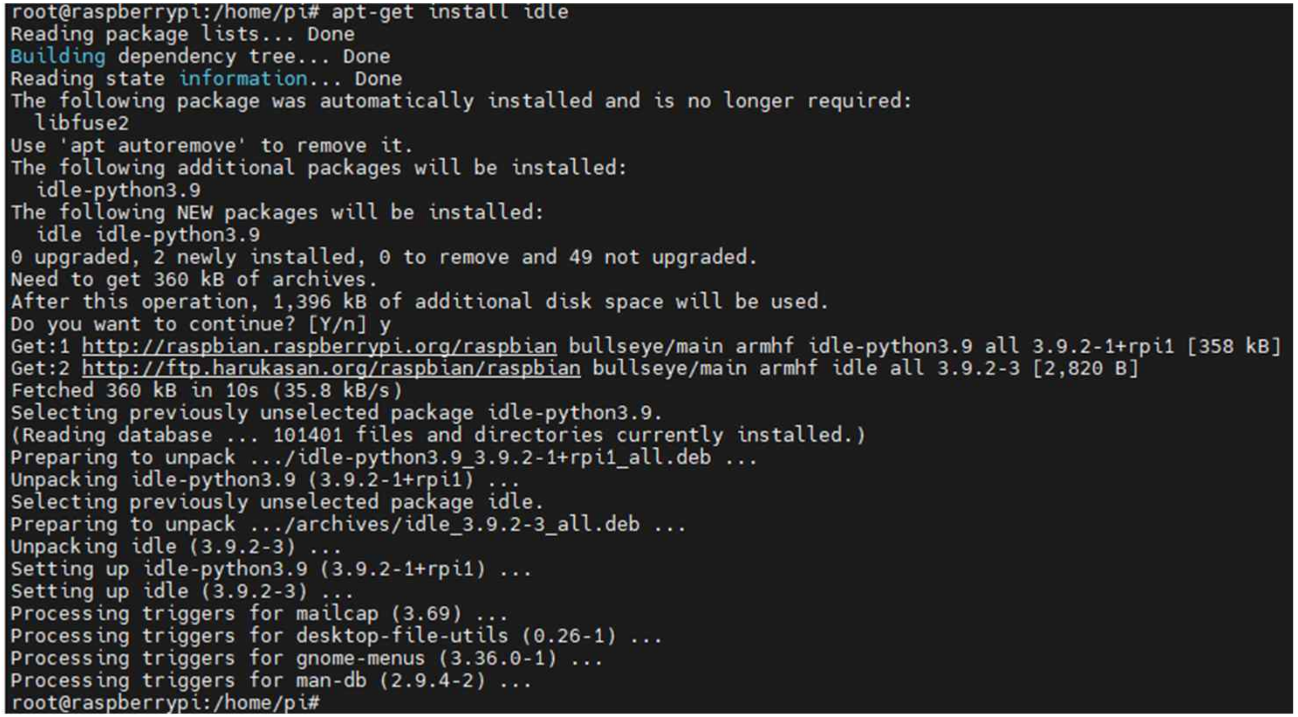


```
pi@raspberrypi:~ $ su
Password:
root@raspberrypi:/home/pi# apt-get update
Hit:1 http://archive.raspberrypi.org/debian bullseye InRelease
Hit:2 http://raspbian.raspberrypi.org/raspbian bullseye InRelease
Reading package lists... Done
root@raspberrypi:/home/pi# python3 -V
Python 3.9.2
root@raspberrypi:/home/pi#
root@raspberrypi:/home/pi# 
```

Raspberry Pi에 Python IDLE 설치

◆ Raspberry Pi에 Python IDLE (Integrated Development Environment) 설치

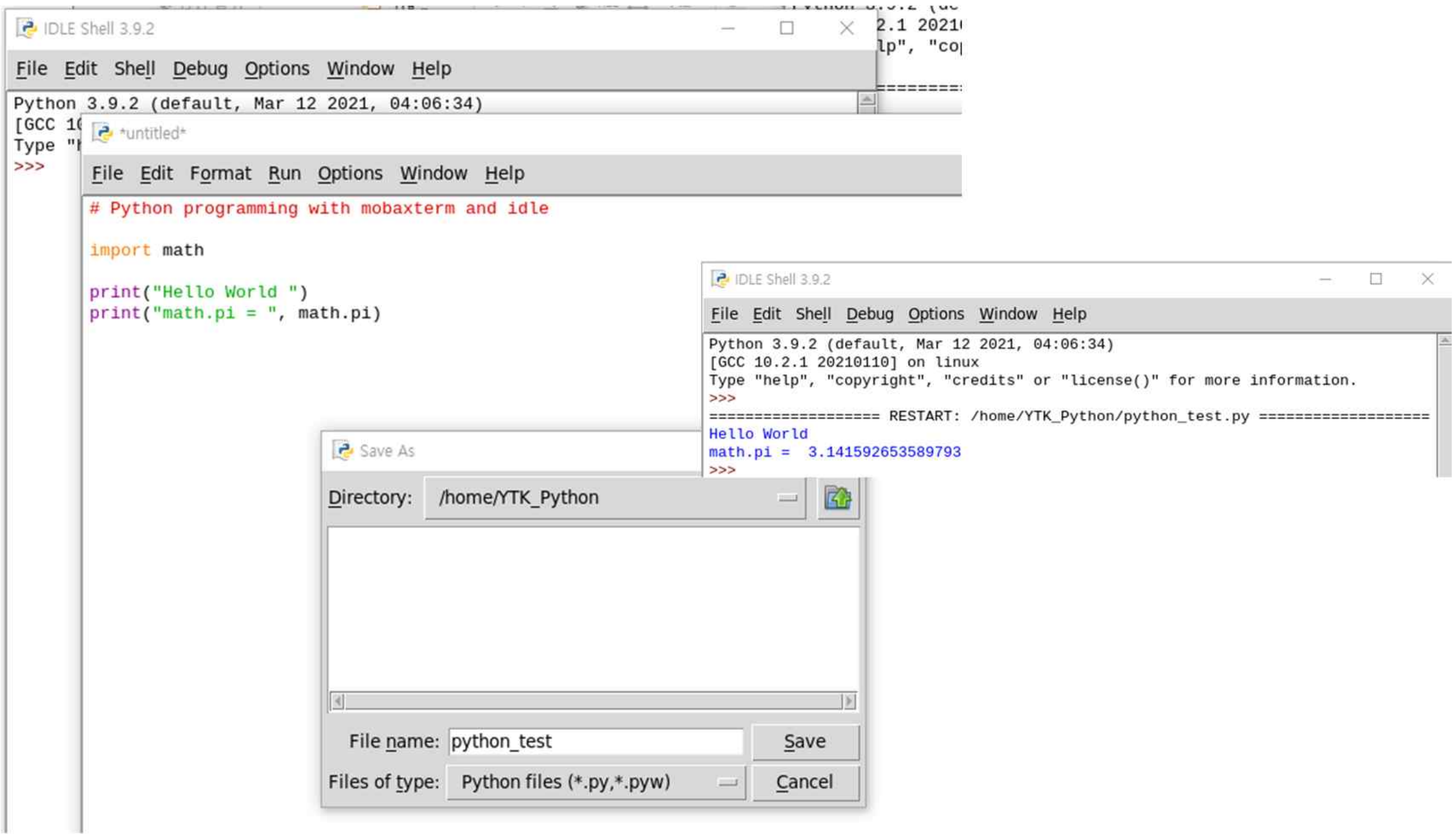
#apt-get install idle



```
root@raspberrypi:/home/pi# apt-get install idle
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following package was automatically installed and is no longer required:
  libfuse2
Use 'apt autoremove' to remove it.
The following additional packages will be installed:
  idle-python3.9
The following NEW packages will be installed:
  idle idle-python3.9
0 upgraded, 2 newly installed, 0 to remove and 49 not upgraded.
Need to get 360 kB of archives.
After this operation, 1,396 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://raspbian.raspberrypi.org/raspbian bullseye/main armhf idle-python3.9 all 3.9.2-1+rpi1 [358 kB]
Get:2 http://ftp.harukasan.org/raspbian/raspbian bullseye/main armhf idle all 3.9.2-3 [2,820 B]
Fetched 360 kB in 10s (35.8 kB/s)
Selecting previously unselected package idle-python3.9.
(Reading database ... 101401 files and directories currently installed.)
Preparing to unpack .../idle-python3.9_3.9.2-1+rpi1_all.deb ...
Unpacking idle-python3.9 (3.9.2-1+rpi1) ...
Selecting previously unselected package idle.
Preparing to unpack .../archives/idle_3.9.2-3_all.deb ...
Unpacking idle (3.9.2-3) ...
Setting up idle-python3.9 (3.9.2-1+rpi1) ...
Setting up idle (3.9.2-3) ...
Processing triggers for mailcap (3.69) ...
Processing triggers for desktop-file-utils (0.26-1) ...
Processing triggers for gnome-menus (3.36.0-1) ...
Processing triggers for man-db (2.9.4-2) ...
root@raspberrypi:/home/pi#
```

