

深圳市海凌科电子有限公司
LD6002C
跌倒检测雷达模组通信协议

目录

1. 通讯协议概述	1
2. 跌倒检测项目	2
消息类型: 查询固件状态 0xFFFF.....	2
消息类型: 报告跌倒检测项目测试结果 0x0E02.....	3
消息类型: 设置安装高度 0x0E04.....	3
消息类型: 设置安装高度后回传结果 0x0E04.....	4
消息类型: 获取雷达参数 0x0E06.....	4
消息类型: 获取雷达参数后回传结果 0x0E06.....	5
消息类型: 设置跌倒阈值 0x0E08.....	5
消息类型: 设置跌倒阈值后回传结果 0x0E08.....	6
消息类型: 设置跌倒灵敏度 0x0E0A.....	6
消息类型: 设置跌倒灵敏度后回传结果 0x0E0A.....	7
消息类型: 高度上传结果 0x0E0E.....	7
消息类型: 打开或关闭 User log 信息 0x010E.....	7
消息类型: 设置报警区域参数 0x0E0C.....	8
消息类型: 设置报警区域参数回传结果 0x0E0C.....	8
消息类型: 雷达初始化设置参数 0x2110.....	9
消息类型: 报告 3D 点云检测项目测试结果 0x0A08.....	9
消息类型: 报告有无人检测项目测试结果 0x0F09.....	10
3. 编程接口	11

1. 通讯协议概述

针对不同的应用项目，列出所有与 TF 帧相关的消息，供用户参考并完成解析，对于文档所出现的消息类，以及消息数据位，是在对应实际项目所配备的。

每一帧数据由一个报头和一个有效载荷组成。这两个部分都可以通过校验和进行保护，以确保拒绝具有格式错误的标头（例如，具有损坏的长度字段）或损坏的有效负载的帧。

帧标头包含帧 ID 和消息类型。帧 ID 随着每条新消息的增加而增加。对于两个对等方，ID 字段的最高位固定为 1 和 0，以避免冲突。

帧 ID 可以在响应中重新使用，以将两条消息连接在一起。类型字段的值将在后面描述。

1.1. 帧结构

毫米波雷达帧中的字段配置如下：

格式	长度 (字节数)	格式含义
SOF	1	起始帧，通常固定为 0x01。
ID	2	帧 ID，MSB 对等位，代表发送包序（从 0 自增到 65535）。
LEN	2	数据帧长度，代表 DATA 字节数（但由于规定的总帧长度限制，规定 DATA 位长度不能超过 1024）。
TYPE	2	消息类型。
HEAD_CKSUM	1	使用 TF_CKSUM_XOR 计算头校验和（从 SOF 位到 TYPE 位先全部异或，后取反）。
DATA	N	长度为 LEN 的数据位。
DATA_CKSUM	1	使用 TF_CKSUM_XOR 计算数据校验和（DATA 位所有字节先全部异或，后取反）。

SOF 位 ~ HEAD_CKSUM 位与 DATA_CKSUM 高字节在前，低字节在后，而 DATA 位为低字节在前，高字节在后。

例如：DATA 数据类型为 uint32 的数据，它的值为 0x12345678，小端序方式传输数据，则为 0x78 0x56 0x34 0x21。ID 数据类型位 uint16 的数据，它的值为 0x1234，大端序方式传输数据，则为 0x12 0x34。

HEAD_CKSUM 位和 DATA_CKSUM 位在计算后，如果超过 1 个字节，将只取最低的 1 个字节数。例如：HEAD_CKSUM 位为 0x1232，最终只取 0x32。

2. 跌倒检测项目

本项目为跌倒项目，数据通过 UART 接口传输，波特率为 115200，数据字长 8 位，停止位 1 位，无奇偶校验位，无硬件流控制。功能实现流程：检测人体高度信息离地面持续 30 帧 * 灵敏度的总帧数（1s 20 帧）持续低于跌倒阈值高度信息，上报跌倒。

