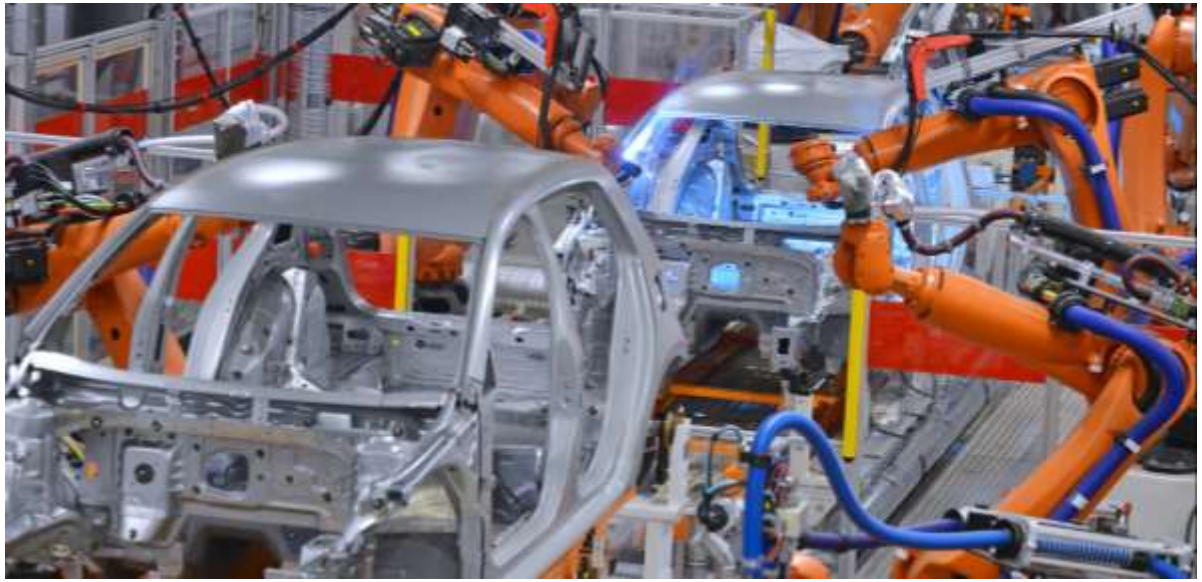
MATLAB EXPO

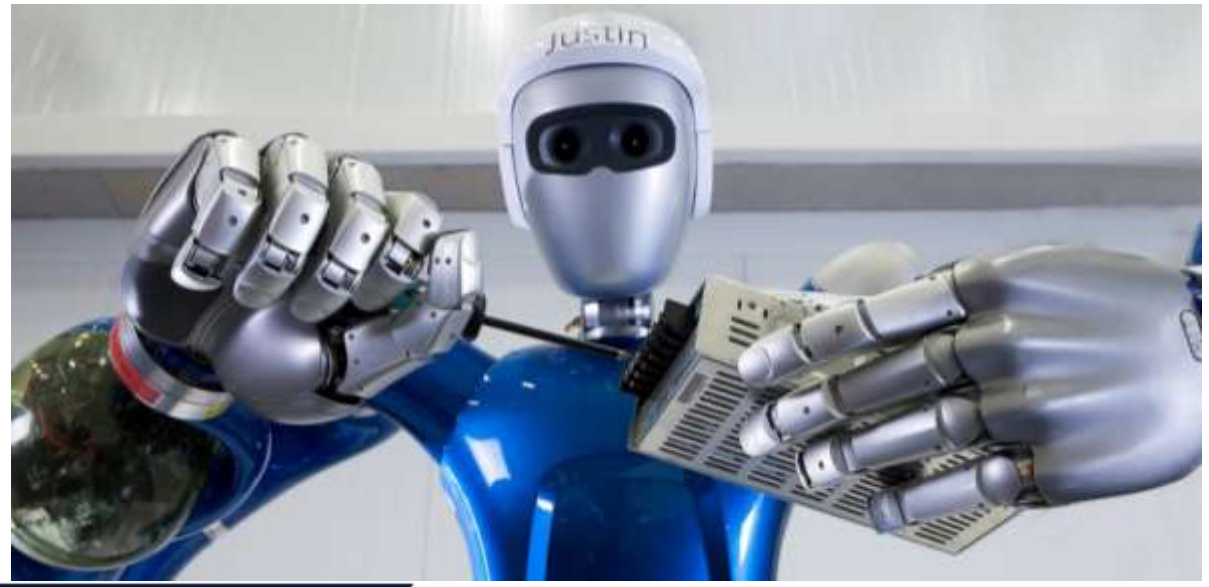
在基于模型的敏捷设计中引入测试驱动开发

