目录

[1引言 1](#_Toc496990098)

[1.1目的 1](#_Toc496990099)

[1.2背景 1](#_Toc496990100)

[1.3项目意义 1](#_Toc496990101)

[2需求规定 2](#_Toc496990102)

[2.1任务概述 2](#_Toc496990103)

[2.2功能模块图 2](#_Toc496990104)

[2.3对具体功能的规定 3](#_Toc496990105)

[3运行环境规定 4](#_Toc496990106)

[3.1硬件要求 4](#_Toc496990107)

[3.2支持软件 4](#_Toc496990108)

[3.3接口 4](#_Toc496990109)

[3.4开发环境 4](#_Toc496990110)

[3.5 控制要求 5](#_Toc496990111)

[4、具体需求 5](#_Toc496990112)

[4.1 功能需求 5](#_Toc496990113)

[4.2 CAN信息数据格式 18](#_Toc496990114)

[4.3.1 可用性 20](#_Toc496990115)

[4.3.2 安全性 20](#_Toc496990116)

[4.3.3 可维护性 21](#_Toc496990117)

[5、验收标准 21](#_Toc496990118)

[5.1 文档验收标准 21](#_Toc496990119)

[5.2 软件验收标准 21](#_Toc496990120)

[5.3功能验收标准 21](#_Toc496990121)

# 1引言

### 1.1目的

本说明书的编写是为了CANTool的功能需求和性能需求，以标准的语言和表述方式整理项目需求，以便于用户对本项目的理解和认识。

### 1.2背景

控制器局域网(CAN)是国际上应用最广泛的现场总线之一。CAN­­­­­­­­­­­­被设计作为汽车环境中的微控制器通讯，在车载各电子控制装置ECU之间交换信息，形成汽车电子控制网络。为了检测和控制CAN bus的信息内容，需要使用CAN bus检测设备。CanTool装置是完成CANbus检测的工具。为了实现CAN数据的显示及控制，需要使用本文提出的CanToolApp软件。

### 1.3项目意义

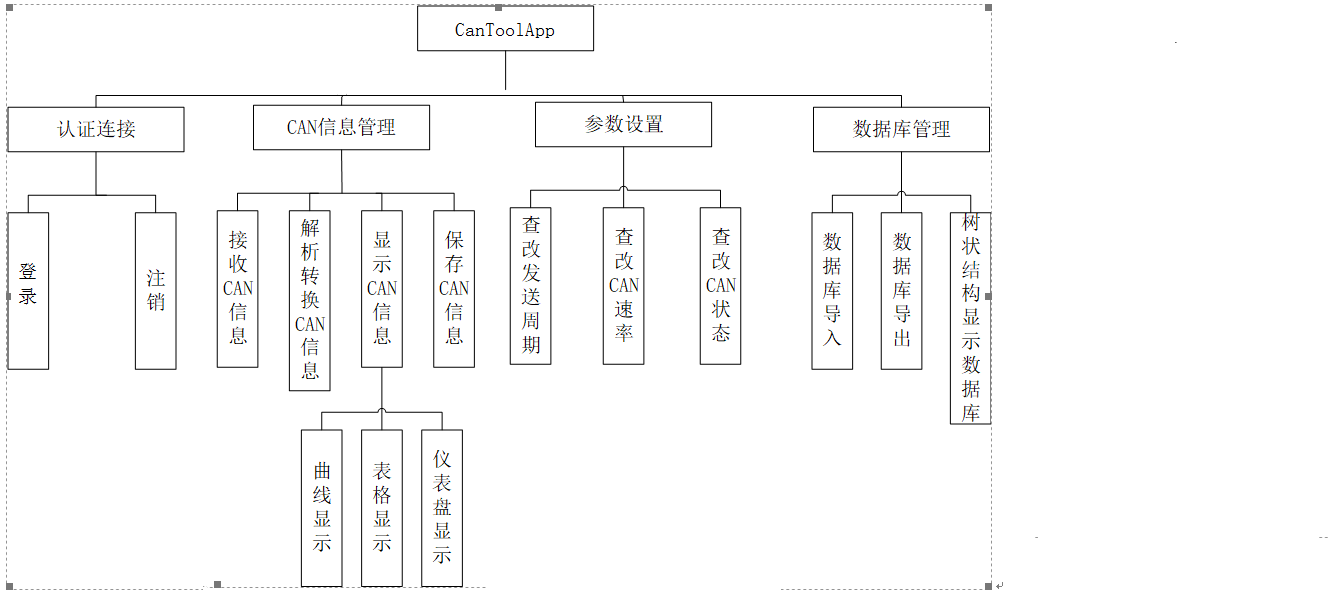
本文提出一个接收CAN信息（这些信息是在连接在CAN总线上的CanTool装置采集的）的App软件，并且能够将这些信息进行解析通过不同方式显示在用户图形界面上。同时在CanToolApp的界面上还可以设定CAN信息，通过GUI按钮将信息发送给CanTool装置，CanTool装置将按照规定的信息格式，将信息发送到CAN总线上。此外，CanToolApp可以设定CAN总线的通信参数，并通过相应的命令设置CanTool装置的CAN通信参数，以使CanTool装置能够与CAN总线上的其他被测ECU进行正常的通信。

# 2需求规定

### 2.1任务概述

该CanTool管理系统主要完成认证连接、参数设置、CAN信息管理、数据库管理四个模块，实现用户接入设备、注销连接、查改CAN信息发送周期、查改CAN速率、查改CAN状态、接收CAN信息、解析转换CAN信息、通过曲线显示实时CAN信息、通过表格和仪表盘显示CAN信息、保存CAN信息、数据库导入导出、通过树状结构显示数据库结构等功能。

### 2.2功能模块图



### 2.3对具体功能的规定

（1）认证连接模块

用户能够选择不同的CanTool装置映射到本机的接口（COM口），通过设置相应COM口波特率、数据位数、停止位数等这些参数来进行与装置的连接。并且这些设定内容会自动保存到CanToolApp设定文件中，供下次使用。

（2）参数设置模块

用户可以设置CanTool装置的CAN速率、更改CAN状态（Open or Close）、设置CAN信息的发送周期。并且可以将这些设定内容保存到CanToolApp设定文件中，供下次使用。

（3）CAN信息管理模块

系统可以接收多个CAN信息，通过CAN信息及CAN信号数据库进行解析，还原出真实物理值，并保存数据到数据库。用户可以选择表格或仪表盘形式显示CAN信息原始数据和其包含的物理值（系统会记录用户选择的显示方式，供下次使用），还可以选择某些接收到的CAN信号，显示其变化的实时物理值曲线。

（4）数据管理模块

用户可以选择将系统中的CAN信息和信号数据库另存为xml和JSON (JavaScript Object Notation)格式导出，也可以将已有的xml或Json格式的数据库导入系统，转换为CAN信息和信号数据库格式。系统可以加载用户提供的CAN信息和信号数据库，以树状结构显示。

# 3运行环境规定

### 3.1硬件要求

CPU：Intel(R) Core(TM) i5-4200U 及以上版本；

内存(RAM)：至少2.00GB；

硬盘 2G以上

### 3.2支持软件

（1）操作系统：兼容window7/8,Linux系统；

（2）浏览器：IE 9及以上版本。

### 3.3接口

外部接口 Arduino nano数据访问接口。

内部接口 Data Access Object数据访问接口。

### 3.4开发环境

**开发语言：**Java

**运行环境：**JDK 1.8+ 7.0.29

**数据库：**MySQL

### 3.5 控制要求

该网站采用目前的技术，后台包含运行两个线程，实现对cantool信息的接受保存以及对实数数据的展示。要求系统启动后线程进入待命状态，系统接入cantool装置后线程启动，系统退出接入后线程在再次进入代码状态。

# 4、具体需求

## 4.1 功能需求

在现代汽车控制技术中，汽车中会使用多个电子控制装置（ECU：Electronic Control Unit）对整车进行控制。而ECU之间的信息交换更多地依赖于CAN(Controller Area Network)总线的网络连接方式来完成。为了检测和控制CAN bus的信息内容，需要使用CAN bus检测设备。CanTool装置是完成CANbus检测的工具。为了实现CAN数据的显示及控制，需要使用本文提出的CanToolApp软件,该软件需要将连接在CAN总线上的CanTool装置采集的CAN信息发送到上位机（移动终端Android、iOS、Windows PC）上。 通过对需求文档进行整理与讨论，我们将Cantool APP的功能主要分为以下四个模块,如下图：

CanToolApp对CAN信息显示处理功能：

1. 显示数据分为3种：

* 从CanTool装置接收的数据
* 用户发送的数据
* 加载的用户提供的数据

2. 显示“接收”数据的处理操作:

* 接受到的数据需要根据CAN信息及信号描述数据库对数据字符串进行解析,然后得到CAN信息中包含的各种CAN信号值。
* 将此CAN信号值进一步进行计算,还原该信号所代表的物理量的信息，并显示在GUI界面上。

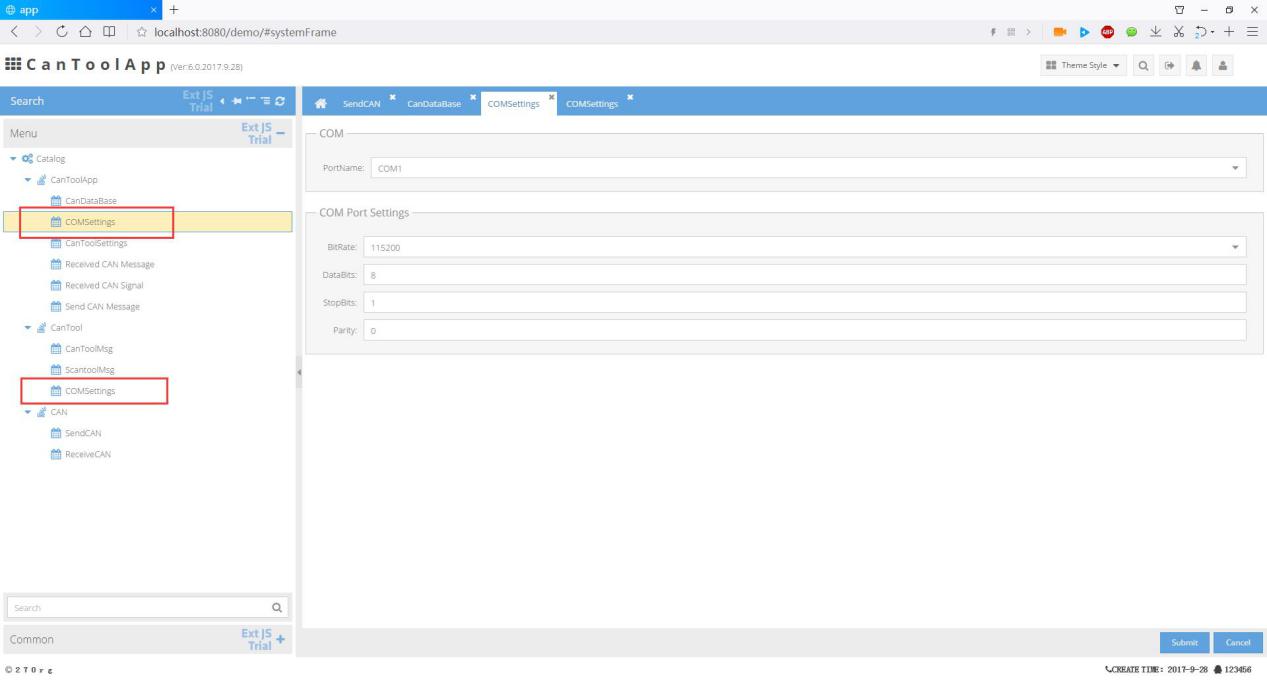
1. 显示“发送”数据的处理操作:

