



文档编号:01.13.006401

YDLIDAR GS5

开发手册



www.ydlidar.cn

目录

| | |
|---------------------------|----|
| 1 工作机制 | 1 |
| 1.1 模式 | 1 |
| 1.2 测距原理 | 1 |
| 2 系统通信 | 2 |
| 2.1 通信机制 | 2 |
| 2.2 系统命令 | 2 |
| 2.3 系统报文 | 3 |
| 3 数据协议 | 4 |
| 3.1 获取版本信息命令 | 4 |
| 3.2 获取版本信息命令 | 4 |
| 3.3 获取设备参数命令 | 5 |
| 3.4 开启扫描命令 | 6 |
| 3.5 停止扫描命令 | 6 |
| 3.6 设置波特率命令 | 7 |
| 3.7 系统复位命令 | 7 |
| 4 数据解析 | 8 |
| 5 OTA 升级 | 11 |
| 5.1 升级流程 | 11 |
| 5.2 送协议 | 11 |
| 5.3 Start_IAP 指令 | 12 |
| 5.4 Running_IAP 指令 | 13 |
| 5.5 Complete_IAP 指令 | 14 |
| 5.6 RESET_SYSTEM 指令 | 15 |
| 5.7 OTA 升级 Q&A | 15 |
| 6 使用注意 | 15 |
| 7 修订 | 17 |

1 工作机制

1.1 模式

YDLIDAR GS5（以下简称 GS5）的系统设置了 2 种工作模式：空闲模式、扫描模式。

- **空闲模式：**GS5 上电时，默认为空闲模式，空闲模式时，GS5 的测距单元不工作，激光器不亮。
- **扫描模式：**当 GS5 进入扫描模式时，测距单元点亮激光器，开始工作，不断对外部环境进行激光采样，并经过后台处理后实时输出。

1.2 测距原理

GS5 是一款近距离固态雷达，测距范围为 70-1000mm。主要由一字线激光器和摄像头构成。一字线激光器发出激光后，被摄像头捕捉到，根据激光器与摄像头的固定结构，结合三角测距原理，我们便可算出物体到 GS5 的距离。再根据摄像头标定后的参数，结合机械参数可以知道被测物体在雷达坐标系中的角度值。由此，我们便获取到了被测物体完整的测量数据。

