



深圳市海凌科电子有限公司

**HLK-LD6002-呼吸心率检测
雷达模组通信协议**

目 录

1. 数据通信链路约定	1
1.1. 通信链路介绍	1
2. 协议约定说明	2
2.1. 协议结构框图	2
2.2. 协议结构简介	2
2.3. TF 帧解析流程	3
2.4. TF 示例帧解析详解	4
3. 通用消息类型	9
3.1. 消息类型：查询固件状态（TYPE:0xFFFF）	9
3.2. 消息类型：返回固件状态（TYPE:0xFFFF）	9
3.3. 消息类型：进入 OTA 升级（TYPE:0x3000）	10
3.4. 消息类型：OTA 升级（TYPE:0x3000）	10
4. 呼吸+心率检测项目	11
4.1. 消息类型：报告有无人检测项目测试结果 0x0F09	11
4.2. 消息类型：报告人员位置 0x0A04	11
4.3. 消息类型：报告相位测试结果 0x0A13	12
4.4. 消息类型：报告呼吸速率测试结果 0x0A14	13
4.5. 消息类型：报告心跳速率测试结果 0x0A15	13
4.6. 消息类型：报告检测目标距离 0x0A16	14
4.7. 消息类型：报告跟踪目标位置信息 0x0A17	14
5. 数据转换解析说明	15
6. 编程接口	15
6.1. 编码 TF 消息	15
6.2. 解码 TF 消息	16
6.3. 示例代码	16
附录 A 文档修订记录	17

1.数据通信链路约定

1.1. 通信链路介绍

服务端是指模组端，客户端是指上位机端 服务端的通信方式：主动上传和被动上传 客户端的通信方式：主动下发。

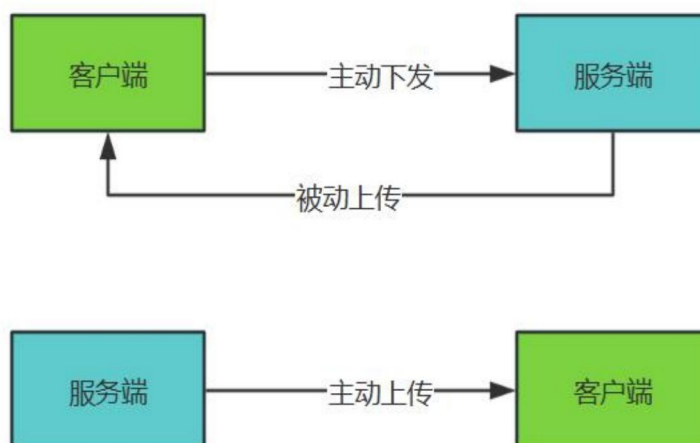
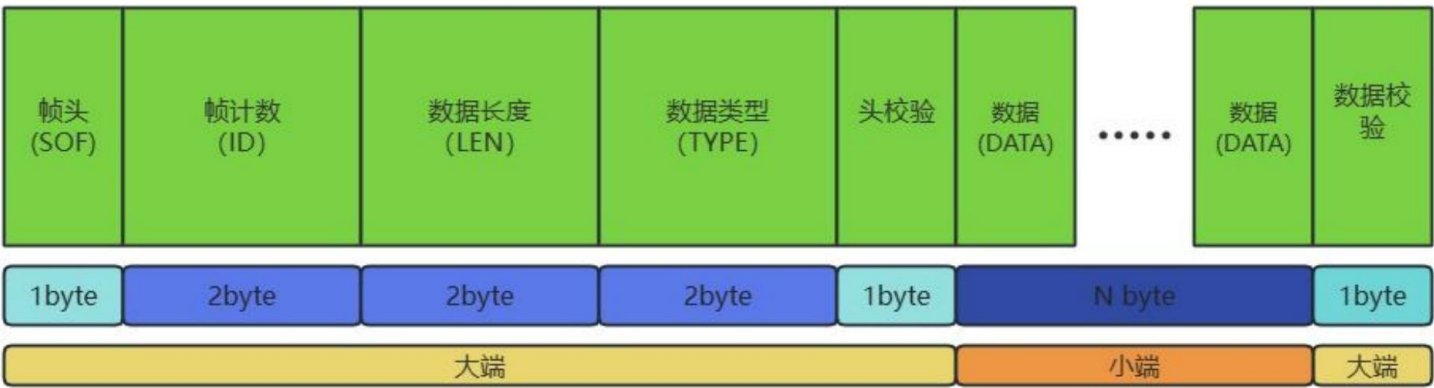