Raspberry Pi를 원격으로 접속한 환경에서 간단한 Python 프로그램 작성 및 실행

◆ mobaxterm에서 root로 접속한 후, 파이썬 IDLE 실행


IDLE Shell 3.9.2
File Edit Shell Debug Options Window Help
untitled
File Edit Format Run Options Window Help
Save As
Save
Cancel

```
Python 3.9.2 (default, Mar 12 2021, 04:06:34)
```

```
# Python programming with mobaxterm and idle

import math

print("Hello World ")
print("math.pi = ", math.pi)
```

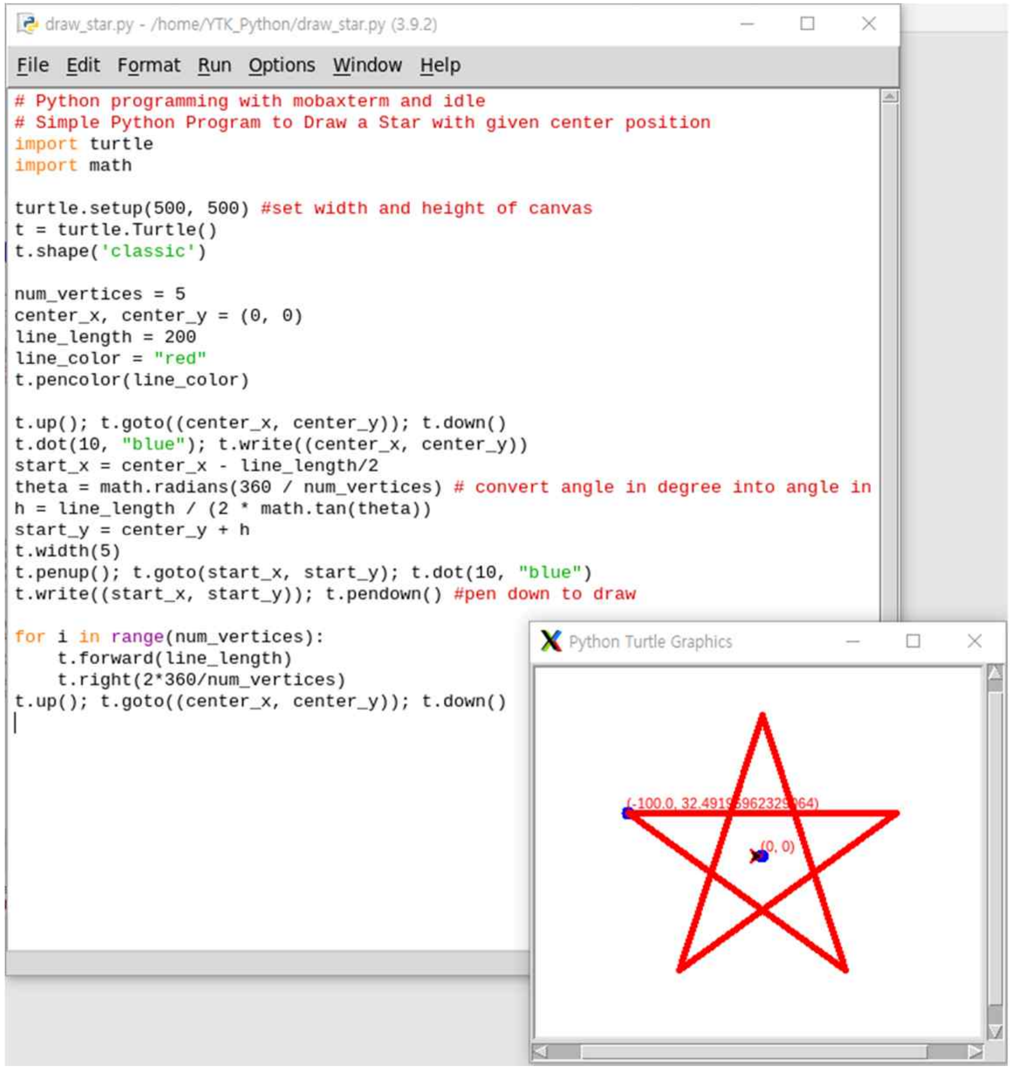
```
Python 3.9.2 (default, Mar 12 2021, 04:06:34)
[GCC 10.2.1 20210110] on linux
Type "help", "copyright", "credits" or "license()" for more information.
>>>
=================== RESTART: /home/YTK_Python/python_test.py ===================
Hello World
math.pi =  3.141592653589793
>>>
```

