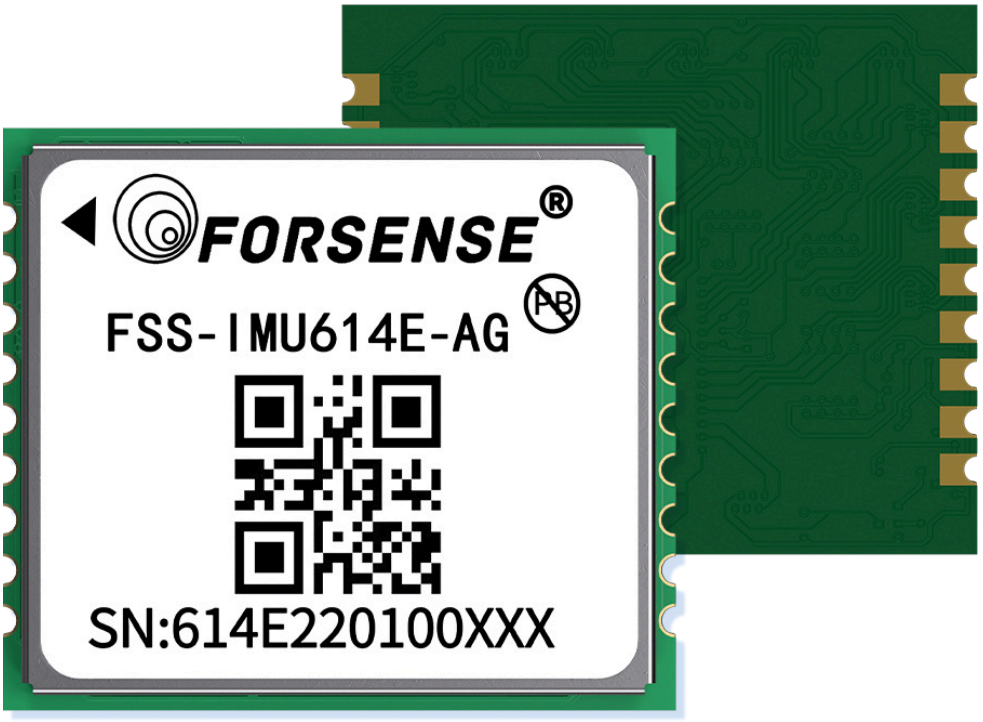
FORSENSE
原极科技

Agricultural Single-Antenna GNSS

IMU614E-AG

Product manual



Forsense (Shanghai) Technology Co., Ltd

Please read this user manual carefully before using this product.

Table of Contents

1. Overview **3**
2. Outline Structure **4**
3. Electrical Characteristics **5**
3.1 Maximum tolerance value 5
3.2 Operating conditions 5
3.3 IO threshold characteristics 5
4. Pin Definition **6**
5. Recommended soldering furnace temperature profile **8**
6. ESD protection **10**
7. Usage example **11**
7.1 Device Installation 11
7.2 Connect RTK board 12
7.2.1 RTK board and the AG moduleRecommended connection method 12
7.2.2 RTK Configuration Requirements 13
7.2.3 RTK differential data import 13
7.3 Parameter Configuration 13
7.3.1 Configure the Primary antenna lever arm 13
7.3.2 Configure rear wheel axle center lever arm 14
7.3.3 Configure vehicle wheelbase 15
7.3.4 Configure RTK Dual antenna installation angle 15
7.3.5 Enable Dual antenna fusion 15
7.3.6 Calibration Dual antenna 16
7.4 Save Parameter 16
8. Output Protocol **17**
8.1 Binary Protocol - AG Data Stream 17
8.2 Binary Protocol - Integrated Navigation Data Stream 18
8.3 NMEA protocol 20
8.4 CAN protocol 28
9. Input Protocol **30**
9.1 Binary Configuration Protocol 30
9.2 String Configuration Protocol 30
9.2.1 ConfigurationPrimary antennaLever Arm 30
9.2.2 Configure rear axle center lever arm 31
9.2.3 Configure vehicle wheelbase 32
9.2.4 Configure RTK Dual antenna installation angle 32
9.2.5 Enable Dual antenna fusion 33
9.2.6 Calibrate Dual antenna 33
9.2.7 Configure the position and velocity projection points output by Integrated Navigation 33
9.2.8 Configure output binary-AG data stream 33
9.2.9 Configure output binary-Integrated Navigation data stream 34
9.2.10 Configure output of NMEA format data stream 34
9.2.11 Configure the current data stream to stop output. 34
9.2.12 Configure Data Output Frequency 34
9.2.13 Configure baud rate 35
9.2.14 Configure CAN baud rate 35
9.2.15 Configure CAN output frequency 35
9.2.16 Print all configuration information 35
9.2.17 Query version number 35
9.2.18 Configure carrier XYZ velocity output 35
9.2.19 Removing gravitational acceleration from acceleration data 35
9.2.20 Configuration installation rotation angle 35
9.2.21 Save Parameter 36
9.3 Configuration via CAN 36
9.3.1 Check version number 36
9.3.2 Configure wheelbase and lever arm 36
9.3.3 Query wheelbase and lever arm 36
9.3.4 Configure CAN output Update Rate and baud rate 37
9.3.5 Query Update Rate and baud rate: 37
9.3.6 Configure rear wheel center lever arm 37
9.3.7 Configure projection points lever arm 37
9.3.8 Query projection points lever arm 37
9.3.9 Configure the Dual antenna installation misalignment angle 37
9.3.10 Query Dual antenna installation deviation angle 37
9.3.11 Configure low-pass filter 38
9.3.12 Query low-pass filter 38
9.3.13 Enable PPS(Standard Frame) 38
9.3.14 Disable PPS(Standard frame) 38
9.3.15 Save Parameter command 38
9.3.16 Configure IMU mounting rotation angle 38
9.3.17 Query configured IMU installation rotation angle 39
9.3.18 Configure whether Dual antenna fusion is enabled 39
10. Heading accuracy test **40**
11. Coordinate System Definition **41**
12. Firmware Upgrade **43**

12.1 via PC software 43
12.1.1 R232 Serial Port Direct Connection 43
12.1.2 Use the included USB-to-CAN module 43
12.2 serial push 44
13. Test Base Plate connection diagram 51
14. ROS driver (Integrated Navigation data stream) 52
14.1 Install ROS serial 52
14.2 Compile Code 52
14.3 Connect the IMU to the system via USB 53
14.4 View Data 53
15. For use with the front wheel gyroscope G200 55
16. Packaging 56
16.1 Tape and reel packaging 56
16.2 carrier tape 56
17. CRC Table Lookup Calculation 57
18. OptionalAccessories 59
19. FAQs List 60
20. Update Log 62

Forsense (Shanghai) Technology Co., Ltd

1. Overview

The IMU614E-AG inertial module uses IMU and GNSS data fusion algorithms to deliver low-cost, high-precision heading and attitude determination resistant to Magnetically-Robust interference. It is especially suited to single antenna controller solutions in the Self-driving Agricultural Machinery field.

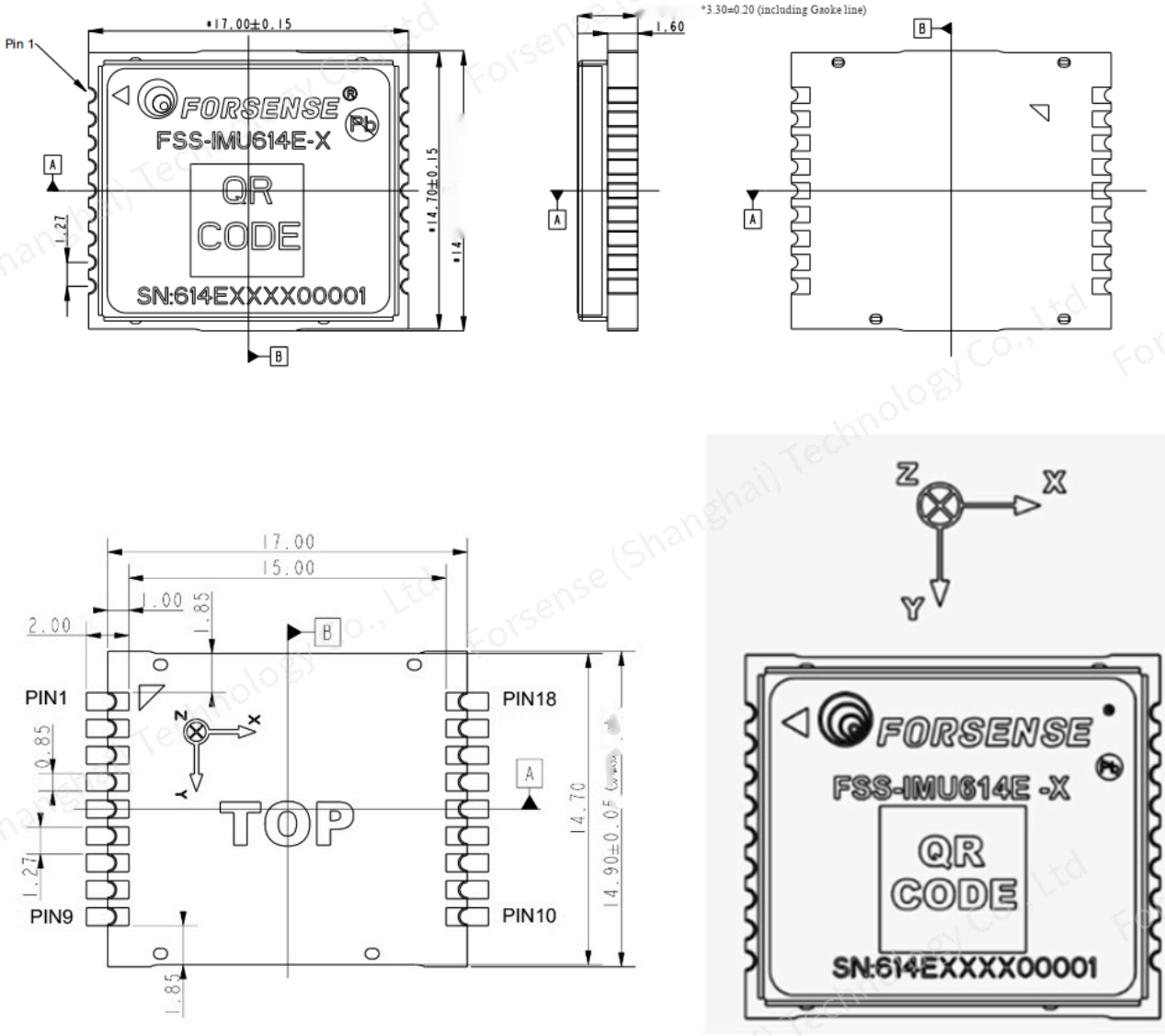
Table 1 Performance Specifications

Performance Specifications	**Attitude Accuracy**	**Roll/Pitch :<0.2° rms** **Heading:<0.3° rms** **(Vehicle speed greater than 1km/h, vehicle-mounted scenario with Ackermann steering structure)**
	Position Reckoning Accuracy	<2% @1σ (In-vehicle scenario, Satellite signal loss 30s, without wheel odometer integration)
	Update Rate	100Hz
	Gyro range	±500°/s
	Gyroscope Bias instability	XY:4°/h Z:3°/h @25℃, ALLAN variance, 1σ
	Accelerometer Range	±6g
	Accelerometer Bias instability	XY:20μg Z:40 μg @25℃, Allan variance, 1σ
Operating conditions	supply voltage	3.3V
	Power Consumption	0.085W
	Operating Temperature (Tot)	-40℃~85℃
	Storage Temperature (Tstg)	-40℃~85℃
	Supported connection protocols	Serial Port: TTL CAN (external circuit required)

2. Outline Structure

Installation requirement: the X-axis points toward the front of the vehicle, and the Y-axis points toward the right side of the vehicle body.

Figure 1 Outline structureand recommended pad dimensions(Unit: mm)



3. Electrical Characteristics

3.1 Maximum tolerance value

Table 2 Absolute Maximum Ratings

Parameter	Symbol	Minimum value	Maximum	Unit
supply voltage	VCC	-0.3	4.0	V
Ground (GND)	GND	-	-	-
Input Pin Voltage (Vin)	Vin	-0.3	VCC+0.2	V
Operating Temperature (Tot)	Tot	-40	85	℃
Storage Temperature (Tstg)	Tstg	-40	85	℃

3.2 Operating conditions

Table 3 Working Conditions

Parameter	Symbol	Minimum value	Typical Value	Maximum value	Unit
supply voltage	VCC	3.2	3.3	3.4	V
Maximum VCC ripple	Vrpp		±40		mV
Power Consumption	P		0.085		W
Operating Temperature (Tot)	T	-40		85	℃
Storage Temperature (Tstg)	T	-40		85	℃

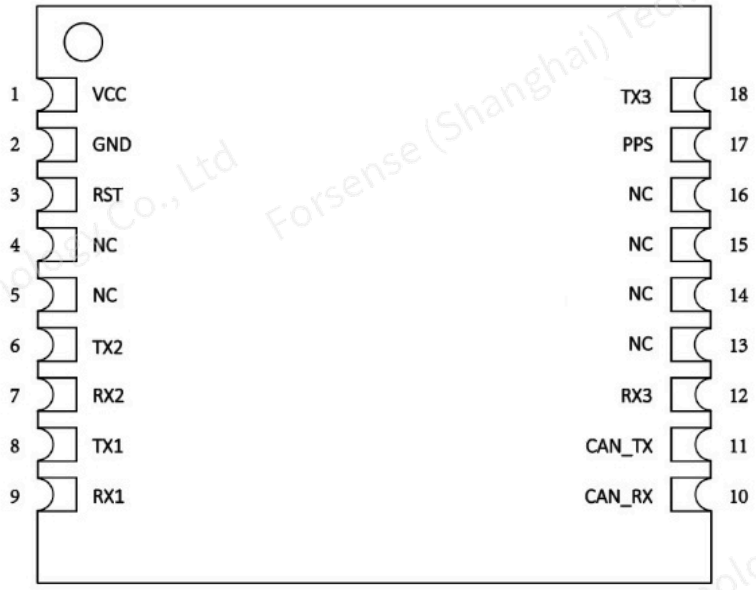
3.3 IO threshold characteristics

Table 4 IO Threshold Characteristics

Parameter	symbol	Minimum value	Typical value	Maximum value	Unit
Input Pin Low Level	Vin_low	0		VCC*0.2	V
Input Pin High Level	Vin_high	VCC*0.7		VCC+0.2	V
Output pin low power level	Vout_low	0		0.45	V
Output pin high power level	Vout_high	VCC-0.45		VCC	V

4. Pin Definition

Figure2 Pin schematic diagram



IMU614E- AG Pin Layout (Top View)

Table 5 Definition of Pin

Pin No.	Pin Name	Pin Description
1	VCC	Power input, +3.3V input, 40mA, ripple no greater than ±40mV
2	GND	Ground (GND)
3	RST	NoneRequiredConnection
4	NC	Connectionless
5	NC	Connectionless
6	TX2	Receive asynchronous data output
7	RX2	Receive asynchronous data input
8	TX1	Receive asynchronous data output (data communication interface (LVTTL))
9	RX1	Receive asynchronous data input (data communication interface (LVTTL))
10	CAN_RX	CAN receive Pin; read data from the bus to the CAN controller
11	CAN_TX	CAN send Pin; read data from the CAN controller to bus driver
12	RX3	Receive asynchronous data input
13	Reserved	Reserved
14	Reserved	Reserved
15	Reserved	Reserved
16	Reserved	Reserved
17	PPS	External synchronous sampling trigger signal; (connect to the RTK pulse-per-second pin)
18	TX3	Receive asynchronous data output

For hardware design information related to the module, refer to the document《IMU614E-XX Hardware Design Manual》.

5. Recommended soldering furnace temperature profile

Figure 3 Welding Furnace Temperature Curve

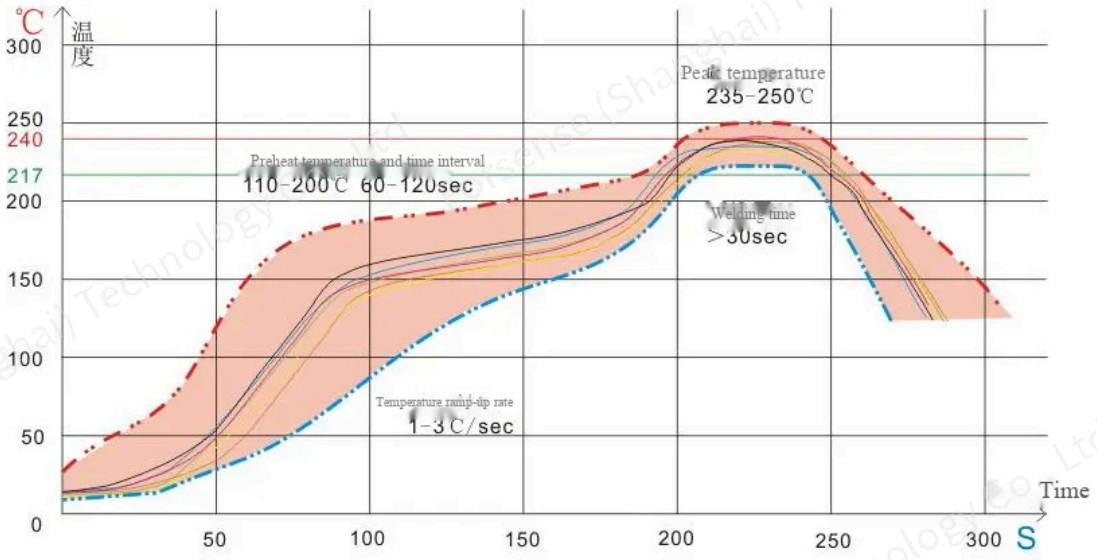


Table 6 Temperature Setting Mode

Item	minimum limit	upper limit	Unit
Maximum temperature rise slope (target=0.8) (Time interval for slope calculation = 60 seconds)	1	3	Degrees/second
Maximum temperature drop slope (time distance for slope calculation = 60 s)	-3	-1	degrees/second
Preheating temperature and time range	60	120	second
Reflow time (period above 217℃)	40	70	Second
Maximum temperature	235	250	Celsius
Maximum Reflow Cycles		1	times

For more SMT information about the module, refer to the document《Forsense-LCC Module_SMT Application Guide》.

