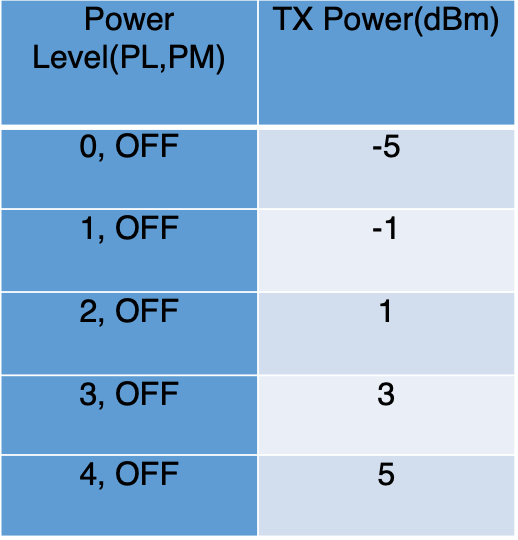
**2020 MCNL 하계인턴 프로젝트 매뉴얼**

1. Xbee s2c 기본 사양 관련

(1) Power control

****

**device.set\_parameter(“PL”, utils.int\_to\_bytes(2))**

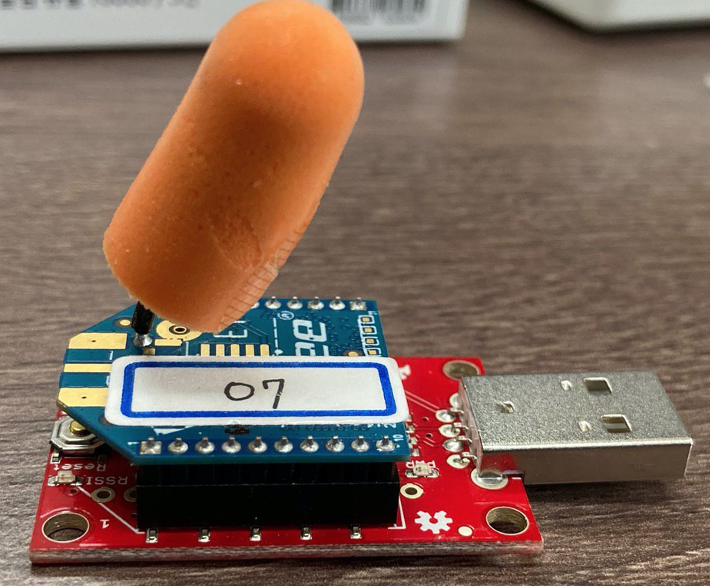
**device.set\_parameter(“PM”, utils.int\_to\_bytes(0))**

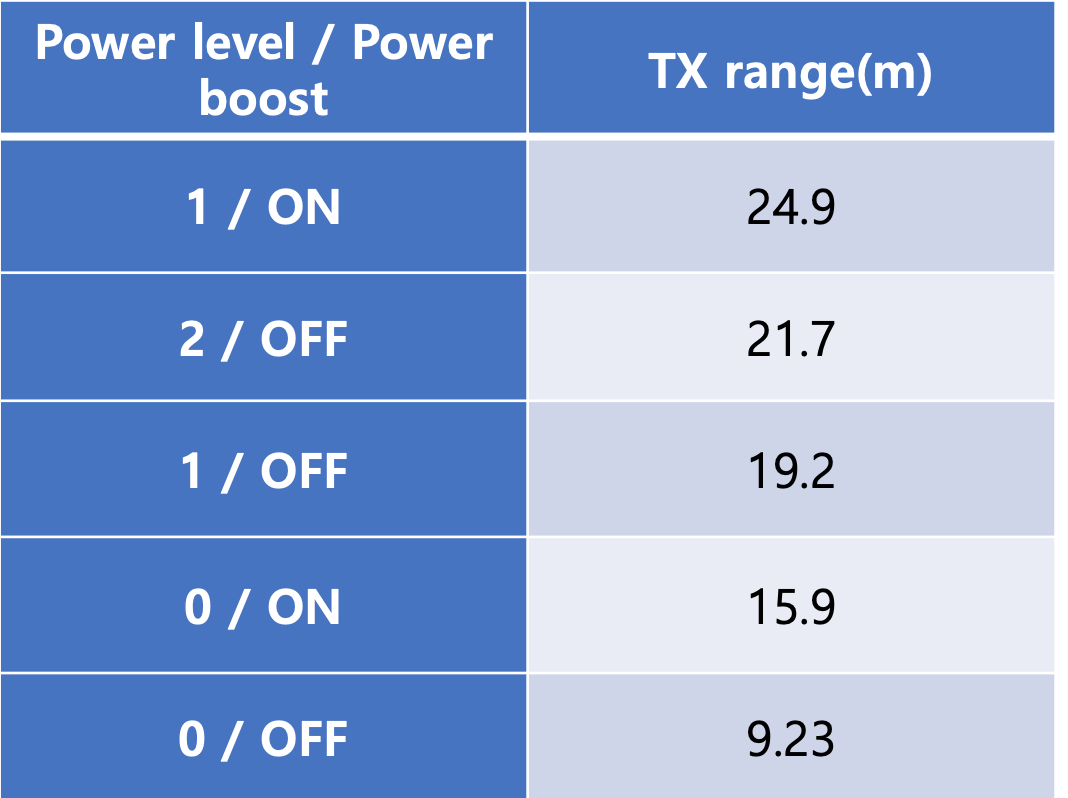
위와 같은 code로 Xbee의 PL, PM 의 parameter들을 control 할 수 있다.

