



DTS 数字输出动态高抗震倾角传感器 产品说明书

V1.0-20250226



产品介绍

DTS 是瑞惯科技自主研发的高性能数字式 MEMS 动态倾角传感器，专为运动载体姿态监测而设计，可精准测量横滚角、俯仰角及航向角等三维姿态参数，特别适用于存在振动或运动工况下的倾角检测。

该传感器采用多轴 MEMS 惯性测量单元 (IMU)，集成三轴加速度计与陀螺仪，并搭载自适应卡尔曼滤波算法，确保在动态环境下仍能实时输出高精度的姿态数据。支持 RS485、RS232 及 TTL 等多种数字接口，兼容性强，便于系统集成。

基于非接触式测量原理，产品无需依赖基准安装面即可实现实时姿态解算。其内置高精度模数转换器 (ADC) 及陀螺仪补偿单元，可自动校正非线性误差、正交耦合、温度漂移及离心加速度效应，有效抑制运动干扰带来的测量偏差，显著提升动态测量精度，适用于长期恶劣工况。

作为一款动态/静态双模式传感器，DTS 具备优异的抗电磁干扰 (EMI) 能力，可耐受高强度机械冲击与振动，广泛应用于工业自动化、工程机械、轨道交通等严苛环境下的姿态测量与控制，是工业级高可靠性姿态感知的理想解决方案。



主要特性

- ★ 量程 ($R \pm 180^\circ$, $P \pm 90^\circ$, $Y \pm 180^\circ$)
- ★ 动态精度 $\pm 0.1^\circ$ ★ 静态精度 $\pm 0.05^\circ$
- ★ RS485/232/TTL 可选



主要应用

- ★ 强冲击振动碎石设备
- ★ 工程车设备测控
- ★ 施工设备调平
- ★ 农用机械测控
- ★ 飞行器姿态控制
- ★ 船舶姿态监测



性能参数

DTS	单位	参数
测量范围 (掉电初始值为 0°)	°	横滚±180°, 俯仰±90°, 方位±180°
测量轴	轴	X 轴 / Y 轴 / Z 轴
横滚俯仰分辨率 ¹⁾	°	0.01
横滚俯仰静态精度@25°C	°	±0.05
横滚俯仰动态精度(rms)@25°C	°	±0.1
陀螺仪	陀螺仪量程	°/s
	零偏稳定性(10s 均值)	°/h
	零偏不稳定性(allan)	°/h
	角度随机游走系数(allan)	°/sqrt(h)
加速度	加速度量程	g
	零偏稳定性(10s 均值)	mg
	零偏不稳定性(allan)	mg
	速度随机游走系数(allan)	m/s/sqrt(h)
零点温度系数 ³⁾ @ -40 ~ 85°C	°/°C	±0.01
灵敏度温度系数 ⁴⁾ @ -40 ~ 85°C	ppm/°C	≤100
上电启动时间	S	≤3.5
响应时间	S	0.01
通讯协议	-	RS485/RS232/TTL 可选
电磁兼容性	-	依照 EN61000 和 GBT17626
平均无故障工作时间 MTBF	小时/次	≥98000
绝缘电阻	兆欧	≥100
抗冲击	-	100g@11ms、三轴向(半正弦波)
抗振动	-	10grms、10 ~ 1000Hz
防护等级	-	IP67 / IP68 (可选)
重量	g	单连接头≤165g(不含电缆线) / 双连接头≤180g(不含电缆线)

1) 分辨率: 指传感器在有效量程内可识别的最小角度变化量, 反映其对微小倾角波动的监测能力。

2) 精度: 在标准环境温度 (25±5°C) 下, 对固定角度进行多次重复测量 (样本数≥16), 计算测量值与真实值的误差均方根 (RMS), 表征传感器的综合测量偏差。

3) 零点温度系数: 传感器在零输入状态下, 其输出值随温度变化的比率, 定义为额定工作温度范围内零点偏移量与常温基准值的比值。

4) 灵敏度温度系数: 传感器满量程输出值随温度变化的稳定性指标, 表征额定温度范围内灵敏度相对于常温参考值的漂移率。

电气参数

参数	条件	最小值	典型值	最大值	单位
供电电压	标准	9	12 24	36	V
工作电流	无负载		40mA(12V)		mA
工作温度		-40		+85	℃
存储温度		-40		+85	℃

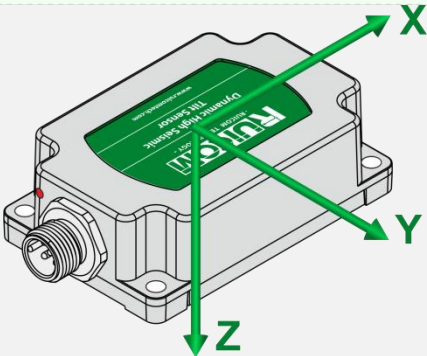
订购信息

型号	封装	信号输出	通信协议	安装方向	应用	防护等级
DTS	3:PCBA 板	232:RS232	BIN:二进制协议	H:水平安装	OB:船载应用	7:IP67
	6:标准外壳	485:RS485	MB:MODBUS 协议	HD:水平向下	EV:工程挖机	8:IP68
	9:防腐蚀外壳	TTL:UART TTL		V:垂直安装	EGP:工程碎石打桩	
				VL:垂直向左	V:车载应用	
				VR:垂直向右	MTI:矿车坡度	

例：DTS6-232BIN-HOB-7：标准外壳/RS232 信号输出/二进制通讯协议/水平安装/船载应用/IP67 防护。

安装轴向

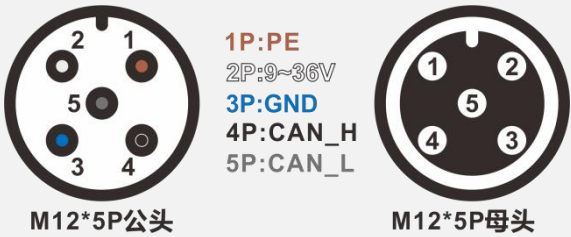
DTS 系列产品采用 ENU（东-北-天）坐标系，遵循右手定则。按照 Z / X / Y 的旋转顺序定义姿态角，其中 Y 轴正方向指向载体前进方向时：绕 Z 轴旋转的角度称为航向角，绕 X 轴旋转的角度称为俯仰角，绕 Y 轴旋转的角度称为横滚角。



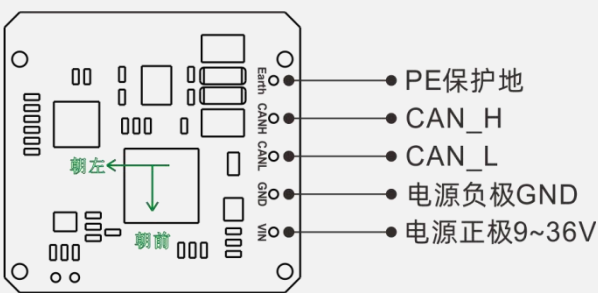
接口定义

针脚/线色	定义	描述
PIN1/棕色	9~36V	电源正极
PIN2/白色	RS232_RXD/TTL_RXD/485_D+	信号输出
PIN3/蓝色	RS232_TXD/TTL_TXD/485_D-	信号输出
PIN4/黑色	GND	电源负极
PIN5/灰色	PE	保护地

