

Beleaf System 主机

本地通讯协议

文档类别	<产品设计类>
文档编号	<>
版 本	<1.0>
状 态	< <input checked="" type="checkbox"/> 草稿 <input type="checkbox"/> 审核 <input type="checkbox"/> 正式>
密 级	< <input type="checkbox"/> 绝密 <input checked="" type="checkbox"/> 机密 <input type="checkbox"/> 普通>
作 者	<曾耸彬>

## 编辑历史

文件名称:					
Beleaf System 主机【本地通讯协议】					
文件说明:					
文件状态: [ ] 草稿 [ ] 正在修改 [✓] 正式发布	文件标识:	产品设计文档			
	适用平台:	嵌入式			
	适用项目:	Beleaf System			
	图例说明:			备注说明	
编辑历史:					
编辑时间	版本	作者	编辑内容		审核人
2022.02.11	V1.0	曾耸彬	初稿		
2022.03.11	V1.1	曾耸彬	将 IO 设备和其他设备分离, 单独命令, 减少名称占用字节空间		

## 目录

编辑历史.....	2
<b>1. 引言.....</b>	<b>6</b>
1.1. 编写目的.....	6
1.2. 文档范围.....	6
1.3. 术语和缩写.....	6
1.4. 网络拓扑图.....	7
1.5. 设计概要.....	8
<b>2. 类型.....</b>	<b>8</b>
2.1. 用途大类.....	8
2.2. 用途小类.....	9
2.3. 分组类型.....	10
2.4. 终端设备功能.....	10
2.5. 终端设备配置.....	11
<b>3. 主机管理（UDP）.....</b>	<b>12</b>
3.1. 主机广播.....	12
<b>4. 从机管理.....</b>	<b>12</b>
4.1. 配置同步.....	12
4.2. 从机注册.....	12
4.3. 从机更名.....	13
4.4. 从机心跳.....	13
4.5. 从机删除.....	13
<b>5. 分组管理.....</b>	<b>13</b>
5.1. 分组增加.....	13

5.2.	分组删除.....	14
5.3.	分组更名.....	14
5.4.	分组启动间隔更改.....	14
6.	传感器管理.....	14
6.1.	普通传感器注册.....	14
6.2.	多路 I0 输入传感器注册.....	15
6.3.	传感器更名.....	15
6.4.	传感器参数更名.....	16
6.5.	普通传感器更换分组.....	16
6.6.	传感器删除.....	16
6.7.	传感器定位.....	16
7.	终端设备管理.....	16
7.1.	普通终端设备注册.....	16
7.2.	多数 I0 输出终端设备注册.....	19
7.3.	普通终端设备配置.....	19
7.4.	终端设备更名.....	19
7.5.	普通终端设备更换分组.....	20
7.6.	终端设备功能更名.....	20
7.7.	终端设备配置更名.....	20
7.8.	多路 I0 输出终端设备端口更换分组.....	20
7.9.	普通终端设备手动控制.....	21
7.10.	多路 I0 输出设备端口手动控制.....	21
7.11.	终端设备删除.....	21
7.12.	终端设备定位.....	21
8.	传感器采集数据管理（UDP）.....	22

---

8.1.	传感器采集数据发送 .....	22
8.2.	虚拟传感器 .....	22
9.	终端设备状态（UDP） .....	22
9.1.	状态发送 .....	22
9.2.	虚拟设备 .....	22
10.	任务管理 .....	23
10.1.	梯形曲线 .....	23
10.2.	任务优先级 .....	23
10.3.	触发条件 .....	24
10.4.	设备任务 .....	24
10.5.	触发与运行 .....	24
10.6.	案例分析 .....	25

