

Maxon + EPOS4 + ROS1 + CSP

Jeong Gowoon

2025.11.30.

Department of Mechanical Engineering
Chonnam National University

Overview

➤ 참고 자료

- https://github.com/Roboprotos/maxon_epos4_ros1
- maxon_EPOS4_ROS1_ros_canopen_Documentation_2022-02.pdf

➤ 수정된 코드 및 github 링크

- https://github.com/Gowoon12/maxon_epos4_ros1_csp.git
- /ShareFolder/04_Lab_Equipment_Related/maxon_epos4_ros1_csp

➤ 테스트 환경

- Ubuntu 20.04 LTS
- ROS Noetic

Catkin workspace 및 패키지 설치

➤ 기존에 사용하던 `catkin_ws`의 의존성 문제를 피하기 위해 새로운 `catkin workspace`를 만드는 것이 좋음

- `mkdir -p ~/maxon_ws/src`
- `cd ~/maxon_ws`
- `catkin_make`
- `source ~/maxon_ws/devel/setup.bash`

➤ 기존에 사용하던 `catkin_ws`의 의존성 문제를 피하기 위해 새로운 `catkin workspace`를 만드는 것이 좋음

➤ `maxon_epos4_ros1` 패키지 설치 (`catkin_make`를 사용하면 오류날 수 있어서 `maxon_epos4_ros_canopen`만 빌드해야함.)

- `cd ~/maxon_ws/src`
- `git clone https://github.com/Roboprotos/maxon_epos4_ros1.git`
- `cd ..`
- `catkin_make --pkg maxon_epos4_ros_canopen`
- `source ~/maxon_ws/devel/setup.bash`

Hardware setup

➤ CAN cable (CAN-USB)

3.3.5.10 CAN 1 (X11) & CAN 2 (X12)

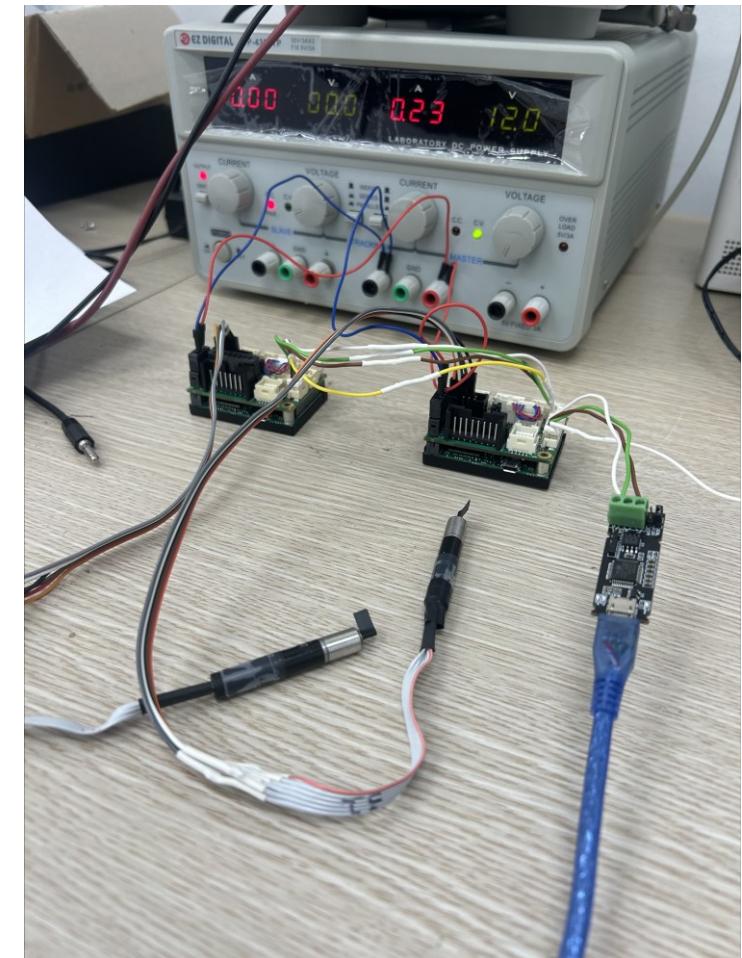
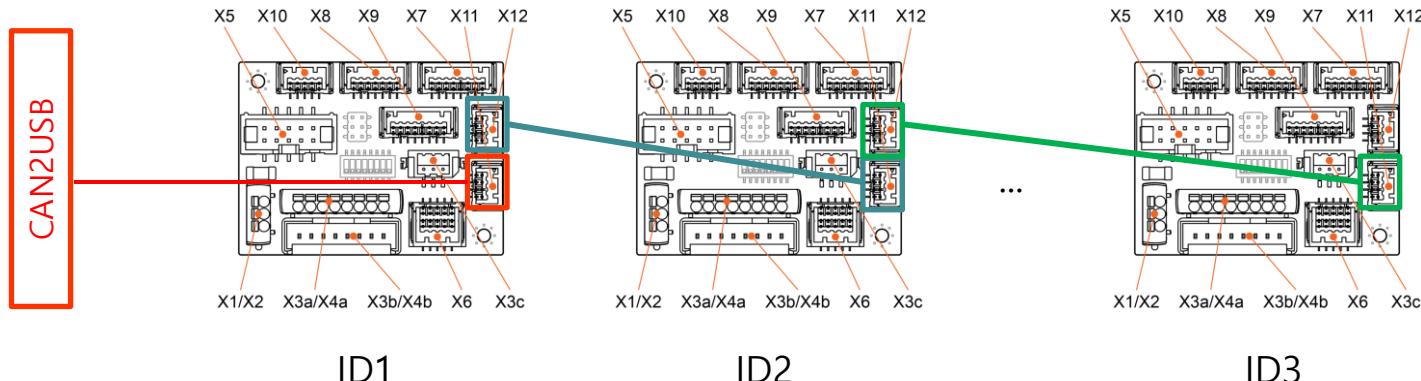


Figure 3-24 CAN 1 connector X11/CAN 2 connector X12

X11/X12 Head A Pin	Prefab Cable Color	520857 Head B Pin	520858 Head B Pin	Signal	Description
1	white	7	1	CAN high	CAN high bus line
2	brown	2	2	CAN low	CAN low bus line
3	green	3	3	GND	Ground
4	Shield	5	4	Shield	Cable shield