连接器针脚定义图



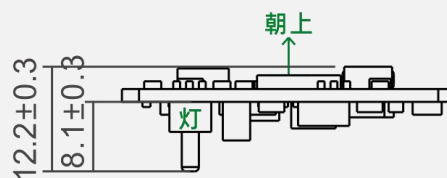
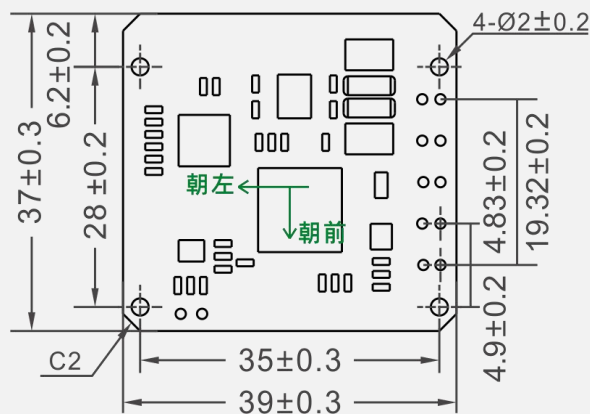
PCBA 板接线定义图





产品尺寸

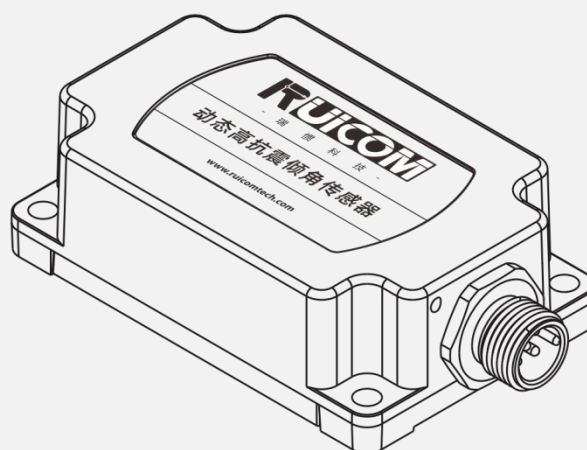
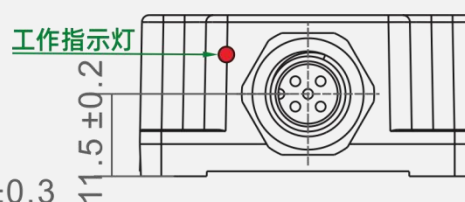
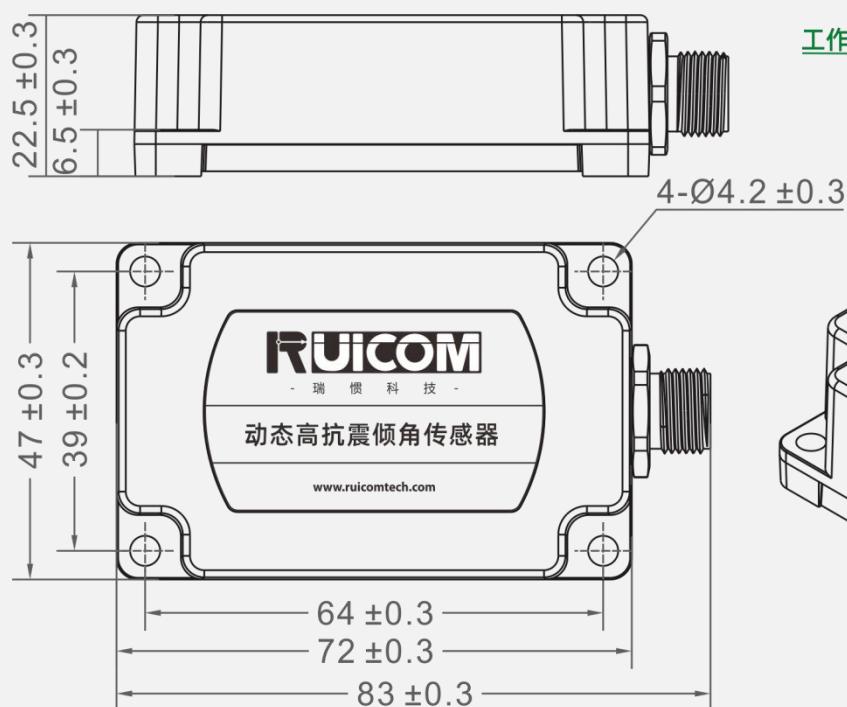
PCBA 尺寸图



