RANGER USER MANUAL

RANGER USER MANUAL
Document version
Important Safety Information
Attention
CONTENTS
1 Introduction tothe RANGER
1.1 Product List
1.2 Performance parameters
1.2 Required for Development
2 Basic Introduction
2.1 Status of the RANGER
2.2 Description of Electrical Interfaces
2.3 Remote Control Instructions
3 Usage and Development
3.1 Operation
3.2 CAN Communication Protocol
3.3 Firmware Upgrade
3.4 RANGER use manual for ROS
3.5 The vehicle body coordinate system
4 Maintenance Instructions
4.1 Maintenance method
5 Product Size

RANGER USER MANUAL



RANGER

AgileX Robotics Team

User Manual V.2.0.1 2023.08

Document version

No.	Version	Date	Edited by	Reviewer	Notes
1	V1.0.0	2022/10/15	何士玉		first draft

2	V2.0.0	2023/03/08	何士玉		• Change appearance picture • Add odometer data feedback • Add remote control data feedback
3	V2.0.1	2023/09/02	谢瑞亲		• Add rendering • Update how to use ROS package

4	V2.0.2	2023/09/07	何士玉		• Change the content of the agreement • Add charging instructions • Added instructions for battery replacement • Add encoder parameters

Before using the robot, any individual or organization must read and understand the manual. If you have any questions about it, please do not hesitate to contact us at support@agilex.ai. It is very important that you should follow and implement all instructions and guidelines in this manual. Please pay extra attention to the warnings.

Important Safety Information

This manual does not cover the design, installation, and operation of a robotic application, nor does it include any equipment that may affect the safety of a robotic system. A robot system that uses the RANGER should be designed and used in compliance with the safety requirements and other standards of the corresponding countries.

Any users of the RANGER should comply with laws and regulations of relevant countries and ensure that there are no obvious hazards in the application of the RANGER. This includes but is not limited to the following:

Responsibility

- Do a risk assessment of the robotic system that uses the RANGER.
- The risk assessment should include additional safety equipment to other machinery.
- Please ensure that the equipment of the whole robotic system, including software and hardware, are designed, and installed correctly.
- The RANGER is not an autonomous mobile robot with anti-collision, anti-fall, biological approach warning, and other safety functions. These safety functions are expected to be developed and assessed by system integrators and end customers under relevant safety regulations and laws to ensure there are not any major dangers and potential safety hazards in their practical applications.
- Read all technical documents: including the risk assessment and this manual.
- Know the possible safety risks before using the RANGER.

Use Environment

- For the first use, please read this manual carefully to understand the basic operation and operating specifications.
- Remote control operation should be in a relatively open area. The RANGER does not have any automatic obstacle avoidance sensors.
- Please use the RANGER under the ambient temperature of -10 ℃~ 40 ℃.
- The RANGER’ s waterproof and dustproof level is IP55 if it is not customized.

Check

- Make sure each device is fully charged.
- Make sure the RANGER has no obvious abnormalities.
- Make sure the remote control has sufficient battery power.

Attention

Operation Precautions

- Make sure the surrounding area is relatively clear during operation
- Remote control within sight range
- The maximum load capacity of RANGER is 150KG. When using it, make sure the payload does not exceed 150KG.
- RANGER When installing an external extension, confirm the position of the center of mass of the extension to ensure that it is at the center of rotation
- When the device alarms for low battery, please charge it in time.
- When something goes wrong with the equipment, please stop using it immediately to avoid secondary damage.

- When an abnormality occurs in the equipment, please contact the relevant technical personnel and do not handle it without authorization.
- Please use the device in an environment that meets the protection level requirements according to the IP protection level of the device.
- Please do not push the cart directly
- The tail extension power supply current does not exceed 15A, and the total power does not exceed 720w.

Battery precautions

- The battery of RANGER products is not fully charged when it leaves the factory. The specific battery voltage and power can be displayed through vol and batt on the RANGER remote control.
- Please do not charge the battery after it is used up. Please charge it in time when the battery of the RANGER remote control is less than 15%.
- Static storage conditions: The optimal storage temperature is -10℃~40℃. When the battery is not in use, it must be charged and discharged once every month, and then stored at full voltage. Do not store the battery Place in fire, or heat the battery. Do not store batteries at high temperatures.
- Charging: You must use the matching dedicated lithium battery charger for charging. Do not charge the battery below 0°C. Do not use non-original standard batteries, power supplies, and chargers.

Precautions for use environment

- The operating temperature of RANGER is -10℃~40℃. Please do not use it in environments with temperatures below -10℃ or above 40℃.
- Do not use it in an environment with corrosive or flammable gases or near flammable substances.
- Please do not use it near heating elements such as heaters or large coil resistors.
- RANGER is waterproof and dustproof to IP55
- It is recommended that the operating environment altitude does not exceed 1000M
- It is recommended that the temperature difference between day and night in the use environment does not exceed 25℃

Safety Precautions

- If you have any questions about the use process, please follow the relevant instruction manual or consult relevant technical personnel.
- Before using the equipment, pay attention to the on-site conditions to avoid improper operation that may cause personal safety problems.
- In case of emergency, press the emergency stop button to power off the equipment.
- Please do not modify the internal device structure without technical support and permission.
- When something goes wrong with the equipment, please stop using it immediately to avoid secondary damage.
- When an abnormality occurs in the equipment, please contact the relevant technical personnel and do not handle it without authorization.

CONTENTS

1 Introduction tothe RANGER

The RANGER is a programmable omnidirectional UGV (UNMANNED GROUND VEHICLE), which is a chassis with a modular design. Compared with the four-wheel differential chassis, the RANGER has obvious advantages when running on ordinary cement roads and asphalt roads. It not only has higher speed and load capacity, but also reduces the wear and tear on the structure and tires. It is also more stable and safer. Compared with Ackermann chassis, the RANGER can realize in-situ steering and has a wider range of applications. The RANGER combines the advantages of differential chassis and Ackermann chassis, which is suitable for various complex terrains. What’ s more, it can be equipped with stereo cameras, LiDAR, GNSS, IMU, manipulators and other equipment to be applied in fields such as unmanned inspection, security, scientific research, exploration and logistics.

