SDLC-BOARD

高速嵌入式开发板

Rev.2020.1215

用户手册

手机版官网

电话：400-025-5057
网址：www.yacer.cn

yacer 亚册
让通信如搭积木般简单

目录

1 概述 1
1.1 简介 1
1.2 应用 1
1.3 特点 2
1.4 功能框图 2
1.5 技术规格 3
1.6 订购信息 4
2 硬件与物理接口 5
2.1 接线方式 5
2.1.1 LG：端子直接接线 5
2.1.2 RM：水平插头接线 5
2.1.3 VM：垂直插头接线 5
2.2 在板 LED 状态指示 6
2.3 以太网接口 ETH1、ETH2：X0 6
2.3.1 功能描述 6
2.3.2 X0 管脚定义 6
2.4 同异步串口 S1：X1 7
2.4.1 功能描述 7
2.4.2 无时钟模式 X1 管脚定义 7
2.4.3 带时钟模式 X1+X2 管脚定义 8
2.4.4 RS-485 终端匹配 8
2.5 同异步串口 S2：X2 9
2.5.1 功能描述 9
2.5.2 X2 管脚定义 9
2.5.3 RS-485 终端匹配 9
2.6 扩展接口 S3：X3 10
2.6.1 功能描述 10
2.6.2 X3 管脚定义 10
2.6.3 CAN 总线终端匹配 10
2.6.4 RS-485 终端匹配 10
2.7 X4：电源/复位/LED 指示接口 11
2.8 X5：DMS-UART 配置管理接口 11
3 构建配置环境 12
3.1 获取配置管理软件 yacer-DMS 12
3.2 运行 yacer-DMS 软件 12
3.3 软件主界面 12
3.4 统计报告 13
3.4.1 控制面板 13
3.4.2 收发指示面板 13
3.4.3 信息显示面板 13

3.5 配置设备 14
3.6 通过以太网口进行配置 15
3.7 通过 DMS-UART 接口进行配置 15
3.8 设置主机 DMS 接口 16
4 功能及配置 17
4.1 以太网接口（ETH1、ETH2） 17
4.1.1 设备别名 17
4.1.2 IP 地址配置 17
4.1.3 默认网关 19
4.2 串口 20
4.2.1 无时钟模式工作方式 20
4.2.2 带时钟模式工作模式 21
4.2.3 波特率 21
4.2.4 同步串口编码格式 22
4.2.5 HDLC-NRZ 参数配置 23
4.2.6 HDLC-NRZI 参数配置 27
4.2.7 HDLC-DBPL 参数配置 27
4.2.8 HDLC-DiffMAN（差分曼彻斯特）参数配置 28
4.2.9 HDLC-MAN（曼彻斯特）参数配置 28
4.2.10 同步 Bit 流参数 29
4.2.11 异步 UART 参数配置 30
4.2.12 异步 HDLC 参数配置 31
4.2.13 CAN 口配置 32
4.3 UDP 转串口配置 33
4.3.1 功能描述 33
4.3.2 协议转换 33
4.3.3 转发配置 34
4.3.4 收 UDP 组播 34
4.4 串口转 UDP 配置 35
4.4.1 功能描述 35
4.4.2 协议转换 35
4.4.3 转发配置 36
4.4.4 UDP Server 如何识别源串口 37
4.5 串口转串口配置 38
5 系统维护 39
5.1 固件版本更新 39
5.1.1 开始更新 39
5.1.2 找到固件版本 39
5.1.3 更新完成 40
5.1.4 更新确认 40
5.2 复位设备 41
5.3 Ping 41

6 机械特性与安装 42
6.1 安装 42
6.2 LG 单板尺寸 42
6.3 RM 单板尺寸 43
6.4 VM 单板尺寸 43
7 协议转换应用 44
7.1 串口数据转换 44
7.1.1 应用数据包与转换模型 44
7.1.2 UDP 报文格式 45
7.1.3 HDLC 帧格式 45
7.1.4 UART 数据包格式 45
7.1.5 UART-HDLC 帧格式 46
7.2 CAN 帧数据转换 47
7.2.1 CAN 数据包与转换模型 47
7.2.2 CAN 数据包格式 48
8 在板编程二次开发 50
8.1 双 CPU 构架 50
8.2 数据交互模型 50
8.3 DMS 支持 51
8.3.1 显示应用 CPU 运行信息 51
8.3.2 应用 CPU 接收 UDP 报文的配置 51
8.3.3 应用 CPU 接收串口数据的配置 52
8.3.4 加载应用 CPU 软件版本 52
8.3.5 加载应用 CPU 的配置 53
8.4 软件开发与编译 53

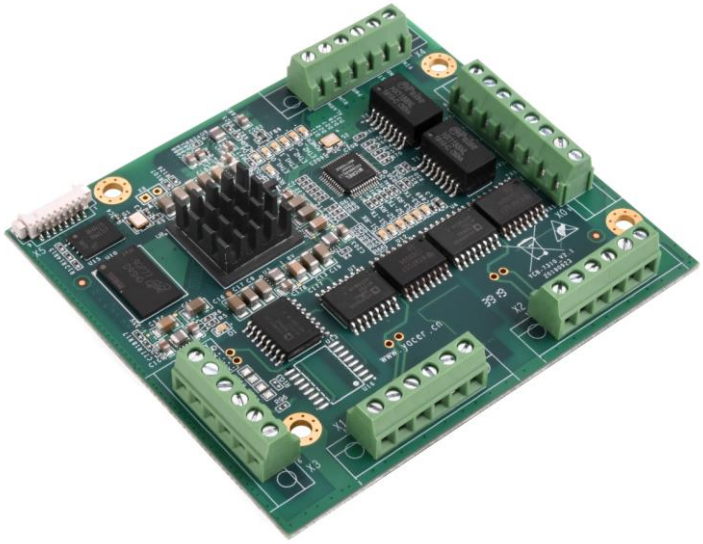
1 概述

1.1 简介

亚册 SDLC-BOARD 嵌入式开发板，提供全特性高速 HDLC/SDLC 同步串口、UART 异步串口、CAN 接口、以及以太网接口，实现接口之间的协议转换。

采用小尺寸、端子接口设计，特别方便用户进行系统集成。

板载应用 CPU 供用户在板编程二次开发，实现业务软件与通信软件一体化无缝结合。



1.2 应用

- 同步串口、异步串口互相转换
- 串口、以太网之间的协议转换
- CAN 口、串口之间数据转换
- CAN 总线、以太网接口转换
- 列车 TCMS 系统、列车机车通信网络（TCN）
- 遥测、测控数据采集传输
- 卫星、电台数据传输
- 航空、航天数据通信
- 嵌入式应用与开发

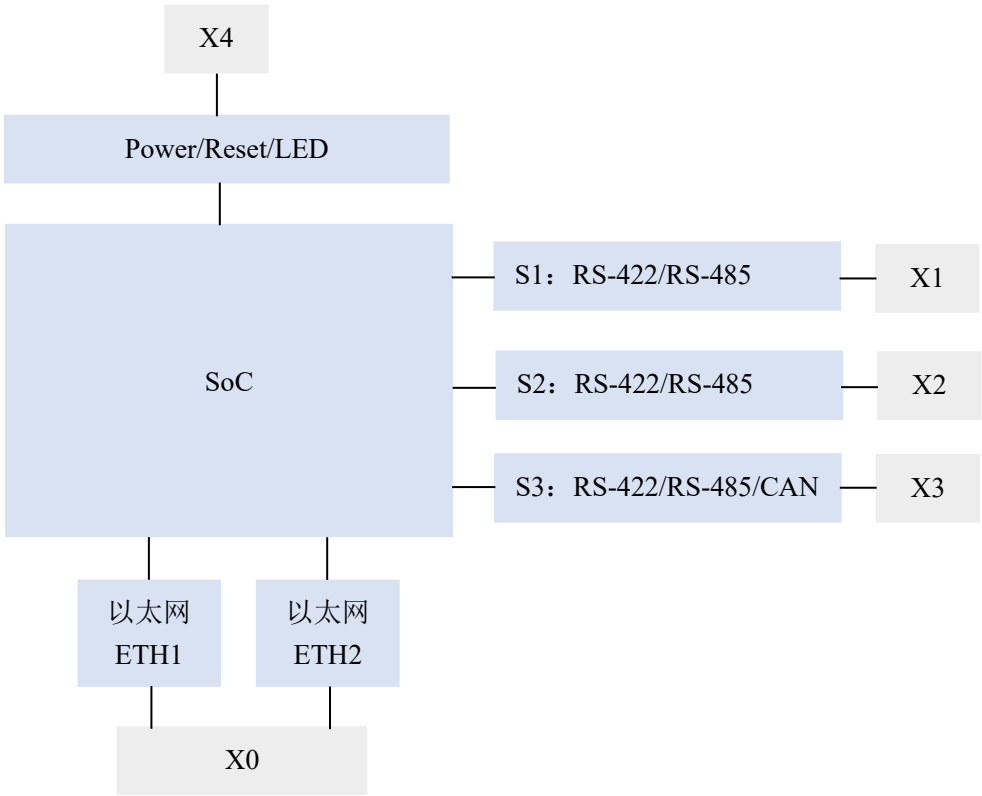
1.3 特点

- 2 路 10/100M 以太网接口，支持以太网交换、双 IP 功能
- 2 路带隔离同异步串口，可选 RS-422 全双工或 RS-485 半双工
- 1 路带隔离扩展接口，可选 RS-422、RS-485 串口或 CAN 总线接口
- 所有串口均支持同步 HDLC 协议、异步 UART 工模式
- 支持 NRZ、NRZI、曼彻斯特、差分曼彻斯特、DBPL 等编码格式
- 板载应用 CPU 供用户在板编程二次开发
- 完备的隔离保护，工业级宽温
- 易集成：尺寸小、端子接口、5～17V 宽压供电

1.4 功能框图

基本功能框图如下图所示，其中：

- X0：为 2 路百兆以太网口 ETH1 和 ETH2；
- X1：同异步串口 S1，可选 RS-422 或 RS-485；
- X2：同异步串口 S2，可选 RS-422 或 RS-485；
- X3：扩展接口 S3，可选 CAN、RS-422 或 RS-485；
- X4：电源、复位及 LED 指示。



1.5 技术规格

同异步串口 S1、S2	
连接器	3.81mm 端子
可选接口类型 （二选一）	RS-485 半双工带隔离串口 RS-422 全双工带隔离串口
工作模式	同步 HDLC、异步 UART、Bit 流
编码格式	NRZ、NRZI、DBPL（Differential Bi-Phase Level）、 曼彻斯特（Manchester）、差分曼彻斯特（Differential Manchester）
波特率	同步 NRZ ≤ 12 Mbps；同步其他 ≤ 6 Mbps 异步 ≤ 3 Mbps
隔离保护	2.5 kVrms
扩展接口 S3	
连接器	3.81mm 端子
可选接口类型 （三选一）	CAN 总线隔离接口（CAN 2.0A、CAN2.0B，ISO 11898） RS-422 全双工带隔离串口 RS-485 半双工带隔离串口
工作模式	同步 HDLC、异步 UART、Bit 流
编码格式	NRZI、DBPL、曼彻斯特、差分曼彻斯特
波特率	串口：同步 ≤ 6 Mbps；异步 ≤ 3 Mbps CAN：50 Kbps ~ 1 Mbps
隔离保护	2.5 kVrms
以太网接口 ETH1、ETH2	
连接器	3.81mm 端子
功能	支持以太网交换、双 IP
速率	10/100 Mbps，支持 MDI/MDIX 自适应
网络协议	TCP/IP
编程接口	UDP Server、UDP Client，支持单播/组播/广播
隔离保护	1.5 kVrms
在板编程二次开发支持	
应用 CPU	ARM Cortex-A9 处理器，主频 250 MHz
内存	DDR3，128 MB
FLASH	6 MB 版本空间，1 MB 配置空间
数据接口	与通信 CPU 基于共享内存交互数据
配置管理	
配置软件	yacer-DMS 配置管理软件
配置接口	以太网接口 DMS-UART 接口（借助于亚册 DMS-UART-8P 配置线）
电源	
供电电压	4.9 ~ 17 VDC
功耗	< 3 W
电源接口	3.81mm 端子

机械特性	
尺寸	长 x 宽：90 mm x 80 mm
重量	100g
工作环境	
工作温度	-40～+75℃
存储温度	-40～+85℃
工作湿度	5～95% RH（无凝结）

注：如用户需要更高的串口速率，请联系厂家。

1.6 订购信息

SDLC-BOARD -	5	5	0	-	LG
X1 接口定义：					
● 全双工 RS-422 串口	4				
● 半双工 RS-485 串口	5				
X2 接口定义：					
● 全双工 RS-422 串口		4			
● 半双工 RS-485 串口		5			
X3 接口定义：					
● 无			0		
● 全双工 RS-422 串口			4		
● 半双工 RS-485 串口			5		
● CAN 总线接口			6		
接线方式：					
● 端子直接接线					LG
● 水平插头接线					RM
● 垂直插头接线					VM

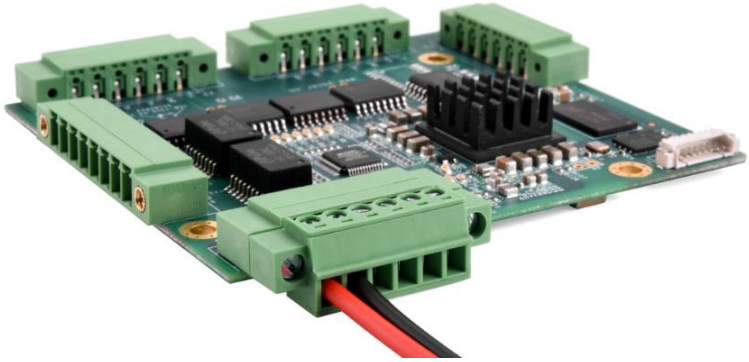
2 硬件与物理接口

2.1 接线方式

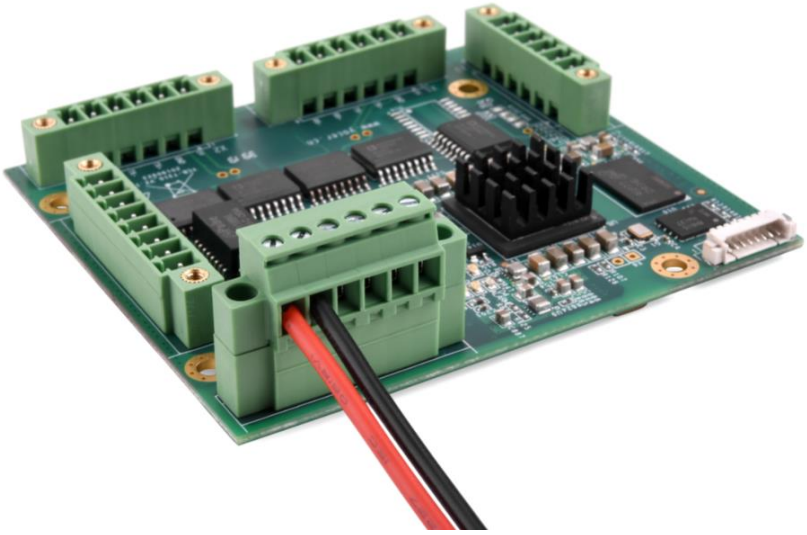
为了适应不同的应用场景，SDLC-BOARD 提供三种不同的接线方式。

2.1.1 LG：端子直接接线

2.1.2 RM：水平插头接线



2.1.3 VM：垂直插头接线



2.2 在板 LED 状态指示

LED	描述
POWER	电源工作指示
RUN	运行指示灯，正常运行时绿灯闪烁
ALARM	告警指示灯，设备启动未就绪或故障时灯亮，正常运行时常灭
ETH0_0	以太网 ETH1 的 LINK/ACT 指示
ETH0_1	以太网 ETH1 的速度指示，100M 时灯亮
ETH1_0	以太网 ETH2 的 LINK/ACT 指示
ETH2_1	以太网 ETH2 的速度指示，100M 时灯亮
S1-RX	接口 S1 接收指示，成功收到一帧数据时闪烁
S1-TX	接口 S1 发送指示，成功发送一帧数据时闪烁
S2-RX	接口 S2 接收指示，成功收到一帧数据时闪烁
S2-TX	接口 S2 发送指示，成功发送一帧数据时闪烁
S3-RX	接口 S3 接收指示，成功收到一帧数据时闪烁
S3-TX	接口 S3 发送指示，成功发送一帧数据时闪烁

