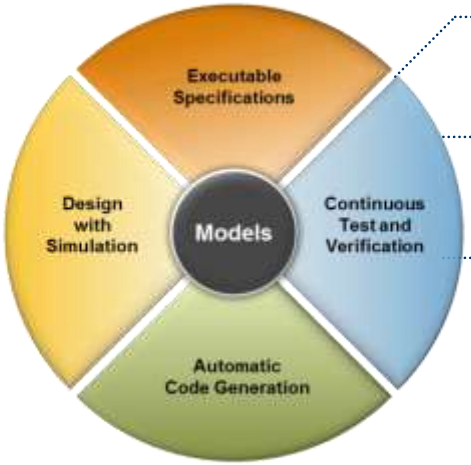
基于模型设计助力高安全关键系统软件开发
——符合行业标准的MBD方案及案例解析

于化龙
高级项目开发部 经理
迈斯沃克（软件）北京有限公司

内容概要

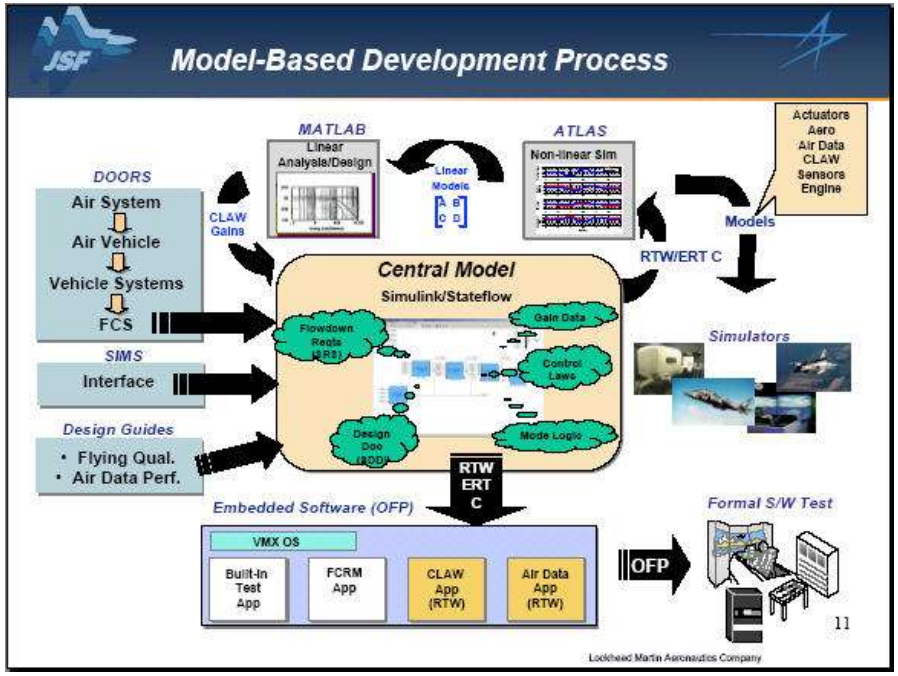


- 国内外高安全关键系统开发与验证的趋势
- 符合行业标准的基于模型设计解决方案
- 轨道交通行业符合EN50128标准项目实践

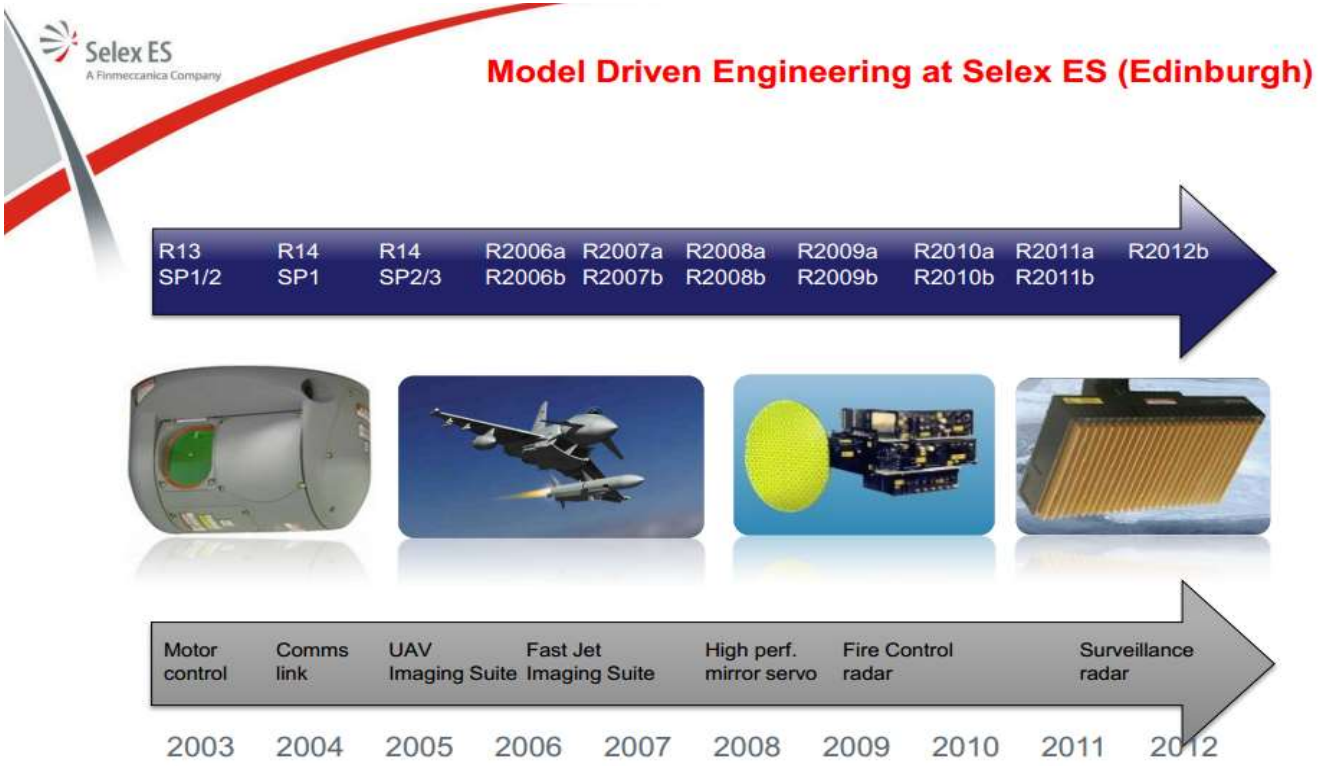
高安全关键系统软件开发面临的挑战

- 系统复杂度不断增大
- 开发周期不断缩短
- 代码量急速增长
- 软件质量与认证要求
- 更加严苛的行业标准
- 更高的安全性考虑
- ……

基于模型设计已成为高安全关键系统软件开发的趋势

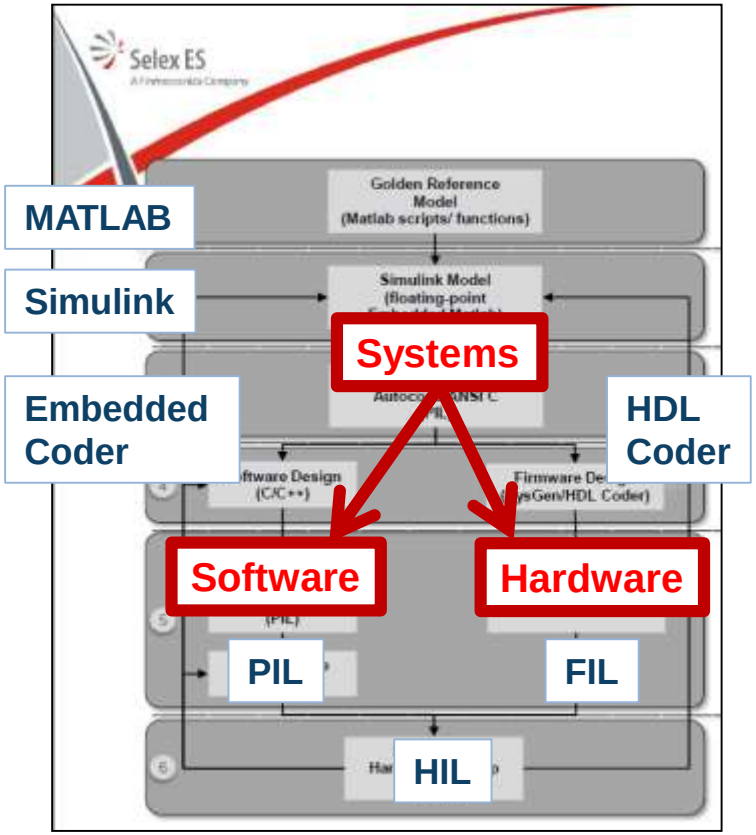


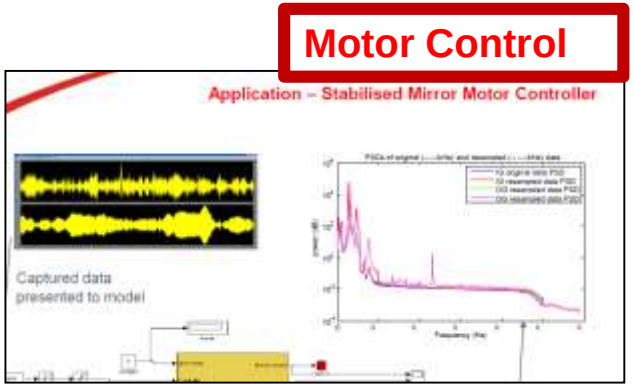
意大利Selex公司基于模型设计技术历程

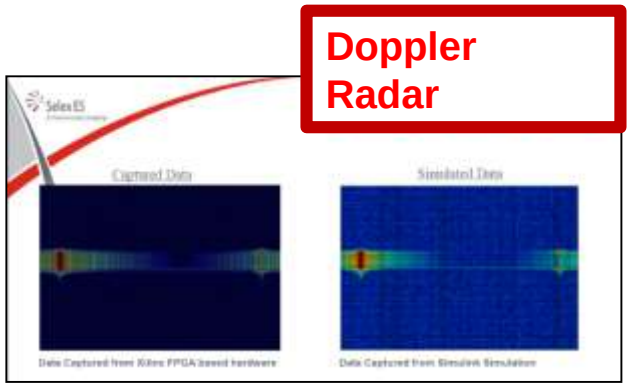


http://www.mathworks.com/tagteam/75977_Concepts_to_Chips_SELEX_ES.pdf

意大利Selex公司飞行系统 （March 2013）







http://www.mathworks.com/tagteam/75977_Concepts_to_Chips_SELEX_ES.pdf

Alstom Generates Production Code for Safety-Critical Power Converter Control Systems (2004)

Challenge

Design and implement real-time power conversion and control systems for trams, metros, and railways

Solution

Use MathWorks tools for Model-Based Design to design, simulate, and automatically generate production code for safety-critical transportation systems

Results

- Development time cut by 50%
- **Defect-free, safety-critical code generated and certified**
- Common language established

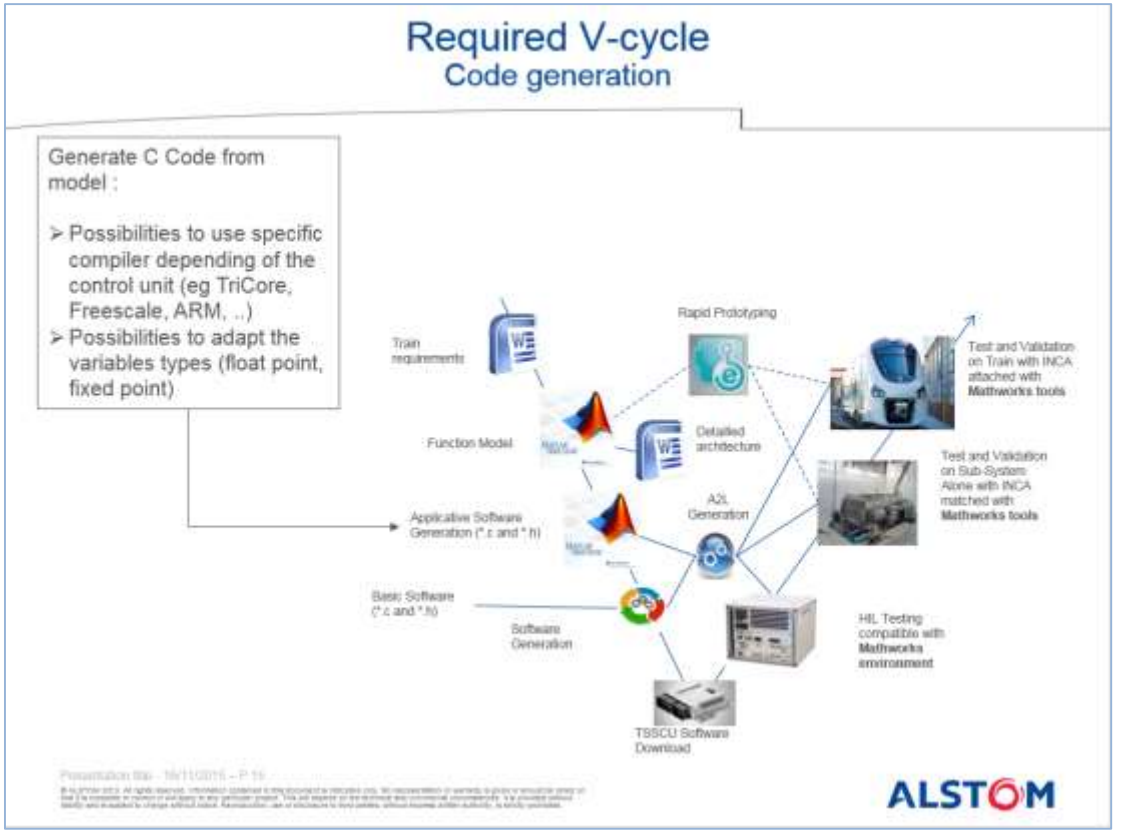
Link to user story

Pendolino tilting train.

