

ROS_PROGRAM_김보람_김호준_이중섭

ROS?

ROS(Robot Operating System)이란 복잡한 로봇의 행동을 간단하게 제작할 수 있게 도와주는 소프트웨어 라이브러리와 툴의 플랫폼이다.

실습 환경

Turtlebot3	OS	ROS
· Single Board Computer (Broadcom BCM2837B0) - Raspberry 4 PI 4B) · OpenCR (STM32F746) · DYNAMIXEL (XL430-W350-T) · LIDAR	· Ubuntu 20.04.03 LTS (Focal Fossa)	· ROS 1.0 Noetic

개발 도구

- Rviz : ROS에서 제공하는 3D visualization tool
- RQT : Qt 기반의 framework for GUI tool
- Gazebo : 3D 시뮬레이터

명령어

roscore	catkin	roslaunch
ROS Master 실행	Package 관리 및 빌드	여러 Node를 실행

ROS Message 통신

최소 단위 실행 프로그램 'node'는 또 다른 node와 message를 통해 데이터를 주고받음.			
방식			
Topic	Service	Action	node 안에서 사용되는 parameter는 외부에서 실시간 변경 가능
Publisher와 Subscriber가 Topic을 주고 받음	응답하는 server와 요청, 응답받는 client 사이의 일회성 통신	비동기 방식의 server와 client 사이의 통신 방식	

중요 URL

- ROS Wiki (<http://wiki.ros.org>)
- Reference Book (<https://github.com/robotpilot/ros-seminar>)

ROS_PROGARM_김보람_김호준_이중섭

ROS 개발환경 구축 관련 명령어

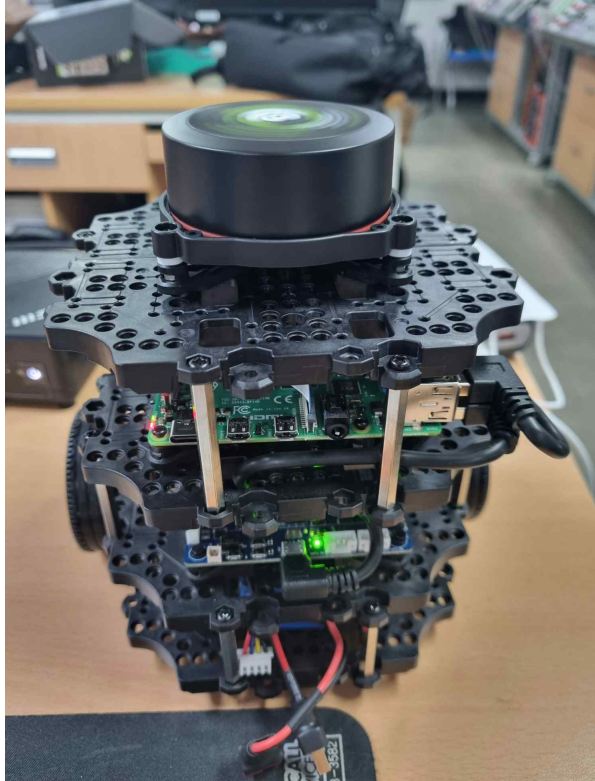
lsb_release -a	ubuntu version 확인
apt-get	package 관리 명령어
sudo	super-user 권한으로 실행
wget	웹에서 파일을 다운로드
chmod 755	사용권한 변경
bash	shell에서의 실행 프로그램 (bashrc-시스템의 기본적인 환경변수들을 저장해둔 것)
.sh (shell script)	여러 개의 명령어 또는 프로그램 실행

mstsc	windows 환경의 원격데스크톱 연결 제어 프로그램
xrdp	원격 접속 허용 프로그램(ubuntu 설치)
xfce	GUI를 위한 것(ubuntu 설치)
sudo dpkg --configure -a	dpkg was interrupted 오류 해결방법
systemctl status xrdp	xrdp 패키지 설치 여부 확인