2.3 以太网接口 ETH1、ETH2：X0

2.3.1 功能描述

ETH1、ETH2 为 2 路 10/100M 以太网接口，双路以太网接口有两种工作模式：

- 以太网交换模式：使能内置以太网交换功能；
- 双 IP 模式：每个以太网接口拥有独立的 IP 地址。

2.3.2 X0 管脚定义

X0 管脚	信号名称
1	ETH1_RX-
2	ETH1_RX+
3	ETH1_TX-
4	ETH1_TX+
5	ETH2_RX-
6	ETH2_RX+
7	ETH2_TX-
8	ETH2_TX+

2.4 同异步串口 S1：X1

2.4.1 功能描述

串口 S1 支持同步 HDLC/SDLC 协议和异步 UART 工作模式，编码格式支持 NRZ、NRZI、曼彻斯特、差分曼彻斯特、DBPL 等。

用户在订购时可以选择下列类型中的一种：

- RS-422 全双工带隔离
- RS-485 半双工带隔离

2.4.2 无时钟模式 X1 管脚定义

当 S1 工作在以下模式时，为无时钟工作模式：

- HDLC-NRZI
- HDLC-MAN（曼彻斯特）
- HDLC-DiffMAN（差分曼彻斯特）
- DBPL
- UART
- UART-HDLC

无时钟模式仅有数据信号，X1 管脚定义如下：

X1 管脚	RS-422 全双工	RS-485 半双工
1	5V_OUT	5V_OUT
2	GND	GND
3	TxD +	Data +
4	TxD -	Data -
5	RxD +	Term +
6	RxD -	Term -

注：5V_OUT 为+5VDC 输出，仅用于总线上拉，不能用于对外供电。不用可悬空。

2.4.3 带时钟模式 X1+X2 管脚定义

当 S1 工作在以下模时，为带时钟工作模式：

- HDLC-NRZ
- 同步 Bit 流

X1 和 X2 合一构成完整的带时钟工作工作信号，包含数据信号和时钟信号，其管脚定义如下：

管脚	RS-422 全双工	RS-485 半双工
X1-1	5V_OUT_1	5V_OUT_1
X1-2	GND_1	GND_1
X1-3	TxD +	Data +
X1-4	TxD -	Data -
X1-5	RxD +	Data_Term +
X1-6	RxD -	Data_Term -
X2-1	5V_OUT_2	5V_OUT_2
X2-2	GND_2	GND_2
X2-3	TxC +	Clock +
X2-4	TxC -	Clock -
X2-5	RxC +	Clock_Term +
X2-6	RxC -	Clock_Term -

注：

- 当 X1 和 X2 合一时，S2 串口编号失效；
- 需要用户短接 GND_1 和 GND_2。

2.4.4 RS-485 终端匹配

在 RS-485 模式下，如果需要做终端匹配，要把匹配电阻跨接在 Term+和 Term-管脚。

2.5 同异步串口 S2：X2

2.5.1 功能描述

串口 S2 支持同步 HDLC/SDLC 协议和异步 UART 工作模式，编码格式支持 NRZI、曼彻斯特、差分曼彻斯特、DBPL 等。

用户在订购时可以选择下列类型中的一种：

- RS-422 全双工带隔离
- RS-485 半双工带隔离

2.5.2 X2 管脚定义

X2 管脚定义如下：

X2 管脚	RS-422 全双工	RS-485 半双工
1	5V_OUT	5V_OUT
2	GND	GND
3	TxD +	Data +
4	TxD -	Data -
5	RxD +	Term +
6	RxD -	Term -

注：5V_OUT 为+5VDC 输出，仅用于总线上拉，不能用于对外供电。不用可悬空。

2.5.3 RS-485 终端匹配

在 RS-485 模式下，如果需要做终端匹配，把匹配电阻跨接在 Term+和 Term-管脚。

2.6 扩展接口 S3：X3

2.6.1 功能描述

S3 为扩展接口，用户可以在出厂时选配为串口或 CAN 总线接口。在配置为串口时，出厂时可以选择下列配置中的一种：

- RS-422：全双工带隔离
- RS-485：半双工带隔离

串口支持同步 HDLC/SDLC 协议和异步 UART 工作模式，编码格式支持 NRZI、曼彻斯特、差分曼彻斯特、DBPL 等。

2.6.2 X3 管脚定义

X3 管脚定义如下：

X3 管脚	CAN	RS-422 全双工	RS-485 半双工
1	5V_OUT	5V_OUT	5V_OUT
2	GND	GND	GND
3	CAN_H	TxD +	Data +
4	CAN_L	TxD -	Data -
5	Term +	RxD +	Term +
6	Term -	RxD -	Term -

注：5V_OUT 为+5VDC 输出，仅用于总线上拉，不能用于对外供电。不用可悬空。

2.6.3 CAN 总线终端匹配

在 CAN 模式下，如果需要做终端匹配，把匹配电阻跨接在 Term+和 Term-管脚。

2.6.4 RS-485 终端匹配

在 RS-485 模式下，如果需要做终端匹配，把匹配电阻跨接在 Term+和 Term-管脚。

2.7 X4：电源/复位/LED 指示接口

X4 管脚定义如下：

X4 管脚	信号名称	描述
1	Power_IN	电源输入，+4.9 ~ 17VDC
2	GND	地
3	Reset	低电平可复位本板
4	LED_POWER	电源工作指示灯，低电平有效
5	LED_RUN	系统告警指示灯，低电平有效
6	LED_ALARM	系统运行指示灯，低电平有效

2.8 X5：DMS-UART 配置管理接口

用于连接亚册 DMS-UART-8P 配置线，利用该配置线 yacer-DMS 可以通过计算机 USB 接口进行配置管理。

3 构建配置环境

3.1 获取配置管理软件 yacer-DMS

用户可通过以下方式获取配置管理软件的压缩包 yacer-DMS.zip：

- SDLC-BOARD 随机 U 盘的软件工具目录；
- 亚册官网 http://www.yacer.cn 的软件频道。

3.2 运行 yacer-DMS 软件

yacer-DMS 为免安装应用软件，对 yacer-DMS.zip 进行解压缩，进入工作目录双击 yacer-DMS.exe 即可运行。

3.3 软件主界面

下图即为配置管理软件的主界面，该界面可以分为三个部分：

- 工具条：功能操作按钮；
- 设备列表：显示在线设备基本信息及运行状态；
- 统计报告：显示接口的收发指示、设备详细信息、以及数据收发统计。

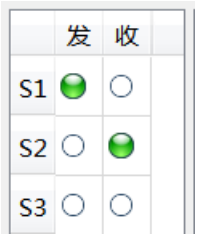
3.4 统计报告

统计报告由三个面板组成：控制面板、收发指示面板、信息显示面板。

3.4.1 控制面板

控制部件	功能
报告刷新周期： 1 秒	设置统计报告刷新周期
刷新	手动刷新统计报告
清零	统计报告清零

3.4.2 收发指示面板



- 发：对应串口每发一帧数据发指示灯闪烁一次。
- 收：对应串口每收一帧数据收指示灯闪烁一次。

3.4.3 信息显示面板

统计报告右侧为信息显示面板，可显示以下内容：

- 设备信息：运行时间、序列号、IP 地址、版本号；
- 串口：所有串口（含 CAN 口）的工作时钟频率、以及数据收发统计；
- UDP 发送：对每个使能的串口转 UDP 条目，显示相关的 UDP Client 发送包数；
- UDP 接收：对每个使能的 UDP 转串口条目，显示相关的 UDP Server 接收包数；
- DMS 服务：显示 SDLC-BOARD 与管理计算机之间的配置管理消息收发统计。

3.5 配置设备

点击工具条上的“设备配置”按钮，或双击设备列表的选中设备，yacer-DMS 弹出配置对话框。

对话框按照接口和功能，把配置项分为多个配置页面。

在对话框最下端包括以下操作按钮：

按钮	功能
导入	打开配置文件，读取配置参数刷新配置对话框
导出	将配置对话框中配置参数导出到文件中进行保存
获取默认配置	用设备出厂的默认配置刷新对话框内容
应用配置并重启设备	将对话框中的配置参数写入设备，并重启设备使配置生效
取消	取消当前配置操作

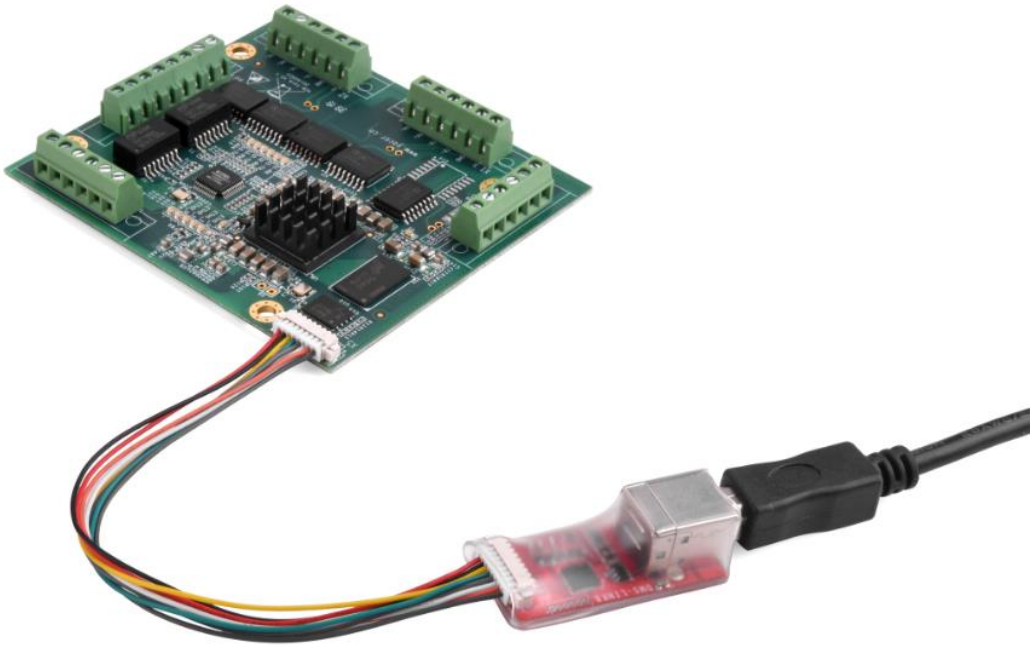
3.6 通过以太网口进行配置

通过网线连接管理计算机与 SDLC-BOARD 的任一网口，在计算机上运行 yacer-DMS 配置管理软件，即可对 SDLC-BOARD 进行运行状态监控和参数配置。

3.7 通过 DMS-UART 接口进行配置

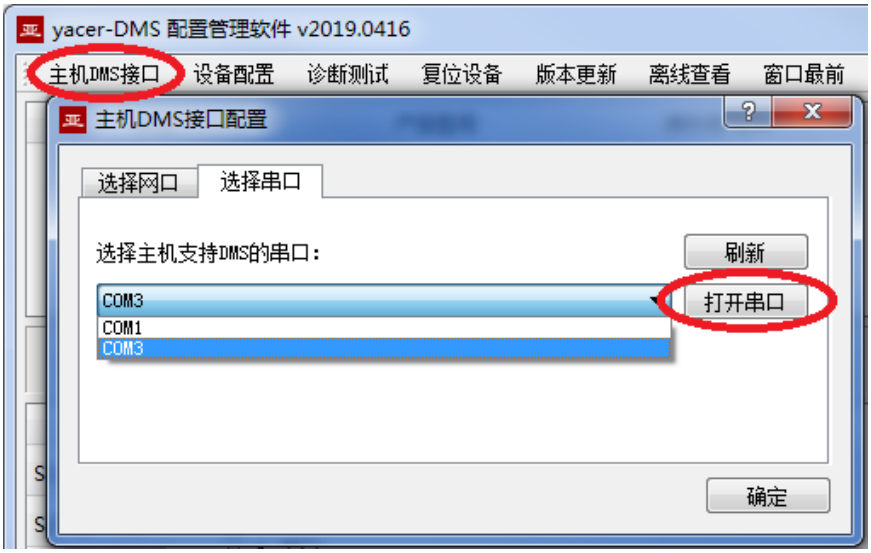
如果 SDLC-BOARD 的以太网接口被占用，用户可以通过 DMS-UART 接口（X5）进行配置。

DMS-UART-8P 配置线为亚册提供的专用配置工具，能够转换 DMS-UART 接口为 USB 接口。

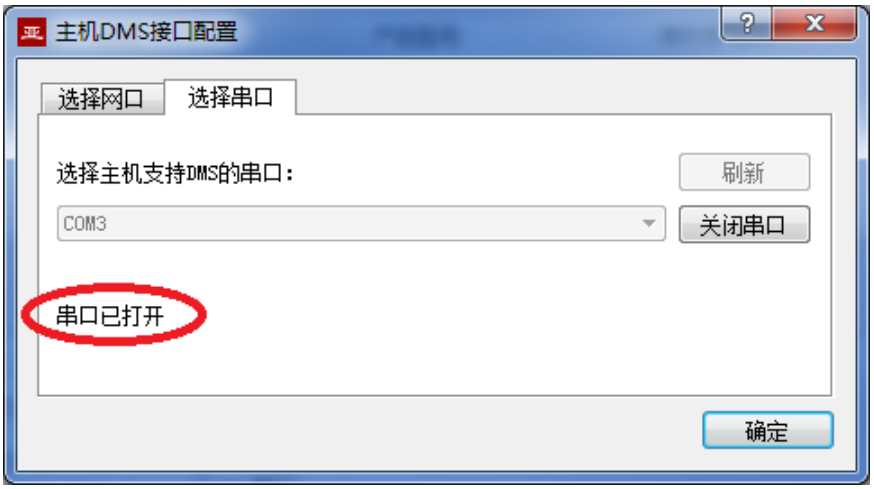


3.8 设置主机 DMS 接口

DMS-UART-8P 配置线接入管理计算机 USB 接口，计算机会增加一个 USB 仿真串口。



选择该仿真串口，点击“打开串口”，如果成功打开串口，则状态如下：



DMS-UART-8P 配置线的详细信息请参考文档《DMS-UART 接口与配置线 用户手册》。

4 功能及配置

4.1 以太网接口（ETH1、ETH2）

4.1.1 设备别名

允许用户为 SDLC-BOARD 设置别名，从而为设备增加描述或助记标识。

4.1.2 IP 地址配置

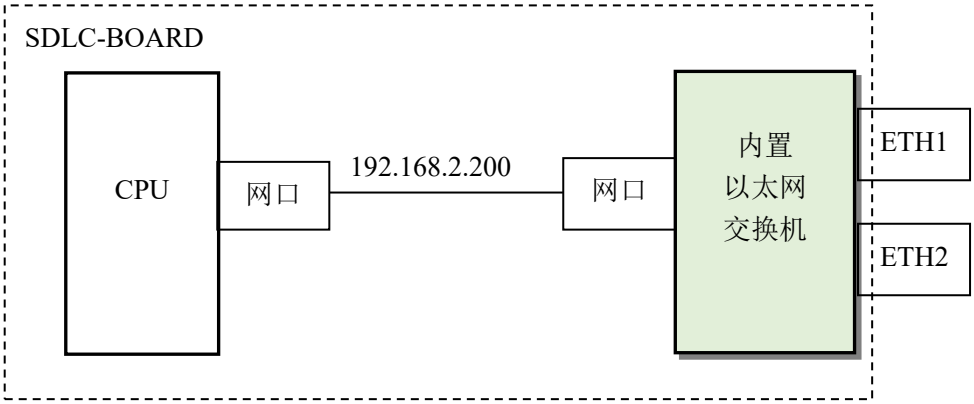
4.1.2.1 以太网交换使能

默认情况下，勾选 ☑ 以太网交换使能 复选框，使能内置以太网交换机，为 ETH1、ETH2 两个网口提供以太网交换功能。

☑ 以太网交换使能

	IP 地址	子网掩码	默认网关
ETH1	192.168.2.200	255.255.255.0	0.0.0.0

使能以太网交换功能后，SDLC-BOARD 对外只有拥有一个 IP 地址，网络功能示意如下：



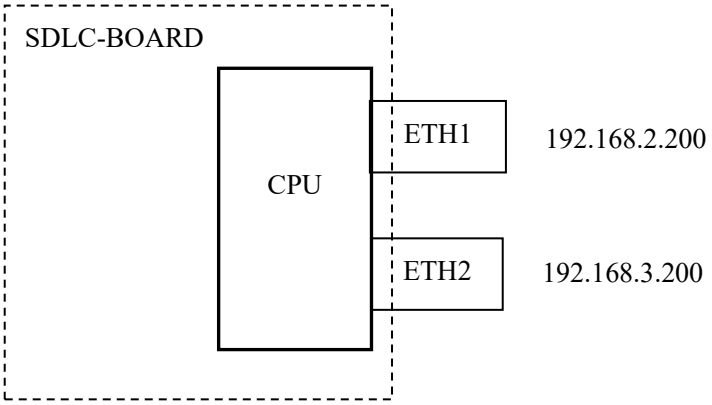
4.1.2.2 双 IP 配置

当取消勾选 ☐ 以太网交换使能 复选框时，ETH1、ETH2 具有独立的 IP 地址，配置时必须确保他们不在同一网段。

☐ 以太网交换使能

	IP 地址	子网掩码	默认网关
ETH1	192.168.2.200	255.255.255.0	0.0.0.0
ETH2	192.168.2.201	255.255.255.0	0.0.0.0