消息类型：查询固件状态 0xFFFF

它支持单向数据传输模式和双向数据传输模式，是用于查询固件状态的消息类型。通常单向数据传输模式是在雷达上电后，就会通过串口发送数据给上位机。如果主机收到下位机版本信息，则表示雷达工作正常，否则表示雷达工作异常/关闭。

上位机发送数据到雷达：TF_ID_FIRMWARE_STATUS				
格式	字节数	基本类型	帧结构	示例帧
SOF	1 byte	uint8	起始帧	01
ID	2 byte	uint16	帧 ID	00 00
LEN	2 byte	uint16	数据帧长度	00 00
TYPE	2 byte	uint16	帧类型	FF FF
HEAD_CKSUM	1 byte	uint8	头校验和	FE

雷达回传数据给上位机：TF_ID_FIRMWARE_STATUS					
格式	字节数	基本类型	帧结构	示例帧	帧含义
SOF	1 byte	uint8	起始帧	01	跌倒检测项目，版本号 为 4.0.18。
ID	2 byte	uint16	帧 ID	00 00	
LEN	2 byte	uint16	数据帧长度	00 04	
TYPE	2 byte	uint16	帧类型	FF FF	
HEAD_CKSUM	1 byte	uint8	头校验和	FA	
DATA	1 byte	uint8	[project_name]	08	
DATA	1 byte	uint8	[major_version]	04	
DATA	1 byte	uint8	[sub_version]	00	
DATA	1 byte	uint8	[modified_version]	12	
DATA_CKSUM	1 byte	uint8	数据校验和	E1	

以下是每个 DATA 位对应的含义：

- [project_name]：表示雷达所运行的项目。
 - 值为 8，表示跌倒检测项目。
- [major_version]：表示雷达主版本号。
- [sub_version]：表示雷达子版本号。
- [modified_version]：表示雷达修订版本号。

消息类型：报告跌倒检测项目测试结果 0x0E02

消息类型为 0x0E02，仅支持单向数据传输模式。

雷达发送数据给上位机：MSG_IND_FALL_STATUS					
格式	字节数	基本类型	帧结构	示例帧	帧含义
SOF	1 byte	uint8	起始帧	01	用于报告跌倒测试结果。
ID	2 byte	uint16	帧 ID	00 00	
LEN	2 byte	uint16	数据帧长度	00 01	
TYPE	2 byte	uint16	帧类型	0E 02	
HEAD_CKSUM	1 byte	uint8	头校验和	F3	
DATA	1 byte	uint8	[is_fall]	01	
DATA_CKSUM	1 byte	uint8	数据校验和	FE	

以下是每个 DATA 位对应的含义：

- [is_fall]：判断是否跌倒。
 - 值为 0：正常。
 - 值为 1：跌倒。

消息类型：设置安装高度 0x0E04

消息类型为 0x0E04，支持双向数据传输模式。

上位机发送数据到雷达：MSG_IND_FALL_SET_HIGH					
格式	字节数	基本类型	帧结构	示例帧	帧含义
SOF	1 byte	uint8	起始帧	01	用于设置雷达安装高度。
ID	2 byte	uint16	帧 ID	00 00	
LEN	2 byte	uint16	数据帧长度	00 04	
TYPE	2 byte	uint16	帧类型	0E 04	
HEAD_CKSUM	1 byte	uint8	头校验和	F0	
DATA	4 byte	float	[high]	00 00 20 40	
DATA_CKSUM	1 byte	uint8	数据校验和	9F	

以下是每个 DATA 位对应的含义：

- [high]：设置雷达安装高度，范围为 1 ~ 5m。

转换成 float：例如[x_point]位为 0x66、0x66、0xA2、0x41，先拼成 uint32 位整形，由于 TF 帧 Data 位小端序，所以值为 0x41A26666，然后进行 float 类型强转，最终结果为：20.3。

```

1. int main(void)
2. {
3.     unsigned int param = 0x41A26666;
4.     float res = *(float *)&param;
5.
6.     printf("data: %f\n", res);
7.     return 0;
8. }
```