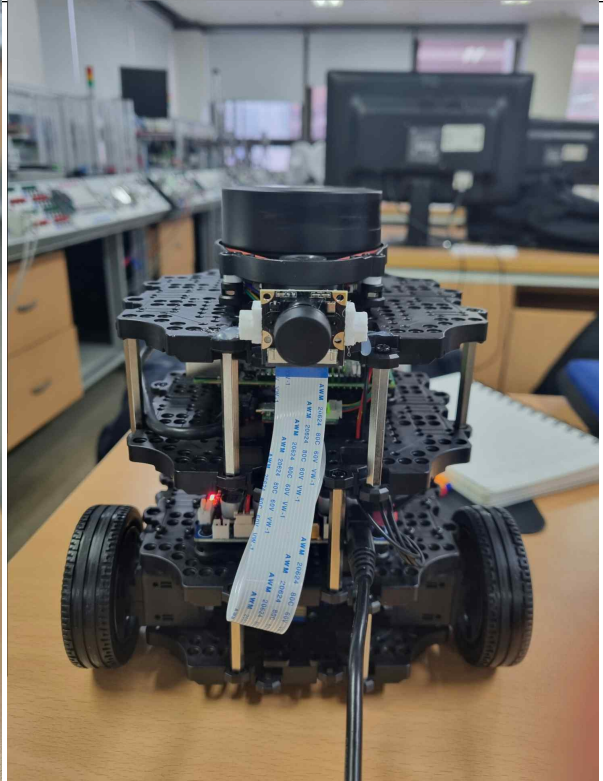
sudo apt-get update	설치된 패키지들의 새 버전 유무 확인
sudo apt-get upgrade	최신 버전으로 업그레이드함
netstat	네트워크 상태 확인 (option-a,n,t,p,u,l)
sudo apt-get remove ros-*	패키지 삭제, 설정파일 남김
sudo apt-get purge ros-*	패키지, 설정파일 삭제
sudo apt autoremove	불필요한 의존성 패키지 함께 삭제
rosversion -d	삭제 확인

cd ~/carkin_ws && catkin_make	경로 & build
echo >>	선택한 파일 하위에 넣어 두는 것
rqt_graph	현재 실행 중인 노드들의 정보를 GUI 형태로 시각화