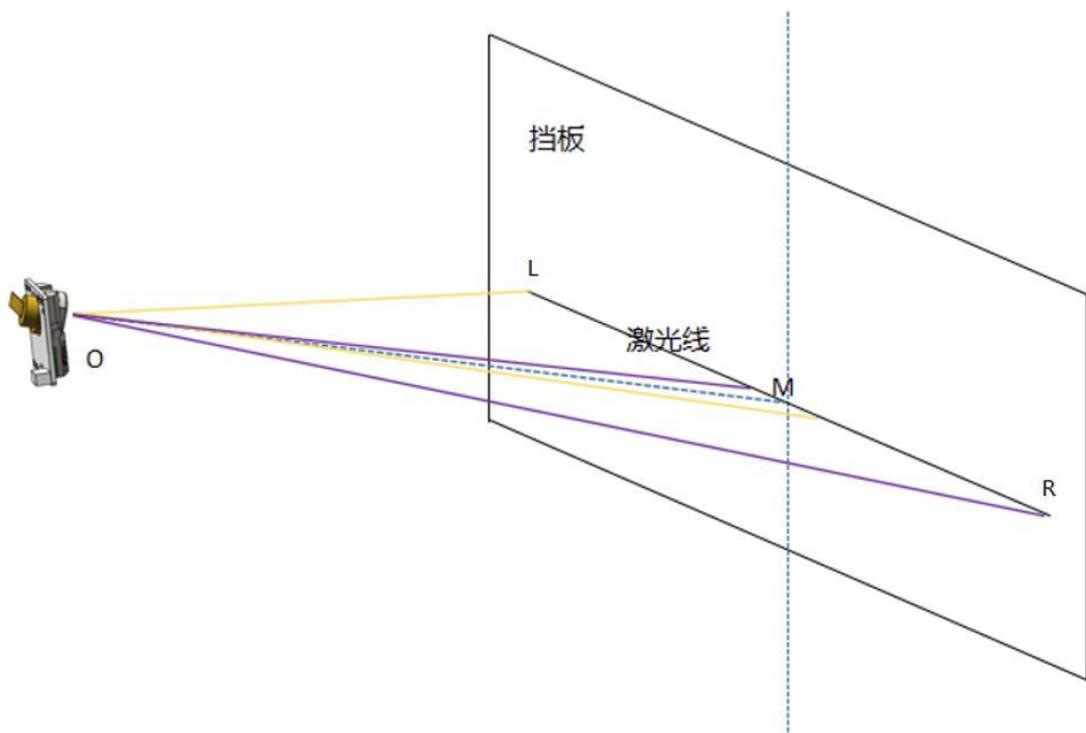


图 1 YDLIDAR GS5 测量示意图

O 为坐标原点，橙色区域为左（L）相机的视角，紫色区域为右（R）相机的视角。

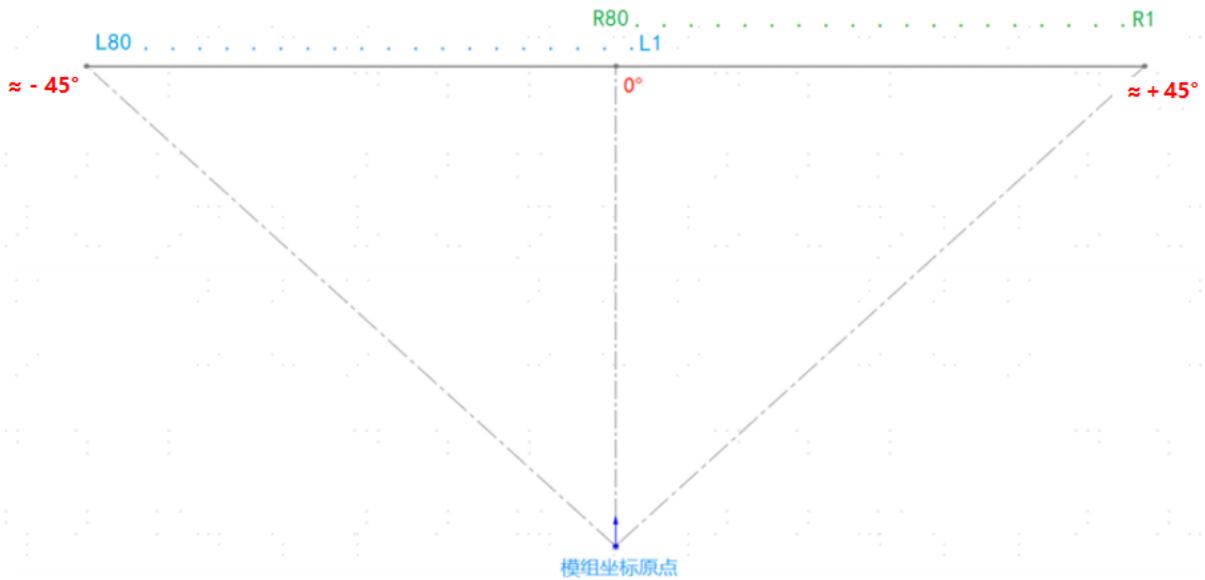


图 2 YDLIDAR GS5 输出点云角度示意图

以模组标点为坐标原点，正前方为坐标系 0° 方向，角度顺时针增加，点云输出时数据 (S1~S160) 排列的顺序为 L1~L80, R1~R80，使用 SDK 计算出的角度和距离均表示在该坐标系中。

2 系统通信

2.1 通信机制

GS5 是通过串口来和外部设备进行命令和数据的交互。当外部设备发送一个系统命令至 GS5，GS5 解析系统命令，会返回相应的应答报文，并根据命令内容，来切换相应的工作状态，外部系统根据报文内容，解析报文，便可获取应答数据。



图 3 YDLIDAR GS5 系统通信机制

2.2 系统命令

外部系统通过发送相关的系统命令，便可设置 GS5 相应的工作状态，获取相应的数据。GS5 对外发布的系统命令如下：

表 1 YDLIDAR GS5 系统命令

| 命令 | 描述 | 模式切换 | 应答模式 |
|------|-------------|------|------|
| 0x60 | 获取设备地址 | 停机模式 | 单次应答 |
| 0x61 | 获取计算参数 | 停机模式 | 单次应答 |
| 0x62 | 获取版本信息 | 停机模式 | 单次应答 |
| 0x63 | 开始扫描，输出点云数据 | 扫描模式 | 持续应答 |
| 0x64 | 停机，停止扫描 | 停机模式 | 单次应答 |
| 0x67 | 设备软重启 | / | 单次应答 |
| 0x68 | 设置串口波特率 | 停机模式 | 单次应答 |

2.3 系统报文

系统报文是系统根据接收的系统命令反馈的应答报文，不同的系统命令，系统报文的应答模式和应答内容也不一样，其中应答模式有四种：无应答、单次应答、多次应答和持续应答。

无应答表示系统不反馈任何报文；单次应答表示系统的报文长度是有限的，应答一次即结束；当系统级联了多个 GS5 时，某些命令的发送会连续收到 GS5 设备的应答；持续应答表示系统的报文长度是无限长的，需要持续发送数据，如进入扫描模式时。

