

基于 SysML 的 RSW 系统模型的研究

刘悦, 谢益武

摘要: SysML (Systems Modeling Language) 是国际对象管理组织(OMG)开发的系统工程可视化体系结构设计语言。本文将系统建模语言为基础, 通过对一个具体的嵌入式系统 -RSW (Rain Sensing Wiper 雨滴自动擦洗系统) 的分析和研究, 从而构建该系统的环境图、需求图和用例图等, 并以此来探讨 SysML 在应用于嵌入式系统方面的方法和特点。

关键字: SysML, 建模, RSW 系统, 嵌入式系统

1. 引言

集成系统体系结构设计是网络化复杂系统建设面临的重要研究议题, 目前世界上在此方面的研究主要集中在强调功能分析的面向过程方法和强调对象及其交互的面向对象方法的综合运用上。结构化方法更多地运用于以完成使命为目标的系统工程过程中, 而面向对象方法则以 UML 为代表, 主导了软件工程设计。SysML(System Modeling Language)是近年 OMG(Object Management Group)组织多家公司和研究单位合作的项目, 力图综合结构化方法和面向对象方法的各自优势, 建立面向系统工程体系结构设计的统一建模语言。相对于软件工程的 UML, SysML 更能满足系统工程对功能分析和结构分解的需求, 将其应用于系统设计是一项很有意义的尝试。

嵌入式系统设计是一个软、硬件结合的协同设计 (Hardware/Software Co-design), 需要不同技术背景的人共同开发。这就带来了一个问题, 如何使这些具有不同技术背景和专长的人联合开发、协同设计。在软、硬件设计领域中, SysML 采用一种简单而直观的图形化方式描述系统设计中的各个问题和细节。不同技术背景的设计师只需懂得简单的 SysML 符号就可以轻易地与对方交流、共同设计。正是因为这些特点, 在嵌入式系统设计中使用 SysML 建模并书写文档通常可以起到事半功倍的效果。

本文将重点讨论如何在 RSW 这个具体的嵌入式系统中使用 SysML 技术, 并用设计出该系统的环境图、需求图和用例图, 从而研究 SysML 的应用于嵌入式系统的方法和特点。

2. SysML 概述及其特点

2.1 SysML 概述

为了满足系统工程的实际需要, 国际系统工程学会 INCOSE(International Council on Systems Engineering)和对象管理组织 OMG 决定在对 UML2.0 的子集进行重用和扩展的基础上, 提出一种新的系统建模语言 -SysML(Systems Modeling Language), 作为系统工程的标准建模语言, 和 UML 用来统一软件工程中使用的建模语言一样, SysML 的目的是统一系统工程中使用建模语言。

SysML 是一种多用途的标准建模语言, 能支持各种复杂系统的详细说明、分析、设计、验证和确认, 这些系统可能包括硬件、软件、信息、过程、人员和设备等。SysML 的定义包括 SysML 语义和 SysML 表示法两个部分。^[1]

2.2 SysML 的特点

2.2.1 SysML 的语义

SysML 为系统的结构模型、行为模型、需求模型和参数模型定义了语义。结构模型强调系统的层次以及对象之间的相互连接关系，包括类和装配。行为模型强调系统中的对象的行为，包括它们的活动、交互和状态历史。需求模型强调需求之间的追溯关系以及设计对需求的满足关系。参数模型强调系统或部件的属性之间的约束关系。SysML 为模型表示法提供了完整的语义。

2.2.2 SysML 的主要特点及应用领域

SysML 是 UML 在系统工程应用领域的延续和扩展。和其他系统工程建模语言相比，SysML 是一种通用的、功能强大的标准建模语言。它消除了不同方法在表达法和术语上的差异，避免了符号表示和理解上不必要的混乱。

SysML 独立于任何一种系统工程过程和方法，但支持过程和方法。SysML 的重点是标准建模语言，不是标准过程或方法。SysML 的应用必然与过程相关，不同的系统工程应用领域要求不同的过程。SysML 的开发者提出的开发过程是模型驱动、以体系结构为中心、迭代递增的过程，这可以说是一个元过程。

2.2.3 SysML 的图形表示

SysML 的图形表示是 SysML 的可视化表示，是用来为系统建模的工具。SysML 定义了九种基本图形来表示模型的各个方面。从模型的不同描述角度来划分，这九种基本图形分成四类：结构图 (Structure Diagram)、参数图 (Parametric Diagram)、需求图 (Requirement Diagram) 和行为图 (Behavior Diagram)。结构图包括类图 (Class Diagram) 和装配图 (Assembly Diagram)，行为图包括活动图 (Activity Diagram)、顺序图 (Sequence Diagram)、时间图 (Timing Diagram)、状态机图 (State Machine Diagram) 和用例图 (Use Case Diagram)。

