

MYBOTSHOP

ROBOTICS

Ascento

User Manual



CONTACT

info@mybotshop.de
support@mybotshop.de
mybotshop.de

Contents

1	Attention	3
2	Disclaimer	4
3	About Ascento	5
4	Handling, Transport & Power — Quick Start	9
5	Network Setup & ROS 2 Operation	14
6	ROS 2 Package & Command Reference	20
7	Gallery	33
8	Getting Help & RMA	35

1 Attention

Attention: Read this document carefully before operating the robot. In case of doubt, consult MYBOTSHOP GmbH before putting the robot into operation. Contact information is available in Chapter 8, Getting Help & RMA.

Do not allow persons who are not familiar with the robot or these instructions to operate it. 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 power-on/power-off procedure and the Soft E-Stop (L2 + B) before switching the robot on.

Caution: Ascento balances on two wheels using a deep-learning control framework. It can traverse flat ground, gravel, grass, small ramps and slopes, but **cannot climb steps or stairs**. Changing the robot's ride height (base_height) deactivates the leg lock that maintains constant height stability, making the robot more susceptible to falling — only do this deliberately and with the surrounding area clear.

2 Disclaimer

The Ascento Research & Development platform documented here is developed and supported by *MYBOTSHOP GmbH*. This documentation covers the Ascento R&D platform only, and is distinct from *Ascento Guard*, a separate security-oriented product. Basic ROS 2 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 GmbH 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 — odometric navigation, map navigation and the Gazebo Fortress simulation — are explicitly under development at the time of writing; treat those sections as a preview rather than a finished workflow. The robot depicted in this manual may differ from the actual delivered product; specifications and features are subject to change without prior notice.

MYBOTSHOP GmbH 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 Ascento

Ascento Research



ascento_blink

Important: This documentation is specifically tailored for the Ascento Research and Development platform. For more information on the R&D version of the robot, please visit the official site for the Ascento Research and Development platform. If you are seeking security solutions, please refer to Ascento Guard , which is designed for security applications. You can find detailed information on Ascento Guard .

Attention: Please note that the robot depicted may differ from the actual delivered product. Specifications and features are subject to change without prior notice.

Introducing Ascento, a cutting-edge autonomous robot designed to excel in a wide array of environments and applications. With its advanced technological capabilities and rugged design, Ascento represents the pinnacle of robotic innovation. This versatile two-wheeled robot is engineered to navigate complex terrains with ease, thanks to its impressive features and robust build. Whether operating in challenging weather conditions or maneuvering through intricate landscapes, Ascento stands out as a reliable and efficient solution for a variety of tasks.



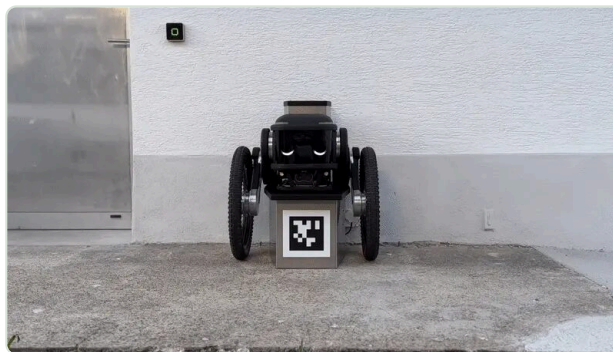
_images/ascento_chill_2.webp

The Ascento robot is a marvel of engineering, built to handle a range of operational demands. With an autonomous speed of 5 km/h, it provides a balance of swift movement and precise control. Its compact size, ranging from 50-90 cm in length, 70 cm in width, and 50 cm in height, combined with a weight of less than 30 kg, ensures agility and ease of transport. Designed to autonomously recharge, Ascento boasts a battery life that supports over 4 hours of continuous operation, with a convenient docking station for effortless recharging.

Equipped with a suite of advanced sensors, including thermal and infrared cameras, a microphone and speakers, GPS, and communication capabilities through 4G and WiFi, Ascento is well-suited for both day and night operations. This comprehensive sensor suite allows it to perform tasks with cm-level precision, making it ideal for applications requiring high accuracy.

Key Features

- **Autonomous Operation:** Ascento can autonomously navigate and perform tasks without human intervention, thanks to its sophisticated onboard sensors and intelligent algorithms.



charging_loop

- **Versatile Terrain Capability:** Its design allows it to operate on any kind of terrain, including slopes and uneven surfaces, making it adaptable to a wide range of environments.



ascento_terrain

- **Fall Recovery System:** The robot features an advanced fall recovery mechanism, enabling it to stand up from any position after a fall, ensuring continuous operation without interruption.



ascento_recovery

- **All-Weather Performance:** Ascento is engineered to function in various weather conditions, including rain, snow, and strong winds, ensuring reliable performance regardless of the environment.



water_splash

- **Obstacle Avoidance:** With its integrated sensors, Ascento can detect and avoid obstacles, allowing for smooth and safe navigation.

