
Summit XL Documentation

Release 0.1

Victor Risager

Mar 08, 2024

CONTENTS

1	Contents	3
1.1	Getting Started	3
1.2	Packages	5
1.3	Automatic Startup of ROS	7
1.4	Docker	7
1.5	Changelog	7

Summit XL is a robot developed by the spanish based company Robotnik. The robot is an outdoor robot with a 4x4 wheel drive and a 270° laser scanner and a front facing depth camera. The robot is used for surveillance and other outdoor tasks that require a sturdy wheel base with good manouverability. In addition the robot can be equipped with mecanum wheels to allow for indoor omnidirectional movement. The robot is based on ROS melodic.

For more information on the specifications of the robot, please refer to: [Robotnik Website](#)

Check out the [Getting Started](#) section for information on how to get the robot up and running.

Note: This documentation is not supposed to replace the official documentation of the robot. It is supposed to be a complement to the official documentation including but not limited to the *README.md* files and manuals. The objective is to provide a more hands on approach to the robot.

CONTENTS

1.1 Getting Started

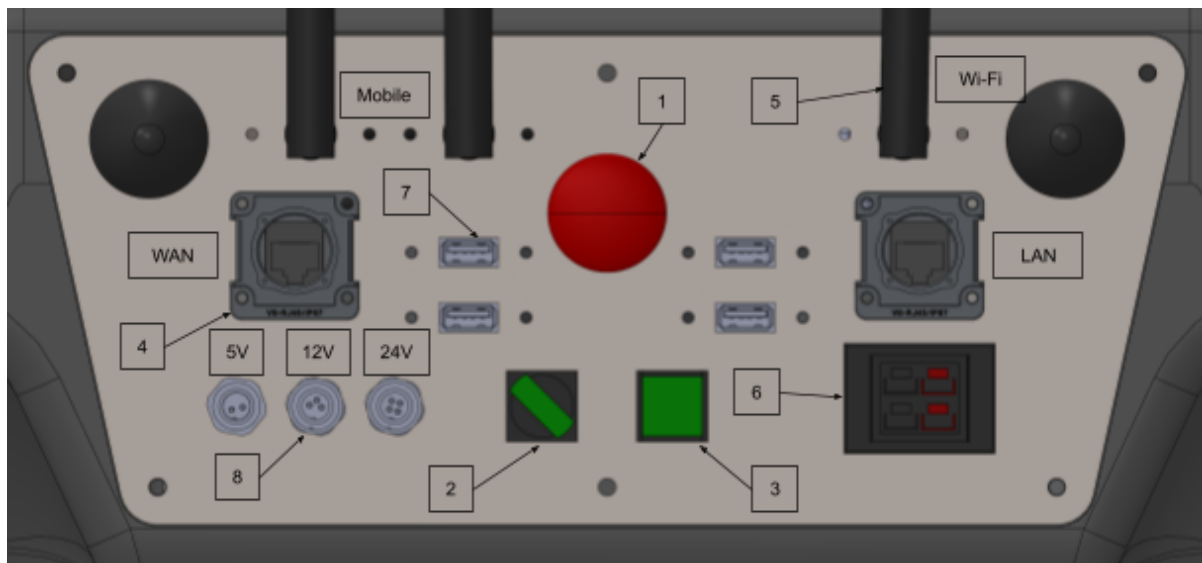
To get started with the robot you need to make sure that you have the following dependencies installed on your pc:

- ROS Melodic
- Ubuntu 18.04

Note: You may prefer to use Docker to containerize the development environment. If you are interested in using Docker, please refer to the [Docker](#) section.

1.1.1 Robot startup

To start up the robot, please refer to this image:



1. Rotate the power switch (2) to the ON position.
2. The CPU button (3) will turn green when the onboard computer is powered on.
3. Connect a monitor, keyboard and mouse to the onboard computer using the HDMI and USB ports (7).

4. The robot will boot up and the robot's computer will automatically start ROS and the robot's packages. For this automatic startup sequence please refer to the *Automatic Startup of ROS* section.
5. Once the robot is booted up, you can see the robot's manual on the desktop. Please read this manual carefully before proceeding.
6. At this point, you can turn on the provided PS4 controller by pressing the PS button. The controller will automatically connect to the robot's computer. To drive the robot, you must keep the *R1* button preessed. To activate the omnidirectional drive, press *o*, and you can increase/decrease the speed using the triangle and cross buttons respectively.

