

# 稳控科技监测设备通用通讯协议接口说明

V1.00

2019年6月

## 关于本文档

- 本文档是稳控科技生产的电子设备通讯协议、数据接口使用说明，适用于所有产品。
- 本文档专门提供给程序开发工程师，若您是产品的终端客户，请直接使用我们专用的软件工具和数据管理平台。
- 表示不同进制数字时会使用明确的标识符号，以 0x 为前缀的数字为 16 进制数据，以 B 为后缀的为 2 进制数据，无任何前缀或后缀的数字为 10 进制数据。
- “\r\n” 表示非可见的回车符和换行符。
- “[]” 表示可选项（非必须内容）
- 除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、建议和举例不构成任何明示或暗示的担保，对此造成的损失不承担任何责任，本手册所述内容最终解释权归我公司所有。

## 目录

## 目录

封面 .....	1
关于本文档 .....	1
目录 .....	2
一、双向通讯协议 .....	4
1.1 标准工业 MODBUS 通讯协议 .....	4
(1) 读取多个连续的寄存器 (0x03/0x04 指令码) .....	4
(2) 修改单个寄存器 (0x06 指令码) .....	4
(3) 修改连续的多个寄存器 (0x10 指令码) .....	5
1.2 自定义 AABB 通讯协议 .....	5
(1) 读取单个寄存器 .....	5
(2) 修改单个寄存器 .....	6
(3) 通用设备地址 .....	6
1.3 自定义字符集指令 .....	6
(1) 读取参数 .....	6
(2) 修改参数 .....	7
(3) 其它常用字符串指令 .....	8
二、监测数据发送协议 .....	8
2.1 主动数据包 .....	8
(1) HEX 格式 .....	8
(2) STR 格式 1.0 .....	9
(3) STR 格式 2.0 .....	9
2.2 UART 数据发送 .....	9
2.3 手机网络-SMS 数据发送 .....	9
2.4 手机网络-TCP 数据发送 .....	10
2.5 手机网络-EMail 数据发送 .....	10
2.6 手机网络-FTP 数据发送 .....	10
2.7 区域射频数据发送 .....	10
三、数据存储格式 .....	10
3.1 FAT 文件系统及通讯协议 .....	10
(1) 进入与退出 DOS 模式 .....	10
(2) 查看文件列表-DIR1 .....	11
(3) 查看文件列表-DIR2 .....	11
(4) 显示 (输出) 文件内容-TYPE .....	11
(5) 新建文件夹-MD .....	11
(6) 新建文件-NF .....	12
(7) 删除文件-DF .....	12
(8) 从设备下载文件-DN .....	12
(9) 向设备上传文件-UP .....	13

(10) 格式化磁盘-FORMAT.....	13
3.2 文件结构说明.....	13
四、参数修改.....	13
4.1 物理接口参数修改.....	13
4.2U 盘参数修改.....	13
4.3 短信指令修改参数.....	14
4.4TCP 服务器修改参数.....	14
4.5FTP 服务器修改参数.....	14
五、开发建议.....	14
5.1 关于设备识别.....	14
5.2 关于设备通道.....	15
5.3 关于数据类型识别.....	15
附件.....	16
校验码算法.....	16
(1) CRC16-MODBUS 算法.....	16
(2) 和校验算法.....	16

## 一、双向通讯协议

设备内部预定义有若干寄存器（参数），设备在寄存器控制下完成工作。

不同的设备，区别仅是寄存器地址定义不同，关于具体寄存器定义（含义）请参阅对应设备的用户手册。

每台设备，至少提供一种或者同时提供几种数字接口：RS232、RS485、手机网络 TCP、射频无线等数字接口，通过这些数字接口，使用本章所述通讯协议，即可完成设备内部寄存器的读写访问操作。