* 发送的数据需要采用同种方式，将用户输入的物理值转换为CAN信号。
* 依据CAN信号描述数据库将属于同一个CAN ID的信号合成为字符串发送给CanTool装置。

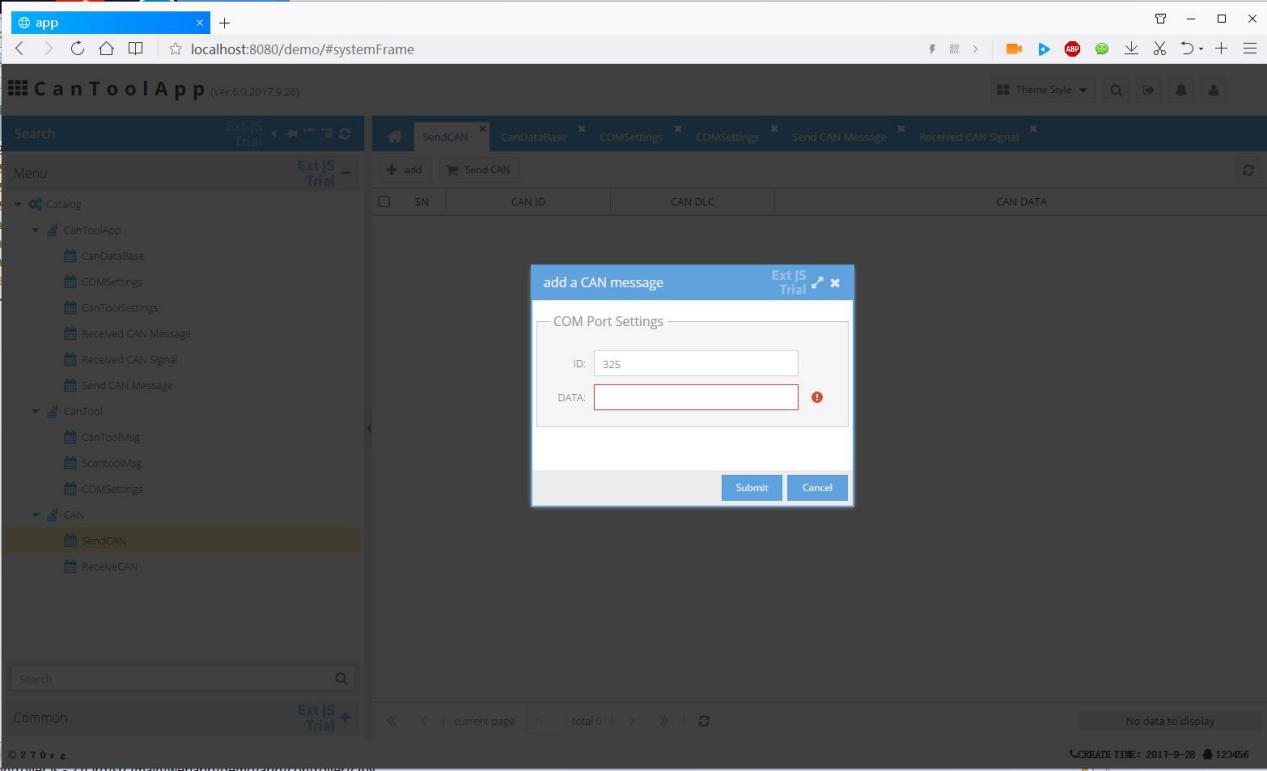
4、显示“加载”数据的处理操作，完成CAN信号数据的解析以及CAN发送信息的组装，同“发送”过程。