消息类型：设置安装高度后回传结果 0x0E04

雷达回传数据给上位机：MSG_IND_FALL_RES					
格式	字节数	基本类型	帧结构	示例帧	帧含义
SOF	1 byte	uint8	起始帧	01	用于设置雷达安装高度后， 回传状态。
ID	2 byte	uint16	帧 ID	00 00	
LEN	2 byte	uint16	数据帧长度	00 01	
TYPE	2 byte	uint16	帧类型	0E 04	
HEAD_CKSUM	1 byte	uint8	头校验和	F5	
DATA	1 byte	uint8	[result]	01	
DATA_CKSUM	1 byte	uint8	数据校验和	FE	

当雷达收到 TYPE 为 0x0E04 时，雷达将会回传数据。[result]有两种结果：

- 值为 0：设置高度失败。
- 值为 1：设置高度成功。

消息类型：获取雷达参数 0x0E06

消息类型为 0x0E06，支持双向数据传输模式。

上位机发送数据到雷达：MSG_IND_FALL_GET_HIGH					
格式	字节数	基本类型	帧结构	示例帧	帧含义
SOF	1 byte	uint8	起始帧	01	获取下位机参数。
ID	2 byte	uint16	帧 ID	00 00	
LEN	2 byte	uint16	数据帧长度	00 00	
TYPE	2 byte	uint16	帧类型	0E 06	
HEAD_CKSUM	1 byte	uint8	头校验和	F6	

上位机下发命令给雷达，请求获取下位机的数据，雷达会回复 类型为 0x0E06 的信号，详细介绍见下

消息类型：获取雷达参数后回传结果 0x0E06

雷达回传数据给上位机：MSG_IND_FALL_GET_HIGH					
格式	字节数	基本类型	帧结构	示例帧	帧含义
SOF	1 byte	uint8	起始帧	01	向上位机发送雷达的参数。
ID	2 byte	uint16	帧 ID	00 00	
LEN	2 byte	uint16	数据帧长度	00 1C	
TYPE	2 byte	uint16	帧类型	0E 06	
HEAD_CKSUM	1 byte	uint8	头校验和	EA	
DATA	4 byte	float	[high]	/	
DATA	4 byte	float	[threshold]	/	
DATA	4 byte	uint32	[sensitivity]	/	
DATA	4 byte	float	[rect_XL]	/	
DATA	4 byte	float	[rect_XR]	/	
DATA	4 byte	float	[rect_ZF]	/	
DATA	4 byte	float	[rect_ZB]	/	
DATA_CKSUM	1 byte	uint8	数据校验和	/	

当雷达收到 TYPE 为 0x0E06 时，雷达将会回传数据。有两种结果：

- 值为 0：获取失败。
- 值为其它：获取成功。

消息类型：设置跌倒阈值 0x0E08

消息类型为 0x0E08，支持双向数据传输模式。

上位机发送数据到雷达：MSG_IND_FALL_SET_THRESHOLD					
格式	字节数	基本类型	帧结构	示例帧	帧含义
SOF	1 byte	uint8	起始帧	01	设置跌倒阈值，雷达默认跌倒阈值为 0.6m。地面处高度为 0m
ID	2 byte	uint16	帧 ID	00 00	
LEN	2 byte	uint16	数据帧长度	00 04	
TYPE	2 byte	uint16	帧类型	0E 08	
HEAD_CKSUM	1 byte	uint8	头校验和	FC	
DATA	4 byte	float	[set_threshold]	9A 99 19 3F	
DATA_CKSUM	1 byte	uint8	数据校验和	DA	

转换成 float：例如[high]位为 0x9A、0x99、0x19、0x3F，先拼成 uint32 位整形，由于 TF 帧 Data 位小端序，所以值为 0x3F19999A，然后进行 float 类型强转，最终结果为：0.6。

```

1. int main(void)
2. {
3.     unsigned int param = 0x41A26666;
4.     float res = *(float *)&param;
5.
6.     printf("data: %f\n", res);

