

战术级 MEMS

6 自由度惯性传感器

FSS-IMU16460-C 产品手册

特性

战术级 MEMS 陀螺仪

- $1.5^{\circ}/\text{hr}$ 零偏不稳定性
- $0.12^{\circ}/\sqrt{\text{hr}}$ 角度随机游走
- $0.04^{\circ}/\text{s}$ 温漂 ($-40^{\circ}\text{C}\sim 85^{\circ}\text{C}$, $\leq 1^{\circ}\text{C}/\text{min}$ @ 1σ)

战术级 MEMS 加速度计

- $20\mu\text{g}$ 零偏不稳定性
- $0.02\text{m/s}/\sqrt{\text{hr}}$ 速度随机游走
- 1.0mg 温漂 ($-40^{\circ}\text{C}\sim 85^{\circ}\text{C}$, $\leq 1^{\circ}\text{C}/\text{min}$ @ 1σ)
-

大范围精细化温度补偿

- -40°C 至 85°C 温度补偿
- 精细化温度标定

独立转台标定

- 独立标定每个模块：灵敏度、零偏、非正交误差

高强度工况耐受

- 超强冲击耐受：2000g (0.5ms, 半正弦, 3 轴)
- 超强振动耐受：10g (10~2KHz, 3 轴)
- 全温环境稳定工作： $-40^{\circ}\text{C}\sim 85^{\circ}\text{C}$
- 100%磁屏蔽

实时而灵活的数字接口、体积小巧

- 高达 1KHz 的可配置输出采样率
- 支持串口
- $22.4\times 24.05\times 9.0\text{mm}$, 重量 8.6g

产品概述

FSS-IMU16460-C 是原极科技倾力打造的 6 自由度 MEMS 惯性传感器模块。标配输出三轴陀螺仪与加速度信息。

高精度、高分辨率, 可捕捉细微的震动与倾斜。大量程的输出, 让大动态下的动作感知成为可能。所有模块出厂前都配置超宽温域的精细化温补与独立标定, 让每个模块都能在各种极限工况下稳定发挥, 同时保证所有产品性能高度一致。

应用领域

- 割草机
- 自动驾驶：水下机器人、工程车
- 精密测量：井下、隧道、震动、倾斜
- 稳定平台：云台、动中通

在标准性能及输出参数的基础上, 原极也为您的特殊需求提供定制化软件及 LOGO 定制服务, 在产品上助您一臂之力!



目录

1. 性能参数	1
1.1 陀螺仪关键指标	1
1.2 加速度计关键指标	2
2. 外形结构	4
3. 电气特性	5
3.1 最大耐受值	5
3.2 工作条件	5
3.3 IO 阈值特性	5
4. 引脚定义	6
5. 通信协议	7
5.1 串口通信协议	7
5.1.1 串口接口参数	7
5.1.2 数据包格式	7
5.1.3 数据流帧——AHRS 数据	8
5.1.4 命令模式 GET 输出——系统状态	10
5.1.5 命令模式 GET 输出——读取参数	11
5.1.6 命令模式 SET 指令	13
5.1.7 命令模式输出——用户命令响应	16
5.1.9 坐标系设置功能	17
5.1.10 串口连接常见问题	20
6. 坐标系定义	21
7. CRC 查表法计算	22
8. 更新记录	25

1. 性能参数

1.1 陀螺仪关键指标

表 1 陀螺仪关键指标

参数	测试条件/备注	最小值	典型值	最大值	单位
测量范围			±500		°/s
零偏不稳定性 X 轴 ¹	@25°C, ALLAN 方差, 1 σ		1.5		°/hr
零偏不稳定性 Y 轴 ¹			1.5		°/hr
零偏不稳定性 Z 轴 ¹			1.5		°/hr
零偏稳定性	国军标, 10s 平滑		4		°/hr
零偏重复性	国军标		xy: 10 z: 20		°/h
分辨率			0.0088		°/s
轴间非正交			0.02		deg
内部低通截止频率	软件可调整		49		Hz
ODR			1000		Hz
测量延时			7		ms
全温范围零偏变化 ²	-40°C~85°C, $\leq 1^\circ\text{C}/\text{min}$ @1 σ		0.04		°/s
随机游走 X 轴 ¹	@25°C, ALLAN 方差, 1 σ		0.15		°/√hr
随机游走 Y 轴			0.15		°/√hr
随机游走 Z 轴			0.12		°/√hr
刻度系数误差			2		‰
刻度系数非线性			200		ppm

注 1: IEEE 标准, 在静态 25°C 环境下 Allan 方差曲线给出

 注 2: 1°C/分钟升温情况下全温零偏变化 1 σ 值

1.2 加速度计关键指标

表 2 加速度计关键指标

参数	测试条件/备注	最小值	典型值	最大值	单位
测量范围			±16		g
零偏不稳定性 X 轴 ¹	@25°C, ALLAN 方差, 1 σ		25		μ g
零偏不稳定性 Y 轴 ¹			25		μ g
零偏不稳定性 Z 轴 ¹			20		μ g
零偏稳定性	国军标, 10s 平滑		50		μ g
零偏重复性	国军标		0.2		mg
分辨率			0.244		mg
轴间非正交			0.02		deg
内部低通截止频率	软件可调整		49		Hz
ODR			1000		Hz
测量延时			7		ms
全温范围零偏变化 ²	-40°C~85°C, $\leq 1^\circ\text{C}/\text{min}$ @1 σ		1.0		mg
随机游走 X 轴 ¹	@25°C, ALLAN 方差, 1 σ		0.02		m/s/ $\sqrt{\text{hr}}$
随机游走 Y 轴			0.02		m/s/ $\sqrt{\text{hr}}$
随机游走 Z 轴			0.02		m/s/ $\sqrt{\text{hr}}$
刻度系数误差			0.5		‰
刻度系数非线性			100		ppm