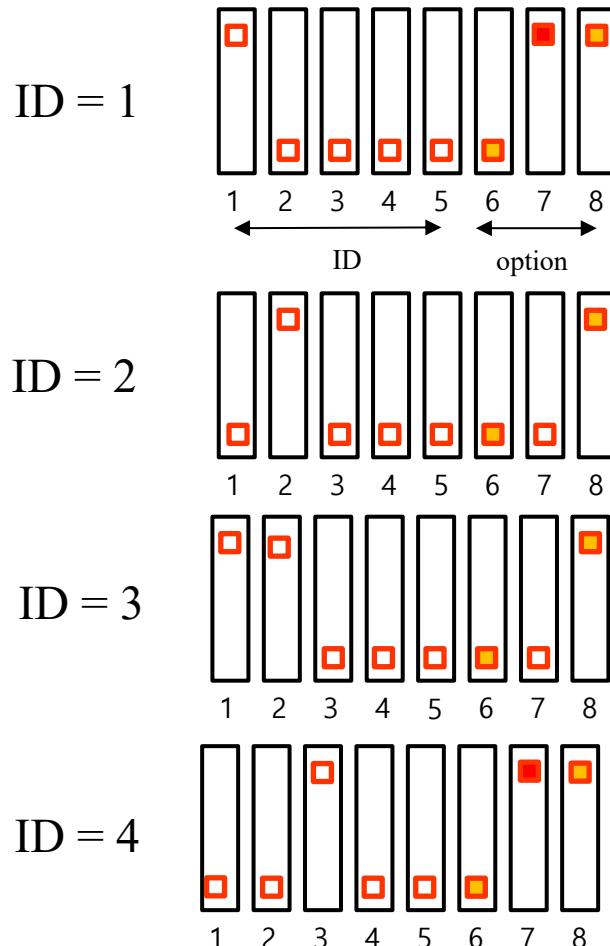


Table 3-35 CAN 1 connector X11/CAN 2 connector X12 – Pin assignment



Hardware setup

➤ DIP switch



3.3.7.2 CAN automatic Bit Rate Detection (Compact CAN)

Controller	Switch	OFF	ON
Compact CAN	6	 ON OFF Automatic bit rate detection deactivated	 ON OFF Automatic bit rate detection activated (factory setting)

Table 3-46 DIP switch SW1 – CAN automatic bit rate detection

3.3.7.3 CAN Bus Termination (Compact CAN)

Controller	Switch	OFF	나머지	ON	처음과 마지막
Compact CAN	7	 Without bus termination (factory setting)	ON OFF	 Bus termination with 120 Ω	ON OFF

Table 3-47 DIP switch SW1 – CAN bus termination

3.3.7.4 Digital Input Level

For details → chapter “3.4.7 Digital I/Os” on page 3-68.

