



24G 生物感知雷达
跌倒检测 - R24FD1B
User Manual (Ver. 1.1) 参考

云帆瑞达科技（深圳）有限公司 发布

目 录

概述.....	2
1. 工作原理.....	2
2. 硬件设计注意事项.....	2
2.1. 电源可参考以下电路设计.....	3
2.2. 使用接线图.....	3
3. 天线与外壳的布局要求.....	4
4. 静电防护.....	4
5. 功能干扰项.....	4
5.1 无人状态，异常输出有人.....	4
5.2 有人状态，异常输出无人.....	5
6. 功能详解.....	5
6.1. 功能点说明.....	5
6.2. 体动幅度参数输出说明.....	6
7. 协议说明.....	6
8. 通讯命令及参数定义.....	7
8.1. 帧结构定义及说明.....	7
8.2. 地址分配及数据信息说明.....	7
附录1：CRC 校验码参考解析代码.....	12
附录 2：运动体征参数解析代码.....	14
历史版本更新说明.....	15

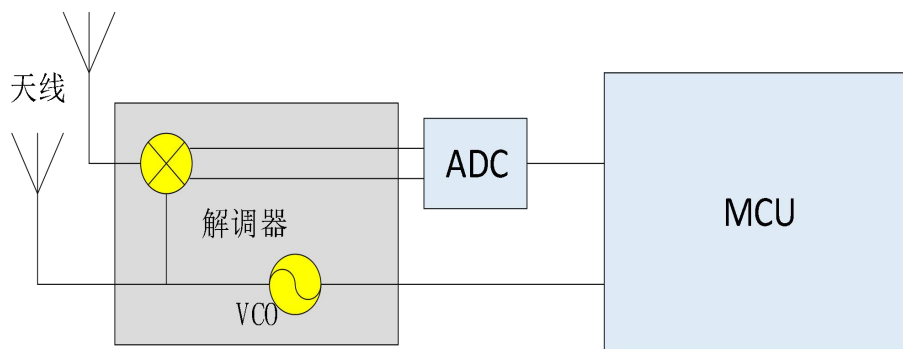
概述

本文档主要阐述该雷达使用事项，各个阶段需要注意的问题点，尽可能降低设计成本和增加产品的稳定性，提升项目的完成效率。

从硬件电路参考设计、雷达天线与外壳的布局要求、如何区分干扰和多功能的标准 UART 协议输出。

本雷达是一个自成体系的隔空感知传感器，由射频天线、雷达芯片和高速主频 MCU 一起组合而成的模组，依赖稳定灵活优越的算法架构核心，解决用户的各种场景探测需求，可搭载上位机或者主机灵活输出探测状态和数据，满足几组 GPIO 可供用户定制开发。

1. 工作原理



雷达发射 24G 频段毫米波信号，被测目标反射电磁波信号，并于发射信号进行解调处理，进而通放大、滤波、ADC 等处理，得到回波解调信号数据。在 MCU 单元对回波信号的幅度、频率、相位进行信息解算，最终实现目标参数（呼吸、运动、微动等）测量及场景评估。