图 1-1 通讯方式

2.协议约定说明

2.1.协议结构框图



2.2.协议结构简介

格式	字节数	存储方式	格式含义
SOF	1	大端序	代表一帧 TF 数据的起始位 ，通常固定为 0x01。
ID	2	大端序	代表发送的包序 ， 可用于检测客户端和服务端数据帧是否连续。
LEN	2	大端序	程序中默认设置总数据帧长度不超过 1024 个字节。
TYPE	2	大端序	代表整个数据帧的消息类型 ， 不同的消息类型所代表的功能不同。
HEAD_CKSUM	1	大端序	代表头帧校验和 ， 从 SOF 位到 TYPE 位先全部进行异或运算 ， 再取反。
DATA	N	小端序	代表数据的有效载荷 ， DATA 位数据长度与 LEN 位相关。例如 LEN 位值为 2 ， 则 DATA 位由两个字节数据组成。
DATA_CKSUM	1	大端序	代表尾帧校验和 ，所有 DATA 位先全部进行 异或运算 ， 再取反。

注：SOF 位~HEAD_CKSUM 位与 DATA_CKSUM 采用大端排序原因 ， 方便查看指令的 TYPE 类型以及数据长度 ， DATA 位采用小端的排序 ， 符合计算器存储顺序 ， 方便解析 DATA 数据（可以强行转换数据类型）。

