



Our heart
beats
embedded.

汽车微控制器

软件开发中的五大挑战

汽车制造商和供应商指南

摘要

作为车辆实时能力和功能安全的关键推动要素，微控制器在汽车行业中继续发挥着至关重要的作用。然而，优化微控制器软件的开发过程通常不是首要任务，市场更关注差异化和创新的应用，而不是在后台工作的嵌入式功能。在通往软件定义车辆的道路上，基本的开发任务往往会被搁置一边。但是，正是这一过程的优化保证了汽车制造商和供应商未来的成功。

为了促进最先进的汽车软件开发过程，ETAS 整理了客户面临的五大挑战：高集成工作，标定复杂，测试耗时，可扩展性和灵活性限制，以及整体网络安全要求。根据 V 模型的具体过程步骤，本白皮书描述了这些挑战并提供了适当的解决方案，这将帮助您提高开发过程的效率、减少工作量、实施更高的安全标准，并最终跟上快速发展的市场步伐。

目录

1. 介绍	4
2. 世界车辆微控制器概览	5
3. 车载单片机软件的开发过程	6
3.1. ECU 生成变化的原因	6
3.2. 开发过程中的主要步骤	7
4. 制造商和供应商面临的主要挑战	9
4.1. 高集成工作	9
4.2. 标定复杂	10
4.3. 测试和调试耗时	11
4.4. 可扩展性和灵活性限制	12
4.5. 整体网络安全需求	13
5. 结论	14

1. 介绍

安全性、实时性、资源优化——这些特性在很大程度上取决于电子控制单元 (ECU)，特别是微控制器，它们是现代车辆中各种重要功能的基础。但是，原始设备制造商 (OEM) 和供应商面临的创新压力往往导致他们忽视 ECU 软件的开发过程，而倾向于更令人兴奋的技术。尽管如此，ECU 仍有很大的改进潜力：更多的自动化、更高的效率、克服安全限制，等等。随着需求的增加和创新速度成为决定性因素，不更新这些基本流程最终会产生不良影响。

在一辆典型的现代汽车开发过程中，有很多这样的基础工作要做。车辆代码的大部分分布在微控制器上，在不被用户注意到的情况下完成其日常工作。然而，这种深度嵌入式软件在车辆中具有非常特殊的责任：汽车部件的故障可能对乘客的健康产生严重影响。因此，汽车行业的要求也很高，尤其是在满足不断变化的安全法规方面。嵌入式微控制器系统的软件开发虽然是一项日常任务，但它极其重要——而且在未来几年里仍将如此。

本白皮书旨在帮助制造商和供应商重新组织其流程，以实现最先进的微控制器软件开发。首先介绍了车辆微控制器的重要性以及为 ECU 代际更换开发新软件所需的步骤。然后深入探讨了主要主题：开发过程中面临的五大主要挑战及可能的解决方案。并非每家汽车公司都会同样受到所有挑战的影响。一些制造商和供应商并不在内部完成所有步骤。其他因素同样具有决定性作用，例如数字化的成熟度、现有工具的组成、遗留软件的数量以及硬件设置。尽管如此，了解全局、常见的困难和潜在的解决方案对汽车行业的每个参与者都是有益的。

2. 世界车辆微控制器概览

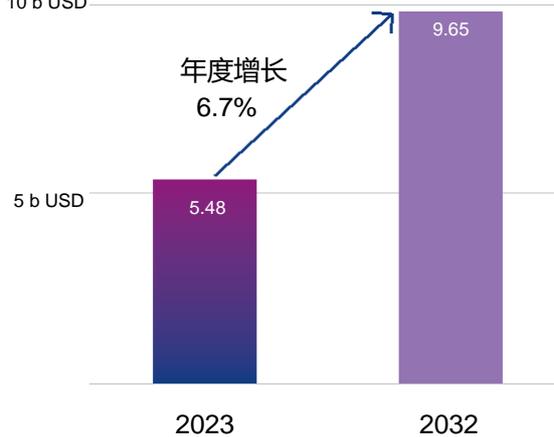
车辆中的大多数控制和通信任务都是由各种微控制器作为 ECU 的核心来完成的：从用于传感器接口和控制的实惠的 8 位微控制器，到用于现代信息娱乐显示器和车辆动态系统的 32 位微控制器。标准的现代车辆包含超过 40 个这样的电子组件，高端车型甚至多达 150 个。这些组件确保了例如安全气囊、ABS（防抱死制动系统）、ESC（电子稳定控制系统）、发动机控制、轮胎压力和电池状态等功能的可靠性。所有这些都后台进行，驾驶员察觉不到。