TurtleBot3 burger 조립

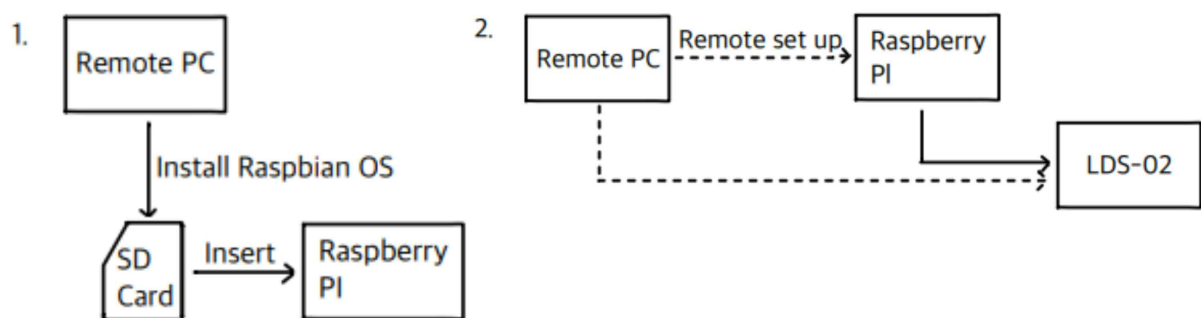


후면



전면

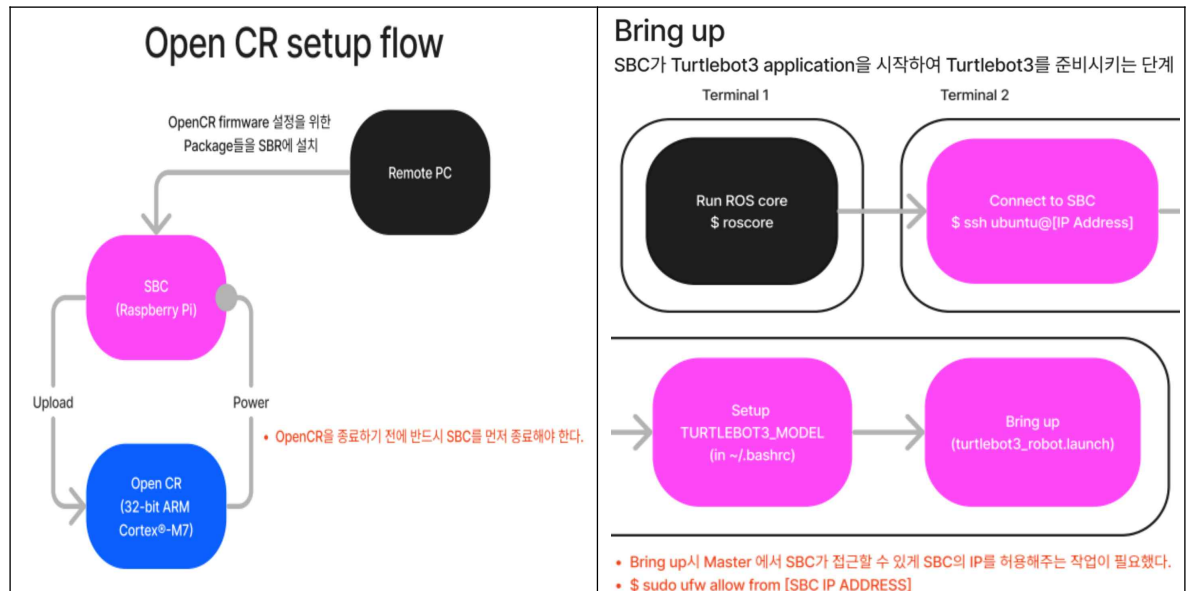
Set up flow



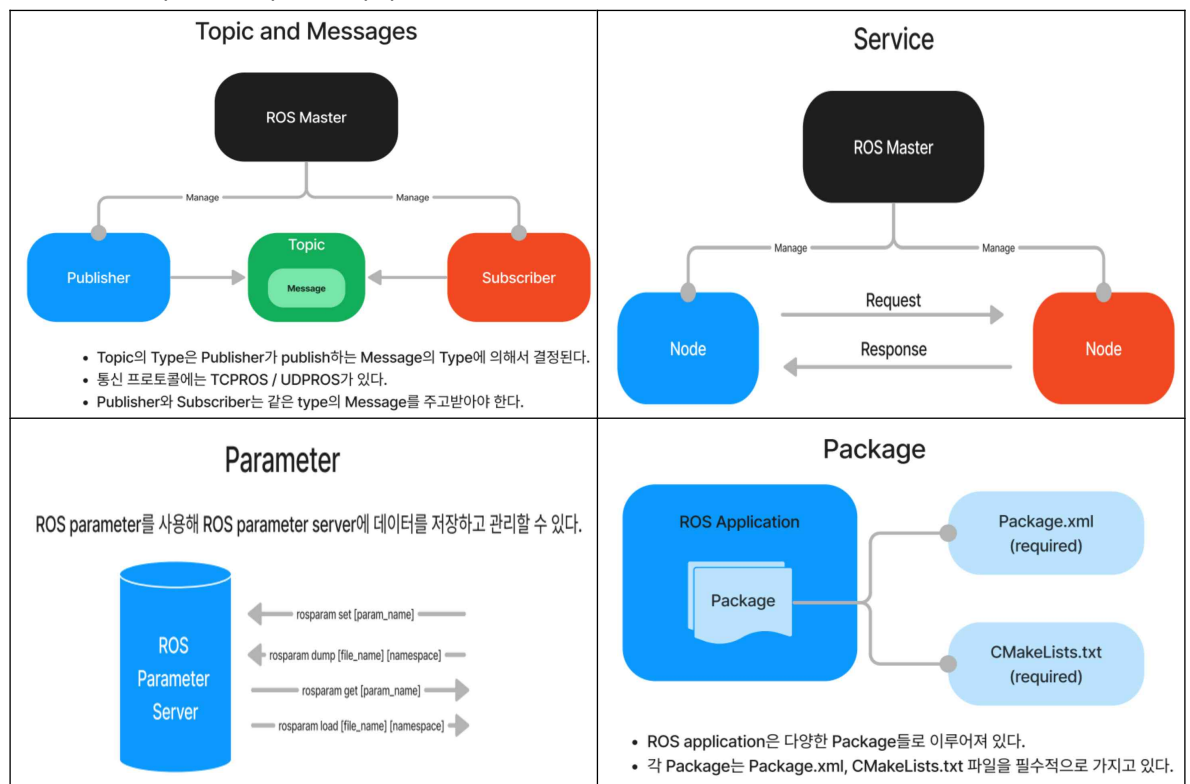
1. Remote PC에서 SD카드로 Raspbian OS Baking 후 Raspberry Board에 삽입
2. Remote PC에서 Raspberry PI 원격 접속 (동일 네트워크에 속해 있어야 함)
3. Raspberry PI 네트워크 설정 및 호스트 네임 설정

