RAC High Precision Satellite Positioning Receiver Specification

RAC high precision GNSS receiver

Achieve high-precision satellite positioning without differential stations and ground-based augmentation networks

Real-time Array Calibration

Product Model: RAC-M1

Hardware version: RAC-C000015R_V1.0

Software Version: GXN5-BSS-UV1.23-180602

Document Revision History

Version	Revision Date	Version Description
V1.0	2016/5/18	New Document
V1.1	2016/12/15	Performance index parameters are detailed
V1.2	2017/6/18	Input command optimization changes
V2.1	2021/6/09	Detail optimization

Disclaimer

Shenzhen BroadGNSS Technology Co., Ltd. reserves the right to modify this manual at any time without prior notice. The company does not assume any form of warranty and assumes no responsibility for errors contained in this manual or for incidental or consequential damages caused by this manual.

Table of contents

1. Product Description

1.1 Product Overview

Product Name:RAC-M1High-precision satellite positioning receiver

Figure 1.1 Appearance of RAC-M1 high-precision satellite positioning receiver

This product is a G-MOUSE-like GPS receiver, a complete GPS satellite positioning receiver with a built-in satellite antenna. It offers comprehensive high-precision positioning capabilities, with dynamic accuracy reaching less than 1.5 meters. It meets the strict requirements of industrial-grade positioning and the needs of personal use.

Through technological innovation, the product has two core competitive advantages: it does not require differential stations or the use of L2 or B3 signals, which significantly reduces costs. Unlike traditional high-precision satellite positioning technology, our technology eliminates the reliance on differential techniques for high-precision positioning, resulting in a substantial reduction in cost. This is our key advantage, making our product a global leader in its category. The product features high precision, high sensitivity, low power consumption, and a compact size. Its extremely high tracking sensitivity greatly expands the coverage area for positioning.

1.2 Product Features

UART (3.3V TTL level)/RS232/IIC interface output for faster applications.

Equipped with KDS 0.5PPM high-precision TCXO.

Rich data output rates: 115200bps (default) [optional: 9600bps].

Output protocol: NMEA 0183 V3.0/UBX.

Supports adjustable data refresh rates: 1Hz to 10Hz.

Self-developed antenna oscillator ensures phase center and geometric center alignment, minimizing measurement errors.Supports optional A-GPS services, magnetometer, and pressure sensor.

GPS, BD, and GLONASS hybrid engine options available.

It is manufactured using lead-free processes and complies with RoHS standards.

1.3 Certification Testing

The positioning chip is AEC-Q100 certified.

The positioning module passes ISO16750 testing.

The positioning product passes ISO7637 testing.

The production line complies with ISO/TS-16949 certification.

1.4 Application

Used in vehicle navigation and law enforcement devices, as well as other portable devices.
Used in drones and automotive anti-theft systems, as well as trajectory tracking products.
Applied in area measurement and distance measurement for surveying products.
High-precision positioning applications in shared electric vehicles, shared cars, and similar services.
High-precision applications in the agricultural plant protection field.
Used in synchronization, UTC, time, and timekeeping applications.
Used in lane-level trajectory recording and GPS data point calibration.

1.5 Performance Indicators

Sensitivity	Track	- 165dBm
	Capture	- 148dBm
TTFF (Time to First Fix)	Cold Start	Average 23S (Open Sky)
	Warm start	Average<5S
	Hot Start	Average<1S
Type	22 Tracking channels, 56 Capture channels	
	Frequency GPS L1 1575.42MHz BDS B1 1561.75MHz	C/Acode
Positioning accuracy	-	Static
	<1.5m	Dynamic

	Coordinate system	Coordinate basis: WGS-84
Speed accuracy	0.1m/s	
Time pulse (configurable)	1s (default)	
Update rate	1-10Hz (default 1Hz, others need to be customized)	
PPS	100ms（1pps=1Hz=1times/second)	
Operational restrictions	High	<18000m
	Speed	<100m/s
	Acceleration	<8g
Data Format	NMEA 0183 general protocol, default 115200 Baud rate, refresh rate 1Hz	

Power Specifications

Power supply	DC 5.0V (Ripple voltage Vpp<200mv）
Working current	50mA Peak 60mA
Standby current	<5uA
Functional Interface	UART/RS232

