文章编号：1005-8451（2014）02-0014-03

基于SCADE的城轨联锁软件开发方法的研究

张　重，刘晓娟，李国瑞
（兰州交通大学 电子与信息工程学院，兰州　730070）

摘　要：CBTC的计算机联锁（CBI）系统是一个复杂且安全性要求非常高的系统，若按照传统的方法进行开发，很难达到其所需要的安全性和可靠性。本文提出基于高安全性应用程序开发环境（SCADE）开发城轨联锁系统软件的方法，能有效解决上述问题。文章主要介绍基于SCADE开发CBI软件的流程，建模方法，形式化验证和代码自动生成的方法。

关键词：城轨联锁；SCADE；形式化方法；建模
中图分类号：U231.7：TP39　**文献标识码：**A

Research on method of software development based on SCADE for Urban Transit interlocking

ZHANG Zhong, LIU Xiaojuan, LI Guorui
(School of Electronocs and Information Engineering, Lanzhou Jiaotong University, Lanzhou　730070, China)

Abstract: Computer based Interlocking(CBI) was a complex system, also a safety critical system. It was difficult to reach the safety and reliability of software needs by the traditional software development process. This paper proposed a method of software development based on Safety-Critical Application Development Environment (SCADE) for Urban Transit interlocking which could effectively solve the above problems. This paper mainly introduced the process of CBI software development based on SCADE, the modeling method, the method of formal verification and code automatically generated.

Key words: Urban Transit interlocking; SCADE; formal methods; modeling

CBTC 系统即基于通信的列车控制系统，主要包括车载列车自动防护（ATP）系统、区域控制器（ZC）、列车自动运行（ATO）系统、列车自动监控（ATS）系统和计算机联锁（CBI）系统。CBI 比较复杂，安全要求为 SIL4 级，为达到所要求的安全级别，采用安全软件开发方法是必要的[1]。本文提出利用基于模型驱动安全软件开发环境（SCADE）开发 CBI 软件的方法，如图 1 所示。使用这种方法既能保证所开发系统的安全性，又能降低开发的难度。

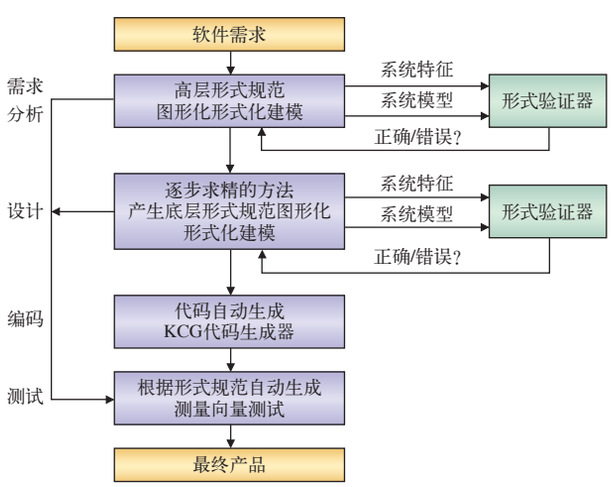

图1　基于SCADE的软件开发流程

1　SCADE简介

高安全性应用程序开发环境（SCADE，Safety Critical Application Development Environment）是一种基于模型驱动的软件开发环境，具有严格的数学理论基础。软件开发的整个流程都可以在 SCADE 中进行。SCADE 可以自动生成直接面向工程的标准 C 代码，从而提高了软件开发的效率。目前，SCADE 在国外已广泛应用于安全需求非常高的领域，如航空、核电、交通控制等[2]。

1.1　SCADE软件建模方法

在 SCADE 中，通过直观的图形符号进行建模，参与构建的图形符号都可以准确地转换为

收稿日期：2013-08-25
作者简介：张　重，在读硕士研究生；刘晓娟，教授。

LUSTRE 语言。LUSTRE 语言是一种形式化语言，能够确保所建模型的精确性和无二义性。在开发软件时，只需掌握 SCADE 基本的图形符号即可，并不需掌握 LUSTRE 语言，这样能有效降低开发的复杂度。在构建模型的过程中，既可以用图形符号构建，构建完成后其自动转换为 LUSTRE 语言，也可以用 LUSTRE 语言直接构建。

