

# Using MIPI Camera

---

## Using MIPI Camera

[1. Environment Preparation](#)

[2. Running Method](#)

[3. Expected Results](#)

[Precautions](#)

To achieve environmental awareness capabilities, robotic products typically incorporate sensors such as cameras and ToF sensors. To reduce the cost of sensor adaptation and usage for users, TogetheROS.Bot encapsulates various commonly used sensors and abstracts them into the `hobot_sensor` module, supporting ROS standard image messages. When the configured sensor parameters do not match the connected camera, the program will automatically adapt to the correct sensor type.

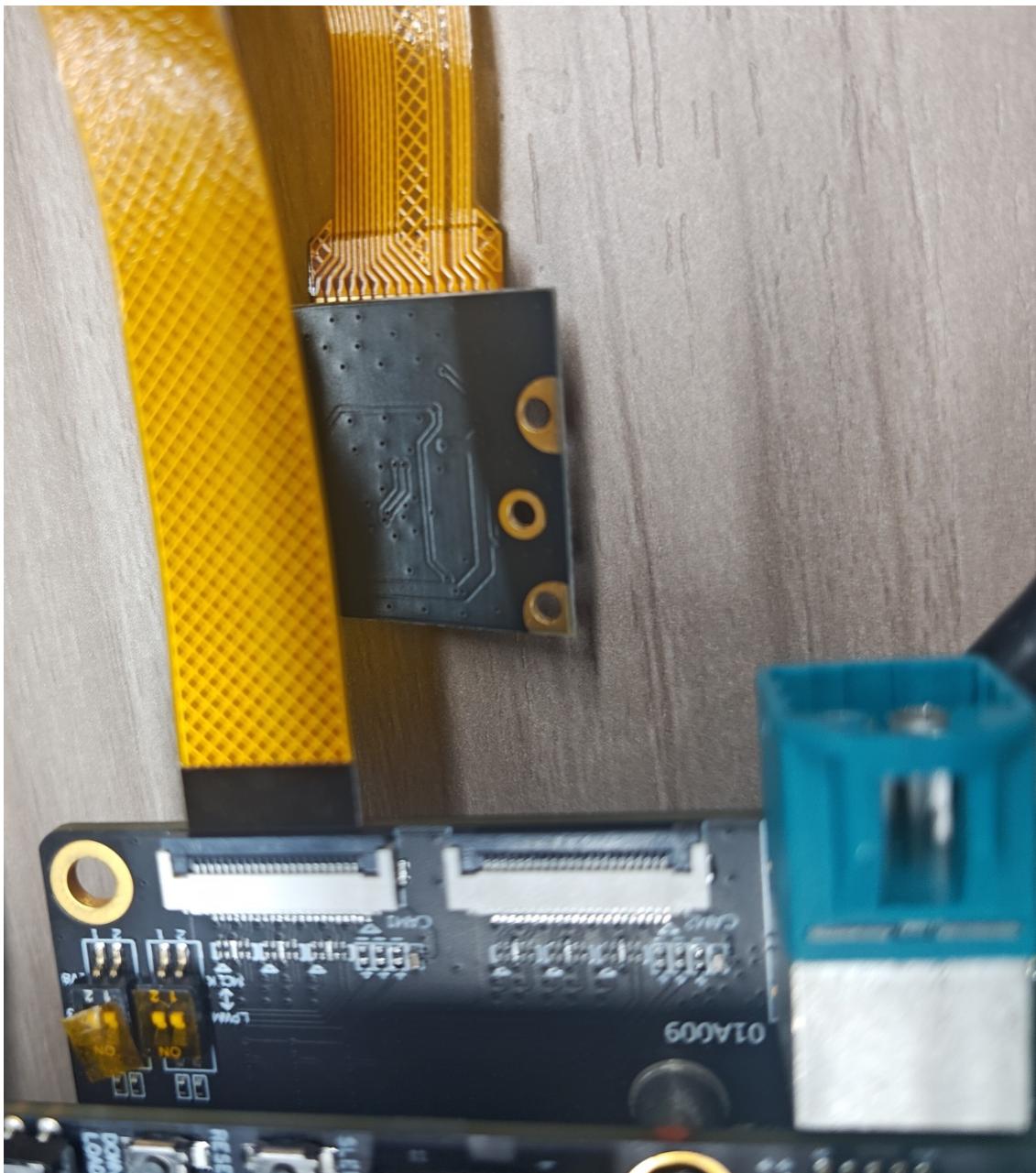
**Note:**

Running this section requires first learning the TogetheROS.Bot visual application chapter and installing the TROS environment.