如图 1^[2]

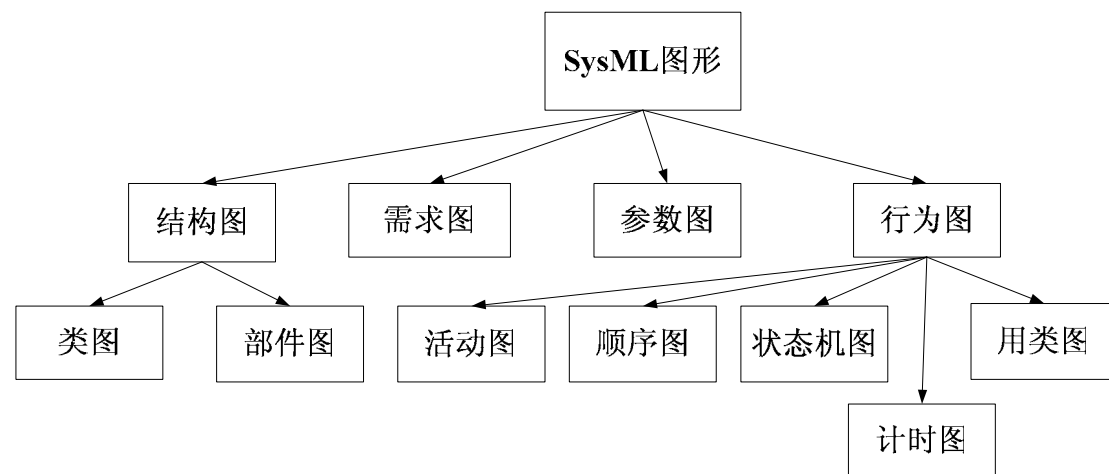


图 1 SysML 的图形表示
Fig1 SysML Diagram figure

3. 雨滴自动擦洗系统概述

现在，汽车中已经安装了越来越多的传感器以增加主动和被动安全性，一种具有极高的

市场渗透性的传感器是雨水传感器，以增加舒适性和安全性。如果汽车有雨水传感器，驾驶者就无需调节雨刮器设置来迅速停止刮片的运动或者得到更好的视角。当在湿路上驾驶时，驾驶者就无需动手来打开雨刮器，所以驾驶者就可以集中精力开车。目前大多数车型都采用的是单电机驱动的雨滴自动擦洗系统。这套系统的优点不言而喻，采用连杆机构驱动 2 个雨刮，成本比较低。但雨刮的噪声和雨雪天气刮刷不净的现象还是时有发生。

雨滴自动擦洗系统 RSW(Rain Sensing Wiper)是一个嵌入系统的真实例子（某种汽车所应用的 Rain Sensing Wipe 系统）。RSW 系统的目标是无论任何时候挡风玻璃外表面检测到液滴时，都能够自动化的进行擦洗（无需用户干预）。此外，依据检测到的液滴量调整雨刷器的速度。

这个系统主要拥有三种主要部件：

- (1) 控制雨刷器行为的软件。
- (2) 执行软件的电子控制单元。
- (3) 固定在挡风玻璃内表面的传感器，它的任务是通过挡风玻璃感应液滴。^[3]

4. SysML 在雨滴自动擦洗系统中的应用

4.1 环境图

环境图构建了系统的范围。在本例中，为系统确定了三类角色：Maintenance（维修用途），Car Electrical System（启动汽车中的系统），Driver（例如手动关闭系统）。需考虑三种外部系统：雨刷器接口，挡风玻璃，和汽车电子系统。注意用户定义的关键字 "external" 是用来限制外部部件的。还要注意汽车电子系统也要为 RSW 提供电力。因此，它应同时属于角色和外部系统。图 2 是 Rain Sensing Wiper 系统的环境图。