"When Alstom delivered a Pendolino train to Czech Railways, the railway application was the first with automatically generated code to receive TUV certification (for EN 50128)"

Han Geerligs
Alstom

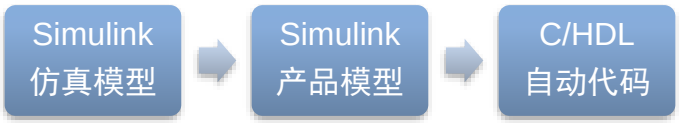
Alstom （Nov. 2013）



基于模型设计的关键在于从仿真到实现

模型：只是用做仿真

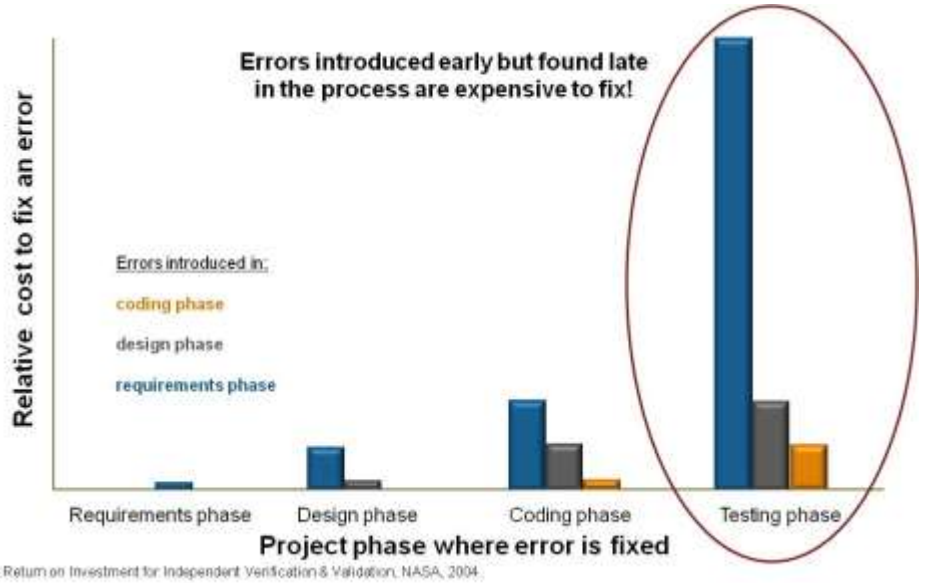
- 设计部门仍然通过手写设计文档来交给软件部门
- 设计部门可能需要维护模型/代码/文档三套设计
- 软件部门仍然根据文档进行手工编码
- 代码编写存在需求理解偏差，易引入认为缺陷



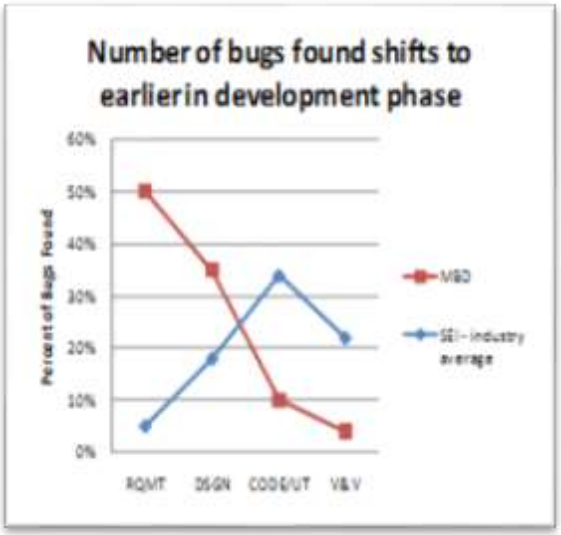
模型：自动生成代码

- ✓ 仿真模型经过配置后可以直接生成代码
- ✓ 设计部门可将仿真模型或者配置好的产品模型交付
- ✓ 软件部门可以对模型进行配置后自动生存代码
- ✓ 通过定制可以实现自动代码与手写代码的自动集成

基于模型设计使早期验证成为可能

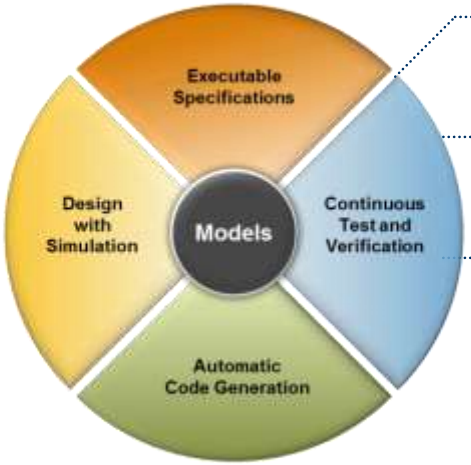


Source: Return on Investment for Independent Verification & Validation, NASA, 2004



基于模型设计：以模型为核心集成设计开发环境

内容概要



- 国内外高安全关键系统开发与验证的趋势
- 符合行业标准的基于模型设计解决方案
- 轨道交通行业符合EN50128标准项目实践

高安全关键系统相关行业标准

DO-178 (Level A)

Honeywell Aerospace USA
Flight Control Systems

ISO 26262

TRW Germany
Electronic parking brake control system

ARP4754 & DO-178

Airbus Helicopters
Certified flight software

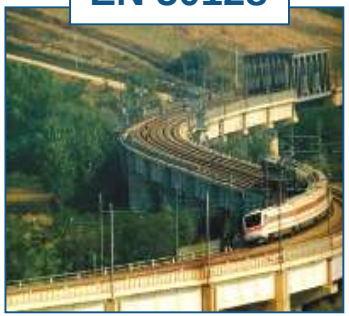
IEC 62304

Weinmann Medical Germany
Transport ventilator

IEC 61508

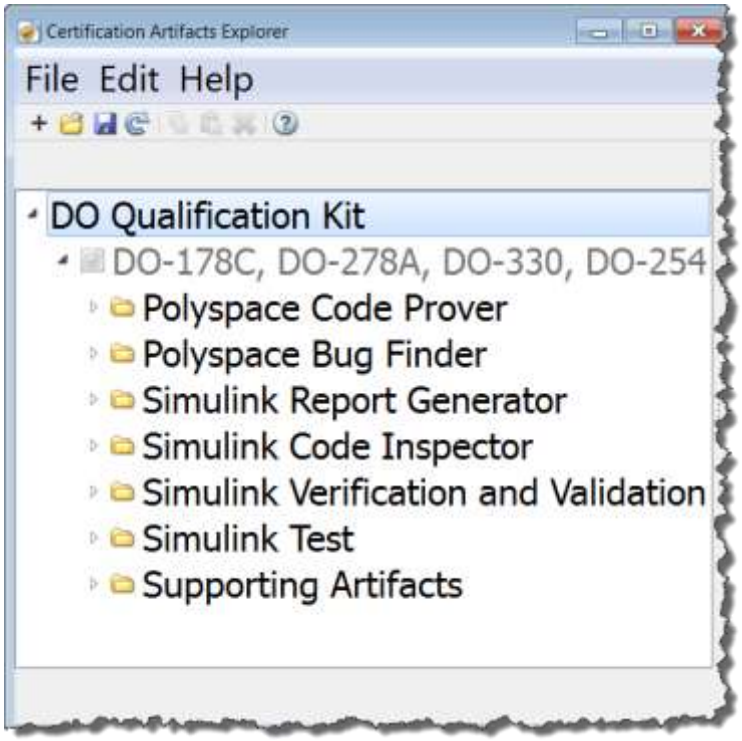
Alstom Grid UK
HDVC Power Systems

EN 50128



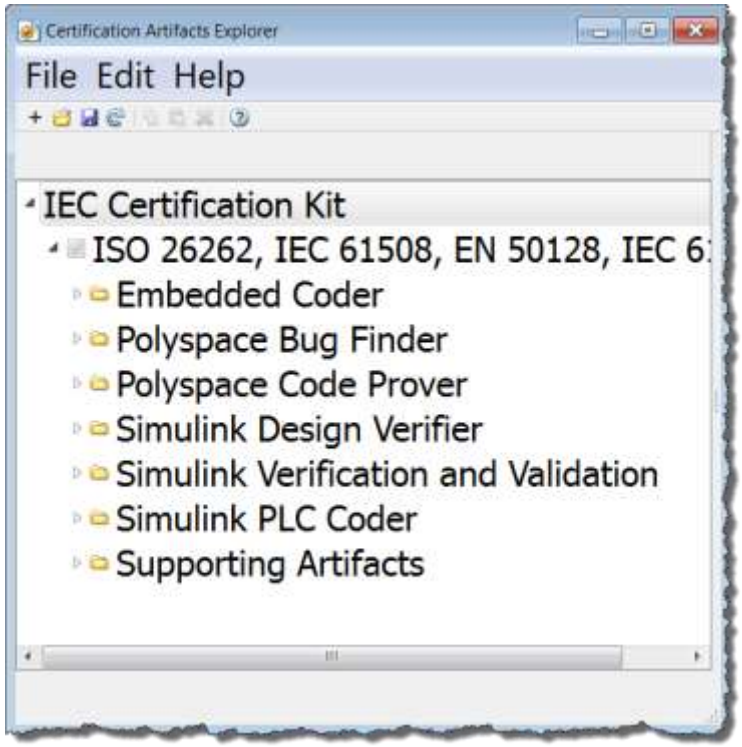
Alstom France
Train Control Systems

DO Qualification Kit



>>qualkitdo

IEC/ISO Certification Kit



>>certkitiec

Certification Mark:

Product: Software Tool for Safety Related Development

Model(s): Embedded Coder™
Real-Time Workshop® Embedded Coder™

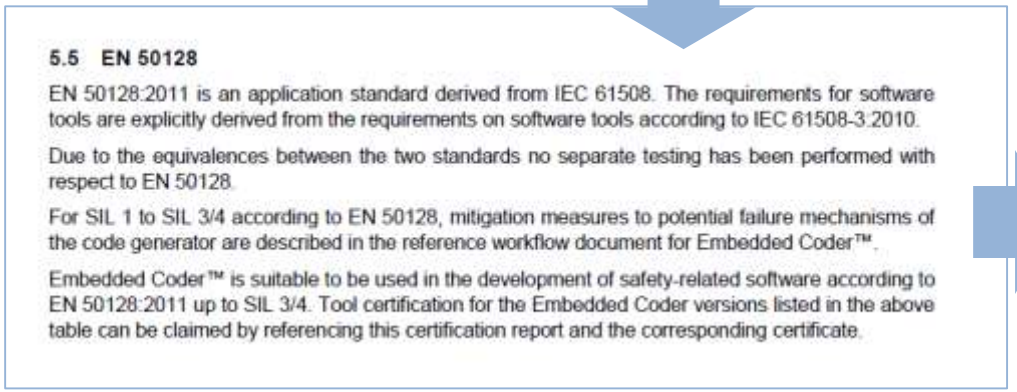
Parameters: The code generator is suitable for use to develop safety-related software according to IEC 61508 and EN 50128.
The code generator is a qualified tool according to ISO 26262.
The report MN72051C is a mandatory part of this certificate.