## 1. 引言

### 1.1. 编写目的

1、根据《Beleaf System 主机【需求规格说明书】》制定本地通讯协议，用于主机跟本地 Room-HUB（从机）设备的通讯。

2、作为软件设计、编码、测试的基础；

3、作为用户和软件开发人员在实现原理、流程、功能及应用等方面达成的共识，并依此作为将来成果验收的参考文档。

### 1.2. 文档范围

主要读者包括：项目负责人、项目软件设计人员、软件开发人员。

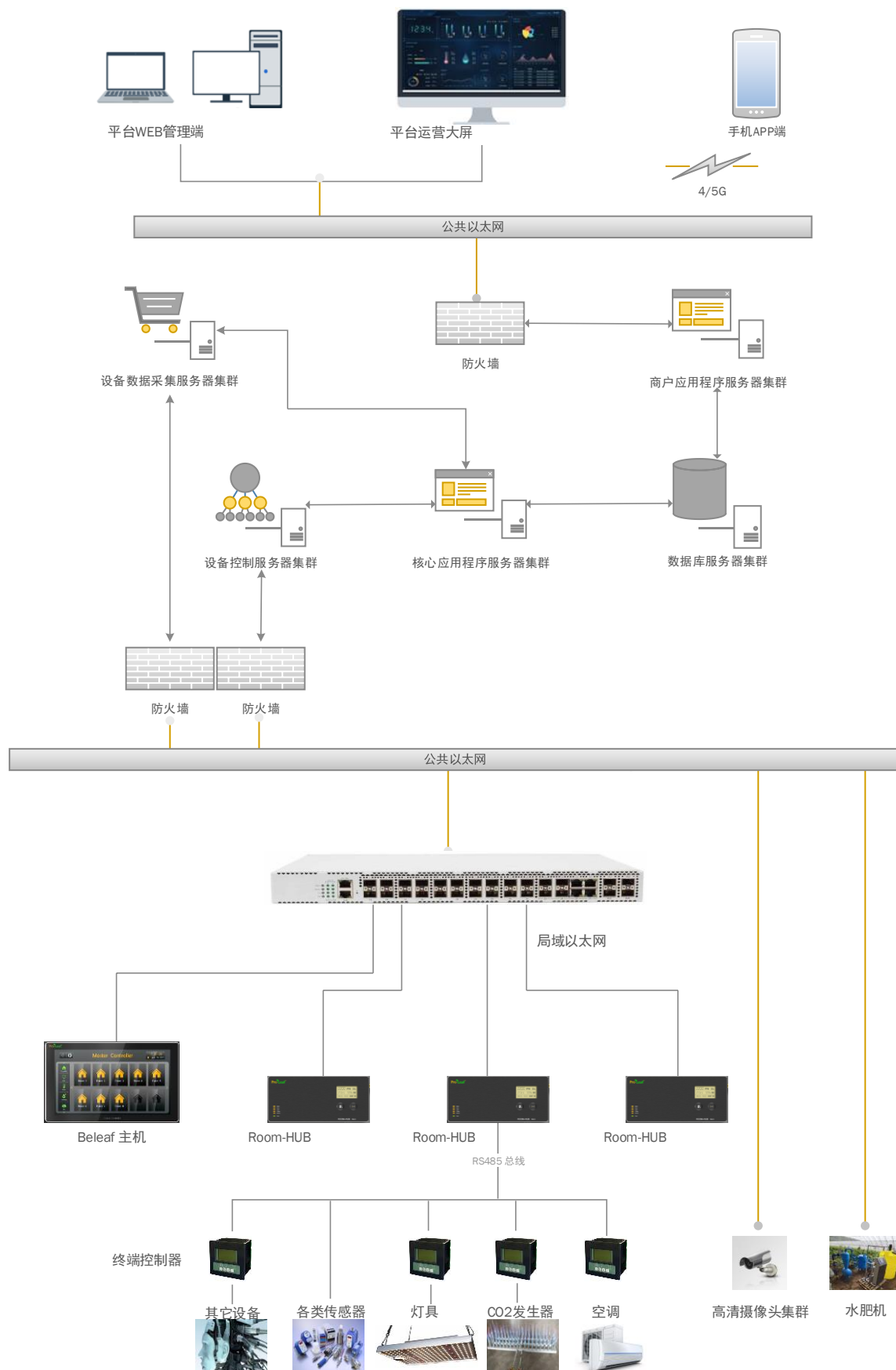
本文档重点是描述主机的需求，对其它部分不做详细描述。

### 1.3. 术语和缩写

提示：列出本文件中用到的专门术语的定义和外文首字母组词的原词组。

序号	术语名称	术语定义
1	Master Controller	主机
2	Room-HUB	从机
3	DTU	终端设备
4		

## 1.4. 网络拓扑图



## 1.5. 设计概要

主机跟本地 Room-HUB 设备（从机）的通讯协议采用 UDP 和 TCP 协议相结合方式，对于传感器数据，终端设备状态等非控制和配置类的数据，采用 UPD 协议。而对于控制和配置类数据，采用 TCP 通讯，保证数据的完整和交换成功。

表格中的功能描述默认都是十六进制数值写法。

无论是主机还是从机，无论是 UDP 还是 TCP，数据报头一致为：

名称	长度	功能
标识	2 bytes	开始标识主机包 AA BB, 从机包 AB CD, 方便编程
总长	2 byte	整个数据报长度, 不包含标识长度
CRC	2 bytes	数据报 CRC 校验, 从应答码开始到结束, 低位在前
应答码	1 byte	主动 0/应答 1, FF 代表功能码错误
功能码	1 byte	从机管理/传感器管理…等编号, 这边暂定文档第一级目录数字, 如从机管理, 代号为 4
	1 byte	从机管理/传感器管理…等具体操作编号, 如注册/删除等, 这边暂定文档第二级目录数字, 如从机管理下的从机更名, 代号为 3
发送者 ID	4 bytes	主机或者从机设备的 ID
序列号	4 bytes	非应答情况下按顺序递增, 应答情况下为原来主动包序列号

如果没有特殊说明，往后的数据报的格式描述，省略了报头描述，实际通信都需要加上报头。

如果没有特殊说明，默认都是 TCP。

## 2. 类型

## 2.1. 用途大类

[illegible]




## 2.2. 用途小类

名称	代号	功能	精度
自定义	0		
温度	0x0100		0.1 ° C
环境温度	0x0101		0.1 ° C
水体温度	0x0101		0.1 ° C
湿度	0x0200		
环境湿度	0x0201		1%
高度	0x0300		1mm
水位高度	0x0301		1mm
气体浓度	0x0400		1PPM
环境 CO2 浓度	0x0401		1PPM
烟雾浓度	0x0402		1PPM
水体浓度	0x0500		1PPM
PH 值	0x0501		1PPM
EC 值	0x0502		1PPM
流量	0x0600		
小管道流量	0x0601		ml/min
大管道流量	0x0602		m3/h
风速	0x0700		
光照	0x0800		
光照百分比	0x0801		%
时间	0x0900		
秒	0x0901		s
无量纲	0x0800		
烟雾报警	0x0801	0 正常/1 报警	无量纲
水位高度上限开关	0x0802	0 正常/1 到上限	无量纲
水位高度下限开关	0x0803	0 正常/1 到下限	无量纲
漏水探测	0x0804	0 正常/1 报警	无量纲
掉电报警	0x0805	0 正常/1 报警	无量纲
杂项	0xF000		
环境光敏	0xF001		
光量子通量密度 (PPFD)	0xF002		1 μ mol/m <sup>2</sup> • s

