USBCAN-I Pro

工业级高性能USB接口CAN卡

用户手册

文档版本 4.30（2020/07/18）

修订历史

版本	日期	原因
V1.00	2013/06/16	创建文档
V2.01	2013/12/20	修正设备工作参数
V3.01	2014/10/22	添加部分参数
V3.50	2016/07/16	添加 OBDII 功能
V4.01	2017/01/13	添加 CANopen 功能
V4.20	2018/07/18	调整文档结构
V4.30	2020/07/18	调整文档结构

目 录

1. 功能简介.....3
1.1 功能概述.....3
1.2 性能特点.....3
1.3 典型应用.....4
2. 设备安装.....5
2.1 设备尺寸.....5
2.2 接口定义及功能.....5
2.3 驱动及软件安装.....6
3. 设备使用.....8
3.1 与 PC 连接.....8
3.2 与 CAN-bus 连接.....8
3.3 CAN 总线终端电阻.....9
3.4 系统状态指示灯.....9
4. 软件使用.....11
4.1 软件启动.....11
4.2 自动识别波特率.....13
4.3 数据接收相关功能.....15
4.4 数据发送相关功能.....20
4.5 总线诊断功能.....23
4.6 汽车电子解析 OBD II 界面.....24
4.7. CANopen 主站功能说明.....29
5. Linux 系统使用说明.....41
6. 二次开发.....45
7. 技术规格.....46
8. 常见问题.....47
9. 免责声明.....50
销售与服务.....51

1. 功能简介

1.1 功能概述

USBCAN-I Pro 是集成 1 路 CAN 接口的高性能型 CAN-bus 总线通讯接口卡。该型号 CAN 卡可兼容 USB2.0 总线全速规范，采用 USBCAN-I Pro 高性能 CAN 接口卡，PC 可以通过 USB 接口快速连接至 CAN-bus 网络，构成现场总线实验室、工业控制、智能小区、汽车电子网络等 CAN-bus 网络领域中数据处理、数据采集的 CAN-bus 网络控制节点。

USBCAN-I Pro 高性能 CAN 接口卡是 CAN-bus 产品开发、CAN-bus 数据分析的强大工具；同时具有体积小巧、即插即用等特点，也是便携式系统用户的最佳选择。USBCAN-I Pro 接口卡上自带 USB 接口，集成 CAN 接口电气隔离保护模块，使其避免由于瞬间过流/过压而对设备造成损坏，增强系统在恶劣环境中使用的可靠性。

USBCAN-I Pro 高性能 CAN 接口卡支持 Windows XP/Win7/Win8/Win10 等 32 位/64 位操作系统，还可支持 Linux 操作系统。我公司为用户提供统一的应用程序编程接口和完整的应用示范代码，含 VC、VB、.Net、Delphi、Labview 和 C++Builder 等开发例程示范，方便用户进行应用程序开发。

USBCAN-I Pro 接口卡可使用我公司自主开发的 ECANTools 通用测试软件，可执行 CAN-bus 报文的收发和监测等功能。

1.2 性能特点

- PC接口符合USB2.0全速规范，兼容USB1.1及USB3.0；
- 集成1路CAN-bus接口，使用DB9接线方式；
- 支持CAN2.0A和CAN2.0B帧格式，符合ISO/DIS 11898规范；
- CAN-bus通讯波特率在5Kbps~1Mbps之间任意可编程；
- 使用USB总线电源供电（DC+5V，130mA）；
- CAN-bus接口采用电气隔离，隔离模块绝缘电压：DC 1500V；
- 最高接收数据流量：14000 fps；
- CAN端接收报文时间戳精度可达1us；

- 支持WinXP、Win7、Win8、Win10等Windows操作系统；
- 支持Linux操作系统；
- 支持ECANTools测试软件；
- 工作温度范围：-20℃~+70℃；

1.3 典型应用

- CAN-bus网络诊断与测试
- 汽车电子应用
- 电力通讯网络
- 工业控制设备
- 高速、大数据量通讯

2. 设备安装

本章介绍了 USB-CAN 接口卡与电脑连接的方法及初次使用电脑连接 USB-CAN 接口卡时的注意事项。

2.1 设备尺寸

设备外形尺寸：（长，含 DB9 接口）87mm ＊（宽）51mm ＊（高）21mm，其示意图如图 2.1 所示。

图 2.1 USBCAN-I Pro 外形尺寸

2.2 接口定义及功能

USBCAN-I Pro 接口卡集成 1 路 USB 接口及 1 路标准 CAN-bus 接口，由 1 个 DB9 接口引出，可以用于连接 1 个 CAN-bus 网络或者 CAN-bus 接口的设备。

USBCAN-I Pro 各接口位置及定义如图 2.2、图 2.3 及表 2.1、表 2.2 所示。

图 2.2 USB 接口位置

端口	名称	功能
USB	USB	USBCAN 供电，与电脑连接

表 2.1 USB 接口定义

图 2.3 CAN-bus 接口位置

端口	名称	功能
CAN/DB9	1	+5V 可选，默认 NC （可定制 DC+5V）
	2	CAN_L 信号线
	3	CAN_GND
	4	无连接
	5	无连接
	6	CAN_GND
	7	CAN_H 信号线
	8	无连接
	9	+5V 可选，默认 NC （可定制 DC+5V）

表 2.2 USBCAN-I Pro 接口卡的 CAN-bus 信号分配

2.3 驱动及软件安装

驱动及软件安装之前，请用户确保自己登陆 Windows 的账户是管理员，或用户账户有安装驱动及软件的相关权限，否则可能导致安装失败。

确认 Windows 账户权限的方法：控制面板-用户账户。

2.3.1 驱动及软件安装

用户可以通过直接安装 ECAN Tools 软件的方式，完成驱动及软件的打包安装。如需手动安装驱动，请进入光盘中的“驱动 driver”文件夹，选择与系统对应（32/64 位）的安装文件（DriverSetup.exe/DriverSetup64.exe）进行手动安装。

2.3.2 驱动及软件卸载

用户可以通过运行上方 DriverSetup.exe/DriverSetup64.exe 后点击“卸载”按钮卸载安装好的设备驱动。

用户可通过“添加/删除程序”中找到 ECANTools 软件对其进行卸载。

设备使用

3.1 与 PC 连接

USBCAN-I Pro 接口卡具有即插即用的特点，因此用户可以使用 PC 机的 USB 接口直接向 USBCAN-I Pro 接口卡供电；若 USB 供电不足，则需选用外部电源供电方式（仅 USBCAN-II Pro 支持外部供电方式）。

3.1.1 USB 总线供电模式

USB 总线供电模式适合于大多数应用场合，例如，当 USBCAN-I Pro 接口卡是 USB 端口连接的唯一设备时。

将 PC 与 USBCAN-I Pro 接口卡通过随机附带的 USB 电缆直接连接，由 USB 电缆向 USBCAN-I Pro 接口卡提供+5V 电源；此时，指示灯 PWR、SYS 点亮，表示设备工作正常且处于待连接状态。

3.2 与 CAN-bus 连接

USBCAN-I Pro 接入 CAN 总线连接方式为将 CAN_H 连 CAN_H，CAN_L 连 CAN_L 即可建立通信。

CAN-bus 网络采用直线拓扑结构，总线最远的 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 3.1 所示。

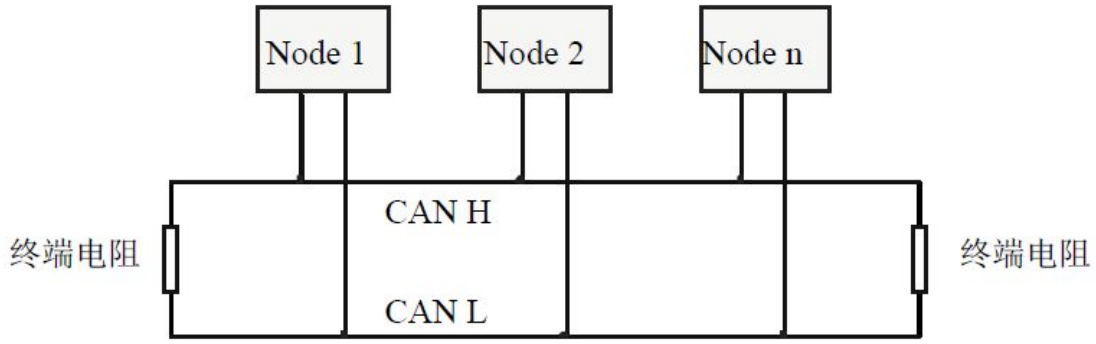


图 3.1 CAN-bus 网络的拓扑结构

注意：CAN-bus 电缆可以使用普通双绞线、屏蔽双绞线。理论最大通信距离主要取决于总线波特率，最大总线长度和波特率关系详见表 3.1。若通讯距离超过 1Km，应保证线的截面积大于Φ1.0mm2，具体规格应根据距离而定，常规是随距离的加长而适当加大。

波特率	总线长度
1 Mbit/s	25m
500 kbit/s	100m
250 kbit/s	250m
125 kbit/s	500m
50 kbit/s	1.0km
20 kbit/s	2.5km
10 kbit/s	5.0km
5 kbit/s	13km

表 3.1 波特率与最大总线长度参照表

3.3 CAN 总线终端电阻

为了增强 CAN 通讯的可靠性，消除 CAN 总线终端信号反射干扰，CAN 总线网络最远的两个端点通常要加入终端匹配电阻，如图 3.2 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集成 120Ω终端电阻。

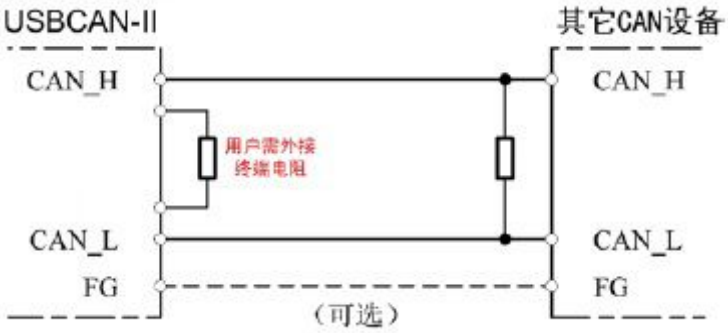


图 3.2 USBCAN-I Pro 与其他 CAN 节点设备连接

注意：USBCAN-I Pro 的 DB9 接口版本在 CAN 总线内部已集成 120Ω终端电阻。

3.4 系统状态指示灯

USBCAN-I Pro 接口卡具有 1 个 PWR 指示灯、1 个 SYS 指示灯、1 个 RX 指示灯、1 个 TX 指示灯来指示设备的运行状态。这 4 个指示灯的具体指示功能见表 3.2，这 4 个指示灯处于各种状态下时，CAN 总线的状态如表 3.3 所示。

指示灯	颜色	指示状态

PWR	绿	电源指示
SYS	绿	系统指示
RX	绿	CAN接收指示
TX	绿	CAN发送指示

表 3.2 USBCAN-I Pro 接口卡指示灯

USBCAN-I Pro 接口卡上电后，四个指示灯同时点亮，之后 PWR 和 SYS 常亮，表明设备已经供电，系统完成初始化；否则，表示存在系统电源故障或其他故障，需联系我公司客服人员。

USB 接口连接正常后，当 PC 端有上位机软件调用 USBCAN 设备时，USB 信号指示灯 SYS 会闪烁。

当 CAN 总线上有数据收发时，对应的 RX、TX 指示灯会有闪烁。

指示灯	状态	指示状态
PWR	亮	电源供电正常
	不亮	电源供电故障
SYS	常亮	设备初始化通过，待机状态
	不亮	设备初始化未通过
	闪烁	PC端有软件调用设备
TX、RX	不亮	CAN总线无数据传输
	闪烁	CAN总线有数据传输

