MYBOTSHOP

QUADRUPED ROBOTICS

Unitree Go2

User Manual

CONTACT

info@quadruped.de

support@quadruped.de

quadruped.de

Contents

1 Attention

Attention: Read this document carefully. In case of doubt, it is essential to consult specialists, experts, or the manufacturer (Unitree Robotics) or distributor (MYBOTSHOP / QUADRUPED Robotics) of the assemblies used. The robot must not be put into operation before clarification. Contact information is available in Chapter 11, Getting Help & RMA.

Do not allow persons who are not familiar with the robotics or these instructions to operate the robot. Robots are dangerous in the hands of untrained users.

Do not operate the robot if you are intoxicated, under the influence of drugs, or unable to concentrate. Familiarize yourself with the low-level joint control safety implications and the emergency damping procedure (L2 long-press + B) before switching the robot on.

Caution: Low-level control bypasses the built-in balancing safeguards entirely — an unsupported GO2 in low-level mode can fall over or damage its joints. Certain athletic actions (e.g. wallow/rollover) can flip the robot onto its back and damage an attached Open Manipulator or D1 arm. If equipped with a robotic manipulator, always return it to its home position before powering the arm down, and review the Safety Guidelines in Chapter 9 before operating.

2 Disclaimer

The GO2 robot documented here is sold and supported by *MYBOTSHOP / QUADRUPED Robotics* as the German distributor for Unitree Robotics. Basic ROS 2 (Foxy) understanding is required to operate the software stack covered in this manual — if you are not familiar with ROS 2, we recommend checking the ROS 2 documentation first.

The client acknowledges and agrees that any information or materials provided by MYBOTSHOP / QUADRUPED Robotics are for R&D and development purposes only. Any services are provided "AS IS" and without any representation or warranty of any kind, express or implied, including but not limited to any warranty of merchantability, fitness for a particular purpose, non-infringement, or any other warranty.

This product is intended for civilian use only. Users should avoid making dangerous modifications or using the robot in hazardous ways. Some features described in this manual (e.g. the Open Manipulator variant, the D1 servo arm, the Steamdeck accessory, or a specific LiDAR option) apply only to specific hardware configurations — always confirm which accessories are fitted to your unit before following an accessory-specific procedure. Note that when connected over the internet, the GO2's large-language-model and login information is transferred to Unitree. For terms, policies and compliance with local laws and regulations, refer to the Unitree Robotics website and the RMA/support process in Chapter 11.

MYBOTSHOP / QUADRUPED Robotics shall not be liable for any damages, including but not limited to direct, indirect, special, incidental, or consequential damages, arising out of or in connection with the use or inability to use the information or materials in this document. This limitation on liability shall apply regardless of the form of action, whether in contract, tort, or otherwise.

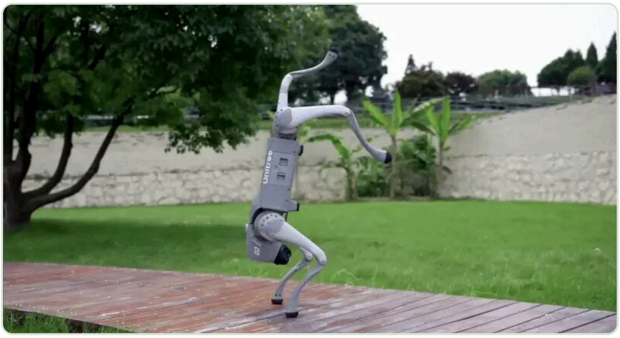
3 About the GO2

GO2 Quadruped Tutorials

This package supplies Sphinx-based tutorial content to assist you with setting up and operating your GO2 quadruped robot. The tutorials topics are listed in the left column, and presented in the suggested reading order.

Important: Please be aware that the large language model and login information when connected over the internet has the information transferred to Unitree throughout.

The Unitree Go2 represents the new frontier of embodied AI — a powerful fusion of robotics and intelligence that redefines the future of bionic quadrupeds. Designed with state-of-the-art perception, advanced AI-driven locomotion, and robust mechanical performance, the Go2 is engineered to explore, learn, and interact with the world around it.

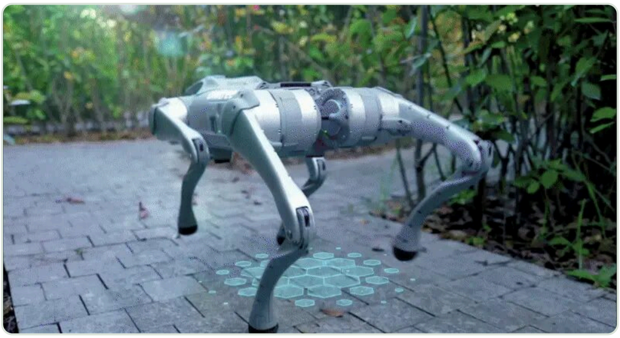


Unitree Go2 Robot

Unitree's relentless innovation and technological leadership have led to the creation of the Go2 — an intelligent, agile, and reliable partner for research, exploration, and real-world applications.

Revolutionary Perception with 4D LiDAR

The Go2 introduces the Unitree self-developed 4D LiDAR L1 , offering a 200% improvement in recognition performance. With an ultra-wide hemispherical field of view (360° × 90°), a minimum detection distance of just 0.05m , and exceptionally small blind spots, the Go2 is capable of true all-terrain recognition and safe navigation.



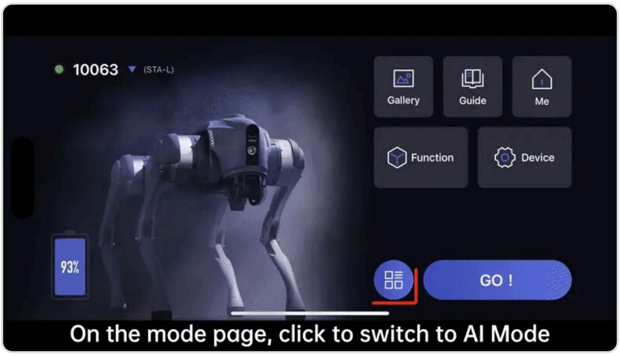
Go2 4D LiDAR

Embodied Intelligence: AI + Robot

Powered by GPT-driven intelligence , the Go2 integrates AI with robotics to achieve a new level of decision-making and adaptability . Through large-scale AI simulation training , the robot masters complex gaits including:

- Upside-down walking
- Adaptive roll-over recovery
- Obstacle climbing
- Agile, stable locomotion in dynamic environments

This makes the Go2 one of the most versatile and intelligent quadrupeds in its class.



Go2 4D LiDAR

Dynamic Movements and Expressive Poses

The Go2 is designed not only for functional mobility but also for expressive interaction . It can perform a wide variety of actions and poses, such as:

- Jumping and stretching
- Shaking hands
- Cheering and pouncing
- Sitting and resting

Enhanced Battery and Power System

With an upgraded 8,000mAh battery (standard) and an optional 15,000mAh ultra-long-life battery , the Go2 achieves up to 150% greater endurance compared to previous models. Its 28.8V system voltage enhances motor efficiency, stability, and overall performance .

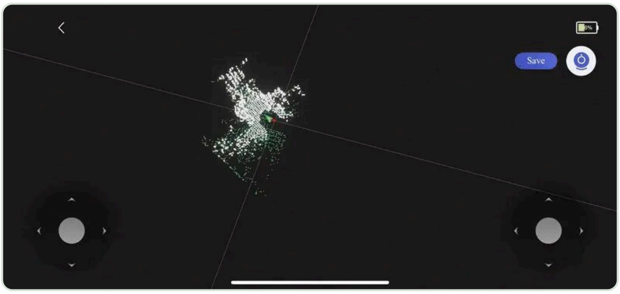
Go2 Battery

Superior Motor Performance

The Go2's redesigned actuators deliver 30% improved performance with a peak joint torque of 45 N·m . A new internal trace connection technique and integrated heat pipe cooling ensure stability during long-term operation, even under demanding workloads.

3D LiDAR Mapping and Autonomy

Using the built-in 4D LiDAR L1 and a dedicated mobile App , users can create 3D point cloud maps and define navigation paths. This enables the Go2 to move autonomously and adapt intelligently to complex environments.



Go2 Battery

ISS 2.0 — Intelligent Side-Follow System

The Go2 is equipped with the latest ISS 2.0 (Intelligent Side-follow System) . With advanced wireless vector positioning and control , it offers:

- 50% improvement in positioning accuracy
- Remote control distance >30m
- Optimized obstacle avoidance strategies

This ensures smooth and intelligent traversal over challenging terrains.

Go2 Battery

OTA — Continuous Evolution

The Go2 is designed to keep evolving . With user authorization , it connects to the cloud-based OTA service , allowing seamless program upgrades that continuously improve performance and expand functionality.

Go2 Battery

Technical Specifications

Core Capabilities	
Recognition System	Standard Ultra-Wide 4D LiDAR L1
Maximum Running Speed	~5 m/s (tested in lab conditions)
Peak Joint Torque	~45 N·m
Wireless Modules	Wi-Fi 6 / Bluetooth / 4G
Battery Endurance	~2–4 hours (depending on configuration and use)
Side-follow System	ISS 2.0

Go2

Intelligent Interaction

- Real-time HD image transmission via the mobile App
- 4G & eSIM support for stable long-range connectivity
- Graphical programming interface with drag-and-drop functions for beginners and developers
- Advanced obstacle avoidance using 4D LiDAR with precise 3D world reconstruction

Pushing Boundaries with Technology

The Unitree Go2 is more than just a quadruped robot — it is a new intelligent creature . With advanced AI, superior hardware, and a constantly evolving system, the Go2 leads the next generation of bionic robotics , setting a new standard for performance, intelligence, and interaction.

Important: The listed specifications may vary based on configuration and usage environment. Maximum torque values differ between joints; 45 N·m represents peak performance at the largest joint. Battery endurance depends on operational conditions and selected battery configuration. Product features may continue to evolve via OTA updates. Always refer to the actual product for the latest details. This product is intended for civilian applications only; avoid unsafe modifications or hazardous usage.

4 Unboxing & Firmware Upgrade

Packaging Description

- The Go2 will be delivered within a secondary cardboard box, as illustrated in the images provided below.

- Within this secondary box, you will discover the primary white box featuring the robot's image and a description of its functionalities.

- Inside the primary box, there is a practical gray-colored travel trolley bag designed for convenient day-to-day transportation of the robot.

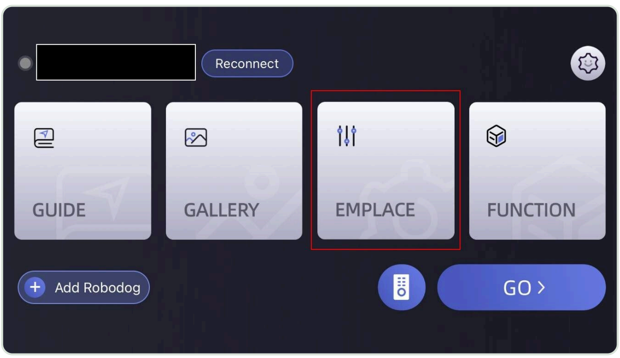
- The final product from the box will look as shown above.

You can access the unboxing video by following the provided link below.

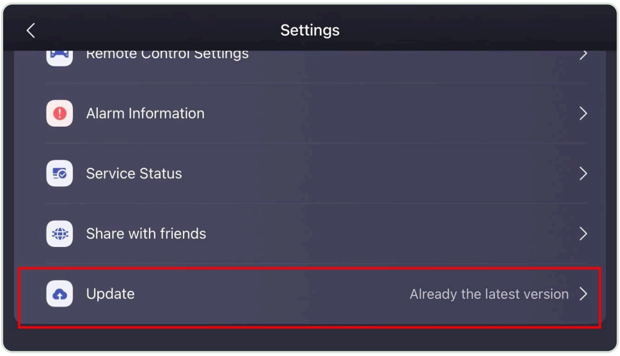
- GO2 Unboxing

Firmware Upgrade

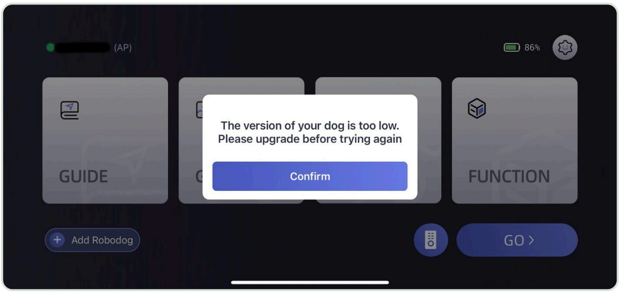
To upgrade your robot's firmware, start by accessing the application's home screen. Proceed to the EMPLACE option,



scroll down to the bottom of the displayed options, and locate the ‘update’ button. Select that option to update your robot’s firmware.



> **Important:** If your robot is operating on an older version of firmware, it may restrict proper app usage and prompt a pop-up as shown below.Therefore, it’s recommended to update the firmware first before using it.

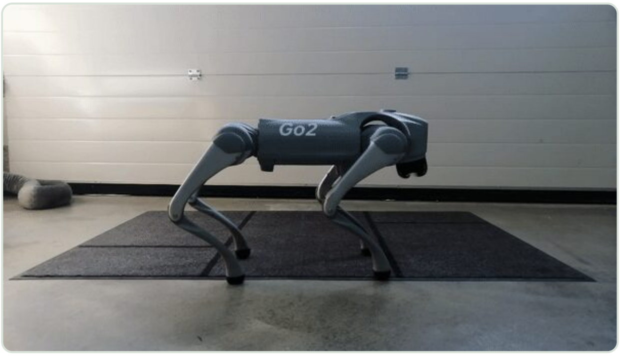


You can access the firmware upgrade video by following the provided link below.

- GO2 Firmware Upgrade

5 App Binding, Remote Control & Calibration

Manuals



The following text are hyperlinks to. Click on them to get redirected to the websites.

- Unitree Documentation

Apps

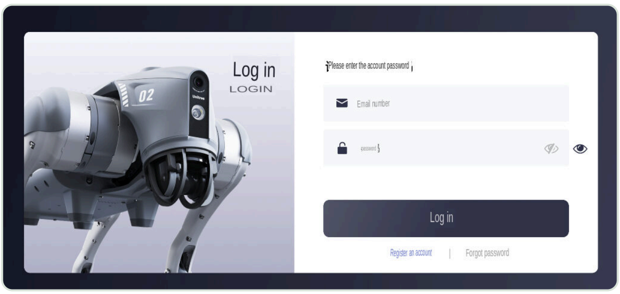
- GO2 iOS
- GO2 Android

App Binding

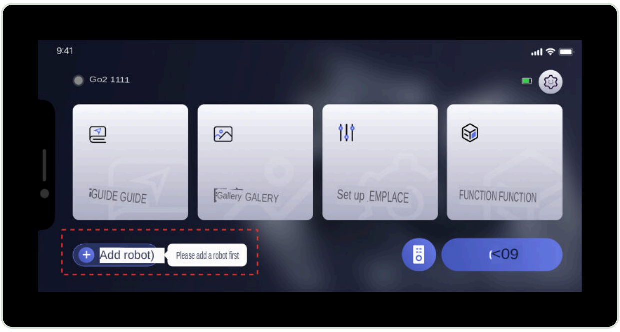
To set up the app, you can either download it from the provided links or search for it directly by the name "Unitree Go" on the App Store or Play Store.

Once you have downloaded the app, please follow these steps:

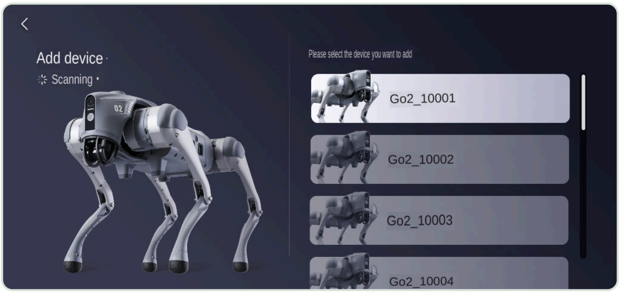
- Download and open the Unitree Go App. Register an account using your email address on the login page of the home screen and log in.