单次应答、多次应答和持续应答的报文采用同一个数据协议，其协议内容为：包头、设备地址、包类型、数据长度、数据段和校验码，通过串口 16 进制输出。

表 2 YDLIDAR GS5 系统报文数据协议

| 包头 | 设备地址 | 包类型 | 数据长度 | 数据段 | 校验码 |
|---------|--------|--------|---------|---------|--------|
| 4 Bytes | 1 Byte | 1 Byte | 2 Bytes | N Bytes | 1 Byte |

字节偏移：

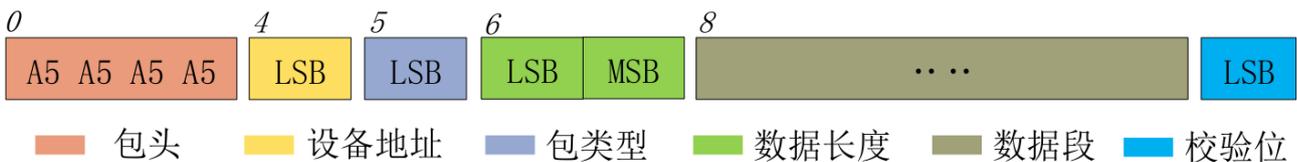


图 4 YDLIDAR GS5 系统报文数据协议示意图

- **包头：**GS5 的报文包头标志为 0xA5A5A5A5；
- **设备地址：**GS5 的设备地址，预留该位，目前固定为 0x01，0x02，0x04；
- **包类型：**系统命令的类型，见表（1）；
- **数据长度：**表示的是应答数据的长度；
- **数据段：**不同系统命令下的应答内容，反馈不同的数据内容，其数据格式也不同；
- **校验码：**校验码。

注：GS5 的数据通信采用的是小端模式，低位在前。

3 数据协议

3.1 获取版本信息命令

当外部设备向 GS5 发送该命令时，GS5 会返回其版本信息。报文为：

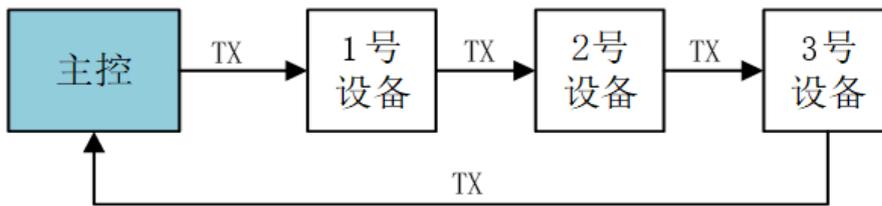
命令发送：

| 包头 | | | | 地址 | 命令类型 | 数据长度 | | 校验码 |
|------|------|------|------|------|------|------|------|------|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | 0x00 | 0x60 | 0x00 | 0x00 | 0x60 |

命令接收：

| 包头 | | | | 地址 | 命令类型 | 数据长度 | | 校验码 |
|------|------|------|------|------|------|------|------|-----|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | 0x0* | 0x60 | 0x00 | 0x00 | LSB |

在级联的情况下，如果串接了 N 个（最大支持 3 个）设备，该命令只返回单次应答，应答中不同的地址代表不同的设备个数。地址：0x01，0x02，0x04，分别对应 1-3 个模组。



级联地址分配：一号模块地址为 0x01，二号模块地址 0x02，三号模块地址 0x04

3.2 获取版本信息命令

当外部设备向 GS5 发送该命令时，GS5 会返回其设备参数，报文为：

命令发送：

| 包头 | | | | 地址 | 命令类型 | 数据长度 | | 校验码 |
|------|------|------|------|------|------|------|------|------|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | 0x00 | 0x62 | 0x00 | 0x00 | 0x62 |

命令接收：

| 包头 | | | | 地址 | 命令类型 | 数据长度 | | 数据段 | | 校验码 |
|------|------|------|------|------|------|------|------|-----|------|-----|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | 0x0* | 0x62 | 0x13 | 0x00 | 版本号 | SN 号 | LSB |

在级联的情况下，如果串接了 N 个（最大支持 3 个）设备，该命令返回 N 个应答，分别回应每个设备的版本信息。

版本号共 3 个字节长度，SN 号共 16 字节长度。

3.3 获取设备参数命令

当外部设备向 GS5 发送该命令时，GS5 会返回其设备参数，报文为：

命令发送：

| 包头 | | | | 地址 | 命令类型 | 数据长度 | | 校验码 |
|------|------|------|------|------|------|------|------|------|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | 0x00 | 0x61 | 0x00 | 0x00 | 0x61 |

