

USB Camera, PTZ Users (Must Read)

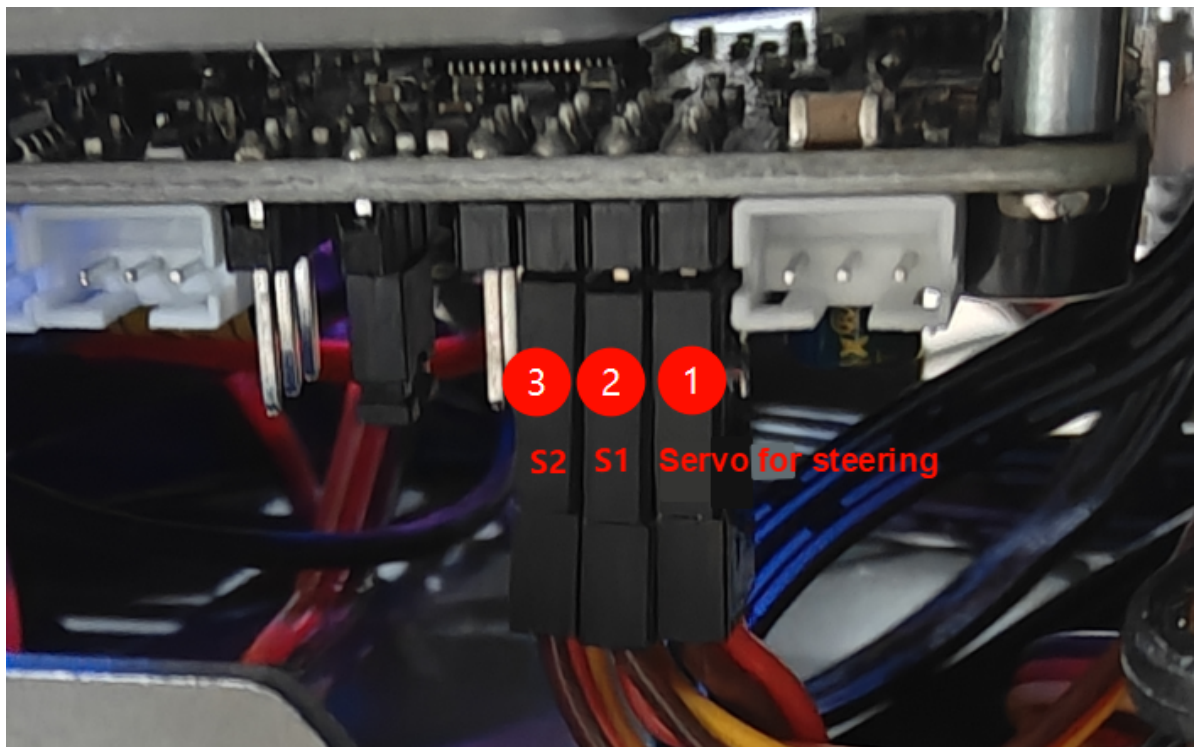
USB Camera, PTZ Users (Must Read)

1. Installation
2. Initial Value Settings
 - 2.1 Servo Initial Values in the Auto-Startup Program
 - 2.2 Initial Servo Values in Vision Cases

