



DNP3.0

服务器端开发包接口 API 使用说明书

版本：V2.0

深圳天勺电力软件有限公司

深圳 中国

1. 开发包简介

该开发包根据已经正式发行的 DNP3.0 规约 最新版本开发而成，采用 C 语言编写，可跨平台使用。

为了最大限度地简化使用，完全封装了 DNP3.0 规约 中的诸多底层细节和各种复杂模型的实现逻辑。用户利用它作开发的时候无需了解底层细节，只需要调用功能接口函数完成相应的功能研发，具有方便、快捷的优点。

本接口说明文档为接口全集，客户可以根据具体业务选择相应的接口进行调用。本开发包用在开发基于 DNP3.0 规约的所有设备中。

现场运行稳定可靠，调用简单高效。

2. 接口描述

2.1 服务应用函数

函数声明	int SZTS_Server_InitAllCfg (void);
功能	获取监控相关配置和全局资源初始化接口，在 DNP 服务前调用
参数	无
返回值	0 - 成功; -1 - 失败
示例	SZTS_Server_InitAllCfg ();

函数声明	int SZTS_Server_MemFree (void);
功能	DNP 资源区释放，在程序退出时调用
参数	无
返回值	0 - 成功; -1 - 失败
示例	SZTS_Server_MemFree ();

函数声明	void SZTS_Server_InitPortCtrl (BYTE byPortNo, BOOLEAN bFirst)
功能	DNP 端口初始化函数，在 SZTS_Server_InitAllCfg 后，DNP 主服务前调用
参数	byPortNo:是指 Port.cfg 配置文件中的端口编号 PortNo。 bFirst: bFirst=TRUE:第一次初始化
返回值	无
示例	SZTS_Server_InitPortCtrl (1 , TRUE);

函数声明	void SZTS_Server_TcpInitial()
功能	DNP 服务 TCP 连接初始化，在 SZTS_Server_InitAllCfg 后，DNP 主服务前调用
参数	无
返回值	无

示例	SZTS_Server_TcpInitial();
----	---------------------------

函数声明	void SZTS_Server_UdpInitial()
功能	DNP 服务 UDP 连接初始化, 在 SZTS_Server_InitAllCfg 后, DNP 主服务前调用
参数	无
返回值	无
示例	SZTS_Server_UdpInitial();

函数声明	void SZTS_Server_initUartStruct ()
功能	DNP 服务串口连接初始化, 在 SZTS_Server_InitAllCfg 后, DNP 主服务前调用
参数	无
返回值	无
示例	SZTS_Server_initUartStruct ();

函数声明	void SZTS_Server_Init (BYTE byPortNo)
功能	DNP 规约变量初始化函数
参数	byPortNo: 是指 Port.cfg 配置文件中的端口编号 PortNo - 1
返回值	无
示例	SZTS_Server_Init (0);

函数声明	int SZTS_Server_NetMsgRecv(UINT iTcpSockNo, BYTE *pbyMsgBuf, BYTE *pbyParamTag)
功能	TCP 报文接收主函数, 对接收报文进行初步判断
参数	UINT iTcpSockNo: 当前 TCP 连接号 BYTE *pbyMsgBuf: 读取的报文内容 BYTE *pbyParamTag: 未用
返回值	接收报文字节数
示例	SZTS_Server_NetMsgRecv(nTcpNo, pConn->pReadBuff, pConn->pReadPara);

函数声明	void SZTS_Server_UDPRecieve(BYTE byPortNo)
功能	UDP 报文接收主函数, 对接收报文进行初步判断
参数	byPortNo: 是指 Port.cfg 配置文件中的端口编号 PortNo - 1
返回值	无
示例	SZTS_Server_UDPRecieve(0);

函数声明	SZTS_Server_Recieve (BYTE byPortNo)
功能	串口报文接收主函数, 对接收报文进行初步判断
参数	byPortNo: 是指 Port.cfg 配置文件中的端口编号 PortNo - 1
返回值	无
示例	SZTS_Server_Recieve (0);

函数声明	void SZTS_Server_Proc(BYTE byPortNo)
功能	DNP 数据报文处理函数，用以处理接收的相关命令报文
参数	byPortNo: 是指 Port.cfg 配置文件中的端口编号 PortNo - 1
返回值	无
示例	SZTS_Server_Proc (0);

函数声明	void SZTS_Server_Send(BYTE byPortNo)
功能	DNP 数据报文发送函数，用以发送相关命令的响应报文
参数	byPortNo: 是指 Port.cfg 配置文件中的端口编号 PortNo - 1
返回值	无
示例	SZTS_Server_Send (0);