Category	Specification
Mechanical Specifications	
Max Speed	5 km/h
Weight	30 - 35 kg, depending on specific payload
Dimensions	55 cm (L) x 60 cm (W) x 55 - 75 cm (H)
Terrain Suitability	All types, including flat, gravel, grass, small ramps, slopes
Balancing Capabilities	Stand-up, sit-down, balance on terrain; cannot traverse steps or stairs
Electrical Specifications	
Battery Duration	Over 7 hours balancing, a bit below during continuous driving
Battery Specifications	50.4V (12S5P), 17.5Ah
Charging	Manual charging with cable (Autonomous docking optional)

Sensors & Accessories	
Sensors	Intel Realsense depth camera, Lidar, thermal camera, infrared camera, GPS, microphone, speaker
Computer	NVIDIA Jetson Orin NX accessible
Payload	Custom payload supported: Lidar, extra computer, etc.
Communication & Features	
Communication	4G, WiFi
Precision	cm-level accuracy
Features	Fall recovery, obstacle avoidance, day and night operation, 2-way communication
Weather Capability	All weather conditions

4 Handling, Transport & Power — Quick Start



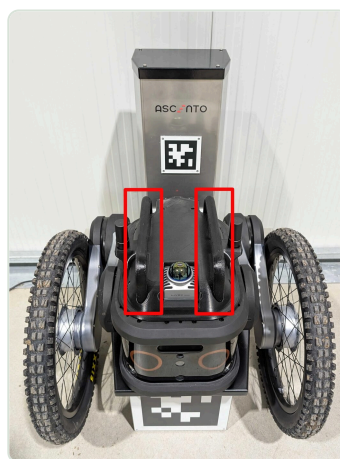
ascento_forward

Ascento Handling and Transport

Before transporting the Ascento robot, ensure proper handling by using the pre-installed handles. These handles are designed for secure grip and even weight distribution, making it easy and safe to carry the robot. Proper handling minimizes the risk of damage and ensures efficient transport.

Single-Person Transport

When transporting the Ascento robot by yourself, it is crucial to handle it properly to avoid injury or damage to the robot. The robot should be lifted using the designated handles located on the top of the main body. These handles are specifically designed to help distribute the robot's weight evenly, allowing you to carry it safely and with greater ease. Ensure that you maintain a stable posture and use both hands to securely grip the handles while lifting.

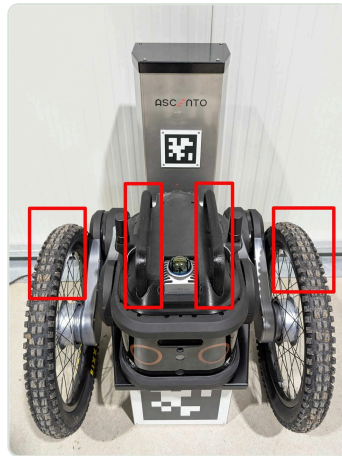


Single-person carrying Ascento

Two-Person Transport (Recommended)

For optimal safety and ease, it is recommended that two people transport the Ascento robot. One person should grip the handle on the left side of the robot's main body and simultaneously support the left wheel. Similarly, the other person should hold the handle on the right side of the

robot's main body and support the right wheel. This method ensures balanced weight distribution and enhanced stability, making it easier to carry the robot without risking accidental drops. Transporting the robot with two people not only reduces the strain on each individual but also provides better control over the robot's balance and movement.



ascento_transport_two_person

Ascento Powering ON

Hardware Setup



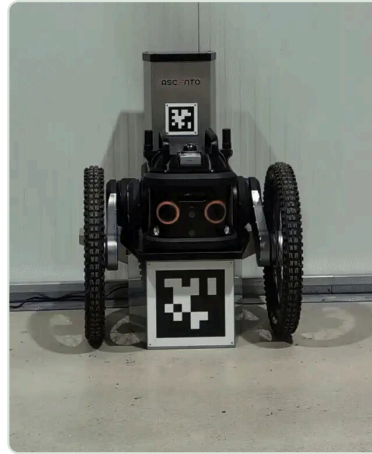
ascento_key

1. Tilt the robot upward.
2. Insert the key into the bottom slot and turn it clockwise to unlock.
3. The robot's flashlights will blink, indicating that the power is on.

Steam Deck Activation

1. Wait for 1 minute after powering up.
2. Verify that the Steam Deck is connected to Ascento's 5G WiFi .
3. Click on the joystick GUI to initiate communication between the Steam Deck and the Ascento robot.

4. Ensure the Ascento is positioned flat on the ground.
5. Perform Homing of the Ascento wheels by pressing L2 + Right . (Required only at first boot or if Ascento's eyes blink red.)
6. Once homed, press L2 + Up to make the robot stand up.



ascento_power_on

Ascento Powering Off

1. To power off, one person should remain close to Ascento .
2. Press Soft E-Stop (L2 + B) —this will cause the robot's motors to dampen, allowing it to gently fall to the ground.
3. Lock the robot with the key.

Ascento Charging

Ascento is designed to be fully weatherproof, with its battery securely integrated inside the robot's body to withstand various environmental conditions. Charging the robot is achieved through a specialized docking station, which provides a convenient and efficient solution for autonomous applications. The docking station features a built-in LED indicator that shows the current charging status of the robot. When the LED is red, it signifies that the robot is in the process of charging. Once the LED turns green, it indicates that the robot is fully charged and ready for deployment.