表 3.3 USBCAN-I Pro 接口卡指示灯状态

4. 软件使用

4.1 软件启动

安装好 Windows 驱动后，连接 CAN 分析仪硬件，打开软件会有如下图所示设置界面。

①“选择设备类型”：需选择对应硬件的版本，**V405/502 版本需选择 V5 类型**，具体版本型号参考设备外壳背面，其他设备类型请不要选择。选择好对应的设备类型后点击打开设备即可看到相应的设备信息。hardware 为我公司自定义硬件版本号，对于用户无任何意义；ID 为硬件 SN 号。

②“打开设备”按钮：点击该按钮，可调取 USBCAN 设备。若显示“USB 设备打开错误!”请检查①中选择的设备是否正确，设备管理器中的驱动是否安装正确。

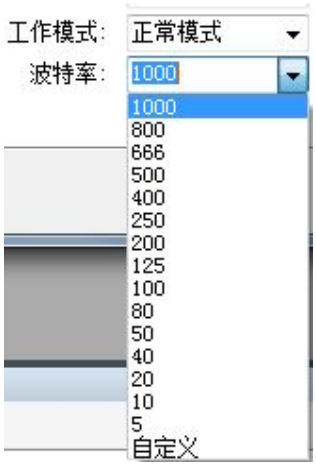
③设备显示窗口：该窗口可显示设备的硬件号及 SN 号码。

④通道选择界面：可在此处切换 CAN 通道的设置窗口。

⑤工作模式：可在此选择正常模式、只听模式和自发自收模式。正常使用时需要您选择默认的正常模式。

⑥波特率选择：您可在此处进行波特率的选择。波特率对于 CAN 总线的通信至关重要，通信前您需要确定目标设备或目标总线的波特率。

选中设备后，在下边可以设置一些具体的工作模式和波特率。如您购买的是双通道设备，可通过选项卡分别对 CAN1、CAN2 进行设置，设置波特率尤为重要，此处设置波特率不可以手动输入，需要通过下拉菜单选择，我公司产品支持如图所示标准波特率：



如您使用的是特殊波特率，请点击自定义按钮，此时需要您输入一个十六进制码，下表中列出了部分特殊波特率的值，如您需要其他波特率请联系我们，我们会帮您计算寄存器设置值。

波特率	参数
83.3kbps	27C017
66.6kbps	2DC013
60kbps	2BC018
35kbps	2EC023
33.3kbps	2EC025
25kbps	2FC02F
13.3kbps	16C0B3
12kbps	2BC07C

⑦自动识别波特率按钮：如您不知道目标总线或设备的波特率，可以选择“波特率自动识别”，自动识别成功条件：**被测设备上电且 CAN 端为活动状态。**

请注意，设置波特率尤其重要，许多客户反映设备连接上之后没有数据，或总线错误，实为波特率没有设置就直接点击确定打开设备。在这里提示您，无论您将我们的设备作为主或从设备使用，只要您将设备接入到 CAN 总线上，必须将设备的波特率设置成与目标设备波特率一致，才可以正常工作。

⑧确定按钮：选择好波特率后，可以点击确定，之后进入软件。此时如果硬件正确启动，SYS指示灯会由常亮变为闪烁状态（连续闪烁2次）。连接成功后界面显示如下图：

4.2 自动识别波特率

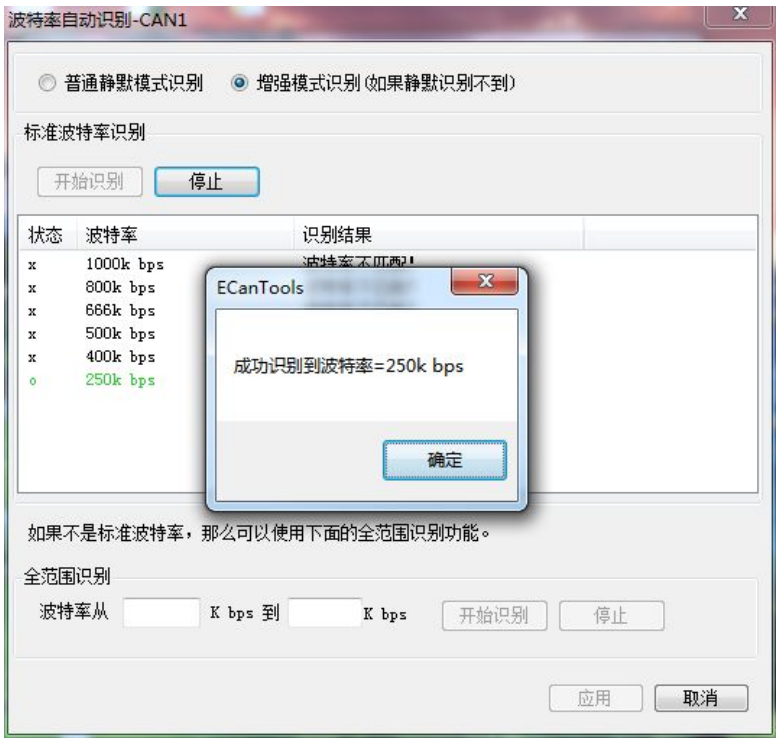
如您不清楚目标设备的波特率，可使用自动识别波特率功能。自动识别波特率有“普通静默模式识别”和“增强模式识别”两种模式可供选择。“普通静默模式识别”的环境要求为，CAN 总线上至少有两个被测设备且互相之间可以正常通讯，要求总线上有活跃的 CAN 数据。“增强模式识别”的环境要求为，被测设备上电且 CAN 通道工作正常，但不要求设备主动发送 CAN 数据，也不要求总线上有活跃的 CAN 数据。自动识别前请确保接线正确，且总线上连入两个 120 欧姆电阻。您可在不对 CAN 总线系统内设备上电的前提下，使用万用表对线路 CANL 与 CANH 两端的电阻值进行测量，应为 60 欧姆左右。

请注意，对于汽车的 CAN 总线系统，请选择“普通静默模式识别”来识别波特率。使用“增强模式识别”可能会造成车辆仪表报错。

波特率自动识别功能按检索范围的不同，可分为两种模式：A. 标准波特率识别（对标准的 CAN 波特率进行一一识别）， B. 全范围波特率识别（手动输入识别范围，软件将对范围内的波特率进行全面匹配）。例如，某 CAN 总线系统的波特率未知，使用标准波特率识别显示“没有找到合适的波特率”时，您可在“全范

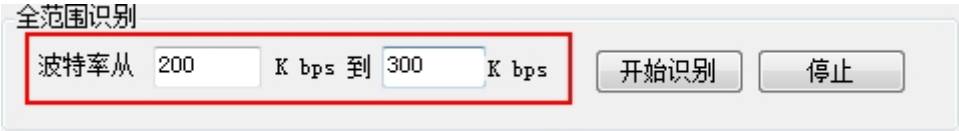
围识别”区域设置波特率从 5kbps 识别到 1000kbps，即可识别到所需的波特率。

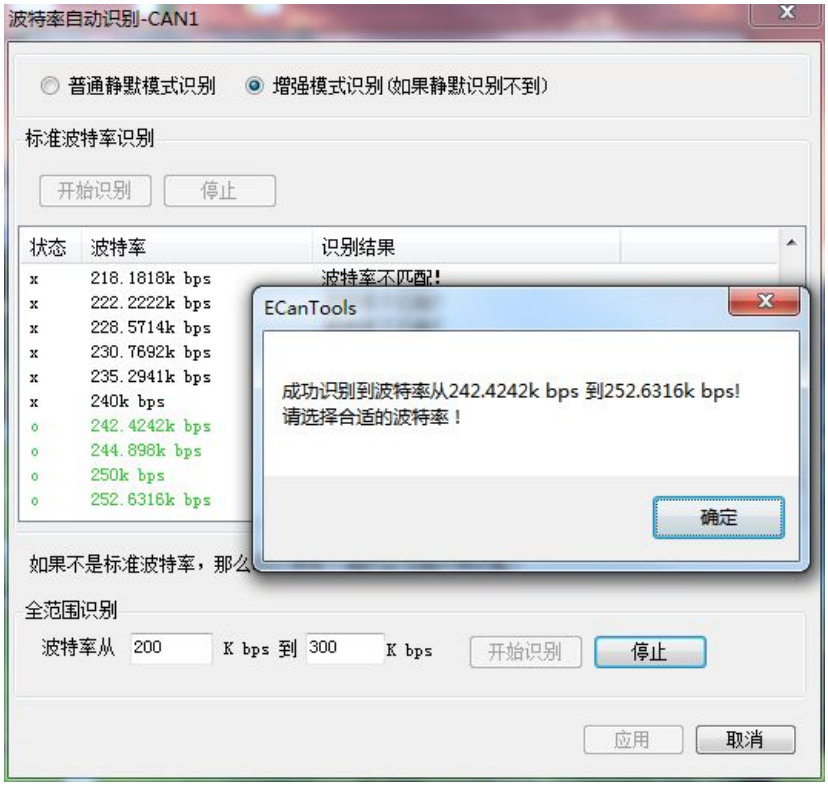
A.标准波特率识别截图：



B.全范围波特率识别截图：

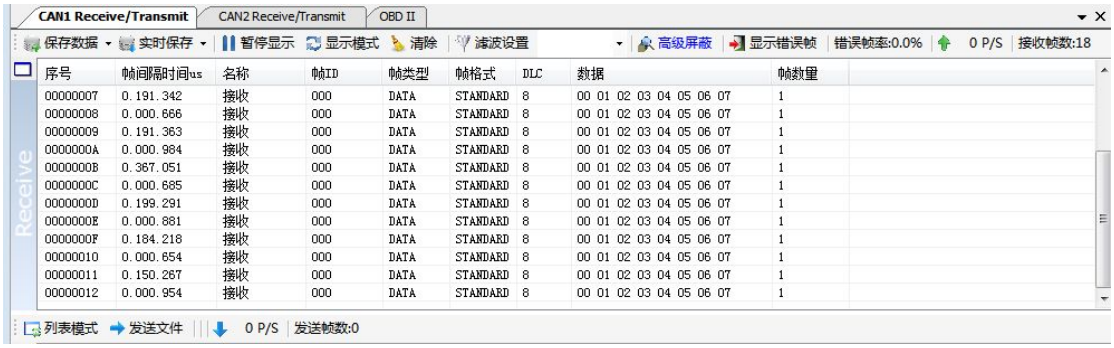
如果标准波特率识别不到未知节点的波特率，可使用全范围波特率进行进一步识别。本功能有助于帮助工程师修正新开发设备的波特率偏差。



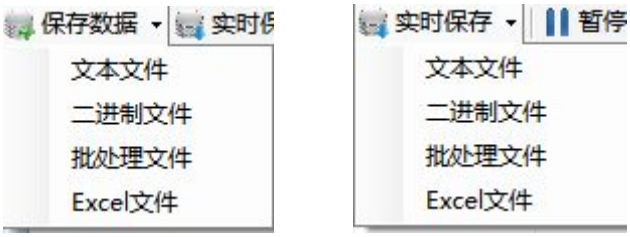


4.3 数据接收相关功能

设备参数设置好后，软件就进入工作状态，如果总线上有数据，这时接收数据窗口就会有数据显示。接收窗口如下图所示：



4.3.1 保存数据功能与实时保存功能



用户可将当前发送/接收列表中的全部数据保存到本地，保存格式详见下表：

文件类型	文件格式	编辑器
文本文件	.txt	记事本
二进制文件	.dat	记事本
批处理文件	.can	记事本
Excel 文件	.csv	WPS Excel

文本文件便于数据保存及后期分析，批处理文件可通过记事本软件进行修改后重新发送回总线。对于批处理文件格式的说明详见 3.4.3。

点击工具条上的“实时保存”，设置实时保存的文本类型和文件名，便可开始数据实时保存功能（即设置保存节点 A），再次点击（即设置保存节点 B），系统会停止保存，并将从开始(A)到结束(B)的数据全部写入保存文件。