Directory: /home/YTK_Python

File name: python_test

Files of type: Python files (*.py,*.pyw)

mobaxterm의 원격 접속 기반 turtle graphic

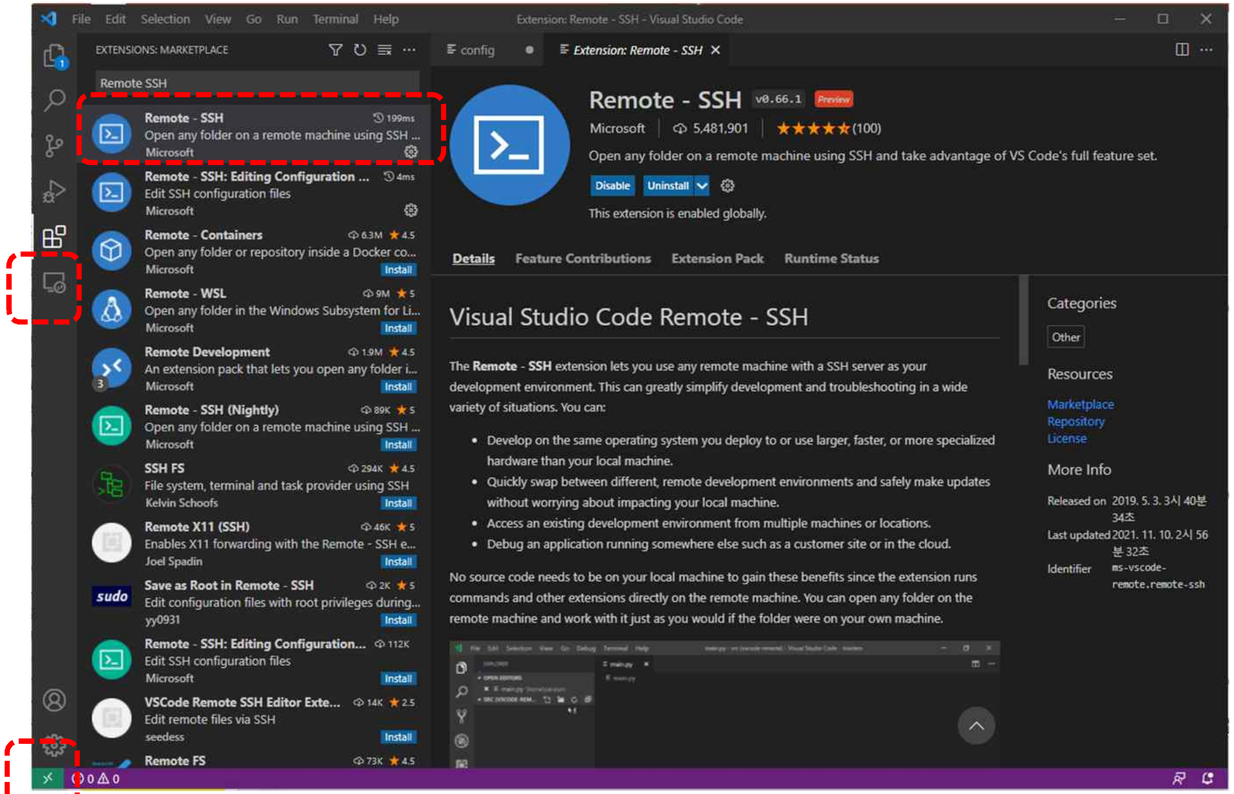

draw_star.py - /home/YTK_Python/draw_star.py (3.9.2)
File Edit Format Run Options Window Help
Python Turtle Graphics
(-100.0, 32.49196962329064)
(0, 0)

```python
# Python programming with mobaxterm and idle
# Simple Python Program to Draw a Star with given center position
import turtle
import math

turtle.setup(500, 500) #set width and height of canvas
t = turtle.Turtle()
t.shape('classic')

num_vertices = 5
center_x, center_y = (0, 0)
line_length = 200
line_color = "red"
t.pencolor(line_color)

t.up(); t.goto((center_x, center_y)); t.down()
t.dot(10, "blue"); t.write((center_x, center_y))
start_x = center_x - line_length/2
theta = math.radians(360 / num_vertices) # convert angle in degree into angle in
h = line_length / (2 * math.tan(theta))
start_y = center_y + h
t.width(5)
t.penup(); t.goto(start_x, start_y); t.dot(10, "blue")
t.write((start_x, start_y)); t.pendown() #pen down to draw

for i in range(num_vertices):
    t.forward(line_length)
    t.right(2*360/num_vertices)
t.up(); t.goto((center_x, center_y)); t.down()
```

VScode로 Raspberry Pi 원격 접속 및 파이썬 프로그래밍

VScode 기반 Raspberry Pi 원격 접속

◆ VScode에서 Remote-SSH를 검색하고, 설치



Remote SSH

- Remote - SSH — 199ms — Open any folder on a remote machine using SSH ... — Microsoft
- Remote - SSH: Editing Configuration ... — 4ms — Edit SSH configuration files — Microsoft
- Remote - Containers — 6.3M ★ 4.5 — Open any folder or repository inside a Docker co... — Microsoft — Install
- Remote - WSL — 9M ★ 5 — Open any folder in the Windows Subsystem for Li... — Microsoft — Install
- Remote Development — 1.9M ★ 4.5 — An extension pack that lets you open any folder i... — Microsoft — Install
- Remote - SSH (Nightly) — 89K ★ 5 — Open any folder on a remote machine using SSH ... — Microsoft — Install
- SSH FS — 294K ★ 4.5 — File system, terminal and task provider using SSH — Kelvin Schoofs — Install
- Remote X11 (SSH) — 46K ★ 5 — Enables X11 forwarding with the Remote - SSH e... — Joel Spadin — Install
- Save as Root in Remote - SSH — 2K ★ 5 — Edit configuration files with root privileges during... — yy0931 — Install
- Remote - SSH: Editing Configuration... — 112K — Edit SSH configuration files — Microsoft — Install
- VSCode Remote SSH Editor Exte... — 14K ★ 2.5 — Edit remote files via SSH — seedess — Install
- Remote FS — 73K ★ 4.5

Remote - SSH v0.66.1 Preview

Microsoft | 5,481,901 | ★★★★★(100)

Open any folder on a remote machine using SSH and take advantage of VS Code's full feature set.

This extension is enabled globally.

Visual Studio Code Remote - SSH

The **Remote - SSH** extension lets you use any remote machine with a SSH server as your development environment. This can greatly simplify development and troubleshooting in a wide variety of situations. You can:

- 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.

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

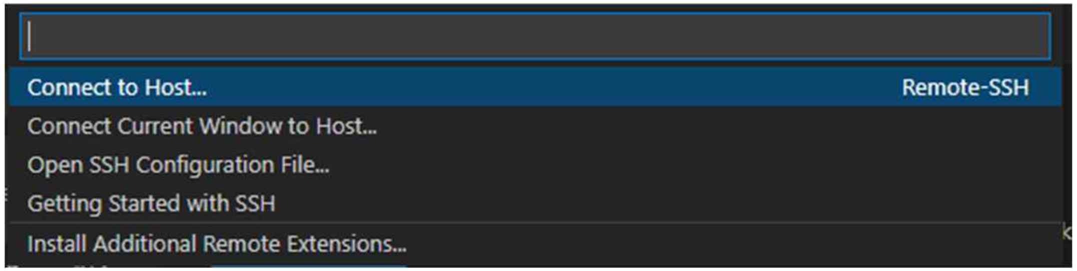
Marketplace
Repository
License

More Info

Released on	2019. 5. 3. 3시 40분 34초
Last updated	2021. 11. 10. 2시 56분 32초
Identifier	ms-vscode-remote.remote-ssh