双 IP 的功能示意如下，SDLC-BOARD 相当于一台配有 2 块网卡的计算机。

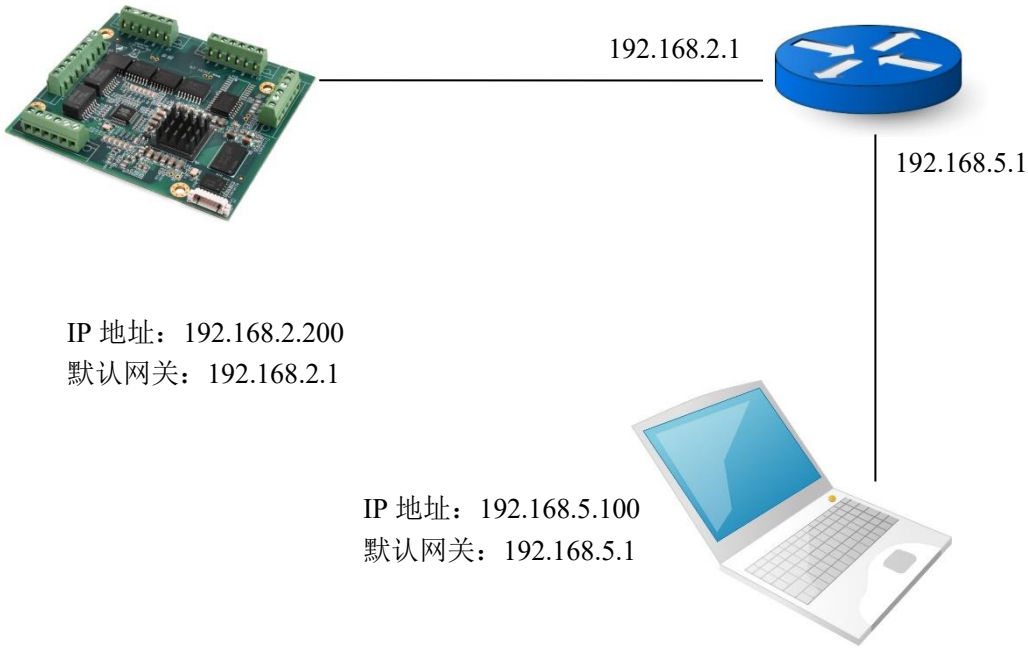


4.1.3 默认网关

缺省情况下，默认网关为 0.0.0.0，表示无网关配置。

如果 SDLC-BOARD 需要与其他网段的主机通信，必须借助于外部路由器。此时 SDLC-BOARD 的 IP 地址，必须和相连路由器端口的 IP 地址在同一网段，同时把该路由器 IP 地址设置为默认网关。

如下图所示，SDLC-BOARD 的 IP 地址为 192.168.2.200，远端计算机的 IP 地址为 192.168.5.100，由于不属于同一网段，二者必须借助于路由器才能够相互通信。SDLC-BOARD、计算机都需要把相连路由器端口的 IP 地址，设置为本设备的默认网关。



4.2 串口

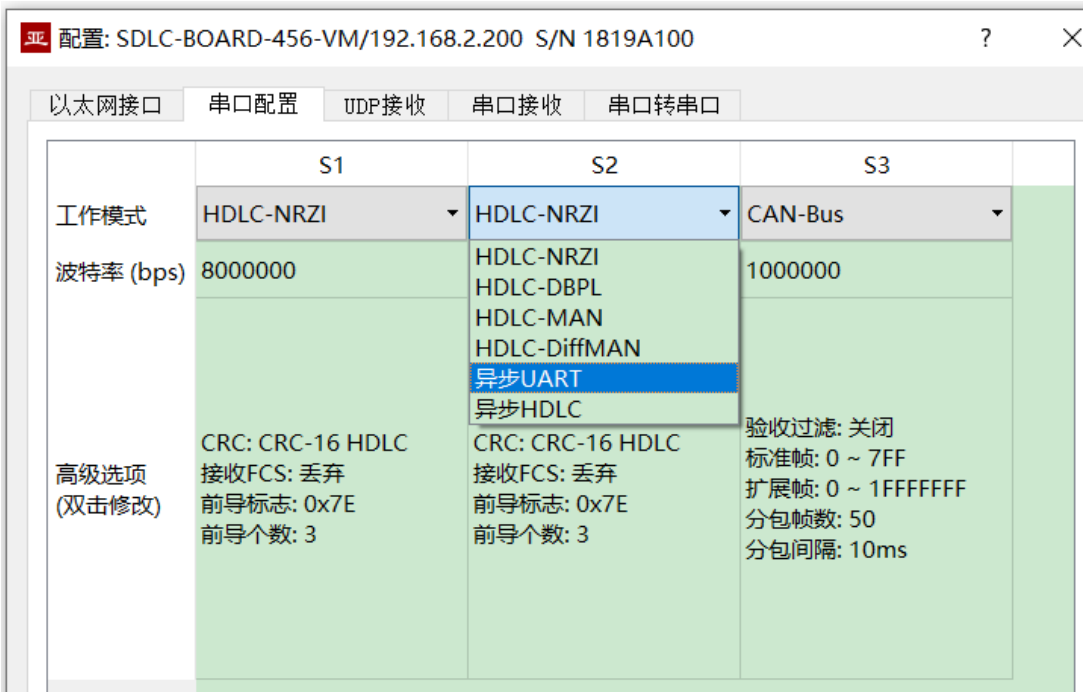
4.2.1 无时钟模式工作方式

S1、S2、S3 出厂时可选配置为 RS-422 全双工或 RS-485 半双工，支持下表所述的同步、异步工作模式。

工作模式		描述
同步	HDLC-NRZI	基于 NRZI 编码的同步 HDLC 协议
	HDLC-DBPL	基于 Differential Bi-Phase Level 编码的同步 HDLC 协议
	HDLC-MAN	基于曼彻斯特编码（Manchester）的同步 HDLC 协议
	HDLC-DiffMAN	基于差分曼彻斯特编码的同步 HDLC 协议
异步	异步 UART	通用异步串口工作模式，类似于通用计算机上的串口
	异步 HDLC	基于 UART 的类 HDLC 通信协议

用户可以通过“工作模式”下拉列表选择所需工作模式。由于每种工作模式的参数配置不同，确定工作模式后，“高级选项”单元格的内容自动根据工作模式进行调整。

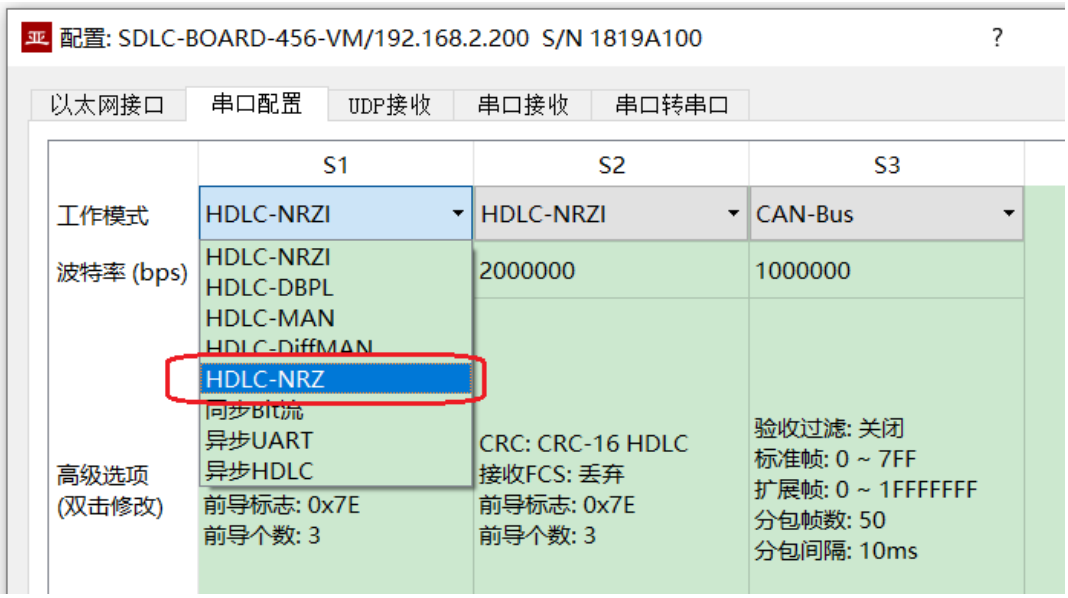
如果需要进一步配置所选工作模式的更多工作参数，鼠标双击“高级选项”所在单元格，即可弹出参数配置对话框。



4.2.2 带时钟模式工作模式

X1 与 X2 合一可以组成一个带时钟的同步串口，能够额外支持以下工作模式：

工作模式		描述
同步	HDLC-NRZ	基于 NRZ 编码的同步 HDLC 协议
	同步 Bit 流	基于接收时钟采样或发送串行 Bit 数据

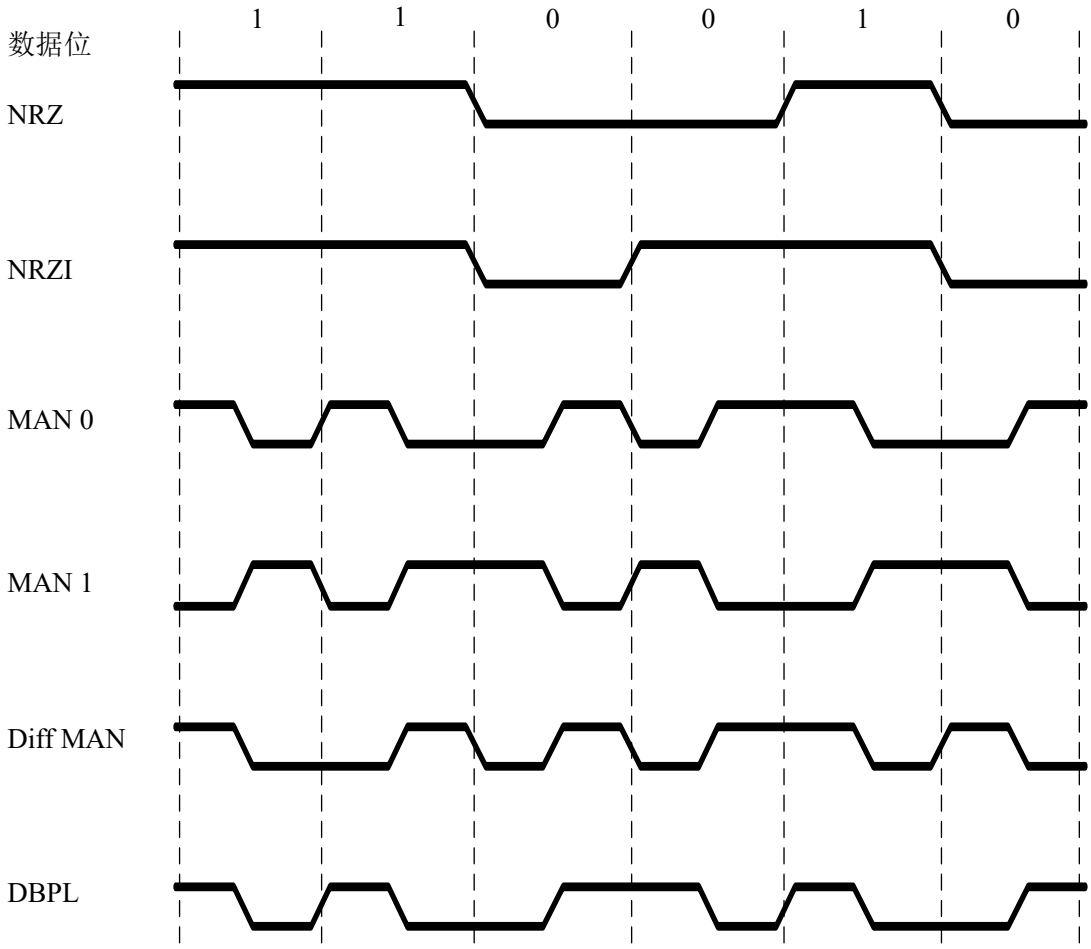


4.2.3 波特率

“波特率”配置了串口的通信速率，对于 HDLC-NRZI、HDLC-DBPL、HDLC-MAN、HDLC-DiffMAN 等同步工作模式，以及所有的异步工作模式来说，通信双方的波特率必须相同才能够保证数据的正确传输。

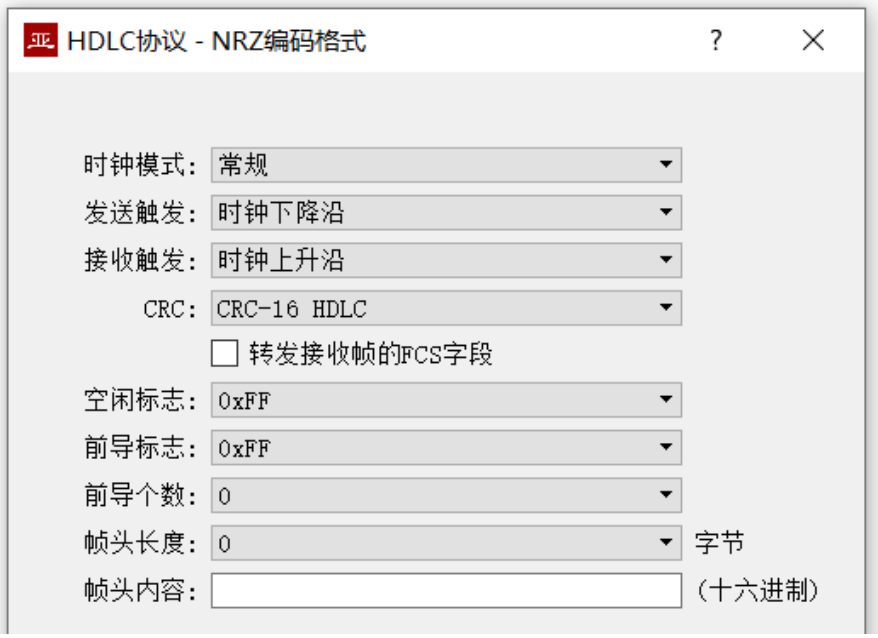
4.2.4 同步串口编码格式

HDLC-NRZ、HDLC-NRZI、HDLC-DBPL、HDLC-MAN、HDLC-DiffMAN 等同步工作方式，链路层采用 HDLC 协议，编码格式区别如下图所示：

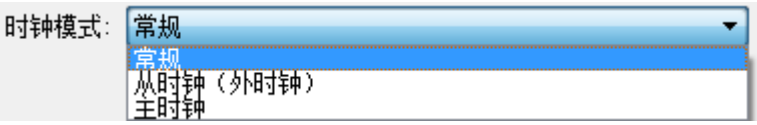


4.2.5 HDLC-NRZ 参数配置

同步 HDLC-NRZ 是常用的同步工作模式，本编码格式依靠收发时钟信号实现数据位同步，因此时钟参数的配置尤为重要。



4.2.5.1 时钟模式

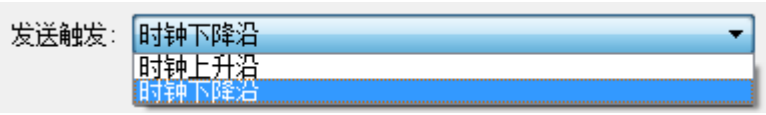


同步串口的时钟模式包括三种，常规、从时钟、主时钟。

时钟模式	发送时钟	接收时钟
常规	本设备产生，通过管脚 TxC 输出	对端设备产生，通过管脚 RxC 输入
从时钟（外时钟）	对端设备产生，取自管脚 RxC TxC 输出自动与 RxC 同步	对端设备产生，通过管脚 RxC 输入
主时钟	本设备产生，通过管脚 TxC 输出	本设备产生，忽略管脚 RxC 输入

从时钟模式也称为外时钟工作模式。当对端为传输设备时，SDLC-BOARD 常配置为从时钟模式，用传输设备提供的时钟发送数据，确保数据在全网的传输基于一个时钟，避免了时钟不同源造成的丢包隐患。