Physical properties

Temperature	Operating temperature	- 30℃~80℃
	Storage temperature	- 40℃~85℃

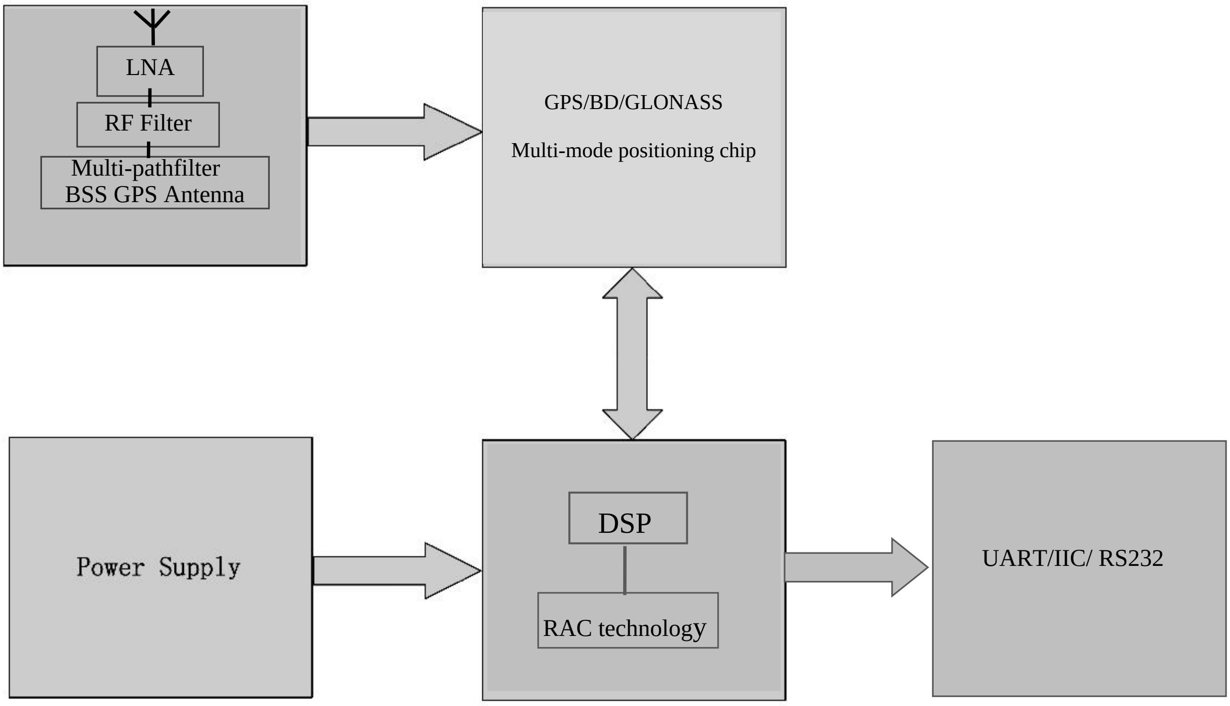
1.6 Product port definition

The product output can support UART or IIC output, etc. The default is UART output (IIC, etc. can be expanded). The specific port definition is shown in the figure below:

Pin NO.	Pin Name	I/O	Remark
1	GND	land	Ground (yellow line)
2	VCC	power supply	DC 5.0VPeak current60mA (Red line)
3	RX	enter	TTL Serial port receiving (blue line)

4	TX	Output	TTL Serial port send (white line)
5	SDA	data	Reserved geomagnetic IIC data (green)
6	SCL	clock	Reserved geomagnetic IIC clock (orange)

1.7 Functional Block Diagram



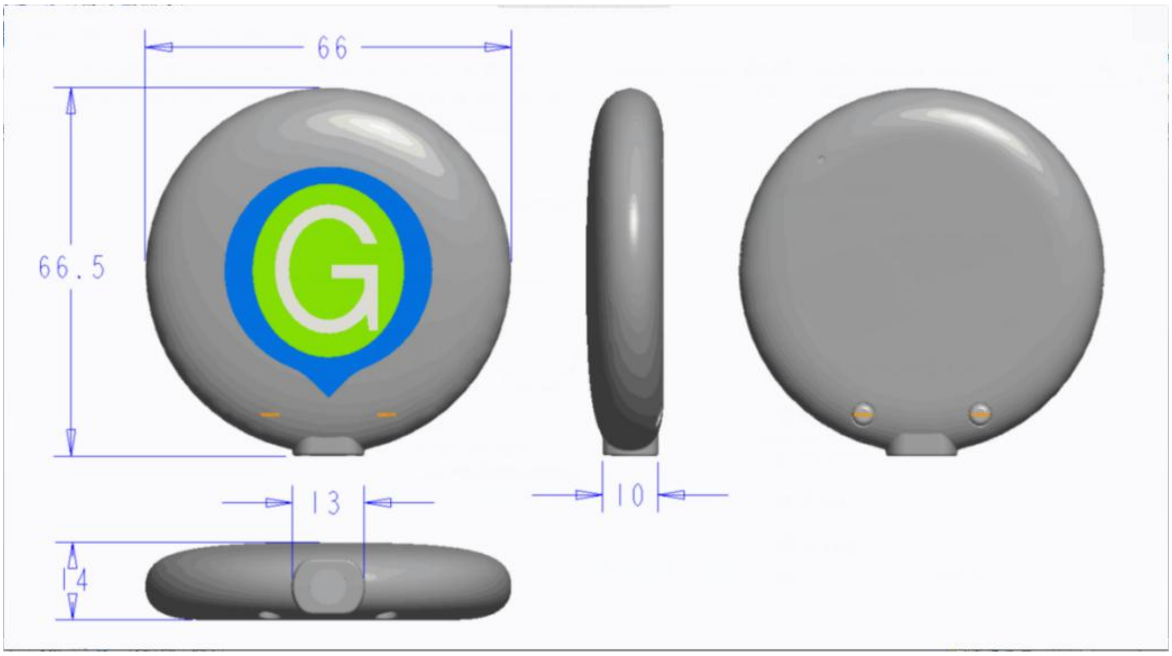
The RAC-M1 high-precision positioning receiver is equipped with a high-precision antenna and a self-developed antenna oscillator to minimize measurement errors. The antenna receives satellite signals, which are transmitted to the chip's RF unit. The RF unit processes the signals and sends them to the baseband unit. The processed data is then output as high-precision positioning data and 1PPS signals via serial port and RS232 interfaces. Additionally, the IIC interface outputs magnetometer data.

130mm

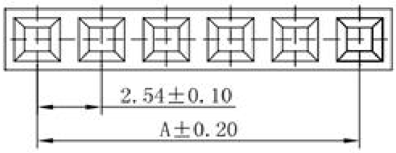
1.8 Product size

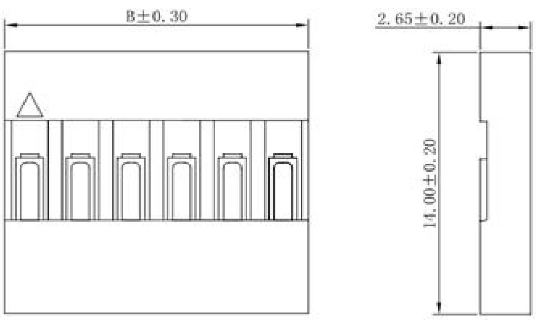
Overall size: diameter 66mm (cable outlet is about 66.5mm)*height 14mm

