CANUSB-I/II 工业级智能 CAN 接口卡使用说明书 V6.4

目 录

第一章 产品简介

1.1 产品概述

CANUSB-I/II 智能 CAN 接口卡兼容 USB1.1 和 USB2.0 总线，带有 1 路/2 路 CAN 接口的工业级智能型 CAN 数据接口卡。采用 CANUSB -I/II 智能 CAN 接口卡，PC 可以通过 USB 总线连接至 CAN 网络，构成实验室、工业控制、智能小区等 CAN 网络领域中数据处理、数据采集。

CANUSB-I/II 智能 CAN 接口卡是 CAN 产品开发、CAN 数据分析的强大工具；同时，具有体积小、即插即用等特点，也是便携式系统用户的最佳选择。

CANUSB-I/II 接口卡上自带光电隔离模块，隔离电压达 2500V，使 CANUSB-I/II 接口卡避免由于地环流的损坏，增强系统在恶劣环境中使用的可靠性。

CANUSB-I/II 智能 CAN 接口卡配有可在 **Win9X/Me、Win2000/XP 、Server 2003、Vista** 下工作的驱动程序，并提供 **VB, VB2003, VC, C++Builder, Delphi, Labview,**下的应用例程。

1.2 性能指标

系统性能：处理器 48MIPS，USB FIFO 1KByte；
帧流量： **业界最优性能，达到CAN的理论极限，实测每秒钟流量超过 6500 帧**
传输方式：CAN2.0A 和 CAN2.0B 协议,USB 接口兼容 USB1.1 和 USB2.0 协议；
通道数目：支持 1-2 路 CAN 控制器，每路均可单独控制；
传输介质：屏蔽或非屏蔽双绞线；
传输速率：CAN 控制器波特率在 5Kbps～1Mbps 之间可选；
通讯接口：CAN-bus 接口采用光电隔离、DC-DC 电源隔离，隔离模块绝缘电压：2500V；
总线长度及节点数：单路总线上最多可接 110 个节点，最长通讯距离 10 公里；
供电形式：可以直接使用 USB 总线电源，无需外部电源；
占用资源：即插即用，资源自动分配；
尺寸： 112mm*84mm*28mm
工作温度：-20℃～+70℃
存储温度：-55℃～+85℃

1.3 应用领域

CAN-bus 产品开发；
CAN-bus 数据分析；
CAN-bus 主从式网络；
CAN-bus 教学应用；
CAN-bus 网关、网桥；
CAN-bus 工业自动化控制系统；
智能楼宇控制、数据广播系统等 CAN-bus 应用系统；
不同 CAN-bus 网络间的数据转换；

1.4 订购信息

型号	工作温度	接口
CANUSB-I	-20℃～+70℃	OPEN5
CANUSB-II	-20℃～+70℃	OPEN5

1.5 产品销售清单

[1] CANUSB-I/II 智能接口转换卡；
[2] USB 电缆线一根；
[3] 光盘 1 张（包括 PC 驱动（Dll, Lib）、接口函数、用户手册、VB, VB2003，VC, C++Builder, Dephi, Labview 例程等）；

1.6 技术支持与服务

一年免费维修、升级，终身维修。
支持邮箱：embededperfect@163.com
支持网站：http://www.embedded-soc.com

第二章 设备安装

2.1 驱动程序安装

CANUSB-I/II智能CAN接口卡使用USB直接供电并提供智能驱动安装包，安装步骤如下：

[1] 点击产品光盘的“\CANUSB\Drivers”目录下安装包安装驱动。

[2] 将CANUSB-I/II智能CAN接口卡通过USB电缆连接倒计算机，提示发现新硬件，选择自动安装软件即可。

2.2 智能CAN接口卡硬件接口描述

CANUSB-I/II智能CAN接口卡集成2 路CAN 通道，每一路通道都是独立的，可以用于连接一个CAN-bus 网络或者CAN-bus 接口的设备。CANUSB-I/II智能CAN接口卡接口布局如下：

