

RS232/484/422 通信模块

AN3485 用户手册

Rev. 1.00



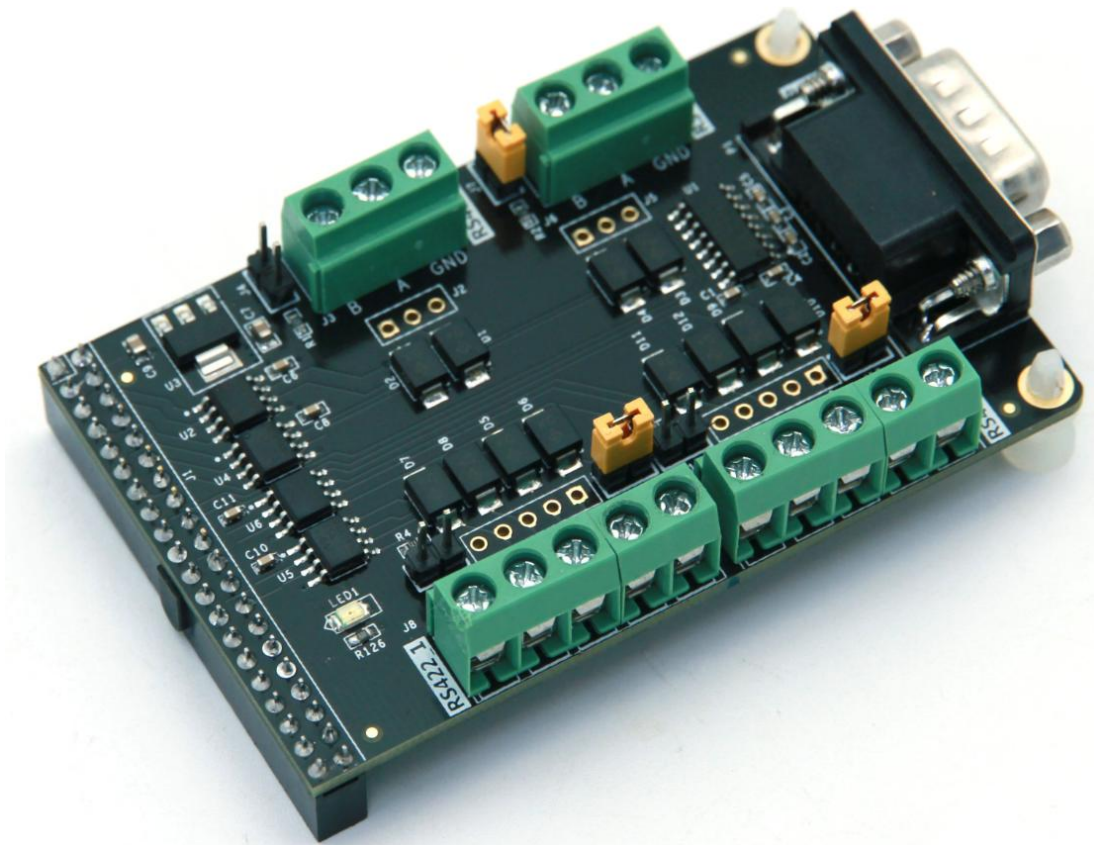
版本记录

版本	时间	作者	描述
Rev1.00	2017-8-15		First Release

第一部分 RS232/485/422 通信模块说明

黑金 AN3845 模块专门为工业现场应用设计的 RS232/485/422 通信模块。它包含一路 RS232 接口，2 路 RS485 和 2 路 RS422 通信接口。配合开发板实现 RS232、485 和 422 的数据远程传输和通信。RS232、485 和 422 接口分别采用 MAX3232、MAX3485 和 MAX3490 芯片作为电平转换芯片。模块留有一个 40 针的排母用于连接开发板，RS232 接口为一个标准的 DB9 串口公座，通过串口线直接连接电脑或者其他设备；RS485 和 RS422 接口采用接线端子跟外部连接，超远距离传输可达上千米，另外 RS485 和 RS422 接口部分带有正负 15KV 的 ESD 防护功能。

AN3845 模块实物照片如下：



AN3845 通信模块正面图

1.1 AN3485 模块的参数说明

以下为 AN3485 通信模块的详细参数:

➤ RS232 接口

- 一路标准的 DB9 公座串行接口；
- 使用 MAX3232 作为 RS232 和 TTL 电平的转换；
- 传输率高达 120Kbps 数据通讯速率

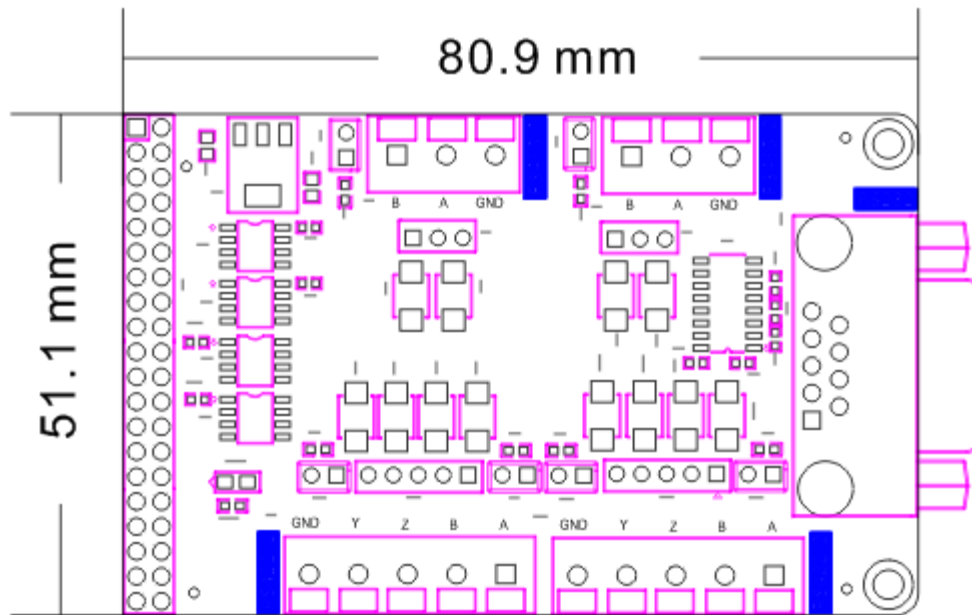
➤ RS485 接口

- 两路 RS485 接口，采用 3 线的接线端子；
- 使用 MAX3485 作为 RS485 和 TTL 的电平转换；
- 工业级设计，抗干扰能力超强，同时采用有效的防雷设计；
- 具有 120 欧匹配电阻，插上跳线帽即可使能匹配电阻，长距离传输时建议短接。
- 支持多机通讯，允许接在最多 128 个设备的总线上
- 传输率高达 500Kbps 数据通讯速率。

➤ RS422 接口

- 两路 RS422 接口，采用 5 线的接线端子；
- 使用 MAX3490 作为 RS422 和 TTL 的电平转换；
- 工业级设计，抗干扰能力超强，同时采用有效的防雷设计；
- 具有 120 欧匹配电阻，插上跳线帽即可使能匹配电阻，长距离传输时建议短接。
- 支持多机通讯，允许接在最多 128 个设备的总线上
- 传输率高达 500Kbps 数据通讯速率。

1.2 AN3485 模块尺寸

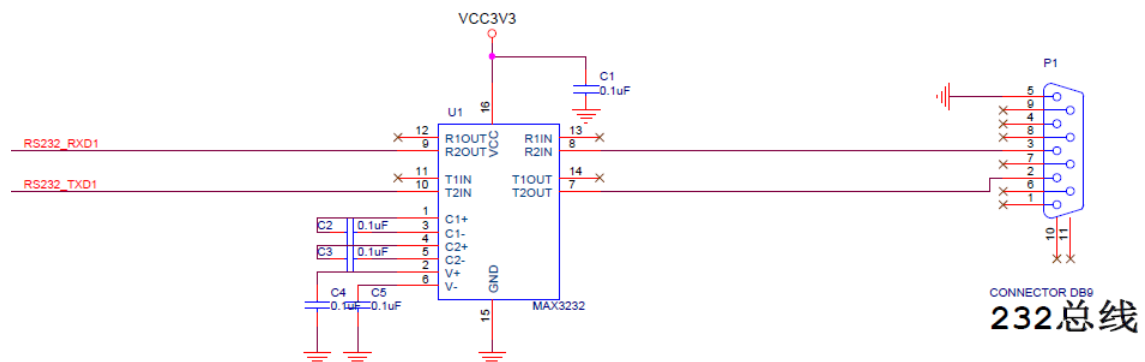


AN3485 通信模块尺寸图

第二部分 模块功能说明

2.1 RS232 电路设计

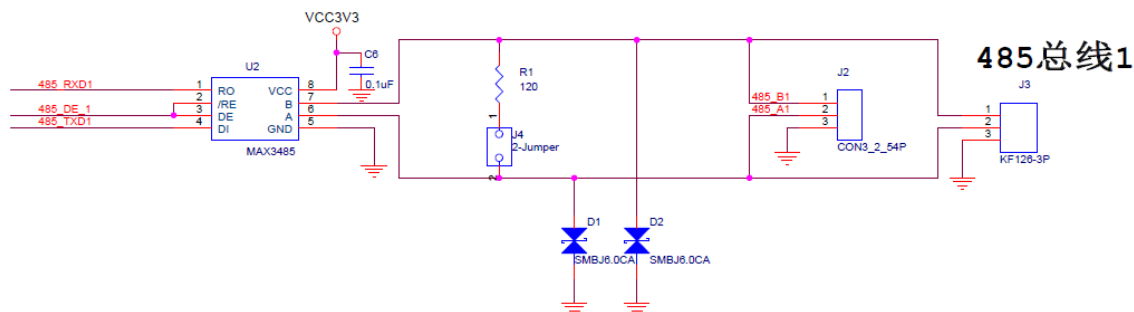
AN3485 模块的 RS232 接口采用 MAX3232 芯片实现 RS232 和 +3.3V TTL 电平的转换。TTL 电平的串口接收和发送信号 (RXD, TXD) 连接到 40 针的连接器上跟外面的 FPGA 芯片或者 ARM 芯片实现串口通信。RS232 串口通信的最高速度为 120kbps，RS232 接口的原理设计图如下图所示。