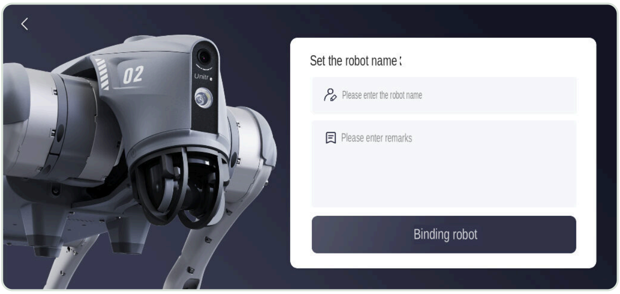
- Click on the home page to add a robot.



- Then, turn on the power of Go2 and simultaneously enable the Bluetooth on your mobile phone. Select the device you want to add for binding.

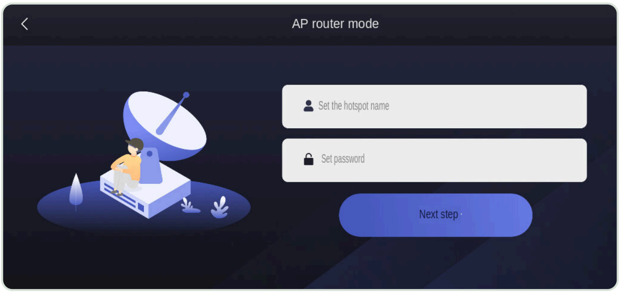


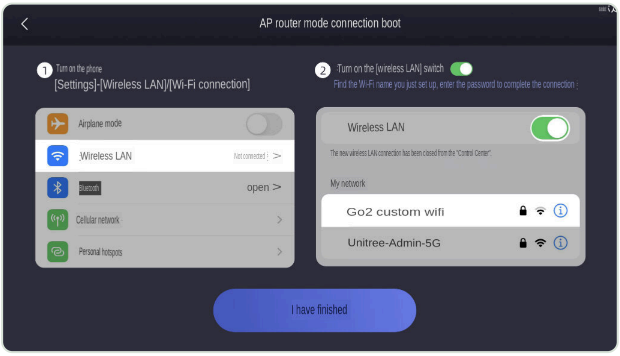
- Customize the robot's information as per your preferences.



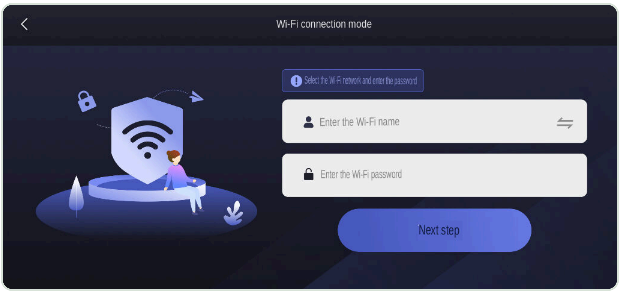
- Go2 needs to establish a connection with the app. After binding the robot, you will have two connection modes: AP router and Wi-Fi. Choose the one that suits your needs.

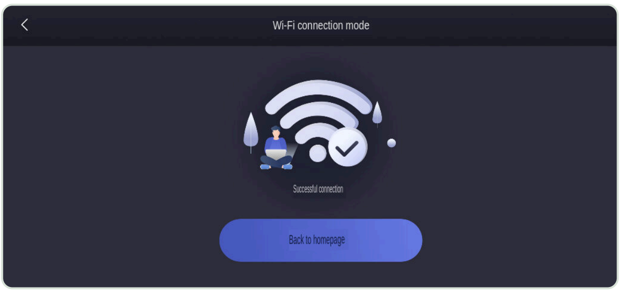
- If you select AP Mode , follow the prompts to set the hotspot name (GO2-XXXXXX) and an 8-digit password. Open the wireless LAN settings on your mobile phone,select the Wi-Fi name you just configured, and enter the password to complete the connection. Once the connection is established, return to the Unitree Go App and click "I'm done" to confirm a successful connection between Go2 and the app.





- If you choose Wi-Fi connection mode , select this mode to establish a connection. Pick an existing Wi-Fi network and enter the password. Wait for the connection to be established. After the Wi-Fi connection is completed, return to the Unitree Go App's home page, confirming that Go2 and the app are successfully connected.





You can access the App Binding video by following the provided link below.

- GO2 App Binding

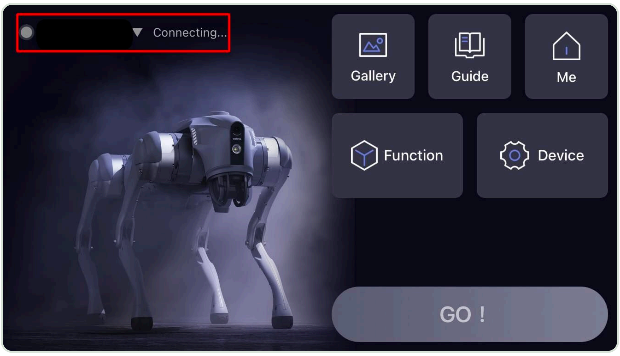
Additionally, the following video demonstrates how to adjust the robot's perspective view within the app after completing the setup.

- GO2 Prospective Switch

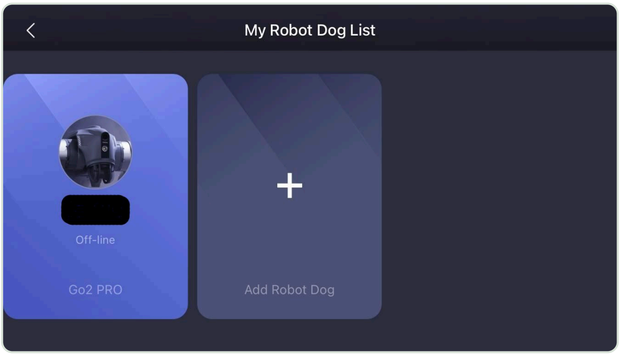
Multiple Binding

To bind more then one Go2 with your app (considering you already have a Go2 binded with the app)please follow the below mentioned steps:

- On the main screen please select the name of your robot mentioned on the top-left corner of the app.



- select the plus sign besides the already binded robot.



- Now follow the same steps as mentioned in the above section to setup your additional Go2

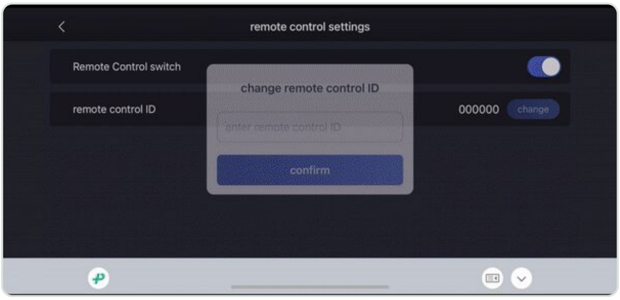
Remote Control Binding

To pair a remote control with the Go2, follow these steps:

- While the robot is powered on and connected to your mobile device, go to EMPLACE -> data -> Remote Control from the app's home screen.

- Choose change to modify the ID of the currently paired remote to match the new one you have.

- Enter the 6-digit ID found on the white sticker on the right front side of the remote, then confirm to bind the remote controller.



For a visual guide on Remote Control Binding, access the video through the provided link below.

- GO2 Accompany Mode

Basic Operations

Basic Movements	Description	Button Combination
Head Up / Down	Adjusts robot's head pitch angle.	Right Rocker (Push Forward / Backward)
Turn Left / Right	Turns robot's head left or right.	Right Rocker (Push Left / Right)
Move Forward / Backward	Moves robot forward or backward.	Left Rocker (Push Forward / Backward)
Sideways Movement	Moves robot sideways left/right.	Left Rocker (Push Left / Right)
Searchlight Switch	Toggles head searchlight ON/OFF. Default OFF.	L2 (Click) + SELECT (Click)
Obstacle Avoidance ON	Enables obstacle avoidance. (Default ON)	X (Click)
Obstacle Avoidance OFF	Disables obstacle avoidance.	Y (Long Press for 3s)

Mode Switches	Description	Button Combination
Damping Mode (Emergency Stop)	Immediately stops all robot motion.	L2 (Long Press) + B (Click)
Stand Lock / Low Down	Locks current pose or robot goes into prone position. Press twice to return to standing.	L2 (Long Press) + A (Click)
Unlock / Default Gait	Resets the robot to its default gait or unlocks it from a locked state.	START
Running	Robot runs at higher speed.	L2 (Long Press) + START (Click)
Pose	Makes the robot stand upright.	SELECT
Normal Gait	Resets the robot to its standard walking gait.	L1 (Long Press) + START (Click)

Classic Gait	Switches the robot to its classic gait style.	RIGHT (Long Press) + START (Click)
Free Walk	Switches the robot to its free walk style.	LEFT (Long Press) + START (Click)
Free Avoid	Enables free avoidance mode.	A (Double Click)
Cross Step	Enables cross-step walking pattern.	B (Double Click)
Bound / Jump	Enables jump or bound mode.	X (Double Click) / Y (Double Click)
Hand Stand / Erect	Performs a handstand or erect pose.	R1 (Double Click) / R2 (Double Click)
Endurance Mode	Enables long-duration, low-energy mode.	L1 (Long Press) + SELECT (Click)

Important: The AIR model currently does not support Free Avoid, Cross Step, Bound/Jump, Hand Stand/Erect .

Note: Please visit Unitree Go2 App to trigger more athletic modes!

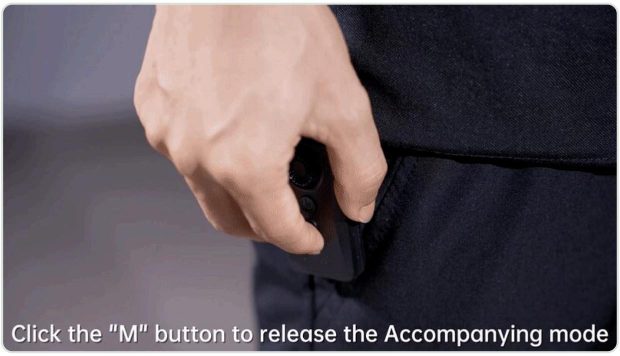
Customised Movements	Description	Button Combination
Stand Up from Fall	Makes robot stand up after a fall.	L2 (Long Press) + X (Click)
Stretch	Robot performs a stretching motion.	R2 (Long Press) + A (Click)
Shake Hands	Robot performs a handshake gesture.	R2 (Long Press) + B (Click)
Love	Robot shows a “love” gesture or pose.	R2 (Long Press) + Y (Click)
Greet	Robot performs a greeting pose.	L1 (Long Press) + A (Click)
Jump Forward	Robot jumps forward.	R1 (Long Press) + A (Click)
Sit Down	Robot sits down.	R1 (Long Press) + B (Click)
Pounce	Robot performs a pounce-like motion.	R1 (Long Press) + X (Click)
Dance 1	Robot performs first dance routine.	L1 (Long Press) + B (Click)
Dance 2	Robot performs second dance routine.	L1 (Long Press) + X (Click)

Accompany Mode

To activate the accompanying mode for the robot, you’ll need the small remote controller that’s included with it.

- Quickly press the “M” button twice on the remote to enter accompanying mode while the robot is turned on. The front headlamp will turn purple, indicating the robot is in accompanying mode. In this mode, the robot will maintain a specific distance and angle from the remote transmitter as it moves.

- To enable obstacle avoidance in accompanying mode, double-click the “L2” button on the remote.



- Press the “M” button once to exit accompanying mode.

For a visual guide on Accompany Mode, access the video through the provided link below.

- GO2 Accompany Mode

Voice Command/ 4G Function

Navigate to FUNCTIONS -> BenBen Dog from the app home screen while the robot is turned on and connected to your mobile device. You’ll find the option to select either Automatic or LongPress. This choice determines whether the robot responds to voice commands instantly as you speak, or if you prefer to activate voice commands by pressing the talk button. This voice command feature allows complete control over the robot, from executing tricks, navigating specific distances and angles, to playing your preferred music.

For a visual guide on Voice commad Mode, access the video through the provided link below.

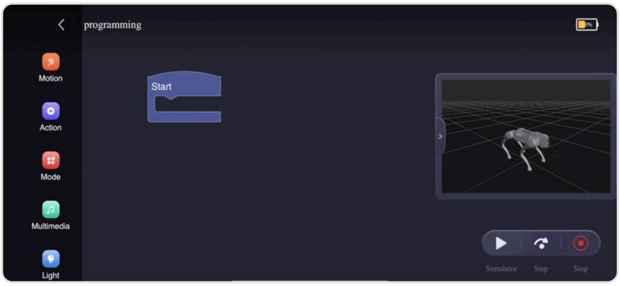
- GO2 Voice command Mode

Note: To enable this Funtion, the robot requires an internet connection through Wi-Fi.

Programming Mode

Navigate to FUNCTIONS -> Programming from the app home screen while the robot is turned on and connected to your mobile device.

It will show you a UI with different commands blocks that you can mix and match for make your robot do certan stuff on the go. You can also visualize the simulation of your code in a small visual box on the right of the sceen.



For a visual guide on Programming Mode, access the video through the provided link below.

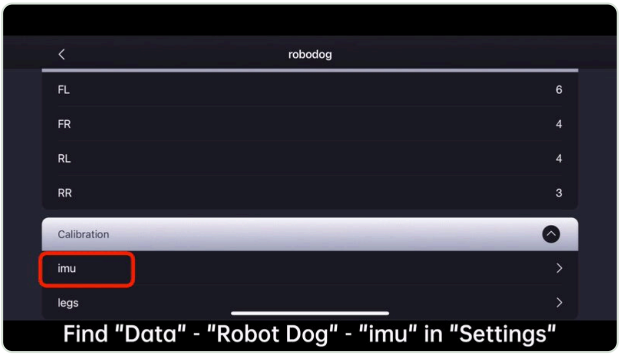
- GO2 Programming Mode

IMU Calibration

To calibrate the built-in IMU on the Go2, please follow these steps:

- Navigate to EMPLACE -> data -> robodog from the app home screen while the robot is turned on and connected to your mobile device. Scroll down to the bottom and select the imu option.

- The interface will prompt you to place the robot on an even and level surface, ensuring all four support pads are touching the ground.
- Then, press the calibration start button. During calibration, the front headlamp will turn blue and start flashing. Once it turns green, the calibration is complete.



- Restart the robot to save the calibration data.

For a visual guide on IMU Calibration, access the video through the provided link below.

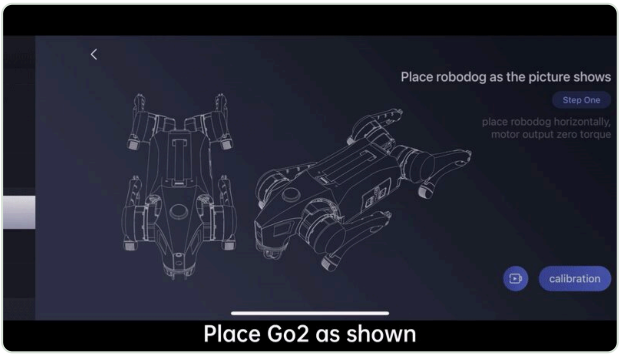
- GO2 IMU Calibration

Leg Calibration

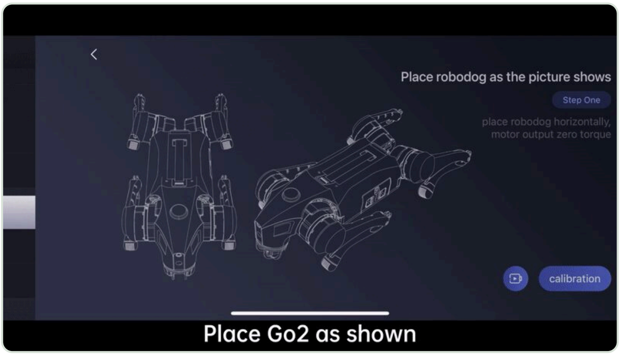
To calibrate the legs on the Go2, please follow these steps:

- Navigate to EMPLACE -> data -> robodog from the app home screen while the robot is turned on and connected to your mobile device. Scroll down to the bottom and select the legs option.

