白皮书

使用 Simulink 满足 AUTOSAR Classic 和 ISO 26262 标准的最佳做法

MathWorks®

汽车行业正在经历一场巨大的变革，这主要体现在所要开发的系统的复杂性及其部署速度方面。这些系统的许多功能都是针对自动驾驶和主动安全而设计的。

特别是，随着生产车辆的特征集功能呈指数级增长，软件设计的重要性也与日俱增。这反过来又增加了交付车载软件的规模、复杂性和总体难度。部署和上市都在以前所未有的速度推进。现在，大多数的 OEM 都在定期提供空口软件的持续升级。为了应对这种软件和电气系统的复杂情况，采用一种结构化的方法来管理和设计这些电气系统势在必行。因此，许多汽车 OEM 和供应商都已决定对 QM 组件和 ASIL 组件分别采用 ASPICE 标准和 ISO® 26262 标准。本白皮书后面部分将重点探讨 ISO 26262 合规性和最佳实践。ISO 26262 分为 12 个部分，其中 10 个部分是规范，2 个部分是指南。该标准旨在帮助组织打造一个统一的流程，用于开发高完整性软件和电气系统。规范部分提出的要求涵盖各种主题，例如安全风险高层次分析、系统和架构设计注意事项，以及实现级软硬件设计注意事项。

AUTOSAR 的目的是创造一种结构化的方法，用来管理电气系统的软件架构和结构。这种方法以其自身的特点，非常适用于自上而下的设计方法，可以帮助实现应用软件的 ISO 26262 合规性。本白皮书探讨了在使用基于模型的设计时可以帮助实现 ISO 26262 合规性的 AUTOSAR 最佳实践。

AUTOSAR 如何帮助实现 ISO 26262 合规性？

AUTOSAR 是基于功能安全开发的。这一点在 AUTOSAR 联盟发布的题为 AUTOSAR 功能安全措施概述的文档中得到了充分体现。该文档描述了适用于安全相关应用的关键概念和构造。本白皮书将通篇引用其中的许多方法。示例包括但不限于：

- 明确定义的架构构造
- 软件重用
- 算法单元封装
- Dem、NvM 和 FiM 等可重用服务

这些方法是许多 OEM 和供应商已决定使用 AUTOSAR 作为其 ECU 架构来帮助采用 ISO 26262 标准的原因所在。下文旨在重点介绍尝试遵循 ISO 26262 合规流程时的一些 AUTOSAR 最佳实践。

AUTOSAR 工具链和工作流注意事项

使用 Simulink 和 Stateflow 对故障检测和软件诊断功能建模

借助 Simulink® 和 Stateflow® 语言，用户能够构建各种不同类型的算法。这种模式同样也适用于在 Simulink 和 Stateflow 内建模的 AUTOSAR 算法。Simulink 和 Stateflow 通常用于开发 AUTOSAR 应用软件组件，但是，工具鉴定问题使得这两款工具往往被忽略，未能用于开发故障检测和诊断等 ASIL 较高的功能。为了解决这一问题，MathWorks 提供了预鉴定工件（帮助进行工具分类的工件）、鉴定工件以及参考工作流（用于针对所有 ASIL 开发算法）。因此，在开发故障检测、成熟、退化模式管理和诊断算法时，不需要放弃仿真环境的好处。为了简化与 AUTOSAR 架构的诊断集成，AUTOSAR Blockset 提供了可以实现 Diagnostic Event

© 2022 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See mathworks.com/trademarks for a list of additional trademarks.
Other product or brand names may be trademarks or registered trademarks of their respective holders.

Manager (Dem) 交互的标准模块。在下面的示例中，使用的是 DiagnosticMonitorCaller 模块。此模块将仿真并生成与符合 AUTOSAR 标准的 Diagnostic Event Manager 兼容的代码。

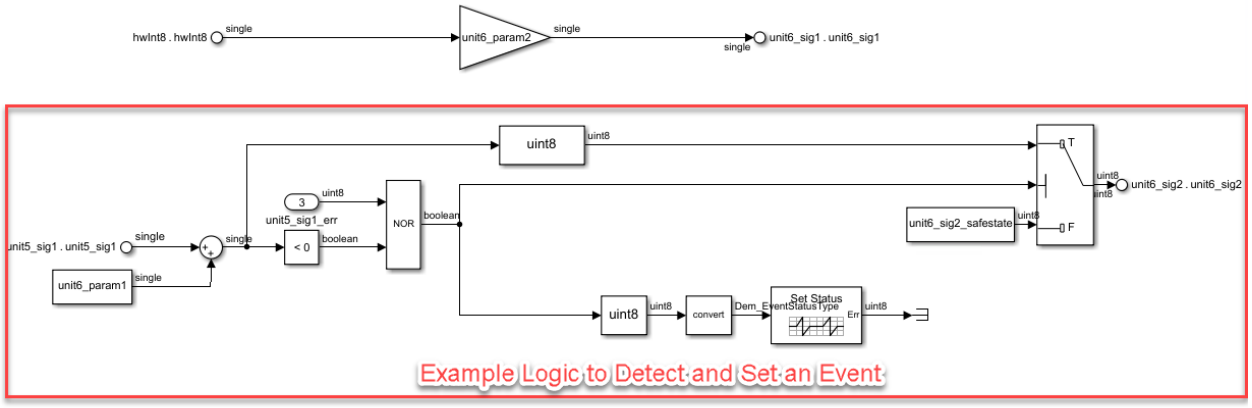

图 1. Simulink 故障检测算法示例。

使用标准的 AUTOSAR 服务（例如 NvM、FiM 和 Dem）

AUTOSAR 标准定义了大多数汽车 ECU 都需要的几种常用软件服务。对于此应用示例，相关服务包括 Diagnostic Event Manager (Dem)、Function Inhibition Manager (FiM) 和 Nonvolatile Memory Manager (NvM)。每个服务都可以在两个 AUTOSAR 应用之间互换使用。有利的做法是直接从 COTS 提供商那里购买这些服务，而不是创建自定义解决方案，因为该解决方案可能存在重用或兼容性问题，需要额外的验证工作。

