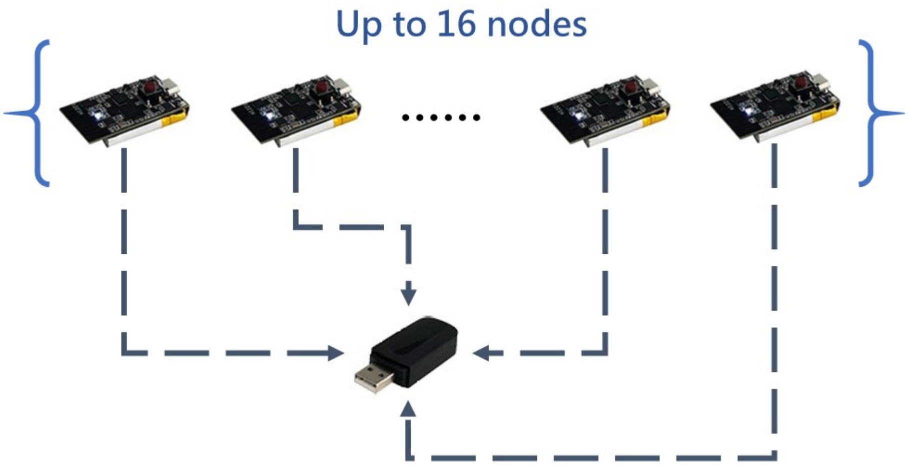
Hi221/Hi221 Dongle User Manual

Hi221 Wireless Transmission Module and Receiver, Rev 1.0

Table of contents

Introduction

Hi221/Hi221 IMU system is a high-performance, small-size, low-latency inertial measurement unit (IMU). This product integrates a 3-axis accelerometer, a 3-axis gyroscope, a 3-axis magnetometer and a low-power microprocessor. It can output 3-dimensional azimuth data based on local geographic coordinates calculated by the sensor fusion algorithm, including roll, pitch and heading angle. The raw sensor data can also be output. Hi221 consists of **Hi221 Dongle(receiver)** and **Hi221(attitude node)**. One Hi221 Dongle can receive up to 16 Hi221 modules form a star network structure. Each Hi221 node can output data up to 100Hz.



Features

Onboard Sensors

- Three-axis gyroscope, maximum range: ±2000°/s output rate 2000Hz
- Triaxial accelerometer, maximum range: ±8g output rate 125Hz
- Triaxial geomagnetic field sensor, maximum range: 800mG Internal sampling rate 100Hz

Communication interface and power supply

- Serial port (compatible with TTL and can be directly connected to 5V or 3.3V serial port devices)
- Supply voltage: 3.3 (+/- 100 mV)
- Maximum peak power consumption: 120mA (RF Tx transmit)

Hardware and size (nodes)

Hardware parameters

parameter	value
Output data interface	UART(TTL 1.8V - 3.3V) or 2.4RF Radio
Operating Voltage	3.3V (± 100mV)
Working current	30mA
Standby Current	20uA
Average working time after charging	8h
Temperature Range	-20℃ - 85℃
Maximum Linear Acceleration	0 - 115 m/s^2
Dimensions	20 x 38 x 8.5mm (W x L x H)
On-board sensor	3-axis accelerometer & 3-axis gyroscope & 3-axis geomagnetic sensors

Coordinate system definition

The carrier system uses the front-left-up (FLU) right-handed coordinate system, and the geographic coordinate system uses the north-west-sky (NWU) coordinate system. The rotation order of Euler angles is ZYX (the Z axis is rotated first, then the Y axis is rotated, and the X axis is finally rotated). The specific definitions are as follows:

- Rotate around the Z axis: Yaw\Yaw\phi (psi) Range: -180° to 180°
- Rotate around the Y axis: Pitch\Pitch\theta (theta) Range: -90° to 90°
- Rotate around the X-axis: roll angle\Roll\psi (phi) range: -180° to180°

If you think of the mod as an aircraft. The X-axis should be considered the direction of the machine head. When the sensor frame coincides with the inertial frame, the ideal output of Euler angles is: Pitch = 0°, Roll = 0°, Yaw = 0°

