RC-RS485采集仪编程手册

（Modbus 版）

V1.0.0

目录

[一、DIS液压式竖向位移采集器概述 3](#_Toc444016263)

[二、HCF700参数 4](#_Toc444016264)

[1、HCF700的参数： 4](#_Toc444016265)

[2、HCF700寄存器列表 4](#_Toc444016266)

[3、HCF700参数的存储格式 5](#_Toc444016267)

[4、HCF700参数出厂值 5](#_Toc444016268)

[三、HCF700指令详解 6](#_Toc444016269)

[1、功能码列表 6](#_Toc444016270)

[2、异常码返回 6](#_Toc444016271)

[3、功能详解 7](#_Toc444016272)

[3.1发送采样指令 7](#_Toc444016273)

[3.2设置传感器类型 8](#_Toc444016274)

[3.3设置采样通道 9](#_Toc444016275)

[3.4重设设备地址 10](#_Toc444016276)

[3.6查看电量 12](#_Toc444016277)

[3.5查看当前状态 13](#_Toc444016278)

[3.7查看系统工作时间 14](#_Toc444016279)

[3.8查看传感器数据 15](#_Toc444016280)

[3.9恢复出厂设置 16](#_Toc444016281)

[四、附录 18](#_Toc444016282)

# 一、RC-RS485采集仪概述

1. 双路485采集器（以下简称RC-RS485）是一款可以接入标准485网络的采集仪，分为上行RS485接口和下行RS485接口，其中上行接口采用标准的Modbus协议格式，下行接口支持定制开发，兼容多种应用RS485接口的传感器。本协议仅描述RS485上行接口的Modbus协议内容。
2. RC-RS485的输出方式为：RS485 方式，遵循 Modbus 协议的 RTU 模式。
3. RC-RS485的通信方式为半双工方式。只有在接收到上位机有 效的指令消息后才向上位机发送回馈消息。空闲状态一直处于接收上位 机指令消息状态。
4. RC-RS485传输数据采用 8 位数据位传输，无校验方式，波特 率可选 4800，9600，19200，115200(波特率在传感器生产时设置，出厂后不可更改)。

数据帧格式为：1 个起始位，8 数据位，无校验位，1 个停止位。

设备不应答非本机地址、格式错误和校验不正确的指令。但为调试方便，当校验码为“0xFF”时，为不校验方式，即MCU响应的指令包括校验码正确和校验码为FF的指令。

# 二、RC-RS485参数

## 1、RC-RS485的参数

产品类型，程序版本号，设备地址，当前工作状态，系统工作时间，

## 2、HCF700寄存器列表

|  |  |  |  |
| --- | --- | --- | --- |
| **地址** | **类型** | **参数** | **数据类型** |
| 0x0000 | 保持寄存器 | 设备地址1 | Uint16 |
| 0x0001 | 保持寄存器 | 已存储的数据量，最大1999条 | Uint16 |
| 0x0002 | 保持寄存器 | 定时采样间隔时长（min）,默认60，0表示持续采样，0xffff表示取消定时采样 | Float |
| 0x0003 | 保持寄存器 | 最早的一条数据编号 |
| 0x0004 | 保持寄存器 |  | Float |
| 0x0005 | 保持寄存器 |  |
| 0x0006 | 保持寄存器 |  | Uint16 |
| 0x0007 | 保持寄存器 |  | Uint16 |
| 0x0008 | 保持寄存器 |  | Float |
| 0x0009 | 保持寄存器 |  |
| 0x000A | 保持寄存器 |  | Float |
| 0x000B | 保持寄存器 |  |
| 0x000C | 保持寄存器 |  | Float |
| 0x000D | 保持寄存器 |  |
| 0x000E | 保持寄存器 |  | Float |
| 0x000F | 保持寄存器 |  |
|  | | |  |
| 0x00003 | 输入寄存器 | 设备类型号 | Uint16 |
| 0x0001 | 输入寄存器 | 本机软件版本 | Uint16 |
| 0x0002 | 输入寄存器 | 系统时间YY | Uint32 |
| 0x0003 | 输入寄存器 | 系统时间MM |
| 0x0004 | 输入寄存器 | 系统时间DD | Uint16 |
| 0x0005 | 输入寄存器 | 系统时间HH | Uint16 |
| 0x0006 | 输入寄存器 | 系统时间mm | Float |
| 0x0007 | 输入寄存器 | 系统时间SS |
| 0x0008 | 输入寄存器 | 备用 | Float |
| 0x0009 | 输入寄存器 | 当前网络上节点数量 |
| 0x000A | 输入寄存器 | 采集仪当前电池电压 | Float |
| 0x000B | 输入寄存器 |  |
| 0x000C | 输入寄存器 |  | Float |
| 0x000D | 输入寄存器 |  |
| 0x000E | 输入寄存器 |  | Uint16 |
| 0x000F | 输入寄存器 |  | Uint16 |

