信息技术与系统

基于 Simulink 自动代码生成技术的 CAN 底层模块库设计

汪　伟，莫官旭，申　健，谢勇波，王文明
（长沙中车智驭新能源科技有限公司，湖南 长沙　410036）

摘　要：针对汽车控制器 CAN 通信报文数据多、传统软件开发方式效率较低的问题，文章提出一种基于 Simulink 自动代码生成技术的 CAN 底层模块库设计方法，其采用 S-Function 设计配置 CAN 通道信息模块和可导入 DBC 文件的 CAN 报文收发模块，编辑 TLC 文件，并使用 RTW（real-time workshop）实现 Simulink 模块自动代码生成。在 Infineon TC21x 系列芯片控制板上进行测试，验证了基于自动代码生成的 CAN 底层模块库的可用性和高效性。测试结果表明，采用该方法设计的 CAN 底层模块库能够在控制板上有效运行，并达到了预期结果，提高了软件开发效率且有利于软件后期维护和功能扩展。

关键词：自动代码生成；CAN 通信；软件复用
中图分类号：U463.6　　**文献标识码：**A　　**文章编号：**2096–5427(2020)03–0093–04
doi:10.13889/j.issn.2096–5427.2020.03.019

Design of CAN Underlying Module Library Based on Simulink Automatic Code Generation Technology

WANG Wei, MO Guanxu, SHEN Jian, XIE Yongbo, WANG Wenming
(Changsha CRRC Intelligent Control and New Energy Technology Co., Ltd., Changsha, Hunan 410036, China)

Abstract: Traditional software development methods for CAN communication in automobile controller have the problem of large message data and low efficiency. This paper proposed a design method of CAN underlying module library based on Simulink automatic code generation technology. S-Function is used to design a configuration CAN channel information module and CAN message transceiver module that can import DBC files, edit TLC files, and use RTW to implement automatic code generation of Simulink modules. The availability and efficiency of the CAN underlying module library based on automatic code generation were tested and verified on the Infineon TC21x chip control board. The test results show that the CAN underlying module library designed by this method can run effectively on the control board and achieve expected results, which improves the development efficiency and is conducive to software maintenance and function expansion.

Keywords: automatic code generation; CAN communication; software reuse

0　引言

汽车控制器之间的通信主要通过 CAN（controller area network）总线实现。随着汽车功能的不断丰富，控制器数量逐渐增多，CAN 报文信息交互频繁，所需处理的数据越来越多；而传统软件开发方式存在开发时间长、代码量大、易出错和不易扩展的不足。另外，在基于模型设计[1-2]的软件开发中，模型生成代码的变量不便与 CAN 底层函数的参数关联，需将模型生成代码中的变量一一赋值至手写代码的底层函数中使用。

基于模型的设计是嵌入式软件快速开发的一种方式，在 Simulink 环境下使用模块库搭建控制策略模型，使用 RTW(real-time workshop) 将模型生成代码，再编译烧写到嵌入式控制器中运行。相比编辑 C 代码的嵌入式软件开发方式，基于模型的设计可视性更高、移植性更强，并且可在 Simulink 上对逻辑进行仿真，简化测试工作。若通过 Simulink 模块实现 CAN 报文收发、装载与解析功能，则能有效提高开发效率和代码可靠性，增强软件移植性[3-4]。据此，本文使用 Simulink 的 S 函数（S-Function）并编写相应的 M 语言脚本和由目标语言编译器（target language

收稿日期：2020–02–07
作者简介：汪伟（1978—），男，高级工程师，研究方向为新能源汽车动力系统集成、匹配与控制。
基金项目：国家重点研发计划（2018YFB1201602，2018YFB1201604）

compiler, TLC）转换的 TLC 文件，设计了一种基于自动代码生成技术的 CAN 底层模块库，用于实现控制器的 CAN 通道初始化和 CAN 报文的接收与发送、解析与装载。

1 模块功能设计

在 Simulink 环境中，需要设计 S-Function 和与之关联的 TLC 文件，以达到生成自定义格式代码的目的[5-7]。首先，在 Matlab 中编辑 M 语言脚本并在模块中调用，用于处理传入模块的数据和设置显示在模块上的字符；然后，在 TLC 文件中对这些数据如何生成代码以及所需包含的头文件进行描述，从而在生成的 C 代码中调用控制器底层函数，其原理如图 1 所示。

