

浙江大学硕士学位论文删减版

基于 AUTOSAR 标准的汽车电子软件
开发平台分析和设计

(删减版)

浙江大学 ESE 工程中心 伍如意
www.smartsar.com support@smartsar.com

密级：_____

浙江大学

硕士学位论文



论文题目 基于 AUTOSAR 标准的汽车电子软件开发平台分析和设计

作者姓名 伍如意

指导教师 姚 敏 教授

杨国青 博士

学科(专业) 计算机应用技术

所在学院 计算机学院

提交日期 2011 年 3 月

A Dissertation Submitted to Zhejiang
University for the Degree of
Master of Engineering



TITLE: Analyze and Design of
Automotive Electronics Software
Development Platform based on
AUTOSAR Standard

Author: Ruyi Wu

Supervisor: Prof.Min Yao

Ph. D. Guoqing Yang

Subject: Computer Application Technology

College: Computer Science and Technology

Submitted Date: March 2011

摘要

随着汽车产业的快速发展，汽车电子软件的开发越来越受到人们的重视。同时，汽车电子软件系统的不断大型化和复杂化的发展趋势，为汽车电子软件的开发手段提出了新的挑战。AUTOSAR 作为一个开放的汽车电子体系架构，引领着汽车电子软件的发展潮流。

本文针对汽车电子软件的发展需求，结合汽车电子软件领域相关技术，提出了基于 AUTOSAR 标准的汽车电子软件开发平台 SmartSAR IDE，并对该平台进行了深入分析和设计，对平台的实现具有重要意义。

本文的主要研究工作包括以下几个方面：

- 1) 从功能、界面、性能、接口等多角度对 SmartSAR IDE 平台进行了系统性的分析，降低了在开发早期因设计错误而引入的风险。同时，在需求分析的基础上，按照功能划分对 SmartSAR IDE 平台进行了总体设计，包括模型设计器、系统配置器、ECU 配置器和 RTE 生成器。
- 2) SmartSAR IDE 平台模型设计器通过组件建模实现系统搭建，系统配置器从系统级别构建 ECU 和总线，ECU 配置器实现 OS、RTE 和基础软件的配置和代码生成，RTE 生成器生成可执行的通信代码。
- 3) 本文提供了统一的数据交换格式设计，实现了各个工具间的无缝集成，为与第三方工具的交互提供标准接口。同时，提供了模型验证功能设计，通过对用户模型的实时验证和批处理验证，保证了用户模型的正确性、一致性和完整性。
- 4) 本文实现了图形化的组件建模环境模型设计器，提供与 Matlab 工具的互操作，并通过数据资源库实现数据模型的重用性。最后，本文通过车身控制系统案例实现了模型设计器的功能应用。

关键词： 汽车电子，软件建模，AUTOSAR，需求分析，模型设计

Abstract

With the rapid development of automotive industry, people pay more and more attention to the development of automotive electronics software. Meanwhile, the large-scale and complex development trends in automotive electronics software system lead to new challenges on automotive electronics software development means. As an automotive electronics open architecture, AUTOSAR leads the development of automotive electronics.

This paper puts forwards an AUTOSAR compliant automotive electronic software development platform SmartSAR IDE based on development requirements of automotive electronic software and relative technology in this field, and analyses the platform deeply and design it in detail, and it is significant to the realization of the platform.

This paper concentrates on the following topics:

- 1) We provide a systematic requirement analysis to SmartSAR IDE from multi-aspects: function, user interface, performance and interfaces, reduce the risk of design error early in development. Meanwhile, according to the classified function points in requirement analysis, we designed the SmartSAR IDE, including model designer, system configurator, ECU configurator and RTE generator.
- 2) SmartSAR IDE model configurator provides system structure by component modeling, while the system configurator builds ECU and buses on system level. And ECU configurator realizes the configuration and code generation of OS, RTE and basic software. RTE generator can generate executable code of communication.
- 3) A uniform data exchange format achieves the integration of different tools, and provides standard interfaces for interacting with third-party tools. Besides, through real-time validation and batch validation for user model, the model validation function realizes the accuracy, consistency and integrity.

- 4) We have implemented the model designer as a graphical modeling environment, providing interoperability with Matlab and model reusability with data resource library. Finally, we have developed a demonstrative control system to illustrate the development process of model designer.

Keywords: Automotive electronics, software modeling, AUTOSAR, requirement analysis, model design

目录

摘要	i
Abstract.....	ii
第 1 章 绪论	1
1.1 课题背景	1
1.2 集成开发环境的发展现状	2
1.3 课题内容和研究意义	5
1.4 论文组织结构	6
1.5 本章小结	6
第 2 章 研究基础	7
2.1 理论基础	7
2.1.1 软件开发模型	7
2.1.2 需求工程	11
2.2 技术基础	13
2.2.1 统一建模语言 UML	13
2.2.2 数据交换格式 XML	14
2.3 AUTOSAR 体系结构	15
2.3.1 AUTOSAR 概述	15
2.3.2 Artop 基础平台	16
2.3.3 AUTOSAR 软件开发平台	17
2.4 本章小结	20
第 3 章 SmartSAR IDE 平台总体分析	21
第 4 章 SmartSAR IDE 平台总体设计	22
第 5 章 SmartSAR IDE 模型设计器的实现与应用	23
第 6 章 总结与展望	24
6.1 工作总结	24
6.2 工作展望	25
参考文献	26
攻读硕士学位期间主要的研究成果	30

致谢	31
----------	----

图目录

图 1.1 2000-2009 年中国汽车产量	1
图 1.2 V 形开发模式	3
图 2.1 瀑布式软件开发模型	7
图 2.2 快速原型开发模型	8
图 2.3 螺旋式模型	9
图 2.4 敏捷开发模式	10
图 2.5 需求工程过程	12
图 2.6 UML 图的分类	13
图 2.7 AUTOSAR 开发过程	16
图 2.8 ARTOP 组件	17
图 2.9 MICROSAR 体系结构	18
图 2.10 SystemDesk 用户界面	19

表目录

…(此处省略中)

第1章 绪论

1.1 课题背景

近年来，中国汽车行业正在以惊人的速度迅猛发展。根据中国汽车工业协会统计，中国汽车产量 10 年增长近 7 倍。如图 1.1 所示，2009 年中国汽车产量达 1379 万辆，同比增长 47.49%。在全球经济恢复的艰难时期，中国汽车工业却首次超越美国，成为全球产销量第一的国家，令世界瞩目^[1]。

根据中国汽车工业协会统计数据显示，2009 年中国私人汽车拥有量已经迅速增长到 4575 万辆^[2]，越来越多的老百姓圆了自己的汽车梦，很多家庭开始买第二三辆车，消费需求越来越大，无不反映着中国汽车行业正走向成熟。汽车电子软件系统作为汽车中不可或缺的一部分，也将迎来它的春天。中国汽车消费者已经从盲目跟风走向个性化发展，消费者更加追求安全、舒适兼具娱乐化的智能汽车，这也对汽车电子软件系统提出了更高的需求和更大的挑战^[3, 4]。

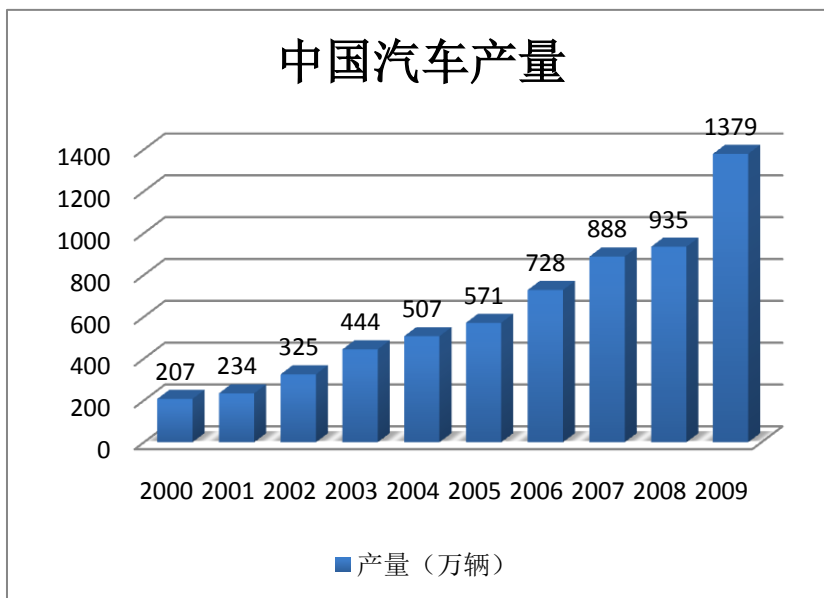


图 1.1 2000-2009 年中国汽车产量

汽车电子软件是依托汽车电子行业和计算机软件的应用而发展起来的。传统的软件开发模式已经不适应如今日益大型化、复杂化的软件系统^[5]，亟需新型的、

高效的软件体系架构来打破汽车电子软件的发展瓶颈。在此时机下，许多国际组织纷纷推出了新型的标准架构，如欧洲的 AUTOSAR 组织所推出的 AUTOSAR 标准^[6]，日本的 JASPAR 标准^[7]等等。