1. 寄存器均为 16 位，参数大于16位的数据占用多个寄存器地址。多个寄存器的数据必须连续读取所有相关寄存器。
2. 不同类型寄存器可以使用相同地址，因为操作不同类型寄存器的功能号也不同。保持寄存器支持读和写，输入寄存器只读。

## 3、RC-RS485参数的存储格式与说明

1. **设备类型**：RS485产品类型：0020.
2. **设备地址**：为 16 位无符号整数，高 8 位为 0，低 8 位有效地址范围为 1-247，0 为广播地址。
3. **当前工作状态**：为 16 位无符号整数，高 8 位为 0，低8位有效，数据范围0-2，0=就绪（未采样或采样完成），1=忙，2=采样中，3=设备不可用，该寄存器写入无效。
4. **系统工作时间**:开机后开始统计工作时间，单位s。
5. **ADC原始度数**：16位无符号数。
6. **液体密度，重力加速度值，高度**值（单位mm）：使用国际标准单位制存储。其中使用4 个字节 32 位单精度浮点数，表示格式采用 IEEE754 标准，大端模式存储数据，在给上位机发送数据时高字数据先发送。例如，液位高度为 123.4mm，那么根据 IEEE754 标准转换出来的4字节数为：0x42 F6 CC CD

## 4、HCF700参数出厂值

|  |  |  |
| --- | --- | --- |
| **寄存器名称** | **默认值** | **备注** |
| 设备地址 | 0x01 |  |
| 液体密度 | 1.0 |  |
| 初始高度 | 0.0 |  |
| 重力加速度值 | 9.8015 | 北京数值（海口为 9.78） |

# 三、RC-RS485指令详解

HCF700提供的指令有:发送采样指令，设置传感器类型，设置采样通道，重设设备地址，查看当前状态，查看电池电压，查看系统工作时间，查看传感器数据及恢复出厂设置。

## 1、功能码列表

|  |  |  |  |
| --- | --- | --- | --- |
| **功能码** | **名称** | **作用** | **备注** |
| 0x03 | 读取保持寄存器 | 在一个或多个保持寄存器中取得当前的二进制值 | 保持寄存器 |
| 0x04 | 读取输入寄存器 | 在一个或多个输入寄存器中取得当前的二进制值 | 输入寄存器 |
| 0x06 | 预置单个寄存器 | 把具体二进值装入一个保持寄存器 | 保持寄存器 |
| 0x10 | 预置多个寄存器 | 把具体的二进制值装入一串连续的保持寄存器 | 保持寄存器 |
| 0x41 | 获取第n条数据 | 0表示最新数据，1-1999条 |  |
| 0x42 | 采样命令 | 命令采集仪采样（可广播、可指定地址采样） |  |
| 0x43 | 更改时间 | 年月日时分秒 |  |
| 0x44 |  |  |  |
| 0x68 | 恢复出厂设置 | 恢复所有保持寄存器出厂配置参数 | 自定义功能码 |

