# Start the Transbot_se container

The ROS environment of Raspberry Pi 5 and the hardware control case of transbot se are both located in Docker. Every time we run a case, we need to confirm that it is in the docker container of transbot se.

```
Starting docker yourself is necessary to run a single case or personal
development.
```

## Preliminary preparation

**Note: The startup program must be closed to run a single case!**

For a tutorial on closing the large boot program, please refer to the tutorial [Transbot SE expansion board control tutorial: closing the large self-starting program].

```
[Preparation before development: Close the startup program]: The tutorial
provides temporary/permanent shutdown of the startup program and
shutdown/restoration of the startup Docker script operation!
```

For first time use, it is recommended to temporarily close the startup program. For simple developers, you can try to permanently close the startup program.

```
If the recovery of the large startup program fails permanently, you can follow
the tutorial to reverse the recovery or burn the system yourself to restore the
factory state.
```

## Start Docker

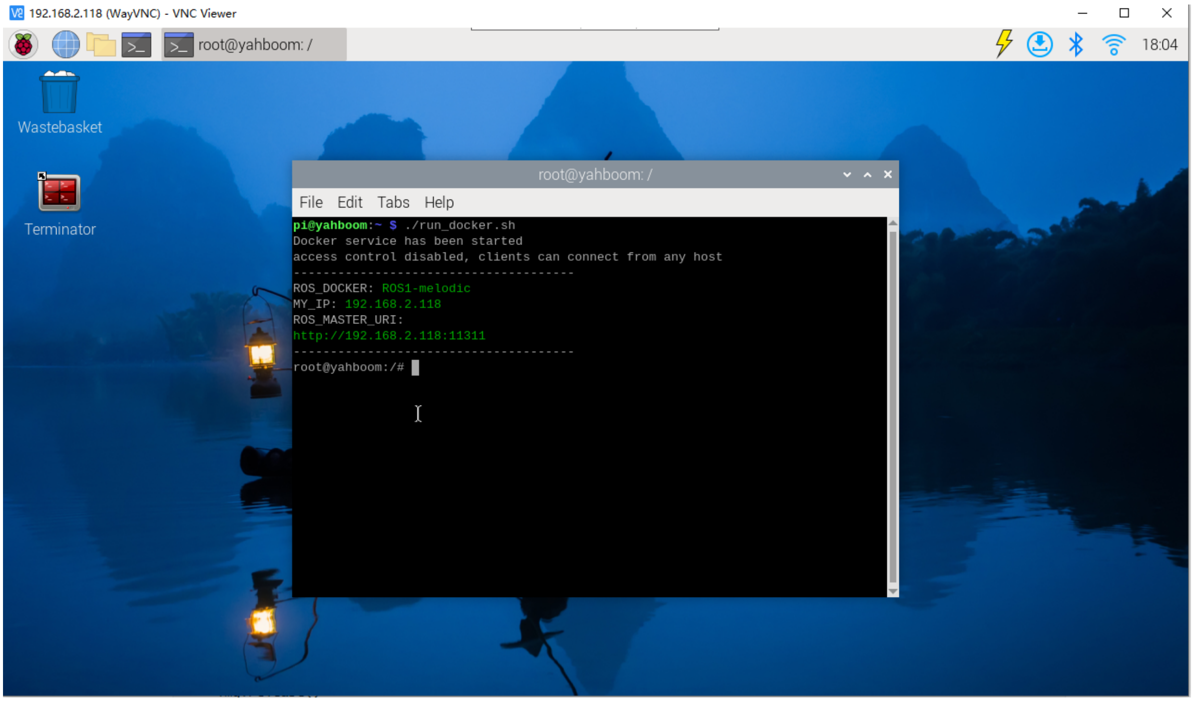
We provide the script file `run_docker.sh` for running the docker image corresponding to transbot se in the home directory on the Raspberry Pi 5 motherboard.

```
#!/bin/bash

# Wait for the Docker service to start
while true; do
     if systemctl is-active --quiet docker; then
         echo "Docker service has been started"
         break
     else
         echo "The Docker service has not started, waiting..."
         sleep 1
     fi
```

```
done


xhost+
docker run -it \
--net=host \
--env="DISPLAY" \
--env="QT_X11_NO_MITSHM=1" \
-v /tmp/.X11-unix:/tmp/.X11-unix \
--security-opt apparmor:unconfined \
-v /home/pi/temp:/root/temp \
--device=/dev/ttyAMA0 \
--device=/dev/video0 \
--device=/dev/input \
--device=/dev/video1 \
yahboomtechnology/ros-melodic:Transbot_Se_V1 /bin/bash
```

Start command

```
./run_docker.sh
```



```
pi@yahboom:~ $ ./run_docker.sh
Docker service has been started
access control disabled, clients can connect from any host
-------------------------------------
ROS_DOCKER: ROS1-melodic
MY_IP: 192.168.2.118
ROS_MASTER_URI:
http://192.168.2.118:11311
-------------------------------------
root@yahboom:/#
```

# Precautions

If the hardware device is not connected correctly, starting docker may fail!

## Example: USB camera

If the camera is not connected or the one identified is not /dev/video0, Docker may not be able to enter correctly or the camera case will report an error.

If we need to cancel the hardware mapping of the USB camera, we can delete the corresponding `--device=/dev/video0 \` information in the startup script.

```
#!/bin/bash

# Wait for the Docker service to start
```

```
while true; do
     if systemctl is-active --quiet docker; then
         echo "Docker service has been started"
         break
     else
         echo "The Docker service has not started, waiting..."
         sleep 1
     fi
done


xhost +
docker run -it \
--net=host \
--env="DISPLAY" \
--env="QT_X11_NO_MITSHM=1" \
-v /tmp/.X11-unix:/tmp/.X11-unix \
--security-opt apparmor:unconfined \
-v /home/pi/temp:/root/temp \
--device=/dev/ttyAMA0 \
--device=/dev/input \
--device=/dev/video1 \
yahboomtechnology/ros-melodic:Transbot_Se_V1 /bin/bash
```