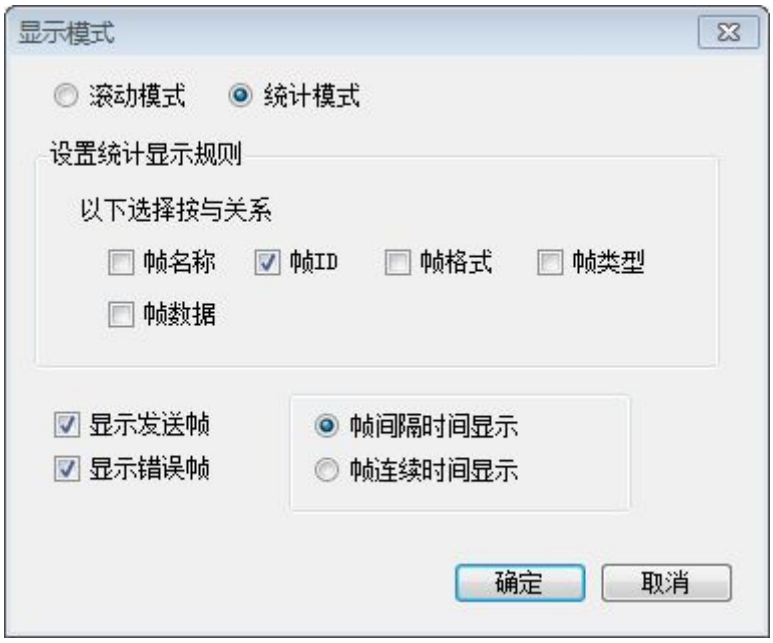
请注意，系统并不会实时写入数据。即在实时保存过程中，保存的文件中并不会有数据。

4.3.2 暂停显示功能

可以点击暂停，即可将目前滚动中的数据窗口暂停，暂停时设备和软件依旧可以正常接收数据，只是数据窗口不会刷新，点击继续显示即可恢复滚动。

4.3.3 显示模式

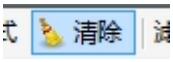
显示模式包括滚动模式和统计模式：



滚动显示是接收到的数据在接收列表中不停的向下滚动，当前窗口看到的是最新的数据；统计列表方式是按设置好的规则分类显示，如可设置同一 ID 的数据包显示统计在一起，后面有统计包数量。统计模式可方便工程师抓取总线上新产生的或有变化的数据。

序号	帧间隔时间us	名称	帧ID	帧类型	帧格式	DLC	数据	帧数量
00000000	0.000.937	接收	000	DATA	STANDARD	8	00 01 02 03 04 05 06 07	38

4.3.4 清除功能



可以清空接收/发送窗口中的数据，以及缓存区中的数据。

4.3.5 滤波设置

接收滤波设置可设置滤波 ID 或 ID 段，**如设置滤波，软件会只显示被设置的滤波 ID（段），不在滤波范围内的 ID 将会被过滤掉。**点击“编辑滤波”，弹出滤波设置窗口，可以设置滤波范围：

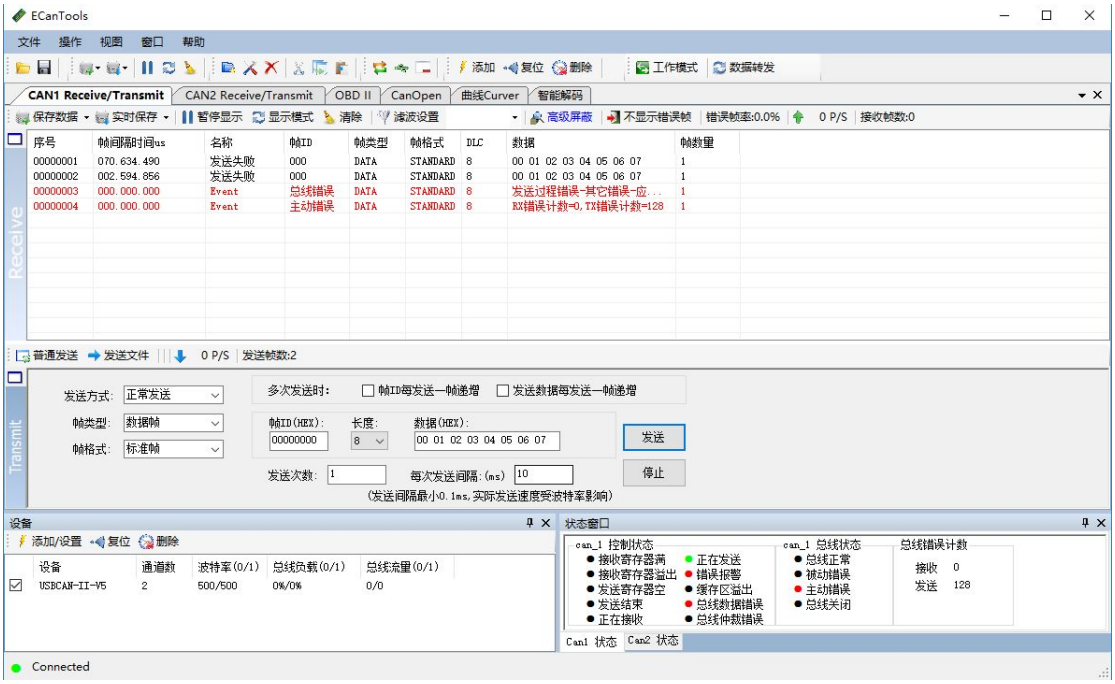
点击“添加滤波”即可开始编辑，输入想要过滤显示的 ID 或 ID 段后点击“保存设置”即可将滤波内容添加到的左侧。

左侧的滤波列表可添加多条数据，勾选相应的滤波段即可选择是否启用。本软件可同时启用多条滤波。**请注意，您需要勾选“设置使能滤波”后方可使滤波**

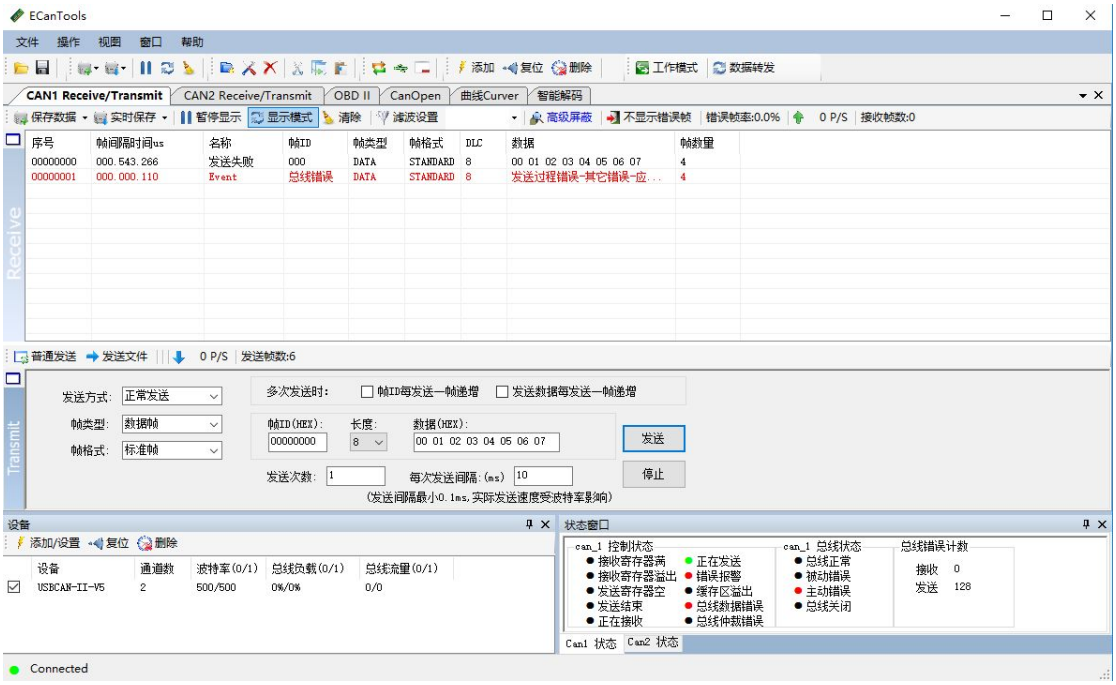
生效。滤波文件可保存并通过“打开滤波文件”进行加载。

4.3.6 接收错误帧显示功能

软件可以捕获总线上的错误帧，当接收到错误帧时，在接收列表中将以飘红的形式显示出来。



错误帧也可通过设置统计模式后进行合并显示。



4.4 数据发送相关功能

4.4.1 普通模式发送数据功能

普通模式可以非常直观的编辑要发送的帧数据，可设置循环发送等特殊功能。

普通模式比较简单，编辑帧信息非常直观，请注意输入数据时每个字节之间需要输入空格，否则将弹出“数据格式错误!”。

4.4.2 列表模式发送数据功能

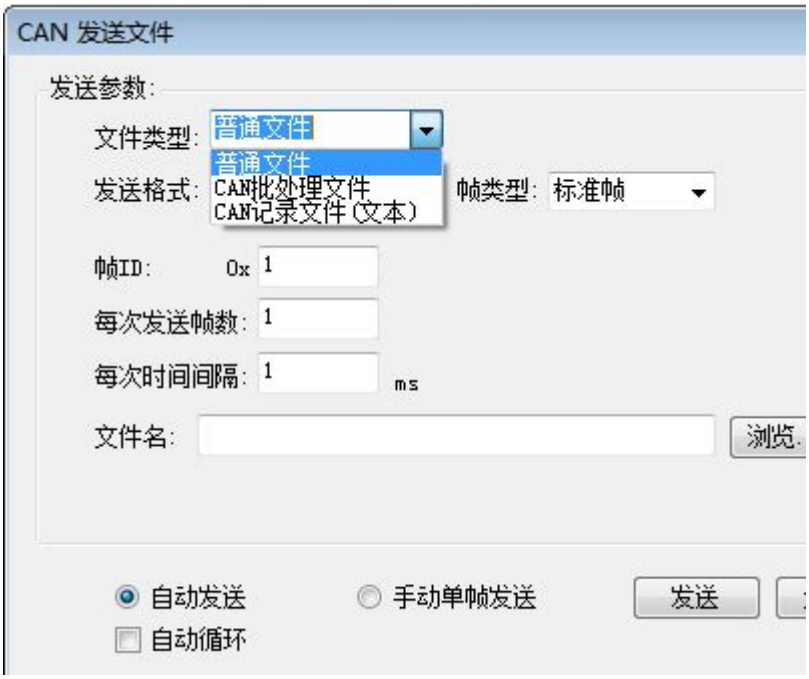
点击列表模式之后，原位置将切换为普通模式。点击新建帧，可以编辑想要发送的数据帧。

编辑数据界面中可设置帧 ID，帧类型，帧格式，帧长度等数据；在高级选项中，可以设置循环发送方式，可设置循环发送间隔时间，循环发送次数，可设置 ID 递增等方式。

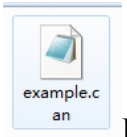
当设置好后，可将 CAN 报文添加到发送队列中：这时用鼠标点击 send 左边的方框可控制发送还是停止；当发送结束后复选框自动回复，循环发送的数据包在“已发送帧数”中可以看到成功发送的数据包数量。