AUTOSAR 标准以“在标准上合作，在实现上竞争”的理念和分层的体系架构吸引了众多的汽车电子软件开发者和供应商，他们纷纷推出了自己的 AUTOSAR 产品，如 EB 公司的 EB Tresos、Vector 公司的 DaVinci Developer、dSpace 公司的 SystemDesk 等等，然而国内尚没有公司或企业推出本土的汽车电子软件开发平台。

浙江大学嵌入式工程中心从 2004 年开始一直致力于 AUTOSAR 标准的研究，并且得到了国家核高基重大专项“实时嵌入式操作系统及开发环境”课题的支持，它的主要研究内容是针对汽车电子软件的强实时、高可靠、网络化、平台化以及标准化发展趋势，研制汽车电子实时嵌入式操作系统及开发环境产品；制定与国际汽车电子开放体系结构（AUTOSAR、OSEK）标准兼容的自主汽车电子基础软件平台体系规范，开发基于国产实时嵌入式操作系统的软硬件参考实现平台。

本文的研究目标是针对 AUTOSAR 标准，结合汽车电子领域基础研究和关键技术，吸收当前 AUTOSAR 软件开发工具的优势，采用 UML 需求分析方法和基于 ARTOP 的平台技术，分析和设计基于 AUTOSAR 标准的汽车电子软件开发平台 SmartSAR IDE。SmartSAR IDE 平台作为一个庞大的集成开发环境，将形成一套完整的汽车电子软件开发工具链，对其进行详细的需求分析和设计是十分重要的。在需求 and 设计阶段发现错误，将大幅减低后期的开发风险和修复代价。

1.2 集成开发环境的发展现状

随着汽车电子行业的快速发展，汽车电子软件的设计方法和开发环境的研究也备受关注。因为只有采用合理的设计方法和高效完备的集成开发环境，才能快速、高效的产出高品质、易扩展的产品，而只有这样的产品才能不被社会淘汰。因此，研发高效的集成开发环境是十分有意义的。

传统的汽车电子软件通常采用“V”形开发模式，它是在快速应用开发模型

的基础上演变而来。V 形开发模式强调软件开发的协作和速度，整个 V 流程包括需求分析，系统设计，详细设计，编码，单元测试，集成测试，系统测试和验收测试，如图 1.2 所示^[8]。

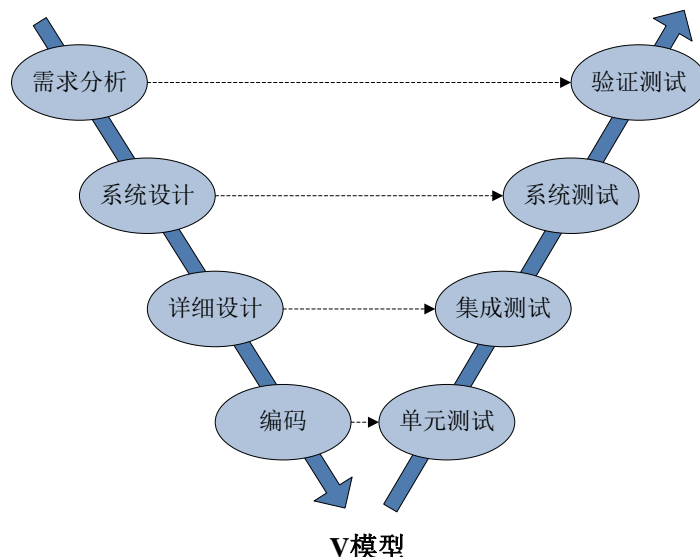


图 1.2 V 形开发模式

V 形开发模式的左侧主要进行软件的分析 and 设计，是设计实现的过程，右侧主要是对左侧的测试和验证，确定是否满足用户的需求。

需求分析阶段：本阶段的主要目标是确认理想的系统模型，分析和收集用户需求，形成用户需求文档。通常，需求文档应说明用户期望的系统功能、性能、接口、安全性等需求，它将作为系统设计阶段的指导文档。此外，验收测试也应在本阶段设计。

系统设计阶段：开发人员需要分析和理解用户需求文档，思考系统的可行性和实现的技术，当需求不可行时要及时反馈给用户修改。本阶段的主要目标是形成系统的总体架构，确定系统的总体方案，形成系统设计文档。该文档应该说明系统的组织，框架，数据结构等等。同时，测试人员在了解系统是如何实现的同时，设计系统的测试方案和测试计划。

详细设计阶段：详细设计是十分底层的设计，涉及到系统具体模块小单元的设计，是开发人员进行编码的依据。本阶段的主要目标是设计模块的详细功能逻

辑、数据库表、伪代码等，产生详细设计文档。同时，测试人员可以对设计进行评审，设计测试用例，完善测试计划。

编码阶段：本阶段开发人员根据详细设计文档进行编码，在编码的同时，测试人员可以进行单元测试，多线程工作，充分的单元测试可以有效地提升程序质量、减少成本。

从 V 形开发模式可以看出，它十分注重系统的质量，将测试工作提前至与需求同步，这也就避免了瀑布模型的不足，总是在代码完成之后进行测试，大大减少了缺陷的发生率和修复代价。

然而，由于 V 流程中的大部分工具缺乏模型可重用性和兼容性，已经无法满足复杂汽车电控系统开发的需求^[9, 10]。模型驱动架构 (Model Driven Architecture)^[11, 12]的提出暂时缓解了这种压力。模型驱动架构方法，把嵌入式应用软件的开发从代码层次提升到模型层次，以模型为中心，通过自动化工具自动完成模型的转换和代码的生成^[13]，从而提高软件的可信度、缩短开发周期。

基于模型驱动软件开发方法的典型的集成开发环境包括 Ptolemy II、GME、SCADE、Rhapsody、Matlab 等。

Ptolemy II^[14]是美国伯克利大学开发的层次化的软件平台，主要用于系统级的建模和仿真。它的主要特点是引入了面向角色的概念，提供多种计算模型。在 Ptolemy II 中，可以通过图形化表述、XML 描述形式、API 编程三种方式进行 workflow 建模。作为一个图形化的建模环境，Ptolemy II 提供简单的可视化交互界面，降低了用户的建模难度，同时作为一个开源系统，在嵌入式领域已经有着广泛的应用。

GME^[15] (Generic Modeling Environment) 是美国范德比尔特大学开发的面向特定领域的建模环境工具集。GME 通过 UML 和 OCL 语言来进行元模型建模，从而生成目标环境的特定模型，这些模型可以通过 XML 格式存储，并生成应用。GME 拥有很强的可扩展性，支持 C++、Visual Basic、C#、Python 等语言^[16]。GME 拥有很多高级特性，其内置的约束管理器增强了建模时的领域约束。同时，它所提供的数据库资源支持模型级的重用。

SCADE^[17] (Safety-Critical Application Development Environment) 是法国爱斯特尔技术公司开发的一个高安全性的应用开发环境。SCADE 支持航空电子、汽车电子、核电站控制等多个嵌入式领域, 提出了一整套嵌入式软件开发的自动化工具。工具覆盖了嵌入式软件开发的整个过程: 图形化建模、模型一致性检查、模型仿真、形式化验证、代码自动生成和文档自动生成等等。同时, SCADE 工具还支持与其他工具的桥接, 如需求分析和追踪工具 DOORS、配置管理工具 Telelogic Synergy、Simulink 等。

IBM Rational Rhapsody Developer^[18] 是一个针对嵌入式实时系统的基于 UML/SysML 的模型驱动开发平台。Rhapsody Developer 提供强大而又灵活的图形化建模环境, 支持从需求收集、设计、实施和测试等软件开发环节。Rhapsody Developer 可以生成支持 C、C++、Java 和 Ada 语言的应用, 开发人员可迅速将其应用到不同的实时操作系统中^[19]。新版的 Rhapsody 还增加了对处理多核的支持, 帮助用户开发多核产品。