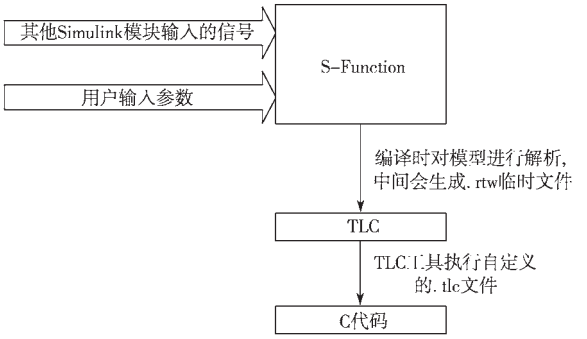


图 1 S-Function 生成自定义代码原理图

Fig. 1 Principle diagram of S-Function custom code

本文使用的 TC21x 芯片平台控制板的 CAN 通信 C 语言底层函数包含的功能有：硬件 CAN 通道设置、发送 CAN 报文及配置接收报文，函数名和所需设置的形参如图 2 所示。

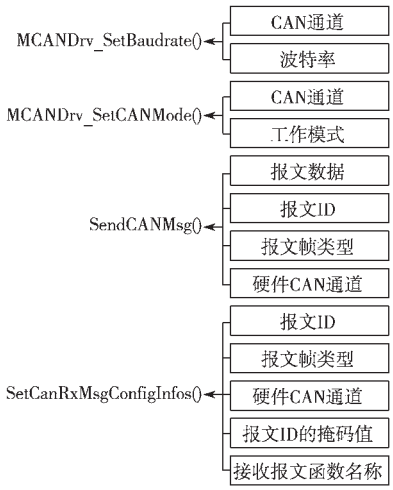


图 2 C 语言底层函数

Fig. 2 C language underlying functions

根据 C 语言的底层函数和 Matlab 支持读取 DBC（database for CAN）文件的功能，将 CAN 底层模块库划分为 CAN 初始化、CAN 报文数据装载与发送、CAN 报文数据接收与解析这 3 种功能 S-Function 模块。CAN 初始化模块用于设置波特率和工作模式，其他两个模块用于读取导入模型的 DBC 文件信息，并调用 CAN 报文收发底层函数来实现对报文数据的处理与收发。根据底层函数所需传入的参数类型，3 个模块的功能结构设计如图 3 所示。