Precautions:

- For module soldering reflow, it is recommended to use eight temperature zones or higher reflow soldering equipment;
- Since the module is a high-precision sensor product, it is relatively sensitive to any deformation:
- If the PCB thickness is less than 1.0 mm, it is recommended to make reflow fixture to prevent the PCB from deforming at high temperatures and affecting the coplanarity of soldering.
- Customers are advised to use high-Tg laminate for PCB mainboards to prevent board deformation during high-temperature reflow, which can cause warpage, extrusion, open solder, and solder bridging defects.
- Due to sensitive components inside the module, the maximum temperature of the customer's reflow soldering machine must not exceed 260℃ (referring to the package top temperature measured on the package body surface).
- It is recommended to use lead-free no-clean solder paste. Recommended solder paste brand/model: Alpha OM-338 SAC305 Sn96.5Ag3.0Cu0.5
- The module contains sensitive components, so secondary reflow should be avoided to prevent degradation of module performance;
- Cooling:

- Controlled cooling ramp rate prevents adverse soldering effects (solder joints becoming more brittle) and mechanical stress inside the product; controlled cooling helps achieve a bright solder surface, fine crystalline grains, and a low contact angle, avoiding shield cover warpage caused by rapid cooling changes.
- Visual Inspection:
- After module soldering, inspect the solder quality using X-ray and optical magnification inspection methods; for details, follow the relevant IPC-A-610F standards.
- **When soldering with a soldering iron, control the temperature between 260℃ and 290℃; do not exceed 3s per soldering operation, and take anti-static measures;**

Forsense (Shanghai) Technology Co., Ltd

6. ESD protection



Static electricity can cause intermittent or permanent circuit damage and is highly harmful to electronic products; analysis shows that most cases are ESD damage;

Therefore, ESD protection for the module is particularly important. Production and transportation processes must strictly follow ESD protection procedures and comply with the following conditions:

- Never touch the module with bare hands, especially at the Pin location.
- SMT placement machines, workbenches, electric soldering irons, and other equipment must be grounded.
- Operators must wear an antistatic wrist strap with a proper grounding wire (cordless antistatic wrist straps must not be used; antistatic gloves are recommended).
- Packaging and PCB must be qualified ESD-safe materials.

7. Usage example

7.1 Device Installation

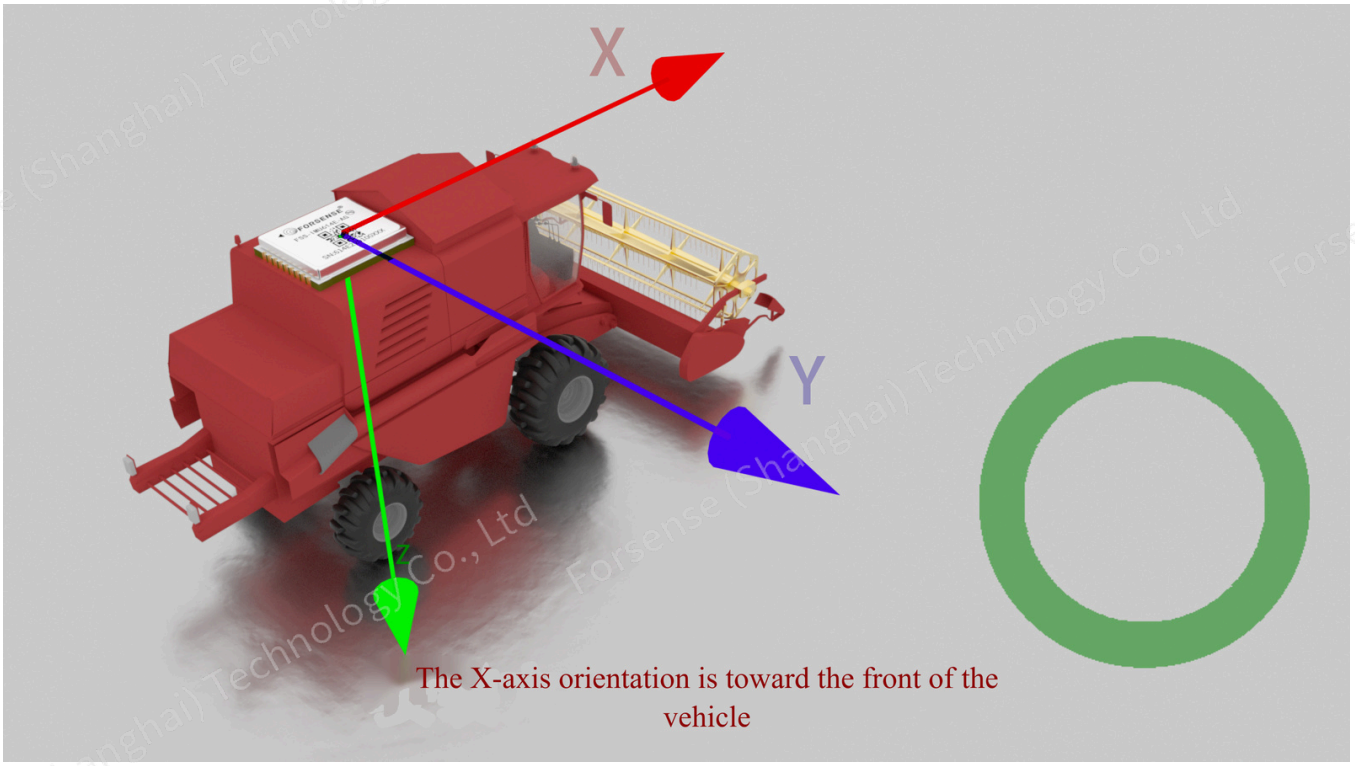
1. The module should be securely fixed to a rigid surface and avoid installation in high-vibration locations.

2. The module installation orientation should maintain the relationship with the vehicle head direction as described in the figure below.

The correct installation diagram is shown below.

X-axis faces the vehicle front

Figure 4 Correct installation diagram

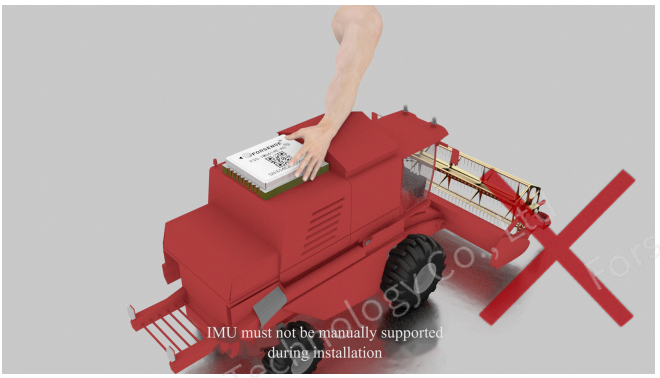


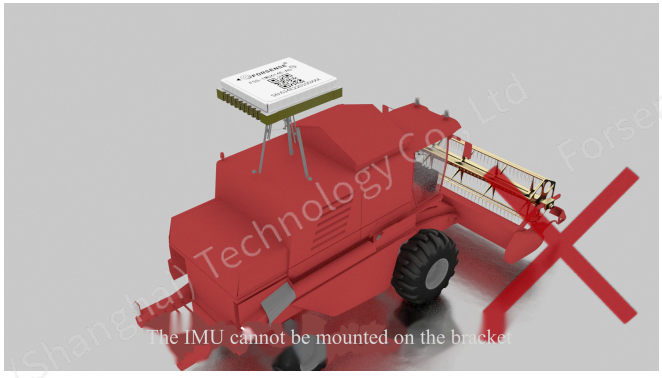
All the following installation methods areIncorrect installation

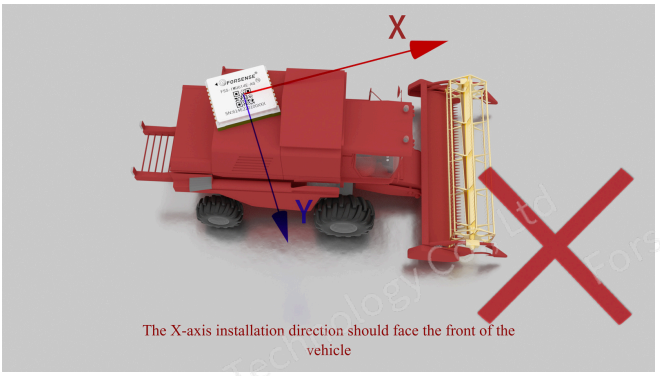










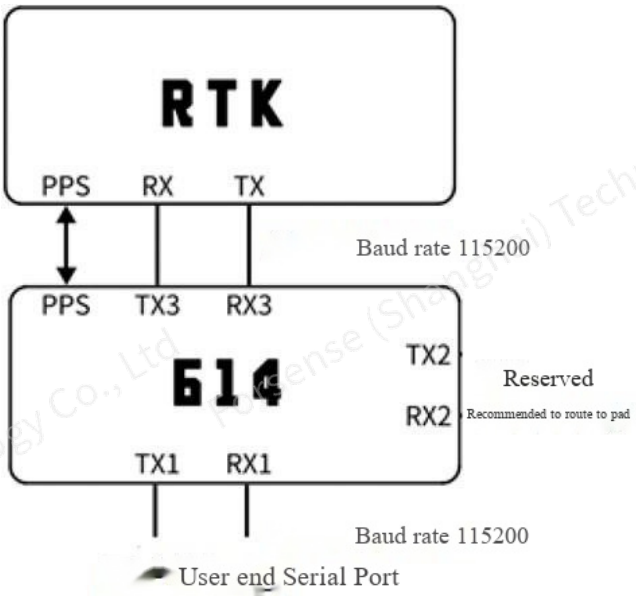




7.2 Connect RTK board

7.2.1 RTK board and the AG moduleRecommended connection method

Figure 5 Connection diagram



7.2.2 RTK Configuration Requirements

Currently, two protocols can be configured: NovAtel and NMEA

NovAtel Protocol:

Configuration command: AT+GNSS_CARD=UNICORE\r\n

Message input requirements:

BESTPOSB 10hz

PSRVELB 10hz

GPGGA 10hz

HEADINGB 10hz (no configuration needed when using single antenna)

Disable the RTK board's inertial navigation assistance-related functions; disable other sentences; baud rate 115200.

pulse per second (PPS): once every 1s, triggered by Rising Edge, pulse width 5ms, aligned to UTC time. High power level must not exceed 5v.

NMEA protocol

Configuration command: AT+GNSS_CARD=OEM\r\n

Message input requirements:

GPGGA 10hz

GPRMC 10hz

HEADINGB 10hz (can be left unconfigured when using single antenna)

Disable the inertial navigation assistance-related functions of the RTK board and disable other sentences. Baud rate 115200.

pulse per second (PPS): once every 1s, triggered by Rising Edge, pulse width 5ms, aligned to UTC time. High power level must not exceed 5v.

7.2.3 RTK differential data import

The user must use user Master to log in to the network CORS account of Qianxun SI or Sixents Technology. RTK board outputs GPGGA messages to the differential server. After the differential server returns the differential data, it forwards it to RTK board, so that RTK enters the fixed solution state.

7.3 Parameter Configuration

7.3.1 Configure the Primary antenna lever arm

For example, configure lever arm vector as X=0.5m, Y=-0.6m, Z=-1.0m

Command: AT+CLUB_VECTOR=0.5,-0.6,-1.0\r\n

Response: GPS_POS_X=0.5,GPS_POS_Y=-0.6,GPS_POS_Z=-1.0\r\n

Note: lever arm vector is the three-dimensional vector (X,Y,Z) of the Primary antenna phase center of the RTK relative to the IMU phase center, in meters. Where,

in the forward-right-down vehicle body coordinate system

If the RTK Primary antenna is in front of the IMU, it is a positive value; otherwise, it is a negative value;

If the RTK Primary antenna is to the right of the IMU, the value is positive; otherwise, it is negative;

If RTK Primary antenna is above the IMU, it is**a negative number**, otherwise it is positive (the antenna is generally above the device).

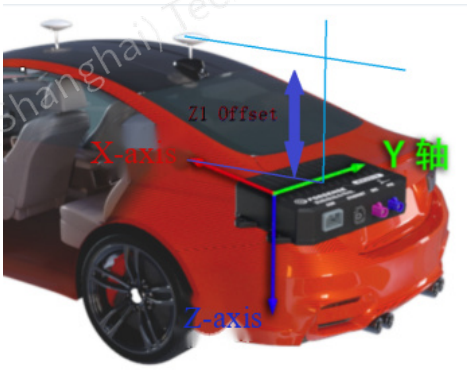
The coordinate system diagram is shown below: (the label must face up; if the IMU is not installed as shown below, the installation orientation must be configured.)

Figure6 Coordinate System Schematic Diagram



Fig.7 Antenna lever arm diagram





7.3.2 Configure rear wheel axle center lever arm

For example, configure lever arm vector to X=0.5m, Y=-0.6m, Z=1.0m

Command: AT+OD_VECTOR=0.5,-0.6,1.0\r\n

Response: OD_POS_X=0.5,OD_POS_Y=-0.6,OD_POS_Z=1.0\r\n

Note: lever arm vector is the three-dimensional vector (X, Y, Z) of the rear wheel axle center relative to the IMU phase center, in meters. where,

in the front-right-down vehicle body coordinate system

If the rear axle center is in front of the IMU, the value is positive; otherwise, it is negative;

If the rear axle center is to the right of the IMU, the value is positive; otherwise, it is negative;

If the rear wheel axle center is below the IMU, it is**positive**, otherwise it is negative (generally, the rear wheel axle center is below the device).

The coordinate system diagram is shown below: (The label must face upward. If the IMU is not installed as shown below, configure the mounting orientation.)



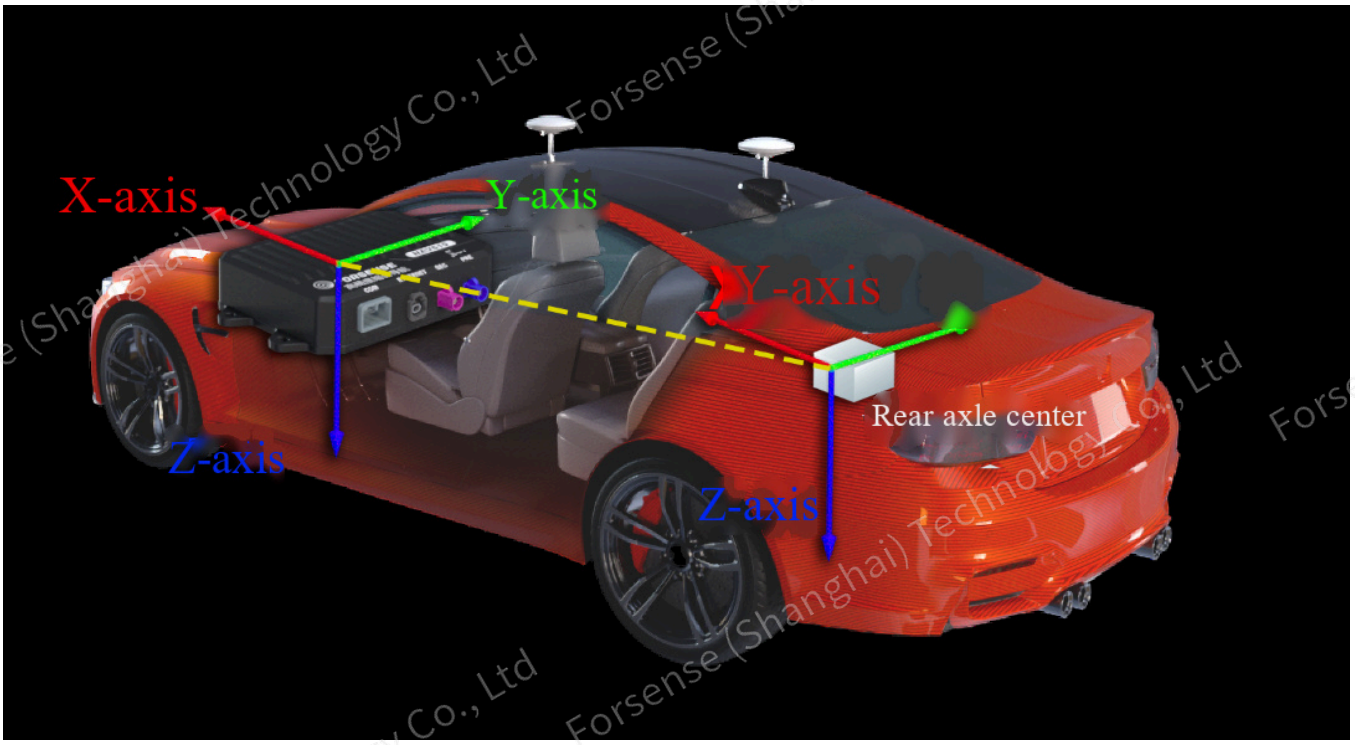


Figure8 Coordinate system schematic

7.3.3 Configure vehicle wheelbase

If the configured vehicle wheelbase is 2m, the configuration command is:

AT+WHEEL_BASE=2\r\n

Note: Save the configuration after setting the commands

7.3.4 Configure RTK Dual antenna installation angle

If the RTK Dual antenna installation angle is configured to 0 degrees, the configuration command is:

Command: AT+RTK_ANGLE=0\r\n

Response: ANGLE=0\r\n

Note: The installation angle is the angle between the ray from Primary antenna to Secondary antenna and the vehicle front direction; clockwise is positive, counterclockwise is negative; the angle input range is -180°~180°

Note: The configuration command takes effect only after it is saved and the device is power-cycled; the spacing distance between Dual antenna must be greater than 50cm.

