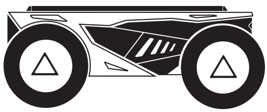


# HUNTER 2.0

AgileX Robotics Team

User Manual V.2.1.0 2021.03

This chapter contains important safety information, before the robot is powered on for he first time, any individual or organization must read and understand this information before using the device. If you have any questions about use, please contact us at support@agilex.ai. Please follow and implement all assembly instructions and guidelines in the chapters of this manual, which is very important. Particular attention should be paid to the text related to the warning signs.

# ⚠ Safety Information

The information in this manual does not include the design, installation and operation of a complete robot application, nor does it include all peripheral equipment that may affect the safety of the complete system. The design and use of the complete system need to comply with the safety requirements established in the standards and regulations of the country where the robot is installed. HUNTER 2.0 integrators and end customers have the responsibility to ensure compliance with the applicable laws and regulations of relevant countries, and to ensure that there are no major dangers in the complete robot application. This includes but is not limited to the following:

## 1.Effectiveness and responsibility

- Make a risk assessment of the complete robot system.
- Connect the additional safety equipment of other machinery defined by the risk assessment together.
- Confirm that the design and installation of the entire robot system's peripheral equipment, including software and hardware systems, are correct.
- This robot does not have a complete autonomous mobile robot, including but not limited to automatic anti-collision, anti-falling, biological approach warning and other related safety functions. Related functions require integrators and end customers to follow relevant regulations and feasible laws and regulations for safety assessment , To ensure that the developed robot does not have any major hazards and safety hazards in actual applications.
- Collect all the documents in the technical file: including risk assessment and this manual.

## 2.Environmental Considerations

- For the first-time use,please read this manual carefully to understand the basic operating content and operating specification.
- No passengers.
- For remote control operation, select a relatively open area to use HUNTER 2.0, because HUNTER 2.0 is not equipped with any automatic obstacle avoidance sensor. Please keep a safe distance of more than 2 meters when operating HUNTER 2.0.
- Use HUNTER 2.0 always between -10°C~45°C ambient temperature.
- If HUNTER 2.0 is not configured with separate custom IP protection, its water and dust protection will be IP22 ONLY.

## 3.Pre-work Checklist

- Make sure each device has sufficient power.
- Make sure HUNTER 2.0 does not have any obvious defects.
- Check if the remote controller battery has sufficient power.
- When using, make sure the emergency stop switch has been released.

## 4.Operation

- In remote control operation, make sure the area around is relatively spacious.
- Make sure to operate the Hunter 2.0 within the visual range.
- The maximum load of HUNTER 2.0 is 150KG. When in use, ensure that the payload does not exceed 150KG.
- When installing an external extension on HUNTER 2.0, confirm the position of the center of mass of the extension and make sure it is at the center of rotation.
- Please charge in time when the device is low battery alarm.
- When HUNTER 2.0 has a defect, please immediately stop using it to avoid secondary damage.
- When HUNTER 2.0 has had a defect, please contact the relevant technical to deal with it, do not handle the defect by yourself.
- Always use HUNTER 2.0 in the environment with the protection level requires for the equipment.
- It is forbidden to push the chassis in the parking status, and the parking can be manually released in an emergency status.
- When charging, make sure the temperature is above 0°C.

## 5.Maintenance

- Regularly check the pressure of the tire, and keep the tire pressure is maintained at 0.8bar.
- If the tire is severely worn or burst, please replace it in time.
- If the battery plan not to use for a long time, it need to be charged periodically in every 2 to 3 months.

# CONTENTS

# 1 HUNTER 2.0 Introduction

HUNTER 2.0 is designed as a programmable UGV( UNMANNED GROUND CHASSIS) upon Ackermann model, of which the chassis is based on Ackermann steering. Therefore, it has similar characteristics to cars but has more significant advantages on Portland cement and asphalt roads over them. Compared to the four-wheel differential chassis, HUNTER 2.0 chassis has a higher load carrying capacity and can reach higher movement speed with less wear of structure and tires for long-term operation. Although HUNTER 2.0 is not designed as suitable for all kinds of terrains, it is equipped with a rocker arm suspension which can pass common obstacles such as speed bumps, etc. Additional components such as stereo camera, laser radar, GPS, IMU and robotic manipulator can be optionally installed on HUNTER 2.0 for advanced navigation and computer vision applications. HUNTER 2.0 is frequently used for autonomous driving education and research, indoor and outdoor security patrolling, environment sensing, general logistics and transportation.

## 1.1 Component list

| Name | Quantity |
|---|---|
| HUNTER 2.0 Robot body | x1 |
| Battery charger(AC 220V) | x1 |
| Aviation plug(male, 4-pin) | x2 |
| Remote control transmitter(optional) | x1 |
| USB to CAN communication module | x1 |
| USB to R232 cable | x1 |

## 1.2 Tech specifications

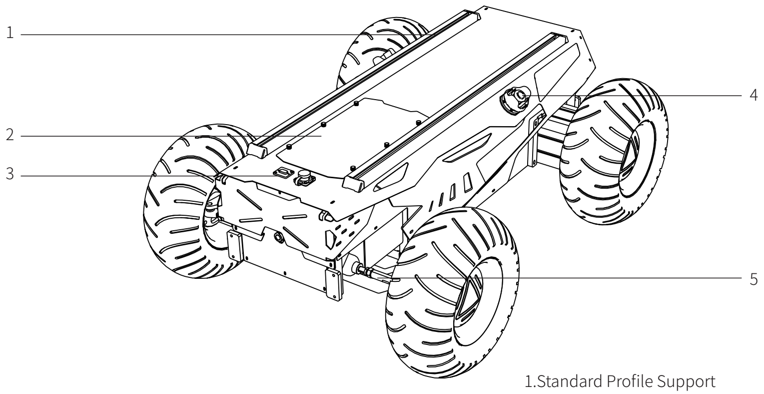
| Parameter Types | Items | Values |
|---|---|---|
| Mechanical specifications | L × W × H (mm) | 980 × 745 × 380 |
| | Wheelbase (mm) | 650 |
| | Front/rear wheel base (mm) | 605 |
| | Weight of chassis body (kg) | 65/70 |
| | Battery | Lithium battery 24V 30Ah/60Ah |
| | Power drive motor | DC brushless 2 ×400W |
| | Steering drive motor | DC brushless 200W |
| | Reduction gearbox | 1:40 |
| | Drive system form | Power off electromagnetic band type brake |
| | Steering | Front wheel Ackermann |
| | Maximum steering angle | 33° |
| | Steering accuracy | 0.5° |
| Motion | No- load MAX speed (m/s) | 1.5 |
| | Minimum turning radius (mm) | 1.6 |
| | Maximum climbing capacity | 10° |
| | Minimum ground clearance (mm) | 105 (Angle 30°) |
| Control | Control mode | Remote control<br>Control command mode |
| | RC transmitter | 2.4G/extreme distance 1km |
| | System interface | CAN |

## 1.3Requirement for development

FS RC transmitter is provided (optional) in the factory setting pf HUNTER 2.0, which allows users to control the chassis of robot to move and turn; CAN and RS232 interfaces on HUNTER 2.0 can be used for user’ s customization.