LED indicator showing charging in progress

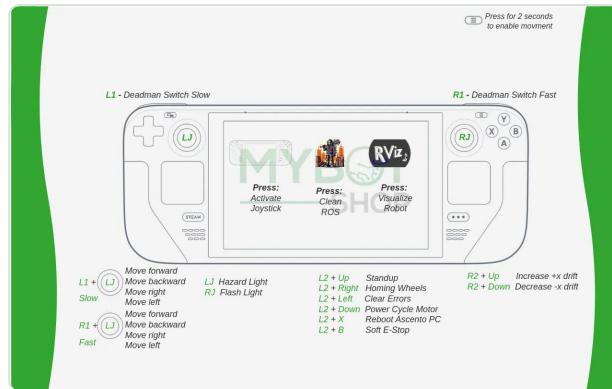
LED indicator showing charging status Complete

Ascento Steam Deck Controller

The Ascento robot can be manually controlled using the Steam Deck controller. To begin, connect the Steam Deck to the Ascento robot's Wi-Fi. Once connected, use the Activate Joystick function to establish communication between the Steam Deck and the robot.

Controller Button Functions

The Steam Deck controller is equipped with various buttons and two joysticks that allow for precise robot control. Below is a detailed description of the button functions:



ascento_steamdeck_controls

Ascento Basic Operation

Important: Always perform motor homing before initiating the Standup command to ensure proper robot alignment.

Deadman Switches

- L1 — Deadman Switch Slow : Enables slow movement when held.
- R1 — Deadman Switch Fast : Enables fast movement when held.

Movement Controls

- L1 + Left Joystick (LJ) — Move Forward; Move Backward; Move Left; Move Right

Lighting and Sensor Functions

- LJ — Hazard Light
- RJ — Flash Light
- Y — Reactivate Sensors

Robot Control Functions

- L2 + Up — Standup
- L2 + Right — Homing Wheels
- L2 + Down — Power Cycle
- L2 + B — Soft E-Stop

Drift Adjustment

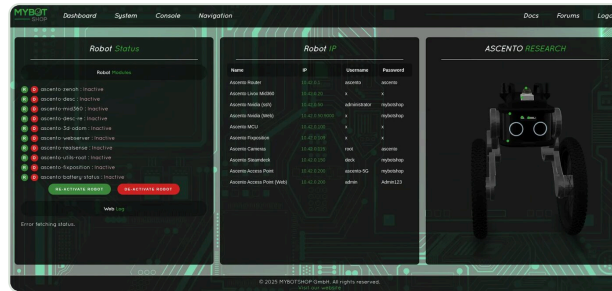
- R2 + Up — Increase +x drift
- R2 + Down — Decrease -x drift

Additional Functions

- Press (Steam Deck Icon) — Activate Joystick
- Press (RViz Icon) — Visualize Robot
- Press (ROS Icon) — Clean ROS

This setup allows for full control of the Ascento robot via the Steam Deck, providing intuitive and efficient maneuvering capabilities.

5 Network Setup & ROS 2 Operation



ascento_forward

Ascento Interface

Note: Ensure your PC is set up correctly for network communication with the robot before proceeding.

Attention: Instructions on how to interface with the robot using Ubuntu 22.04 and ROS 2 Humble .

Static Network Connection

To configure a static network connection between your PC and the robot for the first time, follow these steps:

1. Connect via LAN Cable : Use a LAN cable to connect your PC directly to the robot. The LAN port is located on the back of the robot's top cover.



ascento_network

The ethernet port can be used for setting up communication via LAN.

1. Set Up a Manual Connection : On an Ubuntu system, navigate to Settings → Network . Click on the “+” icon to create a new connection. In the IPv4 settings, change the connection mode to

Manual . Enter the following details: Address IP : 10.42.0.51 Netmask : 24 Save the configuration and restart your network.

2. Verify the Connection : Check the local IP of the host PC: ifconfig Ping the robot to ensure connectivity: ping 10.42.0.50
3. Access the Robot via SSH : Use the following command to connect: ssh -X administrator@10.42.0.50 Default password: mybotshop

Important: Disconnect from other active networks to avoid interference during this process.

IP Addresses

The table below provides a reference for the IP addresses and login credentials for various Ascento components:

Name	IP Address	Username	Password
Ascento Router	10.42.0.1	ascento	ascento
Ascento Livox Mid360	10.42.0.20	x	x
Ascento Nvidia	10.42.0.50	administrator	mybotshop
Ascento Nvidia (Web)	10.42.0.50:9000	x	mybotshop
Ascento MCU	10.42.0.100	x	x
Ascento Fixposition	10.42.0.109	x	x
Ascento Cameras	10.42.0.115	root	ascento
Ascento Steamdeck	10.42.0.150	deck	mybotshop
Ascento Access Point	10.42.0.200	ascento-5G	mybotshop
Ascento Access Point (Web)	10.42.0.200	admin	Admin123

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

Note: Use these credentials responsibly. Change default passwords where security is critical.

Ascento Fixposition

Each new unit has to configure its Fixposition. Several steps are required.

Web Interface

- Option 1 (Recommended)

Connect to the Ascento and open the webpage:

```
http://10.42.0.109/
```

- Option 2

1. Connect your PC to the Fixposition Wi-Fi network with SSID starting with fp- . The Wi-Fi password is 1234567890 .
2. Open a web browser and navigate to: 10 .0.1.1

This will take you to the Fixposition dashboard.