函数声明	BOOLEAN Event_GenerateRecordCOS(WORD wChannel, WORD wIndexYX, BYTE byStatusYX)
功能	生成一条遥信变位 COS 记录
参数	WORD wChannel: 遥信变位点的通道号 WORD wIndexYX: 遥信变位点在 Map 表中的序号 BYTE byStatusYX: 遥信变位点状态
返回值	1-产生 COS 变位; 0-未发生 COS 变位
示例	Event_GenerateRecordCOS(channelYX, indexYX, status);

函数声明	BOOLEAN Event_GenerateRecordSOE(WORD wChannel, WORD wIndexYX, BYTE byStatus, tagPTimeBCD ptTimeAct)
功能	生成一条遥信变位 SOE 记录（带时标）
参数	WORD wChannel: 遥信变位点的通道号 WORD wIndexYX: 遥信变位点在 Map 表中的序号 BYTE byStatus: 遥信变位点状态 tagPTimeBCD ptTimeAct: 遥信变位时间
返回值	1-产生 SOE 变位; 0-未发生 SOE 变位
示例	Event_GenerateRecordSOE(channelYX, indexYX, status, &time);

2.2 装置接口回调函数

函数原型定义	typedef BYTE (*getFlagYCFun)(DWORD);
函数全局变量	getFlagYCFun getFlagYC_fun;
函数功能	通过 Map.cfg 配置文件中的通道号获取相应遥测点的品质值
注册回调方法	int regist_getFlagYCFun(getFlagYCFun f)
原型参数值	DWORD channel: Map.cfg 配置文件中的遥测点位的通道号
原型返回值	BYTE: 返回具体回调函数中的遥测品质
示例	BYTE demo(DWORD channel) {...}; regist_getFlagYCFun (demo);

函数原型定义	typedef WORD (*getValueYCFun)(DWORD);
函数全局变量	getValueYCFun getValueYC_fun;
函数功能	通过 Map.cfg 配置文件中的通道号获取相应的遥测值
注册回调方法	int regist_getValueYCFun(getValueYCFun f)
原型参数值	DWORD channel: Map.cfg 配置文件中的遥测点位的通道号
原型返回值	WORD: 返回具体回调函数中的遥测值或本地值
示例	BYTE demo(DWORD channel) {...}; regist_getValueYCFun (demo);

函数原型定义	typedef BYTE (*getFlagYXFun)(DWORD);
函数全局变量	getFlagYXFun getFlagYX_fun;
函数功能	通过 Map.cfg 配置文件中的通道号获取相应遥信点的品质值
注册回调方法	int regist_getFlagYXFun(getFlagYXFun f)
原型参数值	DWORD channel: Map.cfg 配置文件中的遥信点位的通道号
原型返回值	BYTE: 返回具体回调函数中的遥信品质
示例	BYTE demo(DWORD channel) {...}; regist_getFlagYXFun (demo);

函数原型定义	typedef BYTE (*getValueYXFun)(DWORD);
函数全局变量	getValueYXFun getValueYX_fun;
函数功能	通过 Map.cfg 配置文件中的通道号获取相应的遥信值
注册回调方法	int regist_getValueYXFun(getValueYXFun f)
原型参数值	DWORD channel: Map.cfg 配置文件中的遥信点位的通道号
原型返回值	BYTE: 返回具体回调函数中的遥信值或本地值
示例	BYTE demo(DWORD channel) {...}; regist_getValueYXFun (demo);

函数原型定义	typedef BYTE (*getFlagYMFun)(DWORD);
函数全局变量	getFlagYMFun getFlagYM_fun;
函数功能	通过 Map.cfg 配置文件中的通道号获取相应遥脉点的品质值
注册回调方法	int regist_getFlagYMFun(getFlagYMFun f)
原型参数值	DWORD channel: Map.cfg 配置文件中的遥脉点位的通道号
原型返回值	BYTE: 返回具体回调函数中的遥脉品质
示例	BYTE demo(DWORD channel) {...}; regist_getFlagYMFun (demo);

