

LPMS-ME1

用户手册 ver. 2.0



广州阿路比电子科技有限公司

<http://www.alubi.cn>

历史版本

日期	版本号	更改
2019-11-22	ver. 2.0	<ul style="list-style-type: none">• 将用户手册简化，部分内部移到硬件手册
2019-06-14	ver. 1.17	<ul style="list-style-type: none">• 增加了部分引脚的说明• 增加了部分寄存器的说明
2019-04-10	ver. 1.16	<ul style="list-style-type: none">• 增加了 SPI 功能的说明• 增加了部分寄存器的描述• 修订了 LRC 的计算方式
2018-08-13	ver. 1.15	<ul style="list-style-type: none">• 修订了一术语性错误
2018-03-22	ver. 1.14	<ul style="list-style-type: none">• 修订了 IIC 描述上的错误• 更新了 ME1 引脚描述
2018-01-24	ver. 1.13	<ul style="list-style-type: none">• 修订了通讯接口选择的说明
2017-10-02	ver. 1.12	<ul style="list-style-type: none">• 将数据手册改为用户手册• 修改结构框图• 更新了通信协议例子
2017-08-08	ver. 1.11	<ul style="list-style-type: none">• 增加了 Command Lists 到附录中
2017-05-25	ver. 1.10	<ul style="list-style-type: none">• 增加了 SPI 处于开发中的说明• 增加了磁力计校准例程及详细说明• 修复某些地方的表述错误
2016-10-21	ver. 1.9	<ul style="list-style-type: none">• 模块厚度和外形更新
2016-09-06	ver. 1.8	<ul style="list-style-type: none">• 增添了设置波特率这一例程
2016-08-31	ver. 1.7	<ul style="list-style-type: none">• 增添了参考坐标系章节
2016-08-30	ver. 1.6	<ul style="list-style-type: none">• 修订了发送数据的格式和单位• 增添了通信模式章节• 增添了通信例程
2016-08-29	ver. 1.5	<ul style="list-style-type: none">• 增加高低电平的定义• 增加引脚 15 和 25 的描述• 增加功耗一栏
2016-08-25	ver. 1.4	<ul style="list-style-type: none">• I2C 寄存器名称部分更新• 增添关于数据传输的细节说明
2016-08-18	ver. 1.3	<ul style="list-style-type: none">• UART 部分增加默认波特率• 增加不同模式下数据传输格式
2016-08-16	ver. 1.2	<ul style="list-style-type: none">• I2C 寄存器部分更新• LPMS-ME1 外形更新
2016-07-18	ver. 1.1	<ul style="list-style-type: none">• 初始发布

目录

1. 产品介绍.....	- 1 -
2. 通信协议.....	- 2 -
2.1 LPBUS 协议.....	- 2 -
2.2 UART 通信接口下的通信模式.....	- 2 -
2.3 LPBUS 数据包结构.....	- 3 -
2.4 通信例程.....	- 6 -
将传感器进入 <i>Command Mode</i>	- 6 -
将传感器进入 <i>Streaming Mode</i>	- 7 -
读取传感器配置.....	- 8 -
读取陀螺仪范围.....	- 9 -
设置加速度计范围.....	- 10 -
读取传感器数据.....	- 11 -
保存参数设置.....	- 13 -
读取传感器状态.....	- 14 -
陀螺仪校准.....	- 15 -
磁力计校准.....	- 16 -
设置波特率.....	- 17 -
3. 附录 COMMAND LIST	- 18 -
Summary.....	- 18 -
<i>Acknowledged and Not-acknowledged Identifiers</i>	- 20 -
<i>Configuration and Status Commands</i>	- 20 -
<i>Mode Switching Commands</i>	- 21 -
<i>Gyroscope Settings Command</i>	- 22 -
<i>Accelerometer Settings Command</i>	- 23 -
<i>Magnetometer Settings Command</i>	- 23 -
<i>Data Transmission Commands</i>	- 24 -
<i>Register Value Save and Reset Command</i>	- 26 -
<i>Reference Setting and Offset Reset Command</i>	- 26 -
<i>Filter Settings Command</i>	- 27 -
<i>Device Info</i>	- 28 -



1. 产品介绍

LPMS-ME1 是一款低成本、高精度的 9 轴惯性测量单元。它整合了三轴加速度计、三轴陀螺仪和三轴磁力计等传感器，通过我司独有的算法进行矫正和计算，能提供包括欧拉角、四元数和加速度等信息在内的精确数据。同时，该模块体积小巧，易于组装，方便用户嵌入到自己系统中进行开发和设计。

主要特征：

- MEMS 微型惯性测量单元 (IMU)
- 集成三轴陀螺仪、三轴加速度计、三轴磁力计
- 实时计算传感器的姿态方向、线性加速度以及角速度等数据
- 电源输入：3.3-5.5V
- 通信接口：UART, I2C, SPI
- 封装尺寸：PLCC-28 (12.0x12.0x2.6mm)

应用领域：

- 人体动作捕捉
- 物联网 (IOT) 设备
- 运动性能评估
- 无人机飞行控制



2. 通信协议

2.1 LPBUS 协议

LPBUS 是基于工业标准的 MODBUS 协议所设计的通信协议。这是 LPMS 设备在 UART 通信接口下默认的通信方式。通信协议数据包结构具体请参照以下章节。

一个 LPBUS 通信包含有两项基本的指令形式：**GET** 和 **SET** 指令。指令均由主机（PC、移动数据记录单元等）发送给从机（LPMS 设备）。下面我们会对具体的指令进行详细的介绍，包括指令的类型和传输的数据。