1.1.2 Robot Network setup

1. Connect to the robot's wifi network. The robot's wifi network is called `sh100-210420ab` and the password is `R0b0tn1K`.
2. Add the hostname and IP of your PC to the robot's `/etc/hosts` file. This is necessary for the robot to be able to communicate with your PC. You can do this using `vim` or `nano`:

```
sudo vim /etc/hosts
```

or

```
sudo nano /etc/hosts
```

When inside the file, add the following line to the end of the second section of the file:

```
<your_ip> <your_pc_hostname>
```

This is important such that the robot can resolve your hostname to your IP address. You can find your IP address by running the following command:

```
hostname -I
```

Note: To use SSH, you must deactivate the firewall on the robot. You can do this by running the following command:

```
sudo ufw disable
```

Even though it says that it is disabled after reboots, it is still active after a reboot. I have yet to find a solution to this problem.

1.1.3 Host Setup

1. Add the robot's hostname and IP to your PC's `/etc/hosts` file. The robots hostname and IP is as stated in the manual:

```
192.168.0.200 sh100-210420ab
```

2. Now you should be able to ping the robot from your PC:

```
ping sh100-210420ab
```

And you can even list the topics being published by the robot:


```
rostopic list
```

1.1.4 What is next

- Now that you have the robot up and running, you can start developing your own packages for the robot. You can also use the provided packages to control the robot. Please refer to the [Packages](#) section for more information on the packages that are installed on the robot.
- For information regarding the automated startup process of the robot, please refer to the [Automatic Startup of ROS](#) section.
- It is recommended to use a containerized environment on your host machine to develop code for the robot, for more information regarding this, please refer to the [Docker](#) section.
- In case of any issues with the current setup of the robot, you can see the changelog for the robot in the [Changelog](#) section.

1.2 Packages

The build snapshot of the Summit Robot can be found in `~/catkin_ws/src/summit_packages_release_melodic-devel/` of the robot, and it contains multiple packages that are used to control the robot. More specifically, it contains 148 packages, distributed between the following folders. The packages are:

1.2.1 Command Manager

1.2.2 Deb

1.2.3 Imu

1.2.4 Localization

1.2.5 Manipulation

1.2.6 msgs

1.2.7 Navigation

1.2.8 Others

1.2.9 Robot

1.2.10 Robot Bringup

For official documentation, Please visit the README.md file in `summit_packages_release_melodic-devel/robot_bringup/README.md`

The Robot Bringup package is common configuration package that is suitable for use on all of Robotnik's Products. The objective of this package is to cover as much of the configuration parameters as possible. Therefore this package is a good starting point for any new user.

It contains the relevant environment variables required by the other Robotnik Packages.

In addition it contains bringup files for all of the different components of the Robot.

Installation

When the robot is first build and ready to install software on it, `summit_packages_release_melodic` is installed onto the onboard compututer. Then following the installation instructions in the README.md file in `summit_packages_release_melodic-devel/robot_bringup/README.md` will install the necessary Robot parameters on the computer. The following scripts are located in the `scripts` folder of the `robot_bringup` package.

```
#!/bin/bash
source ~/catkin_ws/devel/setup.bash && roscd robot_bringup
# Robot params
cp -r env/ ~/robot_params

# Boot scripts
cp scripts/bringup.sh ~/
cp scripts/ros_config.sh ~/

# Boot service
sudo cp scripts/bringup-ros.service /etc/systemd/system/
sudo systemctl enable bringup-ros.service
sudo systemctl start bringup-ros.service

echo "source ~/ros_config.sh" >> ~/.bashrc
```

This code does a few things. It copies the environment variables to a folder called `robot_params` in the home directory. As mentioned, the `/robot_params` is the main configurable parameters that the user can tune as they desire.

Since the boot process needs to execute a few things, which is described in detail in *Automatic Startup of ROS*. The `systemctl` service needs to call `bringup.sh`, where most of the startup happens among other things, this is where the `roscore` is executed. The `ros_config.sh` file is a helper file that sets up the environment variables for the user. The `ros_config.sh` file is then sourced in the `.bashrc` file so that the user can use the environment variables in the terminal and it displays some default information for the user.

It also copies the boot scripts to the home directory and the boot service to the `/etc/systemd/system/` directory. The boot service is then enabled and started such that it is persistent to autostart at every boot reboot of the summit robot.

Lastly, notice in the `scripts` folder that there is a `bringup-ros.service`, which defines the service that is started at boot. This service calls the `bringup.sh` script, which is the main script that starts the robot.

1.2.11 Sensors

1.2.12 Simulation

1.2.13 Web

Note: These packages are developed to be applicable for all Robotniks products, so some of them may not be used in the Summit Robot.

1.3 Automatic Startup of ROS

This page describes in detail how the ROS system is started up automatically on the robot.

1.4 Docker

This page describes how to build and run the Docker container for the development environment. Subsequently, it is also possible to setup the development environment using vscode's devcontainer extension.

1.5 Changelog

1.5.1 Networking

This page contains information about the changes made to the summits networking configuration.