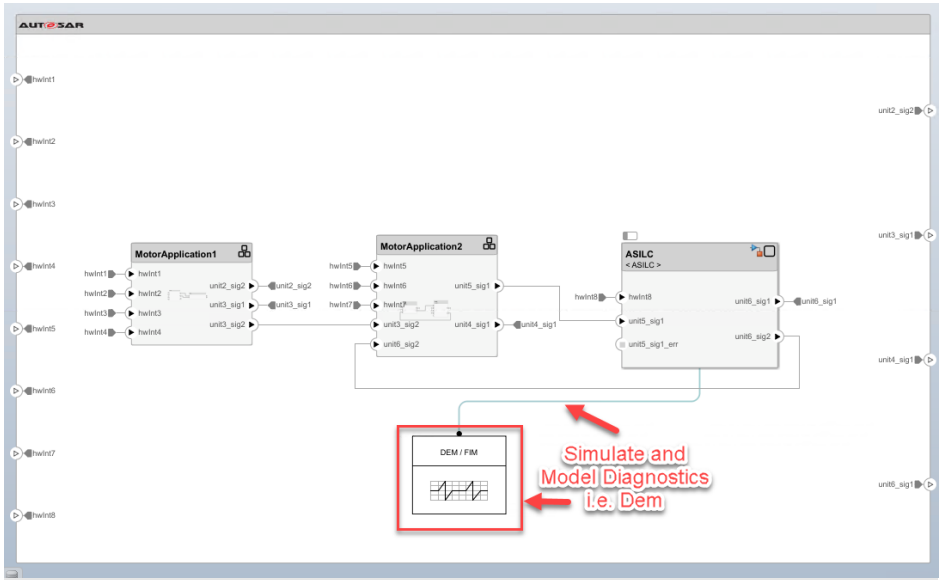

图 2. 带有 Dem RTE 模块的系统级模型示例。

© 2022 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.

Simulink 还简化了这些服务的使用。AUTOSAR Blockset 提供了多个模块，允许组件在 Embedded Coder® 所生成的代码中使用正确的 API 调用这些标准服务。AUTOSAR Blockset 还提供了带有这些服务的桩件的模块。在框架或 AUTOSAR 架构模型中使用时，这些服务提供商模块使工程师无需进行完整的服务实现，即可执行模型在环 (MIL) 或软件在环 (SIL) 测试。

利用经预鉴定/鉴定的 ISO 26262 工具

如果在创建符合 ISO 26262 标准的算法内容时使用了任何自动化工具，则这些工具可能需要经过鉴定。IEC Certification Kit 提供了放心使用基于模型的设计工具所需的工作产品，例如 TCL（工具置信度）、工具鉴定工件、基于模型的设计参考工作流等。下面以 Embedded Coder 参考工作流为例进行说明。功能安全经理可以利用这些工件显著减少他们的工具鉴定工作量。

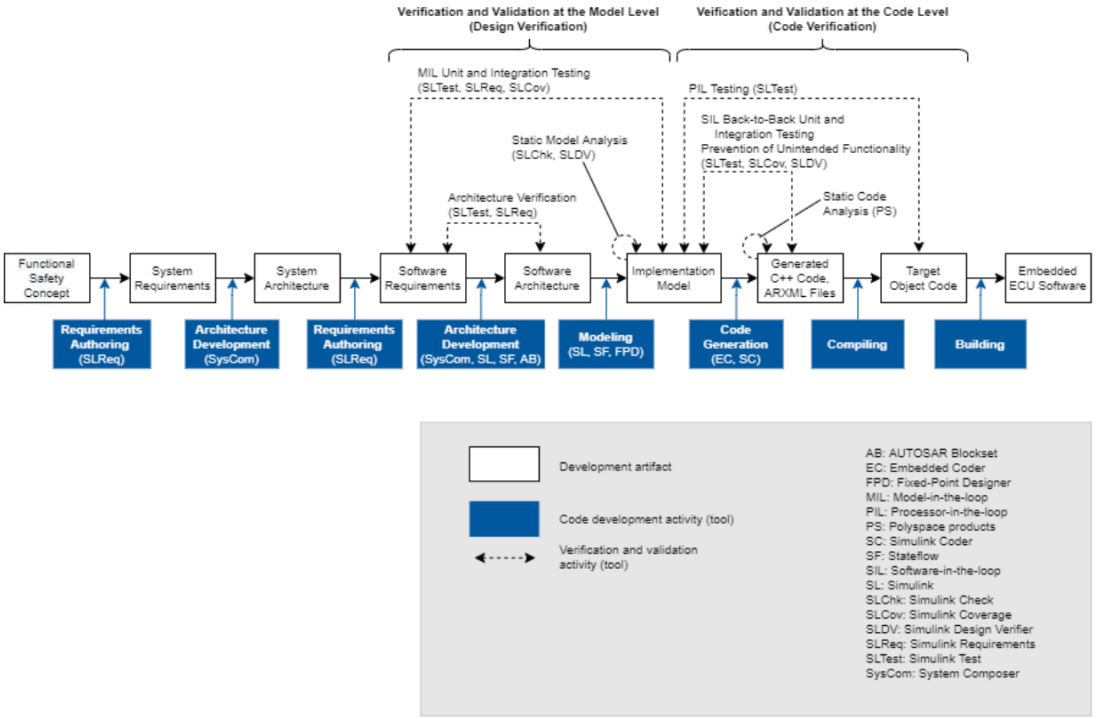

图 3. 符合 ISO 26262 标准的基于模型的设计参考工作流。

© 2022 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See mathworks.com/trademarks for a list of additional trademarks.
Other product or brand names may be trademarks or registered trademarks of their respective holders.