1.1 Product List

Name	Quantity
RANGER robot body	x1
Battery charger (AC 220V)	x1
Aviation plug male（4Pin)	x1
FS remote control (optional)	x1
USB to CAN communication module	x1

1.2 Performance parameters

Parameter Types	Items	Values

Mechanical specifications	L × W × H (mm)	1228×876x520
	Wheelbase (mm)	560
	Axle Track（mm）	890
	Curb weight (kg)	100
	Battery Type	Lithium iron phosphate battery
	Battery parameters	48V24AH(single battery)
	Power drive motor	48v brushless toothed motor
	Drive motor power (w)	600w×4
	Drive motor torque (N.M)	22NM×4
	Steering motor torque (N.M)	35nm×4
	Steering motor power (w)	100
	Steering motor encoder	Multi-turn absolute value
	Drive motor encoder	Hall
	Parking mode	Electronic parking
	Steering	Four wheels four steering
	Suspension form	Independent suspension

Performance parameters	Maximum speed (m/s)	2.5
	Maximum obstacle crossing	100（Vertical obstacles fully loaded）
	Maximum gradeability (°)	25
	Weight (kg)	135(single battery)
	Loading capacity (kg)	150
	Maximum battery life (h)	2-4
	Charging time (h)	1（single battery）
	Single battery capacity (ah)	24（Supports up to four batteries）
	Rated voltage (v)	48
	RC transmitter	2.4G/extreme distance 200M
	System interface	CAN
Functional	Application	Engineering Investigation
		Energy Inspection
		Mine Transportation
		Intelligent Security
		Logistics
		Agricultural Product Collection and Transportation

1.2 Required for Development

The RANGER can be equipped with FS remote control when buying. Users can use it to control the 4WD chassis, complete mode switching, movement and steering. The RANGER has a standard CAN (Controller Area Network) communication interface to facilitate secondary development.

2 Basic Introduction

This part is a basic introduction to the RANGER, mobile robot chassis. After reading this part, users and developers can have an overall understanding about it. As shown in Figure 2.1 below, it is an overview of the RANGER.

Figure 2.1 Overview of the RANGER

The RANGER is based on a modular and intelligent design concept. It adopts a composite design of pneumatic tires and independent suspension on the power module, and has a powerful hub motor, which makes the RANGER swift passing ability and ground adaptability. It can move flexibly on different ground. The hub motor does not require complicated transmission design, making the RANGER smaller and more flexible. An open electrical interface and communication interface are configured at the rear of the RANGER, which is convenient for users to carry out secondary development. The electrical interface adopts aviation waterproof connectors, which is not only conducive to the expansion and use, but also allows the RANGER to be used in some harsh environments. A standard aluminum extension bracket is installed on the RANGER, which is convenient for users to carry external equipment.

2.1 Status of the RANGER

The user can check the status of the RANGER through its CAN message. Please refer to Table 2.1 for specific status.

Status	Description

Voltage	Current battery voltage and capacity information can be displayed via the remote control displays vol and batt
Low Voltage Warning	When the SOC (State of Charge) of the battery is lower than 15% through BMS feedback, the front and rear lights of the RANGER will flash as a reminder. When the battery power is detected lower than 10%, the 4WD chassis will actively cut off the power supply for external equipment and driver to protect the battery. At this time, the chassis will not move and accept external command control.
Detailed Status Information	Check by CAN message

Table 2.1 Status Description Table for the RANGER

2.2 Description of Electrical Interfaces

Figure 2.2 Back View of the RANGER

The RANGER has a circular connector for expansion at its rear. The interface can be used to provide power for external equipment (the load current cannot exceed 15 A, and the voltage range is 42~54.6 V) and transfer information through its CAN communication interface. Its pins

are defined in the figure below. Please note that its power supply is subject to internal control. When the battery voltage is lower than the safe level, it will actively cut off the power supply. Therefore, users need to pay attention to the low voltage alarm of the RANGER before reaching the critical voltage. Do not forget to charge the RANGER after use.

Pin Number	Pin Type	Function and Definition	Note
1	Power supply	VCC	Positive terminal. The voltage range is 42~54.6 V The load current cannot exceed 15 A.
2	Power supply	GND	Negative terminal
3	CAN	CAN_H	CAN Hi (High)
4	CAN	CAN_L	CAN Lo (Low)

Figure 2.3 Pin descriptions of the circular connector

2.3 Remote Control Instructions

Figure 2.4 Introduction of the remote control

As shown in the figure above, the functions of the buttons are defined as follows: SWB is the control mode selection lever, dialed to the top is the command control mode, dialed to the middle or down is the remote control mode; SWA is the light control switch, dialed to the bottom is to turn off the light (need SWB first enters the remote control mode); SWC is the ultrasonic obstacle avoidance enable switch (this version does not support it), and when it is turned to the middle, the ultrasonic obstacle avoidance function is turned on; when SWC is turned to the bottom, it is the parking mode, and the four-wheel four-steering is X-shaped at this time Locking **(it is important to note that when the RANGER is turned off in an X-shaped parking mode, it needs to be turned on twice to start the switch button again)**;

SWD is the switch for Motion mode:

Moving SWD to the top is ① front and rear Ackerman mode (the left joystick controls the speed, and the right joystick controls the steering angle) + ② spin mode (the left joystick does not move, the right joystick controls the spin direction)

Moving SWD to the bottom is the oblique motion mode: the left joystick controls the speed, and the right joystick controls the steering angle (the maximum angle is 90°, which makes the RANGER move laterally);

Zero Point Calibration

1. Set the switches as follows:

- SWA: DOWN position
- SWB: UP position
- SWD: DOWN position

2. Use the **left joystick** to select the target steering motor based on direction:
   - Top-right: front-right steering motor
   - Top-left: front-left steering motor
   - Bottom-left: rear-left steering motor
   - Bottom-right: rear-right steering motor
3. Use the **right joystick (left/right)** to adjust the steering angle.
4. Use the **left scroll wheel** to fine-tune the position incrementally in both directions.
5. **SWC** controls sensitivity:
   - DOWN: Coarse adjustment
   - MID: Medium adjustment
   - UP: Fine adjustment
6. Press **Key1** to set the current position as the zero point.