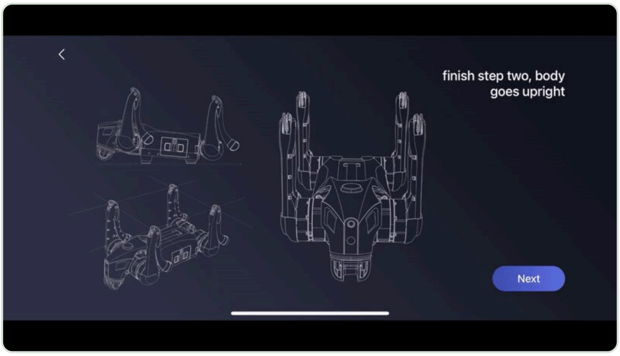
- A popup will show up, select continue which will in return show you steps to take to calibrate the robot.
- The interface will prompt you to place the robot on an even and level surface, ensuring all four support pads are touching the ground and select calibration option.



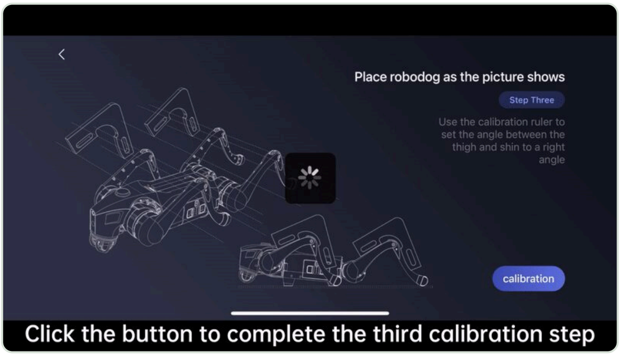
- Align the body motor of the robot in a centered triangular shape as depicted in the images below. Then, press calibrate once more.



- With the robot legs now in an upright position, utilize the calibration ruler to ensure the angle between the thigh and shin forms a right angle.



- Press calibrate again and choose save to store the calibration data. Finally, restart the robot to confirm the calibration.



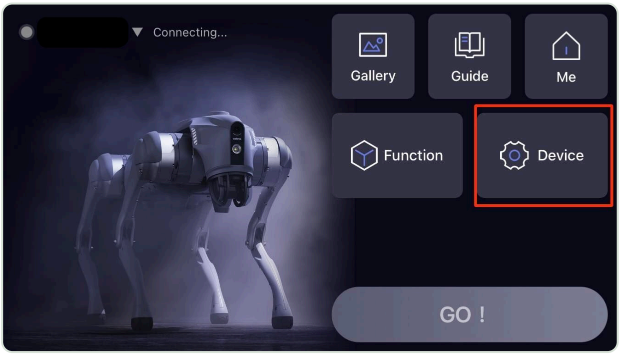
For a visual guide on leg Calibration, access the video through the provided link below.

- GO2 Leg Calibration

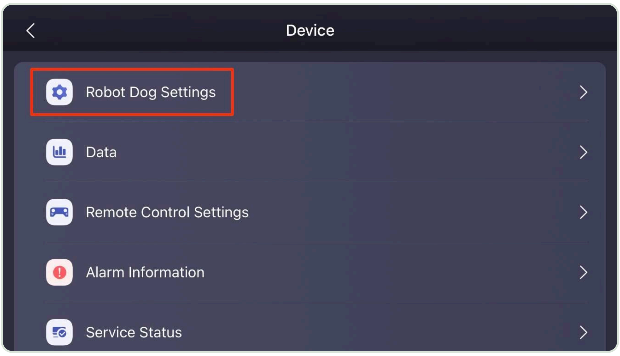
Unbinding

To Unbind the robot please follow the below mentioned steps:

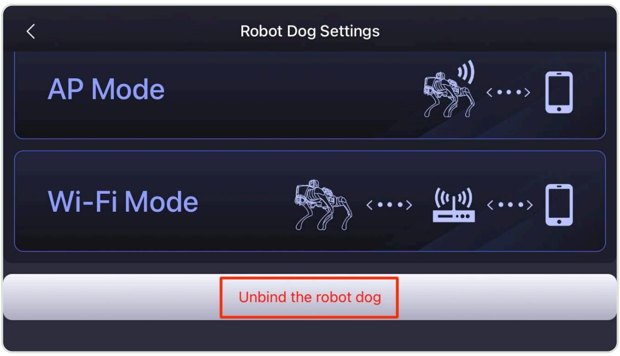
- Login to your go2 app and from the main screen select option Device



- Select Robot Dog Settings option



- Scroll dow to the end of the page and select option unbind the robot dog

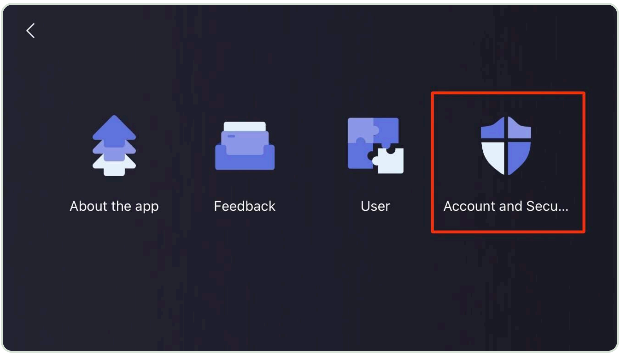


App Logout

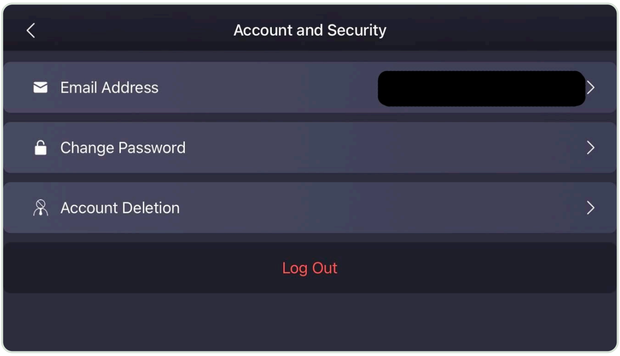
To Logout of the unitree Go2 app please follow the below mentioned steps:

- On the main screen select option Me

- Select Account and Security option



- Scroll dow to the end of the page and select option logout

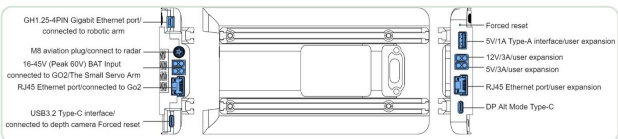


Lidar Assembly & Disassembly

To assemble or diassemble the robot LIDAR manually please follow the video link provided below:

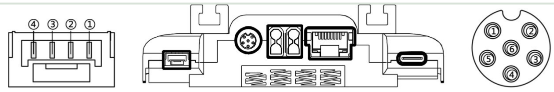
- GO2 LIDAR Assemble and Disassemble

Electrical Interface



Interface	Meaning

XT30U-F Power Interface:	16V-45V (Peak 60V) input: 2 inputs (1 for internal expansion dock power, 1 for external robotic arm) 12V output: 1 output, current output 3A (for external user) 5V output: 1 output, current output 3A (for external user)
Gigabit Ethernet Interface	Total interfaces: 2 ports (standard RJ45) Gigabit Ethernet: 1 channel to GO2, 1 channel for external user
M8 Aviation Plug Interface (Gigabit Ethernet + 12V Power)	Total interfaces: 1 port (standard European aviation plug) Gigabit Ethernet: 1 channel for radar-specific interface Power output: 1 channel 12V/2A for radar power
USB-Type A Interface	Total interfaces: 1 USB3.2 Gen2 Supported modes: USB host only Power output: 5V/1A Transfer speed: 10Gbps
USB-DP Alt Mode Type-C Full-Featured Interface (USB+DP)	Total interfaces: 1 Type-C Supported modes: USB host or DP Alternative Mode - USB: USB3.2 Gen2 - DP: DP1.4 Power output: 5V/3A Transfer speed: 10Gbps
USB-Type-C Interface	Total interfaces: 1 Type-C Interface type: USB3.2 Gen2 Power output: 5V/2A
GH1.25-4PIN Interface (Gigabit Ethernet)	Total interfaces: 1 port Gigabit Ethernet: 1 channel for external user



Gigabit Ethernet interface

Pin Number	Signal
1	Ethernet_TX+
2	Ethernet_TX-
3	Ethernet_RX+
4	Ethernet_RX-

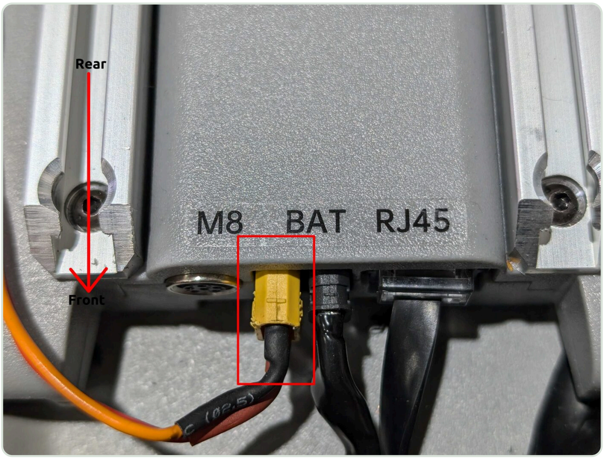
M8 aviation plug interface

Pin Number	Signal
1	Ethernet_TX+
2	Ethernet_TX-
3	Ethernet_RX+
4	Ethernet_RX-
5	+12V
6	GND

External Power-Supply

Warning: The procedure described below is only for development purposes. It turns on the robot without a battery inside from the external power supply. The power supply should atleast be able to provide 30V/20A .

1. Remove the battery from the GO2.
2. Ensure the GO2 is placed on the ground with all four paws as well as knee joints touching the ground. It is advised to place robot on some carpet or similar compressible surface to prevent damage to the joints.
3. Connect the cable to the connection ports (XT30U-F BAT Input) on the GO2's front . The GO2 requires 30V with around 20A of current. At startup it may take up to 20-30A of current. Be vigiliant and make sure the XT30 male pins are in the correct orientation when plugging in.



Steamdeck Controller

Warning: The Steamdeck usage is in experimental phase. Loss of connection or (laggy connection) may lead to the robot moving and/or not stopping immediately. Caution should be taken when using the Steamdeck for the operation of the robot!

Steamdeck - Basic Controls

1. Power button is on the top of the Steamdeck with the power symbol.
2. R2 for mouse left-click
3. L2 for mouse right-click
4. Left trackpad for mouse scrolling
5. Right trackpad for mouse movement
6. L2+R2 for middle mouse button click
7. The 3 dashes button is located on the top right of the Steamdeck next to the Y button, Hold for 3 seconds to gamepadmode and hold for 3 seconds again to go to desktop mode(Switching to desktop mode activates the cursor for the trackpad as well).

Steamdeck Controller

Steamdeck - Initialization with Robot

Before usage:

1. Ensure the Steamdeck is connected to the robot's WiFi access point.

2. Ensure the robot is powered on and operational.
3. Single tap the Joystick Launcher on the screen to activate the controller for the robot. (Do not open multiple instances of the joystick launchers, as this can cause zombie nodes).
4. Single tap the Rviz2 Launcher on the screen to visualize the robot.

Steamdeck - Robot Controls

1. L1 is the dead man's switch for slow speed.
2. R1 is the dead man's switch for high speed.
3. LJ is to move in x and y direction.
4. RJ is to move in yaw direction.
5. B is to release soft switches if the lift reaches maximum height.

Steamdeck - Rviz2 Controls

1. L2+R2 should be held when wanting to move the orbit in the xy plane.
2. R2 should be held to orbit around the orbits origin
3. Left trackpad can be moved up and down to zoom in and zoom out.

Steamdeck - Connection

1. Please follow the instructions carefully above/ or with pictures in here to start the GO2 Steamdeck .

1. Connect to the custom router of the Go2, it may take a few minutes on startup of the robot.

- The IP of the steamdeck is 192.168.123.150
- The IP of the router is 192.168.123.100
- The IP of the Unitree Go2 is 192.168.123.18

1. Once the joystick is opened hold L1 and use the left and right joystick to move the robot. You can change to different movement styles (gaits) for the go2 via the web browser as mentioned below in the console tab.
2. The web browser will automatically open and take you to https://192.168.123.18:9000 .
3. The drivers on GO2 are off by default. You can click on restart all to start all ROS2 services enabling control via the steamdeck+ros2 .
4. The console menu can be used to change between the different gaits. Additionally, more services can be added in /opt/mybotshop/src/mybotshop/go2_webserver/go2_webserver/libroscustom.py

6 ROS 2 Installation – Onboard & External PC

Onboard PC Installation

Note: This repository should already be available and built on GO2's Nividia board if it has been configured by the MYBOTSHOP team.

- Clone this repository into the GO2's Nvidia board.
- Run the bash install script.
- Then build the repository via

```
colcon build --symlink-install
```

- Once built remember to source the install/setup.bash and be sure to add the source to your workspace in the .bashrc file.

External PC Installation

Requires installation of ROS2 Foxy on Ubuntu 20.04 . This is highly recommended if you wish to view the RVIZ on your own computer without latency. All the instructions provided here are to be performed on your own computer.

First Time Setup

Important: Make sure the command source /opt/ros/foxy/setup.bash has not been activated

- Install Dependencies located in the qre_go2/src

```
./go2_install.bash
```

- Make folder called ros2_ws in home directory and clone foxy branch of qre_go2 into it.
- Then build Cyclone in the ros2_ws via.

```
colcon build --packages-select cyclonedds
```

- Source the work

```
source install/setup.bash
```

- Now source ros foxy distro via

```
source /opt/ros/foxy/setup.bash
```

- Next build the repository twice

```
colcon build --symlink-install
```

- Source the built ros workspace via:

```
source /home/$USER/ros2_ws/install/setup.bash
```

- Connect to 123 network with static ip as described in the Network section and then note down the network interface name.
- Edit and source go2_bringup/config/setup.bash with the corrected network interface e.g. enp92s0
- Then source the special configuration for the GO2

```
source /home/$USER/ros2_ws/src/go2_bringup/setup.bash
```

- Now if you are connected via LAN to the GO2 and you do ros2 topic list, the topics should be visible from the go2.

After First Time Setup

- Once the first time setup is performed, then next time simply

```
source /opt/ros/foxy/setup.bash
source / home /$USER/ros2_ws/src/go2_bringup/setup.bash
```

- You can now communicate with ros2 features in the go2. To test it out, launch

```
ros2 launch go2_viz view_robot.launch.py
```

7 Robot Interface – Network, Control, Navigation & Webserver

Robot Interface

Instructions for interfacing with the robot using Ubuntu 20.04 and ROS2 Foxy . This procedure should be followed after setting up and pairing with the GO2 Edu. Furthermore, all of the GO2's functionality should be verified via the app.

The Ethernet port at the back of the GO2 can be utilized for setting up communication via LAN.

Network Interface

For the first time, one needs to connect through a LAN cable to configure the robot's network.

To create a static connection in your PC (not the robots), in Ubuntu go to Settings → Network then click on + and create a new connection.

1. Change the connection to Manual in the IPv4 settings.
2. Set the Address IP as 192.168.123.51 and the Netmask as 24 .
3. Click save and restart your network.

After a successful connection, check the host's local IP by typing in the Host PC's terminal:

```
ifconfig
```

Next, ping the robot:

```
ping 192.168.123.18
```

Access the robot via SSH:

```
ssh -X unitree@192.168.123.18
```

The default password is:

```
123
```

IP Addresses

Robot	IP Addresses	Password
GO2 MCU	192.168.123.161	-
GO2 External	192.168.123.18	123
GO2 Router	192.168.123.100	mybotshop
GO2 Steamdeck	192.168.123.150	mybotshop

Warning: Sometimes other networks can cause disruptions when connecting to the GO2. It is best to have only your connection to the robot active and all others inactive.

Network Verification

- Firstly, connect to the GO2 as described in the Network section Network Interface
- Secondly, open several ssh sessions into the go2 via

```
ssh -X unitree@192 .168.123.18
```

Note: Verify that the robot is processing information by typing the command in an ssh session.

```
ros2 topic echo /sportmodestate
```

Note: If data is displayed then it means that the drivers are running.