4.2.5.2 发送触发



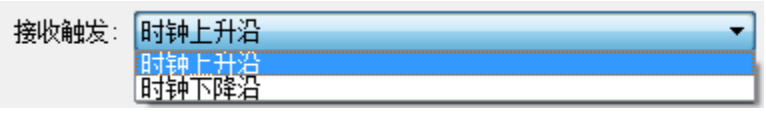
发送触发定义了新数据 bit 的产生时刻：

- 时钟下降沿：在时钟的下降沿，产生新的数据位
- 时钟上升沿：在时钟的上升沿，产生新的数据位

遵循 HDLC 协议规范的通信，应该选择时钟下降沿触发新数据发送。也有一些特殊的应用场合，用户采用非标准的通信方式，采用上升沿触发新数据发送。

4.2.5.3 接收触发

接收触发定义了串口接收数据的采样时刻：



- 时钟上升沿：在 RxC 信号的上升沿，读取 RxD 线上的数据
- 时钟下降沿：在 RxC 信号的下降沿，读取 RxD 线上的数据

遵循 HDLC 协议规范的通信，因为采用下降沿触发新数据，考虑到新数据的稳定时间，为了保证数据的正确读取，接收触发必须配置为时钟上升沿。

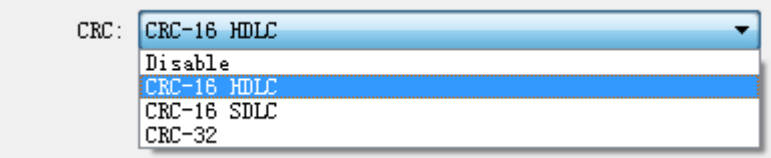
接收触发配置要根据对端设备的发送触发确定：

对端发送触发	本端接收触发
时钟下降沿	时钟上升沿
时钟上升沿	时钟下降沿

4.2.5.4 CRC

为了校验数据通信的正确性，应该使能 CRC 功能。

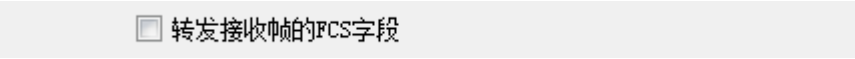
默认情况下，HDLC 通信应该选择 CRC-16-HDLC 校验方式。



CRC	描述
Disable	CRC 不使能： ● 数据发送无 CRC 计算，HDLC 帧无 FCS 字段 ● 数据接收不进行 CRC 校验
CRC-16 HDLC	采用 16 位 ISO HDLC CRC 校验方式
CRC-16 SDLC	采用 16 位 IBM SDLC CRC 校验方式
CRC-32	采用 32 位 ISO HDLC CRC 校验方式

4.2.5.5 转发接收帧的 FCS 字段

该配置仅在 CRC 使能时生效。



HDLC 帧结构如下表所示，其中 FCS 为帧校验序列字段。

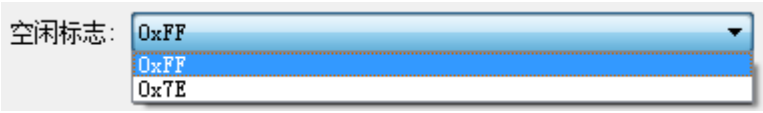
开始标志 Opening Flag	地址字段 Address Field	控制字段 Control Field	信息字段 Information Field	帧校验序列 FCS Field	结束标志 Closing Flag
0x7E	1 字节	1 字节	可变长	CRC 2/4 字节	0x7E
0x7E	用户数据			CRC 2/4 字节	0x7E

如果勾选该复选框，则转发用户数据+FCS 字段。

如果不勾选该复选框，SDLC-BOARD 在接收 HDLC 帧 CRC 校验通过后，抛弃数据末尾 2/4 个字节的 FCS 字段，仅转发用户数据。

4.2.5.6 空闲标志

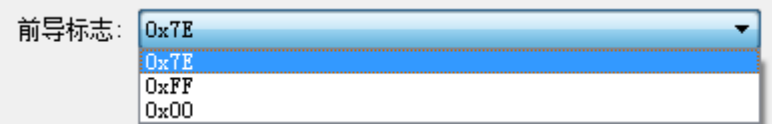
定义 HDLC 帧间填充内容，默认应该选择 0xFF。

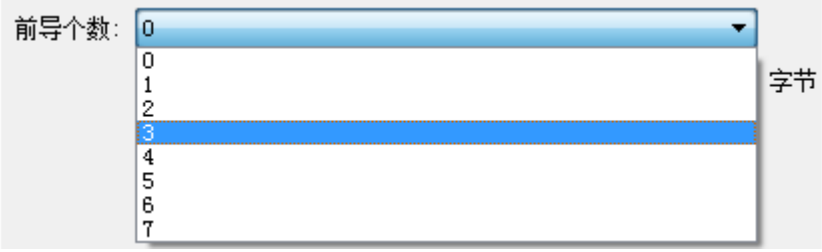


4.2.5.7 前导标志与前导个数

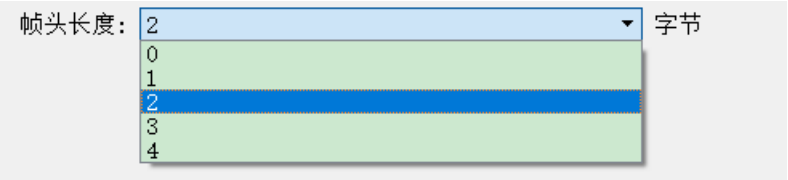
在半双工通信中，往往需要在帧前增加前导标志用于接收方同步，最常用的是增加 2~5 个 0x7E 前导。

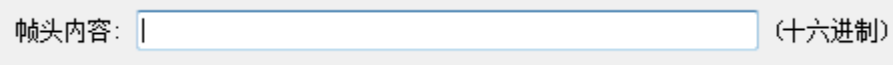
对于全双工应用，往往不需要前导，把前导个数设置为 0 即可（无前导）。





4.2.5.8 帧头长度与帧头内容





以上图为例，帧头长度定义为 2，帧头内容定义为十六进制的 FF 03：

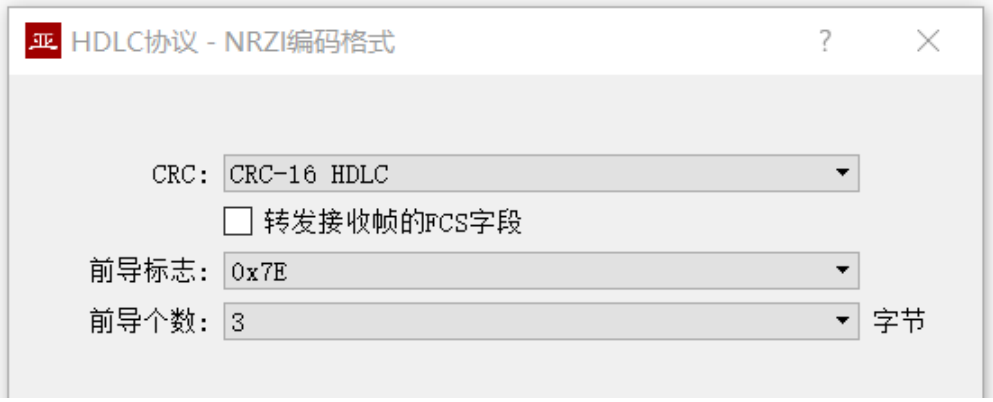
- 在 HDLC 发送时，在用户数据前增加 FF 03，同用户数据一起组成 HDLC 帧数据。
- 在 HDLC 接收时，SDLC-BOARD 把 HDLC 帧数据的前 2 个字节作为帧头抛弃，仅转发后续数据给用户。

开始标志 Opening Flag	帧头	用户数据	帧校验序列 FCS Field	结束标志 Closing Flag
0x7E	0xFF 0x03	可变长	CRC 2/4 字节	0x7E

4.2.6 HDLC-NRZI 参数配置

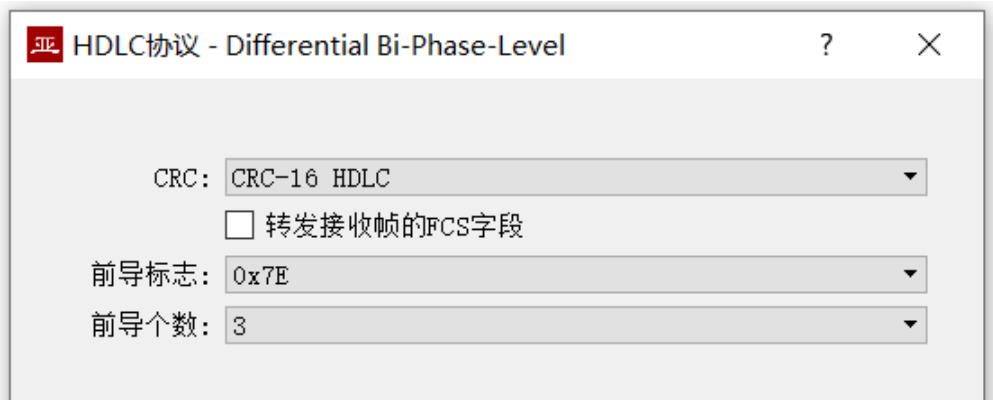
与 NRZ 编码格式不同，NRZI 编码格式的数据中包含时钟信息，只需要通信双方的波特率相同，不再需要时钟模式、发送触发、接收触发等参数。

HDLC-NRZI 的配置参数见下图，参数作用及配置方式同 HDLC-NRZ 工作模式。



4.2.7 HDLC-DBPL 参数配置

HDLC-DBPL 采用 Differential Bi-Phase Level 编码格式，其高级选项对话框如下图所示：

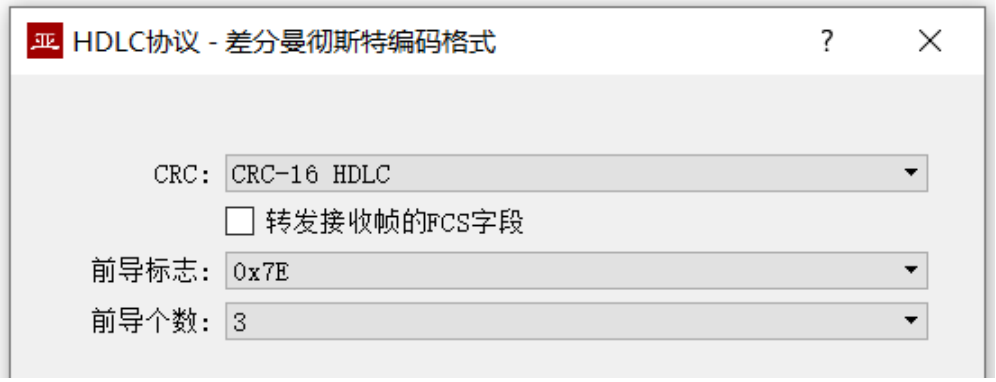


HDLC-DBPL 的参数含义同 HDLC-NRZI 相同。

需要注意的是，很多声称 DBPL 的编码实现其实与差分曼彻斯特一致，因此用户需要仔细参考 5.2.4 章节中编码格式的定义，选择正确的工作模式。

4.2.8 HDLC-DiffMAN（差分曼彻斯特）参数配置

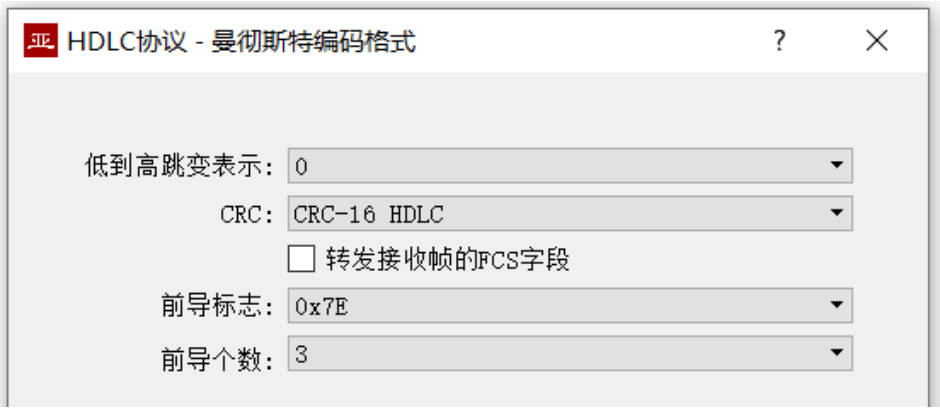
差分曼彻斯特工作模式高级选项对话框如下图所示：



HDLC-DiffMAN 的参数含义同 HDLC-NRZI 相同。

4.2.9 HDLC-MAN（曼彻斯特）参数配置

HDLC-MAN 工作模式高级选项对话框如下图所示：

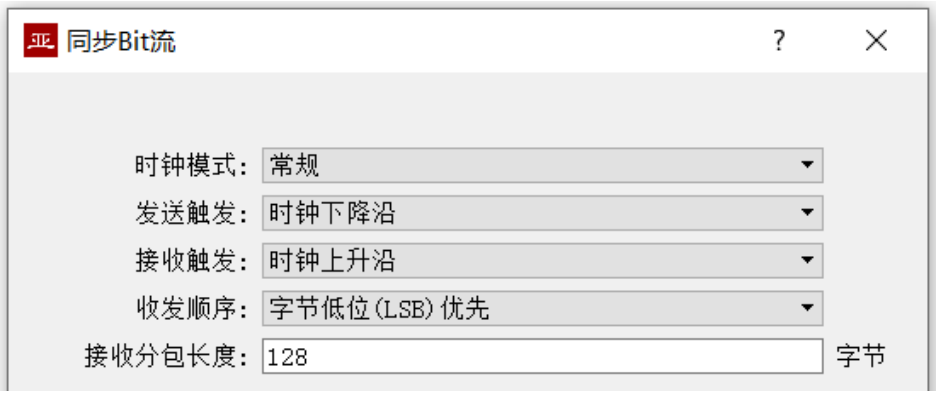


曼彻斯特编码格式除了与 NRZI 相同的配置参数外，还有数据线波形低到高的跳变含义的参数：

- 0：低到高跳变表示逻辑 0；
- 1：低到高跳变表示逻辑 1。

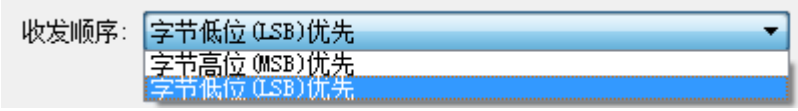
4.2.10 同步 Bit 流参数

每个时钟周期的上升沿或下降沿，在数据线上采样 1bit 数据，每 8bit 组成一个字节，收满一个分包长度的字节后，组成一个 UDP 报文发往目的 IP。



时钟模式、发送触发、接收触发、空闲标志等配置可参考 HDLC-NRZ 参数配置。

线上的 bit 流，在计算机或系统的内存中以字节的形式存放。收发顺序确定了字节和 bit 转换的方式。

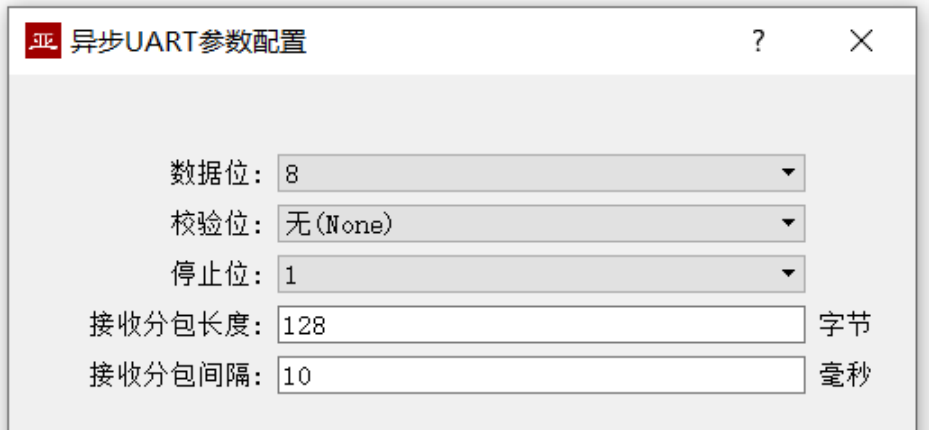


收发顺序	发送操作	接收操作
MSB 优先	字节高位先发	先收到的数据置于字节高位
LSB 优先	字节低位先发	先收到的数据置于字节低位

4.2.11 异步 UART 参数配置

UART 是一种字符流的通信方式，数据位、校验位、停止位定义了异步串口的基本工作参数，该参数必须和对端设备配置完全相同。

一般来说数据位定义为 8 位，即 1 个字节，这样 UART 相当于字节流的通信。



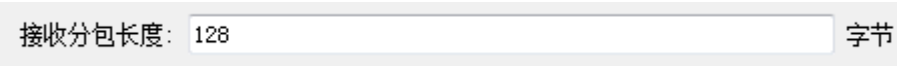
当把 UART 的字节流转换为 UDP 报文或 HDLC 帧时，如果每个字节都转换为一个 UDP 报文进行传输，开销太大、效率太低。

