

• 开发与应用 •

基于 MDA 的汽车嵌入式软件系统架构设计

王 军, 宋顺林

(江苏大学 计算机学院, 江苏 镇江 212013)

摘 要:为了高效管理愈来愈复杂的车辆电子、软件系统,并且在确保产品及服务质量的同时,提高成本效率,以统一建模语言(UML)和系统建模语言(SysML)为基础,参照汽车开放系统架构(AUTOSAR),提出一种基于模型驱动架构的汽车嵌入式软件开发方法,并以汽车引擎点火和敲打控制系统为例进行描述。性能分析表明,该方法缩短了汽车嵌入式软件开发周期,降低了开发成本。

关键词:汽车嵌入式软件;汽车开放系统架构;模型驱动架构;统一建模语言;系统建模语言

中图法分类号:TP311 **文献标识码**:A **文章编号**:1000-7024(2010)09-2048-04

MDA-based automotive embedded software system architecture design

WANG Jun, SONG Shun-lin

(College of Compute, Jiangsu University, Zhenjiang 212013, China)

Abstract: In order to efficiently manage the increasing complexity of vehicle electronics, software system, ensure the quality of products meanwhile, to improve cost-efficiency, based on model driven architecture with reference to AUTOSAR, a UML and SysML-based automotive embedded software development method is presented, and automotive engine ignition and knock control system are taken as an example to describe. Performance analysis shows that the approach reduces the automotive embedded software development cycle and development costs.

Key words: automotive embedded software; AUTOSAR; model driven architecture; UML (unified modeling language); SysML

0 引 言

在过去的10年,汽车系统及软件设计发生了翻天覆地的变化,设计技术和工作流程都在变化。在汽车工程师面临的众多挑战中,最关键的是跟上迅速增加的设计复杂性,许多根本无法跟上。此外,工作流程本身可以使人畏缩,因为许多组织使用并行或分布式的开发方式,相关联的团队和公司工作在不同的时区,说不同的语言,时常使用不同的工具和方法。从而需要工程师寻找新的开发方法,以帮助他们应对这些挑战。从一开始,OMG(object management group)的UML和SysML(systems modeling language)就被设计为满足工程师的架构挑战。工程师使用基于UML/SysML的架构设计,是一个强大的解决方案以迎接这些挑战。因为它们能够有效地捕捉架构。

笔者以AUTOSAR为基础,应用UML/SysML对AUTOSAR系统建模进行描述。讨论基于MDA的汽车嵌入式软件系统架构设计方法,实践表明该方法大大降低了设计复杂性,增强了汽车功能在整个网络化系统的可转移性。

1 AUTOSAR 基本原理

AUTOSAR(automotive open system architecture)^[1]汇集了主

要的汽车制造商、部件供应商及其它电子、半导体和软件系统公司,各成员协作努力制定标准,致力于为汽车工业开发一个开放的、标准化的软件架构。AUTOSAR架构有利于车辆电子系统软件的交换与更新,并为高效管理愈来愈复杂的车辆电子、软件系统提供了一个基础。此外,AUTOSAR在确保产品及服务质量的同时,提高了成本效率。为满足这些要求,AUTOSAR制定了一个分层的软件开发方法,允许企业在标准上合作,实现上竞争。

图1显示在一个单独的ECU(electronic control unit)里面的AUTOSAR分层软件系统。软件被分成3个主要的部分。在最高层,这里是功能系统(functional system),包括所有的特征,组成不同的电气子系统,如雨刷系统,传输控制系统或外部照明系统。由这些系统组成的元素叫做AUTOSAR软件组件(software component, SW-C),AUTOSAR软件组件是一个原子单元,完整的囊括数据,有定义良好的接口,通过AUTOSAR VFB/RTE进行通信。AUTOSAR运行时环境(run time environments, RTE)在应用软件和基础软件(Basic Software)之间。RTE是VFB的实现,它是一组APIs和中间件,连接基础软件到SW-Cs。所有包含在RTE下面的叫做基础软件,目的是抽象硬件并包含操作系统,通信接口(CAN, LIN, FlexRay, MOST),设备

收稿日期:2009-06-09;修订日期:2009-08-05。

基金项目:江苏省产业信息化重点基金项目(1633000004)。

作者简介:王军(1983-),男,山东临沂人,硕士研究生,研究方向为复杂系统体系结构建模、软件工程;宋顺林(1947-),男,江苏溧阳人,教授,博士生导师,研究方向为企业信息化、软件工程、数据挖掘、计算机图形学等。E-mail:petitapetit@163.com