7.3.5 Enable Dual antenna fusion

Note: By default, Dual antenna fusion is disabled and is used only for rapid initialize when stationary. To use Dual antenna heading throughout, enable Dual antenna fusion with the following command: 1 enables Dual antenna fusion, while any other Parameter disables Dual antenna fusion. Note that the Dual antenna installation error must be calibrated at the same time; otherwise, a fixed error will exist relative to the vehicle heading.

Command :AT+DRTK_USE=1\r\n

Response: OK

Note: After saving the configuration commands, a power cycle is required for them to take effect; the spacing between Dual antenna must be greater than 50cm

7.3.6 Calibration Dual antenna

Dual antenna standardThe process is defined as follows

1.Connect Serial Port, enter AT+SETNO\r\n to disable all printing

2. Send command AT+RTK_BIAS_EST=1\r\n to start the calibration process

3. The vehicle drives straight, with vehicle speed not less than 3km/h; calibration is successful after Serial Port returns RTK_BIASE_CORRECT_DONE

4. Send the AT+CONFIG\r\n command to see the printout, RTK_BIAS_FLAG_AND_VALUE=99,XX (calibration angle). Check whether it is reasonable; under normal installation, it is less than 1 degree.

5.AT+SAVE Save the result, then power cycle.

7.4 Save Parameter

After all configuration commands have been set, send the save Parameter command “AT+SAVE\r\n”.

8. Output Protocol

8.1 Binary Protocol - AG Data Stream

Note:

- The CRC check is the CRC verification of all bytes of the frame starting from Frame header, excluding the CRC check bit itself. See the appendix for the calculation method and example routine.
- The frame length is the total number of all data bytes excluding Frame header, the frame ID, the frame length, and the checksum bits.
- little-endian mode, low byte first.

The configuration instruction is:

Command: AT+SETAG

Response: OK\r\n

Table 7 Binary Protocol - AG Data Stream

Content	Type	Relative position
Frame header 1: 0xAA	Uint8	0
Frame header 2: 0x55	Uint8	1
Frame ID: 0x0156	Uint16	2
Frame length: 0x0032	Uint16	4
seconds in GPS week (ms)	Uint32	6
Roll angle (degrees)	Float	10
Pitch angle(degrees)	Float	14
Yaw angle (degrees)	Float	18
Reserved	Float	22
Three-axis Angular velocity (deg/s, Forward-Right-Down)	Float*3	26
Three-axis acceleration (g, front-right-down)	Float*3	38
RTK positioning status	Uint8	50
Attitude significant bits Bit0: 1 indicates valid attitude, 0 invalid Bit1: 1 indicates the front wheel steering angle is valid, 0 indicates invalid	Uint8	51

Status bits: bit0: 1 indicates RTK data is valid, 0 indicates invalid Bit1: 1 indicates PPS signal is valid, 0 indicates invalid Bit2: 1 indicates Integrated Navigation is initialize, 0 indicates it is not initialize (single antenna initialize conditions: RTK positioning status 4 + speed reaches 0.5m/sor above + straight-line driving with acceleration/deceleration, Dual antenna initialize condition: RTK positioning status 4 + RTK Orientation status 50) Bit3: 1 means the front wheel steering angle is valid, 0 invalid Bit4: 1 indicates Integrated Navigation has converged, 0 indicates not converged Bit5: 1 indicates the front wheel gyroscope data is valid, 0 indicates invalid (can be ignored if no external gyroscope is connected) Bit6: 1 indicates the steering wheel motor data is valid, 0 indicates invalid Bit7、Bit8： 01 indicates the vehicle is moving forward (Bit7=1, Bit8=0) 10 indicates vehicle reversing (Bit7=0, Bit8=1) 00 indicates invalid (Bit7=0, Bit8=0)	Uint32	52
CRC check	Uint32	56

8.2 Binary Protocol - Integrated Navigation Data Stream

Note:

- The CRC is calculated over all bytes of the frame starting from Frame header, excluding the CRC bits themselves. See the appendix for the calculation method and example routine.
- The frame length is the total number of all data bytes excluding Frame header, the frame ID, the frame length, and the checksum bits.
- little-endian mode, send the low byte first

The configuration command is:

Command: AT+SETNAV

Response: OK

If configured not to output, the configuration command is:

Command: AT+SETNO\r\n

Response: OK

Table 8 Binary Protocol-Integrated Navigation Data Stream

Content	Type	Relative position
Frame header 1：0xAA	Uint8	0
Frame header 2:0x55	Uint8	1
Frame ID: 0x0166	Uint16	2
Frame length: 0x005E	Uint16	4
seconds in GPS week (ms)	Uint32	6
GPS week number count	Uint16	10
Latitude (degrees × 10000000)	Int32	12
Longitude (degrees × 10000000)	Int32	16
Height (mm)	Int32	20
North velocity (m/s)	Float	24
East velocity (m/s	Float	28
Down velocity (m/s)	Float	32

Roll angle (degrees)	Float	36
Pitch angle(degrees)	Float	40
Yaw angle(degrees)	Float	44
In case of single antenna: AHRS heading(degrees)(no reference value) In case of Dual antenna: RTK Dual antenna heading (degrees) With G200 front wheel gyro connected: front wheel gyro Z-axis Angular velocity(deg/s)	Float	48
Reserved (displayed as front wheel angle when G200 is connected)	Float	52
Accelerometer X-axis (g)	Float	56
Accelerometer Y-axis (g)	Float	60
Accelerometer Z-axis (g)	Float	64
Gyroscope X-axis (deg/s)	Float	68
Gyroscope Y-axis (deg/s)	Float	72
Gyroscope Z-axis (deg/s)	Float	76
IMU Temperature (°C)	Float	80
RTK positioning status (same as the positioning status in GGA) 0: No fix 1:single point positioning 2:Pseudo-range differential positioning 4:fixed solution 5:floating solution	Uint8	84
Number of satellites	Uint8	85
differential delay	Uint8	86
Dual antenna Orientation status 50 indicates oriented Other values indicate not oriented	Uint8	87
Position Dilution of Precision (DOP) (cm) Valid after Integrated Navigation initialize	Uint16	88
Status bits: bit0:1 indicates valid RTK data, 0 indicates invalid Bit1:1 indicates the PPS signal is valid, 0 indicates invalid Bit2:1 indicates Integrated Navigation is initialize, 0 indicates it is not initialize (single antenna initialize condition: RTK positioning status 4 + speed reaches 0.5m/sThe above + straight-line driving with acceleration and deceleration, Dual antenna initialize conditions: RTK positioning status 4 + RTK Orientation status 50) Bit3: 1 means the front wheel steering angle is valid, 0 invalid. Bit4: 1 indicates Integrated Navigation has converged, 0 indicates not converged Bit5:1 indicates the front wheel gyro data is valid; 0 indicates invalid (can be ignored if no external gyro is connected). Bit6: 1 indicates the steering wheel motor data is valid, 0 indicates invalid Bit7、Bit8： 01 indicates the vehicle is moving forward (Bit7=1, Bit8=0) 10 indicates vehicle reversing (Bit7=0, Bit8=1) 00 indicates invalid (Bit7=0, Bit8=0)	Uint16	90
Reserved 1	Uint32	92
Reserved 2	Uint32	96
CRC check	Uint32	100

8.3 NMEA protocol

- Supports outputting the combined data in NMEA format,
- It cannot be output at the same time as the binary data stream. If the NMEA data stream is output, the binary data stream cannot be output. Before switching data streams, first9.2.11send a command to stop the current data stream output.
- The following statements are currently supported. Configuration methodSee 9.2.10Chapter

GPGGA

Chinese explanation of the message

Field	Structure	Description	Style	Example
1	$GPGGA	Message Header		$GPGGA
2	utc	UTC Time (Hour/Minute/Second/Fractional Seconds)	hhmmss.ss	202134
3	lat	Latitude (DDmm.mm)	llll.ll	5106.9847
4	lat dir	Latitude direction (N = North, S = South)	a	N
5	lon	Longitude (DDDmm.mm)	yyyyy.yy	11402.2986
6	lon dir	Longitude direction (E = East, W = West)	a	W
7	quality	Solution status	x	1
8	# sats	Number of satellites involved in the solution, which may differ from the number of visible satellites	xx	2
9	hdop	Horizontal Dilution of Precision (DOP)	x.x	1
10	alt	Antenna height, above or below mean sea level	x.x	1062.22
11	a-units	Antenna height unit, (M = metres)	M	M
12	undulation	Elevation outlier	x.x	-16.271
13	u-units	Elevation outlier unit (M = metres)	M	M
14	age	Differential data age, s	xx	1
15	stn ID	Differential base station ID	xxxx	8
16	*xx	Checksum	*hh	*48
17	[CR][LF]	Statement terminator		[CR][LF]

Official message interpretation

Field	Structure	Description	Symbol	Example
1	$GPGGA	Log header		$GPGGA
2	utc	UTC time status of position (hours/minutes/seconds/ decimal seconds)	hhmmss.ss	202134.00
3	lat	Latitude (DDmm.mm)	llll.ll	5106.9847
4	lat dir	Latitude direction (N = North, S = South)	a	N
5	lon	Longitude (DDDmm.mm)	yyyyy.yy	11402.2986
6	lon dir	Longitude direction (E = East, W = West)	a	W
7	quality	refer to Table: GPS Quality Indicators	x	1
8	# sats	Number of satellites in use. May be different to the number in view	xx	10
9	hdop	Horizontal dilution of precision	x.x	1.0
10	alt	Antenna altitude above/below mean sea level	x.x	1062.22
11	a-units	Units of antenna altitude (M = metres)	M	M
12	undulation	Undulation - the relationship between the geoid and the WGS84 ellipsoid	x.x	-16.271
13	u-units	Units of undulation (M = metres)	M	M
14	age	Age of correction data (in seconds) The maximum age reported here is limited to 99 seconds.	xx	(empty when no differential data is present)
15	stn ID	Differential base station ID	xxxx	(empty when no differential data is present)
16	*xx	Check sum	*hh	*48
17	[CR][LF]	Sentence terminator		[CR][LF]

GPS Quality Indicators

Indicator	Description
0	Fix not available or invalid
1	Single point
	Converging PPP (TerraStar-L)
2	Pseudorange differential
	Converged PPP (TerraStar-L)
	Converging PPP (TerraStar-C PRO, TerraStar-X)
4	RTK fixed ambiguity solution
5	RTK floating ambiguity solution
	Converged PPP (TerraStar-C PRO, TerraStar-X)
6	Dead reckoning mode
7	Manual input mode (fixed position)
8	Simulator mode
9	WAAS (SBAS)[1]

GPRMC

Message description in Chinese

Field	Structure	Description	Style	Example
1	$GPRMC	Message header		$GPRMC
2	utc	Fixed UTC time	hhmmss.ss	144326
3	pos status	Positioning status (A indicates valid positioning, V indicates invalid positioning)	A	A
4	lat	Latitude (DDmm.mm)	llll.ll	5107.0017737
5	lat dir	Latitude direction (N = North, S = South)	a	N
6	lon	Longitude (DDDmm.mm)	yyyyy.yy	11402.3291611
7	lon dir	Longitude direction (E = East, W = West)	a	W
8	speed Kn	Ground speed	x.x	0.08
9	track true	Ground track, referenced to true north	x.x	323.3

10	date	UTC date: (year/month/day)	xxxxxx	210307
11	mag var	Magnetic declination	x.x	0
12	var dir	Magnetic declination direction	a	E
13	mode ind	Positioning mode prompt	a	A
14	*xx	Checksum	*hh	*20
15	[CR][LF]	Sentence terminator		[CR][LF]

Official Message Explanation

Field	Structure	Description	Symbol	Example
1	$GPRMC	Log header		$GPRMC
2	utc	UTC of position	hhmmss.ss	144326.00
3	pos status	Position status (A = data valid, V = data invalid)	A	A
4	lat	Latitude (DDmm.mm)	llll.ll	5107.0017737
5	lat dir	Latitude direction: (N = North, S = South)	a	N
6	lon	Longitude (DDDmm.mm)	yyyyy.yy	11402.3291611
7	lon dir	Longitude direction: (E = East, W = West)	a	W
8	speed Kn	Speed over ground, knots	x.x	0.080
9	track true	Track made good, degrees True	x.x	323.3
10	date	Date: dd/mm/yy	xxxxxx	210307
11	mag var	Magnetic variation, degrees Note that this field is the actual magnetic variation and will always be positive. The direction of the magnetic variation is always positive.	x.x	0.0
12	var dir	Magnetic variation direction E/W Easterly variation (E) subtracts from True course. Westerly variation (W) adds to True course.	a	E
13	mode ind	Positioning system mode indicator, see Table: NMEA Positioning System Mode Indicator	a	A
14	*xx	Check sum	*hh	*20
15	[CR][LF]	Sentence terminator		[CR][LF]

NMEA Positioning System Mode Indicator

Mode	Indicator
A	Autonomous
D	Differential
E	Estimated (dead reckoning) mode
M	Manual input
N	Data not valid

GPHDT (heading information)

Chinese Explanation of the Message

Field	Structure	Description	Style	Example
1	$GPHDT	Message Header		$GPHDT
2	heading	Yaw angle	x.x	75.5554
3	True	True North Angle	T	T
4	*xx	Checksum	*hh	*36
5	[CR][LF]	Sentence terminator		[CR][LF]

Official message interpretation

Field	Structure	Description	Symbol	Example
1	$GPHDT	Log header		$GPHDT
2	heading	Heading in degrees	x.x	75.5554
3	True	Degrees True	T	T
4	*xx	Check sum	*hh	*36
5	[CR][LF]	Sentence terminator		[CR][LF]

GPVTG (Ground Speed Information)

Chinese explanation of the message

Field	Structure	Description	Style	Example
1	$GPVTG	Message header		$GPVTG
2	track true	Orientation angle, referenced to true north	x.x	24.168
3	T	True north indicator	T	T
4	track mag	Orientation angle, referenced to magnetic north	x.x	24.168
5	M	Magnetic north indicator	M	M
6	speed Kn	Horizontal motion speed	x.x	0.4220347
7	N	Speed indicator, knots	N	N
8	speed Km	Horizontal motion speed, km/s	x.x	0.781608
9	K	Speed indicator (K = km/hr)	K	K
10	mode ind	Positioning mode indicator	a	A
11	*xx	Verification value	*hh	*7A
12	[CR][LF]	sentence terminator		[CR][LF]

Official Message Interpretation

Field	Structure	Description	Symbol	Example
1	$GPVTG	Log header		$GPVTG
2	track true	Track made good, degrees True	x.x	24.168
3	T	True track indicator	T	T
4	track mag	Track made good, degrees Magnetic; Track mag = Track true + (MAGVAR correction) See the **MAGVAR** command	x.x	24.168
5	M	Magnetic track indicator	M	M
6	speed Kn	Speed over ground, knots	x.x	0.4220347
7	N	Nautical speed indicator (N = Knots)	N	N
8	speed Km	Speed, kilometres/hour	x.x	0.781608
9	K	Speed indicator (K = km/hr)	K	K
10	mode ind	Positioning system mode indicator, see Table: NMEA Positioning System Mode Indicator	a	A
11	*xx	Check sum	*hh	*7A
12	[CR][LF]	Sentence terminator		[CR][LF]

NMEA Positioning System Mode Indicator

Mode	Indicator
A	Autonomous
D	Differential
E	Estimated (dead reckoning) mode
M	Manual input
N	Data not valid

GPZDA (UTC time and date)

Chinese explanation of the message

Field	Structure	Description	Style	Example
1	$GPZDA	Message Header		$GPZDA
2	utc	UTC Time	hhmmss.ss	220238
3	day	Day, 01 - 31	xx	15
4	month	Month, 01 - 12	xx	7
5	year	Year	xxxx	1992
6	null	Local area code, not available	xx	(empty when no data is present)
7	null	Local area minute description, not available	xx	(empty when no data is present)
8	*xx	Checksum	*hh	*6F
9	[CR][LF]	Statement terminator		[CR][LF]

Official Message Explanation

Field	Structure	Description	Symbol	Example
1	$GPZDA	Log header		$GPZDA
2	utc	UTC time status	hhmmss.ss	220238.00
3	day	Day, 01 to 31	xx	15
4	month	Month, 01 to 12	xx	07
5	year	Year	xxxx	1992
6	null	Local zone description—not available Local time zones are not supported by OEM7 family receivers. Fields 6 and 7 are always null.	xx	(empty when no data is present)
7	null	Local zone minutes description—not available	xx	(empty when no data is present)
8	*xx	Check sum	*hh	*6F
9	[CR][LF]	Sentence terminator		[CR][LF]

The GPATT format is shown in the table below.