Pressing KEY1 in any case = forcibly clear all errors of the RANGER. **Attention! To be used only in special cases where safety is guaranteed.**

POWER is the power button. Press and hold it to power on.

Basic operation process of remote control:

Before starting, you need to ensure that the wheels and chassis of RANGER are parallel and facing forward. After starting the RANGER mobile robot chassis normally, start the remote controller, switch the SWB to remote control mode, and then control the movement of the RANGER platform through the remote controller.

Remote control battery replacement instructions:

The FS remote control uses 5AA batteries as its energy supply. When the remote control display interface Remoter is relatively low, it means that the battery power of the remote control is too low. At this time, you need to open the battery cover on the back of the remote control and replace the battery.

3 Usage and Development

This part mainly introduces the basic operation and usage of the RANGER, and how to carry out secondary development through the external CAN interface and the CAN bus protocol.

3.1 Operation

- **Check**

Check the RANGER status

Check whether there is any obvious abnormality in the RANGER; if so, please contact after-sales support;

When using it for the first time, check whether the power switch in the rear electrical panel is pressed, if pressed, please press it, and then release it. At this time, the power switch is released, and the RANGER is powered off.

- **Power on and off**

The switch marked with "STOP" at the rear of the RANGER is an emergency stop switch. Pressing it will stop the RANGER immediately and turning it clockwise will exit the emergency stop mode.

- **Charge**

Check the battery voltage. The normal voltage range is 45-54V. If the battery voltage is too low, please charge it in time. This product is equipped with a 20A charger by default. Insert the plug of the charger into the charging socket on the back of the chassis, connect the charger to the power supply, and turn on the switch on the charger to enter the charging state.

Battery replacement:

RANGER itself is equipped with a 48v24ah battery. During operation, when the battery power is too low, we can open the battery panel on the right side to quickly replace the battery.

- **Connection of the CAN Cable**

The 4WD chassis is shipped with an circular connector male head. The definition of its lines can refer to the figure below:

Figure 3.1 Overview of the circular connector

Implementation of CAN command control

Start the RANGER chassis normally. At this time, the RANGER chassis will receive instructions from the CAN interface. At the same time, the host can also analyze the status of the current chassis through the real-time data fed back by the CAN bus. For specific protocol content, refer to the CAN communication protocol. (Note: The remote control has the first control right and has the highest priority. By default, the remote control does not turn on the chassis when it is powered on, and the chassis is in standby mode.)

3.2 CAN Communication Protocol

The CAN communication protocol in this product is CAN2.0B standard, its communication baud rate is 500 K, and its message format is the MOTOROLA format. Through the external CAN interface, users can switch the control model and control the linear speed and steering angle of the RANGER. The RANGER will real-time feedback the current movement status information (including the integrated movement information of the vehicle and the detailed movement information of each wheel) and the system status information (including self-diagnostic error codes).

Command	System Status Feedback Command

Node for sending	Node for receiving	ID	Period (ms)	Receive timeout (ms)
Drive-by-wire chassis	Decision-making and control unit	0x211	20ms	None
Data length	0x08			
Byte	Meaning	Data type	Note	
byte [0]	Current vehicle status	unsigned int8	0x00 The system is normal 0x01 0x01 Emergency shut-down mode 0x02 The system is abnormal	
byte [1]	Control mode	unsigned int8	0x00 Standby mode 0x01 Command control mode 0x03 Remote control mode	
byte [2]	High order byte of battery voltage	unsigned int16	Actual voltage X 10 (the unit is 0.1 V)	
byte [3]	Low order byte of battery voltage			
byte [4]	Highest order byte of error message	unsigned int32	Refer to the error message table for details	
byte [5]	High order byte of error message			
byte [6]	Low order byte of error message			
byte [7]	Lowest order byte of error message			

Error message table

Error message		
Byte	Bit	Meaning

byte [4]	bit [0]	Right front steering servo warning (0: unfaulty; 1: faulty)
	bit [1]	Right rear steering servo warning (0: unfaulty; 1: faulty)
	bit [2]	Left rear steering servo warning (0: unfaulty; 1: faulty)
	bit [3]	Left front steering servo warning (0: unfaulty; 1: faulty)
	bit [4]	Reserved, the default value is 0.
	bit [5]	Reserved, the default value is 0
	bit [6]	Reserved, the default value is 0
	bit [7]	Reserved, the default value is 0

byte [5]	bit [0]	Right front steering zero point calibration status (0: unfaulty; 1: faulty)
	bit [1]	Right rear steering zero point calibration status (0: unfaulty; 1: faulty)
	bit [2]	Left rear steering zero point calibration status (0: unfaulty; 1: faulty)
	bit [3]	Left front steering zero point calibration status (0: unfaulty; 1: faulty)
	bit [4]	Steering calibration timeout (0: unfaulty; 1: faulty)
	bit [5]	Safety edge collision status, (0: normal; 1: triggered, the chassis should be stopped immediately.)
	bit [6]	Reserved, the default value is 0
	bit [7]	Reserved, the default value is 0