1.2 基于SCADE的图形化建模

SCADE 提供 2 种方法进行图形化建模：数据流图和安全状态机。

1.2.1 数据流图

数据流图能够很好地对连续控制系统进行建模、数据采集、信号处理、计算等相关工作，数据流图的功能节点即为软件的功能单元，能够构建多层次的功能节点来表示复杂的功能单元，数据流图所采用的算符有算术算符、时序算符、逻辑算符等，这些算符在 SCADE 中都是以直观的图形化方式表现出来，用图形化的方法构建软件模型具有直观、易于掌握等优点。

1.2.2 安全状态机

在 SCADE 中描述离散状态控制系统一般采用安全状态机，安全状态机主要用于描述外部中断处理和内部事物的处理，安全状态机也是用图形化符号表示，其构建的模型的变化控制逻辑是用一系列的状态符号、转移和信号来表示。用从一个状态到另一个状态的转移来表示系统的进展，用信号来触发状态的转移。

2 基于SCADE的CBI系统的建模

CBI 系统主要作用是通过技术方法使信号、道岔和进路按照一定的程序和满足一定的条件才能动作或建立起来，从而使列车能安全高效地运行。根据 CBI 系统的功能需求将其分为几个功能子模块 [3]，并根据它们之间的关系设计出功能模块的组成结构。

2.1 CBI系统功能模块

CBI 系统功能模块如表 1。

2.2 CBI系统结构设计

CBI 主要任务是通过现场设备和 CBTC 其他子系统提供的信息，控制现场设备，并且将安全信息传送给 CBTC 其他子系统 [4]。CBI 通过以下流程完成其功能：

表1　CBI系统功能模块

功能模块	描述
进路命令处理	接收并处理 ATS 发送来的进路命令
轨道空闲处理	接收和处理轨道区段“空闲、占用”状态信息，并把该状态信息转发给其他设备
进路控制	排列、锁闭和解锁进路
道岔控制	解锁、转换和锁闭道岔
信号控制	监控信号状态，并向 ZC 发送行车许可

（1）CBI 接收来至 ATS 发送来的进路命令信息。（2）根据 ZC 传来的相关轨道空闲信息以及相关道岔信息，由进路控制模块进行联锁逻辑判断后进行相应处理，并由道岔控制模块对相关道岔进行控制。（3）根据进路控制单元判定好的信息传送给信号控制模块，信号控制模块将信息进行有效处理后向 ZC 发送行车许可。根据上述描述建立 CBI 系统的结构图，如图 2 所示。

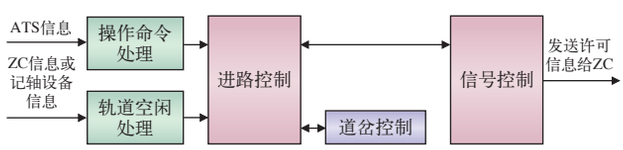

图2　CBI系统结构图

2.3 基于SCADE的CBI系统总体模型

根据上述对功能节点的划分，在 SCADE 的建模工具中定义对应的功能节点。按照图 2 所示的系统结构，在 SCADE 环境中建立 CBI 系统的总体模型，如图 3 所示。

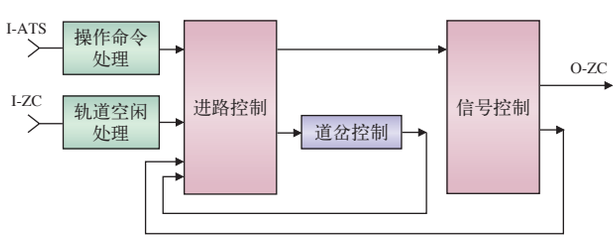

图3　基于SCADE的CBI系统总体功能模型

即用 SCADE 数据流图的功能节点将 CBI 的各功能模块刻画出来。各个功能节点对应 CBI 各功能模块。系统输入为 I-ATS 和 I-ZC，系统输出为 O-ZC。进路命令处理（Command Processing）模块接收 ATS 出来的进路命令信息，经过此功能模块处理，将相应的进路信息发送给进路控制（Route Control）模块。轨道空闲处理 (TVP) 功能模块接收 I- ZC 信息，并做相应处理传送给进路控制模块。进路控制模块接收进路处理模块、