MODBUS、AABB、字符集通讯协议均是被动协议，即：从不主动输出，仅会在收到指令后才会响应。

**注：有些设备需要上电后向其发送@SETM 进入设置模式方可支持通用协议。**

### 1.1 标准工业 MODBUS 通讯协议

所有设备均支持标准 MODBUS 协议中的 0x03 指令码和 0x06 指令码，部分设备支持 0x04 和 0x10 指令码。

#### (1) 读取多个连续的寄存器（0x03/0x04 指令码）

指令数据帧结构

地址码	功能码 0x03	开始地址	寄存器数量	CRC 校验
1 字节	1 字节	2 字节	2 字节	2 字节

返回数据帧结构

地址码	功能码 0x03	数据长度	数据	CRC 校验
1 字节	1 字节	1 字节	n 字节	2 字节

- **地址码：**设备的地址，一般默认为 1（可通过指令修改，下同）
- **功能码：**固定为 1 字节 0x03 或者 0x04，读取连续的多个寄存器
- **开始地址：**要读取的起始寄存器地址，占用 2 字节
- **寄存器数量：**要读取多少个寄存器
- **CRC 校验：**本帧数据的校验码，占用 2 个字节。（校验算法见附录，下同）
- **数据长度：**本帧数据包含的所有寄存器值占用的字节数。

例：读取设备 1 的连续多个寄存器，从寄存器 0 开始连续读取 10 个

主机发送指令：0x01 0x03 0x00 0x00 0x00 0x0A **0xC5 0xCD**

从机返回应答：0x01 0x03 0x14 0x00 0x01 0x00 0x60 0x00 0x00 0x00 0x00 0x00 0x00  
0x00 0x01 0x01 0xF4 0x00 0x00 0x00 0x64 0x00 0xC8 **0x5F 0x8F**（下划线为读取到的 10 个寄存器值）

#### (2) 修改单个寄存器（0x06 指令码）

文档版本：V1.0.0

4 / 16

通讯地址：河北省燕郊开发区创业大厦 12 层 电子邮箱：[INFO@GEO-INS.COM](mailto:INFO@GEO-INS.COM) 电话：+86 0316-3093523

## 指令数据帧结构

地址码	功能码 0x06	寄存器地址	寄存器值	CRC 校验
1 字节	1 字节	2 字节	2 字节	2 字节

## 返回数据帧结构

地址码	功能码 0x06	寄存器地址	寄存器值	CRC 校验
1 字节	1 字节	2 字节	2 字节	2 字节

例：修改设备 1 的寄存器 8 的值为 100

主机发送指令：0x01 0x06 0x00 0x08 0x00 0x64 0x09 0xE3

从机返回应答：0x01 0x06 0x00 0x08 0x00 0x64 0x09 0xE3

**(3) 修改连续的多个寄存器 (0x10 指令码)**

## 指令数据帧结构

地址码	功能码 0x10	起始地址	寄存器数量	字节数	寄存器值	CRC 校验
1 字节	1 字节	2 字节	2 字节	1 字节	n 字节	2 字节

## 返回数据帧结构

地址码	功能码 0x06	起始地址	寄存器数量	CRC 校验
1 字节	1 字节	2 字节	2 字节	2 字节

例：修改设备 1 中寄存器 0~31 的值

主机发送指令 (16 进制指令)：01 10 00 00 00 20 40 00 01 00 60 00 00 00 03 00 00  
00 01 01 F4 00 00 00 C8 C8 C8 00 01 82 35 00 05 03 E8 00 A0 05 DC 06 40 00 05 00  
64 00 00 00 0A 00 0A 00 04 01 77 03 15 00 00 00 00 03 E8 00 01 00 00 00 00 00  
**A3 70 (下划线数据为 32 个寄存器值)**

从机返回应答：0x01 0x10 0x00 0x00 0x00 0x20 0xXX 0xXX

**1.2 自定义 AABB 通讯协议**

AABB 通讯协议是一种非标准自定义协议，相较于 MODBUS 通讯协议，结构更简单，指令生成更容易，便于进行快速测试。AABB 通讯协议支持单寄存器读写两种指令。

**(1) 读取单个寄存器**

## 指令数据帧结构

指令头 0xAA 0xBB	地址码	寄存器地址	和校验
2 字节	1 字节	1 字节	1 字节

## 返回数据帧结构

指令头 0xAA 0xBB	地址码	寄存器地址	寄存器值	和校验
2 字节	1 字节	1 字节	2 字节	1 字节

- **指令头**：固定为 16 进制 AABB
- **地址码**：设备的地址 (1~255，其中地址 255 为通用地址)
- **寄存器地址**：要访问的寄存器地址 (0~127)，寄存器地址字节最高位 (bit7) 是读

