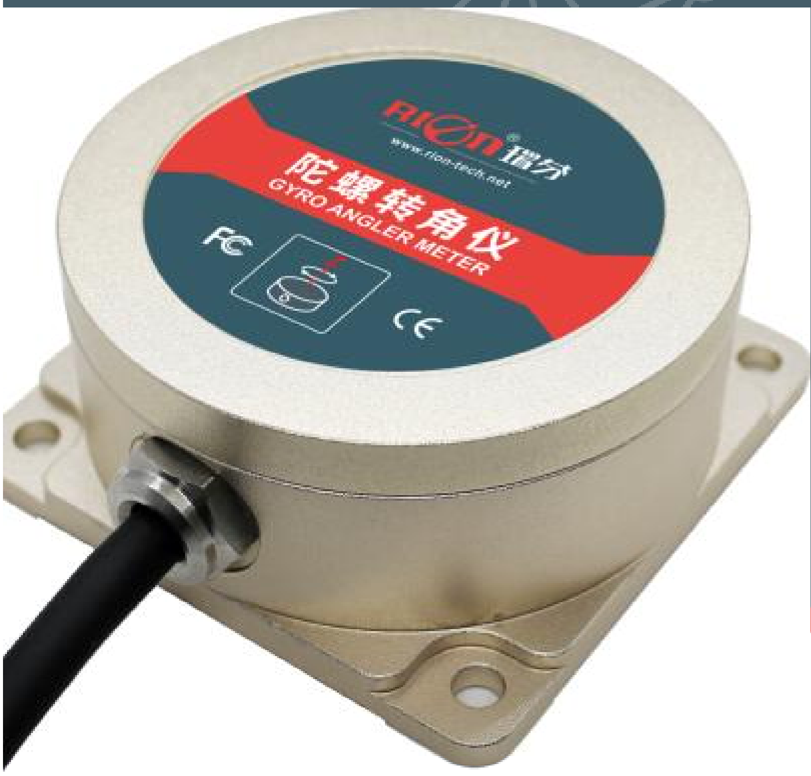


Appearance Patent No.: ZL 201730674512.0

V1.2

CAN OUTPUT ANGULAR GYRO SENSOR

RION TL740D-CAN2.0A/B

Technical Manual

RION QUALIFICATION CERTIFICATION

○ Quality management system certification: GB/T19001-2016 idt ISO19001:2015 standard (certificate No.: 128101)

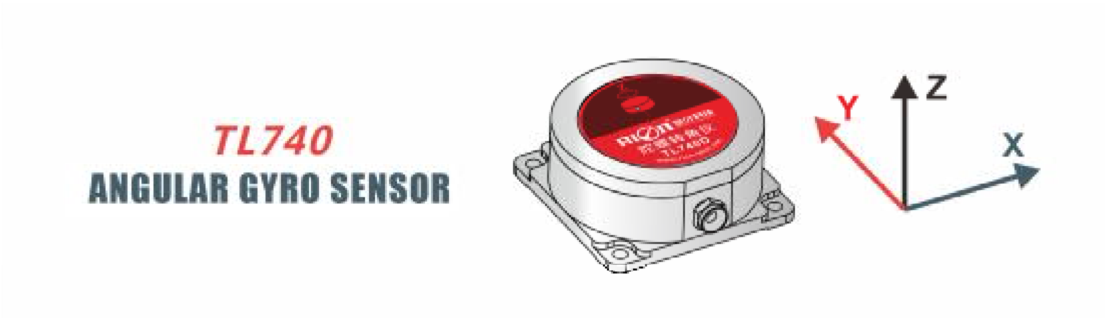
○ High-tech Enterprise (Certificate No.: GR201844204379)

○ CE certification: registration No.: AT18250EC100019

○ China National Intellectual Property Appearance Patent (patent No.: ZL 201730674512.0)

○ Revision date: 2021-7-17

Note: Product functions, parameters, appearance, etc. will be adjusted as technology upgrades. Please contact our sales to confirm when purchasing.



GENERAL DESCRIPTION

TL740D is RION-TECH newly developed horizontal azimuth angular gyro sensor based on latest MEMS inertial measurement platform , by means of the dynamic attitude algorithm for the angular velocity of gyroscope ,it can simultaneously output carrier's azimuth angle .The product inernal integrated RION's Patent Inertial navigation algorithm, through the model of attitude angle data fusion , can solve the gyro short time drift problem as much as possible .