2.2 RS485 电路设计

AN3485 模块的 RS485 接口采用 MAX3485 芯片实现 RS485 和 +3.3V TTL 电平的转换。TTL 电平的 485 信号 (RXD, DE, TXD) 连接到 40 针的连接器上跟外面的 FPGA 芯片或者 ARM 芯片实现 485 数据通信, 其中 DE 信号用来控制 485 的传输方向。

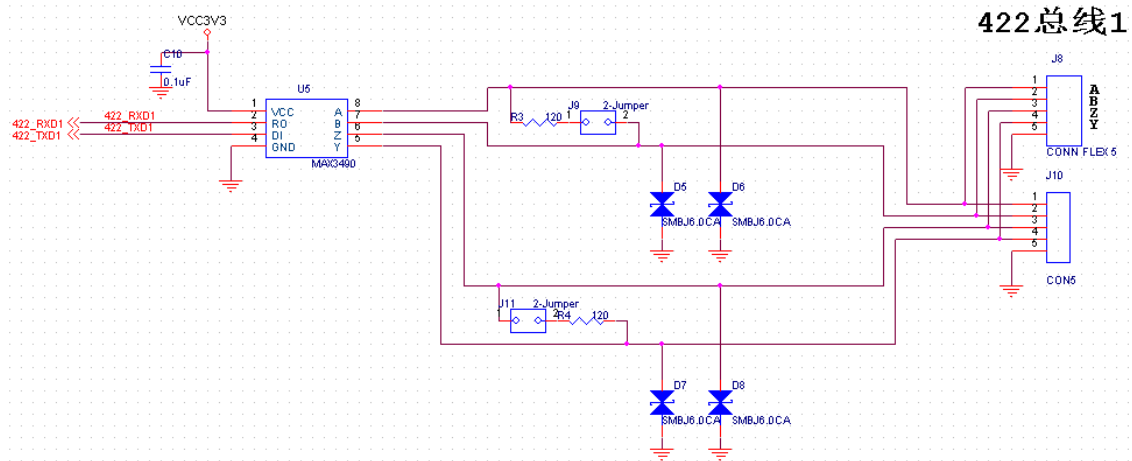
RS485 总线有 120 欧姆的端接电阻, 当 485 接口作为从设备时, 需要把跳线帽连接使能 120 欧姆的端接电阻。另外 RS485 的 A 和 B 总线上分别连接了 TVS 瞬态抑制二极管, 提供正负 15KV 的 ESD 防护功能。RS485 通信的最高速度为 500kbps, RS485 接口的原理设计图如下图所示。



2.3 RS422 电路设计

AN3485 模块的 RS422 接口采用 MAX3490 芯片实现 RS422 和 +3.3V TTL 电平的转换。TTL 电平的 422 信号 (RXD, TXD) 连接到 40 针的连接器上跟外面的 FPGA 芯片或者 ARM 芯片实现 422 数据通信, 因为 RS422 传输是全双工的, 所以没有 DE 信号。

RS422 发送总线 Z-Y 和接收总线 A-B 之间都预留了 120 欧姆的端接电阻, 用户使用时只要连接接收总线 (A-B) 的跳线帽, 使能 120 欧姆的终端匹配。另外 RS422 的接收和发送总线上分别连接了 TVS 瞬态抑制二极管, 提供正负 15KV 的 ESD 防护功能。RS422 通信的最高速度为 500kbps, RS422 接口的原理设计图如下图所示。



2.4 模块 40 针排母的引脚分配：

引脚号	引脚名称	备注
1	GND	地
2	+5V	5V 电源输入
3	-	-
4	-	-
5	485_RXD1	第一路 RS485 数据接收
6	485_DE_1	第一路 RS485 发送使能
7	485_TXD1	第一路 RS485 数据发送
8	-	-
9	-	-
10	-	-
11	485_RXD2	第二路 RS485 数据接收
12	485_DE_2	第二路 RS485 发送使能
13	485_TXD2	第二路 RS485 数据发送
14	-	-
15	-	-
16	-	-
17	RS232_RXD1	RS232 数据接收
18	RS232_TXD1	RS232 数据发送
19	-	-
20	-	-
21	422_RXD2	第二路 RS422 数据接收
22	422_TXD2	第二路 RS422 数据发送
23	-	-
24	-	-
25	422_RXD1	第一路 RS422 数据接收

26	422_TXD1	第一路 RS422 数据发送
27	-	-
28	-	-
29	-	-
30	-	-
31	-	-
32	-	-
33	-	-
34	-	-
35	-	-
36	-	-
37	GND	地
38	GND	地
39	-	-
40	-	-