\* PL(0~4), PM(0-OFF, 1-ON)을 통해 transmit power를 control할 수 있다.

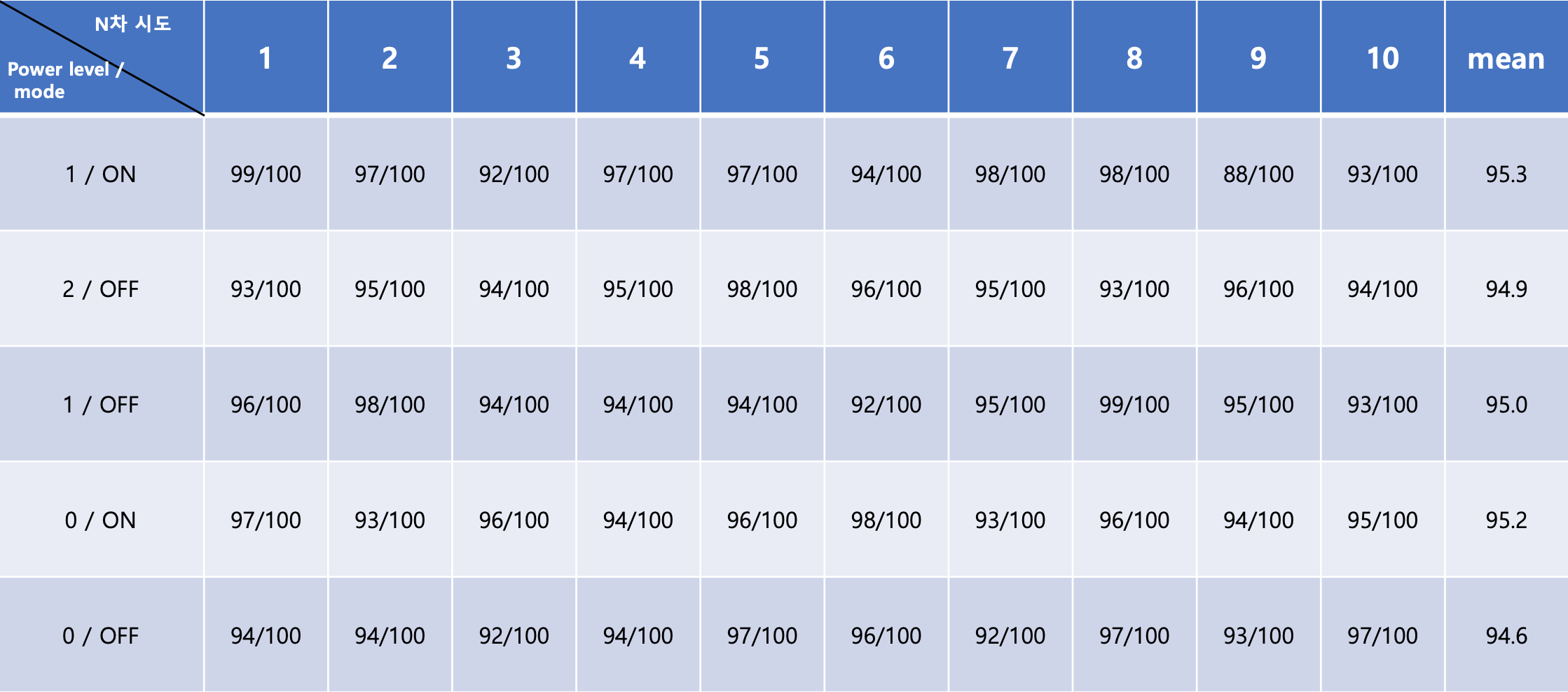
(2) Communication range test

장소 : ECC 삼성홀 앞

장비 : xbee s2c의 안테나에 이어플러그를 꽂아 tx range를 실제보다 줄임(장소 선정을 위해)