目前，OEM 和供应商正面临着越来越短的 E/E 架构创新周期，以及越来越多的车辆型号和功能。因此，基于微控制器的软件开发也必须变得越来越高效。与软件定义汽车相关的趋势，如软硬件解耦、集中式车载计算机或车载云计算，只是在表面上取代了传统的 ECU 方法。实际上，嵌入式系统将继续发挥决定性作用，因为它们是确保实时能力和功能安全达到 ASIL-D 的唯一途径。市场报告预测微控制器领域将有相当大的增长（见图 1）。因此，原始设备制造商和供应商必须调整 ECU 软件的开发过程，以适应新的市场对速度、灵活性和安全性的要求。

嵌入式微控制器系统的软件开发既不是新的，也不是汽车行业所独有的。这些组件安装在电子设备和日常用品中，从牙刷到吸尘器。那为什么在汽车领域开发过程显得特别呢？最重要的方面是从车辆上路的第一分钟起就需要高安全性和可靠性要求，其次是几十年的可维护性。此外，可靠的实时能力对于动力系统控制或车辆动力学等许多系统至关重要：电子控制系统必须始终与车辆或环境中展开的物理过程保持同步。

虽然微控制器编程对于 OEM 和供应商来说是一项周期性的常规任务，但随着每一代的变化，使其适应高监管要求，同时使其在全球 OEM 和供应商网络中尽可能高效是具有挑战性的。对于那些使用长期建立的流程链且工具和方法仅部分兼容当前要求的公司来说，这尤其如此。

图 1：全球微控制器市场增长
10 b USD



汽车微控制器的全球市场规模预计将以每年 6.7% 的速度增长。

来源：<https://www.fortunebusinessinsights.com/de/markt-f-r-mikrocontroller-f-r-die-automobilindustrie-104084>



简要解释：

ECU、MCU、MPU 和 VCU 的协调

汽车系统中使用的电子控制单元（ECU）是管理车辆中特定功能的专用计算机。在这些 ECU 中，微控制器（MCU）处理实时任务，如发动机控制和传感器管理。微处理器（MPU）则用于更高级的 ECU，提供更高的处理能力，用于复杂的应用，如信息娱乐系统和高级驾驶辅助系统（ADAS），通常运行在 Linux 等操作系统上。车辆计算机（VCU）协调这些 ECU，确保整个车辆网络的无缝通信和操作。

3. 车载单片机软件的开发过程

E/E 架构正朝着更加集中的方式进行重组，但基于微控制器的 ECU 将继续存在。嵌入式软件的一般开发过程理论上可以继续保持不变。然而，为了满足对功能性渴求的市场需求，开发周期将急剧加快，正如我们在流行的软件定义车辆 (SDV) 中所遇到的那样。传统的方法无法跟上这种速度，然而，它们也不能简单地在一夜之间被取代。在许多情况下，最佳解决方案是将经过验证的流程和工具与创新解决方案相结合。