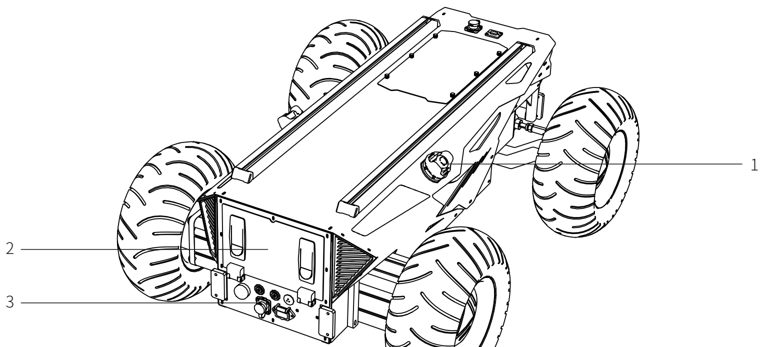
# 2 The Basics

The section provides a brief introduction to the HUNTER 2.0 mobile robot platform, as shown in Figure 2.1 and 2.2




1.Standard Profile Support
2.Top Cabin Panel
3.Top Extend Interface
4.Emergency Stop Switch
5.Ackermann Front Wheel Steering

Figure 2.1 Front View




1.Emergency Stop Switch
2.Battery Change Panel
3.Rear Panel

Figure 2.2 Rear View

Designed as a complete intelligent module, HUNTER 2.0 combines inflatable rubber wheels with independent suspension as its power module, which, along with powerful DC brushless servo motor, enables the chassis of HUNTER 2.0 robot to flexibly move on different ground surfaces with high passing ability and ground adaptability. An emergency stop switch is mounted at the rear end of chassis body, which can shut down power of the robot immediately when the robot behaves abnormally. Water-proof connectors for DC power and communication interfaces are provided both on top and at the rear of the robot, which not only allow flexible connection between the robot and external components but also ensures necessary protection to the internal of the robot even under severe operating conditions.

## 2.1 Status indication

Users can identify the status of chassis body through the voltmeter, the beeper and lights mounted on HUNTER 2.0. For details, please refer to Table 2.1.

| Status | Description |
|---|---|
| Voltage | The current battery voltage can be read from the voltmeter on the rear electrical panel. |
| Replace battery | When the battery voltage is lower than 24.5V (if the BMS is connected, the SOC is judged to be lower than 15%) , the chassis body will give a beep-beep-beep sound as a warning. When the battery voltage is detected as lower than 24V(if the BMS is connected, the SOC is judged to be lower than 10%), HUNTER 2.0 will actively cut off the power supply to external extensions and drive to prevent the battery from being damaged. In this case, the chassis will not enable movement control and accept external command control. |
| Robot powered on | Rear lights are switched on. |

Table 2.1 Descriptions of Chassis Status

## 2.2 Instructions on electrical interfaces

### 2.2.1 Top electrical interface

HUNTER 2.0 provides two 4-pin aviation connectors and one DB9 (RS232) connector. (The current version can be used for upgrade of firmware but do not support for command).The position of the top aviation connector and DB9 interface is shown in Figure 2.3.

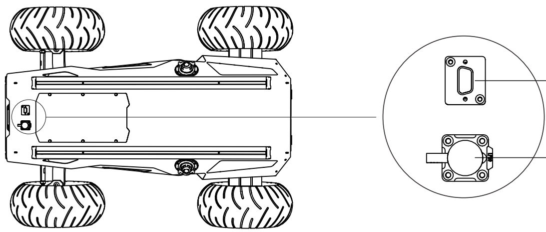
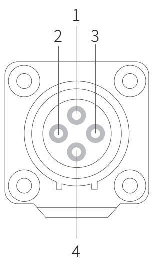



Figure 2.3 Schematic Diagram of HUNTER 2.0 Electrical Interface on Top

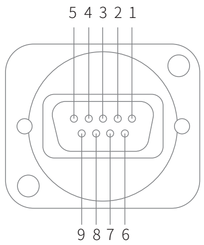
HUNTER 2.0 has each aviation extension interface respectively on top and at rear end which is configured with a set of power supply and a set of CAN communication interface. These interfaces can be used to supply power to extended devices and establish communication. The specific definitions of pins are shown in Figure 2.4.
It should be noted that, the extended power supply here is internally controlled, which means the power supply will be actively cut off once the battery voltage drops below the pre-specified threshold voltage. Therefore, users need to notice that HUNTER 2.0 platform will send a low voltage alarm before the threshold voltage is reached and also pay attention to battery recharging during use.




| Pin No. | Pin Type | Function and Definition | Remarks |
|---|---|---|---|
| 1 | Power | VCC | Power positive, voltage range 21-26.8V, singly maximum current 10 A, total current is less than 15A |
| 2 | Power | GND | Power negative |
| 3 | CAN | CAN_H | CAN bus high |
| 4 | CAN | CAN_L | CAN bus low |

Figure 2.3 Description of Top Aviation Interface Pins

Top DB9 expansion interface pin definition.




| Pin No. | Definition |
|---|---|
| 2 | RS232-RX |
| 3 | RS232-TX |
| 5 | GND |

Figure 2.5 Description of Top DB9 interface

## 2.2.2 Rear electrical interface

The extension interface at rear end is shown in Figure 2.6, where Q1 is the power display; Q2 is the switch of manual parking release; Q3 is the power switch; Q4 is the buzzer; Q5 is CAN and 24V power extension interface; Q6 is charging interface.