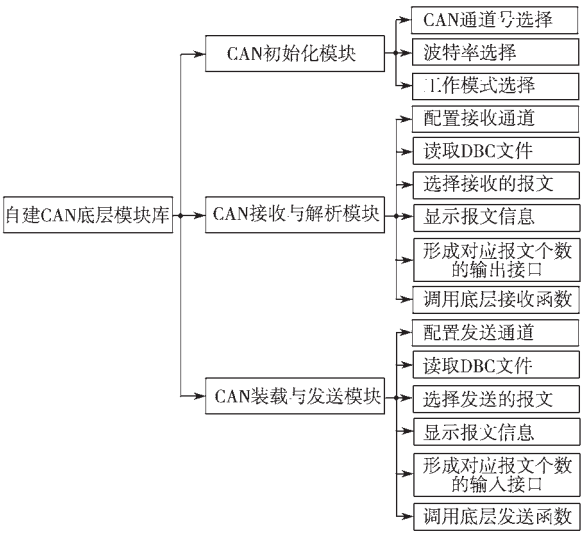


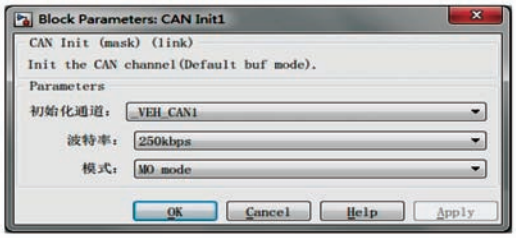
图 3 自建 CAN 底层模块库功能结构

Fig. 3 Functional structure of self-building CAN communication library

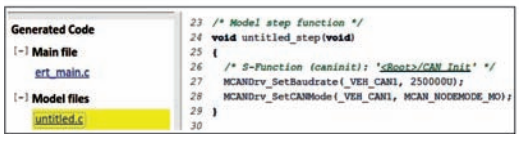
2 功能实现

CAN 底层模块库的各个模型均需图形用户界面（graphical user interface, GUI）与数据处理功能。在 GUI 的设计上，Matlab/Simulink 提供了便捷的 Mask Editor 和 GUIDE 工具供使用者对 Simulink 模型进行封装，可将各模块所需传入的参数类型作为模块对话框的输入项，并为 GUI 上的控件添加回调函数，对输入数据进行处理，最终传入 S-Function 参数用于生成代码。

CAN 初始化所需输入的参数信息较少，可使用 Mask Editor 对硬件 CAN 通道编号，将波特率和工作模式设置为弹出式下拉菜单，并为每个控件对应的 Name 属性命名，以此作为 S-Function 中 S-function parameters 的输入参数名称。实现 CAN 初始化模块同名的 TLC 文件后，可生成代码。所设计的对话框和所生成的代码如图 4 所示。



（a）对话框界面

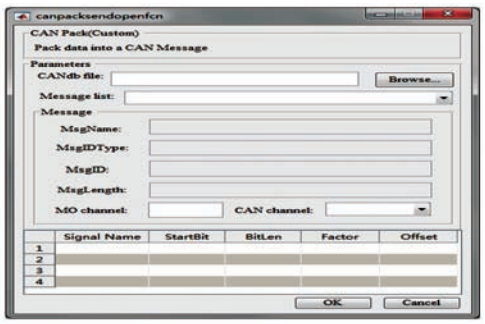


（b）所生成的代码

图 4 CAN 初始化模块的对话框界面与所生成的代码

Fig. 4 Dialog interface and generated code of CAN initial block

CAN 报文装载与发送、CAN 报文接收与解析输入模块参数较多，用 GUIDE 设计对话框界面可以显示更丰富的内容。为了方便观察模块所使用的 CAN 报文中每个信号的信息，用 GUIDE 将 CAN 报文信息的变量名、起始位、信号数据长度、比例系数及偏移量根据起始位置排序，并以表格形式在模块的对话框中显示。报文信息来源于 DBC 文件，因此需为 GUI 的 Browse 和 Message list 编辑框对应的控件设置回调函数，用于选择指定路径的 DBC 文件，并在选择所需报文 ID 后对协议内容进行读取与处理，之后再写入到模块参数中。以 EEC1（ID 为 0x0CF00400）报文中的发动机转速信号为例，CAN 报文装载与发送模块对话框与所生成的代码如图5所示。



（a）对话框

（b）所生成的代码

图 5　CAN 报文装载与发送模块的界面与代码

Fig. 5　Dialog and generated code of CAN message loading and sending block

为了便于在同一模型工程中对不同设置参数的模块进行区分，可通过 Mask Editor 中的 Icon drawing commands 设置模块外观显示的符号与文字。将 CAN 接收与解析模块进一步细分为带回调函数功能的模块和不带回调函数功能的模块，以满足不同需求。CAN 底层模块库各模块封装效果如图 6 所示，从左至右、从上至下分别为 CAN 初始化模块、CAN 报文装载与发送模块、CAN 报文接收与解析模块以及带回调功能的 CAN 报文接收与解析模块。

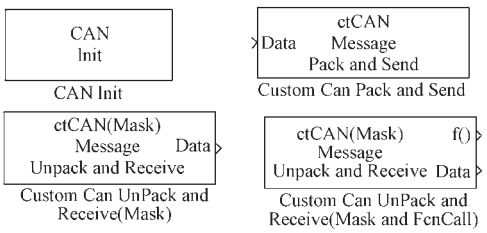


图 6　CAN 底层模块库各模块封装效果

Fig. 6　Encapsulation effect of each block in the CAN underlying module library

完成各模块的功能实现、封装和属性设置后，可将各模块集成为 Simulink 模块库。编写与模块库对应的 slblocks.m 文件，将该文件与模块库文件所在路径添加到 Matlab 的工作路径列表中。在 Simulink Library Browser 中执行 Refresh 操作，CAN 通信模块库可显示在 Simulink 库浏览菜单中，以便查找和使用，效果如图 7 所示。搭建模型时，只需在 CAN 底层模块库中将相应功能的模块拖入到 .mdl 或 .slx 模型文件中即可。每个在工程中使用的模块与 CAN 底层模块库中的父模块存在关联关系；在后期软件维护时，只需对模块库中的父模块功能进行修改，即可改变整个模型文件中各 CAN 模块的功能与生成代码的格式。