Vscode의 Remote-SSH 설치

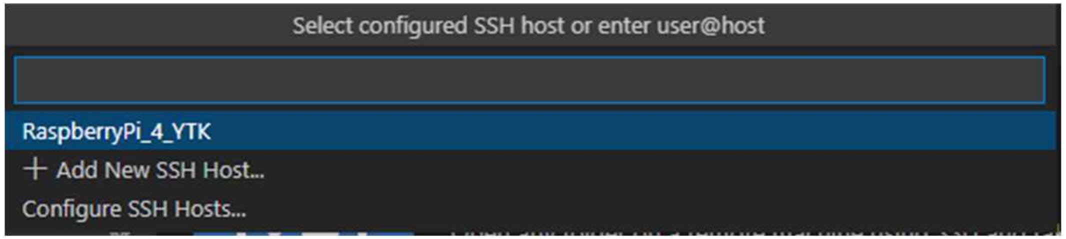
◆ VScode의 좌측하단에 있는 "><" 아이콘을 클릭



Connect to Host... Remote-SSH
Connect Current Window to Host...
Open SSH Configuration File...
Getting Started with SSH
Install Additional Remote Extensions...

◆ Connect to Host를 클릭

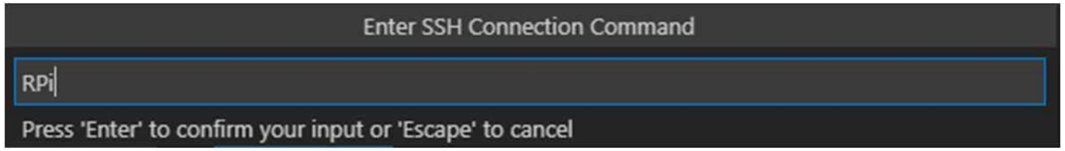
- 이미 설정되어 있는 remote host를 선택/클릭 하거나,
- Add New SSH Host를 클릭



Select configured SSH host or enter user@host
RaspberryPi_4_YTK
+ Add New SSH Host...
Configure SSH Hosts...

Vscode의 Remote-SSH 설치

◆ 새로운 Remote Host 이름을 입력

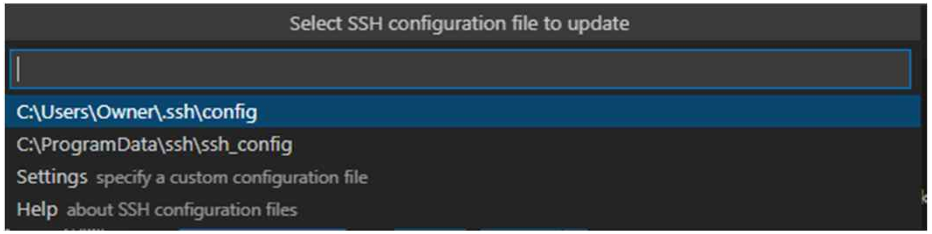


Enter SSH Connection Command

RPi

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

◆ Select SSH configuration file to update에서 C:\Users\Owner\.ssh\config 항목을 클릭한 후, Open Config를 클릭



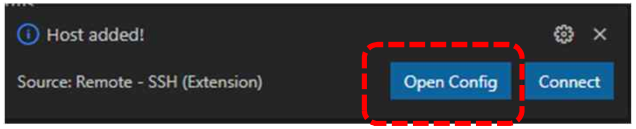
Select SSH configuration file to update

C:\Users\Owner\.ssh\config

C:\ProgramData\ssh\ssh_config

Settings specify a custom configuration file

Help about SSH configuration files



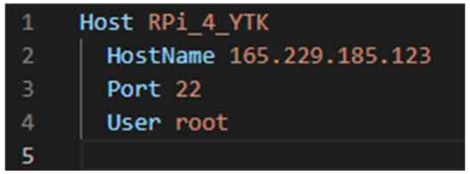
Host added!

Source: Remote - SSH (Extension)

Vscode의 Remote-SSH 설치

◆ SSH Config

- HostName에는 Raspberry Pi의 IP 주소를 입력
- Port는 22, User는 root를 입력
- File 메뉴의 Save를 클릭

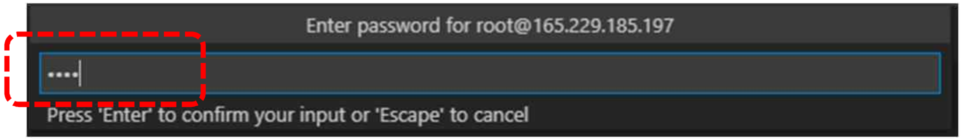


```
Host RPi_4_YTK
  HostName 165.229.185.123
  Port 22
  User root

```

Vscode의 Remote-SSH를 사용한 원격 호스트 접속

◆ 원격 호스트 접속

- VScode의 좌측하단에 있는 "><" 아이콘을 클릭
- 구성되어 있는 remote host를 선택
- Password를 입력



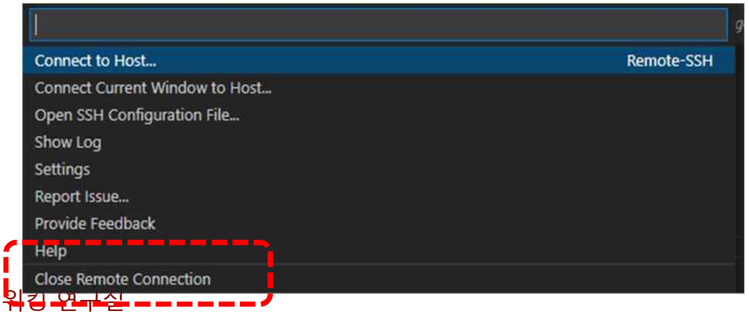
Enter password for root@165.229.185.197

••••

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

◆ 원격 호스트 접속 종료

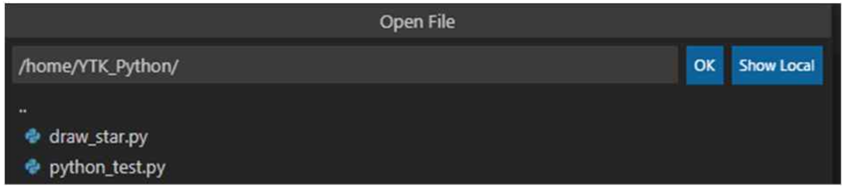
- VScode의 좌측하단에 있는 "><" 아이콘을 클릭; Close Remote Connection



Connect to Host... Remote-SSH
Connect Current Window to Host...
Open SSH Configuration File...
Show Log
Settings
Report Issue...
Provide Feedback
Help
Close Remote Connection