CanToolAPP GUI界面显示：

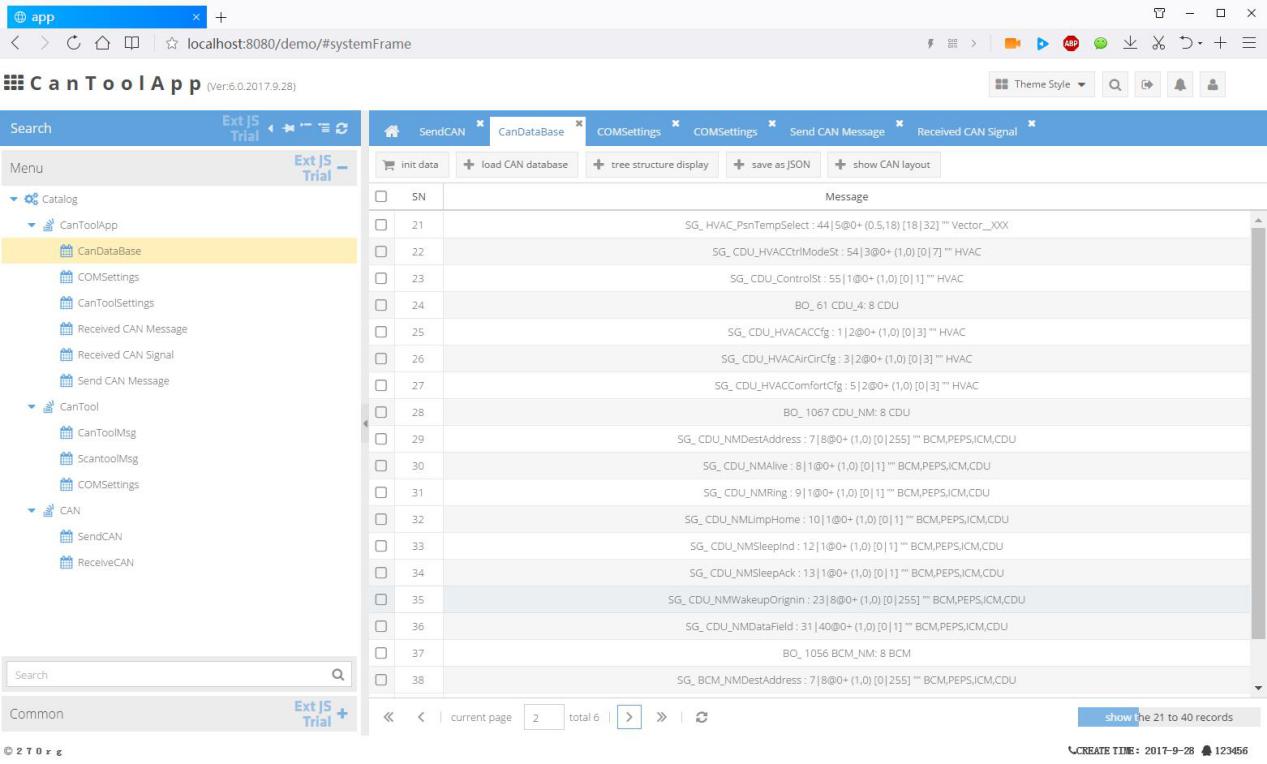
1.COM口基本设定信息界面



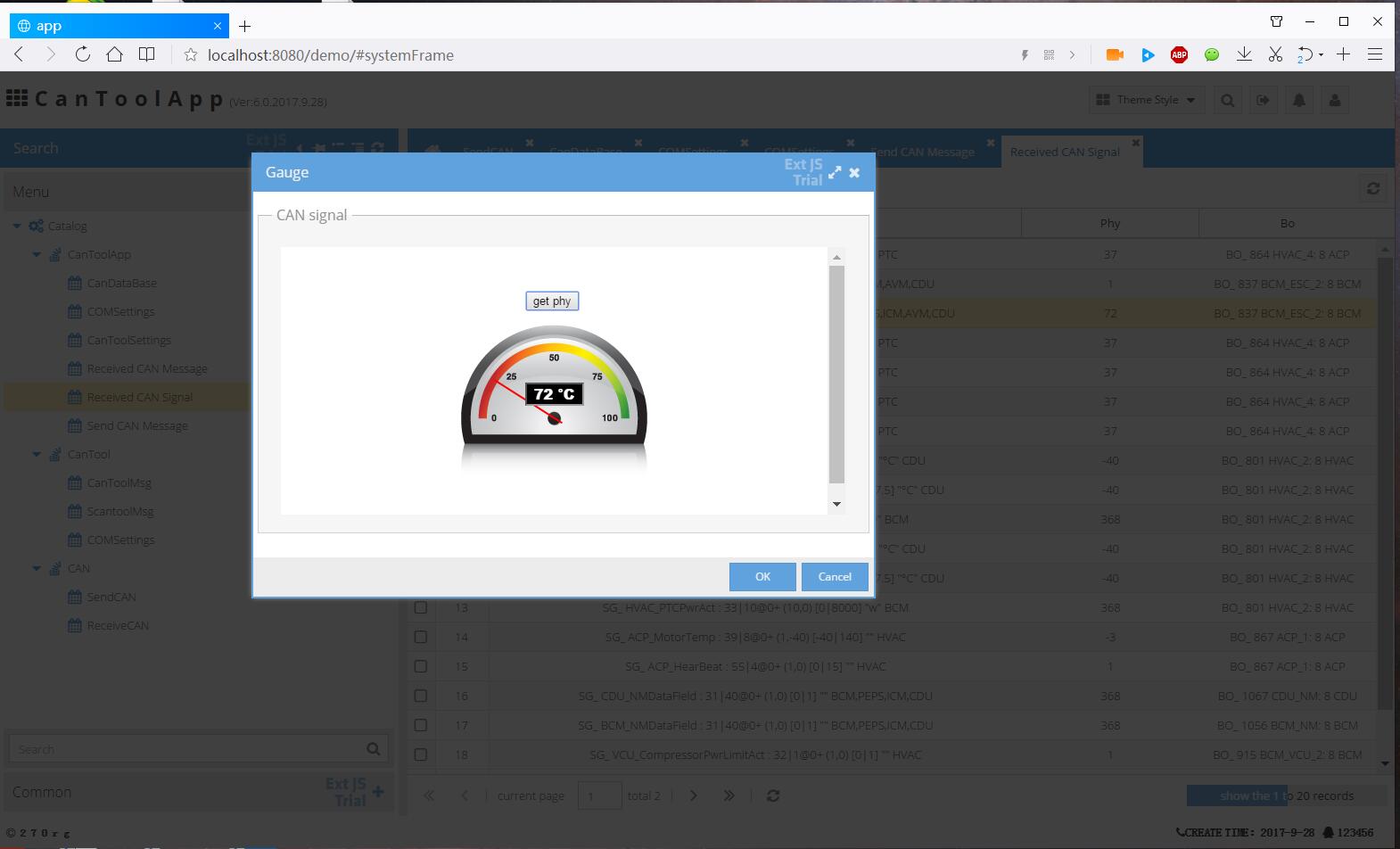
2.Can基本设定信息界面



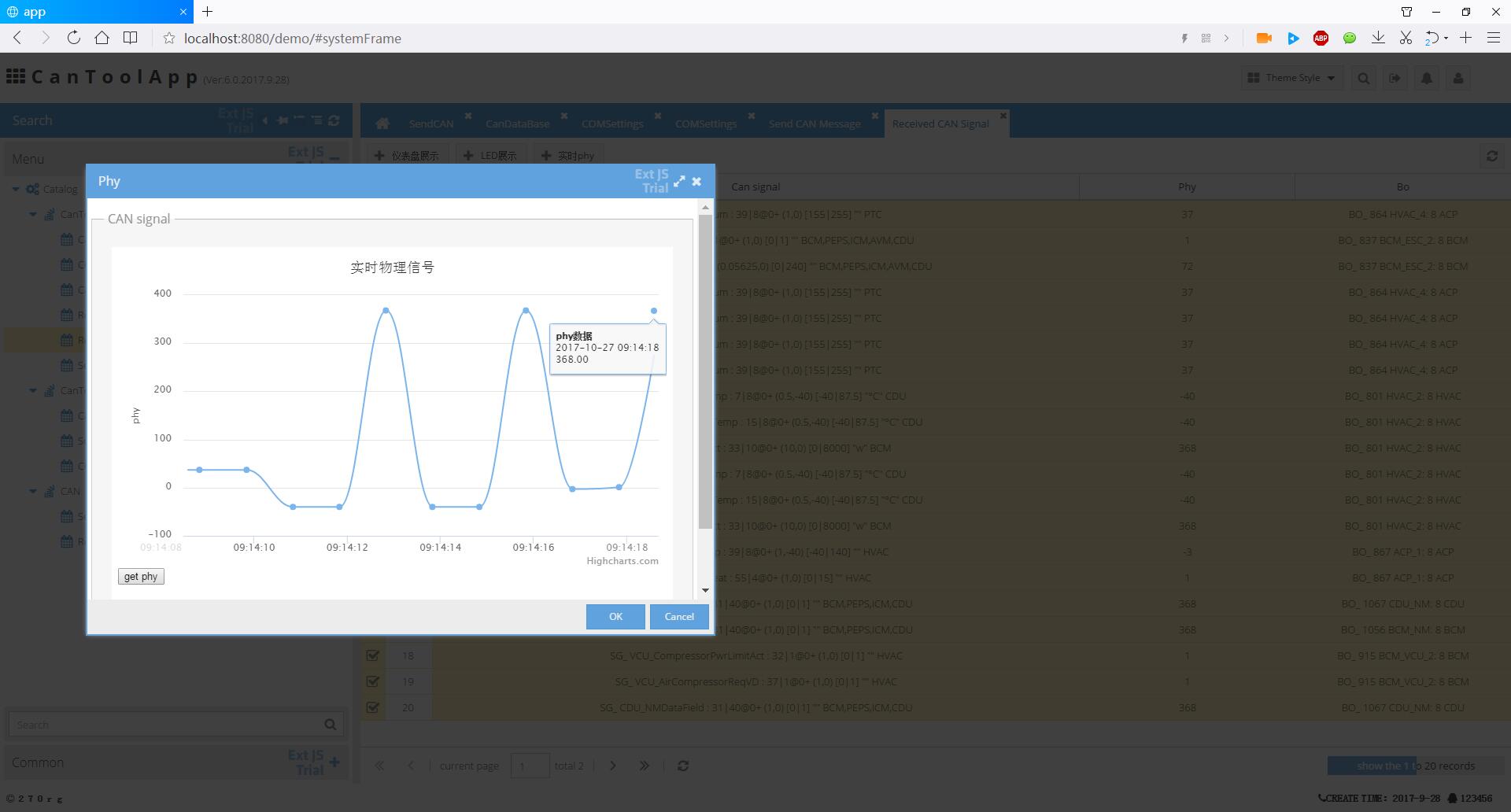
3.Can信息原始数据及物理数据实时显示界面



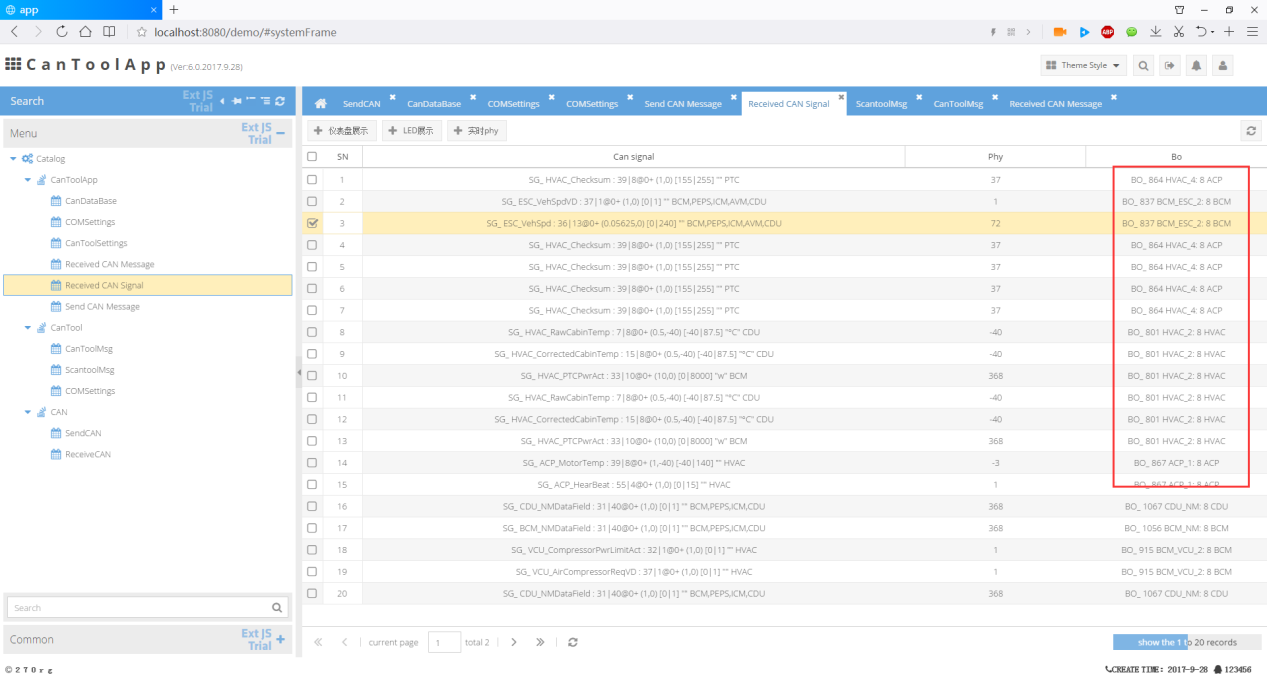
4.Can信号物理值的仪表盘,LED显示。



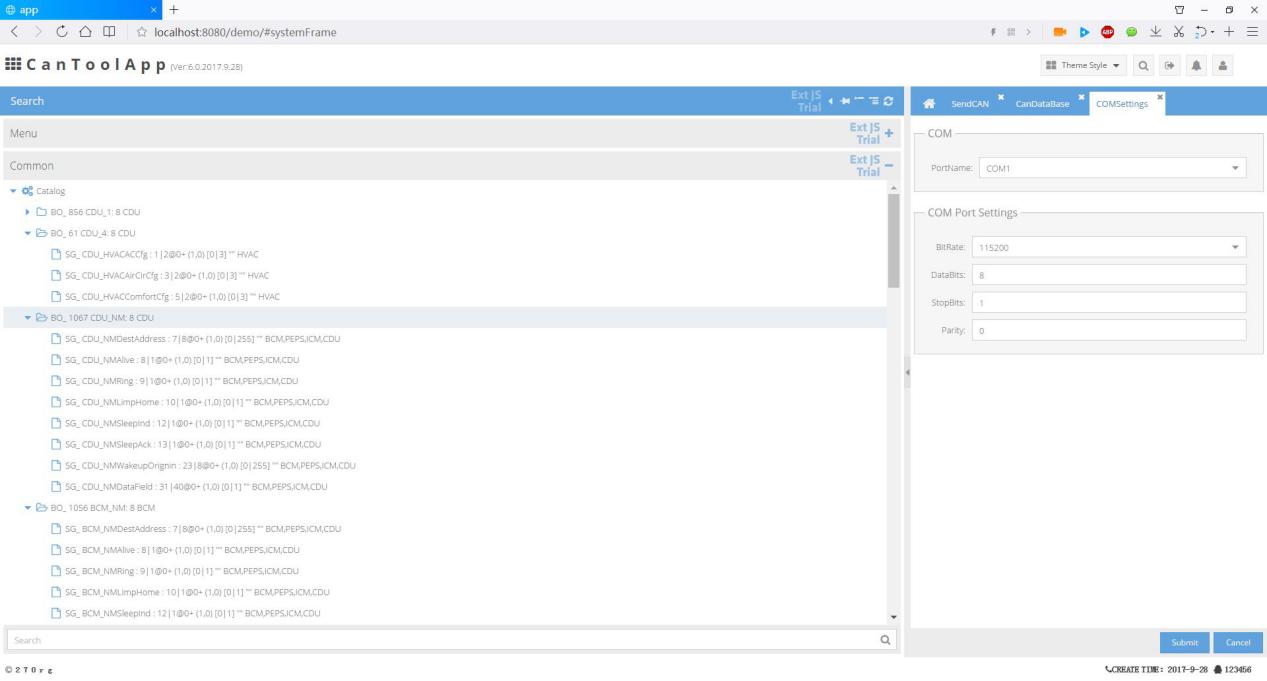