注 1: IEEE 标准, 在静态 25°C 环境下 Allan 方差曲线给出

注 2: 1°C/分钟升温情况下全温零偏变化 1 σ 值

图 1 陀螺仪 ALLAN 方差典型曲线

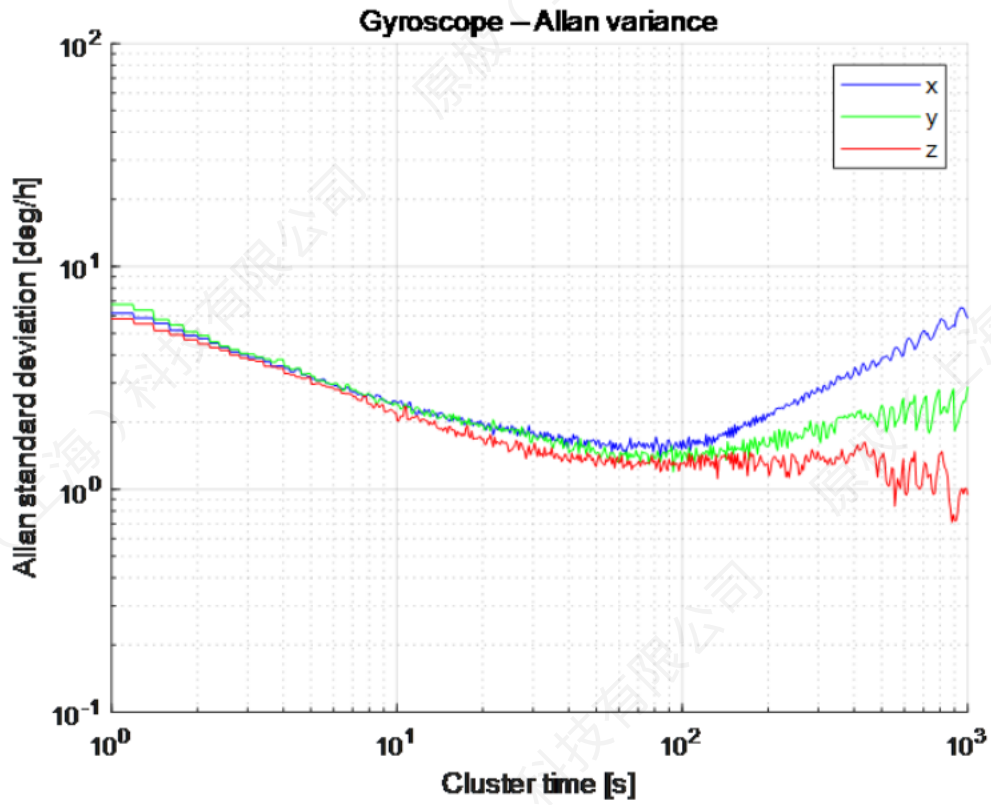
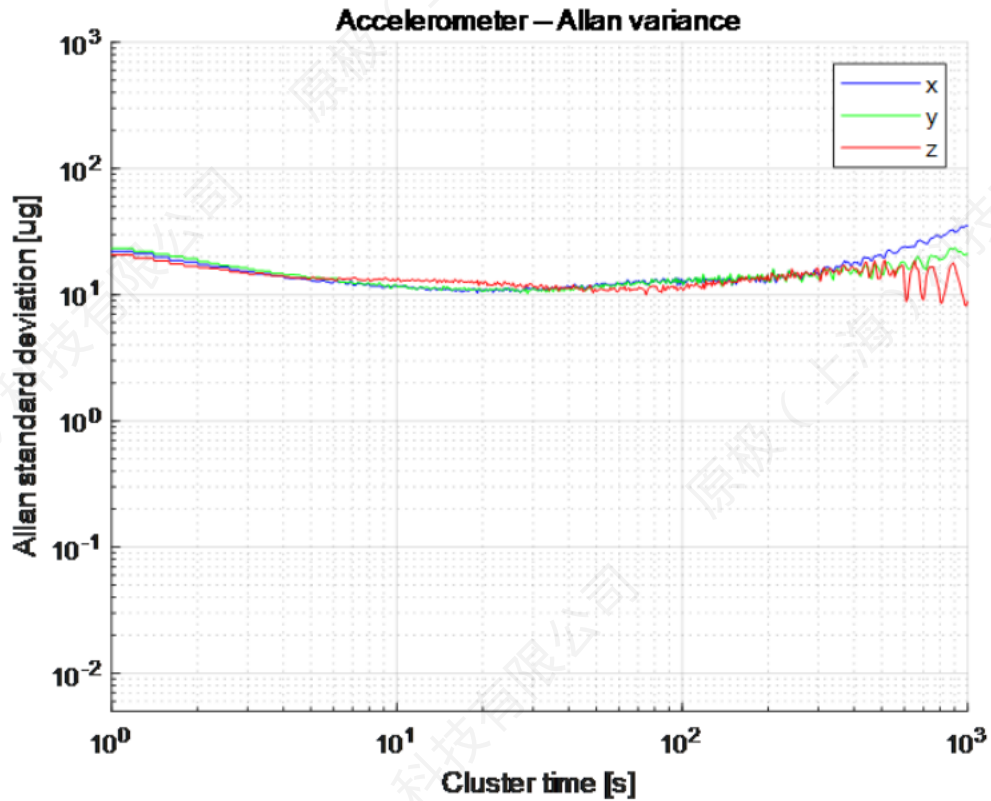
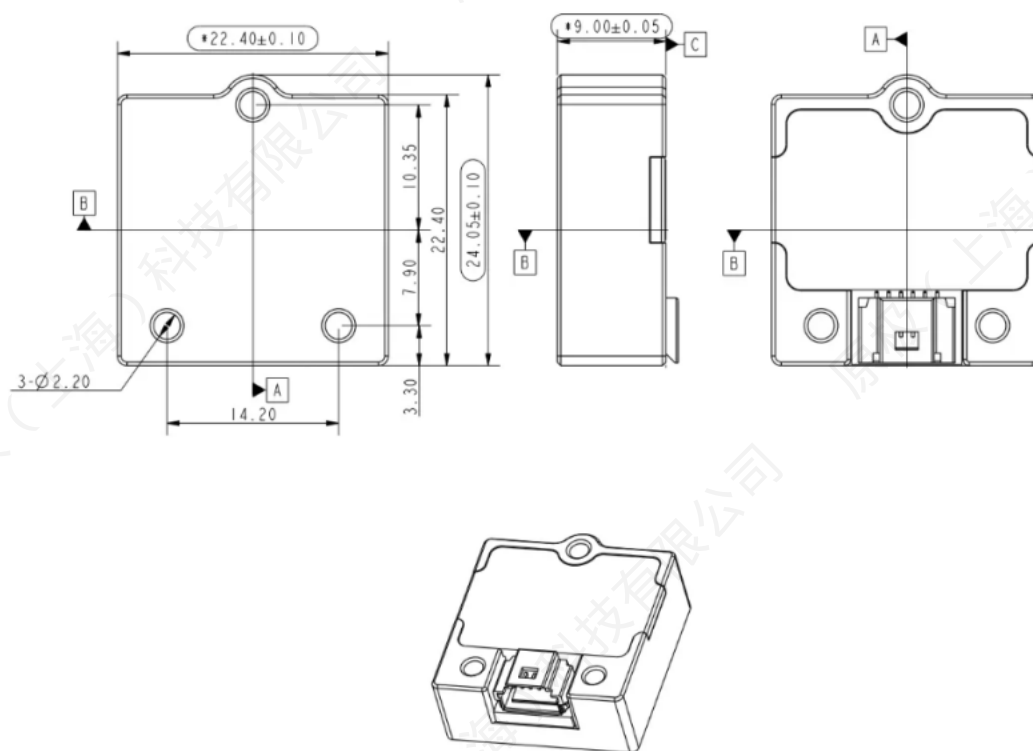