首先，重要的是要了解流程的触发因素，并将该流程中的各个步骤分离出来。

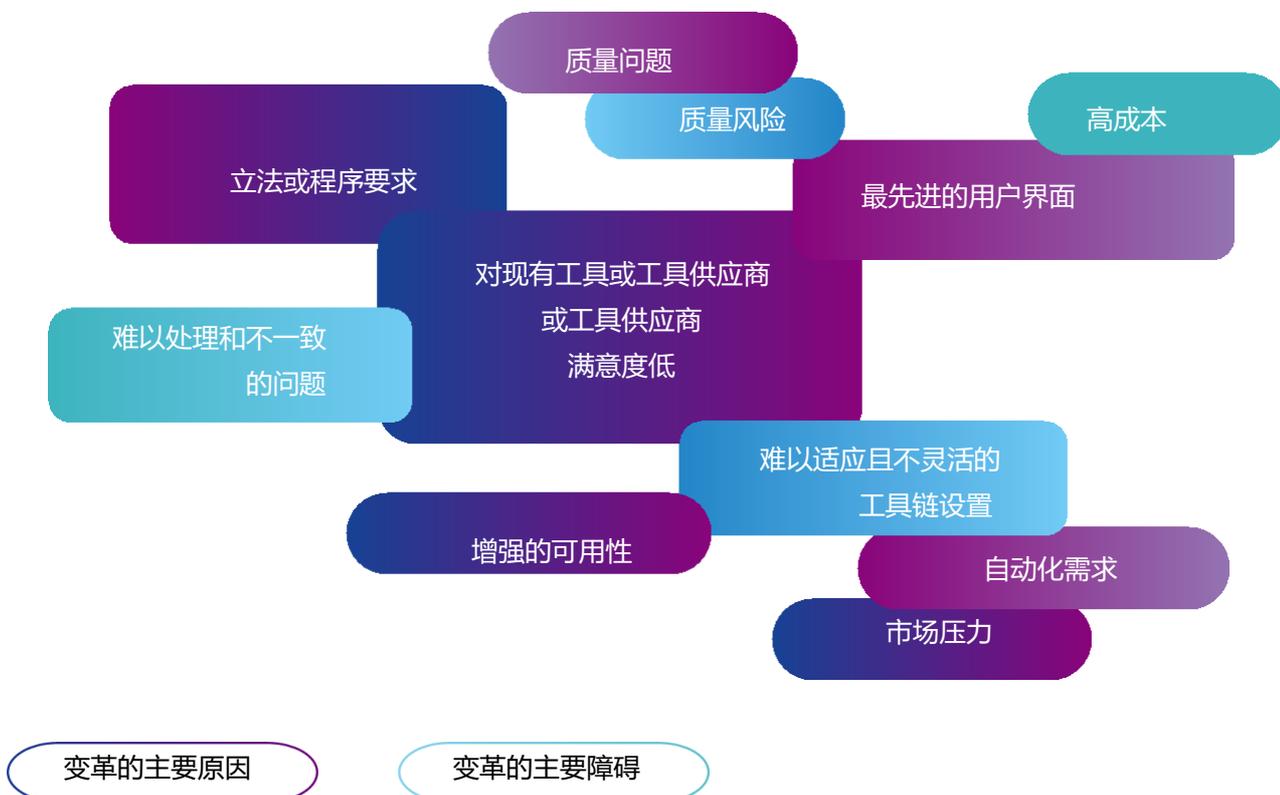
3.1 ECU 生成变化的原因

在深入探讨具体挑战之前，让我们简要地看一下为什么必须首先开发新的 ECU 软件，以及为什么这个过程如此耗时且复杂。一切都始于从一代 ECU 向下一代的转变，这需要对软件进行大幅度的重塑。

在车辆 ECU 软件开发过程中，程序员在开始创建新软件之前，必须投入精力进行工具链设置。每一代新的 ECU 都有其特定的开发过程，取决于具体的领域和合作模式。相应的工具链是为即将到来的这一代单独定义的，因此可能会包含相当大的变化，这与通过更新解决的小调整形成对比。指导原则是尽可能少地（或仅在必要时）从一代到下一代进行更改，因为新的工具链总是需要高昂的迁移成本和项目风险。

从一代到下一代需要进行这些变更的原因有很多（见图 2）。有些是内在的，例如质量问题或对当前工具或工具供应商的不满意程度较低。其他则是由外部因素决定的，例如新的立法或流程要求（例如，ASPICE, ISO 26262 ASIL-D）。市场压力或增强的可用性和最先进的用户界面也起着决定性作用。代际变更最常见的障碍包括高成本、质量风险和不一致的遗留问题，即难以适应、不灵活的工具链设置。主要需求是提高流程中的自动化水平。