This product is specially used for robot car, AVG vehicle azimuth orientation, attitude control and other related applications of the UAV, instead of the traditional robot vehicle magnetic bar guide shortcomings, no need at the site layout of magnetic stripe, is the necessary navigation components for the next generation of robot vehicle automatic tracing and driving.

KEY FEATURES

★ Azimuth angle output
★ Strong vibration resistance
★ Light weight
★ Long life,strong stability
★ Cost-effective
★ All solid state
★ Compact & light design
★ CAN2.0A/CAN2.0B output optional
★ DC9~36V power supply

APPLICATION

★ AGV truck
★ Car Navigation
★ 3D virtual reality
★ Platform stability
★ Auto safety system
★ UAV / Robot
★ Turck-mounted satellite antenna equipment
★ Industrial control

TECHNICAL DATA

TL740D	PARAMETERS
Mesure range	Azimuth Angle (±180°)
Acquisition bandwidth	>100Hz
Resolution	0.01°
Azimuth accuracy	<0.1°/min
positional accuracy	<2mm/m （converted from angle accuracy）
Nonlinear	0.1% of FS
Max angle rate	150°/s
Accelerometer range	±4g
Acceleroemter resultuion	0.001g
Acceleroemter accuracy	5mg
Starting time	5s（Static）
Input Voltage	+9V~36V
Current	60mA(12V)
Working Temp.	-40 ~ +85℃
Storage Temp.	-40 ~ +85℃
Vibration	5g~10g
Impact	200g pk,2ms,½sine
Working life	10 years
Output rate	1Hz~100Hz can set
Output signal	CAN2.0A / CAN2.0B
MTBF	≥50000 hours /times
Insulation resistance	≥100 Megohm
Impact resistance	100g@11ms、3 Axial Direction (Half Sinusoid)
Anti-vibration	10grms、10～1000Hz
Protecting	IP67
Weight	130g(Without cable)

ELECTRICAL CONNECTION

LINE COLOR FUNCTIONS	BLACK	WHITE	GREEN	RED
	GND Power Negative	CAN_L	CAN_H	Vcc 9～36V Power Positive

ORDERING INFORMATION

TL740D-□-□-□

Output Interface

Data output

Application Types

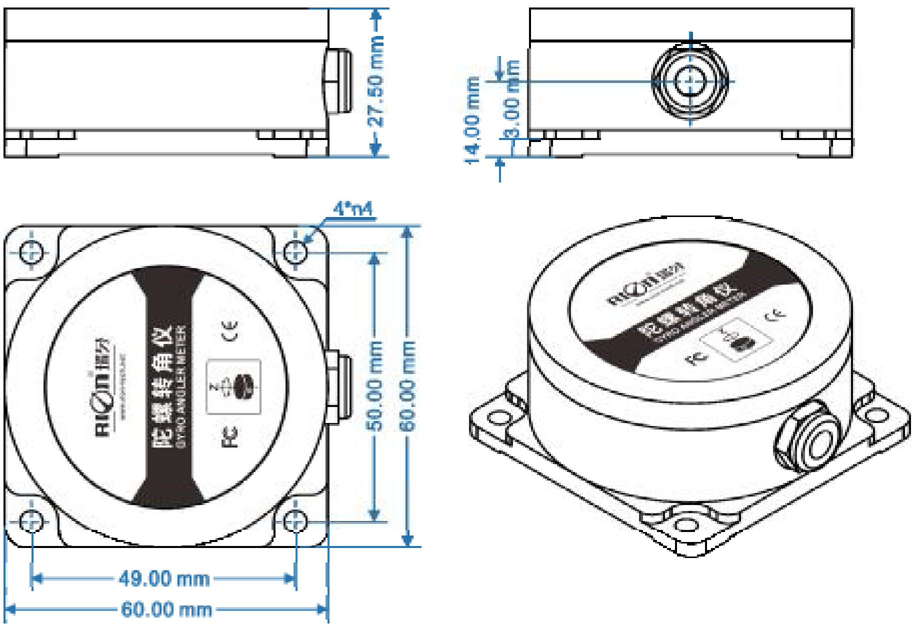
- 1 : Euler angle frame
- 3 : Acceleration frame
- 5 : Gyro Data Frame
- 7 : Standard data
- 4 : Euler angle frame + acceleration frame
- 6 : Euler angle frame + gyro data frame
- 8 : Acceleration frame + gyro data frame
- 9 : Euler angle frame + acceleration frame + gyro data frame

- AGV : AGV application
- AGM : Agricultural machinery application

- C2 : CAN 2.0A
- C3 : CAN 2.0B

E.g :TL740D-C2-1-AGV : CAN2.0A Output Interface/Euler angle frame data output/AGV application.