4.4.3 文件发送功能

选择“发送文件”，即可打开发送文件对话框：



文件类型可设置普通文件或批处理文件(.CAN)，普通文件用于对 CAN 总线设备进行烧写程序，需自行开发烧录软件。这里着重介绍一下批处理文件的发送。



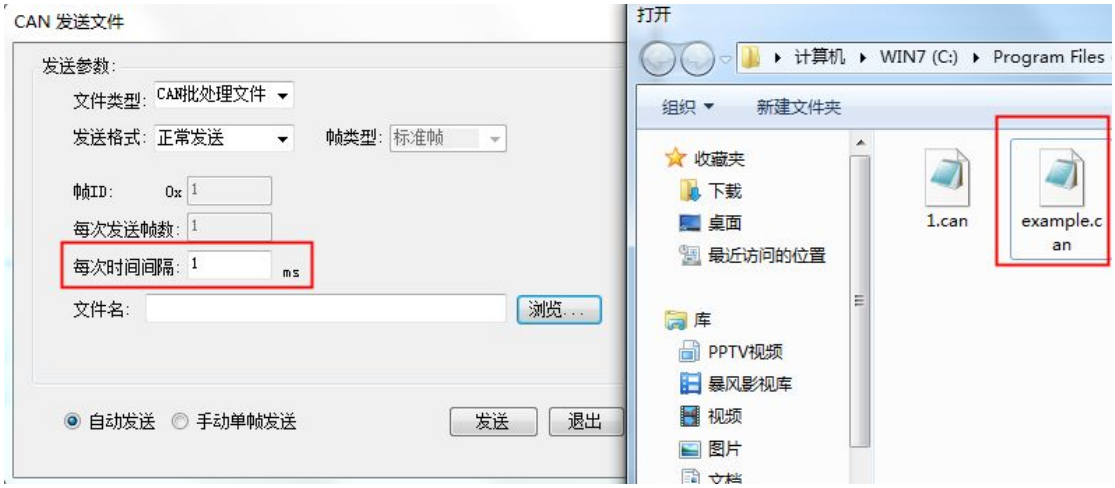
ECANTools 软件可以将接收到的数据保存为批处理文件(.CAN)，批处理文件可以使用记事本方式打开，打开之后会看到保存下来的帧数据内容及格式。如下图所示的“example.can”您可以在 ECANTools 软件中的安装目录中找到。

打开批处理文件(.CAN)后，可以直观的看到保存下来的数据参数，用户可以对这些参数进行修改及添加，各个位置的定义在上图中已经给出，注意批处理文件中不可以存在任何形式的非标准数据，上图中添加注释只为讲解，实际编辑中请勿添加。

	第一组	第二组	第三组	第四组	第五组
解释	帧间隔时间（单位毫秒）	标准/扩展	数据/远程	帧 ID	帧数据

例子	50,	1,	1,	0D223344,	01 02 03 04 05 06 07 08

请注意，您可能会批处理文件(.CAN)的帧间隔时间中发现存在 0 的现象，遇到这类数据 ECANTools 将按照该波特率下的最小发送间隔来发送，并不会出现多帧同时发送的情况。



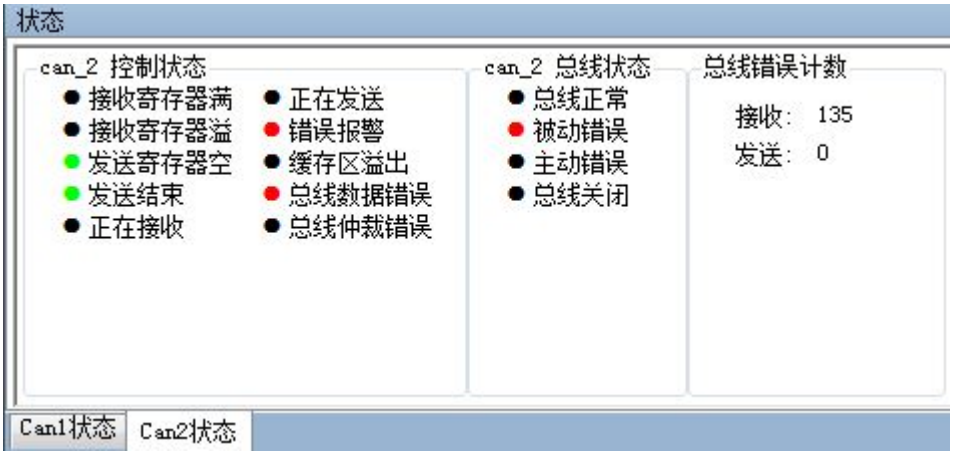
之后选择保存好的批处理文件就可以进行文件发送了，时间间隔默认为 1，请勿改动，如需设置时间间隔请在批处理文件中更改。

4.5 总线诊断功能

使用 ECANTools 软件，用户可通过软件右下角的状态，读出总线是否正常。典型举例：

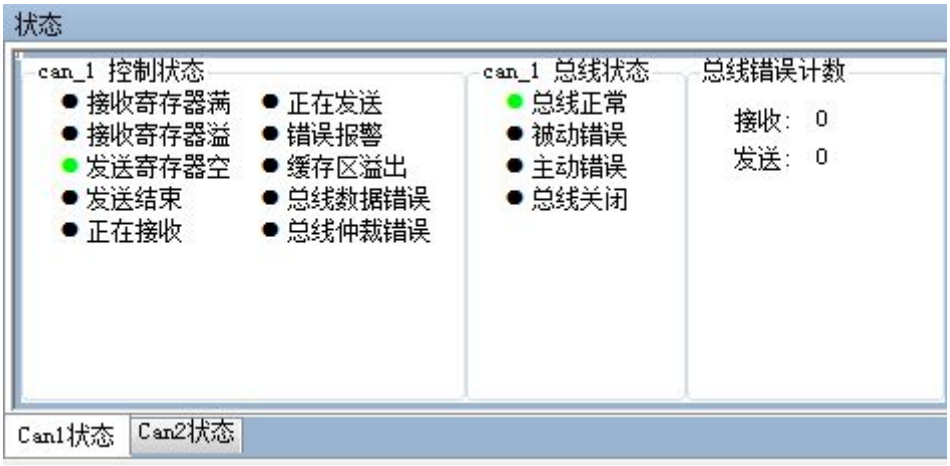
1、主动错误或被动错误：

说明 **波特率设置不正确** 或 **接线有误** 或 **总线空载** 或 **总线无响应**。



2、全是绿灯却没有数据：

说明总线空载，无数据可接收。



4.6 汽车电子解析 OBD II 界面

用户可以使用 ECANTools 软件的汽车信号解析功能。使用该设备接入汽车 OBD 接口即可将汽车内部部分传感器的实际数值解析出来，信号解析功能仅支持家用汽油车 ISO15765 协议。车速、转速、水温可通过软件仪表盘显示。

目前该功能仅支持 CAN1 通道。用户可通过此界面直观的看到汽车当前实时车速、转速和油量的具体数值，便于用户校对汽车仪表盘数值是否准确。

ECANTools 软件还可以读取、解析、清除汽车的故障码，解析 ISO15765 协议中规定的汽车传感器数据，包括：发动机转速、冷却液温度、车辆速度、电压、进气歧管压力、进气温度、空气流速、节气门位置、氧传感器电压、燃油压力等。并且以上这些数据的数值变化可以实时保存在电脑中。

使用时，请首先点选 11Bit 按钮，勾选“循环获取实时数据”。若此时 USBCAN 分析仪连接到汽车动力总线上，则将返回车速、转速、水温等参数。若此时无数据，可切换至 29Bit 重新尝试。若两种模式皆无反馈，请检查通信波特率及接线。

PID 手动获取功能：

您可通过输入 PID 的值来手动获取车辆的各种参数。请注意，需要首先选取正确的发送命令类型，即 11Bit 或 29Bit。详细的 PID 值（十进制）如下表所示。

PID	数据头	最小值	最大值	单位
01	当前故障码数量	0	127	个
04	计算负荷值	0	100	%
05	**发动机冷却液温度**	-40	215	℃
06	短时燃油修正(气缸列 1 和 3)	-100	99.22	%
07	长期燃油修正(气缸列 1 和 3)	-100	99.22	%

08	短时燃油修正(气缸列 2 和 4)	-100	99.22	%
09	长期燃油修正(气缸列 2 和 4)	-100	99.22	%
010	燃油压力计量	0	765	kPa
011	进气歧管绝对压力	0	255	kPa
012	**发动机转速**	0	9999	rpm
013	**车速**	0	255	km/h
014	第一缸点火正时提前角(不包括机械提前)	-64	63.5	°
015	进气温度	-40	215	℃
016	空气流量传感器的空气流量	0	655.35	g/s
017	绝对节气门位置	0	100	%
031	自发动机起动的时间	0	65535	s
033	在 MIL 激活状态下行驶的里程	0	65535	km
034	相对于歧管真空度的油轨压力	0	5177.265	kPa
035	相对于大气压力的油轨压力	0	655350	kPa
044	EGR 指令开度	0	100	%
045	EGR 开度误差 (实际开度-指令开度)/指令开度	-100	99.22	%
046	蒸发冲洗控制指令	0	100	%
047	**燃油液位输入**	0	100	%

048	自故障码被清除之后经历的暖机循环个数	0	255	N/A
049	自故障码被清除之后的行驶里程	0	65535	km
050	蒸发系统的蒸气压力	-8192	8192	Pa
051	大气压	0	255	kPa
060	催化器温度 B1S1	-40	6513.5	℃
061	催化器温度 B2S1	-40	6513.5	℃
062	催化器温度 B1S2	-40	6513.5	℃
063	催化器温度 B2S2	-40	6513.5	℃
066	**控制模块电压**	0	65.535	V
067	绝对负荷值	0	25700	%
068	等效比指令	0	2	N/A
069	相对节气门位置	0	100	%
070	环境空气温度	-40	215	℃
071	绝对节气门位置 B	0	100	%
072	绝对节气门位置 C	0	100	%
073	**加速踏板位置 D**	0	100	%
074	**加速踏板位置 E**	0	100	%
075	**加速踏板位置 F**	0	100	%

076	节气门执行器控制指令	0	100	%
077	MIL 处于激活状态下的发动机运转时间	0	65535	min
078	自故障码清除之后的时间	0	65535	min

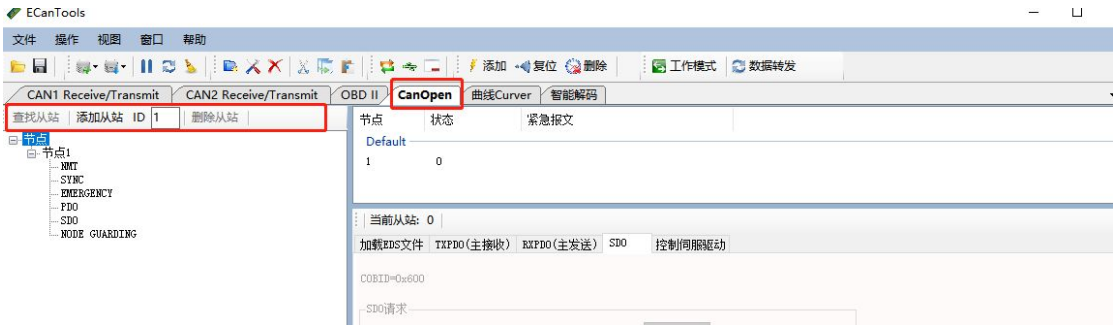
实时保存 TXT 文件功能：

用户可以通过此功能直观的看出汽车内部部分传感器的具体数值并基于这些数据诊断汽车各传感器状态是否正常。用户还可直观的比较某些传感器数值的变化规律和多种传感器数值之间进行比较。

。

4.7 CANopen 主站功能说明

切换到 ECANTools 软件的“CANopen ”选项卡可以进入 CANopen 主站页面，用户可以在左侧点击“查找从站”扫描当前 CAN 总线上的所有 CANopen 从站节点，也可使用手动输入从站节点号的方式，手动添加 CANopen 从站。扫描到或添加好的从站节点会在下方显示，用户可以双击指定的节点号以激活与从站通信的界面。



ECANTools 软件的 CANopen 主站功能可将 USBCAN 设备变成一个 CANopen 主站卡接入 CANopen 网络中，使用电脑搭配 USBCAN 即可完成对 CANopen 从站控制、管理、配置等应用。ECANTools 软件可完成多种常用 CANopen 主站功能，如 NMT（网络管理）、SYNC（同步协议）、EMERGENCY（紧急报文）、PDO（过程数据对象）、SDO（服务数据对象）、NODE GUARDING（节点保护）。