PCBSize: Diameter59mm*high8mm



1.9 Interface terminal size





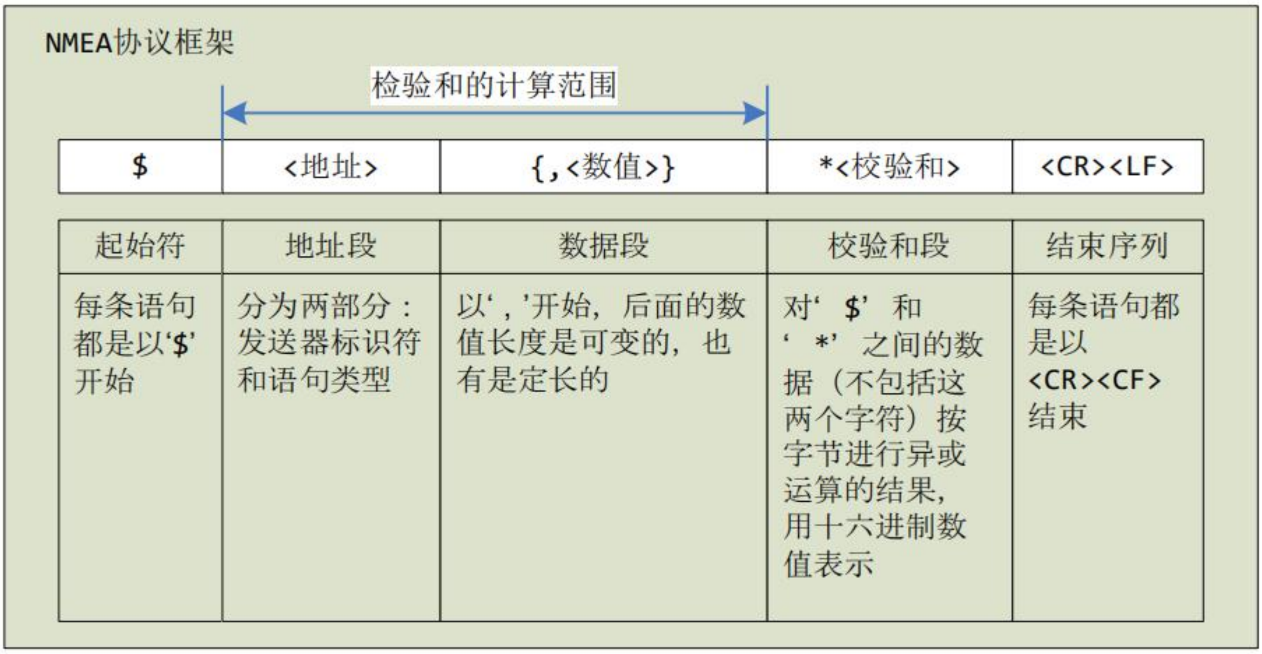
Dimensions: 7.62*10.16mm
1*4P with a 2.54mm pitch, DuPont terminal.

2. Output Statement

2.1 Output Statement

2.1.1 NMEA 0183protocol

NMEA 0183 is a standard developed by the National Marine Electronics Association (NMEA) for marine electronic devices. It has become the unified RTCM (Radio Technical Commission for Maritime Services) standard protocol for GPS navigation equipment. The protocol uses ASCII characters to transmit GPS positioning information, and the data format protocol framework is structured accordingly.



Statement output:

	Information	Illustrate
1	RMC	Recommended minimum dedicated navigation data
2	VTG	Ground speed and heading
3	GGA	Receiver positioning data
4	GSA	DOP and effective satellites
5	GSV	Visible satellites
6	GLL	Geographic location - Latitude/Longitude

2.1.1.1 Minimum dedicated navigation data recommended by RMC

Information	RMC
Describe	Recommended minimum positioning information
Type	Output

Format	$--RMC,UTCtime,status,lat,uLat,lon,uLon,spd,cog,UTCdate,mv,mvE,mode*CS<CR><LF>		
Example	$GNRMC,084103.00,A,2233.39544,N,11356.55665,E,0.035,,220618,,,A*7A		
Parameter Description			
Fields	name	Format	Parameter Description
1	$--RMC	String	Message ID, RMC statement header, '--' is the system identifier
2	UTC time	hhmmss.ss	The current location's UTC time
3	status	character	Position valid flag. V = invalid data A = valid data
4	lat	ddmm.mmmmm	Latitude, the first two characters represent degrees, the following characters represent minutes
5	uLat	character	Latitude direction: N-North, S-South
6	lon	dddmm.mmmmm	Longitude, the first 3 characters represent degrees, the following characters represent minutes
7	uLon	character	Longitude direction: E-East, W-West
8	spd	Numeric	Ground speed, in knots
9	cog	Numeric	True heading over ground, in degrees
10	UTC date	ddmmyy	Date (dd for day, mm for month, yy for year)
11	mv	Numeric	Magnetic declination, in degrees. Fixed to be empty
12	mvE	character	Magnetic declination direction: E-East, W-West. Fixed to empty
13	Mode	character	Mode indication (A = autonomous positioning, D = differential, R = RTK, E = estimation, N = invalid data)
14	navStatus	character	Navigation status indicator (V means the device does not provide navigation status information)NMEA v4.1 and above only
15	CS	Hexadecimal value	Checksum, the XOR result of all characters between $ and * (excluding $ and *)
16	<CR><LF>	character	Carriage return and line feed

2.1.1.2 VTG Ground Speed and Course

Information	VTG		
Describe	Speed over ground and course over ground information.		
Type	Output		
Format	$--VTG,cogt,T,cogm,M,sog,N,kph,K,mode*CS<CR><LF>		
Example	$GNVTG,75.20,T,,M,0.009,N,0.017,K,A*02		
Parameter Description			
Fields	name	Format	Parameter Description
1	$--VTG	String	Message ID, VTG statement header, '--' is the system identifier
2	cogt	Numeric	Course relative to true north, in degrees
3	T	character	True north indication, fixed at T
4	cogm	Numeric	Heading relative to magnetic north, in degrees
5	M	character	Magnetic north indication, fixed at M
6	sog	Numeric	Ground speed, in knots
7	N	character	Speed unit is knot, fixed as N
8	kph	Numeric	Ground speed in kilometers per hour
9	K	character	Speed unit, kilometers per hour, fixed as K