## 2、异常码返回

指令异常回馈消息 当指令操作出现异常时下位机会发送一个回馈消息。 静力水准传感器异常回馈消息格式：

|  |  |
| --- | --- |
| **名称** | **字节数** |
| 设备地址 | 1 |
| 异常功能码 | 1 |
| 异常码 | 1 |
| 16 位 CRC 校验 | 2 |

异常功能码为指令功能号加上 0x80

异常码如下表

|  |  |
| --- | --- |
| 0x01 | 非支持的功能（功能号有效） |
| 0x02 | 寄存器地址不正确 |
| 0x03 | 寄存器值超出范围 |
| 0x04 | 有效操作发生异常 |
| 0x05 | 确认 |
| 0x06 | 功能号无效 |
| 0x07 | 设备忙 |

举例：查看保持寄存器数据，读取数量错误，大于最大寄存器数量，发送01 03 00 00 00 0B 04 0D

HCF700异常返回消息：01 83 02 C0 F1

## 3、功能详解

### 3.1满量程标定

在接收到上位机发送的采样指令后，读取ADC值，存入保持寄存器。加入操作码，

### 3.2压力下传感器标定

在接收到上位机发送的采样指令后，HCF700根据采样指令内容执行传感器采样，然后将数据存储于输入寄存器中。

### 3.3值自动设定指令

在接收到上位机发送的采样指令后，HCF700根据采样指令内容执行传感器采样，然后将数据存储于输入寄存器中。

### 3.4送采样指令

在接收到上位机发送的采样指令后，HCF700根据采样指令内容执行传感器采样，然后将数据存储于输入寄存器。

### 3.4重设设备地址

设备地址出厂默认值为0x01，多个设备集联使用时不允许地址相同的情况，所以需要修改成不同地址，使用功能码0x06写单个保持寄存器，保持寄存器地址为0x0000。

上位机请求消息：

|  |  |
| --- | --- |
| 名称 | 字节数 |
| 设备地址 | 1 |
| 功能码 | 1 |
| 保持寄存器地址 | 2 |
| 采样指令数据 | 2 |
| 16 位 CRC 校验 | 2 |

传感器正常回馈消息：

|  |  |
| --- | --- |
| 名称 | 字节数 |
| 设备地址 | 1 |
| 功能码 | 1 |
| 保持寄存器地址 | 2 |
| 采样指令数据 | 2 |
| 16 位 CRC 校验 | 2 |

举例：上位机向地址为0x01号从机发送“重设设备地址指令”，设置地址为0x02，即主机请求数据帧为：01 06 00 00 00 02 08 0B

|  |  |
| --- | --- |
| 名称 | 字节数 |
| 设备地址 | 0x01 |
| 功能码 | 0x06 |
| 保持寄存器地址高位 | 0x00 |
| 保持寄存器地址低位 | 0x00 |
| 保持寄存器数据高位 | 0x00 |
| 保持寄存器数据低位 | 0x02 |
| 16位CRC校验低位1 | 0x08 |
| 16位CRC校验高位 | 0x0B |

注1：16 位 CRC 校验码是低位在前，高位在后。

HCF700正常反馈消息：02 06 00 00 00 02 08 38

|  |  |
| --- | --- |
| 名称 | 字节数 |
| 设备地址 | 0x02 |
| 功能码 | 0x06 |
| 保持寄存器地址高位 | 0x00 |
| 保持寄存器地址低位 | 0x00 |
| 保持寄存器数据高位 | 0x00 |
| 保持寄存器数据低位 | 0x02 |
| 16位CRC校验低位 | 0x08 |
| 16位CRC校验高位 | 0x38 |