Performance

Attitude angle output accuracy

Attitude Angle	Typical Value
Roll\Pitch - Static	0.8°
Roll/Pitch - Dynamic	2.5°
Heading angle accuracy in motion (in 9-axis mode, no magnetic interference, after calibration)	3°

Gyro

parameter	value
Measuring range	±2000°/s
Nonlinearity	±0.1%
Sampling Rate	2000Hz

Accelerometer

parameter	value
Measuring range	±8G
Nonlinearity	±0.5%
Maximum zero offset	30mG
Sampling Rate	125Hz

Magnetic sensor parameters

parameter	value
Measuring range	±8Gauss
Nonlinearity	±0.1%
Sampling Rate	100Hz

Module data interface parameters (UART)

parameter	value
Serial port output baud rate	115200/460800/921600 optional
Frame output rate	1/25/50/100Hz optional

Module data interface parameters (2.4G RF)

parameter	value
Air Baud Rate	2Mbps
Frame rate (per node)	100Hz
Maximum number of nodes connected to the receiver	16

Sensor calibration

1. The accelerometer and gyroscope are calibrated for scale factor error and non-orthogonal error before leaving the factory, and the calibration parameters are stored in the module.
2. The output of the gyroscope will have a random non-zero bias after each power-on, which is called zero bias repeatability. This random bias cannot be calibrated at the factory. The system is considered to be in a static state within 1S after startup and collects the 1S gyroscope angular velocity as the initial bias. This process is called gyroscope power-on self-calibration. Self-calibration can obtain the best calibration effect when the module is stationary for about 3s after power-on. **If the module is in motion (including slow rotation, vibration, etc.) within 1s of power-on, the heading angle drift phenomenon will increase significantly**. It is recommended to stand still for at least 1s after each power-on.
3. The magnetic sensor (supported by some models) has been calibrated by ellipsoid before leaving the factory, but the magnetic sensor is easily disturbed by the external magnetic field, and generally needs to be re-calibrated by the customer. The factory calibration parameters are of little significance. See the Geomagnetic Calibration section for details.

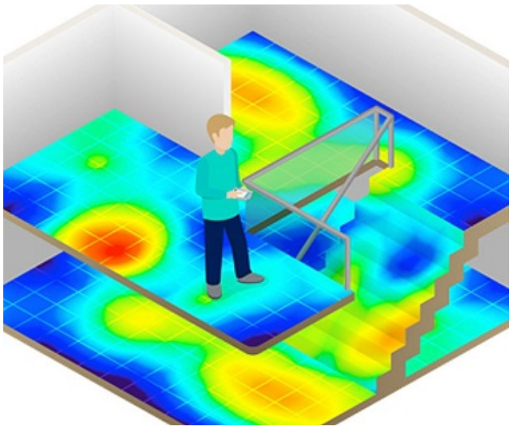
Geomagnetic Calibration

Magnetic Interference Classification

Types	Definitions	Typical sources of interference	Impacts	Measures
Space magnetic field interference	The interference does not move with the movement of the sensor, but is in the world coordinate system	Various fixed magnetic interference sources, furniture, household appliances, cables, steel structures in houses, etc. All interference sources that do not move with the movement of the magnetic sensor	Regardless of whether the magnetic field sensor is well calibrated, the interference of these spatial magnetic fields (or the uneven environmental magnetic field) will distort the spatial geomagnetic field. Geomagnetic compensation will be wrong and the correct heading angle will not be obtained. They are the main culprit that makes indoor geomagnetic fusion difficult to use. This disturbance cannot be calibrated and can seriously affect the geomagnetic performance. Space magnetic field interference is especially serious indoors.	Homogeneous magnetic field detection and shielding of non-homogeneous magnetic field built in the module

Types	Definitions	Typical sources of interference	Impacts	Measures
Interference in the sensor coordinate system	Interference source moves with the movement of the sensor	Module PCB, board fixed with the module, equipment, products, etc. They and the magnetic sensor are regarded as the same rigid body, and they move with the movement of the magnetic sensor	causing hard magnetic/soft magnetic interference to the sensor. These disturbances can be well eliminated by the geomagnetic calibration algorithm.	Geomagnetic Calibration

The figure below is a typical indoor magnetic field distribution map. It can be seen that the spatial magnetic field distortion of the general indoor environment is relatively serious.