采用增量式测试策略

一项适用于 ISO 26262 的测试策略是，采用自下而上的增量式测试方法。这项测试策略始于单元级别，一直向上经过各种集成级别（如下图所示），直到整个软件的集成测试：

- 单元级 MIL 和 SIL 测试
- 单元级静态代码分析
- 功能级 MIL 测试
- 应用级 MIL 测试
- 软件集成级 SIL 测试

这一过程从单元级仿真的快速迭代开始。当开发团队从这些迭代中获得一定的信心后，他们的活动将从模型级转向代码级。在 Simulink 中，可以通过 Simulink Test™、Simulink Coverage™、Simulink Design Verifier™ 和 Requirements Toolbox™ 使用 MIL 和 SIL 测试进行代码级别的仿真。然后，使用 Embedded Coder 生成实现。此外，还可以进行背靠背 MIL/SIL 测试，以确保仿真与所生成代码的行为相符。在单元级测试完成后，第一次集成级测试便可进行了。

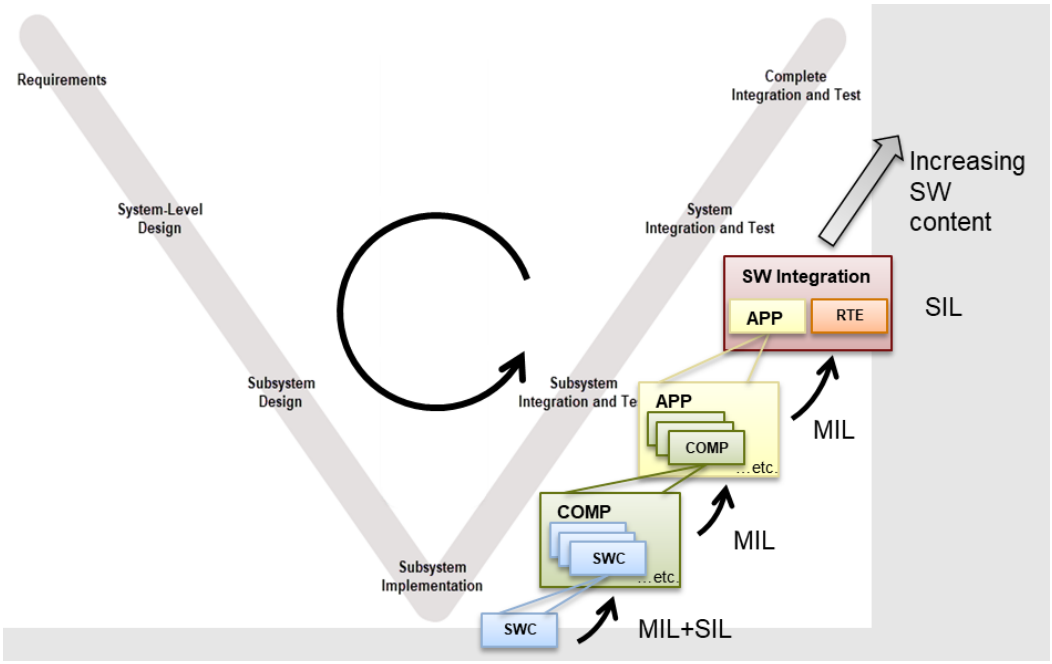

图 4. AUTOSAR 增量式单元和集成测试策略。

ISO 26262 第 6 部分第 10 条规定了集成测试的方法。其中包括多项支持技术，例如基于需求的测试、接口测试、故障注入测试等。这些技术需要应用于将部署到生产环境中的软件。在 AUTOSAR 架构中，自下而上分别为应用软件、基础软件和自动生成的集成代码（RTE 层）。下图说明了这种软件分区，其中带有 MATLAB® 图标的部分表示由 Embedded Coder 生成的部分。

© 2022 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See mathworks.com/trademarks for a list of additional trademarks.
Other product or brand names may be trademarks or registered trademarks of their respective holders.

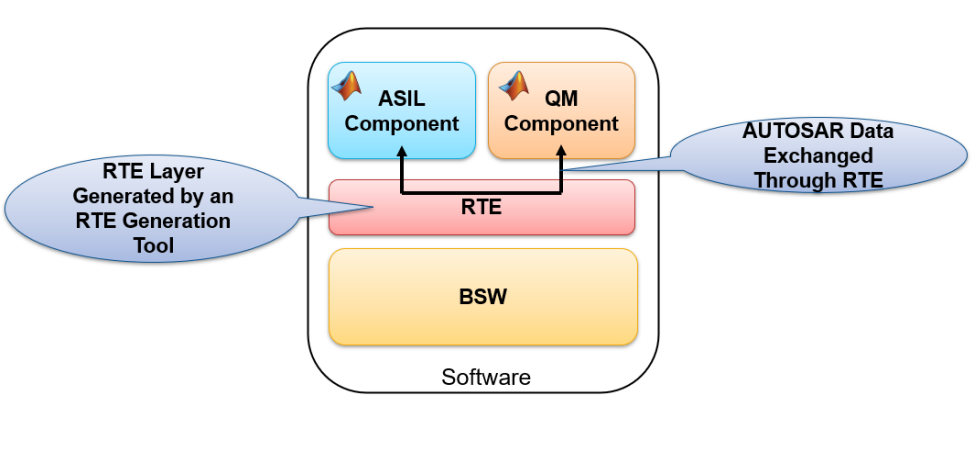

图 5. AUTOSAR 软件架构。

尽管需要在实际目标上进行正式集成测试，但强烈建议在设计周期的早期阶段利用仿真进行集成检查。在 Simulink 中，可以轻松地创建 AUTOSAR 组合和测试框架来执行不同级别的集成。这些集成检查可以增加您在目标上进行最终的集成级测试时可能已捕获所有数据交换不匹配项的可能性。这一步是为了减少在软件部署到目标上后将捕获的不匹配项数。图 6 显示了用于仿真测试的集成级框架的示例。