10	mode	character	Mode indication (A = autonomous positioning, D = differential, R = RTK, E = estimation, N = invalid data)
11	CS	Hexadecimal value	Checksum, all characters between $ and * (excluding $ and *)XOR result
12	<CR><LF>	character	Carriage return and line feed

2.1.1.3 GGA receiver positioning data

Information	GGA		
Describe	Receiver time, location and positioning related data		
Type	Output		
Format	$--GGA,UTCtime,lat,uLat,lon,uLon,FS,numSv,HDOP,msl,uMsl,sep,uSep,diff Age,diffSta *CS<CR><LF>		
Example	$GNGGA,235316.00,2959.99250,S,12000.00900,E,1,06,1.21,62.77,M, 0.00,M,,*7B		
Parameter Description			
Fields	name	Format	Parameter Description
1	$--GGA	String	Message ID, GGA statement header, '--' is the system identifier
2	UTC time	hhmmss.ss	The current location's UTC time
3	lat	ddmm.mmmmm	Latitude, the first two characters represent degrees, the following characters represent minutes
4	uLat	character	Latitude direction: N-North, S-South
5	lon	dddmm.mmmmm	Longitude, the first 3 characters represent degrees, the following characters represent minutes
6	uLon	character	Longitude direction: E-East, W-West
7	FS	Numeric	GPS positioning status (0 = no positioning, 1 = non-differential positioning, 2 = differential positioning)
8	numSv	Numeric	The number of satellites used for positioning, 00~12
9	HDOP	Numeric	Horizontal Dilution of Precision (HDOP＜1.2 is high-precision positioning)
10	msl	Numeric	Altitude, that is, the height of the receiver antenna relative to the geoid high
11	uMsl	character	Altitude unit, meter, fixed character M
12	sep	Numeric	The distance between the reference ellipsoid and the geoid, “-” indicates The geoid is lower than the reference ellipsoid
13	uSep	character	Altitude unit, meter, fixed character M
14	diffAge	Numeric	Differential time (from the last differential signal received)Seconds, non-differential positioning, this item is empty)
15	diffStat	Numeric	The ID of the differential reference station (empty for non-differential positioning)
16	CS	Hexadecimal value	Checksum, all characters between $ and * (excluding $ and *)XOR result
17	<CR><LF>	character	Carriage return and line feed

2.1.1.4 GSA DOP and effective satellites

Information	GSA		
Describe	Satellite number and DOP information used for positioning.		
Type	Output		
Format	$--GSA,smode,FS{,SVID},PDOP,HDOP,VDOP,systemId*CS<CR><LF>		
Example	$GNGSA,A,3,05,21,31,12,18,29,,,,,,,2.56,1.21,2.25,1*01		
Parameter Description			
Fields	Name	Format	Parameter Description
1	$--GSA	String	Message ID, GSA statement header, '--' is the system identifier
2	smode	character	Mode switch indication (M = Manual, A = Automatic)
3	FS	number	Positioning status flag (1 = not positioned, 2 = 2D positioning, 3 = 3D positioning)
4	{,SVID}	Numeric	Satellite number used for positioning
5	PDOP	Numeric	Position Dilution of Precision (PDOP)
6	HDOP	Numeric	Horizontal Dilution of Precision (HDOP)
7	VDOP	Numeric	Vertical Dilution of Precision (VDOP)
8	System Id	Numeric	GNSS system ID defined by NMEA (Note [1])NMEA v4.1 and above only
9	CS	Hexadecimal value	Checksum, all characters between $ and * (excluding $ and *)XOR result
10	<CR><LF>	character	Carriage return and line feed
Remark[1] GNSS system ID			
System ID	Describe		
1	GPS system		
2	GLONASS system		
4	BDS system		

2.1.1.5 GSV visible satellite information

Information	GSV		
Describe	The satellite number and its elevation angle, azimuth angle, carrier-to-noise ratio, etc. of the visible satellite. The number of {satellite number, elevation angle, azimuth angle, carrier-to-noise ratio} parameter groups in each GSV sentence is variable, with a maximum of 4 groups and a minimum of 0 groups.		
Type	Output		
Format	$--GSV,numMsg,msgNo,numSv{,SVID,ele,az,cn0}*CS<CR><LF>		
Example	$GPGSV,3,1,12,02,39,117,25,04,02,127,,05,40,036,24,08,10,052,*7E $GPGSV,3,2,12,09,35,133,,10,01,073,,15,72,240,22,18,05,274,*7B $GPGSV,3,3,12,21,10,316,,24,16,176,,26,65,035,42,29,46,277,18*7A		
Parameter Description			
Fields	Name	Format	Parameter Description
1	$--GSV	String	Message ID, GSV statement header, '--' is the system identifier

2	numMsg	number	Each GSV sentence outputs information about up to 4 visible satellites. Therefore, when the system has more than 4 visible satellites, Multiple GSV statements will be required.
3	msgNo	number	Current statement number
4	numSv	Numeric	Total number of visible satellites
5	{,SVID,ele , az,cn0}	Numeric	They are: Satellite number; Elevation angle, ranging from 0 to 90, in degrees; Azimuth angle, ranging from 0 to 359, in degrees; Carrier-to-noise ratio, ranging from 0 to 99, in degrees. dB, if the current satellite is not tracked, the output is empty.
6	signalId	Numeric	GNSS signal ID defined by NMEA (0 represents all signals)NMEA v4.1 and above only
7	CS	Hexadecimal value	Checksum, all characters between $ and * (excluding $ and *)XOR result
8	<CR><LF>	character	Carriage return and line feed

