

터틀봇3를 위한 ROS 설치 및 제어의 오류 해결 방법

박태환*, 이강희^O

*승실대학교 미디어학과,

^O승실대학교 미디어학과

e-mail: taewhan@soongsil.ac.kr*, kanghee.lee@ssu.ac.kr^O

How to fix errors in ROS installation and control for TurtleBot 3

Tae-Whan Park*, Kang-Hee Lee^O

*Dept. of Digital Media, Soongsil University,

^ODept. of Digital Media, Soongsil University

● 요 약 ●

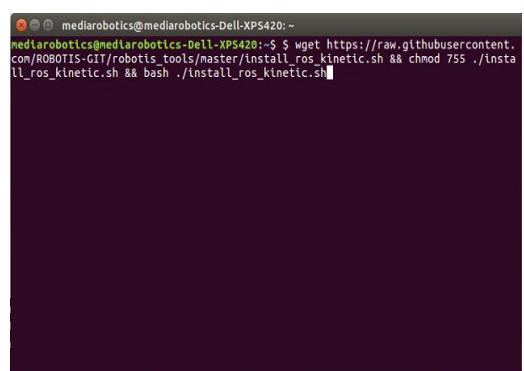
터틀봇3(Turtlebot3)을 제어하기 위하여 퍼시와 터틀봇3 각각에 ROS(Robot Operating System)을 설치하고 제어한다. 터틀봇3는 라즈베리파이 3 보드로 제어되는 오픈소스 로봇이다. 전세계에서 유명한 교육 및 연구용 로봇이지만 설치와 제어 과정에서 여러 오류를 경험하는 사용자들이 있다. 본 논문은 터틀봇3를 처음 사용하는 사용자들을 위하여 설치과정과 설치과정에서 발생하는 오류들에 대하여 다룬다.

키워드: 터틀봇3(Turtlebot3), ROS, 라즈베리파이(Raspberry pi)

I. Introduction

터틀봇3는 전세계에서 유명한 교육 및 연구용 오픈소스 로봇이다. 손쉽게 조립, 유지보수, 교체 및 변형이 가능하며 비교적 저렴한 가격의 로봇이다 [1]. 오픈소스 하드웨어이기 때문에 회로도, PCB 커버, 3D 도면 등 연구 개발에 필요한 대부분의 정보가 공개되어있다. 터틀봇3은 ROS를 이용하여 제어된다. ROS는 Robot Operating System의 약자이다. ROS는 무료로 사용 가능한 오픈소스이며 여러 가지 프로그래밍 언어로 사용 가능하다 [2]. 본 연구는 직접 터틀봇3와 컴퓨터에 ROS를 로보티즈에서 제공하는 매뉴얼을 참고하여 설치를 진행하였다. 설치 시에 발생하거나 발생할 수 있는 오류에 대하여 다룬다.

1. 데스크탑 컴퓨터에 ROS 설치



```
mediarobotics@mediarobotics-Dell-XPS420: ~
mediarobotics@mediarobotics-Dell-XPS420: ~$ wget https://raw.githubusercontent.com/ROBOTIS-GIT/robotis_tools/master/install_ros_kinetic.sh && chmod 755 ./install_ros_kinetic.sh && bash ./install_ros_kinetic.sh
```

Fig. 1. 컴퓨터에 ROS kinetic 설치

II. How to fix errors in ROS installation and control for TurtleBot 3

본 실험은 터틀봇3 버거와 Ubuntu 16.04LTS가 설치된 데스크탑 컴퓨터로 진행한다. 로보티즈에서 제공하는 Turtlebot3 e-manual을 사용하였다 [3]. 설치 과정 뿐만 아니라 설치 과정에서 생길 수 있는 총 3가지 오류에 대하여 다룬다. 라즈비안 설치 후의 오류, 다이나믹셀에 관련된 오류, cartographer 설치할 때의 오류에 대하여 설명한다.

첫 번째로 ROS를 데스크탑 컴퓨터에 설치한다. ROS는 여러 가지 버전이 있으며 본 논문에서는 Kinetic 버전을 설치한다. 그림1처럼 새 터미널 창에 명령어를 입력하면 설치가 진행된다.