1.3 课题内容和研究意义

随着生活水平的不断提升, 人们对汽车的需求量也越来越大, 尤其是私家车, 近几年也迅猛增长。人们在需要一种代步工具的同时, 也对汽车提出了更高的要求, 比如集节能、安全、娱乐等为一体的新型智能汽车将是大众的首选。这也正是对汽车电子软件的提出的需求。应对不断复杂化的系统, 传统的汽车电子软件开发已经到达瓶颈, 与此同时, 以 AUTOSAR 标准为代表的一系列标准化体系渐渐萌芽。国际上的一些汽车电子软件开发商也陆续推出自己的 AUTOSAR 解决方案来抢占市场先机。

本文通过深入调研汽车电子领域的相关技术和开发方法, 结合当前的汽车电子开放体系架构, 在分析了当前其他 AUTOSAR 解决方案的基础上, 设计了一个基于 AUTOSAR 标准的汽车电子软件开发平台 SmartSAR IDE。该平台在设计时考虑到从整体架构到具体模块的实现细节, 以及用户交互界面。SmartSAR IDE 平台采用分层的机制, 做到低耦合, 大大提升了系统的可重用性和可扩展性。本文分析和

设计了 SmartSAR IDE 平台包括从模型设计器、系统配置器、ECU 配置器到 RTE 生成器的完整流程，最后实现了模型设计器，并通过车身控制系统案例实现了模型设计器的应用。

为了使 SmartSAR IDE 平台能够顺利的开发，进行完善的系统分析和设计是十分有价值的。由于 SmartSAR IDE 平台构成庞大，所包含的四大工具不仅功能复杂且需求各异，因此进行多角度的需求分析和详细的设计能够大大降低实现的复杂度，减少因设计错误而引入的开发风险。

1.4 论文组织结构

本文的内容和组织结构如下：

第一章为绪论。该章节描述了本文的项目背景，相关集成开发环境的发展现状，明确了本文的研究内容与意义。

第二章为研究基础。该章节阐述了 SmartSAR IDE 平台的理论基础，包括各种软件开发模型、需求分析方法以及 AUTOSAR 体系结构。同时，在调研现有技术的基础上，对 SmartSAR IDE 平台的基本技术进行了总结。

第三章为 SmartSAR IDE 平台总体分析。该章节从不同角度对 SmartSAR IDE 平台进行了需求分析，包括功能分析、性能分析、界面分析和接口分析。

第四章为 SmartSAR IDE 平台总体设计。该章节分模块对 SmartSAR IDE 平台进行了详细阐述，包括模型设计器、系统配置器、ECU 配置器和 RTE 生成器。

第五章为 SmartSAR IDE 模型设计器的实现与应用。该章节阐述了模型设计器各个功能模块的实现过程，并通过车身控制系统案例对模型设计器进行应用。

第六章为总结与展望。该章节对本文的内容进行了总结，并对未来的工作进行了展望。

1.5 本章小结

本章首先介绍了课题背景，接着分析了集成开发环境的发展现状，包括 V 形开发模式和典型的基于模型驱动的开发环境。然后介绍了本文的主要内容和研究意义，最后介绍了本文的组织结构。

第2章 研究基础

对基于 AUTOSAR 标准的汽车电子软件开发平台进行分析和设计是一个涉及软件工程方法、需求工程方法、AUTOSAR 体系架构等领域的复杂工程，需要高效的软件开发方法和先进的技术基础支持。

2.1 理论基础

2.1.1 软件开发模型

软件开发模型是为了获得高质量的软件系统所需完成的一系列任务的框架。它规定了软件开发过程中各阶段的执行顺序和执行过程。软件开发模型的提出有利于各种人员明确自己当前的任务和活动，有利于协调各人员之间的通信。典型的几种软件开发模型包括瀑布式模型、快速原型模型、螺旋模型、迭代式模型、敏捷开发模型等等。

2.1.1.1 瀑布式模型

瀑布式模型是最早提出的、应用最广泛的软件开发模型，如图 2.1 所示^[20]。瀑布式模型将软件开发过程中的各个阶段规定为以顺序连接的模型，包括系统建模、需求分析与定义、设计、实现、测试和维护。由于各阶段顺序执行，瀑布模型也因此而得名。

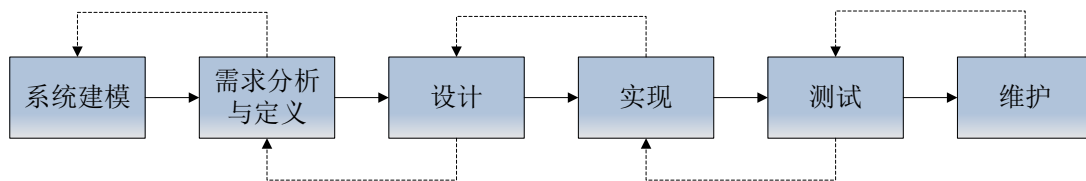


图 2.1 瀑布式软件开发模型

瀑布式模型的突出特点是，它是一种自上而下的模型，各阶段不能重叠，每个阶段都需进行验证，并且产出明确的文档，进行分阶段评审，评审通过后才能进入下一阶段。瀑布式模型的主要优点：

- 模型简单，用户容易熟悉
- 分阶段的形式可以控制项目进程，便于项目管理
- 可以分阶段设置里程碑，便于进行项目跟踪
- 顺序的分阶段模式有利于复杂问题简单化

无疑瀑布式模型给软件开发提供了一种有效的模式，然而在实际开发中也暴露出它的诸多缺点：

- 用户参与度不够，不适用于用于需求不明确的软件项目
- 各阶段比较独立，阶段间移交信息过程中，容易产生偏差
- 以文档为驱动，耗费大量时间，开发人员经常被耽误
- 缺乏灵活性，不适合大型软件的开发

2.1.1.2 快速原型模型

由于许多项目在开始阶段需求都是不明确的，因此快速原型模型针对瀑布式模型的不足进行了改进。为了减少因为软件需求模糊而带来的开发风险，快速原型模型采用的策略是，首先根据基本需求快速建立一个初始原型，然后根据用户的反馈对原型进行改进，直到满足用户的所有需求。图 2.2 描述了快速原型模型的开发过程^[20]。

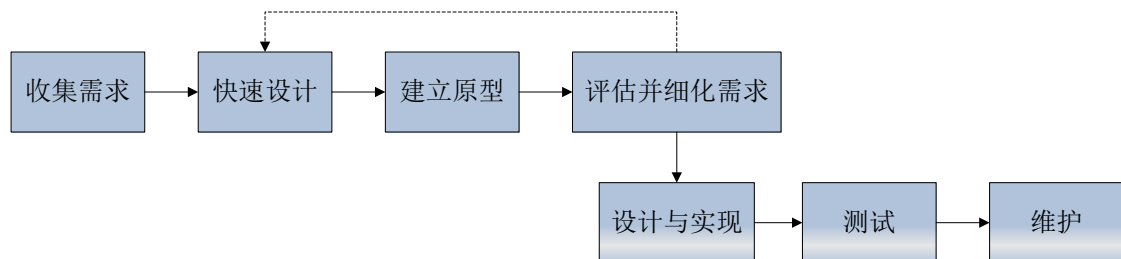


图 2.2 快速原型开发模型

快速原型模型特别适用于用户需求不明确的软件项目，建立原型的主要目的就是为获得完整的需求。当需求明确后，原型可以被抛弃，也可以继续修改原型，在此基础上发展为最终产品。快速原型模型在开发过程上与瀑布式模型类似，但是弥补了瀑布式模型用户参与度不足的缺陷。

快速原型模型的主要优点是快速，开发人员构造快速原型，提高了软件开发效率，减少了开发成本。同时，用户的充分参与，不仅使需求明确，而且减少了以往对用户需求理解不明确以及遗漏需求的情况，降低了软件开发后期返工的代价。但是，快速原型模型也存在一些缺点，开始开发出来的快速原型容易被用户误解为最终产品，用户容易误解开发人员的实际工作量。

2.1.1.3 螺旋式模型

瀑布式模型和快速原型模型都是比较理想化的模型，并没有考虑到实际项目开发过程中存在的大量的风险因素，而对风险控制的不当往往会导致整个项目的失败。螺旋式模型的主要思想是将软件开发过程分为几个螺旋周期，每个周期大致是一个瀑布模型，并在每个周期前加上风险分析。螺旋式模型的开发过程如图 2.3 所示^[21]。每个螺旋周期都包括计划的制定，风险的分析，工程的实施和计划下一周期。