Software Upgrade

To update the software version of the Vision-RTK 2:

- Go to the System → Update panel in the web interface.
- Click inside the marked area to select the SWU file or drag and drop the SWU file there.

Configuration

At the top of the page, the Configuration tab allows you to set up Fixposition parameters according to your requirements.

Fusion Configuration

Available options:

- Autostart — Enables automatic launching of the Fusion engine on system bootup. (Note: Enabling this will not start the Fusion engine immediately.)
- Housing — Choose between Prototype (3D-printed) or Standard (aluminum).
- Tuning Mode — Set expected platform dynamics. For example, use “Slow robot” for typical speeds. Adjust accordingly if operating at high speeds.
- GNSS Extrinsic — Define the position of the GNSS antennas relative to the Vision-RTK 2 sensor frame in meters. These values should be accurate to the millimeter level.

GNSS Configuration

You can configure your own correction data here to receive the desired GNSS corrections.

For example, in Germany, the government provides free correction data to educational institutions, including:

- Username
- Password
- Host Name

- Port

Once acquired, enter this information in the Fixposition GNSS configuration panel at:

```
http://10.42.0.109/#gnss-config
```

Network Configuration

In the Network tab, you can configure:

- Wi-Fi Client — Set the Wi-Fi network that Fixposition should connect to (required for localization startup).
- Wi-Fi Access Point — Change the SSID and password of the Fixposition Wi-Fi network.
- Ethernet — Change the Ethernet IP address to the desired one for PC access. (Do not change this IP if it would disrupt Ascento's connection to Fixposition.)

I/O Configuration

In the Output messages table, select the TCP0 option in all sections except those that turn orange when selected.

Status

After updating and configuring Fixposition, start the Fusion engine by pressing the Start button.

- Monitor the IMU Status — it should display Converging .
- Monitor the Baseline Status — it should be Green .

Verification

For the first-time Fixposition calibration:

- The Ascento must move approximately 50 meters in a figure-8 pattern .
- This movement is necessary for calibrating the IMU and activating Fusion successfully.

Ascento Quick Start

Powering On :

1. Tilt the robot upward, insert the key into the bottom slot, and turn it clockwise to unlock.
2. The robot's flashlights will blink indicating that the power has turned on.
3. Wait for 1 minute after powering up for the PC to bootup.

Powering Off :

- Press Soft E-Stop (L2 + B) — this will cause the robot's motors to dampen, allowing it to gently fall to the ground.

- Lock the robot with the key.

Connecting via LAN :

To quickly establish a connection, use the following command:

```
ssh -X administrator@10.42.0.50
```

Warning: Ensure proper safety measures are in place when operating the robot. If issues arise during operation, refer to the specific troubleshooting commands provided below.

Disengage Estops

To disengage the emergency stops on Ascento:

- Disengage Hard Estop

```
ros2 topic pub /ascento/cmd/hard_estop_set std_msgs/msg/Bool "{data: false}"
```

- Disengage Soft Estop

```
ros2 topic pub /ascento/cmd/soft_estop_set std_msgs/msg/Bool "{data: false}"
```

Motor Homing Start

To initiate motor homing:

```
ros2 topic pub /ascento/cmd/homing_start std_msgs/msg/Empty {}
```

Warning: If homing fails, try power-cycling the robot: `ros2 service call /ascento/cmd/power_reset std_msgs/msg/Trigger {}`

Standup

To command the robot to stand up:

```
ros2 service call /ascento/cmd/standup std_srvs/srv/Trigger {}
```

Visualization

To visualize the robot in RViz:

```
ros2 launch ascento_viz view_robot.launch.py
```

Ascento Tele-operation

To control Ascento using the keyboard:

```
ros2 run teleop_twist_keyboard teleop_twist_keyboard --ros-args --remap  
cmd_vel:=/ascento/cmd/twist
```

Ascento TF Tree

To view the TF tree with remapped topics:

```
ros2 run rqt_gui rqt_gui --ros-args --remap tf:=/ascento_0116/tf --ros-args --remap  
tf_static:=/ascento_0116/tf_static
```

6 ROS 2 Package & Command Reference

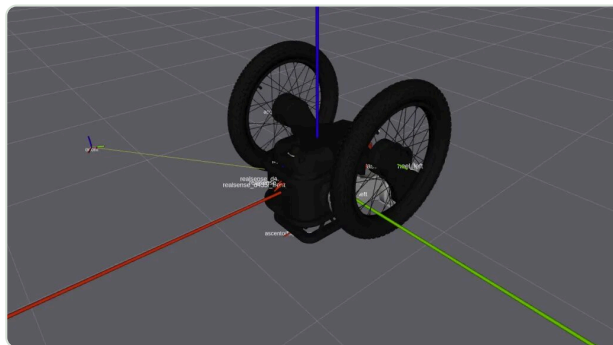
Ascento ROS2 Drivers

Important: Please contact support@mybotshop.de to gain access to the MBS Ascento repository for the latest changes and updates. When contacting support, please include your Mybotshop Order ID as well as your GitHub Username or GitHub Email ID .