单板尺寸: L39×W37×H12.2mm

安装尺寸: L35×W28mm

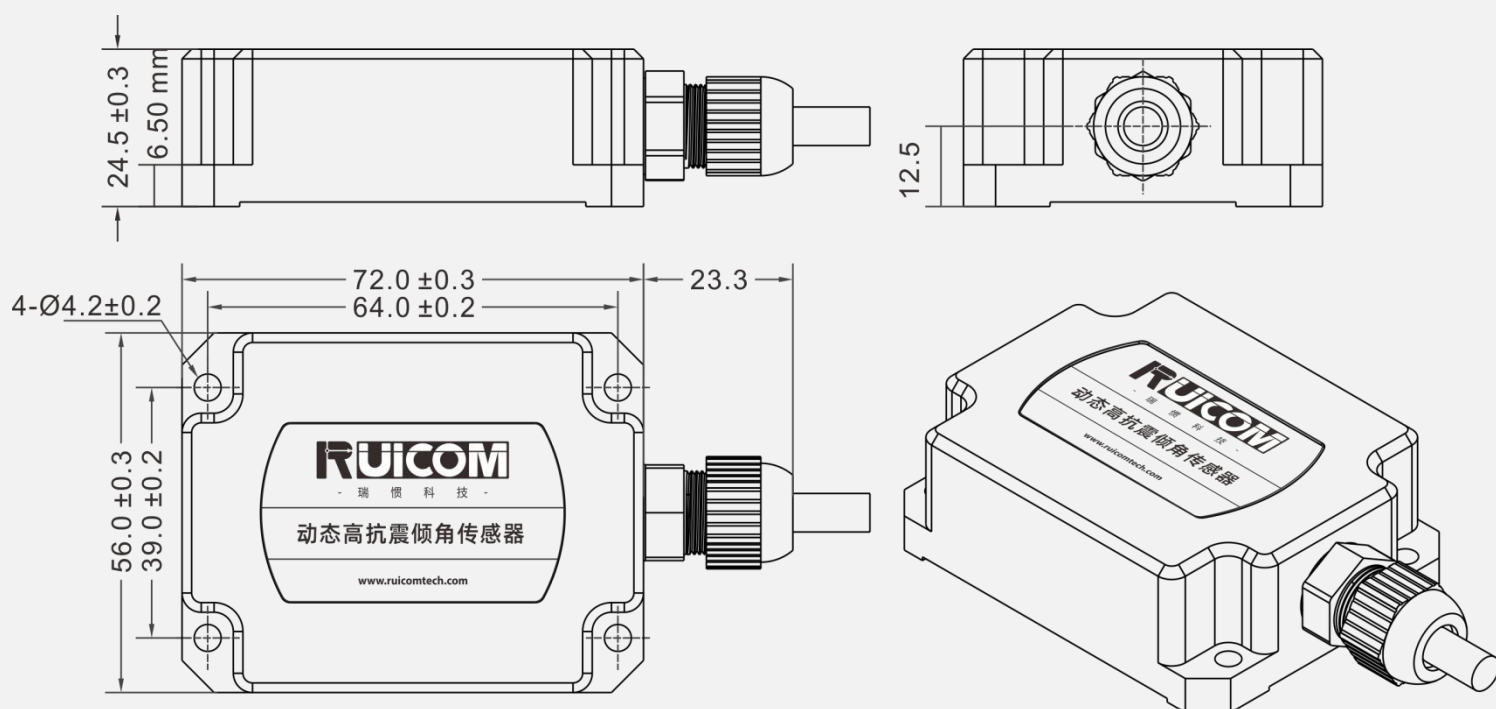
外壳尺寸图



外壳尺寸: L83×W47×H22.5mm

建议螺丝: 4 颗 M5 螺丝

耐腐蚀外壳尺寸图



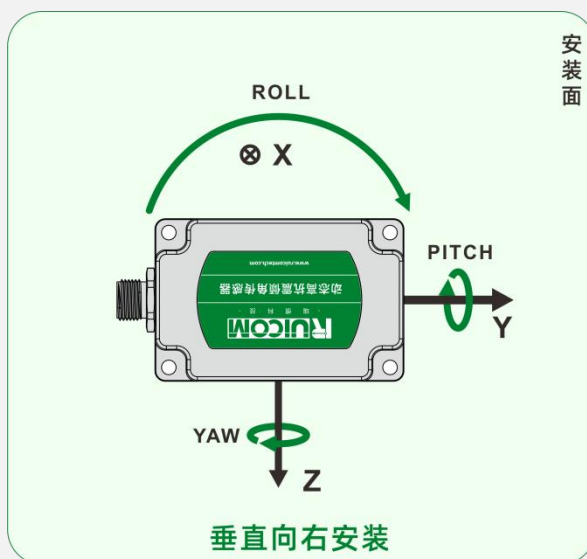
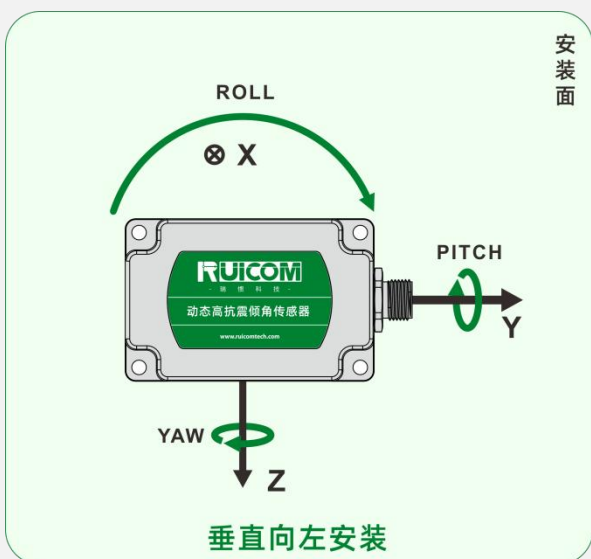
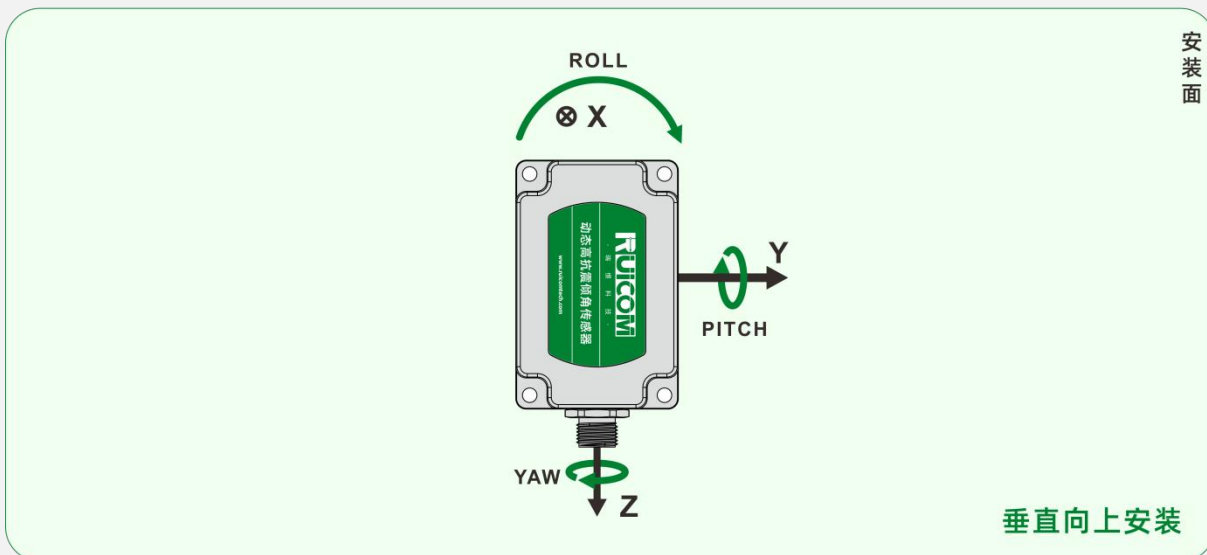
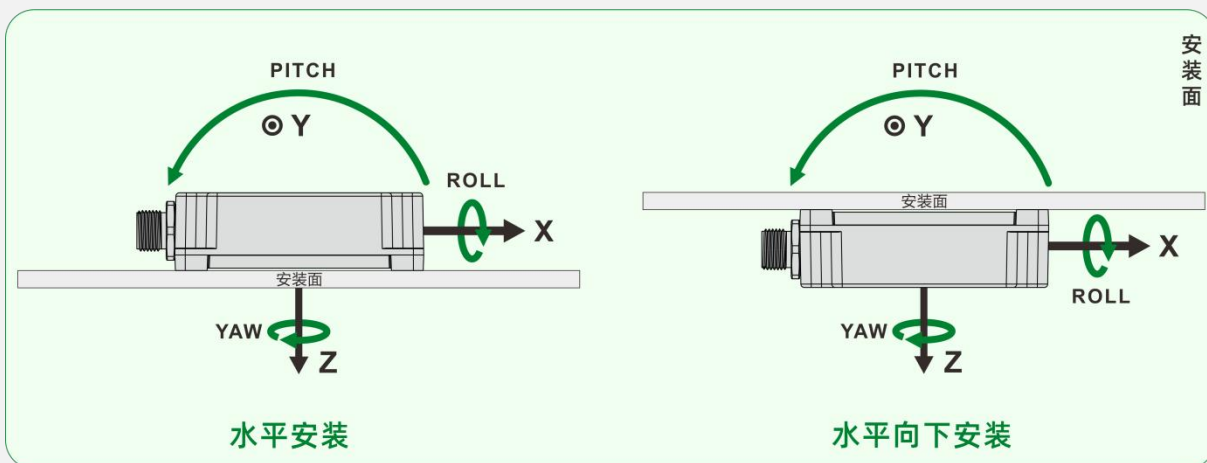
外壳尺寸: L95.3×W56×H24.5mm