2.1.1.6 GLL Geographic Location – Latitude/Longitude

Information	GLL		
Describe	Information such as latitude, longitude, positioning time and positioning status.		
Type	Output		
Format	$--GLL,lat,uLat,lon,uLon, UTCtime,valid,mode*CS<CR><LF>		
Example	$GNGLL,2959.99250,S,12000.00900,E,235316.00,A,A*4E		
Parameter Description			
Fields	Name	Format	Parameter Description
1	$--GLL	String	Message ID, GLL statement header, '--' is the system identifier
2	lat	ddmm.mmmmm	Latitude, the first two characters represent degrees, the following characters represent minutes
3	uLat	character	Latitude direction: N-North, S-South
4	lon	dddmm.mmmmm	Longitude, the first 3 characters represent degrees, the following characters represent minutes
5	uLon	character	Longitude direction: E-East, W-West
6	UTC time	hhmmss.ss	The current location's UTC time
7	valid	character	Data status (A = valid positioning, V = invalid positioning)
8	mode	character	Mode indication (A = autonomous positioning, D = differential, E = estimated, N = invalid data)
9	CS	Hexadecimal value	Checksum, all characters between $ and * (excluding $ and *)XOR result
10	<CR><LF>	character	Carriage return and line feed

2.1.2 UBXprotocol

The UBX protocol is a proprietary format developed by u-blox, and it has the following main features:

1. Compact: Uses 8-bit binary data.
2. Checksum Protection: Implements a low-overhead checksum algorithm for error detection.
3. Modular: Employs a two-stage message identifier, consisting of a class and a message ID.

The structure of a UBX protocol packet includes three parts:

Header: The header is two bytes, 0xB5 0x62, which signals the start of a UBX protocol message.
CLASS: A single byte representing the message category.
ID: A single byte representing a specific parameter within the CLASS.
LENGTH: Indicates the length (in bytes) of the data section.
CK_A and CK_B: Two checksum bytes used to verify the integrity of the data from the CLASS to the Payload.

SYNC Char1	SYNCChar2	CLASS	ID	LENGTH (little endian)	PayLoad	CK_A	CK_B

The following introduces the commonly used UBX data packets.

1. UBX-NAV-POSLLH (0x01 0x02)

Message	NAV-POSLLH					
Description	Geodetic Position Solution					
Message Structure	Header	Class	ID	Length (Bytes)	Payload	Checksum
	0xB5 0x62	0x01	0x02	28	see below	CK_A CK_B
Payload Contents:						

Byte Offset	Number Format	Scaling	Name	Unit	Description
0	U4	-	iTOW	ms	GPS time of week of the navigation epoch.
4	I4	1e-7	lon	Deg	Longitude
8	I4	1e-7	lon	Deg	Latitude
12	I4	-	height	mm	Height above ellipsoid
16	I4	-	oeLh	mm	Height above mean sea level
20	U4	-	Hc	mm	Horizontal accuracy estimate

24	U4	-	vv	mm	Vertical accuracy estimate

2. UBX-NAV-STATUS (0x01 0x03)

Message	NAV-STATUS					
Description	Receiver Navigation Status					
Message Structure	Header	Class	ID	Length (Bytes)	Payload	Checksum
	0xB5 0x62	0x01	0x03	16	see below	CK_ACK_B

Payload Contents:					
Byte Offset	Number Format	Scaling	Name	Unit	Description
0	U4	-	iTOW	ms	GPS time of week of the navigation epoch.
4	U1	-	gpsFix	-	GPSfix Type, this value does not qualify a fix as valid and within the limits. See note on flag gpsFixOk below. 0x00 = no fix 0x01 = dead reckoning only 0x02 = 2D-fix 0x03 = 3D-fix 0x04=GPS+dead reckoning combined 0x05 = Time only fix 0x06..0xff = reserved
5	X1	-	flags	-	Navigation Status Flags
6	X1	-	fixStat	-	Fix Status Information
7	X1	-	flags2	-	further information about navigation output

8	U4	-	ttff	ms	Time to first fix (millisecond time tag)
12	U4	-	msss	mm	Milliseconds since Startup/Reset

3. UBX-NAV-DOP (0x01 0x04)

Message	NAV-DOP					
Description	Dilution of precision					
Message Structure	Header	Class	ID	Length (Bytes)	Payload	Checksum
	0xB5 0x62	0x01	0x04	18	see below	CK_A CK_B
Payload Contents:						
Byte Offset	Number Format	Scaling	Name	Unit	Description	
0	U4	-	iTOW	ms	GPS time of week of the navigation epoch.	
4	U2	0.01	gDOP	-	Geometric DOP	
6	U2	0.01	pDOP	-	Position DOP	
8	U2	0.01	TDOP	-	Time DOP	
10	U2	0.01	vDOP	-	Vertical DOP	
12	U2	0.01	HkDOP	-	Horizontal DOP	
14	U2	0.01	nDOP	-	Northing DOP	
16	U2	0.01	eDOP	-	Easting DOP	

4. UBX-NAV-SOL (0x01 0x06)

Message	NAV-SOL					
Description	Navigation Solution Information					
Message Structure	Header	Class	ID	Length (Bytes)	Payload	Checksum
	0xB5 0x62	0x01	0x06	52	see below	CK_A CK_B

Payload Contents:					
Byte Offset	Number Format	Scaling	Name	Unit	Description
0	U4	-	iTOW	ms	GPS time of week of the navigation epoch.
4	I4	-	f	ns	Fractional part of iTOW (range: +/-500000). The precise GPS time of week in seconds is: (iTOW * 1e-3) + (fTOW * 1e-9)
8	I2	-	week	weeks	GPS week number of the navigation epoch
10	U1	-	gpsFix	-	GPSfix Type, range 0..5 0x00 = No Fix 0x01 = Dead Reckoning only 0x02 = 2D-Fix 0x03 = 3D-Fix 0x04 = GPS + dead reckoning combined 0x05 = Time only fix 0x06..0xff: reserved
11	X1	-	flags	-	Fix Status Flags
12	I4	-	eX	cm	ECEF X coordinate
16	I4	-	eCE	cm	ECEF Y coordinate
20	I4	-	eCEf	cm	ECEF Z coordinate
24	U4	-	pAcc	cm	3D Position Accuracy Estimate
28	I4	-	xvX	cm/s	ECEF X velocity
32	I4	-	VxD	cm/s	ECEF Y velocity
36	I4	-	VxD	cm/s	ECEF Z velocity
40	U4	-	sAcc	cm/s	Speed Accuracy Estimate
44	U2	0.01	pDOP	-	Position DOP
46	U1	-	Reserve	-	Reserved

			d1		
47	U1	-	numSV	-	Number of SVs used in Nav Solution

5. UBX-NAV-PVT (0x01 0x07)

Message	NAV-PVT					
Description	Navigation Position Velocity Time Solution					
Message Structure	Header	Class	ID	Length (Bytes)	Payload	Checksum
	0xB5 0x62	0x01	0x07	92	see below	CK_A CK_B

