Universal Robots ROS Driver

ROS Melodic

김은수

kit 금오공과대학교
Kumoh National Institute of Technology

참고

- https://www.universal-robots.com/ko/
- https://github.com/UniversalRobots/Universal_Robots_ROS_Driver
- https://github.com/UniversalRobots/Universal_Robots_ROS_Driver/blob/master/ur_robot_driver/resources/externalcontrol-1.0.2.urcap
- https://github.com/UniversalRobots/Universal_Robots_ROS_Driver/blob/master/ur_robot_driver/doc/install_urcap_cb3.md

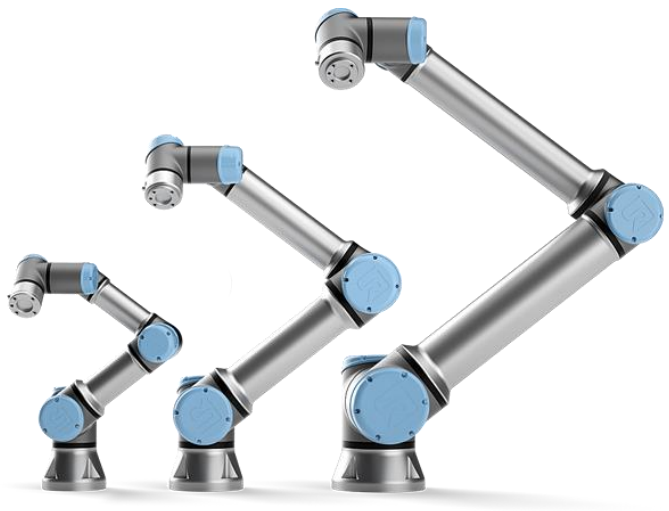
목차

Development Environment

- OS
  - Ubuntu 18.04
- ROS
  - ROS Melodic
- UR3
  - CB3 소프트웨어 버전 >=3.7

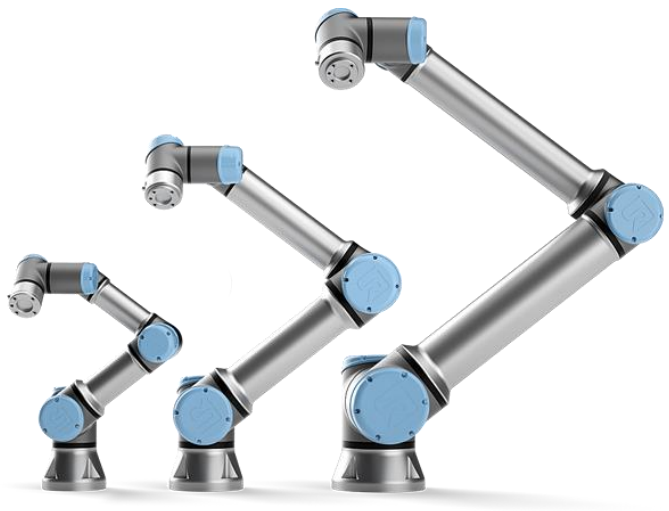
Universal Robots ROS Driver

- Universal Robots ROS Driver
  - 모든 사용자들에게 UR 로봇과 ROS 간에 안정적이고 지속 가능한 인터페이스를 제공



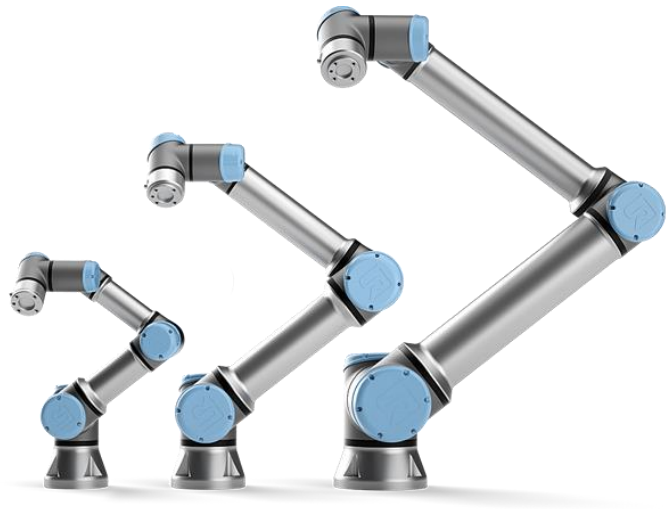
Universal Robots ROS Driver

- 특징
  - 모든 CB3 (소프트웨어 버전 >= 3.7) 및 e-Series (소프트웨어 >=5.1) 로봇에서 작동
  - URCaps 시스템을 사용하면 ROS 측에서 전송된 제어 명령을 처리하는 프로그램이 로봇에서 실행
  - 티치 펜던트(TF)에서보다 복잡한 UR 프로그램의 일부로 ROS 구성 요소를 사용할 수 있음



