

AT32 USB Multi Bridge

前言

AT32 USB 多功能桥接设备提供一个专有的高速USB接口，可以使用多种传输接口与目标进行通信：UART, CAN, SPI, I²C, RS485, USB使用CDC协议进行外设数据透传，可搭配USART IAP, CAN IAP, SPI IAP, I²C IAP等使用，部分系统下需要安装USB CDC驱动。

支持外设：

- USB to USART
- USB to CAN
- USB to SPI
- USB to I²C
- USB to RS485

支持型号列表：

支持型号	AT32F405 系列
------	-------------

目录

1	概述.....	5
1.1	USB Bridge 特点.....	5
1.1.1	外设选择	5
1.1.2	波特率修改	7
1.1.3	硬件资源	8
1.1.4	软件资源	9
1.2	USB 转接 demo 使用	9
2	USB 转 USART.....	10
2.1	USART 转接线连接	10
3	USB 转 CAN 桥接.....	11
3.1	CAN 转接线连接.....	11
3.2	USB CAN 数据转发	11
4	USB 转 SPI 桥接.....	13
4.1	SPI 转接线连接.....	13
4.2	USB SPI 数据转发	13
5	USB 转 I²C 桥接.....	14
5.1	I ² C 转接线连接.....	14
5.2	USB I ² C 数据转发.....	14
6	USB 转 RS485 桥接.....	15
6.1	RS485 转接线连接.....	15
7	版本历史	16

表目录

表 1 CAN 支持波特率	7
表 2 SPI 支持时钟频率	8
表 3 I ² C 支持时钟频率	8
表 4 文档版本历史	16

图目录

图 1 USB SETUP 请求格式	6
图 2 SET LINE CODING 命令	7
图 3 Line Coding Structure.....	7
图 4 AT32-USBHS-Adaptor 图	8
图 5 使用连接图.....	9
图 6 USART 连接图.....	10
图 7 CAN 连接图	11
图 8 SPI 连接图	13
图 9 I2C 连接图	14
图 10 RS485 连接图.....	15

1 概述

1.1 USB Bridge 特点

USB 端使用 USBHS CDC 协议虚拟一个 COM 口，支持多种外设接口，可具体需要选择转接外设接口，同一时刻只支持转一种外设，可通过 USB 控制请求配置当前使用哪一个外设，同时也支持通过 SET LINE CODEING 命令修改外设波特率。

支持外设：

- USB to USART
- USB to CAN
- USB to SPI
- USB to I²C
- USB to RS485

1.1.1 外设选择

可通过 3 种方法来选择外设：

- 通过代码初始化时固定选择支持的外设，后续不再更改
- 通过 USB 控制端点 SETUP 请求来实时选择当前外设
- 通过 USB HID 命令请求来选择当前外设

1.1.1.1 初始化选择外设

USB Multi Bridge 在初始化时可通过函数接口 `usb_multi_bridge_set_type(USB_TO_USART);` 设置当前使用接口，包括如下参数：

USB_TO_USART: USB 转串口

USB_TO_CAN: USB 转 CAN

USB_TO_SPI: USB 转 SPI

USB_TO_I2C: USB 转 I²C

USB_TO_RS485: USB 转 RS485

USB_TO_IDLE: 当前 IDLE 状态，没有转接外设

1.1.1.2 USB 控制请求选择外设

另外可通过 USB 端点 0 SETUP 控制传输命令配置当前支持的转接接口，可参考<Universal Serial Bus Specification 2.0>，如下是 SETUP 数据包格式：

图 1 USB SETUP 请求格式

Offset	Field	Size	Value	Description
0	<i>bmRequestType</i>	1	Bitmap	Characteristics of request: D7: Data transfer direction 0 = Host-to-device 1 = Device-to-host D6...5: Type 0 = Standard 1 = Class 2 = Vendor 3 = Reserved D4...0: Recipient 0 = Device 1 = Interface 2 = Endpoint 3 = Other 4...31 = Reserved
1	<i>bRequest</i>	1	Value	Specific request (refer to Table 9-3)
2	<i>wValue</i>	2	Value	Word-sized field that varies according to request
4	<i>wIndex</i>	2	Index or Offset	Word-sized field that varies according to request; typically used to pass an index or offset
6	<i>wLength</i>	2	Count	Number of bytes to transfer if there is a Data stage