为了提高效率，SDLC-BOARD 将收到的字节流进行缓冲，然后把缓冲下来的若干字节组成一个 UDP 报文发出，这个过程成为分包。

分包由两个参数进行控制，称为分包长度、分包间隔。

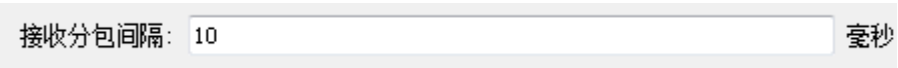
4.2.11.1 接收分包长度

举例来说，如果分包长度设置为 128 字节，那么当 UART 收满 128 字节后，组成一个数据包进行转发。



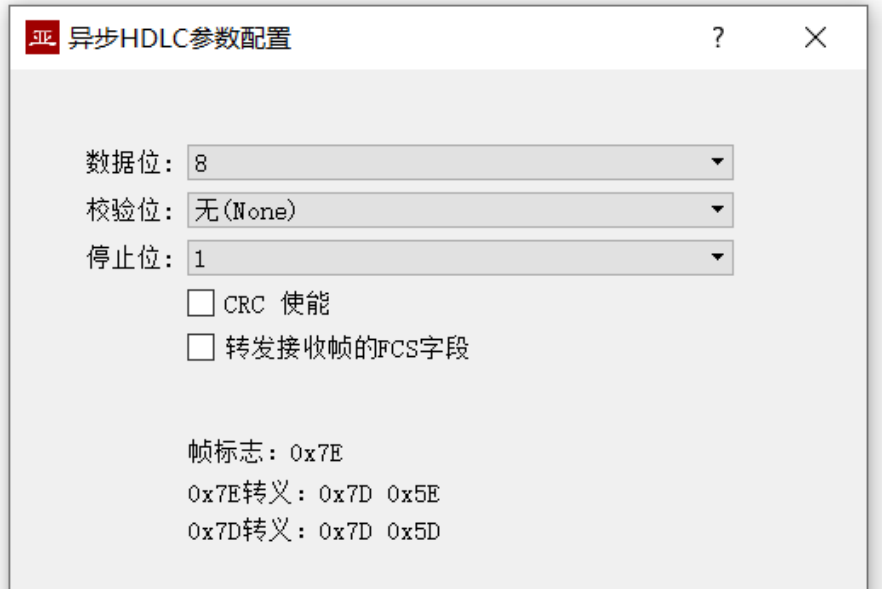
4.2.11.2 接收分包间隔

如上例设置了分包间隔为 10ms，如果 UART 超过 10ms 未收到新的数据，则不论是否收满 128 字节，都会把缓冲区的数据组成一个数据包进行转发。



4.2.12 异步 HDLC 参数配置

异步 HDLC 工作模式是亚册自定义的一种通信方式，在普通 UART 通信之上对字节流进行封装，构成异步 HDLC 帧。这样异步串口就可以和 UDP 报文、同步 HDLC 帧进行基于数据包的通信。



异步 HDLC 帧格式就是把数据包前后增加 0x7E 作为起止标记，帧结构如下：

开始标志	信息字段	帧校验字段	结束标志
0x7E	2 – 1470 字节数据	2 字节 CRC 校验	0x7E

由于信息字段、帧校验字段有可能出现 0x7E，因此在传输前对信息字段、帧校验字段进行字符转义，转义规则如下：

- 0x7E：转义为 0x7D 0x5E 两个字符
- 0x7D：转义为 0x7D 0x5D 两个字符
- 其他字符：不转义

数据发送转义操作如下：

原始数据	实际发送数据
0x7E	0x7D 0x5E
0x7D	0x7D 0x5D
其他	无变化

数据接收转义操作如下：

实际接收数据	数据
0x7D 0x5E	0x7E
0x7D 0x5D	0x7D
其他	无变化

4.2.13 CAN 口配置

S3 在出厂时可以选配为 CAN 总线接口，配置如下：

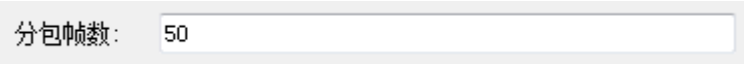
由于 CAN 帧很短，如果每个 CAN 帧都转换为一个 UDP 报文进行传输，开销太大、效率太低。

为了提高效率，SDLC-BOARD 将收到的 CAN 帧进行缓冲，然后把缓冲下来的多个 CAN 帧组成一个数据包进行转发，这个过程成为分包。

分包由两个参数进行控制，称为分包帧数、分包间隔。

4.2.13.1 接收分包帧数

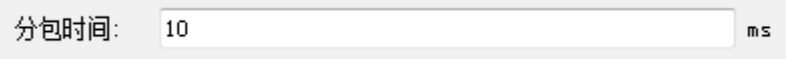
分包帧数最大为 50。



如上图分包帧数为 50，当 SDLC-BOARD 收满 50 个 CAN 帧后，将其组装为一个数据包进行转发。

4.2.13.2 接收分包间隔

如上例设置了分包间隔为 10ms，如果超过 10ms 未收到新的 CAN 帧，则不论是否收满 50 帧，都会把缓冲区的 CAN 帧组成一个数据包进行转发。



4.2.13.3 验收过滤

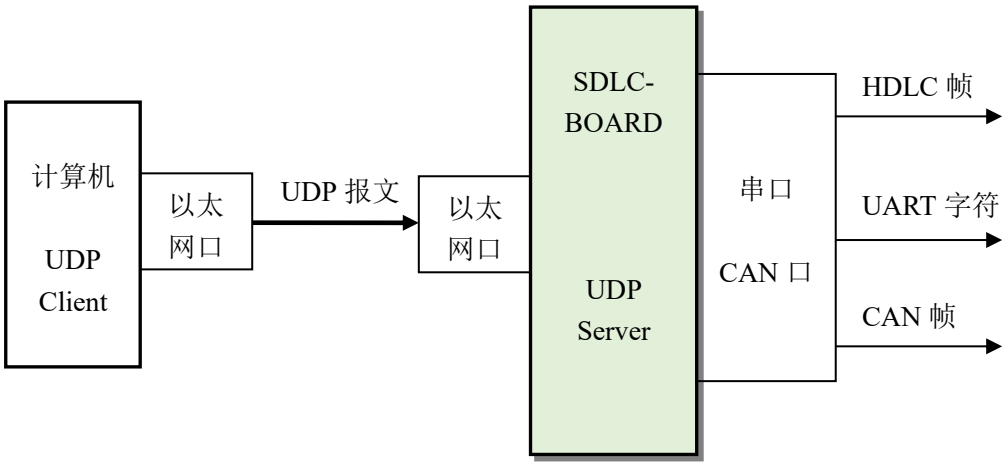
验收过滤允许用户设置允许接收的帧 ID 范围，范围以外的帧自动抛弃。

4.3 UDP 转串口配置

4.3.1 功能描述

借助 SDLC-BOARD，计算机或服务器能够实现同步 HDLC、UART 或 CAN 数据发送。

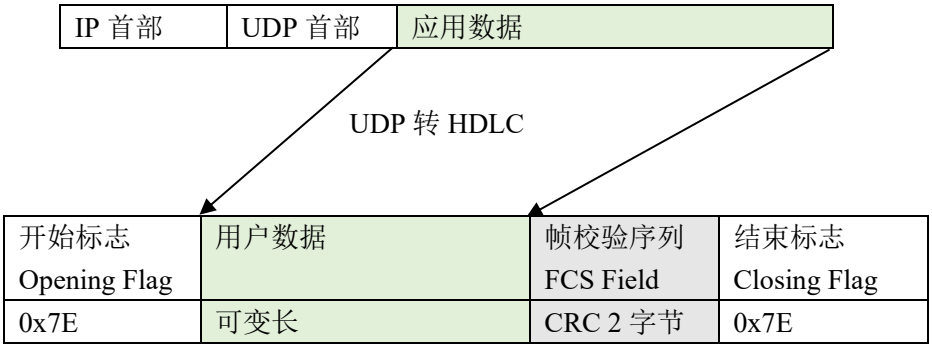
典型的应用如下图所示。计算机作为 UDP Client 通过以太网接口发送 UDP 报文，SDLC-BOARD 对 UDP 报文进行处理，根据配置转换为 HDLC 帧、或 UART 数据、或 CAN 帧，从串口或 CAN 口发送出去。



4.3.2 协议转换

最典型的 UDP 转 HDLC 应用如下图所示，SDLC-BOARD 把 UDP 的应用数据装入 HDLC 帧的用户数据区，然后计算 CRC 并填充 FCS 字段，形成一个完整的 HDLC 帧进行发送。

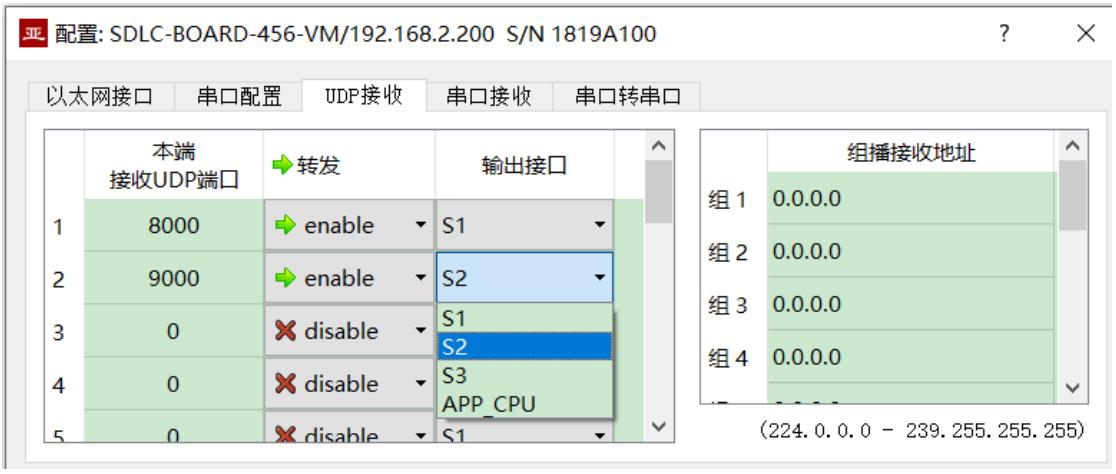
为了减轻计算机的计算负荷、降低用户编程的复杂性，一般情况下，UDP 报文中不包含 HDLC 的 FCS 字段，该字段由 SDLC-BOARD 计算填充。



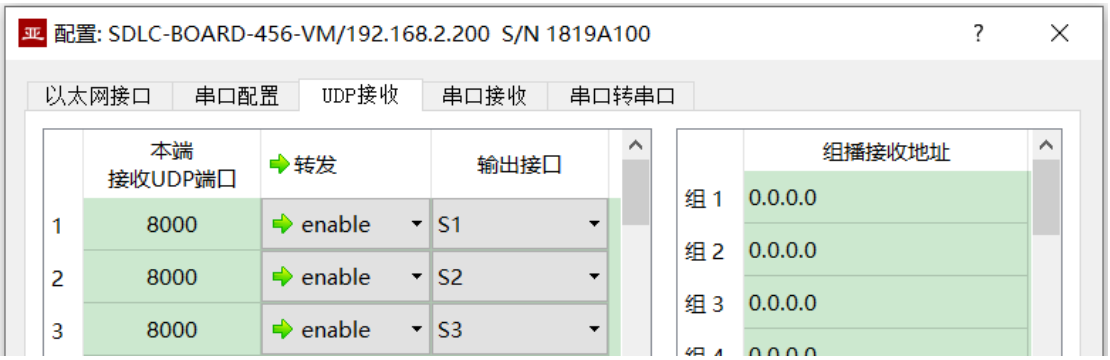
4.3.3 转发配置

对 UDP 转串口进行设置，每一行表示一个 UDP 端口到串口的转发条目，选择“enable”生效。可以实现三种转发策略：

- 转发：指定的 UDP 端口接收的数据可以转发到指定的串口；
- 分路：从同一个 UDP 端口接收的数据可以同时转发到多个串口；
- 汇聚：多个不同的 UDP 端口接收的数据可以转发到同一个串口。



下图的配置实现了从 1 个 UDP 端口收到的数据，同时分发到 3 个串口输出应用：



4.3.4 收 UDP 组播

如果需要接收组播 UDP 报文，在右侧“组播接收地址”列表添加需要加入的组播地址。

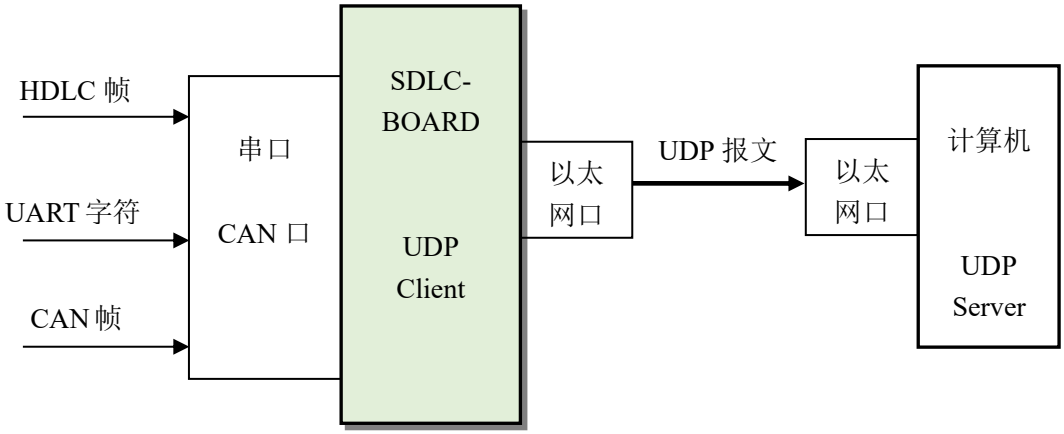
组播地址的范围为 224.0.0.0～239.255.255.255，其中 224.8.8.8 为 SDLC-BOARD 的配置管理地址，用户不能使用该地址。

组播地址配置为 0.0.0.0 表示该条目未生效。

4.4 串口转 UDP 配置

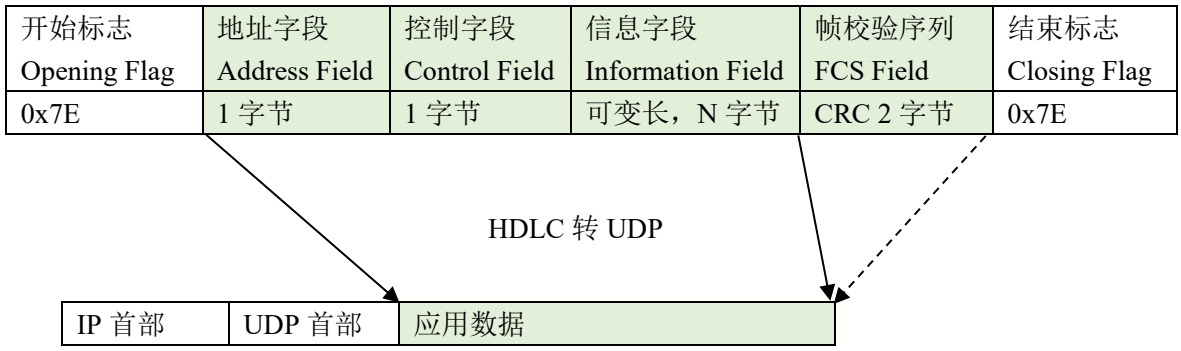
4.4.1 功能描述

SDLC-BOARD 通过同步串口接收 HDLC 帧、通过异步串口接收 UART 字符串并将其分包、通过 CAN 口接收 CAN 帧并组包，将上述数据帧或数据包转换为 UDP 报文，然后根据配置将报文通过以太网口发送给计算机或服务器。



4.4.2 协议转换

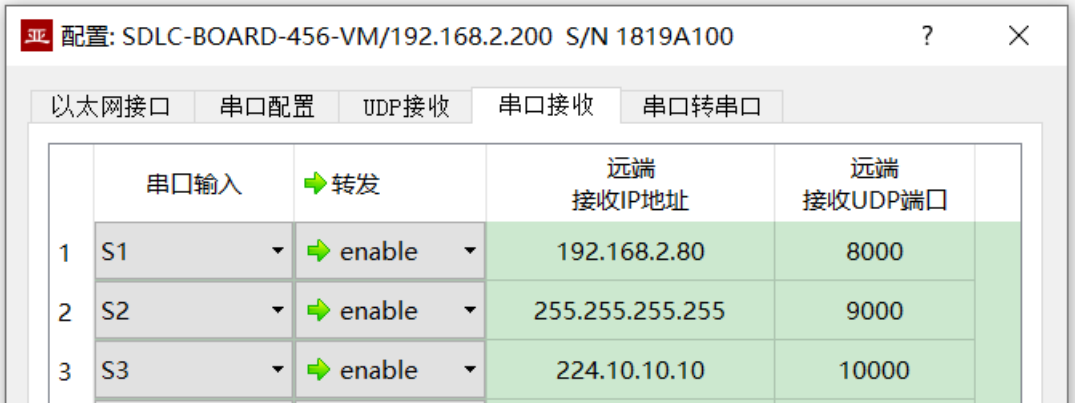
为了确保用户数据的完整性，SDLC-BOARD 把完整的 HDLC 帧置于 UDP 应用数据中，转发给 UDP Server。



4.4.3 转发配置

对串口转 UDP 进行设置，每一行表示一个串口到目标主机 UDP 端口的转发条目，选择“enable”生效。可以实现三种转发策略：

- 从特定串口接收的数据可以转发到特定的目的 IP + UDP 端口；
- 汇聚：多个不同的串口接收的数据可以转发到同一个目的 IP + UDP 端口；
- 分路：从同一个串口接收的数据可以转发到不同的目的 IP 或 UDP 端口。