Universal Robots ROS Driver

- 패키지 내용
  - controller_stopper : 로봇의 상태에 따라 컨트롤러를 중지했다가 다시 시작하는 작은 외부 도구. 로봇이 ROS에서 보낸 명령을 받아들이지 않는 상태일 때 유용
  - ur_calibration : 로봇의 교정 정보를 추출 및 변환
  - ur_controllers : 속도 확장 인식 컨트롤러와 함께 이 드라이버에 도입된 컨트롤러
  - ur_robot_driver : 실제 드라이버 패키지



Universal Robots ROS Driver

- 설치

```
# source global ros
$ source /opt/ros/melodic/setup.bash

# create a catkin workspace
$ mkdir -p catkin_ws/src && cd catkin_ws

# clone the driver
$ git clone https://github.com/UniversalRobots/Universal_Robots_ROS_Driver.git src/Universal_Robots_ROS_Driver

# clone fork of the description. This is currently necessary, until the changes are merged upstream.
$ git clone -b calibration_devel https://github.com/fmauch/universal_robot.git src/fmauch_universal_robot

# install dependencies
$ sudo apt update -qq
$ rosdep update
$ rosdep install --from-paths src --ignore-src -y

# build the workspace
$ catkin_make

# activate the workspace (ie: source it)
$ source devel/setup.bash
```

[PC]

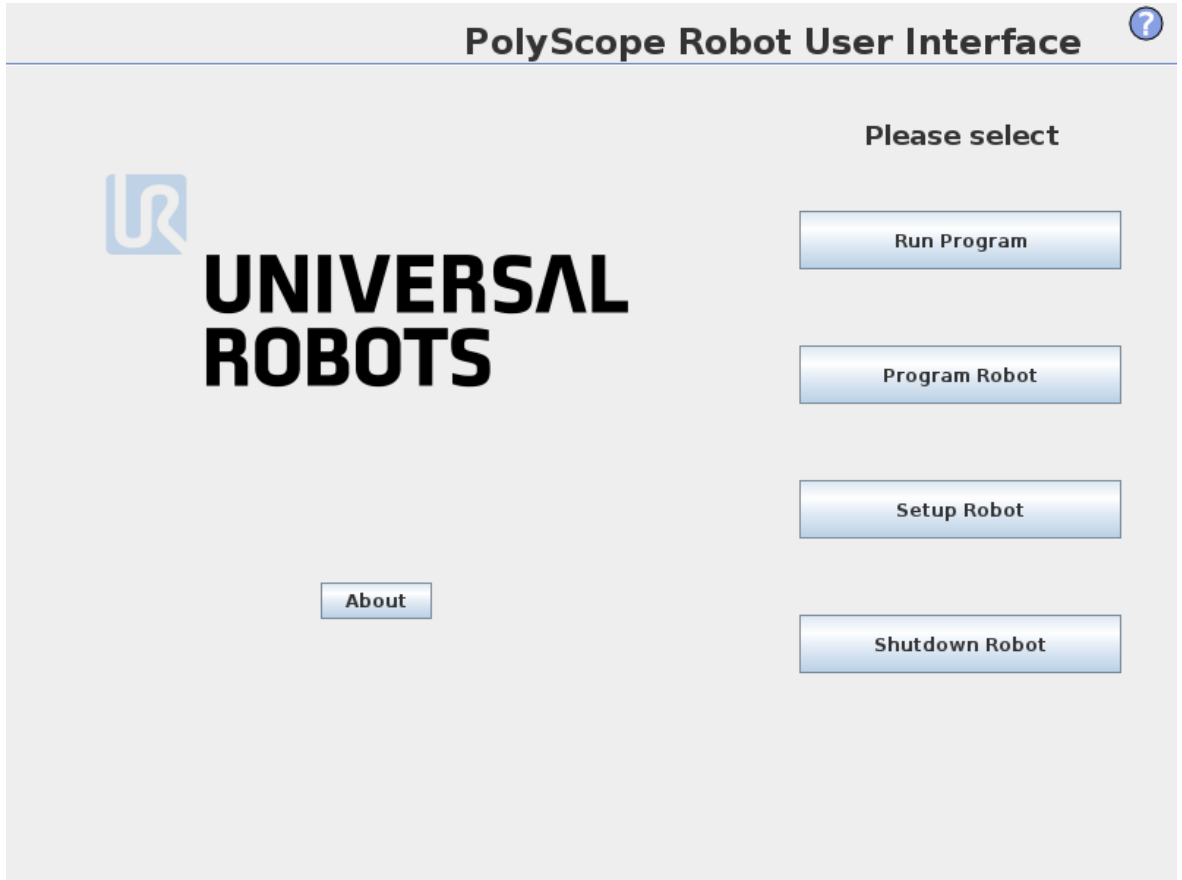
Universal Robots ROS Driver

- ur_robot_driver에 대한 UR 로봇 설정
  - ur_robot_driver를 사용하려면 아래 externalcontrol-1.0.2.urcap 설치
  - https://github.com/UniversalRobots/Universal_Robots_ROS_Driver/blob/master/ur_robot_driver/resources/externalcontrol-1.0.2.urcap
  - USB에 urcap 파일을 담고, TP에 연결