### 3.5查看当前状态

HCF700工作状态实时更新，可随时查看当前工作状态，使用功能码0x04读取多个输入寄存器，寄存器数量为1，输入寄存器地址为0x0001。

上位机请求消息：

|  |  |
| --- | --- |
| 名称 | 字节数 |
| 设备地址 | 1 |
| 功能码 | 1 |
| 输入寄存器地址 | 2 |
| 寄存器数量 | 2 |
| 16 位 CRC 校验 | 2 |

传感器正常回馈消息：

|  |  |
| --- | --- |
| 名称 | 字节数 |
| 设备地址 | 1 |
| 功能码 | 1 |
| 数据字节数 | 1 |
| 工作状态数据 | 2 |
| 16 位 CRC 校验 | 2 |

举例：上位机向地址为0x01号从机发送“查看当前状态指令”，即主机请求数据帧为：01 04 00 01 00 01 60 0A

|  |  |
| --- | --- |
| 名称 | 字节数 |
| 设备地址 | 0x01 |
| 功能码 | 0x04 |
| 输入寄存器地址高位 | 0x00 |
| 输入寄存器地址低位 | 0x01 |
| 寄存器数量高位 | 0x00 |
| 寄存器数量低位 | 0x01 |
| 16位CRC校验低位 | 0x60 |
| 16位CRC校验高位 | 0x0A |

HCF700正常反馈消息：01 04 02 00 00 B9 30

数据为：0x00，表示设备处于就绪状态。

|  |  |
| --- | --- |
| 名称 | 字节数 |
| 设备地址 | 0x01 |
| 功能码 | 0x04 |
| 数据字节数 | 0x02 |
| 数据高位 | 0x00 |
| 数据低位 | 0x00 |
| 16位CRC校验低位 | 0xB9 |
| 16位CRC校验高位 | 0x30 |

### 3.7查看系统工作时间

HCF700开机后会统计连续工作时间，单位为秒，默认15分钟无操作执行稳定性重启，所以最大工作时间为15分钟，即900秒，使用功能码0x04读取多个输入寄存器，寄存器数量为2，输入寄存器地址为0x0002。

上位机请求消息：

|  |  |
| --- | --- |
| 名称 | 字节数 |
| 设备地址 | 1 |
| 功能码 | 1 |
| 输入寄存器地址 | 2 |
| 寄存器数量 | 2 |
| 16 位 CRC 校验 | 2 |

传感器正常回馈消息：

|  |  |
| --- | --- |
| 名称 | 字节数 |
| 设备地址 | 1 |
| 功能码 | 1 |
| 数据字节数 | 1 |
| 工作时间数据 | 4 |
| 16 位 CRC 校验 | 2 |

举例：上位机向地址为0x01号从机发送“查看系统工作时间指令”，即主机请求数据帧为：01 04 00 02 00 02 D0 0B

|  |  |
| --- | --- |
| 名称 | 字节数 |
| 设备地址 | 0x01 |
| 功能码 | 0x04 |
| 输入寄存器地址高位 | 0x00 |
| 输入寄存器地址低位 | 0x02 |
| 寄存器数量高位 | 0x00 |
| 寄存器数量低位 | 0x02 |
| 16位CRC校验低位 | 0xD0 |
| 16位CRC校验高位 | 0x0B |

HCF700正常反馈消息：01 04 04 00 00 02 B2 7A 91

数据为：0x02B2，表示设备已连续工作690秒。

|  |  |
| --- | --- |
| 名称 | 字节数 |
| 设备地址 | 0x01 |
| 功能码 | 0x04 |
| 数据字节数 | 0x04 |
| 工作时间数据31-24位 | 0x00 |
| 工作时间数据23-16位 | 0x00 |
| 工作时间数据15-8位 | 0x02 |
| 工作时间数据7-0位 | 0xB2 |
| 16位CRC校验低位 | 0x7A |
| 16位CRC校验高位 | 0x91 |