5.Can信号实时曲线



6.Can信号在Can信息中的布局图

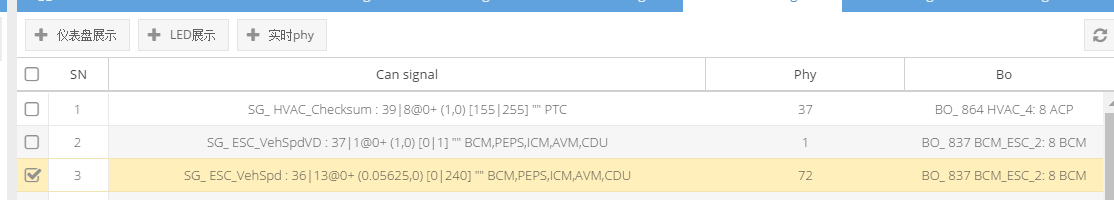


7.Can信号树状结构图

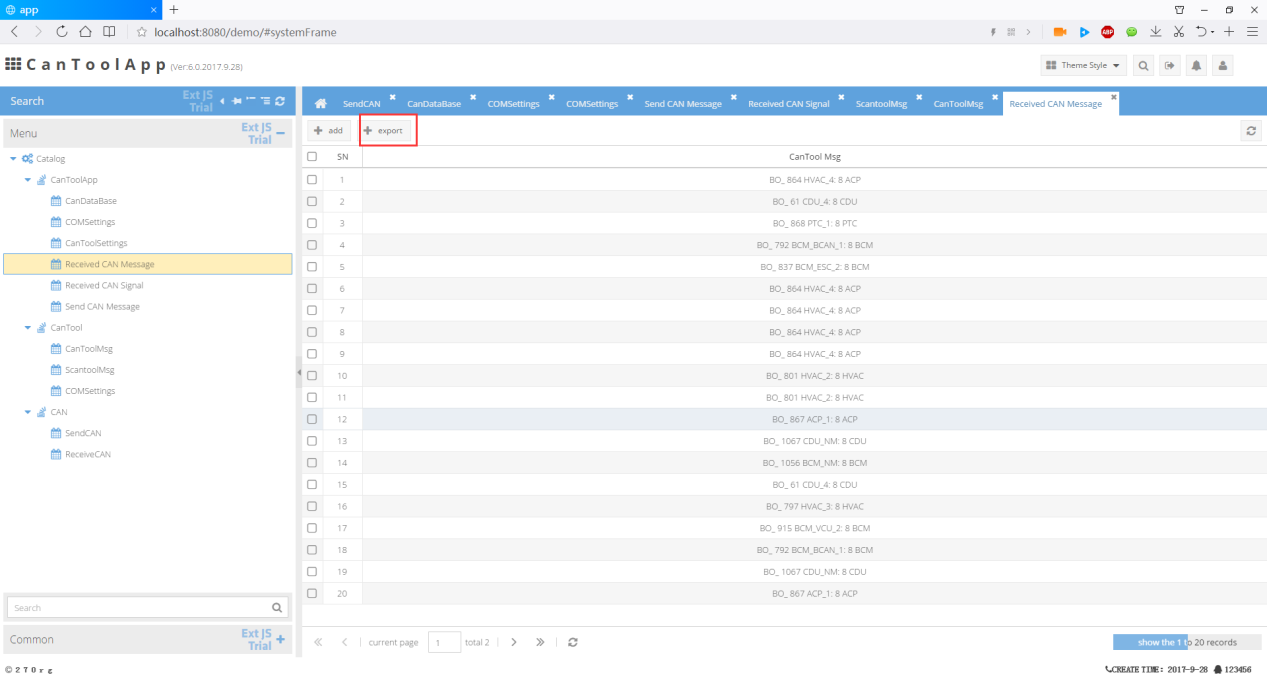


CanToolAPP 的数据存储功能：

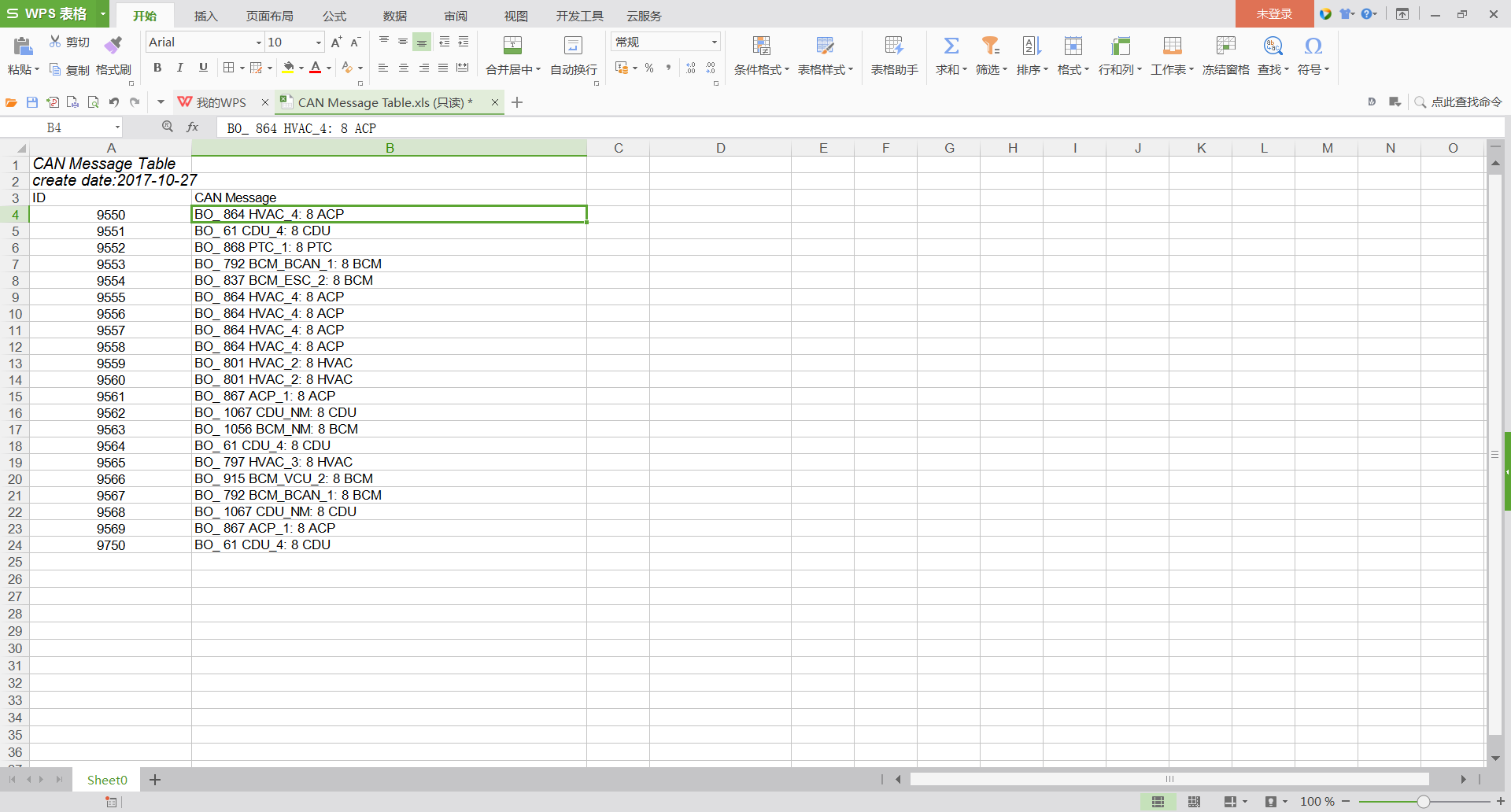
1.用户选择的显示方式可保存到CanToolApp设定文件中，供下次使用。



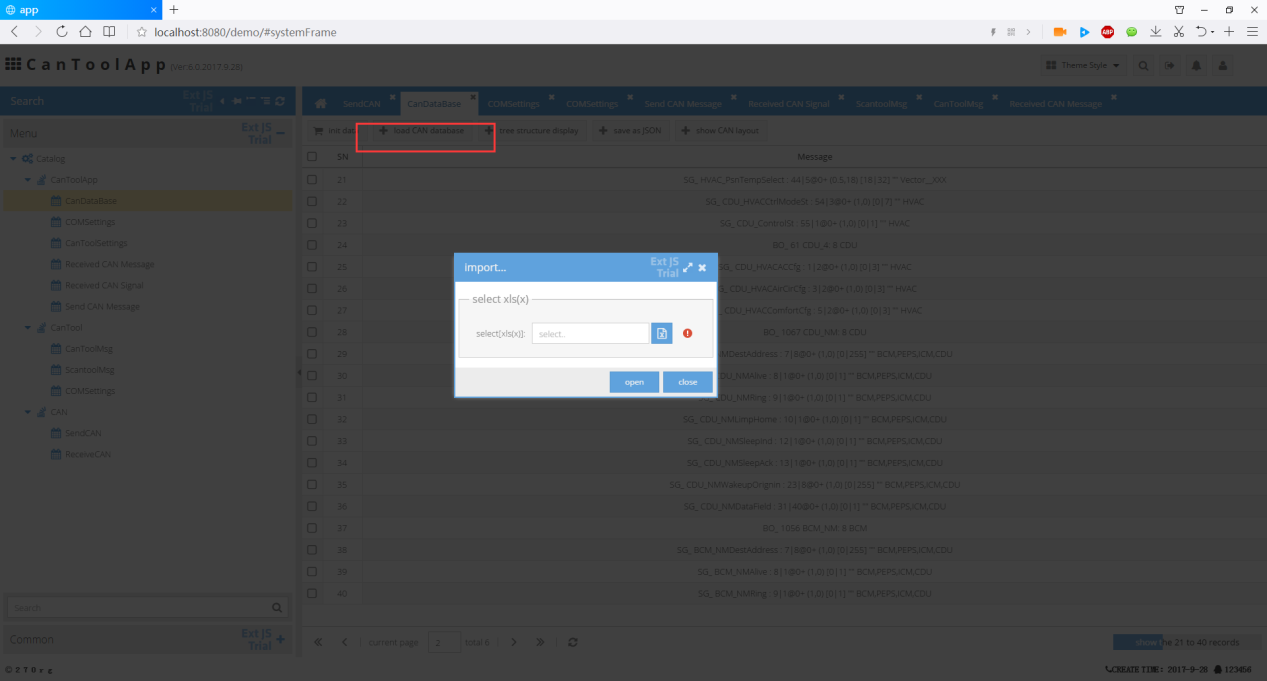
2.可以将接收到的所有CAN信息数据，实时保存为数据文件。格式为CSV格式，或自定义。