Payload Contents:

Byte Offset	Number Format	Scaling	Name	Unit	Description
0	U4	-	iTOW	ms	GPS time of week of the navigation epoch.
4	U2	-	year	y	Year (UTC)
6	U1	-	month	month	Month, range 1..12 (UTC)
7	U1	-	day	d	Day of month, range 1..31 (UTC)
8	U1	-	hour	h	Hour of day, range 0..23 (UTC)
9	U1	-	min	min	Minute of hour, range 0..59 (UTC)
10	U1	-	sec	s	Seconds of minute, range 0..60 (UTC)
11	X1	-	valid	-	Validity flags (see graphic below)
12	U4	-	tAcc	ns	Time accuracy estimate (UTC)
16	I4	-	nano	ns	Fraction of second, range -1e9 .. 1e9 (UTC)
20	U1	-	fixType	-	GNSSfix Type: 0: no fix 1: dead reckoning only 2: 2D-fix

					3: 3D-fix 4: GNSS + dead reckoning combined 5: time only fix
21	X1	-	Flags	-	Fix status flags
22	X1	-	flags2		Additional flags
23	U1	-	numSV	-	Number of satellites used in Nav Solution
24	I4	1e-7	lon	deg	Longitude
28	I4	1e-7	lat	deg	Latitude
32	I4	-	height	mm	Height above ellipsoid
36	I4	-	oeLh	mm	Height above mean sea level
40	U4	-	Hc	mm	Horizontal accuracy estimate
44	U4	-	vv	mm	Vertical accuracy estimate
48	I4	-	Nv	mm/s	NED north velocity
52	I4	-	ewE	mm/s	NED east velocity
56	I4	-	D	mm/s	NED down velocity
60	I4	-	gSpeed	mm/s	Ground Speed (2-D)
64	I4	1e-5	headMot	deg	Heading of motion (2-D)
68	U4	-	sAcc	mm/s	Speed accuracy estimate
72	U4	1e-5	headAcc	deg	Heading accuracy estimate (both motion and vehicle)
76	U2	0.01	pDOP	-	Position DOP
78	U1[6]	-	reserved1	-	Reserved
84	I4	1e-5	headVeh	deg	Heading of vehicle (2-D)
88	I2	1e-2	magDec	deg	Magnetic declination
90	U2	1e-2	magAcc	deg	Magnetic declination accuracy

6. UBX-NAV-VELNED (0x01 0x12)

Message	NAV-VELNED					
Description	Velocity Solution in NED					
Message Structure	Header	Class	ID	Length (Bytes)	Payload	Checksum
	0xB5 0x62	0x01	0x12	36	see below	CK_A CK_B

Payload Contents:					
Byte Offset	Number Format	Scaling	Name	Unit	Description
0	U4	-	iTOW	ms	GPS time of week of the navigation epoch.
4	I4	-	Nv	cm/s	North velocity component
8	I4	-	ewE	cm/s	East velocity component
12	I4	-	D	cm/s	Down velocity component
16	U4	-	speed	cm/s	Speed (3-D)
20	U4	-	gSpeed	cm/s	Ground speed (2-D)
24	I4	1e-5	heading	deg	Heading of motion 2-D
28	U4	-	sAcc	cm/s	Speed accuracy Estimate
32	U4	1e-5	cC	deg	Course/Heading accuracy estimate

7. UBX-NAV-SAT (0x01 0x35)

Message	NAV-SAT					
Description	Satellite Information					
Message Structure	Header	Class	ID	Length (Bytes)	Payload	Checksum
	0xB5 0x62	0x01	0x35	8+12*numxD	see below	CK_A CK_B

Payload Contents:					
Byte Offset	Number Format	Scaling	Name	Unit	Description

0	U4	-	iTOW	ms	GPS time of week of the navigation epoch.
4	U1	-	version	-	Message version (1 for this version)
5	U1	-	numSvs	-	Number of satellites
6	U2	-	reserved1	-	Reserved
8 + 12*N	U1	-	gnssId	-	GNSS identifier for assignment
9 + 12*N	U1	-	svId	-	Satellite identifier for assignment
10+ 12*N	U1	-	cno	dBHz	Carrier to noise ratio (signal strength)
11+ 12*N	I1	-	elev	deg	Elevation (range: +/-90), unknown if out of range
12+ 12*N	I2	-	azim	deg	Azimuth(range 0-360), unknown if elevation is out of range
14+ 12*N	I2	0.1	prRes	m	Pseudo range residual
16+ 12*N	X4	-	flags	-	Bitmask

8.UBX-NAV-TIMEUTC (0x01 0x21)

Message	NAV-TIMEUTC					
Description	UTC Time Solution					
Message Structure	Header	Class	ID	Length (Bytes)	Payload	Checksum
	0xB5 0x62	0x01	0x21	20	see below	CK_A CK_B

Payload Contents:

Byte Offset	Number Format	Scaling	Name	Unit	Description
0	U4	-	iTOW	ms	GPS time of week of the navigation epoch.
4	U4	-	tAcc	ns	Time accuracy estimate (UTC)
8	I4	-	nano	ns	Fraction of second, range -1e9 .. 1e9 (UTC)
12	U2	-	year	y	Year, range 1999..2099 (UTC)