Visualization

You can view the GO2's current state by typing in one of the ssh sessions:

```
ros2 launch go2_viz view_robot.launch.py
```

Tele-operation

You can teleoperate the GO2 by running the following in one of the ssh sessions:

```
ros2 run teleop_twist_keyboard teleop_twist_keyboard
```

Mode Activation

Available GO2 Modes via ROS2 Services:

- balance_stand
- recovery_stand
- stand_down
- stand_up
- stretch
- damp
- sit
- rise_sit
- front_jump
- front_pounce
- front_flip
- stop_move

Available GO2 Functionalities via ROS2 Services:

- Light level : 0-10
- Volume level : 0-10

> **Note:** Pressing the tab button on the keyboard will autocomplete the service request.

- Example of activating mode

```
ros2 service call /go2/modes go2_srvs/srv/Go2Modes request_data:\ recovery_stand
```

- Example of standing down

```
ros2 service call /go2/modes go2_srvs/srv/Go2Modes request_data:\ stand_down
```

- Example of changing light level 0-10

```
ros2 service call /go2/light go2_srvs/srv/Go2Light light_level:\ 3
```

- Example of activating mode

```
ros2 service call /go2/volume go2_srvs/srv/Go2Volume volume_level:\ 4
```

> **Note:** At times the mode does not activate correctly such as stand, in those cases use the recovery stand.

Low-Level Control

The low-level control for GO2 can be directly used via Unitree's provided example in their documentation . Running the provided qre-go2 driver with the low-level commands from Unitree examples should both work concurrently.

An example to test the low-level control is to suspend the go2.

- Verify if the go2 bringup is running via:

```
sudo service ros2 status
```

- Lie down the robot

```
ros2 service call /go2/modes go2_srvs/srv/Go2Modes request_data:\ stand_down
```

- Put motors in damp state

```
ros2 service call /go2/modes go2_srvs/srv/Go2Modes request_data:\ damp
```

- Run low-level example:

```
ros2 run go2_base go2_lowroscontrol
```

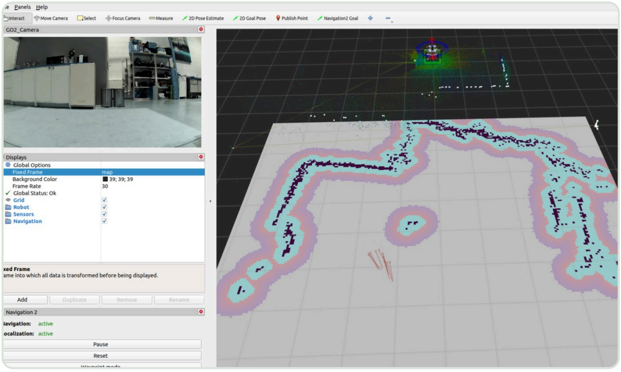
This will move the Rear Left leg of the GO2 with a force of 1N.m . An audible clicking sound can be heard if you try to move the GO2 left leg out of position.

> **Note:** The GO2 low-level mode should only be used if you are familiar with low-level control, and gait planning and wish to create a new/custom controller for the GO2. The GO2 in low-level mode can easily be damaged if used incorrectly.

Navigation

Simultaneous Localization and Mapping (SLAM)

To create a map for map-based navigation, the first step is to create a map of the environment.



Move the robot slowly when building the map

- Launch:

```
ros2 launch go2_navigation slam.launch.py
```

- You can begin mapping using the teleop at 0.2m/s with the keyboard and/or the provided Logitech controller. Once you are satisfied with your map you can export it by running the following command:

```
cd/home/unitree/ros2_ws/src/go2_navigation/maps/
```

```
ros2 run nav2_map_server map_saver_cli -f map_
```

- Rebuild so that the maps can be found (This is required if the map name is not map otherwise it will directly work):

```
colcon build --symlink-install
```

- Then source the environment:

```
source /home/$USER/ros2_ws/install/setup.bash
```

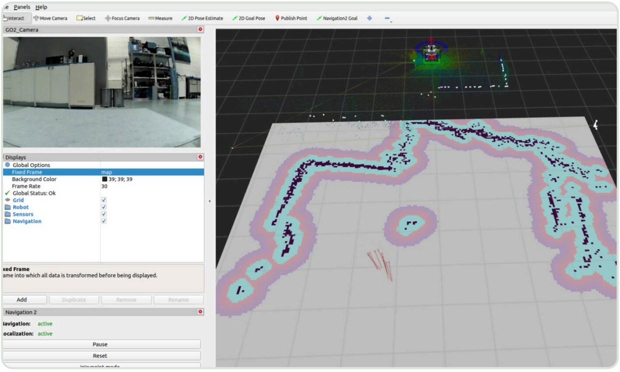
Odometric Navigation

To navigate without creating a map, you can simply launch the odom navi and give goals in RVIZ.

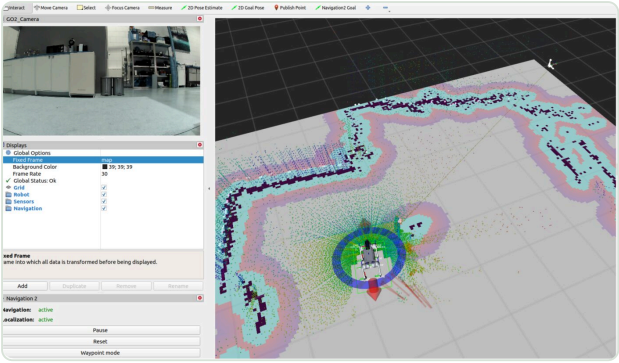
```
ros2 launch go2_navigation odom_navi.launch.py
```

Map Navigation

This will automatically take in the map created and save from the SLAM being map . If the map has been named something other, then in go2 navigation, the parameters have to be updated.



Select the 2D Pose Estimate and click and drag to where the robot corresponds in the map/



Once the robot is in the estimated location, it should self correct when the robot is moving as shown

- Ensure the map is generated and available in the ros package (i.e. after the map is saved, you have placed and performed colcon build):

```
ros2 launch go2_navigation map_navi.launch.py
```

Drivers AutoStartup

Important: The go2 bringup system launch is not required if the go2 startup has been installed. This can be verified by typing in the terminal sudo service ros2 status If it is available and green, then do not launch the go2_bringup as it is running in the background.

Launch the go2 ros driver the communicates and publishes the state of the robot joints.

```
ros2 launch go2_bringup system.launch.py
```

Startup (Optional)

The GO2 ordinarily does not have a startup job unless otherwise specified (In this case the GO2 has a startup job). The GO2 launches the go2_bringup system.launch.py . To verify if the startup job is available in GO2. Run the command:

```
sudo service ros2 status
```

Warning: If an error such as Unit ros2.service could not be found. , then it means that there is no startup installed.

1. The red marker in the service indicates that the startup job has failed.
2. Green marker indicates everything is working correctly.
3. Grey marker indicates that the service has not started yet.

In case of red or grey marker, you may restart the service via:

```
sudo service ros2 restart
```

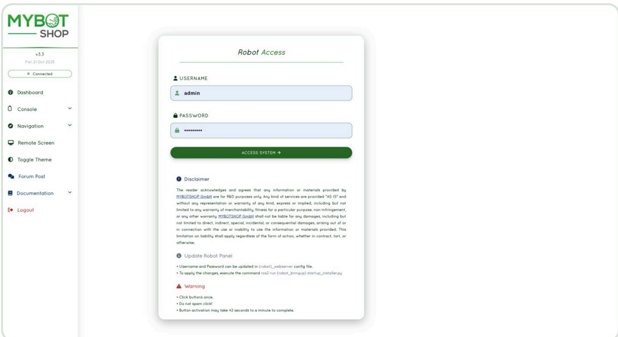
If you want to modify the upstart job or add other ROS launch files to it then it is recommended to add your changes to the main upstart file i.e. system.launch.py located in the go2_bringup package. Once done, save the file and run the following command to update startup job.

```
ros2 run go2_bringup startup_installer.py
```

Webserver

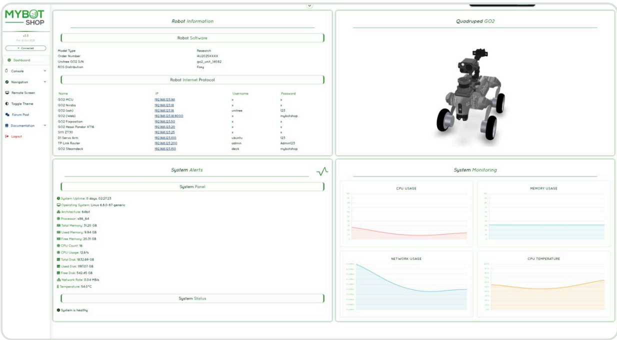
This module should come pre-installed for heavy integration projects. It should be accessible directly at http://192.168.123.18:9000/ or the WiFi ip to which the robot is connected. The GO2 webserver can be configured via the config file in go2_webserver ros2 package located in /opt/mybotshop/src/mybotshop/go2_webserver/config/robot_webserver.yaml

Login



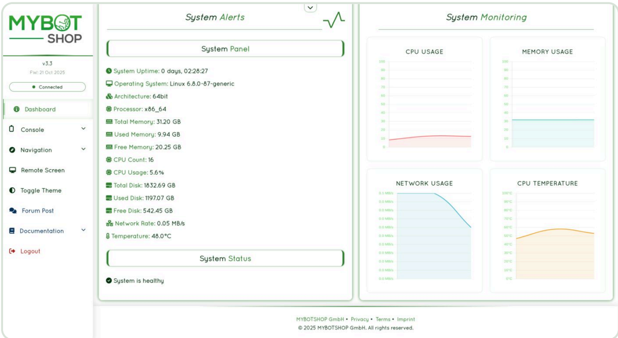
Dashboard

- Enable the GO2 ROS2 Services
- Disable the GO2 ROS2 Services
- Record System GO2 logs



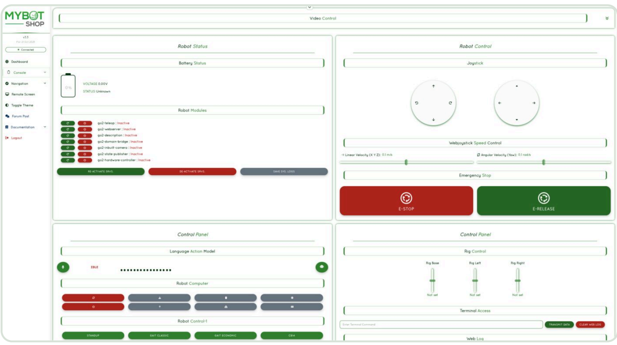
System

- View GO2 external PC system status.
- The battery only runs if the ROS2 drivers are enabled from the dashboard



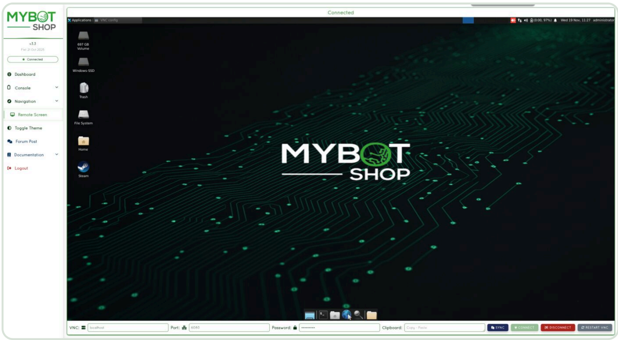
Console

- Movement of the Unitree GO2
- Switch between Gaits
- Adapatable to new ros2 services
- Record ROS2 bags
- Access GO2 external speaker if available
- The console only works if the ROS2 drivers are enabled from the dashboard



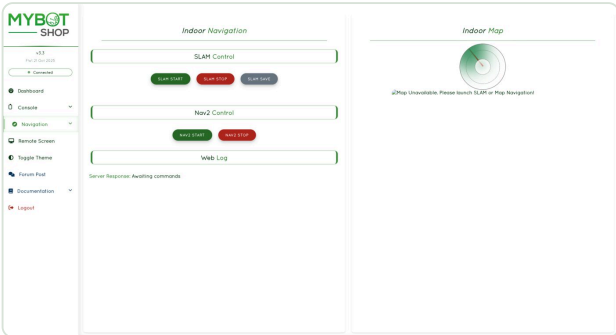
Remote Desktop

- On-board screen of the Unitree GO2's computer



Navi Indoor

- Disabled



Navi Outdoor

- Disabled

Steamdeck

1. Please follow the instructions carefully in here to start the GO2 Steamdeck .
2. Connect to the custom router of the Go2, it may take a few minutes on startup of the robot.

- The IP of the steamdeck is 192.168.123.150
- The IP of the router is 192.168.123.100
- The IP of the Unitree Go2 is 192.168.123.18

1. Once the joystick is opened hold L1 and use the left and right joystick to move the robot. You can change to different movement styles (gaits) for the go2 via the web browser as mentioned below in the console tab.
2. The web browser will automatically open and take you to https://192.168.123.18:9000 .
3. The drivers on GO2 are off by default. You can click on restart all to start all ROS2 services enabling control via the steamdeck+ros2 .
4. The console menu can be used to change between the different gaits. Additionally, more services can be added in /opt/mybotshop/src/mybotshop/go2_webserver/go2_webserver/libroscustom.py

Sensors

Intel Realsense D435i

- To launch the Realsense d435i, launch:

```
ros2 launch go2_depth_camera realsense_d435i.launch.py
```

- By default the Realsense D435i is off.
- The launch file is configured to enable continuous depth stream information from the Realsense d435i without lag. To further change parameters, simply change the configuration in the:

```
go2_depth_camera/launch/realsense_d435i.launch.py
```

Intel Realsense D405

> **Note:** If launching both realsense cameras. Launch them in separate terminals with 10-second delay. Ensure that in their launch files, their serial numbers are assigned.

- To launch the Realsense D405, launch:

```
ros2 launch go2_depth_camera realsense_d405.launch.py
```

- By default the Realsense D405 is off.
- The launch file is configured to enable continuous depth stream information from the Realsense D405 without lag. To further change parameters, simply change the configuration in the

```
go2_depth_camera/launch/realsense_d405.launch.py
```

Mid360 Lidar

- To activate the Livox Mid360 Lidar, launch:

```
ros2 launch go2_livox system.launch.py
```

- By default the Livox Mid360 is off.