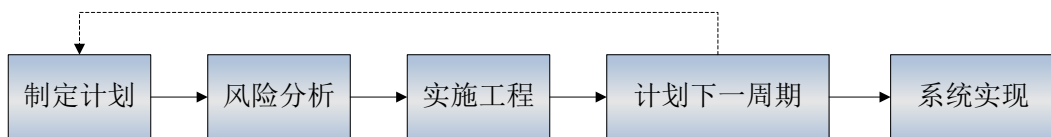


图 2.3 螺旋式模型

螺旋式模型的最大特点是整个开发过程的风险都是受控制的，对于大型项目的开发十分有利。由于每个螺旋周期都会进行风险分析，因此潜在的风险可以及时发现，降低了整个项目的综合风险。然而，这也对开发人员提出了更高的要求，要求开发人员具备丰富的风险评估能力，而大部分开发人员难以达到要求。螺旋式模型并不适合低风险、小规模的项目，每个周期的风险分析会导致昂贵的成本。

2.1.1.4 敏捷开发模型

敏捷开发是从 2001 年开始萌发的一种新型开发模式，如图 2.4 所示，它的主要目标是尽可能早地、快速地实现软件的交付^[22]。目前，很多人对敏捷开发存在误解，认为敏捷开发是一种不需要进行需求分析，不需要编写文档，一味关注

编码实现软件交付的一种开发模式。这种理解是片面的，事实上敏捷开发包括敏捷需求分析、敏捷项目管理、敏捷软件开发三个部分。

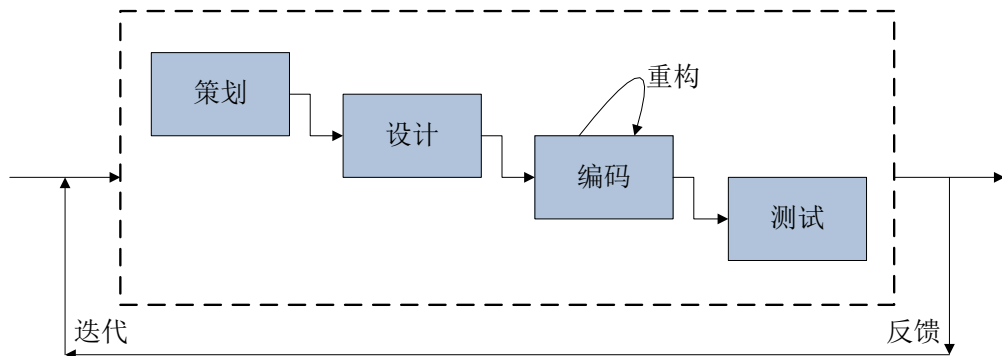


图 2.4 敏捷开发模式

敏捷开发有多种开发模式，常用的开发模式是极限编程（Extreme Programming，简称 XP）和并列争求法（Scrum）。极限编程^[23]是 KenBeck 在 1996 年提出的一种轻量级的、低风险的开发模式。极限编程认为软件需求的变化是不可避免的，传统的在项目起始阶段完成所有需求的定义后再进行开发是不切实际的，在项目开展过程中适应需求的变化才是解决问题的根本方法。极限编程打破了传统的开发模式，拥有自己的核心价值观“沟通、简单、反馈和勇气”，这种价值观启发着开发人员去学习如何应对快速变化和多样化的技术。Scrum^[24]是一种迭代式的软件开发模式，它将整个项目按优先级分为多个冲刺阶段（sprint）来进行，每个 sprint 一般持续两到三周。每个 sprint 开始会开展计划会议，确定该 sprint 的总体目标和个人目标，sprint 结束时会开展总结会议，回顾该 sprint 的任务完成情况，在 sprint 进行过程中，每天会有简短的日常情况会议方便项目经理跟踪项目进度。

然而，敏捷开发模式并不适用于任何软件开发团队，它比较适用于小规模团队，要求团队的成员是积极主动负责任的。此外，开发软件必须是主要目标，要求开发出满足客户需求的有效的产品。敏捷开发的主要优点是可以应对需求的变化，而且能快速发现客户的需求。

2.1.2 需求工程

随着计算机的发展，人们逐渐认识到需求分析和定义的重要性。需求工程是指利用已有的技术和方法进行需求分析，确定用户需求，帮助开发人员理解系统目标的过程。

从软件开发模型可以看出，需求工程是软件开发的起始阶段，为后续阶段的软件开发奠定基础。需求错误的代价会随着软件项目的开展发生变化，越到项目后期，错误代价将会呈现指数级的增长趋势。因此，需求工程在软件开发过程中起着十分重要的作用。

2.1.2.1 需求工程过程

需求工程的目标是待开发的系统进行分析，最终产生精确的软件需求规格说明书。需求工程过程包括软件需求的开发和管理两个部分。需求开发的结果主要是产出需求说明书，需求管理则是根据需求的变化对需求说明书进行内容和版本管理。

如图 2.5 所示^[20]，需求开发分为需求获取、需求建模、需求描述和需求验证四个步骤，每个步骤的主要活动如下：

- 需求获取：通过与用户的交流等多种方式，确定和收集用户需求信息。
- 需求分析：对需求获取阶段获取的用户需求进行分析和提炼，删除错误需求，获得用户真正需要的需求，从而建立需求模型。
- 需求描述：采用适当的描述语言，从需求模型提取信息生成需求说明书，作为用户和开发人员之间的交流文档。
- 需求验证：对需求说明书进行评审，验证其是否正确和完整地描述了用户的需求。

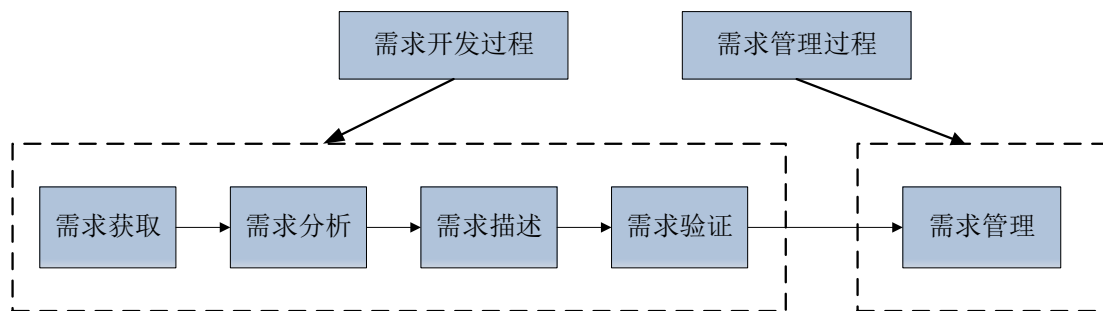


图 2.5 需求工程过程

需求管理贯穿于整个需求工程过程，它的主要任务是有效地管理需求说明书的变更、跟踪需求的状态，保持需求和产品的一致性。

2.1.2.2 需求的特性

需求说明书是需求工程的最终产物，在软件开发过程中起着十分重要的作用。产生一份高质量的需求说明书是一件十分困难的工作，好的需求说明书必须具备一定的特性。比如：无二义性、完整性、正确性、可行性、可验证性^[25]等等。

- 无二义性：要求对每项需求说明都只能有一个明确的解释，由于自然语言很容易产生歧义，而有歧义的需求将会严重阻碍开发人员和用户之间的交流，因此应该尽量用简短的语言表达需求。对可能产生歧义的需求一定要刨根问底，多次讨论，清晰地描述它们。

- 完整性：要求对系统的所有需求都进行描述，如果当软件开发后期才发现遗漏了一项需求，那将会对整个软件项目造成许多负面影响。因此，需求的完整性十分重要，在开发开始之前，一定要进行需求评审，完整的需求才能指导开发人员设计和实现完整的软件系统。

- 正确性：要求每项需求都必须是正确的，这就要求用户必须积极参与，从而捕获用户真正需要的需求，形成软件需求。因为软件需求是开发人员设计和实现的指导性文档，只有正确的需求才能产出正确的实现。

- 可行性：要求每项需求在已有的软件开发环境的限制范围内都必须是可行的。因此，为了避免产生不可行的需求，需求分析过程应该有技术人员参与，负责检查需求在技术上的可行性。

- 可验证性：要求每项需求都能够通过设计一定的测试案例或者其他方法进行验证。如果需求不可检验，那么就不能判断开发人员开发的软件是否会满足用户的需求。因此，需求分析过程也应该有测试人员参与，针对每条需求设计合理的测试案例，确保需求的可验证性。

2.2 技术基础

2.2.1 统一建模语言 UML

统一建模语言^[26]（Unified Modeling Language，简称 UML）是由 OMG 组织提出的一种标准的可视化建模语言。它融合了 OMT、OOSE 和 Booch 方法的特点，汲取了面向对象领域的优势，形成了一种标准的面向对象建模语言^[27]。目前，UML 以其简单、易于理解的特点已经得到了广泛的应用，通常用于数据建模、业务建模、对象建模、组件建模等等。