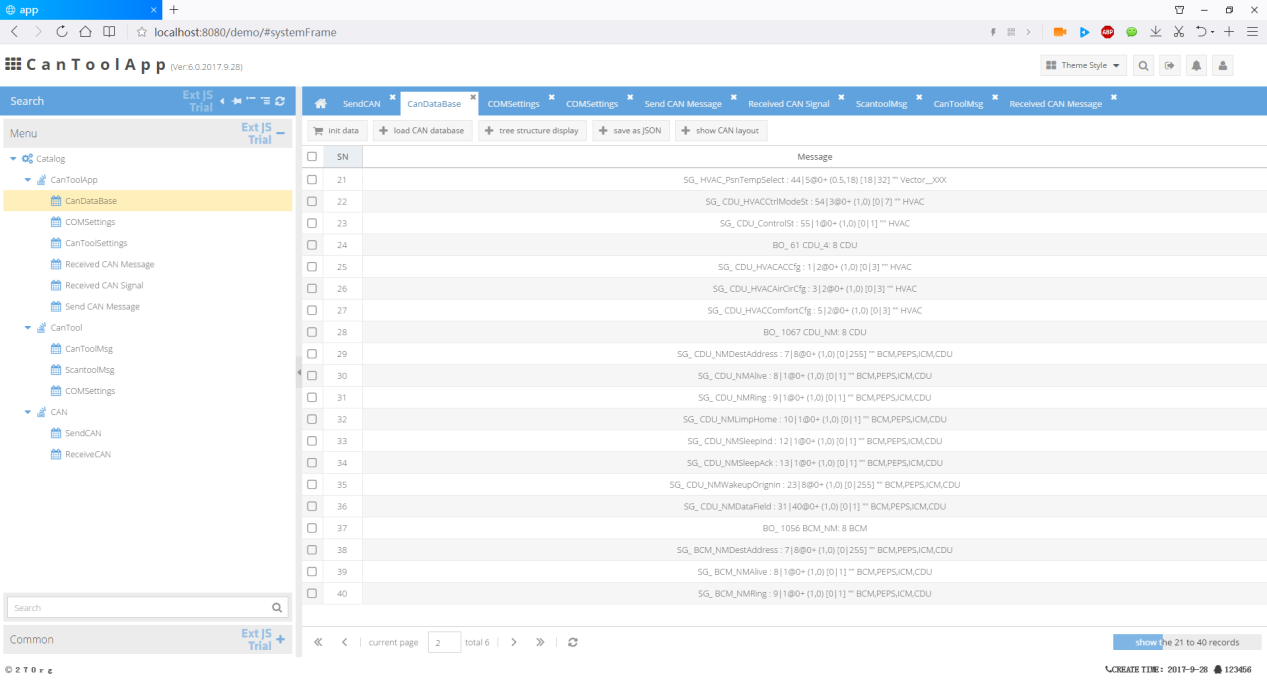
存储结果：



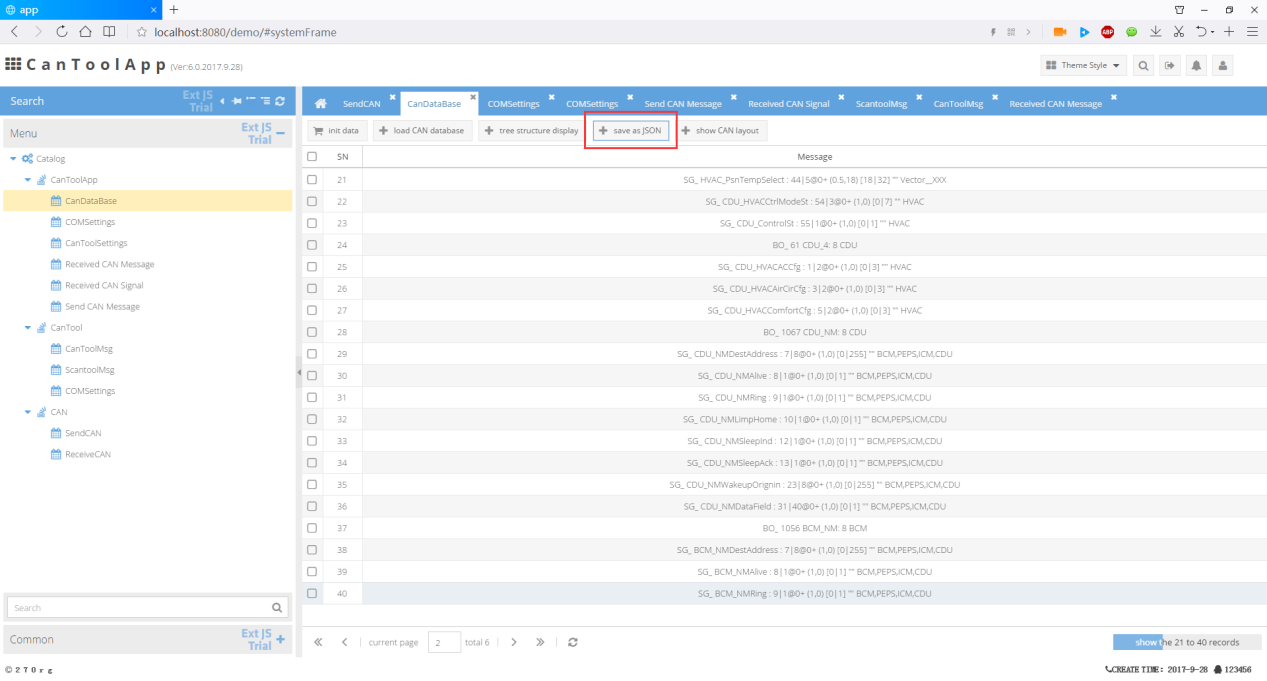
3.可以加载用户提供的CAN信息和信号数据库，加载的数据库文件相关信息，可保存到CanToolApp设定文件中，供下次使用。



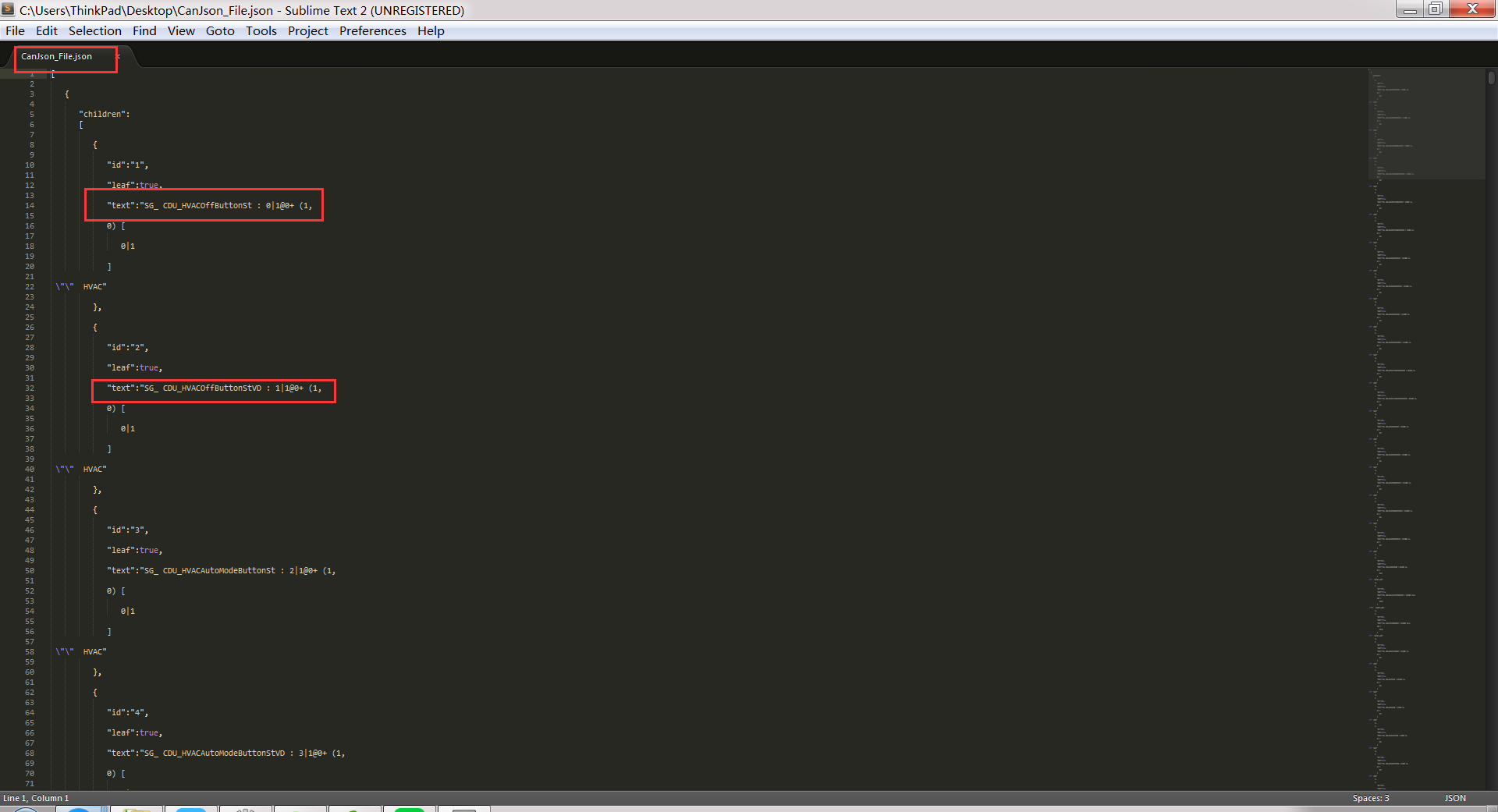
结果：



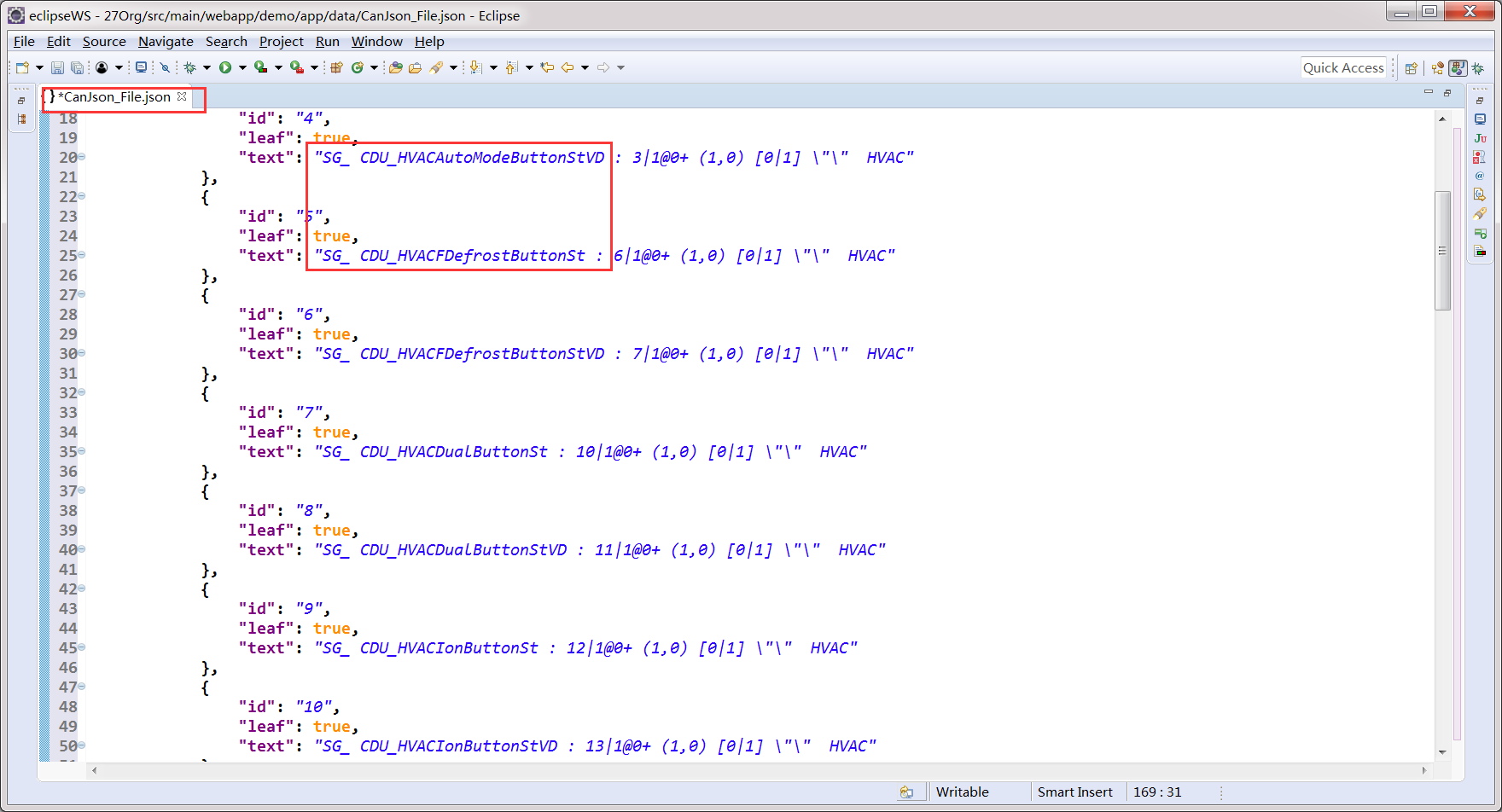
4.可以将用户提供的CAN信息和信号数据库另存为xml和JSON (JavaScript Object Notation)格式。

