## Introduction

This document outlines the steps to follow starting from when a robot is first built to when it is ready to be used in the system.

## Checklist (Summary)

* Visual check to verify that robot is built correctly
* Charge robot
* Burn SD Card
* Verify IP address
* Try to SSH for the first time
* OpenCR Setup
* Add launch commands to bashrc file
* Change namespace files
* Verify everything up until now
* Markers and Motive

## Charge Robot

1. Turn robot off
2. Unplug robot’s red cable that’s attached to its battery
3. Plug in power adapter and connect to blue charging thing
4. Plug the white thing coming out of the battery into the blue charger
   1. Green light = fully charged; red light = charging
5. When charged, first unplug robot from charger and then reattach the red cable

## SD Setup and IP Verification

1. Burn SD card (image??)
2. Insert the SD card with the writing side up into the Raspberry Pi, right underneath it
3. Plug a keyboard with USB and an HDMI cable from a nearby monitor into the Raspberry Pi
4. Turn the robot on (make sure red thing is plugged)
5. The screen should output a bunch of stuff ending with “done,” then hit “Enter”
6. At ubuntu login prompt, enter: username=”ubuntu”, password=”crl1234”
7. Run ifconfig and note the IP address that is listed under “wlan0” under the field “inet”
8. Shut down the robot: sudo shutdown now (the screen should go dark)
9. Turn the robot off and unplug HDMI and USB

## First SSH

Just to test that the Robot has been set up properly and can be accessed via ssh.

1. Turn on the robot
2. In terminal, run the command ssh ubuntu@[ip address]
   1. The robot takes a few minutes to be ready, so you may need to run this command a few times, a few seconds apart
3. Enter the username (ubuntu) and password (crl1234)
4. When you have verified that it works, use the command logout to exit.
5. Turn off the robot

## OpenCR Setup

1. SSH into the robot: ssh ubuntu@[IP\_ADDRESS]
   1. Alternative: you can also plug the robot in using HDMI and USB cords as described earlier in the SD setup step.
2. Follow [these](https://emanual.robotis.com/docs/en/platform/turtlebot3/opencr_setup/#opencr-setup) instructions. **Make sure to select “dashing” at the top**.
   1. Note: I (shaya) got an expired key error when I ran “sudo apt-get update” for a few of the robots, and I fixed it by running: sudo apt-key adv --recv-keys --keyserver keyserver.ubuntu.com [KEY]
3. Verify that it worked using the tests provided at the end (pushing SW1 and SW2 buttons).

*Note: after following the OpenCR setup instructions, the robot should make a sound every time it is turned on.*

## Add Launch Commands to bashrc File

Add a few commands to the robot’s ~/.bashrc file so that every time somebody ssh’s into the robot, it launches.

1. SSH into the robot: ssh ubuntu@[IP\_ADDRESS]

Password: crl1234

1. Access the bash file: nano ~/.bashrc
2. Scroll down to the bottom
3. Add:

if [[ -n $SSH\_CONNECTION ]] ; then

export TURTLEBOT3\_MODEL=burger

ros2 launch my\_tb3\_launcher my\_tb3\_bringup.launch.py

echo "success"

fi

1. Ctr + O; Enter; Ctrl + X
2. Log out: logout

## Namespace Files

Follow this tutorial starting at step 2: <https://discourse.ros.org/t/giving-a-turtlebot3-a-namespace-for-multi-robot-experiments/10756>

All of the relevant directories and files have been created already; you just need to change the parameters/robot name in the following files:

~/turtlebot3\_ws/src/my\_tb3\_launcher/launch:

* hlds\_laser.launch.py (1 change)
* Turtlebot3\_state\_publisher.launch.py (1 change)
* My\_tb3\_bringup.launch.py (1 change)

~/turtlebot3\_ws/src/my\_tb3\_launcher/param:

* Burger.yaml (2 changes)

Naming convention: robot\_1, robot\_2, ...

**Don’t forget to compile according to the instructions. To verify results, use the tests outlined in the instructions.**

## Markers and Motive recognition

Follow the instructions in the Bundle Documentation for robots.