图 2: ECU 生成变化的原因和障碍

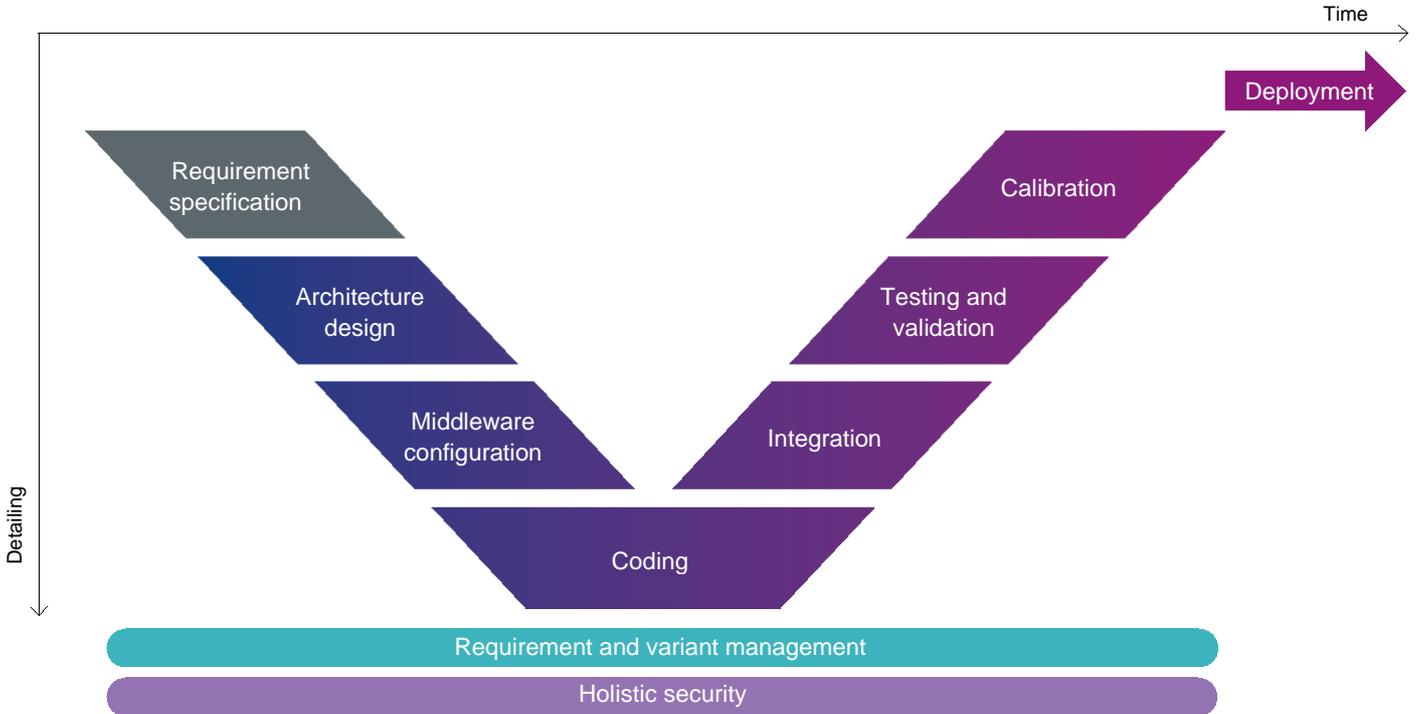


3.2 开发过程中的主要步骤

嵌入式软件开发过程的一个常见示例是 V 模型，它有不同的变体。根据我们的经验，我们将阶段划分为以下主要工作步骤，如下面的图 3 所示。

它们不一定一个接一个地处理。有些可以并行运行，或者必须多次执行。需求和变体管理在所有工作步骤中都是必要的，并且需要多次迭代。

图 3: 车辆软件开发的 V 模型



V 模型将软件开发分为两个主要部分。此版本已适应 ECU 软件开发过程

体系结构设计

作为 ECU 的一部分，微控制器负责特定的互连功能，如发动机或车辆动态控制。因此，非常详细的需求目录是关键。第一步包括识别所有功能、约束、硬件元素、中间件及其相互依赖关系。在这个早期阶段，所有硬件和软件组件都必须符合严格的汽车标准，需要对细节进行细致入微的关注。

中间件配置

中间件是指 ECU 中位于操作系统或硬件与应用软件之间的软件层。它可以作为桥梁，促进车辆软件架构中各个组件之间的通信和数据交换。通过将应用程序和与硬件相关的系统解耦，中间件使独立开发、维护和升级它们成为可能。中间件充当中介，抽象底

层硬件并为软件组件提供标准化接口，以实现无缝交互。

不言而喻，这样的中间件必须符合最高的安全标准，因为 ECU 中的漏洞和合规性问题可能导致未经授权的访问、数据泄露和系统故障，从而危及车辆的安全性和可靠性。确保这种水平的合规性符合严格的行业标准需要广泛的测试和验证，这增加了开发过程的复杂性和时间。因此，中间件的选择至关重要，重点在于成熟度（符合 ISO 26262 ASIL-D）、未来导向（无供应商锁定）和网络安全。只有一个健壮且高度可配置的中间件解决方案，并且具有持续更新，才能保护 ECU 免受不断发展的威胁，并满足所有（不断变化的）法规需求。