byte [6]	bit [0]	driver status (0: unfaulty; 1: faulty)
	bit [1]	Communication connection status with upper layer (0: unfaulty; 1: faulty)
	bit [2]	No. 5 motor driver communication status (0: unfaulty; 1: faulty)
	bit [3]	No. 6 motor driver communication status (0: unfaulty; 1: faulty)
	bit [4]	No. 7 motor driver communication status (0: unfaulty; 1: faulty)
	bit [5]	No. 8 motor driver communication status (0: unfaulty; 1: faulty)
	bit [6]	over temperature protection status (0: normal; 1: triggered)
	bit [7]	Overcurrent protection status (0: normal; 1: triggered)

byte [7]	bit [0]	Battery undervoltage status (0: normal; 1: triggered)
	bit [1]	Ultrasonic obstacle avoidance status (0: normal; 1: triggered)
	bit [2]	Remote control lost connection protection status (0: normal; 1: triggered)
	bit [3]	No. 1 motor driver communication status (0: unfaulty; 1: faulty)
	bit [4]	No. 2 motor driver communication status (0: unfaulty; 1: faulty)
	bit [5]	No. 3 motor driver communication status (0: unfaulty; 1: faulty)
	bit [6]	No. 4 motor driver communication status (0: unfaulty; 1: faulty)
	bit [7]	Reserved, the default value is 0

The motion control feedback frame includes the current linear speed and steering angle of the vehicle.

The details of the protocol are as follows

Command	Motion Control Feedback Command			
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)
Drive-by-wire chassis	Decision-making and control unit	0x221	20ms	None
Data length	0x08			
Byte	Meaning	Data type	Note	

byte [0] byte [1]	High order byte of speed Low order byte of speed	signed int16	Actual speed X 1000 (the unit is 0.001 m/s)
byte [2]	Reserved	-	0X00
byte [3]	Reserved	-	0X00
byte [4]	Reserved	-	0X00
byte [5]	Reserved	-	0X00
byte [6] byte [7]	High order byte of steering angle Low order byte of steering angle	signed int16	Actual steering angle X 100 (the unit is 0.01 °)

The motion control frame includes the linear speed control command and the steering angle control command. The details of the protocol are as follows:

Command			Control Command	
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)
Decision-making and control unit	Node for the chassis	0x111	20ms	500ms
Data length	0x08			
Byte	Meaning	Data type	Note	
byte [0] byte [1]	High order byte of linear speed Low order byte of linear speed	signed int16	Speed of the vehicle, whose unit is mm/s (valid value + -2500; valid value + -1250 when the steering angle > 20°; taking effect in front and rear Ackerman mode and oblique motion mode) Forward direction is positive	
byte [2]	Reserved	-	0X00	

byte [3]	Reserved	-	0X00
byte [4] byte [5]	High order byte of spin speed Low order byte of spin speed	signed int16	Speed of the vehicle, whose unit is mm/s (valid value + -1000, taking effect in spin mode) Counterclockwise spinning is positive.
byte [6] byte [7]	High order byte of steering angle Low order byte of steering angle	signed int16	Steering angle, whose unit is 0.01° (valid value +- 4000 in front and rear Ackerman mode, valid value +- 9000 in oblique motion mode, only taking effect in front and rear Ackerman mode and oblique motion mode)

As shown in Figure 3.2.1, when the RANGER is in front and rear Ackerman mode, the feedback steering angle is (α+β)/2, left steering is negative, and right steering is positive; the feedback speed is the average value of the four wheels' speed (that is, the linear speed of the chassis), reversing is negative, and moving forward is positive. If you need to check the detailed steering angle and speed of each wheel, please refer to 0X271 and 0X281 feedback frames.

Figure 3.2.1 The Ackerman structure of the RANGER

As shown in Figure 3.2.2, when the RANGER is in oblique motion mode, the feedback steering angle is (α1+α2+α3+α4)/4, left steering is negative, and right steering is positive; the feedback linear speed is the average value of the four wheels' speed, reversing is negative, and moving forward is positive. If you need to check the detailed steering angle and speed of each wheel, please refer to the 0x271 and 0x281 feedback frames.

Figure 3.2.2 Wheels control of the RANGER in oblique motion mode

When the chassis is in the spin mode, the steering angle is a constant value , which cannot be changed. At this time, the feedback steering angle is the average value of the absolute values of α1, α2, α3, and α4. The spinning speed of the chassis can be changed by commands, and counterclockwise spinning is positive.

The mode setting frame is used to set the terminal control interface, and the details of the protocol are as follows.

Command	Control Command

Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)
Decision-making and control unit	Node for the chassis	0x421	None	None
Data length	0x01			
Byte	Meaning	Data type	Note	
byte [0]	Control mode	unsigned int8	0x00 Standby mode 0x01 CAN command control mode Boot into standby mode by default	

Control mode description: when the chassis is powered on and the remote control is not connected, the control mode is standby mode. At this time, the chassis only receives control mode commands and does not respond to other commands. To control the chassis using CAN, you need to switch control mode to CAN command control mode first. If the remote control is turned on, the remote control has the highest priority, which can block the control command and switch the control mode. The status setting frame is used to clear system errors, and the details of the protocol are as follows.

Command			Control Command	
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)
Decision-making and control unit	Node for the chassis	0x441	None	None
Data length	0x01			
Byte	Meaning	Data type	Note	

byte [0]	Error clearing command	unsigned int8	0x00 Clear all non-critical faults 0x01~0x08 Clear the communication faults of No. 1~8 motor drivers respectively 0x09 Clear the battery undervoltage fault and try to restore the power supply 0x0a Clear remote control signal loss fault 0x0b~0x0e Clear the steering calibration fault of No. 5~8 motors respectively 0x0f Clear overcurrent fault 0x10 Clear over temperature fault

1. Sample data, the following data is only for testing.

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

1. The chassis turns 10°

byte [0]	byte [1]	byte [2]	byte [3]	byte [4]	byte [5]	byte [6]	byte [7]
0x00	0x00	0x00	0x00	0x00	0x00	0x03	0xe8

In addition to the status of the chassis itself, its feedback information also includes the steering angle and speed of the four wheels, the current of the motor, the encoder information, and the temperature information.