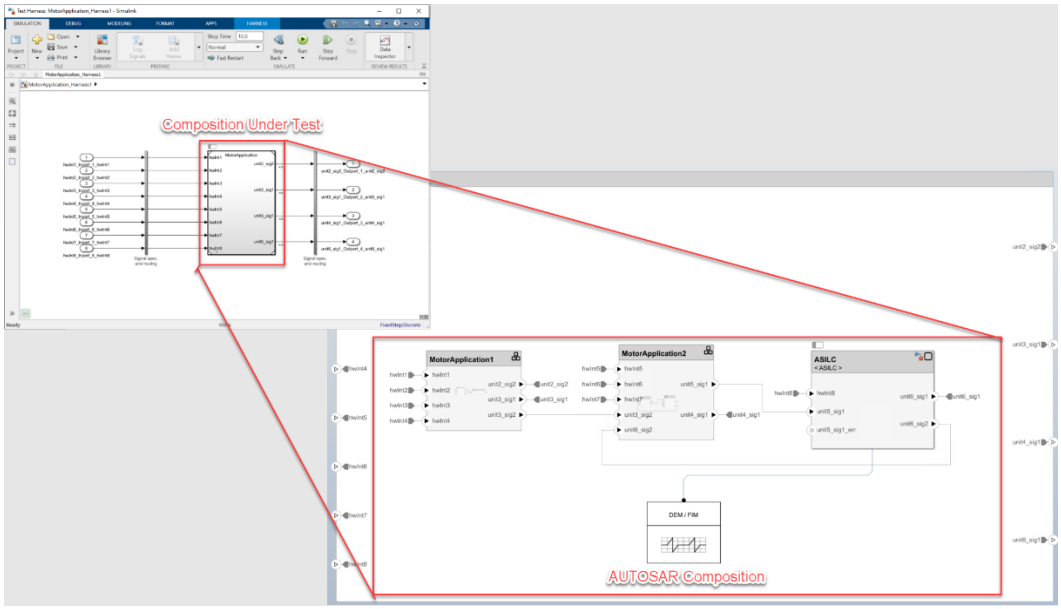

图 6. Simulink 中的集成仿真。

© 2022 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See mathworks.com/trademarks for a list of additional trademarks.
Other product or brand names may be trademarks or registered trademarks of their respective holders.

基于这些因素，正式集成测试策略包括以下内容：

1. 对每个 AUTOSAR 软件组件都进行单元级 MIL 和 SIL 测试
2. 在 Simulink 中使用迭代仿真对 AUTOSAR 组合进行初始集成测试
3. 实现 RTE 生成和整个软件的版本集成*
4. 根据 ISO 26262 要求进行软件集成测试*
5. 使用 Polyspace 进行静态分析

* 在最终目标硬件上完成的活动

AUTOSAR 架构构造

使用软件组件作为单元边界

ISO 26262 第 6 部分深入探讨了针对每个 ASIL 创建符合 ISO 26262 标准的软件开发流程的原则。具体来说，其中着重探讨了单元级和集成级软件活动。另一方面，AUTOSAR 中包含软件组合、原子组件和可运行实体。因此，必须按照 ISO 26262 的语言来使用 AUTOSAR 架构构造。各个入口 C 函数映射到封装在原子软件组件中的可运行实体。在 Simulink 中，组件映射到 Simulink 模型，而可运行实体映射到该模型的不同速率、函数调用子系统或 Simulink 函数模块。相关软件组件会组成表示各个软件功能的组合，这些功能又会进一步组成表示子系统和系统的组合。AUTOSAR 组合是完全虚拟的，因为这种组合实际上并没有相应的代码表示。

那么，以下是选择将原子软件组件定义为 ISO 26262 软件单元的几点有力支持：

- 原子软件组件元素和接口在模型级别进行映射。
- Simulink 模型边界非常适用于创建测试框架和测试用例。
- 模型中的图形化内容将生成到一组封装的 .c 和 .h 文件中。
- 组件中的可运行实体应该高度耦合，并且最好作为一个单元进行测试。

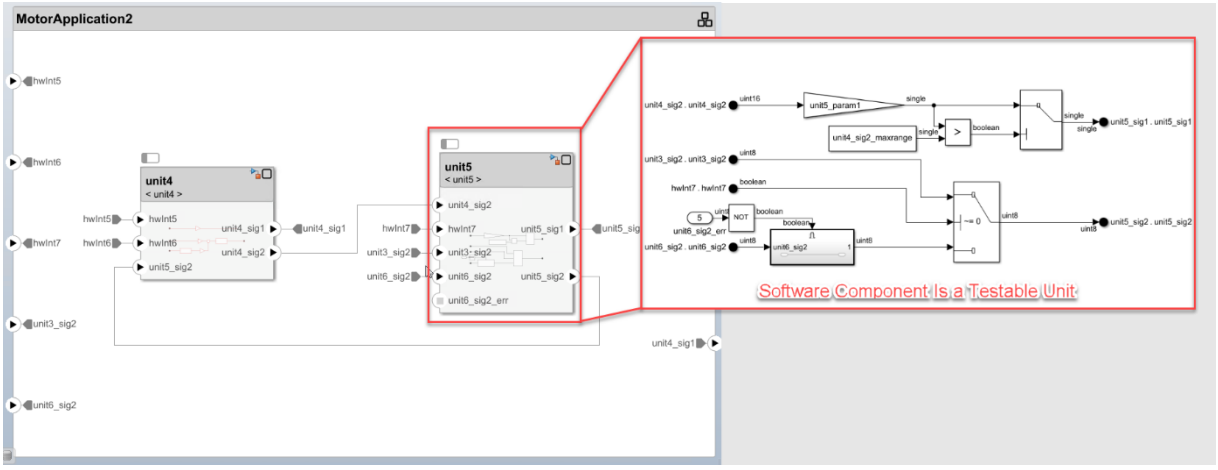

图 7. 作为单元边界的 AUTOSAR 软件组件的示例。

© 2022 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See mathworks.com/trademarks for a list of additional trademarks.
Other product or brand names may be trademarks or registered trademarks of their respective holders.