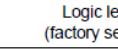
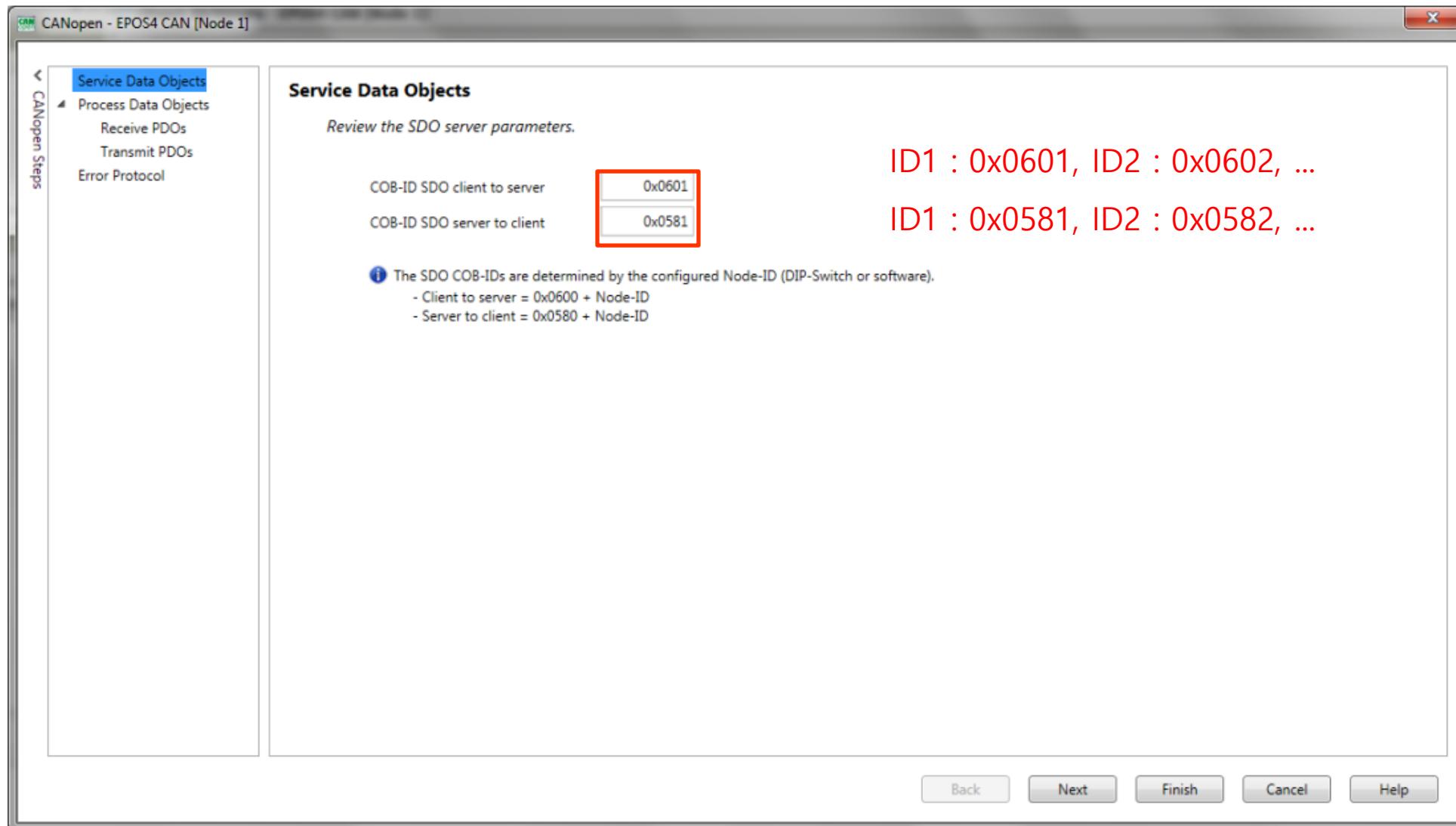
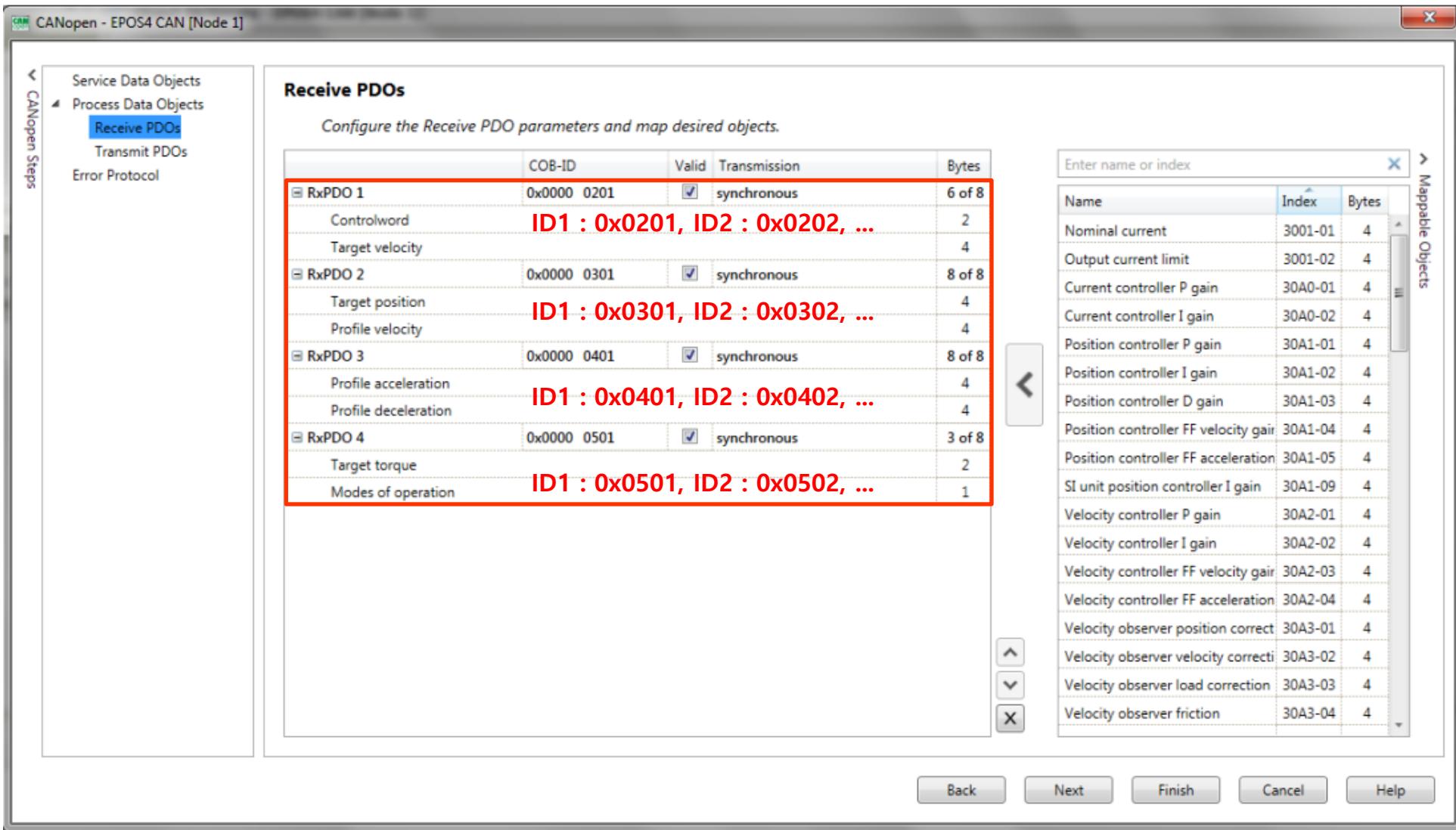
Controller	Switch	OFF	ON
Compact CAN	8	 1 8 R R R R R R R R U U U U U U U U ON OFF Logic level (factory setting)	 1 8 R R R R R R R R U U U U U U U U ON OFF PLC level
Compact EtherCAT	6	 1 6 R R R R R R U U U U U U ON OFF Logic level (factory setting)	 1 6 R R R R R R U U U U U U ON OFF PLC level