建议螺丝: 4 颗 M5 螺丝



工作指示灯说明

1. 产品上电后工作指示灯闪亮;
2. 产品有数据发送时工作指示灯点亮。





二进制通信帧格式

1.0 通讯帧格式

域	帧同步 字节	帧开始 字节	ADDR 地址	CMD 命令域	LEN 长度	Data 数据域	CRC	帧结束字节
字节数	1	1	1	1	2	0 to 504	2	1
说明	Sync. byte	Start of byte	Address	Command	Length of the Data	Data	CRC	End of Tx byte
值	0xFF	0x02	0-FF	-	-	-	-	0x03

Notes:

A. LEN 数据长度域包括数据域(Data)的字节数, MSB 在前, LSB 在后, 长度为 0 表示没有数据域, 最长数据域长度为 504 字节, 最长帧字节为 512 个字节。

B. CRC 的计算从 ADDR 开始, 包括 CMD, 长度域及数据域, MSB 在前, LSB 在后, 按以下函数计算 CRC:

```
uint16 calcCRC(const uint8 *pBuffer, uint16 bufferSize)
{
    uint16 poly = 0x8408;
    uint16 crc = 0;
    uint8 carry;
    uint8 i_bits;
    uint16 j;

    for (j=0; j<bufferSize;j++)
    {
        crc = crc^pBuffer[j];
        for (i_bits=0; i_bits<8;i_bits++)
        {
            carry = crc & 1;
            crc = crc / 2;
            if (carry)
            {
                crc = crc^poly;
            }
        }
    }
    return crc;
}
```

1.1 数据格式及端模式

设备的数据输出为小端模式(如整型, 浮点型, 都是低字节在前, 高字节在后)。

1.2 串口默认

串口默认波特率为 115200 bps, 1 位起始位, 1 stop 位, 无校验。

1.3 默认输出是自动输出, 输出频率是 10Hz.

1.4 默认地址为 0, 范围为 0x00-0xFE 可设, 0xFF 为万能地址。

2.0 通信模式

传感器有两种通信模式：查询模式（问答式）和连续输出模式（按一定的频率自动输出数据）。在连续输出模式下也可进行问答式查询操作。

2.1 查获询模式

采用一问一答模式，用户向传感器发送询问或设置指令，设备对询问或设置指令进行一对一的回复。设备对询问的指令回复有两种格式：应答（CMD=ASK）和回复（CMD = RET），具体可参考第 4 点的命令清单。

2.1.1 应答

应答帧格式如下：

域	帧同步字节	帧开始字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束字节
字节数 (byte)	1	1	1	1	1	1	1	1	1	1
值	0xFF	0x02	0x00	IMU_ACK (0x01)	0x00	0x01	ERROR CODE	0xXX	0xXX	0x03

应答作为命令的回复，由错误代码反映指令的执行情况，有以下应答的错误代码（ERROR CODE）决定设备对询问指令的响应，以下为错误代码的列表：

IMU_Error Code	Value	Description
IMU_NO_ERROR	0x00	命令已正确执行
IMU_ERROR	0x01	命令没有正确执行
IMU_INVALID_FRAME	0x04	无效的命令
IMU_INVALID_PARAMETER	0x09	无效的参数
IMU_NOT_READY	0x0A	传感器未就绪

备注：错误代码组成了 ACK 帧的数据域(DATA)。

2.1.2 回复

回复是对用户指令产生的一对一的响应，不同的询问指令回复的 CMD 及数据域不同，请参考第 4 点的命令清单。

2.2 连续输出模式

连续输出模式是自动输出模式，按一定的频率输出选定的数据项，输出频率可设。

3.0 输出数据类型

3.1 输出数据类型

通过使用 output masks(uint32)设置输出的数据项及格式，用户可选择以下数据项输出：

有效 mask	描述	顺序	支持	Mask 值
IMU_OUTPUT_EULER	姿态角	1	支持	0x00000001
IMU_OUTPUT_GYROSCOPES	标定后三轴角速率	2	支持	0x00000002
IMU_OUTPUT_ACCELEROMETERS	标定后三轴加速度	3	支持	0x00000004

顺序表示数据输出的顺序。

3.2 数据输出格式

3.2.1 姿态角 Roll, Pitch 和 Yaw, 用 real32(float)表示, 12 个字节, 单位是弧度。

存储格式:

Roll	Pitch	Yaw
------	-------	-----

3.2.2 三轴陀螺角速率 Gx, Gy 和 Gz, 用 real32(float)表示, 12 个字节, 单位是 $\text{rad}\cdot\text{S}^{-1}$ (弧度/秒)。

存储格式:

Gx	Gy	Gz
----	----	----

3.2.3 三轴加速度 Ax, Ay 和 Az, 用 real32(float)表示, 12 个字节, 单位是 $\text{m}\cdot\text{S}^{-2}$ 。

存储格式:

Ax	Ay	Az
----	----	----

3.3 输出数据顺序

根据输出 output_masks(uint32) 的不同值, 会产生相应的输出顺序, 默认的 output_masks(uint32) = 0x00000007 如下:

IMU_OUTPUT_EULER|IMU_OUTPUT_GYROSCOPES|IMU_OUTPUT_ACCELEROMETERS
(0x00000001|0x00000002|0x00000004) 则对应的输出的数据顺序及存储格式如下:

Name	Roll	Pitch	Yaw	Gx	Gy	Gz	Ax	Ay	Az
Size	4	4	4	4	4	4	4	4	4
(bytes)	12			12			12		

数据域的解析按以上存储格式进行。

如 output_masks(uint32) = 0x00000001 如下:

IMU_OUTPUT_EULERS (0x00000001) 则对应的输出的数据顺序及存储格式如下:

Name	Roll	Pitch	Yaw
Size	4	4	4
(bytes)	12		

如 output_masks(uint32) = 0x00000005 如下:

IMU_OUTPUT_EULER|IMU_OUTPUT_ACCELEROMETERS (0x00000001|0x00000004) 则对应的输出的数据顺序及存储格式如下:

Name	Roll	Pitch	Yaw	Ax	Ay	Az
Size	4	4	4	4	4	4
(bytes)	12			12		

4.0 命令清单**4.1 设置自动输出数据项***** IMU_SET_DEFAULT_OUTPUT_MASK----- (0x50)**

作用: 用于设置输出 MASK(output_mask(uint32)), 即输出数据类型(Buffer)。

注意: 不具备掉电保存功能, 须执行保存设置命令才有掉电保存功能。

帧格式如下:

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	5	1	1	1
0xFF	0x02	0x00	0x50	0x00	0x05	DATA	0xFF	0xFF	0x03

其中: DATA 部分

DATA	
Reserved.Leave to 0(bytes)	Default output mask(uint32)
1	4

回复:

IMU_ACK

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1		1	1	1	1	1	1	1
0xFF	0x02	0x00	0x01	0x00	0x01	ERROR_CODE	0xFF	0xFF	0x03

4.2 获取自动输出的数据项

* IMU_GET_DEFAULT_OUTPUT_MASK------(0x51)

作用: 用于读取输出 MASK(output mask(uint32))数据项, 用于数据的解析。

帧格式如下:

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	0	1	1	1
0xFF	0x02	0x00	0x51	0x00	0x00	NULL	0xFF	0xFF	0x03

回复: * IMU_RET_DEFAULT_OUTPUT_MASK------(0x52)

作用: 返回 output mask;

帧格式如下:

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	4	1	1	1
0xFF	0x02	0x00	0x52	0x00	0x04	DATA	0xFF	0xFF	0x03

其中: DATA 部分

DATA	
Default output mask(uint32)	
4	

4.3 连续数据输出

* IMU_CONTINUOUS_DEFAULT_OUTPUT------(0x90)

作用：数据项连续输出。

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	See below	1	1	1
0xFF	0x02	0x00	0x90	0x00	0x24	See below	0xFF	0xFF	0x03

Data 部分如下 (total length: 0x0024):

Name	Roll	Pitch	Yaw	Gx	Gy	Gz	Ax	Ay	Az
Size	4	4	4	4	4	4	4	4	4
(bytes)	12			12			12		

数域名	real32	float	unit
Roll	BBFF0F9E	-0.007784	弧度
Pitch	3C35E11F	0.011101	弧度
Yaw	BD2E03F5	-0.042484	弧度
Gx	B99D4848	-0.000300	弧度/秒
Gy	3925ACF1	0.000158	弧度/秒
Gz	389202DA	0.000070	弧度/秒
Ax	3DE1E839	0.110306	m.S ⁻²
Ay	3D9D7DFA	0.076900	m.S ⁻²
Az	C11C1675	-9.755483	m.S ⁻²

4.4 输出模式设置

输出工作方式。

Mode 有两种方式，如下两种：

IMU_QEURY_MODE(查询模式或一问一答模式)：0x00

IMU_CONTINUOUS_MODE (连续输出模式) :0x01

其中 CONTINUOUS_MODE (连续输出模式) 有不同的 Divider 值，对应以下不同的输出频率：

Divider	OUT(Hz)
0	INVALID
1	200
2	100
4	50
8	25
10	20
20	10
40	5
200	1

注：默认为 IMU_CONTINUOUS_MODE(连续输出模式)，频率 10Hz。

4.5 设置输出模式命令

* IMU_SET_CONTINUOUS_MODE----- (0x53)

作用：设置输出模式；

注意：不具备掉电保存功能，须执行保存设置命令才有掉电保存功能。

帧格式如下：

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	3	1	1	1
0xFF	0x02	0x00	0x53	0x00	0x03	如下	0xFF	0xFF	0x03

其中：DATA 部分

DATA		
Reserved.Leave to 0(bytes)	Mode	Divider
1	1	1
0	1	1

注意：以上 Mode = 1，为连续输出，Divider = 1，输出频率为：200Hz。

回复：

IMU_ACK

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	1	1	1	1
0xFF	0x02	0x00	0x01	0x00	0x01	ERROR_CODE	0xFF	0xFF	0x03

4.6 获取输出模式

* IMU_GET_CONTINUOUS_MODE----- (0x54)

作用：用于读取当前模式；

帧格式如下：

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	0	1	1	1
0xFF	0x02	0x00	0x54	0x00	0x00	NULL	0xFF	0xFF	0x03

回复：

* IMU_RET_CONTINUOUS_MODE----- (0x55)

作用：用于返回 Mode 和 Divider；

帧格式如下：

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	2	1	1	1
0xFF	0x02	0x00	0x55	0x00	0x02	DATA	0xFF	0xFF	0x03

其中: DATA 部分

DATA	
Mode(uint8)	Divider(uint8)
1	1

4.7 设置波特率

波特率 (baud rate) 是一个 32 位的整数, 有效的值有以下: 9600, 19200, 38400, 57600, 115200, 256000。

* IMU_SET_PROTOCOL_MODE----- (0x12)

作用: 用于设置通信波特率; (默认为 115200)

注意: 不具备掉电保存功能, 须执行保存设置命令才有掉电保存功能。

帧格式如下:

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	5	1	1	1
0xFF	0x02	0x00	0x12	0x00	0x05	如下	0xFF	0xFF	0x03

其中: DATA 部分

DATA	
Reserved.Leave to 0(bytes)	Baudrate (uint32)
1	4
0	115200

注意: Baud rate(uint32)为小端模式, 低字节在前, 高字节在后。

回复:

IMU_ACK

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	1	1	1	1
0xFF	0x02	0x00	0x01	0x00	0x01	ERROR _CODE	0xFF	0xFF	0x03

4.8 获取波特率

* IMU_GET_PROTOCOL_MODE----- (0x13)

作用: 用于获取通信波特率;

帧格式如下:

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	0	1	1	1
0xFF	0x02	0x00	0x13	0x00	0x00	NULL	0xFF	0xFF	0x03

回复:

*IMU_RET_PROTOCOL_MODE----- (0x14)

作用: 返回当前波特率;

帧格式如下:

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	4	1	1	1
0xFF	0x02	0x00	0x14	0x00	0x04	DATA	0xFF	0xFF	0x03

其中：DATA 部分

DATA
Baud rate
4

4.9 设置传感器地址

传感器地址是通信时的目标地址，只有地址域与传感器的地址相匹配时才能通信成功。

* IMU_SET_ADDR----- (0x0F)

作用：用于设置传感器的地址；（默认为 0x00）

注意：不具备掉电保存功能，须执行保存设置命令才有掉电保存功能。

帧格式如下：

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	1	1	1	1
0xFF	0x02	0x00	0x0F	0x00	0x01	address	0xFF	0xFF	0x03