未定义 I0	0xF100		

### 2.3. 分组类型

名称	代号	功能
传感器	0x0001	
虚拟传感器	0x0002	
终端设备	0x0003	
虚拟终端设备	0x0004	

### 2.4. 终端设备功能

名称	代号	功能
自定义	0	
降温	1	
加热	2	
除湿	3	
加湿	4	
CO2 开关	5	
内循环通风	6	
外循环通风	7	
补光	8	
未定义 I0		

## 2.5. 终端设备配置

名称	代号	功能
自定义	0	
日升日落	1	
死区区间	2	
HID 冷却时间	3	

### 3. 主机管理（UDP）

#### 3.1. 主机广播

主机开机后，每隔 500ms 连续广播发送三次，之后每隔 10 秒发送一次，这样从机可以在任何时候获取主机 IP 地址和 ID 等信息

从机开机之后不发任何信息，直到获取主机 IP 地址之后，才能进行相关的数据通讯。

主机发送：

名称	长度	功能
时间戳	4 bytes	距离 2020 年 1 月 1 日 0 点多少秒
版本号	4 bytes	主机软件版本号，BCD 8421 编码，如 1.72.45 编译成 0x0000,0001,0002,0405，中间点号省略。

### 4. 从机管理

#### 4.1. 配置同步

为了代码逻辑上的简单，减少出错几率，做如下约定：

如果系统上有主机在线，那么从机不接受任何外部设备的配置修改，如手机 APP 通过蓝牙对从机的配置修改等。

如果系统上没有主机在线，后来接入主机，那么之前外部设备对从机的配置将清零，一切以主机配置为主。

主机在从机上线注册之后，如果有对从机的配置进行修改，那么第一次修改前会下发一个配置 ID 号给从机，从机需要记住这个配置 ID，在每次从机注册时发送该 ID 号，用于主机验证从机配置是否被修改过，如果主机发现配置 ID 跟主机上对该从机的配置 ID 不一致，主机将重新下发配置 ID 号，并附带清除配置标志，从机接收后，自己清除本机配置，等待重新接收主机的配置信息。

外部设备在对从机的配置修改前（主机一定是离线的），同样道理需要下发一个配置 ID 号给从机，从机需要记住这个配置 ID。

主机发送：

名称	长度	功能
配置 ID 号	4 bytes	
清除配置	1 byte	0 不清除/1 清除

从机应答，报头格式。

配置 ID 号是一个 32 位的随机数，每次上线后的第一次配置都会重新产生，原则上出现重复的概率非常小。

每次对从机配置时，会检查该配置会影响到哪些终端设备，如果该终端设备处于自动模式，需要提醒并弹出对话框让用户将涉及的终端设备改为手动模式，用户配置完成后，需要用户自己去修改终端设备是否运行在自动模式。

两次配置间隔超过一定时间，比如 1 分钟，需要用户重新输入配置密码。如果用户忘记配置密码，需要跟云平台联系，用手机号或者邮件等方式获取配置密码。

#### 4.2. 从机注册

从机在开机之后并成功获取主机 IP 和时间之后，第一时间向主机注册。

从机掉线后重新上线，第一时间向主机注册。

从机发送：

名称	长度	功能
版本号	4 bytes	从机软件版本号，BCD 8421 编码，如 1.72.45 编译成

		0x0000, 0001, 0002, 0405, 中间点号省略。
配置 ID 号	4 bytes	
心跳间隔	2 bytes	秒
从机名称	16 bytes	从机名称

主机应答，报头格式。

#### 4.3. 从机更名

主机发送：

名称	长度	功能
从机名称	16 bytes	从机名称

从机应答，报头格式。

#### 4.4. 从机心跳

从机发送，报头格式。

主机应答，报头格式。

当从机或者主机在一定时间范围内未收到心跳包或者心跳包回复，会认为对方已经离线，做离线处理工作。

#### 4.5. 从机删除

从机在开机之后并成功获取主机 IP 和时间之后，就会进行注册和心跳通信，当从机或者主机在一定时间范围内未收到心跳包或者心跳包回复，会认为对方已经离线，主机将从机状态移除到离线状态。

当从机离线情况下，主机主动每隔 7 天提醒用户是否删除该从机相关信息和设置，包括传感器和终端设备。

主机具备删除离线从机信息的功能。

主机具备一键清除离线设备，包括离线从机、离线传感器、离线终端设备的信息和设置功能。

### 5. 分组管理

主机可以对传感器、终端设备进行分组，只有同一从机上相同型号的传感器或者终端设备才能分到同一组，不同从机不能同组。

同组的传感器或者终端设备执行相同的动作。

分组的 ID 和名称由主机进行管理，并下发给相关从机。

#### 5.1. 分组增加

主机发送：

名称	长度	功能
分组 ID	4 bytes	
分组名称	16 bytes	
分组类型	2 bytes	如传感器、终端设备等
设备启动间隔	2 bytes	秒，传感器忽略，针对终端设备，同一分组的设备不是同时启动，而是间隔一定时间相继启动，防止电路电流瞬间过载。

从机应答，报头格式。

## 5.2. 分组删除

主机发送：

名称	长度	功能
分组 ID	4 bytes	

从机应答，报头格式。

## 5.3. 分组更名

主机发送：

名称	长度	功能
分组 ID	4 bytes	
分组名称	16 bytes	

从机应答，报头格式

## 5.4. 分组启动间隔更改

主机发送：

名称	长度	功能
分组 ID	4 bytes	
启动间隔	2 bytes	秒

从机应答，报头格式。

# 6. 传感器管理

## 6.1. 普通传感器注册

从机开机后如果没有对普通传感器进行注册，必须向主机注册普通传感器。

从机如果掉线之后重新上线，必须向主机重新注册普通传感器。

从机发送：

名称	长度	功能
传感器 ID	4 bytes	
传感器名称	16 bytes	传感器名称
传感器型号	2 bytes	类似产品型号，不同产品必须不同类型，用于分组判断，只有同型号的产品才能分到同一组
采集数据发送频率	2 bytes	秒
传感器分组 ID	4 bytes	如果没有分组，默认为 0
参数用途大类	2 bytes	0 未知，数值参见参数用途表格，如大气环境控制、土壤环境控制、水体环境控制等。
传感器参数数量 n	1 byte	如 3 合一的环境传感器，具备温度、湿度、光敏三个参数的测量，那数值就是 3
传感器参数 1 用途小类	2 bytes	传感器参数的具体用途，如环境温度等，用途确定，默认的单位就确定，比如温度，那么温度的单位为 0.1 摄氏度等，从机必须按照代号的单位发送给主机
传感器参数 1 名称	16 bytes	