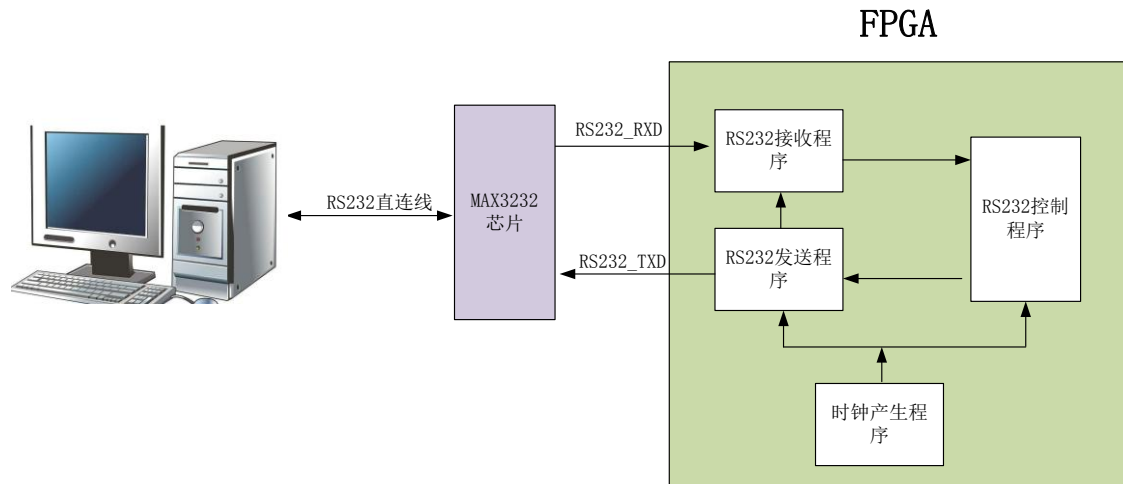
第三部分 RS232 通信程序

3.1 RS232 通信程序介绍

RS232 通信例程 (rs232_test) 的功能主要演示 FPGA 开发板和 AN3485 通信模块的串口接收和发送的功能，用户需要准备一根 RS232 串口线(直连线) 连接开发板到 PC 上进行串口的数据通信。

在例程没有接收到 PC 机发来信息的时候,FPGA 程序会不断的通过 AN3485 的串口向 PC 机发送“ Hello ALINX AN3485” 的信息。当用户从 PC 机的串口工具里发送数据给 AN3485 通信模块到开发板时，FPGA 程序接收到数据后会数据发回给 PC,从而实现 Loopback 的功能。

串口通信例程包含一个 TOP 程序 rs232_test.v 和四个子程序 ,四个子程序分别为 RS232 发送程序 uarttx.v , RS232 接收程序 uartrx.v,时钟产生程序 clkdiv.v 和 RS232 发送控制程序 uartctrl.v。FPGA 程序的设计框图如下图所示:



下面我们来分别对每个程序做一下介绍。

1. 时钟产生程序 clkdiv.v

用来产生 uart 接收和发送波特率为 9600bps 的时钟信号，程序对 50Mhz 的系统时钟进行分频，分频参数 326 计算如下：

这里产生的时钟 clkout 为波特率的 16 倍，假设数据的波特率为 p，则这里的时钟 clkout 的频率为 16*p。以波特率 p 为 9600 为例，系统时钟为 50MHz，则分频系数为 $50000000 / (16 * 9600) = 325.5$ ，取整为 326。

clkout 时钟取 16 倍波特率的目的是为了在 uart 接收的时候对每比特接收的数据有 16 个时钟采样，取中间的采样值，以保证采样不会滑码或误码。

2. 串口发送程序 uarttx.v。

控制串口的一个字节数据发送过程。当空闲状态，串口的发送线路(TX)处于高电位；当收到发送数据指令后，拉低线路一个数据位的时间 T，接着数据按低位到高位依次发送，数据发送完毕后，接着发送奇偶校验位和停止位（停止位为高电位），一帧数据(一个字节)发送结束。

3. 串口接收程序 uartrx.v。

控制串口的一个字节数据接收过程。当空闲状态，串口的接收线路(RX)处于高电位；当检测到线路 RX 的下降沿（线路电位由高电位变为低电位）时说明线路有数据传输，按照约定的波特率从低位到高位接收数据，数据接收完毕后，接着接收并比较奇偶校验位是否正确，如果正确则通知后续设备准备接收数据或存入缓存。