使能 USB TO USART:

<i>bmRequestType</i>	<i>bRequest</i>	<i>wValue</i>	<i>wIndex</i>	<i>wLength</i>
0x40	0x01	0x0000	0x0000	0x0000

使能 USB TO CAN:

<i>bmRequestType</i>	<i>bRequest</i>	<i>wValue</i>	<i>wIndex</i>	<i>wLength</i>
0x40	0x01	0x0001	0x0000	0x0000

使能 USB TO SPI:

<i>bmRequestType</i>	<i>bRequest</i>	<i>wValue</i>	<i>wIndex</i>	<i>wLength</i>
0x40	0x01	0x0002	0x0000	0x0000

使能 USB TO I²C:

<i>bmRequestType</i>	<i>bRequest</i>	<i>wValue</i>	<i>wIndex</i>	<i>wLength</i>
0x40	0x01	0x0003	0x0000	0x0000

使能 USB TO RS485:

<i>bmRequestType</i>	<i>bRequest</i>	<i>wValue</i>	<i>wIndex</i>	<i>wLength</i>
0x40	0x01	0x0004	0x0000	0x0000

1.1.1.3 HID 命令选择外设

通过设备 HID 设备配置当前使用的转接外设，HID VID（0x2E3C），PID(0xFF01)。
 如下是上位机发送数据对于切换的外设：

HID 数据	切换外设
0xA1 0x00	USB TO USART
0xA1 0x01	USB TO CAN
0xA1 0x02	USB TO SPI
0xA1 0x03	USB TO I2C
0xA1 0x04	USB TO RS485

1.1.2 波特率修改

不同转接外设支持不同的波特率或者频率，可通过 SET LINE CODING 命令进行配置。
 注意：受外部电路配置影响，若出现通讯失败可尝试降低通讯速率。

图 2 SET LINE CODING 命令

bmRequestType	bRequestCode	wValue	wIndex	wLength	Data
00100001B	SET_LINE_CODING	Zero	Interface	Size of Structure	Line Coding Structure

Line coding structure: 通过 dwDTERate 参数传递设置的波特率。

图 3 Line Coding Structure

Offset	Field	Size	Value	Description
0	<i>dwDTERate</i>	4	Number	Data terminal rate, in bits per second.
4	<i>bCharFormat</i>	1	Number	Stop bits 0 - 1 Stop bit 1 - 1.5 Stop bits 2 - 2 Stop bits
5	<i>bParityType</i>	1	Number	Parity 0 - None 1 - Odd 2 - Even 3 - Mark 4 - Space
6	<i>bDataBits</i>	1	Number	Data bits (5, 6, 7, 8 or 16).

USART、RS485 支持波特率: 1600 bps~6 Mbps

CAN 支持波特率:

表 1 CAN 支持波特率

支持波特率	DwDTERate 参数
1 Mbps	1000000
500 kbps	500000
250 kbps	250000
125 kbps	125000

SPI 支持时钟频率:

表 2 SPI 支持时钟频率

支持时钟频率	DwDTERate 参数
13.5 MHz	13500000
6.75 MHz	6750000
3.375 MHz	3375000
1.6875 MHz	1687500
0.84375 MHz	843750

I²C 支持时钟频率:

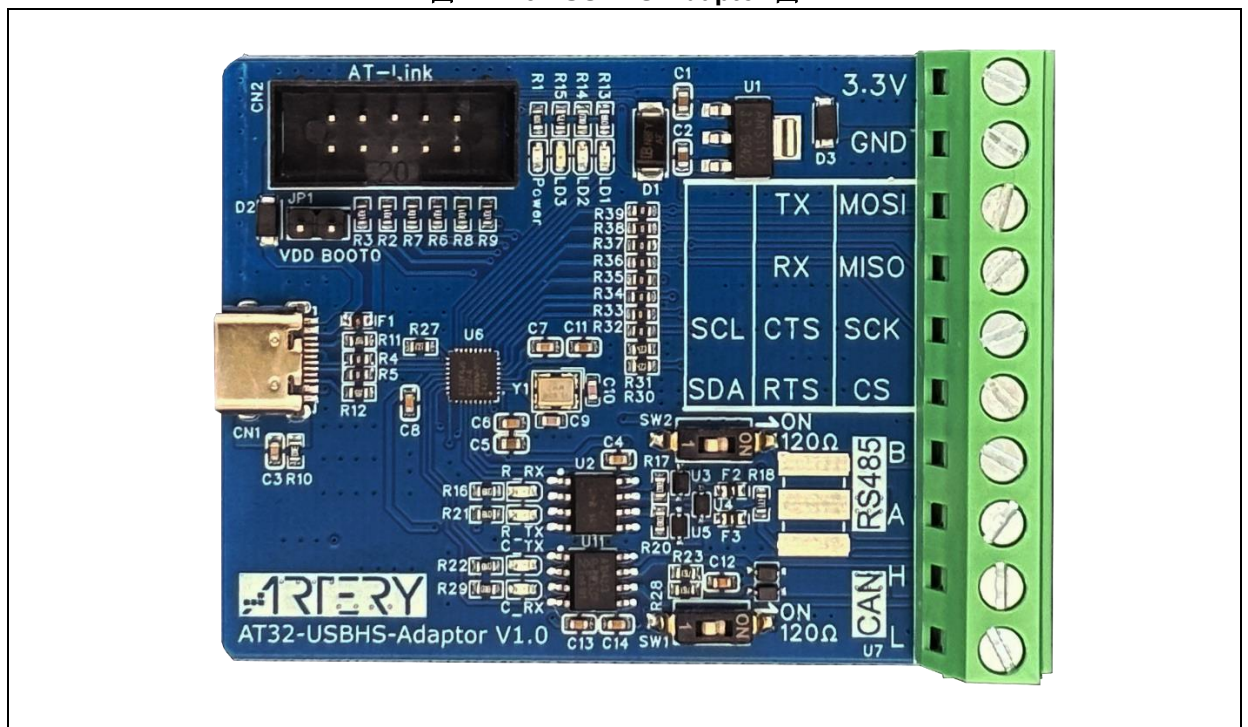
表 3 I²C 支持时钟频率

支持时钟频率	DwDTERate 参数
1 MHz	1000000
400 kHz	400000
200 kHz	200000
100 kHz	100000
50 kHz	50000
10 kHz	10000

1.1.3 硬件资源

1. USB 转接板 《AT32-USBHS-Adaptor》
2. USB 线缆

图 4 AT32-USBHS-Adaptor 图



3. 电源: 设备由 CN1 USB 接口 5V 供电, 并可对外供电 3.3V/350mA。

4. LED 灯

- 电源 Power: 红色 LED 指示 AT32 USBHS Adaptor 板子已供电。
- 模式指示灯: LD1, LD2, LD3 说明如下:

MODE	LD1	LD2	LD3
USB to USART	√		
USB to SPI		√	
USB to I2C			√
USB to CAN	√	√	
USB to RS485	√		√