SIZE

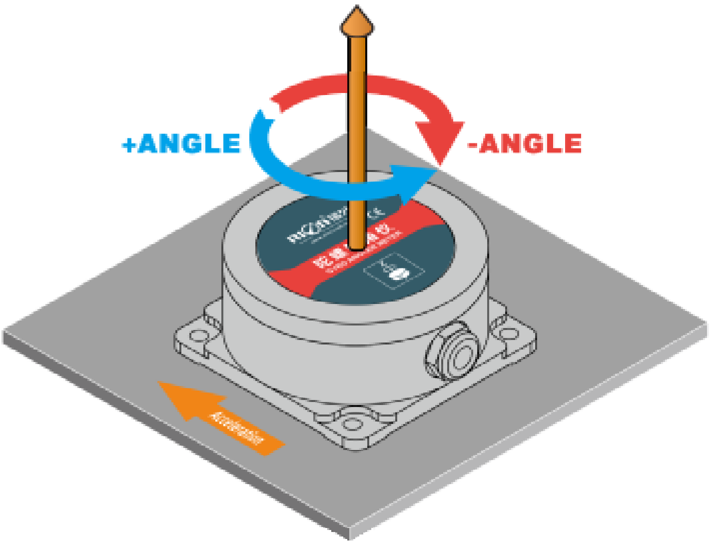


Shell size: L60×W60×H27.5mm

Installation size: L49*W50*H27.5mm

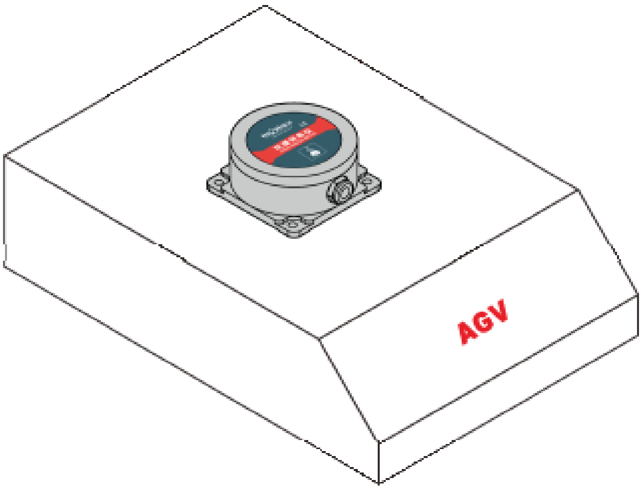
ounting screws: 4 M4 screws

□ INSTALLATION DIRECTION



□ INSTALLATION PRECAUTIONS

1.The angular gyro sensor should be mounted in the center position of the measured object , in order to reduce the influence of linear acceleration on the measurement accuracy. See below diagram as ref.