Fig. 2. ROS 패키지들 설치

- cd ~/catkin_ws/src/
- git clone https://github.com/ROBOTIS-GIT/turtlebot3_msgs.git
- git clone https://github.com/ROBOTIS-GIT/turtlebot3.git
- cd ~/catkin_ws && catkin_make

Fig. 3. catkin 패키지들 설치

ROS를 설치한 후에는 터틀봇3의 제어에 필요한 ROS 패키지들을 그림2와 같이 터미널 명령어를 사용하여 설치한다. catkin 공간에도 필요한 패키지들을 설치한다. catkin_ws 폴더는 그림1과 같은 방법으로 ROS를 설치하면 자동으로 생성된다. catkin은 ROS의 빌드 시스템이며 ROS와 관련된 빌드, 패키지 관리, 패키지 간의 의존 관계들을 편리하게 사용할 수 있게 해준다. 그림3과 같이 catkin 폴더 안의 src 폴더 안에 패키지들을 git clone으로 설치 후에 catkin 폴더로 돌아가서 catkin_make를 실행하면 기본적으로 필요한 패키지들은 설치 완료된다.

Fig. 4. bashrc 파일

패키지를 설치한 후에는 터틀봇3와의 네트워크 통신을 위해 네트워크 설정을 해준다. 터틀봇3와 네트워크 통신을 하기 위해선 터틀봇3와 컴퓨터는 같은 공유기를 사용한다. 터미널창에 ifconfig를 입력하는 등의 방법으로 컴퓨터의 현재 ip 주소를 확인한다.

터미널창에 nano ~/.bashrc를 입력하여 bashrc 파일을 연다. 컴퓨터의 ip주소가 192.168.0.2라고 가정했을 때 그림4와 같이 입력을 한다. 변경 내용을 저장한 후 터미널 창에 source ~/.bashrc 라고 입력 후 실행하면 네트워크 설정까지 끝난다.

Fig. 5. roscore

새 터미널창을 열고 roscore라고 입력 후 실행 하였을 때 그림5와 같이 실행된다면 컴퓨터에 ROS가 성공적으로 설치 완료된다.

2. 터틀봇3의 라즈베리파이에 ROS 설치

두 번째로 터틀봇3의 라즈베리파이에 ROS 설치한다. 라즈베리파이를 위한 마이크로 sd카드가 필요하다. 마이크로 sd카드의 용량은 최소 8GB 이상이여야 한다. 마이크로 sd카드에 일반 라즈비안 대신 로보티즈에서 제공하는 라즈비안 파일을 설치하면 따로 ROS설치를 할 필요없이 사용할 수 있다 [4]. 설치 방법은 일반적인 라즈비안을 설치할 때와 같다.

```
pi@raspberrypi:~ $ sudo apt-get update
Hit:1 http://archive.raspbian.org/raspbian stretch InRelease
Hit:2 http://packages.ros.org/ros/ubuntu stretch InRelease
Get:3 http://packages.ros.org/ros/ubuntu stretch InRelease [4,661 B]
Err:3 http://packages.ros.org/ros/ubuntu stretch InRelease
  The following signatures couldn't be verified because the public key is not available: NO_PUBKEY F42EDFBA17C654
Fetched 4,661 B in 0s (819 B/s)
Reading package lists...
W: An error occurred during the signature verification. The repository is not updated and the previous index files will be used. GPG error: http://packages.ros.org/ros/ubuntu stretch InRelease: The following signatures couldn't be verified because the public key is not available: NO_PUBKEY F42EDFBA17C654
W: Failed to fetch http://packages.ros.org/ros/ubuntu/dists/stretch/InRelease  The following signatures couldn't be verified because the public key is not available: NO_PUBKEY F42EDFBA17C654
W: Some index files failed to download. They have been ignored, or old ones used instead.
```

Fig. 6. sudo apt-get update 에러

창에서 sudo apt-get update입력 후 사용해보면 에러가 그림6처럼 발생한다. 원인은 ROS 패키지 저장소의 키가 변경되었기 때문이다. 그림7에 나온 방법으로 키를 변경해주면 정상적으로 사용 가능하다.