Table 3-48 DIP switch SW1 – Digital input level

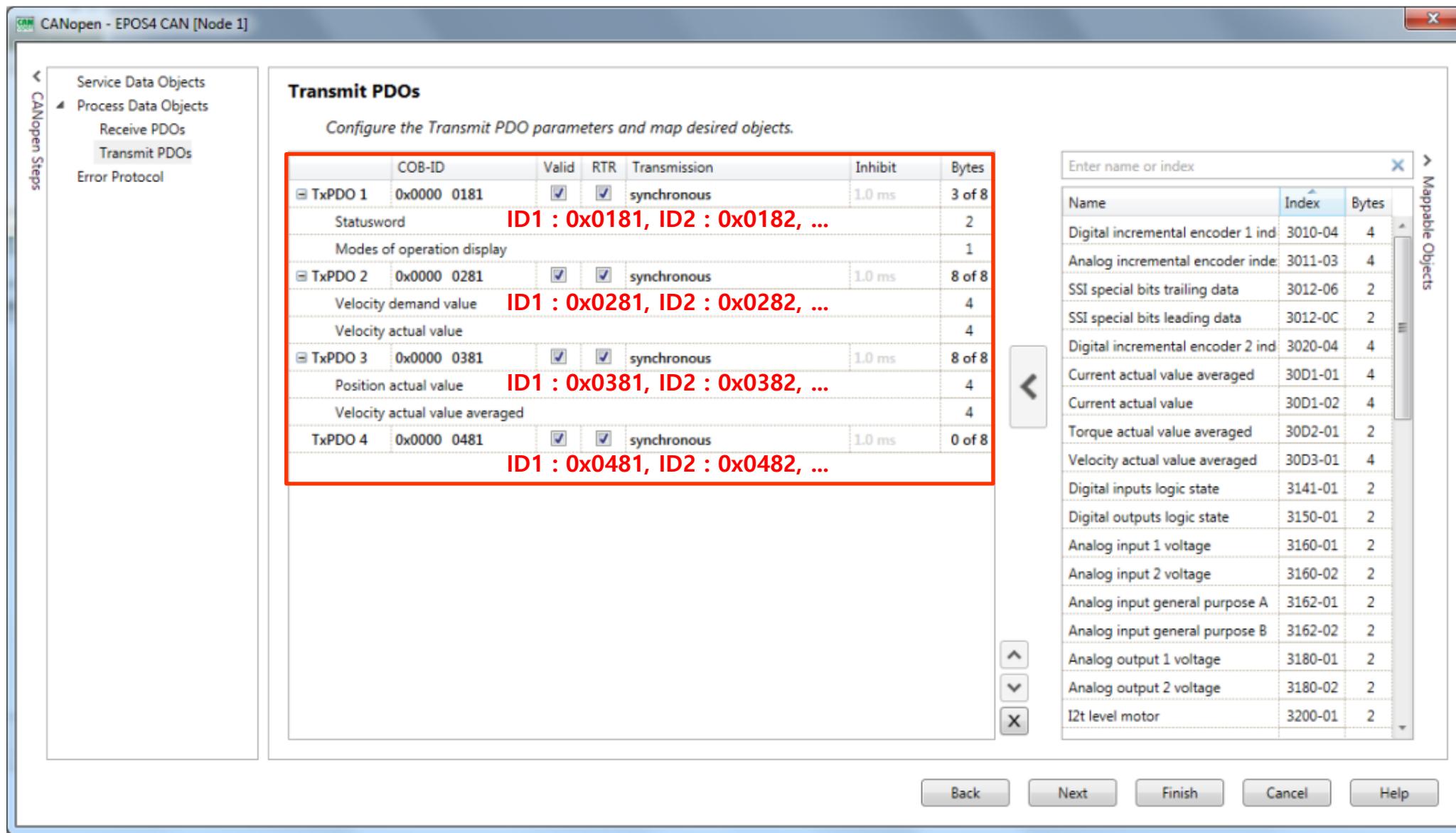
CANopen setting (EPOS studio)



CANopen setting (EPOS studio)



CANopen setting (EPOS studio)



Object Dictionary (EPOS studio)

- Interpolation time period value : **10ms**
- can bit rate : **1kbs** (1000000)
- Wizards -> Parameter Import/Export -> Export Parameters to File
- dcf 파일을 저장하여 코드에서 불러와야 함
- Linux → dcf 수정 필요

dcf 파일 수정

```
[2200]
SubNumber=3
SubNumber=2
ParameterName=Power supply
ObjectType=0x9

[2200sub0]
ParameterName=Highest sub-index supported
ObjectType=0x7
DataType=0x5
AccessType=ro
DefaultValue=2
PDOMapping=0
ObjFlags=1
ParameterValue=2
ParameterValue=1

[2200sub1]
ParameterName=Power supply voltage
ObjectType=0x7
DataType=0x6
AccessType=ro
PDOMapping=0
ObjFlags=3

[2200sub2]
ParameterName=Internal valid logic supply
ObjectType=0x7
DataType=0x1
AccessType=ro
PDOMapping=0
ObjFlags=1
```

```
[6098]
ParameterName=Homing method
ObjectType=0x7
DataType=0x2
AccessType=rww
DefaultValue=7
DefaultValue=0
PDOMapping=0
ObjFlags=0
ParameterValue=7
ParameterValue=0
```

- homming 사용 O : 7
- homming 사용 X : 0

패키지 구성

```
maxon_epos4_ros_canopen
├── CMakeLists.txt
└── config
    ├── canopen_bus_layer.yaml
    ├── controller_1dof_ppm.yaml
    ├── controller_2dof_csp.yaml
    ├── controller_2dof_pvm.yaml
    ├── epos4_50_15_can_ec90flat_gp52c_mile800_node1.dcf
    ├── epos4_50_15_can_ec90flat_gp52c_mile800_node1_hm.dcf
    ├── epos4_50_15_can_ec90flat_gp52c_mile800_node2.dcf
    ├── node_layer_1dof_ppm_hm.yaml
    ├── node_layer_1dof_ppm.yaml
    ├── node_layer_2dof_csp.yaml
    ├── node_layer_2dof_pvm.yaml
    └── ros_layer.yaml
└── launch
    ├── maxon_epos4_canopen_motor_1dof_ppm_hm.launch
    ├── maxon_epos4_canopen_motor_1dof_ppm.launch
    ├── maxon_epos4_canopen_motor_2dof_csp.launch
    └── maxon_epos4_canopen_motor_2dof_pvm.launch
└── LICENSE
└── package.xml
└── scripts
    └── python_example_1dof_ppm.py
└── src
    └── cpp_example_1dof_ppm.cpp
└── urdf
    ├── maxon_epos4_1dof_ppm.xacro
    ├── maxon_epos4_2dof_csp.xacro
    └── maxon_epos4_2dof_pvm.xacro
```

- 통신 및 ros
- 1dof ppm mode
- 2dof csp mode
- 실제 dcf 파일로 교체 필요

코드 수정 필요

➤ controller_2dof_csp.yaml

```

joint_names: [base_link1_joint, link1_link2_joint]

joint_state_controller:
  type: joint_state_controller/JointStateController
  publish_rate: 50

# Settings of the controller and associated drive modes
# drive mode : see http://wiki.ros.org/canopen_402

# position controllers
# example using Cyclic Synchronous Position (8) drive mode

joint_trajectory_controller:
  type: position_controllers/JointTrajectoryController
  joints:
    - base_link1_joint
    - link1_link2_joint
  required_drive_mode: 8
  constraints:
    stopped_velocity_tolerance: 20
    base_link1_joint: {trajectory: 50, goal: 50}
    link1_link2_joint: {trajectory: 50, goal: 50}

```

Here are the 3 modes defined in this documentation:

- Profile Position Mode: required_drive_mode: 1
- Profile Velocity Mode: required_drive_mode: 3
- Cyclic Synchronous Position Mode: required_drive_mode: 8