图 2 加速度计 ALLAN 方差典型曲线



2. 外形结构

图 3 外形结构及尺寸 (单位: mm)



3. 电气特性

3.1 最大耐受值

表 3 最大额定绝对值

参数	符号	范围	单位
供电电压	VCC	-0.3 to 6.5	V
电源地	GND	-	-
输入管脚电压	Vin	-0.3 to 5.8	V
使用温度	Tot	-40 to 85	°C
存储温度	Tstg	-40 to 85	°C

3.2 工作条件

表 4 工作条件

参数	符号	最小值	典型值	最大值	单位
供电电压	VIN	4.5	5	5.5	V
VIN 最大纹波	Vrpp		±40		mV
功耗	P		0.12		W
使用温度	Tot	-40		85	°C
存储温度	Tstg	-40		85	°C

注：当电源电压高于 6V 时，内部电压保护电路将切断电源和设备进入复位状态，直到电压回到工作条件，恢复工作。

3.3 IO 阈值特性

表 5 IO 阈值特性

参数	符号	最小值	典型值	最大值	单位
输入管脚低电平	Vin_low	0		VCC*0.2	V
输入管脚高电平	Vin_high	VCC*0.7		VCC+0.2	V
输出管脚低电平	Vout_low	0		0.45	V
输出管脚高电平	Vout_high	VCC-0.45		VCC	V

注：VCC=3.3V

4. 引脚定义

图 4 引脚示意图

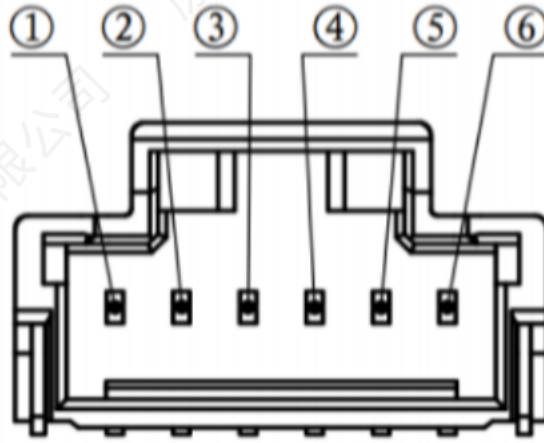


表 6 引脚定义

引脚序号	引脚名称	引脚描述
1	NC	不接
2	NC	不接
3	RX	接收异步数据输入
4	TX	接收异步数据输出
5	GND	电源地
6	VIN	+5V 电源输入

5. 通信协议

5.1 串口通信协议

基于 QT、ROS 和 STM32 的串口协议示例：

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

串口通信具有两种模式：数据流模式(Stream Mode)和命令模式(Command Mode), IMU 在上电初始化完成后，根据参数配置的模式值进入对应模式。

数据流模式：以固定频率周期性输出 AHRS 数据；

命令模式：在此模式下，停止周期性输出，用户通过发送命令与 IMU 进行通信，可通过 GET 指令获取传感器数据、状态、参数等，也可配置 IMU 的参数。

5.1.1 串口接口参数

表 7 串口接口参数

传输速率范围	115200bps ~ 1.5Mbps
默认传输速率	115200bps
开始位	1 bit
数据位	8 bits
停止位	1 bit
奇偶校验	无

5.1.2 数据包格式

IMU 输出和用户输入的数据包结构组成如下：

表 8 IMU 输出和用户输入数据结构

偏移量	数据类型	名称	描述
0	uint8	帧头 1	IMU 输出帧头：0xAA, 0x55 用户输入帧头：0x55, 0xAA
1	uint8	帧头 2	
2	uint16	ID 低位	串口通信帧 ID 的低位字节
3		ID 高位	串口通信帧 ID 的高位字节
4	uint16	数据长度低位	串口通信帧长度的低位字节， length 为 payload 所占字节数，即为 n