将 ASIL 组件划分为多个分组组合

ISO 26262 允许 ECU 应用层包含不同 ASIL 的单元，但前提是可以证明各单元之间互不干扰。如果一个应用需要具有多个 ASIL，则建议将这些软件组件划分为不同的分组组合。例如，可以将一个组合专用于 ASIL D 级组件，而将另一个组合专用于 ASIL B 级组件。之所以用这种方式分割算法，原因是为了清楚地了解哪些组件属于哪个 ASIL。这样做也能使验证和确认过程对于每个 ASIL 来说都是一致的。集成测试可以按照每个 ASIL 所需的严格程度来进行。

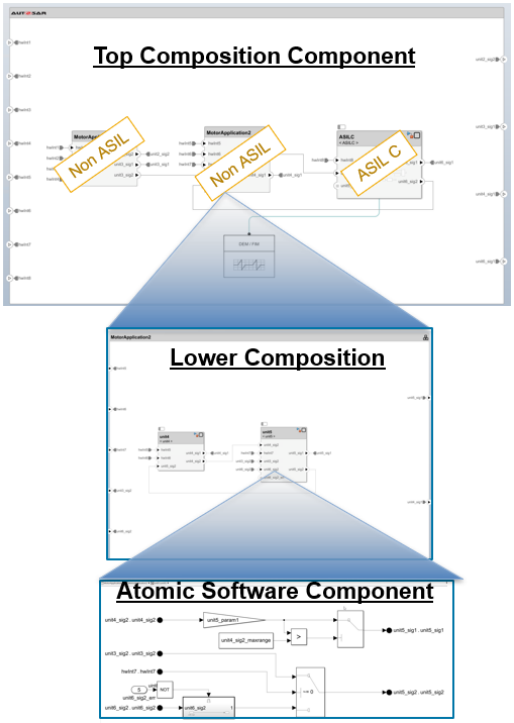

图 8. 划分了 ASIL 组合的软件架构。

将 AUTOSAR 组件划分到内存段的软件地址方法

ISO 26262 举例说明了用于实现无干扰的安全机制和措施。安全机制将以下方面的典型故障考虑在内：

- 时序和执行
- 内存
- 信息交换

© 2022 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.

要缓解内存和信息交换故障，一种方法是针对每个 ASIL 使用不同的内存段。这可以运用 AUTOSAR 的软件地址方法概念轻松实现。软件地址方法可作为基于实现特定的内存段的抽象层。架构师可以利用不同的软件地址方法标记各种 AUTOSAR 元素，例如可运行实体、端口和参数。然后，架构师可以将这些软件地址方法用于微控制器上的不同内存段。软件地址方法也可以在 Simulink 模型中轻松地选择和查看。我们建议针对每个 ASIL 定义一组不同的软件地址方法，以实现最大的内存分区粒度。