GET 指令：读取从机的数据需通过发送 GET 指令实现。一个 GET 请求指令通常不包含任何数据，所请求的数据由从机收到 GET 指令后发出。

SET 指令：从机数据寄存器的值通过发送 SET 指令设置。一个 SET 指令包含要设置的数据，从机的返回值为 **ACK**（代表成功写入寄存器）或者 **NACK**（代表写入寄存器失败）。

详细的指令表请查看附录。

2.2 UART 通信接口下的通信模式

LPMS 设备具有数据流模式（Streaming Mode）和命令模式（Command Mode）两种通信模式。数据流模式下，传感器以设定好的频率不断往外发送数据（发送的数据类型和格式可进行设置）。命令模式下，需通过发送指令与传感器进行通信，可对传感器的参数进行设置。LPMS-ME1 上电后默认的模式为数据流模式（默认频率为 100Hz，默认发送的数据类型见表 2-2）。

图 2.1 为对传感器进行模式转换和参数设置的一般流程。通常首先要使传感器进入命令模式再进行参数的修改或者读取。

注：

1. 仅有四个指令（见图 2.1）可以在数据流模式下使用。

2. 进入命令模式修改参数过后一定要保存设置（图 2.1 中 Step 4），否则断电后修改无效。

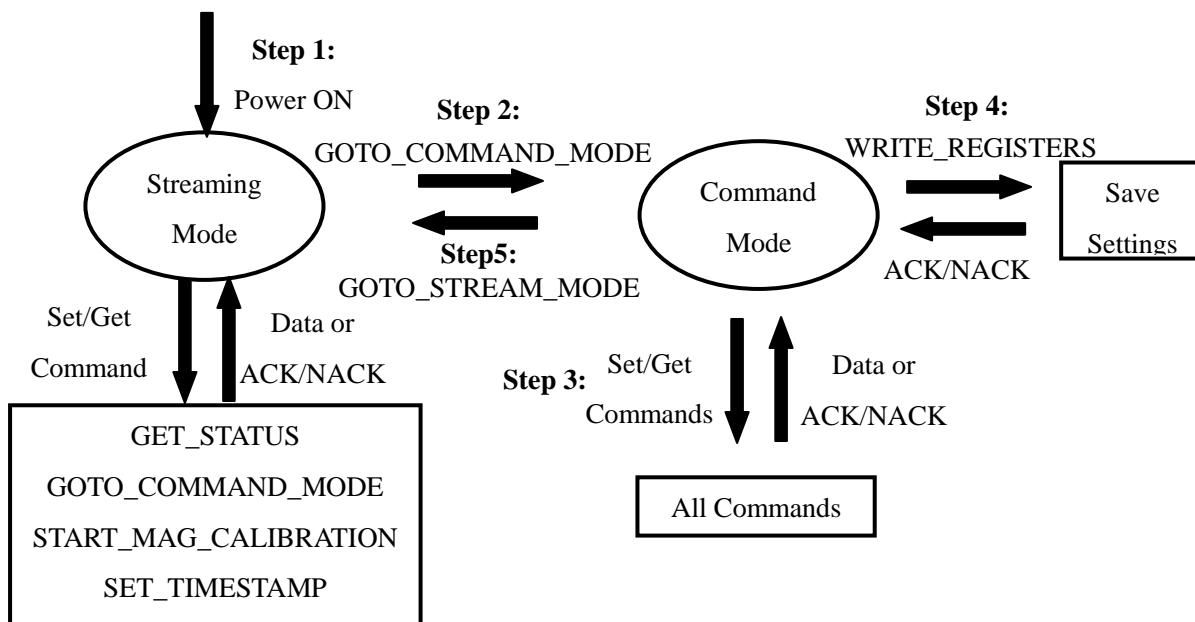


图 2.1. 传感器通信模式转换和参数设置流程

2.3 LPBUS 数据包结构

LPBUS 通信协议的每一个数据包的结构组成如表 2-1 所示。

表 2-1 数据包结构组成

字节#	名称	描述
0	包头	3Ah
1	Sensor ID (低位字节)	包含传感器的 Sensor ID 号的低位字节。ID 号默认值为 1。主机可使用该 ID 号发送 GET/SET 指令到特定的 LPMS 设备中，从机将返回相同的 ID 号。ID 号可通过发送 SET 指令修改。
2	Sensor ID (高位字节)	包含传感器的 Sensor ID 号的高位字节。
3	指令号 (低位字节)	包含所要执行的指令号的低位字节。
4	指令号 (高位字节)	包含所要执行的指令号的高位字节。
5	数据长度 (低位字节)	包含所要传输的数据长度的低位字节。
6	数据长度 (高位字节)	包含所要传输的数据长度的高位字节。
x	数据 (n 个字节)	如果数据长度 n 不等于 0，那么 x=6+n；若 n 为 0，则 x 为空。 如果数据长度不为 0，则该数据含有要传输的数据包，反之该数据为空。



7+n	LRC (低位字节)	LRC 校验的低位字节。为了确保传输数据不失真，我们使用了 LRC 校验和的方法，具体计算如下所示： LRC=sum(Sensor ID, 指令号, 数据长度, 数据)。 计算出来的 LRC 通常与从远程设备传输过来的 LRC 进行比较，如果这两个 LRC 不相等则产生错误报告。
8+n	LRC (高位字节)	LRC 校验的高位字节。
9+n	包尾 (低位字节)	0Dh
10+n	包尾 (高位字节)	0Ah

数据包中的数据部分，以小端格式传输，即低位字节在前，高位字节在后。

数据包中的数据部分，数据格式有两种：

- 32 位浮点型数据格式 (默认设置，时间戳除外，其恒为 32 位无符号整型)
- 16 位无符号整型格式

表 2-2 列出数据包的数据部分能输出的所有类型数据 (最多 8 种数据)，以及其传输顺序及数据格式标志符。数据格式标识符定义请查看表 2-4。

表 2-2 数据包中数据部分的数据顺序及格式 (32 位浮点型模式下)