5		数据长度高位	串口通信帧长度的高位字节， length 为 payload 所占字节数，即为 n
6	uint8	Payload (n 个字节)	数据负载
6+n	Uint32	CRC_CEHCK (32 位数据低字节)	CRC 校验
7+n		CRC_CEHCK (32 位数据中低字节)	
8+n		CRC_CEHCK (32 位数据中高字节)	
9+n		RC_CEHCK (32 位数据高字节)	

注 1：数据以小端格式传输，低字节在前，高字节在后

注 2：crc32 的初值为 1，CRC 计算不包括本身的本帧所有数据，查表计算法见文档末尾

5.1.3 数据流帧——AHRS 数据

表 9 串口 AHRS 数据格式

	帧头	帧头	ID	length	payload	帧尾
数据类型	uint8	uint8	uint16	uint16	A1	uint32
编码	0xAA	0x55	0x0002	0x002C		crc32

注 1：最大输出更新率不大于 200Hz@115200bps

表 10 串口 A1 负载数据格式

offset	名称	数据类型	单位	描述
0	timer	uint32	μs	时间标
4	pitch	float	°	俯仰角
8	roll	float	°	横滚角
12	yaw	float	°	航向角
16	ax	float	g	X 轴加速度
20	ay	float	g	Y 轴加速度
24	az	float	g	Z 轴加速度
28	gx	float	°/s	X 轴角速度

32	gy	float	°/s	Y 轴角速度
36	gz	float	°/s	Z 轴角速度
40	temp	float	°C	IMU 芯片温度

例：获取到 AHRS 数据流：

AA 55 02 00 2C 00 6D 89 16 05 8F C2 65 40 14 AE 07 BF 5C 0F B2 43 25 06 81 3D
 BC 74 13 3C 60 E5 80 BF EC 51 38 BD 0A D7 A3 BB CD CC CC BC D7 A3 EE 41 0C
 BF 84 80

解析如下：

表 11 串口 A1 获取到 AHRS 数据流

描述	原始值	解析值	描述	原始值	解析值
ID	0200	02	Y 轴加速度	BC74133C	0.009g
长度	2C00	44	Z 轴加速度	60E580BF	-1.007g
时间标	6D891605	85363053	X 轴角速度	EC5138BD	-0.045°/s
俯仰角	8FC26540	3.59°	Y 轴角速度	0AD7A3BB	-0.005°/s
横滚角	14AE07BF	-0.53°	Z 轴角速度	CDCCCCBC	-0.025°/s
航向角	5C0FB243	356.12°	imu 芯片温度	D7A3EE41	29.83°C
X 轴加速度	2506813D	0.063g	crc32 校验	0CBF8480	2156183308

5.1.4 命令模式 GET 输出——系统状态

表 12 串口系统状态数据格式

	帧头	帧头	ID	length	payload	帧尾
数据类型	uint8	uint8	uint16	uint16	S1	uint32
编码	0xAA	0x55	0x00FF	N		crc32

注 1：不同 IMU 型号，此帧的长度会有差别，都代表 S1 的长度，需要根据 imu 型号确认。

offset	名称	数据类型	描述
0	Software_ver	uint32	软件版本号
4	Hardware_ver	uint32	硬件版本号
8	rev	uint16	保留字节
10	sn0	uint32	第一 SN 号
14	sn1	uint32	第二 SN 号
18	sn2	uint32	第三 SN 号
22	Board_version	uint32	底板版本号
26	Rev[16]	Uint8	后续都是保留字节

表 13 串口 S1 负载数据格式

注 1：不同 IMU 型号，后续保留字节也不同，需要根据 imu 型号进行确认，IMU614E 为 16 字节。

例：获取系统状态

输入数据：55 AA 01 00 18 00 BD DB 31 34

响应数据：AA 55 FF 00 2A 00 1F 39 03 00 65 6F 01 00 50 83 30 33 35 55 34 50 15 FF 8F 5F FF FF 50 83 FF 1F 29 00 00 00 00 E0 00 07 10 17 08 50 D0 37 10 3B 7A C3 00 02

根据响应数据，解析得到软件版本号 211231(1F 39 03 00)，硬件版本号 94053(65 6F 01 00)。

5.1.5 命令模式 GET 输出——读取参数

表 14 串口参数输入数据格式

	帧头	帧头	ID	length	payload	帧尾
数据类型	uint8	uint8	uint16	uint16	P1	uint32
编码	0x55	0xAA	0x0006	0x0018		crc32

表 15 串口参数输出数据格式

	帧头	帧头	ID	length	payload	帧尾
数据类型	uint8	uint8	uint16	uint16	P1	uint32
编码	0xAA	0x55	0x7530	0x0018		crc32

注 1：读取参数时，IMU 会将数据流关闭，设置完毕后需要重新开启数据流。

表 16 串口 P1 负载数据格式

offset	名称	数据类型	描述
0	Param1	float	获取的参数（输入数据可无视）
4	Param2	float	保留，默认为 0
8	Param3	uint32	设置的参数索引
12	Param4	uint32	保留，默认为 0
16	Param5	Int32	保留，默认为 0
20	Param6	Int32	保留，默认为 0

表 17 串口 P1 负载参数索引表

Param3	Param1	单位
3	串口输出波特率，支持以下波特率 115200、230400、460800、921600、1500000	bps
4	坐标系朝向 (见表 24 坐标系朝向对应表)	
8	X 轴陀螺零偏标定结果，GYRO_X_OFF	°/s
9	Y 轴陀螺零偏标定结果，GYRO_Y_OFF	°/s
10	Z 轴陀螺零偏标定结果，GYRO_Z_OFF	°/s

21	AHRS 输出频率, 默认 100Hz	Hz
31	内部滤波器配置, 定义同 SPI 的 FILTER_CTRL 对照表	

例: 获取 AHRS 输出频率

输入数据: 55 AA 06 00 18 00 00 00 00 00 00 00 00 00 15 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 66 CB 46 AC

响应数据: AA 55 30 75 18 00 00 00 48 42 00 00 00 00 15 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 31 2F A2 0A