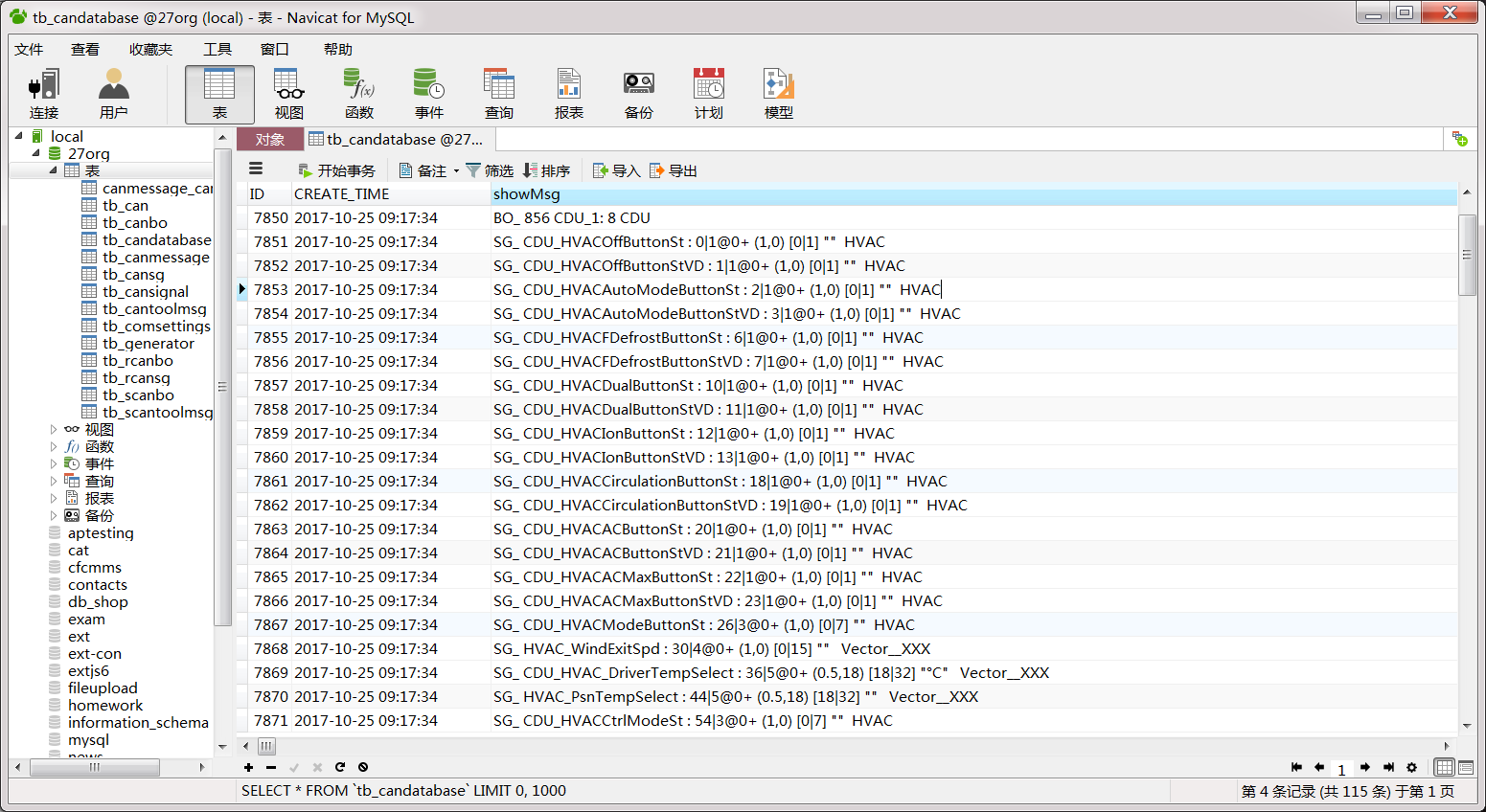


结果：



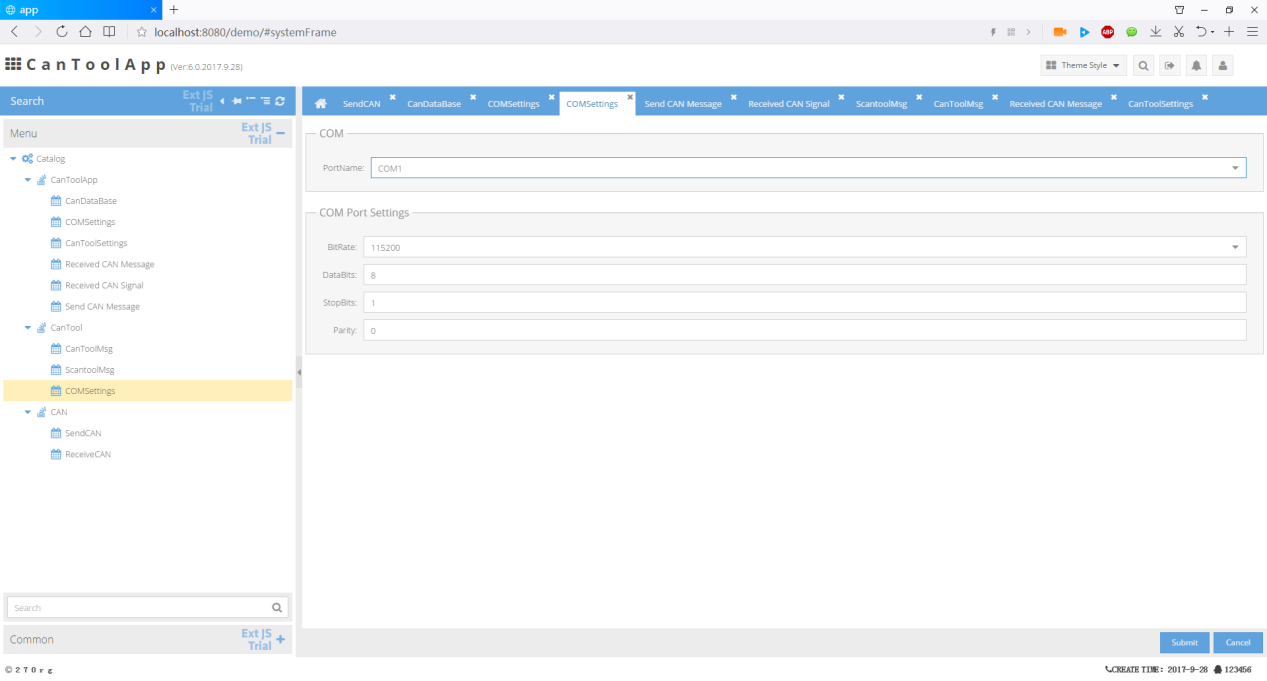
5.可以将xml或Json格式的数据库，转换为CAN信息和信号数据库格式。

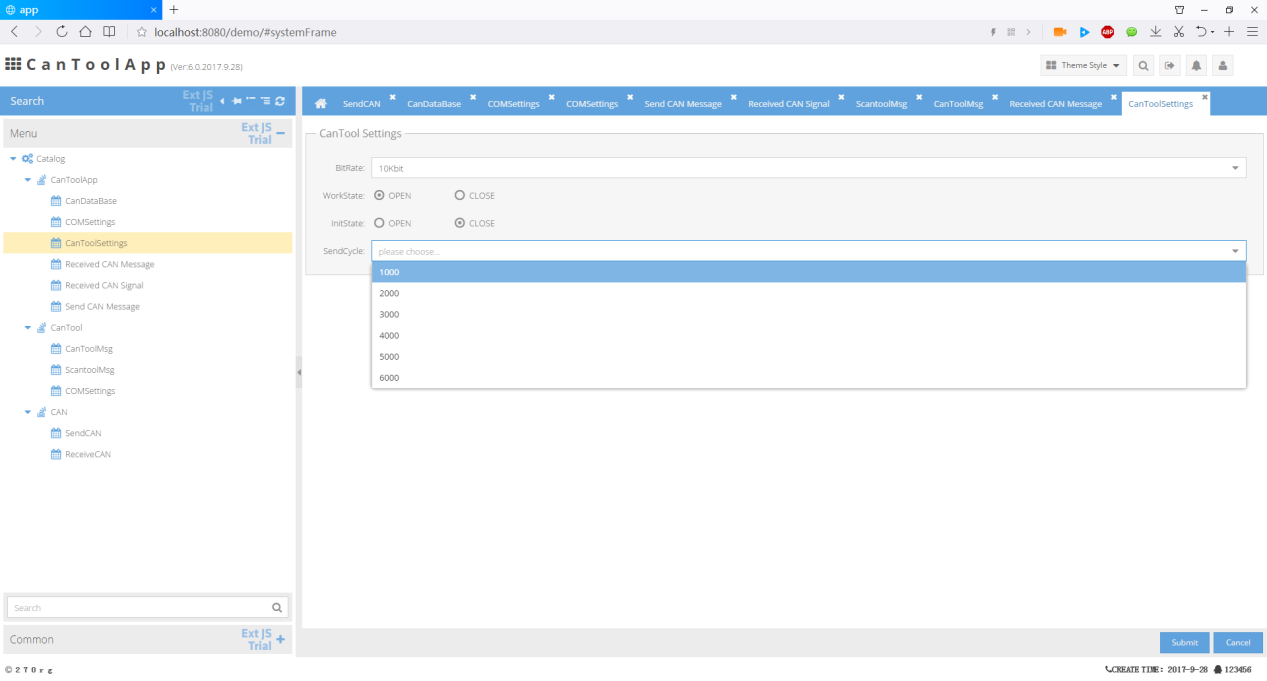




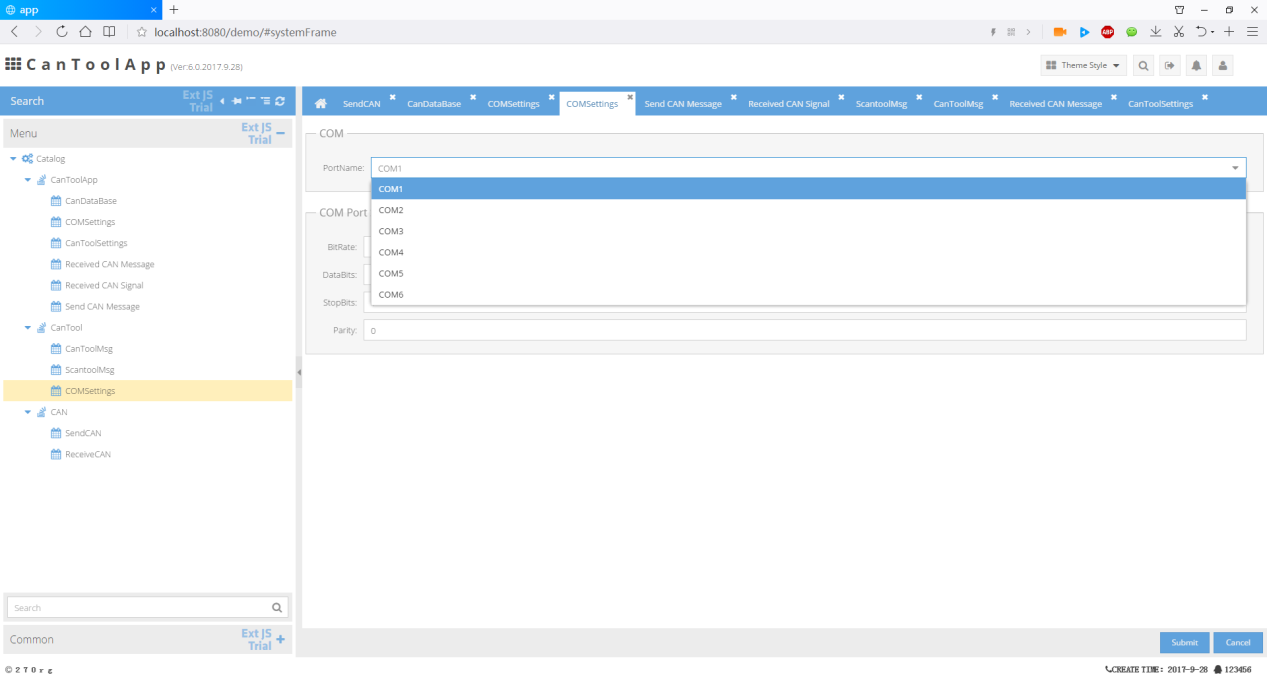
CanToolAPP 通过控制CanTool装置控制Can通信功能：