如上图所示，配置了三个串口转 UDP 条目，分别实现：

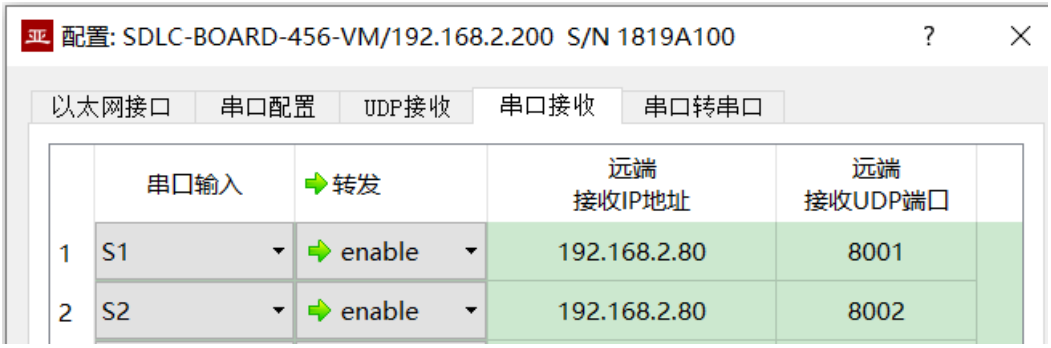
- 串口 S1 转 UDP 单播，目的 IP 地址为 192.168.2.80，目的 UDP 端口为 8000；
- 串口 S2 转 UDP 广播，网络中所有主机均能在 9000 端口收到来自 S2 的数据；
- 串口 S3 转 UDP 组播，网络中只有加入了 224.10.10.10 组的计算机，才能够收到来自 S3 的数据。

4.4.4 UDP Server 如何识别源串口

在汇聚应用方式中，需要把源自多个不同串口的 HDLC 帧，转发给一台服务器或计算机，进行统一处理。在这种情况下，需要有一种策略，让计算机能够知道所收的 UDP 报文，其数据源自哪个串口。

4.4.4.1 根据目的 UDP 端口识别源串口

如下图所示，为每个串口设置不同的转发目的 UDP 端口。作为 UDP Server 计算机在不同的 UDP 端口接收数据：8001 端口所收报文来自串口 S1，8002 端口所收报文来自串口 S2。

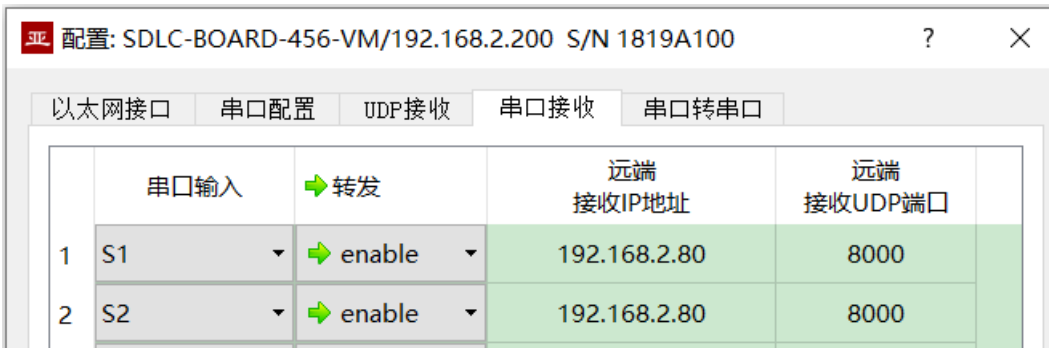


4.4.4.2 根据源 UDP 端口识别源串口

当采用目的 UDP 端口识别源串口方案的时候，UDP Server 需要在多个 UDP 端口上监听和接收数据。当串口数量很多的时候，不仅 UDP Server 的端口资源占用过多，配置及编程的复杂度也提高不少。

为了简化 UDP Server 侧的实现，我们可以采用下例的配置，把每个转换都转发至 UDP Server 的同一端口。**亚册 SDLC-BOARD 在转发时，会自动根据源串口调整 UDP 报文的源端口号，其中串口 S1 转发的 UDP 报文源端口为 8001、串口 S2 的为 8002，以下逐渐递增。**

这样，UDP Server 仅需在一个端口（下例为 8000）监听和接收数据，然后根据源 UDP 端口区分源串口。如果存在多片 SDLC-BOARD，UDP Server 可以通过源 IP 区分源设备。

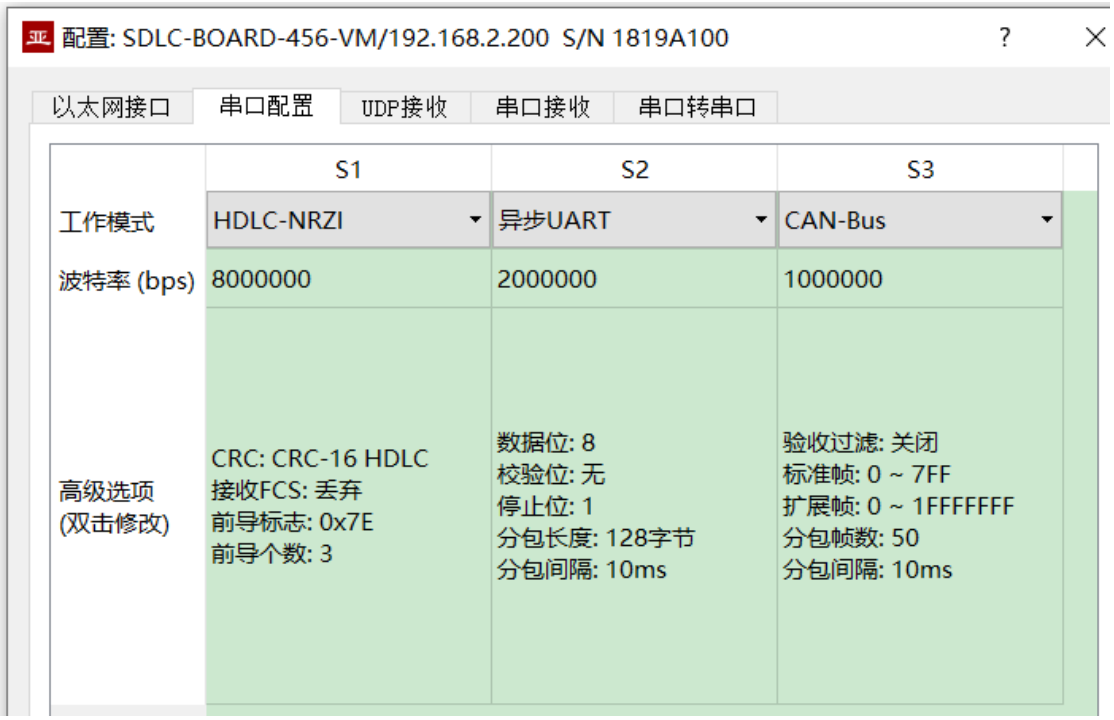


4.5 串口转串口配置

串口转串口能够把指定串口的输入数据，转发到其他串口输出，主要用于：

- 同步串口、异步串口之间的转换；
- 串口、CAN 总线之间的转换。

如下面所示配置，串口 S1 工作在同步 HDLC 模式，S2 工作在异步 UART 模式。其串口转串口配置为 S1、S2 之间的互相转发，则可以实现同、异步串口之间的数据转换。



5 系统维护

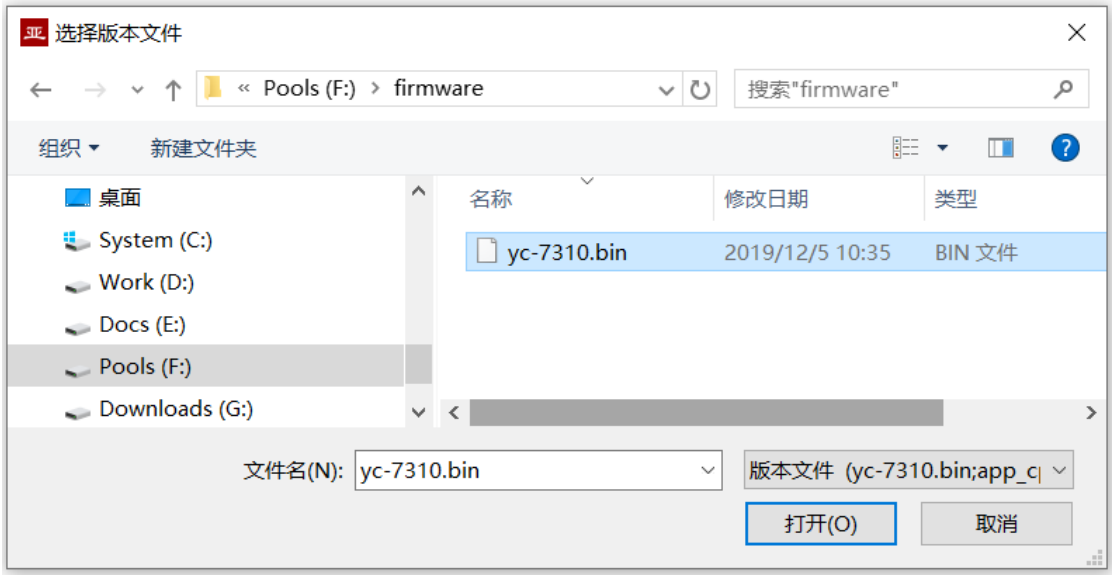
5.1 固件版本更新

5.1.1 开始更新

点击工具条上的 版本更新 按钮，弹出版本更新对话框，点击 开始更新 按钮。

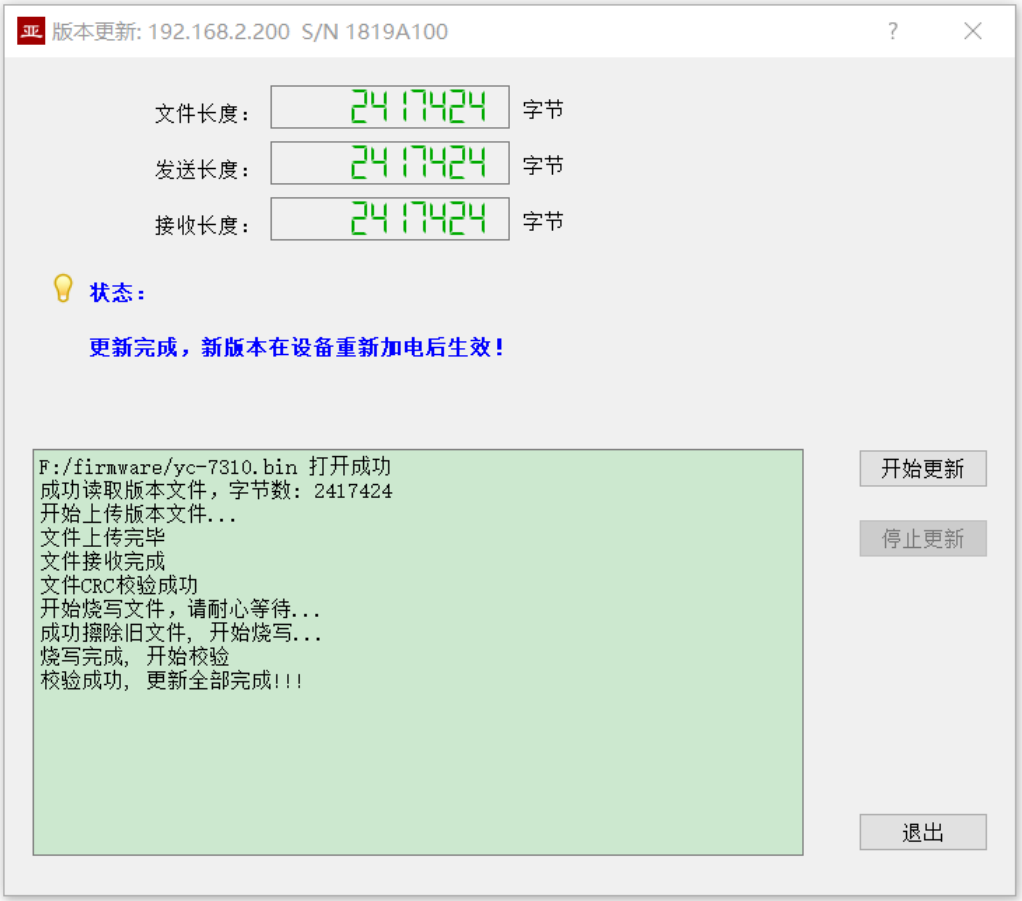
5.1.2 找到固件版本

弹出“选择版本文件”对话框，找到存放最新固件版本所在的文件夹，选中并点击“打开”，即开始更新。



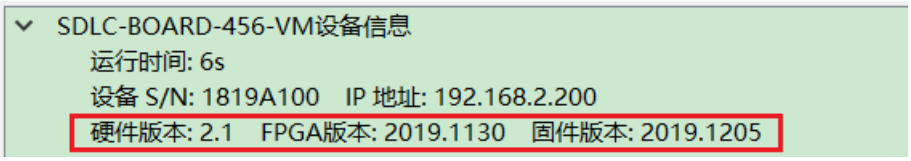
5.1.3 更新完成

更新完成以后页面状态显示“版本更新完成”即表示版本更新完成。



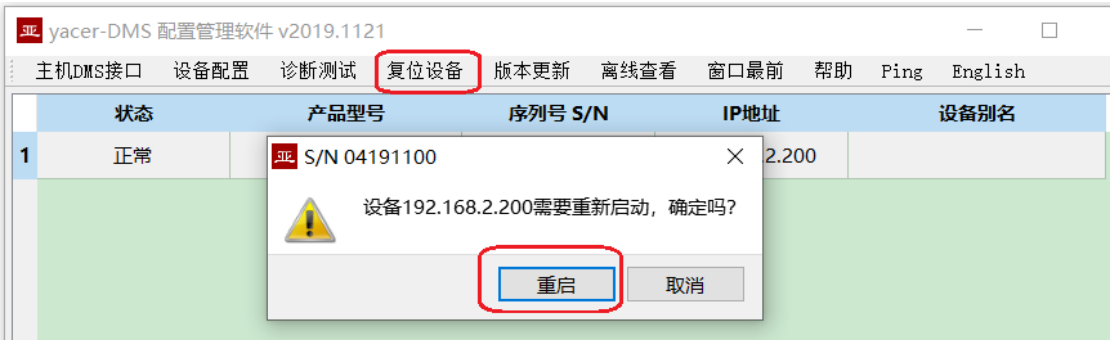
5.1.4 更新确认

更新完成后，将设备重新加电，观察统计报告中的版本信息，通过版本日期确定新版本是否成功更新。



5.2 复位设备

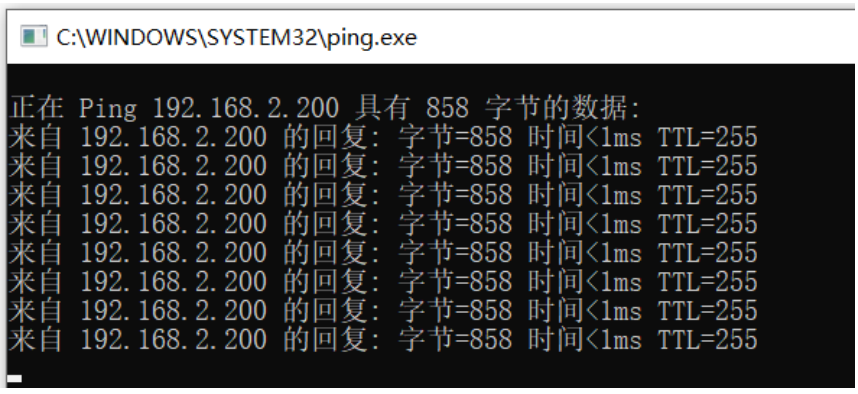
点击工具条上的 复位设备 按钮，弹出设备重启对话框，点击 重启 按钮，设备将进行重新启动。



