ROS 2 Interface Guide for RFT Series (EtherCAT Model)

This guide provides instructions for interfacing the **RFT sensor**, which uses **EtherCAT communication**, within a **ROS 2** environment.

Steps 2 through 4 require a total of **three terminals** .
Each step must be executed in a **separate terminal**.

System Requirements

- ROS 2 must be installed
- Verified in a virtual environment using **Ubuntu 20.04** with **ROS 2 Foxy**

---

1. ROS 2 Workspace Setup and Package Build

1-1) Create your workspace

- Create a directory that will contain your ROS 2 workspace.

  ```
  mkdir -p ~/{your_workspace}
  ```

  ```
  cd ~/{your_workspace}
  ```

1-2) Create the 'src' folder

- The 'src' folder is not generated automatically. You need to create it manually to store ROS 2 packages.

  ```
  mkdir src
  ```

1-3) Unzip your ROS 2 package into the 'src' folder

- Copy 'rft_ethercat_ros2_interface.zip' into 'src' and extract it

  ```
  cp ~/Downloads/rft_ethercat_ros2_interface.zip ~/{your_workspace}/src/
  ```

  ```
  cd ~/{your_workspace}/src
  ```

```
unzip rft_ethercat_ros2_interface.zip
```

1-4) Build the Workspace

- Build all packages in the 'src' folder using 'colcon'.

```
cd ~/{your_workspace}
```

```
colcon build
```

- After the build completes, verify that the following directories are created:
  - build/
  - install/
  - log

---

2. EtherCAT Environment Setup and Interface Execution

- This section describes the complete process required to install dependencies, configure permissions, and run the EtherCAT interface

2-1) Install EtherCAT Python Dependency (pysoem)

```
sudo pip3 install pysoem
```

- Root privileges are required because EtherCAT uses **raw network sockets**.
- Setting network capabilities (**cap_net_raw**) also requires **sudo**.

2-2) Configure Python Permissions for EtherCAT

- Check the Python version in use and grant raw network socket permission.

```
cd /usr/bin
```

```
ls python3.8
```

```
sudo setcap cap_net_raw+ep /usr/bin/python3.8
```

- Applying setcap restricts environment variables.
- This may prevent ROS 2 shared libraries from loading correctly.
- **Therefore, the interface must be executed as root.**
- A permanent non-root solution is under investigation.

2-3) Run the EtherCAT Interface Node

- Switch to the root user:

  sudo su

- Navigate to the workspace and source the ROS 2 environment:
  cd /home/{username}/{your_workspace}

  source /opt/ros/<your_ros2_distro>/setup.bash

  source ./install/setup.bash

- Move to the script directory and run the interface:
  cd ~/{your_workspace}/src/rft_ethercat_ros2_interface/script

  python3 interface.py

- Find the network adapter name using:
  ifconfig
- Identify the active network adapter used for EtherCAT communication (for example: **enp0s3**).
- In the script directory, open the config.txt file and enter the identified adapter name as shown below
- Replace enp0s3 with the actual adapter name detected on your system.

```
# [EtherCAT Configuration]
# Network Interface Name (Check with 'ifconfig' command)
Interface: enp0s3

# Output Rate: 1000, 500, 333, 250, 200, 100 (Hz)
OutputRate_Hz: 500
```