Report
to the
Certificate
Z10 11 12 67052 014
Software Tool for Safety Related Development
Embedded Coder™

Manufacturer
The MathWorks, Inc.
3 Apple Hill Drive
Natick, MA, 01760-2098
USA

Report No. MN72051C
Revision 2.7 dated 2015-05-29

Testing Body
TÜV SÜD Rail GmbH
Embedded Systems



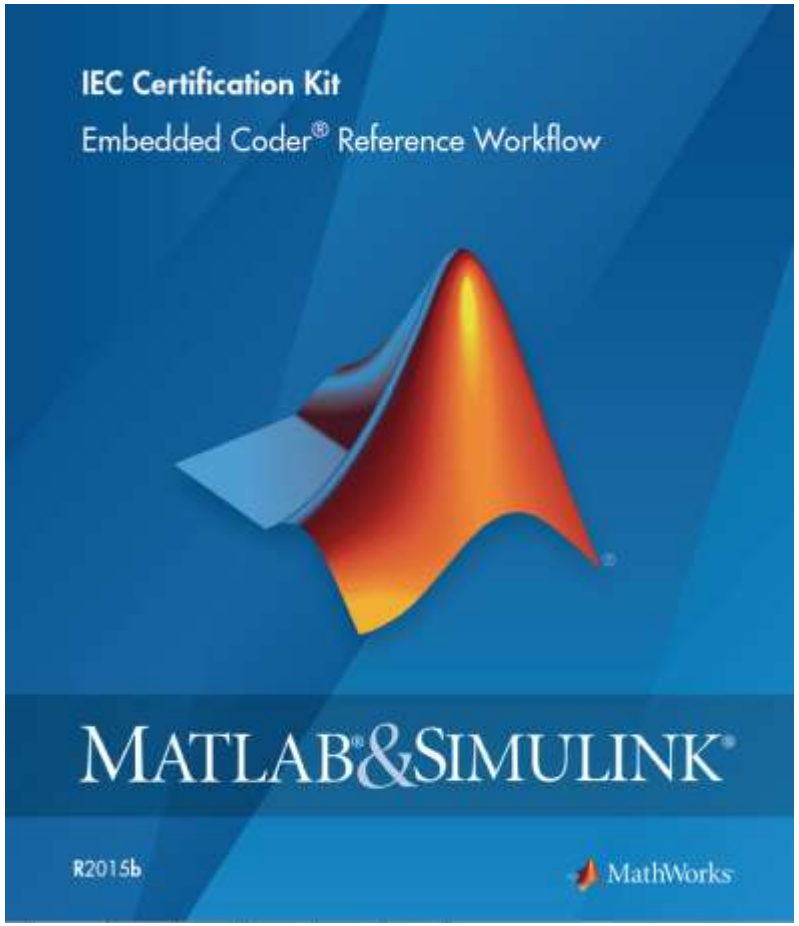
5.5 EN 50128

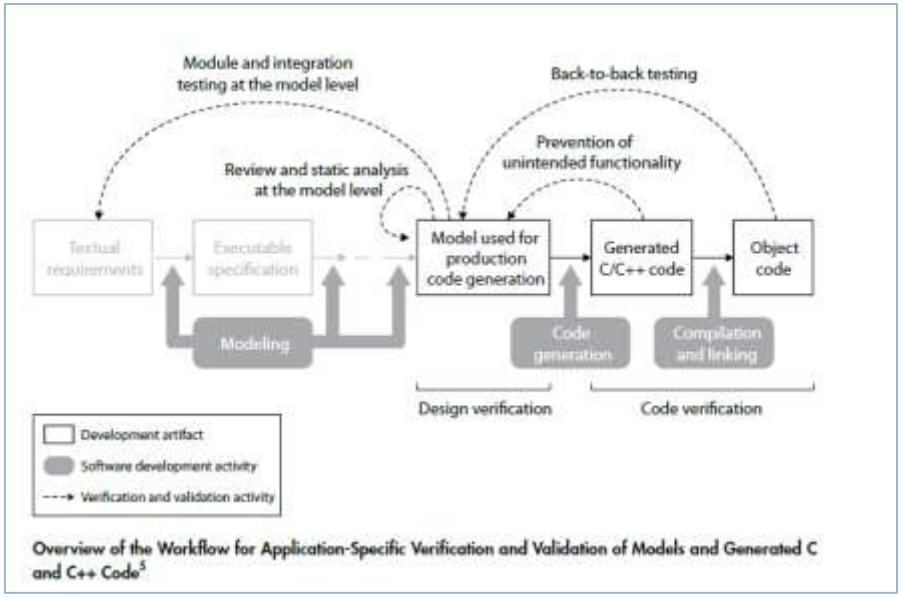
EN 50128:2011 is an application standard derived from IEC 61508. The requirements for software tools are explicitly derived from the requirements on software tools according to IEC 61508-3:2010.

Due to the equivalences between the two standards no separate testing has been performed with respect to EN 50128.

For SIL 1 to SIL 3/4 according to EN 50128, mitigation measures to potential failure mechanisms of the code generator are described in the reference workflow document for Embedded Coder™.

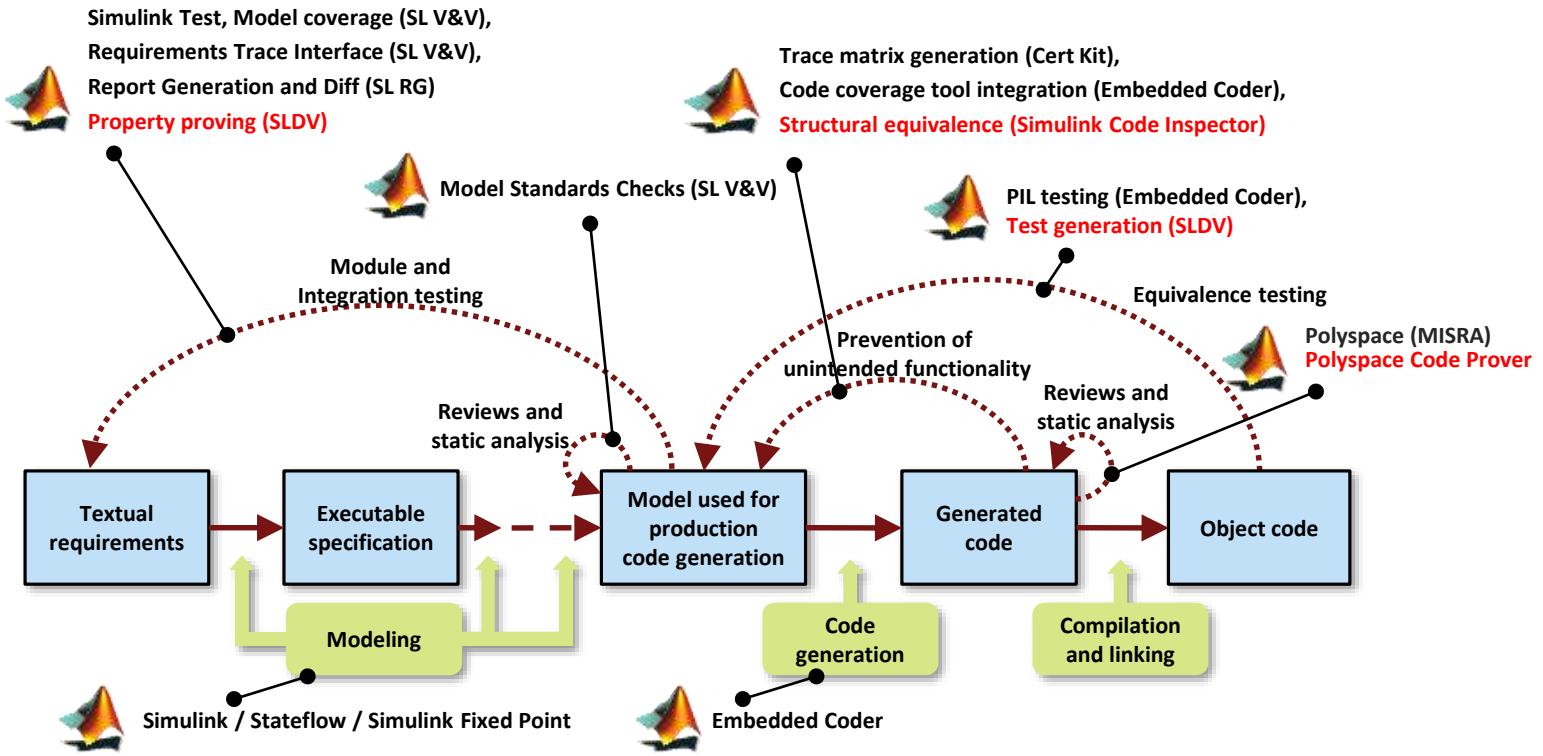
Embedded Coder™ is suitable to be used in the development of safety-related software according to EN 50128:2011 up to SIL 3/4. Tool certification for the Embedded Coder versions listed in the above table can be claimed by referencing this certification report and the corresponding certificate.





Overview of the Workflow for Application-Specific Verification and Validation of Models and Generated C and C++ Code[5]

参考流程


Simulink Test, Model coverage (SL V&V),
Requirements Trace Interface (SL V&V),
Report Generation and Diff (SL RG)
Property proving (SLDV)
Model Standards Checks (SL V&V)
Trace matrix generation (Cert Kit),
Code coverage tool integration (Embedded Coder),
Structural equivalence (Simulink Code Inspector)
PIL testing (Embedded Coder),
Test generation (SLDV)
Module and
Integration testing
Equivalence testing
Polyspace (MISRA)
Polyspace Code Prover
Prevention of
unintended functionality
Reviews and
static analysis
Reviews and
static analysis
Textual
requirements
Executable
specification
Model used for
production
code generation
Generated
code
Object code
Modeling
Code
generation
Compilation
and linking
Simulink / Stateflow / Simulink Fixed Point
Embedded Coder

思考一个问题：工程师为什么无法充分利用模型？

Simulink只用于仿真

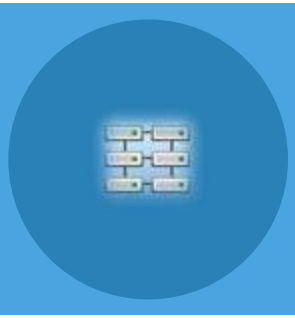
工程师使用Simulink仅用于算法仿真，并没有将Simulink平台作为全流程的开发工具，关键是仍没有突破自动代码生成的技术和心理障碍，模型验证更是无从谈起

模型无法在开发中传递

Simulink模型仿真完成后，工程师又忙于琐碎的文档和编码工作，模型无法向下传递，还没有成为知识传递与积累的载体，模型也没有实现标准化的设计

开发流程无法自动化

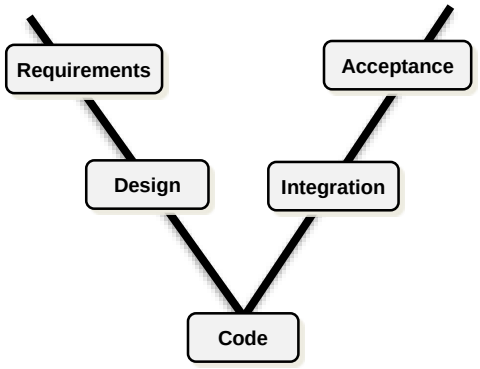
Simulink提供了建模/仿真/验证/代码生成等一些列工具，但这些工具都相对独立，并没有与特定的流程进行集成，所以导致工程师在使用时没有目标，或者就不知何时使用



团队协作过于粗放

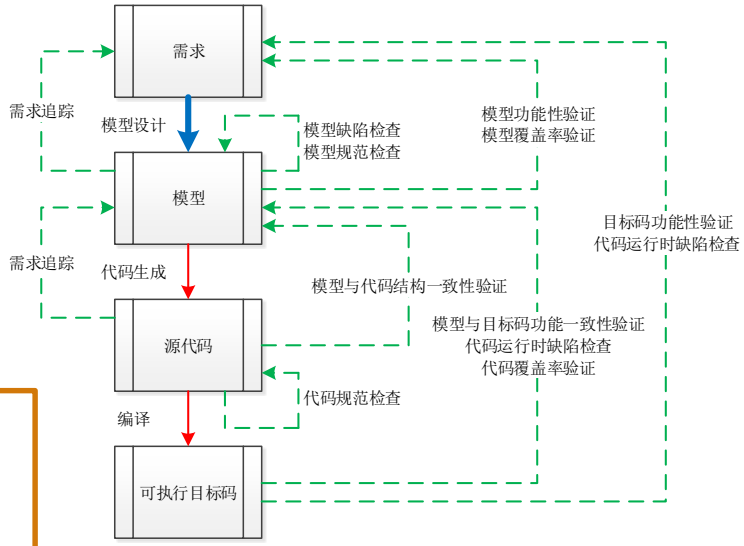
在团队协作中，流程的定义相对粗放，没有规定每一个环节工程师详细的工作职责/使用工具/输入输出等，而且如模型验证等环节都是缺失的，模型没有进入到设计开发流程中