杨超，MathWorks中国应用工程师

MathWorks

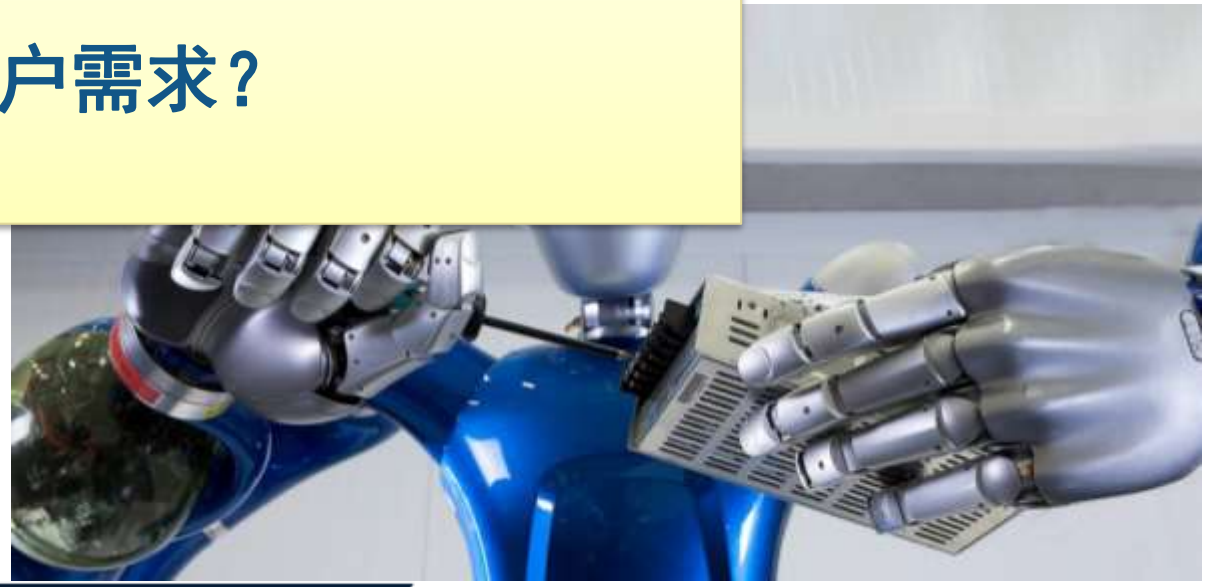
万物皆算法...





万物皆算法...