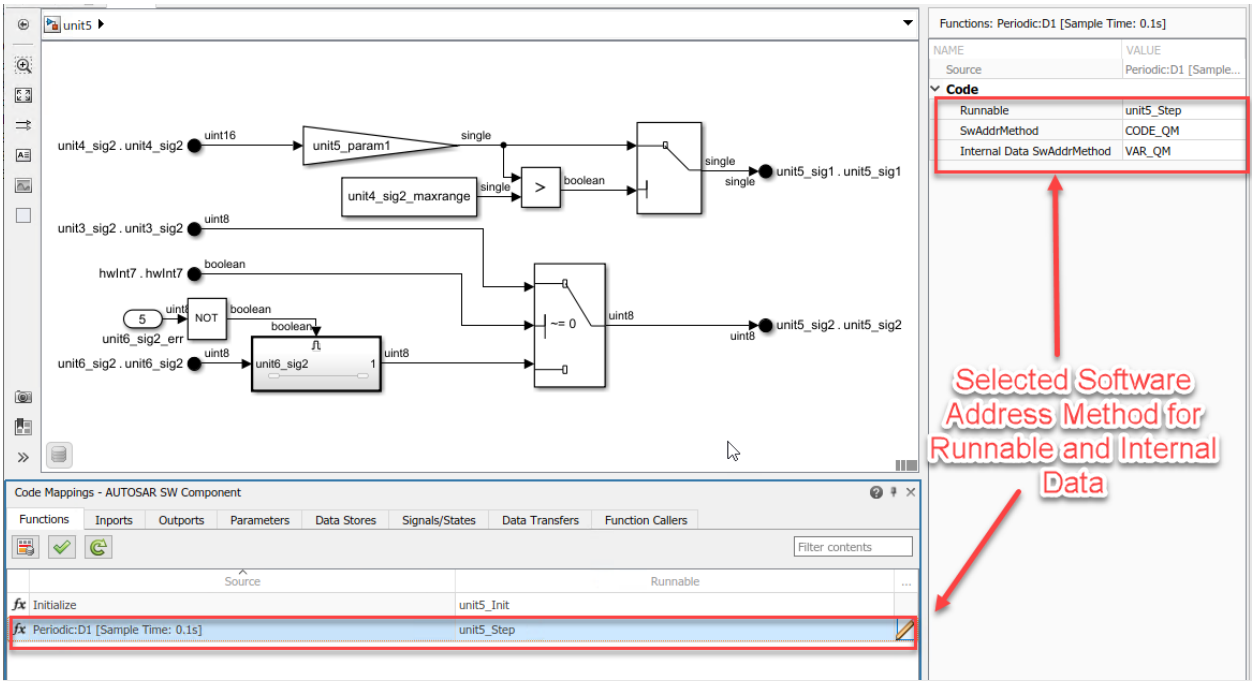

图 9. 对可运行实体和内部数据使用软件地址方法。

从非 AUTOSAR 转换为 AUTOSAR 时重新构建现有代码库

当组织决定采用 AUTOSAR 架构时，则通常需要从以前的非 AUTOSAR 代码库进行迁移。对许多组织来说，只需沿用现有算法并将其包装在 AUTOSAR 构造中的做法很有吸引力。例如，使用复杂的设备驱动程序来包装大量现有的遗留 C 代码。但是，当切换到 AUTOSAR 架构时，这并不是一种可利用的有效方法。更为有效的做法是，以此为契机确保组织的架构经过精心构造，将能实现组织的功能安全目标。因此，务必对软件架构进行评估，并调整组织的架构，同时考虑可用的 AUTOSAR 构造，以及有助于组织实现功能安全目标的构造。但是，对于需要使用现有软件架构支持一次性 AUTOSAR 程序的供应商来说，采用复杂的设备驱动程序方法可能要比投入必要的开发进行重新构建更加可行。

© 2022 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See mathworks.com/trademarks for a list of additional trademarks.
Other product or brand names may be trademarks or registered trademarks of their respective holders.

定义数据管理策略

虽然可行的数据管理策略有许多，但使用 AUTOSAR Blockset 实现 AUTOSAR 构造有助于缩小围绕这个主题的选择范围。作用域限于组件级别的数据应存储在模型工作区中。这样做便于执行代码映射工作流，并防止无意中共享标定和测量定义。模型工作区可以配置为引用外部 M 文件或 MAT 文件作为数据源，以独立于模型文件管理值集。每个组件都应将所引用的数据和数据类型等全局对象（例如 Alias、Bus）存储在特定于组件的数据字典中。根据需要，可以对多个组件或团队之间共享和管理的类型进行分组，并从公共数据字典中对其加以引用。在集成时，组合模型有助于加强组件之间的一致性，因为模型层次结构中数据字典之间的任何重复定义都必须相同。或者，也可以通过在集成时将所有数据字典合并成一个数据字典（在验证条目时）来实现这种一致性。

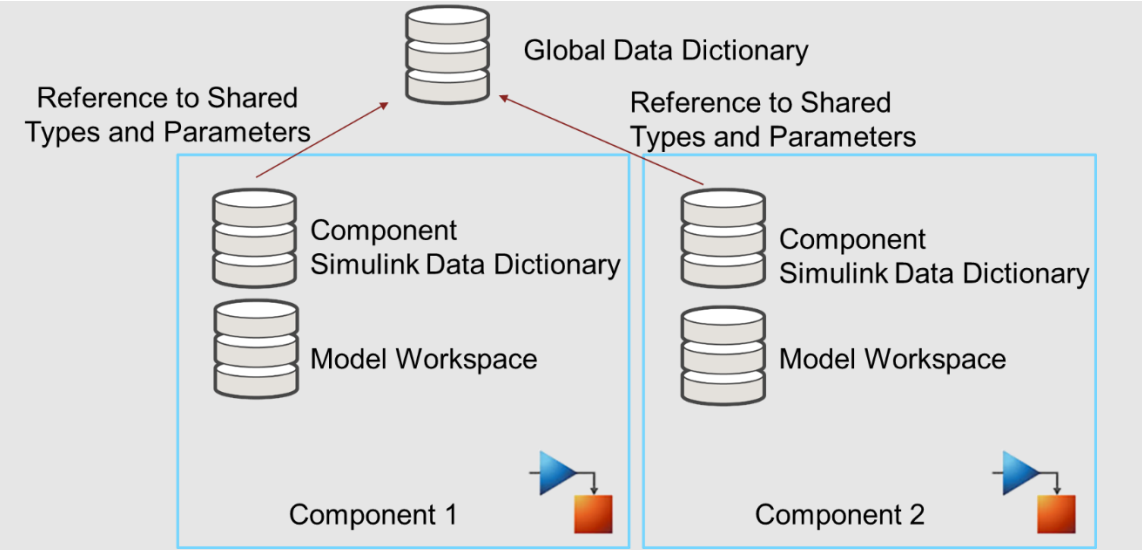

图 10. Simulink 中的 AUTOSAR 数据层次结构示例。

AUTOSAR 数据传输机制

使用 RTE 传递安全关键数据

AUTOSAR 是一个特意分层的软件架构。底层是基础软件 (BSW)。BSW 的上一层是运行时环境 (RTE)。顶层是应用层 (ASW)。所有的 AUTOSAR 端口数据都通过 RTE 传递，并由 RTE 进行管理。这使得 RTE 可以增加额外的保护，例如管理数据的特定内存段以及用于检测传输错误的错误状态。高完整性和安全关键数据应通过 RTE 传递，以便于它们能够受到这种额外的保护。此外，这些组件之间的接口可以在虚拟功能总线级别使用编写工具加以明确定义和管理，从而避开了 ECU 内和 ECU 间的数据传输机制。

© 2022 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See mathworks.com/trademarks for a list of additional trademarks.
Other product or brand names may be trademarks or registered trademarks of their respective holders.