2. 硬件设计注意事项

该雷达的额定供电电压需满足 4.9 - 6V，在正常工作情况下，额定电流要

求 200mA 以上的输入。电源设计，电源纹波需 $\leq 100\text{mv}$ 。

2.1. 电源可参考以下电路设计

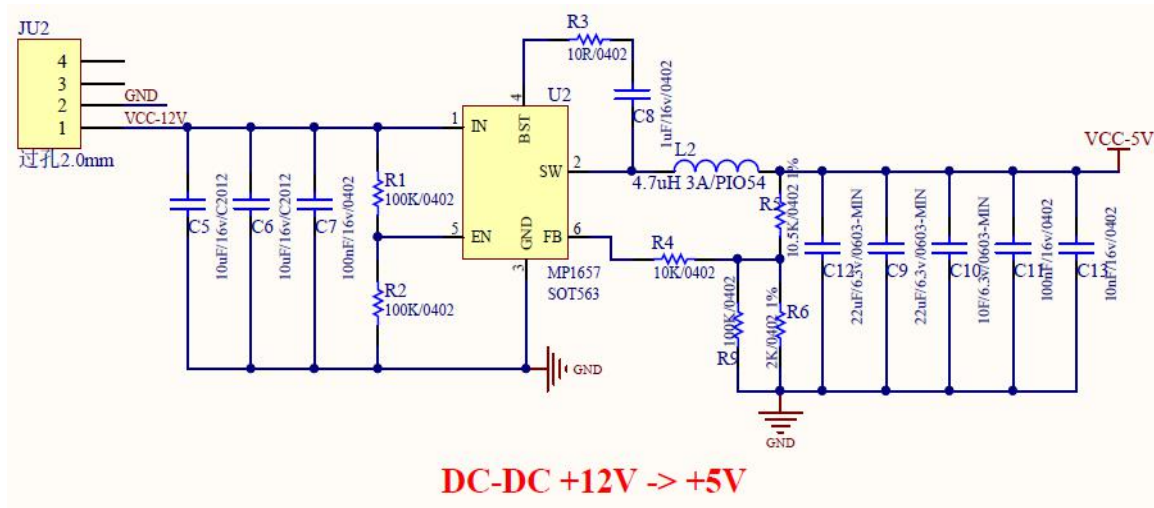


图 1

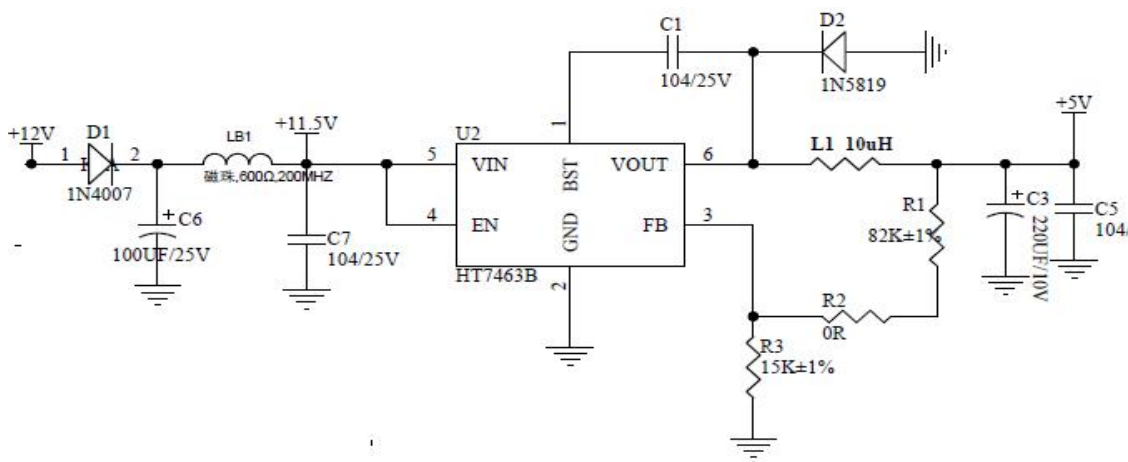


图 2

2.2. 使用接线图

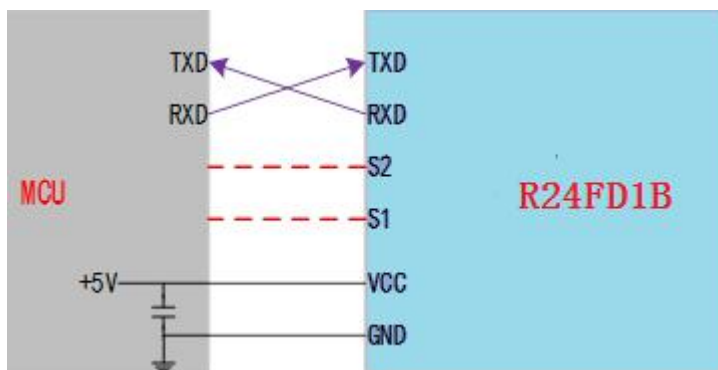


图 3 雷达模块与外设连线示意图

3. 天线与外壳的布局要求

PCBA: 需要保持雷达的贴件高度比其他器件 $\geq 1\text{mm}$

外壳结构: 需要保持雷达天线面和外壳面有 2 - 5mm 距离

外壳探测面: 非金属外壳, 需要平直, 避免弯曲面, 影响整个扫描面积的性能。

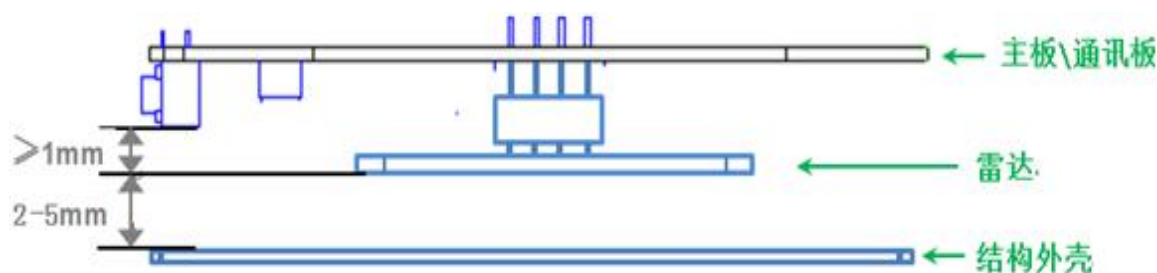


图 4

4. 静电防护

雷达产品内部具有静电敏感电路, 容易遭受静电危害, 因此需要在运输、存储、工作和拿取的过程中充分做好静电防护工作, 不要用手触摸抓取雷达模块天线表面和连接器管脚, 只能触摸其边角部分。

对雷达传感器进行操作时, 请尽量带上防静电手套。

5. 功能干扰项

5.1 无人状态, 异常输出有人

正常状态, 雷达会精准判断人体静坐状态和睡眠的存在, 并输出跌倒, 呼吸, 生命体征等信息

- A. 雷达扫描面积大, 门口, 木板墙的隔壁运动被探测到
调整方法: 降低雷达灵敏度, 雷达提供场景设置
- B. 雷达下方正对运行中空调, 风扇
调整方法: 调整雷达位置, 不要直接面对空调, 风扇
- C. 空调风引起的物体晃动
调整方法: 棉质, 非金属物品不会引起误报, 金属物品需要固定
- D. 雷达没有固定, 振动导致误报

避免支撑晃动，震动

- E. 宠物，飞鸟等偶尔运动物体