顺序#	数据格式标识符	传感器数据
1	UInt32	时间戳 (400Hz 更新率,即 0.0025s)
2	Vector3f	校准后的陀螺仪数据 (rad/s)
3	Vector3f	校准后的加速度计数据 (g)
4	Vector3f	校准后的磁力计数据 (μ T)
5	Vector3f	角速度 (rad/s)
6	Vector4f	四元数 (归一化单位)
7	Vector3f	欧拉角数据 (rad)
8	Vector3f	线性加速度数据 (g)

如用户修改传感器为 16 位无符号整型数据格式，数据将先通过应用一乘法因子后，再进行数据打包传输，以提高数据传输精度，具体请看表 2-3。

表 2-3 数据包中数据部分的数据格式 (16 位无符号整型模式下)

顺序#	数据格式	传感器数据	乘法因子
1	UInt32	时间戳 (400Hz 更新率,即 0.0025s)	无
2	Vector3i16	校准后的陀螺仪数据 (rad/s)	1000
3	Vector3i16	校准后的加速度计数据 (g)	1000



4	Vector3i16	校准后的磁力计数据 (μT)	100
5	Vector3i16	角速度 (rad/s)	1000
6	Vector4i16	四元数 (归一化单位)	10000
7	Vector3i16	欧拉角数据 (rad)	10000
8	Vector3i16	线性加速度数据 (g)	1000

表 2-4 数据格式标识符定义

标识符	描述
UInt32	32 位无符号整型
Int32	32 位带符号整型
Int16	16 位带符号整型
Float32	32 位浮点型
Vector3f	3 元素 32 位浮点型向量
Vector3i16	3 元素 16 位带符号整型向量
Vector4f	4 元素 32 位浮点型向量
Vector4i16	4 元素 16 位带符号整型向量
Matrix3x3f	3x3 浮点型矩阵

传感器数据包中的数据部分，数据类型的传输顺序规定为如表 2-2 和 2-3 中所示，即从顺序#1 到#8 所示的数据类型，最多 8 种。时间戳不能被用户修改，其恒为输出数据。顺序#2 到#8 中的数据，如有任何一数据类型没有被使能输出，则顺延传输下一个被使能的数据，顺序号相对应地往前移。

默认设置下，传感器只输出以下顺序的数据类型（共 7 种）：

1. 时间戳
2. 校准后的陀螺仪数据
3. 校准后的加速度计数据
4. 校准后的磁力计数据
5. 四元数
6. 欧拉角数据
7. 线性加速度数据



2.4 通信例程

以下将列举几个使用 LPBUS 协议的通信例程。

将传感器进入 Command Mode

发送指令（主机→传感器）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	06h	指令号低位字节 (06h = GOTO_COMMAND_MODE)
4	00h	指令号高位字节
5	00h	数据长度低位字节 (GOTO_COMMAND_MODE 指令数据长度为 0)
6	00h	数据长度高位字节
7	07h	校验和低位字节
8	00h	校验和高位字节
9	0Dh	包尾低位字节
10	0Ah	包尾高位字节

数据返回（传感器→主机）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	00h	指令号低位字节 (00h = REPLY_ACK)
4	00h	指令号高位字节
5	00h	数据长度低位字节 (返回 ACK, 无数据)
6	00h	数据长度高位字节
7	01h	校验和低位字节
8	00h	校验和高位字节
9	0Dh	包尾低位字节
10	0Ah	包尾高位字节



将传感器进入 Streaming Mode

发送指令（主机->传感器）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	07h	指令号低位字节 (07h = GOTO_STREAMING_MODE)
4	00h	指令号高位字节
5	00h	数据长度低位字节 (GOTO_STREAMING_MODE 指令数据长度为 0)
6	00h	数据长度高位字节
7	08h	校验和低位字节
8	00h	校验和高位字节
9	0Dh	包尾低位字节
10	0Ah	包尾高位字节

数据返回（传感器->主机）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	00h	指令号低位字节 (00h = REPLY_ACK)
4	00h	指令号高位字节
5	00h	数据长度低位字节 (返回 ACK, 无数据)
6	00h	数据长度高位字节
7	01h	校验和低位字节
8	00h	校验和高位字节
9	0Dh	包尾低位字节
10	0Ah	包尾高位字节



读取传感器配置

发送指令（主机->传感器）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	04h	指令号低位字节 (04h = GET_CONFIG)
4	00h	指令号高位字节
5	00h	数据长度低位字节 (GET 指令数据长度为 0)
6	00h	数据长度高位字节
7	05h	校验和低位字节
8	00h	校验和高位字节
9	0Dh	包尾低位字节
10	0Ah	包尾高位字节

数据返回（传感器->主机）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	04h	指令号低位字节 (04h = GET_CONFIG)
4	00h	指令号高位字节
5	04h	数据长度低位字节 (32 位整型为 4 个字节)
6	00h	数据长度高位字节
7	xxh	配置数据字节 1 (最低位)
8	xxh	配置数据字节 2
9	xxh	配置数据字节 3
10	xxh	配置数据字节 4 (最高位)
11	xxh	校验和低位字节
12	xxh	校验和高位字节
13	0Dh	包尾低位字节
14	0Ah	包尾高位字节

注：xx 的值取决于当前的传感器配置。



读取陀螺仪范围

发送指令（主机->传感器）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	1Ah	指令号低位字节 (1Ah = GET_GYR_RANGE)
4	00h	指令号高位字节
5	00h	数据长度低位字节 (GET_GYR_RANGE 指令数据长度为 0)
6	00h	数据长度高位字节
7	1Bh	校验和低位字节
8	00h	校验和高位字节
9	0Dh	包尾低位字节
10	0Ah	包尾高位字节