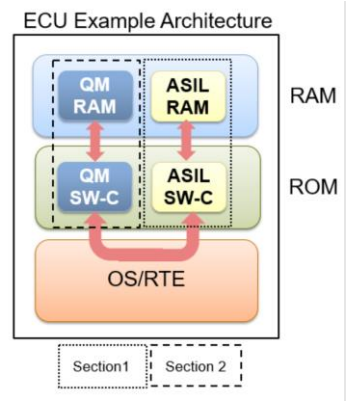

图 11. 软件组件通过 RTE 传输数据。

对高完整性信号使用隐式数据传输

AUTOSAR 提供了多种数据访问模式，用于定义可运行实体与软件和电气系统的组件之间的数据传输。对于 AUTOSAR 端口和可运行实体间变量中的数据元素，两种主要访问类型是显式访问和隐式访问。每种数据访问模式都有各自的优点。隐式传输可确保使用一段数据的每个可运行实体都将访问数据的独立副本，而不会受到该数据的所有其他读取器和写入器的影响。RTE 会为数据元素维护一个数据源，该数据源会在可运行实体开始执行前写入可运行实体副本中，并在可运行实体完成执行后从可运行实体副本中进行更新。因此，如果可运行实体发生中断，则在执行恢复后，可运行实体仍然拥有数据的上一副本。另外，在中断的可运行实体完成执行之前，其他可运行实体将看不到由中断的可运行实体计算的任何中间值。这可以确保数据在一个时间步中不会发生不确定的更改。另一方面，显式数据传输将始终向可运行实体提供由提供者在请求时已为数据元素设置的当前值。

对于有高完整性要求的信号，建议对这些数据元素使用隐式数据传输。这样可确保使用者在执行计算时有一个非易失性数据副本可供操作。在 Simulink 中，可通过将传输模式从显式更改为隐式在单个数据元素级别进行这样的配置。

© 2022 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See mathworks.com/trademarks for a list of additional trademarks.
Other product or brand names may be trademarks or registered trademarks of their respective holders.

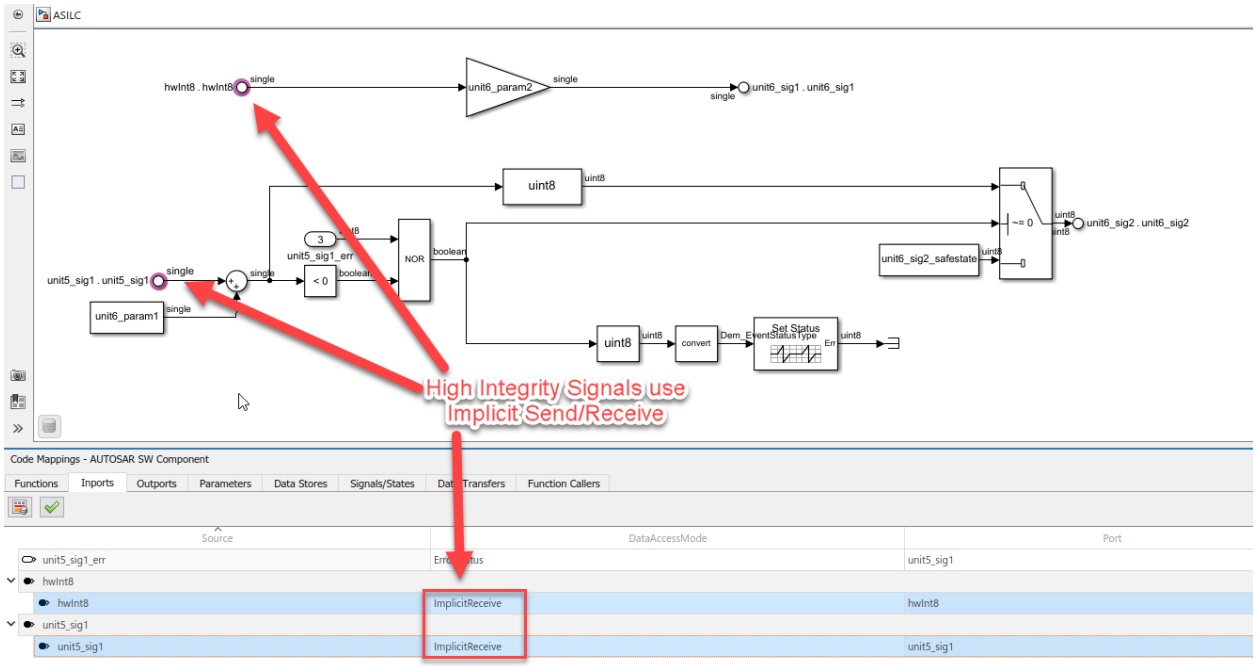

图 12. 在组件级别配置隐式发送/接收。

在不同 ASIL 组件之间传输数据时使用端口错误状态

AUTOSAR 可以针对每个数据元素为任何可运行实体提供错误状态。AUTOSAR 为发送方-接收方接口定义了服务质量属性，例如 `IsUpdated` 和 `ErrorStatus`。`IsUpdated` 属性允许 AUTOSAR 显式接收方检测接收方端口数据元素自上次读取操作执行以来是否接收过数据。当数据空闲时，接收方可以节省计算资源。`ErrorStatus` 值用于捕获数据完整性中的各种故障模式。该值可以通知使用数据的应用软件，数据是否已收到以及是否可以信任。对于任何有高完整性要求的数据信号，建议在应用软件组件中使用错误状态端口，例如在具有不同 ASIL 的软件的各部分之间交换数据时使用。

此错误状态可以在 Simulink 中针对每个数据元素进行配置，方法是构造一个新端口并将其映射到数据元素端口。这样一来，算法便可具有两个 Simulink 端口。一个端口用于数据值，而另一个用于错误状态。

© 2022 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See mathworks.com/trademarks for a list of additional trademarks.
Other product or brand names may be trademarks or registered trademarks of their respective holders.