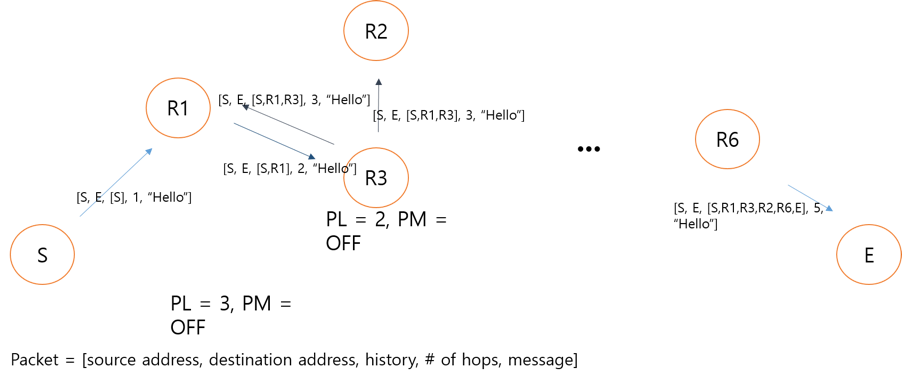
<tx range 측정 결과>

****

<tx range 실험 정리 표>

2. 코드 관련

(1) Ad-hoc network connectivity 구상



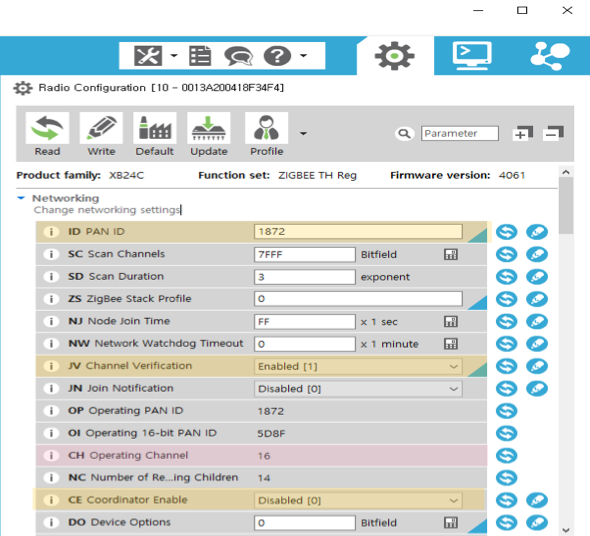
• Broadcasting 방식을 채택한다. TX range 내에 모든 node들에게 pilot을 보내고, ACK를 받음으로써 range 내에 있는 node의 수를 파악한다.

• 이를 바탕으로 TX range를 정하면, broadcasting 방식으로 packet을 전송한다.

• hop > 4인 경우, packet을 버림으로써 비효율적인 packet은 거른다.

(2) XCTU Setting method

**Setting parameters for Broadcasting at XCTU**

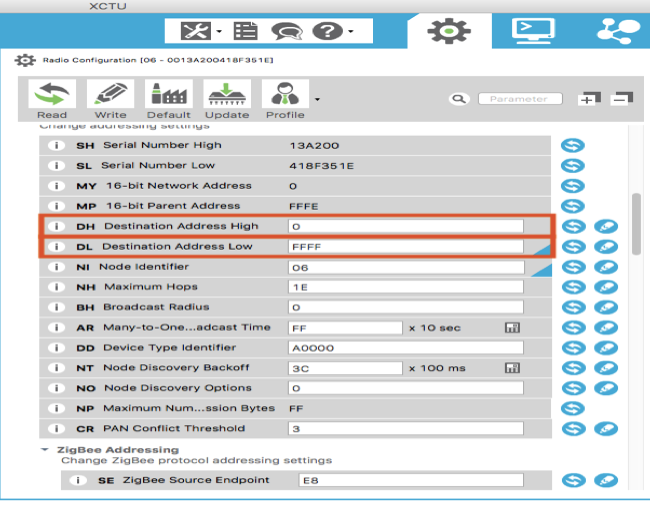


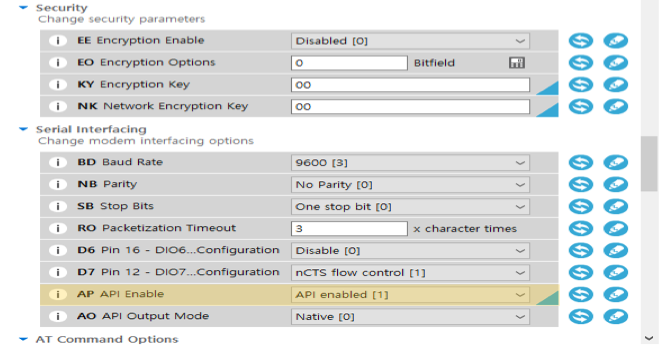
• ID : 통신할 xbee 기기끼리는 같은 ID를 사용해야 한다.

• JV : relay node로 사용할 xbee는 Enabled로 설정해야 한다.

• CH : 같은 channel을 가져야 통신이 가능하므로, 초기에 하나의 coordinator에 remote를 시킴으로써 channel을 통일시킨다. (실험 전, ad-hoc에 사용되지 않는 xbee를 coordinator 로 설정후, remote로 실험에 사용되는 모든 xbee의 channel을 통일시켜야 한다.)