但是如何...
更快交付？
满足极具挑战的客户需求？
保证质量？

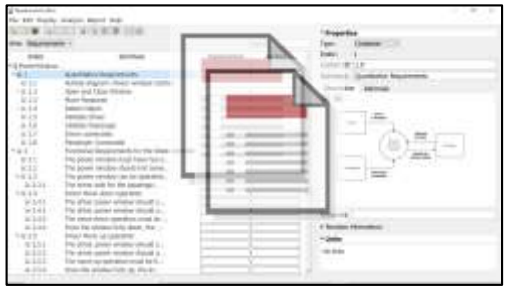


测试驱动的开发流程

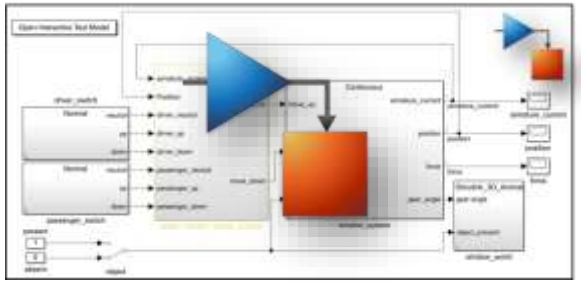
1. 创建测试
2. 实施最少的设计以通过测试
3. 重构

Simulink为测试驱动开发（TDD）提供集成框架

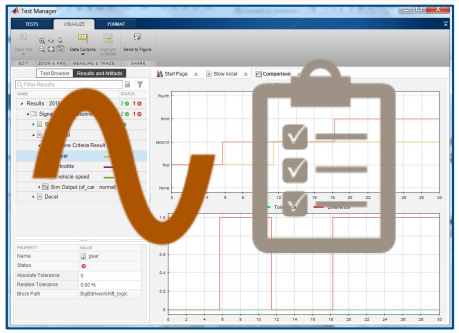