2.3.TF 帧解析流程

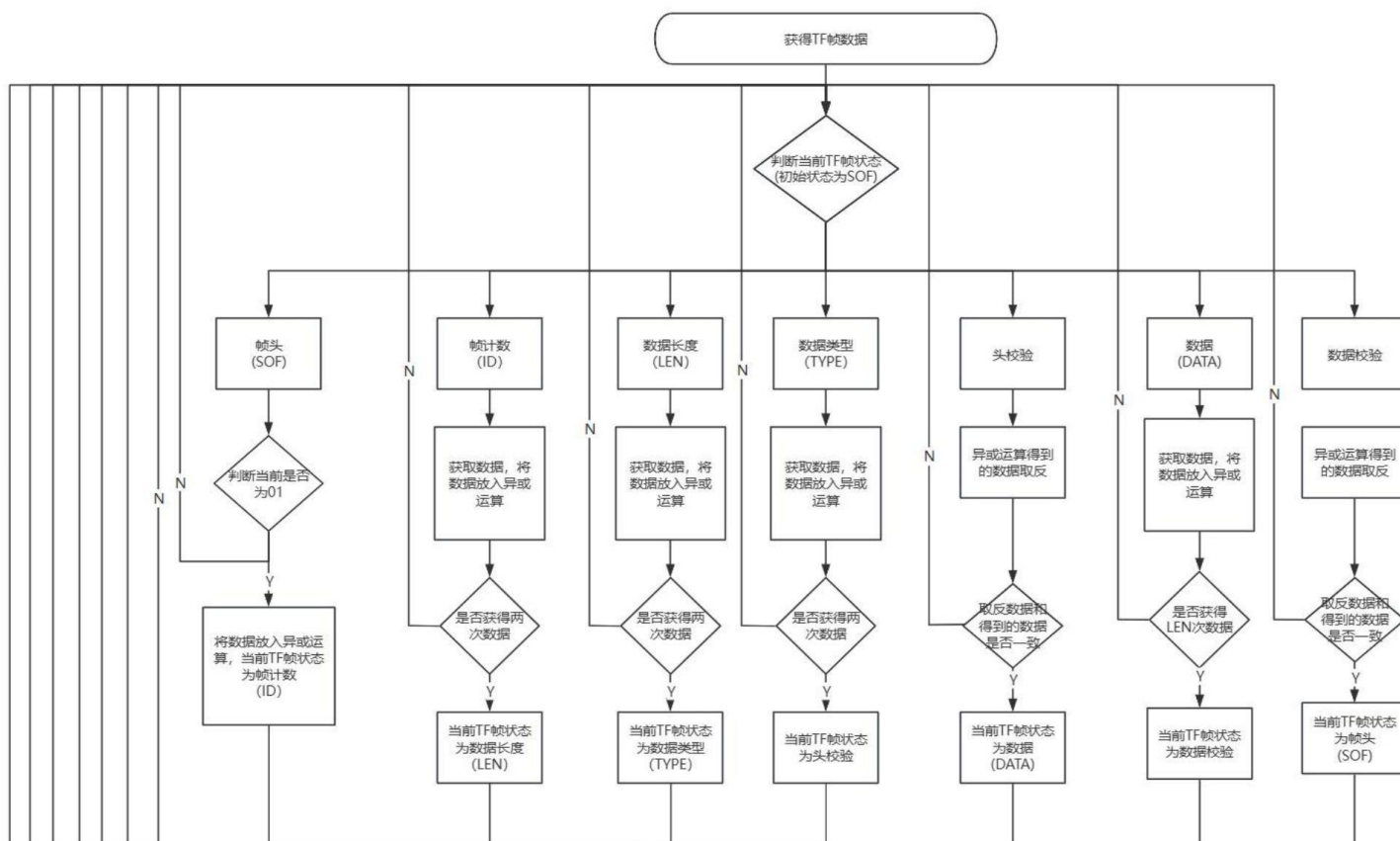