...		
传感器参数 n 用途小类	2 bytes	如果有的话
传感器参数 n 名称	16 bytes	

主机应答：

名称	长度	功能
传感器 ID	4 bytes	

## 6.2. 多路 IO 输入传感器注册

从机开机后如果没有对多路 IO 输入传感器进行注册，必须向主机注册多路 IO 输入传感器。

从机如果掉线之后重新上线，必须向主机重新注册多路 IO 输入传感器。

从机发送：

名称	长度	功能
传感器 ID	4 bytes	
传感器名称	16 bytes	传感器名称
传感器型号	2 bytes	类似产品型号，不同产品必须不同类型
采集数据发送 频率	2 bytes	秒
IO 输入端口 数量 n	1 byte	如 12 个输入端口，那数值就是 12
端口 1 用途小 类	2 bytes	未知为 0，如果知道，填写参数用途小类
端口 1 名称	16 bytes	
...		
端口 n 用途小 类	2 bytes	如果有的话
端口 n 名称	16 bytes	

主机应答：

名称	长度	功能
传感器 ID	4 bytes	

## 6.3. 传感器更名

可以在主机上对传感器进行更名。

主机发送：	长度	功能
名称		
传感器 ID	4 bytes	
传感器名称	16 bytes	传感器名称

从机应答：

名称	长度	功能
传感器 ID	4 bytes	

#### 6.4. 传感器参数更名

可以在主机上对传感器上的某个参数进行更名。

主机发送：

名称	长度	功能
传感器 ID	4 bytes	
参数在注册时的位置	1 byte	从 1 开始
参数名称	16 bytes	

从机应答：

名称	长度	功能
传感器 ID	4 bytes	

#### 6.5. 普通传感器更换分组

可以在主机上对传感器分组进行重新分配。

主机发送：

名称	长度	功能
传感器 ID	4 bytes	
传感器分组 ID	4 bytes	如果 ID 号为 0，代表该传感器被移出分组

从机应答，报头格式。

#### 6.6. 传感器删除

传感器在 3 倍的采集时间内没有任何数据通讯，主机就认为传感器掉线，将传感器状态移除到离线状态。

当传感器离线 7 天情况下，主机主动删除该传感器相关信息和设置。

当传感器离线情况下，主机主动每隔 7 天提醒用户是否删除该传感器相关信息和设置。

主机具备单独的删除离线传感器信息的功能。

主机具备一键清除离线传感器功能。

#### 6.7. 传感器定位

主机上对传感器发送指令，要求传感器收到指令后进行相应的提示，如声光等，让管理人员可以找到。

主机发送：

名称	长度	功能
传感器 ID	4 bytes	
参数在注册时的位置	1 byte	默认 0，如果是多路 I/O 输入设备，则对应各个输入端口

从机应答，报头格式。

### 7. 终端设备管理

#### 7.1. 普通终端设备注册

从机开机后如果没有对普通终端设备进行注册，必须向主机注册普通终端设备。

从机如果掉线之后重新上线，必须向主机重新注册普通终端设备。

从机发送：



名称	长度	功能
设备 ID	4 bytes	
设备名称	16 bytes	
设备类型	2 bytes	类似产品型号，不同产品必须不同类型，用于分组，只有同型号的产品才能分到同一组
设备分组 ID	4 bytes	如果没有分组，默认为 0
设备状态发送频率	2 bytes	秒
参数用途大类	2 bytes	0 未知，数值参见参数用途表格，如大气环境控制、土壤环境控制、水体环境控制等。
设备功能数量 n	1 byte	如降温、加热、除湿，那就是 3 个功能，数值就是 3
设备功能 1 代号	2 bytes	如降温代号，如果功能未定义，填 0，由用户自己定义
设备功能 1 用途小类	2 bytes	类似确定参数单位，设备功能的具体用途，如环境温度等，用途确定，默认的单位就确定，比如温度，那么温度的单位为 0.1 摄氏度等
设备功能 1 名称	16 bytes	如果功能代号 0，默认字符
设备功能 1 参数下限	4 bytes	0 关闭/1 挡/2 挡.../X 挡位，如果只是开关，那 X=1，0 关/1 开，如果是 PPFD，可以设置到上千
设备功能 1 参数上限	4 bytes	0 关闭/1 挡/2 挡.../X 挡位，如果只是开关，那 X=1，0 关/1 开，如果是 PPFD，可以设置到上千
...		
设备功能 n 代号	2 bytes	如果有的话
设备功能 n 名称	16 bytes	
设备功能 n 参数下限	4 bytes	
设备功能 n 参数上限	4 bytes	
设备配置数量 m	1 byte	
设备配置 1 代号	2 bytes	如日升日落，死区区间等
设备配置 1 用途小类	2 bytes	设备配置的具体用途，如环境温度等，用途确定，默认的单位就确定，比如温度，那么温度的单位为 0.1 摄氏度等
设备配置 1 名称	16 bytes	
设备配置 1 下限	4 bytes	
设备配置 1 上限	4 bytes	
...		
设备配置 m 代号	2 bytes	如果有的话
设备配置 m 用途小类	2 bytes	类似确定配置单位，
设备配置 m 名称	16 bytes	
设备配置 m 下限	4 bytes	
设备配置 m 上限	4 bytes	