All ROS2 drivers for Ascento developed by MYBOTSHOP operate under ROS DOMAIN ID=0 and utilize the Zenoh bridge for communication. The drivers are pre-configured and should already be running on the Ascento .

The Ascento ROS2 driver package includes the following:

- Joint States
- Robot Description
- Leg Control (High-level)
- Inertial Measurement Unit Publisher
- Sensor Fusion
- Twist Mux
- Steamdeck Control
- D435i Depth Camera Driver
- Livox Mid360 Lidar Driver
- 2D Scanner
- Fixposition



cascento_rviz

Ascento Startup Services

Several essential Ascento drivers and services run in the background to ensure smooth operation. Below is a list of the main services:

Service Name	Description
--------------	-------------

ascento-description	Launches robot state publisher for visualization
ascento-zenoh	Bridges robot MCU with the onboard Nvidia
ascento-mid360	Activates the mid360 lidar
ascento-3d-odom	Activates 3D Odom and localization
ascento-webserver	Activate Robot webserver
ascento-realsense	Activates the Intelrealsense D435i Camera
ascento-fixposition	Activates fixposition GPS
ascento-battery-status	Displays rviz battery marker
ascento-jointstate-publisher	Launches joint state publisher for visualization

You can query, start, stop, or restart these services using the following commands:

```
sudo service {service_name} status
sudo service {service_name} start
sudo service {service_name} stop
sudo service {service_name} restart
```

Example :

```
sudo service ascento-desc status
```

To remove a service:

```
ros2 run robot_upstart uninstall {service_name}
```

Important: If you encounter an error such as Unit ascento-{service_name}.service could not be found. , this indicates that the startup service is not installed.

Service Status Markers :

- Red Marker : Indicates the startup job has failed. Restart the service

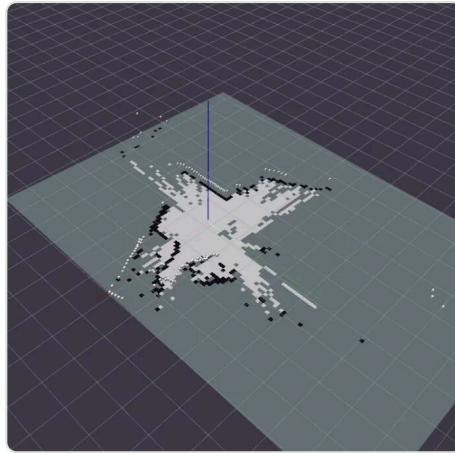
```
sudo service {service_name} restart
```

- Green Marker : Indicates the service is running correctly.
- Grey Marker : Indicates the service has not started.

To modify the upstart job or add additional ROS launch files, update the main upstart file (system.launch.py) in the ascento_bringup package. After making changes:

```
ros2 run ascento_bringup startup_installer.py
```

Ascento Simultaneous Localization and Mapping



ascento_slam

SLAM allows the robot to perform simultaneous localization and mapping. Follow these steps to perform SLAM:

1. Ensure the `ascento_bringup` service is running.
2. Launch the SLAM process: `ros2 launch ascento_nav2 slam.launch.py`
3. Begin mapping using the teleoperation keyboard at a speed of 0.2 m/s.

Exporting the Map :

Once satisfied with your map, export it using:

```
cd /opt/mybotshop/src/mybotshop/ascento_nav2/maps
ros2 run nav2_map_server map_saver_cli -f map_
```

Rebuild the Workspace :

If the map name is not `map_` , rebuild the workspace:

```
colcon build --symlink-install && source install/setup.bash
```

Finally, source the updated environment:

```
source /opt/mybotshop/install/setup.bash
```

Note: Ensure all dependencies are properly installed and the environment is sourced before running these commands.

Ascento Odometric Navigation

Note: Ascento Odometric Navigation is currently under development.

To launch odometric navigation for the robot:

```
ros2 launch ascento_nav2 odom_navi.launch.py
```

Ascento Map Navigation

Note: Ascento Map Navigation is currently under development.

To perform map-based navigation:

1. Ensure a map is generated, saved, and available in the ROS package.
2. Perform a colcon build to integrate the map into the workspace.
3. Launch the map navigation: `ros2 launch ascento_nav2 map_navi.launch.py`

Ascento Fixposition

To connect to Fixposition:

- Connect to Ascento's Wi-Fi network.
- Open a web browser and navigate to

```
http://10.42.0.109
```

Refer to the official Fixposition documentation for integration instructions and parameter tuning.

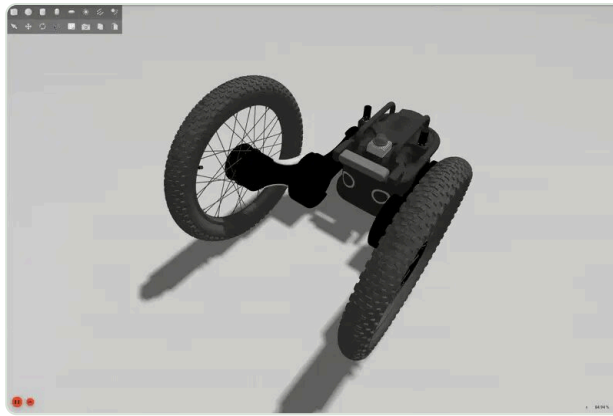
Ascento ZED2i

To launch the ZED2i camera, use the following command:

```
ros2 launch ascento_depth_cameras zed2i.launch.py
```

Note: This service is currently not operational via systemd; please use the manual command.