由于雷达测量微动，灵敏度很高，无法排除此项干扰
- F. 电源干扰，导致偶尔有误判
尽量保持供电电流稳定，减少纹波

5.2 有人状态，异常输出无人

雷达通过电磁波收发，判断人体存在。距离雷达越近，精度越高。

- A. 人体在雷达范围之外
雷达扫描范围，调整安装角度。雷达测量范围，在不同环境，电磁波反射面积不同，扫描面积会有微小差异。
- B. 金属遮挡造成错误输出
过厚的办公桌椅，金属座椅。会阻挡电磁波穿透，造成误判。
- C. 扫描角度差异
雷达没有扫描到躯干部位。造成误判。
- D. 雷达灵敏度过低
雷达提供参数调节，增加灵敏度改善。

6. 功能详解

6.1. 功能点说明

功能点	状态变化时间/功能解释
DP1: 有人/无人	无人到有人，0.5s 内上报 有人到无人，1-2 分钟左右输出无状态
DP2: 有人静止/有人活跃	静态动态切换，0.5 秒以内上报
DP3: 有人靠近设备/ 有人远离设备/有人无方向移动	2 秒输出一次状态
DP4: 体动幅度参数 0 - 100	5 秒输出一次数据 参考（体动幅度参数输出说明）

DP5: 灵敏度设置 0 - 9 档	默认场景模式下, 适配 10 个档位调节
DP7: 场景模式 (床位, 卫生间, 酒店, 卧室, 办公室, 默认模式)	按照面积大小, 适配不同的场景
DP8: 无人误报确认提示	
DP9: 跌倒开关	默认关闭, 开启时, 跌倒功能才生效
DP10: 跌倒状态	两级判断“疑似跌倒”、“跌倒告警” 疑似跌倒——检测到目标跌倒输出状态 跌倒告警——时间 T 静止后, 上报告警
DP11: 跌倒告警上报时间 T	默认 3 分钟, 分为 1-5 分钟设置档位
DP12: 静止驻留告警	分四次时间等级上报

6.2. 体动幅度参数输出说明

体动幅度参数		
0%	无人	环境无人
1%	静止 (睡眠)	只有呼吸而没有肢体运动
2%-30%	微动作	只有轻微头部或者肢体小运动
31%-60%	走动/快速肢体运动	比较慢速的身体移动
61%-100%	跑动/近距离大动作	快速身体移动