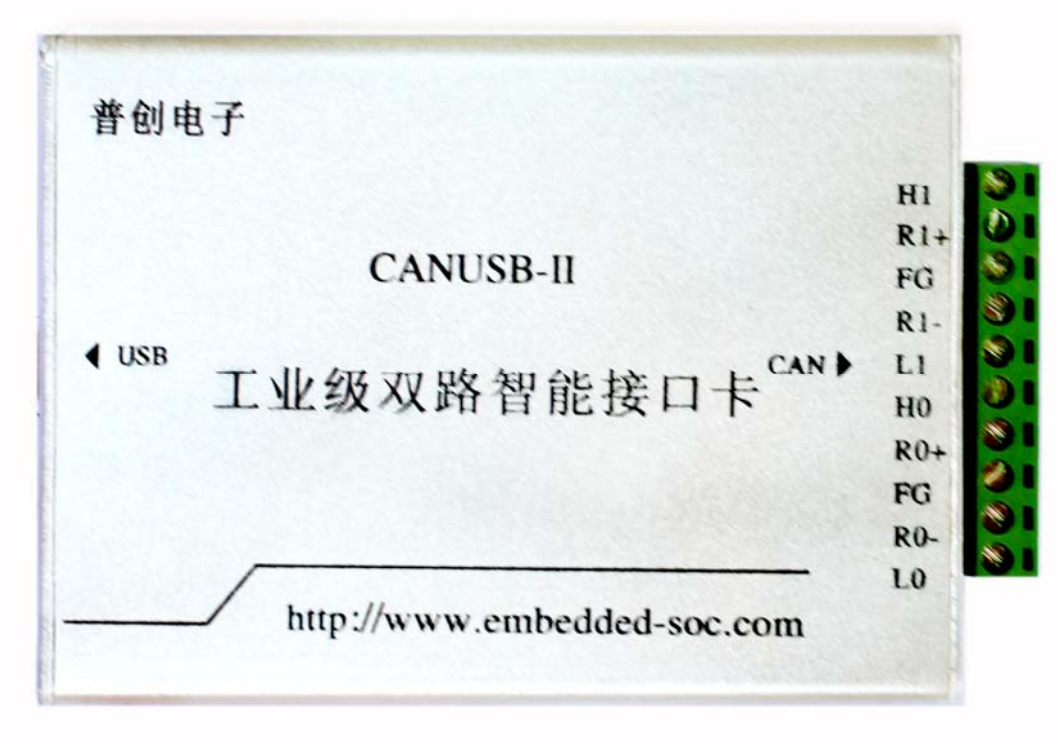


图 1 CANUSB-I/II智能CAN接口卡外围端子

2 路CAN-bus 通道由1 个10 Pin接线端子引出。接线端子的引脚详细定义如表格 1 所示。

表格 1 CANUSB-I/II 接口卡的CAN-bus 信号分配(CANUSB-I型的CAN1接口悬空)

引脚	端口	名称	功能
1	CAN0	CANL0	CANL0 信号线
2		R0－	终端电阻（内部连接到CANL0）
3		FG	屏蔽线 (FG)
4		R0＋	终端电阻（内部连接到CANH0）
5		CANH0	CANH0 信号线
6	CAN1	CANL1	CANL1 信号线
7		R1－	终端电阻（内部连接到CANL1）
8		FG	屏蔽线 (FG)
9		R1＋	终端电阻（内部连接到CANH1）
10		CANH1	CANH1 信号线

2.3 系统连接

CANUSB-I/II接口卡和CAN-bus 总线连接的时候，仅需要将CANL 连CANL，CANH 连CANH 信号。CAN-bus 网络采用直线拓扑结构，总线的2个终端需要安装120Ω 的终端电阻；如果节点数目大于2，中间节点不需要安装120Ω 的终端电阻。对于分支连接，其长度不应超过3米。CAN-bus 总线的连接见图 2 所示。