```

```
7.     return 0;
8. }
```

消息类型：设置跌倒阈值后回传结果 0x0E08

雷达回传数据给上位机：MSG_IND_FALL_SET_THRESHOLD					
格式	字节数	基本类型	帧结构	示例帧	帧含义
SOF	1 byte	uint8	起始帧	01	设置跌倒阈值后，回传状态。
ID	2 byte	uint16	帧 ID	00 00	
LEN	2 byte	uint16	数据帧长度	00 01	
TYPE	2 byte	uint16	帧类型	0E 08	
HEAD_CKSUM	1 byte	uint8	头校验和	F9	
DATA	1 byte	uint8	[value]	01	
DATA_CKSUM	1 byte	uint8	数据校验和	FE	

当雷达收到 TYPE 为 0x0E08 时，雷达将会回传数据。回传的数据有两种结果：

- 值为 0x00：获取失败。
- 值为 0x01：获取成功。

消息类型：设置跌倒灵敏度 0x0E0A

消息类型为 0x0E0A，支持双向数据传输模式。

上位机发送数据到雷达：MSG_IND_FALL_SET_SENSITIVITY					
格式	字节数	基本类型	帧结构	示例帧	帧含义
SOF	1 byte	uint8	起始帧	01	设置跌倒灵敏度
ID	2 byte	uint16	帧 ID	00 00	
LEN	2 byte	uint16	数据帧长度	00 01	
TYPE	2 byte	uint16	帧类型	0E 0A	
HEAD_CKSUM	1 byte	uint8	头校验和	FB	
DATA	4 byte	uint32	[high]	03 00 00 00	
DATA_CKSUM	1 byte	uint8	数据校验和	FE	

设置跌倒灵敏度，初始值 10.(范围 3~30)

消息类型：设置跌倒灵敏度后回传结果 0x0E0A

雷达回传数据给上位机：MSG_IND_FALL_SET_SENSITIVITY					
格式	字节数	基本类型	帧结构	示例帧	帧含义
SOF	1 byte	uint8	起始帧	01	当雷达收到 TYPE 为 0x0E0A 时，雷达将会回传数据。回传的数据有两种结果： 值为 0：获取失败。 值为 1：获取成功。
ID	2 byte	uint16	帧 ID	00 00	
LEN	2 byte	uint16	数据帧长度	00 01	
TYPE	2 byte	uint16	帧类型	0E 0A	
HEAD_CKSUM	1 byte	uint8	头校验和	FB	
DATA	1 byte	uint8	[value]	01	
DATA_CKSUM	1 byte	uint8	数据校验和	FE	

消息类型：高度上传结果 0x0E0E

雷达回传数据给上位机：MSG_IND_FALL_SET_SENSITIVITY					
格式	字节数	基本类型	帧结构	示例帧	帧含义
SOF	1 byte	uint8	起始帧	01	雷达自动上传当前目标点的高度信息（雷达位置高度为 0m）
ID	2 byte	uint16	帧 ID	00 00	
LEN	2 byte	uint16	数据帧长度	00 01	
TYPE	2 byte	uint16	帧类型	0E 0E	
HEAD_CKSUM	1 byte	uint8	头校验和	FF	
DATA	1 byte	Uint32	[value]	01 00 00 00	
DATA_CKSUM	1 byte	uint8	数据校验和	FE	

消息类型：打开或关闭 User log 信息 0x010E

消息类型为 0x010E，只支持单向数据传输模式

上位机发送数据到雷达					
格式	字节数	基本类型	帧结构	示例帧	帧含义
SOF	1 byte	uint8	起始帧	01	0：关闭 User log 信息 1：打开 User log 信息
ID	2 byte	uint16	帧 ID	00 00	
LEN	2 byte	uint16	数据帧长度	00 04	
TYPE	2 byte	uint16	帧类型	01 0E	
HEAD_CKSUM	1 byte	uint8	头校验和	F5	
DATA	4 byte	uint32	[value]	01 00 00 00	
DATA_CKSUM	1 byte	uint8	数据校验和	FE	

