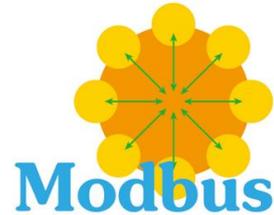


# Modbus 协议在 SDWb 串口屏上的实现方法

## ——Lua 入门概述 2

Modbus 是为使用 PLC 通信而制定的一种串行通信协议。目前已经成为工业自动化领域通信协议的业界标准，是工业控制现场不同电子设备之间常用的连接方式。



SDWb 系列串口屏通过 Lua 脚本编程功能，提供了 Modbus 协议专用的 API 功能函数，全面实现对 Modbus 串口协议的支持，包括 Modbus RTU 主机、Modbus RTU 从机、Modbus ASCII 主机、Modbus ASCII 从机等多种模式，方便串口屏与各种 PLC 等不同电子设备的连接。

### 一. 串口工作模式设置

通过下述 API 接口函数，设置串口工作模式。正确设置串口号、以及串口工作协议等。串口 0 默认为 VGUS 指令集协议，可以配置为 Modbus 协议。串口 1 不支持 Modbus 协议。

```
com_set_work_mode(com_num, work_protocol, baudrate, format)
```

com\_num: 串口编号。编号范围 0-1。

work\_protocol: 设置工作协议。//仅串口 0 支持设置，串口 1 固定为自由协议

0: VGUS 指令集协议。//串口 0 默认模式

1: 自定义串口协议。

2: ModbusRTU 从机。

3: ModbusASCII 从机。

4: ModbusRTU 主机。

5: ModbusASCII 主机。

baudrate: 设置串口波特率。

com\_format: 设置串口格式。

### 二. Modbus 主机 API 函数

串口 0 工作在 Modbus 主机模式，通过调用下列 API 接口函数，主动触发一次 Modbus 串口指令传输。

序号	功能	API 函数
1	获取 MODBUS 异常码	modbus_get_exception_code()
2	设置应答超时时间	modbus_set_timeout (timeout)



3	读线圈（01 功能码）	modbus_fun_01(slave, addr, quantity, coils)
4	读离散输入（02 功能码）	modbus_fun_02(slave, addr, quantity, input)
5	读保持寄存器（03 功能码）	modbus_fun_03(slave, addr, quantity, regs)
6	读输入寄存器（04 功能码）	modbus_fun_04(slave, addr, quantity, input_regs)
7	写单个线圈（05 功能码）	modbus_fun_05(slave, addr, status)
8	写单个寄存器（06 功能码）	modbus_fun_06(slave, addr, reg)
9	写多个线圈（15 功能码）	modbus_fun_15(slave, addr, quantity, coils)
10	写多个寄存器（16 功能码）	modbus_fun_16(slave, addr, quantity, regs)
11	屏蔽写寄存器（22 功能码）	modbus_fun_22(slave, addr, And_Mask, Or_Mask)

### 三. Modbus 从机实现方法

串口 0 工作在 Modbus 从机模式，仅需设置从机地址、和串口工作模式。当串口工作模式设置为 Modbus 从机模式时，SDWb 串口屏会根据接收到的 Modbus 指令，自动做相应的应答处理，整个过程无需 lua 脚本参与。

设置 Modbus 从机设备地址的 API 函数：`modbus_set_slaveaddr (salve)`

slave: 作为 Modbus 从机（RTU 或 ASCII）使用时的设备 ID。范围 1-255。默认值 1。

返回值：无。

注意事项：该 API 需要在初始化回调中调用。仅设置一次。

参考文档：

《基于 VGUS 的 Lua 脚本使用说明》

《Lua 入门概述 1——Lua 脚本可以在 SDWb 串口屏上实现哪些功能》