由于 UART 是异步传输，没有传输同步时钟。为了能保证数据传输的正确性，

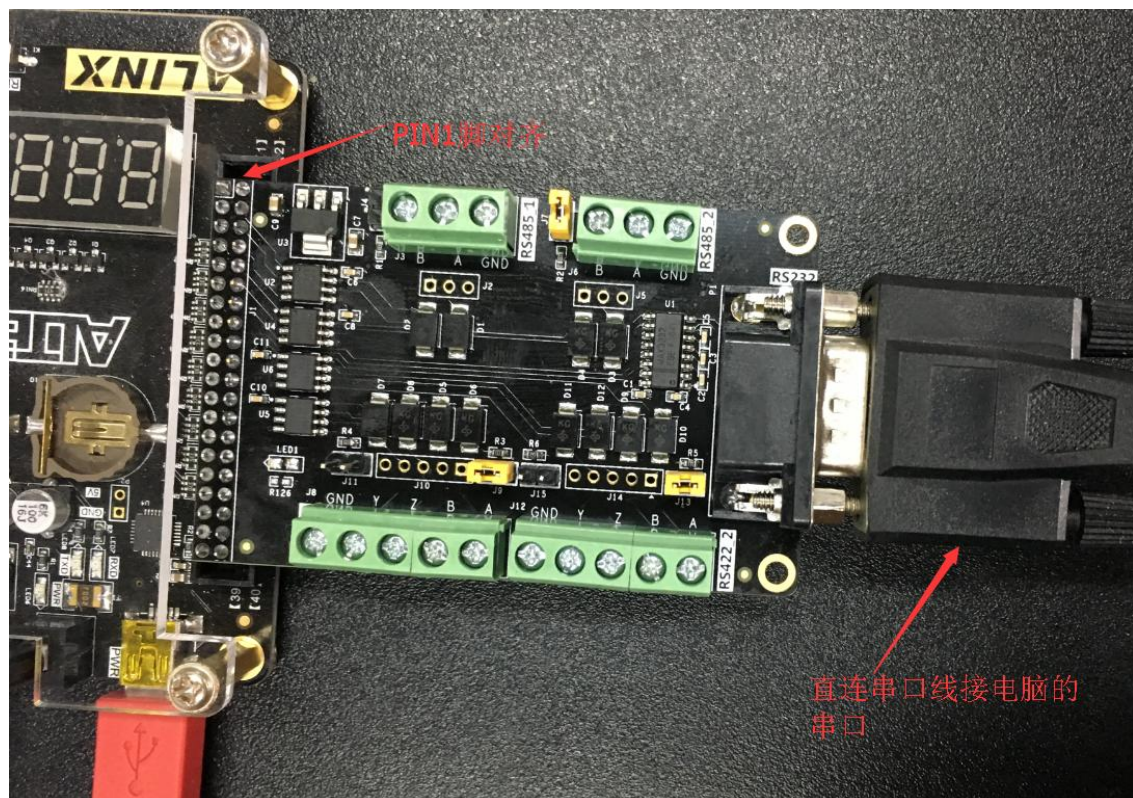
UART 采用 16 倍数据波特率的时钟进行采样。每个数据有 16 个时钟采样，取中间(第 8 个时钟)的采样值，以保证采样不会滑码或误码。一般 UART 一帧的数据位数为 8，这样即使每个数据有一个时钟的误差，接收端也能正确地采样到数据。