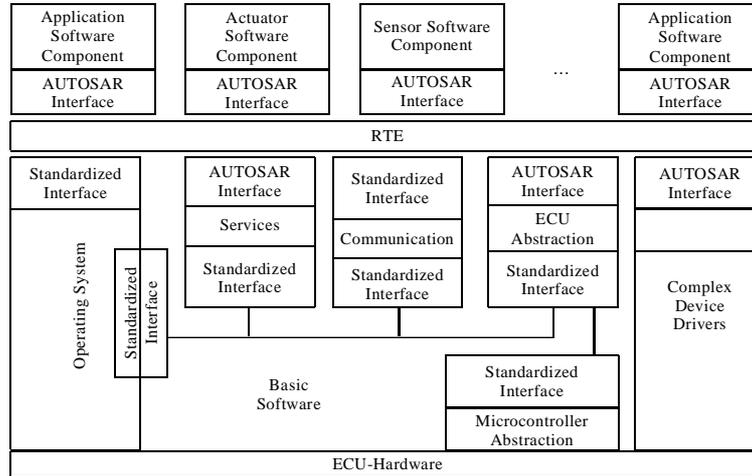


图 1 ECU 软件架构

驱动或其它服务。通过标准化软件组件和基础软件之间的接口,实际效果是,在不同的电气架构(electrical architectures)里,软件组件可以很容易的重定位到不同的 ECUs。

AUTOSAR 虚拟功能总线 (virtual functional bus) 抽象 AUTOSAR 软件组件之间的通信,以一种技术中立的方式,远离硬件,因此,两个软件组件之间的通信,可以发生在同样的 ECU 或通过总线发生在多个 ECU。虚拟功能总线允许两种 AUTOSAR 软件组件之间的通信方式:发送者—接收者模式,一个组件发出信息,另一个组件接听并收到它,客户端—服务器模式,客户端提出要求,服务器端提供请求的服务。

2 模型驱动架构在 AUTOSAR 中的应用

MDA (model driven architecture)^[2]是由 OMG 定义的一个软件开发框架。它是一种基于 UML 以及其它工业标准的框架,支持软件设计和模型的可视化、存储和交换。

2.1 UML/SysML

使用 UML^[3]和 SysML^[4]确定 AUTOSRA 建模环境的好处是,UML 和 SysML 可以超越 AUTOSRA 所覆盖的领域,尤其是需求捕获和细化,行为建模和设计组织。UML/SysML 包(package)的概念以不同的方式将模型元素组织起来,同时,组件被用来确定各种可部署配置,提供模型灵活性和组织性。AUTOSAR 没有详细覆盖的领域是系统需求的捕获和细化过程。在这一点上,UML 和 SysML 可以扮演特别重要的角色。

SysML 需求图(requirement diagrams)可以用于捕获文本需求,然后追踪这些需求到 AUTOSAR 软件组件,可运行实体和 ECUs 并实现它们。此外,需求可以被连接到测试场景,用于验证系统。UML/SysML 用例图(use case diagrams)使得需求组织能够描述不同的系统使用并定义它们,基于典型的系统用例,顺序图(sequence diagrams)描述系统需要履行的不同场景。最后,UML/SysML 状态图(statechart diagrams)和活动图(activity diagrams)提供强大的方法,表达组成 AUTOSAR 软件组件的不同可运行实体的行为。

2.2 AUTOSAR 元模型分层结构

AUTOSAR 定义了一套基于 MDA 的开发方法,其基石就是元模型(meta-model)和元元模型(meta-meta model)。在 OMG

通常的四层结构的基础上,AUTOSAR 多定义了一层 M2 (AUTOSAR 元模型,包括 ECU,端口,映射等)。M4 就是 MOF (meta object facility)。M3(AUTOSAR 元元模型)包括了通用 UML 及 AUTOSAR Profile 来更好的支持模板建模工作,AUTOSAR Profile 中定义了很多构造型(如 AtpClass,IsOfType)以使 UML 更适应汽车领域的建模要求。M2 则是真正用于汽车领域建模的元模型,汽车工程师运用定义在 M2 里的元模型对他们所要解决的问题建模,M1(AUTOSAR 模型)就是汽车工程师的工作成果。M0 是 AUTOSAR 系统在工作中的实现。