数据返回（传感器->主机）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	1Ah	指令号低位字节 (1Ah = GET_GYR_RANGE)
4	00h	指令号高位字节
5	04h	数据长度低位字节 (32 位整型为 4 个字节)
6	00h	数据长度高位字节
7	xxh	配置数据字节 1 (最低位)
8	xxh	配置数据字节 2
9	xxh	配置数据字节 3
10	xxh	配置数据字节 4 (最高位)
11	xxh	校验和低位字节
12	xxh	校验和高位字节
13	0Dh	包尾低位字节
14	0Ah	包尾高位字节

注：xx 的值取决于当前的传感器配置。



设置加速度计范围

发送指令（主机->传感器）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	1Fh	指令号低位字节 (1Fh = SET_ACC_RANGE)
4	00h	指令号高位字节
5	04h	数据长度低位字节 (32 位整型为 4 个字节)
6	00h	数据长度高位字节
7	08h	范围数据字节 1 (范围 8g 为 8d)
8	00h	范围数据字节 2
9	00h	范围数据字节 3
10	00h	范围数据字节 4
11	2Ch	校验和低位字节
12	00h	校验和高位字节
13	0Dh	包尾低位字节
14	0Ah	包尾高位字节

数据返回（传感器->主机）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	00h	指令号低位字节 (00h = REPLY_ACK)
4	00h	指令号高位字节
5	00h	数据长度低位字节 (返回 ACK, 无数据)
6	00h	数据长度高位字节
7	01h	校验和低位字节
8	00h	校验和高位字节
9	0Dh	包尾低位字节
10	0Ah	包尾高位字节



读取传感器数据

发送指令（主机->传感器）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	09h	指令号低位字节 (09h = GET_SENSOR_DATA)
4	00h	指令号高位字节
5	00h	数据长度低位字节 (GET_SENSOR_DATA 指令数据长度为 0)
6	00h	数据长度高位字节
7	0Ah	校验和低位字节
8	00h	校验和高位字节
9	0Dh	包尾低位字节
10	0Ah	包尾高位字节

数据返回（传感器->主机），默认设置下 32-bit 数据格式

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	09h	指令号低位字节 (9d = GET_SENSOR_DATA)
4	00h	指令号高位字节
5	50h	数据长度低位字节 (80 个字节)
6	00h	数据长度高位字节
7-10	xxxxxxxxh	时间戳
11-14	xxxxxxxxh	陀螺仪 x 轴数据
15-18	xxxxxxxxh	陀螺仪 y 轴数据
19-22	xxxxxxxxh	陀螺仪 z 轴数据
23-26	xxxxxxxxh	加速度计 x 轴数据
27-30	xxxxxxxxh	加速度计 y 轴数据
31-34	xxxxxxxxh	加速度计 z 轴数据
35-38	xxxxxxxxh	磁力计 x 轴数据
39-42	xxxxxxxxh	磁力计 y 轴数据
43-46	xxxxxxxxh	磁力计 z 轴数据
47-50	xxxxxxxxh	四元数数据 q0



51-54	xxxxxxxh	四元数数据 q1
55-58	xxxxxxxh	四元数数据 q2
59-62	xxxxxxxh	四元数数据 q3
63-66	xxxxxxxh	欧拉角 x 轴数据
67-70	xxxxxxxh	欧拉角 y 轴数据
71-74	xxxxxxxh	欧拉角 z 轴数据
75-78	xxxxxxxh	线性加速度 x 轴数据
79-82	xxxxxxxh	线性加速度 y 轴数据
83-86	xxxxxxxh	线性加速度 z 轴数据
87	xxh	校验和低位字节
88	xxh	校验和高位字节
89	0Dh	包尾低位字节
90	0Ah	包尾高位字节

注：xx 的值取决于当前的传感器配置和测量值。

若仅使能了加速度计和四元数数据输出，则返回的数据按顺序跳过，见下表。

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	09h	指令号低位字节 (9d = GET_SENSOR_DATA)
4	00h	指令号高位字节
5	20h	数据长度低位字节 (32 个字节)
6	00h	数据长度高位字节
7-10	xxxxxxxh	时间戳
11-14	xxxxxxxh	加速度计 x 轴数据
15-18	xxxxxxxh	加速度计 y 轴数据
19-22	xxxxxxxh	加速度计 z 轴数据
23-26	xxxxxxxh	四元数数据 q0
27-30	xxxxxxxh	四元数数据 q1
31-34	xxxxxxxh	四元数数据 q2
35-38	xxxxxxxh	四元数数据 q3
39	xxh	校验和低位字节
40	xxh	校验和高位字节
41	0Dh	包尾低位字节
42	0Ah	包尾高位字节



保存参数设置

发送指令（主机->传感器）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	0Fh	指令号低位字节 (0Fh = WRITE_REGISTERS command)
4	00h	指令号高位字节
5	00h	数据长度低位字节 (WRITE_REGISTERS 指令数据长度为 0)
6	00h	数据长度高位字节
7	10h	校验和低位字节
8	00h	校验和高位字节
9	0Dh	包尾低位字节
10	0Ah	包尾高位字节

数据返回（传感器->主机）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	00h	指令号低位字节 (00h = REPLY_ACK)
4	00h	指令号高位字节
5	00h	数据长度低位字节 (返回 ACK, 无数据)
6	00h	数据长度高位字节
7	01h	校验和低位字节
8	00h	校验和高位字节
9	0Dh	包尾低位字节
10	0Ah	包尾高位字节