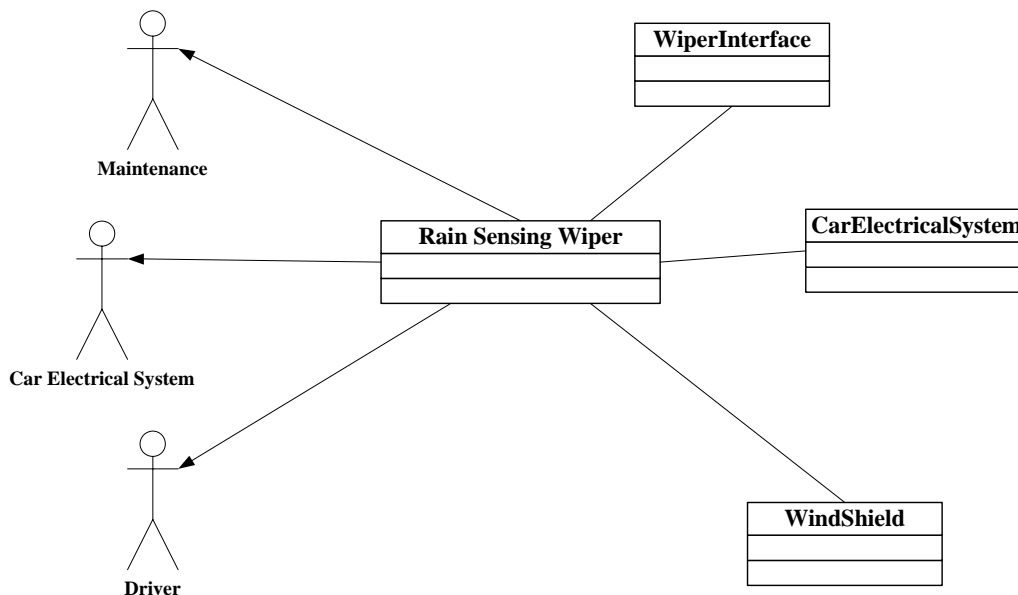


图 2 Rain Sensing Wiper 系统的环境图
Fig2 Rain Sensing Wiper System Condition Diagram

4.2 需求图

SysML 允许需求描述作为模型元素。因此，需求变为产品架构的一部分。这种语言灵活的表达了任何种类、任何关系的基于文本的需求。图 3 展示了 RSW 的需求图。

我们注意到需求图包含功能和非功能需求。SysML 中的需求是抽象类别的——就是说，它们不能被实例化——不包括操作或属性。然而我们可以通过为需求原型增加属性来实现为需求元素增加属性的目的。原型属性的使用允许在设计阶段设置这些属性值。1 需求不能加入到关联与泛化中；但是，一套预定义的关联关系可以帮助描述需求间的关系。我会在下面回顾这些关联关系。

子需求通过使用交叉关系来与它们的父需求关联起来，这些关系指明了嵌入式命名空间。例如，在图 3 中，Automatic Wiping 内的部分子需求通过交叉关系与它联系在一起。父需求相当于封装了嵌入的需求。在某种意义上说，删除父需求将会自动删除所有嵌入其中的需求。另一个封装其他需求的例子是 Core Functions，它包含了两个子需求。为了模型的可读性，用户定义的关键字 "package" 被提供给 Requirement 原型。

在需求分析阶段（例如，分解与细化），依据来源创建新的需求。这些新的需求都与最初的 DeriveRqt 依赖有关。例如，在图 3 中，一组产品的核心功能源自于 Automatic Wiping 下的需求组。选择 DeriveRqt 的名字是为了避免与 UML 2.0 中的 Derive 依赖标准发生混淆。其他例子的起源需求都是对于每个功能的技术性选择（参照图 3 中的 Technical Choices）。我们注意到设计师利用 Rationale 来解释使用固定在挡风玻璃上的传感器的原因。最后一个起源需求是质量需求 System Calibration，它指出系统应被校准。这是在声明狼籍的 RSW 失败之后被加入产品中的需求。这一需求的满足确保了产品不改变传感器与雨刷器的特性。需求间的另一种关系是细化（Refine）。需求细化的例子如图 3 所示。快速启动的需求由可能的速度选择细化：慢速、中等或快速。最后，一种通用的跟踪（Trace）依赖关系用以强调需求是通过不同方式被联系在一起的。图 3 中，手动减速的需求与自动减速联系在一起。需求拥有大量的源属性以存储上文中提到的关系状态。正如我在本系列的第 2 部分解释的那样，当需求关系在需求图外被提出时，这些属性将会变的非常便捷。^[4]