3. Check and Display FT Sensor Data

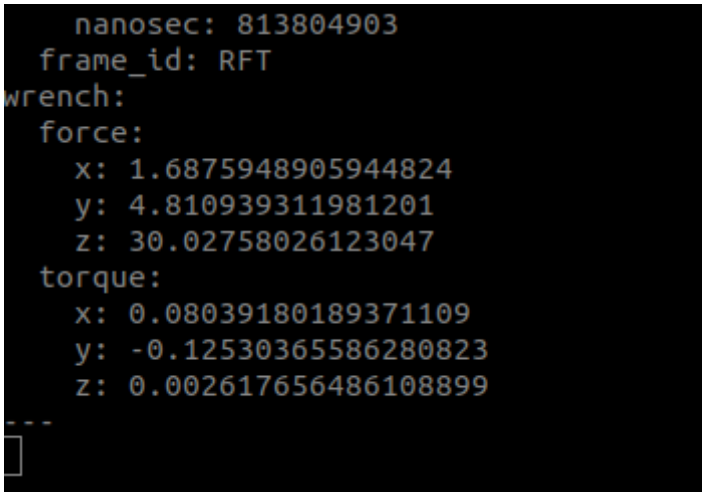
3-1) Open a New Terminal (Root Required)

```
sudo su
cd ~/{your_workspace}
source /opt/ros/<your_ros2_distro>/setup.bash
source ./install/setup.bash
```

3-2) Display FT Data

- Display the FT (Force/Torque) data published by the sensor:

  ```
  ros2 topic echo /RFT/FT_data
  ```



- This command continuously outputs FT sensor data to the terminal.
- Only FT data messages are displayed.

4. Sensor Control via ROS 2 Services

- Sensor control commands are sent using **ROS 2 services**.
- A service client is required to send commands to the sensor.

4-1) Launch the Service Client (Root Required)

- Open a new terminal and source the environment:

  sudo su

  cd ~/{your_workspace}

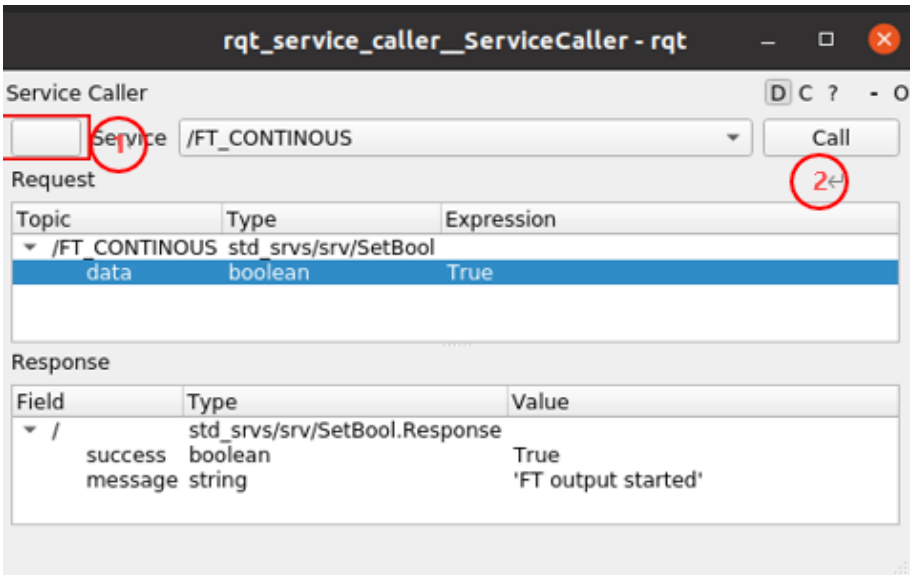
  source /opt/ros/setup.bash

  source ./install/setup.bash

- Launch the service client using rqt_service_caller:

  ros2 run rqt_service_caller rqt_service_caller

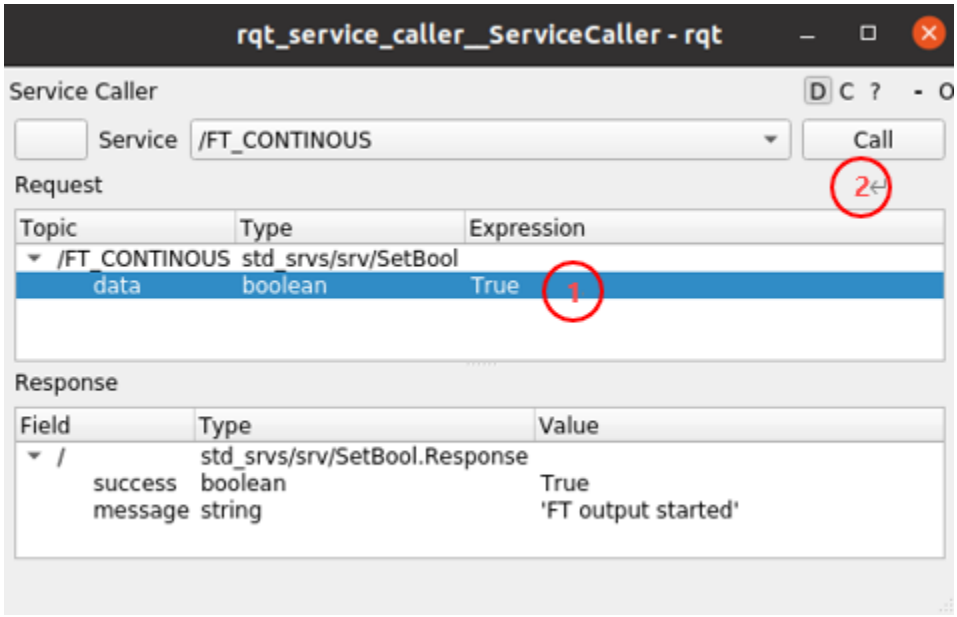
- Once the window opens, click the Update button (1), then select the desired service from the list (2).



- Before using any service other than FT output, /FT_CONTINOUS must be set to '**False**'.
- Other functions will not operate correctly while FT data output is active.
- Output messages from read/set services appear in the terminal used in **Section 2-3**.
- Start/Stop FT output messages appear in the terminal used in **Section 3-2**.
- Parameters returned by read/set services follow the tables in Sections **4-11**.

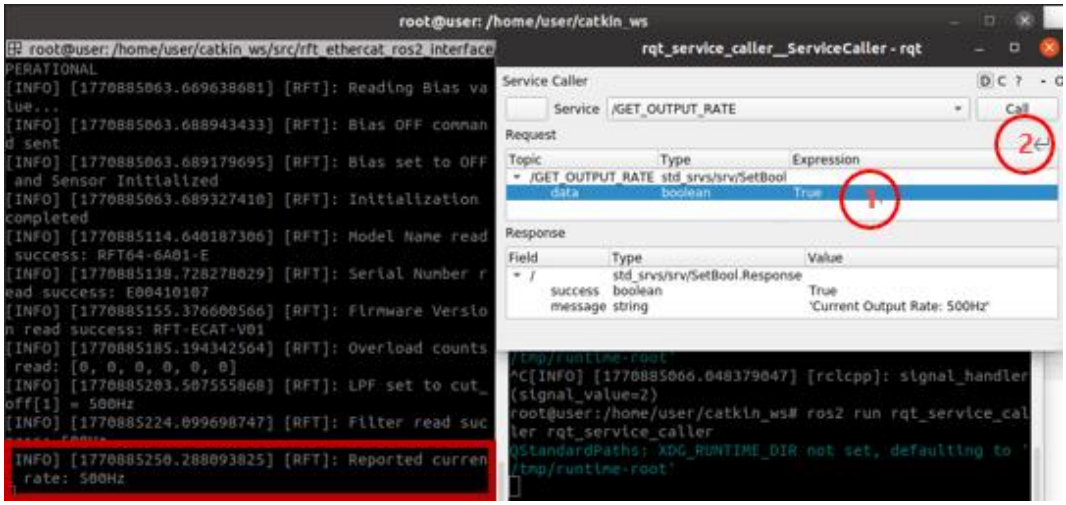
4-2) Stop / Start Data Output

- To stop FT data output:
  - Set the **expression** parameter of / FT_CONTINOUS to '**False**' (1)
  - Click **Call** (2)
- To start FT data output:
  - Set the parameter to '**True**' (1)
  - Click **Call** (2)