编码

一旦定义了体系结构，并且确定了具有特定需求的所有组件，开发人员就可以将系统设计转换为功能软件。所需的功能已经存在于标准流程中。在这里，软件开发人员编写新的（补充）功能，消除错误，并优化或扩展现有功能。挑战在于建立一个确保实现现有功能而无需重写它们的可能性的过程。这意味着高可重用性，同时保持高水平的功能安全、网络安全和代码效率。

集成

在这一点上，所有的工作流都可以聚集在一起：中间件配置、架构设计信息、应用软件以及预校准数据都被构建为可以闪现到微控制器上的代码。

一个主要的挑战在于优化，即平衡汽车功能的性能与新硬件的能力，以满足特定的效率目标和环境条件。这涉及到对中间件进行微调，以实现高效的系统资源使用和实时操作。当然，安全性和合规性也至关重要：满足严格的安全标准和监管要求意味着执行严格的测试和认证流程。

测试和验证

测试和验证阶段确保 ECU 的功能符合有关安全性、性能和可靠性的所有要求和规范。广泛的测试可以识别可能出现的任何潜在问题或故障。在软件密集型车辆的情况下，这个过程可能变得非常复杂和漫长。因此，在尽可能短时间内执行尽可能多的测试对于最小化成本和最大化性能至关重要。

校准

在校准过程中，参数和字段填充数据，确保软件的行为适应物理系统。有些应用程序可能包含成千上万个相互影响的参数。此外，参数需要满足特定的性能目标、环境条件和监管标准。调整必须在以后的更新中实施，例如当排放标准发生变化时。

部署

一旦软件达到“可用”阶段，它将经过最终的批准过程，并完全刷写到目标 ECU 上

4. 制造商和供应商面临的主要挑战

现在我们对流程中涉及的步骤有了一个很好的概述，在此过程中更容易匹配具体的挑战。有些只局限于其中一个步骤，有些则适用于整个过程或从一代到另一代的转变。我们已经确定了我们在整个汽车行业中反复遇到的五大挑战。

4.1 高集成工作

车辆系统几乎总是在棕色环境中开发，即在现有的软件环境中开发，这导致在集成过程中出现许多遗留问题，从而使开发过程变得复杂。各种组件之间多样且复杂的通信接口需要复杂的配置和广泛的测试，以确保无缝互操作性。此外，依赖手动流程进行集成任务会导致错误率增加和开发时间延长。

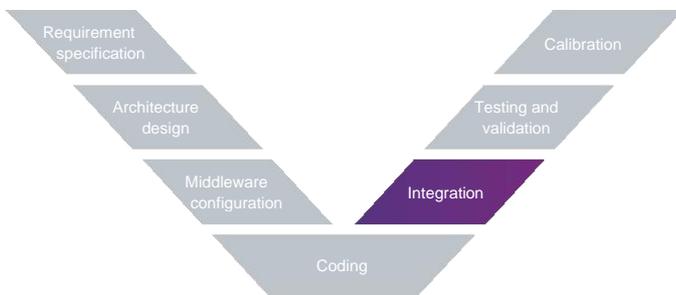


图 4：V 模型过程步骤中的“集成”

解决挑战的方法

渐进式现代化

逐步用现代等效组件更新遗留组件，确保向后兼容，以便顺利过渡和与新系统集成，而不会完全推翻既定流程或冒高投资成本的风险。

集成任务的自动化

实现自动化的工具和脚本来处理重复的集成任务，减少错误率并加速开发周期。

中间件解决方案

采用最先进的中间件来抽象和管理接口的复杂性，提供一致的通信层，从而简化集成工作。采用并实施标准化的通信协议，以简化接口配置，并确保跨不同组件的互操作性。

4.2 复杂校准

校准的耗时特性延长了开发周期，推迟了项目时间表，增加了成本，并限制了效率。理解校准参数的困难可能导致设置中的错误，这需要反复测试和重新校准周期。参数配置错误可能导致微控制器未能满足监管标准，需进行进一步修订，并可能导致昂贵的召回。此外，在复杂项目的情况下，传统的软件文档（有时甚至仍然是手动完成的）达到了其极限，并且也非常容易出错。