函数原型定义	typedef DWORD (*getValueYMFun)(DWORD);
函数全局变量	getValueYMFun getValueYM_fun;
函数功能	通过 Map.cfg 配置文件中的通道号获取相应的遥脉值
注册回调方法	int regist_getValueYMFun(getValueYMFun f)
原型参数值	DWORD channel: Map.cfg 配置文件中的遥脉点位的通道号
原型返回值	DWORD: 返回具体回调函数中的遥脉值或本地值
示例	BYTE demo(DWORD channel) {...};

	regist_getValueYMFun (demo);
--	------------------------------

函数原型定义	typedef BOOLEAN (*judgeAllowRemoteCtrl)(DWORD);
函数全局变量	judgeAllowRemoteCtrl judgeAllowRemoteCtrl_fun;
函数功能	通过 Map.cfg 配置文件中的通道号判断该遥控点是否能够被设置
注册回调方法	int regist_JudgeAllowRemoteCtrl(judgeAllowRemoteCtrl f)
原型参数值	DWORD channel: Map.cfg 配置文件中的遥控点位的通道号
原型返回值	TRUE-能被设置; FALSE-不允许设置
示例	BOOLEAN demoCheckEnableYkCmd(DWORD channel) {...}; regist_JudgeAllowRemoteCtrl(demoCheckEnableYkCmd);

函数原型定义	typedef BOOLEAN (*judgeAllowRemoteSel)(DWORD, BYTE);
函数全局变量	judgeAllowRemoteSel judgeAllowRemoteSel_fun;
函数功能	通过 Map.cfg 配置文件中的通道号判断该遥控点进行遥控选择是否成功
注册回调方法	int regist_JudgeAllowRemoteSel(judgeAllowRemoteSel f)
原型参数值	DWORD channel: Map.cfg 配置文件中的遥控点位的通道号 BYTE Value: 下发遥控选择状态 (合/分)
原型返回值	TRUE-进行遥控选择成功; FALSE-遥控选择失败
示例	BOOLEAN demoCheckEnableYkSel(DWORD channel, BYTE Value) {...}; regist_JudgeAllowRemoteSel(demoCheckEnableYkSel);

函数原型定义	typedef BOOLEAN (*judgeAllowRemoteExe)(DWORD, WORD);
函数全局变量	judgeAllowRemoteExe judgeAllowRemoteExe_fun;
函数功能	通过 Map.cfg 配置文件中的通道号判断该遥控点进行遥控执行是否成功
注册回调方法	int regist_JudgeAllowRemoteExe(judgeAllowRemoteExe f)
原型参数值	DWORD channel: Map.cfg 配置文件中的遥控点位的通道号 WORD Index: 进行遥控选择时的开关序号
原型返回值	TRUE-遥控执行成功; FALSE-遥控执行失败
示例	BOOLEAN demoCheckEnableYkExe(DWORD channel, WORD Index){...}; regist_JudgeAllowRemoteExe(demoCheckEnableYkExe);

函数原型定义	typedef BOOLEAN (*judgeAllowSetCtrl)(DWORD);
函数全局变量	judgeAllowSetCtrl judgeAllowSetCtrl_fun;
函数功能	通过 Map.cfg 配置文件中的通道号判断该遥设点是否能够被设置
注册回调方法	int regist_JudgeAllowSetCtrl(judgeAllowSetCtrl f)
原型参数值	DWORD channel: Map.cfg 配置文件中的遥设点位的通道号
原型返回值	TRUE-能被设置; FALSE-不允许设置
示例	BOOLEAN demoCheckEnableSetCmd(DWORD channel) {...}; regist_JudgeAllowSetCtrl(demoCheckEnableSetCmd);

函数原型定义	typedef BOOLEAN (*judgeAllowSetSel)(DWORD, DWORD);
函数全局变量	judgeAllowSetSel judgeAllowSetSel_fun;

函数功能	通过 Map.cfg 配置文件中的通道号判断该遥设点进行选择是否成功
注册回调方法	int regist_JudgeAllowSetSel(judgeAllowSetSel f)
原型参数值	DWORD channel: Map.cfg 配置文件中的遥设点位的通道号 DWORD Value: 下发设点值
原型返回值	TRUE-设点选择成功; FALSE-设点选择失败
示例	BOOLEAN demoCheckEnableSetSel(DWORD channel, DWORD Value) {...}; regist_JudgeAllowSetSel(demoCheckEnableSetSel);

函数原型定义	typedef BOOLEAN (*judgeAllowSetExe)(DWORD, DWORD);
函数全局变量	judgeAllowSetExe judgeAllowSetExe_fun;
函数功能	通过 Map.cfg 配置文件中的通道号判断该遥设点进行设点执行是否成功
注册回调方法	int regist_JudgeAllowSetExe(judgeAllowSetExe f)
原型参数值	DWORD channel: Map.cfg 配置文件中的遥设点位的通道号 DWORD Index: 进行设点选择时的对象号
原型返回值	TRUE-设点执行成功; FALSE-设点执行失败
示例	BOOLEAN demoCheckEnableSetExe(DWORD channel, DWORD Index){...}; regist_JudgeAllowSetExe(demoCheckEnableSetExe);