主机应答：

名称	长度	功能
设备 ID	4 bytes	

## 案例 1

HVAC 终端设备注册：

名称	长度	功能
设备 ID	4 bytes	0x12345678
设备名称	16 bytes	“HVAC”
设备类型	2 bytes	0x79DF
设备分组 ID	4 bytes	0

设备状态发送频率	2 bytes	10 秒
参数用途大类	2 bytes	1 (室内环境控制)
设备功能数量 n	1 byte	3 (降温, 加热, 通风)
设备功能 1 代号	2 bytes	1 (降温)
设备功能 1 用途小类	2 bytes	0x0101 (环境温度)
设备功能 1 名称	16 bytes	“cool”
设备功能 1 参数下限	4 bytes	0 关闭
设备功能 1 参数上限	4 bytes	5 假设总共 5 档
设备功能 2 代号	2 bytes	2 (加热)
设备功能 2 用途小类	2 bytes	0x0101 (环境温度)
设备功能 2 名称	16 bytes	“heat”
设备功能 2 参数下限	4 bytes	0 关闭
设备功能 2 参数上限	4 bytes	3 假设总共 3 档
设备功能 3 代号	2 bytes	6 (内循环通风)
设备功能 3 用途小类	2 bytes	0x0101 (环境温度)
设备功能 3 名称	16 bytes	“fan”
设备功能 3 参数下限	4 bytes	0 关闭
设备功能 3 参数上限	4 bytes	5 假设总共 5 档

## 案例 2

## LDA 终端设备注册:

名称	长度	功能
设备 ID	4 bytes	0x46289214
设备名称	16 bytes	“LDA”
设备类型	2 bytes	0x749D
设备分组 ID	4 bytes	0
设备状态发送频率	2 bytes	10 秒
参数用途大类	2 bytes	6 (室内光照控制)
设备功能数量 n	1 byte	3 (红光, 白光, PPF)
设备功能 1 代号	2 bytes	8 (补光)
设备功能 1 用途小类	2 bytes	0x0801 (光照百分比)
设备功能 1 名称	16 bytes	“red”
设备功能 1 参数下限	4 bytes	0 关闭
设备功能 1 参数上限	4 bytes	100
设备功能 2 代号	2 bytes	8 (补光)
设备功能 2 用途小类	2 bytes	0x0801 (光照百分比)
设备功能 2 名称	16 bytes	“white”
设备功能 2 参数下限	4 bytes	0 关闭
设备功能 2 参数上限	4 bytes	100
设备功能 3 代号	2 bytes	8 (补光)
设备功能 3 用途小类	2 bytes	0xF002 (PPF)
设备功能 3 名称	16 bytes	“PPF”
设备功能 3 参数下限	4 bytes	0 关闭
设备功能 3 参数上限	4 bytes	2000
设备配置数量	1 byte	1
设备配置 1 代号	2 bytes	1 (日升日落)
设备配置 1 用途小类	2 bytes	0x0901 (时间, 秒)
设备配置 1 名称	16 bytes	“sunrise&sunset”
设备配置 1 下限	4 bytes	0

设备配置 1 上限	4 bytes	3600
-----------	---------	------

## 7.2. 多数 I/O 输出终端设备注册

从机开机后如果没有对多数 I/O 输出终端设备进行注册，必须向主机注册多数 I/O 输出终端设备。  
从机如果掉线之后重新上线，必须向主机重新注册多数 I/O 输出终端设备。

从机发送：

名称	长度	功能
设备 ID	4 bytes	
设备名称	16 bytes	
设备类型	2 bytes	类似产品型号，不同产品必须不同类型
设备状态发送频率	2 bytes	秒
端口数量 n	1 byte	如 I/O 输出有 12 个，那就填 12
端口 1 功能代号	2 bytes	如降温代号，如果功能未定义，填 0，由用户自己定义，如 I/O 输出
端口 1 名称	16 bytes	如果功能代号 0，默认字符
端口 1 分组 ID	4 bytes	
...		
端口 n 功能代号	2 bytes	如果有的话
端口 n 名称	16 bytes	
端口 n 分组 ID	4 bytes	

主机应答：

名称	长度	功能
设备 ID	4 bytes	

## 7.3. 普通终端设备配置

经过注册后的普通终端设备，可以根据主机的设置，配置相关参数。

主机发送：

名称	长度	功能
从机 ID	4 bytes	
设备 ID	4 bytes	
配置在注册时的位置	1 byte	
配置参数	2 bytes	

从机机应答：

名称	长度	功能
终端设备 ID	4 bytes	

## 7.4. 终端设备更名

可以在主机上对终端设备进行更名。

主机发送：	长度	功能
名称		
终端设备 ID	4 bytes	
终端设备名称	16 bytes	

从机机应答：

名称	长度	功能
----	----	----

终端设备 ID	4 bytes	
---------	---------	--

### 7.5. 普通终端设备更换分组

可以在主机上对终端设备上的分组进行重新分配。

主机发送：

名称	长度	功能
终端设备 ID	4 bytes	
终端设备分组 ID	4 bytes	如果 ID 号为 0，代表该终端设备被移出分组
分组名称	16 bytes	