UML 图可以分为 5 大类（包括 9 中图形），包括用例图、静态图、行为图、交互图和实现图，如图 2.6 所示。UML 定义了一组标准的图形符号和文法，便于开发人员进行系统建模。

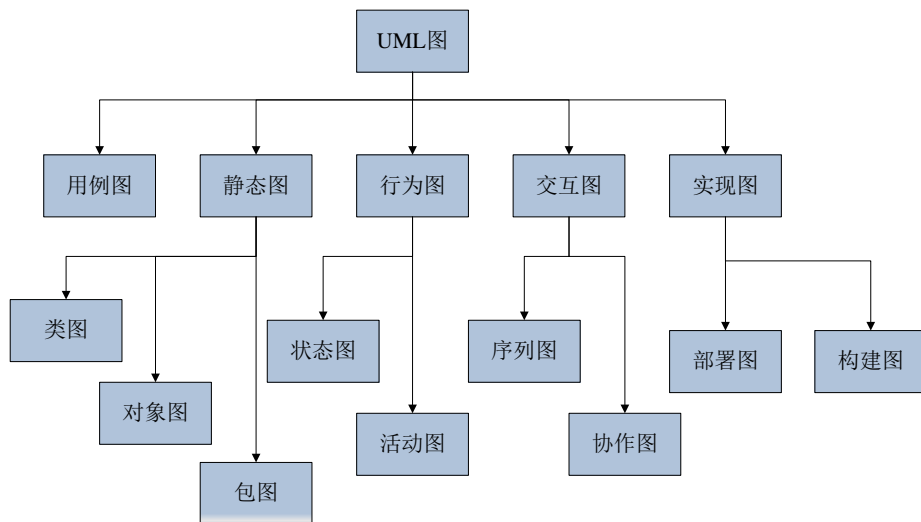


图 2.6 UML 图的分类

用例图（Use case diagram）是从用户的角度帮助开发人员理解系统的功能

需求，包括角色、用例以及它们之间的关系。

静态图 (Static diagram) 包括类图、对象图和包图。类图表示系统的静态结构，定义了系统中类，它们的属性以及类之间的关系。对象图描述的是特定时间系统的结构，比类图更详细。包图描述了一个系统的逻辑划分，以及它们之间的依赖关系。

行为图 (Behavior diagram) 包括状态图和活动图，是对系统的动态属性和交互关系的描述。状态图是对类图的补充，描述某个对象可能的状态以及状态之间的转移。活动图用于描述多个对象之间的过程控制流程。

交互图 (Interactive diagram) 包括序列图和协作图，用于描述消息的传递和对象之间的协作。序列图强调时间顺序，描述不同对象之间的传递的消息以及顺序。协作图则强调对象关系，描述对象以及他们之间的协作关系。

实现图 (Implementation diagram) 包括构件图和部署图。构件图描述了系统中的各构件以及它们之间的依赖关系。部署图描述了系统的物理体系结构，定义了实际设备以及它们之间的连接关系。

2.2.2 数据交换格式 XML

可扩展标记语言 XML (eXtensible Markup Language)^[28]起源于 SGML，是一种简单而又灵活的文本格式。2000 年，W3C 发布了 XML 1.0 版本。

XML 作为一种结构化的数据，以其简单、灵活、可扩展的特点，已经在业界得到了广泛的应用。作为一种数据交换格式，XML 可以有效地分离数据的结构和表示，并且允许用户自定义标记。此外，由于 XML 跨平台的特性，用户容易在不兼容的系统间交互数据。采用 XML 作为交换格式不仅大大提高了数据的交换效率，而且减少了数据交换的复杂性。

XML 能够方便地描述和组织数据，XML 解析器则实现应用程序从 XML 文件中提取所需数据，目前常用的两种解析器：基于 DOM 的解析器和基于 SAX 的解析器。

DOM (Document Object Model) 是 W3C 制定的一套规定解析文件接口的标准。DOM 标准的核心是树模型，解析器首先将 XML 文件读入内存，然后建立一个树模

型记录 XML 文件中的标记和标记的内容，应用程序可以通过操作树模型中的节点获得自己所需的数据。但是 DOM 解析器也存在一个缺点，当 XML 文件很大时，内存树模型会占据较多的内存空间，影响操作时间。

SAX (Simple API for XML) 提供了解析 XML 的 API，它的核心是事件处理机制，和 DOM 解析器相比，占用内存空间少，每次只在内存中保留一个事件的处理对象，处理完毕即可释放内存，可以高效地从 XML 文件获取数据。SAX 解析器的解析流程是对 XML 文件进行顺序扫描，当扫描到文件的开头、结束等特殊节点时，将通知事件处理函数进行处理，然后继续扫描直到文档结束。

2.3 AUTOSAR 体系结构

2.3.1 AUTOSAR 概述

AUTOSAR (AUTomotive Open System ARchitecture) 是一个开放的、标准的汽车电子软件体系架构^[6]。AUTOSAR 由国际上知名的汽车电子制造商、供应商以及工具开发商联合推出，自 2003 年以来目前已经经历了 4 个稳定版本，到达 4.0 版本^[29]。AUTOSAR 组织包括 3 类成员：核心成员、主要成员和相关成员，浙江大学是其中的唯一一家以高校身份加入的成员。

AUTOSAR 的提出，旨在为汽车电子软件行业提供一个统一的开放标准和软件架构，以解决传统汽车电子软件重用性低的缺陷^[30]。AUTOSAR 标准的主要目标：

- 提供标准化的接口，以减少或避免供应商和制造商各自的大量接口；
- 提供运行时环境，使功能与通信分离，提高通信的效率；
- 提供基础软件，以提高软件质量，并专注于有竞争价值的功能；
- 提供微处理器的抽象，使得处理器的更换不再需要上层软件的修改适应；
- 提供标准化的交换格式，使工具链之间可以无缝集成；
- 提高汽车电子软件的安全性、可扩展性等特性。

AUTOSAR 定义了一套标准的汽车电子软件开发过程，用以指导汽车电子软件开发商开发符合 AUTOSAR 标准的汽车电子软件工具。AUTOSAR 软件开发过程如图

2.7 所示^[31]，包括三个级别：系统级、ECU 级和组件级，逐层深入。其中，系统级定义了系统配置的流程，ECU 级定义了 ECU 设计和配置的流程，组件级定义了组件的实现流程。整个 AUTOSAR 软件开发过程为：以组件设计为起点，生成的组件描述作为系统配置的输入，再进行系统配置，然后通过 ECU 抽取得到多个 ECU 初始描述，分别导入 ECU 配置器进行 ECU 配置，完成 ECU 配置后，生成系统可执行代码。各工具之间采用符合 AUTOSAR 标准的统一接口——ARXML 文件。

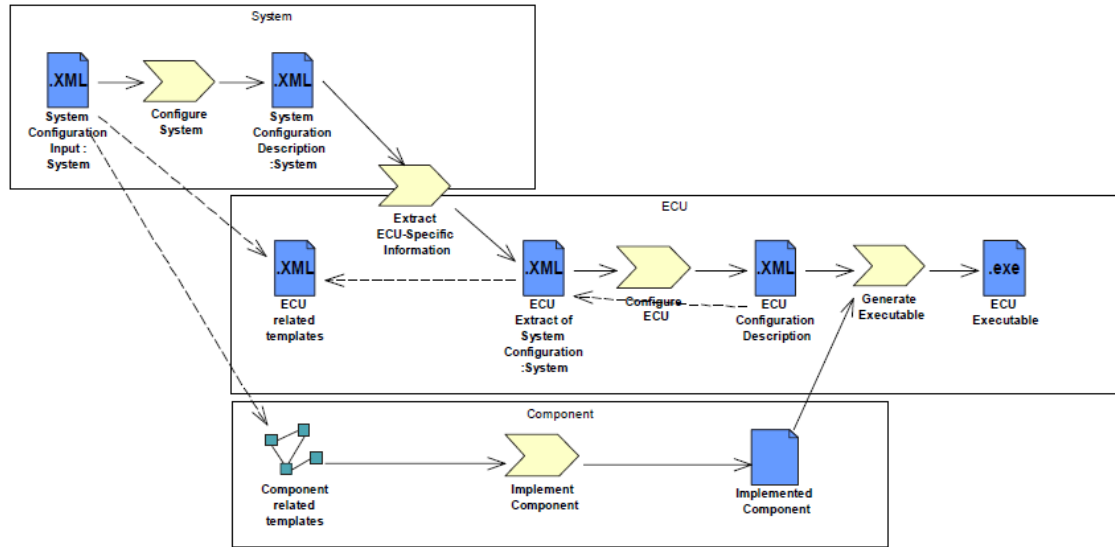


图 2.7 AUTOSAR 开发过程

2.3.2 Artop 基础平台

Artop (AUTOSAR Tool Platform)^[32]是基于 Eclipse 开发的 AUTOSAR 工具平台，是一种用于 AUTOSAR 标准设计和 ECU 设计的基础开发平台。