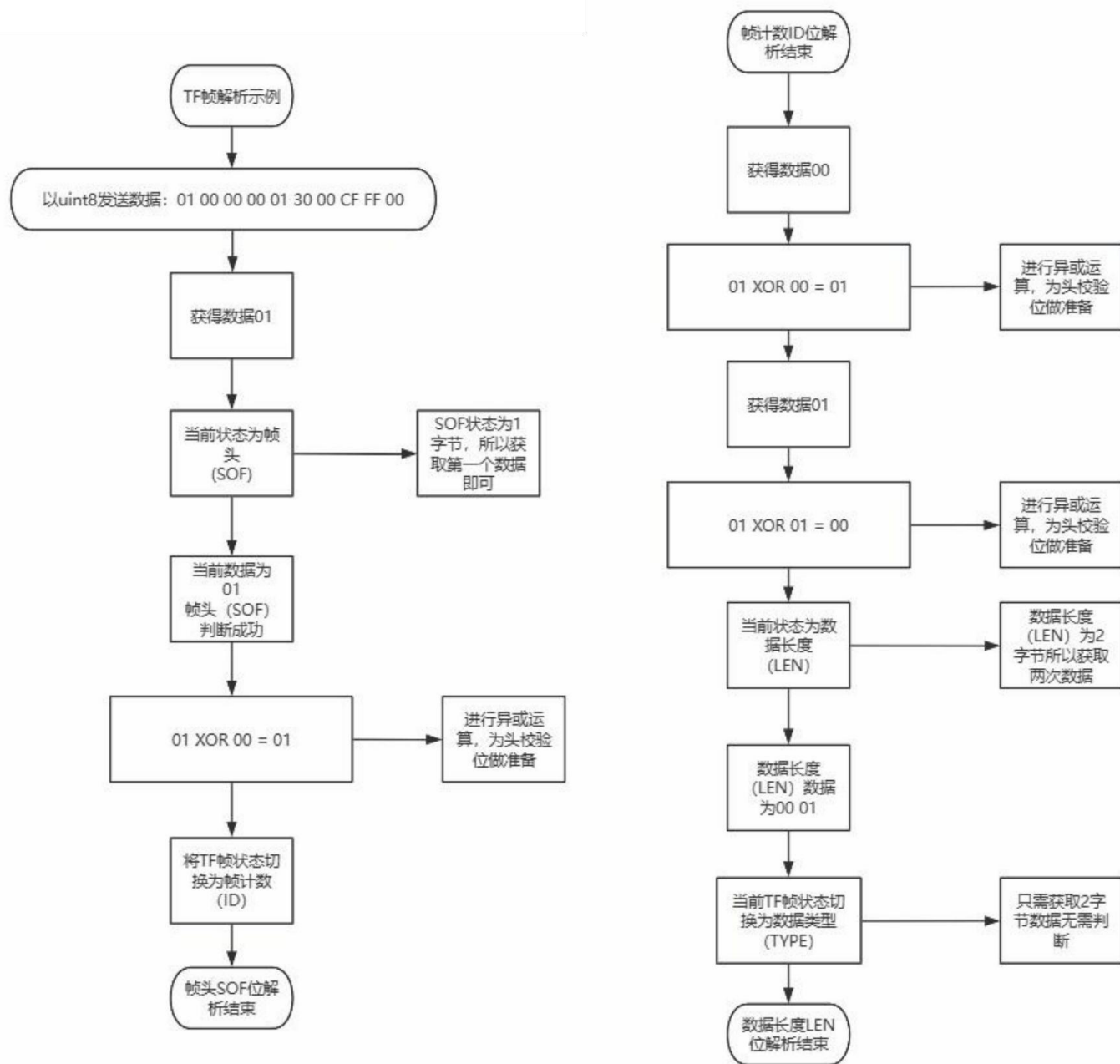
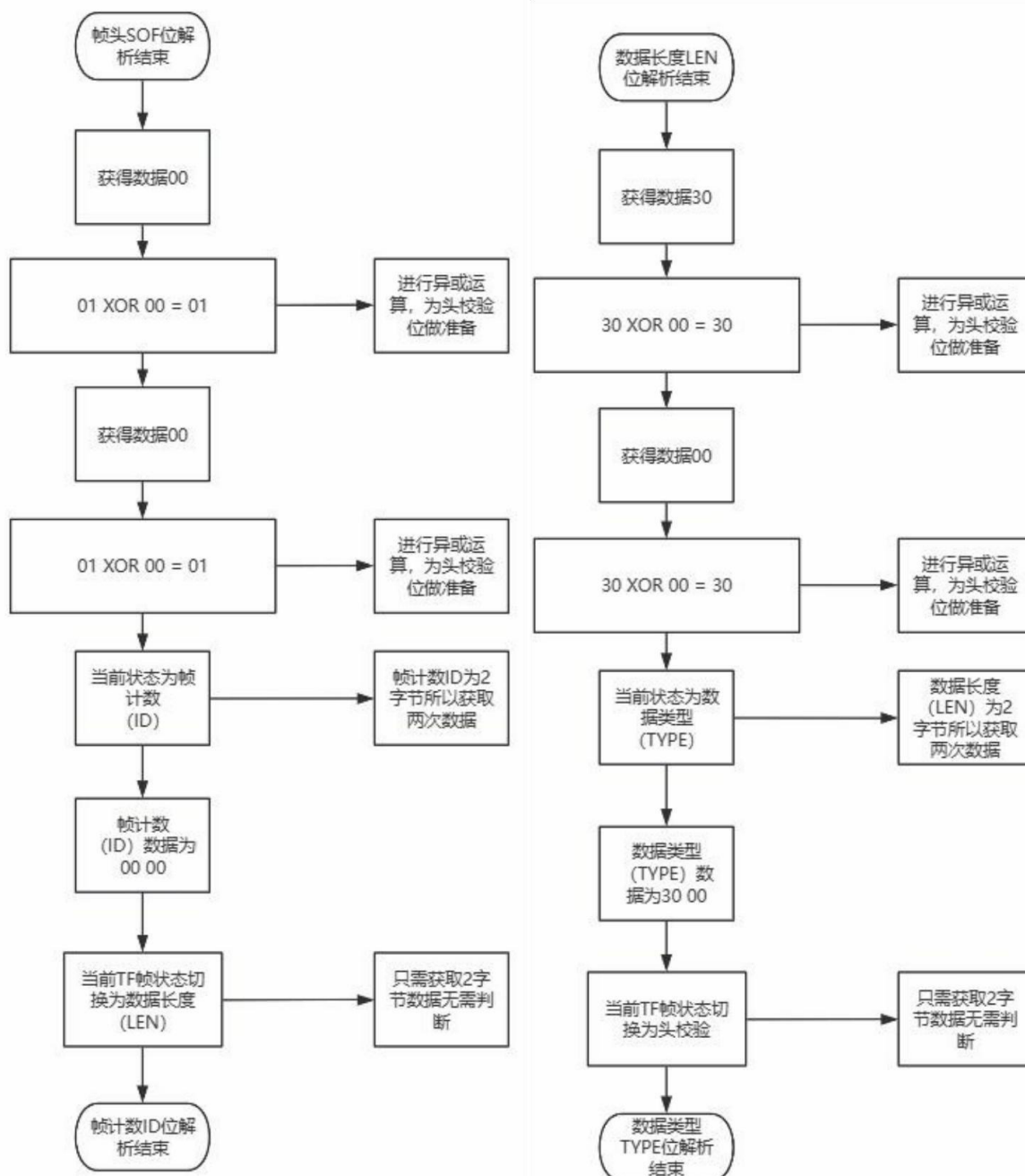