4. 串口发送控制程序 uartctrl.v。

上电后程序不断的向串口发送“Hello ALINX AN3485”字符串, 如果串口有接收到 PC 机发来的数据优先把接收到的数据发回给 PC。

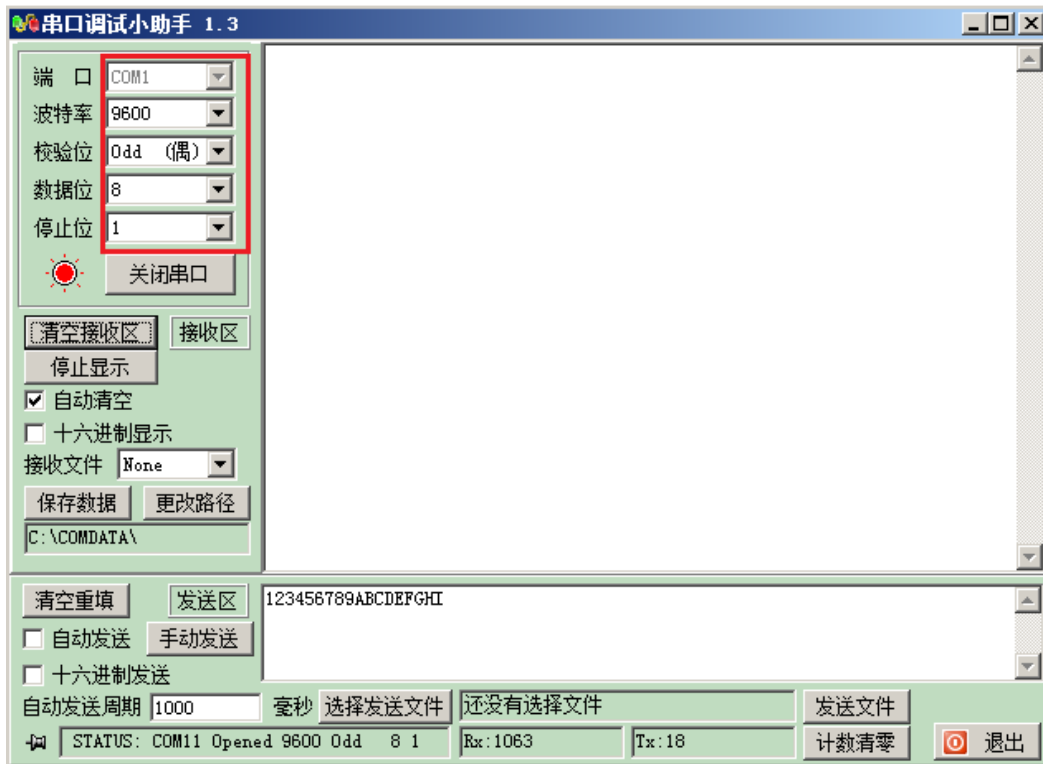
3.2 RS232 通信硬件连接

AN3485 模块和 FPGA 开发板的硬件连接很简单，只要把 AN3485 模块的 40 针的母座 J1 插到 FPGA 开发板的扩展口上，连接器的管脚 1 对齐就好了。以下为黑金 AX301 开发板的 J1 扩展口和 AN3485 通信模块的硬件连接图(如果需要连接开发板 J2 扩展口，管脚需要重新分配)，如果是其它开发板用户需要自己分配管脚。然后我们需要用直连串口线(直连母头)连接 AN3485 模块的 RS232 口和电脑的 RS232 接口。



3.3 RS232 通信实验

开发板上电, 打开串口调试助手, 选择正确的 COM1 或者 COM2, 设置波特率为 9600, 校验位设置为奇校验或偶校验, 停止位为 1。具体配置如下:



下载程序后, 我们可以从显示窗口看到从 FPGA 不断发来的 “Hello ALINX AN3845” 信息。

我们已经在 RS232 通信程序里说明过，不再介绍。

1. RS485 发送程序 rs485_tx.v。

RS485 发送程序跟 RS232 的发送程序基本一样，但 RS485 发送程序多一个发送使能控制信号 DE，这个信号在 RS485 数据发送的时候需要置高，其它的时候置低。

2. RS485 接收程序 rs485_rx.v。

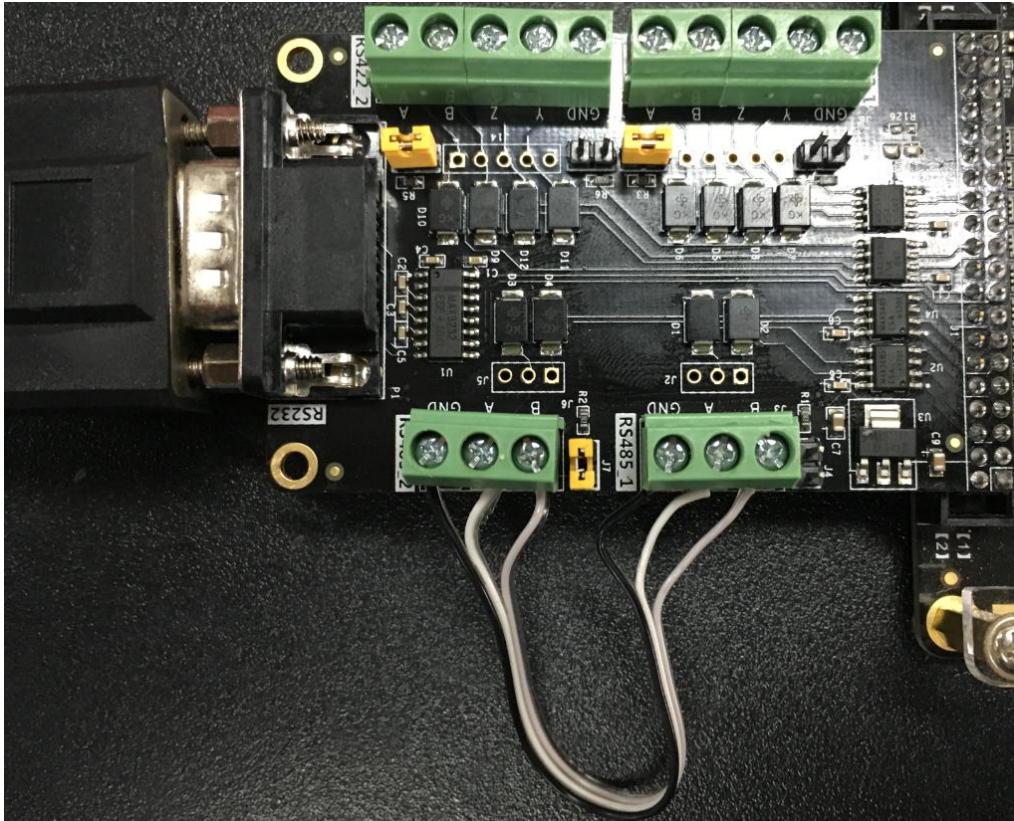
RS485 接收程序是跟 RS232 的接收程序是一样的。

3. RS485 控制程序 rs485_ctrl.v

RS485 控制程序程序判断 RS485 的接收指令，如果接收指令的第一个字为 'S'，存储指令到 RAM 中，直到接收到回车或者换行符结束一条指令的接收。如果接收到一条指令后，会启动 RS485 发送程序，把接收到的指令发送出去，发送的长度为接收到的指令长度。