1.1.4 软件资源

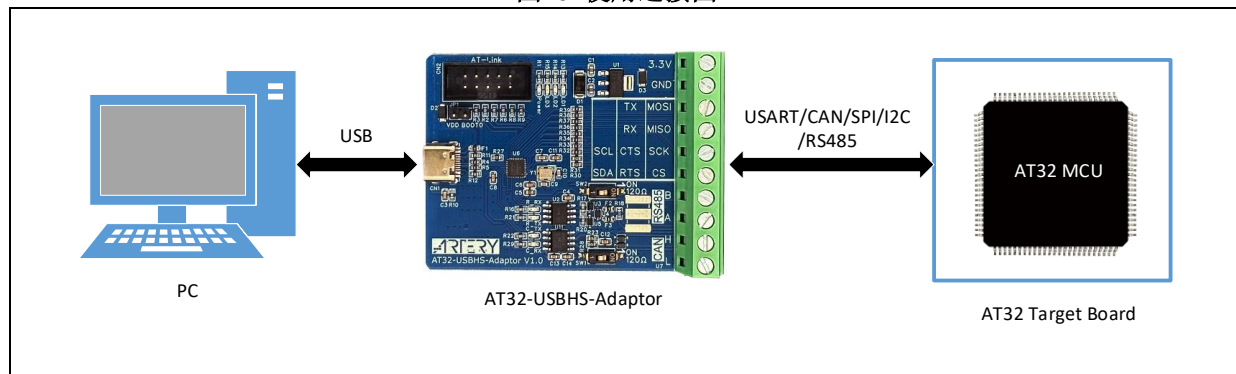
- 1) SourceCode\utilities\usb_multi_bridge

注: 示例工程基于 keil v5 建立, 若用户需要在其他编译环境上使用, 请参考 AT32F403A_407_Firmware_Library_V2.x.x\project\at_start_f403a\templates 中各种编译环境 (例如 IAR6/7/8, keil 4/5, eclipse_gcc) 进行对应修改即可。

1.2 USB 转接 demo 使用

- 1) 打开 usb_multi_bridge 工程源程序, 选择对应需要转接的外设, 通过调用 `usb_multi_bridge_set_type(USB_TO_USART)` 来设置当前转接外设, 编译后下载到转接板。
- 2) 连接 USB 线到 PC, 可在 PC 设备管理器看看到一个虚拟 COM 设备。可通过上位机虚拟串口工具调试。

图 5 使用连接图



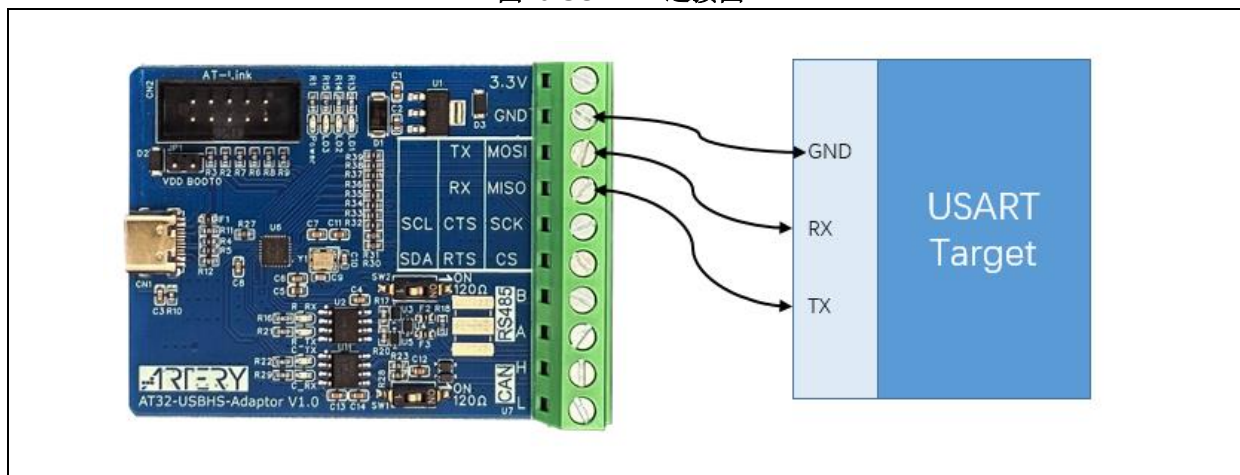
2 USB 转 USART

USB 转 USART 桥接，USB 端实现一个虚拟串口设备，转接板通过 USB 与 PC 上位机通信，然后通过 USART 与下位机进行通信，实现 USB 与 USART 数据的透传功能。注意转接板的 USART 波特率与下位机的 USART 波特率要配置相同。