注：保存修改的参数需要一定的时间，该指令发送后不能马上返回数据，需经过 1~2s 才有数据返回。



读取传感器状态

发送指令（主机->传感器）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节（ID = 1）
2	00h	Sensor ID 高位字节
3	05h	指令号低位字节（05h =GET_STATUS）
4	00h	指令号高位字节
5	00h	数据长度低位字节（GET_STATUS 指令数据长度为 0）
6	00h	数据长度高位字节
7	06h	校验和低位字节
8	00h	校验和高位字节
9	0Dh	包尾低位字节
10	0Ah	包尾高位字节

数据返回（传感器->主机）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节（ID = 1）
2	00h	Sensor ID 高位字节
3	05h	指令号低位字节（05h =GET_STATUS）
4	00h	指令号高位字节
5	04h	数据长度低位字节
6	00h	数据长度高位字节
7-10	xxxxxxxh	返回的状态值
11	xxh	校验和低位字节
12	xxh	校验和高位字节
13	0Dh	包尾低位字节
14	0Ah	包尾高位字节

注：返回状态值的解读请参考附录。



陀螺仪校准

发送指令（主机->传感器）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	16h	指令号低位字节 (16h = START_GYR_CALIBRATION)
4	00h	指令号高位字节
5	00h	数据长度低位字节
6	00h	数据长度高位字节
7	17h	校验和低位字节
8	00h	校验和高位字节
9	0Dh	包尾低位字节
10	0Ah	包尾高位字节

数据返回（传感器->主机）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	00h	指令号低位字节 (00h = REPLY_ACK)
4	00h	指令号高位字节
5	00h	数据长度低位字节 (返回 ACK, 无数据)
6	00h	数据长度高位字节
7	01h	校验和低位字节
8	00h	校验和高位字节
9	0Dh	包尾低位字节
10	0Ah	包尾高位字节

注：发送陀螺仪校准指令后返回 ACK 表示校准开始，校准过程保持传感器静止，10s 后校准完成。校准期间可发送读取传感器状态指令查询（返回值的位 3: Gyroscope calibration running，校准过程该位置 1，校准完成后恢复为 0）。磁力计校准指令与此类似，校准过程时间同为 10s。



磁力计校准

发送指令（主机->传感器）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	11h	指令号低位字节 (11h = START_MAG_CALIBRATION)
4	00h	指令号高位字节
5	00h	数据长度低位字节
6	00h	数据长度高位字节
7	12h	校验和低位字节
8	00h	校验和高位字节
9	0Dh	包尾低位字节
10	0Ah	包尾高位字节

数据返回（传感器->主机）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	00h	指令号低位字节 (00h = REPLY_ACK)
4	00h	指令号高位字节
5	00h	数据长度低位字节 (返回 ACK, 无数据)
6	00h	数据长度高位字节
7	01h	校验和低位字节
8	00h	校验和高位字节
9	0Dh	包尾低位字节
10	0Ah	包尾高位字节

注：发送磁力计校准指令后返回 ACK 表示校准开始，校准过程中使传感器绕 x、y、z 轴不停转动，收集周围的磁场信息，持续转动 10s 后校准完成。校准期间可发送读取传感器状态指令查询（返回值的位 4: Magnetometer calibration running，校准过程该位置 1，校准完成后恢复为 0）。磁力计校准完成后，必须发送保存参数指令保存校准数据。



设置波特率

发送指令（主机->传感器）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	54h	指令号低位字节 (84d = SET_UART_BAUDRATE)
4	00h	指令号高位字节
5	04h	数据长度低位字节
6	00h	数据长度高位字节
7	07h	设置波特率为 921600 bps (具体请查看附录中指令 SET_UART_BAUDRATE)
8	00h	
9	00h	
10	00h	
11	60h	校验和低位字节
12	00h	校验和高位字节
13	0Dh	包尾低位字节
14	0Ah	包尾高位字节

数据返回（传感器->主机）

数据包字节号	值	描述
0	3Ah	包头
1	01h	Sensor ID 低位字节 (ID = 1)
2	00h	Sensor ID 高位字节
3	00h	指令号低位字节 (0d = REPLY_ACK)
4	00h	指令号高位字节
5	00h	数据长度低位字节 (返回 ACK, 无数据)
6	00h	数据长度高位字节
7	01h	校验和低位字节
8	00h	校验和高位字节
9	0Dh	包尾低位字节
10	0Ah	包尾高位字节

注：设置波特率完成并保存设置后，需断电重启传感器才有效。



3. 附录 Command List

Summary

Acknowledged / Not-acknowledged Identifiers				
Identifier	Name	Parameter	Response	Default
0(00h)	REPLY_ACK			
1(01h)	REPLY_NACK			

Get Configuration and Status Info Commands				
Identifier	Name	Parameter	Response	Default
4 (04h)	GET_CONFIG	NONE	Int32	
5 (05h)	GET_STATUS ¹	NONE	Int32	

Mode Switching Commands				
Identifier	Name	Parameter	Response	Default
6 (06h)	GOTO_COMMAND_MODE ¹	NONE	ACK/NACK	
7 (07h)	GOTO_STREAM_MODE	NONE	ACK/NACK	

IMU ID Settings Command				
Identifier	Name	Parameter	Response	Default
20 (14h)	SET_IMU_ID	Int32	ACK/NACK	
21 (15h)	GET_IMU_ID	NONE	Int32	1

Gyroscope Settings Command				
Identifier	Name	Parameter	Response	Default
22 (16h)	START_GYR_CALIBRATION	NONE	ACK/NACK	
25 (19h)	SET_GYR_RANGE	Int32	ACK/NACK	
26 (1Ah)	GET_GYR_RANGE	NONE	Int32	2000dps

Accelerometer Settings Command				
Identifier	Name	Parameter	Response	Default
31 (1Fh)	SET_ACC_RANGE	Int32	ACK/NACK	
32 (20h)	GET_ACC_RANGE	NONE	Int32	4g