根据响应数据, 解析得到输出频率为 50hz (00 00 48 42)。

5.1.6 命令模式 SET 指令

表 18 串口输入命令格式

	帧头	帧头	ID	length	payload	帧尾
数据类型	uint8	uint8	uint16	uint16	R1	uint32
编码	0x55	0xAA	CMD	0x0018		crc32

注 1: CMD 与 R1 关系, 详见 R1 负载参数索引表

表 19 串口 R1 负载数据格式

offset	名称	数据类型	描述
0	Param1	float	设置的参数
4	Param2	float	保留, 默认为 0
8	Param3	uint32	设置的参数索引
12	Param4	uint32	保留, 默认为 0
16	Param5	Int32	保留, 默认为 0
20	Param6	Int32	保留, 默认为 0

表 20 串口 R1 负载参数索引表

CMD	Param1	Param3	描述
1	0	0	触发获取一次系统状态数据
2	0	0	触发获取一次 AHRS 数据
3	<mode>	0	设置输出模式: Mode=1, 数据流输出 AHRS Mode=100, 禁止数据流模式, 进入 COMMAD 模式
5	0	0	保存当前参数到 FLASH
6	0	<value>	读取参数, value 为要读取的参数索引, 即 P1.index, 详见串口应答性输出-参数读取 例如需读取 AHRS 输出频率 (ODR), 则设置 value=21 例如需读取串口波特率, 则设置 value=3 例如需读取内部滤波器, 则设置 value=31 例如需读取坐标系方向, 则设置 value=4

9	0	0	执行软件重启
14	<value>	3	设置串口输出波特率，单位 bps value 的有效值为： 115200, 230400, 460800, 921600, 1500000 value 为其他值时，默认采用 115200bps 设置波特率参数后，需要重启才能生效。 不断电的设置流程：设置波特率，保存参数到 flash，执行软件复位
14	<value>	21	设置周期性 AHRS 数据输出频率，单位 Hz value 的常用值为： 1, 10, 50, 100, 200, 500, 1000 输出频率与串口波特率的推荐对应关系 1000Hz: 921600bps 500Hz: 460800bps 250Hz: 460800bps 200Hz: 460800bps 100Hz: 115200bps
14	<value>	31	内部滤波器配置，定义同 SPI 加速度计和陀螺仪滤波器配置，默认 0xBB，即 47Hz
14	<value>	4	设置 IMU 坐标系朝向，value 的取值范围为 101~124，具体坐标系朝向对应关系见表 24

注 1：请注意本表中数值均为十进制

注 2：可使用上位机命令生成器功能生成对应命令发送，使用方法见本手册上位机使用部分

如执行开启 AHRS 输出：

CMD ID 填入 3，参数 1 填入 1，生成的十六进制数组可以填入串口助手或程序数组中发送给 IMU。

命令生成器

55,aa,03,00,18,00,00,00,80,3f,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,52,d8,8e,e8

CMD ID:

3

参数:

1

1

2

0

3

0

4

0

5

0

6

0

生成命令

发送命令

5.1.7 命令模式输出——用户命令响应

表 21 设置参数串口响应数据格式

	帧头	帧头	ID	length	ACK	Param3	帧尾
数据类型	uint8	uint8	uint16	uint16	uint16	uint16	uint32
编码	0xAA	0x55	0x753D	0x0004	0x7534	参数索引	crc32

表 22 保留参数串口响应数据格式

	帧头	帧头	ID	length	ACK	result	帧尾
数据类型	uint8	uint8	uint16	uint16	uint16	uint16	uint32
编码	0xAA	0x55	0x753D	0x0004	0x0005	0x01	crc32

表 23 串口用户命令响应数据格式

	帧头	帧头	ID	length	command	result	帧尾
数据类型	uint8	uint8	uint16	uint16	uint16	uint16	uint32
编码	0xAA	0x55	0x0064	0x0004	命令 ID	0x01	crc32

例：设置串口输出波特率 115200

输入数据：

55,AA,0E,00,18,00,00,00,E1,47,00,00,00,00,03,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,56,2B,4D,93

响应数据：AA 55 3D 75 04 00 34 75 03 00 A7 98 2A 54

设置周期性 AHRS 数据输出频率 100hz

输入数据：55 AA 0E 00 18 00 00 00 C8 42 00 00 00 00 15 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0A 2B 2C 8D

响应数据：AA 55 3D 75 04 00 34 75 15 00 70 2D B2 48

保存当前参数到 FLASH

输入数据：55 AA 05 00 18 00 C9 2F E6 32

响应数据：AA 55 3D 75 04 00 05 00 01 00 5A CF B1 7C

设置输出模式为 AHRS 数据流

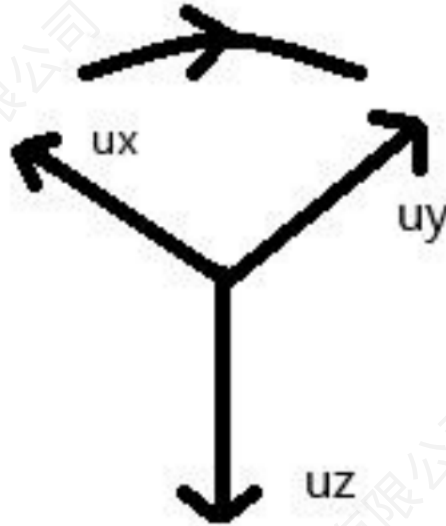
输入数据：55 AA 03 00 18 00 00 00 80 3F 00 52 D8 8E E8