AUTOSAR 元模型(M2 层)是一个 PIM。它的定义独立于加工和处理平台(例如,程序设计语言,数据库,XML 格式),最后将参与到 AUTOSAR 模型(M1 层)的创建和编辑。这个平台无关模型允许汽车领域专家高效率的工作,而不用考虑它们最终是怎样映射到特定平台的。AUTOSAR 元模型分层结构如图 2 所示。

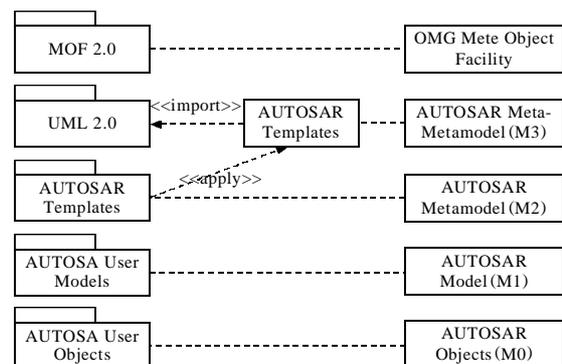


图 2 AUTOSAR 元模型分层结构

3 模型驱动开发工具

3.1 Telelogic Rhapsody

Rhapsody^[5]是 Telelogic 在嵌入式系统市场推出的最先进的 SysML/UML 模型驱动开发环境。Rhapsody 是唯一能同时满足系统工程师和软件开发人员所需的解决方案。其应用遍及众多产业领域,包括汽车、航空航天、医疗和交通运输等。

Rhapsody 提供符合 UML/SysML 标准的工具,可依据特殊

应用领域的建模需要而加以扩充,以提供真正的协同开发环境,无论大或小型的团队都能在此环境里,进行高效率而有价值的沟通。整体性的需求管理和变动追踪功能可确保系统设计能准确地满足客户需求。可测试性设计功能不但可以在流程初期就消除系统的错误,并能全程进行需求验证。Rhapsody 不仅能产生程序框架,它更能产生整个应用程序,所以能有效加速系统开发。以上这些先进技术都以平易近人的格式整合为套件,因而成为软件和系统工程师们的最佳解决方案。

此外,Rhapsody 允许工程师从各种来源集成丰富的外部组件,如 C/C++ 或 Java 代码。Rhapsody 的 R-Block 允许我们为第三方的工具,如 Simulink、StateMate 或 SDL Suite 创建自己的插件。

Telelogic Rhapsody AUTOSAR 软件包是第一个针对汽车行业,基于 UML/SysML,支持 AUTOSAR 的嵌入式软件模型驱动开发环境。汽车行业的工程师现在可以利用基于模型的开发环境来进行符合 AUTOSAR 标准的应用开发,包括设计以及 AUTOSAR 软件组件的交换。

Rhapsody AUTOSAR 软件包提供了一个 AUTOSAR 的建模环境,可以进行 AUTOSAR XML 文档的导入及导出,允许用户使用 AUTOSAR 标准所定义的图、符号以及专业术语来描述汽车 ECU 嵌入式软件系统。这些功能保证用户可以很容易的将 AUTOSAR 的系统模型集成到他们自己的过程以及工具链当中。另外,Rhapsody AUTOSAR 软件包允许工程师来定义 AUTOSAR 的软件组件(Software Component)以及独立于实现方式的通讯机制,这样可以保证软件组件可以被部署到不同结构的平台之上,增加系统灵活度以及软件组件的可重用性。

3.2 系统开发流程

笔者基于对 AUTOSAR 的研究,给出一种基于 UML/SysML 的汽车嵌入式软件开发流程。并且所有的开发过程都在上面所述的集成工具环境里实现。系统开发过程为:需求捕获、功能分析、系统架构设计、子系统架构设计以及系统实现。需求捕获阶段,对汽车的具体系统领域建立理解,用户的要求被转换成一组需求,这些需求确定系统必须做什么和能够做到什么程度。捕获到的需求被导入模型/需求仓库,一旦需求被充分理解,它们被集中到用例。功能分析阶段,每个用例被转换到一个模型,基本的需求通过模型执行得到验证和确认。系统架构设计的焦点是,分配核实过的操作契约(operational contracts)给物理架构。分配是一个可迭代的过程。与汽车领域专家相协作,不同的架构想法和分配策略被分析到,在需求捕获阶段捕获到性能和安全需要被考虑到。接下来的子系统架构设计阶段,决定在物理子系统里的哪一个操作契约将在硬件里执行,哪一个将在软件里执行。不同的设计思想在部署模型里捕获并通过回归测试(tradeoff analysis)得到验证。系统实现阶段,借助 Rhapsody 对模型驱动架构(MDA)的支持,开发者便可在数秒内迅速地将平台无关模型(PIM)落实到具体的车载即时性嵌入式作业系统上。