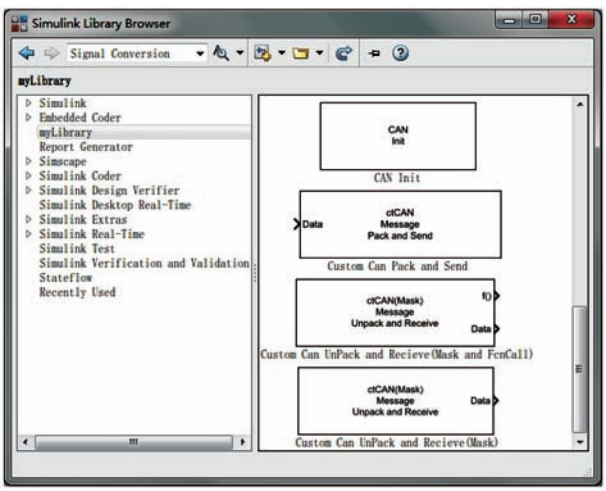


图 7　CAN 通信模块在 Simulink Library Browser 中显示效果

Fig. 7　Display effect of CAN communication block in Simulink Library Browser

3　功能验证

在完成了 CAN 通信模块库的功能设计与封装后，为了验证 CAN 模块是否能在硬件平台上正常运行，本文在 Infineon 的 TC21x 系列控制板上试验 CAN 报文的发送与接收功能。试验环境结构如图 8 所示。

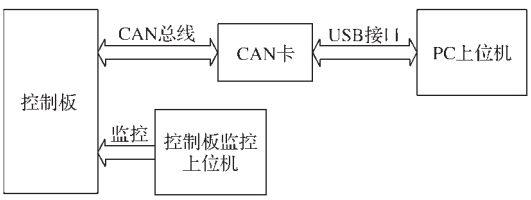


图 8　试验环境组成

Fig. 8　Test environment composition

本次试验分别用 EEC1 报文中的发动机转速值和 TCO1（ID 为 0x0CFE6CEE）报文中的整车车速值作为报文接收解析和报文数据装载与发送的测试对象。其中，发动机转速值在 EEC1 的 BYTE4 和 BYTE5 位置，分辨率为 0.125 (r/min)/bit；整车车速值在 TCO1 的 BYTE7 和 BYTE8 位置，分辨率为 1/256 (km/h)/bit。

搭建如图 9 所示测试模型，其中，CAN 初始化模块和 CAN 报文装载与发送模块分别被置于 Simulink 的原子子系统，用于指定函数名、生成单独的 C 函数以区分不同的模块功能，方便调用。使用 Simulink 编译模型验证了使用模块库所搭建的模型能成功生成代码，各子系统内部如图 10 所示。

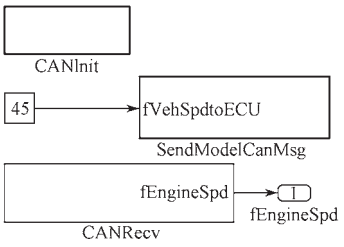


图 9　测试模型

Fig. 9　Test model

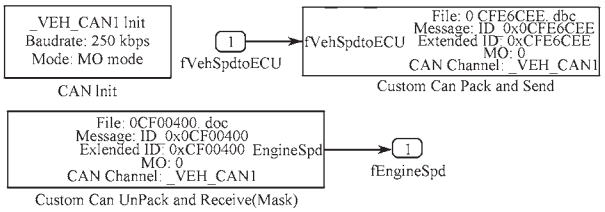


图 10　测试模型子系统内部

Fig. 10　Inner of the test model subsystem

将生成的 C 代码函数加入整个软件工程中调用，编译后烧写到控制器内；使用上位机软件 CANTest 对 TCO1 的报文情况进行采集，模型中车速的设置值为 45，报文发送值为 0x2D00（对应的十进制值为 11520），如图 11 所示，根据协议解析可得实际值为 45 km/h。