Name	Unit	Style	Example	Description
Message header		String	$GNATT	
UTC time		hhmmss.sss	170834.000	17：08：34 UTC
State		Character	1	0: Unavailable 1: Available
Roll angle	degree	3 decimal place	-4.891	The range is ±180°, with the right side defined as downward. is positive
Roll angle Indicator		Character	R	Roll angle indicator
Pitch angle	degree	3 decimal place	3.122	Range is ±90°, defined upward. Is positive
Pitch angle indicator		Character	P	Pitch angle indicator
Yaw angle	degree	3 decimal place	124.005	Range is 0° to 360°, referenced to true north, with clockwise rotation positive
Roll angle uncertainty	degree	3 decimal place	0.432	The range is 0~360
Pitch angle Uncertainty	degree	3 decimal place	0.811	Range is 0~360
Yaw angle uncertainty	degree	3 decimal place	1.202	Range is 0~360
Checksum		HEX	*68	Used to check transmission errors

Name	*Unit*	*Format*	*Example*	*Description*
Sentence Identifier		String	$GNATT	
Time		hhmmss.sss	170834.000	17:08:34 UTC
Status		Character	1	0: invalid 1: valid
Roll Angle	degree	3 decimal places	-4.891	range ±90, right side down defined as positive
Indicator for roll		character	R	Roll indicator
Pitch Angle	degree	3 decimal places	3.122	range ±90, head up defined as positive
Indicator for Pitch		character	P	Pitch indicator
Heading Angle	degree	3 decimal places	124.005	range 0~360, to true North, counter clockwise defined as positive
Roll Angle uncertainty	degree	3 decimal places	0.432	range 0~360
Pitch Angle uncertainty	degree	3 decimal places	0.811	range 0~360
Heading Angle uncertainty	degree	3 decimal places	1.202	range 0~360
Checksum		Hex	*68	Used by program to check for transmission errors

GPCHC (Forsense custom message: GPCHC data protocol)

The GPCHC format is shown in the table below.

$GPCHC,GPSWeek,GPSTime,Heading,Pitch,Roll,gyrox,gyroy,gyroz,accx,accy,accz,Lattitude,Longitude,Altitude,Ve,Vn,Vu,V,NSV1,NSV2,Status,Age,WarmingCs<CR><LF>

Note: If the system status does not contain initialize, it is the GNSS result, including position, velocity, and attitude (Status low nibble: 0 and 1 are GNSS data; 2 and 3 are combined data).

Field	Name	Description	Format	For example
1	Header	GPCHC protocol header	$GPCHC	$GPCHC
2	GPSWeek	Week number from 1980-1-6 to present (GPS time)	www	1980
3	GPSTime	Seconds from Sunday 0:00:00 of the current week to the present (GPS time)	ssssss.ss	16897.68
4	Heading	Yaw angle (0 to 359.99)	hhh.hh	289.19
5	Pitch	Pitch angle (-90 to 90)	+/-pp.pp	-0.42
6	Roll	Roll angle (-180 to 180)	+/-rrr.rr	0.21
7	gyro x	Gyro X-axis, unit (°/s)	+/-ggg.gg	-0.23
8	gyro y	Gyro Y-axis, unit (°/s)	+/-ggg.gg	0.07
9	gyro z	Gyro Z-axis, unit (°/s)	+/-ggg.gg	-0.06
10	acc x	Accelerometer X axis, unit (G)	+/-a.aaaa	0.0009
11	acc y	Accelerometer Y-axis, unit (G)	+/-a.aaaa	0.0048
12	acc z	Accelerometer Z axis, unit (G)	+/-a.aaaa	-1.0037
13	Latitude	Latitude (-90° to 90°)	+/-ll.lllllll	38.8594969
14	Longitude	Longitude (-180° to 180°)	+/-ll.lllllll	121.5150073
15	Altitude	Height, unit (meters)	+/-aaaaa.aa	121.51
16	Ve	East velocity, unit (m/s)	+/-eee.eee	-0.023
17	Vn	Northward velocity, unit (m/s)	+/-nnn.nnn	0.011
18	Vu	Vertical velocity, unit (m/s)	+/-uuu.uuu	0.000
19	V	Vehicle speed, in (m/s)	+/-uuu.uuu	1.500
20	NSV1	Primary antenna 1 Satellite count	nn	14
21	NSV2	Secondary antenna 2 satellite count	nn	6
22	Status	System status (low nibble): 0: initialize ; 1: Satellite navigation mode ; 2: Integrated Navigation mode ; 3: Pure inertial navigation mode ; Satellite status (high nibble): 0: No positioning, no heading; 1: single point positioning heading; 2: Pseudo-range differential positioning heading; 3: Integrated dead reckoning; 4: RTK stable solution positioning and heading; 5: RTK floating solution positioning and heading; 6: single point positioning no heading; 7: Pseudo-range differential positioning no heading; 8: RTK stable solution positioning, no heading; 9: RTK floating solution positioning, no heading;	ss	4
23	Age	differential delay	aa	0
24	Warning	bit0:1:No GPS message, 0:Normal bit1:1:No vehicle message, 0:Normal bit3:1 gyroscope error, 0: normal bit4:1 Accelerometer error, 0: normal	ww	2
25	Cs	Verification	*hh	*47
26	<CR><LF>	Fixed packet trailer		<CR><LF>

8.4 CAN protocol

Table 9 CAN extended frame format 0x19FF CC9A (note: the unit is deg*100, int16 type)

Extended frame ID	1	2	3	4	5	6	7	8
0x19FF CC9A	Roll angle		Pitch angle		Yaw angle		track angle	

Table 10CAN extended frame format 0x19FF CD9A (Note: unit is deg/s*52.0127, int16 type)

Note: Reserved: if connected to G200, output the front wheel angle

Extended Frame ID	1	2	3	4	5	6	7	8
0x19FF CD9A	gyro_x		gyro_y		gyro_z		Reserved	

Table 11 CAN extended frame format 0x19FF CE9A (Note: unit is g*3276.8, int16 type)

Status bits:

Bit0: RTK board data valid flag bit, 1 indicates valid, 0 indicates invalid

Bit1:PPS valid flag bit, 1 indicates valid, 0 indicates invalid

Bit2: front wheel steering angle valid flag bit, 1 indicates valid, 0 indicates invalid

Bit3: Front wheel Gyroscope valid flag bit, 1 indicates valid, 0 indicates invalid

Bit4: RTK positioning fixed solution flag, 1 indicates fixed solution, 0 indicates not fixed solution

Bit5: RTK orientation fixed solution flag, 1 indicates fixed solution, 0 indicates not fixed solution

Bit6: Heading initialize flag, 1 indicates initialize completed, 0 indicates initialize not completed (only for single antenna body gyro)

Extended Frame ID	1	2	3	4	5	6	7	8
0x19FF CE9A	accel_x		accel_y		accel_z		Status bits	

Table12CAN extended frame format 0X19FFCE9B

Note: Speed direction: 1: forward, -1: backward

Extended Frame ID	1	2	3	4	5	6	7	8
0X19FFCE9B	seconds of week				Estimate antenna installation deviation angle		Velocity direction	Reserved

Table13CAN extended frame format 0X19CCFF9A (note: the format is in degrees and must be divided by 1e7, Int32 type)

The device uses the WGS84 coordinate system. By default, longitude and latitude are output as the Primary antenna phase center position; when projection points is configured, they are output as the projection points position;

Latitude is 0° at the equator; a latitude greater than 0° indicates the Northern Hemisphere, and vice versa indicates the Southern Hemisphere;

Longitude is measured with the prime meridian as 0°. A Longitude greater than 0° is in the Eastern Hemisphere; otherwise, it is in the Western Hemisphere.

Extended Frame ID	1	2	3	4	5	6	7	8
0X19CCFF9A	Latitudedegree				LongitudeDegree			

Table14 CAN extended frame format 0X19CCFF9B (Note: Elevation is Ellipsoidal height, divide by 1e7, Int32 type)

Extended Frame ID	1	2	3	4	5	6	7	8
19CCFF9B	Elevation				seconds of week			

Table15 CAN extended frame format 0X19FFD29A (Note: unit is cm/s, int16 type)

Extended Frame ID	1	2	3	4	5	6	7	8
19FFD29A	Northward velocity		East velocity		downward velocity		Ground speed (planar speed)	

Forsense (Shanghai) Technology Co., Ltd

9. Input Protocol

9.1 Binary Configuration Protocol

Note:

- The CRC is calculated over all bytes of the frame starting from Frame header, excluding the CRC field itself. See the appendix for the calculation method and routine.
- The frame length is the total number of all data bytes excluding Frame header, the frame ID, the frame length, and the check bit.
- little-endian mode, low byte sent first.
- This frame must be sent continuously and is not saved on power-off.

Table9 Binary Configuration Protocol

Content	Type	Relative position
Frame header 1:0x55	Uint8	0
Frame header 2: 0xAA	Uint8	1
Frame ID: 0x0101	Uint16	2
Frame length: 0x0018	Uint16	4
(Reserved)	Uint8	6
(Reserved)	Uint16	7
(Reserved)	Uint8	9
Antenna lever arm X (cm)	Int16	10
Antenna lever arm Y (cm)	Int16	12
Antenna lever arm Z (cm)	Int16	14
(Reserved)	Int16	16
(Reserved)	Int16	18
(Reserved)	Int16	20
(Reserved)	Int32	22
(Reserved)	Uint16	26
(Reserved)	Uint8	28
Output protocol: 1: Set the output to the frame described in Section 6.1 2: Set the output reserved protocol (generally not used)	Uint8	29
CRC check bit	Uint32	30

9.2 String Configuration Protocol

9.2.1 ConfigurationPrimary antennaLever Arm

For example, configure lever arm vector as X=0.5m,Y=-0.6m,Z=-1.0m

Command: AT+CLUB_VECTOR=0.5,-0.6,-1.0\r\n

Response: GPS_POS_X=0.5,GPS_POS_Y=-0.6,GPS_POS_Z=-1.0/r/n

Description: lever arm vector is the RTK Primary antenna phase center relative toIMU phase centerof*three-dimensional vector (X,Y,Z), in meters. Where,in the front-right-down vehicle body coordinate system*

If RTK Primary antenna is at the IMU'sfront, then it is positive; otherwise, it is negative;

If the RTK Primary antenna is to theright of the IMU, the value is positive; otherwise, it is negative;

If RTK Primary antenna is on the IMU'supperside, thenit is a**negativenumber**, otherwise it is apositivenumber.

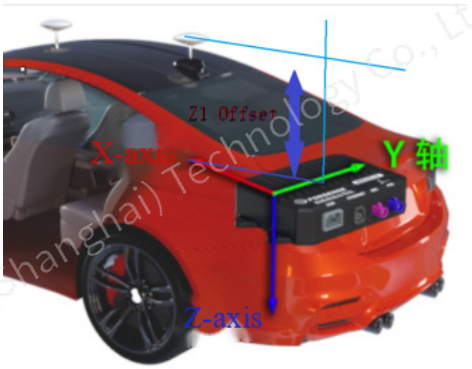
The coordinate system diagram is shown below: (the label must face up; if the IMU is not installed as shown in the figure below, configure the9.2.20section to configure the installation orientation)

Figure9 Coordinate system diagram



Figure 10 Antenna lever arm diagram





9.2.2 Configure rear axle center lever arm

For example, configure lever arm vector as X=0.5m, Y=-0.6m, Z=1.0m

Command: AT+OD_VECTOR=0.5,-0.6,1.0\r\n

Response: OD_POS_X=0.5, OD_POS_Y=-0.6, OD_POS_Z=1.0\r\n

Note: lever arm vector is the three-dimensional vector (X, Y, Z) from the IMU phase center to the rear axle center, in meters, where it is expressed in the front-right-down vehicle body coordinate system.

If the center of the rear wheel axle is in front of the IMU, it is positive; otherwise, it is negative;

If the center of the rear wheel axle is to the right of the IMU, it is positive; otherwise, it is negative;

If the rear wheel axle center is below the IMU, it is**positive**; otherwise it is negative (the rear wheel axle center is generally below the device).

Coordinate system diagram as shown below: (label must face up; if IMU is not installed as shown below, configure9.2.20the installation orientation in the chapter configuration)



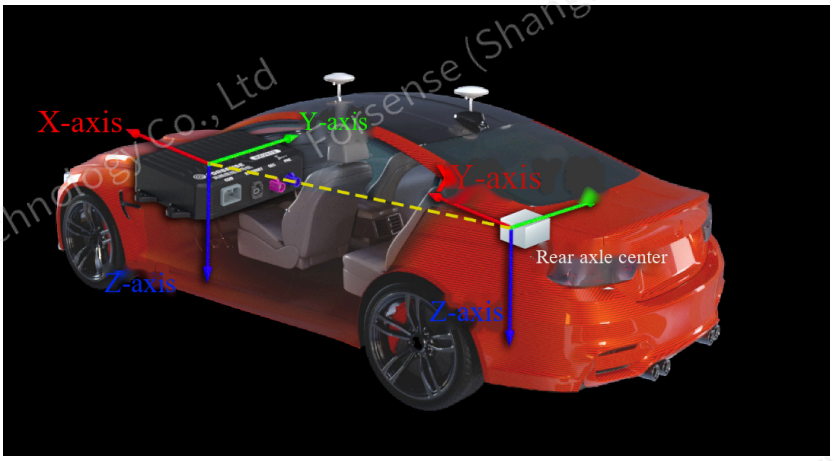


Figure11 Coordinate system diagram

9.2.3 Configure vehicle wheelbase

If the configured vehicle wheelbase is 2m, the configuration command is:

AT+WHEEL_BASE=2\r\n

Note: Save after configuring commands

9.2.4 Configure RTK Dual antenna installation angle

If the RTK Dual antenna installation angle is configured to 0 degrees, the configuration command is:

Command: AT+RTK_ANGLE=0

Response: ANGLE=0

Note: The installation angle is the angle between the ray from Primary antenna to Secondary antenna and the vehicle forward direction; clockwise is positive, counterclockwise is negative. The angle input range is -180°~180°.

Note: Configuration commands take effect only after being saved and the device is power-cycled; the spacing between Dual antenna must be greater than 50cm

Figure12 RTK Dual antenna installation angleSchematic diagram



9.2.5 Enable Dual antenna fusion

Note: By default, Dual antenna fusion is disabled and is used only when stationary for quick initialize. To use the Dual antenna heading throughout, enable Dual antenna fusion as instructed below. 1 enables Dual antenna fusion; all other Parameter disable Dual antenna fusion. Note that the Dual antenna installation error must be calibrated at the same time; otherwise, there will be a fixed error relative to the vehicle body heading.

Command :AT+DRTK_USE=1\r\n

Response: OK

Note: After saving the configuration commands, a power cycle is required for them to take effect; the Dual antenna spacing distance must be greater than 50cm.

9.2.6 Calibrate Dual antenna

Dual antenna markThe process is as follows

1.Connect Serial Port, enter AT+SETNO\r\nDisable all printing

2. Send command AT+RTK_BIAS_EST=1 to start the calibration process

3. Vehicle drives straight, speed no lower than 3km/h; after Serial Port returns RTK_BIASE_CORRECT_DONE, calibration is successful

4. Send AT+CONFIG\r\n command to see the printout: RTK_BIAS_FLAG_AND_VALUE=99,XX (calibration angle). Check whether it is reasonable; under normal installation, it is less than 1 degree.

5.AT+SAVE\r\nSave the result, then power cycle.