• CE : coordinator(enabled)/router(disabled)를 정하는 parameter





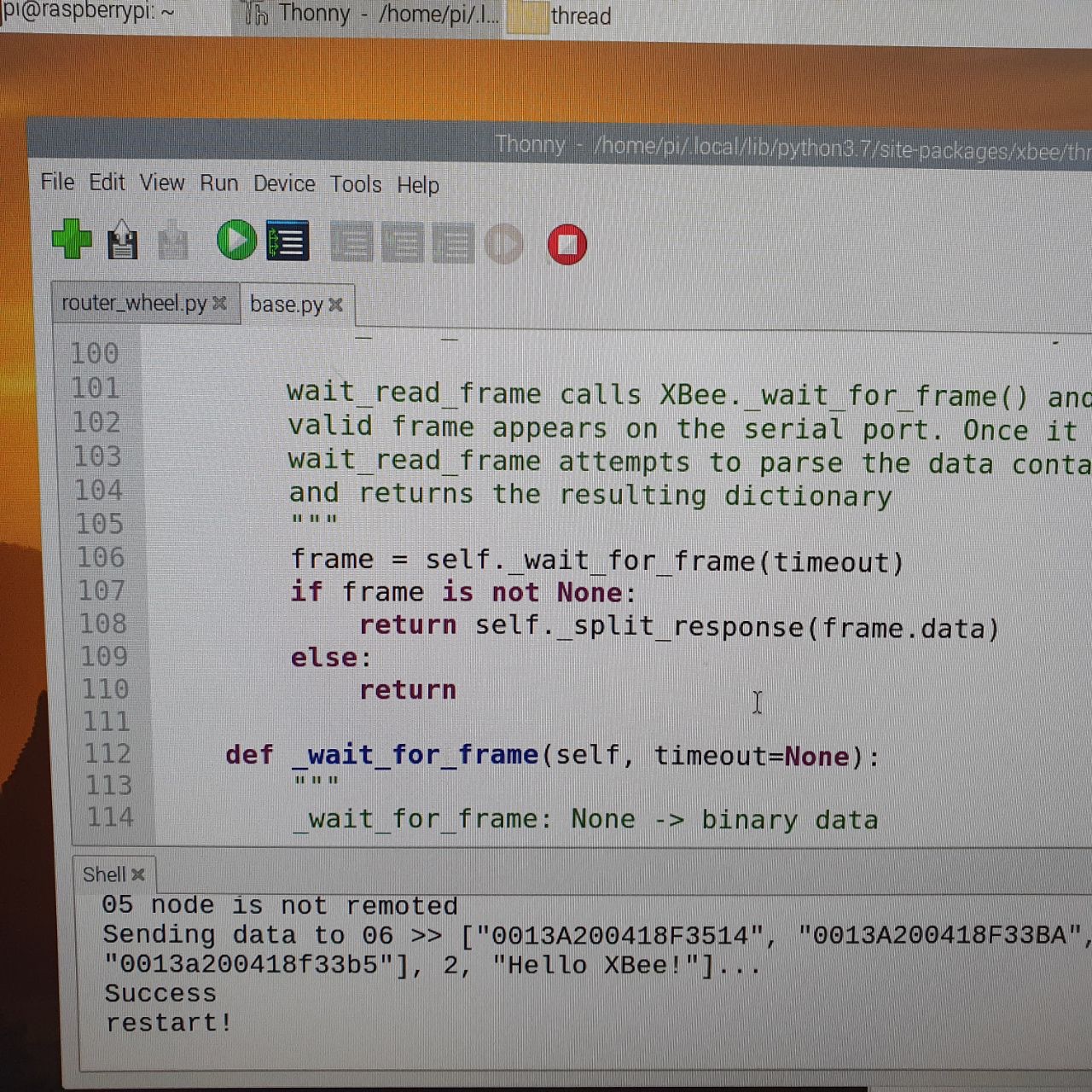
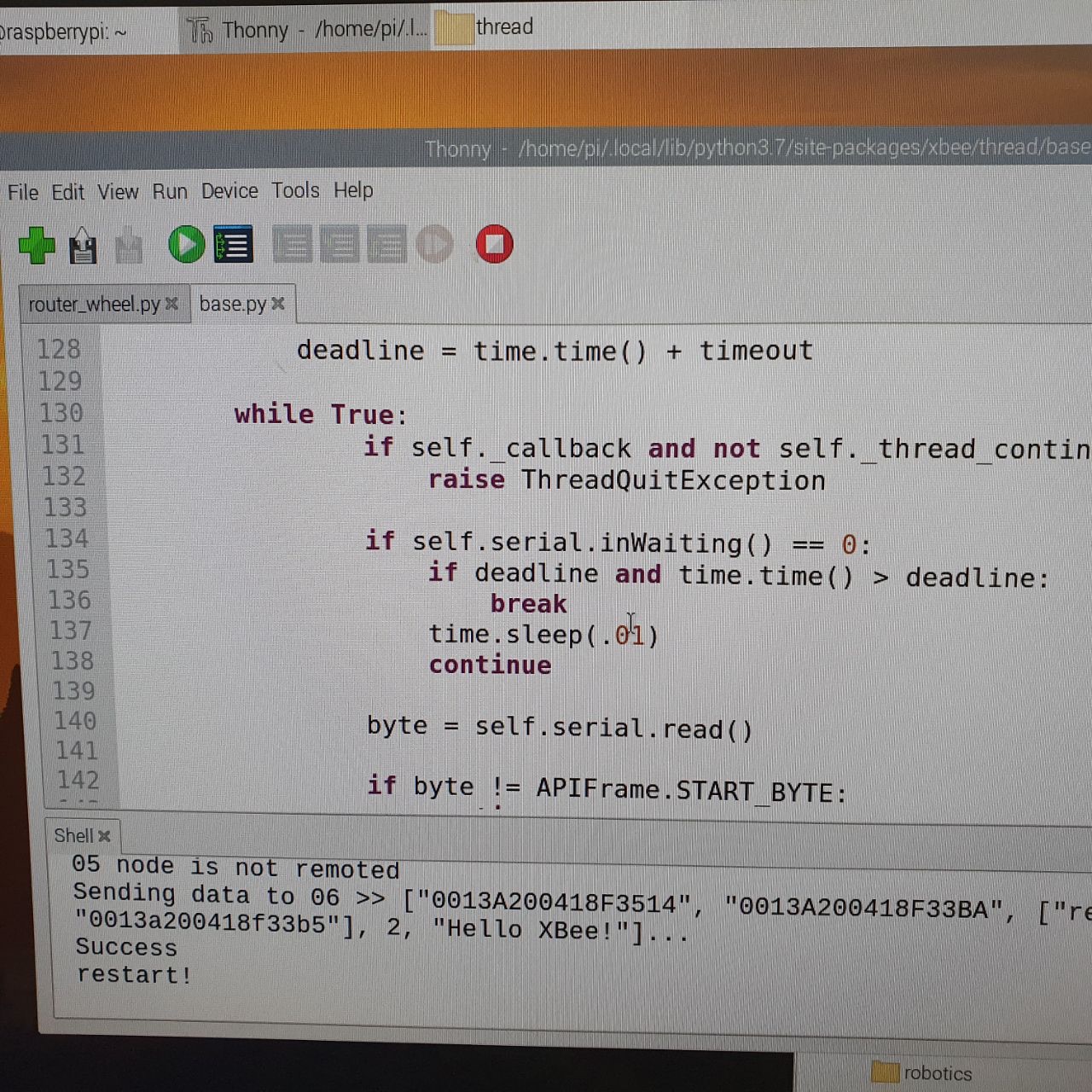
• DH : Broadcasting을 위해서는 0으로 설정해야한다.