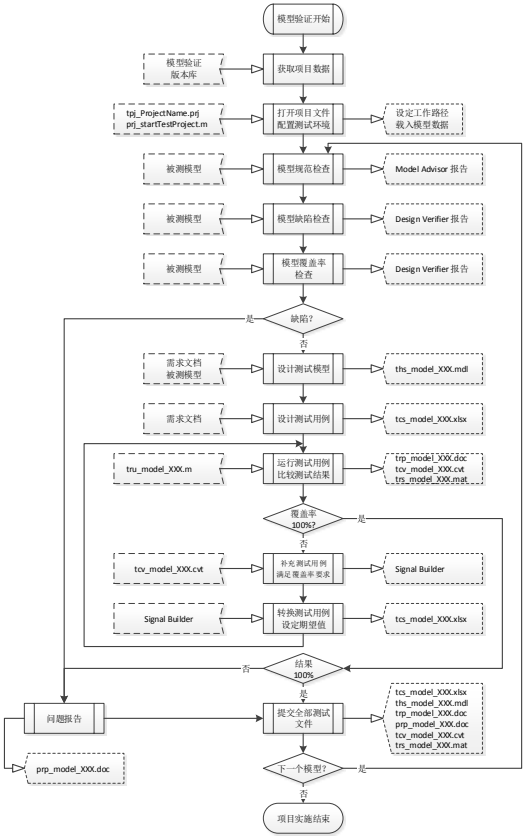
从“V”流程到“微”流程



“微”流程

解决了设计过程中的实现操作定义，每个操作都是可执行的，工程师可以参照它开展工作

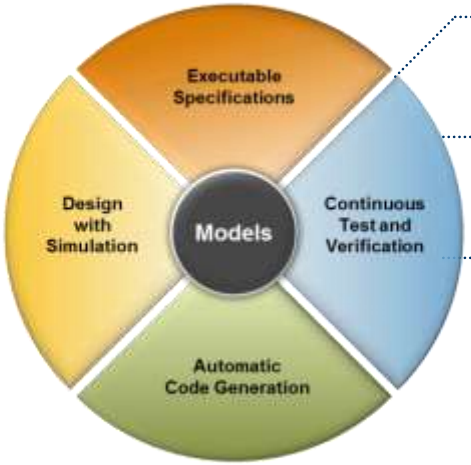




“V”流程

解决了设计过程中主要环节定义，但没有具体的实现，工程师无法参照它开展工作

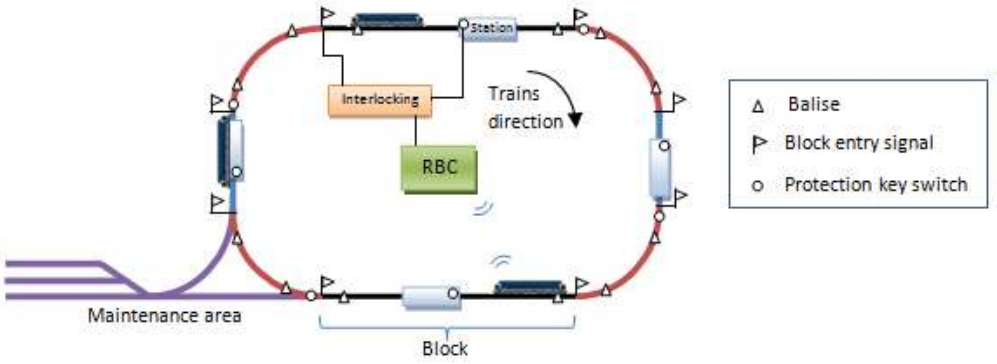
内容概要

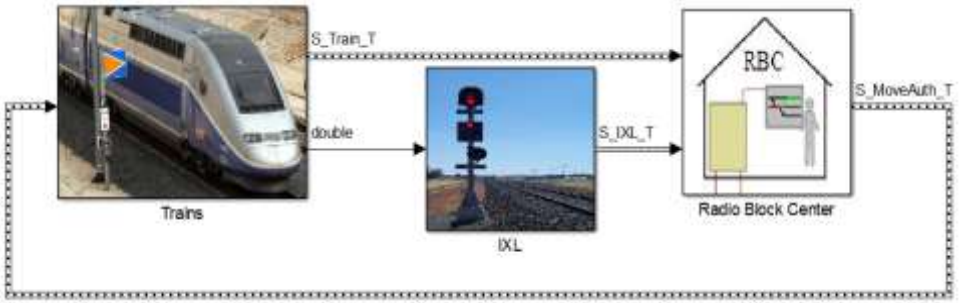


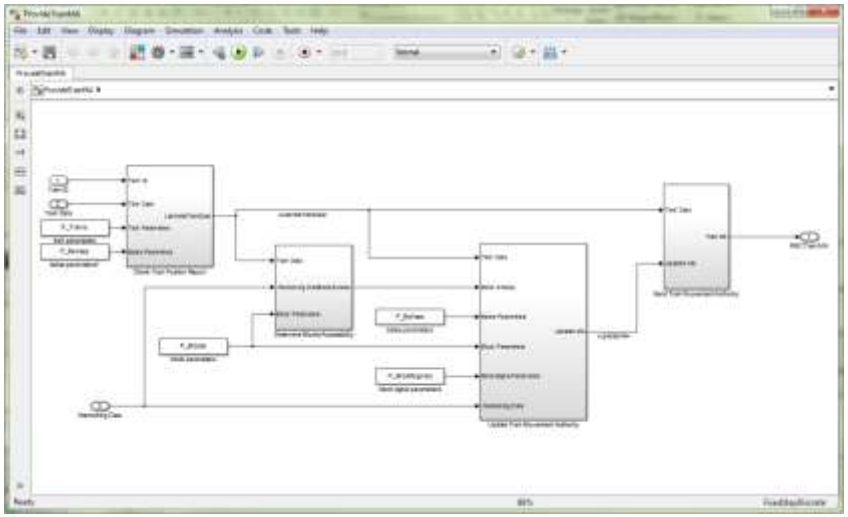
- 国内外高安全关键系统开发与验证的趋势
- 符合行业标准的基于模型设计解决方案
- 轨道交通行业符合EN50128标准项目实践

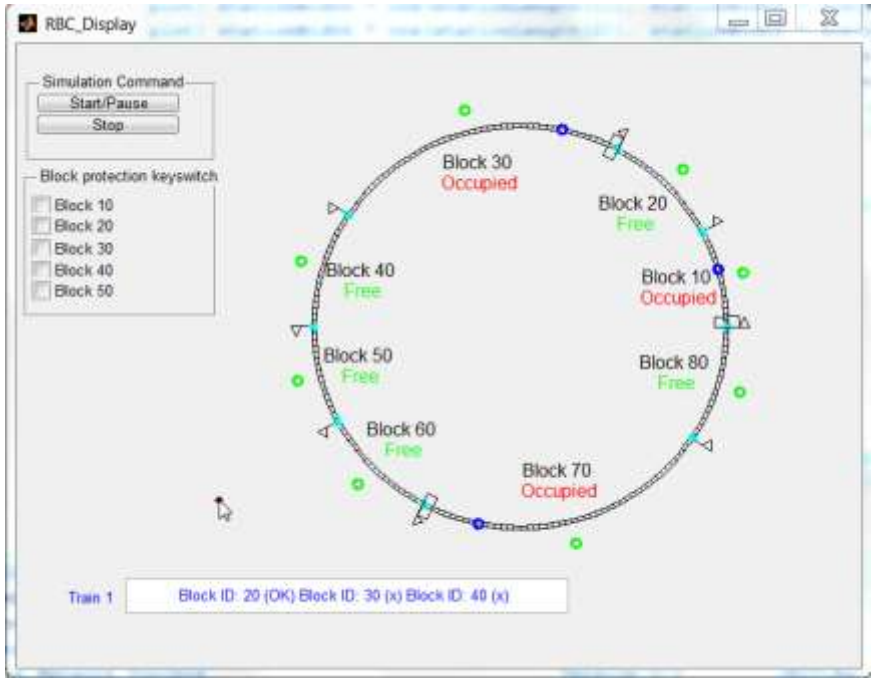
案例项目背景

- 欧洲铁路运输管理系统（ERTMS） Level 2
- Radio Block Center（RBC）系统
- 项目目标
  - 符合SIL4项目的最佳实践
  - EN50128标准符合性指导

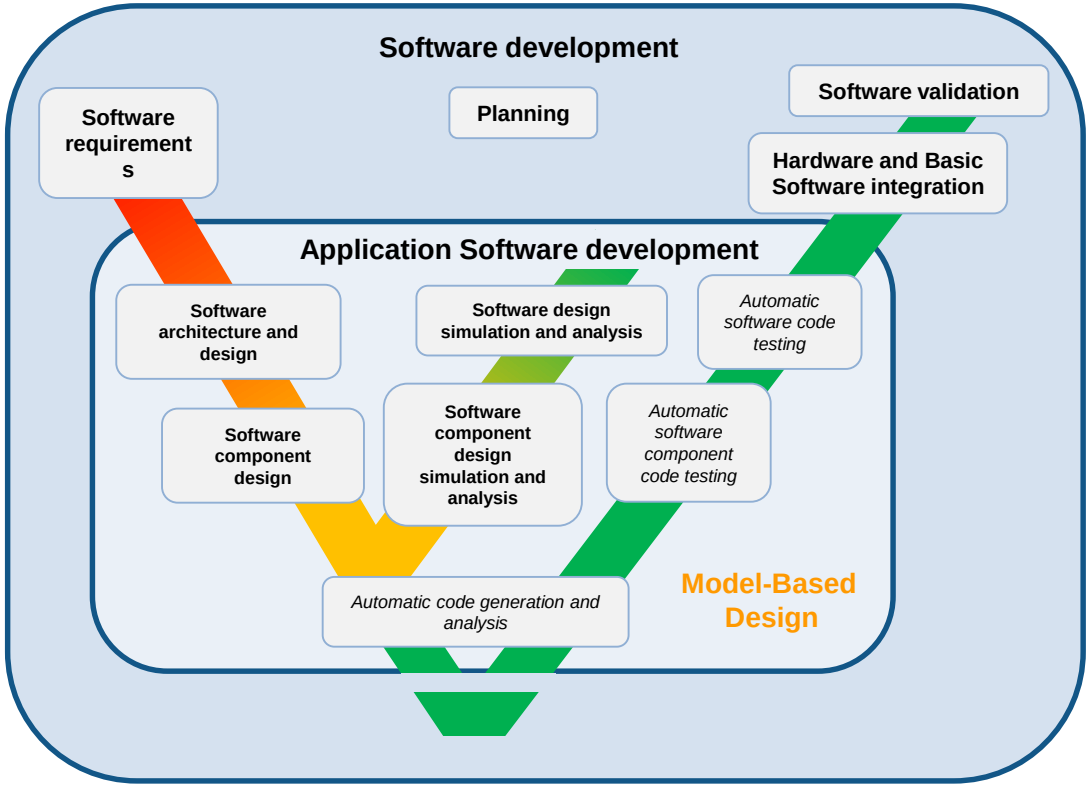




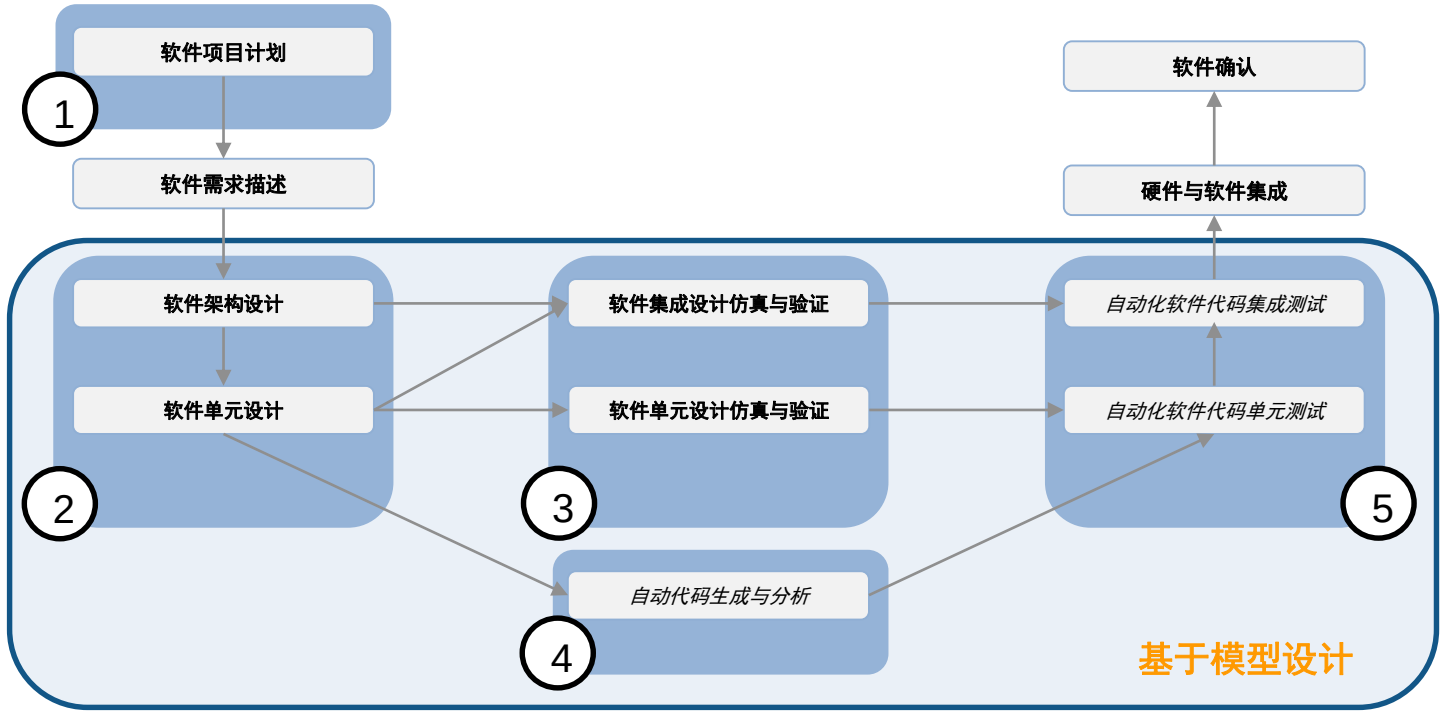



RBC_Display
Simulation Command
Start/Pause
Stop
Block protection keyswitch
Block 10
Block 20
Block 30
Block 40
Block 50
Block 30
Occupied
Block 20
Free
Block 40
Free
Block 10
Occupied
Block 50
Free
Block 80
Free
Block 60
Free
Block 70
Occupied
Train 1
Block ID: 20 (OK) Block ID: 30 (x) Block ID: 40 (x)

基于模型设计：V流程

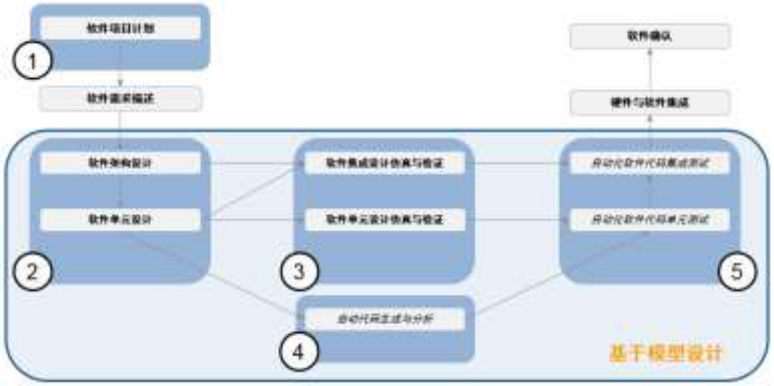

Software development
Software requirements
Planning
Software validation
Hardware and Basic Software integration
Application Software development
Software architecture and design
Software component design
Software design simulation and analysis
Software component design simulation and analysis
Automatic software code testing
Automatic software component code testing
Automatic code generation and analysis
Model-Based Design