ROS 개발환경 구축

1. 주요 개념 도식화



2. ROS명령어 관련 개념 도식화



SLAM을 이용한 MAP 그리기

1. 주요 개념

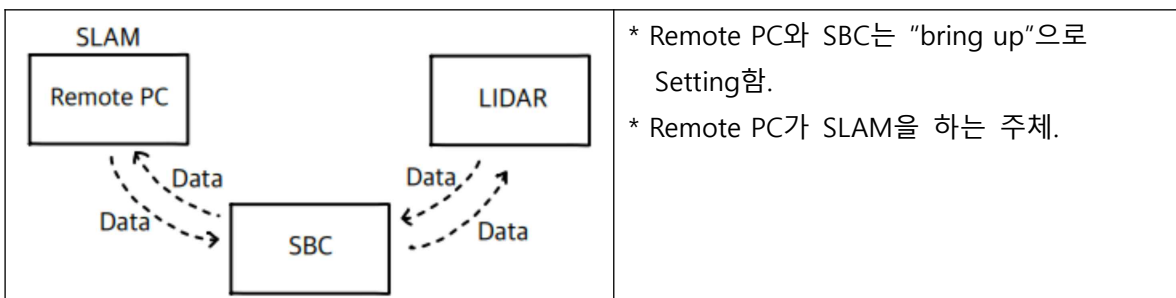
SLAM(Simultaneous localization and Mapping) : 동시적 위치 추정 및 지도 작성