写标志位，为 0 时表示读寄存器，为 1 时表示写寄存器。

- **和校验：**之前所有数据之和， $0xAA+0xBB+地址码+寄存器地址$ ，校验和超过 255 时，仅使用低字节。如下例中，校验和 $=0xAA+0xBB+0x01+0x08=0x016E$ ，则只使用 0x6E 作为最终和校验码。

例：读取设备 1 的寄存器 8 的值

主机发送指令：0xAA 0xBB 0x01 0x08 0x6E

从机返回应答：0xAA 0xBB 0x01 0x08 0x00 0x60 0xCE

## (2) 修改单个寄存器

指令数据帧结构

指令头 0xAA 0xBB	地址码	寄存器地址   0x80	寄存器值	和校验
2 字节	1 字节	1 字节	2 字节	1 字节

写寄存器指令中，寄存器地址字节的最高位应为 1，即地址值与 0x80 做“或”运算。

返回数据帧结构

指令头 0xAA 0xBB	地址码	寄存器地址	寄存器值	和校验
2 字节	1 字节	1 字节	2 字节	1 字节

例：修改设备 1 的寄存器 8 为 100

主机发送指令：0xAA 0xBB 0x01 0x88 0x00 0x64 0x52

从机返回应答：0xAA 0xBB 0x01 0x08 0x00 0x64 0xD2

## (3) 通用设备地址

AABB 通讯协议支持通用地址，无论设备的当前地址为何值，使用 0xFF 作为地址对设备发送读写指令，均可得到正确应答。

例：使用通用地址，读取任一设备的寄存器 8

主机发送指令：0xAA 0xBB 0xFF 0x08 0x6C

从机返回应答：0xAA 0xBB 0x01 0x08 0x00 0xC8 0x36

**注：**当总线上连接有多台设备时（通常为 RS485 总线），使用通用地址时总线上所有设备均会响应指令，导致通讯异常。

**注：**严禁在连接有多台设备的总线中使用通用地址修改设备地址。

## 1.3 自定义字符集指令

使用更加易于手工操作的字符串作为控制指令，字符串相较于上述两种协议，更加直观。字符串指令支持一次发送多条，设备在收到多条指令后会依次执行。

### (1) 读取参数

## 指令数据帧结构

帧头	固定分隔符	参数地址	帧尾
\$GETP[addr]	'='	parAddr	\r\n

## 返回数据帧结构

帧头	参数地址	固定分隔符	寄存器值	帧尾
\$REG	[parAddr]	'='	字符串	\r\n

例：

读取参数 2（忽略设备地址）

主机发送指令：\$GETP=2\r\n

从机返回应答：\$REG[01]=01152\r\n

读取地址为 0x01 的设备中的参数 2

主机发送指令：\$GETP001=2\r\n

从机返回应答：\$REG[01]=01152\r\n

主机发送指令：\$GETP=2

\$GETP=5

\$GETP=72\r\n

从机返回应答：\$BAUDRATE=1152

OK

\$INTE\_SCAN=2

OK

\$TCP\_SEV=www.zoyobj.com

OK\r\n

## （2）修改参数

## 指令数据帧结构

帧头	固定分隔符	寄存器地址	寄存器值	帧尾
\$GETP[addr]	'='	regAddr	字符串	\r\n

## 返回数据帧结构

结果码	备注说明
OK\r\n	修改成功
Err\r\n	修改失败
Read only\r\n	此寄存器是只读的，不可修改
Unknown addr\r\n	寄存器地址错误

例：

修改参数 2 的值为 96

主机发送指令：\$SETP=2, 96#

从机返回应答：OK\r\n

修改设备 1 的参数 2 的值为 96

文档版本：V1.0.0

7 / 16

通讯地址：河北省燕郊开发区创业大厦 12 层 电子邮箱：[INFO@GEO-INS.COM](mailto:INFO@GEO-INS.COM) 电话：+86 0316-3093523