2.1 USART 转接线连接

通过 AT32 USBHS Adaptor 进行转接，需要将转接板上的 RX, TX 和目标板上的 TX, RX 连接,如下是连接图：

图 6 USART 连接图



数据格式支持如下：

数据位数	停止位数	校验	波特率
7, 8, 9	1, 1.5, 2	奇校验, 偶校验, 无校验	1600bps~6Mbps

波特率和数据修改参照 1.1.2 节。

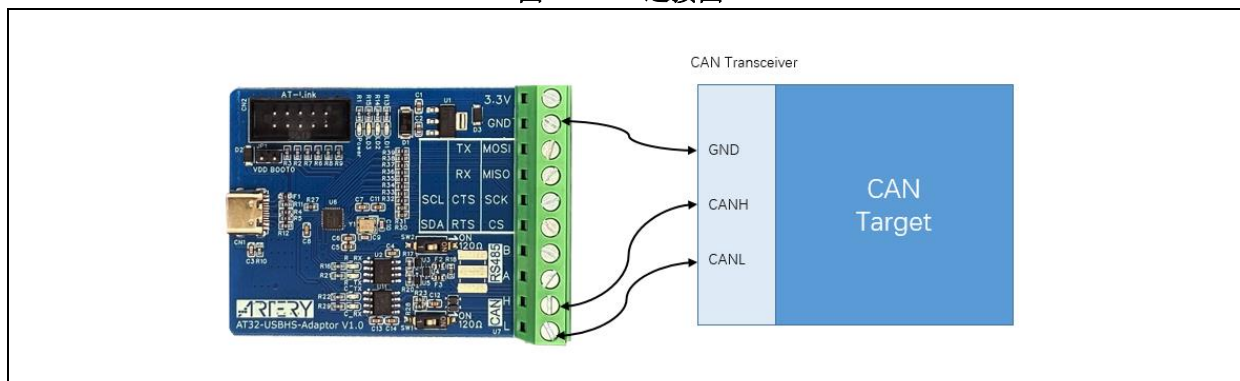
3 USB 转 CAN 桥接

USB 转 CAN 桥接，USB 端实现一个虚拟串口设备，转接板通过 USB 与 PC 上位机通信，然后通过 CAN 与下位机进行通信，实现 USB 与 CAN 数据的透传功能。注意转接板的 CAN 波特率与下位机的 CAN 波特率要配置相同，CAN 默认波特率 500 kbps。

3.1 CAN 转接线连接

通过 AT32 USBHS Adaptor 进行转接，需要将转接板上的 CANH，CANL 和目标板上的 CANH，CANL 连接,如下是连接图：

图 7 CAN 连接图



3.2 USB CAN 数据转发

USB 转 CAN 桥接按照 CAN 协议进行封装，封装格式：

帧 ID（4 字节）	ID 类型(1 字节)	帧类型（1 字节）	帧长（1 字节）	数据(n 字节， n<=8)
------------	-------------	-----------	----------	-------------------

帧 ID：支持 11 位的标准帧 ID，和 29 位扩展帧 ID，按照 LSB 顺序传输。

ID 类型：0 表示标准 ID（此时帧 ID 低 11 位有效），1 表示扩展 ID（此时帧 ID 低 29 位有效）

帧类型：0 表示数据帧，1 表示远程帧

帧长：按照 CAN2.0 的标准，帧长小于等于 8

数据：帧数据

CAN 波特率支持参考 1.1.2 章节。

USB 数据转到 CAN 总线：

USB 到 CAN 按照一包 USB 数据转成一帧 CAN 数据。

收到一包 USB 数据如：0x00 0x00 0x00 0x00 0x00 0x00 0x02 0x11 0x22

前 4 个字节 0x00 0x00 0x00 0x01：表示帧 ID（MSB）

第 5 个字节 0x00：表示是一个标准帧 ID，低 11 位位标准帧 ID

第 6 个字节 0x00：表示是一个数据帧

第 7 个字节 0x02：表示此数据帧长度

第 8,9 字节 0x11,0x22：表示具体帧数据

转换成 CAN 帧后：

标准帧 ID=0x0001

DLC=0x02

DATA0=0x11

DATA1=0x22

CAN 总线数据转到 USB:

CAN 同样按照一帧数据转换成一包 USB 数据:

收到一帧标准数据帧: ID=0x02, DLC=0x03, DATA0=0x11, DATA0=0x22 ,DATA0=0x33

转换成 USB 包之后: 0x00 0x00 0x00 0x02 0x00 0x00 0x03 0x11,0x22,0x33

前 4 个字节 0x00 0x00 0x00 0x02: 表示标准帧 ID

第 5 个字节 0x00: 表示是一个标准帧 ID

第 6 个字节 0x00: 表示是一个数据帧

第 7 个字节 0x03: 表示此数据帧长度

第 8,9,10 字节 0x11,0x22 0x33: 表示具体帧数据

4 USB 转 SPI 桥接

USB 转 SPI 桥接 USB 端实现一个虚拟串口设备，转接板通过 USB 与 PC 上位机通信，然后通过 SPI 与下位机进行通信，实现 USB 与 SPI 数据的透传功能，转接板的 SPI 作为主机。

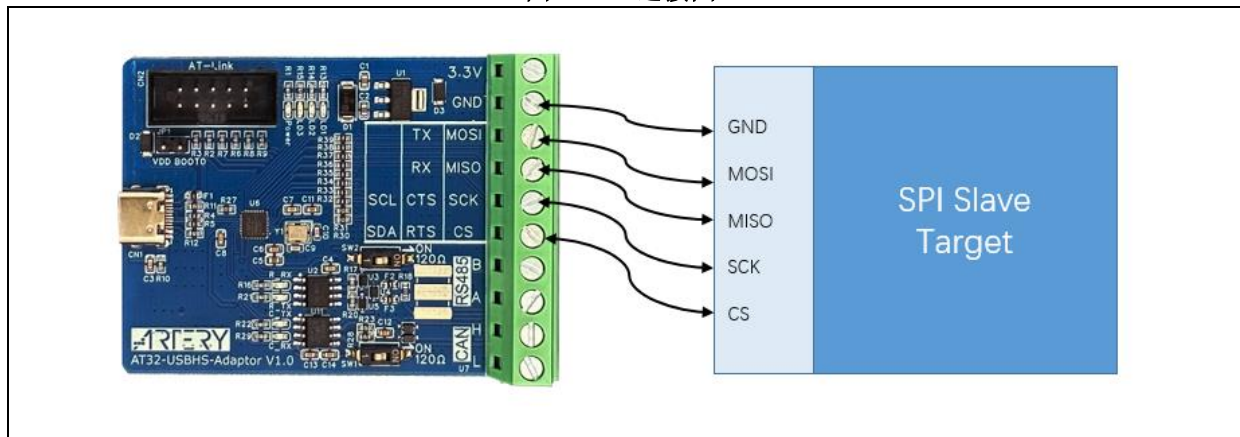
SPI 主机配置：

- SPI 主模式
- 全双工模式
- 8 bit MSB
- Polarity: CPOL High, CPHA low, NSS software

4.1 SPI 转接线连接

通过 AT32 USBHS Adaptor 进行转接，需要将转接板上的 MOSI,MISO,SCK,CS 和目标板上的 MOSI,MISO,SCK,CS 连接,如下是连接图：

图 8 SPI 连接图



4.2 USB SPI 数据转发

SPI 使用全双工进行收发数据，一包数据最大 512 字节，如果需要大于 512，则需要拆包发送。

上位机数据发送：

示例：上位机需要透传 8 个数据给目标板，对应数据：0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08

1. 上位机直接发送 8 个字节 0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08
2. 上位机需要再读取 8 个字节的 dummy data