[PC]

Universal Robots ROS Driver

- ur_robot_driver에 대한 UR 로봇 설정

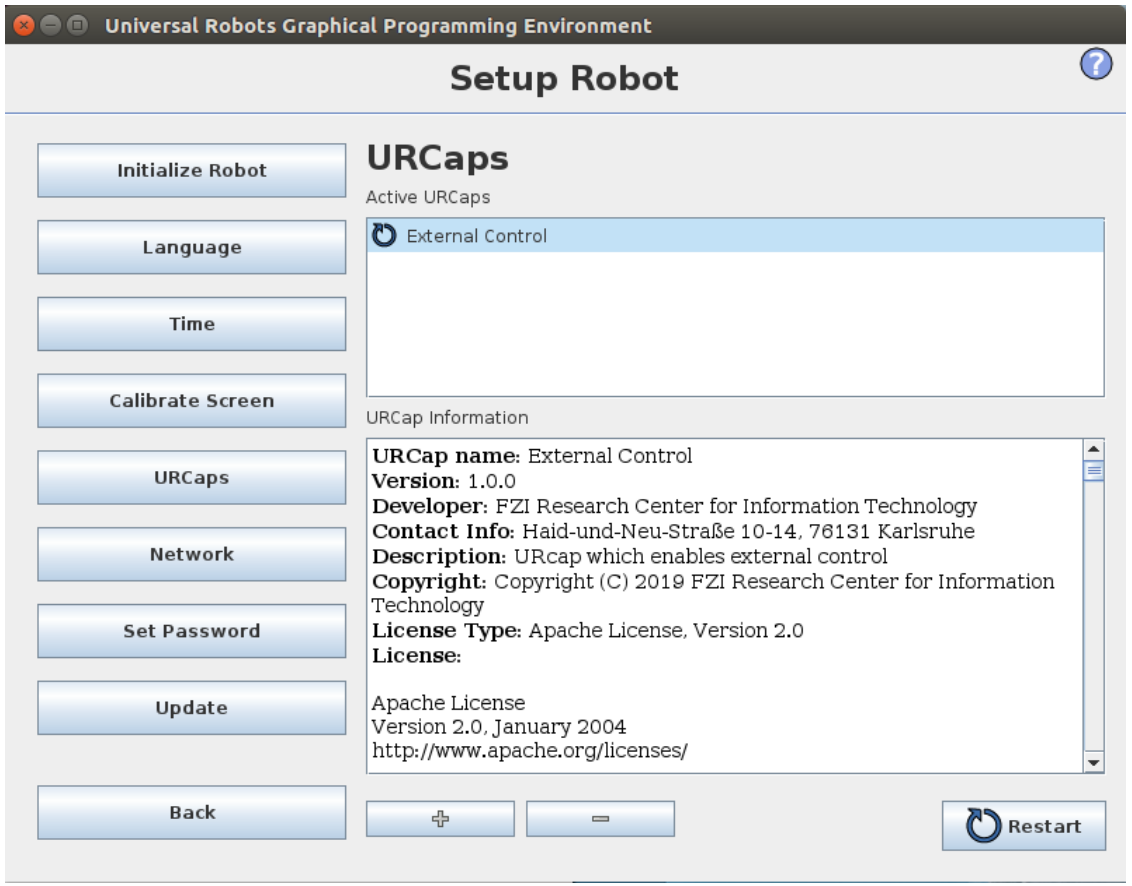


[TP]