Magnetometer Settings Command				
-------------------------------	--	--	--	--



Identifier	Name	Parameter	Response	Default
17 (11h)	START_MAG_CALIBRATION ¹	NONE	ACK/NACK	
33 (21h)	SET_MAG_RANGE	Int32	ACK/NACK	
34 (22h)	GET_MAG_RANGE	NONE	Int32	8Gauss

Data Transmission Commands				
Identifier	Name	Parameter	Response	Default
9 (09h)	GET_SENSOR_DATA	NONE		
10 (0Ah)	SET_TRANSMIT_DATA	Int32	ACK/NACK	
11 (0Bh)	SET_STREAM_FREQ	Int32	ACK/NACK	
66 (42h)	SET_TIMESTAMP ¹	Int32	ACK/NACK	
84 (54h)	SET_UART_BAUDRATE	Int32	ACK/NACK	
85 (55h)	GET_UART_BAUDRATE	NONE	Int32	

Register Value Save and Reset Command				
Identifier	Name	Parameter	Response	Default
15 (0Fh)	WRITE_REGISTERS	NONE	ACK/NACK	
16 (10h)	RESTORE_FACTORY_DEFAULTS	NONE	ACK/NACK	

Reference Setting and Offset Reset Command				
Identifier	Name	Parameter	Response	Default
18 (12h)	SET_ORIENTATION_OFFSET	Int32	ACK/NACK	
82 (52h)	RESET_ORIENTATION_OFFSET	NONE	ACK/NACK	

Filter Settings Command				
Identifier	Name	Parameter	Response	Default
41(29h)	SET_FILTER_MODE	Int32	ACK/NACK	
42(2Ah)	GET_FILTER_MODE	NONE	Int32	1
43(2Bh)	SET_FILTER_PRESET	Int32	ACK/NACK	
44(2Ch)	GET_FILTER_PRESET	NONE	Int32	3

Device Info				
Identifier	Name	Parameter	Response	Default
90(5Ah)	GET_SERIAL_NUMBER	NONE	Char[24]	
92(5Ch)	GET_FIRMWARE_INFO	NONE	Char[16]	

¹**Note:** These commands are executable in both streaming mode and command mode. Other commands are executable only when the sensor is in command mode.



Acknowledged and Not-acknowledged Identifiers

Identifier	0
Name	REPLY_ACK
Description	Confirms a successful SET command.

Identifier	1
Name	REPLY_NACK
Description	Reports an error during processing a SET command.

Configuration and Status Commands

Identifier	4 (0x04)
Name	GET_CONFIG
Description	Get the current value of the configuration register of the sensor. The configuration word is read-only. The different parameters are set by their respective SET commands. E.g. SET_TRANSMIT_DATA for defining which data is transmitted from the sensor.
Parameter	NONE
Response:	Int32

Data format	Bit	Reported State / Parameter
	0 - 2	Stream frequency setting (see SET_STREAM_FREQ)
	3 - 8	Reserved
	9	Reserved
	10	Magnetometer data transmission enabled
	11	Accelerometer data transmission enabled
	12	Gyroscope data transmission enabled
	13	Temperature output enabled
	14	Reserved
	15	Reserved
	16	Angular velocity output enabled
	17	Euler angle data transmission enabled
	18	Quaternion orientation output enabled
	19	Reserved
	20	Reserved
	21	Linear acceleration output enabled
	22	16-bit data output mode enabled
	23	Reserved
	24	Magnetometer compensation enabled
	25	Accelerometer compensation enabled
26	Reserved	
27	Reserved	
28	Reserved	
29	Reserved	
30	Gyroscope auto-calibration enabled	
31	Reserved	



Identifier	5 (0x05)																															
Name	GET_STATUS																															
Description	Get the current value of the status register of the sensor. The status word is read-only																															
Parameter	NONE																															
Response:	Int32																															
Data format	<table border="1"> <thead> <tr> <th>Bit</th> <th>Indicated state</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>COMMAND mode enabled</td> </tr> <tr> <td>1</td> <td>STREAM mode enabled</td> </tr> <tr> <td>2</td> <td>Reserved</td> </tr> <tr> <td>3</td> <td>Gyroscope calibration running</td> </tr> <tr> <td>4</td> <td>Magnetometer calibration running</td> </tr> <tr> <td>5</td> <td>Gyroscope initialization failed</td> </tr> <tr> <td>6</td> <td>Accelerometer initialization failed</td> </tr> <tr> <td>7</td> <td>Magnetometer initialization failed</td> </tr> <tr> <td>8</td> <td>Reserved</td> </tr> <tr> <td>9</td> <td>Gyroscope unresponsive</td> </tr> <tr> <td>10</td> <td>Accelerometer unresponsive</td> </tr> <tr> <td>11</td> <td>Magnetometer unresponsive</td> </tr> <tr> <td>12</td> <td>Flash write failed</td> </tr> <tr> <td>13-31</td> <td>Reserved</td> </tr> </tbody> </table>		Bit	Indicated state	0	COMMAND mode enabled	1	STREAM mode enabled	2	Reserved	3	Gyroscope calibration running	4	Magnetometer calibration running	5	Gyroscope initialization failed	6	Accelerometer initialization failed	7	Magnetometer initialization failed	8	Reserved	9	Gyroscope unresponsive	10	Accelerometer unresponsive	11	Magnetometer unresponsive	12	Flash write failed	13-31	Reserved
Bit	Indicated state																															
0	COMMAND mode enabled																															
1	STREAM mode enabled																															
2	Reserved																															
3	Gyroscope calibration running																															
4	Magnetometer calibration running																															
5	Gyroscope initialization failed																															
6	Accelerometer initialization failed																															
7	Magnetometer initialization failed																															
8	Reserved																															
9	Gyroscope unresponsive																															
10	Accelerometer unresponsive																															
11	Magnetometer unresponsive																															
12	Flash write failed																															
13-31	Reserved																															