上位机数据读取：

示例：上位机需要从目标板读取 8 个数据

1. 上位机发送 8 个字节的 dummy data
2. 上位机读取 8 个字节的的目标板数据

SPI 支持时钟频率参考 1.1.2 章节。

5 USB 转 I²C 桥接

USB 转 I²C 桥接 USB 端实现一个虚拟串口设备，转接板通过 USB 与 PC 上位机通信，然后通过 I²C 与下位机进行通信，实现 USB 与 I²C 数据的透传功能，转接板的 I²C 作为主机。I²C 主机地址 0x0C。

I²C 配置：

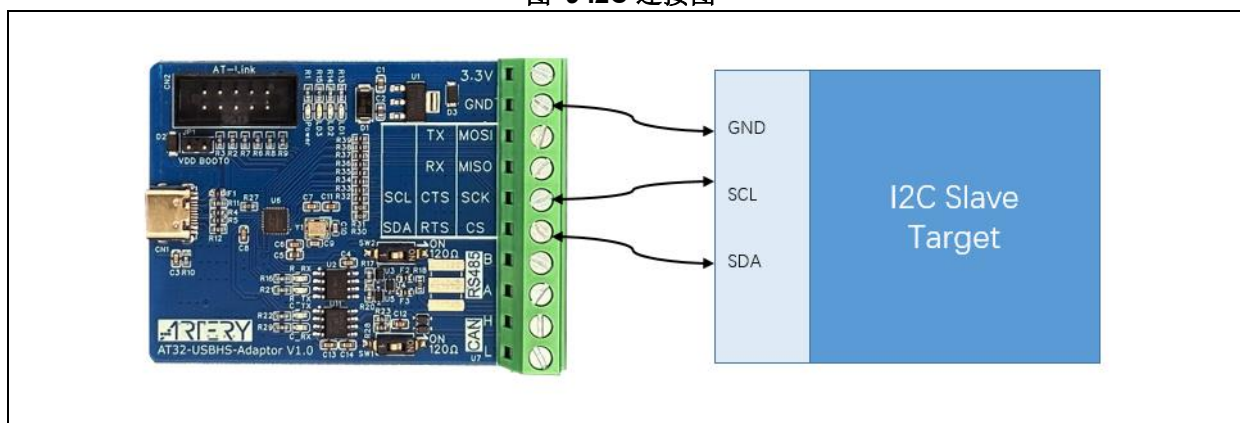
- I²C 主机模式
- 地址 0x0C

5.1 I²C 转接线连接

通过 AT32 USBHS Adaptor 进行转接，需要将转接板上的 SCL,SDA 和目标板上的 SCL,SDA 连接。AT32 USBHS Adaptor 默认安装了 SCL,SDA 线上拉电阻 4.7 k Ω (R30/R31)。

如下是连接图：

图 9 I²C 连接图



5.2 USB I²C 数据转发

使用 I²C 收发数据，一包数据最大 512 字节，如果需要大于 512，则需要拆包发送。

USB 转 I²C 数据封装格式：

方向 (1 byte)	地址 (2 byte)	透传数据长度 (2 Byte)	数据 (n Byte)
-------------	-------------	-----------------	-------------

方向：0x55 表示发送 I²C 数据到目标板，0xAA 表示从目标板读取 I²C 数据

地址：2 字节目标板地址(MSB)

数据长度：发送或者读取的数据长度，长度需要小于等于 507(MSB)

数据：发送或者读取的数据

I²C 支持时钟频率参考 1.1.2 章节。

上位机数据发送：

示例：上位机需要透传 8 个数据给地址为 0xA0 的目标板，对应数据:0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08

上位机发送数据包： 0x55 0x00 0xA0 0x00 0x08 0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08

上位机数据读取：

示例：上位机需要从地址为 0x2C 的目标板读取 8 个数据

1. 上位机发送 0xAA 0x00 0xA0 0x00 0x08 表示要读取 8 个数据

2. 上位机再读取 USB 返回数据进行解析：0xAA 0x00 0xA0 0x00 0x08 0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08