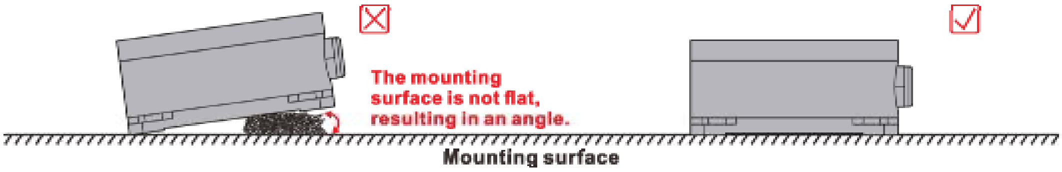


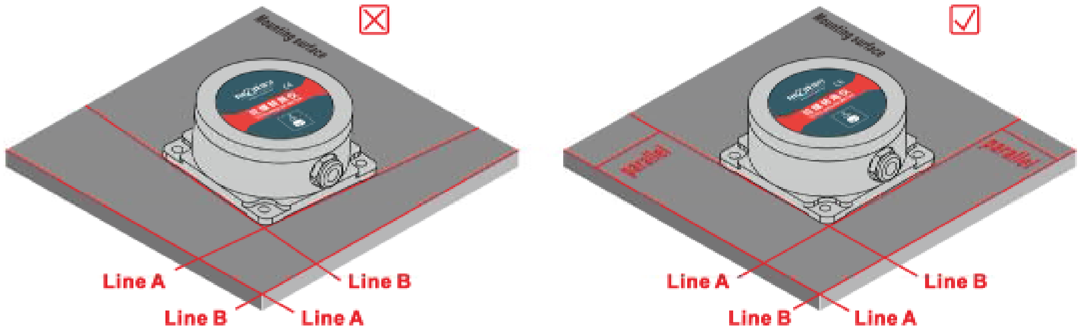
The gyro goniometer is installed in the geometric center of the AGV vehicle

2. The installation of the instrument should be kept parallel to the surface of the measured object, and reduce the influence of the dynamic and acceleration on the angle meter. Incorrect installation will lead to measurement errors, with particular attention to "surface" and” line "

①The mounting surface of the instrument fixing must be close, smooth and stable with the measured surface. If the mounting surface is not smooth, the angle error of angle measurement can be caused easily.

②The axis of the instrument must be parallel to the axis of measurement, and the two axis should not be included angle as far as possible.





3. Do not shake violently during the use of the product, avoid violent vibration, away from the vibration source (if you can not avoid please install the shock absorber), so as not to affect the product measurement accuracy;
4. Try to avoid a sharp acceleration, arrest, sharp turn angular velocity greater than 300 DEG /s movement during use, so as not to affect the measurement precision of products.

RION CAN2.0B PROTOCOL COMMUNICATION PROTOCOL

CAN2.0 protocol supports 2.0 A (11 bit ID) or 2.0 B (29 bit ID). default to CAN2.0A PROTOCOL.

Communication Protocol:

1) Modify node number (node range :0 x01-0x7F), default node number is 0x05 .

Request message format :

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x600+0x05	0x40	0x10	0x10	0x00	Node_ID	0x00	0x00	0x00

Response message format :

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x580+0x05	0x40	0x10	0x10	0x00	Node_ID	0x00	0x00	0x00

Note : if controller send CAN-ID=0x600+0x05, send data：40 10 10 00 10 00 00 00

Sensor return CAN-ID=0x580+0x05, Return data : 40 10 10 00 10 00 00 00 re-power then received frame ID 0 x590(0x580+0x10), Indicates that the frame ID modified successfully.

2) CAN baud rate set(Default baud rate:125Kbps).

Request message format :

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x600+0x05	0x40	0x20	0x10	0x00	Baud	0x00	0x00	0x00

Response message format：

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x580+0x05	0x40	0x20	0x10	0x00	Baud	0x00	0x00	0x00

Note：5th byte(Baud) for 0x00 / 0x01 / 0x02 / 0x03 / 0x04. among 0x00 means baud rate set to 1M bps, 0x01 means baud rate set to 500K bps,0x02 means baud rate set to 250K bps,0x03 means baud rate set to 125K bps,0x04 means baud rate set to 100K bps, Default baud rate 125K bps, After sending this command and receiving the returned data, The sensor needs to be re-powered for the baud rate modification to succeed.

3) Set automatic output cycle time (factory default output cycle 10 ms).

Request message format :

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x600+0x05	0x22	0x00	0x22	0x00	T_L	T_H	0x00	0x00

Response message format :

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x580+0x05	0x22	0x00	0x22	0x00	T_L	T_H	0x00	0x00

The fifth and sixth bytes represent time, the fifth byte is low byte, 6th byte High byte, Time range 10ms~1000ms;

For example：50ms TIME_L=0x32,TIME_H=0x00,

100ms TIME_L=0x64,TIME_H=0x00,

1000ms TIME_L=0xE8,TIME_H=0x03,

Range set 10ms~1000ms, factory default value 100ms(10Hz).

4) Azimuth clear zero

Request message format :

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x600+0x05	0x40	0x10	0x10	0x00	0x10	0x10	0x10	0x10

Response message format :

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x580+0x05	0x40	0x10	0x10	0x00	0x10	0x10	0x10	0x10

Clear the current heading angle to zero.