4 建模设计及部署实现

采用上述汽车嵌入式软件开发流程,笔者以汽车引擎点火和敲打控制系统为例进行介绍,限于文章篇幅,笔者只给出部分描述图形。

4.1 建模设计

4.1.1 需求简介

在一个四冲程引擎,一个循环有 4 个阶段:推进、压缩、燃烧、排放。这些阶段由凸轮轴(camshaft)驱动,它的位置用其角度衡量。实际中,为了最大限度提高功率和燃烧效率,以及减少废气排放,点火通常发生在第二冲程结束前,因此电火花必须在理论燃点(ignition top dead center,ITDC)之前生成。在点火决定点(ignition decision point,IDP),一个电子控制系统决定凸轮轴最佳角位置并生成电火花点燃压缩的混合燃料。

某些情况下,当未燃的混合燃料的温度和压强超过一个临界水平,第二次自动点火就会发生。二次点火产生冲击波并迅速增加汽缸压强。冲击波导致的冲量会在汽缸里激发一个特定频率的共振。如果重敲持续发生,会导致活塞、环、排气阀的损害。

敲打控制系统探测并改正这种现象。一个敲打控制系统由一个或多个噪声传感器组成,还有一个控制器,它通过传感器获得噪声并计算汽缸燃烧阶段需要的校正值。控制器能够使用频谱分析技术^[6]探测到敲打。控制器决定提前或推迟点火以达到纠正的目的。更多的关于火花点火引擎管理和敲打控制的详细资料,请参阅文献[7]。

4.1.2 用例图

图 3 描述敲打控制系统中用例之间的关系,并会由活动图完成。控制器接收由噪声传感器捕获的信号,通过计算相应的逻辑即时(logical instant)对纠正活动做出正确判断。

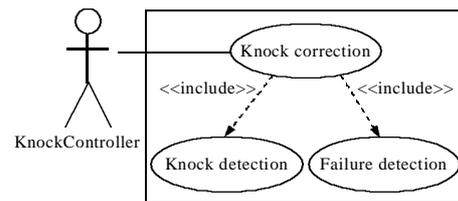


图 3 敲打校正用例图

4.1.3 状态机图

在 UML 状态机中,转换上的标签指定触发器而且必须参考一个事件。标签如“after d”或“at i”确定一个时间事件(Time-Event),前面的是相对即时(relative instant),后面的是完全即时(absolute instant),这些即时含蓄地提及了物理时间。在四冲程引擎,被事件触发的连续阶段同凸轮轴(camshaft)角坐标相关联而不是物理时间即时。凸轮轴角坐标被视为逻辑即时,我们读一个逻辑时钟(camclk)。这个时钟代表一个分离的逻辑时间,它的单位是°CAM(凸轮角度),分辨率是 0.5,最大值是 360。点火控制时间行为简单说明如图 4 所示,详见文献[8]。

4.1.4 活动图

活动图 5 代表信号处理的行为。噪声传感器产生的信号被控制器捕获到,经过格式化处理后传递给敲打探测事件。如果探测失败,控制器传递一个默认的矫正值给点火计算事件;探测成功,控制器传递实际的敲打能量值给计算事件。

4.2 模型验证

模型验证是软件开发的重要阶段,让我们在流程初期就消除系统错误。Simulink 是主要的模型验证工具,它给出了功能块的最详细描述。Simulink 能够生成代码和测试用例,通过