SIYI ZT30

- Gimbal velocity control

```
ros2 topic pub /$GO2_NS/zt30/gimbal/control/cmd go2_interface/msg/CameraGimbalCmd '{yaw:
10, pitch: 0}'
```

- Select Camera Mode Main: Wide, Sub: Zoom (Default) HD Wide | Thermal Zoom HD Zoom | HD Wide HD Wide HD Wide | PIP (Wide Main, Zoom Sub) HD Wide | PIP (Zoom Main, Wide Sub) HD Wide | Dual-split (Wide & Zoom) Thermal Zoom | Thermal, Sub: Zoom Main: Thermal, Sub: Wide Thermal only Dual-split (Thermal + Visible) PIP (Thermal Main, Zoom Sub)
- Normal Camera

```
ros2 service call /$GO2_NS/zt30/camera_mode go2_interface/srv/CameraMode '{camera_mode: 0}'
```

- Thermal

```
ros2 service call /$GO2_NS/zt30/camera_mode go2_interface/srv/CameraMode '{camera_mode: 8}'
```

- Take photo

```
ros2 service call /$GO2_NS/zt30/trigger_photo std_srvs/srv/Trigger {}
```

- Start Recording

```
ros2 service call /$GO2_NS/zt30/trigger_video std_srvs/srv/Trigger {}
```

- Zoom in

```
ros2 service call /$GO2_NS/zt30/zoomin_control std_srvs/srv/Trigger {}
```

- Zoom Out

```
ros2 service call /$GO2_NS/zt30/zoomout_control std_srvs/srv/Trigger {}
```

Manipulation

D1

D1 Driver

```
ROS_DOMAIN_ID=10 ros2 launch d1_controller controller.launch.py
```

D1 Enable Motors

```
ROS_DOMAIN_ID=10 ros2 service call /$GO2_NS/d1/enable_load std_srvs/srv/SetBool "{data:
```

```
true}"
```

D1 Disable Motors

```
ROS_DOMAIN_ID=10 ros2 service call /$GO2_NS/d1/enable_load std_srvs/srv/SetBool "{data:
false}"
```

D1 Trajectory Control

Note: All angles are in radian except for gripper

- Joint poses intuitively

```
d1_link1_joint  : base rotation
d1_link5_joint  : gripper pitch
d1_link6_joint  : gripper roll
d1_link_l_joint : gripper grasp (Max:1.0, Min:-1.0)
```

- Home position

```
ROS_DOMAIN_ID=10 ros2 action send_goal /$GO2_NS/d1/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory "
trajectory:
joint_names: ['d1_link1_joint', 'd1_link2_joint', 'd1_link3_joint', 'd1_link4_joint',
'd1_link5_joint', 'd1_link6_joint', 'd1_link_l_joint']
points:
- positions: [0.0, -1.0, 1.047, 0.0, 0.0, 0.0, 1.0]
   time_from_start: {sec: 1, nanosec: 0}
"
```

- Stand position

```
ROS_DOMAIN_ID=10 ros2 action send_goal /$GO2_NS/d1/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory "
trajectory:
joint_names: ['d1_link1_joint', 'd1_link2_joint', 'd1_link3_joint', 'd1_link4_joint',
'd1_link5_joint', 'd1_link6_joint', 'd1_link_l_joint']
points:
- positions: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 1.0]
   time_from_start: {sec: 1, nanosec: 0}
"
```

- Custom position

```
ROS_DOMAIN_ID=10 ros2 action send_goal /$GO2_NS/d1/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory "
trajectory:
joint_names: ['d1_link1_joint', 'd1_link2_joint', 'd1_link3_joint', 'd1_link4_joint',
```

```
'd1_link5_joint', 'd1_link6_joint', 'd1_link_l_joint']
points:
- positions: [0.0, -0.4, 1.4, 0.0, 0.0, 0.0, 1.0]
   time_from_start: {sec: 1, nanosec: 0}
"
```

Open Manipulator (OM)

OM Driver

- Start Open-manipulator custom ROS driver

```
ros2 launch go2_manipulation openmanipulator.launch.py
```

This should always be running when using the provided moveit and/or joint trajectories

- Launch moveit and state publisher

```
ros2 launch go2_manipulation moveit2.launch.py
```

OM Commands

> **Note:** Be sure to update the namespace accordingly if applicable!

- Test the driver

```
ROS_DOMAIN_ID=10 ros2 action send_goal
/$GO2_NS/open_manipulator/joint_trajectory_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory -f "{
trajectory: {
   joint_names: [joint1, joint2, joint3, joint4],
   points: [
      { positions: [0.1, 0.1, 0.1, 0.1], time_from_start: { sec: 2 } },
      { positions: [-0.1, -0.1, -0.1, -0.1], time_from_start: { sec: 4 } },
      { positions: [0, 0, 0, 0], time_from_start: { sec: 6 } }
   ]
}
}"
```

- Home position (Must be done when turning off) (GO2 Prior to 06 March 2025)

```
ROS_DOMAIN_ID=10 ros2 action send_goal
/go2_unit_001/open_manipulator/joint_trajectory_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory -f "{
trajectory: {
   joint_names: [joint1, joint2, joint3, joint4],
   points: [
      { positions: [0.0, -1.4, 1.2, 1.2], time_from_start: { sec: 2 } },
   ]
```

```
}
}"
```

- Stand position

```
ROS_DOMAIN_ID=10 ros2 action send_goal
/$GO2_NS/open_manipulator/joint_trajectory_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory -f "{
trajectory: {
   joint_names: [joint1, joint2, joint3, joint4],
   points: [
      { positions: [0.0, -0.2, -0.1, -0.24], time_from_start: { sec: 1 } },
   ]
}
}"
```

- Home position (Must be done when turning off) (GO2 After 06 March 2025 - Stl will be added in Go2 Description soon!)

```
ROS_DOMAIN_ID=10 ros2 action send_goal
/$GO2_NS/open_manipulator/joint_trajectory_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory -f "{
trajectory: {
   joint_names: [joint1, joint2, joint3, joint4],
   points: [
      { positions: [0.0, -1.7, 1.4, 1.2], time_from_start: { sec: 2 } },
   ]
}
}"
```

- Close gripper (Must be done when turning off)

```
ROS_DOMAIN_ID=10 ros2 action send_goal
/$GO2_NS/open_manipulator/gripper_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory -f "{
trajectory: {
   joint_names: [gripper],
   points: [
      { positions: [1.0], time_from_start: { sec: 2 } },
   ]
}
}"
```

- Open gripper

```
ROS_DOMAIN_ID=10 ros2 action send_goal
/$GO2_NS/open_manipulator/gripper_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory -f "{
trajectory: {
   joint_names: [gripper],
   points: [
```

```
        { positions: [-1.0], time_from_start: { sec: 2 } },
    ]
}
}"
```

Open Manipulator X (MYBOTSHOP Independent)

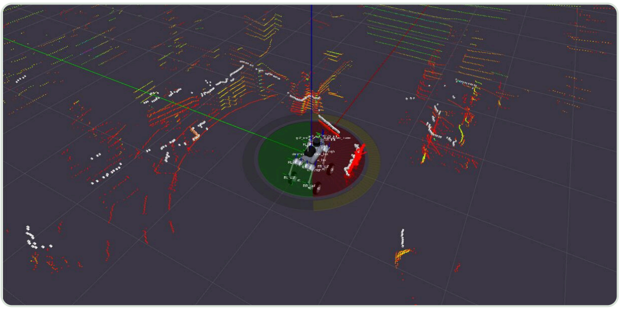
- Independent launch of open manipulator x

```
ros2 launch open_manipulator_x_controller open_manipulator_x_controller.launch.py
```

- Teleop open manipulator x

```
ros2 run open_manipulator_x_teleop teleop_keyboard
```

Lidar - Obstacle Avoidance



Lidar Based Obstacle Avoidance

A ros2 package that enables obstacle avoidance using the Mid360 Lidar. The obstacle avoidance can be enabled/disabled via a ros2 service.

The inner zone will stop the robot if an obstacle is detected, while the outer zone will slow down the robot.

```
ROS_DOMAIN_ID=10 ros2 service call /$GO2_NS/obstacle_avoidance_mode
go2_interface/srv/Go2Modes '{request_data: enable}'
```

```
ROS_DOMAIN_ID=10 ros2 service call /$GO2_NS/obstacle_avoidance_mode
go2_interface/srv/Go2Modes '{request_data: disable}'
```

Simulation

Issac Sim

> **Note:** Instructions for usage are in the humble branch qre-go2-humble

Gazebo (ROS2 Humble)

Update the robot.xacro with the gazebo extension controllers

GO2 Gazebo

```
ros2 launch go2_gazebo go2_fortress_simulation.launch.py
```

GO2W Gazebo

Update the accessories.xacro with the wheels

```
ros2 launch go2_gazebo go2_fortress_simulation.launch.py
```

GO2/GO2-W Effort Trajectory Control Example

- Robot Movement Example

```
ros2 action send_goal /joint_effort_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory -f "{
trajectory: {
   joint_names: [
      'FL_hip_joint', 'FL_thigh_joint', 'FL_calf_joint',
      'FR_hip_joint', 'FR_thigh_joint', 'FR_calf_joint',
      'RL_hip_joint', 'RL_thigh_joint', 'RL_calf_joint',
      'RR_hip_joint', 'RR_thigh_joint', 'RR_calf_joint'
   ],
   points: [
      {
      positions: [0.0, 0.9, -1.5,
                  0.0, 0.9, -1.5,
                  0.0, 0.9, -1.5,
                  0.0, 0.9, -1.5],
      time_from_start: {sec: 2, nanosec: 0}
      }
   ]
}
}"
```

GO2W Wheeled Example

- Move forward

```
ros2 topic pub /velocity_controller/commands std_msgs/msg/Float64MultiArray "{
   data: [-1.0, -1.0, -1.0, -1.0]}"
```

- Move backward

```
ros2 topic pub /velocity_controller/commands std_msgs/msg/Float64MultiArray "{
   data: [10.0, 10.0, 10.0, 10.0]}"
```

- Stop Moving

```
ros2 topic pub /velocity_controller/commands std_msgs/msg/Float64MultiArray "{
   data: [0.0, 0.0, 0.0, 0.0]}"
```

Effort Control Example (Disabled)

- Corresponding Joints

```
'FL_hip_joint', 'FL_thigh_joint', 'FL_calf_joint',
'FR_hip_joint', 'FR_thigh_joint', 'FR_calf_joint',
'RL_hip_joint', 'RL_thigh_joint', 'RL_calf_joint',
'RR_hip_joint', 'RR_thigh_joint', 'RR_calf_joint'
```

- Move joints

```
ros2 topic pub -1 /joint_effort_controller/commands std_msgs/msg/Float64MultiArray "{
   data: [
   -10.0, -30.0, 70.0,
   10.0, -30.0, 70.0,
   -10.0, -30.0, 70.0,
   10.0, -30.0, 70.0]
}"
```

- 0 effort Joints

```
ros2 topic pub -1 /joint_effort_controller/commands std_msgs/msg/Float64MultiArray "{
   data: [
   0.0, 0.0, 0.0,
   0.0, 0.0, 0.0,
   0.0, 0.0, 0.0,
   0.0, 0.0, 0.0]
}"
```

Position Control Open Manipulator X

```
ros2 action send_goal /joint_position_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory -f "{
trajectory: {
   joint_names: [joint1, joint2, joint3, joint4, gripper],
   points: [
      { positions: [0.0, -1.57, 1.57, 0.5, 0.0], time_from_start: { sec: 1 } },
   ]
}
}"
```

Cleanup Gazebo

```
ros2 run go2_gazebo kill_gz.sh
```

8 GO2 Steamdeck – Full Setup & Reference

Note: This chapter is the Steamdeck accessory's own, largely self-contained walkthrough — it repeats some network, gait and webserver material already covered in Chapter 7 for the base ROS 2 driver, but from the Steamdeck's point of view (its own IP, its own controller mapping, its own startup checks). Skip to Chapter 9 if you do not have the Steamdeck accessory.

GO2 Steamdeck

Quick Start

This guide provides instructions for interfacing with the Unitree GO2 Edu robot using Ubuntu 20.04 and ROS2 Foxy .

Go2

Note: Ensure that you have already completed the setup and pairing with your GO2 Edu. Verify all functionality using the mobile app before continuing. Initial setup instructions can be found in the GO2 Docs . This guide builds upon the information available in those docs.

Warning: The documentation below is valid only for the GO2 Edu version (tested until 08-July-2025).

Warning: Unitree is constantly updating their firmware, and at times changing naming conventions may cause breakage in some newer firmwares. If this occurs, either submit an issue or create a pull request with the correct feature updates.

Static Network Connection

For the first-time connection , you must configure the robot’s network via a LAN cable.

1. On your PC (not on the robot), open: Settings → Network → Add (+) New Connection
2. In the IPv4 settings: Change the method to Manual Set the IP Address : 192.168.123.51 Set the Netmask : 24 Save and restart your network connection
3. Confirm that the host's local IP is configured correctly: ifconfig
4. Ping the robot to verify the connection: ping 192 .168.123.18
5. Access the robot via SSH : ssh -X unitree@192.168.123.18 Default password: 123

IP Addresses

IP Address	Device	Username	Password
192.168.123.161	GO2 MCU	x	x
192.168.123.18	GO2 Auxiliary PC	unitree	123
192.168.123.18:9000	GO2 Webserver MBS	x	mybotshop

Warning: Other active networks may interfere with your connection. For best results, disconnect from all other networks and connect only to the GO2.

Network Connection

Connection via LAN:

```
ssh -X unitree@192.168.123.18
```

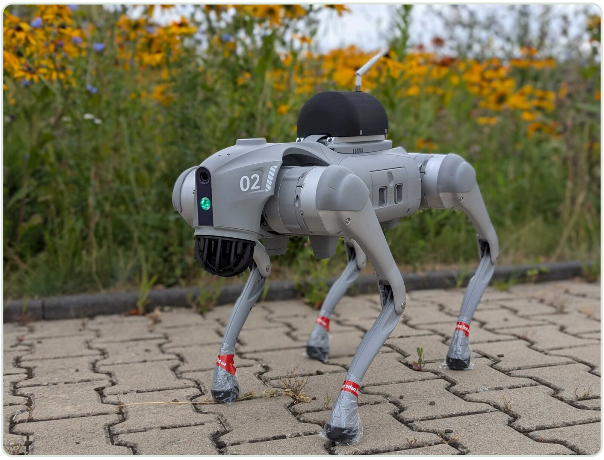
When starting up, keep in mind:

- The Go2 wrapper operates on ROS_DOMAIN_ID=10 . Launch files automatically use Domain ID 10 .
- The namespace can be changed to match your robot's ID for multi-robot systems.

Drivers Startup

- After completing installation, the ROS2 packages will run in the background (if pre-installed by the mybotshop team).

Important: To check which packages must run, see the go2_bringup startup script. Service names are mentioned there and can be referenced directly. Example: sudo service go2-webserver status ros2 launch go2_webserver webserver.launch.py Ensure duplicate services are not running simultaneously. Close unnecessary services before launching new ones.



Go2

Gaits

Available gait modes:

Normal Trot (recommended for Nav2, SLAM applications):

```
ROS_DOMAIN_ID=10 ros2 service call /go2_unit_21576/modes go2_interface/srv/Go2Modes "
{request_data: 'gait_type_classic'}"
```

Flat Terrain Mode (for flat environments):

```
ROS_DOMAIN_ID=10 ros2 service call /go2_unit_21576/modes go2_interface/srv/Go2Modes "
{request_data: 'gait_type_flat_terrain'}"
```

Economic Trot (long runs, battery conservation):

```
ROS_DOMAIN_ID=10 ros2 service call /go2_unit_21576/modes go2_interface/srv/Go2Modes "
{request_data: 'gait_type_economic'}"
```

Fast Trot:

```
ROS_DOMAIN_ID=10 ros2 service call /go2_unit_21576/modes go2_interface/srv/Go2Modes "
{request_data: 'gait_type_trot_run'}"
```

A.I Trot:

```
ROS_DOMAIN_ID=10 ros2 service call /go2_unit_21576/modes go2_interface/srv/Go2Modes "
{request_data: 'gait_type_agile'}"
```

Sports Mode

Important: The Sports Mode feature is deprecated in GO2 v2.2.1+ or with the latest firmware update.

Switch to sports mode:

```
ROS_DOMAIN_ID=10 ros2 service call /go2_unit_21576/modes go2_interface/srv/Go2Modes "
{request_data: 'sports_mode'}"
```

Visualization

Launch RViz visualization:

```
ros2 launch go2_viz view_robot.launch.py
```

GO2 Tele-operation

Run teleoperation with keyboard:

```
ROS_DOMAIN_ID=10 ros2 run teleop_twist_keyboard teleop_twist_keyboard --ros-args --remap
cmd_vel:=/go2_unit_21576/cmd_vel
```

ROS2 Modules

This section describes the ROS2 modules available for the Unitree GO2 platform, their usage, and example commands.

Webserver

The GO2 Webserver module comes pre-installed for heavy integration projects.

It can be accessed directly via:

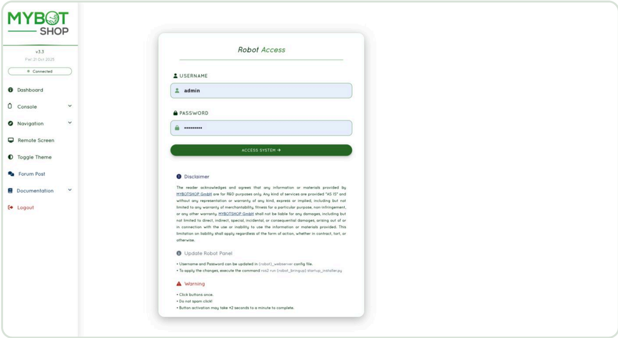
- LAN : http://192.168.123.18:9000/
- Wi-Fi : The IP assigned to the robot

The configuration file is located at:

/opt/mybotshop/src/mybotshop/go2_webserver/config/robot_webserver.yaml

Login

The login page allows authorized access to the GO2 webserver interface.