命令接收：

| 包头 | | | | 地址 | 命令类型 | 数据长度 | 数据段 | 校验码 |
|------|------|------|------|------|------|-----------|------|-----|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | 0x0* | 0x61 | 0x09 0x00 | 设备参数 | LSB |

数据段包括对应设备的参数，分布如下：

| K0 | | B0 | | K1 | | B1 | | Bias |
|-----|-----|-----|-----|-----|-----|-----|-----|------|
| LSB | MSB | LSB | MSB | LSB | MSB | LSB | MSB | LSB |

在级联的情况下，如果串接了 N 个（最大支持 3 个）设备，该命令将返回 N 个应答，分别对应每个设备的参数。

协议接收到的 K、B 为 uint16 型，在代入计算函数前需要转换成 float 型再除以 10000。

`d_compensateK0 = (float)K0/10000.0f;`

`d_compensateB0 = (float)B0/10000.0f;`

`d_compensateK1 = (float)K1/10000.0f;`

`d_compensateB1 = (float)B1/10000.0f;`

Bias 为 int8 型，代入计算函数前需要转换成 float 型再除以 10。

`bias = (float)Bias /10;`

3.4 开启扫描命令

当外部设备向 GS5 发送扫描命令时，GS5 会进入扫描模式，并持续反馈点云数据。报文为：

命令发送：（发送地址 0x00，无论是否级联，将启动所有设备）

| 包头 | | | | 地址 | 命令类型 | 数据长度 | | 校验码 |
|------|------|------|------|------|------|------|------|------|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | 0x00 | 0x63 | 0x00 | 0x00 | 0x63 |

命令接收：（级联的情况下，该命令仅返回一个应答，地址为最大地址，例如：级联了 3 个设备，地址为 0x04。）

| 包头 | | | | 地址 | 命令类型 | 数据长度 | | 校验码 |
|------|------|------|------|------|------|------|------|-----|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | 0x0* | 0x63 | 0x00 | 0x00 | LSB |

接下来，GS5 会持续输出点云数据序列，帧结构如下：

| 包头 | | | | 地址 | 命令类型 | 数据长度 | 数据段 | 校验码 |
|------|------|------|------|------|------|-----------|------|-----|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | 0x0* | 0x63 | 0xB2 0x01 | 点云数据 | LSB |

数据段为系统扫描的点云数据，其按照以下数据结构，以 16 进制向串口发送至外部设备。整包数据长度为 322 Bytes，包含 2 Bytes 的环境数据和 160 个测距点（S1-S160），其中每个测距点为 2 Bytes，高 5 位为强度数据，低 11 位为距离数据，单位 mm。

| ENV | | S1 | | S2 | | ... | | S160 | |
|-----|-----|-----|-----|-----|-----|-----|-----|------|--|
| LSB | MSB | LSB | MSB | LSB | MSB | ... | LSB | MSB | |

图 5 扫描命令应答内容数据结构示意图

3.5 停止扫描命令

当系统处于扫描状态时，GS5 一直在对外发送点云数据，若此时需要关闭扫描，可以发送此命令，令系统停止扫描。发送停止命令后，模组会回复应答指令，系统随即进入待机休眠状态。此时，设备的测距单元处于低功耗模式，激光器关闭。

命令发送：（发送地址 0x00，无论是否级联，将关闭所有设备）

| 包头 | | | | 地址 | 命令类型 | 数据长度 | | 校验码 |
|------|------|------|------|------|------|------|------|------|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | 0x00 | 0x64 | 0x00 | 0x00 | 0x64 |

命令接收:

| 包头 | | | | 地址 | 命令类型 | 数据长度 | | 校验码 |
|------|------|------|------|------|------|------|------|-----|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | 0x0* | 0x64 | 0x00 | 0x00 | LSB |

在级联的情况下，如果串接了 N 个（最大支持 3 个）设备，该命令仅返回一个应答，地址为最大地址，例如：级联了 3 个设备，地址为 0x04。

3.6 设置波特率命令

当外部设备向 GS5 发送该命令时，可设置 GS5 的输出波特率。

命令发送：（发送地址 0x00，仅支持将所有级联设备的波特率设置成一样），报文为：

| 包头 | | | | 地址 | 命令类型 | 数据长度 | 数据段 | 校验码 |
|------|------|------|------|------|------|-----------|--------------|-----|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | 0x00 | 0x68 | 0x01 0x00 | 波特率（1 Bytes） | LSB |

命令接收:

| 包头 | | | | 地址 | 命令类型 | 数据长度 | 数据段 | 校验码 |
|------|------|------|------|------|------|-----------|--------------|-----|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | 0x0* | 0x68 | 0x01 0x00 | 波特率（1 Bytes） | LSB |

