



一、扩展通讯协议解析器控件介绍及脚本写法

（一）图标



（二）属性栏

	属性	功能
	Id	控件在当前页面的 id，不可读写
	Global	全局属性，勾选后全局可调用，其他界面调用时，需要在控件名前面加页名称
	Name	控件名字，编辑工程时可以更改，脚本不可读写
	RxMaxlen	解析器用于接受数据缓存的数组的最大长度，脚本不可读写
	BaudRate	扩展串口的波特率，脚本可读写
	Parity	扩展串口校验位，脚本可读写 0 无校验 1 奇校验 2 偶校验
	Stopbits	扩展串口停止位，脚本可读写 1 停止位 1，2 停止位 2，3 停止位 1.5
	Workmode	扩展串口通信协议，可选自定义或者 modbus 协议，选择 modbus 从机时，需要设置从机地址
	modbusType	主机

（三）脚本写法示例

（以扩展串口名称为 expt10 为例，如图标所示）

- 扩展串口 Modbus 主站发送读取帧（modbusRead）

例如，从设备地址 0x01，功能码 0x03，寄存器起始地址 0x0000，寄存器数量 2，超时时间 50ms

发送格式 `expt10.modbusRead(0x01,0x03,0x00,2,50);`

- 扩展串口 modbus 主站发送写入帧写入数值（expt10.modbusWrite）

例如，从机设备地址 0x01，功能码 06，写入寄存器地址 0x0064，写入值 2，超时时间

发送格式 `expt10.modbuswrite(0x01,0x06,0x0064,0x02,50)`

50ms



- 扩展串口 modbus 主站发送写入帧写入数组（`expt10.modbusWrites`）
例如 从设备地址 0x01, 功能码 0x03, 寄存器起始地址 0x0000, 寄存器数量 2, 数组 `arr[4]`值, 数组长度 4 超时时间 50ms
发送格式 `byte arr[4]={0x01,0x02,0x03,0x04};`
`expt10.modbusWrites(0x01,0x06,0x00,2,arr,4,50);`
- 扩展串口 modbus 从机发送数据（`expt10.modbusSendBytes`）
例如 从机地址 0x01,功能码 0x03, 发送的字节数组 `arr[4]`,发送的数组长度 4, 超时时间 50ms
发送格式 `byte arr[4]={0x01,0x02,0x03,0x04};`
`expt10.modbusSenBytes(0x01,0x03,arr,4,50);`
- Modbus 函数返回值为 0 或者 -1, 用户使用 modbus 发送时一定要读取返回值, 返回为 0 正常执行解析脚本, 返回值为 -1 则通信失败不执行解析脚本
- 扩展串口自定义协议发送字符串（`expt10._uartsend`）
例如 扩展串口发送字符串 “sanyscreen”
格式 `expt10._uartSend(“sanyscreen”)`
- 扩展串口自定义协议发送字节数组（`expt10._uartSendBytes`）
例如 扩展串口发送字节数组 `arr[4]`,起始地址 0, 数组长度 4
格式 `byte arr[4]={0x01,0x02,0x03,0x04};`
`Expt10._uartSendBytes(arr,0,4);`

（四）控制属性（脚本可用）

扩展串口名以 `expt10` 为例

- `Expt10.modbusread` 扩展串口 modbus 发送读取帧
- `Expt10.modbuswrite` 扩展串口 modbus 发送写入帧写入数值
- `Expt10.modbuswrites` 扩展串口 modbus 发送写入帧写入数组
- `Expt10.modbussendbytes` 扩展串口 modbus 从机发送数据
- `Expt10._uartsend` 扩展串口自定义协议发送字符串
- `Expt10._uartsendbytes` 扩展串口自定义协议发送字节数组
- `Expt10.rxbuf` 扩展串口用来缓存接收数据的数组, 该数组为扩展协议解析器固定属性, 不可赋值, 不可清除, 每当扩展串口有数据接收到都会自动缓存新的数据, 旧数据会被清零
- `Expt10.rxlens` `expt10.rxbuf` 数组的长度, 只读属性
- `Expt10.fcode` 扩展串口做 modbus 从站时此属性为主站的下发报文的功能码
- `Expt10.caddr` 扩展串口做 modbus 从站时的地址 可在属性栏配置, 可读写
- `Expt10.addr` 扩展串口接收的数据帧中的地址

（五）通信调试

由于目前在 `vp` 调试界面, 主串口与扩展串口输入输出区域没有做区分, 所以调试时只能同时调试一个串口, 同时调试两个串口请将工程下载到串口屏利用串口助手进行调试, 后续版本的 `vp` 会优化这个功能, 届时可在 `vp` 模拟调试更加方便

（六）常见问题

- 调用扩展串口时, 必须要拖一个扩展协议解析器至特殊控件栏才可使用



- 扩展串口做 modbus 主站，对从站轮询时，可以用定时器定时发送轮询报文
- 工程项目设置里设置的主 从机，从机地址，波特率停止位，校验位都只能设置主串口的参数，扩展串口这些参数设置，都要在扩展协议解析器的控件属性设置，部分可脚本读写设置。
- 扩展串口不能接收识别系统指令，只能使用自定义协议或者 modbus 协议进行通讯
- 采用 modbus 协议通信时出函数写法格式有区别，modbus 协议使用与主串口没有区别，都是采用标准 modbus 协议。
- Modbus.rxbuf 数组第一位为 modbus 协议数据域第一个数据，主站接收时，rxbuf[0]为数据长度，从站接收时 rxbuf[0]为寄存器地址的第一个字节。如果做从站时接受的数据不对，可以用串口助手获取 485 总线的数据进行分析。
- 当工程编译时显示工程路径被拒绝访问，此时找到 pix 工程文件右键属性，将只读属性关闭即可