从机应答，报头格式。

### 7.6. 终端设备功能更名

可以在主机上对终端设备功能进行更名。

主机发送：	长度	功能
名称		
终端设备 ID	4 bytes	
功能在注册时的位置	1 byte	
功能名称	16 bytes	

从机机应答：

名称	长度	功能
终端设备 ID	4 bytes	

### 7.7. 终端设备配置更名

可以在主机上对终端设备配置进行更名。

主机发送：

名称	长度	功能
终端设备 ID	4 bytes	
配置在注册时的位置	1 byte	
配置名称	16 bytes	

从机机应答：

名称	长度	功能
终端设备 ID	4 bytes	

### 7.8. 多路 I/O 输出终端设备端口更换分组

可以在主机上对多路 I/O 输出终端设备上的分组进行重新分配。

主机发送：

名称	长度	功能
终端设备 ID	4 bytes	
端口位置	1 byte	
分组 ID	4 bytes	

从机应答，报头格式。

### 7.9. 普通终端设备手动控制

当主机发送普通设备终端控制指令之后，主机要将该设备转到手动控制状态并将跟该设备有关自动控制的配置停止运行。

主机发送：

名称	长度	功能
终端设备 ID	4 bytes	
设备功能位置	1 byte	
是否自动	1 byte	0 手动/1 自动
设备功能参数	4 bytes	只对手动有效

从机机应答：

名称	长度	功能
终端设备 ID	4 bytes	

### 7.10. 多路 I/O 输出设备端口手动控制

当主机发送多路 I/O 输出终端端口控制指令之后，主机要将该设备端口转到手动控制状态并将跟该设备端口有关自动控制的配置停止运行。

主机发送：

名称	长度	功能
终端设备 ID	4 bytes	
端口位置	1 byte	
是否自动	1 byte	0 手动/1 自动
功能动作	1 byte	只对手动有效，0 停止/ 1 运行

从机机应答：

名称	长度	功能
终端设备 ID	4 bytes	

### 7.11. 终端设备删除

终端设备在 3 倍状态发送时间间隔内没有状态信息发送，就认为终端设备掉线，将终端设备状态移除到离线状态。

当终端设备离线情况下，主机主动每隔 7 天提醒用户是否删除该终端设备相关信息和设置。

主机具备单独的删除离线终端设备信息的功能。

主机具备一键清除离线终端设备功能。

### 7.12. 终端设备定位

主机上对终端设备发送指令，要求终端设备收到指令后进行相应的提示，如声光等，让管理人员可以找到。

主机发送：

名称	长度	功能
终端设备 ID	4 bytes	
功能在注册时的位置	1 byte	默认 0，如果是多路 I/O 输出设备，则对应各个输出端口

从机应答，报头格式。

## 8. 传感器采集数据管理（UDP）

### 8.1. 传感器采集数据发送

根据传感器注册时要求，按一定频率、按各参数位置发送数据。

从机发送：

名称	长度	功能
传感器 ID	4 bytes	
传感器参数数量 n	1 byte	
传感器参数 1 数据	4 bytes	
...		
传感器参数 n 数据	4 bytes	如果有的话

主机不应答

### 8.2. 虚拟传感器

根据设定，同一分组的传感器是同一个型号的，数据格式是一样的，主机将对分组数据进行平均处理，处理结果将用于控制和显示。同时对比传感器间的数据差异，找出可能出现故障的传感器，提示管理人员。

对此，只要分组有两个及以上传感器，就会虚拟出一个传感器，该虚拟传感器的数据就是各传感器的算术平均值。

虚拟传感器也可以是不同传感器但同属性之间的算术运算值虚拟出来，比如温度等

## 9. 终端设备状态（UDP）

### 9.1. 状态发送

根据终端设备注册时要求，按一定频率、按各功能位置发送终端设备状态。

从机发送：

名称	长度	功能
设备 ID	4 bytes	
设备功能数量 n	1 byte	如降温、加热、除湿，那就是 3 个功能，数值就是 3，如 I0 输出有 12 个，那就填 12
设备功能 1 状态	2 bytes	0 停止/1 运行
设备功能 2 状态	2 bytes	如果有的话
...		
设备功能 n 状态	2 bytes	如果有的话

主机不应答

### 9.2. 虚拟设备

根据设定，同一分组的设备是同一个型号的，只要分组有两个及以上设备，就会虚拟出一个设备，主要用途是虚拟设备的功能状态，当分组内所有设备的某个功能都停止时，虚拟设备该功能状态才为 0，否则为 1。

## 10. 任务管理

### 10.1. 梯形曲线

终端设备必须在某个时刻满足一定的条件之后，根据一定的功能设定值进行运行，这些条件或者功能设定值是跟时间相关的，连续的设定值 Y 轴和时间长度 X 轴构成了一条梯形曲线，我们用字符串来表达这条曲线。

主机发送：

名称	长度	功能
梯形曲线 ID	4 bytes	
梯形曲线表达式	256 bytes	

从机应答，报头格式。

梯形曲线表达式我们用字符串表达，格式规定为：

[曲线区间数目, (起始值, 终点值, 持续时长), (起始值, 终点值, 持续时长), ..., (起始值, 终点值, 持续时长)]