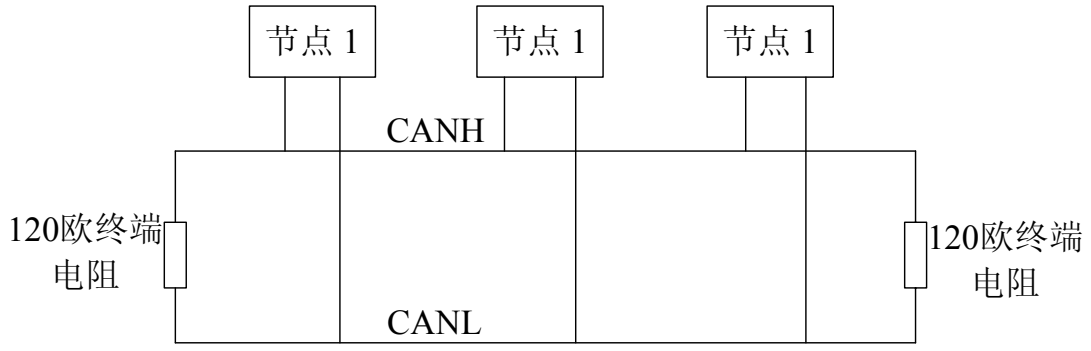


图 2 CAN-bus 网络的拓扑结构

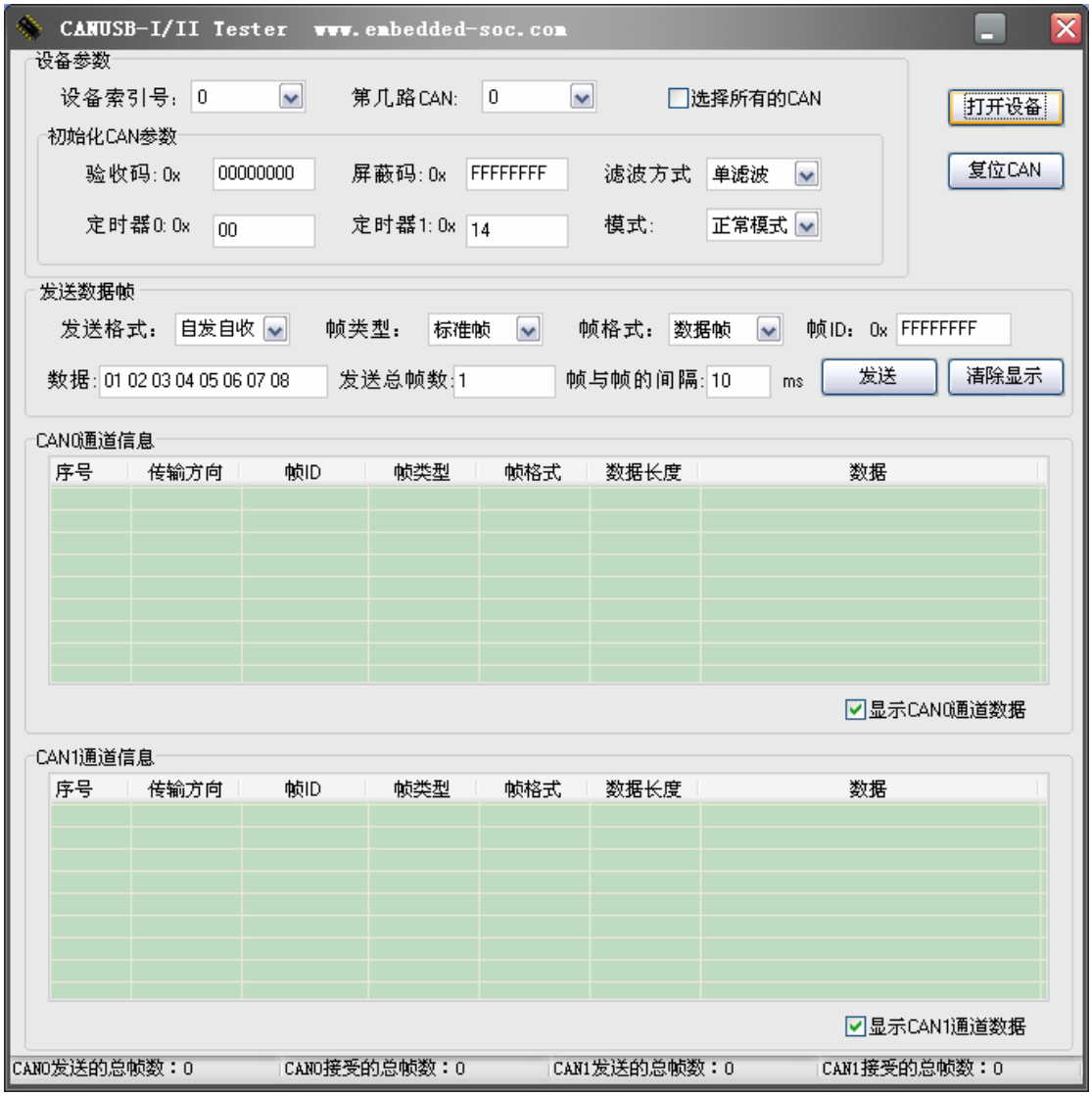
2.4 总线终端电阻

为了增强CAN 通讯的可靠性，CAN 总线网络的两个端点通常要加入终端匹配电阻，如图 2 所示。终端匹配电阻的值由传输电缆的特性阻抗所决定。例如双绞线的特性阻抗为120Ω，则总线上的两个端点也应120Ω终端电阻。当CANUSB-I/II位于CAN-bus 网络的一个端点上时，需要在外部端子上安装120Ω终端电阻，即在“R－”引脚和“R＋”引脚接入终端电阻。