Ascento Gazebo Fortress

*ascento_gazebo*

Note: Ascento Gazebo Simulation is currently under development.

The Fortress Gazebo simulation currently does not support the high-level Ascento interface. It is intended for use with low-level control, training, and testing.

For simulation modifications, refer to the GZ ROS2 Control documentation.

Ascento Simulation

To start the Ascento Fortress simulation:

```
ros2 launch ascento_gazebo fortress_simulation.launch.py
```

Cleanup After Simulation

To clean up the simulation environment after use:

```
ros2 run ascento_gazebo kill_gz.sh
```

Effort Trajectory Control Example

To send a joint trajectory goal:

```
ros2 action send_goal /joint_trajectory_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory -f "{
  trajectory: {
    joint_names: [
      'ascento/hip_left',  'ascento/hip_right',
      'ascento/knee_left', 'ascento/knee_right'
    ],
    points: [
      {
        positions: [-2.0, 2.0, 0.0, 6.28],
        time_from_start: {sec: 2, nanosec: 0}
      }
    ]
  }
}"
```

```
]
}
```

Velocity Control Example

To publish velocity commands to the joint velocity controller:

```
ros2 topic pub -1 /joint_velocity_controller/commands std_msgs/msg/Float64MultiArray "
{data: [0.5, -0.5]}"
```

Ascento Intel RealSense D435i

Color Image	Depth Image
-------------	-------------

Note: The D435i is launched by default through the Ascento bringup . Do not re-run the following command unless the ascento-realsense service has been stopped. `ros2 launch ascento_depth_camera realsense_d435i.launch.py`

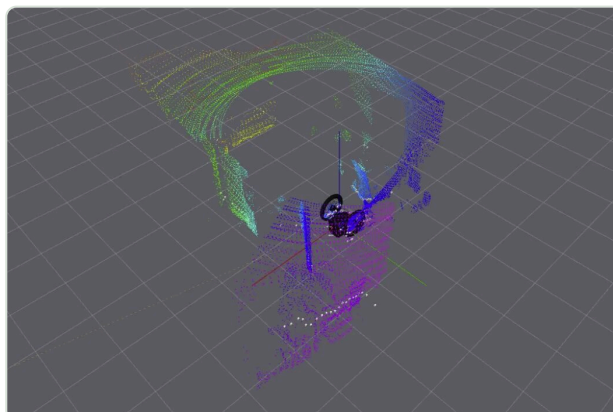
The launch file is pre-configured to enable continuous depth stream from the RealSense D435i without lag.

To modify parameters, edit the configuration file:

```
ascento_depth_camera/launch/realsense_d435i.launch.py
```

Ascento Mid360 LiDAR

Note: The Mid360 LiDAR is launched by default through the Ascento bringup . Do not re-run the following command unless the ascento-mid360 service has been stopped. `ros2 launch ascento_livox system.launch.py`



ascento_lidar

Ascento Data Recording

Important: The ROS 2 bag should be started before launching any drivers; otherwise, it will not capture all the data.

To store data for 30 minutes , limited to 1 GB , use the following command:

```
ros2 bag record -a -b 1000000000
```

Ascento Development

For custom controls and interfaces , refer to the Ascento SteamDeck ROS 2 package as a reference for development.

Ascento Diagnostics

The Ascento Diagnostics system uses different LED colors to indicate various states.

The robot's diagnostics are represented by the following color codes:

Color	State
RED	The robot is in an error state.
WHITE	The robot is operational.
ORANGE	The robot is in a soft e-stopped state.
BLUE	The hardware manager is in none of the above states.

Note: These diagnostics provide immediate visual feedback on the robot's current status.

Ascento Status Commands

This section provides commands to monitor the various statuses of the Ascento Research Robot, including IMU, joint states, battery state, diagnostics, and other key metrics.

The following commands can be used to echo the respective statuses of the robot:

Status	Command
IMU	<code>ros2 topic echo /ascento/imu</code>
Joint States	<code>ros2 topic echo /ascento/joint_states</code>
Battery State	<code>ros2 topic echo /ascento/battery_state</code>
Diagnostic Status	<code>ros2 topic echo /ascento/diagnostic_status</code>

Robot Description	ros2 topic echo /robot_description
Robot Speed	ros2 topic echo /ascento/estimator/speed_bias
Current Height	ros2 topic echo /ascento/estimator/base_height
Motor Homing Status	ros2 topic echo /ascento/homing_status
Head Light Status	ros2 topic echo /ascento/headlight_state
Hazard Light Status	ros2 topic echo /ascento/hazardlight_state
Eyes Status	ros2 topic echo /ascento/eyes_state

Note: These commands provide real-time status updates for the respective components of the robot. Use them as needed for diagnostics or monitoring during operation.

Ascento Service Commands

This section provides service commands to control and configure various features of the Ascento Research Robot.

Important: Always perform motor homing before initiating the Standup command to ensure proper robot alignment.