5) Set output frame type (factory default: angle frame + acceleration frame + gyroscope angle rate frame)

The output data frame includes: angle frame, acceleration frame, gyro angular rate frame and standard frame. Turn on the corresponding frame output, the sensor will output these frames according to the set period. For example, if the output period is 100mS, the angle frame+acceleration frame+gyro angular rate frame is turned on, then the continuous output angle frame+acceleration frame+gyro angular rate frame every 100mS (There will be a small time interval between the angle frame, acceleration frame, and gyro angular rate frame).

Request message format: :

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x600+0x05	0x40	0x30	0x10	0x00	MASK	0x00	0x00	0x00

Response message format :

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x580+0x05	0x40	0x30	0x10	0x00	MASK	0x00	0x00	0x00

The lower 4 bits of the fifth byte Mask are valid. Mask (binary: 0B0000dcba).

a: Represents the angle frame (roll angle, pitch angle, azimuth angle),

1: turn on the output, 0: turn off the output.

b: Represents the acceleration frame (three-axis acceleration), 1: turn on the output, 0: turn off the output.

c: Represents the gyro angular velocity frame (three-axis angular velocity),

1: turn on the output, 0: turn off the output.

d: Represents the standard frame (Z-axis angular rate, forward acceleration, azimuth),

1: turn on the output, 0: turn off the output.

The host sends: 40 30 10 00 07 00 00 00, will turn on the angle frame, acceleration frame and gyro angular velocity frame output, turn off the standard frame output.

The host sends: 40 30 10 00 05 00 00 00, the output of angle frame and gyro angular velocity frame will be turned on, and the output of acceleration frame and standard frame will be turned off.

6) Data parsing

①Data frame types are divided into four types: angle frame, acceleration frame, gyroscope frame, standard frame.

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x580+Node_ID	Data0	Data1	Data2	Data3	Data4	Data5	Data6	flag

Data0-Data6: Represents the data. According to the flag corresponding to the flag, the data is determined to be angle, acceleration, and gyro frame.

flag (ddddccaa): unsigned single byte, 8bit indicates the data frame type and installation measurement method:

aa: indicates the data type of the frame

00: Represents the angle frame (roll angle ±180°, pitch ±90°, azimuth ±180°);

01: Indicates acceleration (±32.765g);

10: Represents the gyroscope (±327.65°/S);

11: Represents the standard frame (Z-axis angular rate + Y-axis forward acceleration + Z-axis azimuth);

cc: reserved.

dddd: reserved.

The following is the analysis of different data types of horizontal measurement methods:

A： Angle data frame

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x580+Node_ID	XL	XH	YL	YH	ZL	ZH	Temp	0x00

There are eight bytes of data after CAN-ID, the first two bytes are XL, XH is the inclination of X axis (ROLL angle), the third and fourth bytes are YL, YH is Y axis (PITCH pitch angle) Inclination, the 5th and 6th bytes ZL, ZH is the inclination of the Z axis (YAW azimuth); the angle value is int16_t, the low byte is first, the high byte is after, and finally divided by 100 to get the angle floating point number. The seventh byte is the temperature value, which is a signed single-byte integer.

Example of angle conversion:

26 15 DA EA 28 23 19 00

Flag= 0x00, indicating that the data is an angle.

The angle data of the X-axis roll angle is represented by a 16-bit signed binary number, the upper 8 bits are XH, and the lower 8 bits are XL.

Use the 16-bit signed binary number to convert to a decimal number, and then divide by 100, the result is the angle.

For example, XL=0x26, XH=0x15, the angle is 54.14°

XH XL

0x15 0x26

0x1526(5414)

The final result：5414/100=54.14°

YH YL

0xEA 0xDA

0xEADA(-5414)

The final result：-5414/100=-54.14°

ZH ZL

0x23 0x28

0x2328(9000)

The final result：9000/100=90.00°

Temp

0x19(25) = 25 Celsius

B： Acceleration Data Frame

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x580+Node_ID	XL	XH	YL	YH	ZL	ZH	0x00	0x01

There are eight bytes of data after CAN-ID, the first two bytes are XL, XH is the acceleration of the X axis, the third and fourth bytes are YL, YH is the acceleration of the Y axis, and the fifth and sixth bytes are ZL, ZH are Z-axis acceleration; acceleration is int16_t, low byte first, high byte last, and finally divided by 1000 to get acceleration floating point number. The seventh byte is reserved.

Examples of acceleration conversion:

26 15 DA EA 28 23 00 01

Flag= 0x01, indicating that the data is acceleration.