2.5 产品使用

2.5.1 CANUSB-I/II接口卡 测试软件概述

该测试软件使用极为方便，点击连接按钮后就可以发送和接受数据了，发送和接受的数据和状态在下面的信息提示框中有很清楚的显示。



2.5.2 CANUSB-I/II接口卡 测试软件详细介绍

[1] 在设备参数部分用户可以选择打开第几路CAN通道，当用户点击了选择所有的CAN时将同时打开二路CAN。(注意CANUSB-I支持CAN0通道一个通道，CANUSB-II才支持CAN0和CAN1二个通道)。

[2] 在设备参数部分用户可以设置验收码，屏蔽码，滤波方式，以及波特率和工作模式。

[3] 点击连接按钮后，设备就开始工作。

[4] 在发送数据帧部分组合了各种可能情况由用户来进行测试。帧ID根据11位还是29位ID从低位截取。（注意从那个CAN通道发送出去由设备参数部分的用户选择的第

几路CAN决定。）

[5] 当插入多个设备时，设备索引号选择对应的设备，同时可以调用库函数获得它们的序列号

第三章 用户编程

用户如果只是利用CANUSB-I/II接口卡进行CAN总线通信测试，可以直接利用随机提供的测试软件，进行收发数据的测试。如果用户打算编写自己产品的软件程序。请认真阅读以下说明。光盘中附带VB、VB2003、VC、C++Builder、Delphi、Labview的完整例程。

3.1函数库数据结构定义

3.1.1 初始化CAN的数据类型

```
typedef struct _INIT_CONFIG{
    DWORD   AccCode;
    DWORD   AccMask;
    DWORD   Reserved;
    UCHAR   Filter;
    UCHAR   Timing0;
    UCHAR   Timing1;
    UCHAR   Mode;
}VCI_INIT_CONFIG,*P_VCI_INIT_CONFIG;
```

AccCode 验收码。

AccMask 屏蔽码。

Reserved 保留。

Filter 滤波方式。

Timing0 定时器0（BTR0）

Timing1 定时器0（BTR0）

Mode 模式。

备注 Timing0 和 Timing1 用来设置 CAN 波特率，几种常见的波特率设置如下：

CAN 波特率	定时器 0	定时器 1
5Kbps	0xBF	0xFF
10Kbps	0x31	0x1C
20Kbps	0x18	0x1C
40Kbps	0x87	0xFF
50Kbps	0x09	0x1C
80Kbps	0x83	0Xff
100Kbps	0x04	0x1C
125Kbps	0x03	0x1C
200Kbps	0x81	0xFA
250Kbps	0x01	0x1C
400Kbps	0x80	0xFA
500Kbps	0x00	0x1C
666Kbps	0x80	0xB6
800Kbps	0x00	0x16

1000Kbps	0x00	0x14

3.1.2 CAN信息帧的数据类型

```
typedef  struct  _VCI_CAN_OBJ{
    DWORD    ID;
    BYTE     SendType;
    BYTE     ExternFlag;
    BYTE     RemoteFlag;
    BYTE     DataLen;
    BYTE     Data[8];
}VCI_CAN_OBJ,*P_VCI_CAN_OBJ;
```

ID 报文ID。

SendType 发送帧类型，=0 时为正常发送，=1 时为自发自收 只有在此帧为发送帧时有意义。

RemoteFlag 是否是远程帧。

ExternFlag 是否是扩展帧。

DataLen 数据长度(<=8)，即Data 的长度。

Data 报文的数据。

3.2接口函数说明

[1] 打开设备

```
BOOL    __stdcall    VCI_OpenDevice(DWORD DevIndex);
```

DevIndex 设备索引号，有一个设备时索引号为0，有两个可以为0或1。

返回值 为1 表示操作成功，0 表示操作失败。

[2] 关闭设备

```
BOOL    __stdcall    VCI_CloseDevice(DWORD DevIndex);
```

DevIndex 设备索引号，有一个设备时索引号为0，有两个可以为0或1。

返回值 为1 表示操作成功，0 表示操作失败。

[3] 初始化CAN

```
BOOL    _stdcall    VCI_InitCan(DWORD DevIndex,DWORD CANIndex,
P_VCI_INIT_CONFIG InitConfig);
```

DevIndex 设备索引号，有一个设备时索引号为0，有两个可以为0或1。

CANIndex 第几路CAN。

InitConfig 初始化参数结构。

InitConfig->AccCode	AccCode 对应SJA1000 中的四个寄存器ACR0，ACR1，ACR2，ACR3，其中高字节对应ACR0，低字节对应ACR3；AccMask 对应SJA1000 中的四个寄存器AMR0，AMR1，AMR2，AMR3，其中高字节对应AMR0，低字节对应AMR3 。
InitConfig->AccMask	
InitConfig->Reserved	保留
InitConfig->Filter	滤波方式，0 表示单滤波，1 表示双滤波
InitConfig->Timing0	定时器0
InitConfig->Timing0	定时器1