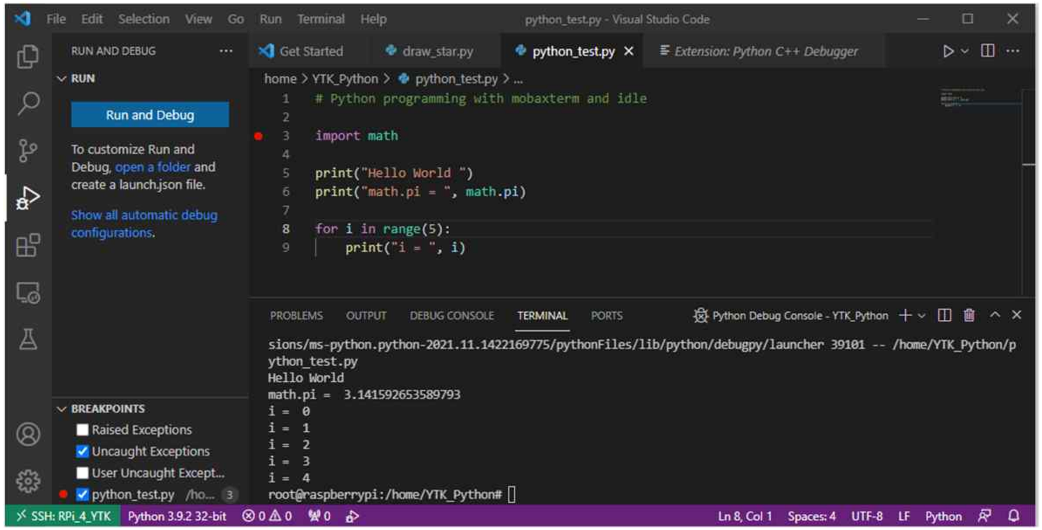
VScode/SSH를 사용하여 파이썬 프로그램 작성

◆ 원격 컴퓨터의 파일 검색



```
/home/YTK_Python/
..
draw_star.py
python_test.py
```

◆ 파이썬 프로그램의 편집, 실행 및 디버깅


File Edit Selection View Go Run Terminal Help
python_test.py - Visual Studio Code
RUN AND DEBUG
RUN
Run and Debug
Get Started
draw_star.py
python_test.py
Extension: Python C++ Debugger
home > YTK_Python > python_test.py > ...
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python Debug Console - YTK_Python
BREAKPOINTS
SSH: RPi_4_YTK
Python 3.9.2 32-bit
Ln 8, Col 1 Spaces: 4 UTF-8 LF Python

To customize Run and Debug, open a folder and create a launch.json file.

Show all automatic debug configurations.

BREAKPOINTS
- ☐ Raised Exceptions
- ☑ Uncaught Exceptions
- ☐ User Uncaught Except...
- ☑ python_test.py /ho... 3

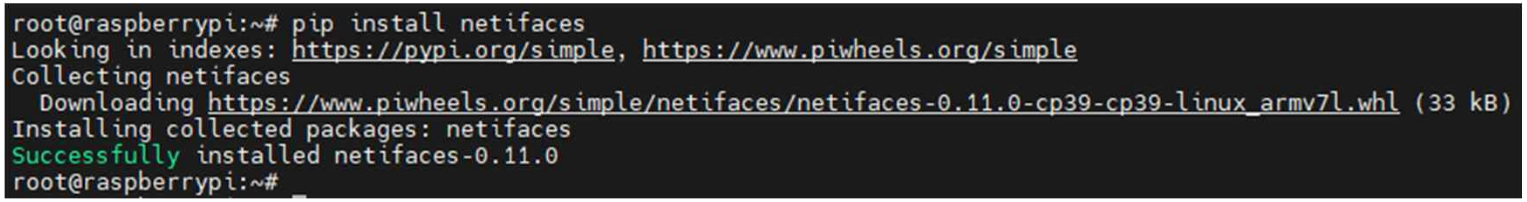
```python
# Python programming with mobaxterm and idle

import math

print("Hello World ")
print("math.pi = ", math.pi)

for i in range(5):
    print("i = ", i)
```

```
sions/ms-python.python-2021.11.1422169775/pythonFiles/lib/python/debugpy/launcher 39101 -- /home/YTK_Python/p
ython_test.py
Hello World
math.pi =  3.141592653589793
i =  0
i =  1
i =  2
i =  3
i =  4
root@raspberrypi:/home/YTK_Python# 
```

Raspberry Pi의 IP 주소 확인

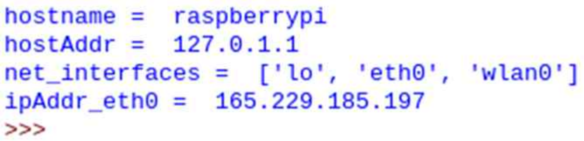
◆ Python의 netifaces 확장모듈 설치

- https://pypi.org/project/netifaces/



```
root@raspberrypi:~# pip install netifaces
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
Collecting netifaces
  Downloading https://www.piwheels.org/simple/netifaces/netifaces-0.11.0-cp39-cp39-linux_armv7l.whl (33 kB)
Installing collected packages: netifaces
Successfully installed netifaces-0.11.0
root@raspberrypi:~# _
```

```
# get IP address of eth0 in Raspberry Pi
import socket, sys, os
import netifaces as ni

hostname = socket.gethostname()
hostAddr = socket.gethostbyname(hostname)
print("hostname = ", hostname)
print("hostAddr = ", hostAddr)

net_interfaces = ni.interfaces()
print("net_interfaces = ", net_interfaces)

ipAddr_eth0 = ni.ifaddresses('eth0')[ni.AF_INET][0]['addr']
print("ipAddr_eth0 = ", ipAddr_eth0)
```



```
hostname =  raspberrypi
hostAddr =  127.0.1.1
net_interfaces =  ['lo', 'eth0', 'wlan0']
ipAddr_eth0 =  165.229.185.197
>>>
```