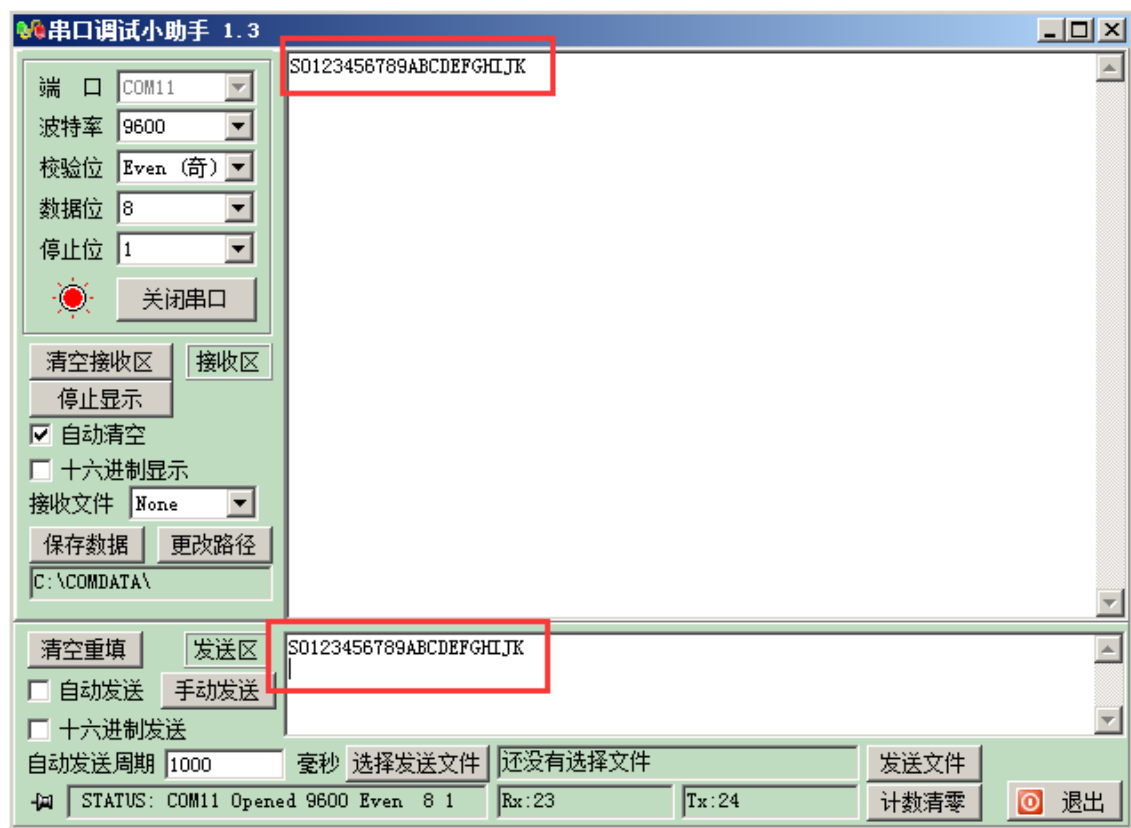
4.2 RS485 通信硬件连接

首先需要把 RS485 第一通道和第二通道的 485 总线用线进行连接 (A 和 A 对接，B 和 B 对接，， GND 对 GND 对接)，不然无法进行数据的通信。同样再把 AN3485 模块插到 AX301 FPGA 开发板的扩展口(J1) ,RS232 接口用串口线跟电脑串口连接。硬件连接如下图所示：



4.3 RS485 通信实验

开发板上电, 下载 rs485_test.sof 文件到 FPGA , 打开串口调试助手 , 选择正确的 COM1 或者 COM2 , 设置波特率为 9600, 校验位设置为奇校验或偶校验 , 停止位为 1。然后串口发送命令 , 以大写的 S 作为开始符 , 后面跟一串数据 , 以回车作为结束符。比如我们这里发送字符串 “S0123456789ABCDEFGHIJK 回车” 后 , 串口能够接收到同样的字符串 , 说明 RS485 数据通信正常。