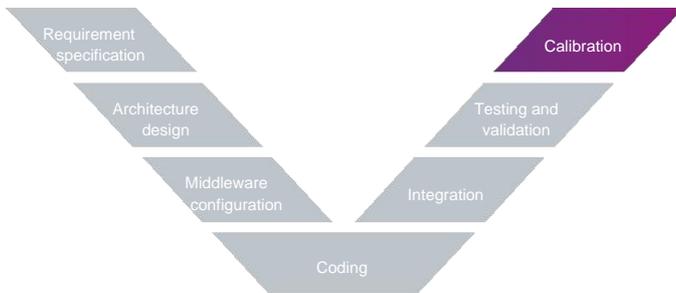


图 5: V 模型过程步骤中的“标定”



解决挑战的方法

增强文档和知识共享

建立一个全面的文件，清楚地解释每个校准参数，其用途、与其他参数的相互依赖关系，及其对系统的影响。实现自动化文档解决方案以减少错误并提高流程效率。

用户友好界面

通过使用只向用户提供相关参数和选项的解决方案，简化校准过程。交互式元素，如可视化滑块和动态图表使理解和调整校准设置变得更容易。

自动化和标准化

使用可以处理常规校准任务的自动化工具。它们减少了手工工作并简化了校准过程，同时还确保了不同项目和团队之间的一致性和可靠性。

仿真与建模

利用仿真和建模工具创建虚拟测试环境，在不对实际硬件产生负面影响的情况下测试校准更改。不同校准设置的影响可以可视化，以便更好地做出决策。此外，基于模型的设计方法将校准参数嵌入系统模型中，使操作和测试更容易。

4.3 测试调试耗时

软件越复杂，就越需要进行测试——测试正在成为创新的瓶颈。因此，效率与可靠性和安全性相结合是未来测试流程的核心要素。此外，不同模块和系统之间的交互可能导致复杂的测试场景，这些场景难以管理和执行，并且在测试开始之前会导致长时间的集成过程。更重要的是，深度嵌入的微控制器通常具有有限的处理能力、内存和存储，这可能会限制可以直接在硬件上运行的测试类型和范围。因此，传统的调试和测试工具可能不适合上述情况。

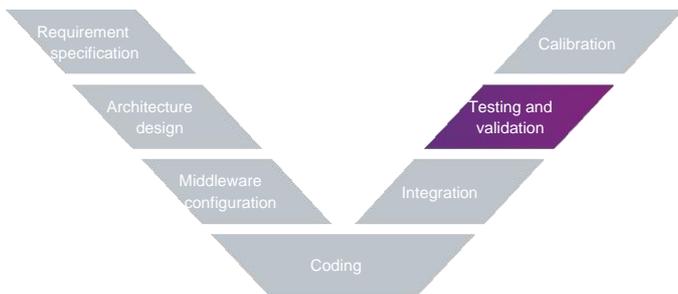


图 6：V 模型过程步骤中的“测试和验证”



解决挑战的方法

基于模型的开发

使用仿真工具在虚拟环境中对系统进行建模和测试。这减少了基于硬件的测试时间，因为可以立即检测到编程错误并向程序员提供反馈。

高级调试工具

使用电路仿真器（ICE）进行实时调试，并使用跟踪和分析工具来分析执行流、性能瓶颈和内存使用情况，从而帮助进行有效的调试和优化。

Software-in-the-Loop 测试

在虚拟环境中执行测试（前加载），这使得在早期阶段检测错误成为可能。并行化和延时也加速了虚拟环境中的测试。

持续集成/持续部署

实现 CI/CD 管道来自动化构建、测试和部署过程，减少手工工作并加速迭代循环。自动化测试可以及早发现问题，减少开发过程中修复错误的时间和成本。

单元测试和测试驱动开发

采用 TDD 实践，以确保在编码之前完成测试，并实现高测试覆盖率。在集成之前为组件开发和运行单元测试，以简化问题隔离和解决。

代码生成

从模型自动生成代码，确保一致性并减少人为错误的可能性。

自动化框架

使用脚本自动化复杂的测试场景，以确保一致性并减少手工工作。使用代码覆盖工具来覆盖所有的执行路径和边缘情况。

并行和连续测试

无论何时执行更改，都要持续测试系统。使用多个测试平台或虚拟环境并行运行测试，以减少测试时间。优化硬件资源分配，实现高效利用，减少瓶颈。

4.4 可拓展性和灵活性方面的限制

僵化的单体软件设计限制了修改和扩展，增加了更新和添加新功能的复杂性。专有系统和缺乏标准化导致供应商锁定，并使与新功能或第三方组件的集成变得复杂且容易出错。此外，软件设计中模块化不足阻碍了代码的高效重用，并妨碍了新功能的无缝集成，影响了整体系统的适应性。