InitConfig->Mode	模式，0 表示正常模式，1 表示只听模式

返回值 为1 表示操作成功，0 表示操作失败。

[4] 复位CAN设备

BOOL __stdcall VCI_ResetCan(DWORD DevIndex ， DWORD CANIndex);

DevIndex 设备索引号，有一个设备时索引号为0，有两个可以为0或1。

CANIndex 第几路CAN。

返回值 为1 表示操作成功，0 表示操作失败。

[5] 发送一帧数据

BOOL __stdcall VCI_Transmit(DWORD DevIndex ， DWORD CANIndex, P_VCI_CAN_OBJ *pSend);

DevIndex 设备索引号，有一个设备时索引号为0，有两个可以为0或1。

CANIndex 第几路CAN。

pSend 指向信息帧结构体，其参数介绍请看函数库数据结构部分。

返回值 为1 表示操作成功，0 表示操作失败。

[6] 接受数据

DWORD __stdcall VCI_Receive(DWORD DevIndex ， DWORD CANIndex，PVCI_CAN_OBJ pReceive ， DWORD Len ， DWORD WaitTime);

DevIndex 设备索引号，有一个设备时索引号为0，有两个可以为0或1。

pReceive 用来接收的数据帧结构体数组的首指针。

Len 读取多少帧的数据。

WaitTime 以毫秒为单位。

返回值 返回实际读取到的帧数。如果返回值为0，则表示没有读到数据。

[7] 获取缓冲区中尚未读取的帧数

DWORD __stdcall VCI_GetReceiveNum(DWORD DevIndex,DWORD CANIndex);

DevIndex 设备索引号，有一个设备时索引号为0，有两个可以为0或1。

CANIndex 第几路CAN。

返回值 返回缓冲区中尚未读取的帧数。

[8] 清空缓冲区中的数据

BOOL __stdcall VCI_ClearBuffer(DWORD DevIndex,DWORD CANIndex);

DevIndex 设备索引号，有一个设备时索引号为0，有两个可以为0或1。

CANIndex 第几路CAN。

返回值 为1 表示操作成功，0 表示操作失败。

[9] 读取序列号

BOOL _stdcall VCI_ReadDevSn(DWORD DevIndex, PCHAR DevSn);

DevIndex 设备索引号，有一个设备时索引号为0，有两个可以为0或1。

DevSn 序列号。

返回值 为1 表示操作成功，0 表示操作失败。

3.3 接口库函数使用方法

首先，把库函数文件都放在工作目录下。总共有五个文件CAN_TO_USB.h，CAN_TO_USB.lib（For VC），CAN_TO_USBbc.lib(For BCB)，SiUSBXp.dll，CAN_TO_USB.dll

上述库文件支持采用VB, VC, C++Builder,Delphi等工具进行编程，当用户采用动态链接时不需要用使用CAN_TO_USB.lib（For VC），CAN_TO_USBbc.lib(For BCB)。

3.3.1 VC 调用动态库的方法（静态链接）

（1）在.CPP 中包含CAN_TO_USB.h头文件；
（2）在工程文件中加入CAN_TO_USB.lib 文件。

3.3.2 C++ Builder 调用动态库的方法（静态链接）

（1）在.CPP 中包含CAN_TO_USB.h 头文件；
（2）在工程文件中加入CAN_TO_USBbc.lib文件。

3.3.3 VB 调用动态库的方法

通过以下方法进行声明后就可以调用了。

语法:

```
[Public | Private] Declare Function name Lib "libname" [Alias "aliasname"] [([arglist])] [As type]
```

Declare 语句的语法包含下面部分：

Public（可选）

用于声明在所有模块中的所有过程都可以使用的函数。

Private（可选）

用于声明只能在包含该声明的模块中使用的函数。

Name（必选）

任何合法的函数名。动态链接库的入口处（entry points）区分大小写。

Libname（必选）

包含所声明的函数动态链接库名或代码资源名。

Alias（可选）

表示将被调用的函数在动态链接库 (DLL) 中还有另外的名称。当外部函数名与某个函数重名时，就可以使用这个参数。当动态链接库的函数与同一范围内的公用变量、常数或任何其它过程的名称相同时，也可以使用 Alias。如果该动态链接库函数中的某个字符不符合动态链接库的命名约定时，也可以使用 Alias。