关于 CANopen 协议的应用层描述、设备子协议及接口规范等内容，您还可参阅由[德]Holger Zeltwanger 著，周立功等译，由北京航空航天大学出版社出版的《现场总线 CANopen 设计与应用》一书。

下表列出了 CANopen 常见报文类型所对应的功能码及帧 ID 范围。

报文类型	功能码	COB-ID 范围(Hex)
NMT	0000	000h
SYNC	0001	080h
EMERGENCY	0001	081h~0FFh
TIME	0010	100h
PDO1(发送)	0011	181h~1FFh
PDO1(接收)	0100	201h~27Fh
PDO2(发送)	0101	281h~2FFh
PDO2(接收)	0110	301h~37Fh
PDO3(发送)	0111	381h~3FFh
PDO3(接收)	1000	401h~47Fh
PDO4(发送)	1001	481h~4FFh
PDO4(接收)	1010	501h~57Fh
SDO(发送)	1011	581h~5FFh
SDO(接收)	1100	601h~67Fh
NMT Error Control	1110	701h~77Fh

4.7.1 NMT 命令

NMT 命令（Network Management）提供网络管理（如初始化、启动和停止节点，侦测失效节点）服务。这种服务是采用主从通讯模式（所以只有一个 NMT 主节点）来实现的。

用户可以使用 NMT 命令更改从站节点的运行状态。

Start remote node——设置节点进入操作状态

Stop remote node——设置节点进入停止状态

Enter pre-operational state——设置节点进入预操作状态

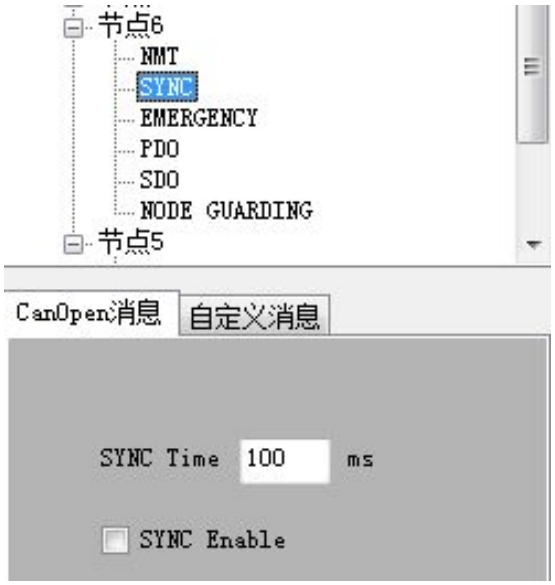
Reset node——设置节点复位

Reset communication——设置节点复位通信

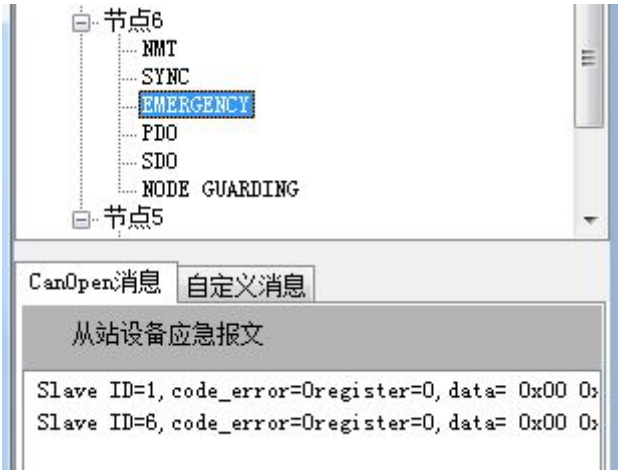
4.7.2 SYNC 报文

SYNC 是主站发出的同步报文，所有设置了同步 PDO 的从站根据 SYNC 报文与主站进行通信。

用户可通过ECANTools软件循环发送SYNC报文,所有的同步PDO会根据SYNC报文发送。


节点6
NMT
SYNC
EMERGENCY
PDO
SDO
NODE GUARDING
节点5
CanOpen消息
自定义消息
SYNC Time
100
ms
SYNC Enable

4.7.3 EMERGENCY 报文



应急指示报文由设备内部出现的致命错误触发，由相关应用设备以最高优先级发送到其它设备。适用于中断类型的错误报警信号。

ECANTools 软件可接收到从站节点发出的 EMERGENCY 报文，该报文由出现紧急状态的从节点发出，ECANTools 软件的 CANopen 主站功能可接收并处理此紧急报文。

4.7.4 PDO

CANopen 节点之间使用 PDO 进行数据传送，主要用于传输实时数据。

过程数据对象（PDO），全称为 Process data object。它与 SDO 的通信区别在于，PDO 属于过程数据，单向传输，无需接收节点回应 CAN 报文来进行确认，属于“生产消费”模型。

ECANTools 软件默认有 4 对发送和接收 PDO，即 TPDO/RPDO。

4.7.5 SDO

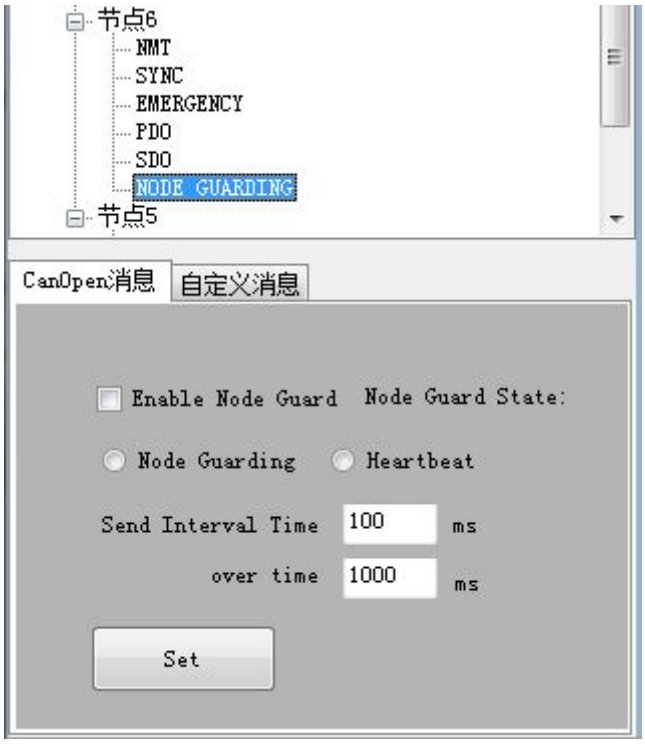
CANopen 主站模块的 SDO 功能分为发送 SDO 和接收 SDO，用于读写对象字典。SDO 为服务数据对象，英文全称 Service data object，有指定被接收节点的地址（Node ID），并且需要指定的接收节点回应 CAN 报文来确认已经接收，如果超时没有确认，则发送节点将会重新发送原报文。这种通讯方式属于常见的“服务器客户端”的通信模型，即我们常说的轮询式。

ECANTools 软件中的 SDO 功能，可通过编辑索引、子索引的方式与从站进行信息交互，实现了对对象字典中条目的读写。

索引和对象（OD）表

索引	对象
0000h	保留
0001h~001Fh	基本的数据类型
0020h~003Fh	复杂的数据类型
0040h~005Fh	生产商相关数据类型
0060h~007Fh	设备描述的基本的数据类型
0080h~009Fh	设备描述的复杂数据类型
00A0h~0FFFh	保留
1000h~1FFFh	通讯参数
2000h~5FFFh	制造商的特殊设备描述文件
6000h~9FFFh	标准设备描述文件
A000h~BFFFh	标准接口描述文件
C000h~FFFFh	保留

4.7.6 NODE GUARDING



ECANTools 软件中的节点保护（NODE GUARDING）功能有两种实现方式——主站主动发送命令来询问从站节点的状态（Node Guarding）、从站节点以心跳的方式周期传送它的状态（Heartbeat）。

ECANTools 软件还可通过编辑自定义消息手动模拟 CANopen 数据，用户可直接通过编写功能码、从站 node ID、数据的方式直接向 CANopen 从站发送数据。

功能码对应功能如下表所示：

----配置 PDO 0x1800+n

----它的 COB-ID 是 0x387

----该 PDO 始终触发传输

----它必须包含数据：data X(2bytes),data Y(4bytes),

按下面的顺序

--data X 定义在索引 0x6000,子索引 03

--data Y 定义在索引 0x2010,子索引 21

1-- 索引 1800+n，子索引 01：写 COBID（4bytes）

2-- 索引 1800+n，子索引 02：写传输字节《t》(1byte)

t=1to0xF0:PDO 在每接到《t》个 SYNC 后被传输

t=FD :在接收到 PDO 请求（rtr=1）后传输

t=FF:根据事件进行传输，节点自发发送 PDO

3-- 索引 1A00+n:定义第 n 个数据的映射

子索引 0：写嵌入到 PDO 中的数据个数（1byte），本例是，写入《2》

子索引 1：定义在哪里寻找嵌入的第一个数据和大小。(8bytes)

格式是：index (2 bytes) - subindex (1 byte) - size in bits (1 byte)

本例，写入《60000310》

子索引 2：定义在哪里寻找嵌入的第二个数据和大小。(8bytes)

本例，写入《20102120》

配置节点 5 的 PDO 1802 在每 3 个 SYNC 传输，发送的 SDO（s）应为

605 23 02 18 01 00 00 87 03

605 2F 02 18 02 03 00 00 00

605 2F 02 1A 00 02 00 00 00

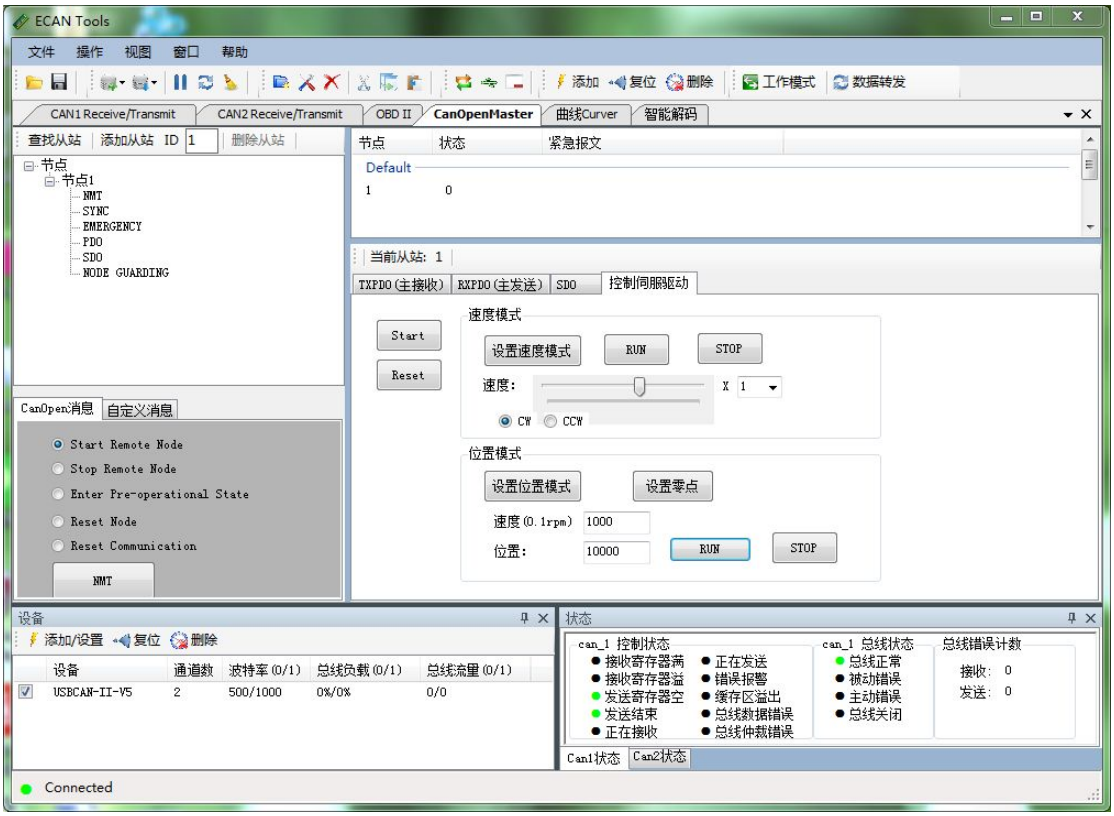
605 23 02 1A 01 10 03 00 60

605 23 02 1A 02 20 21 10 20

4.7.7 控制伺服驱动

该功能仅支持符合 CANopen402 标准的伺服电机设备，使用该功能时请提前确认。

现阶段仅开放“设置速度模式”供您对伺服电机设备进行简单控制。连接伺服电机设备的 CAN 总线后，依次点击“查找从站”—“Start”—“设置速度模式”—“RUN”，即可将伺服电机驱动。