其中，数据段为波特率参数，包含四种波特率（bps），分别为：230400、512000、921600、1500000 分别对应代号 0-3（注意：三模块串联波特率需 ≥ 921600 ，默认为 921600）。

在级联的情况下，如果串接了 N 个（最大支持 3 个）设备，该命令仅返回一个应答，地址为最大地址，例如：级联了 3 个设备，地址为 0x04。

设置波特率后，需要软重启设备。

3.7 系统复位命令

当外部设备向 GS5 发送该命令时，GS5 会进入软重启，系统复位并重新启动。

命令发送：（发送地址 Address，只能是确切的级联地址：0x01/0x02/0x04）

| 包头 | | | | 地址 | 命令类型 | 数据长度 | 校验码 |
|------|------|------|------|---------|------|-----------|-----|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | Address | 0x67 | 0x00 0x00 | LSB |

命令接收:

| 包头 | | | | 地址 | 命令类型 | 数据长度 | 校验码 |
|------|------|------|------|---------|------|-----------|-----|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | Address | 0x67 | 0x00 0x00 | LSB |

Address 为级联链路中需要配置的模组的地址：

复位 1 号模组： Address =0x01

复位 2 号模组： Address =0x02

复位 3 号模组： Address =0x04

4 数据解析

表 3 数据结构描述

| 内容 | 名称 | 描述 |
|---------|------|-------------------------------------|
| K0(2B) | 设备参数 | (uint16) 相机角度参数 k0 系数 (见 3.3 小节) |
| B0(2B) | 设备参数 | (uint16) 相机角度参数 b0 系数 (见 3.3 小节) |
| K1(2B) | 设备参数 | (uint16) 相机角度参数 k1 系数 (见 3.3 小节) |
| B1(2B) | 设备参数 | (uint16) 相机角度参数 b1 系数 (见 3.3 小节) |
| BIAS | 设备参数 | (unit8) 当前相机角度参数 bias 系数 (见 3.3 小节) |
| ENV(2B) | 环境数据 | 环境光强度 |
| Si(2B) | 测距数据 | 低 11 位为距离，高 5 位为强度值 |

➤ 距离解析

距离解算公式： $Distance_i = (Si_MSB \ll 8 | Si_LSB) \& 0x01ff$ ，单位为 mm。

强度计算： $Quality_i = Si_MSB \gg 1$

➤ 角度解析

将激光的出射方向作为传感器的前方，将激光圆心在 PCB 平面的投影作为坐标原点，以 PCB 平面法线为 0 度方向建立极坐标系。顺时针方向，角度逐渐增大。

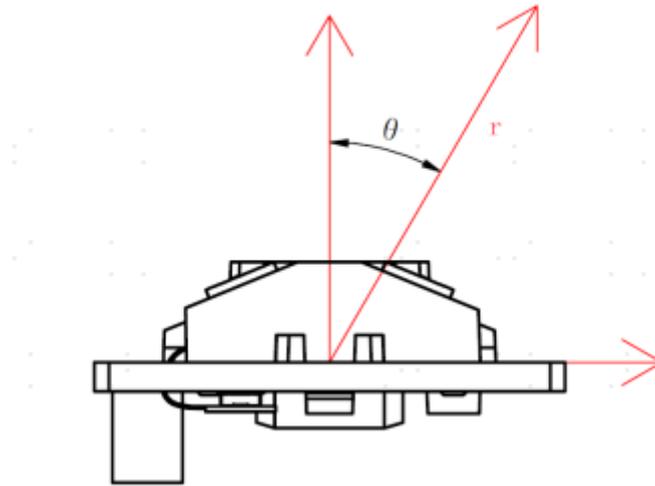


图 6 YDLIDAR GS5 坐标系示意图 (GS5 俯视图)