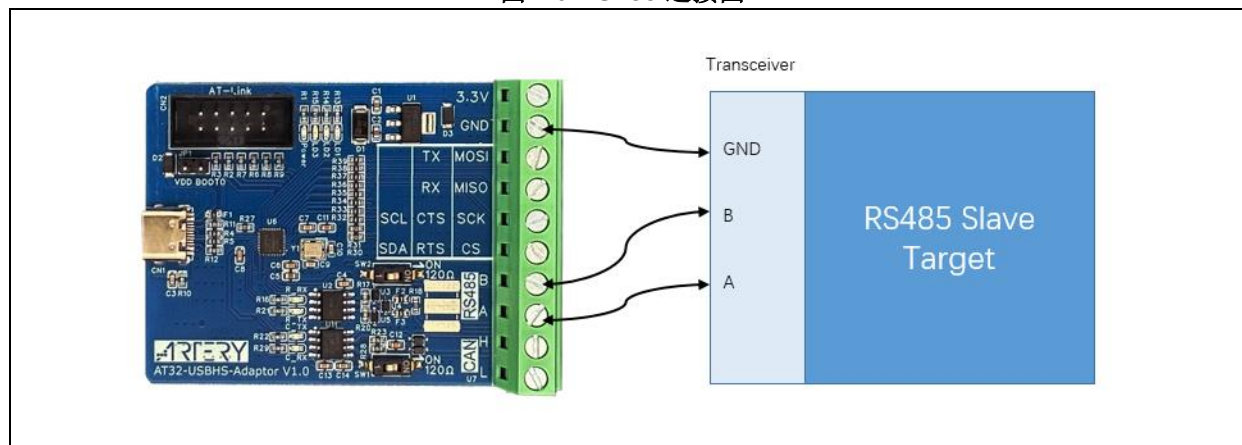
6 USB 转 RS485 桥接

USB 转 485 桥接 USB 端实现一个虚拟串口设备，转接板通过 USB 与 PC 上位机通信，然后通过 485 与下位机进行通信，实现 USB 与 485 数据的透传功能。

6.1 RS485 转接线连接

通过 AT32 USBHS Adaptor 进行转接，需要将转接板上的 RS485-A, RS485-B 和目标板上的 A,B 连接，如下是连接图：

图 10 RS485 连接图



7 版本历史

表 4 文档版本历史

日期	版本	变更
2024.07.18	2.0.0	初始版本
2024.08.20	2.0.1	修改文档描述 添加 LED 模式指示说明

重要通知 - 请仔细阅读

买方自行负责对本文所述雅特力产品和服务的选择和使用，雅特力概不承担与选择或使用本文所述雅特力产品和服务相关的任何责任。

无论之前是否有过任何形式的表示，本文档不以任何方式对任何知识产权进行任何明示或默示的授权或许可。如果本文档任何部分涉及任何第三方产品或服务，不应被视为雅特力授权使用此类第三方产品或服务，或许可其中的任何知识产权，或者被视为涉及以任何方式使用任何此类第三方产品或服务或其中任何知识产权的保证。

除非在雅特力的销售条款中另有说明，否则，雅特力对雅特力产品的使用和/或销售不做任何明示或默示的保证，包括但不限于有关适销性、适合特定用途（及其依据任何司法管辖区的法律的对应情况），或侵犯任何专利、版权或其他知识产权的默示保证。

雅特力产品并非设计或专门用于下列用途的产品：（A）对安全性有特别要求的应用，例如：生命支持、主动植入设备或对产品功能安全有要求的系统；（B）航空应用；（C）航天应用或航天环境；（D）武器，且/或（E）其他可能导致人身伤害、死亡及财产损害的应用。如果采购商擅自将其用于前述应用，即使采购商向雅特力发出了书面通知，风险及法律责任仍将由采购商单独承担，且采购商应独立负责在前述应用中满足所有法律和法规要求。

经销的雅特力产品如有不同于本文档中提出的声明和/或技术特点的规定，将立即导致雅特力针对本文所述雅特力产品或服务授予的任何保证失效，并且不应以任何形式造成或扩大雅特力的任何责任。

© 2024 雅特力科技 保留所有权利