注意：ADDR 要当前的要更改的传感器地址，DATA 部分为要设置的传感器新地址。

回复：

IMU_ACK

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	1	1	1	1
0xFF	0x02	0x00	0x01	0x00	0x01	ERROR _CODE	0xFF	0xFF	0x03

注意：回复的 ADDR 为之前的传感器地址。

4.10 获取传感器地址

* IMU_GET_ADDR----- (0x0D)

作用：用于获取传感器地址；

帧格式如下：

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	0	1	1	1
0xFF	0x02	0x00	0x0D	0x00	0x00	NULL	0xFF	0xFF	0x03

回复：

*IMU_RET_ADDR----- (0x0E)

作用：返回传感器地址；

帧格式如下:

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	1	1	1	1
0xFF	0x02	0x00	0x0E	0x00	0x01	ADDR	0xFF	0xFF	0x03

其中: DATA 部分

DATA
ADDR
1

4.11 设置测量安装方式 (水平, 水平向下, 垂直向上, 垂直向左, 垂直向右五种安装方式。)

*** IMU_SET_FIXED_MODE----- (0x20)**

作用: 用于设置传感器的安装方式; (默认为 0x00, 水平方向)

注意: 不具备掉电保存功能, 须执行保存设置命令才有掉电保存功能。

帧格式如下:

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	1	1	1	1
0xFF	0x02	0x00	0x20	0x00	0x01	Fixed_ Mode	0xFF	0xFF	0x03

注意: Fixed_Mode: 0x00:水平方式, 0x01:垂直向上, 0x02:垂直向左, 0x03:垂直向右, 0x04:水平向下。

回复: IMU_ACK

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	1	1	1	1
0xFF	0x02	0x00	0x01	0x00	0x01	ERROR _CODE	0xFF	0xFF	0x03

4.12 获取测量安装方式

*** IMU_GET_FIXED_MODE----- (0x21)**

作用: 用于获取传感器的安装方式;

帧格式如下:

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	0	1	1	1
0xFF	0x02	0x00	0x21	0x00	0x00	NULL	0xFF	0xFF	0x03

回复:

***IMU_RET_FIXED_MODE----- (0x22)**

作用: 返回传感器的安装方式;

帧格式如下:

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	1	1	1	1
0xFF	0x02	0x00	0x22	0x00	0x01	Fixed_ Mode	0xFF	0xFF	0x03

其中: DATA 部分

DATA
Fixed_Mode
1

4.13 设置绝对/相对零点

传感器的零点分为: 绝对零点和相对零点;

设为绝对零点传感器输出倾角的参考面是地理水平面; 设为相对零点时传感器输出倾角的参考面是相对于安装面, 当传感器的安装测量面与水平面有误差时, 可设置相对零点将当前的角度 (横滚角与俯仰角) 清零, 此时传感器输出的角度变化是相对于安装面。

* IMU_SET_OFFSET_MODE-----(0x25)

作用: 用于设置传感器的零点; (默认为 0x00, 绝对零点)

注意: 不具备掉电保存功能, 须执行保存设置命令才有掉电保存功能。

帧格式如下:

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	1	1	1	1
0xFF	0x02	0x00	0x25	0x00	0x01	Offset_ Mode	0xFF	0xFF	0x03

注意: Offset_Mode: 0x00: 绝对零点; 01: 相对零点。

回复: IMU_ACK

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	1	1	1	1
0xFF	0x02	0x00	0x01	0x00	0x01	ERROR_ _CODE	0xFF	0xFF	0x03

4.14 获取零点类型

* IMU_GET_OFFSET_MODE-----(0x26)

作用: 用于获取当前传感器的零点类型;

帧格式如下:

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	0	1	1	1
0xFF	0x02	0x00	0x26	0x00	0x00	NULL	0xFF	0xFF	0x03

回复:

***IMU_RET_OFFSET_MODE-----(0x27)**

作用：返回传感器的安装方式；

帧格式如下：

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	1	1	1	1
0xFF	0x02	0x00	0x27	0x00	0x01	OFFSET_ Mode	0xFF	0xFF	0x03

其中：DATA 部分

DATA
OFFSET_Mode
1

Offset_Mode: 0x00: 绝对零点； 01: 相对零点。

4.15 设置重力量级

*** IMU_SET_GRAVITY_MAGNITUDE----- (0xB1)**

作用：设置当前重力量级大小；

注意：不具备掉电保存功能，须执行保存设置命令才有掉电保存功能。

帧格式如下：

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	5	1	1	1
0xFF	0x02	0x00	0xB1	0x00	0x05	DATA	0xFF	0xFF	0x03

其中：DATA 部分

DATA
Reserved.Leave to 0(bytes) magnitude (real32)
1 4

Magnitude: 为重力量级大小。浮点类型，小端模式。

回复：

IMU_ACK

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	1	1	1	1
0xFF	0x02	0x00	0x01	0x00	0x01	ERROR _CODE	0xFF	0xFF	0x03

4.16 获取重力量级

*** IMU_GET_GRAVITY_MAGNITUDE----- (0xB2)**

作用：获取当前重力量级大小；

帧格式如下:

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	0	1	1	1
0xFF	0x02	0x00	0xB2	0x00	0x00	NULL	0xFF	0xFF	0x03

回复:

* IMU_RET_GRAVITY_MAGNITUDE----- (0xB3)

作用: 返回当前重力量级大小;

帧格式如下:

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	4	1	1	1
0xFF	0x02	0x00	0xB3	0x00	0x04	DATA	0xFF	0xFF	0x03

其中: DATA 部分

DATA
magnitude (real32)
4

4.17 设置保存命令

设置的参数不具有掉电保存功能, 更改设置参数后须执行设置保存命令, 以保存更新后的参数。

* IMU_SAVE_SETTINGS----- (0x24)

作用: 用于保存所有的设置参数到 EEPROM, 具有掉电保存功能。没有数据域, 帧格式如下:

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	0	1	1	1
0xFF	0x02	0x00	0x24	0x00	0x00	NULL	0xFF	0xFF	0x03

回复:

IMU_ACK

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	1	1	1	1
0xFF	0x02	0x00	0x01	0x00	0x01	ERROR_CODE	0xFF	0xFF	0x03

备注: ERROR CODE 参考 6 内容;

4.18 默认数据查询命令 (当输出模式 Mode = 0, 一问一答模式时)

* IMU_GET_DEFAULT_OUTPUT----- (0x56)

作用: 获取当前默认的输出数据项。

帧格式如下:

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	0	1	1	1
0xFF	0x02	0x00	0x56	0x00	0x00	NULL	0xFF	0xFF	0x03

回复:

***IMU_RET_DEFAULT_OUTPUT----- (0x57)**

作用: 返回默认的数据项输出。

当连续自动输出的 outputmask = 0x00000007j 时, Data 部分如下:

帧格式如下:

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	24	1	1	1
0xFF	0x02	0x00	0x57	0x00	0x24	See below	0xFF	0xFF	0x03

Data 部分如下 (total length: 0x0024):