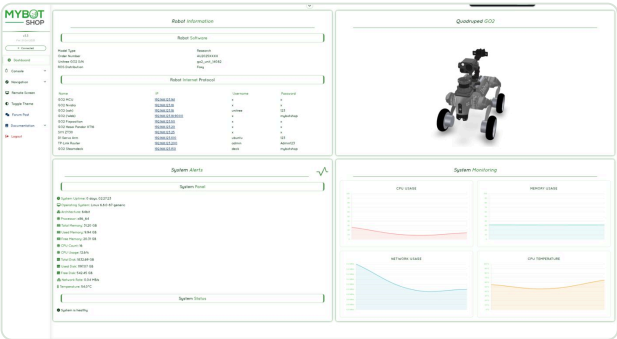


GO2 Webserver Login

Dashboard

The Dashboard provides key system controls:

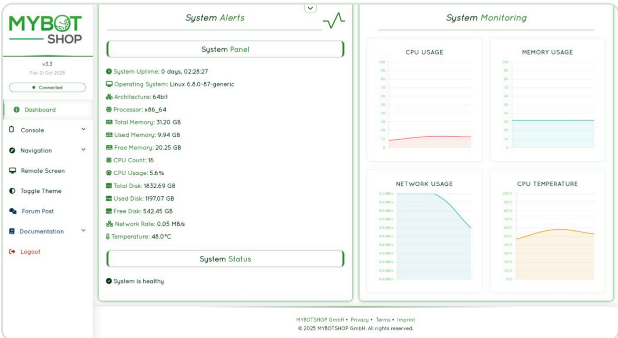
- Enable or disable the GO2 ROS2 services.
- Record system logs for debugging or analysis.



GO2 Webserver Dashboard

System

The System page displays the external PC status of the Unitree GO2. - Battery status is only active when ROS2 drivers are enabled from the Dashboard.



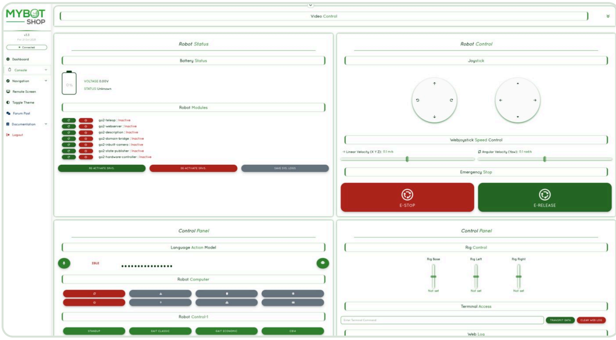
GO2 Webserver System Status

Console

The Console allows direct control over the Unitree GO2:

- Move the robot and switch between different gaits.
- Adaptable to new ROS2 services.

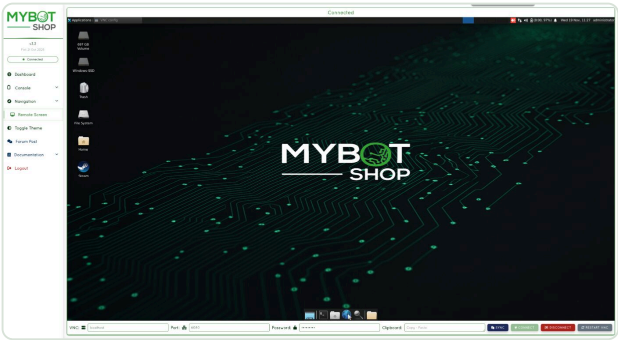
- Record ROS2 bags for later playback.
- Access the GO2 external speaker if available.



GO2 Webserver Console

Remote Desktop

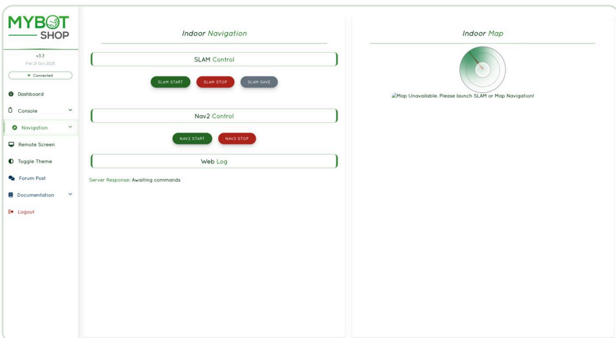
Access the on-board screen of the GO2's computer directly via the webserver.



GO2 Webserver Remote Desktop

Navi Indoor

This feature is coming soon .



GO2 Webserver Navi Indoor

Navi Outdoor

This feature is coming soon .

LLM Interface

This feature is coming soon .

Mode Activation

The following modes are available via ROS2 services:

- balance_stand
- recovery_stand
- stand_down
- stand_up
- stretch
- damp
- sit
- rise_sit
- front_jump
- front_pounce
- front_flip
- stop_move

Note: These functions can be extended in the go2_platform.cpp file to support new firmware features from the Unitree team.

Available functionalities via ROS2 services:

- Light level: 0–10
- Volume level: 0–10

Example — Activating stand-up mode :

```
ROS_DOMAIN_ID=10 ros2 service call /go2_unit_001/modes go2_interface/srv/Go2Modes "
{request_data: 'stand_up'}"
```

Wait a few seconds, then switch to balance stand:

```
ROS_DOMAIN_ID=10 ros2 service call /go2_unit_001/modes go2_interface/srv/Go2Modes "
{request_data: 'balance_stand'}"
```

Example — Standing down :

```
ROS_DOMAIN_ID=10 ros2 service call /go2_unit_001/modes go2_interface/srv/Go2Modes "
{request_data: 'stand_up'}"
```

Note: First use stand_up to remove balance mode, then stand_down . This is required for the latest firmware and GO2-W.

```
ROS_DOMAIN_ID=10 ros2 service call /go2_unit_001/modes go2_interface/srv/Go2Modes "
{request_data: 'stand_down'}"
```

Example — Changing light level (0–10):

```
ROS_DOMAIN_ID=10 ros2 service call /go2_unit_21576/light go2_interface/srv/Go2Light "
{light_level: 10}"
```

Example — Changing volume level (0–10):

```
ROS_DOMAIN_ID=10 ros2 service call /go2_unit_21576/volume go2_interface/srv/Go2Volume "
{volume_level: 4}"
```

Note: Sometimes modes (e.g., stand) may fail to activate. In such cases, use the recovery stand mode.

Realsense Depth Cameras

Warning: If running both Realsense cameras , launch them in separate terminals with a 10s delay to avoid conflicts.

Realsense D405 :

```
ros2 launch go2_depth_camera realsense_d405.launch.py
```

Realsense D435i :

```
ros2 launch go2_depth_camera realsense_d435i.launch.py
```

Note: By default, both Realsense D435i and D405 are disabled. The launch file is preconfigured to stream depth data continuously from D435i without lag. Modify parameters in: go2_depth_camera/launch/realsense_d4XX.launch.py

Go2

Lidar Modules

Livox Mid360 :

```
ros2 launch go2_lidars livox_mid360.launch.py
```

Hesai Pandar XT-16 :

```
ros2 launch go2_lidars hesai.launch.py
```

After launching, access the sensor's web UI and configure:

- Spin rate : 300
- Return mode : strongest return
- Azimuth : 30 → 330

Fixposition Setup

Install dependencies:

```
sudo apt update
sudo apt install libyaml-cpp-dev libboost-all-dev zlib1g-dev libeigen3-dev linux-libc-dev
nlohmann-json3-dev
```

Build Fixposition:

```
distrobox enter ros2_humble
rm /opt/mybotshop/src/third_party/gps/07Aug2025_fixposition_humble/COLCON_IGNORE
cp -r /opt/mybotshop/src/third_party/gps/07Aug2025_fixposition_humble
/home/unitree/fixposition_humble
cd /home/unitree/fixposition_humble
source /opt/ros/humble/setup.bash
./setup_ros_ws.sh -r 2
./create_ros_ws.sh /home/unitree/fixposition_humble_ws
cd /home/unitree/fixposition_humble_ws
colcon build --symlink-install --cmake-args -DBUILD_TESTING=OFF && source
install/setup.bash
```

Run Fixposition:

```
distrobox enter ros2_humble
source /opt/ros/humble/setup.bash
source /home/unitree/fixposition_humble_ws/install/setup.bash
ros2 launch fixposition_driver_ros2 fixposition_humble.launch.py
```

Run converter (new terminal):

```
ros2 launch odom_to_tf_ros2 odom_to_tf.launch.py
```

Open Manipulator Control

Start the custom ROS driver:

```
ros2 launch go2_manipulation openmanipulator.launch.py
```

Note: This driver must always be running when using MoveIt or joint trajectories.

Launch MoveIt & State Publisher:

```
ros2 launch go2_manipulation moveit2.launch.py
```

Open Manipulator Commands

Note: Update the namespace accordingly if required.

Test driver:

```
ROS_DOMAIN_ID=10 ros2 action send_goal
/go2_unit_001/open_manipulator/joint_trajectory_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory -f "{
  trajectory: {
    joint_names: [joint1, joint2, joint3, joint4],
    points: [
      { positions: [0.1, 0.1, 0.1, 0.1], time_from_start: { sec: 2 } },
      { positions: [-0.1, -0.1, -0.1, -0.1], time_from_start: { sec: 4 } },
      { positions: [0, 0, 0, 0], time_from_start: { sec: 6 } }
    ]
  }
}"
```

Home position (before turning off, GO2 prior to 06 Mar 2025):

```
ROS_DOMAIN_ID=10 ros2 action send_goal
/go2_unit_001/open_manipulator/joint_trajectory_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory -f "{
  trajectory: {
    joint_names: [joint1, joint2, joint3, joint4],
    points: [
      { positions: [0.0, -1.4, 1.2, 1.2], time_from_start: { sec: 2 } },
    ]
  }
}"
```

Stand position:

```
ROS_DOMAIN_ID=10 ros2 action send_goal
/go2_unit_21576/open_manipulator/joint_trajectory_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory -f "{
  trajectory: {
    joint_names: [joint1, joint2, joint3, joint4],
    points: [
      { positions: [0.0, -0.2, -0.1, -0.24], time_from_start: { sec: 1 } },
    ]
  }
}"
```

Home position (GO2 after 06 Mar 2025):

```
ROS_DOMAIN_ID=10 ros2 action send_goal
/go2_unit_21576/open_manipulator/joint_trajectory_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory -f "{
  trajectory: {
    joint_names: [joint1, joint2, joint3, joint4],
    points: [
      { positions: [0.0, -1.7, 1.4, 1.2], time_from_start: { sec: 2 } },
    ]
  }
}"
```

Close gripper (before turning off):

```
ROS_DOMAIN_ID=10 ros2 action send_goal
/go2_unit_21576/open_manipulator/gripper_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory -f "{
  trajectory: {
    joint_names: [gripper],
    points: [
      { positions: [1.0], time_from_start: { sec: 2 } },
    ]
  }
}"
```

Open gripper:

```
ROS_DOMAIN_ID=10 ros2 action send_goal
/go2_unit_21576/open_manipulator/gripper_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory -f "{
  trajectory: {
    joint_names: [gripper],
    points: [
      { positions: [-1.0], time_from_start: { sec: 2 } },
    ]
  }
}"
```

Navigation

This section covers SLAM , navigation , startup services , and simulation tools for the Unitree GO2 robot.

SLAM (Simultaneous Localization and Mapping)

Warning: The SLAM feature is not yet ported to the new system.

1. Ensure go2_bringup is running.
2. Launch SLAM: ros2 launch go2_navigation slam.launch.py
3. Begin mapping using teleoperation at 0.2 m/s (keyboard control recommended).
4. Once satisfied with your map, export it: ROS_DOMAIN_ID = 10 ros2 run nav2_map_server map_saver_cli -f /opt/mybotshop/src/mybotshop/go2_navigation/maps/custom_map
5. Rebuild so that custom maps can be located (required if the map name is not `map_`): cd /opt/mybotshop/ && colcon build --symlink-install && source /opt/mybotshop/install/setup.bash

Odometric Navigation

Note: The Nav2 Goal Tool and Nav2 Goal Plugin do not work due to namespace issues. Instead, use 2D Goal Pose in RViz or custom scripts.

Launch odometric navigation:

```
ros2 launch go2_navigation odom_navi.launch.py
```

Map Navigation

Note: The Nav2 Goal Tool and Nav2 Goal Plugin do not work due to namespace issues. Use 2D Goal Pose or scripts instead.

1. Ensure a map is generated and available in the ROS package. (After saving, place the map in the correct location and rebuild with colcon build .)
2. Launch map-based navigation: ros2 launch go2_navigation map_navi.launch.py

Startup Services

By default, the GO2 does not include a startup job unless explicitly configured. It only launches:

go2_bringup/system.launch.py

To verify if a startup service exists:

```
sudo service go2-<ros2 module name> status
```

- Error message : Unit <ros2 module name>.service could not be found. → No startup service is installed.
- Markers : Color Meaning Red Startup job has failed Green Service is running correctly Grey Service has not yet started

Restart a service if red/grey:

```
sudo service go2-<ros2 module name> restart
```

To modify or extend startup jobs, edit:

go2_bringup/startup_installer.py

Then update:

```
ros2 run go2_bringup startup_installer.py
```

GO2 Steamdeck

Go2

Follow the initialization guide:

Steamdeck Initialization Docs

1. Connect to the custom router of the GO2 (startup may take several minutes).

Default IP addresses:

Device	IP Address
Steamdeck	192.168.123.150
Router	192.168.123.100
Unitree GO2	192.168.123.18

1. Once the joystick opens, hold L1 and use the left/right joysticks to move the robot.

2. Change gaits via the web browser console (opens automatically at 192.168.123.18:9000).
3. By default, ROS2 drivers are off . Use Restart All to enable services for Steamdeck + ROS2 control.
4. Additional services can be added via: /opt/mybotshop/src/mybotshop/go2_webserver/go2_webserver/libroscustom.py

GO2 Camera

Camera Calibration

Note: Calibration is already provided. Requires a 7x6 checkerboard .

Run calibration:

```
ROS_DOMAIN_ID=10 ros2 run camera_calibration cameracalibrator --size 8x6 --square 0.025 --
camera /go2_unit_001/sensor/camera
```

GO2 Isaac Sim

Note: Instructions for usage are located in the humble branch : qre-go2-humble .

GO2 Gazebo Classic (Under Development)

Launch Gazebo simulation:

```
ros2 launch go2_gazebo_classic simulation.launch.py
```

Example — controlling GO2 in Gazebo:

```
ros2 run go2_platform go2_gazebo_demo
```

GO2 Gazebo Citadel

Important: Deprecated

Debugging & Miscellaneous

This section provides instructions for debugging, network setup, camera streaming, manipulator control, and recommended hardware enhancements .