9.2.7 Configure the position and velocity projection points output by Integrated Navigation

If the output is configured to the projection points result set by Integrated Navigation, the configuration instruction is:

Command : AT+PROJ_VECTOR=1.0,2.0,3.0\r\n

Response: PROJ_VECTOR_X=1.0, PROJ_VECTOR_Y=2.0, PROJ_VECTOR_Z=3.0\r\n

Note: Integrated Navigation output defaults to the antenna phase center coordinates. To output coordinates for another position, you need to configure the lever arm vector of the IMU phase center relative to this projection points position; the configuration method is the same as9.2.1 Lever arm configuration

9.2.8 Configure output binary-AG data stream

Command: AT+SETAG\r\n

Response: OK

If configured to stop output

Command: AT+SETNO\r\n

Response: OK

9.2.9 Configure output binary-Integrated Navigation data stream

Command: AT+SETNAV\r\n

Response: OK

If configured to stop output

Command: AT+SETNO

Response: OK

9.2.10 Configure output of NMEA format data stream

If NEMA sentence output is configured, the 8.2 Integrated Navigation data stream is not output

If you need to switch to 8.2 Integrated Navigation data stream output requires first following9.2.11 instruction to stop the current data stream output

The configuration commands are as follows:

GPGGA

Example: at 5HzOutput GPGGA sentence at frequency: AT+GPGGA=5\r\n

Response: OK

GPRMC

Example: with 1HzOutput the GPRMC sentence at a frequency: AT+GPRMC=1

Response: OK

GPHDT (heading information)

Example: with 1HzFrequency output GPHDT statement: AT+GPHDT=1

Response: OK

GPVTG (ground speed information)

Example: with 1HzOutput GPVTG sentence at frequency: AT+GPVTG=1

Response: OK

GPZDA (UTC time and date)

Example: with 1HzFrequency output GPZDA sentence: AT+GPZDA=1\r\n

Response: OK

GPATT (custom message)

Example: using 1HzGPATT sentence output frequency: AT+GPATT=1\r\n

Response: OK

If the configuration stops output

Command: AT+SETNO

Response: OK

9.2.11 Configure the current data stream to stop output.

Command: AT+SETNO\r\n

Response: OK\r\n

9.2.12 Configure Data Output Frequency

If the data output frequency is configured to 10Hz, then the configuration command is:

Command: AT+OUTRATE=10\r\n

Response: OK

9.2.13 Configure baud rate

Only baud rates of 115200 or 230400 are supported for configuration; the default baud rate is 115200.

If IMU serial port baud rate is configured to 230400, the configuration command is:

Command: AT+BAUD=230400

Response: BAUD=230400\r\n

Note: Configuration commands take effect only after saving and a power cycle.

9.2.14 Configure CAN baud rate

Only baud rates 250K, 500K, and 1M can be configured; the default baud rate is 500K.

Example: If the CAN baud rate is configured to 500K, the configuration command is:

Command: AT+CAN_BAUD=500\r\n

Response: OK

Note: Configuration commands take effect only after saving and a power cycle.

9.2.15 Configure CAN output frequency

If the CAN data output frequency is configured to 10Hz, the configuration command is:

Command: AT+CAN_ODR=10

Response: OK

Note: supports up to 100Hz，takes effect immediately after modification, and remains in effect after power-off and restart once saved

9.2.16 Print all configuration information

If querying all configured information, the configuration instruction is:

AT+CONFIG\r\n

9.2.17 Query version number

AT+VERSION\r\n

9.2.18 Configure carrier XYZ velocity output

After configuration, once initialize is complete, the north, east, and down velocities in the combined data stream will switch to the carrier's XYZ three-axis velocities.

1 enables carrier speed output; all other Parameter are disabled.

Command: AT+NAV_OUTPUT_XYZ=1

Response: OK

9.2.19 Removing gravitational acceleration from acceleration data

1. Remove gravity acceleration command: AT+DEDUCTIONG=1\r\n

After sending, an OK response indicates successful transmission and takes effect immediately.

Send AT+SAVE\r\n to save the configuration; otherwise it will be lost after a restart.

2. Restore gravity acceleration command: AT+DEDUCTIONG=0

Returning OK after sending indicates successful transmission, effective immediately.

Send AT+SAVE\r\n to save the configuration; otherwise, it will be lost after a restart.

9.2.20 Configuration installation rotation angle

Currently, only the following rotation angles are supported.

Rotate 180 degrees around the x-axis

z-axis rotation 90 180 270

The configuration instructions are as follows

If the installation rotation angle is X-axis rotation of 180 degrees, the configuration command is:

Command: AT+INSTALL_ANGLE=180,0,0

Response: INST_ANGLE_X=180.000,INST_ANGLE_Y=0.000,INST_ANGLE_Z=0.000

If the installation rotation angle is a Z-axis rotation of 180 degrees, the configuration command is:

Command: AT+INSTALL_ANGLE=0,0,180\r\n-

Response: INST_ANGLE_X=0.000,INST_ANGLE_Y=0.000,INST_ANGLE_Z=180.000

9.2.21 Save Parameter

The above Parameter, including lever arm, data stream, output frequency, etc., all require entering asave command once after configuration, and it takes effect only after a restart

Command: AT+SAVE

Response: OK

9.3 Configuration via CAN

Note: All CAN messages are little-endian mode, low byte first.

9.3.1 Check version number

Example:

ID = 0x19FFF326

Data = 0x9A 0x07 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF

Response

Example: (version number 1805)

ID = 0x19FFF29A

Data = 0x26 0xC7 0x00 0x00 0x07 0x0D 0xFF 0xFF

Note: The first two bytes 0x26 0xC7 are fixed Frame header

9.3.2 Configure wheelbase and lever arm

Example

ID = 0X0DFFC126

Data = wheelbase 2 bytes + lever arm 2 bytes * 3 (unit: cm)

Response

ID = 0X19FFC09A

Data = wheelbase 2 bytes + lever arm 2 bytes*3 (unit: cm)

9.3.3 Query wheelbase and lever arm

ID = 0X0DFFC226

Data =0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF

Response

ID = 0X19FFC29A

Data = wheelbase 2 bytes + lever arm 2 bytes * 3 (unit: cm)

9.3.4 Configure CAN output Update Rate and baud rate

Baud rate supports 1M, 500K, 250K (represented as 0x01, 0x02, 0x03 respectively)

Update Rate supports 20, 50, 100Hz (represented as 0x14, 0x32, 0x64 respectively)

Restart after configuration to take effect.

ID = 0x19FFF326

Data = 0x9A 0x4A baud rate Update Rate 0xFE 0xFB 0xF9 0xFF

Response

ID =0x19FFF59A

Data =0xFF 0x4A baud rate Update Rate 0xFF 0xFF 0xFF 0xFF

9.3.5 Query Update Rate and baud rate:

ID = 0x19FFF426

Data =0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF

Response

ID =0x19FFF49A

Data =0xFF 0x4A baud rate Update Rate 0XFF 0XFF 0XFF 0xFF

9.3.6 Configure rear wheel center lever arm

ID = 0X0BC00114

Data=lever arm 2 bytes*3+FF*2 (unit: cm)

Response

ID=0X0BCBB114

Data=lever arm 2 bytes*3+00*2 (unit: cm)

9.3.7 Configure projection points lever arm

ID = 0X0BC00111

Data=lever arm 2 bytes*3+FF*2 (unit: cm)

9.3.8 Query projection points lever arm

ID = 0X0BCFF111

Data=0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF

Response

ID = 0X0BCBB111

Data=lever arm 2 bytes*3+00*2 (unit: cm)

9.3.9 Configure the Dual antenna installation misalignment angle

ID = 0X0DFFC326

Data=Angle(init16)+FF*6 (Unit: degree*100)

9.3.10 Query Dual antenna installation deviation angle

ID = 0X0DFFC526

Data=0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF

Response

ID = 0X19FFC29A

Data=Angle+FF*6 (Unit: degrees*100)

9.3.11 Configure low-pass filter

ID = 0X0DFFC626

Data=filtered value 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF

Response

ID=0X19FF69A

Data=filtered value 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF

9.3.12 Query low-pass filter

ID=0X0DFFC726

Data=0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF

Response

ID=0X19FF79A

Data=Filtered value 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF

9.3.13 Enable PPS(Standard Frame)

ID=0x400

Data=0X00 0X00 0X00 0X00 0X00 0X00 0X00 0X00

Response

ID=0XBCBB116

Data=0X00 0X00 0X00 0X00 0X00 0X00 0X00 0X00

9.3.14 Disable PPS(Standard frame)

ID=0x399

Data=0X00 0X00 0X00 0X00 0X00 0X00 0X00 0X00

Response

ID=0xBCBB116

Data=0X01 0X00 0X00 0X00 0X00 0X00 0X00 0X00

9.3.15 Save Parameter command

ID = 0X0DFFCF26

Data =0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF

Response

ID = 0X19FFCF9A

Data=0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF

9.3.16 Configure IMU mounting rotation angle

The first 6 bytes are grouped in pairs, with each pair representing the rotation angle configured for one axis.

Example: rotate 180 degrees around the X-axis

ID:0X0BC00110

Data =B4 00 00 00 00 00 00 00

9.3.17 Query configured IMU installation rotation angle

ID:0X0BCFF110

Data =00 00 00 00 00 00 00 00

Response:

ID:0X0BCBB110

Data =B4 00 00 00 00 00 00 00

9.3.18 Configure whether Dual antenna fusion is enabled

When configured to 0x99, it is enabled; any other value disables it.

Example: Configure Dual antenna fusion as enabled

ID:0X0BC00115

Data =99 00 00 00 00 00 00 00

Response

ID:0X0BCBB115

Data =01 00 00 00 00 00 00 00

10. Heading accuracy test

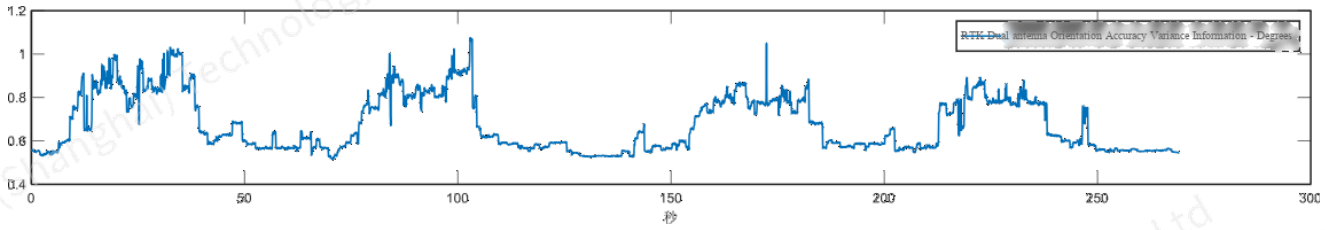
Test conditions:

- Using a tractor as the test vehicle, conduct straight-line, turning, and other tests at speeds of 1~18km/h.
- Use Unicore UM982The Yaw angle output by the board serves as the ground truth reference. The baseline length is 1.5 m
- The Yaw angle of the single antenna Integrated Navigation module have all been time-synchronized with the RTK Yaw angle via PPS signal.

Test results:

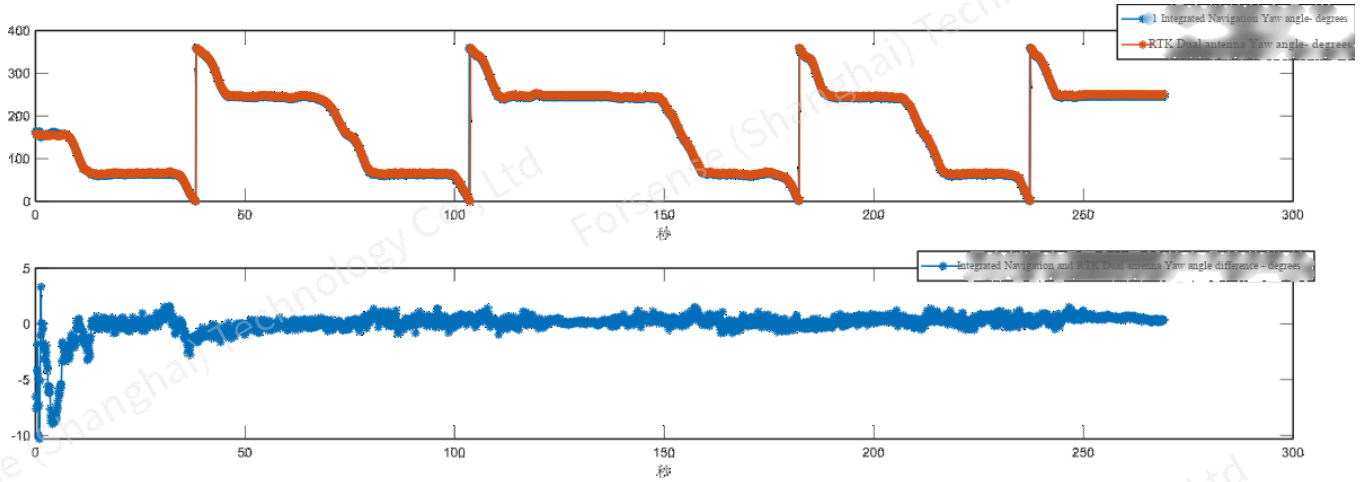
- Yaw angle accuracy of the single antenna Integrated Navigation module within 5s after startupconvergesto within 1 degree.

Figure 13 Orientation accuracy variance information/Figure



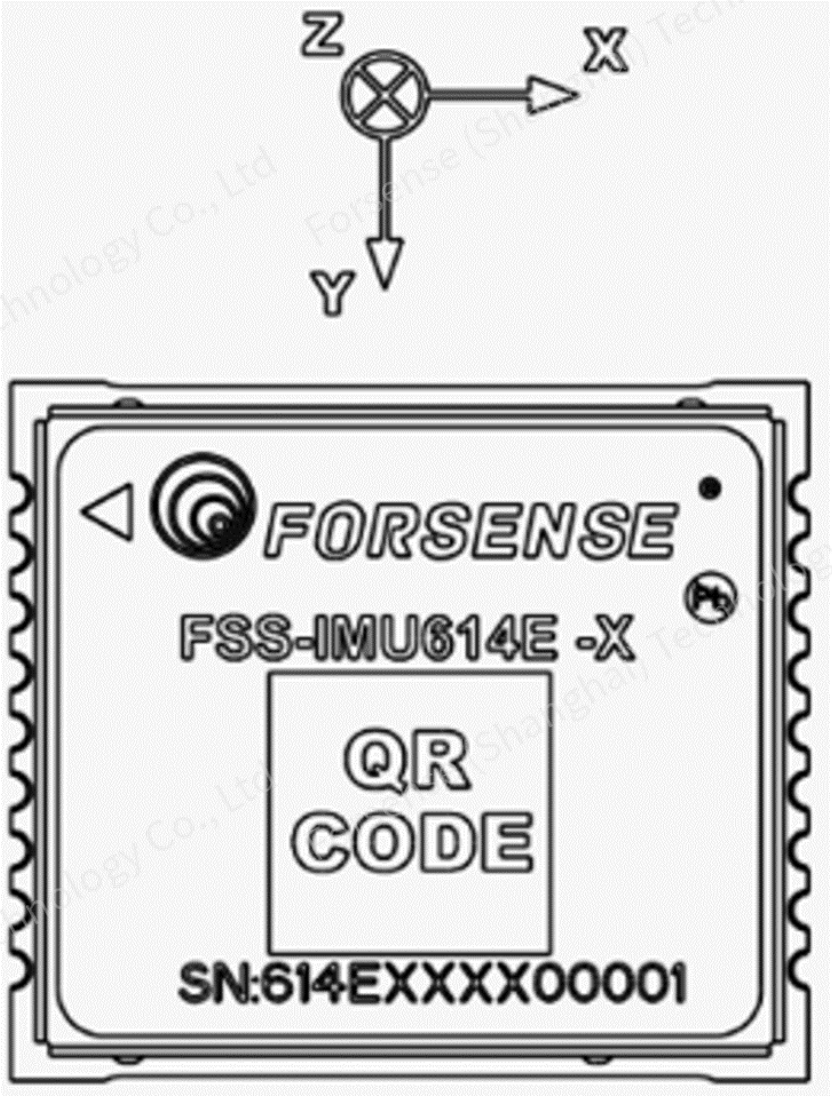
- The Yaw angle accuracy of the single antenna Integrated Navigation module is better than 0.3 degreesrms。

Figure 14 Yaw angle/Figure



11. Coordinate System Definition

Figure 15 Coordinate System Schematic



The coordinate system of this product uses a Forward-Right-Down (FRD) coordinate system. The Euler angle ranges are as follows:

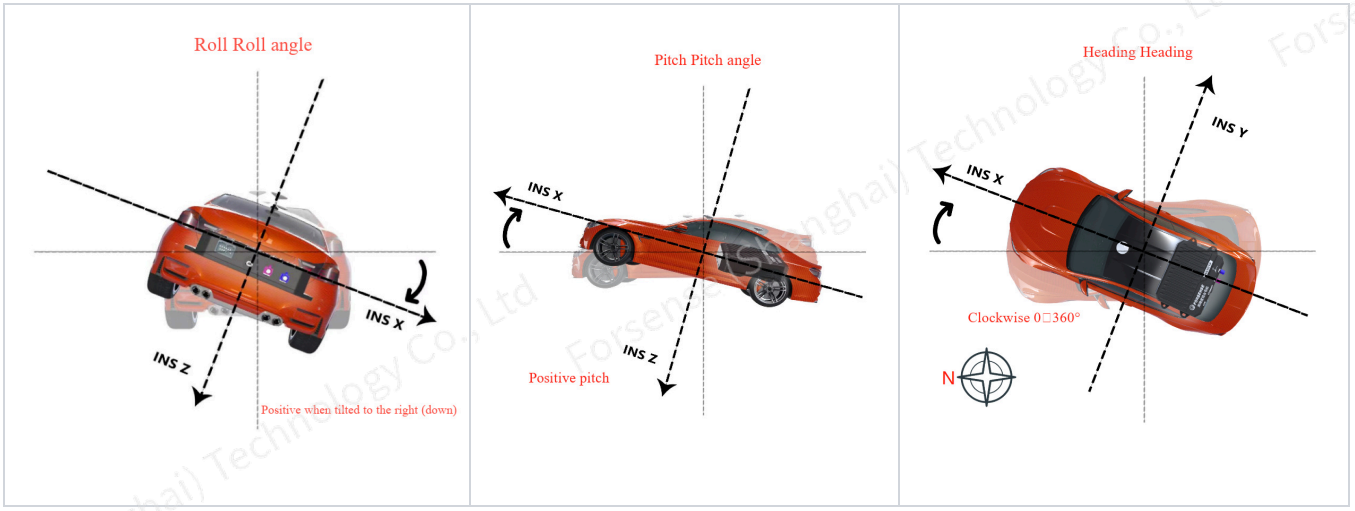
Rotation about the Z-axis: Yaw angle Yaw range: 0° ~360°;

Rotation about the X axis: Roll angle Roll range: -180°~180°;

Rotation around the Y-axis: Pitch angle Pitch range: -90°~90°.