7. 协议说明

本协议应用于 24G 毫米波睡眠探测雷达与上位机之间的通信。

本协议概要介绍了雷达工作流程, 对接口协议组成架构进行了简单介绍, 并给出了相关雷达工作所需要控制命令及数据, 串口通信定义如下:

- ◇ 接口电平: TTL
- ◇ 波特率: 9600bps
- ◇ 停止位: 1
- ◇ 数据位: 8
- ◇ 奇偶校验: 无

8. 通讯命令及参数定义

8.1. 帧结构定义及说明

A、帧结构定义

起始码	数据长度		功能码	地址码 1	地址码 2	数据	校验码	
0X55	Lenth_L	Lenth_H	Command	Address_1	Address_2	Data	Crc16_L	Crc16_H
1 Byte	1 Byte	1 Byte	1 Byte	1 Byte	1 Byte	n Byte	1 Byte	1 Byte

B、帧结构说明

- a. 起始码：1Byte，固定为 0X55。
- b. 数据长度：2 Byte，低字节在前，高字节在后。
 ✚ 长度=数据长度+功能码+地址码 1+地址码 2+数据+校验码。
- c. 功能码：1Byte
 ✚ 读命令：0X01
 ✚ 写命令：0X02
 ✚ 被动上报命令：0X03 ✚
 主动上报命令：0X04
- d. 地址码：地址码 1 表示功能分类，地址码 2 表示具体功能。 ✚
 见地址分配及数据信息说明。
- e. 数据：n Byte
- f. 校验码：2 Byte，低字节在前，高字节在后。
 采用 CRC16 校验，参考代码见附录 1。

8.2. 地址分配及数据信息说明

24G 生物感知雷达接口内容					
	功能码	地址码 1	地址码 2	数据	备注
1	读命令 0x01	标识查询 0x01	设备 ID 0x01		
2			软件版本 0x02		
3			硬件版本 0x03		
4			协议版本 0x04		
5		雷达信息查询 0x03	环境状态 0x05		
6		体征参数 0x06			
7		系统参数查询 0x04	阈值档位 0x0C		
8		场景设置 0x10			

9		其他信息查询 0X05	跌倒功能开关 0X0B		查询 当前跌倒功能开关 状态
10			跌倒报警时间查询 0X0C		查询 当前跌倒报警时间
			跌倒灵敏度查询 0X0E		查询 当前跌倒灵敏度

24G 生物感知雷达接口内容

1	写命令 0x02	系统参数 0x04	阈值档位 0x0C	枚举范围 0~9	分别对应 0 1 2 3 4 5 6 7 8 9 档 (默认 6 档) 档 位越大, 越灵敏
2			场景设置 0x10	默认模式 0x00	
3				区域探测 (顶装) 0x01	
4				卫生间 (顶装) 0x02	
5				卧室 (顶装) 0x03	
6				客厅 (顶装) 0x04	
7				办公室 (顶装) 0x05	
8				酒店 (顶装) 0x06	
9		重启 0X04			
10		跌倒功能开关 0x0B	关 0x00 开 0x01		
		其他功能 0X05	跌倒灵敏度设置 0X0E	0X00 0 档	默认跌倒灵敏度为 6 档 档位越小, 越灵敏 档位越大, 越不灵 敏
	0X01 1 档				
	0X02 2 档				
	0X03 3 档				
	0X04 4 档				
	0X05 5 档				
	0X06 6 档				
	0X07 7 档				
	0X08 8 档				
	0X09 9 档				
11		开始 OTA 升级 0X08	4byte 整形数据 (固件包大小) + nbyte (软件版本号)		

12			升级包传输 0X09	包偏移 (4byte) + 数据包 (1024byte)	
13			升级结束信息 0X0A	固定字符 0X0F	

24G 生物感知雷达接口内容

1	被动上报命令 0x03	上报模块标识 0x01	设备 ID 0x01	12 Byte 数据	
2			软件版本 0x02	10 Byte 数据	
3			硬件版本 0x03	8 Byte 数据	
4			协议版本 0x04	8 Byte 数据	

24G 生物感知雷达接口内容

1	被动上报命令 0x03	上报雷达信息 0X03	环境状态 0x05	无人状态 00 FF FF	
2				有人静止 01 00 FF	
3				有人运动 01 01 01	
4			体征参数 0x06	4 Byte Float 数据	