基于模型设计：实施环节


软件项目计划
1
软件需求描述
软件架构设计
软件单元设计
2
软件集成设计仿真与验证
软件单元设计仿真与验证
3
自动代码生成与分析
4
软件确认
硬件与软件集成
自动化软件代码集成测试
自动化软件代码单元测试
5
基于模型设计

基于模型设计在项目中的实施

① 项目计划：基于模型的软件开发项目计划
② 软件设计：基于模型的软件架构与单元设计
③ 模型验证：基于模型的设计仿真与验证
④ 代码生成：基于模型的自动代码生成与分析
⑤ 代码验证：基于模型的代码集成与验证



① 基于模型的软件开发项目计划

使用Certification Kit对流程进行定义

- ✓ 明确定义项目范围
- ✓ 在软件质量保证计划（SQAP）中根据**TÜV SÜD for SIL4**批准的流程详细制定项目开发流程
- ✓ 必须详细描述流程的细节并且映射到**EN50128**表格中

CERTIFICATE

No. Z10 11 12 67052 014

Holder of Certificate: The MathWorks, Inc.
3 Apple Hill Drive
Natick MA 01760-2098
USA

Factory(ies): 67052

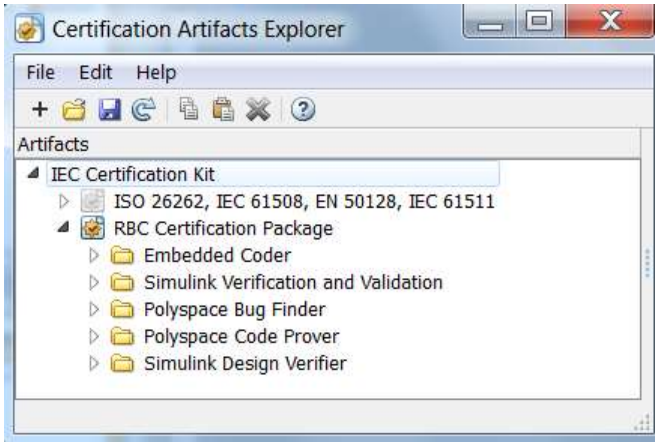
Certification Mark:

标准和符合性验证

- ✓ 通过**Model Advisor**中EN50128检查项自动验证模型标准的符合性
- ✓ 通过**Polyspace**中MISRA AC AGG检查工具自动验证代码标准的符合性

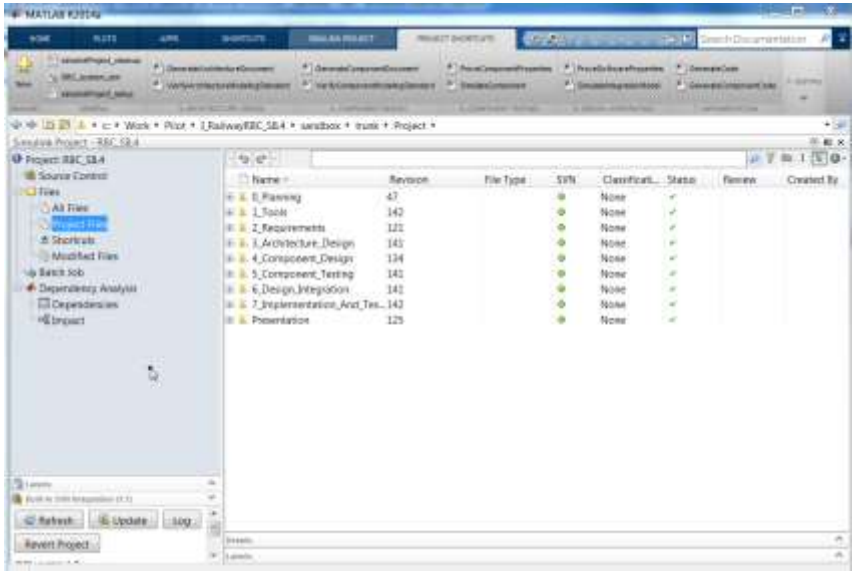
使用Certification Kit对工具进行认证

- 对项目建立认证包**实例**
- Certification Kit 针对每个MATLAB版本都会进行升级
- **自动**运行工具的测试程序并且生成报告



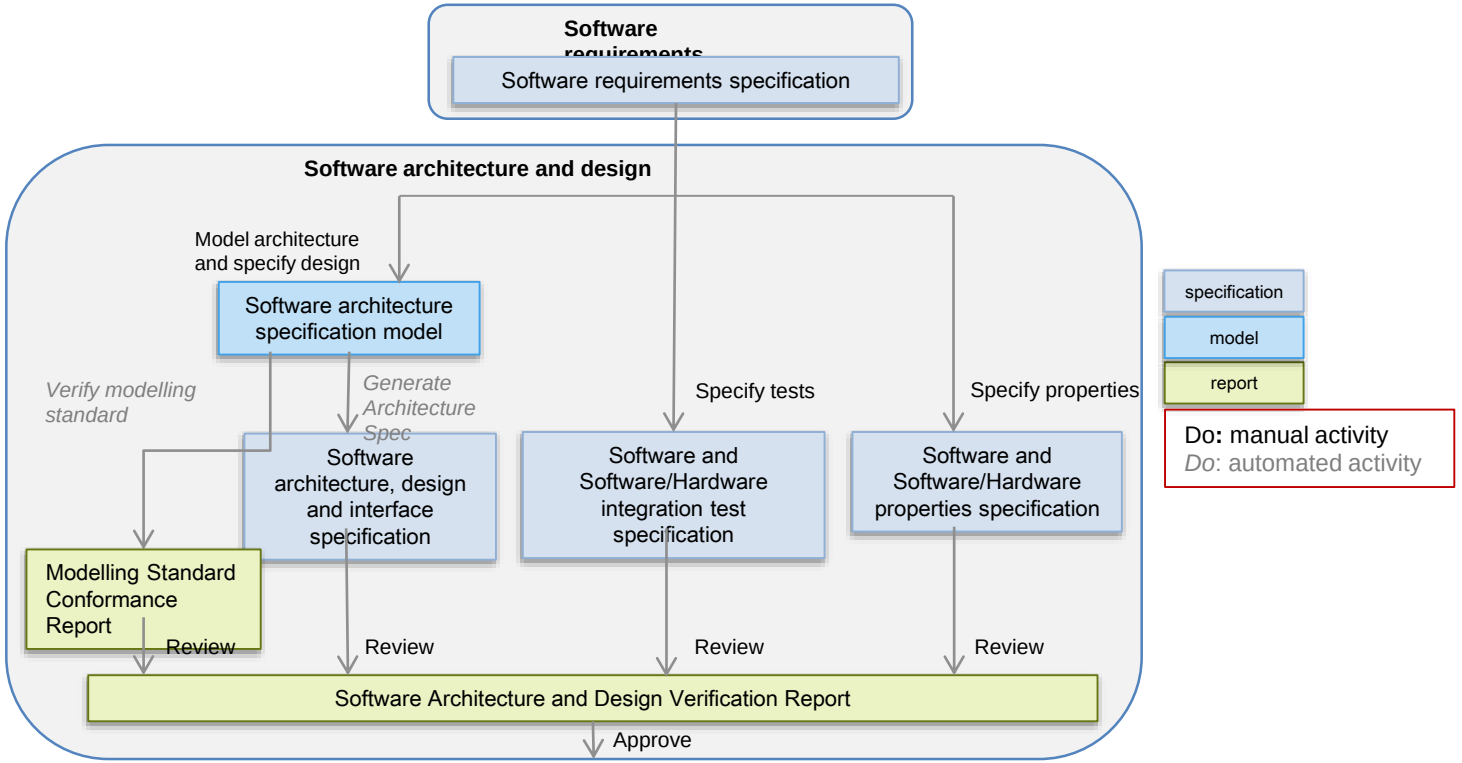
使用Simulink Project管理项目

- ✓ 确保工具配置的一致性
- ✓ 通过行为执行的自动化快捷操作确保流程的符合性
- ✓ 在CM下保持项目团队的合作
- ✓ 提供完整项目的依赖关系分析
- ✓ 通过自动化脚本实现行为执行的批处理操作



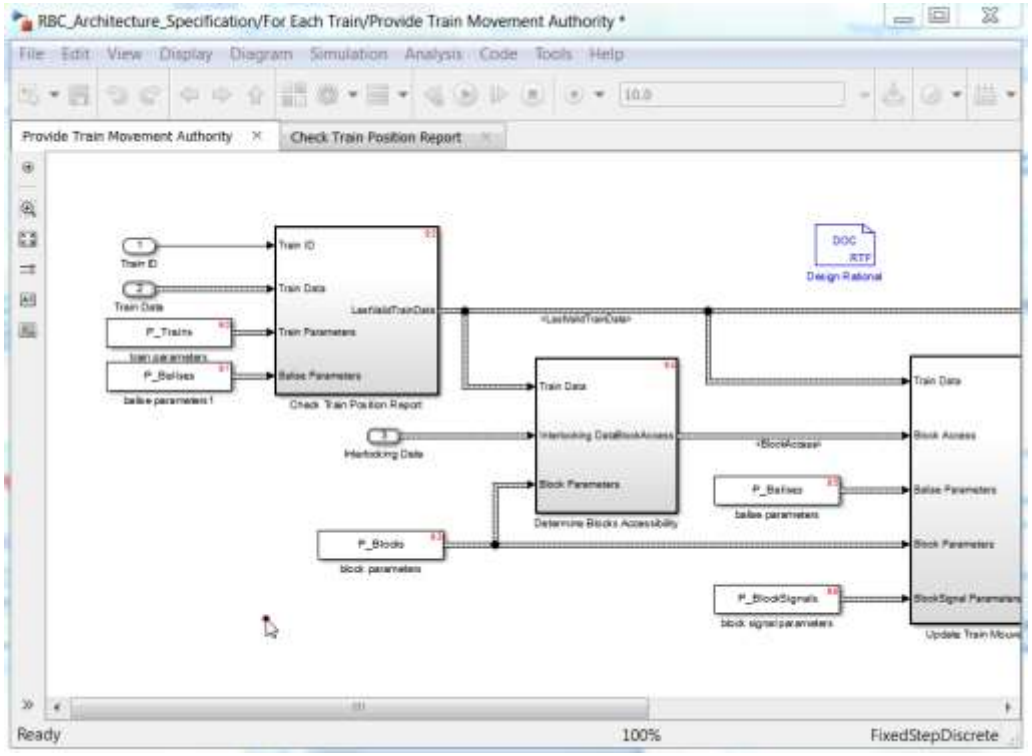
② 基于模型的软件架构与单元设计

软件架构设计


Software
Software requirements specification
Software architecture and design
Model architecture and specify design
Software architecture specification model
Verify modelling standard
Generate Architecture Spec
Specify tests
Specify properties
Software architecture, design and interface specification
Software and Software/Hardware integration test specification
Software and Software/Hardware properties specification
Modelling Standard Conformance Report
Review
Review
Review
Review
Software Architecture and Design Verification Report
Approve
specification
model
report
Do: manual activity
Do: automated activity

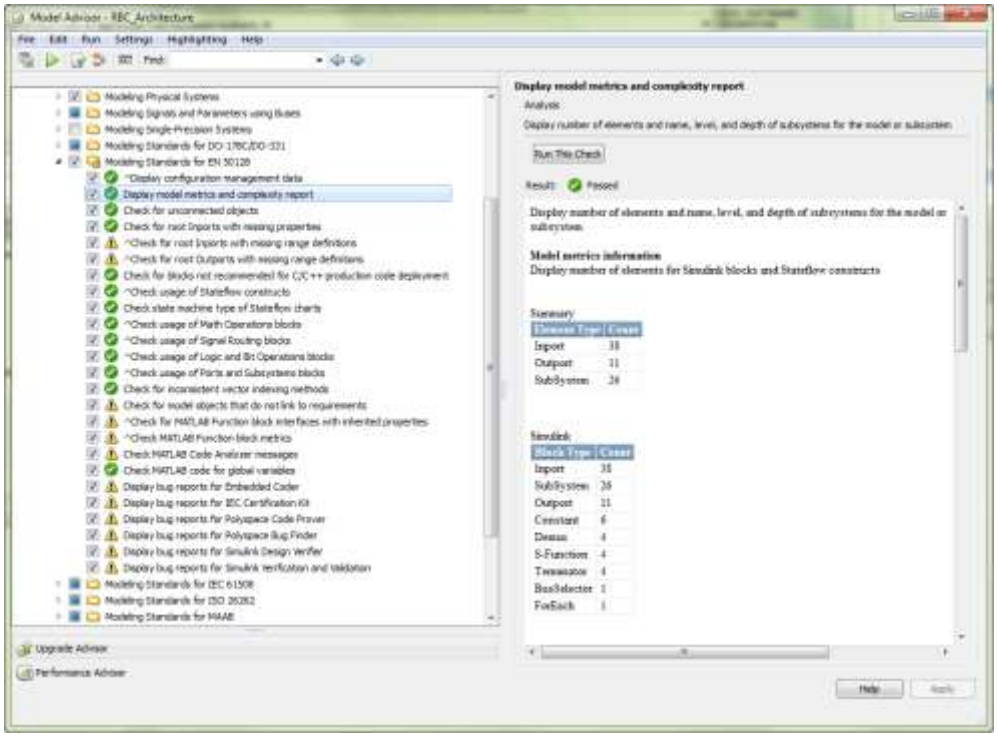
软件架构设计中的开发活动

- 架构模型限定在应用层软件
- 开发软件架构描述模型
  - 使用Simulink图形化描述模块划分和数据流
  - 使用MATLAB描述接口特性
  - 使用富文本注释在模型中描述设计依据和需求
  - 建立模型与需求的追踪
  - 软件架构设计文档通过模型自动生成

