

# Raspberry Pi 4 초기 설정 및 CC 활용을 위한 환경 구축

1. 개요
2. 준비물
3. Boot SD 만들기
4. 부팅하기
5. 환경 구축

## 1. 개요

기체의 임무가 간단한 경우 성능이 좋은 CC를 사용할 필요가 없다. 따라서 간단한 제어 코드 정도는 Raspberry Pi로 실행해도 문제가 없다. 본 문서에서는 Raspberry Pi 4를 CC로 활용할 수 있도록 구축하는 방법을 설명한다. Raspberry Pi의 성능과 용량이 한정적이기에 최소한의 설치만 진행하도록 주의한다.

## 2. 준비물

Companion Computer	Raspberry Pi 4
Power Source	USB-C, 15W or larger
Boot SD	Micro SD, 16GB or larger
	Micro HDMI→HDMI 케이블
	모니터
	마우스, 키보드 (무선도 가능)
	Computer for writing Boot SD

## 3. Boot SD 만들기

Raspberry Pi는 자체 용량이 적기 때문에 외부 디스크를 활용한다. 이를 위해 Raspberry Pi에는 Micro SD 슬롯이 있다. 여기에 Boot용 SD 카드를 넣고 전원을 공급하면 자동으로 SD 카드로 부팅이 된다. Boot SD 카드는 아래 링크를 따라하면 된다. Boot SD 카드를 만드는 과정에서 SD 카드가 초기화 되기 때문에 중요한 파일은 백업을 하도록 한다.

<https://ubuntu.com/tutorials/how-to-install-ubuntu-desktop-on-raspberry-pi-4#2-prepare-the-sd-card>

Ubuntu 환경에서 Boot SD를 만드는 경우, rpi-imager 설치 후 실행은 그냥 터미널에 아래 명령을 실행하면 된다.

```
$ rpi-imager
```

rpi-imager는 계속 업데이트 되기 때문에 링크의 UI와 조금 다를 수 있다. 아래의 순서를 따라 진행한다.

### ①. Device 선택

사용할 Raspberry Pi의 모델명을 선택하면 된다. 이 경우 Raspberry Pi 4를 선택한다.

### ②. OS 선택

OS는 ubuntu desktop 22.04.5 LTS (64-BIT)를 선택한다. Server가 아닌 Desktop 버전을 설치하는 것이 개발할 때 조금 더 편하다.

### ③. SD 카드를 선택

기본 설정으로 시스템 디스크는 선택하지 못하도록 돼있지만, 디스크를 두 개 이상 사용 중이거나 파티션을 나눈 경우 조심하여 선택한다. 드라이브 유형과 용량을 확인하면 좋다.

### ④. Write

완료 후 SD 카드를 컴퓨터에서 제거한다.

## 4. 부팅하기

라즈베리파이에 sd카드를 삽입하고 전원을 연결하면 자동으로 부팅한다. 부팅하기 전에 키보드, 모니터, 마우스를 모두 연결해놓아야 켜졌을 때 모두 정상적으로 작동한다.

이후 일반적인 우분투 설치와 같이 설치를 한다. 라즈베리파이의 성능이 좋은 편은 아니기에 다소 느릴 수 있다.

## 5. 환경 구축

### 5-1. ROS2 Humble 설치

Raspberry Pi에서는 시뮬레이션을 실행할 필요가 없기 때문에 ros 설치 단계에서는 `ros-humble-desktop`이 아닌 `ros-humble-ros-base`를 설치한다. 필요에 따라서 `ros-humble-dev-tools`와 `rqt` 등의 추가 기능을 설치한다. `rqt`만 따로 설치하는 경우 아래의 링크의 1.1을 따라하면 된다. 사용 중인 ROS2의 버전을 잘 입력하도록 한다.

<https://wiki.ros.org/rqt/UserGuide/Install/Groovy>

### 5-2. PX4 환경 구축

PX4 Firmware 및 toolchains는 설치할 필요가 없다. 하지만 uORB 메시지를 활용하는 노드를 작성한 경우 `MicroXRCEAgent`와 `px4_msgs`를 설치해야한다. 아래의 링크를 따라 Raspberry Pi 환경을 설정하고 `MicroXRCEAgent` 등 필요한 자원을 설치한다.

[https://docs.px4.io/main/en/companion\\_computer/pixhawk\\_rpi#ubuntu-setup-on-rpi](https://docs.px4.io/main/en/companion_computer/pixhawk_rpi#ubuntu-setup-on-rpi)

[https://docs.px4.io/main/en/companion\\_computer/pixhawk\\_rpi#ros-setup-on-rpi](https://docs.px4.io/main/en/companion_computer/pixhawk_rpi#ros-setup-on-rpi)

MAVLink Communication은 MAVLink를 활용한 노드를 작성했을 경우 따라한다. uXRCE-DDS는 FC의 환경을 설정하는 것이기 때문에 CC인 Raspberry Pi 4에서는 할 필요 없다.

`px4_msgs`는 “개발환경 구축 방법”의 3-5. (선택사항) `px4_msgs workspace` 구축하기를 따라서 똑같이 하면 된다. 이때 `colcon build` 대신 `colcon build --executor sequential`을 활용한다. `colcon build`를 실행할 경우 해당 패키지 내의 코드를 병렬로 컴파일한다. 그러나 Raspberry Pi의 성능이 좋지 않기 때문에 멈추는 등의 문제가 발생 할 수 있다. 따라서 `--execute sequential` 태그를 활용하여 순차적인 컴파일을 실행하도록 한다. 한시간 정도 소요되니 인내심을 갖고 기다린다.

### 5-3. SSH 설치 및 연결

Raspberry Pi를 CC로 활용하기 위해서는 원격으로 접근이 가능해야 한다. 이를 위해 지상국 컴퓨터에서 SSH 클라이언트인 Putty를 이용하고, Raspberry Pi에서는 SSH 서버를 열어놓는다. SSH 서버를 열기 위해 SSH server/client를 설치한다.

```
$ sudo apt update && sudo apt upgrade
$ sudo apt-get install openssh-server
$ sudo apt-get install ssh
```

이후 서버 서비스를 실행한다.

```
$ sudo service ssh start
```

아래 명령어를 통해 ssh service의 상태를 확인할 수 있다.

```
$ service ssh status
```

이후 아래 명령어를 통해 Raspberry Pi의 ip를 확인한다. IP는 사용 중인 WiFi에 따라 바뀌기 때문에 새로운 네트워크에 연결할 때마다 확인을 해준다.

```
$ ip a
```

아래와 같은 실행 결과가 뜬다.

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: wlan0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether 00:00:a1:99:1a:1a brd ff:ff:ff:ff:ff:ff
    altname wlan0
    inet 192.168.11.22/33 brd 192.168.44.55 scope global dynamic noprefixroute wlan0
        valid_lft 00000sec preferred_lft 00000sec
    inet6 aa00::0aa0:0000:000a:aa00/00 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

여기서 파란색으로 표시한 192.168.12.34가 활용하게 될 Raspberry Pi의 IP 주소이다.

이후 지상국에서 Putty를 실행하고 Raspberry Pi의 IP 주소를 입력한다. 다른 설정을 바꿀 필요 없다. 실행하면 보안 관련 주의 안내가 뜨지만 무시하고 연결하면 된다. 이후 로그인할 user와 비밀번호를 입력할 때 **sudo** 권한이 있는 user로 로그인 하는 것이 편하다. 로그인까지 완료하면 터미널 창이 뜨고, 해당 창에서 Raspberry Pi를 제어할 수 있다.