The X-axis acceleration data is represented by a 16-bit signed binary number, the upper 8 bits are XH, and the lower 8 bits are XL.

Use the 16-bit signed binary number to convert to a decimal number, and then divide by 1000, the result is the acceleration.

For example, XL=0x26, XH=0x15, the acceleration is 5.414g

XH XL

0x15 0x26

0x1526(5414)

Final results：5414/1000=5.414g

YH YL

0xEA 0xDA

0xEADA(-5414)

Final results：-5414/1000=-5.414g

ZH ZL

0x23 0x28

0x2328(9000)

Final results：9000/1000=9.000g

C： Gyro Data Frame

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte

0x580+Node_ID	XL	XH	YL	YH	ZL	ZH	0x00	0x02

There are eight bytes of data after CAN-ID, the first two bytes are XL, XH is the X-axis gyro value, the third and fourth bytes are YL, YH is the Y-axis gyro value, and the fifth and sixth bytes are ZL, ZH are the Z-axis gyro value; the gyro value is int16_t, the low byte is first, the high byte is after, and finally divided by 100 to get the gyro floating point number. The seventh byte is reserved.

Examples of gyro data conversion:

26 15 DA EA 28 23 00 02

Flag= 0x02, indicating that the data is a gyro.

The X-axis gyro data value is represented by a 16-bit signed binary number, the upper 8 bits are XH, and the lower 8 bits are XL.

Use the 16-bit signed binary number to convert to a decimal number, and then divide by 100, the result is the gyro value.

For example,XL=0x26,XH=0x15, The gyro is 54.14°/S

XH XL

0x15 0x26

0x1526(5414)

Final results：5414/100=54.14°/S

YH YL

0xEA 0xDA

0xEADA(-5414)

Final results：-5414/100=-54.14°/S

ZH ZL

0x23 0x28

0x2328(9000)

Final results：9000/100=90.00°/S

D： Standard data frames

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x580+Node_ID	ZrateL	ZrateH	YaccL	YaccH	ZangL	ZangH	0x00	0x03

There are eight bytes of data after CAN-ID, the first two bytes are ZrateL, ZrateH is the Z-axis gyro angular rate value, the third and fourth two bytes are YaccL, YaccH is the Y-axis acceleration,

The fifth and sixth bytes are ZangL, ZangH is the Z-axis azimuth value; all data items are int16_t, the low byte first, high byte last, Zrate and Zang are divided by 100 to get the Z-axis gyro angular rate and Z-axis azimuth. Yacc divided by 1000 to get the Y-axis body acceleration, the seventh byte is reserved byte 0x00, and the eighth byte is 0x03.

Examples of standard frame conversion:

26 15 DA EA 28 23 00 03

Flag= 0x03, indicating that the data is a gyro angular rate frame.

ZrateL=0x26, ZrateH=0x15, the gyro value is 54.14°/S

ZrateH ZrateL

0x15 0x26

0x1526(5414)

Final results：5414/100=54.14°/S

YaccH YaccL

0xEA 0xDA

0xEADA(-5414)

Final results：-5414/1000=-5.414g

ZangH ZangL

0x23 0x28

0x2328(9000)

Final results：9000/100=90.00°

② Periodic output of data frame: 4 kinds of data frames can be output in any combination. AGV generally chooses standard data frame output. For attitude control, you can choose angle frame output, or choose angle frame, acceleration frame and gyro angular rate output.

A. Single frame of angle output (output one frame of angle frame in cycles, the default output mode), the message format is as follows:

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x580+Node_ID	XL	XH	YL	YH	ZL	ZH	Temp	0x00

B. Dual-frame angular acceleration output (double-frame, double-frame output in cycles), the message format is as follows:

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x580+Node_ID	XL	XH	YL	YH	ZL	ZH	Temp	0x00

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x580+Node_ID	XL	XH	YL	YH	ZL	ZH	0x00	0x01

C. Three-frame angular gyro acceleration output (three frames, three consecutive output frames according to the cycle), the message format is as follows：

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x580+Node_ID	XL	XH	YL	YH	ZL	ZH	Temp	0x00

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x580+Node_ID	XL	XH	YL	YH	ZL	ZH	0x00	0x01

CAN-ID	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
0x580+Node_ID	XL	XH	YL	YH	ZL	ZH	0x00	0x02