Mode Switching Commands

Identifier	6 (0x06)
Name	GOTO_COMMAND_MODE
Description	Switch to command mode. In command mode the user can issue commands to the firmware to perform calibration, set parameters etc.
Parameter	NONE
Response:	ACK (success) or NACK (error)

Identifier	7 (0x07)
Name	GOTO_STREAM_MODE
Description	Switch to streaming mode. In this mode data is continuously streamed from the sensor, and some commands cannot be performed until the sensor receives the GOTO_COMMAND_MODE command.
Parameter	NONE
Response:	ACK (success) or NACK (error)

IMU ID Setting Command

Identifier	20 (0x14)
Name	SET_IMU_ID
Description	Set sensor ID
Parameter	Int32
Response:	ACK (success) or NACK (error)



Identifier	21 (0x15)
Name	GET_IMU_ID
Description	Get sensor ID
Parameter	None
Response:	Int32

Gyroscope Settings Command

Identifier	22 (0x16)
Name	START_GYR_CALIBRATION
Description	Start the calibration of the gyroscope sensor
Parameter	NONE
Response:	ACK (success) or NACK (error)

Identifier	25 (0x19)												
Name	SET_GYR_RANGE												
Description	Set the current range of the gyroscope												
Parameter	Int32 <table border="1"> <thead> <tr> <th>Range (deg/s)</th> <th>Identifier</th> </tr> </thead> <tbody> <tr> <td>125</td> <td>125</td> </tr> <tr> <td>245</td> <td>245</td> </tr> <tr> <td>500</td> <td>500</td> </tr> <tr> <td>1000</td> <td>1000</td> </tr> <tr> <td>2000</td> <td>2000</td> </tr> </tbody> </table>	Range (deg/s)	Identifier	125	125	245	245	500	500	1000	1000	2000	2000
Range (deg/s)	Identifier												
125	125												
245	245												
500	500												
1000	1000												
2000	2000												
Response:	ACK (success) or NACK (error)												

Identifier	26 (0x1A)												
Name	GET_GYR_RANGE												
Description	Get current gyroscope range.												
Parameter	NONE												
Response:	Int32 <table border="1"> <thead> <tr> <th>Range (deg/s)</th> <th>Identifier</th> </tr> </thead> <tbody> <tr> <td>125</td> <td>125</td> </tr> <tr> <td>245</td> <td>245</td> </tr> <tr> <td>500</td> <td>500</td> </tr> <tr> <td>1000</td> <td>1000</td> </tr> <tr> <td>2000</td> <td>2000</td> </tr> </tbody> </table>	Range (deg/s)	Identifier	125	125	245	245	500	500	1000	1000	2000	2000
Range (deg/s)	Identifier												
125	125												
245	245												
500	500												
1000	1000												
2000	2000												

**Accelerometer Settings Command**

Identifier	31 (0x1F)										
Name	SET_ACC_RANGE										
Description	Set the current range of the accelerometer										
Parameter	Int32 <table border="1"><thead><tr><th>Range</th><th>Identifier</th></tr></thead><tbody><tr><td>2g</td><td>2</td></tr><tr><td>4g</td><td>4</td></tr><tr><td>8g</td><td>8</td></tr><tr><td>16g</td><td>16</td></tr></tbody></table>	Range	Identifier	2g	2	4g	4	8g	8	16g	16
Range	Identifier										
2g	2										
4g	4										
8g	8										
16g	16										
Response:	ACK (success) or NACK (error)										

Identifier	32 (0x20)										
Name	GET_ACC_RANGE										
Description	Get the current range of the accelerometer										
Parameter	NONE										
Response:	Int32 <table border="1"><thead><tr><th>Range</th><th>Identifier</th></tr></thead><tbody><tr><td>2g</td><td>2</td></tr><tr><td>4g</td><td>4</td></tr><tr><td>8g</td><td>8</td></tr><tr><td>16g</td><td>16</td></tr></tbody></table>	Range	Identifier	2g	2	4g	4	8g	8	16g	16
Range	Identifier										
2g	2										
4g	4										
8g	8										
16g	16										

Magnetometer Settings Command

Identifier	17 (0x11)
Name	START_MAG_CALIBRATION
Description	Start the calibration of the magnetometer sensor
Parameter	NONE
Response:	ACK (success) or NACK (error)

Identifier	33 (0x21)										
Name	SET_MAG_RANGE										
Description	Set the current range of the gyroscope										
Parameter	Int32 <table border="1"><thead><tr><th>Range</th><th>Identifier</th></tr></thead><tbody><tr><td>4 Gauss</td><td>4</td></tr><tr><td>8 Gauss</td><td>6</td></tr><tr><td>12 Gauss</td><td>12</td></tr><tr><td>16 Gauss</td><td>16</td></tr></tbody></table>	Range	Identifier	4 Gauss	4	8 Gauss	6	12 Gauss	12	16 Gauss	16
Range	Identifier										
4 Gauss	4										
8 Gauss	6										
12 Gauss	12										
16 Gauss	16										
Response:	ACK (success) or NACK (error)										



Identifier	34 (0x22)	
Name	GET_MAG_RANGE	
Description	Get current magnetometer range.	
Parameter	NONE	
Response:	Int32	
	Range	Identifier
	4 Gauss	4
	8 Gauss	6
	12 Gauss	12
	16 Gauss	16

Data Transmission Commands

Identifier	9 (0x09)
Name	GET_SENSOR_DATA
Description	Retrieves the latest set of sensor data. A data packet will be composed as defined by SET_TRANSMIT_DATA. The currently set format can be retrieved with the sensor configuration word.
Parameter	NONE
Response:	See the LPBUS protocol explanation for a description of the measurement data format.