Notice

In the indoor environment, the spatial magnetic interference is particularly serious, and the spatial magnetic interference cannot be eliminated by calibration. In an indoor environment, although the module has a built-in homogeneous magnetic field detection and shielding mechanism, the accuracy of the heading angle of the 9-axis mode largely depends on the degree of indoor magnetic field distortion. room, workshop, underground garage, etc.), even after calibration, the heading angle accuracy of 9-axis may not be as good as that of 6-axis or even large angle errors.

Difference between 6-axis and 9-axis mode

Because the geomagnetic field is very susceptible to spatial interference, great care should be taken when using the 9-axis mode. The following table lists the usage recommendations for different usage occasions and working conditions

Mode	Applicable Environment	Typical Application	Advantages	Disadvantages	Precautio
6-axis mode	Various environments	Low dynamic attitude detection such as gimbal, indoor robot	1. Good attitude angle output stability 2. Completely free from magnetic field interference	Heading angle drifts slowly over time	cannot be compensat
9-axis mode	No magnetic interference environment	1. Compass, north finding system The surveyor will not do large-scale walking)	1. The heading angle will not drift with time 2. Once the geomagnetic field is detected, the heading angle can be quickly corrected to point north	Any magnetic interference will cause the heading angle accuracy to decline. In the case of severe indoor interference, the heading angle cannot point to the correct direction. In addition, the metal structure and motor of the mobile robot will generate very strong magnetic interference, so the mobile robot platform is not suitable for 9-axis mode.	Geomagnet sensor needs to calibrate before first use

The module's automatic geomagnetic calibration system can only deal with the fixed magnetic field interference installed with the module. If there is magnetic field interference in the installation environment, the interference must be fixed, and the distance between the interference magnetic field and the module will not change after installation (for example: the module is installed on an iron material, because iron will have magnetic field interference, At this time, the iron and the module need to be rotated and calibrated together, and the iron will not be separated from the compass (relative displacement) during use. Once separated, it needs to be re-calibrated. If the size of the iron is not fixed, or The distance change from the compass is not fixed. This kind of interference cannot be calibrated. Even if the calibration is successful, the accuracy will be very poor, and it can only be installed away from it. The safe distance is controlled above 40CM).