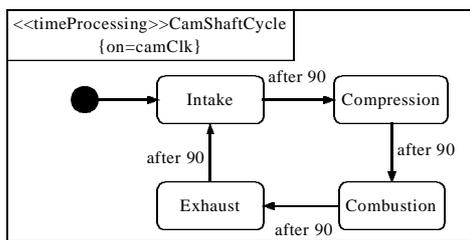


图 4 时间行为状态机

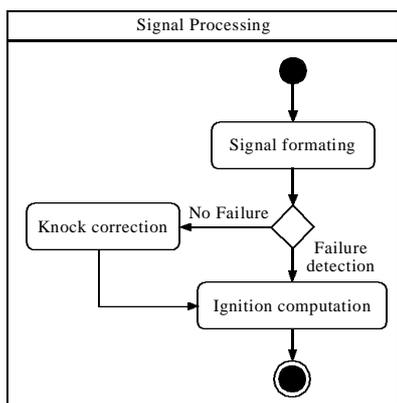


图 5 信号处理活动

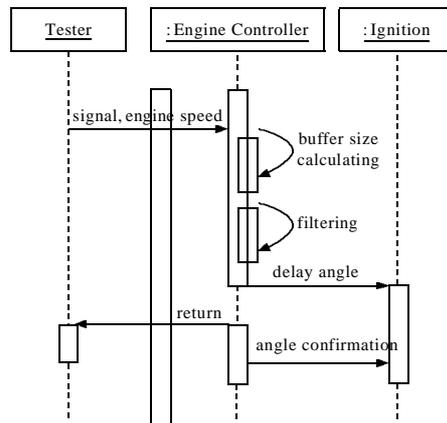


图 6 模拟验证产生的点火控制顺序

模拟对设计进行验证。图 6 是 Simulink 对系统模型进行仿真时捕获系统行为产生的。测试装置生成的模拟敲打信号和引擎转速传递给控制器, 控制器对信号进行过滤和计算。纠正过的信号传递给点火事件, 由点火事件计算出特定的点火逻辑即时。确认过的逻辑即时返回给控制器和测试装置。

4.3 自动生成代码

以上所建模型都是平台无关模型 (PIM)。借助 Rhapsody 对模型驱动架构 (MDA) 的支援, Rhapsody 可以从构建的平台无关模型自动生成应用程序代码, 并将其整合到实际的执行框架中, 便生成可执行的应用程序。生成的程序代码既清晰又具有高度可读性, 凡是能读取这些整合性程序码 (C、C++、Java、Ada) 的商用 IDE, 都能让工程师对模型或程序代码进行除错工作。因此, 不但生产力提升, 维护成本也降低了。

5 结束语

本文使用 UML/SysML 为汽车嵌入式软件开发描述了一些系统工程技术。这些技术提供了一个系统的整体视角, 包括需求分析, 系统功能分析, 架构设计。他们提供了一个可重

复使用的架构/软件机会, 并且允许开发团队考虑利用本文介绍的 MDA 思想。工程师能够审查不同的功能部署, 模拟可以运行并核实系统功能, 性能和时效性。所有这些通过使用 UML/SysML 得到实现, UML/SysML 可以作为信息的主要仓库, 不同工具之间通过 XML/XMI 交换信息。

参考文献:

- [1] Autosar Specifications Release 3.1[EB/OL]. http://www.autosar.org/find02_07.php,2009.
- [2] MDA Specifications [EB/OL]. <http://www.omg.org/mda/specs.htm>,2005.
- [3] Unified Modeling Language 2.0[EB/OL]. <http://www.omg.org/spec/UML/2.0/>,2005.
- [4] Systems Modeling Language 1.1[EB/OL].<http://www.omg.org/spec/SysML/1.1/>,2008.
- [5] Telelogic Rhapsody [EB/OL]. <http://www.telelogic.com/products/rhapsody/index.cfm>,2009.
- [6] Zadnik M,Vincent F,Vingerhoeds R.SI engine knock detection method robust to resonance frequency changes[R].Piscataway: Institute of Electrical and Electronics Engineers Inc,2007.
- [7] Bosch Robert.Bosch Automotive handbook[M].Massachusetts: Bentley Publishers,2004.
- [8] Andr C,Mallet F,Peraldi-Frati M A.A multiform time approach to real-time system modeling: Application to an automotive system[R].Piscataway:Institute of Electrical and Electronics Engineers Inc,2007.

(上接第 2047 页)

- [7] 张明媛,袁永博,白广斌.基于现代信息技术的城市灾害应急管理系统[J].防灾减灾工程学报,2008,28(3):382-287.
- [8] Murray E J.Modeling emergency response systems[C].40th Annual Hawaii International Conference on System Sciences,2007: 22-30.
- [9] 张友生,田俊国,殷建民.信息系统项目管理师辅导教程[M].北京:电子工业出版社,2005:172-173.
- [10] LANGE C, CHAUDRON M, MUSKENS J, In practice: UML

- software architecture and design description[J].IEEE Software, 2006,23(2):40-46.
- [11] ARISHOLM E,BRIAND L,HOVE S,et al.The impact of UML documentation on software maintenance: an experimental evaluation [J]. IEEE Transactions on Software Engineering, 2006,32(6):365-381.
- [12] 蔡敏,徐慧慧,黄炳强.UML 基础与 Rose 建模教程[M].北京:人民邮电出版社,2006:75-76.