- 시작 화면 – 로봇 설정 - URCaps

Universal Robots ROS Driver

- ur_robot_driver에 대한 UR 로봇 설정

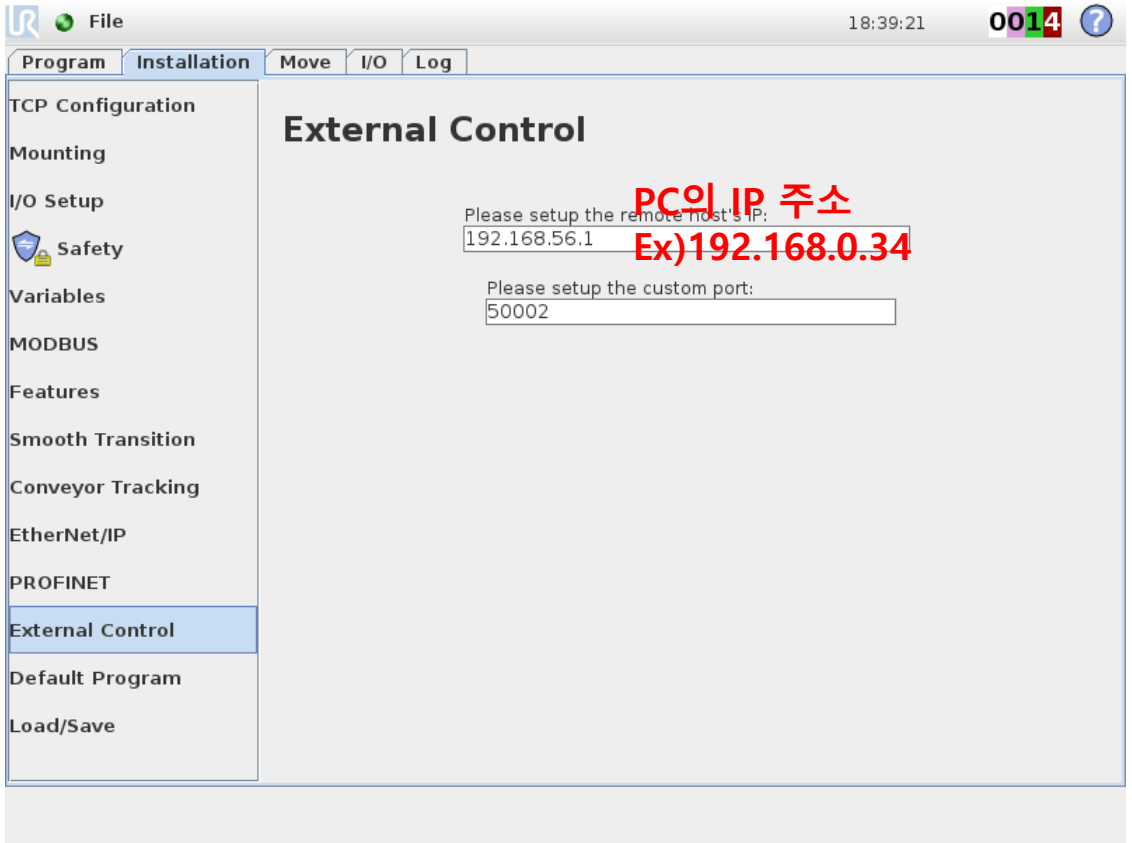


- 하단의 + 부호 클릭
  => 연결된 USB의 urcap 파일 표시됨
- 파일을 선택하고 열기, Restart

[TP]