14	U1	-	month	month	Month, range 1..12 (UTC)
15	U1	-	day	d	Day of month, range 1..31 (UTC)
16	U1	-	hour	h	Hour of day, range 0..23 (UTC)
17	U1	-	min	min	Minute of hour, range 0..59 (UTC)
18	U1	-	sec	s	Seconds of minute, range 0..60 (UTC)
19	X1	-	valid		Validity Flags

9. UBX -MON-HW (0x0A 0x09)

Message	MON-HW					
Description	Hardware Status					
Message Structure	Header	Class	ID	Length (Bytes)	Payload	Checksum
	0xB5 0x62	0x0A	0x09	60	see below	CK_A CK_B
Payload Contents:						
Byte Offset	Number Format	Scaling	Name	Unit	Description	
0	X4	-	pinSel	-	Mask of Pins Set as Peripheral/PIO	
4	X4	-	pinBank	-	Mask of Pins Set as Bank A/B	
8	X4	-	pinDir	-	Mask of Pins Set as Input/Output	
12	X4	-	pinVal	-	Mask of Pins Value Low/High	
16	U2	-	noisePerMS	-	Noise Level as measured by the GPS Core	
18	U2	-	agc	-	AGC Monitor (counts SIGHI xor SIGLO, range 0 to 8191)	
20	U1	-	aStatus	-	Status of the Antenna Supervisor State Machine (0=INIT, 1=DONTKNOW, 2=OK, 3=SHORT,4=OPEN)	
21	U1	-	aPower	-	Current PowerStatus of Antenna (0=OFF, 1=ON, 2=DONTKNOW)	
22	X1	-	flags	-	Flags	

23	U1	-	reserved1	-	Reserved
24	X4	-	usedMask	-	Mask of Pins that are used by the Virtual Pin Manager
28	U1[17]	-	VP	-	Array of Pin Mappings for each of the 17 Physical Pins
45	U1	-	jamInd	-	CW Jamming indicator, scaled (0 = no CW jamming, 255 = strong CW jamming)
46	U1[2]	-	reserved2	-	Reserved
48	X4	-	pinIrq	-	Mask of Pins Value using the PIO Irq
52	X4	-	pullH	-	Mask of Pins Value using the PIO Pull High Resistor
56	X4	-	pullL	-	Mask of Pins Value using the PIO Pull Low Resistor

10. Geomagnetic output format

Message	Earth's magnetism				
Description	Output format of geomagnetic field				
Message Structure	Header	Class	Length (Bytes)	Payload	Checksum
	0x55 0xAA	0x01	06	see below	CK_A CK_B
Payload Contents:					

Byte Offset	Number Format	Scaling	Name	Unit	Description
0	I1	-	YAxis low position	mxD	GeomagneticYaxis
1	I1	-	YAxis height	mxD	GeomagneticYaxis
2	I1	-	XAxis low position	mxD	GeomagneticXaxis
3	I1	-	XAxis height	mxD	GeomagneticXaxis
4	I1	-	ZAxis low position	mxD	GeomagneticZaxis
5	I1	-	ZAxis height	mxD	GeomagneticZaxis

2.2 Reference Program

2.2.1 NMEA0183Protocol Verification Reference Program

```
unsigned char Calc_GPS_Sum( const char* Buffer ) {

    unsigned char i, j, k, sum;
    sum = 0;
    for ( i = 1; i < 255; i++ ) //ifrom1The beginning is to flash the $ start character {

        if ( ( Buffer[i] != '*' ) && ( Buffer[i] != 0x00 ) ) //Determine the end character {

            sum ^= Buffer[i]; //GPSThe checksum algorithm isXOR
        }
        else
        {
            break;
        }
    }
    j = Buffer[i + 1]; //Take the two characters after the
    terminator k = Buffer[i + 2];

    if ( isalpha( j ) ) //Determines whether the character is an English letter. If it is an English letter, it returns a non-zero value, otherwise it returns zero. {

        if ( isupper( j ) ) //If the character is an uppercase English letter, it returns a non-zero value, otherwise it returns zero {

            j -= 0x37;//Forced to16Base }

        else {

            j -= 0x57;//Forced to16Base }

    }
    else
    {
        if ( ( j >= 0x30 ) && ( j <= 0x39 ) ) {

            j -= 0x30;//Forced to16Base
        }
    }

    if ( isalpha( k ) ) //Determines whether the character is an English letter. If it is an English
    letter, it returns a non-zero value, otherwise it returns zero. {
        if ( isupper( k ) ) //If the character is an uppercase English letter, it returns a non-
        zero value, otherwise it returns zero {
            k -= 0x37; //Forced to16Base
        }
        else
        {
            k -= 0x57; //Forced to16Base
        }
    }
```

```
        else {

                if ( ( k >= 0x30 ) && ( k <= 0x39 ) ) {

                        k -= 0x30; //Forced to16Base }

        }

        j = ( j << 4 ) + k; //Force merge to16Base //
        gps_sum = j;

        if ( sum == j )
        {
                return Valid; //Checksum OK
        }
        else {

                return Invalid; //Checksum Error
        }
}
```

2.2.2 UBXProtocol Verification Reference Program

```
//fromCLASSarrivePayLoadAll data
unsigned int CalcCheckAB(unsigned char *Bytes, unsigned char len) {

        unsigned char i,a,b ;
        unsigned int result;
result = 0;
a = 0;
b = 0;
    for ( i = 0; i < len ; i++) {

a+=Bytes[i];
b+=a;
    }
result = a<<8|b;
return result;
}
```

3. Contact Us

Shenzhen BroadGNSS Technology Co., Ltd.

BroadGNSS Technology Co.,Ltd.

Contact: Mr. LiMobile phone: 18988798557QQ:843570942

Address: Room 402, Shengye Space, No. 52, Tiezai Road, Bao'an District, Shenzhen

www.broadgnss.com

4. Appendix 1 (Test Method)

4.1 Overview

This document mainly introduces the basic test methods of high-precision satellite positioning receivers, mainly to detect whether the equipment is powered on and works normally and outputs data, and whether the positioning accuracy is normal.