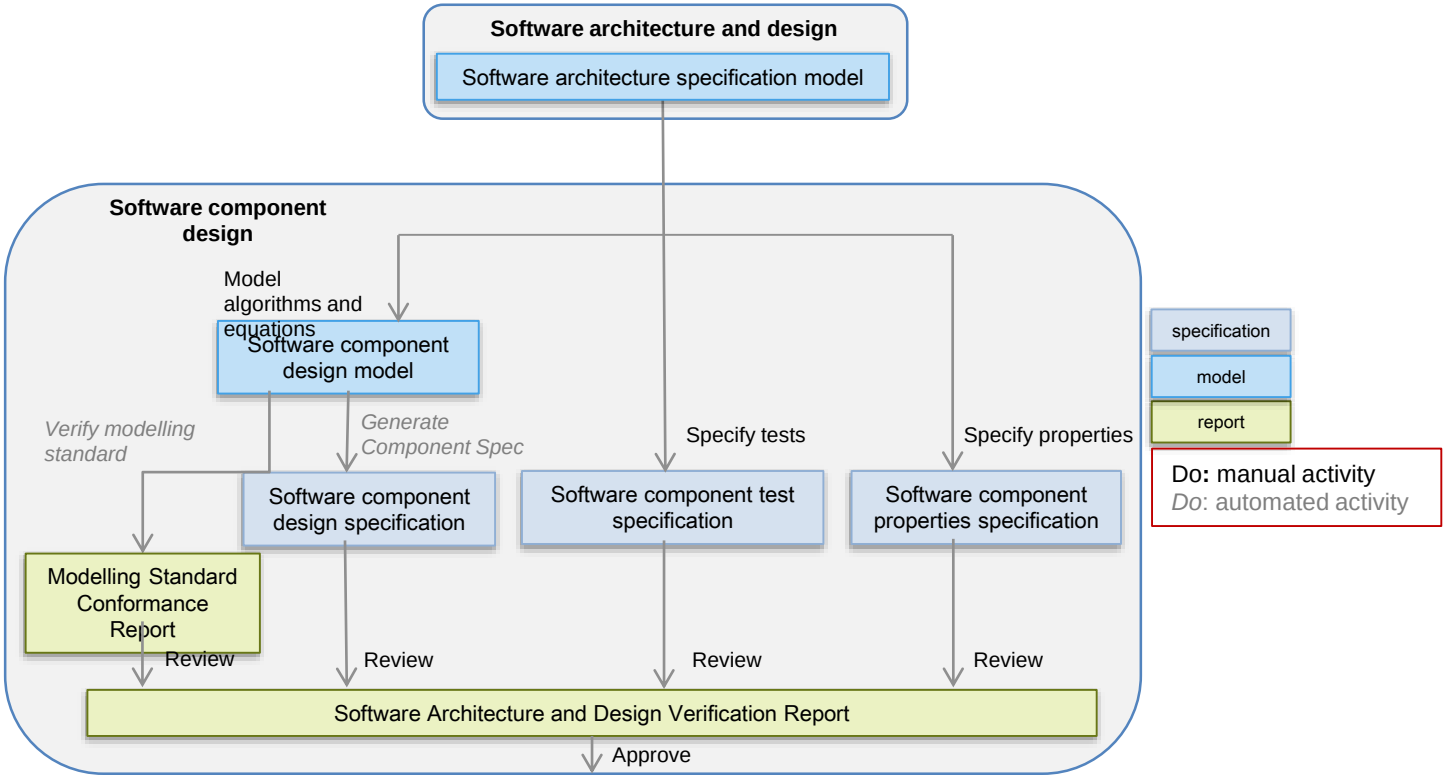


软件架构设计中的验证活动

- 描述软件集成测试
  - 软件集成测试用例与需求建立追踪
- 验证模型规范的符合性
- 验证测试描述
  - 同级审核
- 验证软件架构描述
  - 同级审核

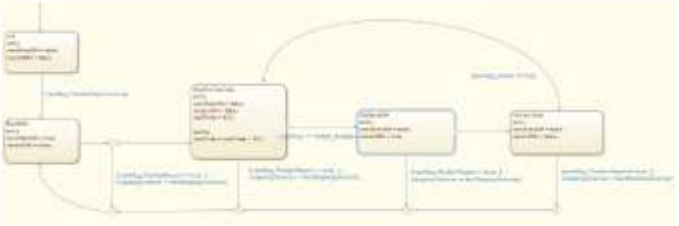


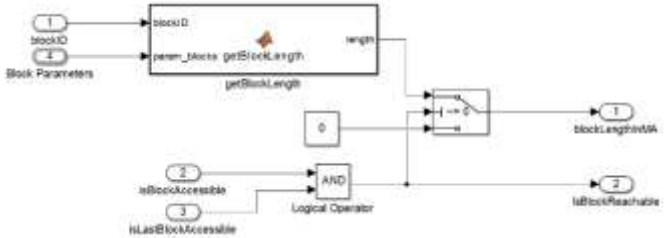
软件单元设计


Software architecture and design
Software architecture specification model
Software component design
Model algorithms and equations
Software component design model
Verify modelling standard
Generate Component Spec
Specify tests
Specify properties
Software component design specification
Software component test specification
Software component properties specification
Modelling Standard Conformance Report
Review
Review
Review
Review
Software Architecture and Design Verification Report
Approve
specification
model
report
Do: manual activity
Do: automated activity

软件单元设计中的开发活动

- 开发软件单元描述模型
  - 软件架构模型描述单元接口
  - 根据架构模型中的单元需求开发具体的算法和方程
  - 设计依据包含在模型中的富文本注释中
  - 软件单元需求与模型建立追踪
  - 软件单元描述文档通过模型自动生成

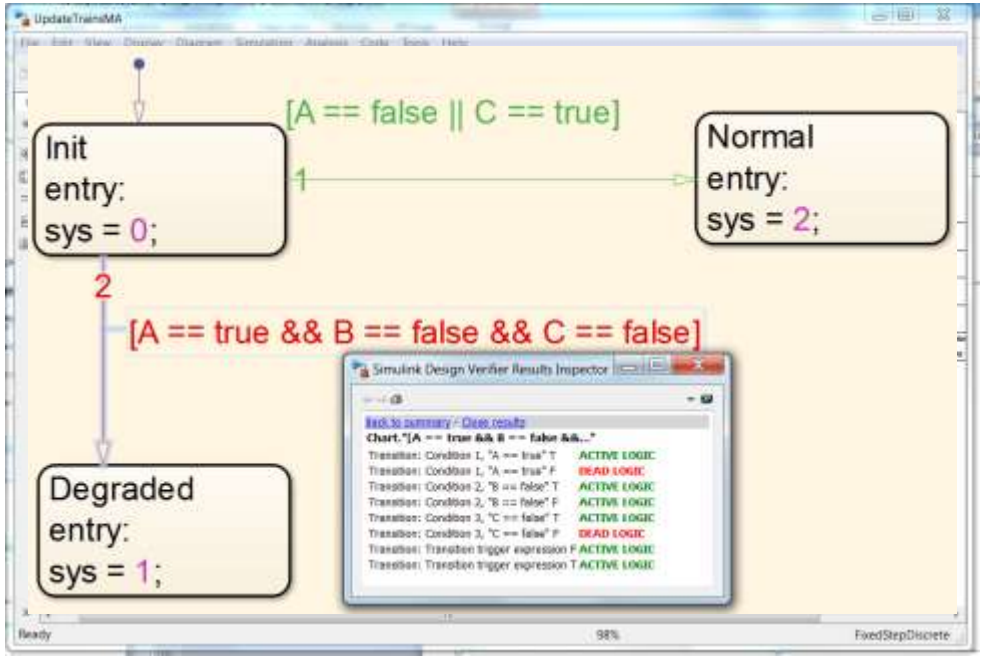





```
function trainIDIsValid = CheckTrainIDValid(TrainData,param_trains, param
%#codegen

% init
trainIDIsValid = false;
baliseIDValid = false;

for idx = uint32(1):numel(param_trains.trainID)
    if TrainData.trainID == param_trains.trainID(idx)
        trainIDIsValid = true;
    end
end

for idx = uint32(1):numel(param_balises.ID)
    if TrainData.lastBaliseID == param_balises.ID(idx)
        if TrainData.distanceFromLastBalise < param_balises.distanceToNex
            baliseIDValid = true;
        end
    end
end
```

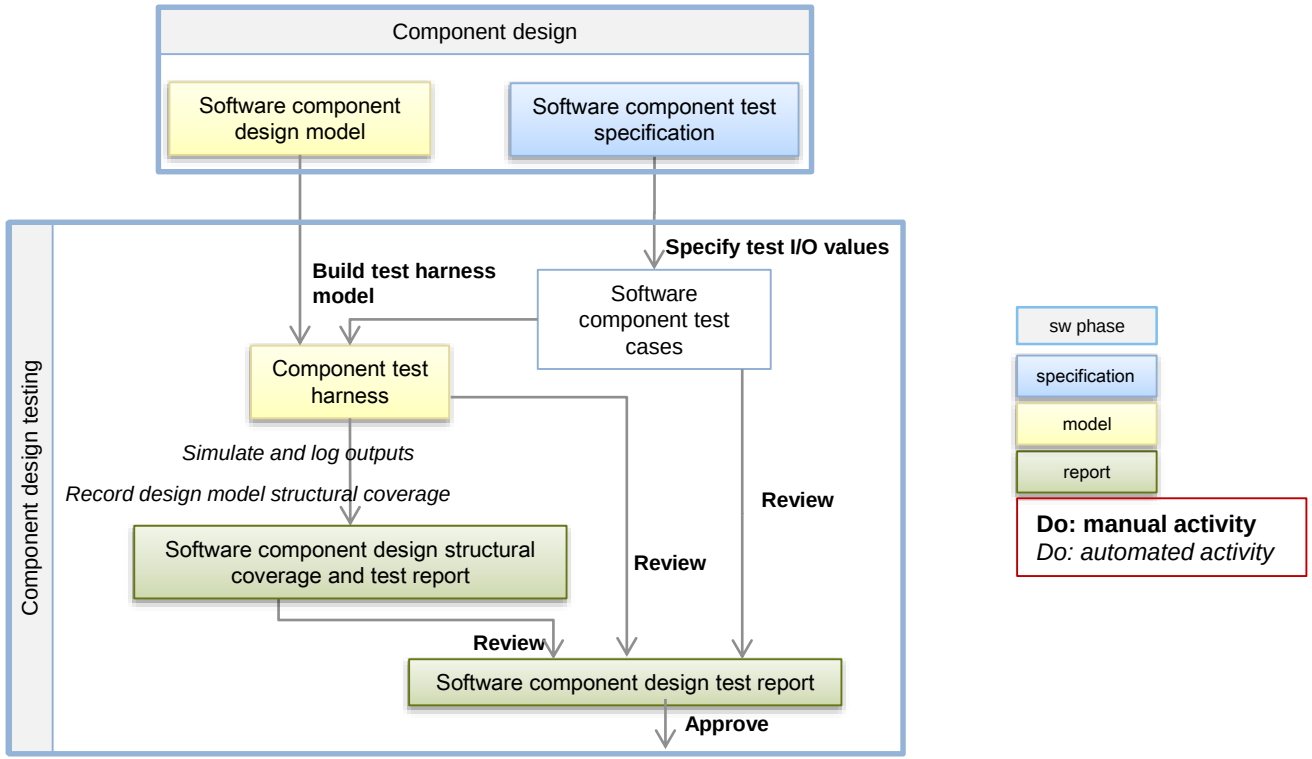
软件单元设计中的验证活动

- 验证软件单元模型
  - 模型规范符合性自动化验证
  - 软件单元描述同级审核
    - 需求追踪型验证
    - 设计鲁棒性验证
    - 设计可测试性验证
- 详尽验证单元测试描述
  - 软件单元测试描述同级审核



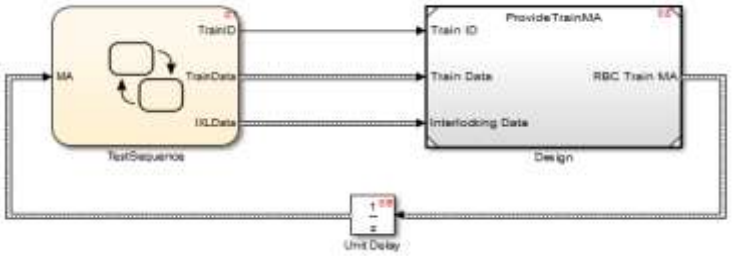
③ 基于模型的设计仿真与验证

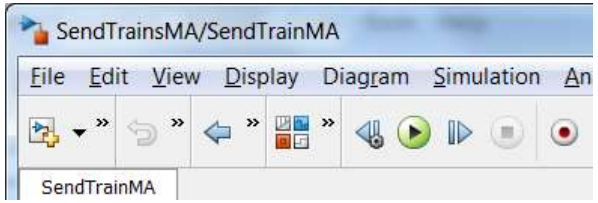
模型单元测试


Component design
Software component design model
Software component test specification
Component design testing
Specify test I/O values
Build test harness model
Software component test cases
Component test harness
Simulate and log outputs
Record design model structural coverage
Review
Software component design structural coverage and test report
Review
Review
Software component design test report
Approve
sw phase
specification
model
report
Do: manual activity
Do: automated activity