24G 生物感知雷达接口内容

1	被动上报命令 0x03	上报系统信息 0X04	阈值档位 0X0C	当前档位值 (0X00~0X09)	
2			默认模式 0x00		
			区域探测 (顶装) 0x01		
			卫生间 (顶装) 0x02		
			卧室 (顶装) 0x03		
			客厅 (顶装) 0x04		
			办公室 (顶装) 0x05		
			酒店 (顶装) 0x06		

24G 生物感知雷达接口内容

1			反馈 OTA 升级开始 0X08	失败 0X00	
2				成功 0X01	

3	被动上报命令 0x03	上报其他信息 0X05	反馈 OTA 传输 0X09	固定字符 0X0F			
			跌倒功能开关 0X0B	关 0X00			
开 0X01							
4					跌倒报警时间 0X0C	1min 0X00	
						2min 0X01	
						3min 0X02	
						4min 0X03	
						5min 0X04	
						6min 0X05	
						7min 0X06	
	8min 0X07						
	9min 0X08						
	10min 0X09						
			跌倒灵敏度设置回 复 0X0E	0X00 0 档			
				0X01 1 档			
				0X02 2 档			
				0X03 3 档			
				0X04 4 档			
				0X05 5 档			
				0X06 6 档			
				0X07 7 档			
				0X08 8 档			
				0X09 9 档			

24G 生物感知雷达接口内容					
1	主动上报 命令 0X04	上报模块标识 0X01	软件版本 0X02		OTA 升级完成/重上 电都会上报一次
2		上报雷达信息 0x03	环境状态 0x05	无人状态 00 FF FF	
3				有人静止 01 00 FF	
4				有人运动 01 01 01	
5				运动体征参数 0X06	4Byte Float 数据
6		接近远离状态 0x07	固定字 符 0x01 0x01	无 0x01	
7				接近 0x02	
8				远离 0x03	
9		上报其他信息 0X05	心跳包 0X01	无人状态 00 FF FF	
10				有人静止 01 00 FF	
11				有人运动 01 01 01	
12				异常复位 0X02	0X0F

24G 生物感知雷达接口内容					
1	跌倒雷达数 据上报 0x06	报警 0x01	跌倒报警 0x01	疑似跌倒 0X00	报警时间节点: 5min/10min/30min/ 60min
				真实跌倒 0X01	
				无跌倒 0X02	
2		驻留报警 0X02	无 0x00		
			第一次 0X01		
			第二次 0X02		
			第三次 0X03		
			第四次 0X04		

- 说明:**
- 1) 读写命令为上位机向雷达发送指令。
 - 2) 上报命令为雷达向上位机发送信息。
 - 3) 跌倒灵敏度为 0~9 档，默认 6 档，档位越小越灵敏
 - 4) 人体灵敏度为 0~9 档，默认 6 档，档位越大越灵敏