Below is a list of service commands and their purposes:

Command	Description
Standup	Makes the robot stand up. ros2 service call /ascento/cmd/standup std_srvs/srv/Trigger {}
Head Light	Turns the headlight on or off. ros2 service call /ascento/cmd/headlight_set std_srvs/srv/SetBool "{data: true}"
Hazard Light	Turns the hazard light on or off. ros2 service call /ascento/cmd/hazardlight_set std_srvs/srv/SetBool "{data: true}"
Eyes State	Toggles the eyes state on or off. ros2 service call /ascento/cmd/eyes_set std_srvs/srv/SetBool "{data: true}"
Powercycle Robot	Restarts the robot power system. ros2 service call /ascento/cmd/power_reset std_srvs/srv/Trigger {}
Speed Bias Increase	Increases the robot's speed bias by 0.002 per request. ros2 service call /ascento/cmd/speed_bias_increase std_srvs/srv/Trigger {}
Speed Bias Decrease	Decreases the robot's speed bias by 0.002 per request. ros2 service call /ascento/cmd/speed_bias_decrease std_srvs/srv/Trigger {}
MCU Reboot	Reboots the robot's MCU. ros2 service call /ascento/cmd/reboot_computer std_srvs/srv/Trigger {}
Motor Homing Start	Starts the motor homing process. ros2 service call /ascento/cmd/homing_start std_srvs/srv/Trigger {}

Motor Homing Cancel	Cancels the motor homing process. ros2 service call /ascento/cmd/homing_cancel std_srvs/srv/Trigger {}
---------------------	--

Note: These commands are critical for controlling the robot's operational features. Ensure the robot is powered on and connected before executing them.

Ascento Publish Commands

This section provides commands for publishing messages to control and configure various functionalities of the Ascento Research Robot.

Warning: The Ascento operates using a deep-learning control framework . Changing the height deactivates the leg lock , which maintains constant height stability. This makes the robot more susceptible to falling .

Below is a list of publish commands and their purposes:

Command	Description
Standard Height	Sets the robot's base height. Minimum height: 0.3m. Maximum height: 0.8m. ros2 topic pub /ascento/cmd/base_height_meter_set std_msgs/msg/Float64 "{data: 0.3}"
Command Velocity	Publishes velocity commands to the robot. ros2 topic pub /ascento/cmd/twist geometry_msgs/msg/Twist
Error Clear	Clears error states of the robot. ros2 topic pub /ascento/cmd/clear_error std_msgs/msg/Empty {}
Hard E-Stop	Engages or disengages the hard emergency stop. Engage ros2 topic pub /ascento/cmd/hard_estop_set std_msgs/msg/Bool "{data: true}" Dis-engage ros2 topic pub /ascento/cmd/hard_estop_set std_msgs/msg/Bool "{data: false}"
Soft E-Stop	Engages or disengages the soft emergency stop. Engage ros2 topic pub /ascento/cmd/soft_estop_set std_msgs/msg/Bool "{data: true}" Dis-engage ros2 topic pub /ascento/cmd/soft_estop_set std_msgs/msg/Bool "{data: false}"

Attention: Ensure that commands for E-Stop and velocity are executed responsibly to avoid unintended robot behavior.

Note: These commands allow for precise control of the robot during operation. Confirm the robot is connected and powered on before issuing commands.

Ascento Installation

This section provides instructions for installing and configuring the Ascento Research Robot software.

Important: This repository should already be available and built on Ascento's Nvidia board if it has been configured by the MYBOTSHOP Team.

Environment Setup

1. Update `.bashrc` Add the following lines to your `.bashrc` file for automatic ROS2 activation: #
Ascento ROS2 Activation source /opt/ros/humble/setup.bash source
/opt/mybotshop/src/mybotshop/ascento_bringup/hooks/setup.bash
2. Create Workspace Directory Create the required directory and set ownership: sudo mkdir
/opt/mybotshop sudo chown -R administrator:administrator /opt/mybotshop

Repository Setup

1. Clone Repository and Run Installers Clone the repository to your local machine, copy it to the Ascento PC, and run the installer scripts: cd /opt/mybotshop/src/mybotshop/ sudo chmod +x ascento_dependencies.sh ./ascento_dependencies.sh cd
/opt/mybotshop/src/mybotshop/ascento_webserver sudo chmod +x webserver_installer.sh
./webserver_installer.sh
2. Build the ROS2 Workspace Build the workspace and source the environment: cd
/opt/mybotshop colcon build --symlink-install source install/setup.bash

Zenoh Configuration

1. Verify Zenoh Version The required version is 0.11 . Check the installed version: zenoh-bridge-ros2dds --version
2. Remove Incorrect Version If the version is incorrect, remove it: sudo apt-get remove zenoh-bridge-ros2dds -y
3. Install Correct Version Navigate to the appropriate directory and install based on hardware (e.g., Nvidia → arch64): cd
/opt/mybotshop/src/mybotshop/ascento_hardware/launch/zenoh/arch64/ sudo dpkg -i zenoh-bridge-ros2dds_0.11.0_arm64.deb
4. Verify Again zenoh-bridge-ros2dds --version
5. Set Ascento ID Modify the startup configuration in ascento_bringup/hooks/ to reflect the correct Ascento number.