Aliasname（可选）

动态链接库。如果首字符不是数字符号 (#)，则 aliasname 是动态链接库中该函数入口处的名称。如果首字符是 (#)，则随后的字符必须指定该函数入口处的顺序号。

Arglist（可选）

代表调用该函数时需要传递参数的变量表。

Type（可选）

Function 返回值的数据类型；可以是 Byte、Boolean、Integer、Long、Currency、Single、Double、

Decimal（目前尚不支持）、Date、String（只支持变长）或 Variant，用户定义类型，或对象类型。

arglist 参数的语法如下：

[Optional] [ByVal | ByRef] [ParamArray] varname[( )] [As type]

部分描述：

Optional（可选）

表示参数不是必需的。如果使用该选项，则 arglist 中的后续参数都必需是可选的，而且必须都使用Optional 关键字声明。如果使用了 ParamArray，则任何参数都不能使用 Optional。

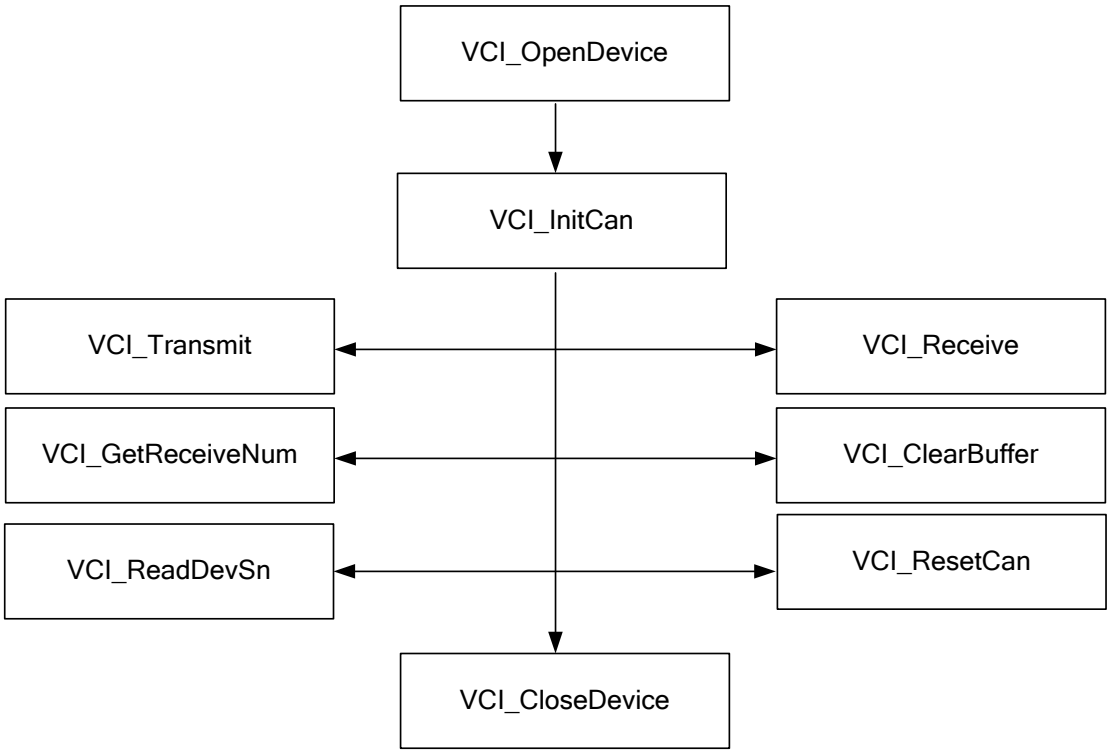
ByVal（可选）

表示该参数按值传递。

ByRef（可选）

表示该参数按地址传递。

3.4 接口库函数使用流程



附录

关于波特率寄存器BTR, 验收滤波器ACR，屏蔽器AMR 等更详细的资料，可参考SJA1000 独立CAN 控制器的芯片手册。

1. CAN2.0B标准帧

CAN 标准帧信息为11个字节，包括两部分：信息和数据部分。前3个字节为信息部分。

	7	6	5	4	3	2	1	0
字节1	FF	RTR	X	X	DLC （数据长度）			
字节2	（报文识别码）					ID.10-ID.3		
字节3	ID.2-ID.0			X	X	X	X	X
字节4	数据1							
字节5	数据2							
字节6	数据3							
字节7	数据4							
字节8	数据5							
字节9	数据6							
字节10	数据7							
字节11	数据8							

字节1 为帧信息。第7 位（FF）表示帧格式，在标准帧中，FF＝0；第6 位（RTR ）表示帧的类型，RTR=0 表示为数据帧，RTR=1 表示为远程帧；DLC 表示在数据帧时实际的数据长度。字节2、3 为报文识别码，11 位有效。字节4～11 为数据帧的实际数据，远程帧时无效。