The details of the protocol are as follows:

PS: The eight motor numbers of the chassis are: No. 1 is the right front wheel motor, No. 2 is the right rear wheel motor, No. 3 is the left rear wheel motor, No. 4 is the left front wheel motor, No. 5 is the right front steering motor, No. 6 is the right rear steering motor, No. 7 is the left rear steering motor, and No. 8 is the left front steering motor.

Feedback information of speed, current, and position of motor

Command	High-speed feedback information frame for motor driver

Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)
Drive-by-wire chassis	Decision-making and control unit	0x251~0x258	20ms	None
Data length	0x08			
Byte	Meaning	Data type	Note	
byte [0] byte [1]	High order byte of motor speed Low order byte of motor speed	signed int16	The current speed of the motor, whose unit is RPM (Revolutions Per Minute)	
byte [2] byte [3]	High order byte of motor current Low order byte of motor current	signed int16	The present current of the motor, whose unit is 0.1 A	
byte [4] byte [5] byte [6] byte [7]	Highest order byte of position High order byte of position Low order byte of position Lowest order byte of position	signed int32	The current position of the motor, whose unit is the number of pulses	

Feedback of temperature voltage and status of motor

Command		Low-speed information feedback frame for motor driver		
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)

Drive-by-wire chassis	Decision-making and control unit	0x261~0x268	100ms	None
Data length	0x08			
Byte	Meaning	Data type	Note	
byte [0]	High order byte of driver voltage	unsigned int16	The current driver voltage, whose unit is 0.1 V	
byte [1]	Low order byte of driver voltage			
byte [2]	High order byte of drive temperature	signed int16	The unit is 1 ℃.	
byte [3]	Low order byte of driver temperature			
byte [4]	Motor temperature	signed int8	The unit is 1 ℃.	
byte [5]	Driver status	unsigned int8	See Table 2 for details	
byte [6]	Reserved	-	0X00	
byte [7]	Reserved	-	0X00	

Table 2 Driver status

Byte	Bit	Meaning

byte[5]	bit[0]	Power supply voltage status (0: normal; 1: too low)
	bit[1]	Motor temperature status (0: normal; 1: over temperature)
	bit[2]	The current status of the driver(0: normal; 1: over-current)
	bit[3]	Driver temperature status (0: normal; 1: over temperature)
	bit[4]	Sensor status (0: Normal; 1: Abnormal)
	bit[5]	Driver status (0: Normal; 1: Abnormal)
	bit[6]	Drive enable status (0: enable; 1: disable)
	bit[7]	Reserved

Steering angle feedback of four wheels

Command		Information feedback frame of four wheels' steering angle		
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)
Drive-by-wire chassis	Decision-making and control unit	0x271	20ms	None
Data length	0x08			
Byte	Meaning	Data type	Note	
byte [0] byte [1]	High order byte of steering angle of No. 5 motor Low order byte of steering angle of No. 5 motor	signed int16	The Current steering angle, whose unit is 0.01 °	

byte [2] byte [3]	High order byte of steering angle of No. 6 motor Low order byte of steering angle of No. 6 motor	signed int16	The Current steering angle, whose unit is 0.01 °
byte [4] byte [5]	High order byte of steering angle of No. 7 motor Low order byte of steering angle of No. 7 motor	signed int16	The Current steering angle, whose unit is 0.01 °
byte [6] byte [7]	High order byte of steering angle of No. 8 motor Low order byte of steering angle of No. 8 motor	signed int16	The Current steering angle, whose unit is 0.01 °

Rotational speed feedback of four wheels

Command		Information feedback frame of four wheels' rotational speed		
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)
Drive-by-wire chassis	Decision-making and control unit		20ms	None
Data length	0x08	0x281		
Byte	Meaning	Data type	Note	

byte [0] byte [1]	High order byte of rotational speed of No. 1 motor Low order byte of rotational speed of No. 1 motor	signed int16	The current rotational speed, whose unit is mm/s
byte [2] byte [3]	High order byte of rotational speed of No. 2 motor Low order byte of rotational speed of No. 2 motor	signed int16	The current rotational speed, whose unit is mm/s
byte [4] byte [5]	High order byte of rotational speed of No. 3 motor Low order byte of rotational speed of No. 3 motor	signed int16	The current rotational speed, whose unit is mm/s
byte [6] byte [7]	High order byte of rotational speed of No. 4 motor Low order byte of rotational speed of No. 4 motor	signed int16	The current rotational speed, whose unit is mm/s

The motion mode switching command is used to change motion model of the chassis, and the details of the protocol are as follows

Command	Current motion mode feedback command

Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)
Drive-by-wire chassis	Decision-making and control unit	0x291	20ms	None
Data length	0x02			
Byte	Meaning	Data type	Note	
byte [0]	Current motion mode	unsigned int8	0x00 front and rear Ackerman mode 0x01 oblique motion mode 0x02 spin mode 0x03 Parking mode	
byte [1]	Whether the chassis is in the process of switching the motion model	unsigned int8	0x00 switching is completed. 0x01 in the process of switching motion mode The chassis does not respond to speed control commands in the process of switching motion mode.	