➤ node_layer_2dof_csp.yaml

```

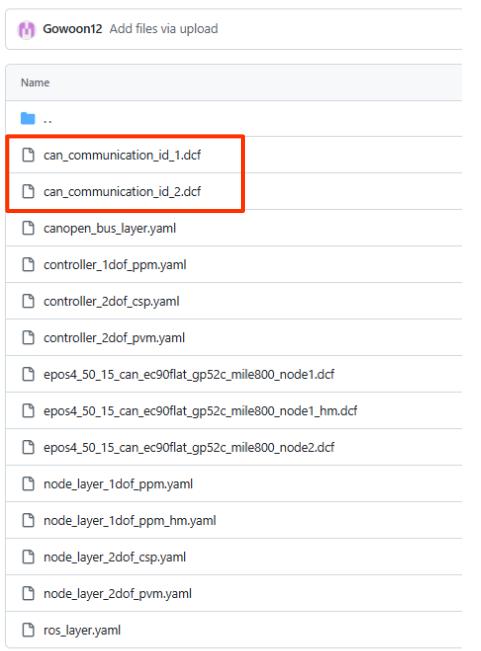
# struct syntax
nodes:
  node1:
    id: 1
    name: base_link1_joint
    eds_pkg: maxon epos4 ros canopen # optional package name for relative path
    eds_file: "config/can_communication_id_1.dcf" # path to EDS/DCF file
  node2:
    id: 2
    name: link1_link2_joint
    eds_pkg: maxon epos4 ros canopen # optional package name for relative path
    eds_file: "config/can_communication_id_2.dcf" # path to EDS/DCF file

defaults: # optional, all defaults can be overwritten per node
# eds_pkg: my_config_package # optional package name for relative path
# eds_file: "my_config.dcf" # path to EDS/DCF file
  dcf_overlay: # "ObjectID": "ParameterValue" (both as strings)
    "60C2sub1": "10" # Interpolation time period value, to be set as the sync/interval_ms value present in canopen_bus_layer.yaml file, 10ms is recommended
    "607Dsub1": "-2147483648" # disable software min limit
    "607Dsub2": "2147483647" # disable software max limit

# "6084": "10000" # Profile deceleration
# "6085": "10000" # Quick-stop deceleration
# "6098": "0" # No homing operation required
# "1016sub1": "0x7F0064" # heartbeat timeout of 100 ms for master at 127
# "1017": "100" # heartbeat producer

# canopen_chain_node settings ..
  motor_allocator: canopen::Motor402::Allocator # select allocator for motor layer
#  motor_layer: settings passed to motor layer (plugin-specific)
  switching_state: 2 # (Operation_Enable), state for mode switching. Drive mode of operation from canopen_402 wiki
  pos_to_device: "rint(rad2deg(pos)*400*16/360)" # rad -> inc, for a MILE 800 CPT encoder and a 26:1 gearbox, 3200 = 4*800 --> 4 * 100
  pos_from_device: "deg2rad(obj6064*360/400/16)" # actual position [inc] -> rad
  vel_to_device: "vel" # rad/s -> mdeg/s
  vel_from_device: "obj606C" # actual velocity [mdeg/s] -> rad/s
  eff_to_device: "rint(eff)" # just round to integer
  eff_from_device: "0" # unset

```



CAN open setup

- CAN-USB 꽂은 후, `/sbin/lsusb`로 연결 확인
 - `sudo ip link set can0 type can bitrate 1000000`
 - `sudo ip link set can0 up`
 - `candump can0`

```
gowoon@gowoon-15ZD990-VX50K:~$ lsusb
Bus 004 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 003 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 001 Device 030: ID 1d50:606f OpenMoko, Inc. canable gs_usb
Bus 001 Device 003: ID 04f2:b678 Chicony Electronics Co., Ltd LG Camera
Bus 001 Device 005: ID 8087:0aaa Intel Corp.
Bus 001 Device 002: ID 062a:4101 MosArt Semiconductor Corp. Wireless Keyboard/Mouse
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
gowoon@gowoon-15ZD990-VX50K:~$ sudo ip link set can0 type can bitrate 1000000
[sudo] password for gowoon:
gowoon@gowoon-15ZD990-VX50K:~$ sudo ip link set can0 up
gowoon@gowoon-15ZD990-VX50K:~$ candump can0
    can0  701  [1]  00
    can0  702  [1]  00
```



1dof ppm (예제)

- (모든 terminal : source ~/maxon_ws/devel/setup.bash)
- roscore
- roslaunch maxon_epos4_ros_canopen maxon_epos4_canopen_motor_1dof_ppm.launch
- rosservice call /maxon/driver/init
 - publish 테스트
- rostopic pub /maxon/canopen_motor/base_link1_joint_position_controller/command std_msgs/Float64 – 10
 - python 예제
- rosrun maxon_epos4_ros_canopen python_example_1dof_ppm.py
 - cpp 예제
- rosrun maxon_epos4_ros_canopen ccp_example_1dof_ppm.cpp
- rostopic list
- rostopic echo /maxon/joint/

```
$ cd  
~/catkin_ws/src/maxon_epos4_ros1/maxon_epos4_ros_canopen/scripts
```

```
$ chmod +x python_example_1dof_ppm.py
```

You can check that it worked by looking whether the following command displays "x" in the user permissions:

```
$ ls -l
```

```
cyril@cyril-tx2: ~/catkin_ws/src/maxon_epos4_ros1/maxon_epos4_ros_canopen/scripts  
cyril@cyril-tx2:~/catkin_ws/src/maxon_epos4_ros1/maxon_epos4_ros_canopen/scripts$ ls -l  
total 4  
-rwxr-xr-x 1 cyril cyril 1258 août 24 15:48 python_example_1dof_ppm.py
```

1dof ppm (예제)