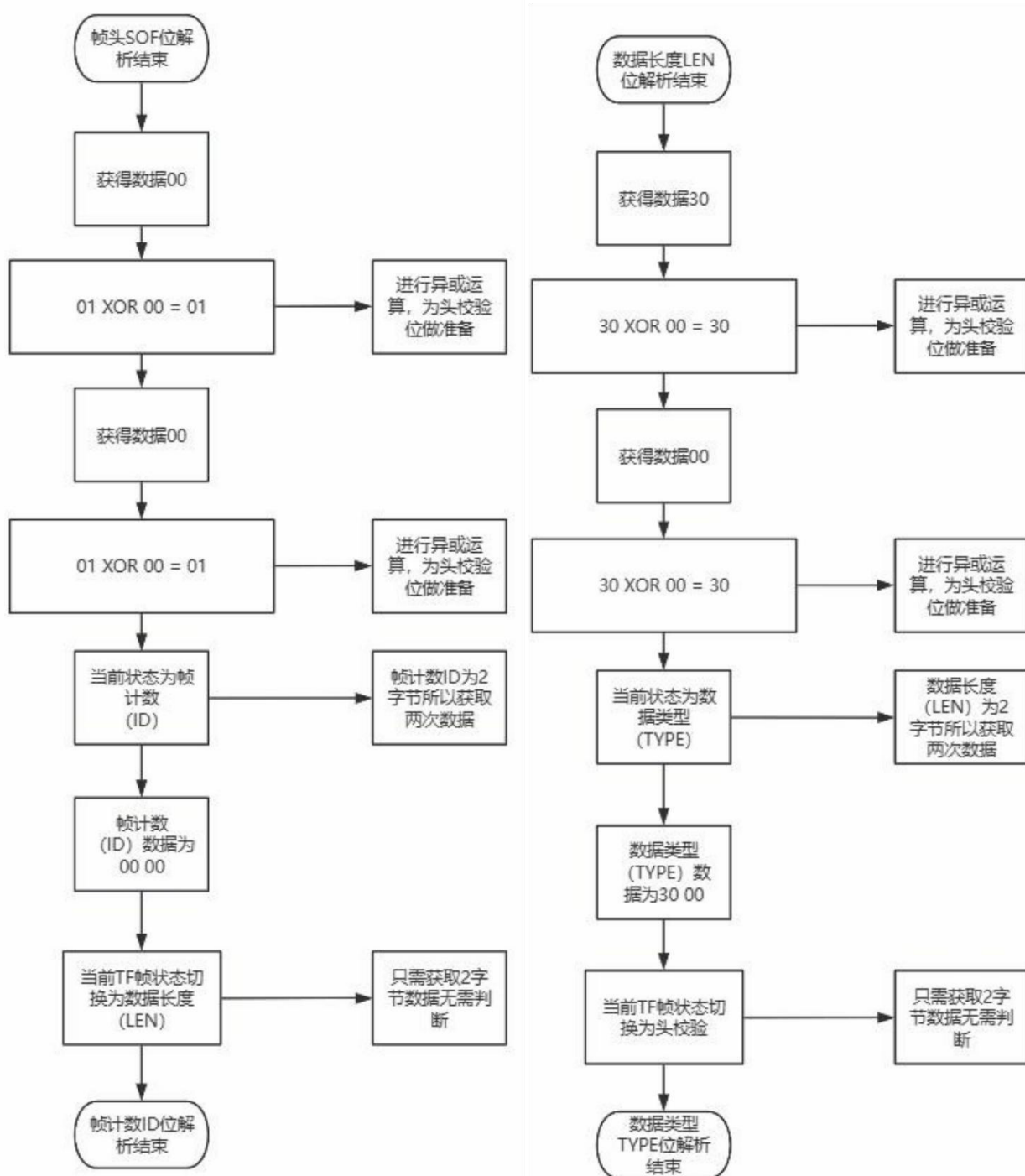