响应数据：AA 55 64 00 04 00 03 00 01 00 E7 87 E3 AD

5.1.9 坐标系设置功能

设置固件坐标系，在上位机当中显示对应固件设计坐标系

图 5 固件原始坐标系



按照上图规则，当 x 和 y 轴确定之后， z 轴确定。 Z 轴垂直于 X 轴到 Y 轴的面。

$X/Y/Z$ 三轴的朝向总共有二十四种，如下表所示：

表 24 坐标系朝向对应表

朝向 (value)	XAxis	YAxis	ZAxis	说明
101	+Ux	+Uy	+Uz	默认朝向
102	-Ux	-Uy	+Uz	
103	-Uy	+Ux	+Uz	
104	+Uy	-Ux	+Uz	
105	-Ux	+Uy	-Uz	
106	+Ux	-Uy	-Uz	
107	+Uy	+Ux	-Uz	
108	-Uy	-Ux	-Uz	
109	-Uz	+Uy	+Ux	
110	+Uz	-Uy	+Ux	
111	+Uy	+Uz	+Ux	

112	-Uy	-Uz	+Ux
113	+Uz	+Uy	-Ux
114	-Uz	-Uy	-Ux
115	-Uy	+Uz	-Ux
116	+Uy	-Uz	-Ux
117	-Ux	+Uz	+Uy
118	+Ux	-Uz	+Uy
119	+Uz	+Ux	+Uy
120	-Uz	-Ux	+Uy
121	+Ux	+Uz	-Uy
122	-Ux	-Uz	-Uy
123	-Uz	+Ux	-Uy
124	+Uz	-Ux	-Uy

如何更改坐标系为 102 朝向：

CMD ID 填入 14，参数 1 填入 102，参数 3 填入 4，生成的十六进制数组可以填入串口助手或程序数组中发送给 IMU。

串口号: COM11 & USB S 波特率: 115200 刷新

固件版本: 220811
 硬件版本: 00
 最新版本: 6142
 配置前区地址码: 0
 校准前区地址码: 0
 主从机: 从机
 序列号: 38373454415015FF
 405+FFFF

命令生成器

55 aa 0e 00 18 00 00 00 cc
 42 00 00 00 00 04 00 00 00 00 00 00 00 00 00 00 05 17 c2 99

CMD ID: 14
 参数: 1 102 2 0 3 4
 4 0 5 0 6 0

生成命令 发送命令 串口数据显示

命令生成器使用说明
 举例说明:
 6. 执行并启动AHRS输出: CMD ID填入3, 参数1填入1, 点击生成命令按钮, 则生成相应的命令, 生成的十六进制数组可以填入串口助手或点云发送命令按钮(前提是已打开串口通讯)将数据发送给IMU

命令索引表

命令ID	参数1	参数3	功能描述
1	0	0	触发获取一次系统状态数据
2	0	0	触发获取一次AHRS数据
3	<mode>	0	设置输出模式: Mode=1, 数据流输出AHRS Mode=100, 禁止数据流模式, 进入COMMAND模式
5	0	0	保存当前参数到FLASH
6	0	<value>	读取参数, value为要读取的参数索引, 例如: 读取串口输出波特率, 则设置value=3; 读取AHRS输出频率(ODR), 则设置value=21; 读取内部波特率配置, 则设置value=31
9	0	0	执行软件复位

设置串口输出波特率, 单位bps, value的有效值为:

如何读取坐标系朝向:

CMD ID 填入 06，参数 3 填入 4，生成的十六进制数组可以填入串口助手或程序数组中发送给 IMU。



例： 设置坐标系为 115 朝向

输入数据:

55,aa,0e,00,18,00,00,00,e6,42,00,00,00,00,04,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,46.6a.4e.86

响应数据: AA 55 3D 75 04 00 34 75 04 00 60 0E 6B 1B

参考表 21 解析得到参数索引为 04，设置成功

读取坐标系：

输入数据: 55 AA 06 00 18 00 00 00 00 00 00 00 00 00 04 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 69 64 09 E4

响应数据: AA 55 30 75 18 00 00 00 E6 42 00 00 00 00 04 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 B2 2F 2D 4E