Debugging

RQT GUI

Launch the RQT GUI for ROS2:

```
ROS_DOMAIN_ID=10 ros2 run rqt_gui rqt_gui --ros-args --remap tf:=/go2_unit_21576/tf --ros-
args --remap tf_static:=/go2_unit_21576/tf_static
```

TF Visualization

View TF tree in RQT:

```
ROS_DOMAIN_ID=10 ros2 run rqt_tf_tree rqt_tf_tree --force-discover --ros-args --remap
tf:=/go2_unit_21576/tf --ros-args --remap tf_static:=/go2_unit_21576/tf_static
```

Generate TF frames diagram:

```
ROS_DOMAIN_ID=10 ros2 run tf2_tools view_frames.py --force-discover --ros-args --remap
tf:=/go2_unit_21576/tf --ros-args --remap tf_static:=/go2_unit_21576/tf_static
```

Miscellaneous

Internet via LAN

Connect the GO2 to a router with internet access via LAN:

```
sudo ip link set eth0 down && sudo ip link set eth0 up
sudo dhclient eth0
```

Wait 10–30 seconds, then update packages:

```
sudo apt-get update
```

Update date and time automatically:

```
sudo date -s "$(wget --method=HEAD -qSO- --max-redirect=0 google.com 2>&1 | sed -n 's/^
*Date: *//p')"
```

Fix GO2 USB 2.0 → 3.0 Port (2024 Models)

1. Backup the current device tree:

```
sudo cp /boot/dtb/kernel_tegra234-p3767-0000-p3768-0000-a0.dtb kernel_tegra234-p3767-0000-
```

```
p3768-0000-a0.dtb.bak
```

1. Download the latest device tree:

Latest Equipment Tree

1. Copy the updated device tree to the boot directory:

```
sudo cp kernel_tegra234-p3767-0000-p3768-0000-a0.dtb /boot/dtb/ -r
```

1. Restart the system for changes to take effect.

Sync Host and GO2 Computers

Synchronize source files using rsync :

```
rsync -avP -t --delete -e ssh src unitree@192.168.123.18://opt/mybotshop
```

Camera Stream via Terminal (Non-ROS)

View video stream directly from terminal (replace multicast-iface with the network interface, e.g., eth0):

```
gst-launch-1.0 udpsrc address=230.1.1.1 port=1720 multicast-iface=eth0 ! application/x-rtp,
media=video, encoding-name=H264 ! rtph264depay ! h264parse ! avdec_h264 ! videoconvert !
autovideosink
```

Faster version :

```
gst-launch-1.0 udpsrc address=230.1.1.1 port=1720 multicast-iface=eth0 ! \
    application/x-rtp, media=video, encoding-name=H264 ! \
    rtph264depay ! queue max-size-buffers=1 ! h264parse ! queue ! avdec_h264 ! \
    videoconvert ! autovideosink
```

Open Manipulator X (Independent)

Launch Open Manipulator X Controller:

```
ros2 launch open_manipulator_x_controller open_manipulator_x_controller.launch.py
```

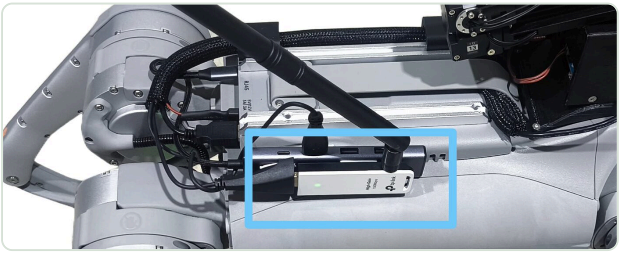
Teleoperate Open Manipulator X:

```
ros2 run open_manipulator_x_teleop teleop_keyboard
```

Enhancements

WiFi Stick

We recommend using a long-range WiFi stick , such as TL-WN722N , or any high-end long-range WiFi stick for remote operation.



Go2

Connection Hub

For connecting a screen or expanding ports, we recommend a USB-C hub such as Ugreen USB-C Hub with HDMI for testing and debugging.

Latest QRE GO2 Updates

> **Important:** The latest updates are available from the qre_go2 GitHub repository . If access is required, email support@mybotshop.de with your GitHub username and purchase ID.

9 Open Manipulator – Control & Safety Guidelines

GO2 Open Manipulator

Open Manipulator OM X ROS Control

Openmanipulator Home Pose	Openmanipulator Ready Pose

- Start Open manipulator X custom ROS driver

```
ros2 launch go2_moveit system_openmanipulator.launch.py
```

Note: This should always be running when using the provided moveit and/or joint trajectories

- Launch moveit and state publisher

```
ros2 launch go2_manipulation system.launch.py
```

- Test the driver

```
ros2 action send_goal /open_manipulator/joint_trajectory_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory -f "{
trajectory: {
    joint_names: [joint1, joint2, joint3, joint4],
    points: [
    { positions: [0.1, 0.1, 0.1, 0.1], time_from_start: { sec: 2 } },
    { positions: [-0.1, -0.1, -0.1, -0.1], time_from_start: { sec: 4 } },
    { positions: [0, 0, 0, 0], time_from_start: { sec: 6 } }
    ]
}
}"
```

- Home position (Must be done when turning off)

```
ros2 action send_goal /open_manipulator/joint_trajectory_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory -f "{
trajectory: {
    joint_names: [joint1, joint2, joint3, joint4],
    points: [
    { positions: [0.0, -1.4, 1.2, 1.2], time_from_start: { sec: 2 } },
    ]
}
}"
```

- Close gripper (Must be done when turning off)

```
ros2 action send_goal /open_manipulator/gripper_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory -f "{
trajectory: {
    joint_names: [gripper],
    points: [
    { positions: [1.0], time_from_start: { sec: 2 } },
    ]
}
}"
```

- Open gripper

```
ros2 action send_goal /open_manipulator/gripper_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory -f "{
trajectory: {
    joint_names: [gripper],
    points: [
    { positions: [-1.0], time_from_start: { sec: 2 } },
    ]
}
}"
```

Precautionary Measures

• Be careful with certain go2 actions such as wallow . This will cause the robot to flip over onto its back damaging the arm.

• Power off the GO2 correctly when power is low from the battery.

• Ensure the Open manipulator is in home position before killing its driver, otherwise it will fall.

Known Issues

• Open Manipulator not responding

• This may occur due to wrong joint positions of the dynamixels.

• The solution is to either power cycle the entire robot or only the open manipulator arm ensuring that the open manipulator's driver is turned off.

Safety Guidelines: Robotic Manipulator

When deploying robotic manipulators, it is imperative to prioritize safety procedures to mitigate risks and ensure secure operations. The following guidelines delineate key safety measures when working with robotic manipulators.

Work Area Safety

- Maintain a clean and well-organized work area. Cluttered or inadequately lit spaces can impede the proper functioning of sensors and hinder precise manipulation.
- Avoid operating robotic manipulators in hazardous environments, such as those containing corrosive substances, extreme temperatures, or sharp objects that may damage the manipulator.
- Ensure that only authorized personnel are present in the vicinity during manipulator operation to prevent interference and ensure a safe working environment.

Electrical Safety

- Ensure the manipulator's power system adheres to electrical safety standards. Regularly inspect and maintain power components to prevent malfunctions.
- Implement safeguards to protect the manipulator from adverse environmental conditions, such as exposure to moisture or extreme temperatures.
- Regularly inspect power cables and connections, promptly replacing damaged components to minimize the risk of electrical issues.

Manipulation Safety

- Implement collision detection systems to prevent unintended contact with objects, humans, or other equipment during manipulation tasks.
- Define and enforce safety zones around the manipulator's workspace to minimize the risk of unintended interactions with personnel or other objects.
- Regularly calibrate and test the manipulator's sensors and systems to ensure precise and reliable performance during manipulation tasks.

Emergency Response

- Install an emergency stop mechanism to swiftly halt manipulator operation in unforeseen circumstances or emergencies.
- Clearly mark and communicate emergency stop locations within the manipulator's operational area.
- Conduct regular emergency response drills to ensure personnel are familiar with procedures for handling unexpected situations during manipulator operation.

Data Security and Privacy

- Implement robust cybersecurity measures to safeguard the manipulator's control systems and data from unauthorized access or manipulation.
- Ensure compliance with privacy regulations when collecting, storing, or transmitting data captured by the manipulator's sensors.

Human Interaction Safety

- Integrate sensors and communication systems to detect and respond to the presence of humans in the manipulator's vicinity.
- Clearly communicate the manipulator's operational status and intentions using visual and audible signals to alert nearby individuals.
- Establish protocols for safe human-robot collaboration, particularly in shared workspaces where manipulators are in operation.

Residual Risks

Despite the implementation of safety measures, certain residual risks may persist. These include:

- Impairment of sensor functionality.
- Risk of collisions during complex manipulation tasks.
- Cybersecurity vulnerabilities.
- Unintended human interactions due to unforeseen circumstances.

Robotic manipulators are sophisticated technologies that demand correct usage to avoid accidents and ensure a secure environment. Please adhere to proper procedures diligently, prioritizing both precision and safety.

Safety Guidelines: Autonomous Robot

When deploying autonomous mobile robots, prioritizing safety procedures is essential to prevent accidents and ensure secure operations. The following guidelines outline key safety measures when working with an autonomous mobile robot:

Work Area Safety

Maintain a clean and well-lit work area. Cluttered or poorly illuminated spaces can impede the proper functioning of sensors and navigation systems.

- Avoid operating autonomous mobile robots in explosive atmospheres, such as areas with flammable liquids, gases, or dust. The robot's components may pose a risk in such environments.
- Keep bystanders and unauthorized personnel at a safe distance during robot operation to prevent interference with autonomous navigation.

Electrical Safety

- Ensure the robot's power system adheres to electrical safety standards. Regularly inspect and maintain power components to prevent malfunctions.
- Implement mechanisms to protect the robot from adverse weather conditions, such as rain or wet environments.
- Regularly inspect power cables and connections and replace damaged components promptly to minimize the risk of electrical issues.

Navigation Safety

- Implement obstacle detection and avoidance systems to prevent collisions with objects, people, or other robots.
- Define and enforce safety zones within the robot's operational area to minimize the risk of unintended interactions with personnel or other equipment.
- Regularly calibrate and test the robot's navigation sensors and systems to ensure accurate and reliable performance.

Emergency Response

- Install an emergency stop mechanism to quickly halt the robot's operation in case of unforeseen circumstances or emergencies.
- Clearly mark and communicate emergency stop locations throughout the robot's operational area.
- Conduct regular emergency response drills to ensure personnel are familiar with procedures for handling unexpected situations.

Data Security and Privacy

- Implement robust cybersecurity measures to protect the robot's control systems and data from unauthorized access or manipulation.
- Ensure compliance with privacy regulations when collecting, storing, or transmitting data captured by the robot's sensors.

Human Interaction Safety

- Integrate sensors and communication systems to detect and respond to the presence of humans in the robot's vicinity.
- Clearly communicate the robot's operational status and intentions using visual and audible signals to alert nearby individuals.
- Establish protocols for safe human-robot collaboration, especially in shared workspaces.

Residual Risks

Despite the implementation of safety measures, certain residual risks may persist. These include:

- Impairment of sensor functionality.
- Risk of collisions in crowded or dynamic environments.
- Cybersecurity vulnerabilities.
- Unintended human interactions due to unforeseen circumstances.

Autonomous Mobile Robots (AMR) are advanced technologies that require correct usage to avoid accidents and ensure a secure environment. Learn and follow the proper procedures diligently; prioritize both quality and safety.

10 Hardware & Specification Reference

Model Variants

The Go2 series is available in three distinct models — AIR , PRO , and EDU . Below is a comparative specification table for each model:

Model	AIR	PRO	EDU
Mechanical Specifications			
Dimension (Standing)	70cm x 31cm x 40cm	70cm x 31cm x 40cm	70cm x 31cm x 40cm
Dimension (Crouching)	76cm x 31cm x 20cm	76cm x 31cm x 20cm	76cm x 31cm x 20cm
Weight (with battery)	≈15kg	≈15kg	≈15kg
Material	Aluminium alloy + High-strength engineering plastic	Aluminium alloy + High-strength engineering plastic	Aluminium alloy + High-strength engineering plastic
Electronic Parameters			
Voltage	28V–33.6V	28V–33.6V	28V–33.6V
Max Working Power	≈3000W	≈3000W	≈3000W
Performance Parameters			
Payload	≈7kg (Max 10kg)	≈8kg (Max 10kg)	≈8kg (Max 12kg)
Speed	0–2.5 m/s (Max 5m/s)	0–3.5 m/s (Max 5m/s)	0–3.7 m/s (Max 5m/s)
Max Climb Drop Height	≈15cm	≈16cm	≈16cm
Max Climb Angle	30°	40°	40°
Computing Power	—	8-core High-perf CPU	8-core High-perf CPU
Knee Joint Parameters			
Max Joint Torque	—	≈45N·m	≈45N·m
Knee Motors	12 sets	12 sets	12 sets

Range of Motion	Body: -48° ~ 48° Thigh: -200° ~ 90° Shank: -156° ~ -48°	Body: -48° ~ 48° Thigh: -200° ~ 90° Shank: -156° ~ -48°	Body: -48° ~ 48° Thigh: -200° ~ 90° Shank: -156° ~ -48°
Intra-joint Circuit	√	√	√
Heat Pipe Cooler	√	√	√
Sensors			
3D LiDAR (Wide-angle)	√	√	√
Wireless Position Module	—	√	√
HD Wide-angle Camera	√	√	√
Foot-end Force Sensor	—	—	√
Features			
Basic Actions	√	√	√
Auto-scaling Strap	—	√	—
Intelligent OTA Upgrade	√	√	√
RTT 2.0 Pictures	√	√	√
Graphical Programming	√	√	√
Front Lamp (3W)	√	√	√
Wi-Fi 6 Dual Band	√	√	√
Bluetooth 5.2/4.2/2.1	√	√	√
4G Module	—	√	√
Voice Function	—	√	√
ISS 2.0 Side-follow	—	√	√
Obstacle Detection	√	√	√
Charging Pile Support	—	√	√
Secondary Development	—	—	√
Accessories			

Manual Controller	Optional	Optional	√
High Computing Module	—	—	NVIDIA Jetson Orin (Optional,40–100Tops)
Battery	8000mAh (x1)	8000mAh (x1)	15000mAh (x1)
Battery Life	1–2h	1–2h	2–4h
Charger	33.6V 3.5A	33.6V 3.5A	33.6V 9A (Fast)
Warranty	6 months	12 months	12 months

Note: Max joint torque refers to the largest joint motor. Actual torque varies by motor. Specifications may vary in different scenarios/configurations. Refer to actual product. Voice functions include offline commands, intercom, and media playback. For development, refer to the Secondary Development Manual . Warranty terms: See Product Warranty Brochure . Non-mainland versions of 4G support 43 European countries and select Asian regions.

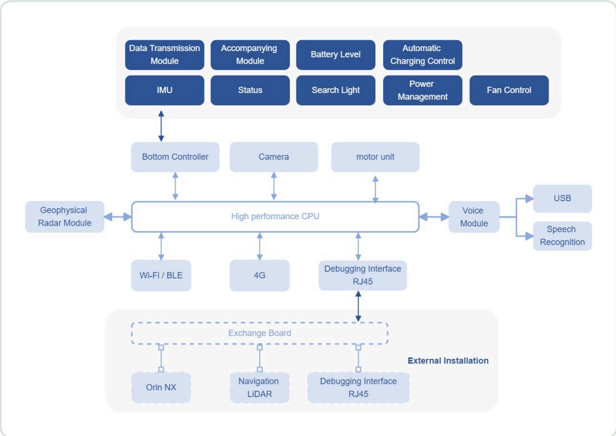
Tip: Some functions require manual operation or secondary development due to hardware/software limits. Go2 is a civilian-use robot . Avoid unsafe modifications or hazardous use. Visit the Unitree Robotics Website for latest terms, updates, and compliance guidelines.