Universal Robots ROS Driver

- ur_robot_driver에 대한 UR 로봇 설정

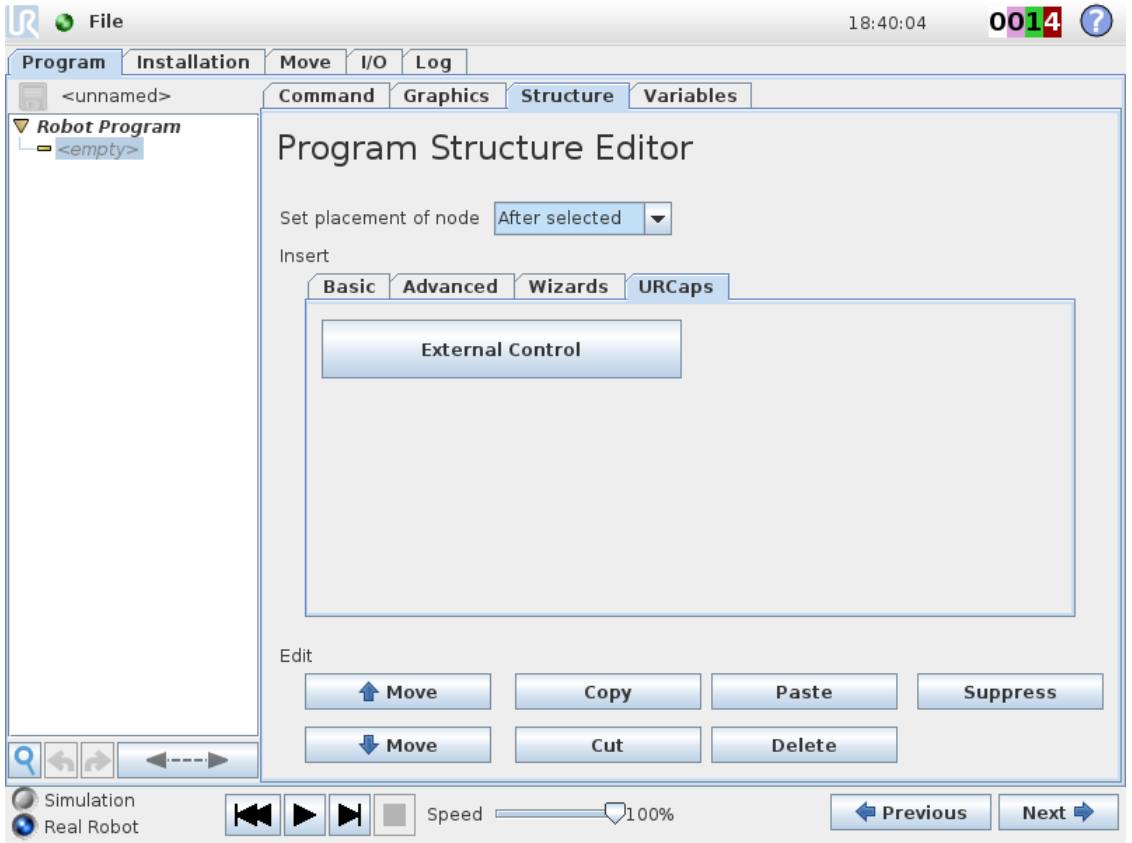


[TP]

- 재부팅 후 설치 섹션에서 External Control 클릭
- ROS 드라이버를 실행할 외부 PC의 IP 주소를 설정
- custom port는 그대로 유지

Universal Robots ROS Driver

- ur_robot_driver에 대한 UR 로봇 설정

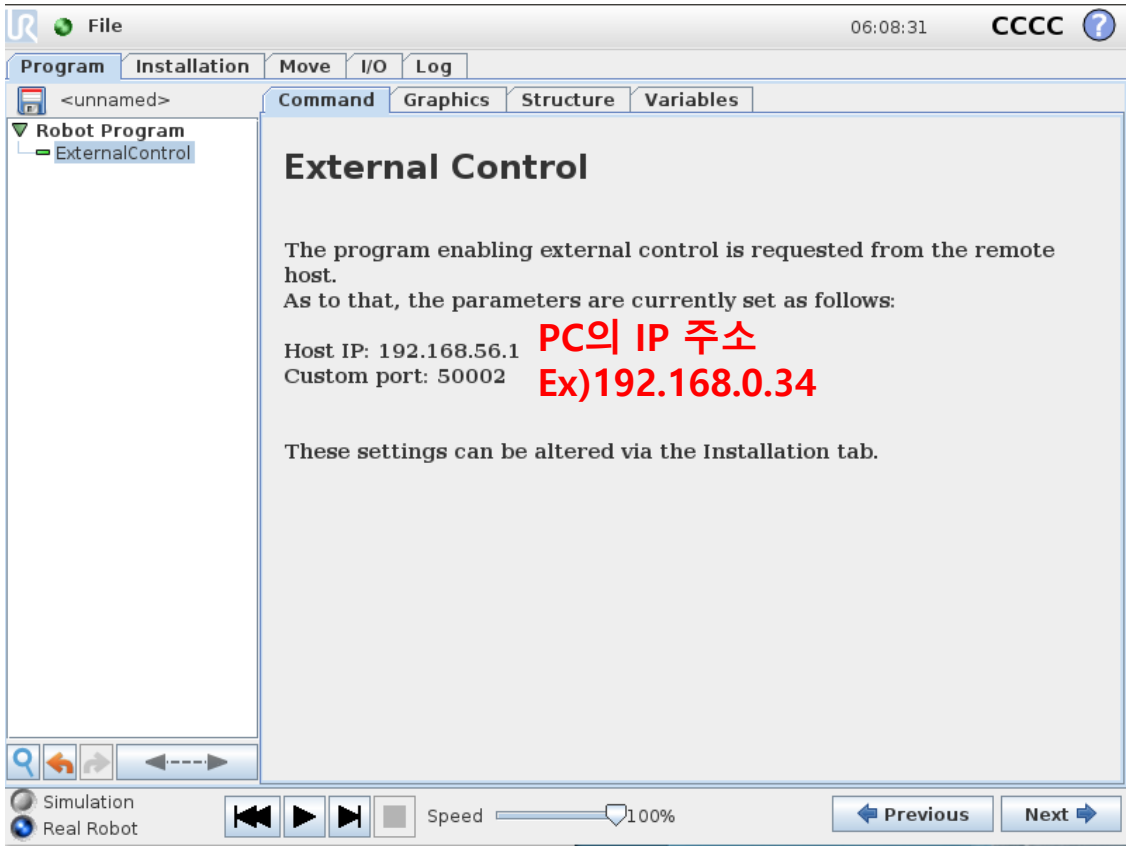


- 새 URCap을 사용하려면 새 프로그램 작성하고 구조-URCaps 들어가서 External Control 노드를 프로그램 트리에 삽입

[TP]

Universal Robots ROS Driver

- ur_robot_driver에 대한 UR 로봇 설정



[TP]