5.3 Ping

点击工具条上的 Ping 按钮，DMS 自动对所选设备启动 ping 命令，以检查配置管理计算机和 SDLC-BOARD 之间的网络连接是否正常。

执行 Ping 命令前，请首先确保计算机和 SDLC-BOARD 的 IP 地址在同一网段。



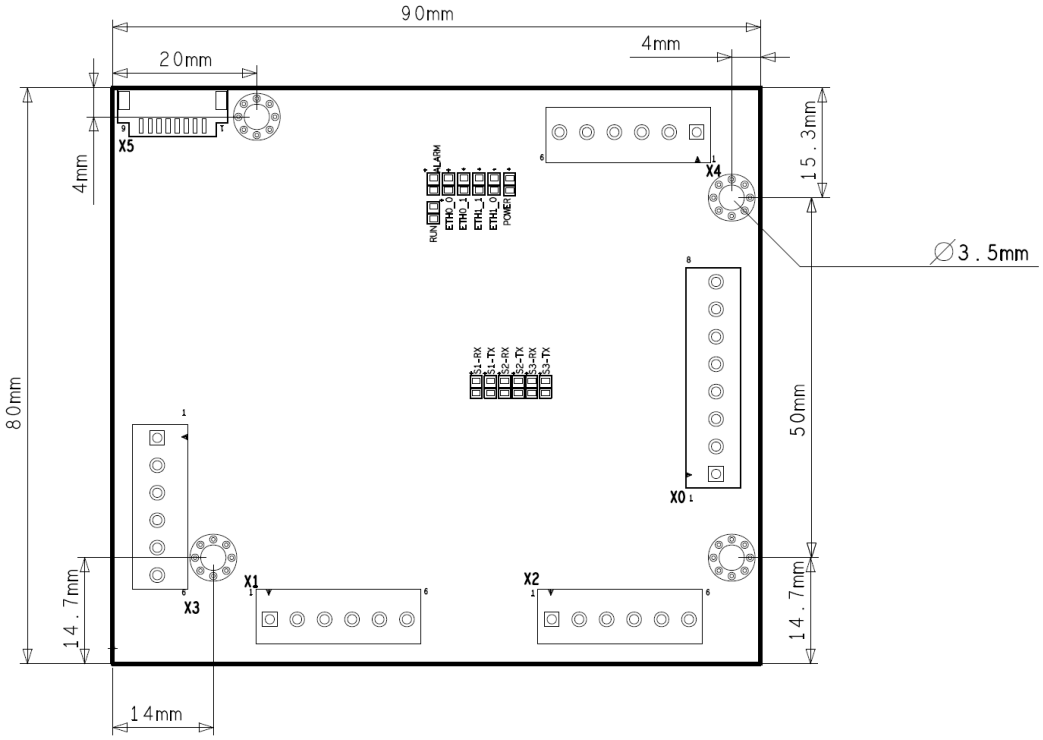
6 机械特性与安装

6.1 安装

采用 4 个 M3 螺钉固定，安装孔径 ＝3.5mm。

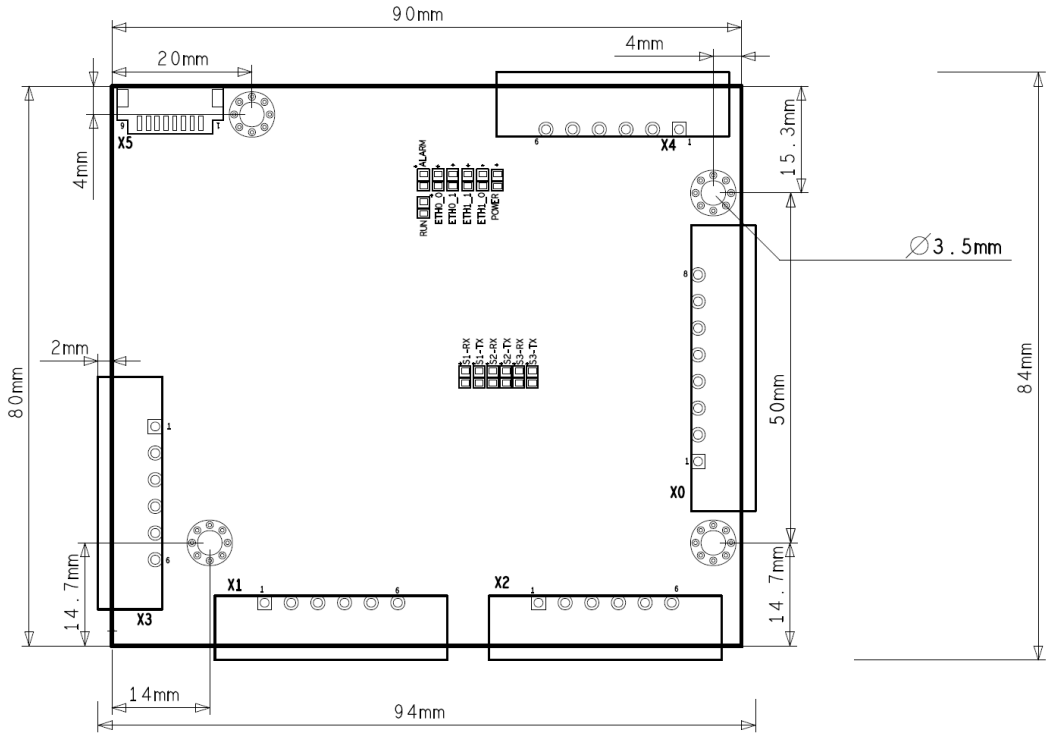
6.2 LG 单板尺寸

高度占用 130mm。



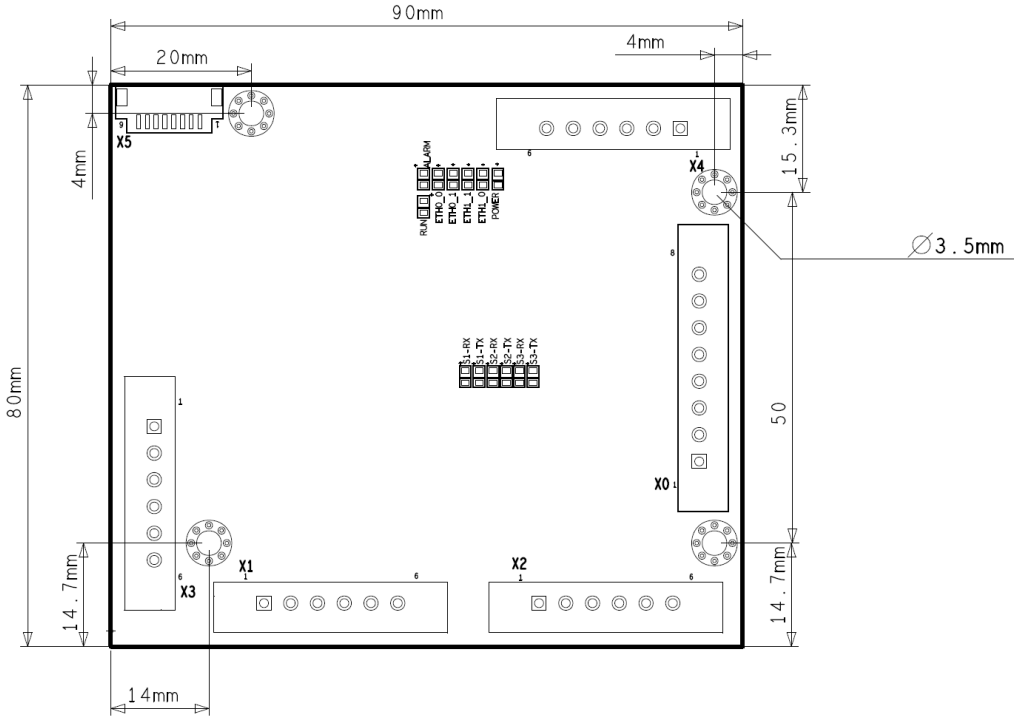
6.3 RM 单板尺寸

含插头高度占用 130mm。



6.4 VM 单板尺寸

含插头占用高度：230mm



7 协议转换应用

7.1 串口数据转换

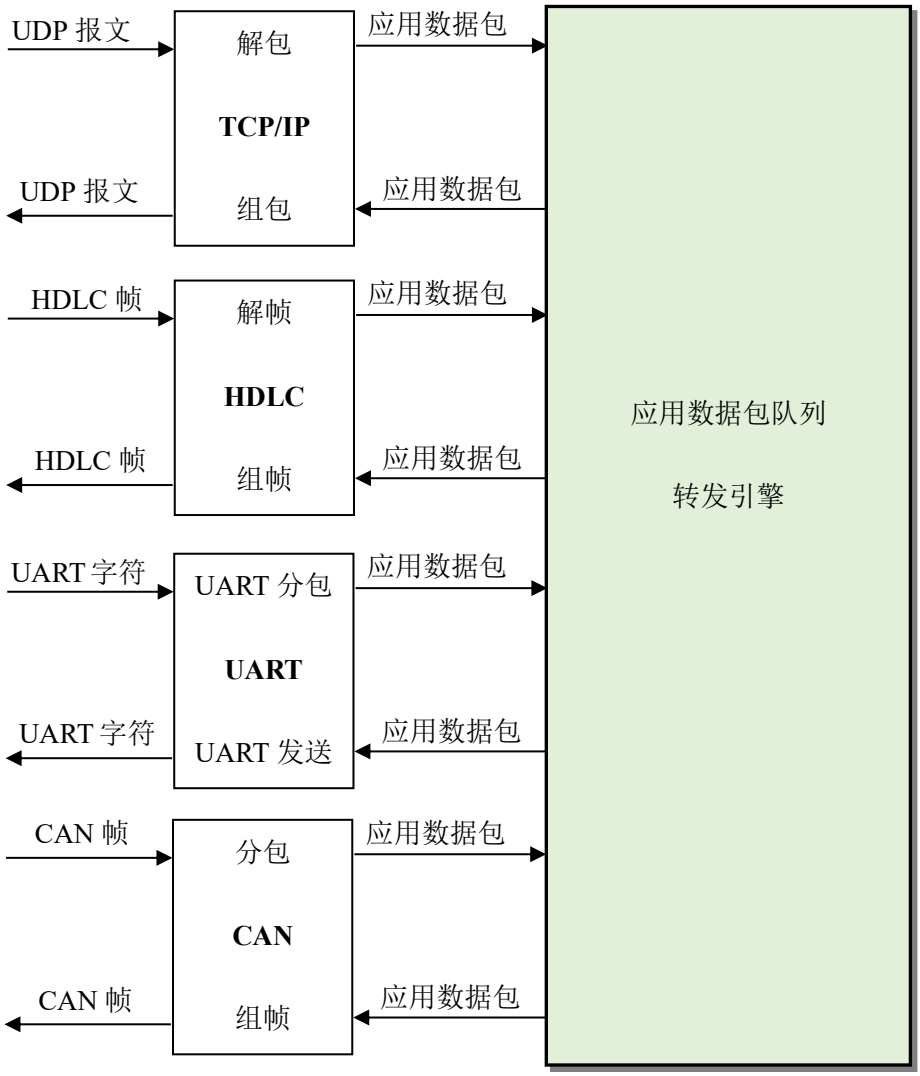
7.1.1 应用数据包与转换模型

串口数据转换包括：

- 串口与 UDP 之间的协议转换
- 同步串口、异步串口之间的数据转换

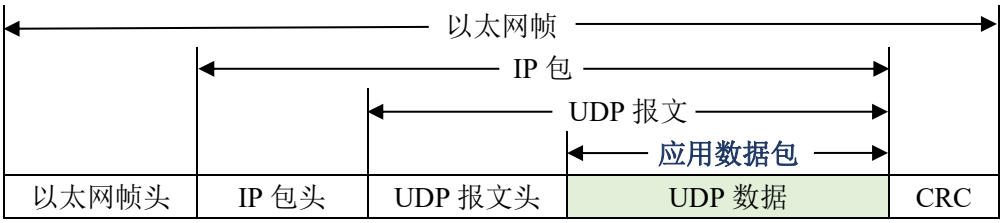
在接收时，不同类型接口的接收处理模块，对数据进行解包或解帧操作，提取应用数据包，送入系统的队列。

SDLC-BOARD 的转发引擎读取应用数据包队列，根据转发配置送至各接口的发送模块。发送模块负责应用数据包的组帧或组包操作，以产生不同类型的协议包或数据帧，通过物理接口发送出去。



7.1.2 UDP 报文格式

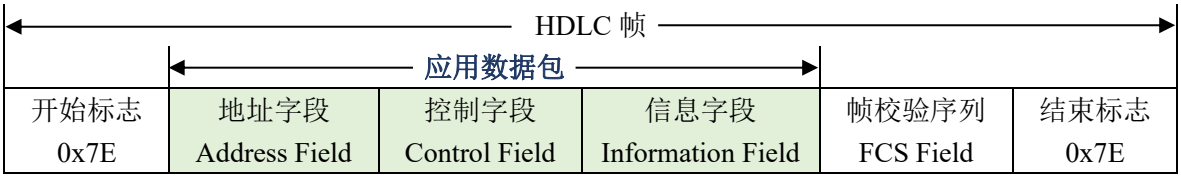
在 UDP 协议中，应用数据包封装在 UDP 报文的数据区中，每个 UDP 报文包含一个完整的应用数据包。



7.1.3 HDLC 帧格式

一个完整的 HDLC 帧由位于开始标志和结束标志之间的若干字段组成，包括地址字段、控制字段、信息字段及用于 CRC 校验的的 FCS 字段。

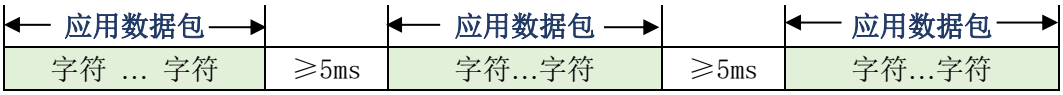
对于 SDLC-BOARD 来说，并不区分地址字段、控制字段和信息字段，而是把他们统一作为应用数据包交给上层应用填写和处理。



7.1.4 UART 数据包格式

当串口工作在异步 UART 模式时，从串口收到的是无头无尾的字符流，其中并没有用来进行分包或分帧的信息。

SDLC-BOARD 采用时间信息来分包，允许用户定义 UART 的分包时间。举例来说假如分包时间为 5ms，那么当超过 5ms 没有收到新的字符时，则认为一个数据包接收完毕。

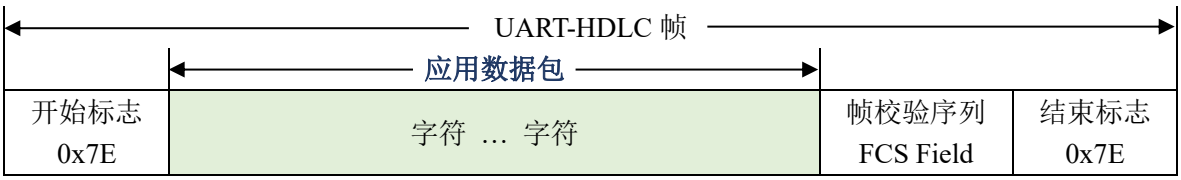


在实际应用中，包间隔期间期间不能有数据传输，导致通信带宽存在浪费，而且波特率越高浪费约严重。

7.1.5 UART-HDLC 帧格式

UART-HDLC 工作模式采用另一种策略为 UART 提供分包能力。如下图所示，数据发送方计算应用数据包的 CRC，并在头尾增加 0x7E 作为开始和结束标志，构建一个 UART-HDLC 帧。

这种策略不需要额外增加分包时间，可以充分利用通信带宽，但增加了通信双方处理的复杂度。



由于应用数据包和帧校验字段有可能出现 0x7E，因此收发双方需要对应用数据包和帧校验字段进行字符转义，转义规则如下：

- 0x7E：转义为 0x7D 0x5E 两个字符
- 0x7D：转义为 0x7D 0x5D 两个字符
- 其他字符：不转义

数据发送转义操作如下：

原始数据	实际发送数据
0x7E	0x7D 0x5E
0x7D	0x7D 0x5D
其他	无变化

数据接收转义操作如下：

实际接收数据	数据
0x7D 0x5E	0x7E
0x7D 0x5D	0x7D
其他	无变化

7.2 CAN 帧数据转换

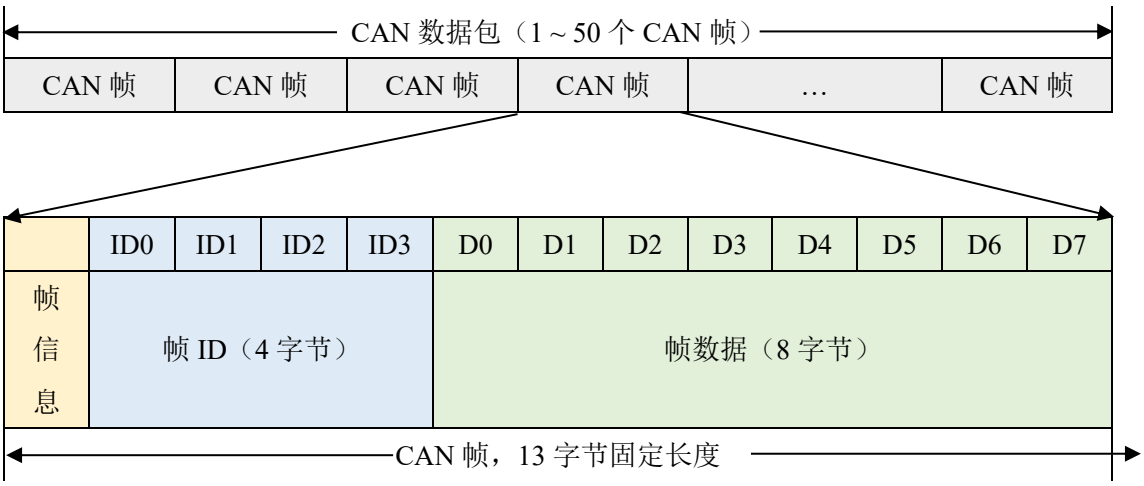
7.2.1 CAN 数据包与转换模型

由于 CAN 帧很短，为了提高转发效率，将多个 CAN 数据帧组成一个 CAN 数据包，每个 CAN 数据包对应一个其他接口的应用数据包。

7.2.2 CAN 数据包格式

7.2.2.1 包格式

CAN 数据包由 1~ 50 个 CAN 帧组成，每个 CAN 帧的固定长度为 13 字节。



7.2.2.2 帧信息

帧信息为 1 字节长度，其格式定义如下：

- FF：标志帧和扩展帧标识，1 为扩展帧，0 为标准帧；
- RTR：远程帧和数据帧标识，1 为远程帧，0 为数据帧；
- DLC：CAN 实际数据的长度。

帧信息	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	FF	RTR	保留	保留	DLC.3	DLC.2	DLC.1	DLC.0