• DL : Broadcasting을 위해서는 FFFF로 설정해야한다.

• AP : API enable로 설정해야 python에서 parameter control이 가능하다.

코드 실행을 위해선 xbee library 수정 필요. (라즈베리파이는 python3.7, 3.8 모두에서 고쳐야함.)

1. /home/pi/.local/lib/python3.7/site-packages/xbee/thread/base.py
2. 위의 경로로 들어가서, 기존의 library에서 아래의 사진에 첨부된 코드로 변경



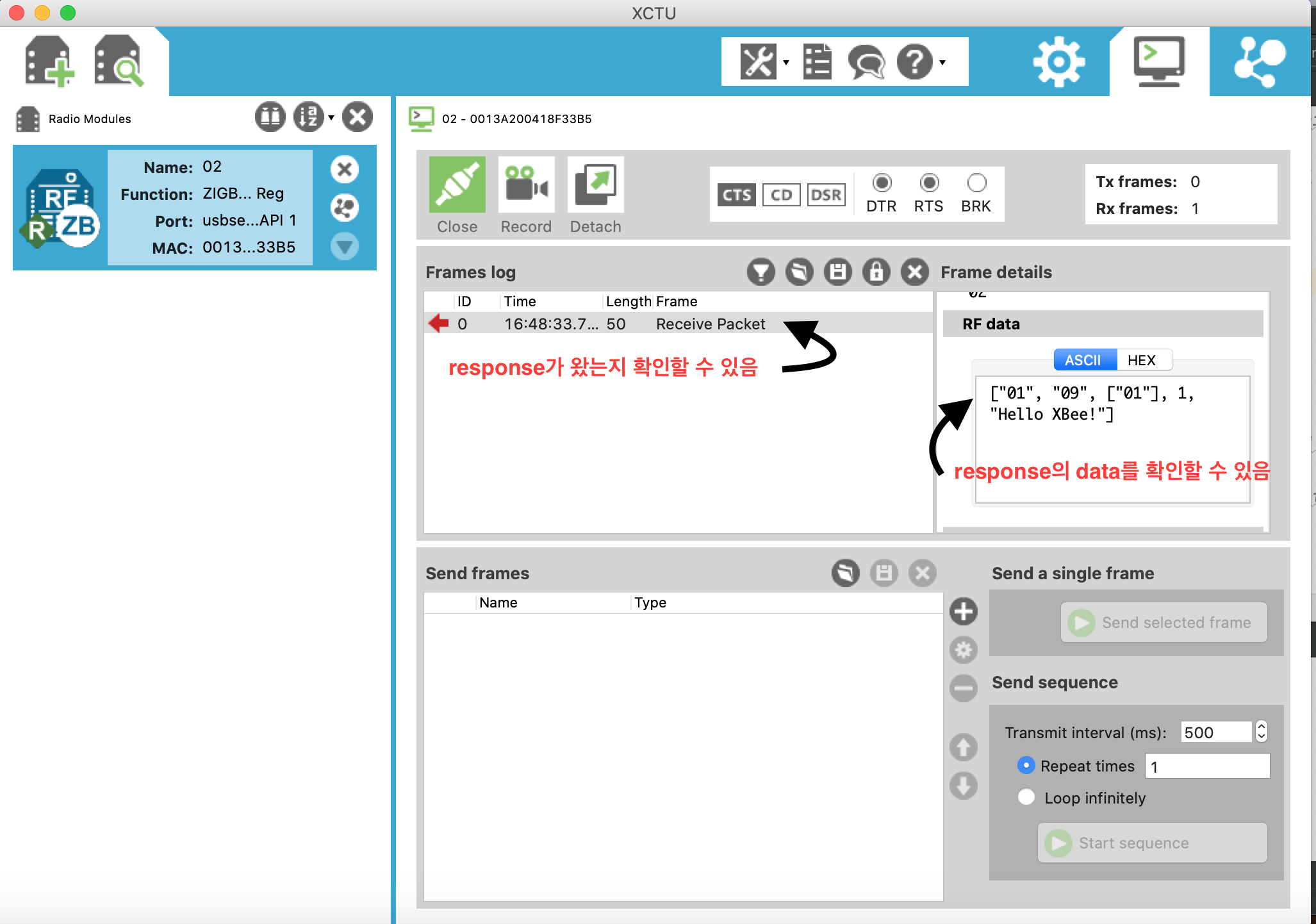
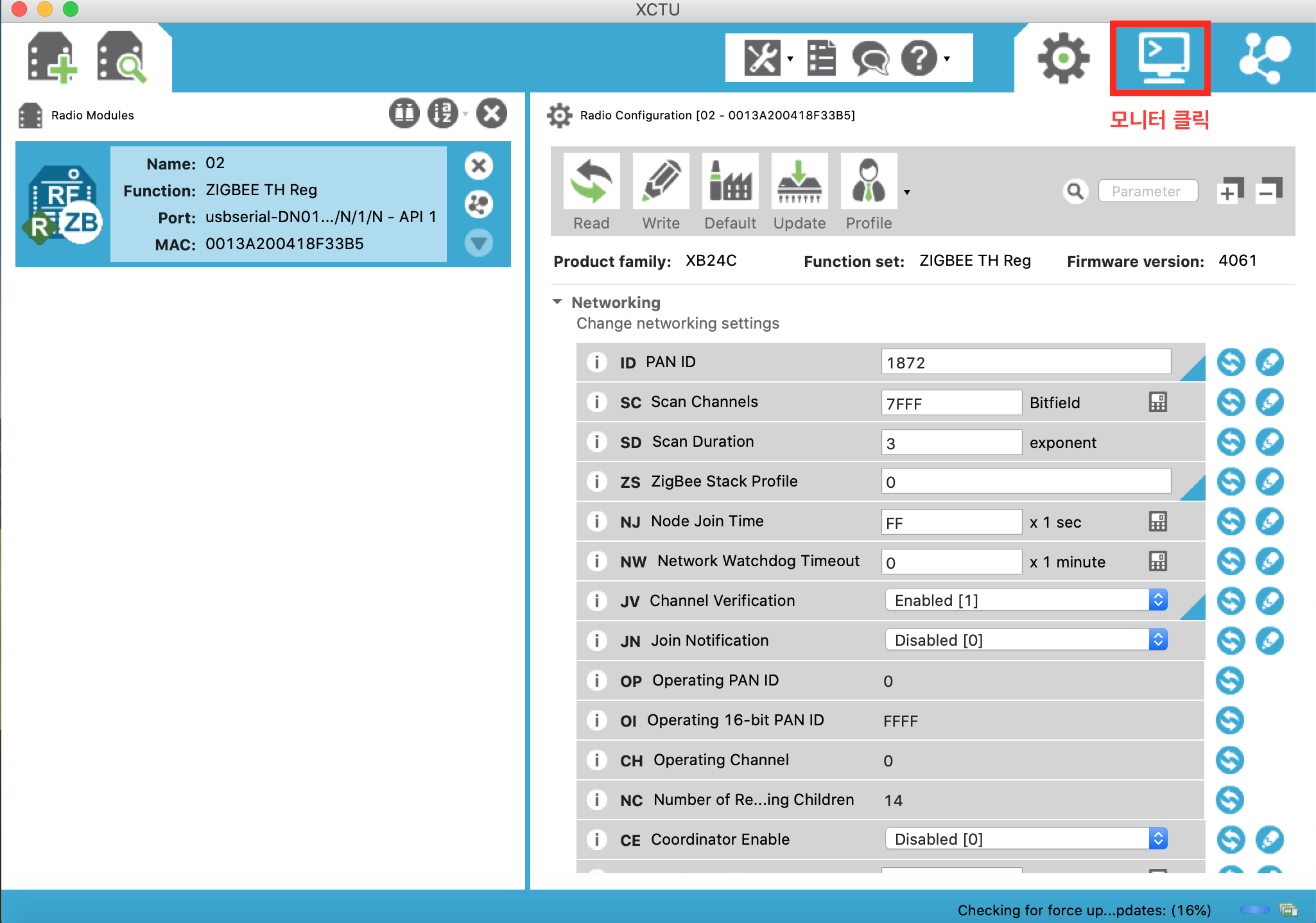
(3) Source node 코드