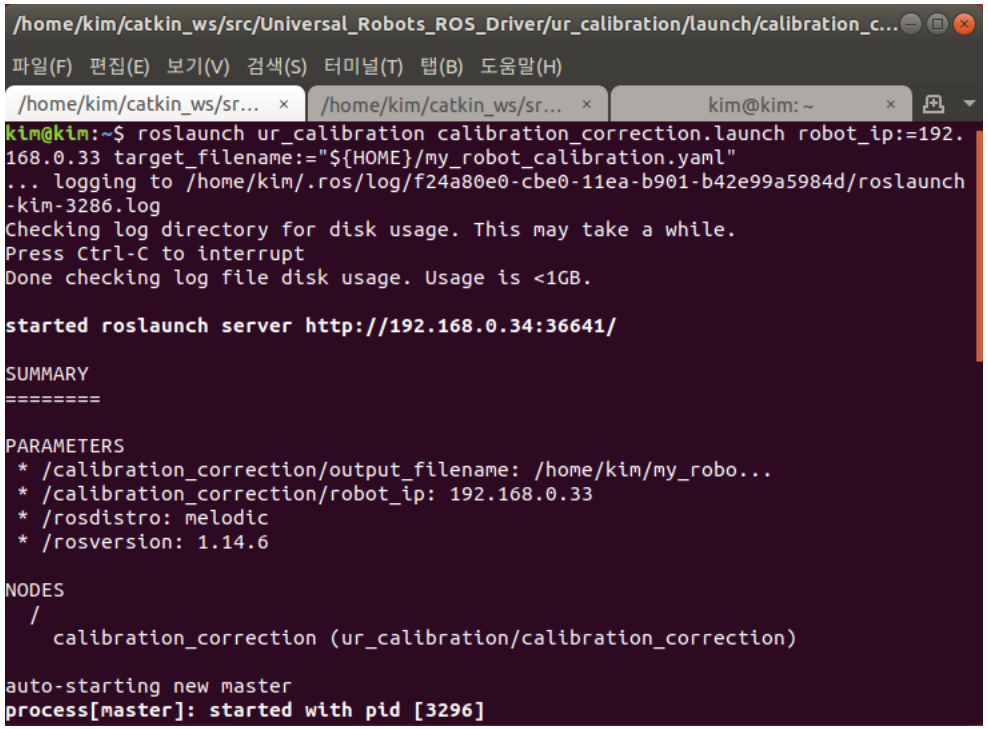
- 명령 탭을 다시 클릭하면 설치 내에 입력한 설정이 표시됨
- 확인한 다음 프로그램 저장

Universal Robots ROS Driver

- 교정 정보 추출

```
$ roslaunch ur_calibration calibration_correction.launch  \
   robot_ip:=<robot_ip> target_filename:="${HOME}/my_robot_calibration.yaml"
```

UR3의 IP 주소
Ex)192.168.0.33



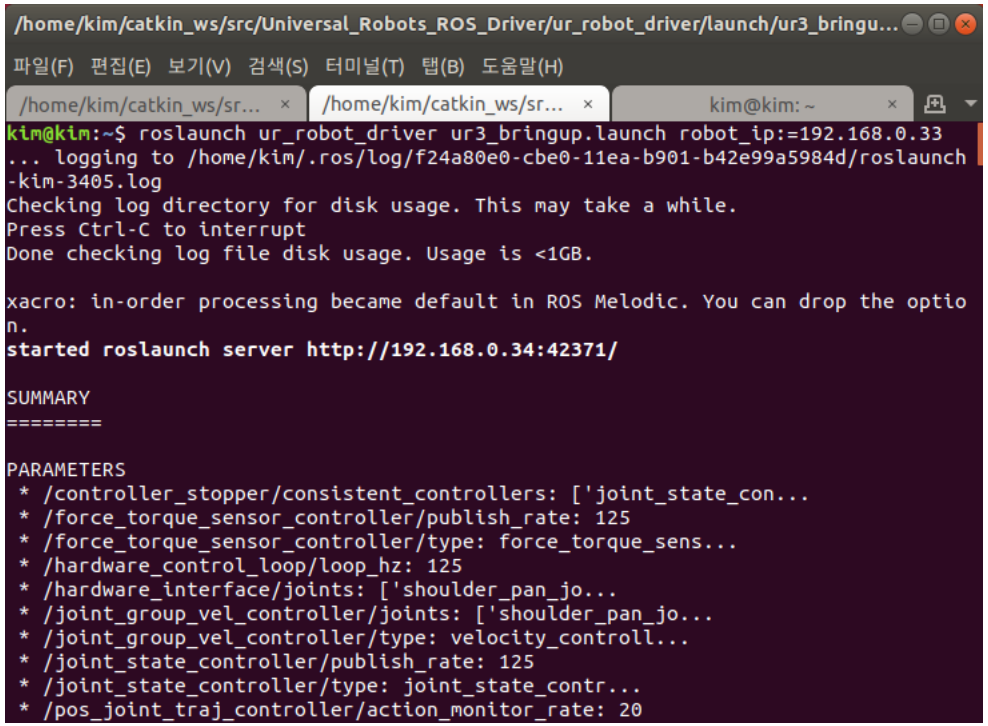
[PC]