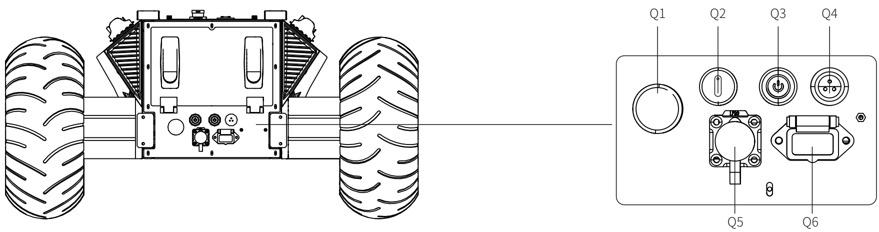
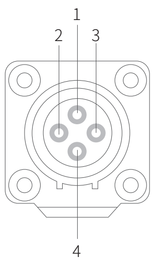



Figure 2.6 Rear View

Specific definitions for pins of Q5 are shown in Figure 2.7. The rear panel provides the same CAN communication interface and 24V power interface with the top one (two of them are internally inter-connected). The pin definitions are given in Figure 2.7.




| Pin No. | Pin Type | Function and Definition | Remarks |
|---|---|---|---|
| 1 | Power | VCC | Power positive, voltage range 21-26.8V, single maximum current 10 A, total current is less than 15A |
| 2 | Power | GND | Power negative |
| 3 | CAN | CAN_H | CAN bus high |
| 4 | CAN | CAN_L | CAN bus low |

Figure 2.7 Description of Rear Aviation Interface Pins

## 2.3 Remote control instructions

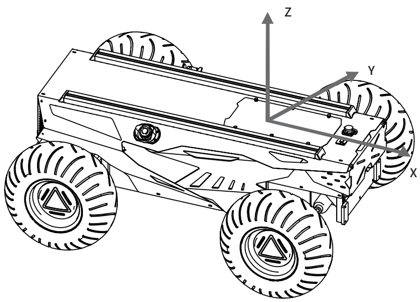
FS RC transmitter is provided (optional) for HUNTER 2.0. In this product, we use the left-hand-throttle design. Refer to Figure 2.8 for its definition and function.

The function of the button is defined as: SWC and SWD are temporarily disabled. Among which SWA is the parking switch lever, turn to the top to release the parking mode, turn to bottom is the parking mode. (The remote control can be performed normally after the parking mode is released.) SWB for control mode selection, top position for command control and the middle position for remote control mode; S1 is the throttle button, which controls the forward and backward of HUNTER 2.0; S2 controls the steering of the front wheels, and POWER is the power button, press and hold it to turn on.


Figure 2.8 Schematic Diagram of Buttons on FS RC transmitter

## 2.4 Instructions on control demands and movements

A reference coordinate system can be defined and fixed on the chassis body as shown in Figure 2.9 in accordance with ISO 8855.




As shown in Figure 2.9, the chassis body of HUNTER 2.0 is in parallel with X axis of the established reference coordinate system. In the remote control mode, push the remote control stick S1 forward to move in the positive X direction, push S1 backward to move in the negative X direction. When S1 is pushed to the maximum value, the movement speed in the positive X direction is the maximum, When pushed S1 to the minimum, the movement speed in the negative direction of the X direction is the maximum; the remote control stick S2 controls the steering of the front wheels of the car body, push S2 to the left, and the chassis turns to the left, pushing it to the maximum, and the steering angle is the largest, S2 Push to the right, the car will turn to the right, and push it to the maximum, at this time the right steering angle is the largest. In the control command mode, the positive value of the linear velocity means movement in the positive direction of the X axis, and the negative value of the linear velocity means movement in the negative direction of the X axis; the steering angle is the steering angle of the inner wheel.

Figure 2.9 Schematic Diagram of Reference Coordinate System for Chassis Body

# 3 Getting Started

This section introduces the basic operation and development of the HUNTER 2.0 platform using the CAN bus interface.

## 3.1 Use and operation

The basic operating procedure of startup is shown as follows:

### Check

- Check the condition of HUNTER 2.0. Check whether there are significant anomalies; if so, please contact the after-sale service personal for support;
- Check the state of emergency-stop switches. Make sure both emergency stop buttons are released;
- For first-time use, check whether Q3 (drive power supply switch) on the rear panel has been pressed down; if so, please release it, and then the drive will be powered off.

### Startup

- Press Q3 button, and normally, the voltmeter will display correct battery voltage and front and rear lights will be both switched on;
- Check the battery voltage, the normally voltage range is 24~26.8V, if there is continuous “beep-beep-beep...” sound from beeper,it means the battery voltage is low, please charge the battery.

### Shutdown

- Press the button Q3 to cut off the power supply.

### Emergency stop

- Press down emergency push button on the top of HUNTER 2.0 chassis body.

## Basic operating procedure of remote control

- After the chassis of HUNTER 2.0 mobile robot is started correctly, turn on the RC transmitter and push the SWB to the remote control mode, then, HUNTER 2.0 platform movement can be controlled by the RC transmitter.

## Parking

- The parking brake adopts a power off electromagnetic band type brake to realize the parking function, so when the chassis is running, the parking function must be turned off before moving;
- In the remote control mode, SWA is the parking function switch. You can control movement after turning the stick to the top to turn off the parking function. Turn the stick to the bottom to turn on the parking mode, if the chassis speed is not 0 at this time, it will automatically decelerate to 0 and turn on the parking function.
- In the command mode, the parking mode is the default when the power is turned on. At this time, there is no response to the speed command, and the parking release command needs to be sent before the speed command can be sent for control. If you need to park after the motion control is completed, just send a parking command.
- When the emergency stop is triggered, the parking will automatically start. At this time, released the emergency stop, no matter where the remote control SWA is located, it needs to be unlocked again for normal movement. If the power fails to be re-powered after a power failure (such as low battery voltage), you can use the Q2 knob switch to manually unlock the parking to facilitate moving the chassis or trailer. It should be noted that the manual (tail knob switch Q2) unlocking the parking has the highest priority, which will invalidate the parking in the program, so it is limited to special circumstances. Please close it in time after use.
- Ramp parking, when HUNTER 2.0 is on the slope, if the speed is 0, HUNTER 2.0 will check the current automatically. When it reaches a certain value and continues for a period of time, HUNTER 2.0 will turn on the ramp parking function automatically. After receiving the motion command again, the ramp parking will relieve automatically and start to running.

# 3.2 Charging and battery replacement

HUNTER 2.0 is equipped with a 10A charger by default to meet customers' recharging demand.

### The detailed operating procedure of charging is shown as follows:

- Make sure the electricity of HUNTER 2.0 chassis is powered off. Before charging, please make sure the power switch in the rear control console is turned off;
- Insert the charger plug into Q6 charging interface on the rear control panel;
- Connect the charger to power supply and turn on the switch in the charger. Then, the robot enters the charging state.