### 3.8查看传感器数据

当执行完成采样指令后， 读取相应寄存器即可。

### 3.9恢复出厂设置

恢复出厂设置将设备所有可配置数据均恢复到出厂设置，见[出厂参数](#_4、振弦采集仪参数出厂值)，当不知道设备地址时使用广播指令，即地址为0，此操作需慎重使用，特别是集连多个设备时，会将所有设备都恢复出厂配置，未避免误操作，设置有4个字节操作码，分别是0x33 0x55 0x77 0x99，使用功能码0x68（自定义操作码）。

上位机请求消息：

|  |  |
| --- | --- |
| 名称 | 字节数 |
| 设备地址 | 1 |
| 功能码 | 1 |
| 操作码 | 4 |
| 16 位 CRC 校验 | 2 |

传感器正常回馈消息：

|  |  |
| --- | --- |
| 名称 | 字节数 |
| 设备地址 | 1 |
| 功能码 | 1 |
| 保持寄存器地址 | 2 |
| 寄存器数量 | 2 |
| 数据字节数 | 1 |
| 所有保持寄存器数据 | 20 |
| 16 位 CRC 校验 | 2 |

举例：上位机向地址为0x01号从机发送“恢复出厂设置指令”，即主机请求数据帧为：01 68 33 55 77 99 98 CD

|  |  |
| --- | --- |
| 名称 | 字节数 |
| 设备地址 | 0x01 |
| 功能码 | 0x68 |
| 操作码1 | 0x33 |
| 操作码2 | 0x55 |
| 操作码3 | 0x77 |
| 操作码4 | 0x99 |
| 16位CRC校验低位 | 0x98 |
| 16位CRC校验高位 | 0xCD |

HCF700正常反馈消息：01 68 00 00 00 0A 14 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FF 6C B3

|  |  |
| --- | --- |
| 名称 | 字节数 |
| 设备地址 | 0x01 |
| 功能码 | 0x68 |
| 保持寄存器地址高位 | 0x00 |
| 保持寄存器地址低位 | 0x00 |
| 保持寄存器数量高位 | 0x00 |
| 保持寄存器数量低位 | 0x0A |
| 数据字节数 | 0x14 |
| 设备地址高位 | 0x00 |
| 设备地址低位 | 0x01 |
| 采样方式高位 | 0x00 |
| 采样方式低位 | 0x00 |
| 传感器类型63-48位高位 | 0x00 |
| 传感器类型63-48位低位 | 0x00 |
| 传感器类型47-32位高位 | 0x00 |
| 传感器类型47-32位低位 | 0x00 |
| 传感器类型31-16位高位 | 0x00 |
| 传感器类型31-16位低位 | 0x00 |
| 传感器类型15-0位高位 | 0x00 |
| 传感器类型15-0位低位 | 0x00 |
| 通道选择数据63-48位高位 | 0x00 |
| 通道选择数据63-48位低位 | 0x00 |
| 通道选择数据47-32位高位 | 0x00 |
| 通道选择数据47-32位低位 | 0x00 |
| 通道选择数据31-16位高位 | 0x00 |
| 通道选择数据31-16位低位 | 0x00 |
| 通道选择数据15-0位高位 | 0xFF |
| 通道选择数据15-0位低位 | 0xFF |
| 16 位 CRC 校验低位 | 0x6C |
| 16 位 CRC 校验高位 | 0xB3 |

# 四、附录

**Modbus CRC校验函数（C语言）**

unsigned short int CRC16\_Check(unsigned char \*Pushdata, unsigned char length)

{

unsigned short int Reg\_CRC=0xffff;

unsigned char i,j;

for( i = 0; i<length; i++)

{

Reg\_CRC^= \*Pushdata++;

for (j = 0; j<8; j++)

{

if (Reg\_CRC & 0x0001)

Reg\_CRC=Reg\_CRC>>1^0xA001;

else

Reg\_CRC >>=1;

}

}

return Reg\_CRC;

}