Universal Robots ROS Driver

- 빠른 시작

```
$ roslaunch ur_robot_driver <robot_type>_bringup.launch robot_ip:=192.168.0.33
```

ur3



[PC]

Universal Robots ROS Driver

- TP에서 ExternalControl 삽입된 프로그램 실행

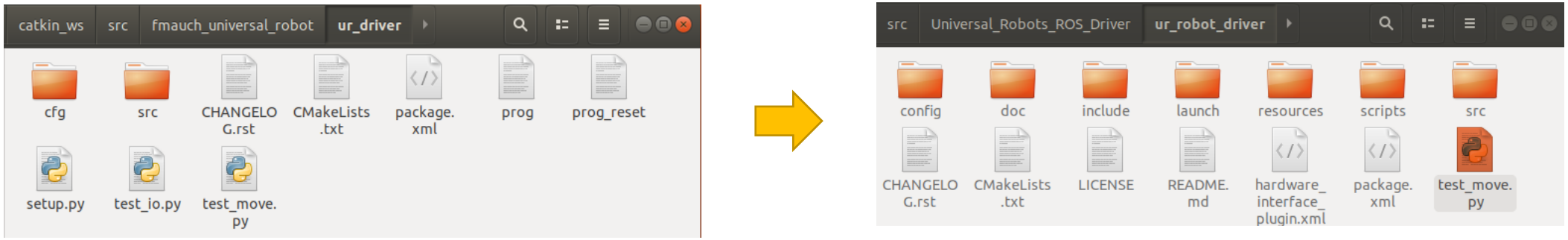
- 실행 후, 터미널에 위와 같이 뜨면 성공

```
[ INFO] [1595399285.009571277]: Robot requested program
[ INFO] [1595399285.009740909]: Sent program to robot
[ INFO] [1595399285.107951536]: Robot ready to receive control commands.
```

[TP]

Universal Robots ROS Driver

- fmauch_universal_robot/ur_driver/test_move.py를 Universal_Robots_ROS_Driver/ur_robot_driver로 이동



[PC]

Universal Robots ROS Driver

- test_move.py 텍스트편집기 열기
- 안전을 위해 Q1, Q2, Q3 값 수정

열기(O) test_move.py ~/catkin_ws/src/Universal_Robots_ROS_Driver/ur_robot_driver 저장(S)

```
#!/usr/bin/env python
import time
import roslib; roslib.load_manifest('ur_driver')
import rospy
import actionlib
from control_msgs.msg import *
from trajectory_msgs.msg import *

JOINT_NAMES = ['shoulder_pan_joint', 'shoulder_lift_joint', 'elbow_joint',
               'wrist_1_joint', 'wrist_2_joint', 'wrist_3_joint']
#Q1 = [2.2,0,-1.57,0,0,0]
#Q2 = [1.5,0,-1.57,0,0,0]
#Q3 = [1.5,-0.2,-1.57,0,0,0]

Q1 = [0.,-1.57,0.,-1.57,0.,0.]
Q2 = [0.,-1.57-1.57/6,0.,-1.57-1.57/6,0.,0.]
Q3 = [0.,-1.57+1.57/6,0.,-1.57+1.57/6,0.,0.]



client = None
```

- /follow_joint_trajectory을 /scaled_pos_joint_traj_controller/follow_joint_trajectory로 수정

```
try:
    rospy.init_node("test_move", anonymous=True, disable_signals=True)
    client = actionlib.SimpleActionClient('/scaled_pos_joint_traj_controller/follow_joint_trajectory',
```

[PC]