4-3) Read Data Output Rate

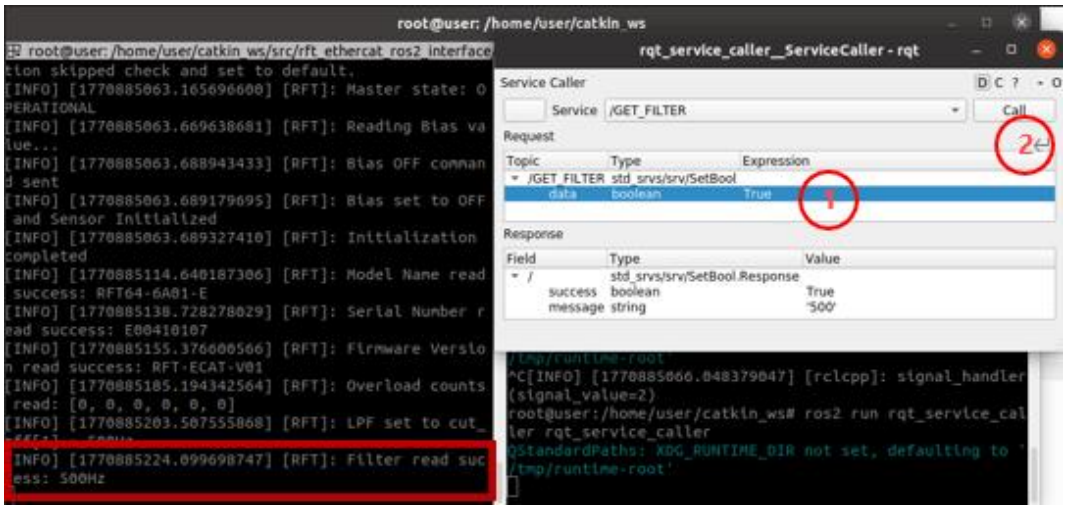
- Select the /GET_OUTPUT_RATE service.
- Set the **expression** parameter to '**True**'(1).
- Click **Call** (2).



- The current data output rate parameter is displayed.

4-4) Read Filter Setting

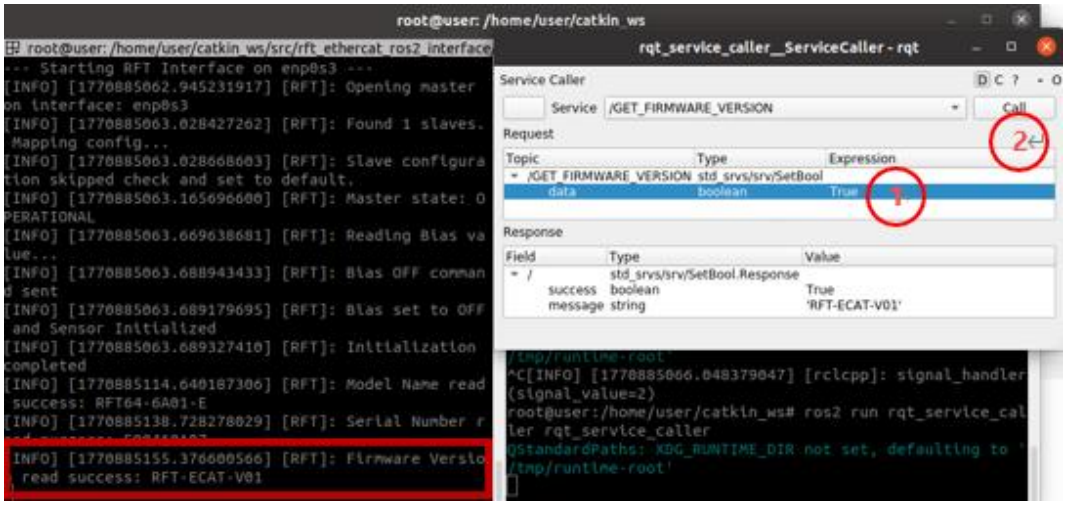
- Select the /GET_FILTER service.
- Set the **expression** parameter to '**True**'(1).
- Click **Call** (2).