The image shows two terminal windows side-by-side. The left terminal window displays the command line and its output:

```
18:28 ● /home/gowoon/maxon_ws/src/maxon_epos4_... EN
gowoon@gowoon-15ZD990-VX50K:~$ source ~/maxon_ws/devel/setup.bash
gowoon@gowoon-15ZD990-VX50K:~$ rosrun maxon_epos4_ros_canopen maxon_epos4_canopen_motor_1dof_ppm.launch
... logging to /home/gowoon/.ros/log/e340703c-cdce-11f0-97ba-579f6ff8d57f/roslaunch-gowoon-15ZD990-VX50K-83747.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://gowoon-15ZD990-VX50K:36333/
SUMMARY
=====
CLEAR PARAMETERS
* /maxon/canopen_motor/
PARAMETERS
* /maxon/canopen_motor/base_link1_joint_position_controller/joint: base_link1_joint
* /maxon/canopen_motor/base_link1_joint_position_controller/required_drive_mode: 1
* /maxon/canopen_motor/base_link1_joint_position_controller/type: position_controller...
* /maxon/canopen_motor/bus/device: can0
* /maxon/canopen_motor/bus/master_allocator: canopen::SimpleMa...
* /maxon/canopen_motor/defaults/eff_from_device: 0
* /maxon/canopen_motor/defaults/motor_allocator: canopen::Motor402...
* /maxon/canopen_motor/defaults/pos_from_device: obj0064
* /maxon/canopen_motor/defaults/pos_to_device: pos
* /maxon/canopen_motor/defaults/swit...
* /maxon/canopen_motor/defaults/vel_from_device: obj006C
* /maxon/canopen_motor/defaults/vel_to_device: vel
* /maxon/canopen_motor/heartbeat/msg: 77#05
* /maxon/canopen_motor/heartbeat/rate: 20
* /maxon/canopen_motor/joint_group_position_controller/joints: ['base_link1_joint']
* /maxon/canopen_motor/joint_group_position_controller/required_drive_mode: 1
* /maxon/canopen_motor/joint_group_position_controller/type: position_controller...
* /maxon/canopen_motor/joint_names: ['base_link1_joint']
* /maxon/canopen_motor/joint_state_controller/publish_rate: 50
* /maxon/canopen_motor/joint_state_controller/type: joint_state_contr...
* /maxon/canopen_motor/nodes/node1/eds_file: config/can_commun...
* /maxon/canopen_motor/nodes/node1/eds_pkgs: maxon_epos4_ros_c...
* /maxon/canopen_motor/nodes/node1/id: 1
* /maxon/canopen_motor/nodes/node1/name: base_link1_joint
* /maxon/canopen_motor/sync/interval_ms: 10
* /maxon/canopen_motor/sync/overflow: 0
* /maxon/robot_description: <?xml version="1.....
/rostdistro: noetic
*/rosversion: 1.17.4

NODES
/maxon/
canopen_motor (canopen_motor_node/canopen_motor_node)
controller_spawner (controller_manager/controller_manager)

ROS_MASTER_URI=http://localhost:11311
```

A red box highlights the last few lines of the log output:

```
process[maxon/canopen_motor-1]: started with pid [83772]
process[maxon/controller_spawner-2]: started with pid [83773]
[INFO] [1764494909.688459134]: Using fixed control period: 0.010000000
[INFO] [1764494934.969875385]: Initializing...
[INFO] [1764494934.970096165]: Current state: 1 device error: system:0 internal_error: 0 (OK)
[INFO] [1764494934.970233940]: Current state: 2 device error: system:0 internal_error: 0 (OK)
[ERROR] [1764494937.223159489]: EMCY received: 81#0000000000000000
[INFO] [1764494937.223441034]: Initializing successful
Loaded '/maxon/canopen_motor/joint_state_controller'
Loaded '/maxon/canopen_motor/base_link1_joint_position_controller'
Started ['/maxon/canopen_motor/joint_state_controller'] successfully
Started ['/maxon/canopen_motor/base_link1_joint_position_controller'] successfully
[maxon/controller_spawner-2] process has finished cleanly
Log file: /home/gowoon/.ros/log/e340703c-cdce-11f0-97ba-579f6ff8d57f/maxon-controller_spawner-2*.log
```

The right terminal window shows the command:

```
18:28 ● gowoon@gowoon-15ZD990-VX50K:~ EN
gowoon@gowoon-15ZD990-VX50K:~$ source ~/maxon_ws/devel/setup.bash
gowoon@gowoon-15ZD990-VX50K:~$ rosrun maxon_epos4_ros_canopen maxon_epos4_canopen_motor_1dof_ppm.launch
success: True
message: ''
gowoon@gowoon-15ZD990-VX50K:~$
```

▶ 새로운 창에서 초기화 필요

- **rosservice call /maxon/driver/init** → 이것만 하면 됨
- **rosservice call /maxon/driver/halt**
- **rosservice call /maxon/driver/recover**
- **rosservice call /maxon/driver/shutdown**

The terminal window shows the log output for the driver initialization process:

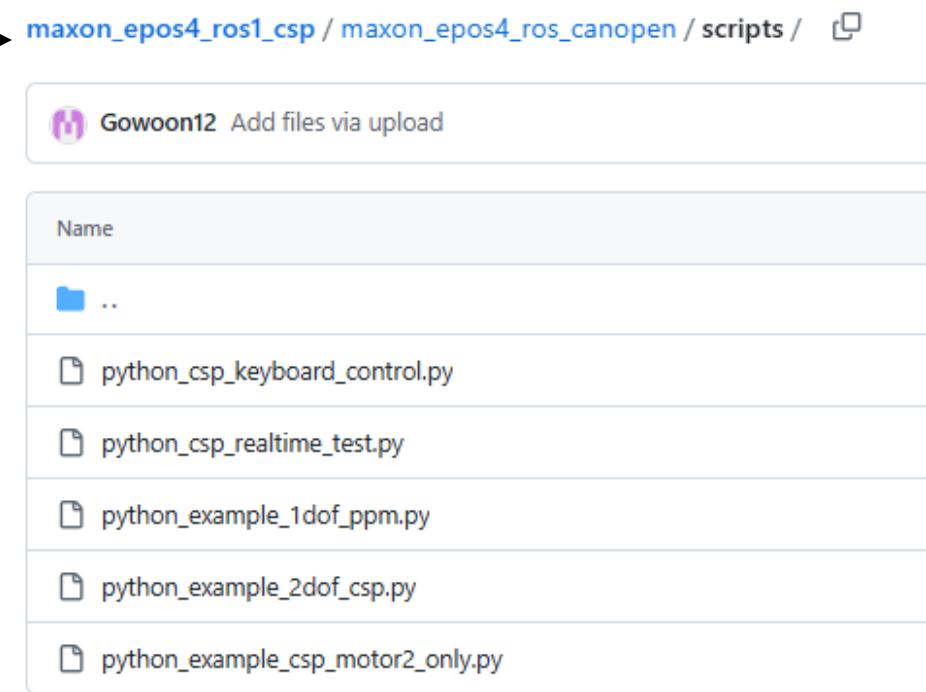
```
NODES
/maxon/
canopen_motor (canopen_motor_node/canopen_motor_node)
controller_spawner (controller_manager/controller_manager)

ROS_MASTER_URI=http://localhost:11311

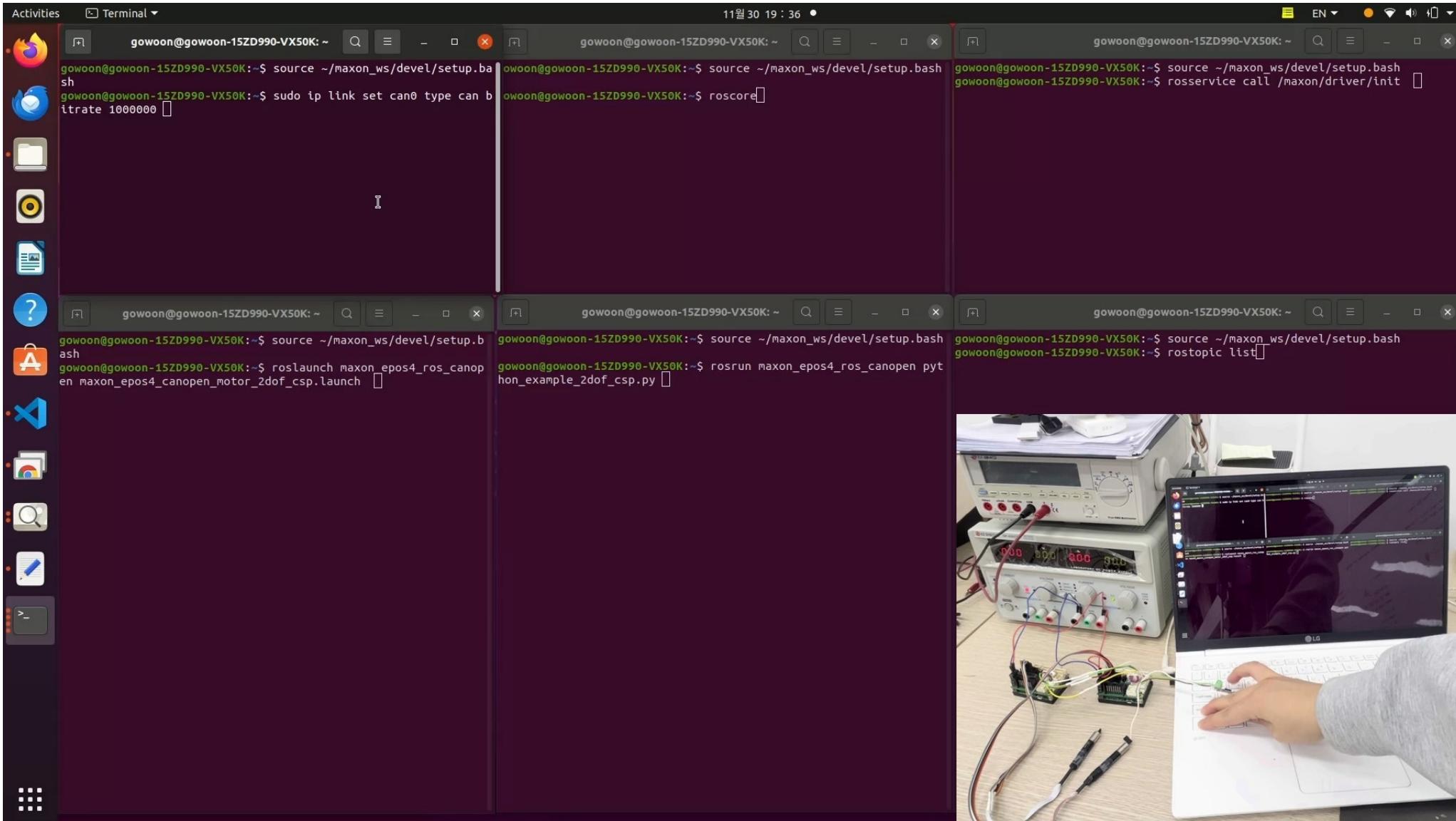
process[maxon/canopen_motor-1]: started with pid [83772]
process[maxon/controller_spawner-2]: started with pid [83773]
[INFO] [1764494909.688459134]: Using fixed control period: 0.010000000
[INFO] [1764494934.969875385]: Initializing...
[INFO] [1764494934.970096165]: Current state: 1 device error: system:0 internal_error: 0 (OK)
[INFO] [1764494934.970233940]: Current state: 2 device error: system:0 internal_error: 0 (OK)
[ERROR] [1764494937.223159489]: EMCY received: 81#0000000000000000
[INFO] [1764494937.223441034]: Initializing successful
Loaded '/maxon/canopen_motor/joint_state_controller'
Loaded '/maxon/canopen_motor/base_link1_joint_position_controller'
Started ['/maxon/canopen_motor/joint_state_controller'] successfully
Started ['/maxon/canopen_motor/base_link1_joint_position_controller'] successfully
[maxon/controller_spawner-2] process has finished cleanly
Log file: /home/gowoon/.ros/log/e340703c-cdce-11f0-97ba-579f6ff8d57f/maxon-controller_spawner-2*.log
```

2dof csp mode

- (모든 terminal : source ~/maxon_ws/devel/setup.bash)
- roscore
- roslaunch maxon_epos4_ros_canopen maxon_epos4_canopen_motor_2dof_csp.launch
- rosservice call /maxon/driver/init
- **python 예제 (새로 작성함)** → [maxon_epos4_ros1_csp / maxon_epos4_ros_canopen / scripts /](#)
- rosrun maxon_epos4_ros_canopen python_example_2dof_csp.py
- rostopic list
- rostopic echo /maxon/joint/