主机发送指令：\$SETP001=2,96#

从机返回应答：OK\r\n

### (3) 其它常用字符串指令

字符串指令	备注说明
@SETM[addr]\r\n	进入参数设置模式
@STTN[addr]=x\r\n	设置透明传输端口号
\$SAVE[addr]	永久保存参数
\$INFO[addr]	查看设备基本信息
\$REST[addr]	设备重启
\$STDN[addr]	设备关机
\$RSTP[addr]	恢复出厂参数
\$STFC[addr]	当前参数写入到出厂参数
\$STDF[addr]	设置为默认参数
\$DEBUG	

## 二、监测数据发送协议

监测数据发送是指设备工作于定时采发模式时，每次采集完毕后经由数字接口主动输出的数据帧。

### 2.1 主动数据包

主动输出的监测数据帧有三种版本，HEX 格式、STR 格式 1.0 和 STR 格式 2.0，这三种数据可通过设备参数任选一种。

#### (1) HEX 格式

16 进制数据，数据帧长度 91 字节。

数据地址	数据长度	备注说明
0	1	数据类型码 0x01: 手动测试数据 0x02: 自动发送数据
1~2	2	设备 ID 号
3~6	4	数据记录号
7~12	6	时间戳 BCD 码年月日时分秒
13	1	预留，无定义
14	1	手机网信号强度 0~31
15~16	2	设备温度，单位 0.1℃
17~18	2	供电电压，单位 0.01V
19~20	2	充电电压，单位 0.01V



21~24	4	脉冲传感器计数值
25~88	64	32 通道监测数据
89~90	2	MODBUS 协议 CRC 校验

### (2) STR 格式 1.0

16 进制字符串，数据帧长度 156 字节。每 2 个字符转换为 1 字节数据，例如：字符串“1A”表示 0x1A（即 10 进制的 26）

数据地址	数据长度	备注说明
0~1	2 字符	设备 ID 号
2~5	4 字符	数据记录号
6~7	2 字符	预留，无定义
8~9	2 字符	手机网信号强度 0~31
10~11	2 字符	设备温度，单位℃
12~15	4 字符	脉冲传感器计数值
16~19	4 字符	供电电压，单位 0.01V
20~23	4 字符	充电电压，单位 0.01V
24~151	128 字符	32 通道监测数据
152~155	4 字符	MODBUS 协议 CRC 校验

### (3) STR 格式 2.0

用数据名称标识和 10 进制数字字符串表示的数据序列，更宜阅读。

数据字符串	备注说明
BATV=nnnnn	供电电压，单位 0.01V nnnnn 为固定的 5 个数字字符，下同
CHGV=nnnnn	充电电压，单位 0.01V
SIGV=nnnnn	手机网信号强度 0~31
TEMP=nnnnn	设备温度，单位 0.1℃
CHxx=nnnnn	32 通道数据，xx 为 2 个数字字符表示的通道号

## 2. UART 数据发送

经由设备的 RS232、RS485 等物理接口直接发送上述数据帧，无任何其它附加内容。

当选择数据输出协议为“STR2.0”时，不同数据项之间用\r\n分隔。

## 2.3 手机网络-SMS 数据发送

直接向目标手机号码发送上述数据帧。

当选择数据输出协议为“STR2.0”时，不同数据项之间用\r\n分隔。

## 2.4 手机网络-TCP 数据发送

以 TCP 客户身份连接预设的 TCP 服务器，并在连接成功后发送以下数据帧：

设备的唯一识别码 UDID+ “>” +上述数据帧。

例：15B87911B123456>数据

## 2.5 手机网络-EMail 数据发送

利用手机网络，向预设的邮箱发送数据，邮件内容如下：

```
Device ID 15B87911B123456
BATV CHGV SIGV TEMP
10282 00000 00016 00031
GMT CH01 CH02 CH03 CH04 CH05 CH06 CH07 CH08.....
2016-04-30 15:08:01 00000 00000 00000 00000 00000 00000 00000.....
```