2. CAN2.0B扩展帧

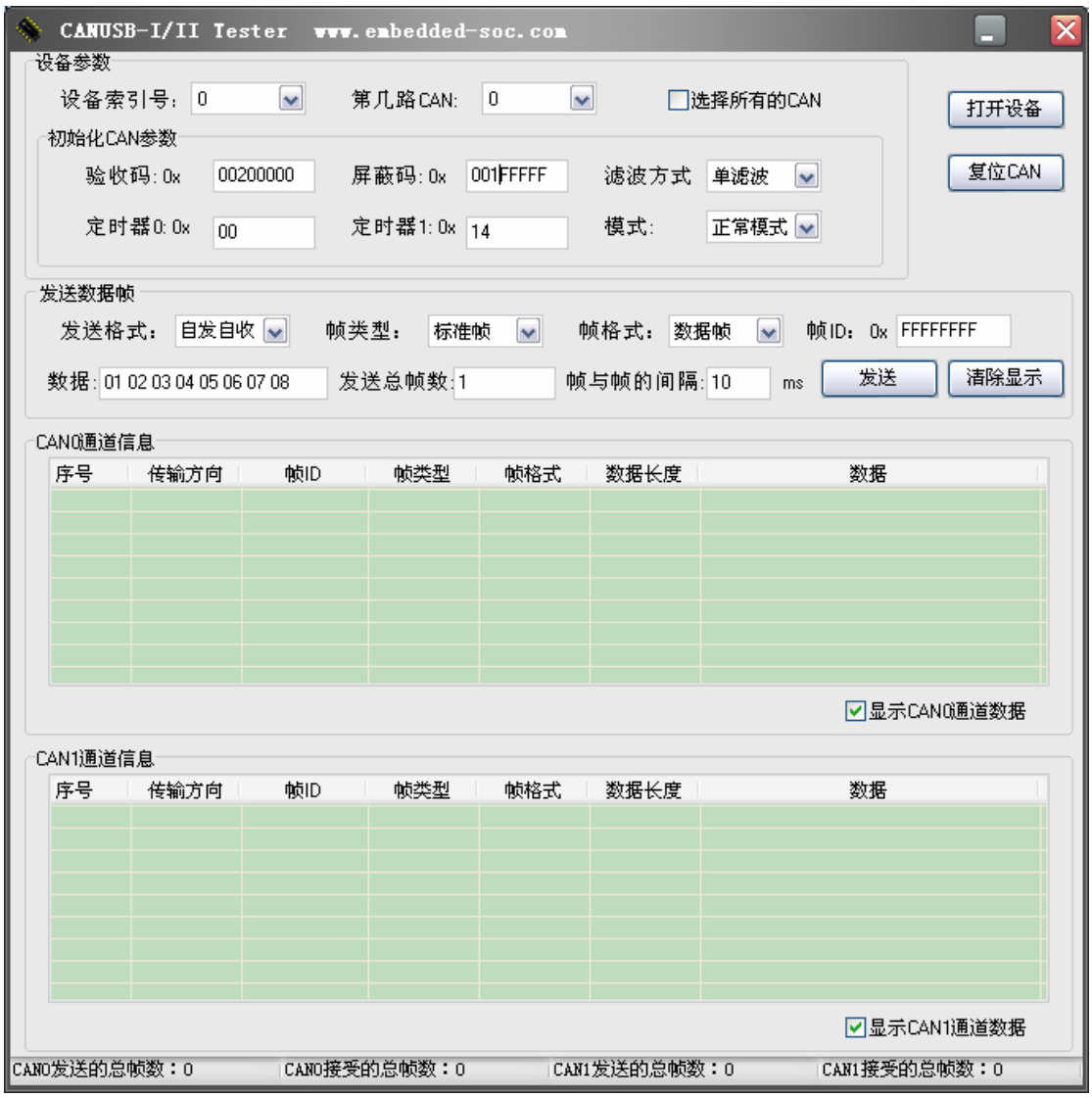
扩展帧CAN 扩展帧信息为13 个字节，包括两部分，信息和数据部分。前5个字节为信息部分。

	7	6	5	4	3	2	1	0
字节1	FF	RTR	X	X	DLC （数据长度）			
字节2	（报文识别码）					ID.28-ID.21		
字节3	ID.20-ID.13							
字节4	ID.12-ID.5							
字节5	ID.4-ID.0					X	X	X
字节6	数据1							
字节7	数据2							
字节8	数据3							
字节9	数据4							
字节10	数据5							
字节11	数据6							
字节12	数据7							
字节13	数据8							