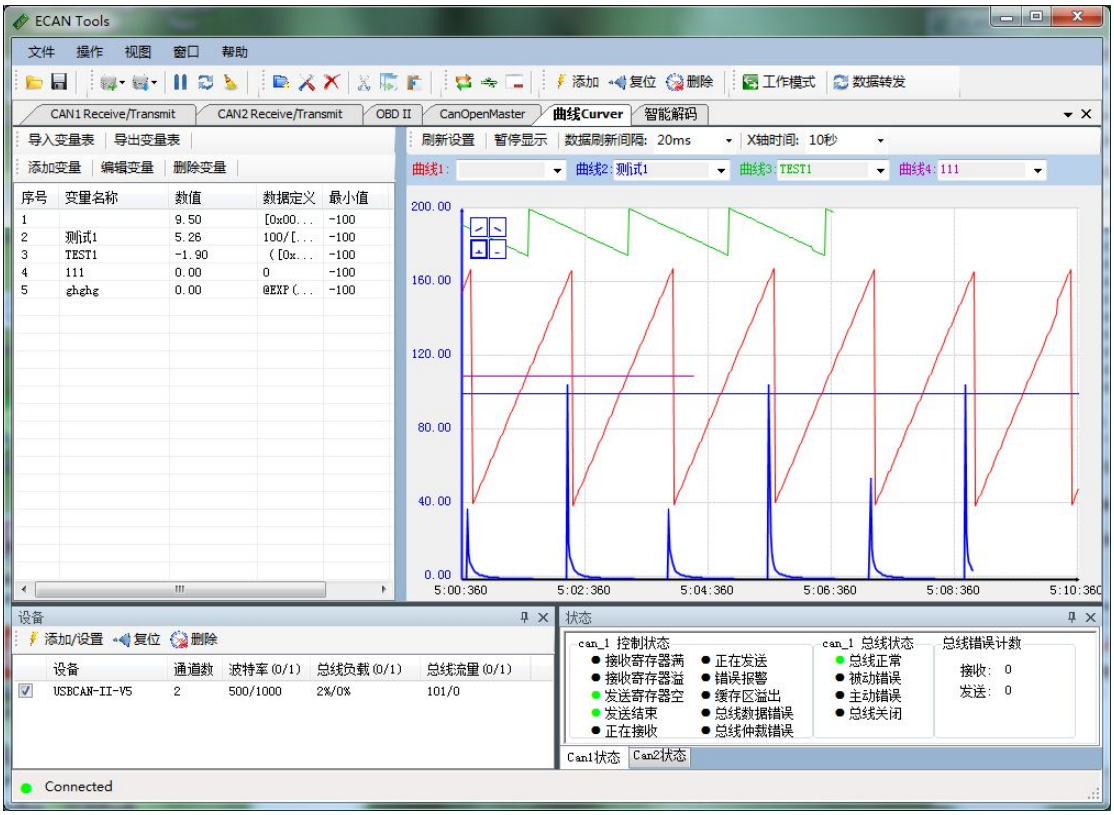

ECAN Tools
文件 操作 视图 窗口 帮助
添加 复位 删除 工作模式 数据转发
CAN1 Receive/Transmit
CAN2 Receive/Transmit
OBD II
CanOpenMaster
曲线Curver
智能解码
查找从站 添加从站 ID 1 删除从站
节点
节点1
NMT
SYNC
EMERGENCY
PDO
SDO
NODE GUARDING
节点 状态 紧急报文
Default
1 0
当前从站: 1
TXPDO(主接收) RXPDO(主发送) SDO 控制伺服驱动
速度模式
Start
Reset
设置速度模式 RUN STOP
速度:
X 1
CW CCW
位置模式
设置位置模式 设置零点
速度(0.1rpm) 1000
位置: 10000 RUN STOP
CanOpen消息 自定义消息
Start Remote Node
Stop Remote Node
Enter Pre-operational State
Reset Node
Reset Communication
NMT
设备
添加/设置 复位 删除
设备 通道数 波特率(0/1) 总线负载(0/1) 总线流量(0/1)
USBCAN-II-V5 2 500/1000 0%/0% 0/0
状态
can_1 控制状态
接收寄存器满 正在发送
接收寄存器溢 错误报警
发送寄存器空 缓存区溢出
发送结束 总线数据错误
正在接收 总线仲裁错误
can_1 总线状态
总线正常
被动错误
主动错误
总线关闭
总线错误计数
接收: 0
发送: 0
Can1状态 Can2状态
Connected

4.8 曲线 Curver 功能说明

数据数值变化也可通过曲线实时显示其变化轨迹，如下图所示。用户最多可选择四个变量在同一界面同时显示。

用户可以通过曲线功能选择最多四个您所需要的变量，并通过观察曲线的方式实时了解每个数据的变化，并且还可直观的比较多个数值之间的相对变化，可用于调整发动机怠速时转速、扭矩等。

在使用曲线功能时请注意，将变量设置好后再接收数据，数据接收时新建的变量无法在曲线中显示



点击“添加变量”“编辑变量”即可添加和编辑变量，这些变量可以对接收的 CAN 总线的原始数据进行简单的数学运算，并在以曲线的方式显示出来。可以添加多组变量，变量名称可包含字母，数字，汉字，空格和下划线，方便识别。

曲线功能中可提供的特殊运算如下表所示，**请严格按照运算形式书写！**

运算符	运算功能	书写格式（以正弦为例）：使用特殊运算符时，运算符括号内需要有至少一则运算。例如：@SIN(数据源)是
sin/cos/tan	正弦/余弦/正切	
sqr	求一个非负实数算数平方根	
log	求一个数以 e 为底的对数（ln x）	

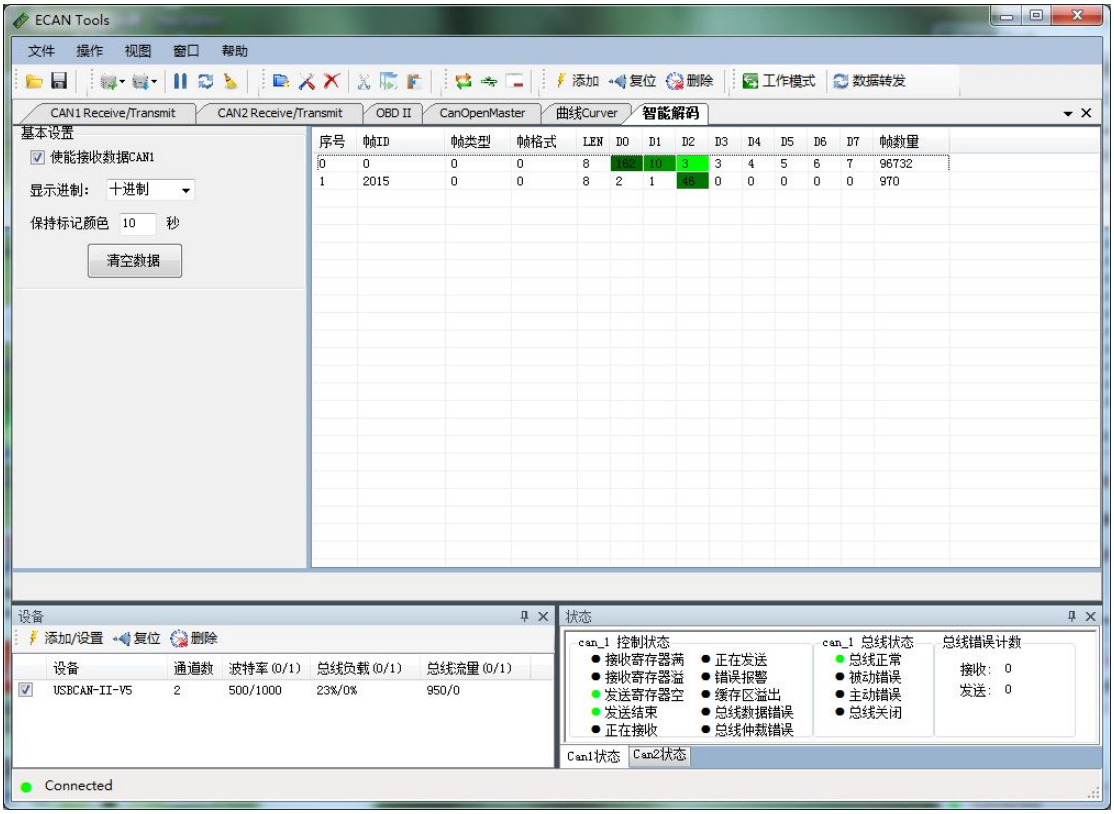
abs	求一个数的绝对值	错误的书写方式，会提示非数字，正确的书写方式为@SIN(数据源*1)
exp	求以 e 为底的指数函数的值（e^x）	
Max	变量当前最大值	
Min	变量当前最小值	

“添加数据源”就是将收到的 CAN 总线原始数据中某一帧的某一字节的数据作为数据源添加到变量计算定义中，添加时先点击“添加数据源”，然后依次填充“CAN ID”、“标准帧（扩展帧）”、“数据”等信息，点击确定，会在“变量计算定义”中生成一条相应的语句，该语句也可自行编写，编写时请遵循该格式。

在曲线界面中，“  ”功能是切换到上一条曲线，“  ”功能是切换到下一条曲线；“  ”功能是将当前颜色曲线坐标轴上移，“  ”功能是将当前颜色曲线坐标轴下移。

4.9 智能解码功能说明

智能解码功能现仅支持对 CAN1 通道特定帧 ID 的帧数据变化频率的显示，颜色随着变化频率的大小改变，变化越快，颜色越深，可以调整为十六、十、二进制显示。



5. Linux 系统使用说明

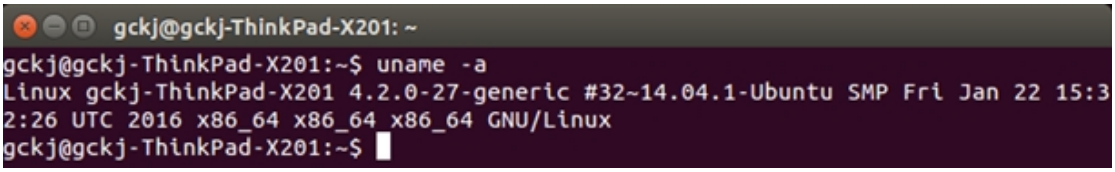
广成科技 USBCAN 分析仪系列产品支持各种版本 Linux 操作系统，我公司会向客户提供 32/64 位 Linux 系统驱动及二次开发相关文档，用户可自行在 Linux 系统中开发使用。

Linux 系统中使用我公司设备的大体方法如下：①获取系统管理员权限；②拷贝必要的文件到系统 GCC 编译目录中；③切换目录到 USBCAN 驱动文件夹进行编译；④运行测试程序。具体操作方法如下：

1. 查询 linux 版本号，确认系统类型（32/64 位）。

输入：uname -a

（由结果可知我们 Linux 系统类型是 64 位的）



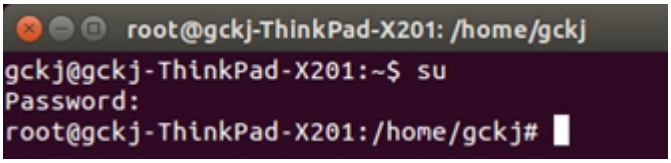
2. 确定 Linux 系统类型后，拷贝对应的 USBCAN 驱动文件到系统中。