- The current filter setting parameter is displayed.

4-5) Read Firmware Version

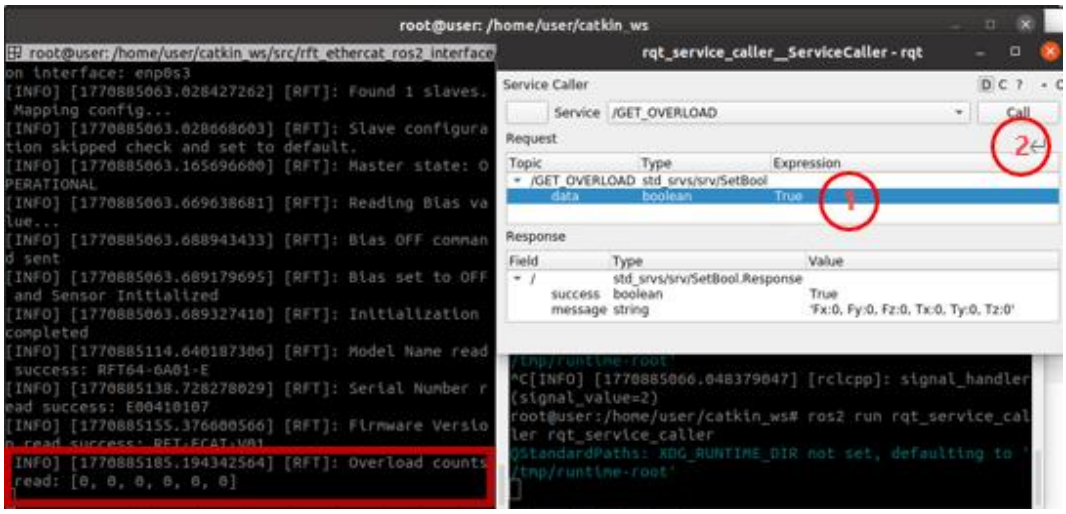
- Select the /GET_FIRMWARE_VERSION service.
- Set the **expression** parameter to '**True**' (1).
- Click **Call** (2).



- The firmware version is displayed.

4-6) Read Count of Overload Occurrence

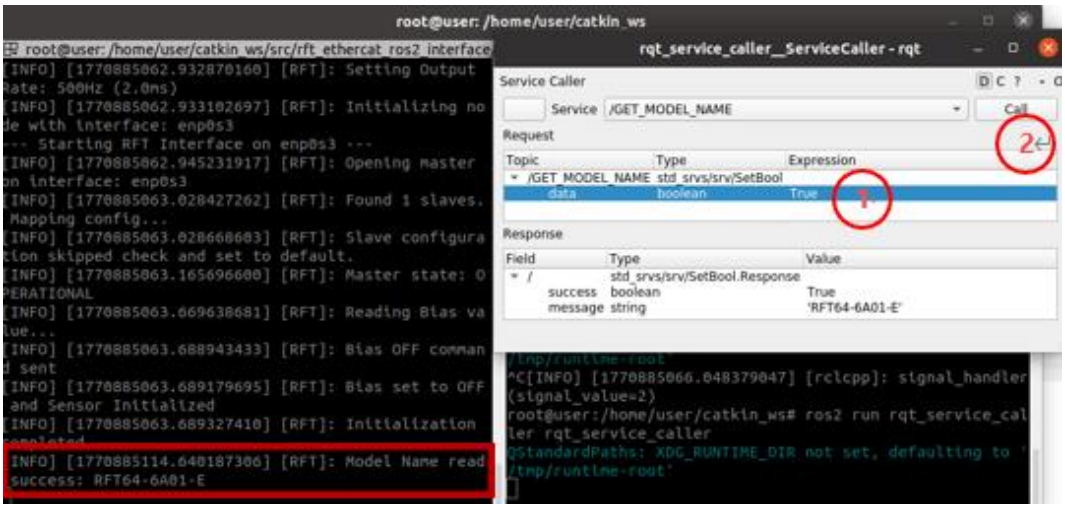
- Select the /GET_OVERLOAD service.
- Set the **expression** parameter to '**True**' (1).
- Click **Call** (2).