7.2.2.3 帧 ID

帧 ID 占用 4 个字节，标准帧和扩展帧的 ID 位数不同。

标准帧 ID：标准帧 ID 为 11 位，取值范围为 0x000～0x7FF，有效填充为 ID.10～ID.0。

		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
标准帧 ID 4 字节	ID0								
	ID1								
	ID2						ID.10	ID.9	ID.8
	ID3	ID.7	ID.6	ID.5	ID.4	ID.3	ID.2	ID.1	ID.0

扩展帧 ID：扩展帧 ID 为 29 位，取值范围为 0x00000000 ~ 0x1FFFFFFF，有效填充为 ID.28～ID.0。

		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
标准帧 ID 4 字节	ID0				ID.28	ID.27	ID.26	ID.25	ID.24
	ID1	ID.23	ID.22	ID.21	ID.20	ID.19	ID.18	ID.17	ID.16
	ID2	ID.15	ID.14	ID.13	ID.12	ID.11	ID.10	ID.9	ID.8
	ID3	ID.7	ID.6	ID.5	ID.4	ID.3	ID.2	ID.1	ID.0

7.2.2.4 帧数据

帧数据占用 8 个字节空间，有效数据长度为 0～8 字节，第一个字节为有效数据的起始字节，有效长度由帧信息中的 DLC 值确定。

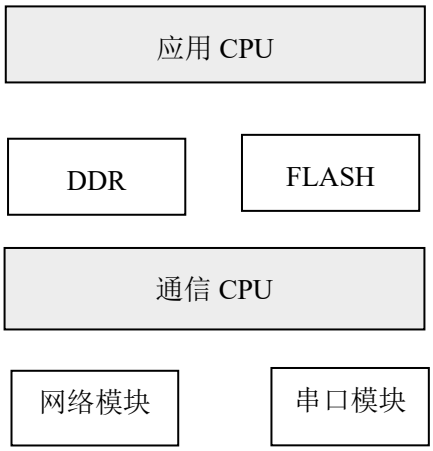
帧数据 8 字节	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
	Data 0	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7

8 在板编程二次开发

8.1 双 CPU 构架

SDLC-BOARD 核心包含两个独立的 CPU，二者通过共享内存交互数据：

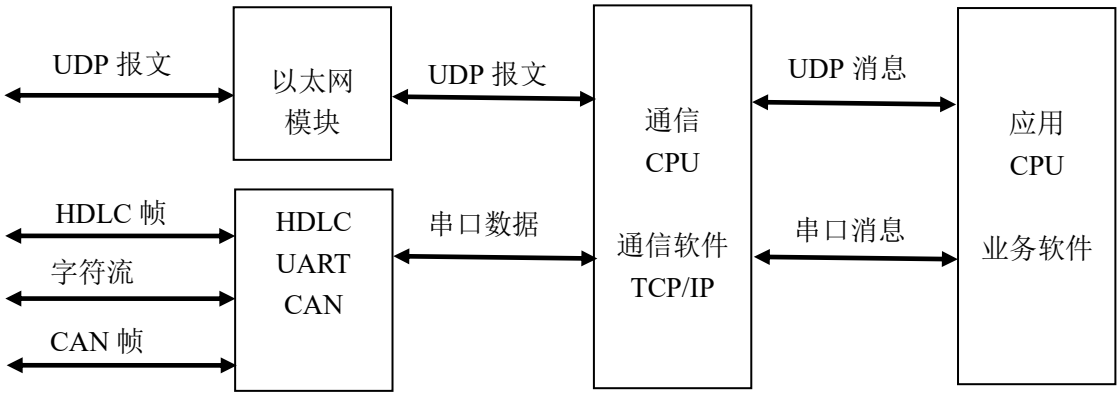
- 通信 CPU：提供网络、串口通信支持，提供配置管理支持；
- 应用 CPU：运行用户二次开发的业务软件，对来自通信 CPU 的数据进行处理加工。



8.2 数据交互模型

系统数据流如下所示，其中：

- UDP 接收流程：通信 CPU 的 TCP/IP 协议栈接收 UDP 报文，转换为 UDP 消息通过共享内存发送给应用 CPU；
- UDP 发送流程：应用 CPU 把 UDP 消息通过共享内存发送给通信 CPU，通信 CPU 的 TCP/IP 协议栈进行处理，转换为 UDP 报文通过以太网模块发出；
- 串口接收流程：通信 CPU 通过串口模块接收数据，通过共享内存交给应用 CPU 读取和处理；
- 串口发送流程：应用 CPU 把待发送的串口数据，通过共享内存送给通信 CPU，再通过串口模块组帧发出。

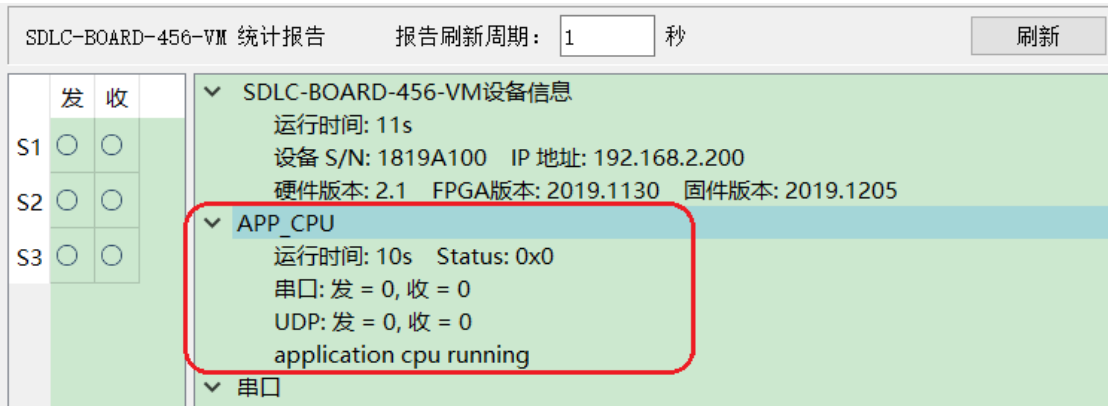


8.3 DMS 支持

8.3.1 显示应用 CPU 运行信息

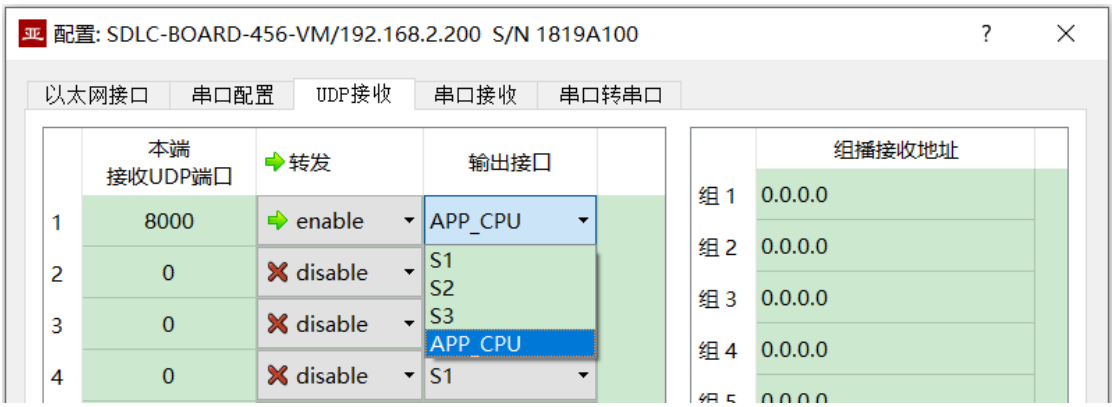
DMS 统计报告显示可以显示应用 CPU（APP_CPU）信息：

- 运行时间：应用 CPU 启动以来的运行时间；
- Status：应用 CPU 运行状态码，具体含义由用户定义；
- 串口收发统计：应用 CPU 从串口发送或接收的数据包计数；
- UDP 收发统计：应用 CPU 采用 UDP 收发的报文计数；
- APP_CPU 运行状态字符串。



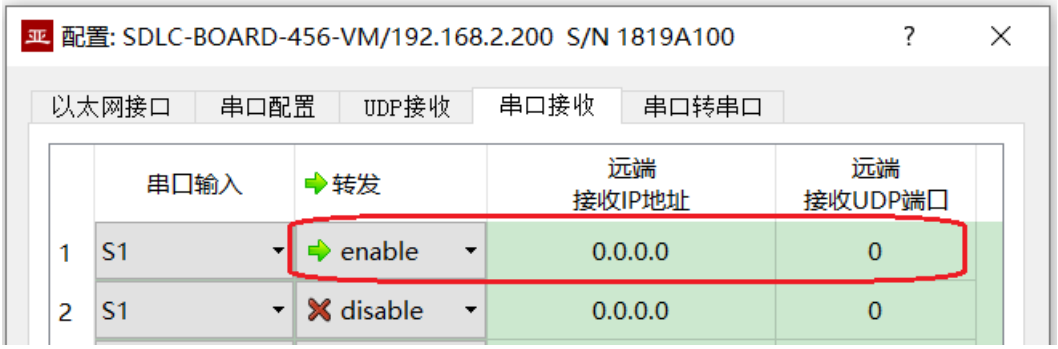
8.3.2 应用 CPU 接收 UDP 报文的配置

如下图所示，当 UDP 接收的输出接口选择 APP_CPU 时，从 8000 端口接收的 UDP 报文转发给应用 CPU 进行处理。



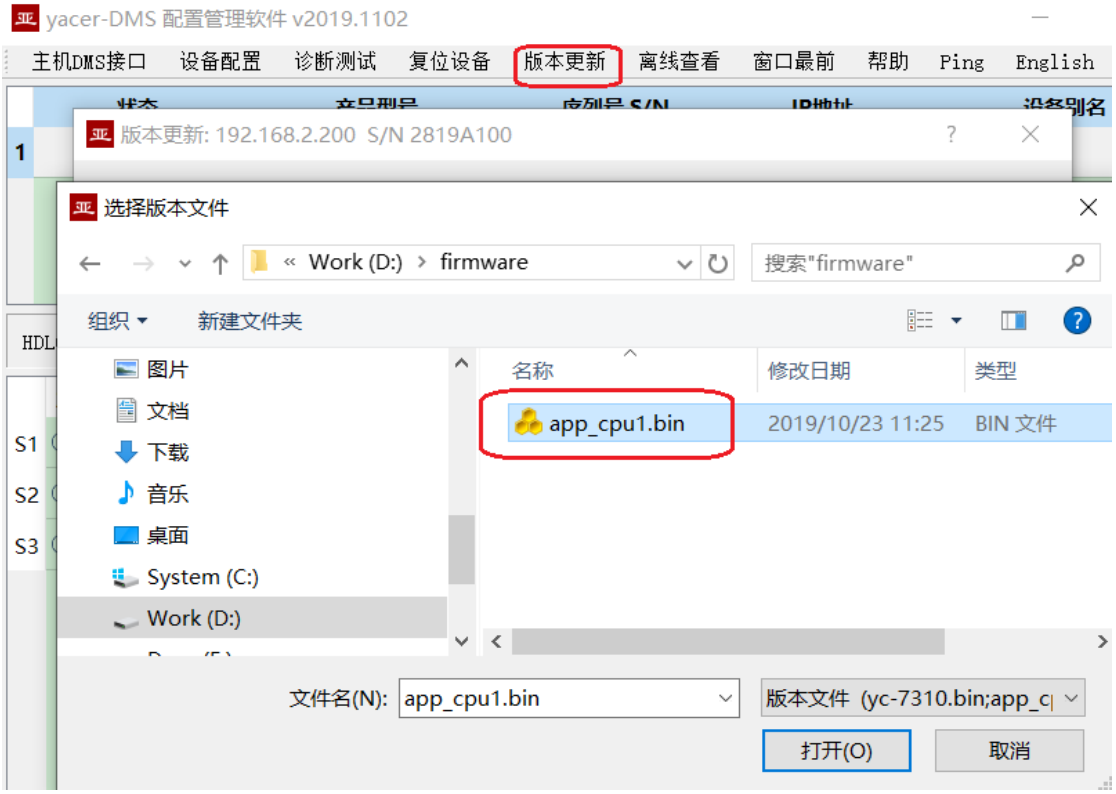
8.3.3 应用 CPU 接收串口数据的配置

如下图所示，使能串口-S1 的转发，远端 IP 地址或远端 UDP 端口如果为 0，则从 S1 接收的串口数据转发给应用 CPU 进行处理。



8.3.4 加载应用 CPU 软件版本

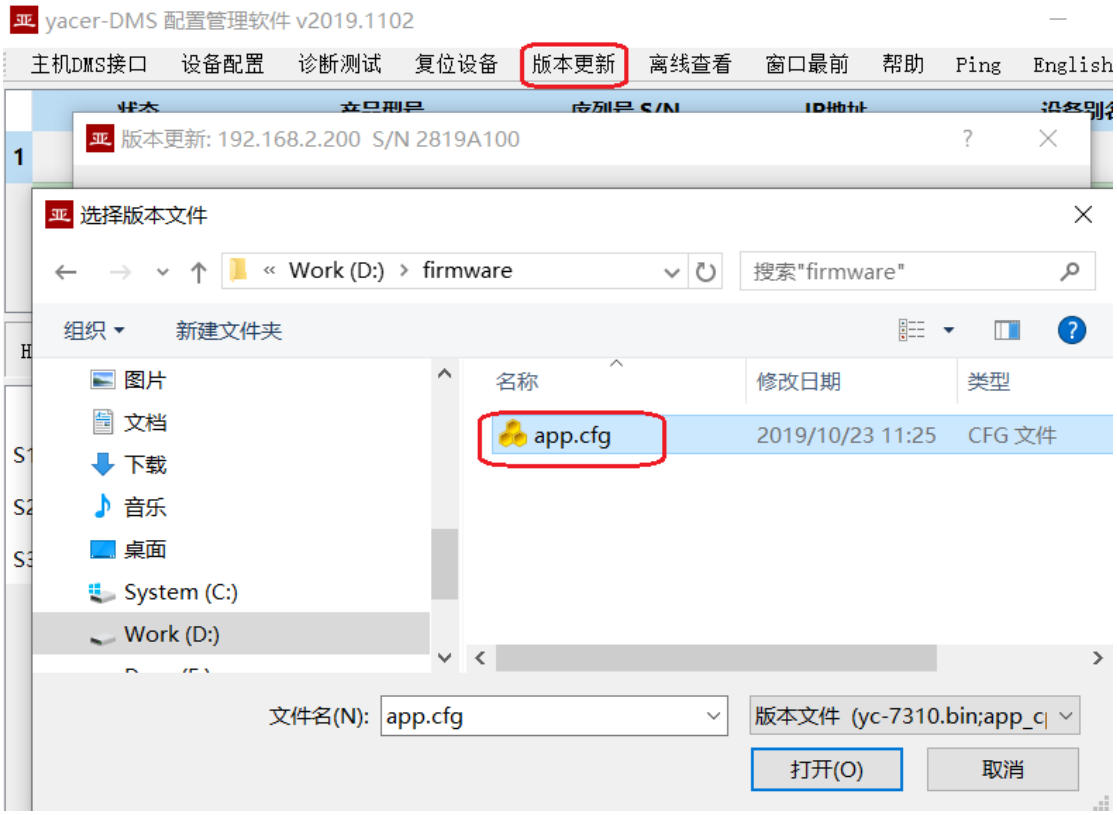
用户开发的软件版本，生成名称为 app_cpu1.bin 的版本文件，借助 DMS 的版本更新功能进行上传和烧录，烧录完成后复位设备即可加载运行最新的应用 CPU 软件版本。



8.3.5 加载应用 CPU 的配置

应用 CPU 的配置必须为 app.cfg，其内容和格式由用户自行定义，长度不能超过 1M 字节。

利用 DMS 同样可以上传和烧录配置文件至 FLASH，当系统复位后，新的配置即可启用。



8.4 软件开发与编译

请联系厂家技术支持。