（此例子中我们把驱动文件放到系统桌面）

3. 获取管理员权限，以便于后续安装驱动等操作。

输入：su

（输入 su 指令后要求输入管理员密码，输入正确的密码即可获取管理员权限）



4. 进入 USBCAN 驱动文件夹，拷贝 libusb.so、libusb-1.0.so、libECanVci.so.1 到 gcc 编译库目录下。（默认路径为/usr/lib）

输入：cp libusb.so libusb-1.0.so libECanVci.so.1 /usr/lib

（默认路径为/usr/lib）

```
root@gckj-ThinkPad-X201:/home/gckj# cd Desktop/linux64
root@gckj-ThinkPad-X201:/home/gckj/Desktop/linux64# cp libusb.so libusb-1.0.so l
ibECanVci.so.1 /usr/lib
root@gckj-ThinkPad-X201:/home/gckj/Desktop/linux64#
```

5. 进入 gcc 编译库文件夹，把 libECanVci.so.1 和 libECanVci.so 链接到一起。

输入：ln -sv libECanVci.so.1 libECanVci.so

```
root@gckj-ThinkPad-X201:/usr/lib# ln -sv libECanVci.so.1 libECanVci.so
‘libECanVci.so’ -> ‘libECanVci.so.1’
root@gckj-ThinkPad-X201:/usr/lib#
```

6. 再次进入 USBCAN 驱动文件夹，编译。

输入：make

```
root@gckj-ThinkPad-X201:/# cd /home/gckj/Desktop/linux64
root@gckj-ThinkPad-X201:/home/gckj/Desktop/linux64# make
removed ‘test’
gcc -o test test.c -lpthread -lECanVci -lusb
root@gckj-ThinkPad-X201:/home/gckj/Desktop/linux64#
```

7. 运行测试程序测试 USBCAN 收发。

输入：./test

```
root@gckj-ThinkPad-X201:/home/gckj/Desktop/linux64# ./test
test [DevType] [DevIdx] [ChMask] [Baud] [TxType] [TxSleep] [TxFrames]
    example: test 16 0 3 0x1400 0 1 1000
                     | | | |      | | |
                     | | | |      | | |1000 frames / channel
                     | | | |      | |
                     | | | |      | |tx > sleep(3ms) > tx > sleep(3ms) ....
                     | | | |      |
                     | | | |      |0-normal, 1-single, 2-self_test, 3-single_self_
test, 4-single_no_wait....
                     | | | |
                     | | | |0x1400-1M, 0x1c03-125K, ....
                     | | |
                     | | |bit0-CAN1, bit1-CAN2, bit2-CAN3, bit3-CAN4, 3=CAN1+CAN2,
 7=CAN1+CAN2+CAN3
                     | |
                     | |Card0
                     |
                     |1-usbcan,  ....
root@gckj-ThinkPad-X201:/home/gckj/Desktop/linux64#
```

输入./test 后会出现系统提示及例子，其中：

第一位（16）：设备类型，单通道设备输入 3，双通道输入 4；

第二位（0）：设备索引号，当只接入一台 USBCAN 时为 0；

第三位（3）：打开第几路 CAN，打开 CAN1 输入 1，打开 CAN2 输入 2 同时打开 CAN1 和 CAN2 输入 3；

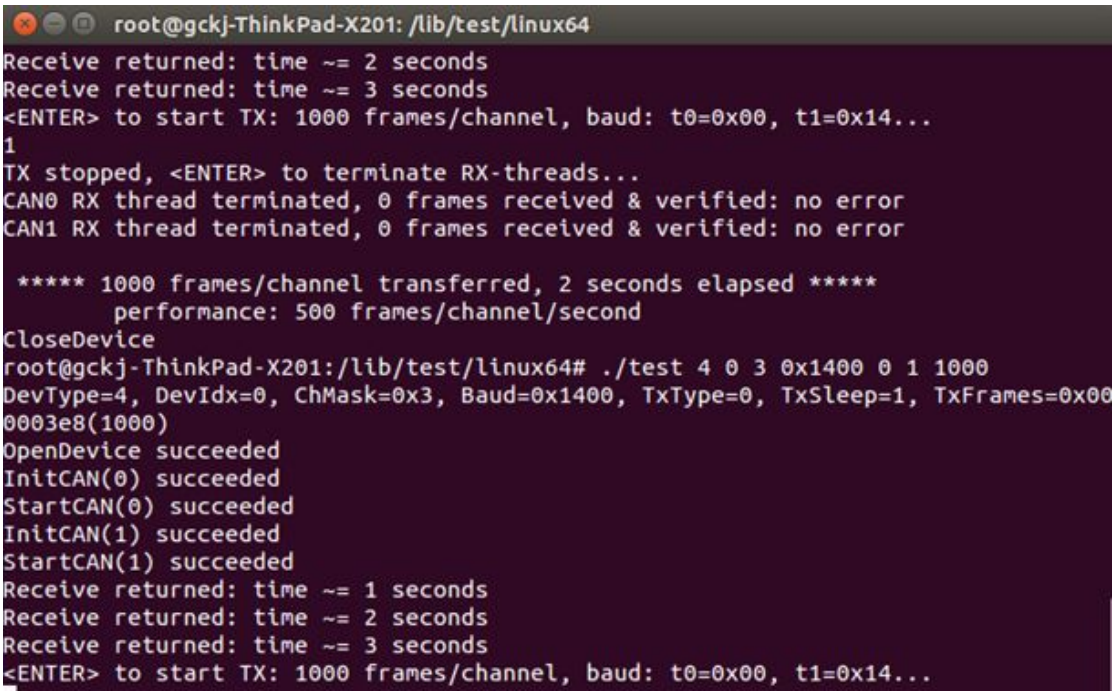
第四位（0x1400）：设置 CAN 总线波特率，0x1400 表示波特率为 1000K，其他波特率的值详见“EcanVCI 动态库使用手册”；

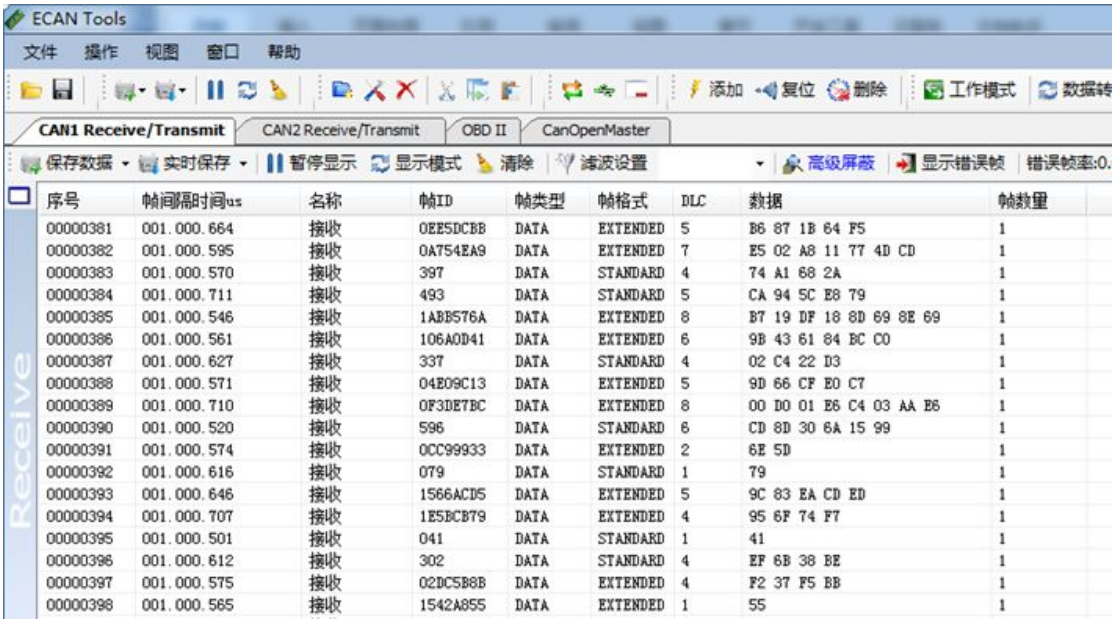
第五位（0）：工作模式，0 为正常模式，其他工作模式详见“EcanVCI 动态库使用手册”；

第六位（1）：发送时间间隔，单位 ms；

第七位（1000）：发送次数。

8. 运行测试程序后即可使用其他 USBCAN 设备收到他发出的数据





序号	帧间隔时间us	名称	帧ID	帧类型	帧格式	DLC	数据	帧数量
00000381	001.000.664	接收	0EE5DCBB	DATA	EXTENDED	5	B6 87 1B 64 F5	1
00000382	001.000.595	接收	0A754EA9	DATA	EXTENDED	7	E5 02 A8 11 77 4D CD	1
00000383	001.000.570	接收	397	DATA	STANDARD	4	74 A1 68 2A	1
00000384	001.000.711	接收	493	DATA	STANDARD	5	CA 94 5C E8 79	1
00000385	001.000.546	接收	1ABB576A	DATA	EXTENDED	8	B7 19 DF 18 8D 69 8E 69	1
00000386	001.000.561	接收	106A0D41	DATA	EXTENDED	6	9B 43 61 84 BC C0	1
00000387	001.000.627	接收	337	DATA	STANDARD	4	02 C4 22 D3	1
00000388	001.000.571	接收	04E09C13	DATA	EXTENDED	5	9D 66 CF E0 C7	1
00000389	001.000.710	接收	0F3DE7BC	DATA	EXTENDED	8	00 D0 01 E6 C4 03 AA E6	1
00000390	001.000.520	接收	596	DATA	STANDARD	6	CD 8D 30 6A 15 99	1
00000391	001.000.574	接收	0CC99933	DATA	EXTENDED	2	6E 5D	1
00000392	001.000.616	接收	079	DATA	STANDARD	1	79	1
00000393	001.000.646	接收	1566ACD5	DATA	EXTENDED	5	9C 83 EA CD ED	1
00000394	001.000.707	接收	1E5BCB79	DATA	EXTENDED	4	95 6F 74 F7	1
00000395	001.000.501	接收	041	DATA	STANDARD	1	41	1
00000396	001.000.612	接收	302	DATA	STANDARD	4	EF 6B 38 BE	1
00000397	001.000.575	接收	02DC5B8B	DATA	EXTENDED	4	F2 37 F5 BB	1
00000398	001.000.565	接收	1542A855	DATA	EXTENDED	1	55	1

6. 二次开发

我公司为二次开发的用户提供标准的接口函数库，包括：ECANVCI.h、ECANVCI.lib、ECANVCI.dll。该接口函数库均为标准格式，用户可以在VC、VB、Labview等编程环境中，对这些接口函数声明调用，具体使用方法详见“ECAN动态库使用手册”。图6.1为常用结构体名称及函数库调用流程。