Note: For now, the battery needs about 4 hours to be fully recharged from 21V, and the voltage of fully-recharged battery is about 26.8V.

## Battery replacement

- Turn off the power switch of the HUNTER 2.0 chassis.
- Press the button lock on the battery replacement panel and open the battery panel.
- Unplug the currently connected battery interface, respectively (XT60 power connector) (BMS connector) lock
- Take out the battery, pay attention to this process, the battery is forbidden to hit and collide.
- Install the battery that will be used, and then plug the connector back.
- Turn off the power to replace panel, press the lock.

# 3.3 Battery replacement

HUNTER 2.0 provides CAN and RS232 (not open to current version) interfaces for user customization. Users can select one of these interfaces to conduct command control over the chassis body.

## 3.3.1 CAN message protocol

HUNTER 2.0 adopts CAN2.0B communication standard which has a communication baud rate of 500K and Motorola message format. Though external CAN bus interface, the moving linear speed and the rotational angle of chassis can be controlled; HUNTER 2.0 will feedback on the current movement status information and its chassis status information in real time.
The system status feedback command includes the feedback information about current status of chassis body, control mode status, battery voltage and system failure. The description is given in Table 3.1.

Table 3.1 Feedback Frame of HUNTER 2.0 Chassis System Status

| Command Name | System Status Feedback Frame | | | |
|---|---|---|---|---|
| Sending node | Receiving node | ID | Cycle (ms) | Receive-timeout (ms) |
| Steer-by-wire chassis | Decision-making control unit | 0x211 | 100ms | None |
| Data length | 0×08 | | | |
| Position | Function | Data type | Description | |
| byte [0] | Current status of chassis body | unsigned int8 | 0×00 System in normal condition<br>0×01 Emergency stop mode (not enabled)<br>0×02 System exception | |
| byte [1] | Mode control | unsigned int8 | 0×00 Standby mode<br>0×01 CAN command control mode<br>0×02 Remote control mode | |
| byte [2]<br>byte [3] | Battery voltage higher 8 bits<br>Battery voltage lower 8 bits | unsigned int16 | Actual voltage × 10 (with an accuracy of 0.1V) | |
| byte [4]<br>byte [5] | Failure information higher 8 bits<br>Failure information lower 8 bits | unsigned int16 | See notes for details<br>[Description of Failure Information] | |
| byte [6] | Parking(brake) state | unsigned int8 | 0×00 Brake unlocked state 0×01 Brake locked state | |
| byte [7] | Parity bit (checksum) | unsigned int8 | 00-255 counting loops, which will be added once every command sent | |

| Description of Failure Information | | |
|---|---|---|
| Byte | Bit | Meaning |
| byte [4] | bit [0] | Status error of drive (0: No failure 1: Failure) |
| | bit [1] | Upper communication connection status (0: No failure 1: Failure) |
| | bit [2] | Reserved, default 0 |
| | bit [3] | Reserved, default 0 |
| | bit [4] | Reserved, default 0 |
| | bit [5] | Reserved, default 0 |
| | bit [6] | Reserved, default 0 |
| | bit [7] | Reserved, default 0 |
| byte [5] | bit [0] | Battery under-voltage failure (0: No failure 1: Failure) |
| | bit [1] | Reserved, default 0 |
| | bit [2] | Remote control loss protection (0: No failure 1: Failure) |
| | bit [3] | Steering motor drive communication failure (0: No failure 1: Failure) |
| | bit [4] | Rear right motor drive communication failure (0: No failure 1: Failure) |
| | bit [5] | Rear left motor drive communication failure (0: No failure 1: Failure) |
| | bit [6] | Reserved, default 0 |
| | bit [7] | Front wheel steering encoder disconnection failure (0: No failure 1: Failure) |

The command of movement control feedback frame includes the feedback of current linear speed and turning angle of chassis body. For the detailed content of protocol, please refer to Table 3.2.

## Table 3.2 Movement Control Feedback Frame

| Command Name | Movement Control Feedback Command | | | |
|---|---|---|---|---|
| Sending node | Receiving node | ID | Cycle (ms) | Receive-timeout (ms) |
| Steer-by-wire chassis | Decision-making control unit | 0x221 | 20ms | None |
| Date length | 0×08 | | | |
| Position | Function | Data type | Description | |
| byte [0] | Moving speed higher 8 bits | signed int16 | Actual speed × 1000 (with an accuracy of 0.001rad) | |
| byte [1] | Moving speed lower 8 bits | | | |
| byte [2] | Reserved | - | 0x00 | |
| byte [3] | Reserved | - | 0x00 | |
| byte [4] | Reserved | - | 0x00 | |
| byte [5] | Reserved | - | 0x00 | |
| byte [6] | Corner higher 8 bits | signed int16 | Actual internal steering angle ×1000 (unit 0.001rad) | |
| byte [7] | Corner lower 8 bits | | | |

The control frame includes linear speed control command, front wheel internal steering angle control command. For its detailed content of protocol, please refer to Table 3.3.

## Table 3.3 Control Frame of movement Control Command

| Command Name | Control Command | | | |
|---|---|---|---|---|
| Sending node | Receiving node | ID | Cycle (ms) | Receive-timeout (ms) |
| Decision-making control unit | Chassis node | 0x111 | 20ms | 500ms |
| Date length | 0×08 | | | |
| Position | Function | Data type | Description | |
| byte [0] | Linear speed higher 8 bits | signed int16 | Chassis moving speed, unit mm/s (effective value+ -1500) | |
| byte [1] | Linear speed lower 8 bits | | | |
| byte [2] | Reserved | — | 0x00 | |
| byte [3] | Reserved | — | 0x00 | |
| byte [4] | Reserved | — | 0x00 | |
| byte [5] | Reserved | — | 0x00 | |
| byte [6] | Corner higher 8 bits | signed int16 | Internal steering angle unit 0.001rad (effective value+ -576) | |
| byte [7] | Corner lower 8 bits | | | |

The mode setting frame is use to set the control interface of HUNTER 2.0. The detailed content of the protocol is as follows.

| Command Name | Control Mode Setting Command | | | |
|---|---|---|---|---|
| Sending node | Receiving node | ID | Cycle (ms) | Receive-timeout (ms) |
| Decision-making control unit | Chassis node | 0x421 | None | None |
| Date length | 0×01 | | | |
| Position | Function | Date type | Description | |
| byte [0] | Control mode | unsigned int8 | 0×00 Standby mode<br>0×01 CAN command mode<br>Power-on enters standby mode default | |

Description of control mode: In case the HUNTER 2.0 is powered on and the RC transmitter is not connected, the control mode is defaulted to standby mode. At this time, the chassis only receives control mode command, and does not respond other commands. To use CAN for control need to switch CAN command mode at first. If the RC transmitter is turned on, the RC transmitter has the highest authority, can shield the control of command and switch the control mode.