- Overload count values are displayed in the following order:
  - Fx, Fy, Fz, Tx, Ty, Tz

4-7) Read Model Name

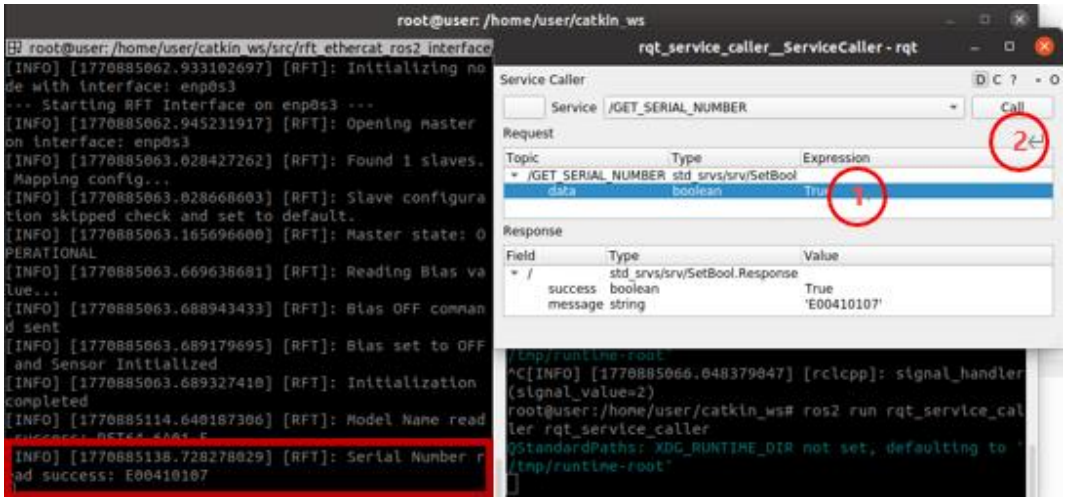
- Select the /GET_MODEL_NAME service.
- Set the **expression** parameter to '**True**' (1).
- Click **Call** (2).



- The sensor model name is displayed.

4-8) Read Serial Number

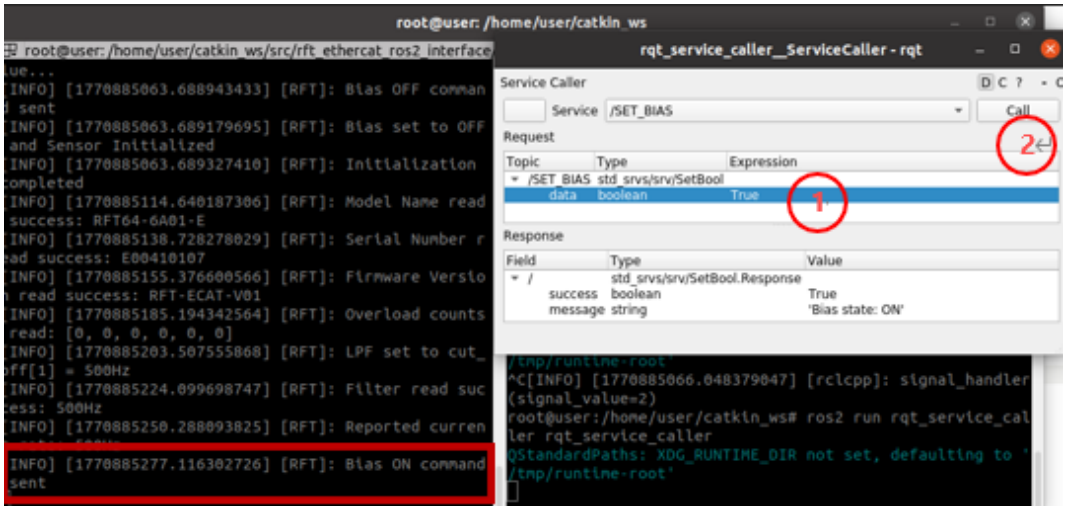
- Select the /GET_SERIAL_NUMBER service.
- Set the **expression** parameter to '**True**' (1).
- Click **Call** (2).