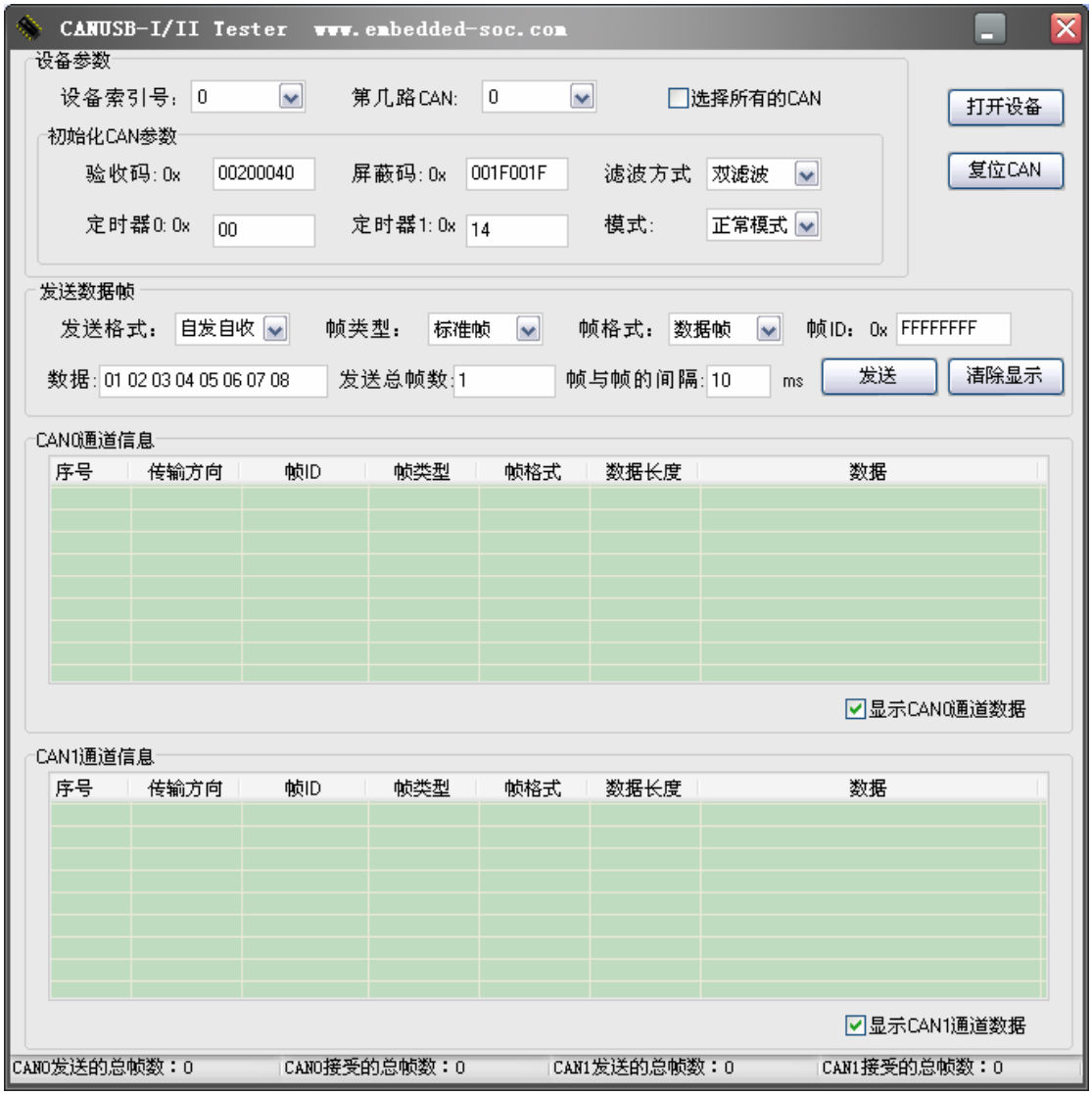
字节1 为帧信息。第7 位（FF）表示帧格式，在扩展帧中，FF＝1；第6 位（RTR ）表示帧的类型，RTR=0 表示为数据帧，RTR=1 表示为远程帧；DLC 表示在数据帧时实际

的数据长度。字节2～5 为报文识别码，其高29 位有效。字节6～13 为数据帧的实际数据，远程帧时无效。

3. 验收和屏蔽寄存器设置示例



只能收到 ID 为 1 的设置（单滤波，标准帧）



只能收到 ID 为 1 或者 2 的设置（双滤波，标准帧）