Status setting frame is use to clear the system errors. The detailed content of the protocol is as follows.

| Command Name | Status Setting Command | | | |
|---|---|---|---|---|
| Sending node | Receiving node | ID | Cycle (ms) | Receive-timeout (ms) |
| Decision-making control unit | Chassis node | 0x441 | None | None |
| Date length | 0×01 | | | |
| Position | Function | Date type | Description | |
| byte [0] | Errors clearing command | unsigned int8 | 0×00 Clear all not serious failure<br>0×01 Clear steering motor drive communication failure<br>0×02 Clear rear right motor drive communication failure<br>0×03 Clear rear left motor drive communication failure<br>0×05 Clear battery under-voltage failure<br>0×06 Clear steering encoder communication failure<br>0×07 Clear remote control signal loss failure | |

[Note] Example data: The following data is only used for testing
1.The chassis moves forward at 0.15m/s ( It need to unlock parking by command before running)

| byte [0] | byte [1] | byte [2] | byte [3] | byte [4] | byte [5] | byte [6] | byte [7] |
|---|---|---|---|---|---|---|---|
| 0x00 | 0x96 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 |

2.The chassis steering 0.2rad

| byte [0] | byte [1] | byte [2] | byte [3] | byte [4] | byte [5] | byte [6] | byte [7] |
|---|---|---|---|---|---|---|---|
| 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0xC8 |

The chassis status information will be feedback, and what’ s more, the information about motor current, encoder and temperature are also included. The following feedback frame contains the information about motor current, encoder and motor temperature.
The motor numbers of the four motors in the chassis correspond to: steering No. 1, right rear wheel No. 2, and left rear wheel No. 3

## Motor speed current position information feedback

| Command Name | Motor Drive High Speed Information Feedback Frame | | | |
|---|---|---|---|---|
| Sending node | Receiving node | ID | Cycle (ms) | Receive-timeout (ms) |
| Steer-by-wire chassis | Chassis node | 0x251~0x253 | 20ms | None |
| Date length | 0×08 | | | |
| Position | Function | Data type | Description | |
| byte [0] | Motor speed higher 8 bits | signed int16 | Current speed of the motor Unit RPM | |
| byte [1] | Motor speed lower 8 bits | | | |
| byte [2] | Motor current higher 8 bits | signed int16 | Motor current Unit 0.1A | |
| byte [3] | Motor current lower 8 bits | | | |
| byte [4] | Position highest bits | signed int32 | Current position of the motor Unit: pulse | |
| byte [5] | Position second-highest bits | | | |
| byte [6] | Position second-lowest bits | | | |
| byte [7] | Position lowest bits | | | |

| Command Name | Motor Drive Low Speed Information Feedback Frame | | | |
|---|---|---|---|---|
| Sending node | Receiving node | ID | Cycle (ms) | Receive-timeout (ms) |
| Steer-by-wire chassis | Decision-making control unit | 0x261~0x263 | 100ms | None |
| Date length | 0×08 | | | |
| Position | Function | Data type | Description | |
| byte [0] | Drive voltage higher 8 bits | unsigned int16 | Current voltage of drive unit 0.1V | |
| byte [1] | Drive voltage lower 8 bits | | | |
| byte [2] | Drive temperature higher 8 bits | signed int16 | Unit 1°C | |
| byte [3] | Drive temperature lower 8 bits | | | |
| byte [4] | Motor temperature | signed int8 | Unit 1°C | |
| byte [5] | Drive status | unsigned int8 | See the details in [Drive control status] | |
| byte [6] | Reserved | — | 0x00 | |
| byte [7] | Reserved | — | 0x00 | |

| Drive Status | | |
|---|---|---|
| Byte | Bit | Description |
| byte [5] | bit [0] | Whether the power supply voltage is too low (0:Normal 1:Too low) |
| | bit [1] | Whether the motor is overheated (0:Normal 1:Overheated) |
| | bit [2] | Whether the drive is over current (0:Normal 1:Over current) |
| | bit [3] | Whether the drive is overheated (0:Normal 1:Overheated) |
| | bit [4] | Sensor status (0:Normal 1:Abnormal) |
| | bit [5] | Drive error status (0:Normal 1:Error) |
| | bit [6] | Drive enable status (0:Normal 1:Disability) |
| | bit [7] | Reserved |

Parking control command is use to control the motor brake of the driving wheel. The detailed content of the protocol is as follows.

| Command Name | Parking Control Command | | | |
|---|---|---|---|---|
| Sending node | Receiving node | ID | Cycle (ms) | Receive-timeout (ms) |
| Decision-making control unit | Chassis node | 0×131 | None | None |
| Date length | 0×01 | | | |
| Position | Function | Date type | Description | |
| byte [0] | Parking command | unsigned int8 | 0×00 Turn off parking (unlock the brake)<br>0×01 Turning on parking (lock the brake)<br>The brake need to be unlocked to control the speed of the chassis | |

| Command Name | Steering Zero Setting Command | | | |
|---|---|---|---|---|
| Sending node | Receiving node | ID | Cycle (ms) | Receive-timeout (ms) |
| Decision-making control unit | Chassis node | 0x431 | None | None |
| Date length | 0×01 | | | |
| Position | Function | Date type | Description | |
| byte [0] | Setting current position to zero | unsigned int8 | Setting current position to zero<br>Fixed value: 0×AA | |

| Command Name | Steering Zero Setting Feedback Command | | | |
|---|---|---|---|---|
| Sending node | Receiving node | ID | Cycle (ms) | Receive-timeout (ms) |
| Decision-making control unit | Chassis node | 0x43A | None | None |
| Date length | 0×01 | | | |
| Position | Function | Date type | Description | |
| byte [0] | Reply the steering zero setting | unsigned int8 | 0×EE Setting current position to zero successfully | |

| Command Name | BMS Data Feedback | | | |
|---|---|---|---|---|
| Sending node | Receiving node | ID | Cycle (ms) | Receive-timeout (ms) |
| Steer-by-wire | Decision-making control unit | 0x361 | 500ms | None |
| chassis | 0×08 | | | |
| Date length | Function | Date type | Description | |
| byte [0] | Battery SOC | unsigned int8 | Range 0~100 | |
| byte [1] | Battery SOH | unsigned int8 | Range 0~100 | |
| byte [2] | Battery voltage higher 8 bits | unsigned int16 | Unit: 0.01V | |
| byte [3] | Battery voltage lower 8 bits | | | |
| byte [4] | Battery current higher 8 bits | signed int16 | Unit: 0.1A | |
| byte [5] | Battery current lower 8 bits | | | |
| byte [6] | Battery temperature higher 8 bits | signed int16 | Unit: 0.1°C | |
| byte [7] | Battery temperature lower 8 bits | | | |