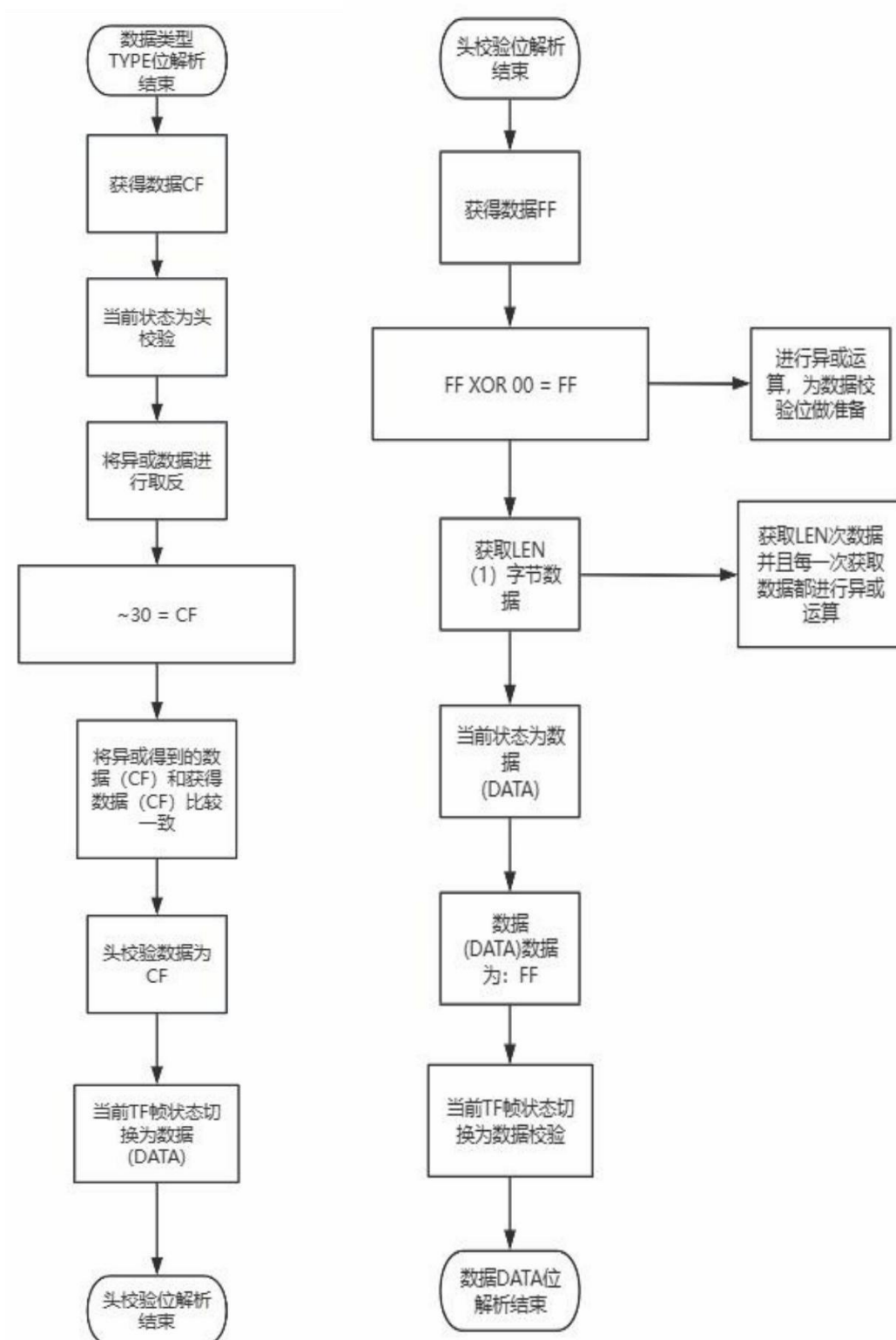
图 2-1 TF 帧解析流程

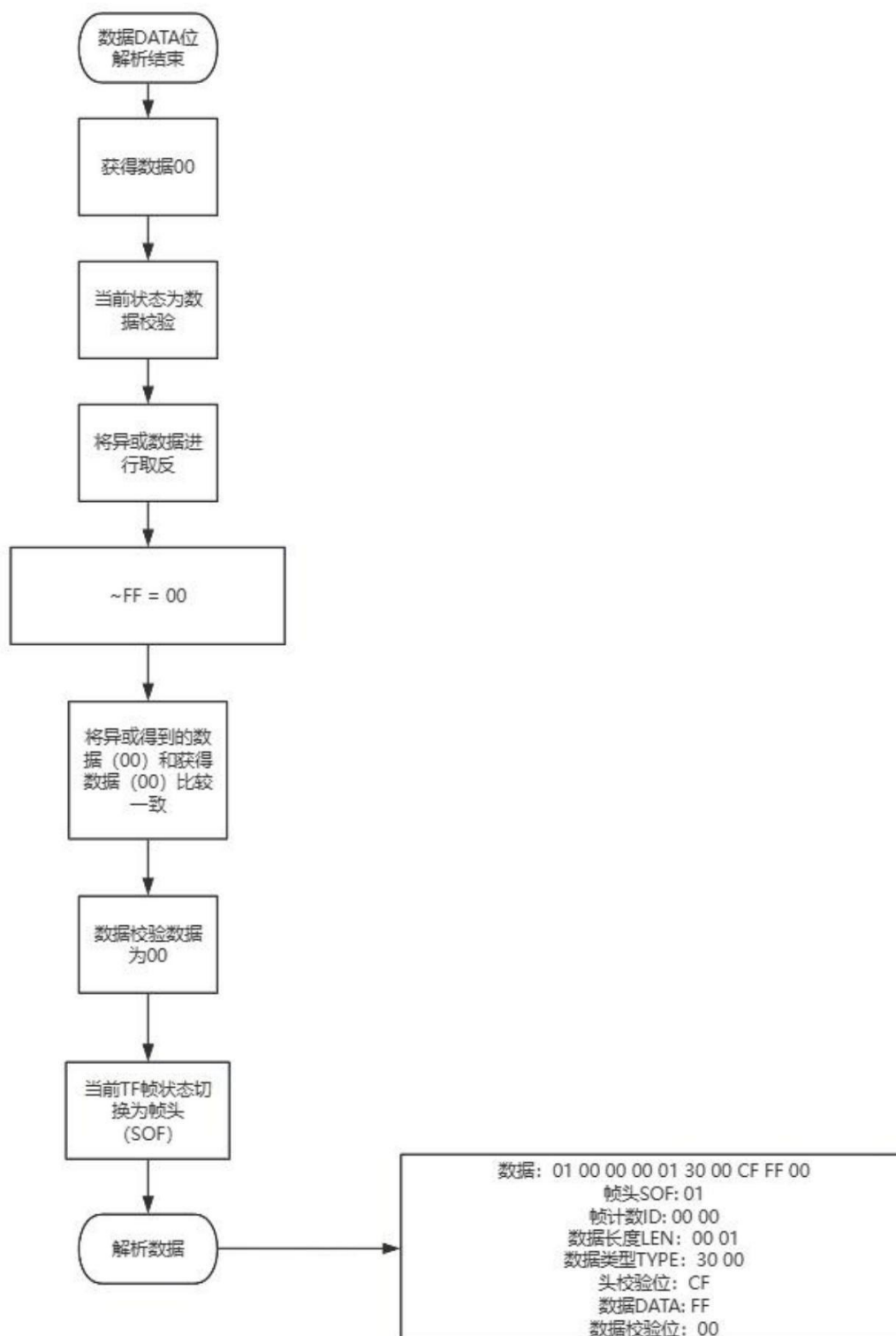
2.4.TF 示例帧解析详解











3.通用消息类型

3.1.消息类型：查询固件状态（TYPE:0xFFFF）

通信方式：	主动下发		
格式	字节数	基本类型	帧结构
SOF	1 byte	uint8	起始帧
ID	2 byte	uint16	帧 ID
LEN	2 byte	uint16	数据帧长度
TYPE	2 byte	uint16	帧类型
HEAD_CKSUM	1 byte	uint8	头校验和

示例 TF 帧：01 00 01 00 00 FF FF FF

3.2.消息类型：返回固件状态（TYPE:0xFFFF）

通信方式：	被动上传		
格式	字节数	基本类型	帧结构
SOF	1 byte	uint8	起始帧
ID	2 byte	uint16	帧 ID
LEN	2 byte	uint16	数据帧长度
TYPE	2 byte	uint16	帧类型
HEAD_CKSUM	1 byte	uint8	头校验和
DATA（见下文）			
DATA_CKSUM	1 byte	uint8	数据校验和

以下是 DATA 位对应的含义：

DATA			
字节数	基本类型	帧结构	帧含义
1 byte	uint8	[project_name]	表示雷达所运行的项目 0：表示存在感知项目 1：表示呼吸检测项目。 2：表示手势检测项目。 3：表示测距项目。 4：表示人员计数项目。 5：表示 3D 点云检测项目。
1 byte	uint8	[major_version]	表示雷达主版本号
1 byte	uint8	[sub_version]	表示雷达子版本号
1 byte	uint8	[modified_version]	表示雷达修订版本号