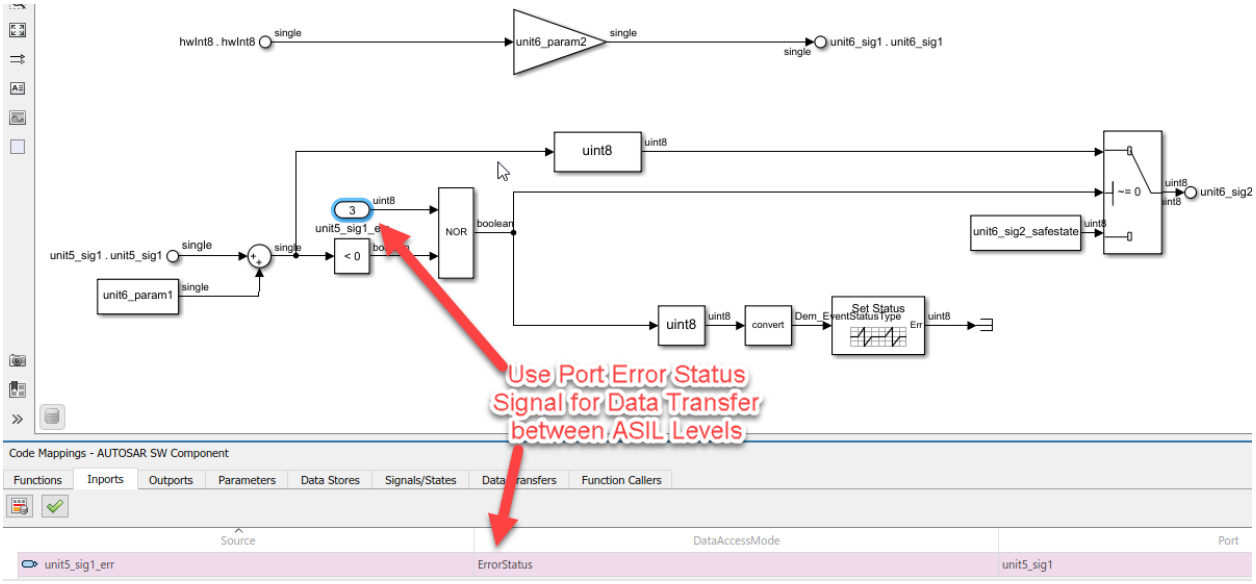

图 13. 配置端口错误状态。

使用端到端保护

端到端保护是由 RTE 实现提供的一种保护，旨在检测通过 AUTOSAR 端口在软件组件之间进行数据传输时的错误。在使用端到端保护时，返回的 `ErrorStatus` 将有所不同，以反映可通过端到端保护捕获的潜在故障模式。软件应用应该检查这些值，并确定如何对这些类型的故障做出响应。通过这些端口的数据传输，既可以通过 CAN 通信等媒介进行，也可以通过同一 ECU 上两个组件之间的 RTE 调用进行。对于任何 AUTOSAR 端口，都可以实施端到端保护。我们建议对通过 CAN、LIN 或以太网等通信通道传输的任何高完整性 AUTOSAR 数据使用端到端保护。

端到端保护可以在 AUTOSAR 架构工具中定义。此设置也可以在 Simulink 模型中管理和查看。图 14 说明了如何在 Simulink 中为数据元素设置端到端保护。

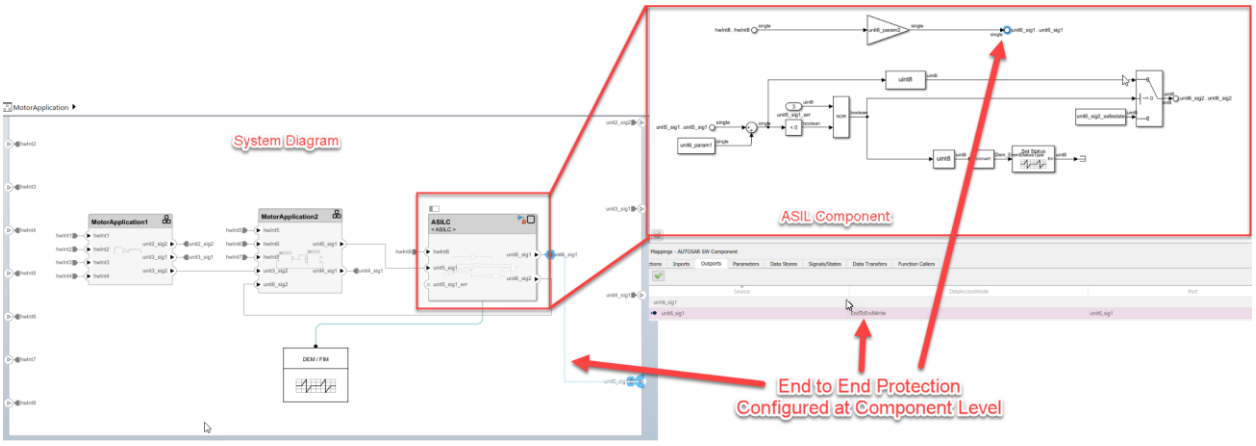

图 14. 在组件级别配置端到端保护。

© 2022 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.

小结

本白皮书旨在研究 ISO 26262 和 AUTOSAR 的最佳实践。开发 AUTOSAR 是为了对 ISO 26262 进行补充，并提供便于实现 ISO 26262 合规性的构造。本白皮书中的最佳实践是指 AUTOSAR 构造和工作流项目，它们可在 Simulink 中用来帮助将 ISO 26262 标准应用于以下方面：

- AUTOSAR 工具链和工作流
- AUTOSAR 架构构造
- AUTOSAR 数据传输机制

本白皮书中探讨的最佳实践也是对使用 Simulink 部署 AUTOSAR 的 10 个最佳实践这一白皮书的有力补充。在确定贵组织的 ISO 26262 工作流时，应该先评估并使用这些最佳实践。此外，开发组织应该明确哪些 AUTOSAR 构造应该成为内部的建议标准。MathWorks 咨询服务已经帮助众多客户建立了符合 ISO 26262 标准的工作流，并在基于模型的设计的合格用例基础上针对他们的 AUTOSAR 架构制定了相应的策略。

后续步骤

面向 AUTOSAR 的 MATLAB 和 Simulink

mathworks.com/solutions/automotive/standards/autosar

联系 MathWorks 咨询服务

mathworks.com/services/consulting.html

© 2022 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See mathworks.com/trademarks for a list of additional trademarks.
Other product or brand names may be trademarks or registered trademarks of their respective holders.