轨道空闲检查模块、信号控制（Signal Control）模块、道岔控制（Switch control）模块相应信息，处理后将信息输出给道岔控制模块和信号控制模块。信号控制模块接收进路控制模块信息，做相应处理后将信息输出给信号控制模块。

3　基于SCADE的CBI模型的形式化验证

在SCADE中可以对所建模型进行形式验证，从而保证模型的正确性，它内置了ProverSL作为形式化验证引擎，对所建模型进行形式验证时，设计一个符合安全需求的特性观察器，对模型进行形式验证，如果验证结果表明模型符合安全需求，系统会自动给出一个安全证明，如果验证结果表明不符合安全需求，系统回给出一个反例，帮助进一步修正。这就能够保证软件的安全性并且提高验证的自动化程度。基于SCADE的形式验证大体分为4步，结合上述所建CBI模型的实例对其流程进行说明，如图4所示。

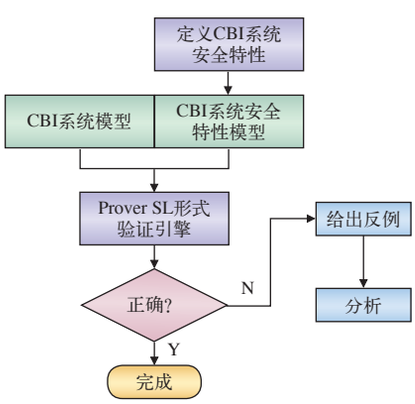

图4　CBI系统形式化验证的SCADE实现

（1）提取所建模型的安全需求，并进行规范。（2）用数据流图将提取出的CBI安全需求进行描述。（3）创建一个模型将CBI系统的安全性需求描述和软件需求描述结合到一起。（4）用形式化验证引擎对上述结合在一起的模型进行验证，如果模型安全，系统会给一个安全证明，如果模型不安全，会给出一个反例。

4　代码的自动生成与分析

4.1　代码自动生成与集成

SCADE能够自动生成面向工程的C代码，SCADE代码生成器满足DO-178B航空A级标准。所生成的代码集成方式为工程人员自定义一个结构体来调度自动生成功能函数块的接口，功能函数块为结构体的成员，编写基层支持软件并添加主函数，在主程序中调用主函数，并将所有代码在VC++6.0环境下编译集成。

4.2　代码测试与分析

本文应用clock函数测量手写代码和自动生成代码的执行时间。通过测试表明，自动生成的代码执行效率高于手写代码，如表2所示。

表2　执行时间的比较

执行步数（步）	执行时间（手写）	执行时间（自动生成）
10万	46 ms	38 ms
50万	264 ms	231 ms
200万	1 022 ms	967 ms
1 000万	11 650 ms	10 543 ms

5　结束语

研究表明，SCADE图形化建模直观、易于掌握、利于软件开发和后期维护。SCADE可以对其所建模型进行形式验证，保证软件开发的安全性。SCADE可自动生成直接面向工程的高安全的C代码，使用SCADE开发城轨联锁软件不仅能提高软件的开发效率，而且能有效保证其安全性，是开发城轨软件比较好的方法。

参考文献：

[1] CENELEC.EN50129: Railway Application-Communication, Signalling, Processing Systems-Safey Related Electronic System for Signailing[S]. 2003.

[2] 张合军．基于SCADE的无人机飞行控制系统软件设计[D]. 南京：南京航空航天学，2007.

[3] 刘晓娟，张雁鹏，汤自安．城市轨道交通智能控制系统[M]. 北京：中国铁道出版社，2010.

[4] 王　鲲，龙广钱，王　俊，等．关于城市轨道交通CBTC计算机联锁子系统的研究[J]. 现代城市轨道交通，2012（4）：1-3.

[5] 颜雯清．SCADE平台下C代码的自动生成[J]. 计算机仿真，2009（10）：24-28.

责任编辑　杨利明