The motion mode switching command is used to change motion model of the chassis, and the details of the protocol are as follows

Command			Control command	
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)
Decision-making and control unit	Node for the chassis	0x141	None	None
Data length	0x01			
Byte	Meaning	Data type	Note	

byte [0]	Motion mode	unsigned int8	0x00 front and rear Ackerman mode (default) 0x01 oblique motion mode 0x02 spin mode 0x03 Parking mode

Odometer information feedback frame is as follows

Front wheel

Command	Odometer information feedback			
Node for sending	Node for receiving	ID	Period （ms）	Receive timeout (ms)
Drive-by-wire chassis	Decision-making and control unit	0×311	20ms	None
Data length	0x08			
Byte	Meaning	Data type	Note	
byte [0] byte [1] byte [2] byte [3]	Front left wheel odometer highest byte Front left wheel odometer second highest byte Front left wheel odometer second lowest byte Front left wheel odometer lowest byte	signed int32	Chassis left wheel odometer feedback, Unit: mm	

byte [4] byte [5] byte [6] byte [7]	Front right wheel odometer highest byte Front right wheel odometer second highest byte Front right wheel odometer second lowest byte Front right wheel odometer lowest byte	signed int32	Chassis right wheel odometer feedback, Unit: mm

Rear wheel

Command	Odometer information feedback			
Node for sending	Node for receiving	ID	Period （ms）	Receive timeout (ms)
Drive-by-wire chassis	Decision-making and control unit	0×312	20ms	None
Data length	0x08			
Byte	Meaning	Data type	Note	

byte [0] byte [1] byte [2] byte [3]	Rear left wheel odometer highest byte Rear left wheel odometer second highest byte Rear left wheel odometer second lowest byte Rear left wheel odometer lowest byte	signed int32	Chassis left wheel odometer feedback, Unit: mm
byte [4] byte [5] byte [6] byte [7]	Rear right wheel odometer highest byte Rear right wheel odometer second highest byte Rear right wheel odometer second lowest byte Rear right wheel odometer lowest byte	signed int32	Chassis right wheel odometer feedback, Unit: mm

The remote control information feedback frame is as follows

Command	Remote control information feedback			
Node for sending	Node for receiving	ID	Period （ms）	Receive timeout (ms)

Drive-by-wire chassis	Decision-making and control unit	0x241	20ms	None
Data length	0x08			
Byte	Meaning	Data type	Note	
byte [0]	Remote control SW feedback	unsigned int8	bit[0-1]: SWA: 2-Upper gear, 3-Lower gear bit[2-3]: SWB: 2-Upper gear, 1-Middle gear, 3-Lower gear bit[4-5]: SWC: 2-Upper gear, 1-Middle gear, 3-Lower gear bit[6-7]: SWD: 2-Upper gear, 3-Lower gear	
byte [1]	Right joystick left/right	unsigned int8	Value range: [-100,100]	
byte [2]	Right joystick up/down	unsigned int8	Value range: [-100,100]	
byte [3]	Left joystick up/down	unsigned int8	Value range: [-100,100]	
byte [4]	Left joystick left/right	unsigned int8	Value range: [-100,100]	
byte [5]	left knob VRA	unsigned int8	Value range: [-100,100]	
byte [6]	Reserved	--	0x00	
byte [7]	Counting verification	unsigned int8	0-255 cycle count	

Ultrasonic Feedback Command

Command	Ultrasonic Feedback Command			
Node for sending	Node for receiving	ID	Period (ms)	Receive timeout (ms)

Drive-by-wire chassis	Decision-making and control unit	342	50ms	None
Data length	0×08			
Byte	Meaning	Data type	Note	

byte【0】	High order byte of the distance measured by No. 1 ultrasonic sensor	int16	The distance unit is nm. Positive value: valid data -1: the sensor is offline. -2: sensor data validation failed -3: abnormal sensor data -4: system exception
byte【1】	Low order byte of the distance measured by No. 1 ultrasonic sensor		
byte【2】	High order byte of the distance measured by No. 2 ultrasonic sensor	int16	
byte【3】	Low order byte of the distance measured by No. 2 ultrasonic sensor		
byte【4】	High order byte of the distance measured by No. 3 ultrasonic sensor	int16	
byte【5】	Low order byte of the distance measured by No. 3 ultrasonic sensor		

byte【6】	High order byte of the distance measured by No. 4 ultrasonic sensor	int16	
byte【7】	Low order byte of the distance measured by No. 4 ultrasonic sensor		

Command	Ultrasonic Feedback Command			
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)
Drive-by-wire chassis	Decision-making and control unit	343	50ms	None
Data length	0×08			
Byte	Meaning	Data type	Note	

byte【0】	High order byte of the distance measured by No. 5 ultrasonic sensor	int16	The distance unit is nm. Positive value: valid data -1: the sensor is offline. -2: sensor data validation failed -3: abnormal sensor data -4: system exception
byte【1】	Low order byte of the distance measured by No. 5 ultrasonic sensor		
byte【2】	High order byte of the distance measured by No. 6 ultrasonic sensor	int16	
byte【3】	Low order byte of the distance measured by No. 6 ultrasonic sensor		
byte【4】	High order byte of the distance measured by No. 7 ultrasonic sensor	int16	
byte【5】	Low order byte of the distance measured by No. 7 ultrasonic sensor		

byte【6】	High order byte of the distance measured by No. 8 ultrasonic sensor	int16	
byte【7】	Low order byte of the distance measured by No. 8 ultrasonic sensor		

Ultrasonic obstacle avoidance setting command

Command	Ultrasonic obstacle avoidance setting command			
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)
Decision-making and control unit	Node for the chassis	0×451	None	None
Data length	0×01			
Byte	Meaning	Data type	Note	
byte[0]	Set the ultrasonic obstacle avoidance	unsigned int8	0×00 Disable ultrasonic obstacle avoidance 0×01 Enable ultrasonic obstacle avoidance	

Feedback command of ultrasonic obstacle avoidance setting

Command	Feedback command of ultrasonic obstacle avoidance setting			
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)

Node for the chassis	Decision-making and control unit	0×45A	None	None
Data length	0×01			
Byte	Meaning	Data type	Note	
byte[0]	Current status of ultrasonic obstacle avoidance	unsigned int8	0×00 Ultrasonic obstacle avoidance is disabled. 0×01 Ultrasonic obstacle avoidance is enabled.	