The schematic diagrams for roll, pitch, and Yaw angle degrees are shown below:

Figure 16 Schematic diagram of roll, pitch, and Yaw angle



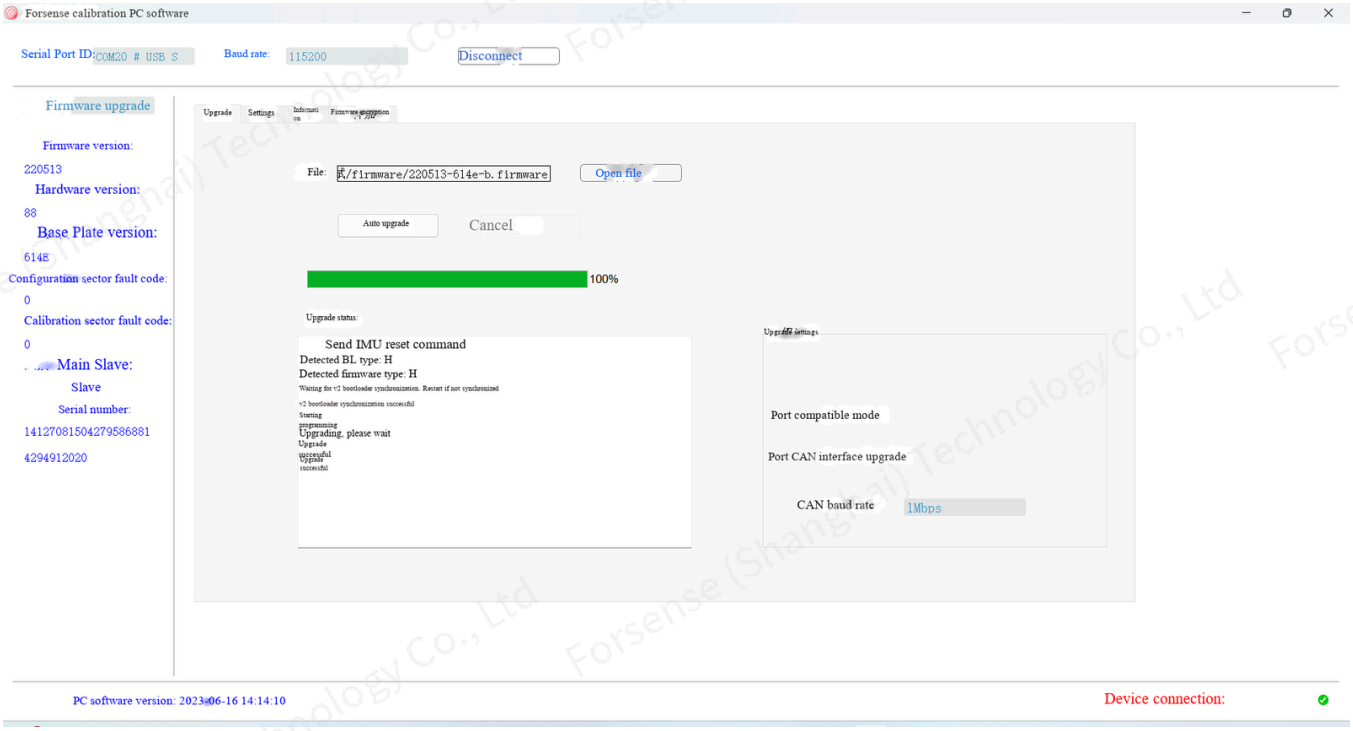
12. Firmware Upgrade

12.1 via PC software

12.1.1 R232 Serial Port Direct Connection

Use Forsense IMU to test PC software — Select Firmware Upgrade — Open Firmware — Click Auto Upgrade.

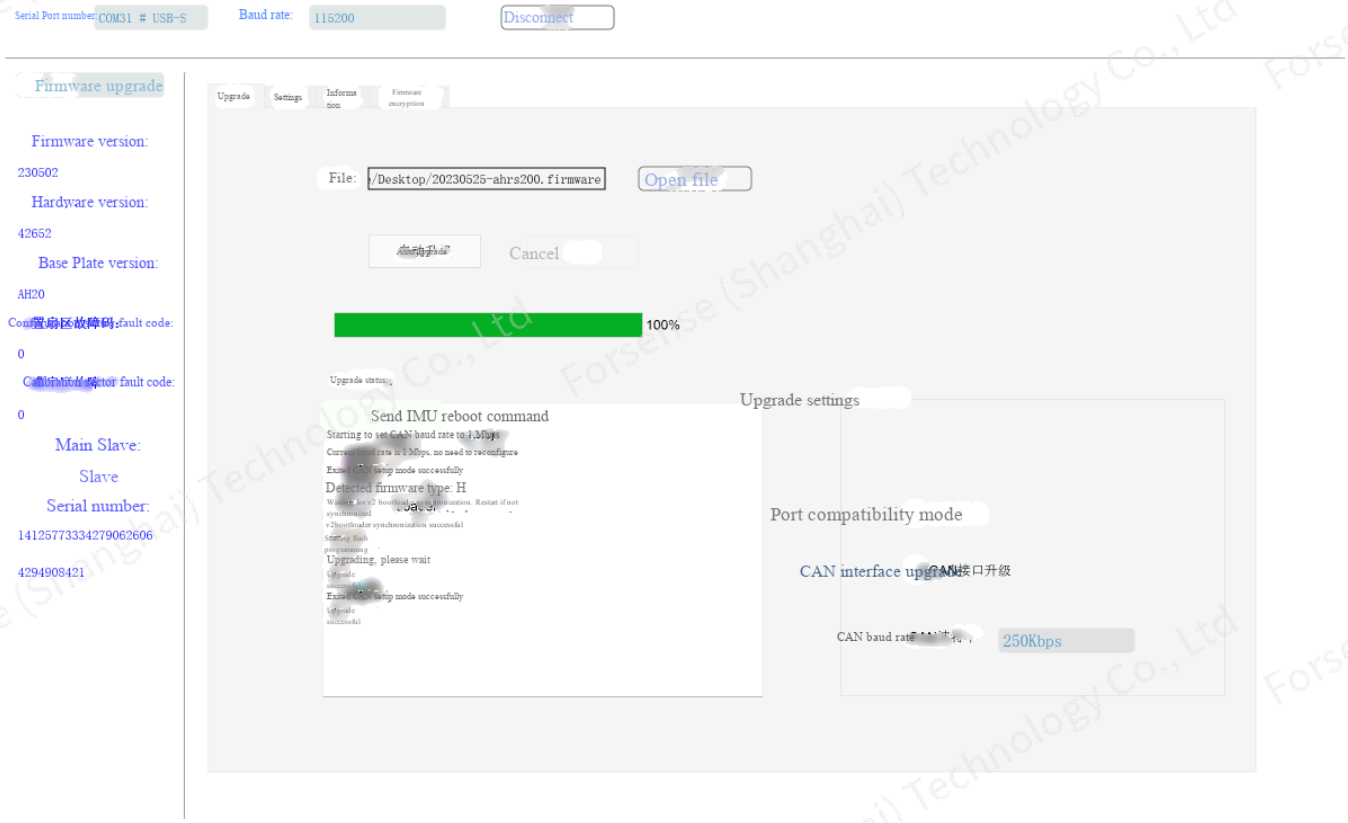
Figure 17 PC software Firmware Upgrade Page-Serial Port/Figure



12.1.2 Use the included USB-to-CAN module

Use Forsense IMU to test PC software——Select Firmware Upgrade——Open Firmware——Select CAN Interface Upgrade——Set the Baud Rate of the Upgraded Firmware——Click Auto Upgrade.

Figure 18 PC software Firmware Upgrade Page - USB to CAN/Figure



12.2 serial push

The upgrade consists of the following steps:

First step: Send the upgrade command

Send an upgrade command to tell the module to prepare for upgrade. After the command is sent, the device will record an upgrade flag in a certain area of flash, then perform a soft reboot. At this point, the IMU module will enter

In BOOTLOADER.

PC software sends the update command as follows:

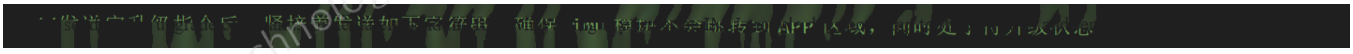
```
cmd_bl[34] ={0x55,0xaa,0xbb,0x88,0x18,0x00,0x00,0x00,0xc8,0x42,0x00,0x00,
0x48,0x43,0x2c,0x01,0x00,0x00,0x90,0x01,0x00,0x00,0xf4,0x01,0x00,0x00,
0x58,0x02,0x00,0x00,0x40,0x97,0x46,0x6a};
```

At this moment imu the module will restart and enter bootloader 中

Step two: Send**HC32MCU_FORSENSE** String

After the IMU module enters the bootloader, the IMU module will actively send the HC32_UPLOADER string. During this period, PC software needs to send the HC32MCU_FORSENSE string. Once the IMU module receives this string, it will not jump to the APP and will remain in the bootloader in a state awaiting program upgrade, and at the same time

will stop actively sending any messages



```
QString str = "HC32MCU_FORSENSE" ;
for(int i=0;i<10;i++)
{
  _port_device->write(str.toLatin1()); sleep_ms(50);
}
```

Third step: send the synchronization command

When the IMU module is in the bootloader awaiting upgrade, it does not actively send messages and only passively responds to commands from PC software. At this point, after PC software sends the command Send_CMD_LONG(0X21,0,0,0,0,0,0), it waits for a response from the IMU module. After the IMU receives this command, it immediately replies with Send CMD ACK (0x64,0x10,0)

Data. Once PC software determines that this data has been received, synchronization is considered complete.

```
PC software and imu Module synchronization process:
1. PC software sends a synchronization command with command code 0x21, then waits for response from the imu module
Send_CMD_LONG(0x21,0,0,0,0,0,0);
2. Upon receiving the command code 0x21 from the imu module, it sends response data, with data command as follows:
Send_CMD_ACK(0x64,0x10,0);
3. PC software checks whether it has received imuthe response data. If received, synchronization is complete
Note: See Send CMD LONG and Send CMD ACK functions at the end of the article
```

Send erase command

PC software sends the erase command Send_CMD_LONG(0x23,0,0,0,0,0,0). After the command is sent, the IMU module will completely erase all contents in the APP area and inform PC software of the final execution result. PC software will decide whether to send the erase command again based on the execution result. Once the erase is successful,IMUThe module's APP cannot be restored. In addition, PC software must wait until IMU Only after receiving a successful erase response can the next step be performed; otherwise, it may cause the subsequent upgrade to fail.

```
1. PC software sends an erase command with command code 0x23, after sending this command, patiently wait imu for the execution result
Send_CMD_LONG(0x23,0,0,0,0,0,0);
2. imu After receiving the command the APP area data will be completely erased, and the final execution result will be replied to {(TERM_0000)} ACK(0x64, 0x10, 0); // indicates successful erase
send_CMD_
```

```
send_CMD_ACK(0x64,0x11,0); //表示擦除失败，需要重新擦除
注意：上位机必须收到擦除成功的响应，才能进行下一步
```

Step five: Send the upgrade data packet

After successful erasure, the most important step of sending firmware data begins. The data is sent using the function Send_Upload_Data. PC software divides the upgrade firmware into packets, each with a fixed size of 64 bytes. The last packet, if it is less than 64 bytes, is sent according to the actual byte count. Each data frame contains the valid data length and the offset address of this packet in the entire firmware. Each time PC software sends a data frame, it must waitIMU the module's response; after determining that imu has successfully obtained this frame of data, then send the next frame. IMU After the module successfully receives the data packet from PC software, it will send response data and write to the specified flash address according to offset address.

If writing to flash fails, send a failure command; if writing succeeds, do not send.

Example: Send the 1000-byte upgrade file uint8_t Upgrade_Data[1000] to the imu module

// Send the first packet:

1.1 Send bytes 0~63 via function to the measurement module

```
Send_Upload_Data（0x27,0,0,0x40,Upgrade_Data);
```

// In the above function, the first Parameter 0x27 is a fixed value: the second Parameter is a fixed value: the third Parameter 0 is offset address.

The fourth Parameter 0x40 is the valid byte length: the fifth Parameter is the starting address of the transmitted data

1.2 After the mu successfully receives the data, it sends response data; and writes to the specified flash address based on offset address

```
send_CMD_ACK(0x753D,0x00,0);
```

// In the above function, the first Parameter 0x753D is a fixed value: the second Parameter is a fixed value, the third Parameter is offset address.

If the imu module fails to write data into flash during the process, it sends a failure command send_CMD_ACK(0x64,0x11,0)

// Send the second packet:

2.1 After PC software receives imu response data, it sends the second packet data Send_Upload_Data(0x27,0,0x40,0x40,Upgrade_Data+0x40);

2.2 After the imu successfully receives the second packet, it sends a response: send_CMD_ACK(0x753D,0x00,0x40)

2.3 If the imu module fails to write data into flash during the process, it sends a failure command send_CMD_ACK(0x64,0x11,0);

// Send the third packet

3.1 After PC software receives imu response data, it sends the third packet data Send_Upload_Data(0x27,0,0x80,0x40,Upgrade_Data+0x80);

3.2 After the imu successfully receives the third packet, it sends a response: send_CMD_ACK(0x753D,0x00,0x80);

3.3 If the imu module fails to write data into flash during the process, it sends a failure command send_CMD_ACK(0x64,0x11,0);

1/ Send the fifteenth packet:

15.1 PC software received After i response data: send the fifteenth data packet Send_Upload_Data（0x27,0,0x380,0x40,Upgrade_Data+0x380);

15.2 imu After successfully receiving the fifteenth data packet: send response send_CMD_ACK(0x753D,0x00,0x380);

15.3 imu If writing fails during the module's data write to human flh, send failure command send_CMD_ACK(0x64,0x11,0);

Send the sixteenth packet:

15.1 PC software received After imu response data: send the sixteenth data packet Send_upload_Data (0x27, 0x3c00x28, Upgrade_Data+0x3c0)

15.2 imu After successfully receiving the sixteenth data packet: send response: send_CMD_ACK(0x753D,0x00,0x3C0);

15.3 imu If writing fails during the module's data write to flh, send failure command send_CMD_ACK(0x64,0x11,0);

发送。 Send end

Step Six: Acquire**CRC** Checksum

In general, the firmware upgrade command format is imu614e-ag#CRC1373387121.firmware, the CRC string is immediately followed by the precomputed CRC checksum value. After the upgrade file is sent, PC software needs to send a verification command to determine whether the upgrade file received by the imu module is erroneous. After PC software sends Send_CMD_LONG(0x29,0,0,0,0,0,0) command, obtain the CRC check digit calculated by the imu module itself,

If PC software determines that CRC checksum value is incorrect, the upgrade should be restarted beginning with the erase command of the fourth step.

PC software sends request to obtain crc verification code command, wait Firmresponse

```
Send_CMD_LONG(0x29,0,0,0,0,0,0);
```

imu 模块自动发送 crc verification code data:

```
send_CMD_ACK(0x753C,0x10,crc32_data);
```

wherein crc32_data 值为 imu 模块中 crc32 data

Step seven: Send restart command

After PC software determines that the crc check value is correct, the upgrade sends a restart command and succeeds.

```
判断 crc 校验值正确后，发送重启命令：
    Send_CMD_LONG(0x30,0,0,0,0,0,0);
```

After the firmware upgrade is complete, power off and restart the device, then read the version number to determine whether the upgrade succeeded.