## 1. Environment Preparation

---

- Connect the power supply.
- Connect the MIPI camera module to the development board's MIPI CSI interface. Note that the black side should be facing upwards.



- Connect the development board to the monitor via an HDMI cable.
- Log in to the RDK S100(P) via MobaXterm or other SSH tools.

## 2. Running Method

---

- Log in to the RDK via SSH and determine the camera model; here, `IMX219` is used as an example.
- Start the `hobot_sensor` node using the following command:

```
# Configure the tros.b environment
source /opt/tros/humble/setup.bash
```

```
# Launched using the launch method
ros2 launch mipy_cam mipy_cam.launch.py mipy_video_device:=IMX219
```

- If the program outputs the following information, it means the node has started successfully.

```
[INFO] [launch]: All log files can be found below /root/.ros/log/2022-06-11-15-16-13-641715-ubuntu-8852  
[INFO] [launch]: Default logging verbosity is set to INFO  
[INFO] [mipi_cam-1]: process started with pid [8854]  
...
```

### 3. Expected Results

- To view the IMX219 camera image on the web, since raw data is being published, JPEG image encoding is required. Two separate terminals are needed: one to subscribe to the MIPI data and encode it into JPEG, and another to publish it via a web service.

```
# Configure the tros.b environment  
source /opt/tros/humble/setup.bash
```

```
# Start encoding  
ros2 launch hobot_codec hobot_codec_encode.launch.py  
  
# Open another terminal  
# Configure the tros.b environment  
source /opt/tros/humble/setup.bash  
  
# Start the WebSocket connection.  
ros2 launch websocket websocket.launch.py websocket_image_topic:=/image_jpeg  
websocket_only_show_image:=true
```

- Open a browser (Chrome/Firefox/Edge) on your PC and enter `http://IP:8000` (where IP is the RDK IP address). Click on the web interface display in the upper left corner to see the real-time video output from the F37 camera.



- Query the camera intrinsic parameters on your PC (the specific data will depend on the camera calibration file). The command and results are as follows:

```
root@ubuntu:~# source /opt/ros/humble/setup.bash
```

```
root@ubuntu:~# ros2 topic echo /camera_info  
header:
```

```
stamp:  
  sec: 1662013622  
  nanosec: 672922214  
frame_id: default_cam  
height: 1080  
width: 1920  
distortion_model: plumb_bob  
d:  
  - 0.169978  
  - -0.697303  
  - -0.002944  
  - -0.004961  
  - 0.0  
k:  
  - 1726.597634  
  - 0.0  
  - 904.979671  
  - 0.0  
  - 1737.359551  
  - 529.123375  
  - 0.0  
  - 0.0  
  - 1.0  
r:  
  - 1.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 1.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 1.0  
p:  
  - 1685.497559  
  - 0.0  
  - 881.6396  
  - 0.0  
  - 0.0  
  - 1756.460205  
  - 526.781147  
  - 0.0  
  - 0.0  
  - 0.0  
  - 1.0  
  - 0.0  
binning_x: 0  
binning_y: 0  
roi:  
x_offset: 0  
y_offset: 0  
height: 0  
width: 0  
do_rectify: false
```

## Precautions

- Camera Connection and Disconnection Precautions
- It is strictly forbidden to connect or disconnect the camera while the development board is powered on, as this can easily damage the camera module.**
- If you encounter abnormal startup of the hobot\_sensor node, you can troubleshoot the problem using the following steps:
    - Check the hardware connection.
    - Check if the tros.b environment is set up.
    - Check if the parameters are correct; please refer to [README.md](#) for details.
    - Note that the tros environment must be set up in every interface.
    - It is recommended to run the examples in root mode.