- 值为 0x00：关闭 User log 信息。
- 值为 0x01：打开 User log 信息。

消息类型：设置报警区域参数 0x0E0C

消息类型为 0x0E0C，支持双向数据传输模式

上位机发送数据到雷达					
格式	字节数	基本类型	帧结构	示例帧	帧含义
SOF	1 byte	uint8	起始帧	01	设置跌倒报警区域范围，设置范围 0.3~1.5
ID	2 byte	uint16	帧 ID	00 00	
LEN	2 byte	uint16	数据帧长度	00 10	
TYPE	2 byte	uint16	帧类型	0E 0C	
HEAD_CKSUM	1 byte	uint8	头校验和	EC	
DATA	4 byte	float	[rect_XL]	00 00 00 3F	
DATA	4 byte	float	[rect_XR]	00 00 00 3F	
DATA	4 byte	float	[rect_ZF]	00 00 00 3F	
DATA	4 byte	float	[rect_ZB]	00 00 00 3F	
DATA_CKSUM	1 byte	uint8	数据校验和	FF	

设置跌倒报警区域范围，设置范围 0.3~1.5

消息类型：设置报警区域参数回传结果 0x0E0C

雷达发送数据到上位机					
格式	字节数	基本类型	帧结构	示例帧	帧含义
SOF	1 byte	uint8	起始帧	01	0：失败 1：成功
ID	2 byte	uint16	帧 ID	00 00	
LEN	2 byte	uint16	数据帧长度	00 04	
TYPE	2 byte	uint16	帧类型	0E 0C	
HEAD_CKSUM	1 byte	uint8	头校验和	F8	
DATA	1 byte	uint8	[value]	01	
DATA_CKSUM	1 byte	uint8	数据校验和	FE	

- 值为 0x00：获取失败。
- 值为 0x01：获取成功。

消息类型：雷达初始化设置参数 0x2110

消息类型为 0x2110，支持单向数据传输模式

上位机发送数据给雷达					
格式	字节数	基本类型	帧结构	示例帧	帧含义
SOF	1 byte	uint8	起始帧	01	雷达初始化设置参数 High:2.4 Threshold:0.6 Sensitivity:10 Rect_XL:1.5 Rect_XR:1.5 Rect_ZF:1.5 Rect_ZB:1.5
ID	2 byte	uint16	帧 ID	00 00	
LEN	2 byte	uint16	数据帧长度	00 00	
TYPE	2 byte	uint16	帧类型	2110	
HEAD_CKSUM	1 byte	uint8	头校验和	CF	

当收到 TYPE 为 0x2110 时，雷达初始化设置参数。

High:2.4

Threshold:0.6

Sensitivity:10

Rect_XL:1.5; Rect_XR:1.5 ; Rect_ZF:1.5; Rect_ZB:1.5;

设置的内容和 TYPE 类型 0x0E06 的内容相关

消息类型：报告 3D 点云检测项目测试结果 0x0A08

消息类型为 0x0A08，仅支持单向数据传输模式(开启 User log 后自动上传)。

雷达发送数据给上位机：MSG_IND_3D_CLOUD_RES					
格式	字节数	基本类型	帧结构	示例帧	帧含义
SOF	1 byte	uint8	起始帧	01	用于报告 3D 点云检测结果。其中消息类型为 0x0A08 是点云信息
ID	2 byte	uint16	帧 ID	00 00	
LEN	2 byte	uint16	数据帧长度	\	
TYPE	2 byte	uint16	帧类型	0A 08	
HEAD_CKSUM	1 byte	uint8	头校验和	\	
DATA	4 byte	int32	[target_num]	\	
DATA	4 byte	int32	[cluster_index]	\	
DATA	4 byte	float	[x_point]	\	
DATA	4 byte	float	[y_point]	\	
DATA	4 byte	float	[z_point]	\	
DATA	4 byte	float	[speed]	\	
DATA_CKSUM	1 byte	uint8	数据校验和	\	

以下是每个 DATA 位对应的含义：

- [target_num]: 目标个数。
- [cluster_index]: 聚类目标 ID。
- [x_point]: x 坐标, 单位: m。
- [y_point]: y 坐标, 单位: m。
- [z_point]: z 坐标, 单位: m。
- [speed]: 速度, 单位: m/s。