gmapping : Map을 그리는 Package

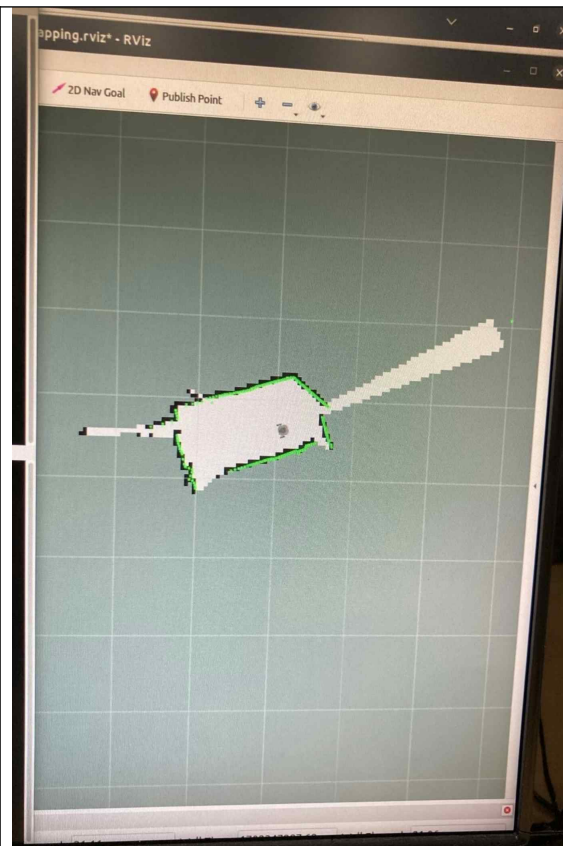
ToF(Time of Flight) : 빛(레이저)을 이용해 거리와 시간을 계산하여 Object의 위치 추측

Navigation : 목적지까지 이동할 수 있는 알고리즘

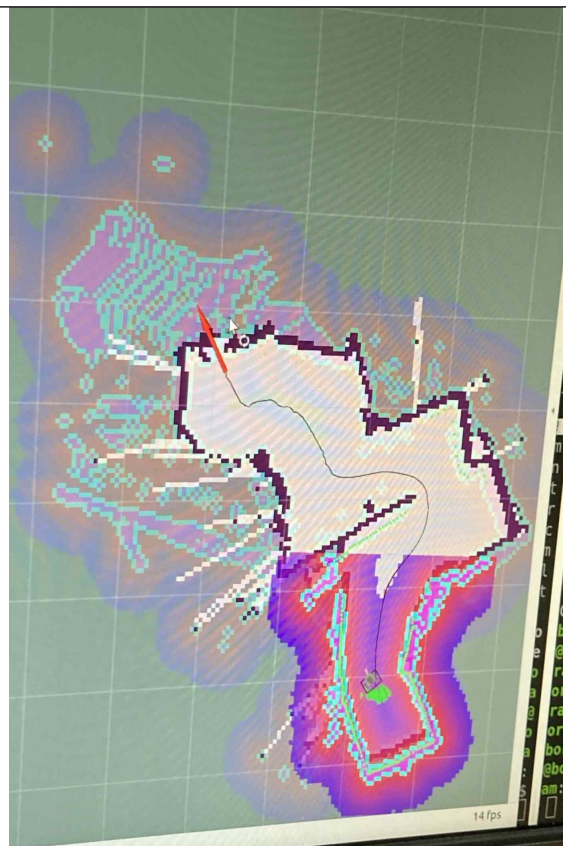
2. 순서 도식화



3) 관련 사진



Map 그리기



Navigation

SLAM을 이용한 MAP 그리기(오차값 줄이기)

1. 주요 개념

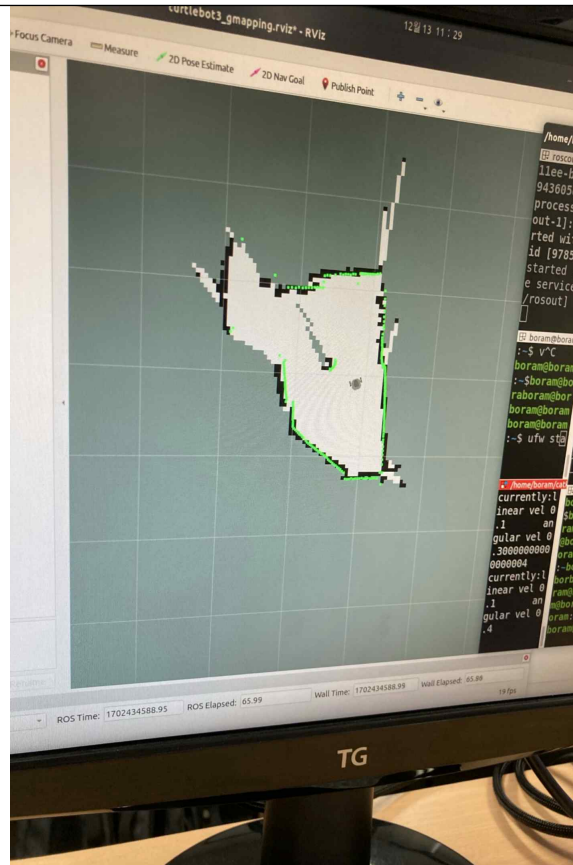
Particle Filter(Sequential Monte Carlo or Monte Carlo Localization) : Noise가 있는 환경에서 측정된 data를 filter로 사용해 실제 위치를 추정하는 방법. 가능성이 높은 Particle만 남기고 가능성이 낮은 Particle은 지움.