Identifier	10 (0x0A)	
Name	SET_TRANSMIT_DATA	
Description	Set the current transmit data	
Parameter	Int32	
	Bit	Reported State / Parameter
	0 - 9	0
	10	Magnetometer data transmission enabled
	11	Accelerometer data transmission enabled
	12	Gyroscope data transmission enabled
	13	Temperature output enabled
	14	0
	15	0
	16	Angular velocity output enabled
	17	Euler angle data transmission enabled
	18	Quaternion orientation output enabled
	19	0
	20	0
	21	Linear acceleration output enabled
	22	16-bit data output mode enabled
	23	0
24	Magnetometer compensation enabled	
25	Accelerometer compensation enabled	
26-31	0	
Response:	ACK (success) or NACK (error)	



Identifier	11 (0x0B)		
Name	SET_STREAM_FREQ		
Description	Set the current streaming frequency		
Parameter	Int32		
	Frequency (Hz)	Identifier	Bit : 0~2 (GET_CONFIG return data)
	5	5	000
	10	10	001
	25	25	010
	50	50	011
	100	100	100
	200	200	101
400	400	110	
Response:	ACK (success) or NACK (error)		

Identifier	66 (0x42)		
Name	SET_TIMESTAMP		
Description	Set the current sensor timestamp counter. Counter updates at 400Hz, i.e. setting timestamp counter equates to setting the timestamp to 1s.		
Parameter	Int32		
Response:	ACK (success) or NACK (error)		

Identifier	84 (0x54)		
Name	SET_UART_BAUDRATE		
Description	Set the current UART baudrate		
Parameter	Int32		
	Baud rate	Identifier	
	19200	0	
	38400	1	
	57600	2	
	115200	3	
	230400	4	
	256000	5	
460800	6		
921600	7		
Response:	ACK (success) or NACK (error)		

Identifier	85 (0x55)		
Name	GET_UART_BAUDRATE		
Description	Get the current UART baudrate		
Parameter	NONE		



Response:	Int32	
	Baud rate	Identifier
	19200	0
	38400	1
	57600	2
	115200	3
	230400	4
	256000	5
	460800	6
921600	7	

Register Value Save and Reset Command

Identifier	15 (0x0F)
Name	WRITE_REGISTERS
Description	Write the currently set parameters to flash memory.
Parameter	NONE
Response:	ACK (success) or NACK (error)

Identifier	16 (0x10)
Name	RESTORE_FACTORY_DEFAULTS
Description	Reset the LPMS parameters to factory default values. Please note that upon issuing this command your currently set parameters will be erased.
Parameter	NONE
Response:	ACK (success) or NACK (error)

Reference Setting and Offset Reset Command

Identifier	18 (0x12)						
Name	SET_OFFSET						
Description	Sets the orientation offset using one of the three offset methods.						
Parameter	Int32						
	<table border="1"> <thead> <tr> <th>Mode</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>Object reset</td> <td>0</td> </tr> <tr> <td>Heading reset</td> <td>1</td> </tr> </tbody> </table>	Mode	Value	Object reset	0	Heading reset	1
	Mode	Value					
Object reset	0						
Heading reset	1						
Response:	ACK (success) or NACK (error)						

Identifier	82 (0x52)
Name	RESET_ORIENTATION_OFFSET
Description	Reset the orientation offset to 0 (unity quaternion).
Parameter	NONE
Response:	ACK (success) or NACK (error)



Filter Settings Command

Identifier	41 (0x29)	
Name	SET_FILTER_MODE	
Description	Set the sensor filter mode	
Parameter	Int32	
	Mode	Value
	Gyroscope only	0
	Accelerometer + gyroscope (Kalman filter)	1
	Accelerometer+ gyroscope+ magnetometer (Kalman filter)	2
	Accelerometer + gyroscope (DCM filter)	3
Accelerometer + gyroscope + Magnetometer (DCM filter)	4	
Response:	ACK (success) or NACK (error)	

Identifier	42 (0x2A)	
Name	GET_FILTER_MODE	
Description	Get the sensor filter mode	
Parameter	NONE	
Response:	Int32	
	Mode	Value
	Gyroscope only	0
	Accelerometer + gyroscope (Kalman filter)	1
	Accelerometer+ gyroscope+ magnetometer (Kalman filter)	2
	Accelerometer + gyroscope (DCM filter)	3
Accelerometer + gyroscope + Magnetometer (DCM filter)	4	

Identifier	43 (0x2B)	
Name	SET_FILTER_PRESET	
Description	Set one of the filter parameter presets for accelerometer and magnetometer covariance strength	
Parameter	Int32	
	Correction strength	Value
	Weak	0
	Medium	1
	Strong	2
Dynamic	3	
Response:	ACK (success) or NACK (error)	



Identifier	44 (0x2C)	
Name	GET_FILTER_PRESET	
Description	Get current filter preset	
Parameter	NONE	
Response:	Int32	
	Correction strength	Value
	Dynamic	0
	Strong	1
	Medium	2
	Weak	3

Device Info

Identifier	90 (0x5A)	
Name	GET_SERIAL_NUMBER	
Description	Get sensor serial number	
Parameter	NONE	
Response:	Char[24] Character array of length 24	

Identifier	92 (0x5C)	
Name	GET_FIRMARE_INFO	
Description	Get firmware info	
Parameter	NONE	
Response:	Char[16] Firmware name - version	



广州阿路比电子科技有限公司——版权所有——2019

Guangzhou Alubi Electronic Technology Co.,Ltd.

<http://www.alubi.cn>