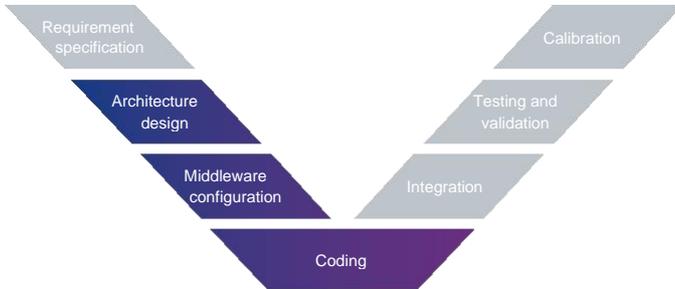


图 7: V 模型过程步骤中的“体系结构设计”、“中间件配置”和“编码”



解决挑战的方法

模块化设计和标准化接口

模块化设计和标准化接口 实施模块化、基于组件的架构和标准化的通信协议，以打破单一结构，便于更新和扩展。

开放标准和互操作性

在工具链中实施开放标准和设计系统时要考虑到互操作性，以避免供应商锁定，并简化新功能或第三方组件的集成。避免不遵守通用标准的供应商，从而引发供应商锁定。

中间件和抽象层

利用中间件和抽象层将应用程序逻辑相互分离，并将其与硬件细节分离，从而确保更容易更新和改进可伸缩性。

模块化和代码重用

设计具有高度模块化和可重用组件的软件，以方便新功能的无缝集成，提高整体系统的适应性。

4.5 整体网络安全需求

网络威胁的多样性和不断发展的性质，包括恶意软件、黑客攻击和未经授权的访问，使保护 ECU 系统的任务复杂化。不幸的是，微控制器的计算资源有限，因此很难在不影响性能的情况下实现全面的安全措施。此外，遵守不同地区和行业的严格和不同的网络安全法规和标准增加了开发过程的复杂性。



图 8：V 模型中所有流程步骤作为一个整体挑战

解决挑战的方法

多层安全架构

使用多层安全方法（包括加密、身份验证、入侵检测和引导机制）全面解决各种威胁情况。

资源受限环境的安全性

使用针对微控制器有限计算资源量身定制的轻量级，高效的安全协议，以确保在不影响性能的情况下提供强大的保护。

与监管标准保持一致

确保持续监控并适应不断发展的网络安全法规和标准；将合规要求纳入开发生命周期，并定期进行审计以确保遵守法规。

5. 结论

尽管微控制器是 E/E 架构中最小的组件，但它们在任何设置中都是至关重要的——从功能特定的控制单元到与领域无关的区域 ECU。投资于微控制器软件开发的改进将在未来几十年内保持汽车市场竞争力的重要因素。目标是提高速度、降低成本并保证安全性，同时考虑到遗留问题和日益复杂的车辆。这只能通过重新考虑既定流程来实现，包括逐步解决本白皮书中提出的挑战。像 ETAS 这样经验丰富的合作伙伴提供的复杂解决方案，允许 OEM 和供应商优化其深度嵌入的软件开发，使其能够应对未来的（市场）需求和（法律）法规。这种强大的基础创造了全球汽车行业的长期竞争力。



关于 ETAS

ETAS 有限公司成立于 1994 年，是罗伯特博世的全资子公司，在欧洲，北美，南美和亚洲的所有主要汽车市场都有本地业务。

ETAS 在软件开发解决方案、车辆操作系统、车辆云服务、数据采集和处理解决方案、集成客户解决方案和网络安全等领域为实现软件定义车辆提供全面的解决方案。

作为网络安全领域的行业先驱，我们通过成熟的软件产品和专业的安全服务组合，帮助客户管理网络安全相关的复杂性，降低网络风险，并最大限度地发挥其业务潜力。

ETAS 汽车安全解决方案正在保护全球数以百万计的车辆系统，并为软件定义车辆的网络安全制定标准。



关于我们的解决方案

作为一家拥有 30 多年经验的领先汽车供应商，ETAS 拥有广泛且成熟的解决方案组合。下面，我们根据您当前的需求和市场要求，专门为 ECU 软件开发过程编写了一系列软件包。