根据表 15 与表 16，解析得到参数 1 为 115 (float)，参数 3 为 04。即坐标系为 115 朝向。

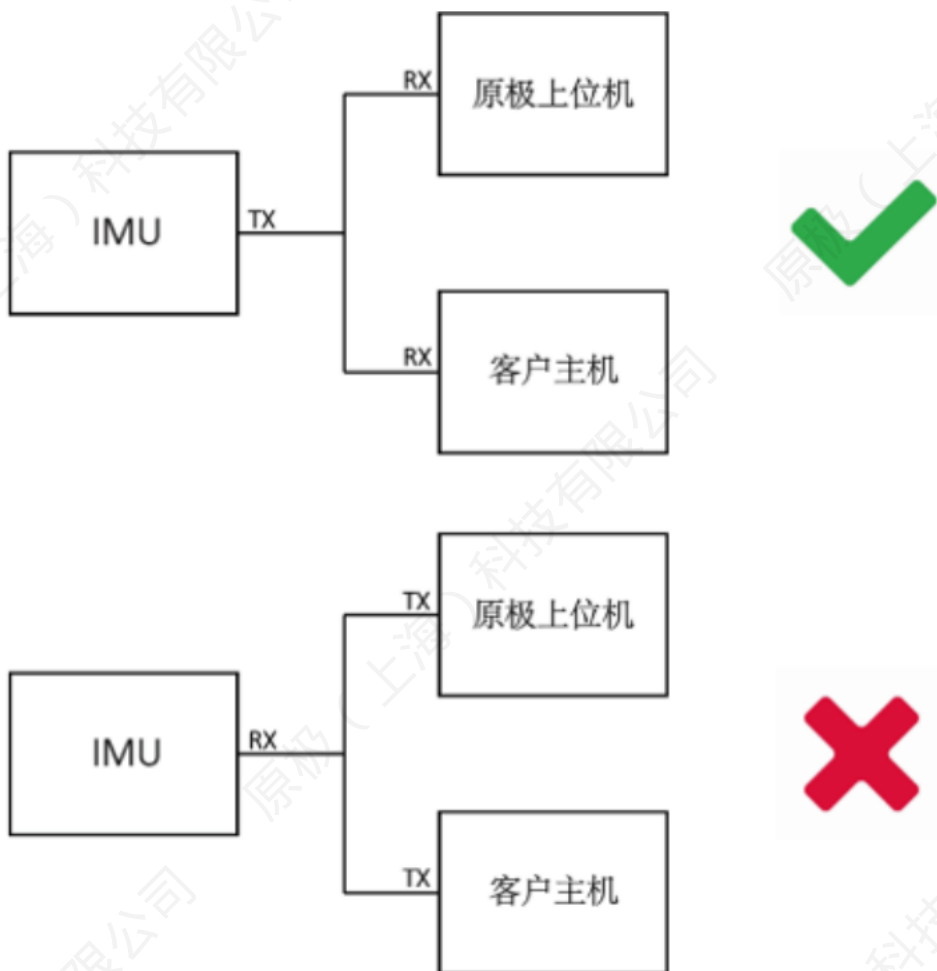
5.1.10 串口连接常见问题

1) IMU 的 RX 不能接 2 个主机 TX

串口的 RX 不能同时接 2 个 TX，所以如果需要连接原极上位机时，需要断开其与用户主机的串口通信，否则上位机只能接收到数据，不能发送命令给 IMU。

如下图所示：

图 6 串口连接方式示意图



注：IMU TX 可接多路 RX，RX 不可接多路 TX；
IMU 串口不可同时连接客户主机和原极上位机；
IMU 可以预留另外一路串口专门连接原极上位机。

2) 获取不到版本号

检查串口线是否丢包，推荐使用 FT232 芯片的串口线，CH340、PL2303 数据线在高波特率时 (>115200bps) 会丢包

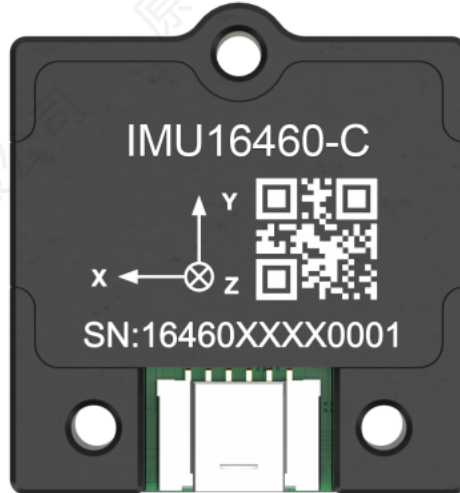
建议串口线直连，不建议串联，如 RS422 的接口接电脑，直接使用 RS422 转 USB 线，不要用 RS422 转 RS232+RS232Z 转 USB 线串联。

3) 上位机曲线显示卡顿

如果是 FT232 数据线，用系统管理员打开上位机，自动配置串口延时
手动在设备管理器中配置串口延时。

6. 坐标系定义

图 7 坐标系示意图



本产品坐标系使用 前-右-下(FRD)坐标系，欧拉角范围如下：

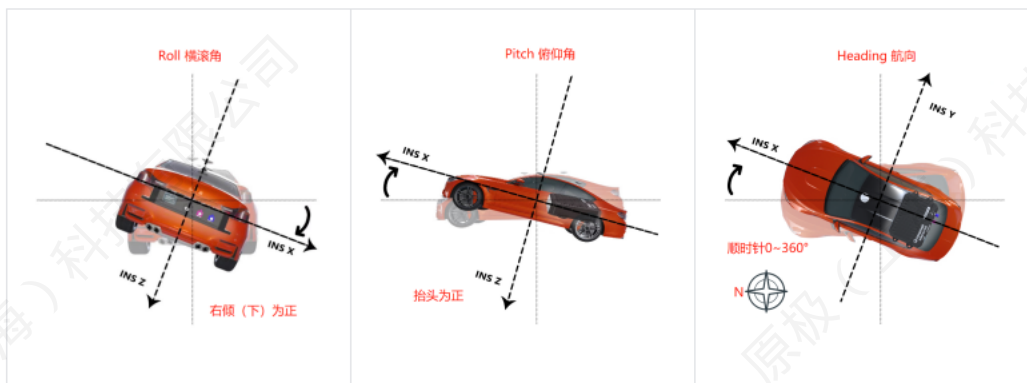
绕 Z 轴方向旋转：航向角 Yaw 范围： $0^{\circ} \sim 360^{\circ}$ ；