Final Setup

1. Run Startup Installers ros2 run ascento_bringup startup_installer.py
2. Update Hostname Set the PC name for the robot: sudo hostnamectl set-hostname ASCENTO-0000
3. Update Web Interface Manually update the robot software version in the index.html file served by the web interface.

Ascento External Computer Installation

This section provides steps for setting up the Ascento software environment on an external computer.

1. Clone and Build the Workspace

Clone the mbs_ascento package and build it:

```
colcon build --symlink-install
source install/setup.bash
```

1. Configure Zenoh

Zenoh version 0.11 is required. Follow these steps:

- Check Zenoh Version

```
zenoh-bridge-ros2dds --version
```

- Remove Incorrect Version

```
sudo apt-get remove zenoh-bridge-ros2dds -y
```

- Install Correct Version (Assuming AMD architecture for external computers)

```
cd ros2_ws/src/mybotshop/ascento_hardware/launch/zenoh/amd/
sudo dpkg -i zenoh-bridge-ros2dds_0.11.0_amd64.deb
```

- Verify Installation

```
zenoh-bridge-ros2dds --version
```

1. Environment Configuration (Critical)

Important: This step is critical . Ensure the following environment variables are set in every terminal session. Adjust the CYCLONEDDS_CONFIG_PATH according to your system's setup.

```
export ASCENTO_NS="ascento_0116"

# Example configuration path:
CYCLONEDDS_CONFIG_PATH=/home/administrator/projects/bipeds/ascento/mbs_ascento/src/mybotshop/ascento_bringup/config/cyclonedds.xml
export RMW_IMPLEMENTATION=rmw_cyclonedds_cpp
export CYCLONEDDS_URI="${CYCLONEDDS_CONFIG_PATH}"
```

1. Launch Zenoh and RViz2

- Launch Zenoh

```
ros2 launch ascento_hardware zenoh.launch.py
```

- Launch RViz2

```
ros2 launch ascento_viz view_robot.launch.py
```

Access to Ascento PC

This section explains how to access the Ascento PC remotely using VNC.

VNC Ascento Startup

The VNC server is typically started by default. If not, use the following commands to manage it manually:

- Start VNC Server

```
vncserver :1 -geometry 1920x1080 -depth 24 -localhost no
```

- Check Running VNC Sessions

```
vncserver -list
```

- Kill VNC Session

```
vncserver -kill :1 && rm -rf /tmp/.X1-lock /tmp/.X11-unix/X1
```

VNC Remote Startup

On your remote PC, follow these steps to access the Ascento interface:

1. Install TigerVNC Viewer `sudo apt install tigervnc-viewer`
2. Allow VNC Port Through Firewall `sudo ufw allow 5901/tcp`
3. Connect via VNC `vncviewer 10.42.0.50:1`

Miscellaneous

This section includes additional commands and processes to enhance the usage and maintenance of the Ascento Research Robot.

Time Synchronization

1. Sync System Time Use the following command to sync the system time with Google's server:
`sudo date -s "$( wget --method = HEAD -qSO- --max-redirect = 0 google.com 2 > & 1 | sed -n 's/^*Date: */p' ) "`
2. Sync Host Computer and Ascento Computer Synchronize the files between the host computer and the Ascento computer: `rsync -avP -t --delete -e ssh src administrator@10.42.0.50://home/administrator/ros2_ws`
3. Sync Host Computer and Steamdeck Synchronize the files between the host computer and the Steamdeck: `rsync -avP -t --delete -e ssh src deck@10.42.0.150://home/deck/ros2_ws`

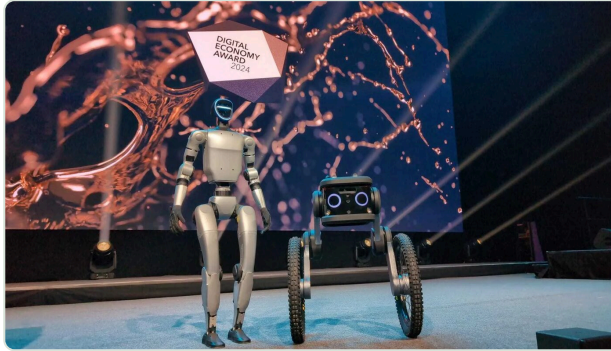
URDF Conversion

Convert a Xacro file to a URDF file using the following command:

```
ros2 run xacro xacro
/home/administrator/ros2_ws/src/mybotshop/ascento_description/xacro/robot.xacro >
/home/administrator/ros2_ws/src/mybotshop/ascento_description/xacro/ascento.urdf
```

7 Gallery

A few reference photos of the Ascento platform, from the manufacturer's own gallery page. The full media collection is linked at the end of this chapter.



Ascento on stage at the Digital Economy Award 2024



Ascento, resting posture



Ascento, patrolling outdoors



Ascento, headlight detail

Explore More

The full image and media collection is hosted on HiDrive: [Ascento Media Gallery](#). For any questions, reach out at support@mybotshop.de.

8 Getting Help & RMA

RMA / Support

How to Raise a Return Merchandise Authorization (RMA) Case

At MYBOTSHOP GmbH, 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



ascento_support

If the issue cannot be resolved remotely through our forum:

1. Email us at support@mybotshop.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: 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.

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 .