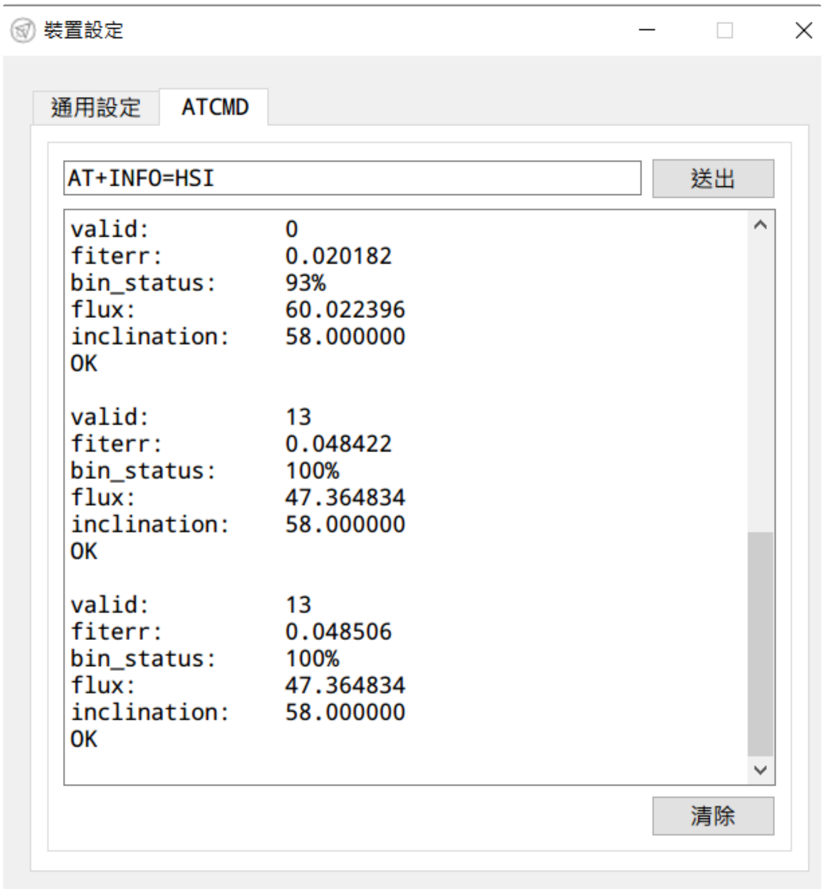
Calibration method

This module integrates an active and non-intervention geomagnetic calibration algorithm. The module will automatically collect geomagnetic field information and estimate hard and soft magnetic calibration parameters. After successful calibration, the calibration information will be saved on the module Flash. **Users can achieve geomagnetic calibration without any operation/command.** When using it for the first time, the module will automatically collect the surrounding geomagnetic field and try to calculate the calibration parameters of the geomagnetic sensor. When using the module for the first time and need to use the 9-axis mode, the following calibration operations should be performed:

In the smallest possible range, slowly let the module move and rotate, or perform a figure-of-8 movement or 360 degrees around each axis, so that the module can experience as many poses as possible. In general, if the geomagnetic interference is within an acceptable range, the calibration can be completed. If the latter is in the same geomagnetic environment (same location), there is no need to re-calibrate. If the module has not been successfully calibrated, it means that the surrounding geomagnetic field interference is relatively large.

The geomagnetic calibration status can be viewed using AT commands:

Send the `AT+INFO=HSI` command, the module will print the current state of the geomagnetic calibration system:



Parameter display	**Meaning**	**Explanation**
valid	valid flag	0: There are no valid calibration parameters (no calibration or calibration has never been successful). Non-0: Geomagnetic calibration completed
fiterr	Fitting residual	The smaller the residual, the better the parameter fitting effect, usually below 0.03, the calibration result is good enough. If the fitting result is always > 0.1, it means that the geomagnetic interference is very large, and it is best to calibrate again to get better calibration results. Fit residuals grow slowly over time.
flux	Local magnetic field	The most recent estimated geomagnetic field strength by the fitter, in uT
inclination	local magnetic inclination angle	magnetic inclination angle estimated by the most recent fitter, in °

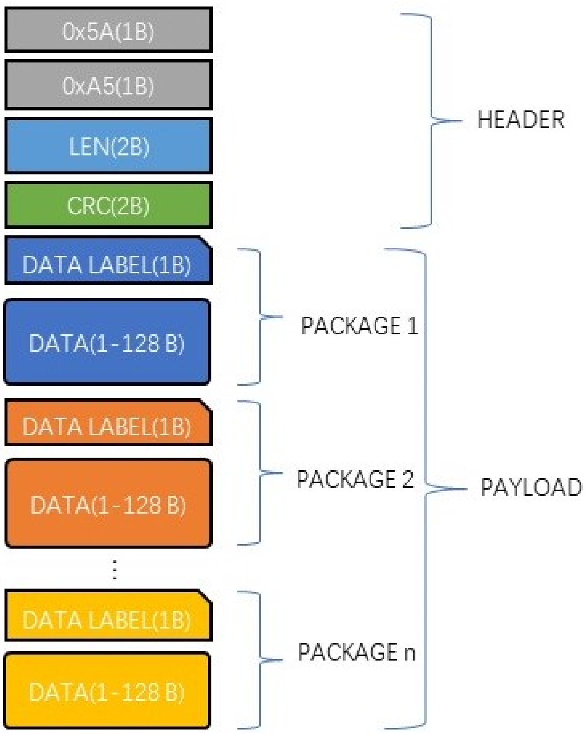
- Although the geomagnetic parameter estimation can automatically collect data online, the geomagnetic calibration parameters are automatically and dynamically fitted. However, if the surrounding geomagnetic environment changes (for example, it needs to switch to another room or indoor and outdoor), it is best to repeat the manual calibration operation.