绕 X 轴方向旋转：横滚角 Roll 范围： $-180^{\circ} \sim 180^{\circ}$ ；

绕 Y 轴方向旋转：俯仰角 Pitch 范围： $-90^{\circ} \sim 90^{\circ}$ 。

横滚、俯仰、航向角度示意图如下：

图 8 横滚、俯仰、航向角示意图



7. CRC 查表法计算

建议直接参考示例代码。

注 1：数据以小端格式传输，低字节在前，高字节在后

注 2：crc32 的初值为 1，CRC 计算不包括本身的本帧所有数据

C++

```
static const uint32_t crc32_tab [ ] = {  
0x00000000, 0x77073096, 0xee0e612c, 0x990951ba, 0x076dc419,  
0x706af48f,  
0xe963a535, 0x9e6495a3, 0x0edb8832, 0x79dcb8a4, 0xe0d5e91e,  
0x97d2d988,  
0x09b64c2b, 0x7eb17cbd, 0xe7b82d07, 0x90bf1d91, 0x1db71064,  
0x6ab020f2,  
0xf3b97148, 0x84be41de, 0x1adad47d, 0x6ddde4eb, 0xf4d4b551,  
0x83d385c7,  
0x136c9856, 0x646ba8c0, 0xfd62f97a, 0x8a65c9ec, 0x14015c4f,  
0x63066cd9,  
0xfa0f3d63, 0x8d080df5, 0x3b6e20c8, 0x4c69105e, 0xd56041e4,  
0xa2677172,  
0x3c03e4d1, 0x4b04d447, 0xd20d85fd, 0xa50ab56b, 0x35b5a8fa,  
0x42b2986c,  
0xdbbbc9d6, 0xacbcf940, 0x32d86ce3, 0x45df5c75, 0xdcd60dcf,  
0xabd13d59,  
0x26d930ac, 0x51de003a, 0xc8d75180, 0xbfdd06116, 0x21b4f4b5,  
0x56b3c423,  
0xcfba9599, 0xb8bda50f, 0x2802b89e, 0x5f058808, 0xc60cd9b2,  
0xb10be924,  
0x2f6f7c87, 0x58684c11, 0xc1611dab, 0xb6662d3d, 0x76dc4190,  
0x01db7106,  
0x98d220bc, 0xefd5102a, 0x71b18589, 0x06b6b51f, 0x9fbfe4a5,  
0xe8b8d433,  
0x7807c9a2, 0x0f00f934, 0x9609a88e, 0xe10e9818, 0x7f6a0dbb,  
0x086d3d2d,  
0x91646c97, 0xe6635c01, 0x6b6b51f4, 0x1c6c6162, 0x856530d8,  
0xf262004e,  
0x6c0695ed, 0x1b01a57b, 0x8208f4c1, 0xf50fc457, 0x65b0d9c6,
```

0x12b7e950,
 0x8bbbeb8ea, 0xfcb9887c, 0x62dd1ddf, 0x15da2d49, 0x8cd37cf3,
 0xfbd44c65,
 0x4db26158, 0x3ab551ce, 0xa3bc0074, 0xd4bb30e2, 0x4adfa541,
 0x3dd895d7,
 0xa4d1c46d, 0xd3d6f4fb, 0x4369e96a, 0x346ed9fc, 0xad678846,
 0xda60b8d0,
 0x44042d73, 0x33031de5, 0xaa0a4c5f, 0xdd0d7cc9, 0x5005713c,
 0x270241aa,
 0xbe0b1010, 0xc90c2086, 0x5768b525, 0x206f85b3, 0xb966d409,
 0xce61e49f,
 0x5edef90e, 0x29d9c998, 0xb0d09822, 0xc7d7a8b4, 0x59b33d17,
 0x2eb40d81,
 0xb7bd5c3b, 0xc0ba6cad, 0xedb88320, 0x9abfb3b6, 0x03b6e20c,
 0x74b1d29a,
 0xead54739, 0x9dd277af, 0x04db2615, 0x73dc1683, 0xe3630b12,
 0x94643b84,
 0x0d6d6a3e, 0x7a6a5aa8, 0xe40ecf0b, 0x9309ff9d, 0x0a00ae27,
 0x7d079eb1,
 0xf00f9344, 0x8708a3d2, 0x1e01f268, 0x6906c2fe, 0xf762575d,
 0x806567cb,
 0x196c3671, 0x6e6b06e7, 0xfed41b76, 0x89d32be0, 0x10da7a5a,
 0x67dd4acc,
 0xf9b9df6f, 0x8ebeeff9, 0x17b7be43, 0x60b08ed5, 0xd6d6a3e8,
 0xa1d1937e,
 0x38d8c2c4, 0x4fdff252, 0xd1bb67f1, 0xa6bc5767, 0x3fb506dd,
 0x48b2364b,
 0xd80d2bda, 0xaf0a1b4c, 0x36034af6, 0x41047a60, 0xdf60efc3,
 0xa867df55,
 0x316e8eef, 0x4669be79, 0xcb61b38c, 0xbc66831a, 0x256fd2a0,
 0x5268e236,
 0xcc0c7795, 0xbb0b4703, 0x220216b9, 0x5505262f, 0xc5ba3bbe,
 0xb2bd0b28,
 0x2bb45a92, 0x5cb36a04, 0xc2d7ffa7, 0xb5d0cf31, 0x2cd99e8b,
 0x5bdeae1d,
 0x9b64c2b0, 0xec63f226, 0x756aa39c, 0x026d930a, 0x9c0906a9,

```
0xeb0e363f,  
0x72076785, 0x05005713, 0x95bf4a82, 0xe2b87a14, 0x7bb12bae,  
0x0cb61b38,  
0x92d28e9b, 0xe5d5be0d, 0x7cdcefb7, 0x0bdbdf21, 0x86d3d2d4,  
0xf1d4e242,  
0x68ddb3f8, 0x1fda836e, 0x81be16cd, 0xf6b9265b, 0x6fb077e1,  
0x18b74777,  
0x88085ae6, 0xff0f6a70, 0x66063bca, 0x11010b5c, 0x8f659eff,  
0xf862ae69,  
0x616bffd3, 0x166ccf45, 0xa00ae278, 0xd70dd2ee, 0x4e048354,  
0x3903b3c2,  
0xa7672661, 0xd06016f7, 0x4969474d, 0x3e6e77db, 0xaed16a4a,  
0xd9d65adc,  
0x40df0b66, 0x37d83bf0, 0xa9bcae53, 0xdebb9ec5, 0x47b2cf7f,  
0x30b5ffe9,  
0xbdbdf21c, 0xcabac28a, 0x53b39330, 0x24b4a3a6, 0xbad03605,  
0xcdd70693,  
0x54de5729, 0x23d967bf, 0xb3667a2e, 0xc4614ab8, 0x5d681b02,  
0x2a6f2b94,  
0xb40bbe37, 0xc30c8ea1, 0x5a05df1b, 0x2d02ef8d,  
}  
uint32_t crc_crc32 (uint32_t crc, const uint8_t *buf, uint32_t  
size ) {  
    for (uint32_t i=0; i<size ; i++) {  
        crc = crc32_tab [ (crc ^ buf [i] ) & 0xff] ^ (crc >> 8 );  
    }  
    return crc;  
}
```

8. 更新记录

版本	日期	状态/注释
版本 1.0	2024. 05. 16	首版
版本 1.1	2024. 06. 05	更新引脚定义