雷达传输的原始数据转换到上图坐标系中，需要进行一系列的运算，转换函数如下（详细请参考 SDK）：

```
double pixelU = n, Dist, theta, tempTheta, tempDist, tempX, tempY;
if (n < 80)//区分左右相机数据
{
    pixelU = 80 - pixelU;
    if (d_compensateB0 > 1) {
        tempTheta = d_compensateK0 * pixelU - d_compensateB0;
    }
    else
    {
        tempTheta = atan(d_compensateK0 * pixelU - d_compensateB0) * 180 / M_PI;
    }
    tempDist = (dist - angle_p_x) / cos((angle_p_angle + bias - (tempTheta)) * M_PI / 180);
    tempTheta = tempTheta * M_PI / 180;
    tempX = cos((angle_p_angle + bias) * M_PI / 180) * tempDist * cos(tempTheta) +
            sin((angle_p_angle + bias) * M_PI / 180) * (tempDist * sin(tempTheta));
    tempY = -sin((angle_p_angle + bias) * M_PI / 180) * tempDist * cos(tempTheta) +
            cos((angle_p_angle + bias) * M_PI / 180) * (tempDist * sin(tempTheta));
    tempX = tempX + angle_p_x;
    tempY = tempY - angle_p_y;
    Dist = sqrt(tempX * tempX + tempY * tempY);
    theta = atan(tempY / tempX) * 180 / M_PI;
}
else
{
    pixelU = 160 - pixelU;
    if (d_compensateB1 > 1)
    {
        tempTheta = d_compensateK1 * pixelU - d_compensateB1;
    }
    else
    {
        tempTheta = atan(d_compensateK1 * pixelU - d_compensateB1) * 180 / M_PI;
```

```

}
tempDist = (dist - angle_p_x) / cos((angle_p_angle + bias + (tempTheta)) * M_PI / 180);
tempTheta = tempTheta * M_PI / 180;
tempX = cos(-(angle_p_angle + bias) * M_PI / 180) * tempDist * cos(tempTheta) + sin(-
(angle_p_angle + bias) * M_PI / 180) * (tempDist * sin(tempTheta));
tempY = -sin(-(angle_p_angle + bias) * M_PI / 180) * tempDist * cos(tempTheta) + cos(-
(angle_p_angle + bias) * M_PI / 180) * (tempDist * sin(tempTheta));
tempX = tempX + angle_p_x;
tempY = tempY + angle_p_y;
Dist = sqrt(tempX * tempX + tempY * tempY);
theta = atan(tempY / tempX) * 180 / M_PI;
}
if (theta < 0)
{
theta += 360;
}
*dstTheta = theta;
*dstDist = Dist;

```

➤ 校验码解析

校验码采用单字节累加对当前数据包进行校验，四个字节包头和校验码本身不参与校验运算，校验码解算公式为：

$$Checksum = ADD_1^{end}(C_i) \quad i = 1, 2, \dots, end$$

ADD_1^{end} 为累加公式，表示将元素中从下标 1 到 end 的数进行累加。

| | |
|-----------|-------|
| 设备地址 | C_1 |
| 命令类型 | C_2 |
| 数据长度 LSB | C_3 |
| 数据长度 MSB | C_4 |
| 数据段 LSB 0 | C_5 |
| 数据段 LSB 1 | C_6 |
| ... | ... |
| 数据段 MSB N | C_end |

图 7 校验累加示意图

5 OTA 升级

5.1 升级流程

级联升级前置条件：使用0x60指令对三个模组都分配过地址。避免地址冲突导致升级失败。
升级过程中注意：每条指令均有应答，需确认应答报文及其含义；确认后即可进行下一步。

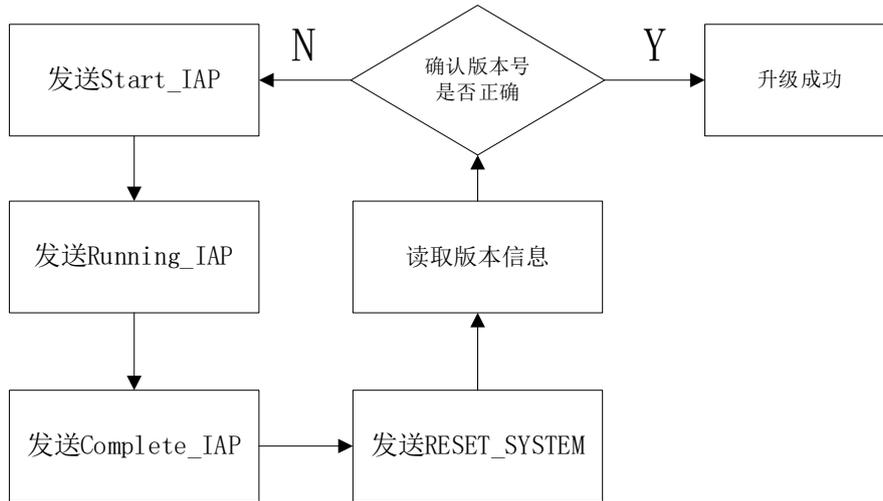


图 8 升级流程图

5.2 送协议

表 4 OTA 数据协议格式（小端）

| 参数 | 长度 (BYTE) | 描述 |
|----------------|-----------|-------------------|
| Packet_Header | 4 | 数据包头，固定为 A5A5A5A5 |
| Device_Address | 1 | 指定设备的地址 |
| Pack_ID | 1 | 数据包 ID（数据类型） |
| Data_Len | 2 | Data 数据段的长度，0-82 |
| Data | n | 数据，n = Data_Len |
| Check_Sum | 1 | 校验和，除去包头后的其余字节校验和 |

| | | | | | | | | |
|---------------|------|------|------|---------|---------|-----------|---------|-----------|
| Packet_Header | | | | Address | Pack_ID | Data_Len | Data | Check_Sum |
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | LSB | LSB | LSB MSB | n Bytes | LSB |