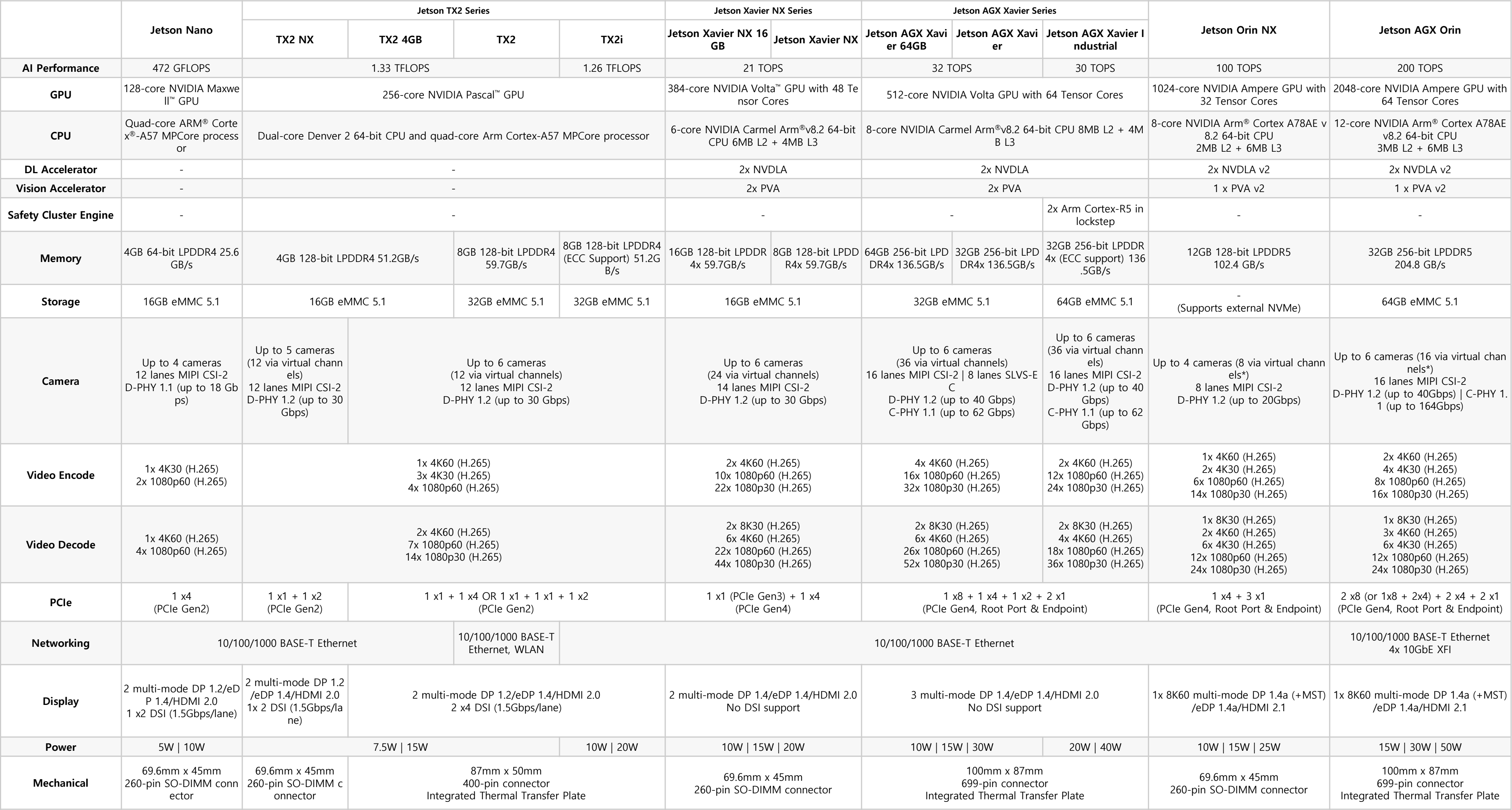
Jetson Developer Kit (Nano/TX2/Xavier)

Jetson Developer Kit

◆ Jetson Developer Kit

	Jetson Nano Developer Kit	Jetson TX2 Developer Kit	Jetson Xavier NX Developer Kit	Jetson AGX Xavier Developer Kit
AI Performance	0.5 TFLOPS (FP16)	1.3 TFLOPS (FP16)	6 TFLOPS (FP16) 21 TOPS (INT8)	5.5-11 TFLOPS (FP16) 20-32 TOPS (INT8)
GPU	128-core NVIDIA Maxwell™ GPU	256-core NVIDIA Pascal™ GPU architecture with 256 NVIDIA CUDA cores	NVIDIA Volta architecture with 384 NVIDIA CUDA® cores and 48 Tensor cores	512-Core Volta GPU with Tensor Cores
CPU	Quad-core ARM A57 @ 1.43 GHz	Dual-Core NVIDIA Denver 2 64-Bit CPU Quad-Core ARM® Cortex®-A57 MPCore	6-core NVIDIA Carmel ARM®v8.2 64-bit CPU 6 MB L2 + 4 MB L3	8-Core ARM v8.2 64-Bit CPU, 8 MB L2 + 4 MB L3
Memory	2GB/ 4GB 64-bit LPDDR4 25.6 GB/s	8GB 128-bit LPDDR4 1866 MHz - 59.7 GB/s	8 GB 128-bit LPDDR4x @ 51.2GB/s	32 GB 256-Bit LPDDR4x | 137 GB/s
Power Consumption	5-10W	7.5-15W	10-15W	10-30W

(source: https://www.nvidia.com/en-us/
https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/)