示例 TF 帧：01 04 AF 00 04 FF FF 51 06 03 0A 00 F0

3.3.消息类型：进入 OTA 升级 (TYPE:0x3000)

通信方式：	主动下发		
格式	字节数	基本类型	帧结构
SOF	1 byte	uint8	起始帧
ID	2 byte	uint16	帧 ID
LEN	2 byte	uint16	数据帧长度
TYPE	2 byte	uint16	帧类型
HEAD_CKSUM	1 byte	uint8	头校验和

示例 TF 帧：01 00 01 00 00 30 00 CF

3.4. 消息类型：OTA 升级 (TYPE:0x3000)

通信方式：	被动上传		
格式	字节数	基本类型	帧结构
SOF	1 byte	uint8	起始帧
ID	2 byte	uint16	帧 ID
LEN	2 byte	uint16	数据帧长度
TYPE	2 byte	uint16	帧类型
HEAD_CKSUM	1 byte	uint8	头校验和
DATA (见下文)			
DATA_CKSUM	1 byte	uint8	数据校验和

以下是 DATA 位对应的含义：

DATA			
字节数	基本类型	帧结构	帧含义
1 byte	uint8	[tmpCode Info]	FF:待升级

示例 TF 帧：01 00 03 00 01 30 00 CC FF 00

4.呼吸+心率检测项目

4.1.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 02	
TYPE	2 byte	uint16	帧类型	0F 09	
HEAD_CKSU	1 byte	uint8	头校验和	F9	
DATA	2 byte	uint8	[is_human]	01 00	
DATA_CKSU	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 00 00：无人。
- 值为 01 00：有人。