当点云存在 N 个的时候, x、y、z、dop_idx 也存在 N 个, cluster_id 距离相近点云 ID 相同, 示例:

```

01 41 A1 00 68 0A 08 74 05 00 00 00 00 00 00 62 F6 C2 BD 38 24 F7 3E E6 A8 41 3E 39 63 18 BC 00
起始帧 帧 ID 数据帧长度 帧类型 头校验和 点云个数 点云 ID x0 坐标 y0 坐标 z0 坐标 速度 V0
00 00 00 D5 94 A5 BD 14 9E 18 3F 69 2C FA 3C 39 63 18 BC 00 00 00 00 73 5C 2F BE 3C 4F F8 3E EE EE
点云 ID x1 坐标 y1 坐标 z1 坐标 速度 V1 点云 ID x2 坐标 y2 坐标 z2 坐标
E1 3E 39 63 18 BC 00 00 00 00 BF 56 CA BD F8 E4 EE 3E 39 CC 65 3E 39 63 18 3C 00 00 00 00 48 0B A3
速度 V2 点云 ID x3 坐标 y3 坐标 z3 坐标 速度 V3 点云 ID x4 坐标
BD 37 92 18 3F E0 2F 16 3D 39 63 18 3C 6F
y4 坐标 z4 坐标 速度 V4 数据校验和
    
```

消息类型：报告有无人检测项目测试结果 0x0F09

消息类型为 0x0F09, 仅支持单向数据传输模式。

雷达发送数据给上位机: MSG_IND_FALL_STATUS					
格式	字节数	基本类型	帧结构	示例帧	帧含义
SOF	1 byte	uint8	起始帧	01	用于报告有无人测试结果。
ID	2 byte	uint16	帧 ID	00 00	
LEN	2 byte	uint16	数据帧长度	00 01	
TYPE	2 byte	uint16	帧类型	0F 09	
HEAD_CKSUM	1 byte	uint8	头校验和	F9	
DATA	1 byte	uint8	[is_human]	01	
DATA_CKSUM	1 byte	uint8	数据校验和	FE	

以下是每个 DATA 位对应的含义：

- [is_human]: 判断是否有人。
 - 值为 0: 无人。
 - 值为 1: 有人。

3. 编程接口

3.1. 编码 TF 消息

```
void tinyFrameTx(TF_TYPE type, uint8 *data, TF_LEN len);
```

其中 type 为发送数据类型，uint16 类型，例如人员检测数据结果上报，数据类型为 0x0A10。见 4.2.1.6 详述。

uint8* data 是发送数据的地址。

len 为发送数据的长度，uint16 类型。

3.2. 解码 TF 消息

```
TinyFrameRx tinyFrameRx(void);
```

成功接收消息后，接收的数据返回到一个 TinyFrameRx 类型的变量。

其中每个 CKSUM 的解析如下所示：

HEAD_CKSUM：TF 帧头校验和【从第一个字节开始到 HEAD_CKSUM 位的上一个字节】

DATA_CKSUM：TF 数据校验和【DATA 的第一个到 DATA_CKSUM 位的上一个字节】

其中计算 CKSUM 的方法 c 代码如下所示：

```
1. unsigned char getCksum(unsigned char *data, unsigned char len)
2. {
3.     unsigned char ret = 0;
4.
5.     for (int i = 0; i < len; i++)
6.         ret = ret ^ data[i];
7.
8.     ret = ~ret;
9.
10.    return ret;
11. }
12.
```

3.3. 示例代码

如果想要解析 TF 帧数据的 demo（包含 Linux 环境与 Keil μVision5 环境下的 C 语言 demo、Python 语言 demo），可以直接与销售沟通获得。（不提供应用代码方面技术支持）

Revision History

Revison	Release Date	Description
V1.0	2023/11/28	第一次发行
V1.1	2024/06/17	1、增加 TF 帧结构说明 2、增加有无人信息 0F09 上报 3、补充点云数据解析说明 4、4、跌倒灵敏度设置范围更改为【3~30】