Solution

Refer to New GPG keys deployed for packages.ros.org for the full write-up, but as a summary these are the commands for Ubuntu (and Debian):

Remove the old key

- On Ubuntu 16.04 and newer (with ROS Kinetic, Melodic, etc):

```
sudo apt-key del 421C365BD9FF1F717815A3E95523BAE8B01FA116
```

- On Ubuntu 15.10 and older (ie: 14.04, from here):

```
sudo apt-key del B01FA116
```

Import the new key

This command should work on all versions of Ubuntu:

```
# add the new key
sudo apt-key adv --keyserver 'http://keyserver.ubuntu.com:80' --recv-key C1CF6E31E6BADE8866B17284F42EDFBA17C654
```

Final step

At this point make sure to run a `sudo apt update`. You should receive no errors and subsequent `sudo apt install` commands should also now work.

Fig. 7. ROS 패키지 저장소 키 변경 [5]

라즈베리파이에서 OpenCR보드를 사용하기 위해 rosrun turtlebot3_bringup create_udev_rules를 터미널창에 입력 후 실행한

다.

터틀봇3와 컴퓨터와 네트워크 통신을 위해서 네트워크 설정한다. ifconfig를 터미널 창에 입력 후 실행하는 등의 방법으로 라즈베리파이의 IP주소를 확인한다. nano ~/.bashrc를 터미널 창에 입력 후 실행시켜 bashrc 파일을 연다. export ROS_MASTER_URI= 이후의 부분은 그림4와 똑같이 컴퓨터의 IP주소를 입력한다. export ROS_HOSTNAME= 이후 부분에는 라즈베리파이의 IP주소를 입력한다.

3. 터틀봇3의 OpenCR보드 설정

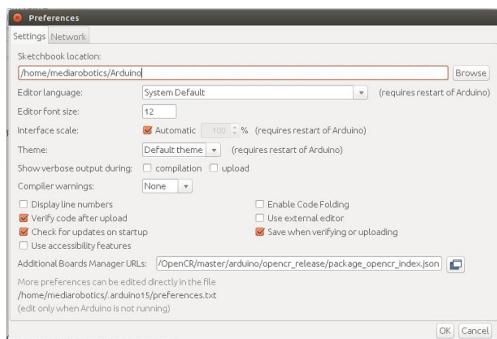


Fig. 8. 아두이노 환경설정

세 번째로 터틀봇3의 OpenCR보드 설정을 한다. 최신 버전의 아두이노 IDE를 컴퓨터에 설치한다. 아두이노 IDE를 실행 후 파일 탭에서 환경설정을 실행한다. 환경설정에서 그림8과 같이 추가보드 매니저 URLs(Additional Boards Manager URLs)에 https://raw.githubusercontent.com/ROBOTIS-GIT/OpenCR/master/arduino/opencr_release/package_opencr_index.json라고 입력한다.



Fig. 9. OpenCR 보드 매니저

도구 탭에서 보드, 보드 매니저를 실행 후 그림9와 같이 OpenCR을 검색 후 다운받는다.

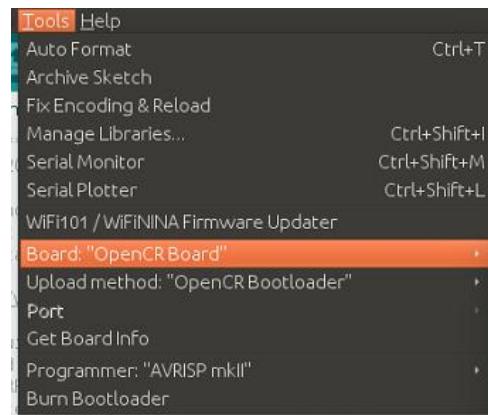


Fig. 10. Board "OpenCR Board"

도구 탭에서 보드 선택 후 OpenCR Board를 선택하여 그림10과 같은 상태만든다. 컴퓨터와 OpenCR보드를 연결한다. 도구 탭의 포트에서 연결가능한 포트를 선택한다.

다이나믹셀 설정을 위하여 아두이노의 예제파일 들 중 turtlebot3_setup_motor를 사용한다. 이 예제를 실행 할 때 반드시 하나의 모터만 OpenCR 보드에 연결된 상태로 실행해야한다. 배터리 또한 연결된 상태로 진행한다. 예제는 파일탭에서 예제, turtlebot3, turtlebot3_setup 탭에서 다이나믹셀 설정을 위한 예제코드 turtlebot3_setup_motor을 찾을 수 있다. 예제 실행 후 시리얼 모니터창을 연다. 만약 실행이 잘 안될 경우엔 OpenCR 보드의 리셋 버튼을 잠시 누른 후에 다시 예제를 실행한다.