The feedback data of BMS for all batteries, and the details of the protocol are as follows

Command		The feedback data of BMS		
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)
Drive-by-wire chassis	Decision-making and control unit	0x361	500ms	None
Data length	0x08			
Byte	Meaning	Data type	Note	
byte [0]	Battery SOC (State of Charge)	unsigned int8	Range 0~100	
byte [1]	Battery SOH (State of Health)	unsigned int8	Range 0~100	
byte [2] byte [3]	High order byte of battery voltage Low order byte of battery voltage	unsigned int16	Unit: 0.01 V	

byte [4] byte [5]	High order byte of battery current Low order byte of battery current	signed int16	Unit: 0.1 A
byte [6] byte [7]	High order byte of battery temperature Low order byte of battery temperature	signed int16	Unit: 0.1 ℃

Command		The feedback data of BMS		
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)
Drive-by-wire chassis	Decision-making and control unit	0x362	500ms	None
Data length	0x04			
Byte	Meaning	Data type	Note	
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: Charging overcurrent	
byte [2]	Warning Status 1	unsigned int8	BIT1: Overvoltage; BIT2: Undervoltage; BIT3: High temperature; BIT4: Low temperature; BIT7: Discharge overcurrent	
byte [3]	Warning Status 2	unsigned int8	BIT0: Charging overcurrent	

The feedback data of BMS for No. 1 battery

Command		The feedback data of BMS		
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)
Drive-by-wire chassis	Decision-making and control unit	0x363	500ms	None
Data length	0x08			
Byte	Meaning	Data type	Note	
byte [0]	Battery SOC	unsigned int8	Range 0~100	
byte [1]	Battery SOH	unsigned int8	Range 0~100	
byte [2] byte [3]	High order byte of battery voltage Low order byte of battery voltage	unsigned int16	Unit: 0.01 V	
byte [4] byte [5]	High order byte of battery current Low order byte of battery current	signed int16	Unit: 0.1 A	
byte [6] byte [7]	High order byte of battery temperature Low order byte of battery temperature	signed int16	Unit: 0.1 ℃	
Command			The feedback data of BMS	
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)

Drive-by-wire chassis	Decision-making and control unit	0x364	500ms	None
Data length	0x04			
Byte	Meaning	Data type	Note	
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: Charging overcurrent	
byte [2]	Warning Status 1	unsigned int8	BIT1: Overvoltage; BIT2: Undervoltage; BIT3: High temperature; BIT4: Low temperature; BIT7: Discharge overcurrent	
byte [3]	Warning Status 2	unsigned int8	BIT0: Charging overcurrent	

The feedback data of BMS for No. 2 battery

Command		The feedback data of BMS		
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)
Drive-by-wire chassis	Decision-making and control unit	0x365	500ms	None
Data length	0x08			
Byte	Meaning	Data type	Note	
byte [0]	Battery SOC	unsigned int8	Range 0~100	
byte [1]	Battery SOH	unsigned int8	Range 0~100	

byte [2] byte [3]	High order byte of battery voltage Low order byte of battery voltage	unsigned int16	Unit: 0.01 V	
byte [4] byte [5]	High order byte of battery current Low order byte of battery current	signed int16	Unit: 0.1 A	
byte [6] byte [7]	High order byte of battery temperature Low order byte of battery temperature	signed int16	Unit: 0.1 ℃	
Command			The feedback data of BMS	
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)
Drive-by-wire chassis	Decision-making and control unit	0x366	500ms	None
Data length	0x04			
Byte	Meaning	Data type	Note	
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: Charging overcurrent	

byte [2]	Warning Status 1	unsigned int8	BIT1: Overvoltage; BIT2: Undervoltage; BIT3: High temperature; BIT4: Low temperature; BIT7: Discharge overcurrent
byte [3]	Warning Status 2	unsigned int8	BIT0: Charging overcurrent

The feedback data of BMS for No. 3 battery

Command		The feedback data of BMS		
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)
Drive-by-wire chassis	Decision-making and control unit	0x367	500ms	None
Data length	0x08			
Byte	Meaning	Data type	Note	
byte [0]	Battery SOC	unsigned int8	Range 0~100	
byte [1]	Battery SOH	unsigned int8	Range 0~100	
byte [2] byte [3]	High order byte of battery voltage Low order byte of battery voltage	unsigned int16	Unit: 0.01 V	
byte [4] byte [5]	High order byte of battery current Low order byte of battery current	signed int16	Unit: 0.1 A	

byte [6] byte [7]	High order byte of battery temperature Low order byte of battery temperature	signed int16	Unit: 0.1 ℃	
Command			The feedback data of BMS	
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)
Drive-by-wire chassis	Decision-making and control unit	0x368	500ms	None
Data length	0x04			
Byte	Meaning	Data type	Note	
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: Charging overcurrent	
byte [2]	Warning Status 1	unsigned int8	BIT1: Overvoltage; BIT2: Undervoltage; BIT3: High temperature; BIT4: Low temperature; BIT7: Discharge overcurrent	
byte [3]	Warning Status 2	unsigned int8	BIT0: Charging overcurrent	

The feedback data of BMS for No. 4 battery

Command		The feedback data of BMS		
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)

Drive-by-wire chassis	Decision-making and control unit	0x369	500ms	None
Data length	0x08			
Byte	Meaning	Data type	Note	
byte [0]	Battery SOC	unsigned int8	Range 0~100	
byte [1]	Battery SOH	unsigned int8	Range 0~100	
byte [2] byte [3]	High order byte of battery voltage Low order byte of battery voltage	unsigned int16	Unit: 0.01 V	
byte [4] byte [5]	High order byte of battery current Low order byte of battery current	signed int16	Unit: 0.1 A	
byte [6] byte [7]	High order byte of battery temperature Low order byte of battery temperature	signed int16	Unit: 0.1 ℃	