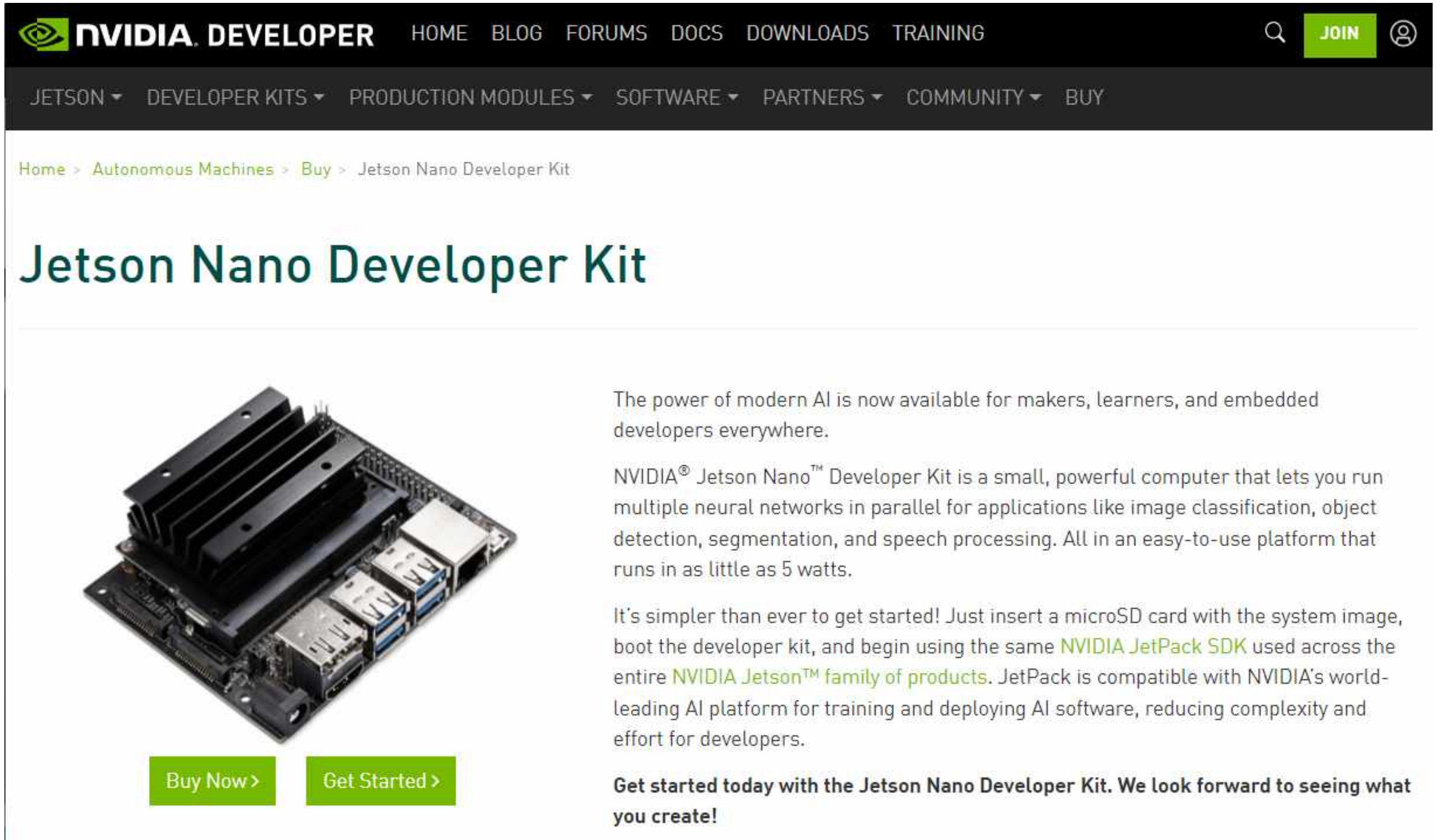


	Jetson Nano	Jetson TX2 Series				Jetson Xavier NX Series		Jetson AGX Xavier Series			Jetson Orin NX	Jetson AGX Orin
		TX2 NX	TX2 4GB	TX2	TX2i	Jetson Xavier NX 16 GB	Jetson Xavier NX	Jetson AGX Xavier 64GB	Jetson AGX Xavier	Jetson AGX Xavier Industrial		
AI Performance	472 GFLOPS	1.33 TFLOPS			1.26 TFLOPS	21 TOPS		32 TOPS		30 TOPS	100 TOPS	200 TOPS
GPU	128-core NVIDIA Maxwell™ GPU	256-core NVIDIA Pascal™ GPU				384-core NVIDIA Volta™ GPU with 48 Tensor Cores		512-core NVIDIA Volta GPU with 64 Tensor Cores			1024-core NVIDIA Ampere GPU with 32 Tensor Cores	2048-core NVIDIA Ampere GPU with 64 Tensor Cores
CPU	Quad-core ARM® Cortex®-A57 MPCore processor	Dual-core Denver 2 64-bit CPU and quad-core Arm Cortex-A57 MPCore processor				6-core NVIDIA Carmel Arm®v8.2 64-bit CPU 6MB L2 + 4MB L3		8-core NVIDIA Carmel Arm®v8.2 64-bit CPU 8MB L2 + 4MB L3			8-core NVIDIA Arm® Cortex A78AE v8.2 64-bit CPU 2MB L2 + 6MB L3	12-core NVIDIA Arm® Cortex A78AE v8.2 64-bit CPU 3MB L2 + 6MB L3
DL Accelerator	-	-				2x NVDLA		2x NVDLA			2x NVDLA v2	2x NVDLA v2
Vision Accelerator	-	-				2x PVA		2x PVA			1 x PVA v2	1 x PVA v2
Safety Cluster Engine	-	-				-		-		2x Arm Cortex-R5 in lockstep	-	-
Memory	4GB 64-bit LPDDR4 25.6GB/s	4GB 128-bit LPDDR4 51.2GB/s		8GB 128-bit LPDDR4 59.7GB/s	8GB 128-bit LPDDR4 (ECC Support) 51.2GB/s	16GB 128-bit LPDDR4x 59.7GB/s	8GB 128-bit LPDDR4x 59.7GB/s	64GB 256-bit LPDDR4x 136.5GB/s	32GB 256-bit LPDDR4x 136.5GB/s	32GB 256-bit LPDDR4x (ECC support) 136.5GB/s	12GB 128-bit LPDDR5 102.4 GB/s	32GB 256-bit LPDDR5 204.8 GB/s
Storage	16GB eMMC 5.1	16GB eMMC 5.1		32GB eMMC 5.1	32GB eMMC 5.1	16GB eMMC 5.1		32GB eMMC 5.1		64GB eMMC 5.1	- (Supports external NVMe)	64GB eMMC 5.1
Camera	Up to 4 cameras 12 lanes MIPI CSI-2 D-PHY 1.1 (up to 18 Gbps)	Up to 5 cameras (12 via virtual channels) 12 lanes MIPI CSI-2 D-PHY 1.2 (up to 30 Gbps)	Up to 6 cameras (12 via virtual channels) 12 lanes MIPI CSI-2 D-PHY 1.2 (up to 30 Gbps)			Up to 6 cameras (24 via virtual channels) 14 lanes MIPI CSI-2 D-PHY 1.2 (up to 30 Gbps)		Up to 6 cameras (36 via virtual channels) 16 lanes MIPI CSI-2 | 8 lanes SLVS-EC D-PHY 1.2 (up to 40 Gbps) C-PHY 1.1 (up to 62 Gbps)		Up to 6 cameras (36 via virtual channels) 16 lanes MIPI CSI-2 D-PHY 1.2 (up to 40 Gbps) C-PHY 1.1 (up to 62 Gbps)	Up to 4 cameras (8 via virtual channels*) 8 lanes MIPI CSI-2 D-PHY 1.2 (up to 20Gbps)	Up to 6 cameras (16 via virtual channels*) 16 lanes MIPI CSI-2 D-PHY 1.2 (up to 40Gbps) | C-PHY 1.1 (up to 164Gbps)
Video Encode	1x 4K30 (H.265) 2x 1080p60 (H.265)	1x 4K60 (H.265) 3x 4K30 (H.265) 4x 1080p60 (H.265)				2x 4K60 (H.265) 10x 1080p60 (H.265) 22x 1080p30 (H.265)		4x 4K60 (H.265) 16x 1080p60 (H.265) 32x 1080p30 (H.265)		2x 4K60 (H.265) 12x 1080p60 (H.265) 24x 1080p30 (H.265)	1x 4K60 (H.265) 2x 4K30 (H.265) 6x 1080p60 (H.265) 14x 1080p30 (H.265)	2x 4K60 (H.265) 4x 4K30 (H.265) 8x 1080p60 (H.265) 16x 1080p30 (H.265)
Video Decode	1x 4K60 (H.265) 4x 1080p60 (H.265)	2x 4K60 (H.265) 7x 1080p60 (H.265) 14x 1080p30 (H.265)				2x 8K30 (H.265) 6x 4K60 (H.265) 22x 1080p60 (H.265) 44x 1080p30 (H.265)		2x 8K30 (H.265) 6x 4K60 (H.265) 26x 1080p60 (H.265) 52x 1080p30 (H.265)		2x 8K30 (H.265) 4x 4K60 (H.265) 18x 1080p60 (H.265) 36x 1080p30 (H.265)	1x 8K30 (H.265) 2x 4K60 (H.265) 6x 4K30 (H.265) 12x 1080p60 (H.265) 24x 1080p30 (H.265)	1x 8K30 (H.265) 3x 4K60 (H.265) 6x 4K30 (H.265) 12x 1080p60 (H.265) 24x 1080p30 (H.265)
PCIe	1 x4 (PCIe Gen2)	1 x1 + 1 x2 (PCIe Gen2)	1 x1 + 1 x4 OR 1 x1 + 1 x1 + 1 x2 (PCIe Gen2)			1 x1 (PCIe Gen3) + 1 x4 (PCIe Gen4)		1 x8 + 1 x4 + 1 x2 + 2 x1 (PCIe Gen4, Root Port & Endpoint)			1 x4 + 3 x1 (PCIe Gen4, Root Port & Endpoint)	2 x8 (or 1x8 + 2x4) + 2 x4 + 2 x1 (PCIe Gen4, Root Port & Endpoint)
Networking	10/100/1000 BASE-T Ethernet			10/100/1000 BASE-T Ethernet, WLAN	10/100/1000 BASE-T Ethernet							10/100/1000 BASE-T Ethernet 4x 10GbE XFI
Display	2 multi-mode DP 1.2/eDP 1.4/HDMI 2.0 1 x2 DSI (1.5Gbps/lane)	2 multi-mode DP 1.2/eDP 1.4/HDMI 2.0 1x 2 DSI (1.5Gbps/lane)	2 multi-mode DP 1.2/eDP 1.4/HDMI 2.0 2 x4 DSI (1.5Gbps/lane)			2 multi-mode DP 1.4/eDP 1.4/HDMI 2.0 No DSI support		3 multi-mode DP 1.4/eDP 1.4/HDMI 2.0 No DSI support			1x 8K60 multi-mode DP 1.4a (+MST)/eDP 1.4a/HDMI 2.1	1x 8K60 multi-mode DP 1.4a (+MST)/eDP 1.4a/HDMI 2.1
Power	5W | 10W	7.5W | 15W			10W | 20W	10W | 15W | 20W		10W | 15W | 30W		20W | 40W	10W | 15W | 25W	15W | 30W | 50W
Mechanical	69.6mm x 45mm 260-pin SO-DIMM connector	69.6mm x 45mm 260-pin SO-DIMM connector	87mm x 50mm 400-pin connector Integrated Thermal Transfer Plate			69.6mm x 45mm 260-pin SO-DIMM connector		100mm x 87mm 699-pin connector Integrated Thermal Transfer Plate			69.6mm x 45mm 260-pin SO-DIMM connector	100mm x 87mm 699-pin connector Integrated Thermal Transfer Plate