当设备具备邮件发送功能时，会同时具备一次发送多条数据的功能，多条数据发送时，邮件正文为实时数据，其它历史数据会以附件形式发送到邮箱，附件名称为“DATA.txt”。

## 2.6 手机网络-FTP 数据发送

利用手机网络，向预设的 FTP 服务器发送数据文件，文件内容与上述邮件内容相同。

## 2.7 区域射频数据发送

利用设备自带的无线射频功能，向外广播一帧数据，数据帧格式可设置为任意一种。

# 三、数据存储格式

## 3.1 FAT 文件系统及通讯协议

对于具备内部 FAT 系统的设备，每次数据发送同时，会在设备内部同步保存。通过 RS232 或者 RS485 接口可完成对内部文件的访问，比如：下载数据文件、删除旧的无用数据文件等。

设备内部 FAT 文件系统的访问必须在特定的工作模式下进行(下文称为“DOS 模式”)。

### (1) 进入与退出 DOS 模式

通过向设备发送“@ENDOS”进入 DOS 模式。

对于某些设备，可能需要先向设备发送“@SETM”指令使设备可以响应外部指令，再发送“@ENDOS”指令。

处于 DOS 模式的设备，向其发送“@EXDOS”即可退出 DOS 模式。

**(2) 查看文件列表-DIR1**

语法：DIR1 路径\r\n

设备返回指定路径下所有文件夹及文件

例：

发送指令 DIR1 0:\r\n	发送指令 DIR1 0:/dat\r\n
设备返回	0:/dat
0:	0:/dat/DAT_1605.txt
0:/sys	0:/dat/DAT_1606.txt
0:/dat	0:/dat/12345.txt
0:/dat/DAT_1605.txt	0:/dat/
0:/dat/DAT_1606.txt	OK
0:/dat/12345.txt	
0:/log	
0:/sms	
0:/tts	
0:	
OK	

**(3) 查看文件列表-DIR2**

与 DIR1 近似，文件名称后增加了文件大小数据，举例如下：

发送指令 DIR2 0:\r\n	发送指令 DIR2 0:/dat\r\n
设备返回	0:/dat
0:	0:/dat/DAT_1605.txt 796
0:/sys	0:/dat/DAT_1606.txt 2183
0:/dat	0:/dat/12345.txt 2183
0:/dat/DAT_1605.txt 796	0:/dat/
0:/dat/DAT_1606.txt 2183	OK
0:/dat/12345.txt 2183	
0:/log	
0:/sms	
0:/tts	
0:	
OK	

**(4) 显示（输出）文件内容-TYPE**

语法：TYPE 文件名\r\n

例：向设备发送“TYPE 0:/dat/DAT\_1605.txt\r\n”

设备立即返回文件的内容。

**(5) 新建文件夹-MD**

语法：MD 文件夹\r\n

例：向设备发送“MD 0:/ABC”  
设备返回“MDO:/ABC\r\nOK\r\n”

#### (6) 新建文件-NF

语法：NF 文件名\r\n

例：向设备发送“NF 0:/ABC/1234.txt\r\n”

设备返回“NF0:/ABC/1234.txt\r\nOK\r\n”

向设备发送“DIR2 0:/ABC\r\n”

设备返回

0:/ABC

0:/ABC/1234.txt 0

0:/ABC/

OK

#### (7) 删除文件-DF

语法：DF 文件名\r\n

例：向设备发送“DF 0:/ABC/1234.txt\r\n”

设备返回“DF0:/ABC/1234.txt\r\nOK\r\n”

向设备发送“DIR2 0:/ABC\r\n”

设备返回

0:/ABC

0:/ABC

OK

#### (8) 从设备下载文件-DN

语法：

- A. 向设备发送>>：DN 文件名\r\n
- B. 从设备返回<<：OK\r\n
- C. 向设备发送>>：请求码，格式见下文说明。
- D. 从设备返回<<：一帧数据，格式见下文说明。
- E. 判断接收到的数据帧是否是最后一条，若不是则继续步骤 C。
- F. 结束。