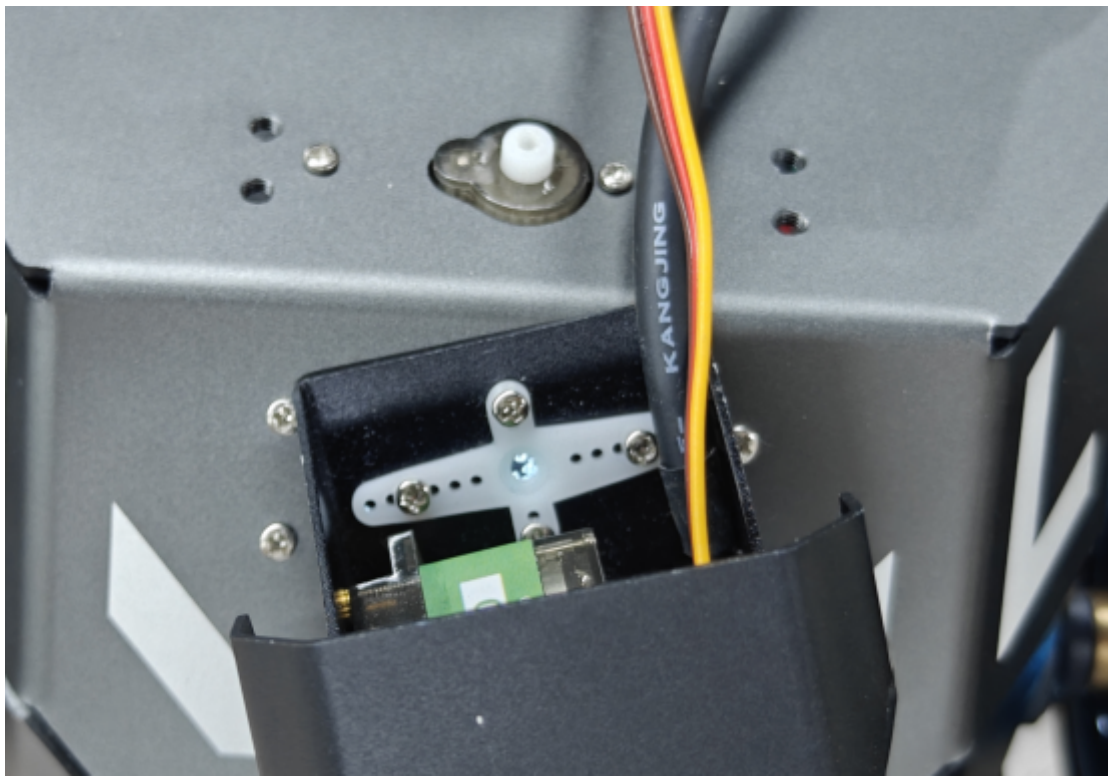
Note: For the ROSMASTER-A1 robot's PTZ, perform the following steps before assembly to initialize the angles.

1. Installation

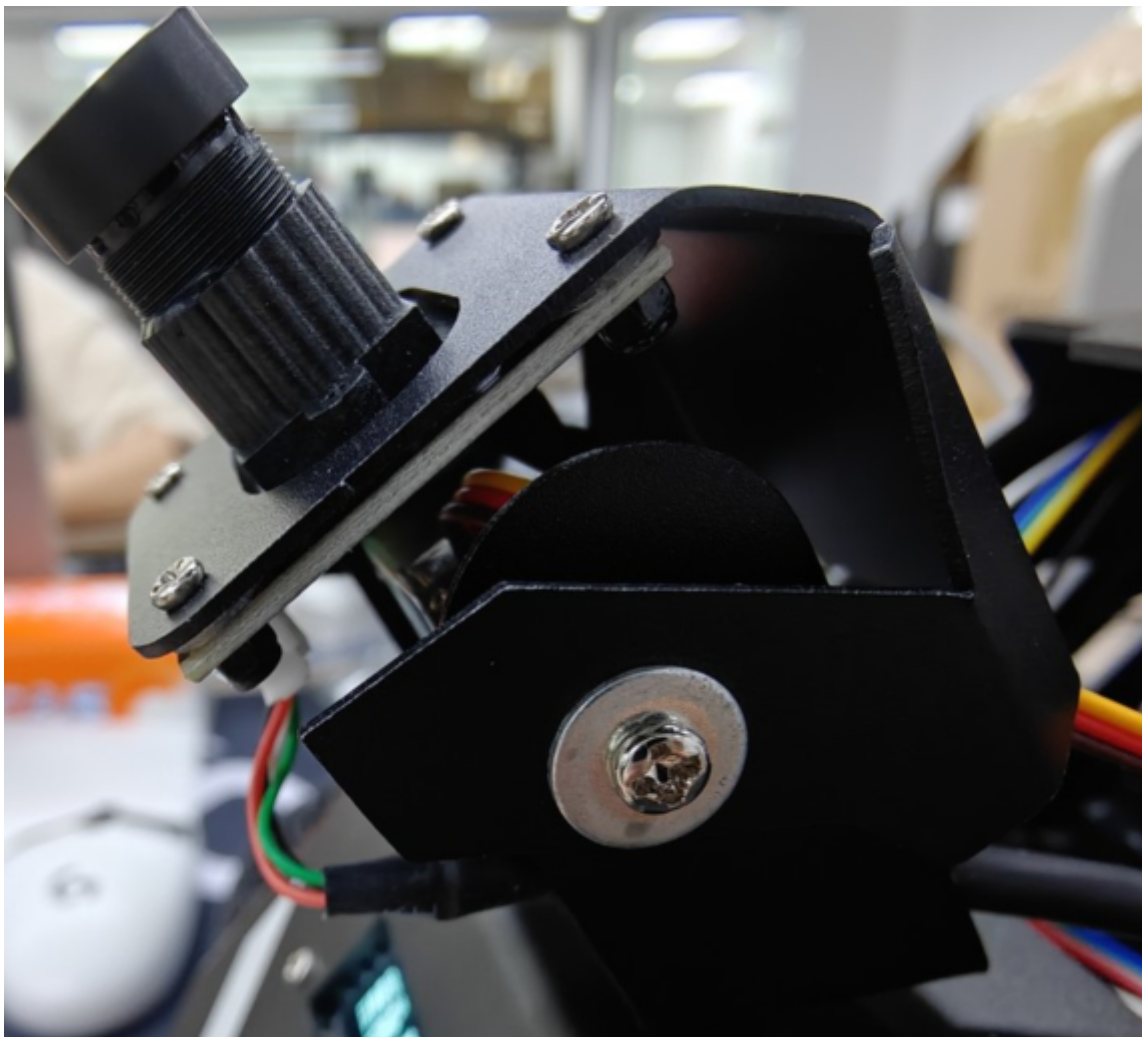
1. With the robot powered off, connect the corresponding servo cables (S1 for the horizontal servo, S2 for the vertical servo).