表 5 OTA 升级指令

| 指令类型 | Pack_ID | 描述 |
|--------------|---------|-----------------|
| Start_IAP | 0x0A | 上电后发送该指令以启动 IAP |
| Running_IAP | 0x0B | 运行 IAP, 传输数据包 |
| Complete_IAP | 0x0C | IAP 结束 |
| ACK_IAP | 0x20 | IAP 应答 |
| RESET_SYSTEM | 0x67 | 复位重启指定地址的模组 |

5.3 Start_IAP 指令

命令发送:

| Packet_Header | | | | Address | Pack_ID | Data_Len | | Data | Check_Sum |
|---------------|------|------|------|---------|---------|----------|------|----------|-----------|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | LSB | 0x0A | 0x12 | 0x00 | 18 Bytes | LSB |

数据段 Data 格式:

Data[0~1]: 默认为 0x00;

DATA[2~17]: 为固定的字符验证码:

0x73 0x74 0x61 0x72 0x74 0x20 0x64 0x6F 0x77 0x6E 0x6C 0x6F 0x61 0x64 0x00
0x00

参考发送报文:

A5 A5 A5 A5 01 0A 12 00 00 00 73 74 61 72 74 20 64 6F 77 6E 6C 6F 61 64 00 00
C3

命令接收: (由于存在 FLASH 扇区操作, 此返回延时较大且存在波动, 约在 80ms~700ms 之间)

| Packet_Header | | | | Address | ACK | Data_Len | | Data | Check_Sum |
|---------------|------|------|------|---------|------|----------|------|---------|-----------|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | LSB | 0x20 | 0x04 | 0x00 | 4 Bytes | LSB |

接收数据格式:

Address: 为模组地址;

ACK: 默认为 0x20, 表示该数据包为应答包;

Data[0~1]: 默认为 0x00;

Data[2]: 0x0A 表示响应的命令为 0x0A;

Data[3]: 0x01 表示接收正常, 0 表示接收异常;

参考接收:

A5 A5 A5 A5 01 20 04 00 00 00 0A 01 30

5.4 Running_IAP 指令

命令发送:

| Packet_Header | | | | Address | Pack_ID | Data_Len | | Data | Check_Sum |
|---------------|------|------|------|---------|---------|----------|------|----------|-----------|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | LSB | 0x0B | 0x52 | 0x00 | 82 Bytes | LSB |

升级时会对固件进行拆分, 数据段 (Data) 开头两字节表示该段数据相对于固件首字节的偏移量。

数据段格式:

| Package Shift | | 字符串验证码 | 固件数据 |
|---------------|---------|------------------|-------------------|
| Data[0] | Data[1] | Data[2]~Data[17] | Data[18]~Data[81] |

Data[0~1]: $\text{Package_Shift} = \text{Data}[0] + \text{Data}[1] * 256;$

Data[2]~Data[17]: 为固定的字符串验证码:

0x64 0x6F 0x77 0x6E 0x6C 0x6F 0x61 0x64 0x69 0x6E 0x67 0x00 0x00 0x00 0x00
0x00

Data[18]~Data[81]: 固件数据;

参考发送报文:

A5 A5 A5 A5 01 0B 52 00 00 00 64 6F 77 6E 6C 6F 61 64 69 6E 67 00 00 00 00 00
+ (Data[18]~Data[81]) + Check_Sum

命令接收:

| Packet_Header | | | | Address | ACK | Data_Len | | Data | Check_Sum |
|---------------|------|------|------|---------|------|----------|------|---------|-----------|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | LSB | 0x20 | 0x04 | 0x00 | 4 Bytes | LSB |

接收数据格式:

Address: 为模组地址;

ACK: 默认为 0x20, 表示该数据包为应答包;

Data[0~1] : Package_Shift = Data[0]+ Data[1]*256 表示响应的固件数据偏移量，建议在升级过程中检测应答时判断该偏移量作为一个保护机制。

Data[2]=0x0B 表示响应的命令为 0x0B;

Data[3]=0x01 表示接收正常，0 表示接收异常;

参考接收:

A5 A5 A5 A5 01 20 04 00 00 00 0B 01 31

5.5 Complete_IAP 指令

命令发送:

| Packet_Header | | | | Address | Pack_ID | Data_Len | | Data | Check_Sum |
|---------------|------|------|------|---------|---------|----------|------|----------|-----------|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | LSB | 0x0C | 0x16 | 0x00 | 22 Bytes | LSB |

数据段 Data 格式:

| 默认 | 字符串验证码 | 加密标志位 |
|-----------------|------------------|-------------------|
| Data[0]~Data[1] | Data[2]~Data[17] | Data[18]~Data[21] |

Data[0~1]: 默认为 0x00;

Data[2]~Data[17]: 为固定的字符串验证码:

0x63 0x6F 0x6D 0x70 0x6C 0x65 0x74 0x65 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00

0x00

Data[18]~Data[21]: 加密标志位, uint32_t 型, 加密固件为 1, 非加密固件为 0;

参考发送报文:

A5 A5 A5 A5 01 0C 16 00 00 00 63 6F 6D 70 6C 65 74 65 00 00 00 00 00 00 00 00 00 00 00 00 + (uint32_t 加密标志) + Check_Sum

命令接收:

| Packet_Header | | | | Address | ACK | Data_Len | | Data | Check_Sum |
|---------------|------|------|------|---------|------|----------|------|---------|-----------|
| 0xA5 | 0xA5 | 0xA5 | 0xA5 | LSB | 0x20 | 0x04 | 0x00 | 4 Bytes | LSB |

接收数据格式:

Address: 为模组地址;

ACK: 默认为 0x20, 表示该数据包为应答包;

Data[0~1]: 默认为 0x00;

Data[2]: 0x0C 表示响应的命令为 0x0C;

Data[3]: 0x01 表示接收正常, 0 表示接收异常;

参考接收报文:

```
A5 A5 A5 A5 01 20 04 00 00 00 0C 01 32
```

5.6 RESET_SYSTEM 指令

具体参见 [3.7 章节系统复位命令](#)。

5.7 OTA 升级 Q&A

- 1) Q: 发送复位指令后如何判断复位成功? 是否需要延时?
A: 可根据复位指令的应答包来判断执行成功; 建议收到应答后添加 500ms 延时再进行后续操作。
- 2) Q: 级联状态下升级功能异常可能是什么原因?
A: ①确认物理链路是否正确, 如确认是否能收到三个模组的点云数据; ②确认三个模组地址没有冲突, 可尝试重新分配地址; ③复位待升级的模组后重试;
- 3) Q: 升级过程被断电重启等情况打断后应该如何处理?
A: 重新发送 Start_IAP 指令即可重新升级。
- 4) Q: 级联升级后为什么读取到的版本号为 0?
A: 说明模组升级不成功, 需要复位该模组后重新升级。

6 使用注意

- 1) 在和 GS5 进行命令交互时, 除了停止扫描命令, 其他命令不能在扫描模式下进行交互, 这样容易导致报文解析错误。
- 2) GS5 上电不会自动开启测距, 需要发送启动扫描命令让其进入扫描模式, 需要停止测距时发送停止扫描命令让其停止扫描并进入休眠。
- 3) 正常启动 GS5, 我们建议的流程为:
第一步, 发送获取设备地址命令, 获取当前级联个数, 并配置地址;
第二步, 发送获取版本命令, 获取版本号;
第三步, 发送获取设备参数命令, 获取设备的角度参数用于数据解析;

第四步，发送启动扫描命令，获取点云数据。

4) 关于 GS5 透视窗透光材料的设计建议：

如果为 GS5 设计前面罩透视窗，其透光材料建议选用透红外光的 PC，要求透光区域为平面（平面度 $\leq 0.05\text{mm}$ ），并且该平面内的所有区域，在 780nm 至 1000nm 波段的透光率均大于 90%。

5) 关于导航板对 GS5 进行反复开关的推荐操作流程：

为了降低导航板的功耗，如果需要对 GS5 进行反复上下电的操作，建议在关电之前，先发送停止扫描命令（见 3.4 章节），然后将导航板的 TX 和 RX 配置为高阻态，再将 VCC 拉低进行关闭。下一次上电时，先拉高 VCC，再将 TX 和 RX 配置为正常的输出输入态，然后延时 300ms 后，再对线激光进行命令交互。

6) 关于 GS5 各指令发送后的最大等待时间：

获取地址：延时 800ms，获取版本：延时 100ms；

获取参数：延时 100ms，开启扫描：延时 400ms；

停止扫描：延时 100ms，设置波特率：延时 800ms；

设置沿边模式：延时 800ms，启动 OTA：延时 800ms；

7 修订

| 日期 | 版本 | 修订内容 |
|------------|-----|------|
| 2023-07-24 | 1.0 | 初撰 |