| Command Name | BMS Data Feedback | | | |
|---|---|---|---|---|
| Sending node | Receiving node | ID | Cycle (ms) | Receive-timeout (ms) |
| Steer-by-wire chassis | Decision-making control unit | 0x362 | 500ms | None |
| Date length | 0×04 | | | |
| Position | Function | Data type | Description | |
| byte [0] | Alarm Status 1 | unsigned int8 | BIT1: Overvoltage BIT2: Undervoltage BIT3: High temperature BIT4: Low temperature BIT7: Discharge overcurrent | |
| byte [1] | Alarm Status 2 | unsigned int8 | BIT0: Charge overcurrent | |
| byte [2] | Warning Status 3 | unsigned int8 | BIT1: Overvoltage BIT2: Undervoltage BIT3: High temperature BIT4: Low temperature BIT7: Discharge overcurrent | |
| byte [3] | Warning Status 4 | unsigned int8 | BIT0: Charge overcurrent | |

| Command Name | Mileage Feedback | | | |
|---|---|---|---|---|
| Sending node | Receiving node | ID | Cycle (ms) | Receive-timeout (ms) |
| Steer-by-wire chassis | Decision-making control unit | 0x311 | 20ms | None |
| Date length | 0×08 | | | |
| Byte | Function | Data type | Description | |
| byte [0] | Left wheel mileometer highest bit | signed int32 | Chassis left wheel mileometer feedback, unit:mm | |
| byte [1] | Left wheel mileometer second-highest bit | | | |
| byte [2] | Left wheel mileometer second-lowest bit | | | |
| byte [3] | Left wheel mileometer lowest bit | | | |
| byte [4] | Right wheel mileometer highest bit | signed int32 | Chassis right wheel mileometer feedback, unit:mm | |
| byte [5] | Right wheel mileometer second-highest bit | | | |
| byte [6] | Right wheel mileometer second-lowest bit | | | |
| byte [7] | Right wheel mileometer lowest bit | | | |

## 3.3.2 CAN cable connection

2 aviation male plugs are supplied along with HUNTER 2.0 as shown in Figure 3.2. Users need to lead wires out by welding on their own. For wire definitions, please refer to Table 3.2.

## 3.3.3 Implementation of CAN command control

Correctly start the chassis of HUNTER 2.0 mobile robot, and turn on RC transmitter. Then, switch to the command control mode, i.e. toggling S1 mode of RC transmitter to the top. At this point, HUNTER 2.0 chassis will accept the command from CAN interface, and the host can also parse the current state of chassis with the real-time data fed back from CAN bus. For the detailed content of protocol, please refer to CAN communication protocol.


Figure 3.2 Schematic Diagram of Aviation Male Plug

## 3.4 Firmware upgrades

In order to facilitate users to upgrade the firmware version used by HUNTER 2.0 and bring customers a more complete experience, HUNTER 2.0 provides a firmware upgrade hardware interface and corresponding client software. A screenshot of this application is shown in Figure 3.3.

### Upgrade preparation

- Serial cable × 1
- USB-to-serial port × 1
- HUNTER 2.0 chassis × 1
- Computer (Windows operating system) × 1

### Upgrade procedure

- Before connection, ensure the robot chassis is powered off;
- Connect the serial cable onto the serial port at rear end of HUNTER 2.0 chassis;
- Connect the serial cable to the computer;
- Open the client software;
- Select the port number;
- Power on HUNTER 2.0 chassis, and immediately click to start connection (HUNTER 2.0 chassis will wait for 3s before power-on; if the waiting time is more than 3s, it will enter the application); if the connection succeeds, "connected successfully" will be prompted in the text box;
- Load Bin file;
- Click the Upgrade button, and wait for the prompt of upgrade completion;
- Disconnect the serial cable, power off the chassis, and turn the power off and on again.

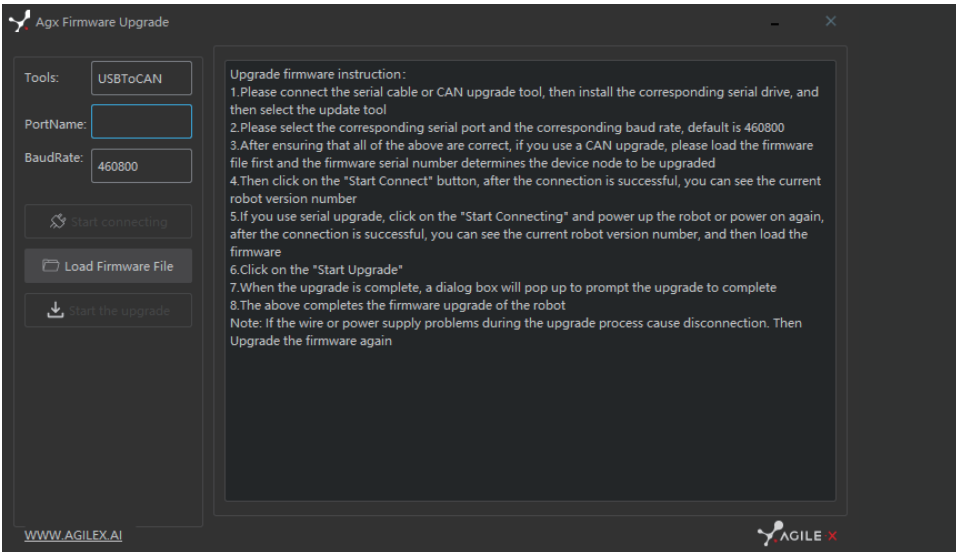



Figure 3.3 Client Interface of Firmware Upgrade

# 3.5 HUNTER 2.0 ROS Package

ROS provide some standard operating system services, such as hardware abstraction, low-level device control, implementation of common function, interprocess message and data packet management. ROS is based on a graph architecture, so that process of different nodes can receive, and aggregate various information (such as sensing, control, status, planning, etc.) Currently ROS mainly support UBUNTU.