Accessories Overview

Accessory	Description	Notes	Availability	Picture
Manual Controller	Remote handheld controller	Optional for AIR & PRO; Standard in EDU	EDU	
Standard Battery	8000mAh / 236.8Wh	≈2.5kg	Default on AIR & PRO	
Long Endurance Battery	15000mAh / 432Wh	≈2.5kg	Default on EDU	
Charger (Standard)	33.6V 3.5A	Slow charge ~2h	AIR & PRO	
Fast Charger	33.6V 9A	Fast charge ~1h15	EDU + optional upgrade	
Foot End Sensor	Mounted on foot ends	Provides advanced force detection	EDU only	
Charging Pile	Dock-type charging pile	Supports auto-charging	PRO & EDU	
Expansion Dock (Orin Nano 8GB)	High-performance module	Up to 40 TOPs computing power	Optional	
Expansion Dock (Orin NX 16GB)	Advanced high-computing module	Up to 100 TOPs computing power	Optional	

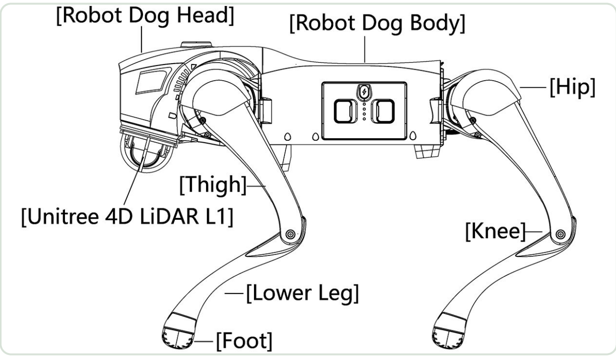
Dual Cameras	106mm × 25mm × 41mm	Weight ≈108g, 3W power	Optional	
Depth Camera (Intel D435i)	124mm × 29mm × 26mm	1280×720@30fps, 848×480@90fps	Optional	
3D Navigation Radar (MID-360)	65mm × 65mm × 60mm	Horizontal FOV 360° / Vertical 59°	Optional	
3D Navigation Radar (Hesai XT16)	Ø100–103mm × 76mm	Horizontal 360° / Vertical 30°	Optional	
Small Servo Arm (D1 / D1-550)	Robotic manipulator arm	6 DOF + claw clamp	Optional	
Protective Bracket (Small)	Support fixture	Provides hardware protection	Optional	

Hardware Architecture

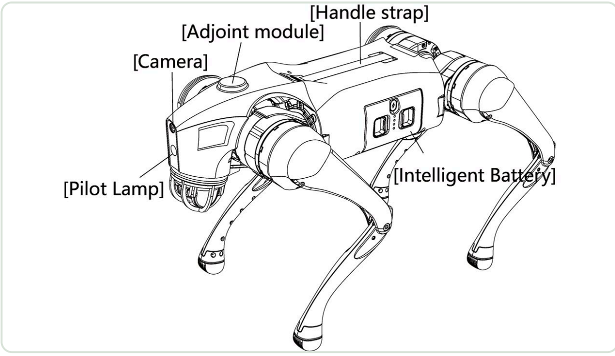


Go2 Hardware Architecture

Hardware



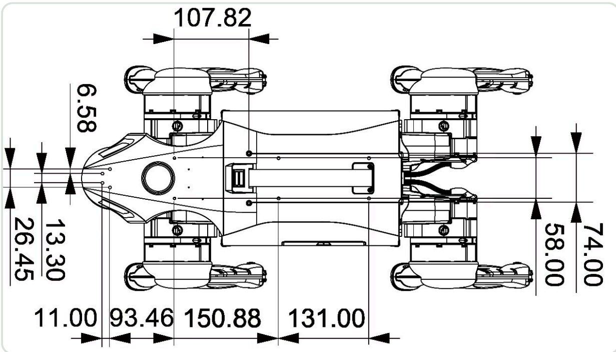
Go2 Hardware View 1



Go2 Hardware View 2

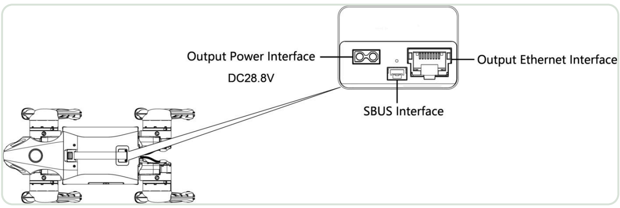
Back Load Installation Hole

> **Note:** The installation holes are designed for standard M4 screws and units are in millimeters (mm).



Go2 Installation Holes

Electrical Interface



Go2 Electrical Interface

Power Interface

DC 28.8V output Connected to Orin NX 8/16GB high computing power module BAT input.

Ethernet Interface

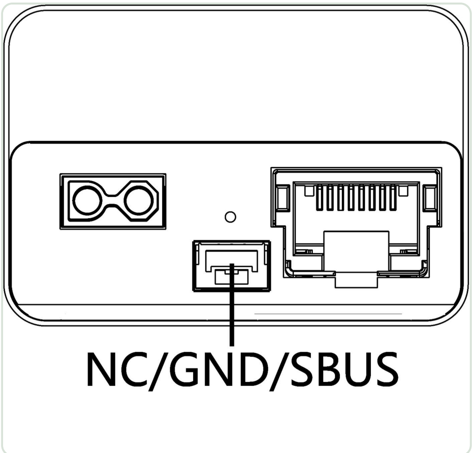
Standard RJ45 interface Connected to User PC / Orin NX 8/16GB via RJ45 Ethernet.

SBUS Interface

Used for communication connection on a universal remote control.

⚠️ This interface does not provide power output.

Interface definition (from left to right): NC / GND / SBUS



Go2 SBUS Interface

Lidar Specifications

The Unitree 4D LiDAR-L1 is equipped with Unitree's proprietary 4D radar technology (3D spatial positioning + 1D grayscale). It delivers high-speed laser ranging at up to 21,600 samples per second , providing real-time 3D environmental data. This enables advanced motion-based obstacle avoidance , ensuring reliable performance across diverse and challenging terrains.

Specifications

Parameter	Details
Model	L1
Dimensions	75 mm (W) × 75 mm (D) × 65 mm (H)
Supply Voltage	12 V DC
Laser Wavelength	905 nm
Field of View	360° × 90°

Pilot Lamp Status Indicators

Color	Flashing / State	Meaning
	Flashing	Switching on
	Permanently on	Powered on, default obstacle avoidance on
	Permanently on	Obstacle avoidance closure

	Permanently on	Range mode
	Permanently on	Accompanying mode on state
	Low flash rate	Motor & IMU calibration in progress
	Low flash rate	Low battery warning (crouch down in 10 min)
	Low flash rate	System abnormality, boot/hardware failure → Contact after-sales service
	Fast flash rate	Motor & IMU calibration failed
	Permanently on	Head indicator light

Notes

Light Priority Modes

The robot uses a priority-based light indication system:

Mode	Light Indicator
Companion Mode	Purple light (always on)
Endurance Mode	Yellow light (always on)
Obstacle Avoidance Off	Blue light (always on)

Camera Specifications

The onboard camera supports dual-stream video output at different resolutions and frame rates:

Parameter	Details
Supported Streams	720p @ 15 fps, 1080p @ 15 fps
Aperture	F2.2
Field of View	120°
Transmission	Real-time HD image transmission via app (smooth at 720p @ 15 fps in clear environments)

Joint Serial Number

A quadruped robot functions similarly to an animal, with a trunk (body) and four symmetrical legs (front and rear). The robot's four legs are grouped into front and rear pairs , each with a consistent coordinate system and joint range of motion (with slight variations between forelegs and hind legs).

Leg Numbering:

Leg	Description

0	FR – Front Right
1	FL – Front Left
2	RR – Rear Right
3	RL – Rear Left

Joint Numbering:

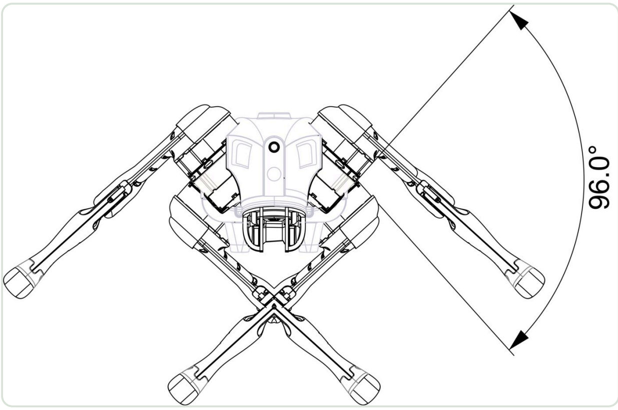
Joint | Description	
0	Hip (body joint)
1	Thigh joint
2	Calf joint

Example: FR_thigh = Right Front Leg, Thigh Joint.

Joint Limits

Joint Motion Limits:

Joint	Range of Motion
Hip (Body Joint)	-48° ~ +48°
Thigh Joint	Vertical = 0° Forward = Positive Hind Legs: -260° ~ +30° Forelegs: -200° ~ +90°
Calf Joint	-156° ~ -48°



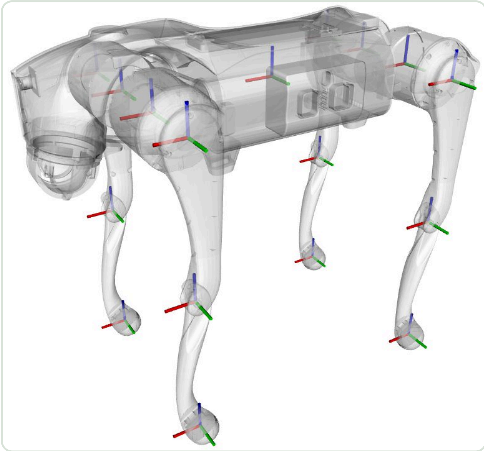
Go2 4D LiDAR

Coordinate Sys. & Joint Axes

- Hip Joint Rotation Axis: X-axis
- Thigh & Calf Joint Rotation Axis: Y-axis
- Rotation Direction: Positive direction follows the right-hand rule

When all joints are set to zero degrees :

- Red axis = X-axis
- Green axis = Y-axis
- Blue axis = Z-axis



Go2 4D LiDAR

Important: ⚠️ Note: The calf joint cannot physically reach the zero-degree position due to its mechanical limits. The illustration only represents the theoretical zero-point reference .

Robot Specifications

Dimensions

Parameter	Specification
Naked machine (standing) – L × W × H	70 cm × 31 cm × 40 cm
Naked machine (lying down) – L × W × H	76 cm × 31 cm × 20 cm
Net weight (excluding battery)	~15 kg
Degrees of Freedom	12
Maximum Speed	3.7 m/s (up to 5 m/s)

Environment

Parameter	Specification
Operating Temperature	5 C ~ 35 C (optimal in fair weather)
Slope Tolerance	±40°
Maximum Step Height	16 cm

Lighting	3 W searchlight

Battery

Specification	BT2-05	BT2-06
Weight	2.5 kg	2.5 kg
Capacity	8000 mAh / 236.8 Wh	15000 mAh / 432 Wh
Rated Voltage	DC 29.6 V	DC 28.8 V
Charging Limit Voltage	DC 33.6 V	DC 33.6 V
Runtime	1 – 2 h	3 – 5 h
Charging Method	Slow charging	Fast charging / Docking
Charging Current	3.5 A	9 A
Charging Duration	~2 h	~1 h 15 min

Internal Systems

Specification	Details
Processor	8-core processor
4G Connectivity	Built-in patch SIM card
Wi-Fi	Wi-Fi 6 Dual Band Wireless (802.11x)
Bluetooth	Versions 5.2 / 4.2 / 2.1
Storage	64 GB
Output Power Supply	DC 28.8 V (battery voltage)
Connection Port	RJ45 Ethernet port

Expansion Dock

Specification	Orin Nano 8GB	Orin NX 16GB
Power Supply Voltage	16–60 V DC	16–60 V DC
Computing Power	Up to 40 TOPS	Up to 100 TOPS
Functional Interfaces	USB 3.0 Type-A ×1 USB 3.0 Type-C ×1 USB 2.0 Type-C ×1 Gigabit Ethernet (RJ45)×2 100 Mbps Ethernet (GH1.25-4PIN) ×1 M8 aviation plug ×1	USB 3.0 Type-A ×1 USB 3.0 Type-C ×1 USB 2.0 Type-C ×1 Gigabit Ethernet (RJ45)×2 100 Mbps Ethernet (GH1.25-4PIN) ×1 M8 aviation plug ×1

3D Navigation Radar

Go2 supports two types of LiDAR (MID-360 and Hesai XT16R), both compatible with SLAM navigation.

Specification	MID-360	Hesai XT16R
Dimensions	65 × 65 × 60 mm	Ø100.0/103.0 × 76 mm
Power Supply Voltage	9–27 V DC	9–36 V DC
Laser Wavelength	905 nm	905 nm
Field of View	Horizontal 360° Vertical -7° ~ +52°	Horizontal 360° Vertical -15° ~ +15°

Depth Camera

Specification	D435i
Size	124 × 29 × 26 mm
Minimum Depth Distance	0.105 m
Depth Resolution	1280 × 720 @ 30 fps 848 × 480 @ 90 fps
Field of View	86° × 57° (±3°)

Small Servo Arm

Specification	D1	D1-550
Weight	~2.37 kg	~3.15 kg
Degrees of Freedom	6 DOF + 1 claw clamp	6 DOF + 1 claw clamp
Load Capacity	500 g (ideal)	500 g (incl. clamp)
Max Arm Span	550 mm (excl. clamp) 670 mm (incl. clamp)	550 mm (excl. clamp) 670 mm (incl. clamp)
Motor Type	Servo motor	Servo motor
Working Radius	—	550 mm
Power Demand	24 V 2.5 A (max 5 A)	24 V 10 A (15–48 V)
Power	60 W	240 W
Controller	—	Integrated (4× Cortex- A55 @ 1.8 GHz)
Communication	RJ45 (Ethernet)	RJ45 (Ethernet 100 Mbps) + Type-C (debug)
Control Method	DDS subscription	DDS subscription

Control Cycle	—	10 Hz
Torque (per joint)	—	J0: 3.3 Nm J1: 3.3 Nm J2: 1.7 Nm J3: 1.7 Nm J4: 1.7 Nm J5: 1.7 Nm J6: 1.7 Nm
Joint Motion Range	J1: ±135° J2: ±90° J3: ±90° J4: ±135° J5: ±90° J6: ±135°	J0: ±135° J1: ±90° J2: ±90° J3: ±135° J4: ±90° J5: ±135° Claw: 0–65 mm

Remote Control with Screen

Specification	Details
Display	5.5” 1080p HD LCD touchscreen
System	Android 9, 2 GB RAM, 16 GB storage
Size (antenna retracted)	189 × 138 × 41 mm
Weight	850 g
Battery	10,200 mAh, 7.4 V, 2S Li-ion
Charging Protocol	PD 20 W (fast charging)
Charging Time	~5 h (20 W fast charging)
Runtime	Up to 13 h
Functional Interfaces	HDMI (video out, industry version only) USB-A (external device / file transfer) Type-C (charging & transfer) Firmware upgrade port (GH1.25 4PIN) SIM card slot TF card slot Tripod mount (1/4” threaded hole)

Dual Cameras

Specification	Details
Size	106 × 25 × 41 mm
Weight	108 g
Power Consumption	3 W
Searchlight Power	8 W
Luminous Flux	120–140 lm @ 1 W
Searchlight Type	Transparent

Dual Cameras + Three-in-One Module

Specification	Details

Size	270 × 143 × 133 mm
Weight	690 g ± 10 g
Power Supply Voltage	16–60 V DC
Brightness	3000 lm
Light Output Angle	15°
Power	Lighting: 30 W Speaker: 30 W Red/Blue: 10 W
Effective Sound Range	500 m
Maximum Sound Range	800 m
Maximum Sound Pressure	130 dB
Audio Formats	mp3 / wav / m4a / flac / aac
Control Method	Remote app
Functions	Real-time shouting Audio recording & playback Text-to-speech Voice + lighting control Flash / constant light Brightness adjustment Red/Blue flash & mode switching

11 Getting Help & RMA

RMA / Support

How to Raise a Return Merchandise Authorization (RMA) Case

At QUADRUPED Robotics, we are committed to providing excellent support to ensure your issues are resolved promptly. If you encounter any problems with our products, please follow the steps below to raise a Return Merchandise Authorization (RMA) case:

Step 1: Open a Topic in Our Forum

1. Visit MYBOTSHOP Forum .
2. Open a new topic describing your issue in detail. Our support team and community members will attempt to assist you in resolving the problem remotely.

Step 2: Contact Support for RMA Number

If the issue cannot be resolved remotely through our forum:

1. Email us at support @ quadruped . de .
2. Provide a detailed description of the issue and include any troubleshooting steps already taken.
3. Our support team will evaluate your case and, if necessary, issue a RMA number along with a return address or a shipping label.

Important Shipping Information

Note: Please note: All goods must be sent back to the following address: QUADRUPED Robotics GmbH c / o MYBOTSHOP GmbH Willy - Messerschmitt - Strasse 12 50126 Bergheim Germany Do not ship goods without obtaining a RMA number. Parcels sent without a RMA number will be declined and returned to the sender. Do not ship goods to QUADRUPED Robotics GmbH Office in Leverkusen!

Help Resources

In case of any issues, help can be taken from any of the following forums/resources:

1. MYBOTSHOP Forum : The forum is actively monitored by support teams at MYBOTSHOP GmbH .
2. Github Issues : Issues can be reported in the repository.
3. Online QA Sites: Questions can be asked on any of the QA sites like StackOverflow and Answers ROS .

12 Getting Involved

Getting Involved

Contact support @ mybotshop . de for more information.

13 FAQ

Frequently Asked Questions

(Coming Soon)