Command			The feedback data of BMS	
Node for sending	Node for receiving	ID	Period（ms）	Receive timeout (ms)
Drive-by-wire chassis	Decision-making and control unit	0x36A	500ms	None
Data length	0x04			

Byte	Meaning	Data type	Note
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: Charging overcurrent
byte [2]	Warning Status 1	unsigned int8	BIT1: Overvoltage; BIT2: Undervoltage; BIT3: High temperature; BIT4: Low temperature; BIT7: Discharge overcurrent
byte [3]	Warning Status 2	unsigned int8	BIT0: Charging overcurrent

3.3 Firmware Upgrade

To facilitate users to upgrade the firmware of the 4WD chassis and bring customers a better experience, the 4WD chassis provides a hardware interface and a software for upgrading firmware. The GUI (Graphical User Interface) of the software is shown in the figure below.

Upgrade Preparation

Songling CAN debugging module X 1

micro USB cable X 1

the 4WD chassis X 1

a computer (WINDOWS OS (Operating System)) X 1

Upgrade Process

1.Plug in the USBTOCAN module on the computer, and then open the AgxCandoUpgradeToolV1.3_boxed.exe software (the sequence cannot be wrong, first open the software and then plug in the module, the device will not be recognized).

2.Click the Open Serial button, and then press the power button on the car body. If the connection is successful, the version information of the main control will be recognized, as shown in the figure.

3.Click the Load Firmware File button to load the firmware to be upgraded. If the loading is successful, the firmware information will be obtained, as shown in the figure

4.Click the node to be upgraded in the node list box, and then click Start Upgrade Firmware to start upgrading the firmware. After the upgrade is successful, a pop-up box will prompt.

3.4 RANGER use manual for ROS

ROS (Robot Operating System) provides some standard operating system services, such as hardware abstraction, low-level device control, implementation of commonly used functionality, message-passing between processes, and package management. ROS is based on a graph architecture, where processing takes place in nodes that may receive, post, and multiplex various information (such as sensor data, control, state, planning, and other messages). Currently ROS mainly supports UBUNTU OS.

Development Preparation

Hardware Equipment

- CANlight CAN communication module X1
- Laptop X1
- AGILEX Ranger mobile robot chassis X1
- Paired remote control FS-i6s for the AGILEX Ranger X1
- Circular connector on the rear of the AGILEX Ranger X1

Tested Development Environment

- Ubuntu 18.04/Ubuntu 22.04
- ROS1/ROS2
- Git

Hardware Connection and Preparation

- Lead out the CAN cable from the top aviation plug or tail plug, and connect CAN_H and CAN_L in the CAN cable to the CAN_TO_USB adapter respectively;
- Turn on the knob switch on the chassis of the mobile robot and check whether the emergency stop switches on both sides are released;
- Connect CAN_TO_USB to the laptop's USB port. The connection diagram is shown in Figure 3.4.

Figure 3.4 The CAN wiring diagram

ROS Installation and Environment Setup

Please refer to http://wiki.ros.org/kinetic/Installation/Ubuntu for details

Test the Communication between CANABLE hardware and CAN

Set the CAN-TO-USB adapter

- Enable the gs_usb kernel module

sudo modprobe gs_usb

- Set baud rate to 500k and enable the CAN-TO-USB adapter

sudo ip link set can0 up type can bitrate 500000

- If there is no error in the previous steps, you can check the CAN devices with the command below

ifconfifig -a

- Install and use can-utils to test hardware

sudo apt install can-utils

- If the CAN-TO-USB adapter has been connected to the RANGER and the RANGER has been powered on, the command below can be used to monitor the data from the RANGER.

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 BUNKER ROS PACKAGE Download and compile

- Download ros dependencies

$ sudo apt install libasio-dev

$ sudo apt install ros-$ROS_DISTRO-teleop-twist-keyboard

- Clone and compile the bunker_ros source code

$ cd ~/catkin_ws/src

$ git clone --recursive https://github.com/agilexrobotics/ugv_sdk.git

$ git clone https://github.com/agilexrobotics/ranger_ros.git

$ cd ..

$ catkin_make

- Reference：https://github.com/agilexrobotics/bunker_ros

Start the ROS node

- Start the base node

roslaunch ranger_bringup ranger_minimal.launch

Note that the usb_to_can module equipped with Songling needs to be enabled before starting. The enabling command is as follows: rosrun ranger_bringup bringup_can2usb.bash. This command only needs to be executed once each time the usb_to_can module is powered on.

- Run the keyboard_control node

roslaunch ranger_bringup ranger_teleop_keyboard.launch

Pay attention to the terminal printout and use the designated keys to control the ranger movement.

Github ROS development package directory and usage instructions

*_base:: The core node for the chassis to send and receive hierarchical CAN messages. Based on the communication mechanism of ros, it can control the movement of the chassis and read the status of the bunker through the topic.

*_msgs: Define the specific message format of the chassis status feedback topic

*_bringup: startup files for chassis nodes and keyboard control nodes, and scripts to enable the usb_to_can module

3.5 The vehicle body coordinate system

Taking the center of the vehicle body as the coordinate origin, facing forward is the positive direction of the x-axis, facing left is the positive direction of the y-axis, and facing upward is the positive direction of the z-axis.

4 Maintenance Instructions

4.1 Maintenance method

The vehicle maintenance

When the tires are severely worn, please replace them in time.

Battery maintenance

If the battery is not used for a long time, please don't forget to charge it every 2 to 3 months.

5 Product Size