## Development Preparation

Hardware preparation

- CANlight can communication module ×1
- Thinkpad E470 notebook ×1
- AGILEX HUNTER 2.0 mobile robot chassis ×1
- AGILEX HUNTER 2.0 remote control FS-i6s ×1
- AGILEX HUNTER 2.0 top aviation power socket ×1

Use example environment description

- Ubuntu 16.04 LTS (This is a test version, tasted on Ubuntu 18.04 LTS)
- ROS Kinetic (Subsequent versions are also tested)
- Git

## Hardware connection and preparation

- Lead out the CAN wire of the HUNTER 2.0 top aviation plug or the tail plug, and connect CAN_H and CAN_L in the CAN wire to the CAN_TO_USB adapter respectively;
- Turn on the knob switch on the HUNTER 2.0 mobile robot chassis, and check whether the emergency stop switches on both sides are released;
- Connect the CAN_TO_USB to the usb point of the notebook. The connection diagram is shown in Figure 3.4.

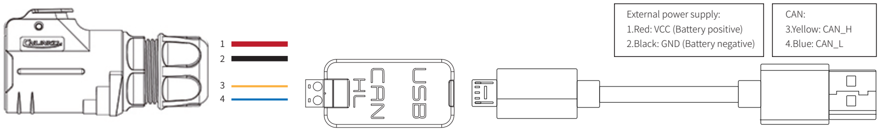



Figure 3.4 CAN connection diagram

## ROS installation and environment setting

For installation details, please refer to

http://wiki.ros.org/kinetic/Installation/Ubuntu

## Test CANABLE hardware and CAN communication

Setting CAN-TO-USB adaptor

- Enable gs_usb kernel module
  $ sudo modprobe gs_usb
- Setting 500k Baud rate and enable can-to-usb adaptor
  $ sudo ip link set can0 up type can bitrate 500000
- If no error occurred in the previous steps, you should be able to use the command to view the can device immediately
  $ ifconfifig -a
- Install and use can-utils to test hardware
  $ sudo apt install can-utils
- If the can-to-usb has been connected to the HUNTER 2.0 robot this time, and the car has been turned on, use the following commands to monitor the data from the HUNTER 2.0 chassis
  $ candump can0
- Please refer to:
  [1]https://github.com/agilexrobotics/agx_sdk
  [2]https://wiki.rdu.im/_pages/Notes/Embedded-System/Linux/can-bus-in-linux.html

## AGILEX HUNTER 2.0 ROS PACKAGE download and compile

- Download ros package
  $ sudo apt install ros-$ROS_DISTRO-teleop-twist-keyboard
  $ sudo apt install ros-$ROS_DISTRO-joint-state-publisher-gui
  $ sudo apt install ros-$ROS_DISTRO-ros-controllers
  $ sudo apt install ros-$ROS_DISTRO-webots-ros
  $ sudo apt install libasio-dev
- Clone compile hunter_2_ros code
  $ cd ~/catkin_ws/src
  $ git clone https://github.com/agilexrobotics/hunter_2_ros.git
  $ git clone https://github.com/agilexrobotics/agx_sdk.git
  $ cd ~/catkin_ws
  $ catkin_make
  Please refer to: https://github.com/agilexrobotics/hunter_2_ros

## Start the ROS node

- Start the based node
  $ roslaunch bunker_bringup hunter_2_robot_base.launch
- Start the keyboard remote operation node
  $ roslaunch bunker_bringup hunter_2_teleop_keyboard.launch

# 4 Attention

This section includes some precautions that should be paid attention to for HUNTER 2.0 use and development.

## 4.1Battery

- The battery supplied with HUNTER 2.0 is not fully charged in the factory setting, but its specific power capacity can be displayed on the voltmeter at real end of HUNTER 2.0 chassis or read via CAN bus communication interface. The battery recharging can be stopped when the green LED on the charger turns green. Note that if you keep the charger connected after the green LED gets on, the charger will continue to charge the battery with about 0.1A current for about 30 minutes more to get the battery fully charged;
- Please do not charge the battery after its power has been depleted, and please charge the battery in time when low battery level alarm is on;
- Static storage conditions: The best temperature for battery storage is -10°C to 45°C; in case of storage for no use, the battery must be recharged and discharged once about every 2 months, and then stored in full voltage state. Please do not put the battery in fire or heat up the battery, and please do not store the battery in high-temperature environment;
- Charging: The battery must be charged with a dedicated lithium battery charger; lithium-ion batteries cannot be charged below 0°C (32°F) and modifying or replacing the original batteries are strictly prohibited.
- HUNTER 2.0 only supports the replacement and use of the battery provided by us, and the battery can be charged separately.

## 4.2 Operational environment

- The operating temperature of HUNTER 2.0 is -10°C to 45°C;
- please do not use it below -10°C and above 45°C ;
- The requirements for relative humidity in the use environment of HUNTER 2.0 are: maximum 80%, minimum 30%;
- Please do not use it in the environment with corrosive and flammable gases or closed to combustible substances;
  Do not place it near heaters or heating elements such as large coiled resistors, etc.;
- Except for specially customized version (IP protection class customized), HUNTER 2.0 is not water-proof, thus please do not use it in rainy, snowy or water-accumulated environment;
- The elevation of recommended use environment should not exceed 1,000m;
- The temperature difference between day and night of recommended use environment should not exceed 25°C;

## 4.4 Other notes

- When handling and setting up, please do not fall off or place the chassis upside down;
- For non-professionals, please do not disassemble the chassis without permission.

## 4.3 Electrical/extension cords

- For the extended power supply on top, the current should not exceed 10A and the total power should not exceed 240W;
- For the extended power supply of top and tail, each ports must not be greater than 24V10A, the total output current must not be greater than 15A, total power should not exceed 360W.
- When the system detects that the battery voltage is lower than the safe voltage class, external power supply extensions will be actively switched to. Therefore, users are suggested to notice if external extensions involve the storage of important data and have no power-off protection.