4.2.消息类型：报告人员位置 0x0A04

消息类型为 0x0A04，仅支持单向数据传输模式。0x0A04 为目标数据，0x0A08 为点云数据。

雷达发送数据给上位机：MSG_IND_HUMAN_DETECTION_3D_TGT_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	用于报告人员位置。
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 04	
HEAD_CKSU	1 byte	uint8	头校验和	/	
DATA	4 byte	int32	[target_num]	/	
DATA	4 byte	float	[x]	/	
DATA	4 byte	float	[y]	/	
DATA	4 byte	float	[z]	/	
DATA	4 byte	int32	[dop_idx]	/	
DATA	4 byte	Int32	[cluster_id]	/	
...	
DATA_CKSU	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。
 [x] : x 坐标，数据类型 float，单位: 米(m)。
 [y] : y 坐标，数据类型 float，单位: 米(m)。
 [z] : z 坐标，数据类型 float，单位: 米(m)。
- [dop_idx] : 数据类型 int32，速度 dop_idx。
 [cluster_id] : 数据类型 int32，聚类目标 ID。

注：当目标存在 N 个的时候，x、y、z、dop_idx、cluster_id 也存在 N 个。

注：此条协议 Z 轴输出为 0。

4.3.消息类型：报告相位测试结果 0x0A13

消息类型为 0x0A13，仅支持单向数据传输模式。

雷达发送数据给上位机					
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	用于输出总相位、心跳相位、呼吸相位结果。
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	0A 13	
HEAD_CHKSUM	1 byte	uint8	头校验和	\	
DATA	4 byte	float	[total phase]	\	
DATA	4 byte	float	[breath phase]	\	
DATA	4 byte	float	[heart phase]	\	
DATA_CHKSUM	1 byte	uint8	数据校验和	\	

4.4.消息类型：报告呼吸速率测试结果 0x0A14

消息类型为 0x0A14，仅支持单向数据传输模式。

雷达发送数据给上位机					
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	用于报告呼吸速率测试结果。
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	0A 14	
HEAD_CHKSUM	1 byte	uint8	头校验和	\	
DATA	4 byte	float	[rate]	\	
DATA_CHKSUM	1 byte	uint8	数据校验和	\	

4.5.消息类型：报告心跳速率测试结果 0x0A15

消息类型为 0x0A15，仅支持单向数据传输模式。

雷达发送数据给上位机					
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	用于报告心跳相位测试结果。
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	0A 15	
HEAD_CHKSUM	1 byte	uint8	头校验和	\	
DATA	4 byte	float	[rate]	\	
DATA_CHKSUM	1 byte	uint8	数据校验和	\	

4.6.消息类型：报告检测目标距离 0x0A16

消息类型为 0x0A16，仅支持单向数据传输模式。

雷达发送数据给上位机					
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	用于报告检测距离。
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	0A 16	
HEAD_CHKSUM	1 byte	uint8	头校验和	\	
DATA	4 byte	uint 32	[flag]	\	
DATA	4 byte	float	[range]	\	
DATA_CHKSUM	1 byte	uint8	数据校验和	\	

注：标志为 1 时，输出距离（单位：cm）标志为 0 时，不输出距离

4.7.消息类型：报告跟踪目标位置信息 0x0A17

消息类型为 0x0A17，仅支持单向数据传输模式。

雷达发送数据给上位机					
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	用于报告检测距离。
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	0A 17	
HEAD_CHKSUM	1 byte	uint8	头校验和	\	
DATA	4 byte	float	[x]	\	
DATA	4 byte	float	[y]	\	
DATA	4 byte	float	[z]	\	
DATA_CHKSUM	1 byte	uint8	数据校验和	\	

注：输出距离（单位：m）

5. 数据转换解析说明

A、以下是 DATA 位的数据转换：

转换成 float: 例如[rate]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 =
4.
5.     printf("data: %f\n", res);
6.     return 0;
7. }
```

B、以下是每个 CKSUM 的解析：

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.
4.     for (int i = 0; i < len;
5.         i++)
6.         ret = ret ^ data[i];
7.
8.
9.
10.    return ret;
11. }
```

6. 编程接口

6.1. 编码 TF 消息

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

其中 type 为发送数据类型，
uint16 类型 Uint8* data 是发
送数据的地址。

Len 为发送数据的长度，uint16 类型。

6.2.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。

6.3.示例代码

如果想要解析 TF 帧数据的 demo(包含 Linux 环境与 Keil μ Vision5 环境下的 C 语言 demo、Python 语言 demo)，可以直接与销售沟通获得，或直接从资料下载里下载《[TF_Demo.rar](#)》文件即可。

附录 A 文档修订记录

版本号	修订范围	日期
V1.0	初始文档	2023 年 10 月 18 日
V1.1	修改文档	2024 年 7 月 29 日
V1.2	增加协议内容及修改格式	2025 年 6 月 7 日