(source: https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/)

Jetson Nano Developer Kit

- https://developer.nvidia.com/embedded/jetson-nano-developer-kit


NVIDIA. DEVELOPER
HOME BLOG FORUMS DOCS DOWNLOADS TRAINING
JOIN
JETSON DEVELOPER KITS PRODUCTION MODULES SOFTWARE PARTNERS COMMUNITY BUY
Home > Autonomous Machines > Buy > Jetson Nano Developer Kit
Buy Now >
Get Started >

Jetson Nano Developer Kit

The power of modern AI is now available for makers, learners, and embedded developers everywhere.

NVIDIA® Jetson Nano™ Developer Kit is a small, powerful computer that lets you run multiple neural networks in parallel for applications like image classification, object detection, segmentation, and speech processing. All in an easy-to-use platform that runs in as little as 5 watts.

It's simpler than ever to get started! Just insert a microSD card with the system image, boot the developer kit, and begin using the same NVIDIA JetPack SDK used across the entire NVIDIA Jetson™ family of products. JetPack is compatible with NVIDIA's world-leading AI platform for training and deploying AI software, reducing complexity and effort for developers.

Get started today with the Jetson Nano Developer Kit. We look forward to seeing what you create!

Jetson Nano Developer Kit

GPU	128-core Maxwell
CPU	Quad-core ARM A57 @ 1.43 GHz
Memory	4 GB 64-bit LPDDR4 25.6 GB/s
Storage	microSD (not included)
Video Encode	4K @ 30 | 4x 1080p @ 30 | 9x 720p @ 30 (H.264/H.265)
Video Decode	4K @ 60 | 2x 4K @ 30 | 8x 1080p @ 30 | 18x 720p @ 30 (H.264/H.265)
Camera	2x MIPI CSI-2 DPHY lanes
Connectivity	Gigabit Ethernet, M.2 Key E
Display	HDMI and display port
USB	4x USB 3.0, USB 2.0 Micro-B
Others	GPIO (General Purpose Input/Output), I^{2}C, I^{2}S, SPI, UART
Mechanical	69 mm x 45 mm, 260-pin edge connector

Jetpack SDK

- **NVIDIA Jetpack SDK (https://developer.nvidia.com/embedded/jetpack)**
  - NVIDIA JetPack SDK is the most comprehensive solution for building end-to-end accelerated AI applications. All Jetson modules and developer kits are supported by JetPack SDK.
  - JetPack SDK includes the Jetson Linux Driver Package (L4T) with Linux operating system and CUDA-X accelerated libraries and APIs for Deep Learning, Computer Vision, Accelerated Computing and Multimedia. It also includes samples, documentation, and developer tools for both host computer and developer kit, and supports higher level SDKs such as DeepStream for streaming video analytics and Isaac for robotics.
  - JetPack 4.6 is the latest production release, and supports all Jetson modules including Jetson AGX Xavier Industrial module. JetPack 4.6 includes support for Triton Inference Server, new versions of CUDA, cuDNN and TensorRT, VPI 1.1 with support for new computer vision algorithms and python bindings, L4T 32.6.1 with Over-The-Air update features, security features, and a new flashing tool to flash internal or external media connected to Jetson.
  - See highlights below for the full list of features added in JetPack 4.6
  - In addition to the L4T-base container, CUDA runtime and TensorRT runtime containers are now released on NGC for JetPack 4.6.

Key Features in Jetpack

◆ Key Features in Jetpack

- https://developer.nvidia.com/embedded/jetpack

OS	Features
CUDA Toolkit for Host	▪ NVIDIA L4T provides the bootloader, Linux kernel 4.9, necessary firmwares, NVIDIA drivers, sample filesystem based on Ubuntu 18.04, and more
CUDA Toolkit for Jetson	▪ Ubuntu with cross-development support
Computer Vision	▪ OpenCV ▪ VPI (Vision Programing Interface) is a software library that provides Computer Vision / Image Processing algorithms implemented on PVA[1] (Programmable Vision Accelerator), GPU and CPU
cuDNN	▪ CUDA Deep Neural Network
TensorRT	▪ a high performance deep learning inference runtime for image classification, segmentation, and object detection neural networks. ▪ TensorRT is built on CUDA.
Multimedia API	▪ low level APIs for flexible application development ▪ Camera application API ▪ Scalable Video Coding (SVC) H.264 encoding, YUV444 8, 10 bit encoding and decoding ▪ Sensor driver API

Homework 14

Homework 14

14.1 Raspberry Pi에 Raspbian을 설치하고, 원격에서 접속할 수 있도록 구성하라. mobaxterm을 사용하여 원격에서 Raspberry Pi에 접속하여 별 (star)를 그리는 파이썬 프로그램을 Raspberry Pi 상에서 실행하도록 하고, 결과를 capture하라.

14.2 Homework 12.4에서 구현하였던 파이썬 스레드 및 tkinter GUI 기반의 채팅 프로그램에서 TCP server 기능을 Raspberry Pi에 설치하고, desktop PC에 TCP client 기능을 설치한 후, 채팅이 실행하도록 하고, 결과를 capture 하라. Raspberry Pi의 Raspbian 환경에서 정상적으로 실행할 수 있도록 필요한 수정을 추가할 것.

참고문헌

<Jetson Nano, Jetpack>

[1] Jetson Nano Developer Kit, https://developer.nvidia.com/embedded/jetson-nano-developer-kit.

[2] NVIDIA Jetson Nano OS 설치 및 초기화, https://wendys.tistory.com/141.

[3] Jetson Download Center, https://developer.nvidia.com/embedded/downloads.