Serial port protocol

After the module is powered on, the frame data is output at the factory frame rate (usually 100) by default. The frame format is as follows:

PRE	TYPE	PAYLOAD LENGTH(2B)	CRC (2B)	PAYLOAD (0-512B)
0x5A	0xA5	LEN	CRC	DATA

field name	value	length (bytes)	description
PRE	0x5A	1	fixed at 0x5A
TYPE	0xA5	1	fixed at 0xA5
LEN	1-512	2	The length of the data field in the frame, low byte first. Length indicates the length of the data field (PAYLOAD), excluding PRE , TYPE , LEN , CRC fields.
CRC	-	2	16-bit CRC checksum of frame data for all fields (PRE,TYPE,LEN, PAYLOAD) except CRC itself. LSB (low byte first)
PAYLOAD	-	1-512	Data carried by one frame. The PAYLOAD field consists of several **sub-packets**. Each packet contains a packet label and data. The tag determines the type and length of the data.



CRC implementation function:

```c
/*
    currectCrc: previous crc value, set 0 if it's first section
    src: source stream data
    lengthInBytes: length
*/
static void crc16_update(uint16_t *currectCrc, const uint8_t *src, uint32_t lengthInBytes)
{
    uint32_t crc = *currectCrc;
    uint32_t j;
    for (j=0; j < lengthInBytes; ++j)
    {
        uint32_t i;
        uint32_t byte = src[j];
        crc ^= byte << 8;
        for (i = 0; i < 8; ++i)
        {
            uint32_t temp = crc << 1;
            if (crc & 0x8000)
            {
                temp ^= 0x1021;
            }
            crc = temp;
        }
    }
    *currectCrc = crc;
}
```

data pack

Packet Overview

data tag	packet length (including tag 1 byte)	name	remarks
0x91	76	IMUSOL (IMU data set)	Hi221 output packet
0x62	Variable length	GWSOL (Wireless Node Data Collection)	Hi221Dongle Output Packet

0X91 (IMUSOL)

A total of 76 bytes. Integrate the sensor raw output of the IMU and the attitude solution data.

Byte Offset	Type	Size	Unit	Description
0	uint8_t	1	-	Packet label: 0x91
1	uint8_t	1	-	ID
2	-	6	-	Reserved
8	uint32_t	4	ms	Timestamp information, accumulated since the system is powered on, increments by 1 every millisecond
12	float	12	1G (1G = 1 gravitational acceleration)	X, Y, Z axis acceleration
24	float	12	deg/s	X, Y, Z angular velocity
36	float	12	uT	Magnetic field strength in X, Y, Z axes
48	float	12	deg	Node Euler angle set, in order: Roll angle (Roll), Pitch angle (Pitch), Yaw angle (Yaw)
60	float	16	-	Node quaternion set, in order WXYZ

0x62 (GWSOL)

Preset packets for wireless receivers. The first 8 bytes of the data packet are receiver information. The latter is divided into N data blocks. Each data block describes the attitude data of a node (up to 16 nodes are supported). The size of each data block is 76 bytes, and the data structure is the same as 0x91.

This protocol packet has a large amount of data. It is recommended to adjust the Hi221Dongle baud rate to 460800 or 921600 to obtain the highest frame rate output. The format is as follows:

Byte Offset	Size	Type	Unit	Description
0	1	uint8_t	-	Packet label: 0x62
1	1	uint8_t	-	GWID, receiver network ID
2	1	uint8_t	-	N, this frame contains the number of node data blocks
3	5	-	-	Reserved
----Start of node data block----	-	-	-	*Data structure is the same as 0x91*
8+76*N(N=0-15)	1	uint8_t	-	Packet label: 0x91
9+76*N(N=0-15)	1	uint8_t	-	ID of node N
10+76*N	10	-	-	Reserved
20+76*N	12	float	-	Node N triaxial acceleration
32+76*N	12	float	-	Node N triaxial angular velocity
44+76*N	12	float	-	Node N-axis magnetic field strength
56+76*N	12	float	-	Node N Euler angles
68+76*N	16	float	-	Node N Quaternion
----End of node data block----	-	-	-	-----------

Factory Default Packet

The factory default definition of packet data carried in a frame is as follows:

Products	Default Output Packets
Hi221	0x91
Hi221Dongle	0x62

Example of data frame structure

The data frame is configured as `0x91` packets

Use the serial port assistant to sample a frame of data, a total of 82 bytes, the first 6 bytes are the frame header, length and CRC check value. The remaining 76 bytes are data fields. Suppose the data is received into the C array `buf`. As follows:

5A A5 4C 00 6C 51 **91** 00 A0 3B 01 A8 02 97 BD BB 04 00 9C A0 65 3E A2 26 45 3F 5C E7 30 3F E2 D4 5A C2 E5 9D A0 C1 EB 23 EE C2 78 77 99 41 AB AA D1 C1 AB 2A 0A C2 8D E1 42 42 8F 1D A8 C1 1E 0C 36 C2 E6 E5 5A 3F C1 94 9E 3E B8 C0 9E BE BE DF 8D BE

- The first step: judge the frame header, get the data field length and frame CRC:

Frame header: 5A A5

Frame data field length: 4C 00: (0x00<<8) + 0x4C = 76

Frame CRC check value: 6C 51: (0x51<<8) + 0x6C = 0x516C

- Step 2: Check CRC

```
    uint16_t payload_len;
    uint16_t crc;

    crc = 0;
    payload_len = buf[2] + (buf[3] << 8);

    /* calulate 5A A5 and LEN filed crc */
    crc16_update(&crc, buf, 4);

    /* calulate payload crc */
    crc16_update(&crc, buf + 6, payload_len);
```

The CRC value obtained is 0x516C. The frame CRC check is passed.

- Step 3: Receive data

The data field of the packet starts from 0x91. In C language, you can define a structure to easily read data:

Define the 0x91 packet structure as follows:

```
__packed typedef struct
{
    uint8_t     tag;                /* data packet tag */
    uint8_t     id;
    uint8_t     rev[6];             /* reserved */
    uint32_t    ts;                 /* timestamp */
    float       acc[3];
    float       gyr[3];
    float       mag[3];
    float       eul[3];             /* eular angles: Roll,Pitch,Yaw */
    float       quat[4];            /* quaternion */
}id0x91_t;
```

__packed is a compiler keyword (under Keil), indicating that the structure is tightly aligned by bytes, and each element of the structure corresponds to the structure definition of the 0x91 data packet. When receiving data, you can directly memcpy the received array to the structure: (note that the structure must be aligned with 4 bytes), where buf points to the frame header, and buf[6] points to the data field in the frame.

```
    /* Receive data and interpret data using 0x91 packet structure definition */
    __align(4) id0x91_t dat; /* struct must be 4 byte aligned */
    memcpy(&dat, &buf[6], sizeof(id0x91_t));
```

Finally get the dat data result:

```
id : 0
timestamp : 310205
acc : 0.224 0.770 0.691
gyr : -54.708 -20.077 -119.070
mag : 19.183 -26.208 -34.542
eul(R/P/Y) : 48.720 -21.014 -45.512
quat : 0.855 0.310 -0.310 -0.277
```

The data frame is configured as 0x62 packets

Use the serial port assistant to sample a frame of data, a total of 166 bytes, the first 6 bytes are the frame header, length and CRC check value. The remaining 160 bytes are data fields. Suppose the data is received into the C array `buf`. As follows:

5A A5 A0 00 B5 DC **62 00 02** 00 00 00 00 00 **91** 01 00 00 00 00 00 00 00 00 00 00 EC 51 B8 BD 12 83 40 3E 64 3B 8F 3F 33 33 E7 C2 CD CC 3C C1 33 33 0B C1 9A 99 99 41 9A 99 A9 40 CD CC 94 C1 80 D3 28 41 E8 CD 9B 40 B1 45 2C C0 78 97 7E 3F 83 20 BE 5C 3D 14 24 3D 32 69 DF BC **91** 04 00 00 00 00 00 00 00 00 00 00 CD CC CC 3E 6F 12 83 3C 56 0E 6D 3F 9A 99 C7 C2 33 33 63 41 9A 99 35 42 00 00 9 33 33 63 41 00 00 D0 C1 F6 E6 A1 40 C0 93 A6 C1 FB 73 F5 C0 00 1D 7B 3F C8 E7 FF 3C DA 6C 3B BE 5E 99 6C BD

- The first step: judge the frame header, get the data field length and frame CRC:

Frame header: `5A A5`

Frame data field length: `A0 00`: (0x00<<8) + 0xA0 = 160

Frame CRC check value:`B5 DC`:(0xDC<<8) + 0xB5 = 0xDCB5

- Step 2: Check the CRC, the process is the same as the 0x91 data packet

The calculated CRC value is 0xDCB5. The frame CRC check is passed.