(4) Relay node 코드

(3,4 코드는 파이썬 파일로 전달)

(5) Destination node

아래와 같은 방법으로 XCTU를 통해 response를 확인할 수 있다.



\* 모니터를 연결 한 port는 파이썬에서 코드 실행을 할 수 없다는 점을 주의해야 한다.

3. 라즈베리 파이 관련

(1) Raspberry pi setting method

1. 16기가 SD 카드에 Raspberry Pi Imager를 설치한다. (SD 카드 포맷시 / 기존의 SD 카드를 그대로 사용해도 되지만, 사용하던 SD카드에 깔린 여러 파일 때문에 오류가 생길 수 있으므로 포맷을 추천함)

2. 라즈베리파이 기본 설정을 완료한다.

3. 명령 창을 킨다.

4. 프로젝트에 맞는 파이썬 버전과 pip 버전을 설치한다.

* python3.7이 default로 깔려 있으나, python3.8 을 사용하기 위해서는 다운로드가 필요하다.
* pip 설치 관련 : <https://studyhard24.tistory.com/232>
* 환경변수 경로 설치가 중요함

(2) autostart method

1. sudo nano ~/.bashrc

2. 가장 하단에 fi 라고 적힌 부분 바로 밑에 2~6 입력

3. alias python=/usr/local/bin/python3.8

4. alias pip=/usr/local/bin/pip3.8

5. echo Running at boot

6. python /home/pi/Desktop/untitled/filename.py (실행할 파일이 있는 경로로 설정해야함!)

7. 2~6 까지 적고 ctrl+X 누르고 Y 누르고 enter 누르기

8. sudo python3.8 -m pip install —upgrade pip

9. pip -V 를 써보고 3.8 ver으로 나오는지 확인

10. python-V를 써보고 3.8 ver으로 나오는지 확인

11. library 설치 (sudo pip install library’s name)