Name	Roll	Pitch	Yaw	Gx	Gy	Gz	Ax	Ay	Az
Size	4	4	4	4	4	4	4	4	4
(bytes)	12			12			12		

4.19 获取特殊指定的数据项输出

*** IMU_GET_SPECIFIC_OUTPUT----- (0x58)**

作用: 获取特殊指定的 MASK 数据项。

帧格式如下:

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	4	1	1	1
0xFF	0x02	0x00	0x58	0x00	0x04	DATA	0xFF	0xFF	0x03

其中: DATA 部分

DATA
Output masks(uint32)
4

特定指定的数据项 Output masks 为以下任一指定的数据项:

IMU_OUTPUT_EULER|IMU_OUTPUT_GYROSCOPES|IMU_OUTPUT_ACCELEROMETERS 中的任何一个或多个组合。

回复:

*** IMU_RET_SPECIFIC_OUTPUT----- (0x59)**

作用: 返回特殊指定的数据项输出。

帧格式如下:

帧同步 字节	帧开始 字节	ADDR	CMD	LEN (MSB)	LEN (LSB)	DATA	CRC (MSB)	CRC (LSB)	帧结束 字节
1	1	1	1	1	1	0x24	1	1	1
0xFF	0x02	0x00	0x59	0x00	0x24	DATA	0xFF	0xFF	0x03

DATA 部分如下 (total length: 0x0024):

Name	Roll	Pitch	Yaw	Gx	Gy	Gz	Ax	Ay	Az
Size	4	4	4	4	4	4	4	4	4
(bytes)	12			12			12		

Output masks(uint32) = 0x00000007

IMU_OUTPUT_EULER|IMU_OUTPUT_GYROSCOPES|IMU_OUTPUT_ACCELEROMETERS;

MODBUS 通讯协议

一. 数据帧格式:

RTU 模式;

通讯参数: 波特率 9600 bps (出厂默认) ;

数据帧: 1 个起始位, 8 位数据, 偶校验, 1 个停止位。

注意, 使用前请仔细阅读以下项目:

1 > 根据 MODBUS 通信协议规范, 相邻数据帧之间的时间间隔应不小于 3.5 个字符传输时间(以 9600bps 波特率为例, 该间隔时间为 $3.5 \times (11 \text{ 位/字符}) \div 9600 \approx 4\text{ms}$)。为确保通信可靠性, 本传感器采用更为严格的时间间隔标准, 将帧间最小间隔提升至 5ms。建议用户在实际应用中确保相邻数据帧发送间隔不低于 5ms, 以保证通信的稳定性和数据完整性。

主机发送命令 - - 5ms 空闲 - - 从机回覆命令 - - 5ms 空闲 - - 主机发送命令.....

2 > MODBUS 协议中规定了广播地址----0 的相关内容, 本传感器同样也能接受广播地址内容, 但一律不会进行回覆(除读取地址码功能)。所以广播地址 0 就可以作为以下用途, 仅供参考。

1.将总线上挂载的所有本型号的倾角传感器的地址全部设为某一个地址。

2.将总线上挂载的所有本型号的倾角传感器全部设为相对/绝对零点。

3.对整条总线上的该型号传感器进行测试, 即主机向总线发送 0 地址询问角度命令, 通信指示灯能闪烁的即通信正常。

3 > 为了提高系统的可靠性, 设置地址命令和设置绝对/相对命令, 设置波特率, 设置输出频率及设置奇偶校验位这三种命令都必须连续两次发送才会有效。“连续两次发送”是指两次都发送成功(从机每次都有回覆), 且两次问答必须前后连续, 即主机不能在两次问答的中间插入其他数据帧, 否则, 该种命令就会被锁定, 设置过程参考如下: 发送设置地址命令 - - 等待从机发送的设置成功命令 - - (不能出现其它命令) 再次发送设置地址命令 - - 等待从机发送的设置成功命令 - - 修改成功

4 > 当正常通信累计到一定次数后, 通信指示灯才会闪烁一次。

二. 读取角度数据: Modbus 功能码 03H

主机查询命令：		从机响应：		
传感器地址	01H	传感器地址	01H	
功能码	03H	功能码	03H	
寄存器首地址	00H	数据长度 14 个字节	0EH	
数据长度 4 个字	02H	数据字 1 低 8 位	C4H	横滚角
	00H	数据字 1 高 8 位	96H	
	07H	数据字 2 低 8 位	03H	
CRC	A5C8H	数据字 2 高 8 位	00H	俯仰角
		数据字 3 低 8 位	F8H	
		数据字 3 高 8 位	9BH	
		数据字 4 低 8 位	01H	方位角
		数据字 4 高 8 位	00H	
		数据字 5 低 8 位	6BH	
		数据字 5 高 8 位	54H	温度
		数据字 6 低 8 位	02H	
		数据字 6 高 8 位	00H	
		数据字 7 高 8 位	03	温度
		数据字 7 低 8 位	06	
		CRC	C40CH	

读取测量数据命令应用举例:									
主机发送		01H	03H	00H	02H	00H	07H	A5H	C8H
从机回复									
01H	03H	0EH	C4H	96H	03H	00H	F8H	9BH	01H
00H	6BH	54H	02H	00H	03H	06H	C4H	0CH	

注: 从机回复帧的数据域为 C4H, 96H, 03H, 00H, F8H, 9BH, 01H, 00H, 6BH, 54H, 02H, 00H, 03H, 06H

横滚角 ($\pm 180^\circ$) 为数据域的第 1-4 字节, 俯仰角 ($\pm 90^\circ$) 为数据域的第 5-8 字节, 方位角 ($\pm 180^\circ$) 为 9-12 字节, 低字节在前, 高字节在后, 第 13-14 字节为温度。角度的表示方法为点数表示法, 一个点对应 0.001° , $0.001 \times (\text{点数} - \text{偏移})$ 即为角度。如测量范围为 $\pm 90^\circ$, 偏移量为 90000, 则总共点数为 180000 点。所以 0 对应 -90° , 180000 对应 $+90^\circ$, 90000 对应 0° 。如测量范围为 $\pm 180^\circ$, 偏移量为 180000, 则总共点数为 360000 点。所以 0 对应 -180° , 180000 对应 0° , 360000 对应 180° 。

以上述数据帧为例: 角度的转换过程如下:

- 1) 得到当前角度点数, 注意低字节在前, 横滚角为 000396C4H, 俯仰角为 00019BF8H, 方位角 0002546BH。
- 2) 转换为十进制, 横滚角: 396C4H $\rightarrow$ 235204, 俯仰角: 19BF8H $\rightarrow$ 105464, 方位角: 2546BH $\rightarrow$ 152683。
- 3) 横滚角减去偏移量 180000 (注意: 该数值是与测量范围相关的一个量, 测量范围值乘 1000,