Function definition:

1. The Send_CMD_LONG function is defined as follows:

```
struct  MULTI_LONG_CMD_STRUCT
{
uint8_t header1;
uint8_t header2;
uint16_t id;
uint16_t length;
float param1;
float  param2;
uint32_t param3;
uint32_t param4;
int32_t param5;
int32_t param6;
uint32_t check_crc;
}__attribute__((packed));
```

```
void  :Send_CMD_LONG(uint16_t cmd_id,float cm1,float cm2,uint32_t  cm3,uint32_t cm4,int32_t cm5,int32_t cm6)
{
    uint8_t check_sum=0;
    struct MULTI_LONG_CMD_STRUCT data_cmd_long __attribute__((packed));
    data_cmd_long.header1=0x55;
    data_cmd_long.header2=0xAA;
    data_cmd_long.id=cmd_id;
    data_cmd_long.length=sizeof(data_cmd_long)-10;
    data_cmd_long.param1=cm1;
    data_cmd_long.param2=cm2;
    data_cmd_long.param3=cm3;
    data_cmd_long.param4=cm4;
    data_cmd_long.param5=cm5;
    data_cmd_long.param6=cm6;
    int len=sizeof(data_cmd_long)-4;
    uint32_t check_crc=1;
    data_cmd_long.check_crc=crc_crc32(check_crc,(uint8_t *)(&data_cmd_long), len);
    send((uint8_t *)(&data_cmd_long),sizeof(data_cmd_long));
}
```

2.The Send_CMD_ACK function is defined as follows:

```
struct  CMD_ACK_STRUCT
{
uint8_t header1;
uint8_t header2;
uint16_t id;
uint16_t length;
uint32_t command;/*<   Command ID (of acknowledged command).*/
uint32_t result;/*<   Result of command.*/
uint32_t check_crc;
}__attribute__((packed));

void Send_CMD_ACK(uint16_t cmd_id,uint16_t ack_id,uint32_t result)
{
    uint32_t check_crc=0;
    struct CMD_ACK_STRUCT data_cmd_ack __attribute__((packed));
    data_cmd_ack.header1=0xAA;
    data_cmd_ack.header2=0x55;
    data_cmd_ack.id=cmd_id;
    data_cmd_ack.length=sizeof(data_cmd_ack)-10;
    data_cmd_ack.command=ack_id;
    data_cmd_ack.result=result;
    int len=sizeof(data_cmd_ack)-4;
    check_crc=1;
    data_cmd_ack.check_crc=crc_crc32(check_crc,(uint8_t *)(&data_cmd_ack), len);
    Cout((uint8_t *)(&data_cmd_ack),sizeof(data_cmd_ack));
}
```

3.The Send_Upload_Data function is defined as follows:

```
struct  UPLOAD_DATA
{
uint8_t header1;
uint8_t header2;
uint16_t id;
uint16_t length;
uint8_t param[64];
uint32_t offset;
uint16_t size;
uint8_t cmd;
uint32_t check_crc;
}__attribute__((packed));

struct UPLOAD_DATA  upload_data;
void Send_Upload_Data(uint8_t cmd_id,uint8_t cmd,uint32_t offset,uint16_t size,uint8_t*  param)
{
    upload_data.header1=0x55;
    upload_data.header2=0xAA;
    upload_data.id=cmd_id;
    upload_data.length=sizeof(UPLOAD_DATA)-10;
    upload_data.cmd=cmd;
    for(int i=0;i<size;i++)
    upload_data.param[i] = *(param+i);
    upload_data.offset=offset;
    upload_data.size=size;
    int len=sizeof(UPLOAD_DATA)-4;
    uint32_t check_crc=1;
    upload_data.check_crc=crc_crc32(check_crc,(uint8_t *)(&upload_data), len);
    send((uint8_t *)(&upload_data),sizeof(UPLOAD_DATA));
}
```

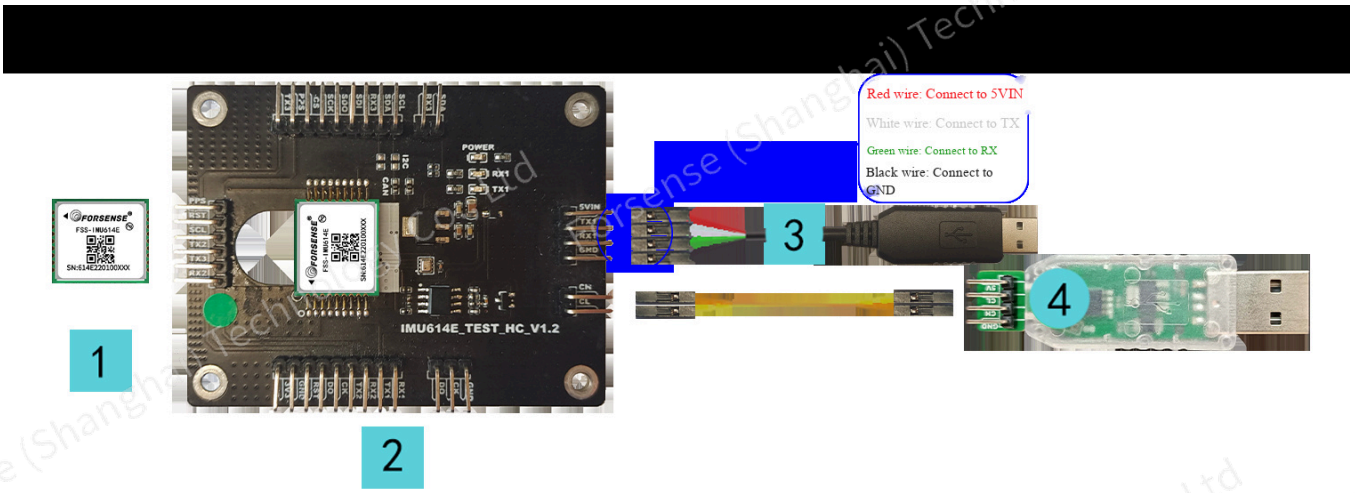
4. The CRC32 verification function is as follows:

```
static const uint32_t crc32_tab[] = {
    0x00000000, 0x77073096, 0xee0e612c, 0x990951ba, 0x076dc419, 0x706af48f,
    0xe963a535, 0x9e6495a3, 0x0edb8832, 0x79dcb8a4, 0xe0d5e91e, 0x97d2d988,
    0x09b64c2b, 0x7eb17cbd, 0xe7b82d07, 0x90bf1d91, 0x1db71064, 0x6ab020f2,
    0xf3b97148, 0x84be41de, 0x1adad47d, 0x6ddde4eb, 0xf4d4b551, 0x83d385c7,
    0x136c9856, 0x646ba8c0, 0xfd62f97a, 0x8a65c9ec, 0x14015c4f, 0x63066cd9,
    0xfa0f3d63, 0x8d080df5, 0x3b6e20c8, 0x4c69105e, 0xd56041e4, 0xa2677172,
    0x3c03e4d1, 0x4b04d447, 0xd20d85fd, 0xa50ab56b, 0x35b5a8fa, 0x42b2986c,
    0xdbbbc9d6, 0xacbcf940, 0x32d86ce3, 0x45df5c75, 0xdcd60dcf, 0xabd13d59,
    0x26d930ac, 0x51de003a, 0xc8d75180, 0xbfd06116, 0x21b4f4b5, 0x56b3c423,
    0xcfba9599, 0xb8bda50f, 0x2802b89e, 0x5f058808, 0xc60cd9b2, 0xb10be924,
    0x2f6f7c87, 0x58684c11, 0xc1611dab, 0xb6662d3d, 0x76dc4190, 0x01db7106,
    0x98d220bc, 0xefd5102a, 0x71b18589, 0x06b6b51f, 0x9fbfe4a5, 0xe8b8d433,
    0x7807c9a2, 0x0f00f934, 0x9609a88e, 0xe10e9818, 0x7f6a0dbb, 0x086d3d2d,
    0x91646c97, 0xe6635c01, 0x6b6b51f4, 0x1c6c6162, 0x856530d8, 0xf262004e,
    0x6c0695ed, 0x1b01a57b, 0x8208f4c1, 0xf50fc457, 0x65b0d9c6, 0x12b7e950,
    0x8bbeb8ea, 0xfcb9887c, 0x62dd1ddf, 0x15da2d49, 0x8cd37cf3, 0xfbd44c65,
    0x4db26158, 0x3ab551ce, 0xa3bc0074, 0xd4bb30e2, 0x4adfa541, 0x3dd895d7,
    0xa4d1c46d, 0xd3d6f4fb, 0x4369e96a, 0x346ed9fc, 0xad678846, 0xda60b8d0,
    0x44042d73, 0x33031de5, 0xaa0a4c5f, 0xdd0d7cc9, 0x5005713c, 0x270241aa,
    0xbe0b1010, 0xc90c2086, 0x5768b525, 0x206f85b3, 0xb966d409, 0xce61e49f,
    0x5edef90e, 0x29d9c998, 0xb0d09822, 0xc7d7a8b4, 0x59b33d17, 0x2eb40d81,
    0xb7bd5c3b, 0xc0ba6cad, 0xedb88320, 0x9abfb3b6, 0x03b6e20c, 0x74b1d29a,
    0xead54739, 0x9dd277af, 0x04db2615, 0x73dc1683, 0xe3630b12, 0x94643b84,
    0x0d6d6a3e, 0x7a6a5aa8, 0xe40ecf0b, 0x9309ff9d, 0x0a00ae27, 0x7d079eb1,
    0xf00f9344, 0x8708a3d2, 0x1e01f268, 0x6906c2fe, 0xf762575d, 0x806567cb,
    0x196c3671, 0x6e6b06e7, 0xfed41b76, 0x89d32be0, 0x10da7a5a, 0x67dd4acc,
```

Forsense (Shanghai) Technology Co., Ltd

13. Test Base Plate connection diagram

Figure19 Schematic diagram of module connection PC software



	Name	Quantity
1	IMU614E series module	1 piece
	Accessory name	Quantity
2	CAN version test Base Plate	1 piece
3	TTL Serial Port cable	1 piece
4	USB to CAN module	1 piece

14. ROS driver (Integrated Navigation data stream)

ROS driver download link:

https://data.forsense-imu.com/page/download.html



14.1 Install ROS serial

Install the ROS serial package; this example relies on the serial package provided by ROS to implement serial communication.

First, run the following command to download and install the serial package:

sudo apt-get install ros -melodic-serial

Then enter roscd serial the command to enter the serial download location; if the installation is successful, the following will appear:

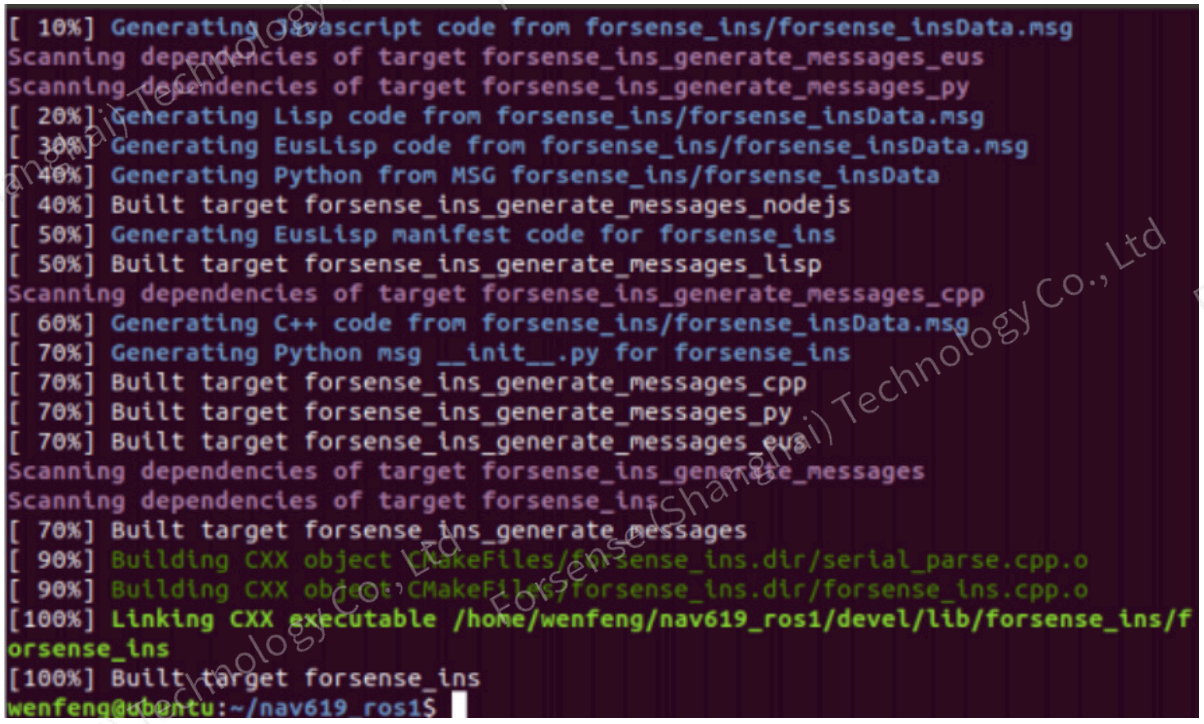
Following information:

/opt/ros/melodic/share/serial

14.2 Compile Code

cd FS982_ros/

catkin_make

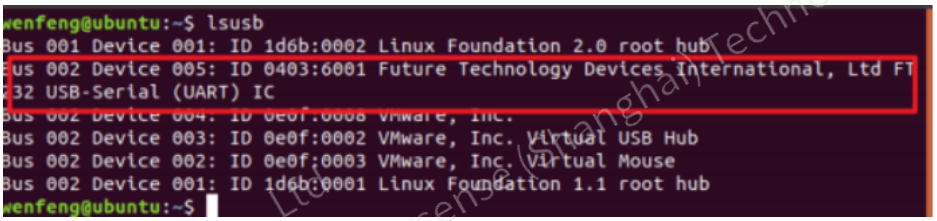


Compilation complete

14.3 Connect the IMU to the system via USB

Check whether connected:

lsusb



View USB port number:

ls /dev/ttyU*



Configure to enable USB to serial port privileges:

sudo chmod 777 /dev/ttyUSB0

14.4 View Data

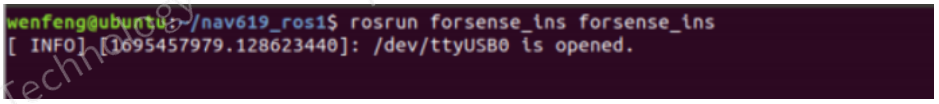
Run roscore to start ROS.

Return to the serial_imu_ws folder and execute

source devel/setup.bash

Run rosrun to start

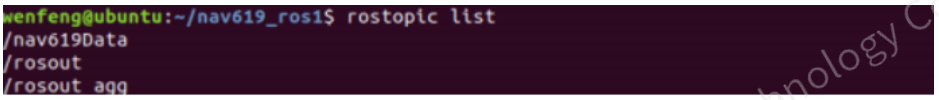
rosrun forsense_ins forsense_ins



Open new window

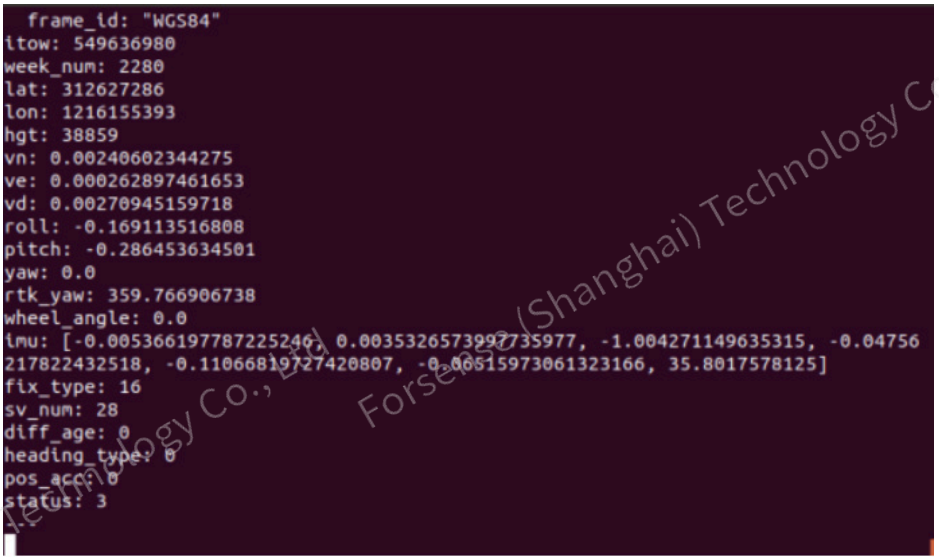
source devel/setup.bash

rostopic list



Enter the command to view IMU data

rostopic echo /FS982Data


frame_id: "WGS84"
itow: 549636980
week_num: 2280
lat: 312627286
lon: 1216155393
hgt: 38859
vn: 0.00240602344275
ve: 0.000262897461653
vd: 0.00270945159718
roll: -0.169113516808
pitch: -0.286453634501
yaw: 0.0
rtk_yaw: 359.766906738
wheel_angle: 0.0
imu: [-0.00536619778722524 6, 0.003532657399773 5977, -1.004271149635315, -0.04756
217822432518, -0.11066819727420807, -0.00515973061323166, 35.8017578125]
fix_type: 16
sv_num: 28
diff_age: 0
heading_type: 0
pos_acc: 0
status: 3

15. For use with the front wheel gyroscope G200

1.G200 Installation requirements:

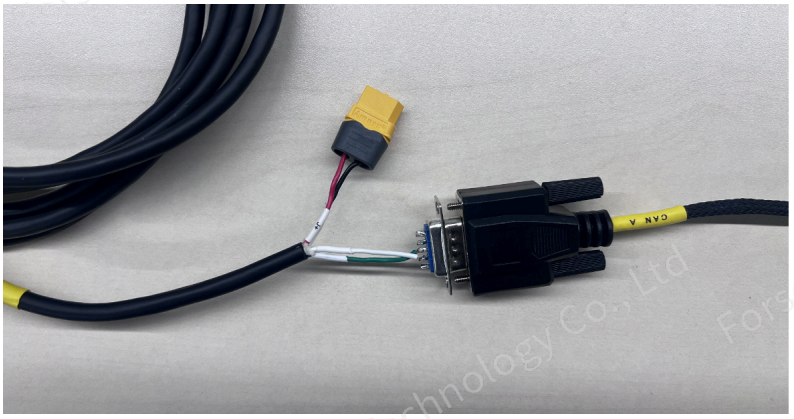
A.G200 must be installed on the front axle of the front wheel (a horizontal rigid position that rotates together with the front wheel)

{"translation": "B. Install G200 with the cable outlet pointing toward the rear of the vehicle"}</translation>

2. The CAN interface of G200 is connected to CAN A of the harness, and the power supply is 5V.

Figure21 Wiring diagram





3. First measure the vehicle's wheelbase. Taking a vehicle wheelbase of 2m as an example, connect Master's Serial Port to the Serial Port assistant, send AT+WHEEL_BASE=2, and receive the reply WHEEL_BASE=2; then send the save command AT+SAVE, and receive the reply OK. The device must be powered on again.