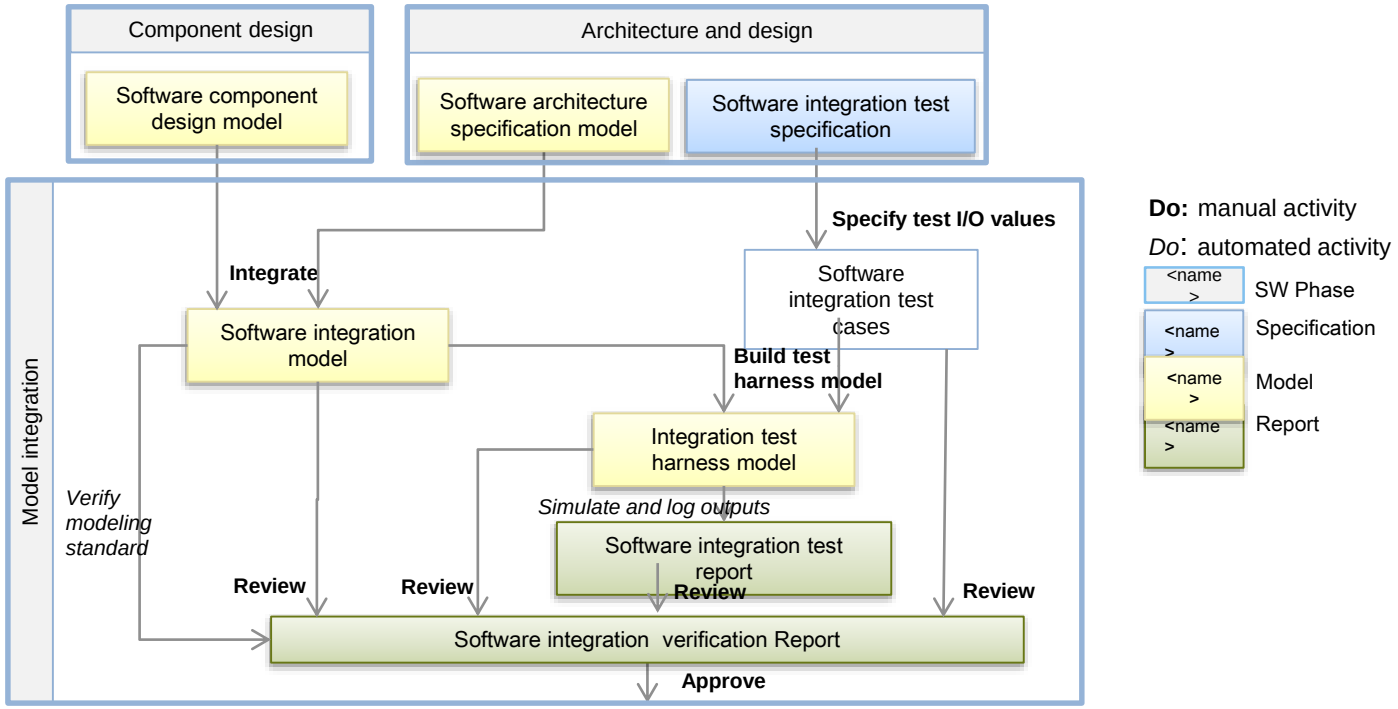
模型单元测试中的活动

- 创建测试模型
  - 引用被测设计模型并将其连接到测试输入
  - 比较测试的期望值与模型输出值
  - 启动单元模型的结构覆盖率测量
  - 启动信号信号范围覆盖率测量
- 执行测试程序
  - 载入测试数据到MATLA工作空间
  - 仿真测试模型
  - 统计累积覆盖率并生成报告
- 审核模型单元测试描述和报告



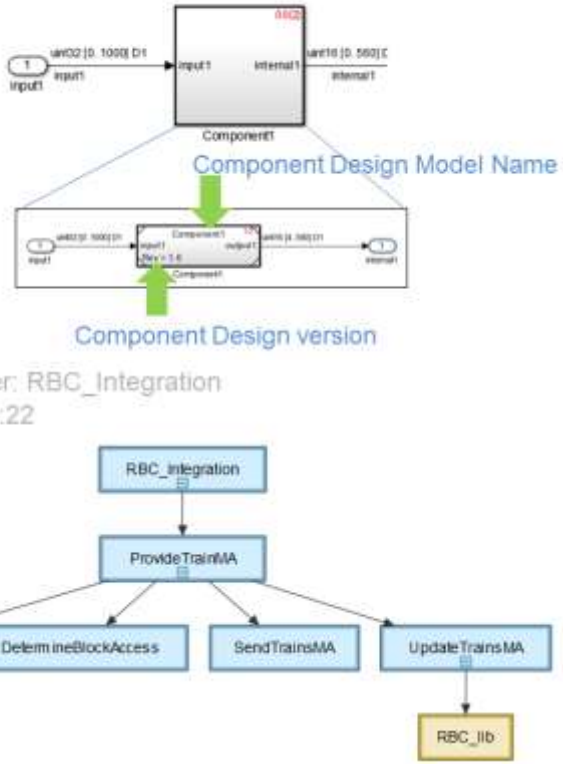


模型集成与测试


Component design
Software component design model
Architecture and design
Software architecture specification model
Software integration test specification
Specify test I/O values
Integrate
Software integration model
Software integration test cases
Build test harness model
Integration test harness model
Model integration
Verify modeling standard
Simulate and log outputs
Software integration test report
Review
Review
Review
Review
Software integration verification Report
Approve
Do: manual activity
Do: automated activity
<name>
SW Phase
<name>
Specification
<name>
Model
<name>
Report

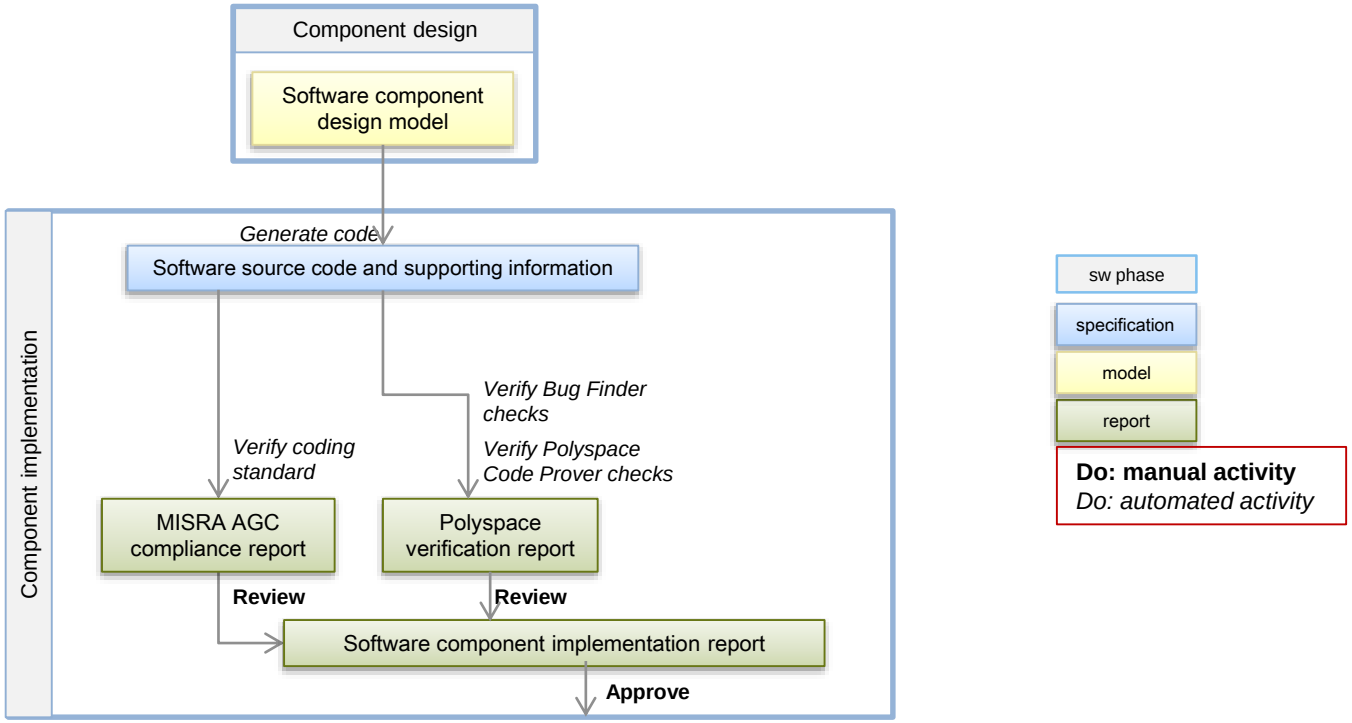
模型集成与测试中的活动

- 开发:
  - 将单元模型集成到软件架构模型中
- 测试:
  - 创建测试模型
  - 执行测试程序
- 验证:
  - 验证集成模型的规范符合性
  - 审核集成模型
  - 审核测试报告



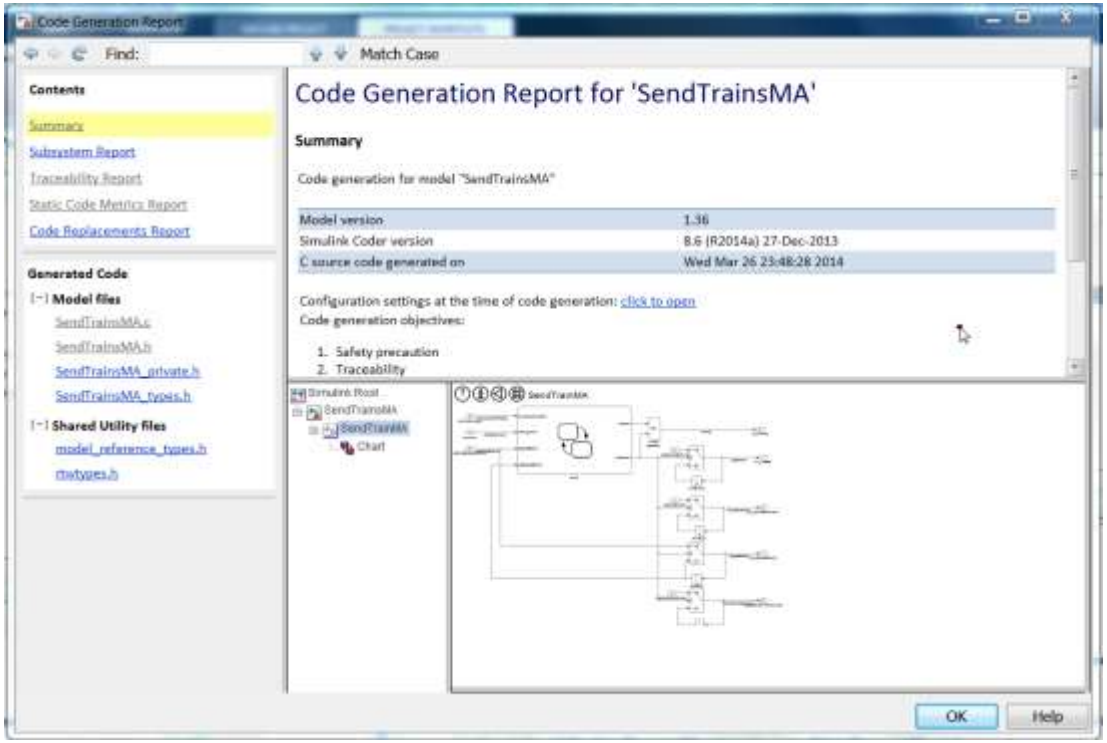
④ 基于模型的自动代码生成

自动代码生成与分析


Component design
Software component design model
Component implementation
Generate code
Software source code and supporting information
Verify coding standard
Verify Bug Finder checks
Verify Polyspace Code Prover checks
MISRA AGC compliance report
Polyspace verification report
Review
Review
Software component implementation report
Approve
sw phase
specification
model
report
Do: manual activity
Do: automated activity