2. Leave the PTZ servos disassembled. Power on the robot. The S1 and S2 servos will reset to their initial angles.



3. Install the PTZ servo. Be careful not to adjust the horizontal servo gear angle at this point. Turn off the power to the robot and tighten the screws.



2. Initial Value Settings

2.1 Servo Initial Values in the Auto-Startup Program

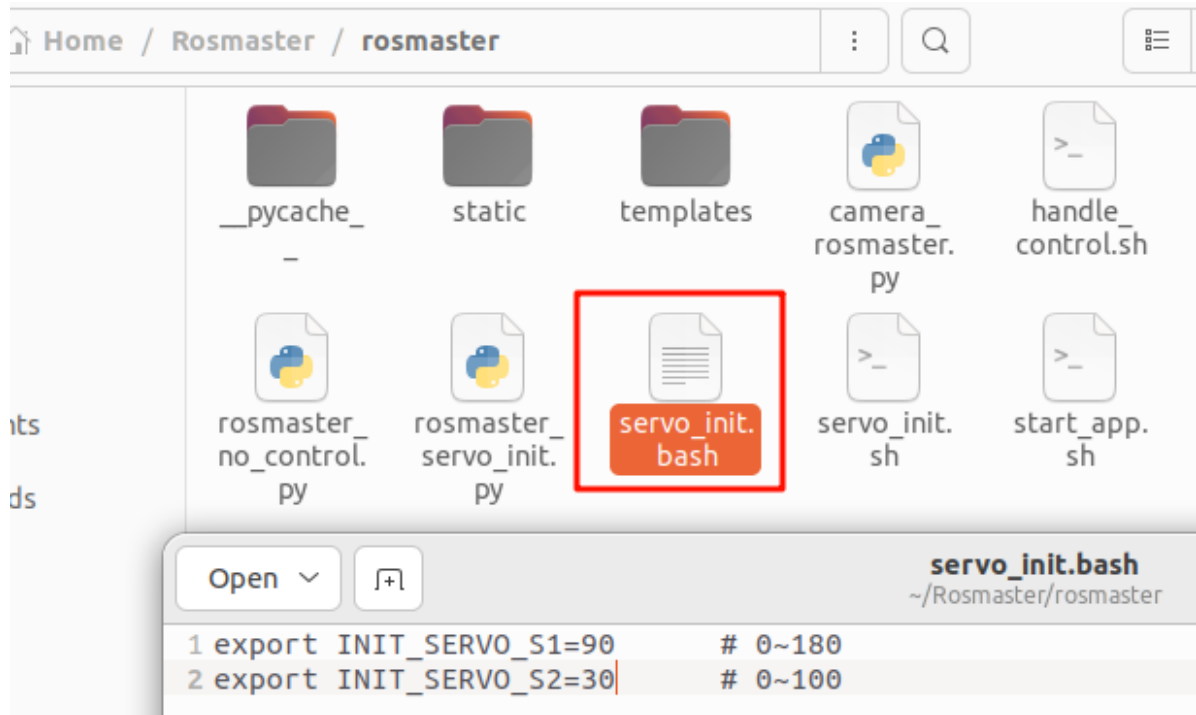
Each time the robot is powered on, the driver board sets the PTZ to an initial angle of (90,90). After entering the system, the Servo Initial Position Program automatically starts and sets the PTZ to a preset angle of (90,30).