通常情况下，总会在整个产品周期中提出需求，并且需要额外的需求图加以表示。因此，产品需求一般规定为一组需求图。

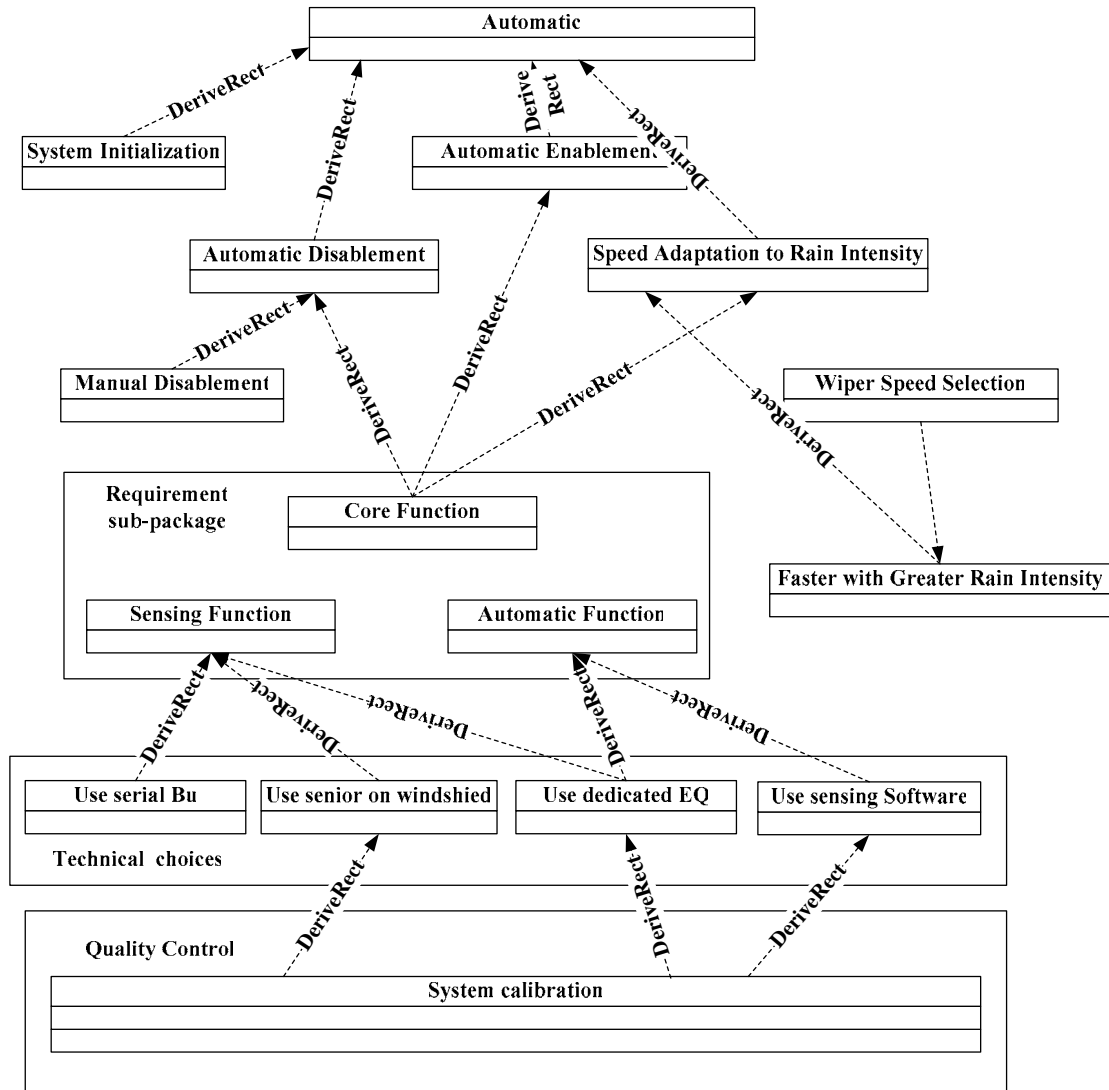


图3 Rain Sensing Wiper 系统的 SysML 需求图
Fig3 Rain Sensing Wiper System Requirement Diagram