- Step 3: Obtain 0x62 frame network segment, number of nodes and other information

  ```c
  uint8_t gwsol_tag = buf[6];
  uint8_t gwid = buf[7]; /* GWID segment */
  uint8_t node_cnt = buf[8]; /* number of nodes received */
  ```

  GWSOL label: `0x62` = 0x62

  Receiver segment: `0x00` = 0

  Number of nodes in this frame: `0x02` = 2

- Step 4: Receive data from each node

The 0x62 data packet is actually a combination of multiple 0x91 data packets, with the header plus the receiver message. In C language, you can define 0x91 structure to read data easily:

Define the 0x91 packet structure as follows: (take Keil as an example)

```c
__packed typedef struct
{
    uint8_t tag; /* data packet tag */
    uint8_t id;
    uint8_t rev[6]; /* reserved */
    uint32_t ts; /* timestamp */
    float acc[3];
    float gyr[3];
    float mag[3];
    float eul[3]; /* eular angles:R/P/Y */
    float quat[4]; /* quaternion */
}id0x91_t;

```

`__packed` is a compiler keyword (under Keil), indicating that the structure is tightly aligned by bytes, and each element of the structure corresponds to the structure definition of the 0x91 data packet. When receiving data, just memcpy the received array directly to the structure: (note that the structure must be 4-byte aligned), where `buf` points to the data header.

```c
    /* Receive data and interpret data using 0x91 packet structure definition */
    __align(4) id0x91_t id0x91; /* struct must be 4 byte aligned */

    int i;
    for(i=0; i<node_cnt; i++)
    {
        memcpy(&id0x91, &buf[8+6] + i*sizeof(id0x91_t), sizeof(id0x91_t)); /* 8+6: 6 is the length of the frame header information, and 8 is the length of the receiver header in the 0x62 packet */

        /* process the packet data
        ...
        printf("node_id:%d\r\n", id0x91->id);
         */
    }
```

Finally, the data result is obtained: This frame of data contains 2 nodes with IDs 1 and 4 respectively. The specific information is as follows:

```
id : 1
timestamp : 0
acc : -0.090 0.188 1.119
gyr : -115.600 -11.800 -8.700
mag : 19.200 5.300 -18.600
eul(R/P/Y) : 10.552 4.869 -2.692
quat : 0.994 0.093 0.040 -0.027

id : 4
timestamp : 0
acc : 0.400 0.016 0.926
gyr : -99.800 14.200 45.400
mag : 18.500 14.200 -26.000
eul(R/P/Y) : 5.059 -20.822 -7.670
```

```
quat : 0.981 0.031 -0.183 -0.058
```

AT command

When using the serial port to communicate with the module, the module supports the AT command set to configure/view the module parameters. AT commands always start with the ASCII code `AT`, followed by control characters, and the last end with a carriage return line feed `\r\n`. You can use the serial port debugging assistant to test:

General module AT commands are as follows

Command	Function	Power-off save (Y)	Immediately effective (Y), reset effective (R)	Remarks
AT+ID	Set module user ID	Y	R	
AT+INFO	Print the information of module	N	Y	
AT+ODR	Set module serial port output frame rate	Y	R	
AT+BAUD	Set baud rate of serial port	Y	R	
AT+EOUT	Data output switch	N	Y	
AT+RST	Reset module	N	Y	
AT+TRG	Single output trigger	N	Y	Not supported
AT+SETPTL	Set output protocol	Y	Y	Default Hi221:0x91, Hi221 Dongle:0x62
AT+MODE	Set the working mode of the module	Y	R	
AT+GWID	Set wireless receiver ID	Y	R	

AT+ID

Set mod user ID

Example AT+ID=1

AT+INFO

Print module information, including product model, version, firmware release date, etc.

AT+ODR

Set the module serial port output frame rate. Save when power off. Restart the module to take effect

Example Set serial port output rate to 100Hz: AT+ODR=100

Note: When the output frame rate is set relatively high (such as 200), the default 115200 baud rate may not meet the output bandwidth requirements. At this time, it is necessary to set the module baud rate to a high value (such as 921600), so that the module can output a high output stably. frame rate.