cost(충돌값)map : 장애물 회피. global과 local 2종류 존재.

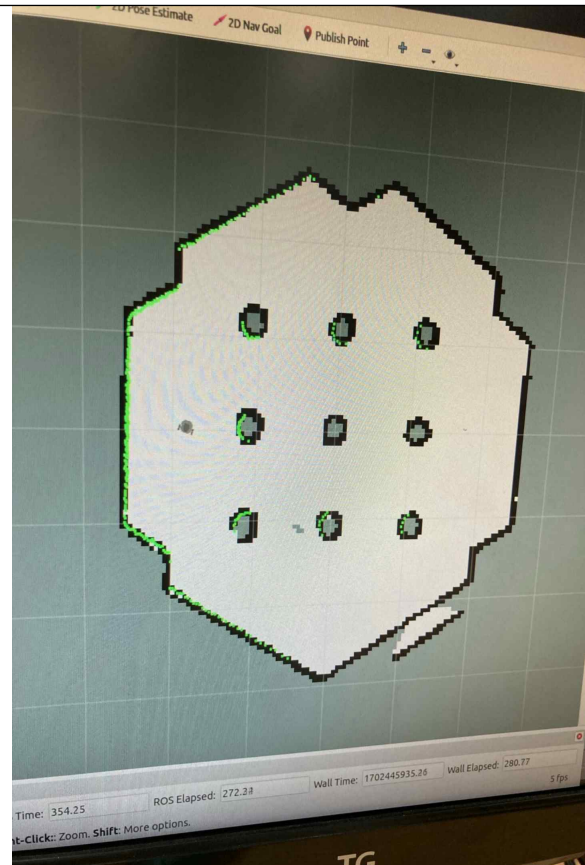
collision avoidance : 충돌 전에 회피하는 DWA 알고리즘 사용

Dynamic Window Approach : 하드웨어의 한계로 최대 허용 속도가 존재. 이 때 turtlebot의 행위를 Approach라고 함.

2. 관련 사진



Mapping



시뮬레이션을 이용한 Mapping

Autonomous Driving

1. 주요 개념

Camera calibration : 카메라 보정

intrinsic calibration : 카메라 렌즈의 광학적 요소를 보정.

extrinsic calibration : 카메라와 외부 공간과의 기하학적 관계에 의한 보정.

2. 순서

- 1) Remote PC(roscore)에서 원격으로 SBC 접속한 다음, bulid
- 2) SBC : 공간정리 -> 'open cv' down -> making -> noetic camera 실행
- 3) Remote pc에서 실행
- 4) 카메라 보정 : Intrinsic camera calibration, Extrinsic camera calibration, Lane detection
- 5) 수정한 매개변수 반영
- 6) Remote PC에서 원격으로 Raspberry PI에 접속한 다음, 반영값 확인하기 (calibration -> action)
- 7) 자율주행 진행 (문제 발생 시, 매개변수 재설정)

3. 관련 사진