# 5 Q&A

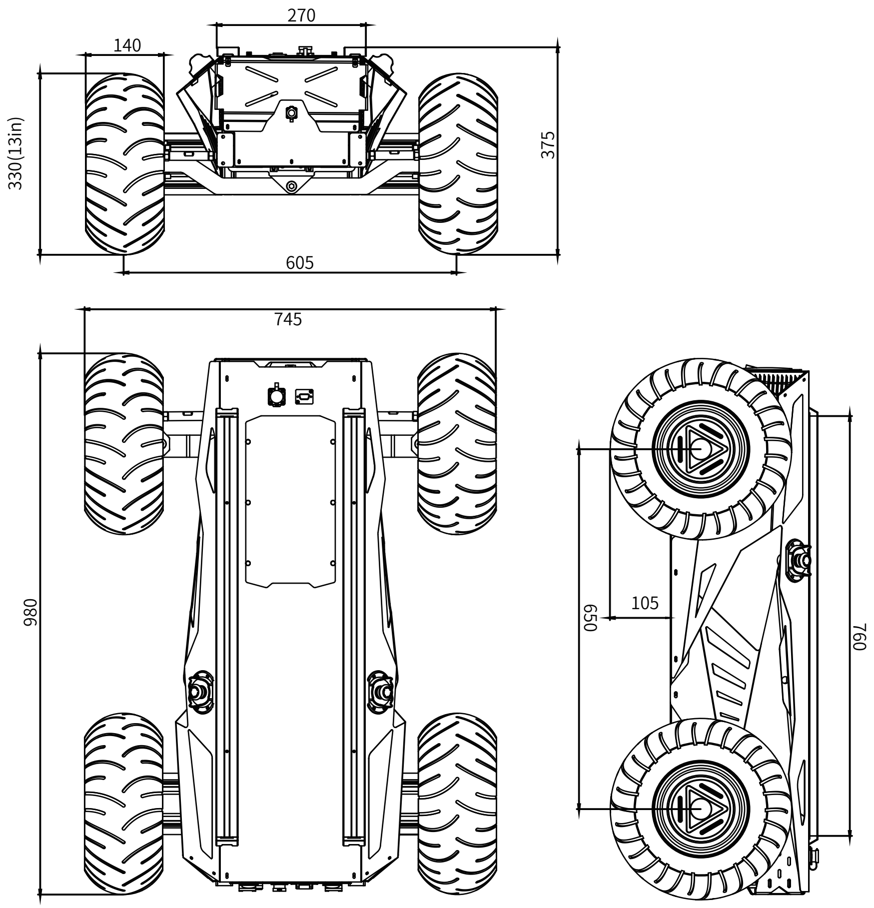
Q: HUNTER 2.0 is started up correctly, but why cannot the RC transmitter control the chassis body to move?
A: First, check whether the drive power supply is in normal condition, whether the drive power switch is pressed down and whether E-stop switches are released; then, check whether the control mode selected with the top left mode selection switch on the RC transmitter is correct, check whether the parking switch is turn off.

Q: HUNTER 2.0 remote control is in normal condition, and the information about chassis status and movement can be received correctly, but when the control frame protocol is issued, why cannot the chassis body control mode be switched and the chassis respond to the control frame protocol?
A: Normally, if HUNTER 2.0 can be controlled by a RC transmitter, it means the chassis movement is under proper control; if the chassis feedback frame can be accepted, it means CAN extension link is in normal condition. Please check the CAN control frame sent to see whether the data check is correct and whether the control mode is in command control mode. You can check the status of error flag from the error bit in the chassis status feedback frame.

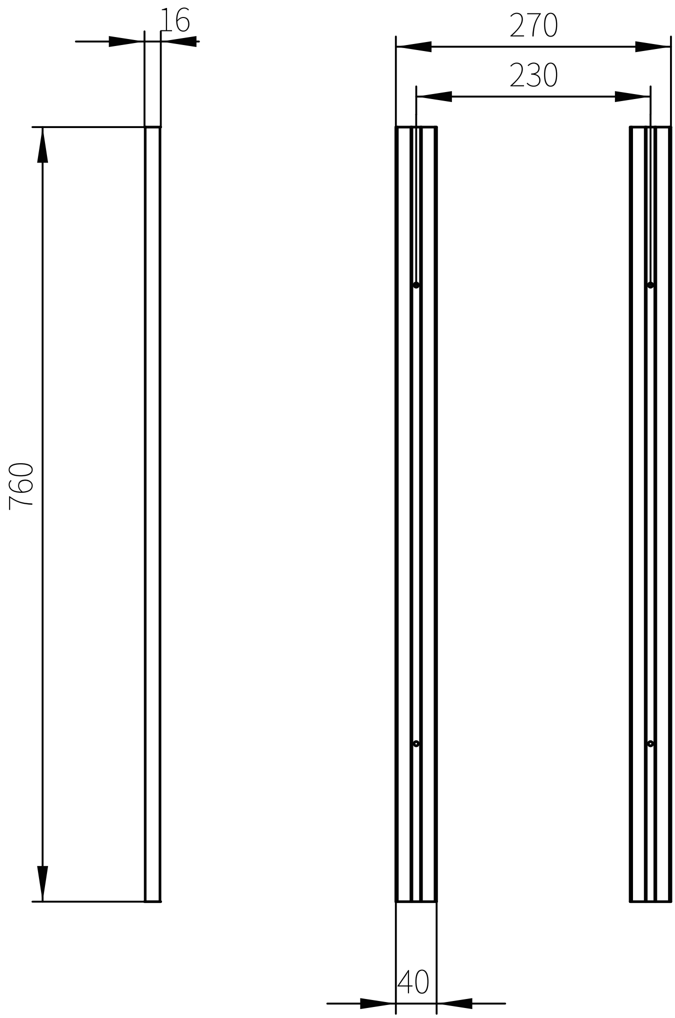
Q: HUNTER 2.0 gives a "beep-beep-beep..." sound in operation, how to deal with this problem?
A: If HUNTER 2.0 gives this "beep-beep-beep" sound continuously, it means the battery is in the alarm voltage state. Please charge the battery in time.

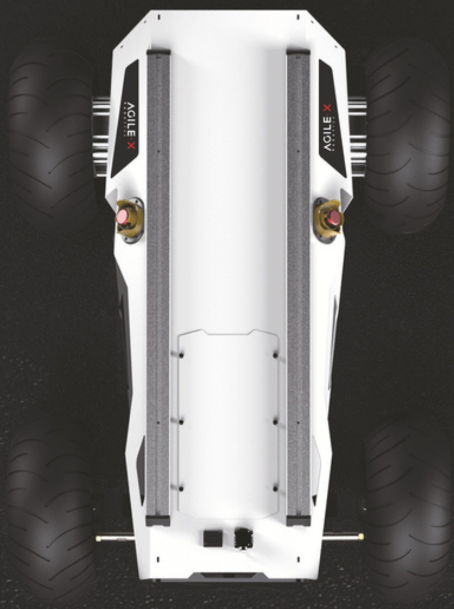
# 6 Product Dimensions

## 6.1 Illustration diagram of product external dimensions



6.2 Illustration diagram of top extended support dimensions





**Distributeur officiel**

sales@generationrobots.com

+33 5 56 39 37 05

www.generationrobots.com