请求码，请求对方回传一帧文件数据

帧头	请求码	帧尾
0x5A 0x5A	2 字节	0x5C

- **请求码 0x00 0x00**：首次发送时表示请求开始传输第 0 帧数据，之后发送表示请求发送下一帧数据。
- **请求码 0xF1 0x11**：请求对方重复传输一次数据帧。一般用于接收数据并发现校验错误时要求对方再次发送一次。

每帧数据固定为 108 字节，格式为：

帧头	数据帧号	有效数据长度	数据段	校验	帧尾
0x5A 0x5A	2 字节	1 字节 (0~100)	100 字节	CRC 2 字节	0x5B
注：当收到有效数据长度为 0 的数据帧时，表示文件传输结束					
注：CRC 校验采用 MODBUS CRC16 算法，见附录。					

### (9) 向设备上传文件-UP

语法：

- 向设备发送>>：UP 文件名\r\n
- 从设备返回<<：就绪码（请求码）
- 向设备发送>>：一帧数据，格式见上文说明。若是最后一帧数据，则跳转至步骤 E，否则跳转至步骤 B。
- 结束。

### (10) 格式化磁盘-FORMAT

语法：FORMAT\r\n

设备返回：“formatting...\r\n\r\nOK\r\n”

## 3.2 文件结构说明

```

BASEINFO  EQUITYPE LF01
BASEINFO  HWVER  1.00
BASEINFO  HWVER  1.00
BASEINFO  DATATYPE  30
BASEINFO  DATETIME  2000/01/09 01:00:45
BASEINFO  CREATEBY  LF01
BASEINFO  SN  1234567890123456
TITEL  RECNUM  DATETIME  CH01  CH02  CH03  CH04  CH05  CH06  CH07  CH08  CH09  CH10  CH11  CH12  CH13  CH14  CH15  CH1
RECORD  0  2000/01/09 01:00:45  0E7F  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000
RECORD  0  2000/01/09 01:10:47  0E89  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000
RECORD  0  2000/01/09 08:02:00  0EAF  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000
RECORD  0  2000/01/09 08:04:00  0E92  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000
RECORD  0  2000/01/09 08:05:00  0EA6  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000
RECORD  0  2000/01/09 08:06:00  0EB9  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000
RECORD  0  2000/01/09 08:08:00  0EFB  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000

```

## 四、参数修改

### 4.1 物理接口参数修改

使用前述三种协议之一，通过连接 RS232 或者 RS485 完成参数修改。

对于字符串参数，建议使用\$指令集中的\$SETP 指令。

### 4.2U 盘参数修改

创建指令配置文件 PARSCMD\$.txt，在文件内添加\$指令，每个指令占用一行。将此文件复制到 U 盘。

将 U 盘插入设备，开机，设备分自动检测 U 盘内的 PARSCMD\$.txt 文件并执行\$指令。

PARSCMD\$.txt 文件可以复制到 U 盘根目录，也可以复制到 U 盘内的设备 UDID 文件夹，分别用于所有设备和 UDID 一致的设备。（文件夹名称必须为设备 UDID 的后 6 位字符）

### 4.3 短信指令修改参数

使用任意短信发送工具（如：短信猫或者手机），编辑短信指令发送给设备内的 SIM 卡（手机号）。

短信指令语法：

\$SETP=参数地址, 参数值#

\$SETP=参数地址, 参数值#

.....

一次短信可包含多条指令，不同指令之间用回车分隔。

设备收到并正确执行指令后，会返回收到的短信内容以表示执行成功。

注：仅被设置为“数据中心电话号码”的手机号码发出的为合法指令，其它手机号码发送的任何短消息会被设备忽略。

### 4.4 TCP 服务器修改参数

设备与 TCP 服务器建立链接后，服务器可向设备发送任何符合设备通讯协议的指令，交互过程与 RS232 完全相同。

### 4.5 FTP 服务器修改参数

设备在每次采发结束时主动检测预设的 FTP 服务器上是否存在 UDID/PARSCMD\$.txt 文件，若存在则下载执行，执行完毕后删除 FTP 服务器上的指令文件。