Artop 希望为所有 AUTOSAR 开发工具提供公共基础平台。因为所有的 AUTOSAR 开发工具都需要一些基本功能，而这些功能没有哪家开发商可以有独到的实现方法，与其让每个公司分别实现这些基本功能，不如提供一个公共的基础平台来改进工具的质量、互操作性和开发效率。Artop 就是基于这样的想法而诞生的，Artop 源码对于 AUTOSAR 成员和合作者免费开放，它的发展也由他们共同推动。

Artop 提供了一系列基本构建组件，如图 2.8 所示^[33]。其中，Artop AUTOSAR 层提供了 AUTOSAR 不同版本的元模型实现、AUTOSAR 验证引擎、XSD 的序列化、

工作区间管理器等等。Artop Eclipse 补充层也是 Artop 用户组开发的 Artop 基础平台，但是没有 AUTOSAR 特定的部分，因此其他非 AUTOSAR 工具也可以共享。

Artop 的主要目标是使 AUTOSAR 工具与 AUTOSAR 标准日益一体化，用基础共享的方式建立 AUTOSAR 商业软件平台，改善 AUTOSAR 工具商之间的合作，使其在 Artop 上层领域进行良性竞争。

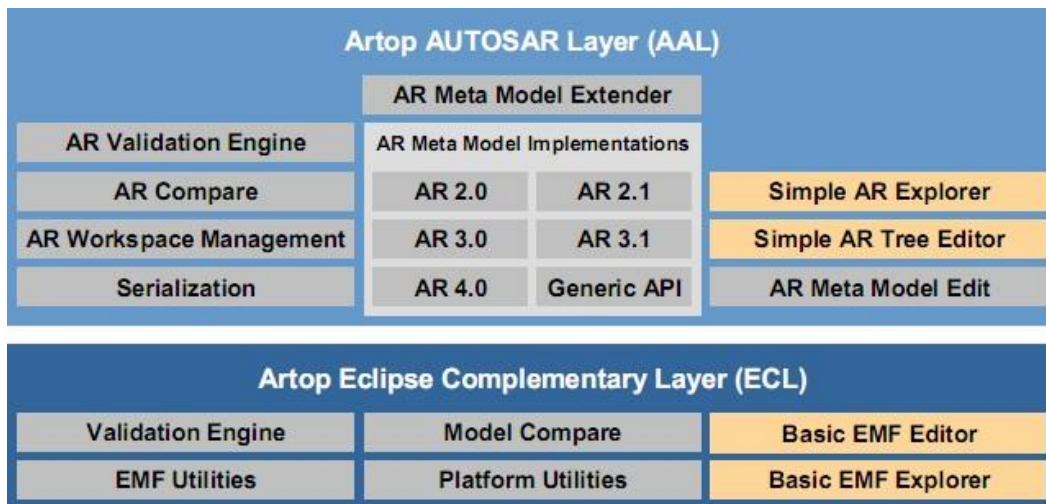


图 2.8 ARTOP 组件

2.3.3 AUTOSAR 软件开发平台

随着 AUTOSAR 标准的影响力日益扩大，许多汽车电子软件开发商也推出了自己的 AUTOSAR 解决方案，典型代表是 Vector、MathWorks、dSPACE、EB 等公司。它们都围绕 AUTOSAR 为目标，或是全新开发，或是整合原有的产品和工具，旨在提供 AUTOSAR 专用开发方法和工具平台，用于快速构建符合 AUTOSAR 标准的汽车电子软件。

Vector^[34]公司提出的 AUTOSAR 解决方案，包括 DaVinci 工具链和 MICROSAR 配置工具。DaVinci 是 Vector 提供的一条汽车分布式电子系统设计工具链，包括三大组件：

- DaVinci System Architect：用于车辆的系统级设计；
- DaVinci Network Designer：用于网络通信设计，支持对不同总线（包括 CAN、LIN 和 FlexRay）进行系统架构和通信序列的设计；

- DaVinci Developer: 用于 ECU 的开发, 不仅能集成针对不同嵌入式器件的代码, 而且能自动生成 ECU 各控制算法模块的软件接口代码。

MICROSAR 配置工具提供对基础软件的配置。如图 2.9 所示^[35], MICROSAR 包括操作系统、微控制器抽象层、通信栈(支持 CAN、LIN、FlexRay 等总线)、诊断、运行时环境等模块。开发人员可以忽略不同硬件平台的差异, 先进行应用程序的开发, 并且可以利用 CANoe 进行仿真和调式。

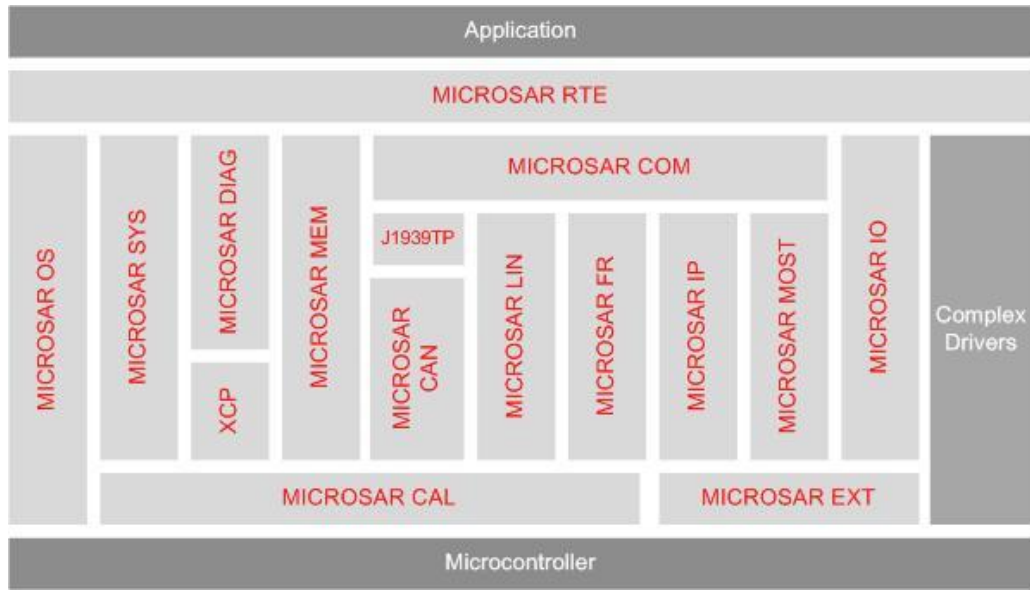


图 2.9 MICROSAR 体系结构

此外, Vector 还与 MathWorks 公司合作, 用户使用 Vector 的 DaVinci SystemArchitect 设计组件架构, 然后将组件描述导出到 Simulink 中进行组件内部行为设计, MathWorks 的 Real-Time Workshop Embedded Coder 可以自动生成组件代码, 并回读到 DaVinci 中, 然后用户可以使用 DaVinci 工具链配置 MICROSAR RTE。

dSpace 公司的 AUTOSAR 解决方案主要包括 SystemDesk 和 TargetLink 两个工具。SystemDesk 支持 AUTOSAR2.1 和 3.0 版本, 主要致力于应用层和系统设计的部分, 如图 2.10 所示^[36, 37]。SystemDesk 的主要功能:

- 提供图形化的组件设计, 包括接口和内部行为设计
- 提供硬件拓扑设计和网络通信设计

- 支持多种系统仿真功能，包括开环和闭环仿真、VFB 系统仿真、CAN 总线仿真、虚拟仿真运行等。
- 生成 RTE（运行时环境）
- 支持基本的 ECU 配置和基础软件配置