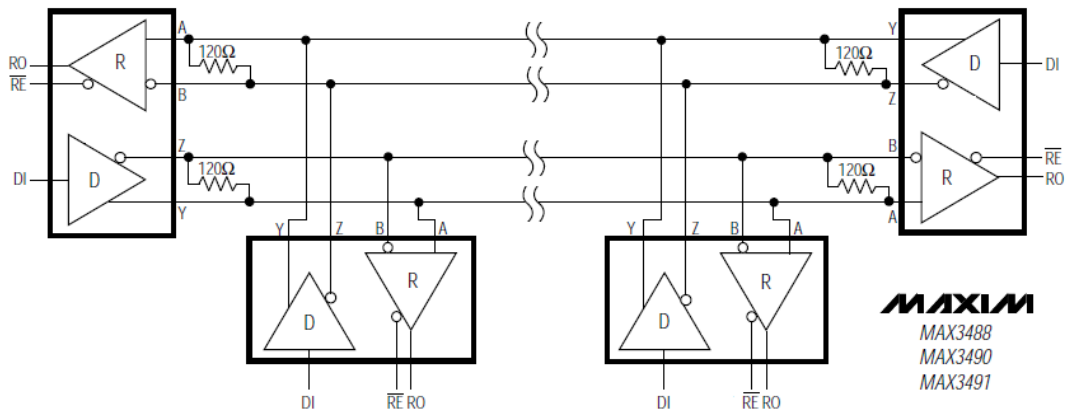


这里一定要注意,命令需要大写的'S'开始,回车或者换行符结束,否则 RS485 通信是不成功的。

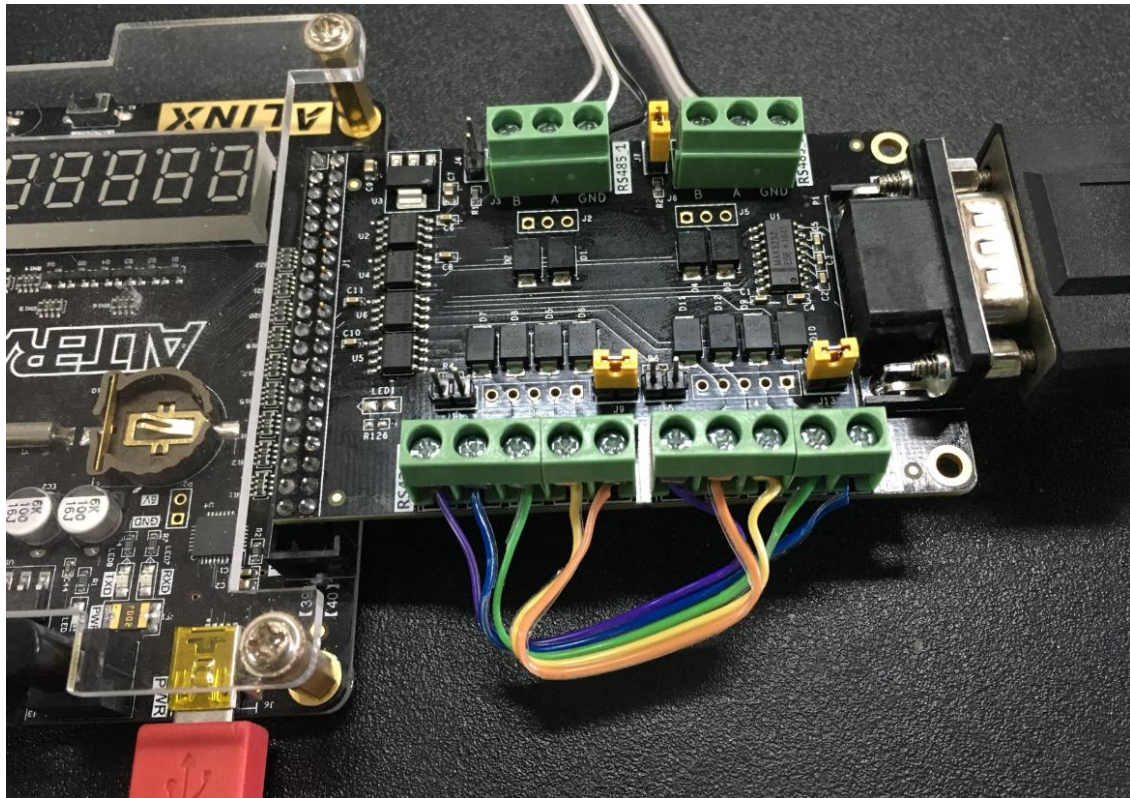
第五部分 RS422 通信程序

RS422 的通信程序基本上跟 RS485 的例程是一样的,只是 RS422 不需要 DE 信号,这里我们不再对程序进行说明。另外实验方法也是跟 RS485 通信一样,从电脑发送指令给 RS232 串口,通过第一路的 RS422 发送,第二路的 RS422 接收命令后,再通过第二路 RS422 发送回第一路的 RS422,第一路 RS422 接收到后命令后再返回给 RS232 串口到电脑。

RS422 第一路和第二路之间的总线连接需要特别注意, A 和 B 为数据的 RS422 的数据接收, Y 和 Z 为 RS422 的数据发送。所以第一路的 A 和 B 需要跟第二路的 Y 和 Z 对连,第一路的 Y 和 Z 需要跟第二路的 A 和 B 对连。下图为 MAX3490 芯片手册的连接示意图。



AN3485 模块 2 路 RS422 数据通信的硬件连接图如下所示 (A 和 Y 相连 , B 和 Z 相连 , GND 和 GND 相连), **注意线不要接反。**



实验方法和现象大家参考 RS485 通信的章节。