

电化学设备通信协议文档

目录

- [1. 协议概述](#)
- [2. 通信规范](#)
- [3. 协议状态机](#)
- [4. 命令格式](#)
- [5. 响应格式](#)
- [6. 完整通信流程](#)
- [7. 错误处理](#)
- [8. 数据格式](#)
- [9. 技术指标](#)
- [10. 附录](#)

协议概述

本通信协议用于电化学设备（如循环伏安法 CV 仪表）的参数配置和数据采集。协议基于串口通信，支持以下功能：

- 参数配置**：设置电位范围、扫描速率、循环次数等测试参数
- 测试控制**：启动、停止测试
- 实时数据传输**：接收电位和电流数据
- 模拟模式**：支持无硬件设备的测试

主要特性

- ✓ 串口通信 (RS232/USB)
- ✓ 实时数据流传输
- ✓ 队列式响应处理
- ✓ 多线程架构
- ✓ 模拟模式支持
- ✓ 完整的状态管理

通信规范

物理层

参数	值	说明
接口	RS232/USB	串口通信

参数	值	说明
波特率	115200	默认波特率（可配置）
数据位	8	标准配置
校验位	N (无)	无奇偶校验
停止位	1	1个停止位
超时	1s	串口读取超时

数据层

- 编码格式：UTF-8
- 行结束符：\r\n (CRLF) 或 \n (LF)
- 分隔符：逗号 , (用于数据字段分隔)
- 字段类型：
 - 浮点数：电压值、电流值（4位小数精度）
 - 整数：参数值
 - 特殊符号：控制响应

协议状态机

状态定义

状态	值	说明
IDLE	0	初始状态，未连接
PARAMETER_SET	1	参数已设置
WAITING_ACK	2	等待参数确认响应
STARTING_TEST	3	测试即将开始
RECEIVING_DATA	4	正在接收数据
TEST_COMPLETE	5	测试完成
ERROR	6	错误状态

命令格式

1. 参数设置命令 (P - Parameter)

格式： P ,,,,,,

完整参数列表：

P <start_v>,<end_v>,<scan_dir>,<scan_rate>,<start_v2>,,<vertex_v>,,,,,<scan_interval>,,<current_range>,<current_range>,,,,

参数说明：

位置	参数名	类型	单位	范围	说明
1	start_v	float	V	-3.0~3.0	起始电位
2	end_v	float	V	-3.0~3.0	结束电位
3	scan_dir	int	-	1/-1	扫描方向（1=正向，-1=负向）
4	scan_rate	float	V/s	0.001~10.0	扫描速率
5	start_v2	float	V	-3.0~3.0	第二扫描起始点
6	cycles	int	-	1~100	循环次数
7	vertex_v	float	V	-3.0~3.0	顶点电位（-1默认自动）
8-11	param8-11	int	-	-	保留参数
12	scan_interval	float	V/s	0.001~10.0	采样间隔
13	param13	int	-	-	保留参数
14-15	current_range	int	μA	1~1000	电流量程
16-18	param16-18	int	-	-	控制参数

示例： P -1.0,1.0,1,0.2,-1.0,2,-1,0,0,10,100,0.2,20,50,50,2,0,1,

2. 启动测试命令 (S - Start)

格式： S

说明： 单个字符 S， 用于启动参数中配置的CV测试

发送方式： 原始字节发送（不需要回车符）

响应格式

1. 参数确认响应 (#)

格式: #\r\n

含义: 设备已收到并解析参数, 参数配置成功

期望后续状态: PARAMETER_SET

2. 开始响应 (*)

格式: *\r\n

含义: 设备已开始接收测试数据, 测试正在进行中

期望后续状态: RECEIVING_DATA

3. 完成响应 (@)

格式: @\r\n

含义: 数据接收完成, 测试结束

期望后续状态: TEST_COMPLETE

4. 数据行 (Data)

格式: ,, \r\n

字段说明:

字段	类型	单位	格式	说明
voltage	float	V	-X.XXXX	当前电位值
current	float	μA	X.XXXX	当前电流值

示例:

-1.0000,2.5432,
0.0000,5.2341,
1.0000,-0.1234,

发送频率: 约 16 Hz (每 62ms 一条数据)

5. 错误响应

目前协议没有定义特殊错误响应, 错误通过状态转移和超时来处理。

完整通信流程

标准测试流程

主机端通信流程：

- 1. 建立串口连接
- 2. 发送参数设置命令 P <参数列表>，
- 3. 等待参数确认响应 #\r\n
- 4. 发送启动命令 S
- 5. 等待开始响应 *\r\n
- 6. 循环接收数据行 ,, \r\n
- 7. 接收完成响应 @\r\n
- 8. 断开连接

时序示例

时间	事件	说明
00:00	连接设备	建立串口连接
00:01	发送参数	发送 P 命令
00:02	收到确认	收到 # 响应
00:03	发送启动	发送 S 命令
00:04	收到开始	收到 * 响应
00:05-00:25	接收数据	约 320 条数据行 (20秒 × 16Hz)
00:25	收到完成	收到 @ 响应
00:26	处理数据	保存和处理数据

错误处理

常见错误情况

错误	原因	解决方案
连接失败	串口不存在或被占用	检查端口号，关闭其他应用
参数确认超时	设备未正确响应	检查波特率，重启设备
数据不完整	测试中断或超时	增加 timeout 值，检查设备
无效数据格式	串口数据损坏	检查波特率和线缆

超时处理

数据接收超时可分为两类：

- **参数确认超时**：5秒内未收到 # 响应表示参数设置失败
- **测试数据超时**：60秒内未收到 @ 响应表示测试中断

异常处理

设备应在以下情况下返回错误或异常状态：

- 参数值超出范围
- 串口通信错误
- 测试过程中设备故障
- 数据接收中断

数据格式

CSV 输出格式

标准 CSV 格式包含两列：电位(V) 和 电流(μ A)

示例数据行：

-1.0000,2.5432
-0.9999,2.4521
-0.9998,2.3641
1.0000,-0.1234

数据特征

- **数据点数量**：取决于扫描速率和时间
 - 例如：0.2 V/s 的扫描速率，范围 -1V 到 1V（共 2V），采样间隔 0.2 V/s
 - 总时间 $\approx 10 \text{ 秒} \times 2 \text{ 次循环} = 20 \text{ 秒}$
 - 数据点数 $\approx 20 \text{ 秒} \times 16 \text{ Hz} = 320 \text{ 个}$
- **数据精度**：4 位小数（0.0001 级别）

文件命名规则

- **数据文件**：`cv_data_YYYYMMDD_HHMMSS.csv`
 - **图形文件**：`cv_curve_YYYYMMDD_HHMMSS.png`
-

技术指标

性能

指标	值
最大数据点率	16 Hz
数据精度	4 位小数
响应延迟	< 100 ms
最大并发测试	1
内存占用	< 10 MB

兼容性

项目	值
Python 版本	3.7+
操作系统	Windows, Linux, macOS
串口接口	RS232, USB Virtual COM

附录

A. 参数范围表

参数	最小值	最大值	单位	说明
start_v	-3.0	3.0	V	起始电位
end_v	-3.0	3.0	V	结束电位
scan_rate	0.001	10.0	V/s	扫描速率
cycles	1	100	-	循环次数
current_range	1	1000	μA	电流量程

B. 状态转移表

当前状态	事件	下一状态
IDLE	设备连接	IDLE
IDLE	发送参数	PARAMETER_SET
PARAMETER_SET	收到 #	WAITING_ACK

当前状态	事件	下一状态
WAITING_ACK	确认完成	PARAMETER_SET
PARAMETER_SET	发送开始命令	STARTING_TEST
STARTING_TEST	收到 *	RECEIVING_DATA
RECEIVING_DATA	接收数据行	RECEIVING_DATA
RECEIVING_DATA	收到 @	TEST_COMPLETE
任何状态	异常	ERROR