1.CanToolApp可以设定CAN总线的通信参数，并通过相应的命令设置CanTool装置的CAN通信参数，以使CanTool装置能够与CAN总线上的其他被测ECU进行正常的通信。

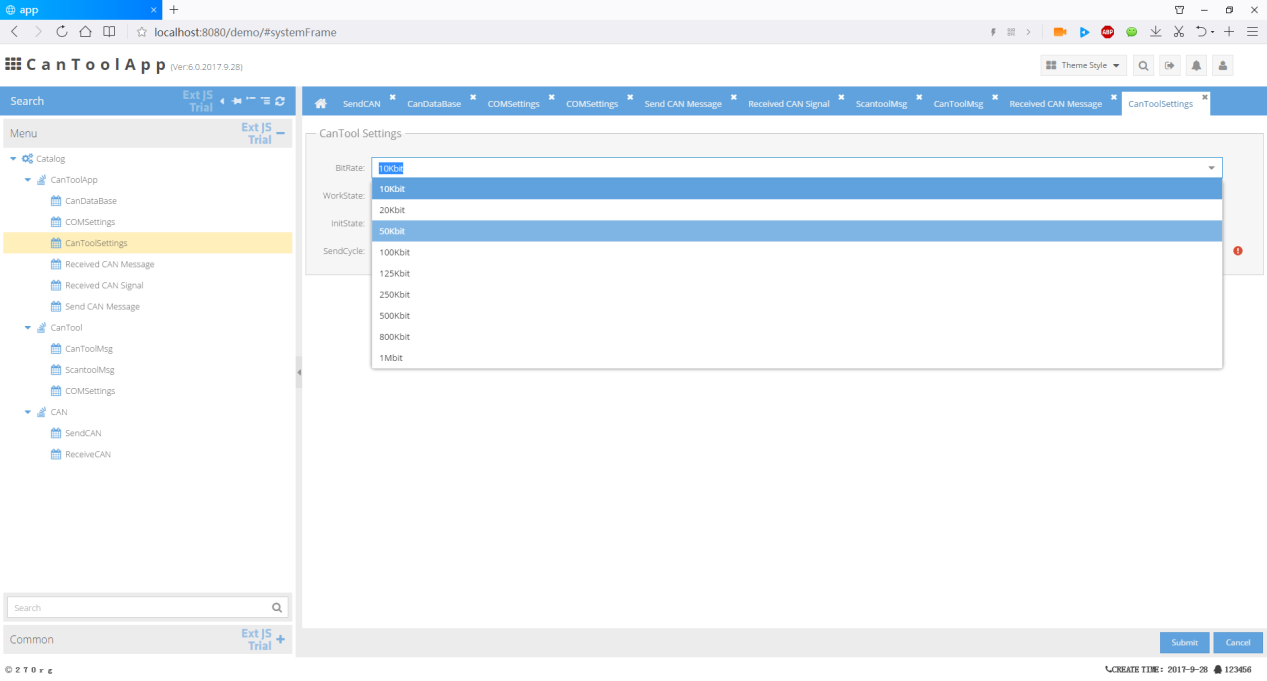


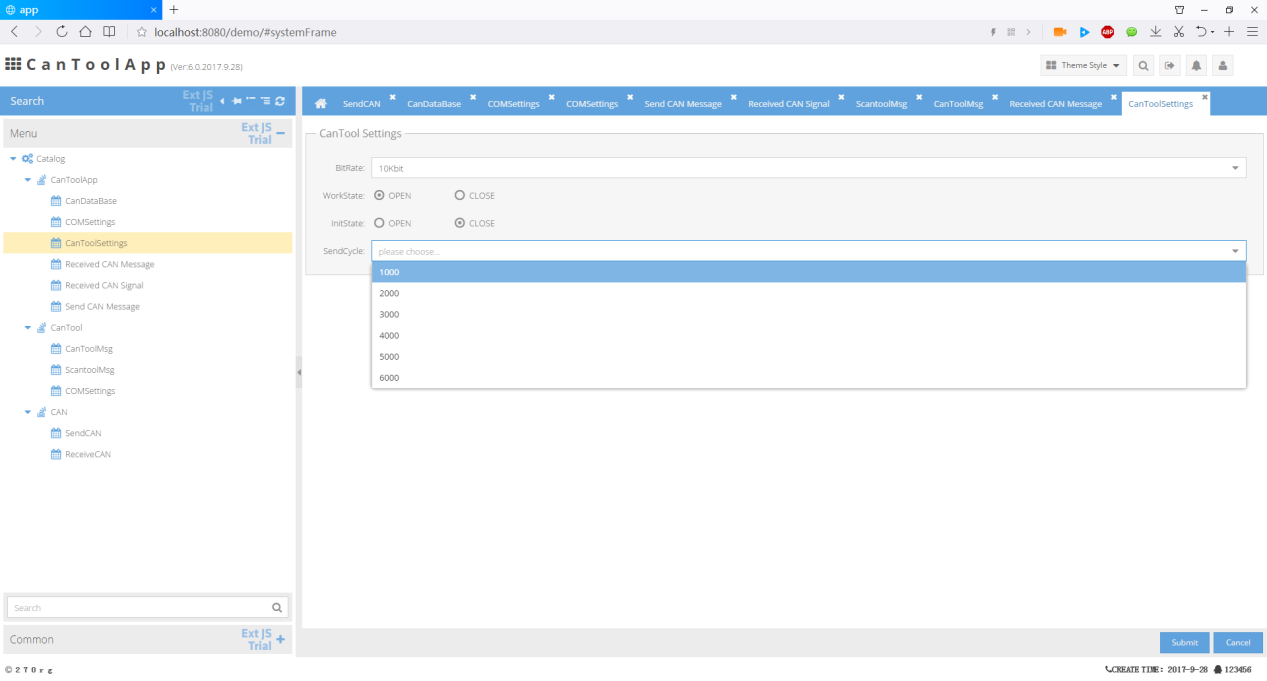


2.能够搜索到本机所有可使用的COM口，并在弹出式ComboBox中以列表方式让用户选择CanTool装置在上位机中映射的COM口。并设置相应COM口波特率115200、数据位数8、停止位数1。