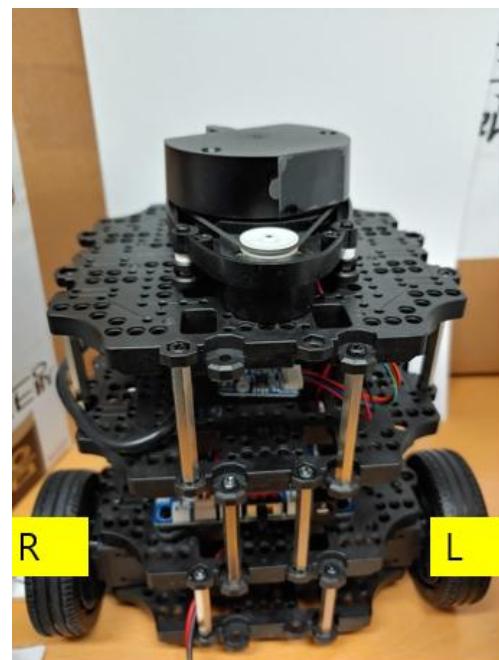


Fig. 11. 터틀봇3 버거 정면

시리얼 모니터창에서 다이나믹셀 설정을 할 수 있다. 왼쪽 모니터만 연결된 상태라고 가정했을 때 시리얼 모니터창에 1을 입력하면

OpenCR보드에 연결된 다이나믹셀을 찾아낸다. Y를 입력하면 찾아낸 모터를 왼쪽 다이나믹셀로 설정한다. 설정이 제대로 되었다면 시리얼 모니터창에 ok라고 출력된다. 시리얼 모니터창에 3을 입력하면 왼쪽으로 설정한 다이나믹셀이 잠시 동안 작동한다. 왼쪽 오른쪽은 그림11을 보고 구분할 수 있다. 왼쪽 다이나믹셀이 제대로 작동한다면 그 다음은 오른쪽 다이나믹셀만 연결 후 시리얼 모니터창에 2를 입력하여 설정을하고 4를 입력하여 테스트를 한다.

양쪽 다이나믹셀 설정을 완료한 후 파일의 예제, turtlebot3,turtlebot3_burger 템에서 turtlebot3_core예제를 열고 실행한다. 실행 후 OpenCR보드의 sw1을 몇 초간 누르면 앞으로 30cm 이동하며 sw2를 몇 초간 누르면 180도 회전을 한다.

다이나믹셀 설정 과정에서 다이나믹셀의 고장이나 OpenCR보드의 고장, 혹은 연결 문제 때문에 정상적으로 작동이 안될 수가 있다. 이 경우 로보티즈 사이트에서 제공하는 자가진단 방법을 이용하면 문제의 원인을 파악할 수 있다 [6].

tall-pkgs-key-not-working/
[6] Self Diagnosis Dynamixel, <https://www.robotis.com/module/selfcheck.php>

III. Conclusions

본 논문은 ROS를 터틀봇 3에 설치 적용 및 제어하면서 나오는 오류 사항들을 기술한 논문이다. 본 논문을 통해 오픈소스의 문제점 및 접근 방향에 대한 좋은 가이드라인이 생길 수 있음을 확신한다. 이를 토대로 차후 본 저자들 같은 초보 ROS 연구자들이 ROS의 버그 및 어려움을 쉽게 발견 보완할 수 있는 오픈소스 튜토리얼을 기대한다.

ACKNOWLEDGEMENT

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 SW중심 대학사업의 연구결과로 수행되었음(2018-0-00209)

REFERENCES

- [1] Erico Guizzo, "The TurtleBot3 Teacher," IEE Spectrum, Vol. 54, issue8, pp19-20, Aug 2017
- [2] Morgan Quigley, "ROS: an open-source Robot Operating System," ICRA workshop on open source software, Vol. 3, No.3, pp. 1-6, 2009.
- [3] Turtlebot e-manual, <http://emanual.robotis.com/docs/en/platform/turtlebot3/overview/>
- [4] Rasbian Download, <http://www.robotis.com/service/download.php?no=1738>
- [5] ROS Remove old Key and import the new key. <https://answers.ros.org/question/325039/apt-update-fails-cannot-import-key/>