4.2 Test content

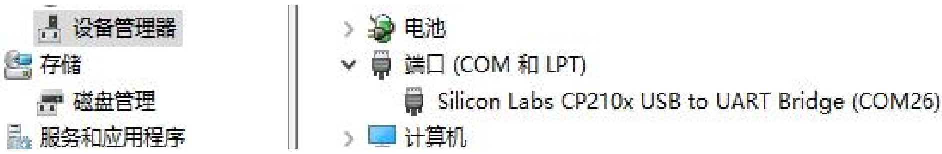
4.2.1Test Preparation

1. Install the USB to TTL serial port board driver (CH340).
2. Install serial port tools on the PC, such as the "sscom5.12" serial tool.
3. Connect the serial port board to the locator.

北斗GPS接收机 模块接口端		USB TO TTL 串口板		
黑线 GND	↔	GND		
红线 DC5V	↔	DC5V	↔	电脑USB端口
黄线 RX	↔	TX		
白线 TX	↔	RX		

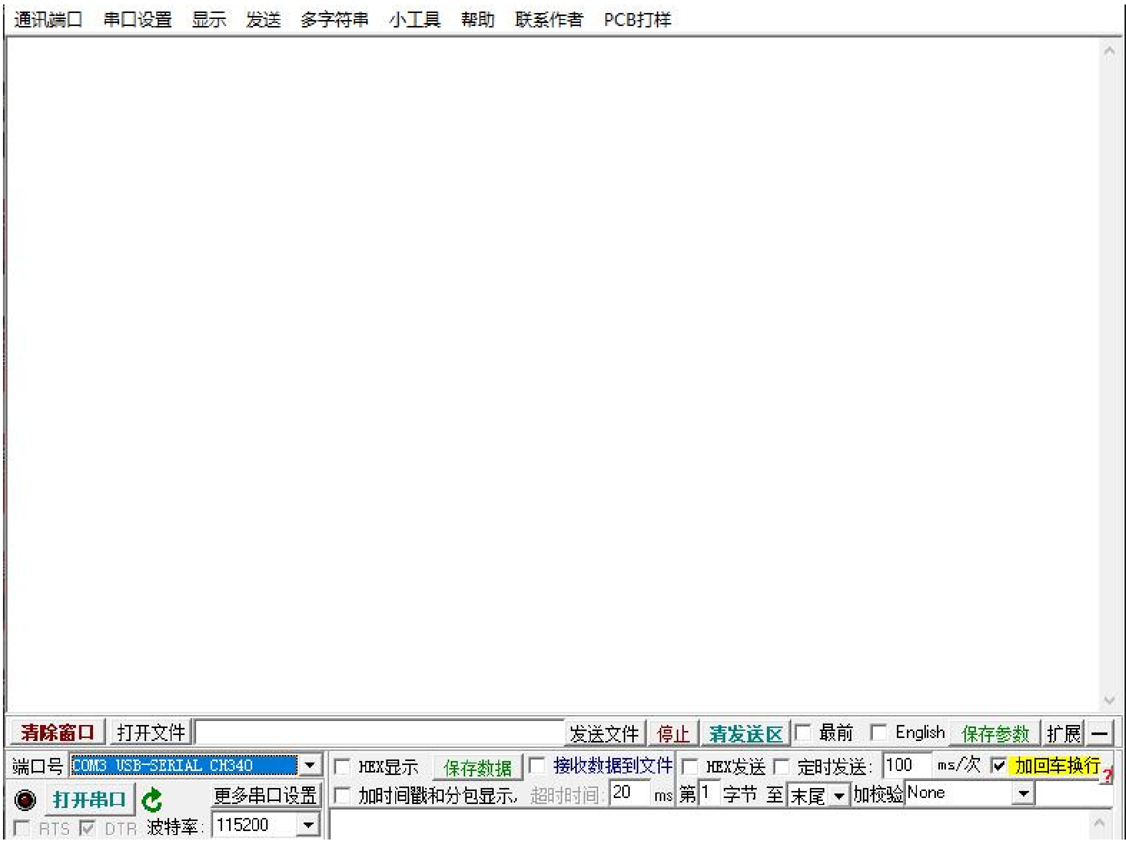


4. Ensure the test environment is in an open area with a clear sky.
5. After completing the above steps, check the computer's device manager to ensure the corresponding COM port driver is functioning correctly.



4.2.2 Test steps

1. Open serial port tools, such as "sscom5.12" or HyperTerminal, and select a baud rate of 115200 and the corresponding COM port number.



If the data exceeds 1MB, check the "Receive data to file" option. Also, ensure the "Add carriage return and line feed" option is checked before testing.

(2) When the serial port outputs the expected information, the device is successfully positioned.

$GPRMC,033301.30,A,2236.361122,N,11351.963318,E,0.005,,280917,,,A*70
$GPRMC,033301.40,A,2236.361121,N,11351.963311,E,0.004,,280917,,,A*7C
$GPRMC,033301.50,A,2236.361124,N,11351.963312,E,0.008,,280917,,,A*77
$GPRMC,033301.60,A,2236.361124,N,11351.963312,E,0.004,,280917,,,A*78
$GPRMC,033301.70,A,2236.361123,N,11351.963306,E,0.004,,280917,,,A*7B
$GPRMC,033301.80,A,2236.361123,N,11351.963306,E,0.007,,280917,,,A*77
$GPRMC,033301.90,A,2236.361124,N,11351.963297,E,0.004,,280917,,,A*7B
$GPRMC,033302.00,A,2236.361124,N,11351.963297,E,0.006,,280917,,,A*73

$GPRMC,033302.10,A,2236.361123,N,11351.963296,E,0.005,,280917,,,A*77
$GPRMC,033302.20,A,2236.361123,N,11351.963290,E,0.007,,280917,,,A*70
$GPRMC,033302.30,A,2236.361123,N,11351.963290,E,0.005,,280917,,,A*73

4.2.3 Placement