ETAS AUTOSAR 解决方案：

简单、安全、可靠的软件集成

我们为 OEM 和供应商提供软硬件无缝集成的解决方案包，以及针对网络威胁的全面保护。此外，我们的虚拟测试解决方案支持开发生命周期中的早期集成，从而减少了测试时间并提高了执行速度。

ETAS RTA-CAR (经典 AUTOSAR)

它是一个用于开发和配置的工具集

经典 AUTOSAR 软件组件，可实现 ECU 的高效集成、代码生成和验证

+

ESCRYPT CyscurHSM

是一款创新且灵活的高速同步安全固件，可确保 ECU 的安全启动、车载通信、ECU 组件保护和刷写

+

ESCRYPT CyscurLIB

它是一个专门针对嵌入式系统需求的加密库

+

虚拟 ECUs

在早期发育阶段进行复杂的测试

ETAS 虚拟化环境解决方案：

基于云的软件在环测试

我们为云中的 SIL 测试提供模块化的工具和服务。它们支持虚拟 ECU 的创建、调试和预校准，以及虚拟工件的集成和仿真，并在云中并行执行。工程师可以测试和迭代大量的测试用例，从而大大提高效率和模拟速度。此外，基于云的解决方案是可扩展的，并且支持软件的持续集成和部署，这对于现代开发过程尤为重要。

ETAS VECU-BUILDER

它是一种生成虚拟 ECU 的工具，用于在软件在环 (SIL) 设置中验证和验证汽车微控制器软件

+

COSYM (系统联合仿真)

它是一个强大的仿真和集成平台早期在系统级别测试和验证软件开发阶段

+

ETAS MODEL-SYMLATOR

它使用户能够在云中执行模拟

+

ESCRYPT CyscurFUZZ

它是一种最先进的模糊测试工具，有助于满足当前的法规和标准

ETAS 标定解决方案：

将数据转化为有价值的信息

我们的无缝工具耦合解决方案自动化数据搜索、分析和传输。通过这些经过验证的产品集成在一起，我们提高了 ECU 校准和诊断的效率、准确性和团队协作。

ETAS EHANDBOOK

为 ECU 软件的文档化提供了新的工具，在不同的抽象层次上以交互式 and 图形化的方式显示功能的逻辑

+

INCA (集成标定和应用工具) 包含灵活的工具，用于汽车电子系统的标定，诊断和验证

实时标定验证解决方案：

解锁数据洞察

我们的解决方案包有助于实时验证标定和快速反馈，允许快速调整和高品质的结果。这种集成的方法最大限度地减少了延迟，增强了团队协作，加快了开发周期，推动了 ECU 项目的成功，并确保了有效的项目完成。

ETAS EHANDBOOK

提供了一个即时访问详细的 ECU 文件

+

ETAS MDA (测量数据分析器)

简化数据分析和可视化，揭示趋势和问题的精确标定和优化 ECU 性能

+

ETAS EATB (ETAS 分析工具箱)

自动化和简化测试，为全面的见解和明智的决策提供强大的分析和报告工具

+

ETAS ASCMO (高级模拟标定，建模和优化)

允许 ECU 行为的详细模拟和先进标定更高的性能和效率



欢迎联系

Anthony Esteban

Customer Chief Engineer, ETAS

[Get in touch on LinkedIn Contact Form](#)



所提供的所有信息均为一般性信息，并非针对任何特定个人或实体的情况。虽然我们尽力提供准确和最新的信息，但不能保证这些信息与收到之日一样准确，也不能保证这些信息在将来仍然准确。在没有适当的专业意见和没有彻底审查有关情况的事实之前，任何人都不应根据这些信息采取行动。

©ETAS GmbH 版权所有。最后更新日期：2024 年 9 月

ETAS 有限责任公司

Borsigstraße 24, 70469 Stuttgart, Germany

T +49 711 3423-0, info@ETAS.com

您对 ETAS 产品

或解决方案感兴趣吗？

请访问 www.etas.cn

或者在社交媒体上关注我：