4. Device connection FS982-AG decode PC software V3.3.4, enter the home page of PC software, select Serial Port number, baud rate (default 115200), click connect, the device information on the left is displayed to indicate successful connection, click the mode start at the bottom left, correctly connect to G200, the front wheel gyro significant bits shows green, otherwise red, after the vehicle initialize is successful, the front wheel angle significant bits displays green, otherwise red, after the vehicle moves, it will output front wheel angle data in real time.

Figure22G200 correct connection diagram

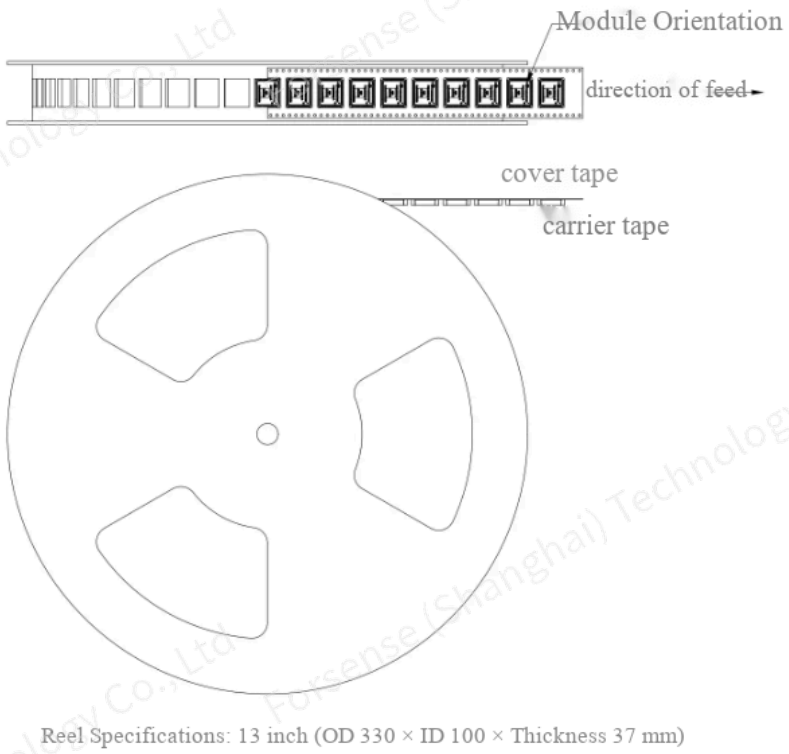


16. Packaging

The IMU614E-AG module uses sealed tape-and-reel packaging. Enables efficient production.

16.1 Tape and reel packaging

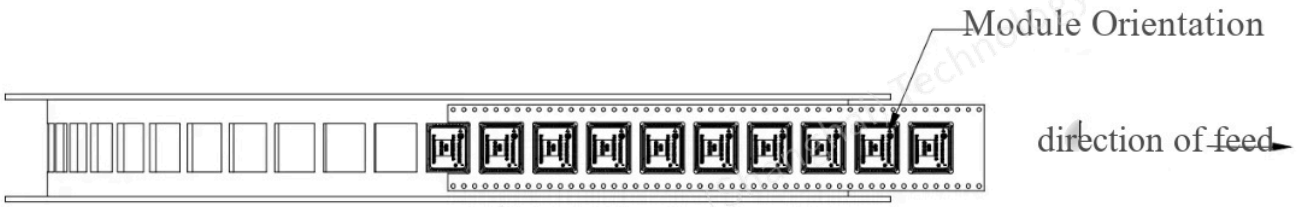
Figure23 Tape and reel packaging diagram



16.2 carrier tape

The position and orientation of the IMU614E-AG module as placed on the carrier tape at the factory are shown in the figure below:

Figure 24 Module position and orientation at carrier tape



17. CRC Table Lookup Calculation

It is recommended to refer directly to the example code.

Note 1: Data is transmitted in little-endian format, with the low byte first and the high byte last.

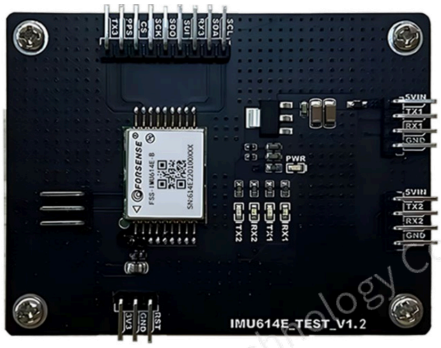
Note 2: The initial value of crc32 is 1; the CRC calculation does not include all data of this frame itself.

```
static  const  uint32_t  crc32_tab [ ]  =  {
0x00000000, 0x77073096, 0xee0e612c, 0x990951ba, 0x076dc419, 0x706af48f,
0xe963a535, 0x9e6495a3, 0x0edb8832, 0x79dcb8a4, 0xe0d5e91e, 0x97d2d988,
0x09b64c2b,  0x7eb17cbd,  0xe7b82d07,  0x90bf1d91,  0x1db71064,  0x6ab020f2,
0xf3b97148,  0x84be41de,  0x1adad47d,  0x6ddde4eb,  0xf4d4b551,  0x83d385c7,
0x136c9856,  0x646ba8c0,  0xfd62f97a,  0x8a65c9ec,  0x14015c4f,  0x63066cd9,
0xfa0f3d63,  0x8d080df5,  0x3b6e20c8,  0x4c69105e,  0xd56041e4,  0xa2677172,
0x3c03e4d1,  0x4b04d447,  0xd20d85fd,  0xa50ab56b,  0x35b5a8fa,  0x42b2986c,
0xdbbbc9d6,  0xacbcf940,  0x32d86ce3,  0x45df5c75,  0xdcd60dcf,  0xabd13d59,
0x26d930ac,  0x51de003a,  0xc8d75180,  0xbfd06116,  0x21b4f4b5,  0x56b3c423,
0xcfba9599,  0xb8bda50f,  0x2802b89e,  0x5f058808,  0xc60cd9b2,  0xb10be924,
0x2f6f7c87,  0x58684c11,  0xc1611dab,  0xb6662d3d,  0x76dc4190,  0x01db7106,
0x98d220bc,  0xefd5102a,  0x71b18589,  0x06b6b51f,  0x9fbfe4a5,  0xe8b8d433,
0x7807c9a2,  0x0f00f934,  0x9609a88e,  0xe10e9818,  0x7f6a0dbb,  0x086d3d2d,
0x91646c97,  0xe6635c01,  0x6b6b51f4,  0x1c6c6162,  0x856530d8,  0xf262004e,
0x6c0695ed,  0x1b01a57b,  0x8208f4c1,  0xf50fc457,  0x65b0d9c6,  0x12b7e950,
0x8bbeb8ea,  0xfcb9887c,  0x62dd1ddf,  0x15da2d49,  0x8cd37cf3,  0xfbd44c65,
0x4db26158,  0x3ab551ce,  0xa3bc0074,  0xd4bb30e2,  0x4adfa541,  0x3dd895d7,
0xa4d1c46d,  0xd3d6f4fb,  0x4369e96a,  0x346ed9fc,  0xad678846,  0xda60b8d0,
0x44042d73,  0x33031de5,  0xaa0a4c5f,  0xdd0d7cc9,  0x5005713c,  0x270241aa,
0xbe0b1010,  0xc90c2086,  0x5768b525,  0x206f85b3,  0xb966d409,  0xce61e49f,
0x5edef90e,  0x29d9c998,  0xb0d09822,  0xc7d7a8b4,  0x59b33d17,  0x2eb40d81,
0xb7bd5c3b,  0xc0ba6cad,  0xedb88320,  0x9abfb3b6,  0x03b6e20c,  0x74b1d29a,
0xead54739,  0x9dd277af,  0x04db2615,  0x73dc1683,  0xe3630b12,  0x94643b84,
0x0d6d6a3e,  0x7a6a5aa8,  0xe40ecf0b,  0x9309ff9d,  0x0a00ae27,  0x7d079eb1,
0xf00f9344,  0x8708a3d2,  0x1e01f268,  0x6906c2fe,  0xf762575d,  0x806567cb,
0x196c3671,  0x6e6b06e7,  0xfed41b76,  0x89d32be0,  0x10da7a5a,  0x67dd4acc,
0xf9b9df6f,  0x8ebeeff9,  0x17b7be43,  0x60b08ed5,  0xd6d6a3e8,  0xa1d1937e,
0x38d8c2c4,  0x4fdff252,  0xd1bb67f1,  0xa6bc5767,  0x3fb506dd,  0x48b2364b,
0xd80d2bda,  0xaf0a1b4c,  0x36034af6,  0x41047a60,  0xdf60efc3,  0xa867df55,
0x316e8eef,  0x4669be79,  0xcb61b38c,  0xbc66831a,  0x256fd2a0,  0x5268e236,
0xcc0c7795,  0xbb0b4703,  0x220216b9,  0x5505262f,  0xc5ba3bbe,  0xb2bd0b28,
0x2bb45a92,  0x5cb36a04,  0xc2d7ffa7,  0xb5d0cf31,  0x2cd99e8b,  0x5bdeae1d,
0x9b64c2b0,  0xec63f226,  0x756aa39c,  0x026d930a,  0x9c0906a9,  0xeb0e363f,
0x72076785,  0x05005713,  0x95bf4a82,  0xe2b87a14,  0x7bb12bae,  0x0cb61b38,
0x92d28e9b,  0xe5d5be0d,  0x7cdcefb7,  0x0bdbdf21,  0x86d3d2d4,  0xf1d4e242,
0x68ddb3f8,  0x1fda836e,  0x81be16cd,  0xf6b9265b,  0x6fb077e1,  0x18b74777,
0x88085ae6,  0xff0f6a70,  0x66063bca,  0x11010b5c,  0x8f659eff,  0xf862ae69,
0x616bffd3,  0x166ccf45,  0xa00ae278,  0xd70dd2ee,  0x4e048354,  0x3903b3c2,
0xa7672661,  0xd06016f7,  0x4969474d,  0x3e6e77db,  0xaed16a4a,  0xd9d65adc,
0x40df0b66,  0x37d83bf0,  0xa9bcae53,  0xdebb9ec5,  0x47b2cf7f,  0x30b5ffe9,
0xbdbdf21c,  0xcabac28a,  0x53b39330,  0x24b4a3a6,  0xbad03605,  0xcdd70693,
0x54de5729,  0x23d967bf,  0xb3667a2e,  0xc4614ab8,  0x5d681b02,  0x2a6f2b94,
0xb40bbe37,   0xc30c8ea1,   0x5a05df1b,   0x2d02ef8d,
}
uint32_t  crc_crc32 (uint32_t  crc,  const  uint8_t  *buf,  uint32_t size ) {
for   (uint32_t  i=0;  i<size ;  i++)  {
```

```
crc = crc32_tab [ (crc ^ buf [i ] ) & 0xff] ^ (crc >> 8 ) ;
}
return crc;
}
```

Forsense (Shanghai) Technology Co., Ltd

18. OptionalAccessories



IMU614E-X test Base Plate



TTL Serial Port cable



USB to CAN module

19. FAQs List

Issue	Solution
PC software cannot connect	Please check whether Serial Port is occupied or the device is powered on normally. If PC software disconnects during connection, Serial Port may be loose; try unplugging and reinserting the Serial Port cable, then reopen PC software.
Positioning status is always 1	Confirm whether the differential data is properly connected, and check if the baud rate, differential account, mounting point, and other information are correct
Dual antenna Orientation status never reaches 50	Please investigate the following factors: 1. If Orientation status is always 0, check whether Secondary antenna is correctly connected, and whether the antenna feeder is damaged. 2. If Orientation status is not 0 but never reaches 50, confirm whether the test environment is open, and whether the distance of Dual antenna is above 50 cm.
Signal is normal when connected to active antenna, no signal when connected to passive antenna	passive antenna feeder must not exceed 1.5M
Signal is normal when antenna is connected to active power splitter, no signal when antenna is powered by the device	Check whether the antenna feeder is damaged or if static electricity caused by plugging/unplugging the antenna while powered may have led to abnormal power supply circuit. The current circuit design includes a fuse, and normal operation can be restored after power cycling.
Unable to initialize	initialize must meet the following conditions, single antenna: Positioning status fixed solution + driving speed above 0.5m/s + straight-line driving with acceleration/deceleration after stopping Dual antenna: Positioning status fixed solution + Orientation status 50
Serial Port packet loss	Please investigate the following factors: 1. Serial Port cable needs to support at least 115200 baud rate 2. Computer Serial Port latency needs to be set to 2ms
RTK data is invalid	Please investigate the following factors: 1. Confirm whether RTK board is connected to the AG module according to the connection requirements recommended in Chapter 7 of the manual, 2. Whether the baud rate and input message comply with the RTK configuration requirements in Chapter 8 of the manual
How to determine whether the physical connection of G200 is correct	Determine whether bit5 is valid by parsing the number from Status bits
When connected to G200, the front wheel gyroscope significant bits is invalid	Use a CAN analyzer to check whether the CAN baud rate of G200 matches that of the AG. If not, adjust them to be consistent. Refer to the configuration page in G200 and the AG manual Parameter for specific modification instructions
When connected to G200, the front wheel gyroscope significant bits is valid, but the front wheel angle is abnormal	Please investigate the following factors: 1. Whether the AG has completed initialize, as correct front wheel angle output requires initialize to be completed; 2. Whether the wheelbase is configured—failure to configure the wheelbase will result in invalid front wheel steering angle
After CAN configuration and arm length, Dual antenna angle, etc., the result does not match the configuration	Check whether configured as follows: little-endian mode, little endian, unit in CM, negative numbers require two's complement
Can NMEA protocol and binary protocol be output simultaneously?	Not supported, only one of the two protocols can be output at a time
When connected to G200, the front wheel steering angle is asymmetric when turned fully left or fully right	Please investigate the following factors: 1. Whether G200 is installed horizontally; 2. If powered on only when stationary, power on at 0 degrees, or manually drive straight for a while to correct it
Why are there still satellites detected when the antenna is unplugged?	UM982 has strong signal tracking capability. If there are signal sources nearby, the system may continue tracking for a period via coupling even after the antenna is disconnected. In actual satellite loss-of-lock scenarios, the antenna remains connected, and external signals cannot enter, which does not affect usage
Cover the antenna with a metal pot lid to simulate loss of lock, status remains fixed solution	Incorrect test method: The metal pot lid can only block signals from directly above; lateral signals may reflect off the roof to the bottom of the pot lid and then reflect back to the antenna, causing a false fix
ROS driver cannot read data	The device was not configured to output combined data stream by default at power-up; configuring default output of combined data stream at power-up can resolve this

Large heading error	The following factors may be responsible: 1. Incorrect lever arm and axle distance configuration, with units magnified by a factor of 10, 2. Confirm whether rigidly fixed 3. Whether far from strong vibration sources
Excessive error compared to reference attitude	Confirm whether installation error has been excluded during statistics (typically, the mean error between the device and the reference is considered the installation error)
How to perform two's complement operation when configuring CAN commands in little-endian mode	Two's complement is applied to negative numbers. For -20, X - 65536 = -20, so X = 65516. The hexadecimal value of 65516 is FFEC. The hexadecimal value of 65535 is FFFF. Since counting starts from 0, the maximum value is 65536. The two's complement value is obtained by subtracting the current result from the maximum value.
What impact does having copper poured under the device have?	No effect. Just avoid routing high-frequency signals.

Forsense (Shanghai) Technology Co., Ltd

20. Update Log

Latest version of the manual: IMU614E-AG_Datasheet_Product Manual（en）

Version	Date	Status/Comment
Version 1.0	2023.07.28	First release
Version 1.1	2023.10.07	Update coordinate system definition
Version 1.2	2023.12.14	Add attachment
Version 1.3	2024.02.04	Update electrical characteristics
Version 1.4	2024.02.27	Update reflow soldering profile and ESD protection precautions
Version 1.5	2024.03.26	Add slope mode / usage description in combination with G200
Version 1.6	2024.06.12	1. Add some new AT commands 2. Add CAN protocol latitude and longitude, Elevation message 3. Add the FAQs list
Version 1.7	2024.05.10	Add connection PC software example
Version 1.8	2024.07.05	1. Added some new AT commands 2.Add latitude and longitude to the CAN protocol, Elevation message 3. Add FAQs list
Version 1.9	2024.09.30	1. Add support for RTK input of NMEA protocol content 2. Add recommended pad size
Version1.10	2025.01.22	Adjust the table and figure order
Version 1.11	2025.04.21	Delete slope mode
Version 1.12	2025.12.03	Update outline dimension drawing
Version 1.13	2025.12.31	IncreaseCANInstruction