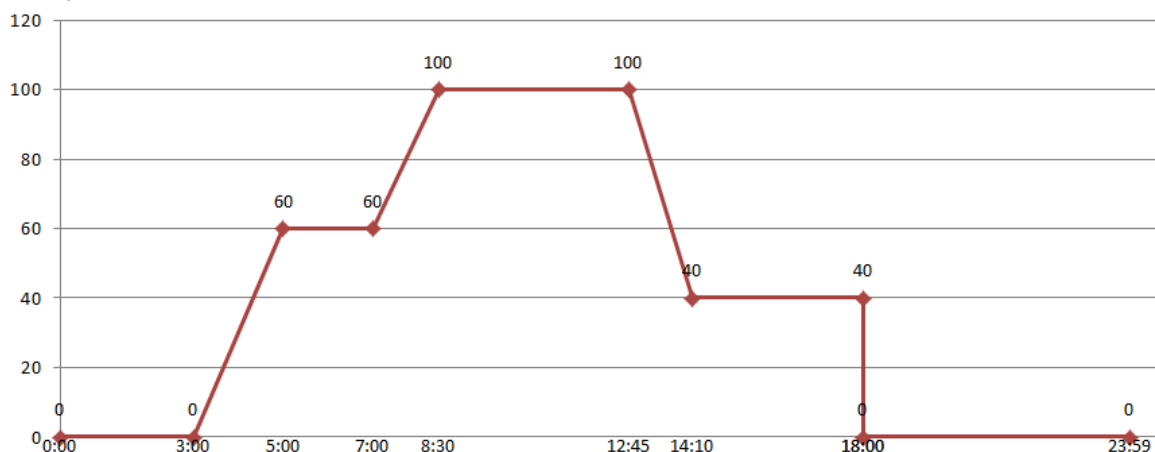
曲线区间数目：构成梯形曲线的直线数量

起始值：每段直线左端点的 Y 轴坐标值

终点值：每段直线右端点的 Y 轴坐标值

持续时长：单位秒，不是时刻，如果持续时长为 0，代表没有时间限制。

比如：



总共有 8 段曲线，假设它的 ID 号为 1，表达式为：

[8, (0, 0, 10800), (0, 60, 7200), (60, 60, 7200), (60, 100, 5400), (100, 100, 15300), (100, 40, 5100), (40, 40, 13800), (0, 0, 21600)]

如上图的第一段直线，时长 3 小时，即 10800 秒。

系统默认已经有两条曲线[1, (0, 0, 0)], ID 号为 0, 和[1, (1, 1, 0)], ID 号为 1

### 10.2. 任务优先级

每个任务都有个优先级，当优先级高的任务条件成立触发后并对终端设备发出控制指令，如果有低优先级的任务试图对同一终端设备相同功能发出控制指令，将不再执行低优先级任务。也就是高优先级任务屏蔽低优先级任务。

任务优先级分为四级 1, 2, 3, 4：最高优先级 1 为紧急事件，2 为定时器事件，3 为优先件级，4 为普通事件。

### 10.3. 触发条件

将某些事件作为触发条件，进行相关的设备动作，如

主机发送：

名称	长度	功能
触发条件 ID	4 bytes	
触发条件优先级	1 byte	1/2/3/4
触发条件表达式	128 bytes	

从机应答，报头格式。

触发条件表达式规定为：

((输入参数)关系运算符(#梯形曲线 ID#)) [&&...&&((输入参数)关系运算符(#梯形曲线 ID#))]

[]代表可选。

其中“输入参数”可能是：

1. “0”
2. “1”
3. “\$传感器 ID:传感器在注册时位置\$”
4. “\$虚拟传感器 ID:传感器在注册时位置\$”
5. “\$设备 ID:设备功能在注册时位置\$”
6. “\$虚拟设备 ID: 设备功能在注册时位置\$”

可以为同一设备的同一功能设置多个触发条件，触发条件按照任务优先级依次执行，当某个表达式运算结果为“真”时，触发成功，执行任务，之后继续判断后面的触发条件，如果控制的设备功能已经触发过，则跳过不判断和执行。

### 10.4. 设备任务

设定某个终端设备的某个功能的运行曲线

主机发送：

名称	长度	功能
设备任务 ID	4 bytes	
设备或分组 ID	4 bytes	
触发功能在注册时的位置	1 byte	
梯形曲线 ID	4 bytes	

从机应答，报头格式。

### 10.5. 触发与运行

将触发条件 ID 和设备任务 ID 关联起来，这样当触发条件成立之后，相应的设备能够根据设定曲线动作。

主机发送：

名称	长度	功能
----	----	----



设备运行 ID	4 bytes	
触发条件 ID	4 bytes	
设备任务 ID	4 bytes	
开始时间	4 bytes	秒，任务开始时间，从 2020 年 1 月 1 日 0 点 00 分 0 秒开始算
持续时间	4 bytes	秒，任务运行时间，梯形曲线反复执行，直达运行时间结束，0 表示不限时长
任务延时	2 bytes	秒，当触发条件成立后，延时一点时间后再执行，可用于防止抖动等

从机应答，报头格式。

## 10.6. 案例分析

### 案例 1:

从 2022 年 3 月 3 日 14 点 03 分 56 秒开始，每隔 47 小时执行 1 小时的灌溉，也就是每两天灌溉一次，无限循环

1. 定义梯形曲线 ID 3，表达式为：[2, (0, 0, 169200), (1, 1, 3600)]，即 47 小时（169200 秒）停止，1 小时（3600 秒）工作
2. 定义触发条件 ID 4，优先级为 2，表达式为：(1) == (梯形曲线 ID 3)
3. 定义触发条件 ID 5，优先级为 2，表达式为：(0) == (梯形曲线 ID 3)
4. 定义设备任务 ID 6，假如控制灌溉电磁阀的 IO 设备 ID 为 7，使用端口 1（注册功能），因为电磁阀只有开和关，我们可以复用梯形曲线 ID 1
5. 定义设备运行 ID 8，将触发条件 ID 4 和设备任务 ID 6 关联起来，开始时间为 2022 年 3 月 3 日 14 点 03 分 56 秒（68479436 秒），持续时间为 0，不停循环。
6. 定义设备运行 ID 9，将触发条件 ID 5 和设备任务 ID 6 关联起来，开始时间为 2022 年 3 月 3 日 14 点 03 分 56 秒（68479436 秒），持续时间为 0，不停循环。