⚠ Due to the servo's inherent structure, the gears will inevitably be offset by an angle. Therefore, setting it to 90 degrees does not necessarily result in the camera being horizontally centered.

Modification method:

- ORIN users:

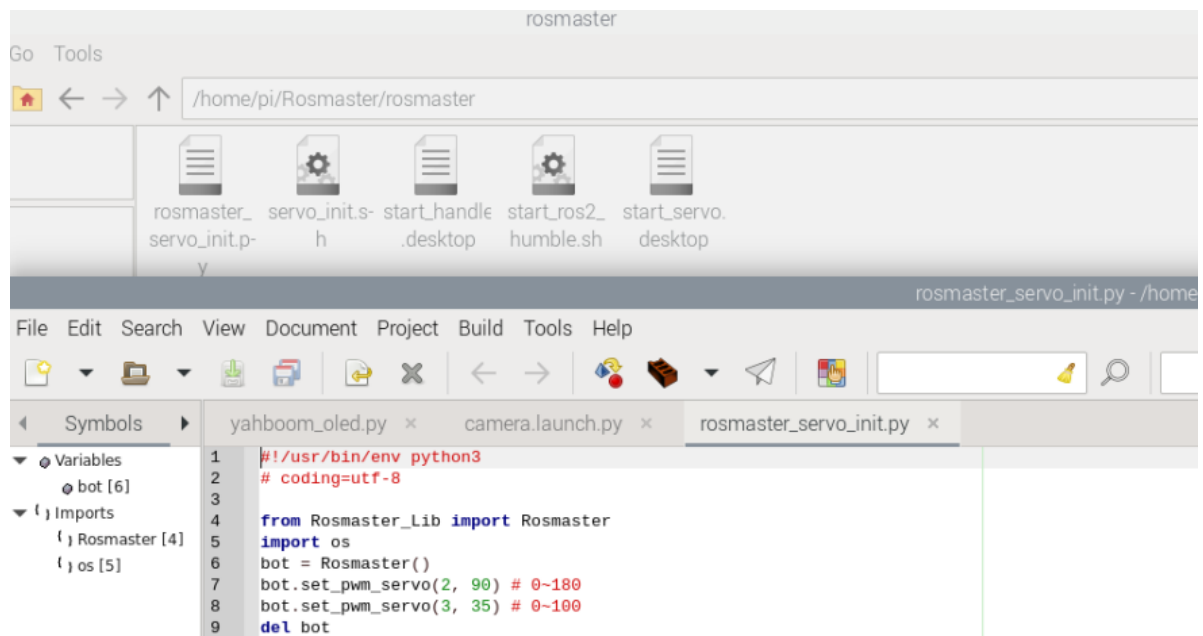
You can modify the parameters in `/home/jetson/Rosmater/rosmaster/servo_init.bash`.



🐘 This configuration will be applied to the `Servo initial position program` and the `Power-on auto-start joystick control program`.

- pi5/jetson nano users:

You can modify the parameters in `/home/pi/Rosmaster/rosmaster`.



👉 This configuration will be applied to the `Servo Initial Position Program`

After entering Docker, **(If you don't understand the operation, please refer to the Docker tutorial)**

```
vim run_handle.sh
```

```

-----
ROS_DOMAIN_ID: 61 | ROS: humble
my_robot_type: A1 | my_lidar: tmini | my_camera: nuwa
-----
root@raspberrypi:~# vim run_handle.sh

```

```

root@raspberrypi: ~
File Edit Tabs Help
#!/bin/bash

bash -c "source /opt/ros/humble/setup.bash;

source /root/yahboomcar_ros2_ws/yahboomcar_ws/install/setup.bash;

echo "Controller_control";

export INIT_SERVO_S1=90;
export INIT_SERVO_S2=35;

export ROS_DOMAIN_ID=61;

ros2 launch yahboomcar_ctrl yahboomcar_joy_launch.py;"

exit 0

```

Save the file and exit.