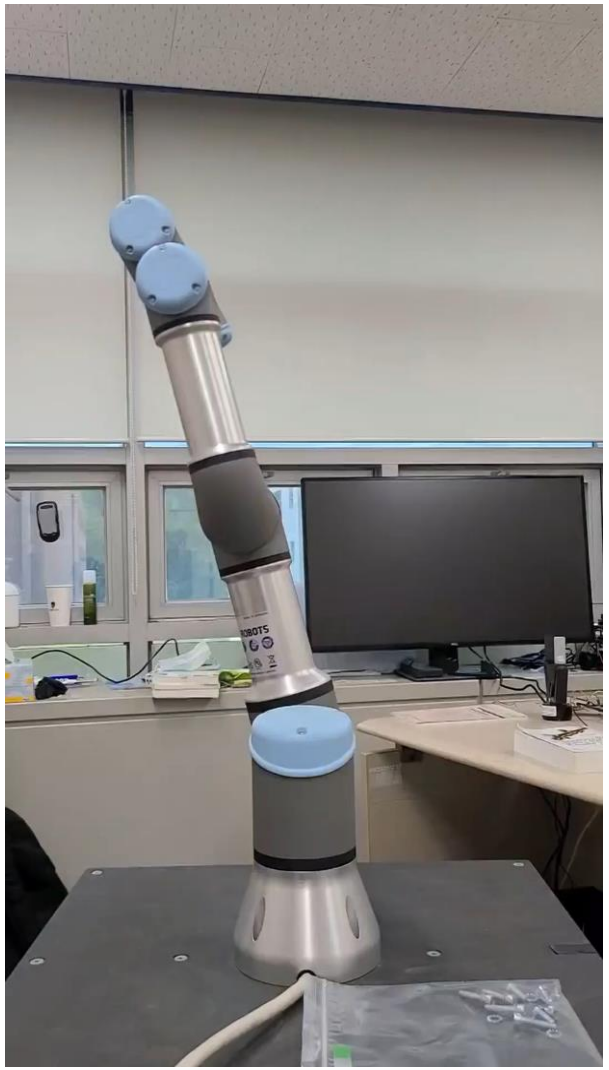
Universal Robots ROS Driver

- test_move.py 저장 후 실행

```
$ rosrun ur_robot_driver test_move.py
```

```
kim@kim:~$ rosrun ur_robot_driver test_move.py
Waiting for server...
Connected to server
```

- ROS Driver를 통하여 ROS에서 UR3 Control 성공!!

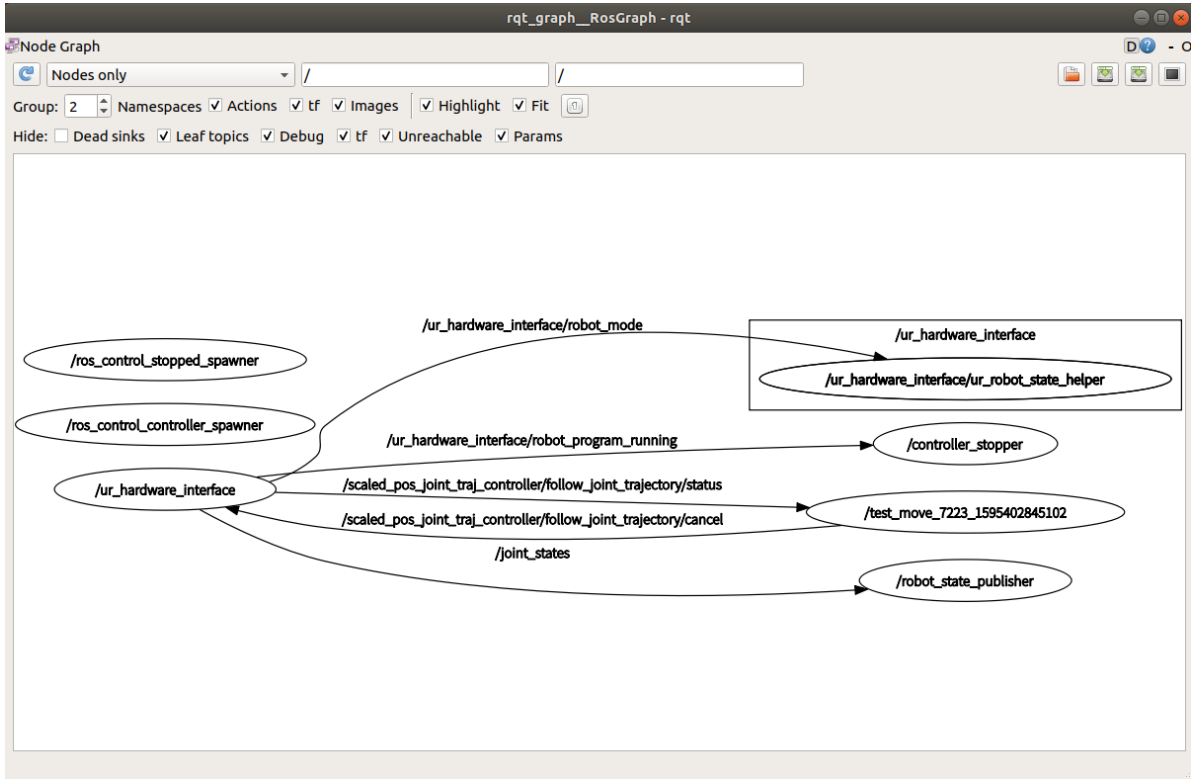


[PC]

Universal Robots ROS Driver

- rqt_graph

```
$ rqt_graph
```



[PC]

- THANK YOU