当触发条件 ID 4 成立，说明当前时刻梯形曲线 ID 3 的值为 1，执行设备运行 ID 8，将电磁阀打开。

当触发条件 ID 5 成立，说明当前时刻梯形曲线 ID 3 的值为 0，执行设备运行 ID 9，将电磁阀关闭。

### 案例 2:

A 白天（6:00~18:00）运行，夜晚（0~6:00, 18:00~24:00）不运行

B 当 CO2 大于 500PPM 时，停止 CO2 发生器。

C 当 CO2 小于 200PPM 时，启动 CO2 发生器。

1. 使用系统梯形曲线 ID 0，表达式为：[1, (0, 0, 0)]，值为 0 的一条直线，控制 CO2 发生器关
2. 使用系统梯形曲线 ID 1，表达式为：[1, (1, 1, 0)]，值为 1 的一条直线，控制 CO2 发生器开
3. 定义梯形曲线 ID 3，表达式为：[3, (0, 0, 21600), (1, 1, 43200), (0, 0, 21600)]，用于定时器。
4. 定义梯形曲线 ID 4，表达式为：[1, (500, 500, 0)]，值为 500 的一条直线。
5. 定义梯形曲线 ID 5，表达式为：[1, (200, 200, 0)]，值为 200 的一条直线
6. 定义触发条件 ID 6，优先级为 2，表达式为：(0) == (梯形曲线 ID 3)
7. 定义触发条件 ID 7，优先级为 3，假设 CO2 浓度传感器的 ID 为 24，参数位置为 1，表达式为：(\$24:1\$) > (梯形曲线 ID 4)
8. 定义触发条件 ID 8，优先级为 3，表达式为：(\$24:1\$) < (梯形曲线 ID 5)
9. 定义设备任务 ID 9，假设控制 CO2 发生器的 IO 设备 ID 为 26，使用端口 1（注册功能），关联梯形曲线 ID 0
10. 定义设备任务 ID 10，假设控制 CO2 发生器的 IO 设备 ID 为 26，使用端口 1（注册功能），关联梯形曲线 ID 1

11. 定义设备运行 ID 11, 将触发条件 ID 6 和设备任务 ID 9 关联起来, 开始时间为 2020 年 1 月 1 日 0 点 0 分 0 秒 (0 秒), 持续时间为 0, 不停循环。
12. 定义设备运行 ID 12, 将触发条件 ID 7 和设备任务 ID 9 关联起来, 开始时间为 2020 年 1 月 1 日 0 点 0 分 0 秒 (0 秒), 持续时间为 0, 不停循环。
13. 定义设备运行 ID 13, 将触发条件 ID 8 和设备任务 ID 10 关联起来, 开始时间为 2020 年 1 月 1 日 0 点 0 分 0 秒 (0 秒), 持续时间为 0, 不停循环。

当触发条件 ID 6 成立, 说明当前时刻是黑夜, 设备运行 ID 11 可执行, 执行设备任务 ID 9, 运行梯形曲线 ID 0, 关闭 CO2 发生器。

当触发条件 ID 7 成立, 说明当前时刻 CO2 大于 500PPM, 执行设备运行 ID 12, 执行设备任务 ID 9, 运行梯形曲线 ID 0, 关闭 CO2 发生器, 停止补充 CO2。

当触发条件 ID 8 成立, 说明当前时刻 CO2 小于 200PPM, 设备运行 ID 13 可执行, 执行设备任务 ID 10, 运行梯形曲线 ID 1, 启动 CO2 发生器, 补充 CO2。

至于 CO2 发生器如何工作, 由控制 CO2 发生器的 IO 设备去控制, 例如使用 PID 算法等。

### 案例 3:

A LDA 灯光控制器有 4 个模拟量输出, 两两一组, 分成两组, 同一组的端口分别控制白和红两种灯光。

B 白天 (6:00~18:00) 用第一组, 黑夜 (0~6:00, 18:00~24:00) 用第二组。

C 要求白天上午 6:00~12:00 点 白光 80%, 红光 70%, 白天下午白光 60%, 红光 90%

1. LDA 上报两个数据包, 分别注册两个设备 ID 28 和 29, 每个 ID 注册两个功能 (白和红) 和一个属性 (日升日落)
2. 这边只讲白天的控制设备 ID 28, 黑夜的类似
3. 先设置设备 ID 28 的属性, 日升日落时间, 日升日落逻辑由 LDA 处理。
4. 定义梯形曲线 ID 2, 准备给白光, 表达式为 :  
[4, (0, 0, 21600), (80, 80, 21600), (60, 60, 21600), (0, 0, 21600)]。
5. 定义梯形曲线 ID 3, 准备给红光, 表达式为 :  
[4, (0, 0, 21600), (70, 70, 21600), (90, 90, 21600), (0, 0, 21600)]。
6. 定义触发条件 ID 4, 优先级为 3, 表达式为: (1) = (系统梯形曲线 ID 1)
7. 定义设备任务 ID 5, LDA 设备 ID 为 1, 使用功能 1 (白光), 关联梯形曲线 ID 2
8. 定义设备任务 ID 6, LDA 设备 ID 为 1, 使用功能 2 (红光), 关联梯形曲线 ID 3
9. 定义设备运行 ID 7, 将触发条件 ID 4 和设备任务 ID 5 关联起来, 开始时间为 2020 年 1 月 1 日 0 点 0 分 0 秒 (0 秒), 持续时间为 0, 不停循环。
10. 定义设备运行 ID 8, 将触发条件 ID 4 和设备任务 ID 6 关联起来, 开始时间为 2020 年 1 月 1 日 0 点 0 分 0 秒 (0 秒), 持续时间为 0, 不停循环。