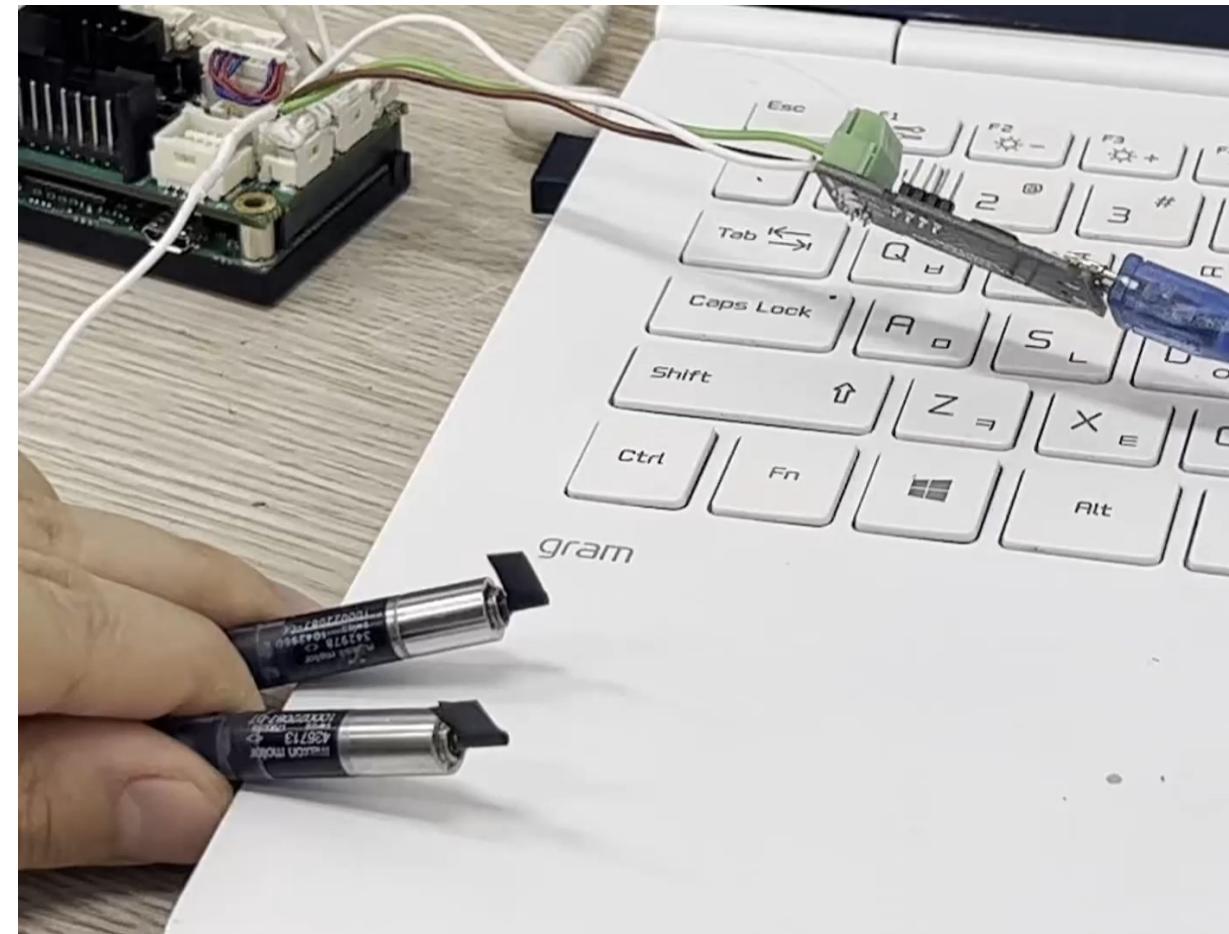
2dof csp mode



2dof csp mode

➤ 약 100hz

```
gwown@gwown-15ZD990-VX50K: ~
[INFO] [1764499188.483207]: Δt = 0.009882 sec (101.2 Hz)
[INFO] [1764499188.493193]: Δt = 0.009985 sec (100.2 Hz)
[INFO] [1764499188.503330]: Δt = 0.010072 sec (99.3 Hz)
[INFO] [1764499188.513115]: Δt = 0.009896 sec (101.1 Hz)
[INFO] [1764499188.523200]: Δt = 0.010001 sec (100.0 Hz)
[INFO] [1764499188.533436]: Δt = 0.010184 sec (98.2 Hz)
[INFO] [1764499188.543104]: Δt = 0.009776 sec (102.3 Hz)
[INFO] [1764499188.553404]: Δt = 0.010200 sec (98.0 Hz)
[INFO] [1764499188.563376]: Δt = 0.010011 sec (99.9 Hz)
[INFO] [1764499188.573198]: Δt = 0.009875 sec (101.3 Hz)
[INFO] [1764499188.583206]: Δt = 0.009995 sec (100.0 Hz)
[INFO] [1764499188.593165]: Δt = 0.009976 sec (100.2 Hz)
[INFO] [1764499188.603390]: Δt = 0.010202 sec (98.0 Hz)
[INFO] [1764499188.613207]: Δt = 0.009798 sec (102.1 Hz)
[INFO] [1764499188.623198]: Δt = 0.010015 sec (99.8 Hz)
[INFO] [1764499188.633383]: Δt = 0.010126 sec (98.8 Hz)
[INFO] [1764499188.643177]: Δt = 0.009864 sec (101.4 Hz)
[INFO] [1764499188.653137]: Δt = 0.009978 sec (100.2 Hz)
[INFO] [1764499188.663300]: Δt = 0.010082 sec (99.2 Hz)
[INFO] [1764499188.673177]: Δt = 0.009949 sec (100.5 Hz)
[INFO] [1764499188.683221]: Δt = 0.010027 sec (99.7 Hz)
[INFO] [1764499188.693213]: Δt = 0.010013 sec (99.9 Hz)
[INFO] [1764499188.703401]: Δt = 0.010170 sec (98.3 Hz)
[INFO] [1764499188.713174]: Δt = 0.009764 sec (102.4 Hz)
[INFO] [1764499188.723251]: Δt = 0.010019 sec (99.8 Hz)
[INFO] [1764499188.733214]: Δt = 0.010014 sec (99.9 Hz)
[INFO] [1764499188.743281]: Δt = 0.010022 sec (99.8 Hz)
[INFO] [1764499188.753122]: Δt = 0.009916 sec (100.8 Hz)
[INFO] [1764499188.763321]: Δt = 0.010126 sec (98.8 Hz)
[INFO] [1764499188.773350]: Δt = 0.010021 sec (99.8 Hz)
[INFO] [1764499188.783104]: Δt = 0.009841 sec (101.6 Hz)
[INFO] [1764499188.792931]: Δt = 0.009928 sec (100.7 Hz)
[INFO] [1764499188.803219]: Δt = 0.010168 sec (98.3 Hz)
[INFO] [1764499188.813042]: Δt = 0.009917 sec (100.8 Hz)
```



Thank you