PARSCMD\$.txt 文件内容要求与前述 U 盘修改参数相同。

## 五、开发建议

### 5.1 关于设备识别

本节所述设备识别主要应用于监测软件平台开发。

(1) 基于有线连接的监测系统，一般情况下所有设备均接入 RS485 总线，主机通过通讯协议可实时获取设备运行状态和实时监测数据。采用有线连接的监测系统，建议使用设备地址来区分不同的设备。

(2) 无线短信收发数据的监测系统，建议软件平台在接收短信数据时检索出短信信息中的发送方手机号码来区分不同的监测设备。

(3) 无线 TCP 服务器监测系统，监测设备在向服务器发送的数据包中包含有设备的 UDID 信息，建议此时软件平台使用 UDID 来区分不同设备。

(4) EMAIL 和 FTP 监测系统，邮件或者 FTP 数据文件中均包含有设备的 UDID 信息，建议使用 UDID 来识别不同设备。

对于监测平台来说，理想的情况是可以处理上述任意一种数据来源，建议在服务器数据库中建立一个专门的设备信息表，表结构如下：

字段	类型	备注说明
设备 ID	字符串, 20 字节	由平台生成的设备唯一识别码, 非必须
设备名称	字符串, 100 字节	可以作为软件平台中对设备的唯一识别信息
设备地址	16 位整型, 2 字节	
手机号码	字符串, 30 字节	有些物联网卡号码会达到 20 位或以上
UDID	字符串, 30 字节	设备的 UDID 一般小于 16 个字符
所属于监测项目		此设备服务于哪个监测项目

不管数据来源是哪一种, 接收到数据检索出识别信息 (设备地址、手机号码或者 UDID) 后, 在设备信息表中检索, 最终均识别为“设备别名”。

## 5.2 关于设备通道

无论使用何种型号的设备, 设备主动发送的通讯协议均相同 (前述“主动数据包”协议之一), 设备每个通道连接何种传感器以及传感器原始数据如何转换为物理数据需要在安装时由安装人员提供, 这些信息用于软件平台的数据管理与解算, 故此建议软件平台内包含有针对每个通道 (每个传感器) 的专用基本信息表, 表结构建议如下:

字段	类型	备注说明
通道 ID	字符串, 20 字节	由平台生成的每个通道的唯一识别码, 非必须
通道名称	字符串, 100 字节	形如“XXXX 位移”、“XXXX 传感器”等
所属设备	字符串	此通道属于哪台监测设备
通道号	16 位整型, 2 字节	1~32
计算方法		能够将设备传回的此通道的原始数据解析计算为物理数据的描述信息或者必须计算参数, 可能需要多个字段
物理单位	字符串, 20 字节	mm、KPa、Hz 等

## 5.3 关于数据类型识别

接收到设备发来的一帧数据后, 首先应识别数据帧协议类型 (HEX、STR1.0 或者 STR2.0), 再进行数据解析。建议的识别方法如下:

接收到短信、TCP、RS232 (或者 RS485) 数据后

判断是否包含有“BATV=”, 若包含则识别为 STR2.0 数据帧, 若不包含, 则继续判断

判断数据长度为 156 时, 识别为 STR1.0, 数据长度为 91 时识别为 HEX 数据帧。

短信数据包中包含有设备手机号码、TCP 数据流中包含有设备 UDID、RS232 数据流中包含有设备地址, 任意一种信息均可从设备信息表中检索出对应的唯一一台设备。

## 附件

### 校验码算法

#### (1) CRC16-MODBUS 算法

```
unsigned int crc16(unsigned char *dat, unsigned int len)
{
    unsigned int crc=0xffff;
    unsigned char i;

    while(len!=0)
    {
        crc^=*dat;
        for(i=0;i<8;i++)
        {
            if((crc&0x0001)==0)
                crc=crc>>1;
            else
            {
                crc=crc>>1;
                crc^=0xa001;
            }
        }
        len-=1;
        dat++;
    }
    return crc;
}
```

#### (2) 和校验算法

```
unsigned char AddCheck(unsigned char *dat, unsigned char count)
{
    unsigned char i, Add=0;
    for (i=0; i<count; i++)
        Add+=dat[i];
    return Add;
}
```