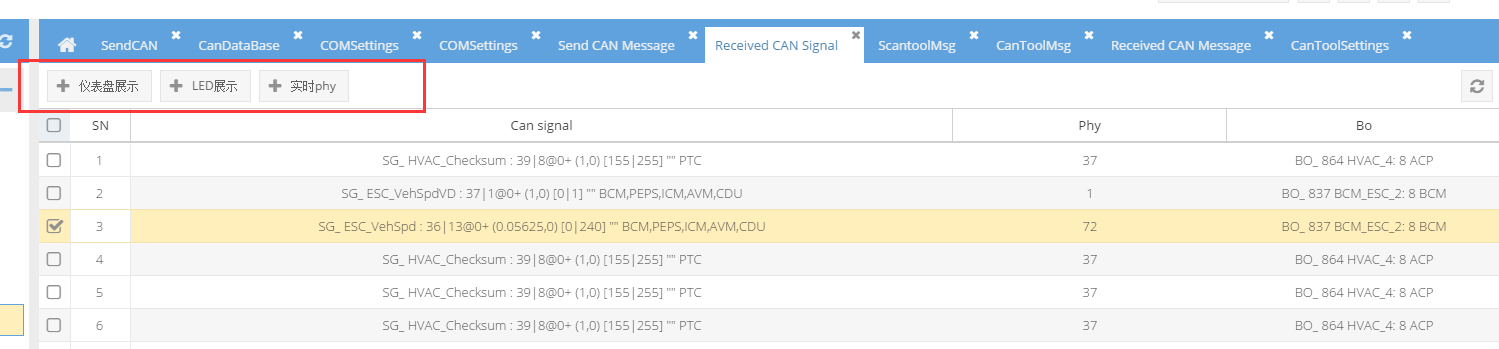


3.能够实现CANtool装置的CAN速率设置、进入CAN工作状态（Open）、进入CAN初始化状态（ Close）。

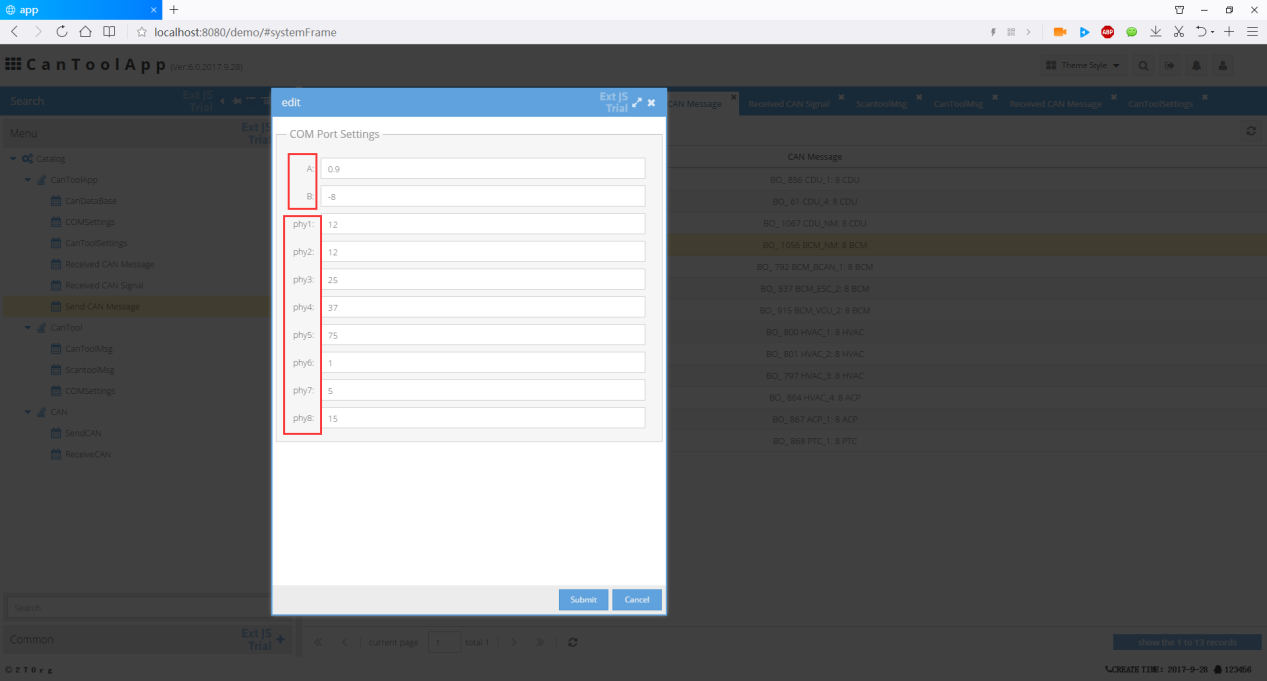




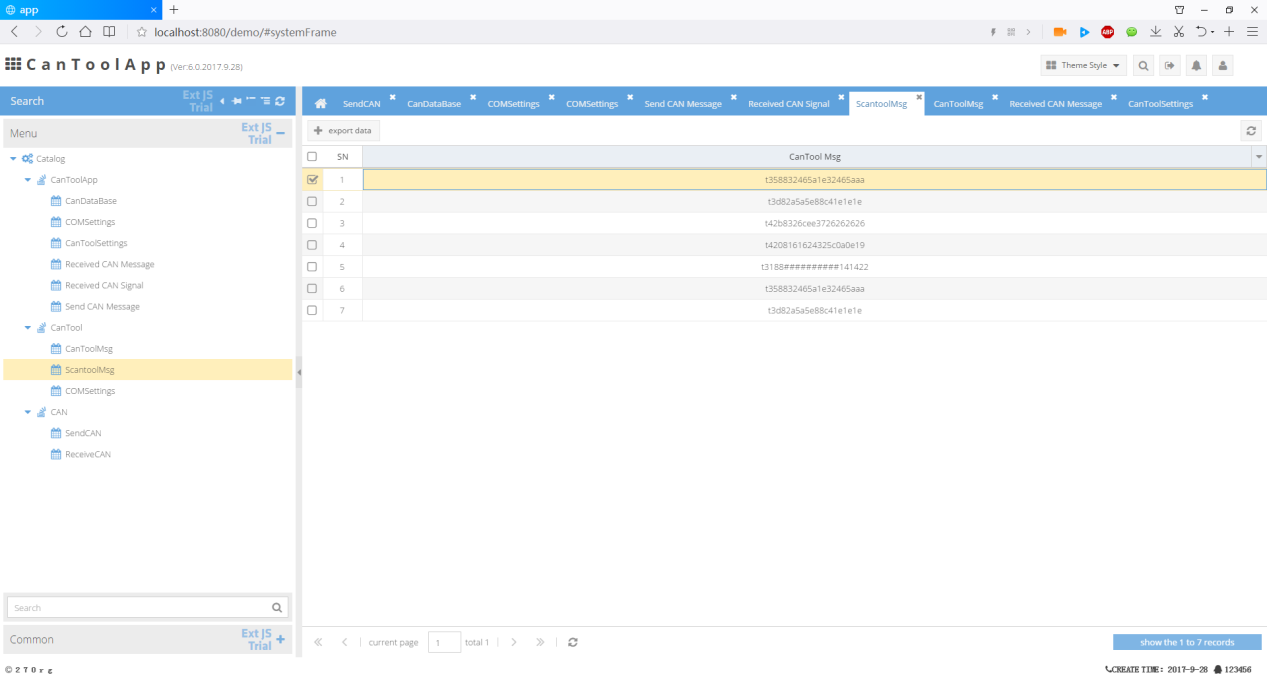
4.控制显示方式,显示时可以让用户选择仪表盘方式显示接收到CAN信号物理值。



5.允许用户设定CAN信息中的CAN信号物理值,App可将用户设定的物理值转换为CAN信号值，将CAN信息中包含的所有CAN信号合成完整的CAN信息后，发送给CanTool装置，发送到CAN总线上。



发送结果：



## 4.2 CAN信息数据格式

表 1 “CAN信息Message描述数据库”格式

| 字段名 | 类型 | 样例 | 说明 |
| --- | --- | --- | --- |
| CANmessage标识符 | char[32] | BO\_ | 固定为BO\_ |
| id | uint32 | 2148606241 | 十进制数值2148606241转换为16进制为0x80112121，其中msb=bit31=1表示是CAN扩展帧，bit28~bit0是实际的CANID值=0x00112121 |
| 100 | 十进制数值100转换为16进制为0x00000064，其中msb=bit31=0表示是CAN标准帧，bit10~bit0是实际的CANID值=0x013 |
| Message Name | char[32] | EngMsg1 | 字符串，最长32字节 |
| 分隔符 | char[1] | : | 固定为 : |
| DLC | unsigned char | 8 | 范围:0—8, 表示此CAN信息的DATA的长度为8byte |
| Node Name | char[32] | BODY\_ECU | 字符串，最长32字节,发送此信息的Node名。也是ECU名 |
| 注：每个字段间间隔符除注明外，均使用空格作为分隔符，上述样例在数据库中保存的格式为。BO\_开始于新的一行第0列，信息的末尾为”\n”即DOS换行符”0x0d 0x0a”。  BO\_ 2148606241 Ext1: 8 Node\_Body | | | |

表 2 “CAN信号Signal描述数据库”格式

| 字段名 | 类型 | 样例 | 说明 |
| --- | --- | --- | --- |
| CAN signal 标识符 | char[32] | SG\_ | 固定为SG\_ |
| Signal Name | char[32] | EngMsg1 | 字符串，最长32字节 |
| 分隔符 | char[1] | : | 固定为 : |
| 起始位| bit长度@bit格式 | unsigned char[10] | 23|12@0+ | 23表示起始位编号  12 表示此CAN信息的DATA的长度为12bit  0+表示数据按Motorola的bit顺序排列,数据最高位MSB放在bit编号23的位置，即23bit为MSB,bit28为LSB，数据是左对齐的，即放置不在bit编号bit23,bit22…bit16; bit31,bit30,bit29,bit28的位置，注意此编号还是规定使用intel格式时的编号。从数据库的编号开始从左到右，并向高位数据字节依次排列MSB🡺LSB |
| 16|12@1+ | 16表示起始位编号  12 表示此CAN信息的DATA的长度为12bit  0+表示数据按Intel的bit顺序排列,数据最低位LSB放在bit编号16的位置，即16bit为LSB,bit27为MSB，数据是右对齐的，及从CAN数据库的编号开始依次从右向左,并向高位字节放置信号的LSB🡺MSB。 |
| (A,B) | Double | (0.1, -10) | A:分辨率，1LSB的物理值精度，B：物理值的偏移量offset。  Phy=A\*x+B, x为CAN信号的数值，phy为CAN信号对应的物理值 |
| [C|D] | double | [-41|368.5] | 物理值的范围：Min=C到MAX=D |
| 物理单位 | Char[32] | “℃” | 带有双引号的字符串，可以为空:””, |
| Node Name | char[255] | BODY\_ECU | 接收该信号的节点Node名列表（也是ECU名）字符串，最长32字节。如果多个ECU接收此信号，则用逗号将多了节点名隔开，例如：BCM,PEPS,ICM,CDU |
| 注：每个字段间间隔符除注明外，均使用空格作为分隔符，上述样例在数据库中保存的格式为。SG\_开始于新的一行第1列，信息的末尾为”\n”即DOS换行符”0x0d 0x0a”。  SG\_ sig1bittable : 8|1@1+ (1,0) [0|1] "" AC\_ECU  SG\_ Sig\_moto1 : 7|12@0+ (0.1,-41) [-41|368.5] "DegC" Node\_Body  SG\_ CDU\_NMWakeupOrignin : 23|8@0+ (1,0) [0|255] "ms" BCM,PEPS,ICM,CDU | | | |

**4.3属性**

### 4.3.1 可用性

* 界面优美，操作简单。当涉及“删除”等危险操作时，系统会先给予确认提示
* 并发性：能够保证一定数量的用户同时在线使用
* 容错能力：系统具有一定的容错能力，非硬件或通讯故障均能保持运行

### 4.3.2 安全性

* 故障处理：系统应能够记录错误信息、生成错误日志，并及时恢复现场
* 访问控制：对普通用户设置访问权限，防止访问越界
* 安全存储：数据库保存在服务器端，定期备份，确保数据不丢失

### 4.3.3 可维护性

* 使用 Github 进行源码管理，保留各版本的源代码
* 文档及代码均遵循事先预定的规范
* 提供用户反馈渠道，确保及时发现和修复 Bug

# 5、验收标准

## 5.1 文档验收标准

  文档编写符合国际文档编写规范。

* 软件需求规格说明书
* 项目开发计划书

## 5.2 软件验收标准

  软件一切功能正常，运行流畅，不卡顿，不闪退。

* 网页端
* 服务器端（用于支持网页正常运行）

## 5.3功能验收标准

  需求四象限及各阶段验收标准如图：