自动代码生成

- 可读性与可追溯性
- 自动生成具有代码和模型双向链接的报告
- 提供代码生成指标
- 通过配置确保C代码中的算法结构

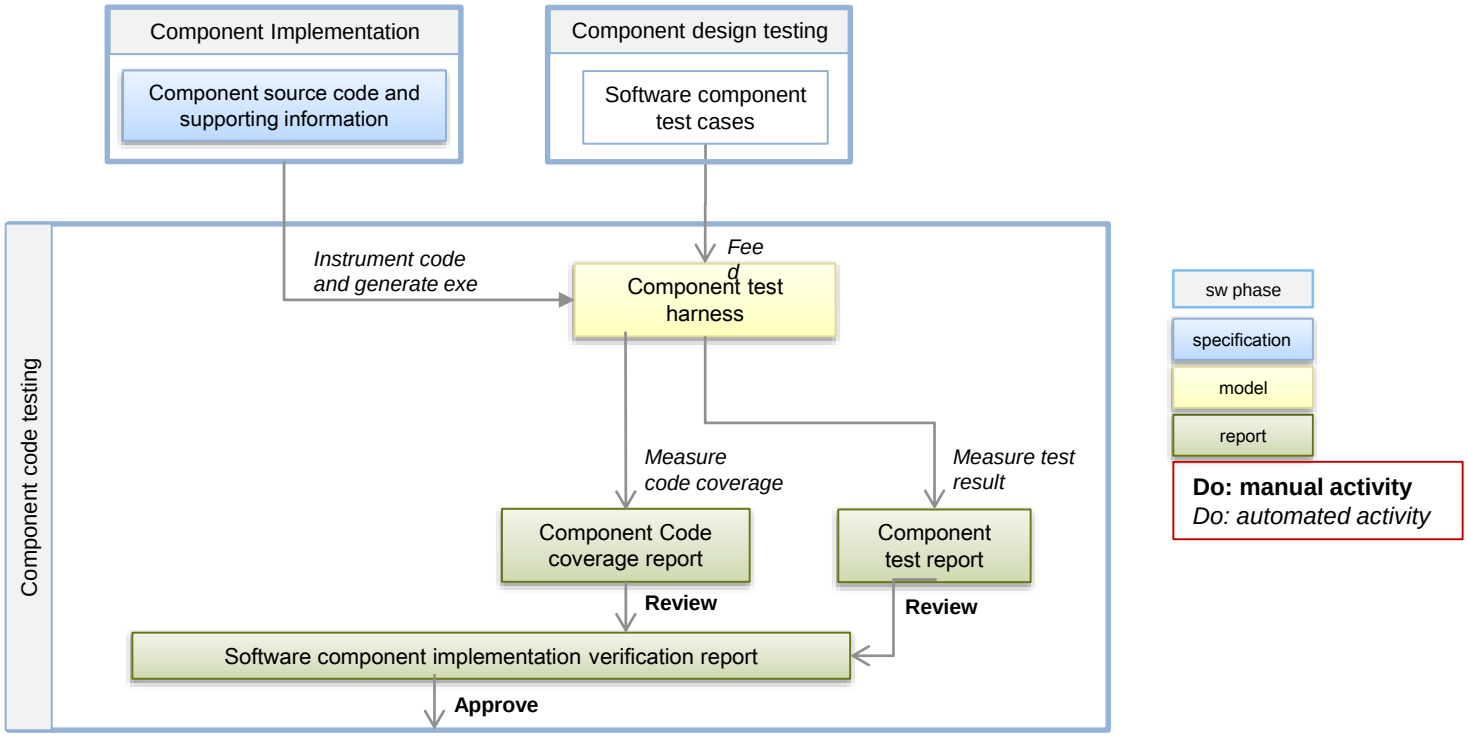


自动代码分析

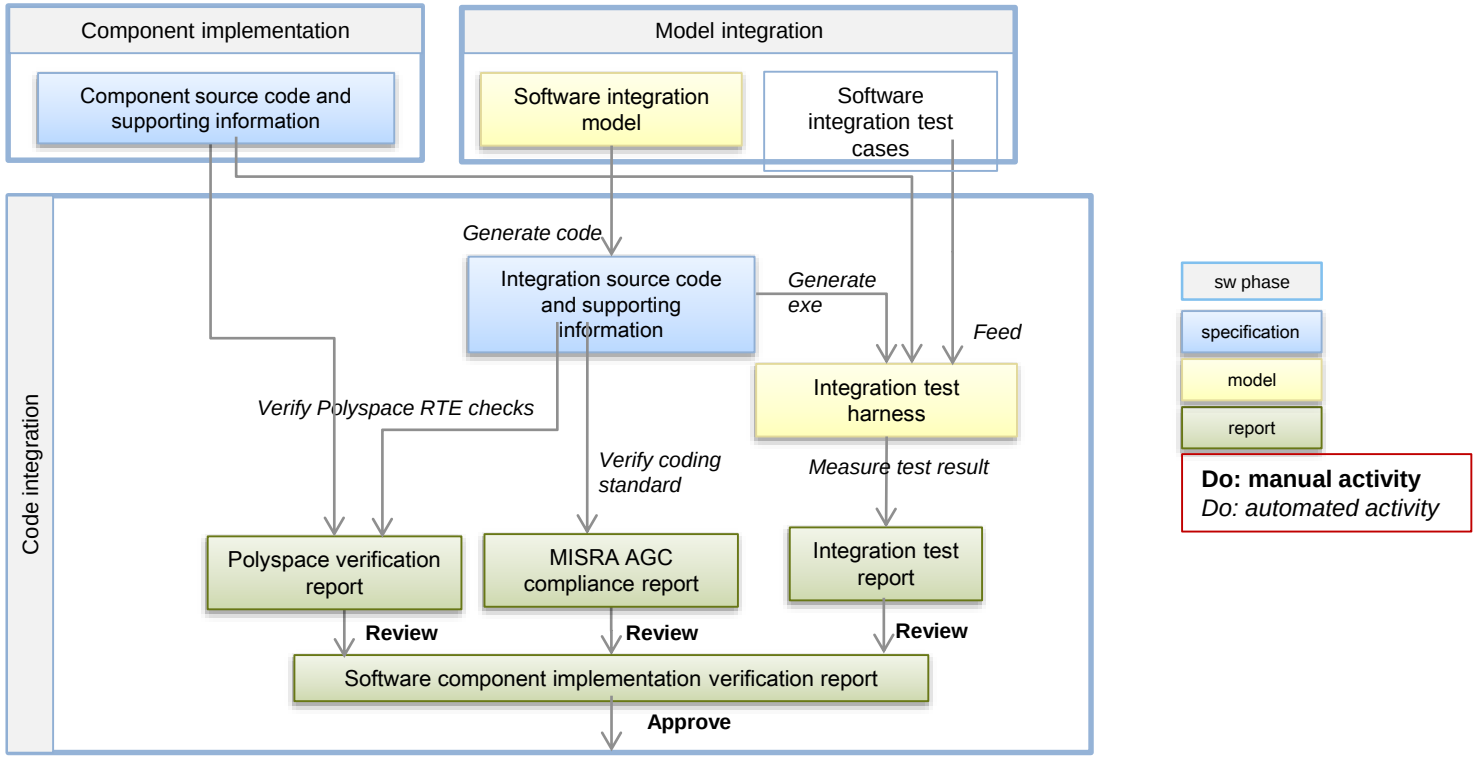
- 使用Polyspace Bug Finder验证生成代码与**MISRA**标准的符合性
- 使用Polyspace Code Prover验证生成代码的鲁棒性

⑤ 基于模型的代码集成与验证

代码单元测试


Component Implementation
Component source code and supporting information
Component design testing
Software component test cases
Component code testing
Instrument code and generate exe
Fee d
Component test harness
Measure code coverage
Measure test result
Component Code coverage report
Component test report
Review
Review
Software component implementation verification report
Approve
sw phase
specification
model
report
Do: manual activity
Do: automated activity

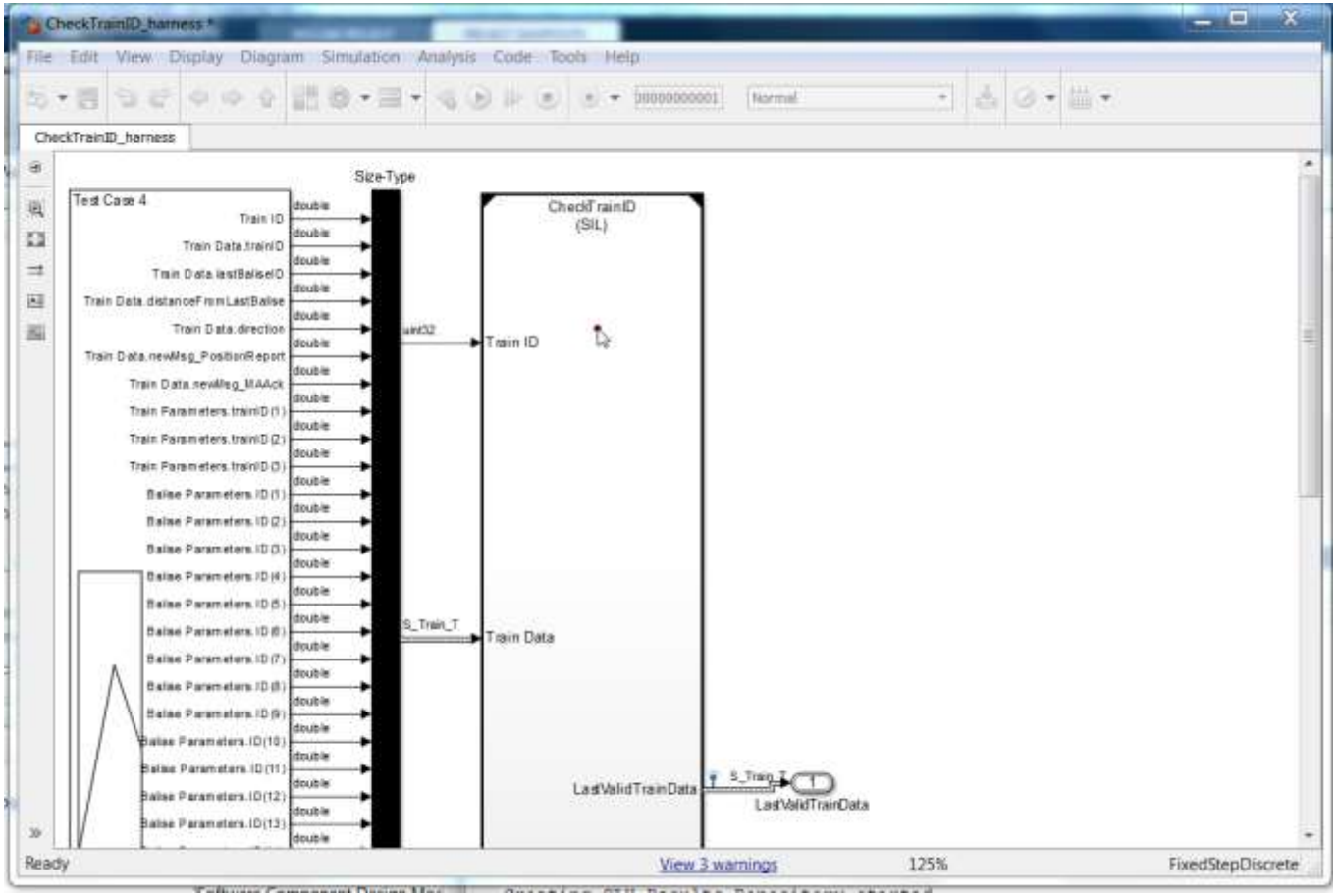
代码集成测试


Component implementation
Component source code and supporting information
Model integration
Software integration model
Software integration test cases
Code integration
Generate code
Integration source code and supporting information
Generate exe
Feed
Integration test harness
Verify Polyspace RTE checks
Verify coding standard
Measure test result
Polyspace verification report
MISRA AGC compliance report
Integration test report
Review
Review
Review
Software component implementation verification report
Approve
sw phase
specification
model
report
Do: manual activity
Do: automated activity

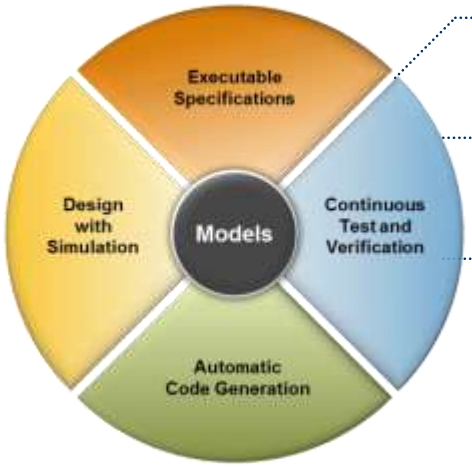
自动化代码测试

- ✓ 使用Simulink Design Verifier自动生成背靠背测试用例或正重用基于需求的测试用例
- ✓ 自动化软件在环测试或者处理器在环测试
- ✓ 使用集成的第三方工具LDRA统计代码覆盖率
- ✓ 使用Simulink Data Inspector比较模型与代码的输出

自动化代码测试


CheckTrainID_harness *
File Edit View Display Diagram Simulation Analysis Code Tools Help
CheckTrainID_harness
Size-Type
Test Case 4
Train ID
Train Data.trainID
Train Data.lastBaliseID
Train Data.distanceFromLastBalise
Train Data.direction
Train Data.newMsg_PositionReport
Train Data.newMsg_MAAck
Train Parameters.trainID (1)
Train Parameters.trainID (2)
Train Parameters.trainID (3)
Balise Parameters.ID (1)
Balise Parameters.ID (2)
Balise Parameters.ID (3)
Balise Parameters.ID (4)
Balise Parameters.ID (5)
Balise Parameters.ID (6)
Balise Parameters.ID (7)
Balise Parameters.ID (8)
Balise Parameters.ID (9)
Balise Parameters.ID (10)
Balise Parameters.ID (11)
Balise Parameters.ID (12)
Balise Parameters.ID (13)
double
CheckTrainID
(SIL)
Train ID
Train Data
LastValidTrainData
LastValidTrainData
Ready
View 3 warnings
125%
FixedStepDiscrete

总结与问题



- 国内外高安全关键系统开发与验证的趋势
- 符合行业标准的基于模型设计解决方案
- 轨道交通行业符合EN50128标准项目实践