- The sensor serial number is displayed.

4-9) Set Bias

- To disable bias:
  - Set /SET_BIAS to '**False**' (1).
  - Click **Call** (2).
- To enable bias:
  - Set the parameter to '**True**' (1).
  - Click **Call** (2).



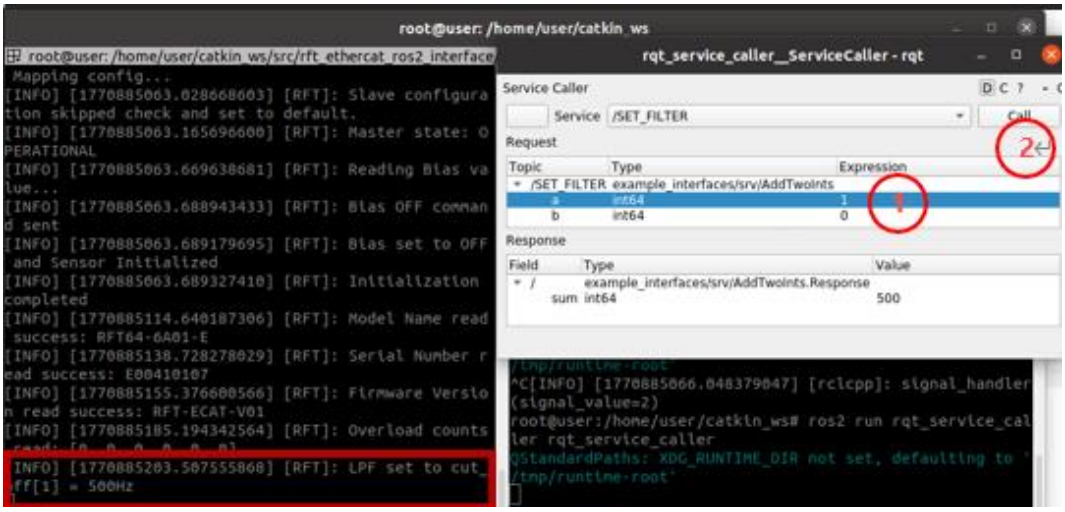
4-10) Set Data Output Rate

- The data output rate can be configured in the config.txt file located in the script directory.
- Change only the numeric value of the output rate to one of the values listed on the 5th line.
- After saving 'config.txt', **restart the Python interface executed in Step 2-3** for the change to take effect.

```
1 # [EtherCAT Configuration]
2 # Network Interface Name (Check with 'ifconfig' command)
3 Interface: enp0s3
4
5 # Output Rate: 1000, 500, 333, 250, 200, 100 (Hz)
6 OutputRate_Hz: 500
```

4-11) Set Filter

- Select the /SET_FILTER service.
- Enter the filter parameter value (not the filter type) in **request.a** (1).
- Click **Call** (2).



- Refer to the table below for filter parameters and cutoff frequencies:

Filter Type	Filter Parameter	Cutoff Frequency [Hz]
0(0x00)	0(0x00)	No filter
1(0x01)	0(0x00)	No filter
1(0x01)	1(0x01)	500
1(0x01)	2(0x02)	300
1(0x01)	3(0x03)	200
1(0x01)	4(0x04)	150
1(0x01)	5(0x05)	100
1(0x01)	6(0x06)	50
1(0x01)	7(0x07)	40
1(0x01)	8(0x08)	30
1(0x01)	9(0x09)	20
1(0x01)	10(0x0A)	10
1(0x01)	11(0x0B)	5
1(0x01)	12(0x0C)	3
1(0x01)	13(0x0D)	2
1(0x01)	14(0x0E)	1