4.3 用例图

SysML 提供了继承于 UML 2.0 的用例图。在图 4 中，我们用主要的用例（由椭圆图显示）展示了外部角色的交互作用。

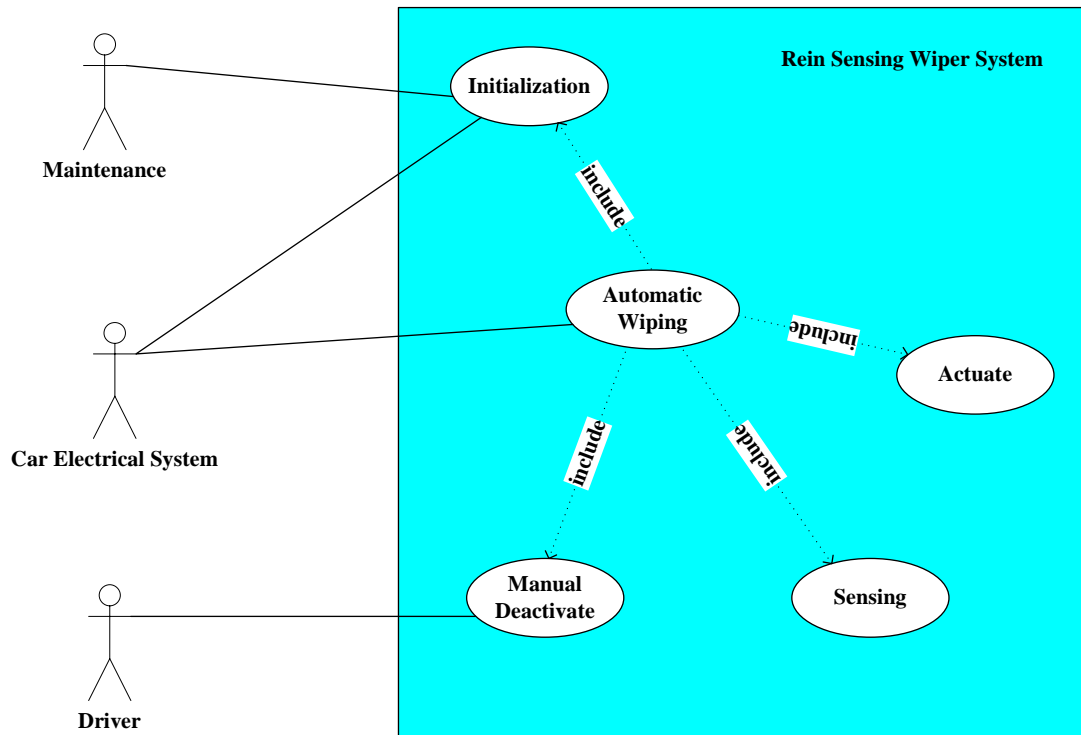


图 4 Rain Sensing Wiper 系统的 SysML 用例图

Fig4 Rain Sensing Wiper System Use Case Diagram

用例图展示了三类角色并分别介绍了各自的用例。被称为 Automatic Wiping 的关键用例是由一系列子用例组成。这种等级关系通过包括 (Include) 依赖关系形成。^[5]

5. 结论

本文首先概要地介绍了 SysML, 以及 SysML 的语义、特点和应用范围, 并说明的 SysML 九种基本图形。然后通过构建一个具体的嵌入式系统模型来探讨 SysML 在应用于嵌入式系统方面的方法和特点。限于篇幅, 只构建了该系统的环境图、需求图和用例图, 但是足以看出 SysML 在嵌入式系统建模的特点和优势。SysML 是一种先进实用、功能强大的标准建模语言, 能对系统工程的各种问题建模。在应用 SysML 解决系统工程问题时, 必须有相应的 SysML 支持环境来支持。目前, 作为 SysML 合作伙伴的各种工具开发商正在致力于 SysML 建模与仿真环境的开发, 这就促进 SysML 技术的发展, 普及 SysML 技术的应用, 从而有力地推动系统工程理论和实践的发展。

参考文献

- [1] System Modeling Language (SysML) Specification 03 April 2006
- [2] Yves Vanderperren, Wim Dehance :UML2 and SysML, an Approach to Deal with Complexity in SoC/NoC Design
- [3] Mathew Hause, Francis Thom and Alan Moore: Inside SysML
- [4] SysML and UML2 Support for Activity Modeling U.S. National Institute of Standards and Technology, 100 Bureau Drive, Stop 8263, Gaithersburg, MD 20899-8263
- [5] Jiang Cai-yun, Wang Wei-ping, Li Qun: SysML: A New Systems Modeling Language
- [6] 吴娟, 王明哲: 《DoDAF 产品集的 SysML 模型支持》 兵工自动化 2006 年第 25 卷第 2 期
- [7] OMG SysML Specification
- [8] <http://www.uml.org.cn/>

Based on SysML RSW System Modeling Research

Liu yue, Xie Yiwu

Department of Computer Science and Technology, Dalian Maritime University, Liaoning Dalian
(116026)

Abstract

SysML (Systems Modeling Language) is a system management visual Structural language which is developed by international object management organization. This paper bases on SysML and provides a concrete example-RSW(Rain Sensing Wiper System)which was analysed and researched ,designed this system Condition Diagram, Requirement Diagram and Use Case Diagram and so on.

Keywords: SysML, Modeling, RSW System, Embeded System