TargetLink 工具包支持与 SystemDesk 协同工作，主要用于生成 SystemDesk 中的模型代码，并进行代码的产品级优化。

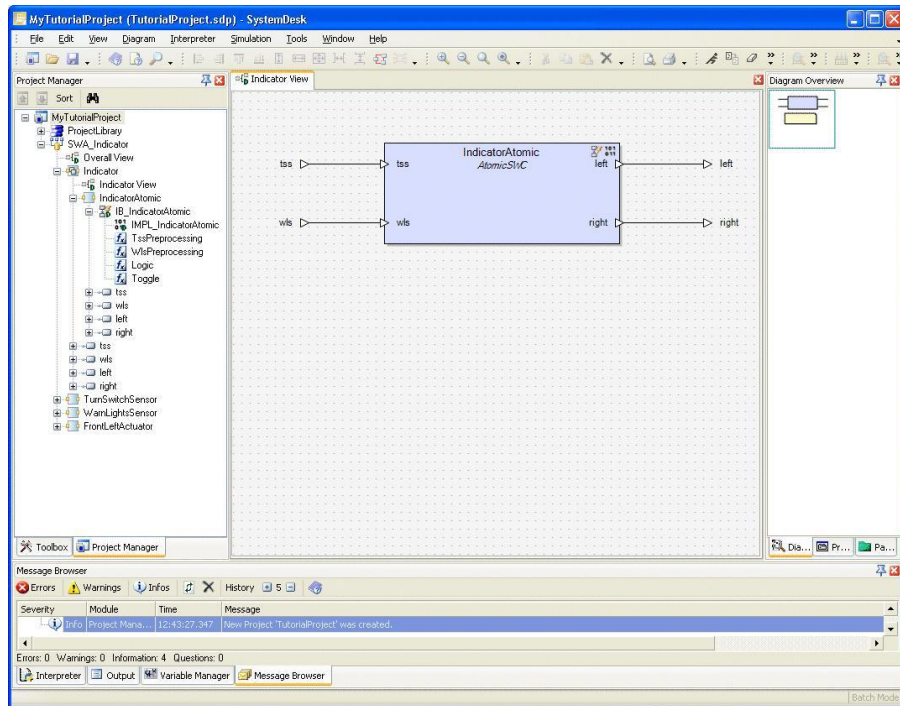


图 2.10 SystemDesk 用户界面

EB 公司^[38]提供的 EB tresos studio 产品是一个符合 AUTOSAR 标准的 ECU 配置和代码生成工具。EB tresos studio 支持基础软件模块的配置、一致性验证和代码生成。同时，EB 的 AUTOSAR 解决方案还包括 EB tresos AutoCore^[39]，主要实现 OS、RTE、内存管理、通信栈等功能。

EB tresos studio 也支持 AUTOSAR 2.1 和 3.0 标准，与 dSpace 合作，推出了一套完整的 ATUSOAR 解决方案。EB tresos 主要负责 ECU 的配置和生成，由 SystemDesk 完成组件设计和系统设计。

MathWorks^[40]公司的 Matlab 产品因其在控制领域强大的影响力被大家所熟知。

2008 年, MathWorks 发布的 R2008 版本的 Simulink 开始支持 AUTOSAR 模型的简单设计以及组件描述与代码的生成。其后续的 R2009 与 R2010 版本也在不断地增加对 AUTOSAR 的支持。Simulink 为开发 AUTOSAR 组件提供了便利的内部行为设计功能, 并且可以自动生成符合 AUTOSAR 标准的组件描述和代码。

此外, NEC^[41]、ETAS^[42]、Freescale^[43]、Softing^[44] 等公司也提出了自己的 AUTOSAR 工具, 但都以某个模块为主, 未能提供完整的解决方案。如 NEC 主要致力于微控制器抽象层的研究, ETAS 则主要关注 OS 和 RTE, 等等。

2.4 本章小结

本章介绍了本文的研究基础。其中, 领域基础介绍了多种软件开发模型和需求工程过程, 以及高质量的需求应具备的特性。技术基础介绍了统一建模语言 UML, XML 交换格式。最后介绍了本文所基于的 AUTOSAR 标准、AUTOSAR 公共基础开发平台 Artop 以及当前的 AUTOSAR 软件开发平台。

第3章 SmartSAR IDE 平台总体分析

本章在深入调研汽车电子软件领域和技术的基础上，结合当前 AUTOSAR 工具软件的优点，提出了基于 AUTOSAR 标准的汽车电子软件开发平台——SmartSAR IDE。本章将分别从功能、界面、接口、性能等角度对 SmartSAR IDE 平台的进行需求分析。

...(此处省略中)

第4章 SmartSAR IDE 平台总体设计

通过对 SmartSAR IDE 平台的总体分析，SmartSAR IDE 平台被划分为四大工具，自上而下依次为模型设计器、系统配置器、ECU 配置器和 RTE 生成器。本章将在上一章节总体分析的基础上对各个工具进行总体设计。对 SmartSAR IDE 平台的设计是各个工具进行实现的基础，并为各个工具的实现提供了指导性准则。

...(此处省略中)

第5章 SmartSAR IDE 模型设计器的实现与应用

…(此处省略中)

第6章 总结与展望

6.1 工作总结

本文以汽车电子软件的开发为背景，研究了领域内多种软件开发模型和需求工程方法，在深入调研汽车电子软件领域和技术的基础上，吸取当前汽车电子软件开发工具的优点，针对 AUTOSAR 标准和汽车电子领域的特殊需求，提出了基于 AUTOSAR 标准的汽车电子软件开发平台——SmartSAR IDE。

本文的主要工作和特点包括：

- 1) 本文从功能、性能、接口、界面等多方面对 SmartSAR IDE 平台进行总体需求分析，降低了在软件开发早期引入错误的概率，提高了软件系统的开发效率。
- 2) SmartSAR IDE 平台提供了集模型设计器、系统配置器、ECU 配置器、RTE 生成器于一体的完整汽车电子开发过程，使得汽车电子软件开发更加规范、高效。
- 3) SmartSAR IDE 模型设计器采用图形化的建模环境设计，系统配置器和 ECU 配置器采用可视化的配置界面设计，为用户提供了简单、友好、灵活、易操作的交互界面，大大降低了用户的开发成本。
- 4) SmartSAR IDE 平台采用符合 AUTOSAR 标准的数据定义格式，为各个工具提供了统一的交互接口，便于工具的集成和扩展，同时也为与其他 AUTOSAR 工具的交互提供了便利。
- 5) SmartSAR IDE 平台提供了模型验证功能设计，通过对模型的实时验证和及时反馈，保证了用户建立模型的正确性和完整性，降低了系统开发的风险。
- 6) 本文完成了 SmartSAR IDE 平台模型设计器的实现，并在车身控制系统上实现了对模型设计器的应用。

6.2 工作展望

…(此处省略中)

参考文献

- [1] 中国汽车报. 解读中国汽车产销量跃居世界第一[EB/OL]. [2010-01-11].
<http://www.cnautonews.com/>.
- [2] 中国汽车工业协会 [EB/OL]. [2010-06-07].
<http://www.caam.org.cn/newslst/a34-1.html>.
- [3] Z. Yu, X. Zhou, D. Zhang, et al. Supporting Context-Aware Media Recommendations for Smart Phones[J]. IEEE Pervasive Computing, 2006, 5:68-75.
- [4] J. Ma, L. T. Yang, B. O. Apduhan, et al. Towards a smart world and ubiquitous intelligence: A walkthrough from smart things to smart hyperspaces and UbicKids[J]. International Journal of Pervasive Computing and Communications, 2005, 1:53-68.
- [5] H. Heinecke, W. Damm, B. Josko, et al. Software components for reliable automotive systems[C]. in Proceedings of the conference on Design, automation and test, Europe, 2008, 549-554.
- [6] AUTOSAR. AUTomotive Open System ARchitecture[EB/OL]. [2009-11-13].
<http://www.autosar.org/>.
- [7] JasPar. Japan Automotive Software Platform and Architecture[EB/OL]. [2008-08-20]. https://www.jaspar.jp/english/index_e.php.
- [8] K. Grimm. Software technology in an automotive company: major challenges[C]. in Proceedings of the 25th international conference on Software Engineering, 2003, 498-503.
- [9] M. Broy. Challenges in automotive software engineering[C]. in Proceedings of the 28th International Conference on Software Engineering, Shanghai, China, 2006, 42.
- [10] A. Pretschner, M. Broy, I. H. Kruger, et al. Software Engineering for Automotive Systems: A Roadmap[C]. presented at the Future of Software Engineering. Minneapolis, Minnesota, USA, 2007.