👉 This configuration will be applied to the `Handle Control Program` that starts automatically at startup

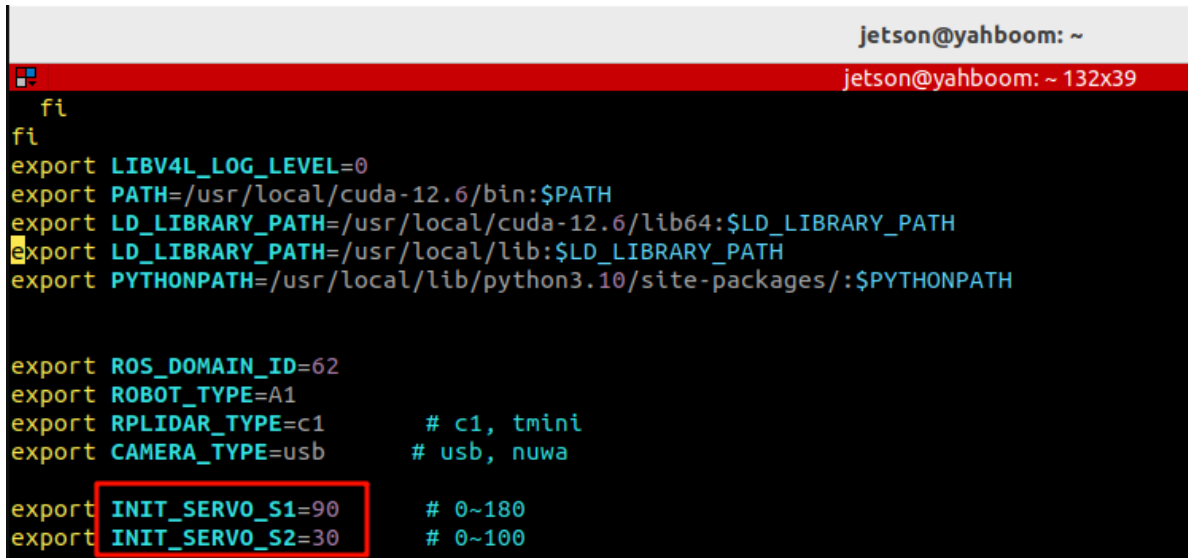
2.2 Initial Servo Values in Vision Cases

In most vision tracking/following cases, the servo is reset to an initial value when it is started. This is done by retrieving the servo angles preset in the environment variables.

Modification method:

- ORIN users:

```
sudo vim ~/.bashrc
```



```
jetson@yahboom: ~
jetson@yahboom: ~ 132x39
fi
fi
export LIBV4L_LOG_LEVEL=0
export PATH=/usr/local/cuda-12.6/bin:$PATH
export LD_LIBRARY_PATH=/usr/local/cuda-12.6/lib64:$LD_LIBRARY_PATH
export LD_LIBRARY_PATH=/usr/local/lib:$LD_LIBRARY_PATH
export PYTHONPATH=/usr/local/lib/python3.10/site-packages/:$PYTHONPATH

export ROS_DOMAIN_ID=62
export ROBOT_TYPE=A1
export RPLIDAR_TYPE=c1      # c1, tmini
export CAMERA_TYPE=usb     # usb, nuwa

export INIT_SERVO_S1=90    # 0~180
export INIT_SERVO_S2=30    # 0~100
```

- pi5/jetson nano users:

After entering Docker, (If you don't understand the operation, please refer to the [Docker tutorial](#))

```
sudo vim ~/.bashrc
```



```
# ~/.bashrc: customized by the user
#f1
export ROS_DOMAIN_ID=61
export ROBOT_TYPE=A1
export RPLIDAR_TYPE=tmini    # c1, tmini
export CAMERA_TYPE=nuwa     # usb, nuwa

export INIT_SERVO_S1=90    # 0~180
export INIT_SERVO_S2=35    # 0~100

echo "-----"
echo -e "ROS_DOMAIN_ID: \033[32mROS_DOMAIN_ID\033[0m | \033[34mROS: $(printenv ROS_DISTRO)\033[0m"
echo -e "my_robot_type: \033[32mROBOT_TYPE\033[0m | my_lidar: \033[32mSRPLIDAR_TYPE\033[0m | my_camera: \033[32mCAMERA_TYPE\033[0m"
echo "-----"
```

Save the file and exit.

👉 This configuration will be applied in the [Visual Tracking/Following Example](#)