序号	传输方向	帧ID	帧格式	帧类型	数据长度	数据(HEX)
00022755	接收	0x0cfe6cee	数据帧	扩展帧	0x08	00 00 00 00 00 00 00 2d
00022756	接收	0x0cfe6cee	数据帧	扩展帧	0x08	00 00 00 00 00 00 00 2d
00022757	接收	0x0cfe6cee	数据帧	扩展帧	0x08	00 00 00 00 00 00 00 2d
00022758	接收	0x0cfe6cee	数据帧	扩展帧	0x08	00 00 00 00 00 00 00 2d
00022759	接收	0x0cfe6cee	数据帧	扩展帧	0x08	00 00 00 00 00 00 00 2d
00022760	接收	0x0cfe6cee	数据帧	扩展帧	0x08	00 00 00 00 00 00 00 2d
00022761	接收	0x0cfe6cee	数据帧	扩展帧	0x08	00 00 00 00 00 00 00 2d

图 11　报文采集结果

Fig. 11　Result of message collecting

在 CANTest 的基本操作栏中，设置报文帧 ID 和数据值发送给控制板，如图 12 所示。本文设置 EEC1 的发动机转速报文原始值为 0x1450（十进制值为 5200，根据协议，解析值为 650 r/min），通过控制板的上位机监控板可得所接收的内容为 650，如图 13 所示。

图 12　模拟发送报文

Fig. 12　Simulated message sending

图 13　上位机监控板接收到的发动机转速值

Fig. 13　Speed value of engine in the upper computer monitoring board

通过以上测试结果可看出，自建的 CAN 底层模块库接收与解析功能、数据装载与发送功能能在控制板中能正常运行，且协议解析内容与 DBC 文件中设置内容相符。

4　结语

本文通过 Simulink 自建 CAN 底层模块库，在 Infineon 控制板上实现了对数据发送和接收功能底层函数的调用，并将 DBC 文件的信息导入到模块中；结合控制板已实现的 C 语言底层函数，分别建立 CAN 报文接收与解析、CAN 报文装载与发送功能模块；通过 Simulink 自动生成代码技术验证了自建模块库所生成的代码在控制器上的运行结果。在基于 TC21x 系列芯片控制板的试验环境中的测试结果表明：自建的 CAN 底层模块库可正常调用底层的 CAN 报文发送和接收函数功能，并能解析报文或根据协议装载成原始报文。使用该方法，不需要人为对照 CAN 协议来编辑 C 代码和对 CAN 报文解析与装载，模型会自动根据 DBC 文件来对 CAN 报文信息进行处理，相比传统软件开发，降低了编写软件时对照报文协议的错误率，节省了开发时间，方便后期维护工作。由于本文模块库在代码中生成了较多单精度或双精度型变量，这会占用较多的控制板数据存储空间，因此优化生成代码中的数据类型是本文方案需要改进之处。

参考文献：

[1] 何宇 . 基于 Simulink 的飞控基础模型库的设计及生成代码性能优化研究 [D]. 成都 : 电子科技大学 , 2019: 2-3.

[2] 戴计生 , 陈俊波 , 李程 . 基于 Simulink 的嵌入式控制软件开发环境的设计 [J]. 大功率变流技术 , 2014(5): 7-11,32.

[3] 万彪 . 一种 Simulink 模块封装的自动代码生成技术研究 [J]. 机床与液压 , 2019, 47(10): 166-169.

[4] 刘少飞 . 基于 Simulink 的自建模块库设计及应用 [J]. 现代车用动力 , 2016(8): 4-8.

[5] 孙忠萧 . Simulink 仿真及代码生成技术入门到精通 [M]. 北京 : 北京航空航天大学出版社 , 2015.

[6] 赵强 , 赵仁德 , 王平 . 基于 C MEX S- 函数的 SVPWM 仿真研究 [J]. 变流技术与电力牵引 , 2008(5): 1-4, 24.

[7] 李延红 , 张书朋 , 李成 . 一种基于 M 语言的 MBD 软件开发自动化脚本工具 [J]. 汽车电器 , 2020(2): 47-52.