12. sudo pip install digi-xbee

13. sudo pip install numpy

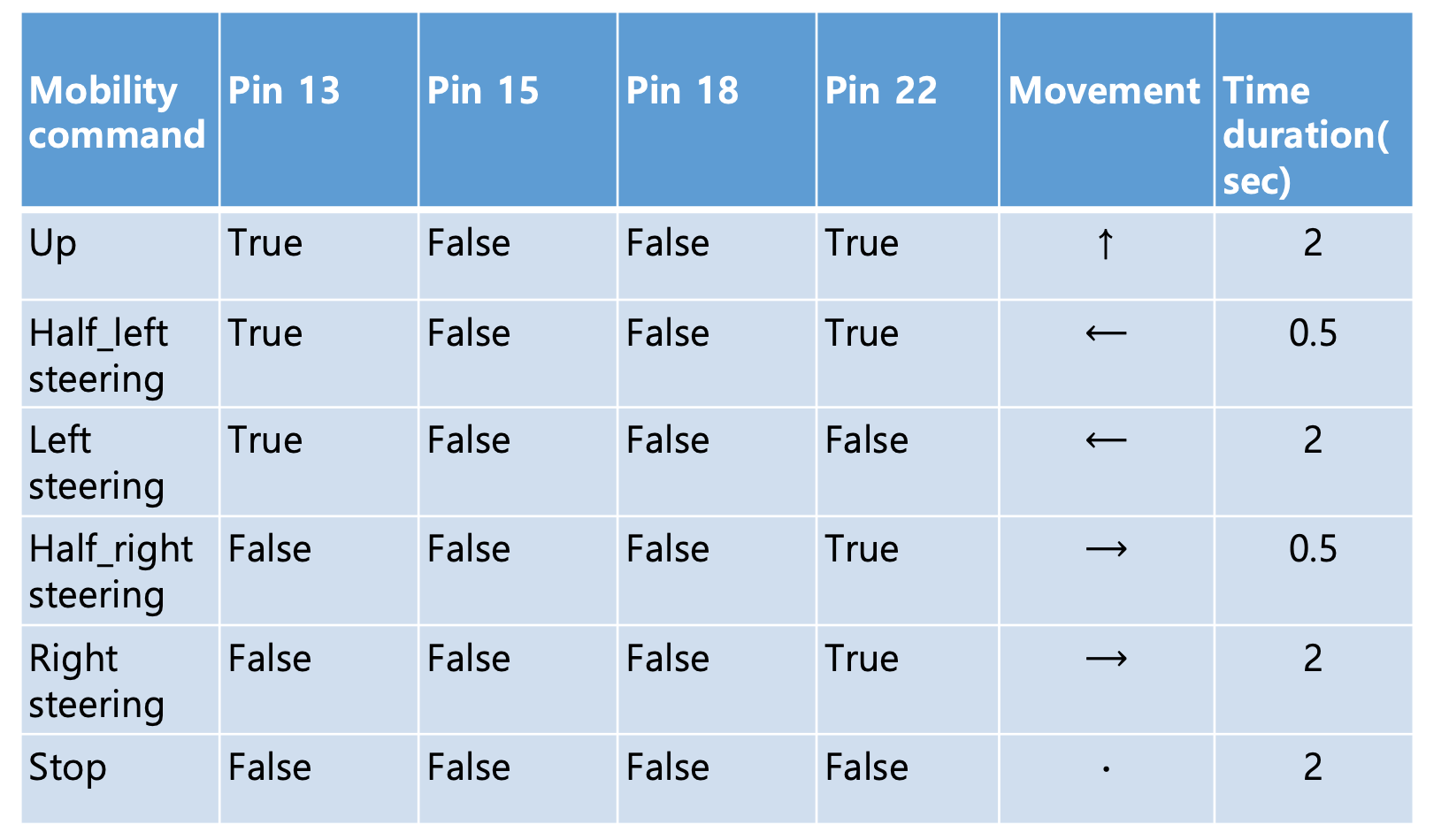
14. sudo pip install pandas

15. sudo pip install xbee

16. 15번까지 완료 후 터미널창을 껐다 다시 켜보기 (터미널에서 자동 실행을 빠져나오려면 Crtl+C)

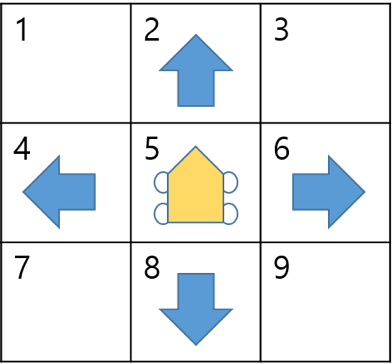
4. 모빌리티 관련

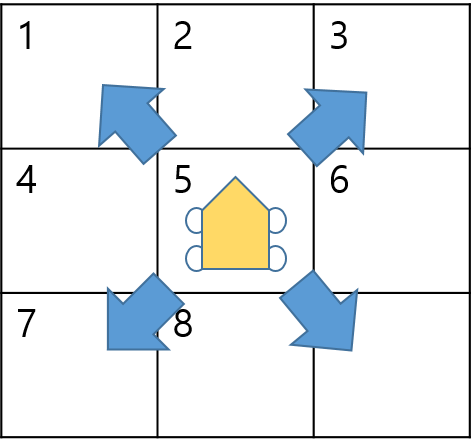
(1) Mobility control



+) half\_up, half\_down, half\_leftsteering, half\_rightsteering 도 mobility command에 추가하였다.

위 4개의 추가 command 들은 45도 방향으로 움직이는 것을 목적으로 한다.



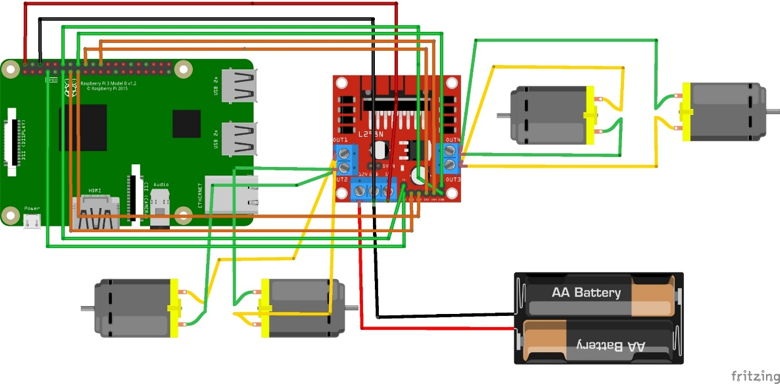


(2) RC car preparation

RC-car 제작을 위해 필요한 준비물

: Raspberry-Pi3, RC-car 프레임 키트, motor driver, 점퍼선, battery(AA Battery\*4)

<라즈베리파이,모터를 모터드라이버로 연결>



<Raspberry-Pi Pin number(GPIO)>