附录 1: CRC校验码参考解析代码

```
1. const unsigned char cuc_CRCHI[256]=
2. {
3.     0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
4.     0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
5.     0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
6.     0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
7.     0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
8.     0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
9.     0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
10.    0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
11.    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
12.    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
13.    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
14.    0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
15.    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
16.    0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
17.    0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
18.    0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
19.    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
20.    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
21.    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
22.    0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
23.    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
24.    0x00, 0xC1, 0x81, 0x40
25. };
```

```
1. const unsigned char cuc_CRCLo[256]=
2. {
3.     0x00, 0xC0, 0xC1, 0x01, 0xC3, 0x03, 0x02, 0xC2, 0xC6, 0x06, 0x07, 0xC7,
4.     0x05, 0xC5, 0xC4, 0x04, 0xCC, 0x0C, 0x0D, 0xCD, 0x0F, 0xCF, 0xCE, 0x0E,
5.     0x0A, 0xCA, 0xCB, 0x0B, 0xC9, 0x09, 0x08, 0xC8, 0xD8, 0x18, 0x19, 0xD9,
6.     0x1B, 0xDB, 0xDA, 0x1A, 0x1E, 0xDE, 0xDF, 0x1F, 0xDD, 0x1D, 0x1C, 0xDC,
7.     0x14, 0xD4, 0xD5, 0x15, 0xD7, 0x17, 0x16, 0xD6, 0xD2, 0x12, 0x13, 0xD3,
8.     0x11, 0xD1, 0xD0, 0x10, 0xF0, 0x30, 0x31, 0xF1, 0x33, 0xF3, 0xF2, 0x32,
9.     0x36, 0xF6, 0xF7, 0x37, 0xF5, 0x35, 0x34, 0xF4, 0x3C, 0xFC, 0xFD, 0x3D,
10.    0xFF, 0x3F, 0x3E, 0xFE, 0xFA, 0x3A, 0x3B, 0xFB, 0x39, 0xF9, 0xF8, 0x38,
11.    0x28, 0xE8, 0xE9, 0x29, 0xEB, 0x2B, 0x2A, 0xEA, 0xEE, 0x2E, 0x2F, 0xEF,
12.    0x2D, 0xED, 0xEC, 0x2C, 0xE4, 0x24, 0x25, 0xE5, 0x27, 0xE7, 0xE6, 0x26,
13.    0x22, 0xE2, 0xE3, 0x23, 0xE1, 0x21, 0x20, 0xE0, 0xA0, 0x60, 0x61, 0xA1,
14.    0x63, 0xA3, 0xA2, 0x62, 0x66, 0xA6, 0xA7, 0x67, 0xA5, 0x65, 0x64, 0xA4,
15.    0x6C, 0xAC, 0xAD, 0x6D, 0xAF, 0x6F, 0x6E, 0xAE, 0xAA, 0x6A, 0x6B, 0xAB,
16.    0x69, 0xA9, 0xA8, 0x68, 0x78, 0xB8, 0xB9, 0x79, 0xBB, 0x7B, 0x7A, 0xBA,
17.    0xBE, 0x7E, 0x7F, 0xBF, 0x7D, 0xBD, 0xBC, 0x7C, 0xB4, 0x74, 0x75, 0xB5,
18.    0x77, 0xB7, 0xB6, 0x76, 0x72, 0xB2, 0xB3, 0x73, 0xB1, 0x71, 0x70, 0xB0,
19.    0x50, 0x90, 0x91, 0x51, 0x93, 0x53, 0x52, 0x92, 0x96, 0x56, 0x57, 0x97,
20.    0x55, 0x95, 0x94, 0x54, 0x9C, 0x5C, 0x5D, 0x9D, 0x5F, 0x9F, 0x9E, 0x5E,
21.    0x5A, 0x9A, 0x9B, 0x5B, 0x99, 0x59, 0x58, 0x98, 0x88, 0x48, 0x49, 0x89,
22.    0x4B, 0x8B, 0x8A, 0x4A, 0x4E, 0x8E, 0x8F, 0x4F, 0x8D, 0x4D, 0x4C, 0x8C,
23.    0x44, 0x84, 0x85, 0x45, 0x87, 0x47, 0x46, 0x86, 0x82, 0x42, 0x43, 0x83,
24.    0x41, 0x81, 0x80, 0x40
25. };

1. static unsigned shortint us_CalculateCrc16(unsigned char *lpuc_Frame, unsigned shortint lus_Len)
2. {
3.     unsigned char luc_CRCHi = 0xFF;
4.     unsigned char luc_CRCLo = 0xFF;
5.     int li_Index=0;
6.
7.     while(lus_Len--)
8.     {
9.         li_Index = luc_CRCLo ^ *(lpuc_Frame++);
10.        luc_CRCLo = (t_BYTE)( luc_CRCHi ^ cuc_CRCHi[li_Index]);
11.        luc_CRCHi = cuc_CRCLo[li_Index];
12.    }
13.    return (unsigned shortint )(luc_CRCLo << 8 | luc_CRCHi);
14. }
```

附录 2：运动体征参数解析代码

```
typedef union
{
    unsigned char Byte[4];
    float Float;
}Float_Byte;

void main()
{
    Float_Byte fb;
    fb.Byte[0] = 0x9A;
    fb.Byte[1] = 0xFB;
    fb.Byte[2] = 0xE7;
    fb.Byte[3] = 0x3F;
    printf("%f\r\n",fb.Float);
}
```

历史版本更新说明

Revision	Release Data	Summary
V1.0_0212	2020/02/12	初稿
V1.1_0319	2021/03/19	重新调整
V1.2_0528	2021/5/28	加上跌倒灵敏度调节
V1.3_0628	2021/6/28	加上 人体灵敏度的解释和 跌倒灵敏度的解释