AT+BAUD

Set baud rate of serial port, available options: `9600/115200/460800/921600`

When more than 2 nodes are connected and the frame rate is found to be less than 100Hz, BAUD must be set to 460800

Example `AT+BAUD=115200`

Notice

- Special attention should be paid to using this command. If the wrong baud rate is entered, it will lead to inability to communicate with the module
- After the baud rate parameter is set, it will be saved after power off, and the reset module will take effect. The baud rate of CH-Center software should also be modified accordingly.
- When upgrading firmware, you need to switch back to 115200 baud rate.

AT+EOUT

Serial port output switch

Example Open serial port output `AT+EOUT=1` Close serial port output `AT+EOUT=0`

AT+RST

reset module

Example `AT+RST`

AT+GWID

It can be configured through the AT+GWID command. The GWID attribute determines the RF frequency of the receiver and the node. Only when the GWID of the node and the GWID of the receiver are the same, the module and the receiver can communicate. GWID is equivalent to a wireless network segment. When using multiple receivers at the same location to form multiple star networks, it must be ensured that the GWID (network segment) of each receiver is different.

Example Three Hi221 nodes and one Hi221 Dongle receiver. Set the Hi221 Dongle network segment to GWID=0, set the self IDs of the three Hi221 nodes to 0, 1, 2 and connect to the Hi221 Dongle:

Hardware Preparation: A USB cable, a PC with the CH-Center software installed.

1. Connect the receiver (Hi221 Dongle) to the PC with a USB cable, open the CH-Center software configuration interface, and send AT commands in sequence:

   `AT+GWID=0`
   `AT+RST`

2. Connect node 0 (Hi221) to PC via USB cable, open CH-Center software configuration interface, and send AT commands in sequence:

AT+GWID=0
AT+ID=0
AT+RST

3. Connect node 1 (Hi221) to PC with USB cable, open CH-Center software configuration interface, and send AT commands in sequence:

   AT+GWID=0
   AT+ID=1
   AT+RST

4. Connect node 2 (Hi221) to PC with USB cable, open CH-Center software configuration interface, and send AT commands in sequence:

   AT+GWID=0
   AT+ID=2
   AT+RST

Note:

- The last AT+RST sent is because all configuration changes can be reset to take effect. If AT+RST is not sent, it needs to be powered on again to take effect.
- It is best to unplug other devices when connecting to each node and receiver to avoid multiple serial port selection errors.

AT+MODE

Set the mod working mode

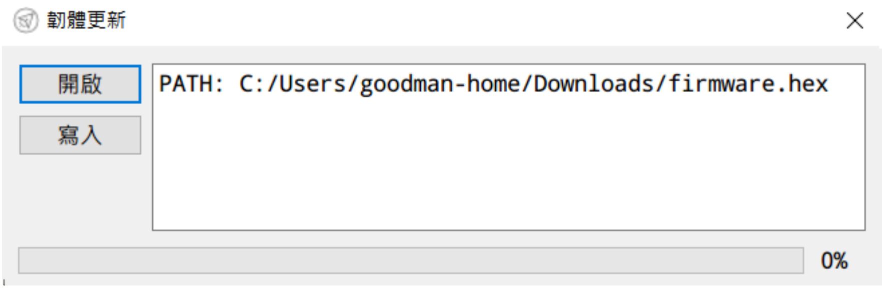
example

- Set the module to work in 6-axis mode (non-magnetic calibration) AT+MODE=0
- Set the module to work in 9-axis mode (the geomagnetic field sensor is involved in heading angle correction) AT+MODE=1

Appendix A - Firmware Upgrade and Factory Reset

This product supports firmware upgrade, and you do not need to upgrade the firmware by yourself under normal use.
Firmware upgrade steps:

- Connect the module, open the host computer, and set the baud rate of both the module and the host computer to 115200. Open the firmware upgrade window
- Click the connection button, if the module connection information appears. It means that the upgrade system is ready, click Open File, select the firmware with the filename .hex, and then click Download.
- After completion, it will prompt completion, close the serial port at this time, power on the module again, and the module upgrade is complete.


韌體更新
開啟
寫入
PATH: C:/Users/goodman-home/Downloads/firmware.hex
0%