- [11] A. Kleppe, J. Warmer, and W. Bast. MDA Explained: The Model Driven Architecture: Practice and Promise[M]. Addison Wesley Longman Publishing Co., Inc., 2003.
- [12] OMG. MDA[EB/OL]. [2010-05-20]. <http://www.omg.org/mda>.
- [13] T. Meservy and K. Fenstermacher. Transforming Software Development: An MDA Road Map[J]. Computer, 2005, 9:52-58.
- [14] Berkeley. Ptolemy[EB/OL]. [2010-10-24]. <http://ptolemy.eecs.berkeley.edu>.
- [15] Vanderbilt. GME[EB/OL]. [2008-06-12]. <http://www.isis.vanderbilt.edu/Projects/gme>.
- [16] ISIS. GME 5 User's Manual[R]. Institute for Software Integrated Systems, Vanderbilt University. 2005.
- [17] Esterel-Technologies. SCADE(Safety-Critical Application Development Environment)[EB/OL]. [2008-12-15]. <http://www.esterel-technologies.com/>.
- [18] IBM. Rational Rhapsody[EB/OL]. [2010-05-10]. <http://www-01.ibm.com/software/awdtools/rhapsody/>.
- [19] B. P. Douglass and C. Evangelist. Model Driven Architecture and Rhapsody[R]. I-Logix, 2000.
- [20] 毋国庆, 梁正平, 袁梦霆. 软件需求工程[M]. 机械工业出版社, 2008.
- [21] T. Matthews. A process review and appraisal of the system engineering capability for the Florida Department of Transportation[R]. Florida Department of Transportation ITS Office, 2003.
- [22] R. S. Pressman 著, 郑人杰, 马素霞等译. 软件工程实践者的研究方法[M]. 机械工业出版社, 2009.
- [23] J. Kivi, D. Haydon, J. Hayes, et al. Extreme programming: a university team design experience[C]. in Electrical and Computer Engineering, 2000 Canadian Conference, Halifax, NS, 2000, 816.
- [24] K. Schwaber and M. Beedle. Agile software development with Scrum[M]. Prentice Hall Upper Saddle River, NJ, 2002.
- [25] 黄国兴, 周勇. 软件需求工程[M]. 清华大学出版社, 2008.
- [26] OMG. UML[EB/OL]. [2010-05-20]. <http://www.omg.org/uml>.

- [27] P.Green, M. Edwards, and S. Essa. UML for System- Level Design, Forum on Design Languages[C]. in Proceedings of FDL 2001, Lyon, France, Sept. 3- 7, 2001.
- [28] T. Bray, J. Paoli, C. M. Sperberg-McQueen, et al. Extensible markup language (XML) 1.0[J]. W3C recommendation, 2000, 6.
- [29] S. Fürst and e. al. AUTOSAR – A Worldwide Standard is on the Road[R]. AUTOSAR GbR, Muniche, 2009.
- [30] U. Freund. Mult-level system integration based on AUTOSAR[C]. presented at the Proceedings of the 30th international conference on Software engineering, Leipzig, Germany, 2008.
- [31] A. GbR. Specification of the Virtual Functional Bus V1.0.1. 2008.
- [32] M. Rudorfer, S. Voget, and S. Eberle. Artop (AUTOSAR Tool Platform) whitepaper[R]. 2008.
- [33] Artop. AUTOSAR Tool Platform(Artop)[EB/OL]. [2010-06-20]. <http://www.artop.org/>.
- [34] Vector. AUTOSAR - A Choice for the Future! [EB/OL]. [2010-11-06]. http://www.vector.com/vi_autosar_solutions_en.html.
- [35] MICROSAR. MICROSAR - Your Basic Software for AUTOSAR[EB/OL]. [2010-12-02]. http://www.vector.com/vi_microsar_en.html.
- [36] dSPACE. dSPACE[EB/OL]. [2010-12-23]. <http://www.dspace.com/>.
- [37] O. Niggemann and M. Beine. SystemDesk and TargetLink –AUTOSAR-Compliant Development at System and Function Level[R]. 2007.
- [38] EB. EB tresos: The Product Family for the Development of ECU Software[EB/OL]. [2010-12-25]. http://www.elektrobit.com/what_we_deliver/automotive_software/products.
- [39] EB. tresos AutoCore[EB/OL]. [2010-12-25]. <http://www.elektrobit.fi/>.
- [40] MathWorks. MATLAB and Simulink for Technical Computing[EB/OL]. [2010-09-03]. <http://www.mathworks.com/>.
- [41] NEC. NEC[EB/OL]. [2009-12-02]. <http://www.nec.com/>.
- [42] ETAS. ETAS and AUTOSAR[EB/OL]. [2009-10-01]. <http://www.etas.com/zh/>.

-
- [43] Freescale. AUTOSAR OS and MCAL[EB/OL]. [2010-11-18].
<http://www.freescale.com/>.
 - [44] Softing. AUTOSAR ECU Software[EB/OL]. [2009-12-02].
<http://www.softing.com/home/>.
 - [45] R. Wu, H. Li, J. Wang, et al. A Hierarchical Modeling Method for AUTOSAR Software Components[C]. presented at the The 2nd International Conference on Computer Engineering and Technology (ICCET), Chengdu, China, 2010.

攻读硕士学位期间主要的研究成果

项目：

- [1] 基于模型驱动的汽车电子安全软件开发技术，国家 863 项目，项目编号：2007AA01Z129，主要参与者，项目组组长，负责项目的研发和验收工作。
- [2] 面向嵌入式软件的生产线，国家 863 项目，项目编号：2007AA010304，主要参与者，并负责项目验收工作。
- [3] 汽车电子虚拟服务总线及其模型设计工具，国家核高基项目，项目编号：2009ZX01038-001-06，主要参与者，负责项目的总体分析和设计工作，并承担模型设计器的实现工作。

发表论文：

- [1] Ruyi Wu, Hong Li, Jinbo Wang, Yuhao Yang, "A Hierarchical Modeling Method for AUTOSAR Software Components", In The 2nd International Conference on Computer Engineering and Technology (ICCET2010), April 2010, Chengdu, China, pp184-188.(EI 索引)

专利：

- [1] 李红、杨国青、伍如意、王超、杨玉皓、汪金波，UML 模型到 AUTOSAR 软件组件模型的转换方法，申请号：201010118856.0，已受理
- [2] 杨国青、胡博、王冬冬、伍如意、杨玉皓、汪金波、赵民德，于内存数据结构的模块间实时同步方法，申请号：10186142.3，已受理
- [3] 李红、胡博、杨玉皓、汪金波、王超、伍如意、王冬冬，基于索引的汽车电子软件组件模型仓库的实现方案，申请号：201010182629.4，已受理

软件著作权：

- [1] 杨国青、李红、赵坚、黄凯、胡博、吕攀、张莉苹、伍如意，等，基于模型驱动的汽车电子安全软件开发平台 SmartOSEK IDE，登记号：2009SR049616

致谢

两年半的研究生生涯即将画上一个圆满的句号，我也将离开校园去开启人生的另一个篇章。在此，我想对所有关心和支持我的可爱的人们致以最真诚的谢意。

首先感谢吴朝晖教授，您带领的 CCNT 实验室团队创造了一个广阔的平台，让我能在这个舞台上尽情地挥洒，迎接一个又一个新的挑战。您高昂的激情、新颖的理念深深地感染着我，是我学习的目标。

感谢我的导师姚敏教授，您的平易近人、渊博的学识、严谨的态度给我留下了深刻的印象，是我学习的榜样，谨向您表示由衷的谢意。感谢 ESE 团队的所有老师们：赵民德、顾宗华、杨国青、李红。感谢你们一直以来的关心和帮助，让我在实验室能够快速成长，并深深地感受着这个大家庭的温暖。

特别感谢杨国青和李红老师。感谢杨老师在论文撰写期间给予的悉心指导，您一直以来的关心、鼓励和支持，是我不断克服困难继续前行的动力。您源源不断的热情、独到的思维和深刻的见解是我努力的方向和前进的目标。感谢李老师无论在工作还是生活上对我的无私帮助和悉心照顾，您的细心、随和、严谨深刻地影响着我，是我学习的榜样。

感谢实验室 08 级的伙伴们：王冬冬、王超、李楠、彭威、周霖、周丽、张翼、胡琦，感谢你们与我一起经历美好而难忘的研究生生活。感谢可爱的师弟师妹们：耿莉莉、汪金波、杨玉皓、钟晓峰、孙征、刘敏、邓俊、田帅、赵庆玲等，你们为实验室带来了青春和活力。感谢已经毕业的师兄师姐们：张莉苹、彭巍、胡博、赵坚、吕攀、姜旭峰等，祝你们前程似锦。

感谢我的父母在我漫长的求学路上始终如一的支持和鼓励，无论遇到多大的困难和挫折，你们永远是最坚强的后盾和最温馨的港湾，我的每一点进步都离不开你们的支持。我会继续努力，去开创更美好的明天。

伍如意

二零一一年一月求是园