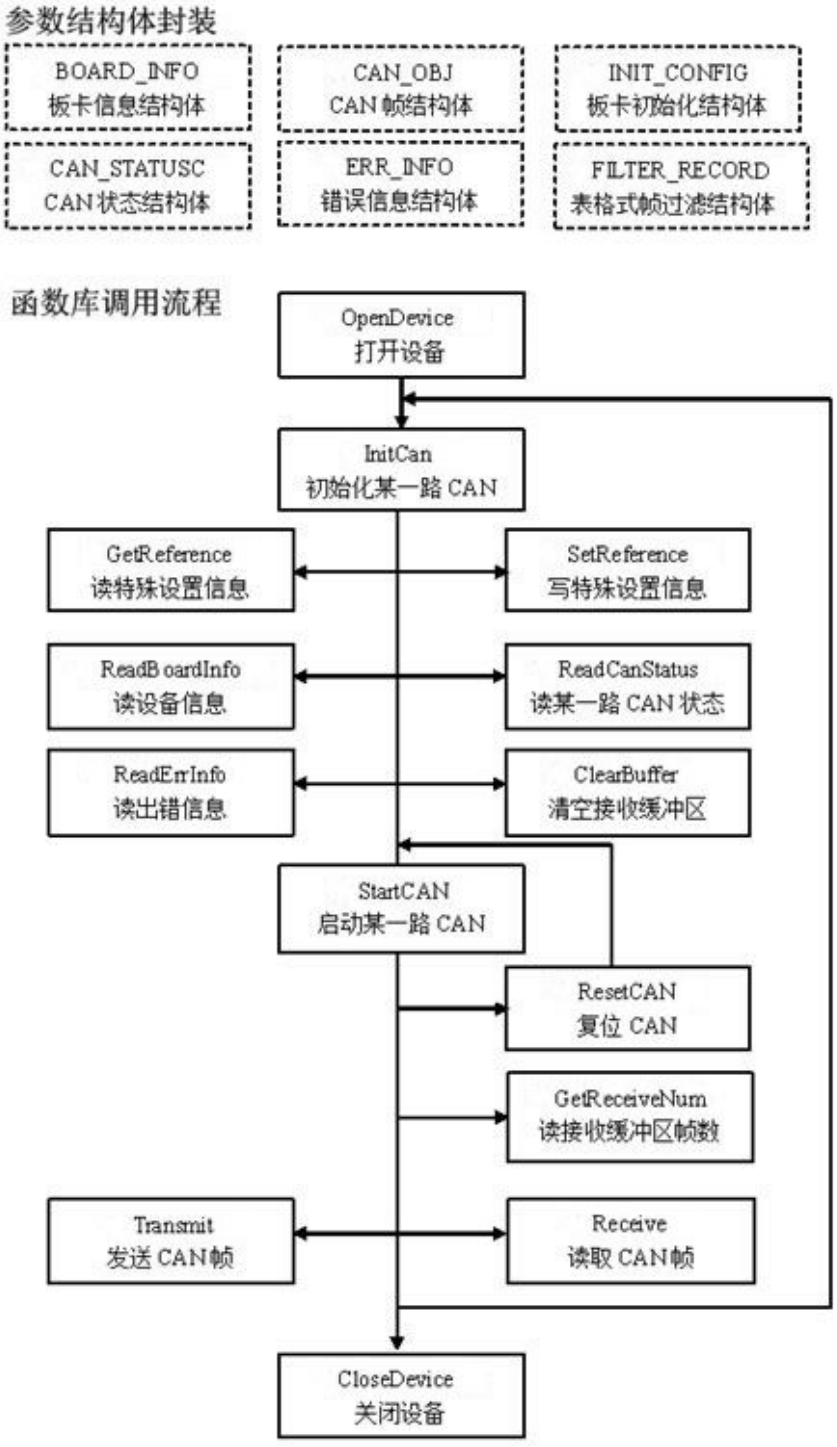


图 6.1 二次开发流程

7. 技术规格

连接方式	
PC端	USB接口，A类型
CAN端	DB9
接口特点	
USB接口	USB2.0全速接口，兼容USB1.1，USB3.0
CAN接口	遵循ISO 11898标准，支持CAN2.0A/B
CAN波特率	5Kbit/s~1Mbit/s
电气隔离	1500V，DC-DC
CAN终端电阻	已集成
供电电源	
供电电压	+5V DC （USB接口）
供电电流	最大130mA
环境试验	
工作温度	-40℃~+85℃
工作湿度	15%~90%RH，无凝露
EMC测试	EN 55024:2011-09 EN 55022:2011-12
防护等级	IP 20
基本信息	
外形尺寸	87mm *51mm *21mm
重量	90g

8. 常见问题

1. 在 ECANTools测试软件中，为何显示“打开设备错误”？

可能产生此类故障的原因是：设备类型选择不正确或 USBCAN 设备驱动没有正常安装。在 PC 的设备管理器中检查 USBCAN 设备属性，看看有没有“！”或“？”在 USBCAN 设备前面；若有，则检查硬件/软件冲突，并重新安装 USBCAN 设备驱动程序。

2. 是否一定需要使用 120Ω 终端匹配电阻？

建议 120Ω终端匹配电阻用于吸收端点反射，提供稳定的物理链路。当进行单节点的自发自收测试时必须连接该 120Ω的终端电阻构成回路，否则无法进行自发自收测试。USBCAN 高性能 CAN 接口卡内部已经连接有 120Ω的终端电阻。

3. 一台计算机能否安装多块USBCAN接口卡？

旧版的接口不支持多卡同时操作，但是目前的 USBCAN 接口卡，支持多达 8 个同一型号的 USBCAN 接口卡同时操作。

4. USBCAN-I Pro接口卡最高的数据转换率是多少？

USBCAN 接口卡的单一 CAN 通道最高支持 8000 fps 的 CAN 总线数据转换，这里提到的帧是指标准帧 8 个数据的数据帧，如果是小于 8 字节数据或者远程帧可能会更快。另外，最高数据流量会受 PC 性能的限制。

5. 为何CAN状态指示灯不亮？

因为 USBCAN 接口卡的所有操作是受 PC 机控制的，只有 PC 机发送了启动 CAN 通讯的命令后，CAN 状态指示灯才会有意义。

6. 为何调用接口函数时系统非法操作？

首先在使用接口函数时请认真阅读函数说明，保证输入参数合法，特别注意指针(地址)的传递，或参照提供的例子程序，倘若问题还是未能解决，可联系我们的技术支持。

7. USBCAN接口卡的通讯波特率如何设置？

设备提供了一组常用的波特率的设置值，若要使用其他的波特率，请联系广成科技客服进行计算。需要注意：USBCAN 接口卡的 CAN 控制器使用 24MHz 时钟，用户自定义波特率时要根据该时钟频率进行计算。

8. 系统进入待机或睡眠状态是否影响接收？

会有影响。这时所有处理将停止，最大可能导致硬件接收缓冲溢出错误。若有程序打开设备将尝试阻止系统进入待机或睡眠状态，从而保证系统正常工作。

使用 USBCAN 接口卡时，请禁止系统的待机和睡眠功能。

9. 如何处理应用中的错误？

错误主要分为函数调用错误和 CAN-bus 通讯错误两种。 函数调用错误一般由参数错误引起，如：设备号超出范围，类型号错误等，用 Win32 函数 GetLastError 返回的错误号是 87，还有的是对未打开的设备进行操作，实际是对一个非法句柄操作，根据具体函数调用情况都有相应的 Win32 标准错误码提供，用户可以使用 GetLastError 进行错误分析，这部分除错工作一般应该在设计时完成。

对于 CAN-bus 通讯错误，一般由 CAN 网络引起，也可能因用户设置不当而引起，如：波特率设置不一致、没有启动 CAN 控制器便调用发送函数等。大部分错误已经在设备驱动中作了简单的处理，如果要进行更深层次的错误分析和处理，可以调用 ReadCANStatus 函数。

另外需要注意的是数据溢出中断错误，它的产生有两种可能：(1) 软件接收缓区冲溢出。 这说明应用程序无法及时处理接收到的数据，这时用户应该优化应用程序或更改通讯策略。 (2)硬件接收缓冲区溢出。产生这种错误是由于接收端 PC 中断延迟太大而引起的。只能通过 提升计算机性能或协调其余节点适当降低发送速度来解决。

10. 打开关闭设备要注意哪些事项？

USBCAN-I Pro 接口卡提供 1 个 CAN 端口且不允许共享方式打开设备，同一个设备不可被不同进程通过调用 OpenDevice 函数多次打开。OpenDevice 和 CloseDevice 函数一般在应用程序初始化和退出时只需要调用一次。当关闭设备时若能当前端口不再使用，应该先调用 ResetCAN 函数使当前端口脱离 CAN 总线，设备驱动程序只会在最后一个设备句柄关闭时才自动调用 ResetCAN 退出 CAN 总线的连接。

11. 如何使用中断方式操作通讯卡？

USBCAN 接口卡不提供直接操作中断的接口，因为中断已经在驱动程序中处理了。 需要在应用程序中操作中断的多数原因是：程序不知道数据何时能到达设备，需要得到一个接收消息的触发才能从缓冲读取数据。解决这个问题的一般手段是使用多线程（或多任务）。 即启动一个新的线程，在线程中循环调用 Receive 函数来查询接收缓冲。Receive 内部已实现了阻塞机制，在缓冲里没有数据时会挂起调用线程，这时不会占用 CPU 的时间，应用程序仍然可以处理其他事务。

12. 如何更好的使用Transmit发送函数？

USBCAN 接口卡的驱动提供约 128 帧发送缓冲 FIFO，每次 Transmit 调用最多发送约 128 帧数据。发送设备的发送速度由当前计算机软硬件性能决定，一般连续发送速度在 2000 fps 左右(标准数据帧 11Bytes，1Mbps)，若发送速度过快将有可能使远端接收设备数据溢出而失去响应，这样用户可在应用编程中适当添加延时以降低发送速度。

发送过程中每一帧都有超时限制，单帧发送时超时时间约 2 秒，一次发送多帧时最后一 帧发送超时为 2 秒，其余为 1 秒。发送超时一般由于 CAN 总线繁忙且当前节点优先级较低时发生，并不是函数调用或通讯错误，用户可以编程实现重发(一般中低速网络极少发生发送超时事件)。因此，在系统设计时注意保证 CAN 总线占用不应该超过总线容量的 60-70%。

13. 如何更好的使用Receive函数？

设备驱动提供 100000 帧软件接收缓冲区，这为应用编程人员提供了充足的反应处理时间。当软件接收缓冲数据溢出时设备驱动程序将调用 ResetCAN 复位 CAN 总线，同时置位 CAN status 的数据溢出中断标志位，注意软件缓冲溢出和 CAN 控制器硬件缓冲溢出都是使用该标志位。

接收函数提供 Wait 参数适合用于多线程编程，函数内部封装一个阻塞函数，其参数 Wait 含义如同 Win32 的 WaitForSingleObject 的 dwMilliseconds 参数(请参考 Win32API 说明)，它为 Receive 指定一个超时返回时间，单位为毫秒。

当 Wait 为 0 时函数调用时立即返回当前成功读取到的帧数，若接收缓冲为空则返回 0。当 Wait 非 0 时，若函数调用时接收缓冲中已经有数据则马上返回成功读取的帧数，若这时接收缓冲为空，函数将等待一个指定的超时到达或接收到数据才返回成功接收的帧数。当 Wait 为 0xFFFFFFFF 时为无限等待直到有数据接收到，建议不要把 Wait 设得过大，无限等待更应该注意。

nFrames 等于 0 时函数实际是一个通知消息返回，不要求读接收缓冲区，是一个特殊的技巧性用法。注意：若在主线程中调用 Receive 函数并且 Wait 非 0 则有可能引起应用程序暂时性的失去响应。若通过查询方式接收，一般应该把 Wait 设为 0。

9. 免责声明

感谢您购买广成科技的 GCAN 系列软硬件产品。GCAN 是沈阳广成科技有限公司的注册商标。本产品及手册为广成科技版权所有。未经许可，不得以任何形式复制翻印。在使用之前，请仔细阅读本声明，一旦使用，即被视为对本声明全部内容的认可和接受。请严格遵守手册、产品说明和相关的法律法规、政策、准则安装和使用该产品。在使用产品过程中，用户承诺对自己的行为及因此而产生的所有后果负责。因用户不当使用、安装、改装造成的任何损失，广成科技将不承担法律责任。

关于免责声明的最终解释权归广成科技所有。

销售与服务

沈阳广成科技有限公司

地址：辽宁省沈阳市浑南区长青南街 135-21 号 5 楼
邮编：110000
网址：www.gcgd.net
全国销售与服务电话：400-6655-220
售前服务电话与微信号：13889110770
售前服务电话与微信号：18309815706
售后服务电话与微信号：13840170070
售后服务电话与微信号：17602468871