如果测量范围 $\pm 180^\circ$ 的偏移量为 180×1000 ，测量范围为 $\pm 90^\circ$ 的偏移量为 90×1000 ），横滚角：
 $235204 - 180000 = 55204$ ，俯仰角： $105464 - 90000 = 15464$ ，方位角：
 $152683 - 180000 = -27317$ 。

4) 得到最终角度，横滚角： $55204 \times 0.001 = 55.204^\circ$ ，俯仰角： $15464 \times 0.001 = 15.464^\circ$ ，方位角： $-27317 \times 0.001 = -27.317^\circ$ 。

温度为数据域的第 13-14 字节

解析：第 13 字节 的高四位为符号位，0：正温度，1：负温度，低四位为十位；

第 14 字节 的高四位为个位，低四位为十分位；

如从机回复 03H，06H 温度为： 30.6°C 。

三. 设置传感器相对/绝对零点:

Modbus 功能码 06H

设置相对/绝对零点命令:		从机响应:	
传感器地址	01H	传感器地址	01H
功能码	06H	功能码	06H
访问寄存器	00H	寄存器	00H
首地址	10H	地址	10H
该字若为非零则为	00 H	该字若为非零则为相	00H
相对零点，为零则	FFH / 00H	对零点，为零则为绝	FFH / 00H
为绝对零点	相对 / 绝对	对零点	相对/ 绝对
CRC	C84FH/ 880FH	CRC	C84FH/ 880FH

命令都必须连续两次发送才会有效，且两次中间不能有其他命令。

设置相对/绝对零点命令应用举例:								
主机发送	01H	06H	00H	10H	00H	FFH	C8H	4FH
从机回复	01H	06H	00H	10H	00H	FFH	C8H	4FH

注：0010 为寄存器地址，该寄存器控制传感器输出为相对零点，还是绝对零点。若为非零（如上述例子中，被写入了 00FFH），则输出为相对零点。相反，若为零（将第 5，第 6 字节改为 00H），则为绝对零点。最后两字节为 CRC 校验和。

四. 设置传感器地址: (传感器地址出厂默认为 1)

设置传感器地址码命令:		从机响应:	
传感器地址	01H	传感器地址	01H
功能码	06H	功能码	06H
寄存器	00H	寄存器	00H
地址	11H	地址	11H
传感器的新地址	00H	传感器的新地址	00H
	04H		04H
CRC	D80C	CRC	D80C

命令都必须连续两次发送才会有效，且两次中间不能有其他命令。

设置传感器地址命令应用举例：

主机发送	01H	06H	00H	11H	00H	04H	D8H	0CH
从机回复	01H	06H	00H	11H	00H	04H	D8H	0CH

注：0011H 为寄存器地址，该寄存器控制传感器地址。上述例子中，传感器的地址被改为了 0004H，最后两字节为 CRC 校验和。

五. 设置传感器波特率: (出厂默认是 9600bps)

设置传感器波特率命令：		从机响应：	
传感器地址	01H	传感器地址	01H
功能码	06H	功能码	06H
寄存器地址	00H	寄存器地址	00H
	12H		12H
传感器的波特率	00H	传感器的波特率	00H
	XX		XX
CRC	CRC LH	CRC	CRC LH

XX： A0H:4800 A1H:9600 A2H:19200 A3H:38400 A4H:115200

命令都必须连续两次发送才会有效，且两次中间不能有其他命令。

设置传感器波特率命令应用举例：

主机发送	01H	06H	00H	12H	00H	A2H	A8H	76H
从机回复	01H	06H	00H	12H	00H	A2H	A8H	76H

注：0012H 为寄存器地址，该寄存器控制传感器波特率。上述例子中，把传感器的波特率设置成 19200，最后两字节为 CRC 校验和。

六. 读取传感器地址码：

读取传感器地址码命令：		从机响应：	
传感器地址	00H	传感器地址	01H
功能码	42H	功能码	42H
寄存器地址	00H	返回数据长度	02H
	11H		
数据长度	00H	传感器地址	01H
	01H		01H
CRC	CRC LH	CRC	CRC LH

读取传感器地址命令应用举例：

主机发送	00H	42H	00H	11H	00H	01H	E9H	D1H
从机回复	01H	42H	02H	01H	01H	6CH	28H	

七. 设置传感器通讯字符格式: (出厂默认是偶校验)

设置传感器通讯字符格式命令:		从机响应:	
传感器地址	01H	传感器地址	01H
功能码	06H	功能码	06H
寄存器	00H	寄存器	00H
地址	09H	地址	09H
传感器的新格式	00H	传感器的新格式	00H
	00H		00H
CRC	59 C8	CRC	59 C8

注意: 命令都必须连续两次发送才会有效, 且两次中间不能有其他命令, 重新上电有效。

设置传感器通讯字符格式应用举例:

主机发送	01H	06H	00H	09H	00H	00H	59H	C8H
从机回复	01H	06H	00H	09H	00H	00H	59H	C8H

以上举例为把字节格式设置成: 一个起始位+8 个数据位 无奇偶校验 +1 个停止位, 重新上电后有效; 出厂默认是一个起始位+8 个数据位 偶校验 +1 个停止位。

注: 0009 为寄存器地址, 该寄存器控制传感器通讯字符格式。

0000H: 一个起始位+8 个数据位 无校验 +1 个停止位。

0001H: 一个起始位+8 个数据位 奇校验 +1 个停止位。

0002H: 一个起始位+8 个数据位 偶校验 +1 个停止位。

八. 设置传感器自动输出频率: (出厂默认是 0Hz, 查询方式)

设置传感器自动输出频率命令:		从机响应:	
传感器地址	01H	传感器地址	01H
功能码	06H	功能码	06H
寄存器	00H	寄存器	00H
地址	13H	地址	09H
传感器的输出频率	00H	传感器的输出频率	00H
	XXH		XXH
CRC	CRC LH	CRC	CRC LH

XXH:

00H: 0Hz (查询方式); 01H: 5Hz; 02H: 10Hz; 03H: 20Hz;

04H: 25Hz; 05H: 50Hz; 06H: 100Hz;

注意: 命令都必须连续两次发送才会有效, 且两次中间不能有其他命令。

设置传感器自动输出频率命令应用举例:

主机发送	01H	06H	00H	13H	00H	04H	79H	CCH
从机回复	01H	06H	00H	13H	00H	04H	79H	CCH

以上举例为把字节格式设置成: 自动输出频率为: 25Hz。

九. 传感器航向角清零命令:

传感器航向角清零命令:		从机响应:	
传感器地址	01H	传感器地址	01H
功能码	06H	功能码	06H
寄存器地址	00H	寄存器地址	00H
15H		15H	
传感器清零命令	00H	传感器清零命令	00H
	FFH		FFH
CRC	D8 4E	CRC	D8 4E

传感器航向角清零应用举例:								
主机发送		01H	06H	00H	15H	00H	FFH	D8H 4EH
从机回复								
01H	06H	00H	15H	00H	FFH	D8H	4EH	

以上举例为把当前航向角清零。