需求



实现



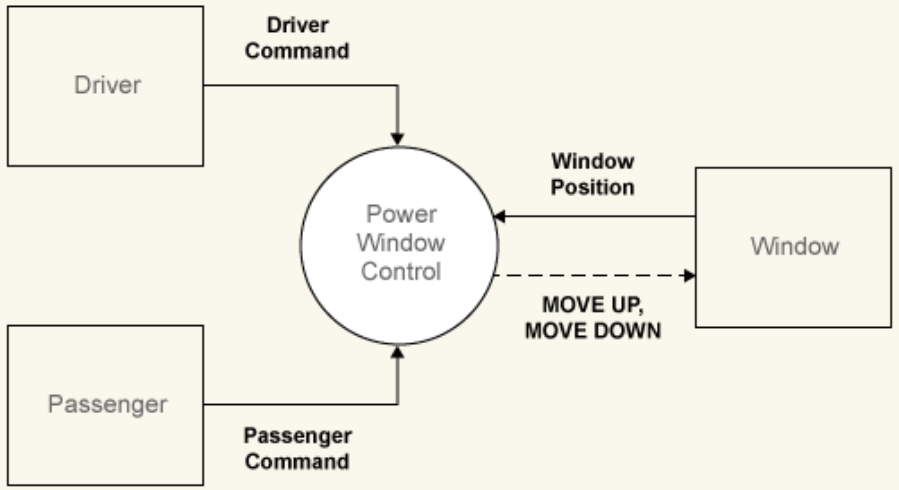
测试

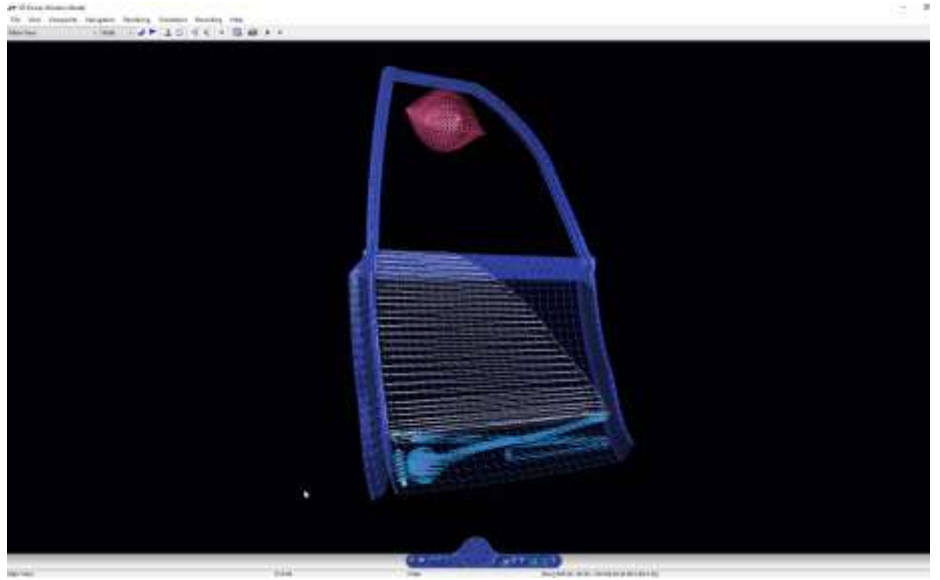


从客户的顶层需求出发

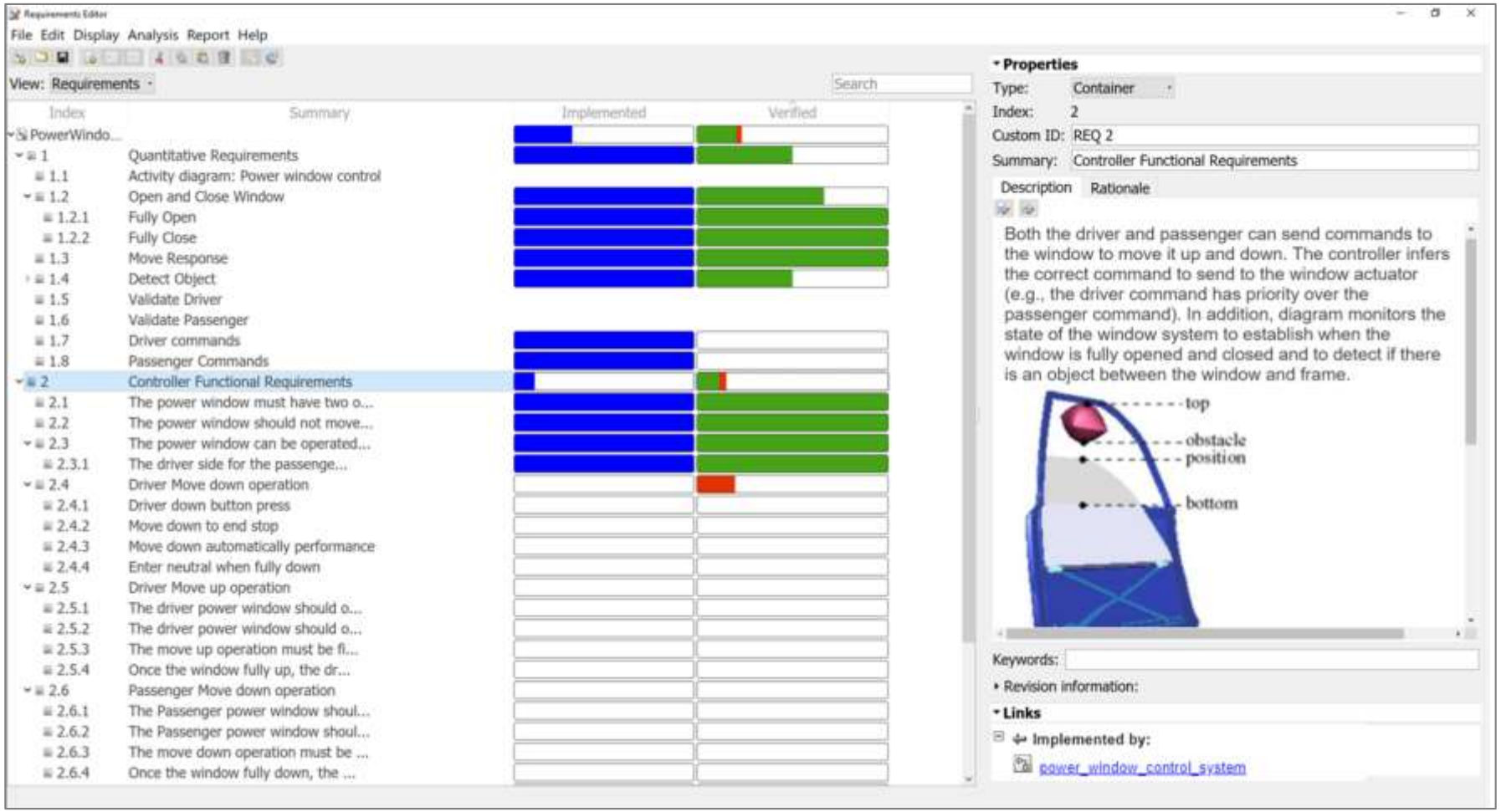
使用者需求:

- 驾驶员和乘客应都能够控制车窗
- 如果检测到阻碍，车窗应停止关闭
- 车窗应能够全开和全关



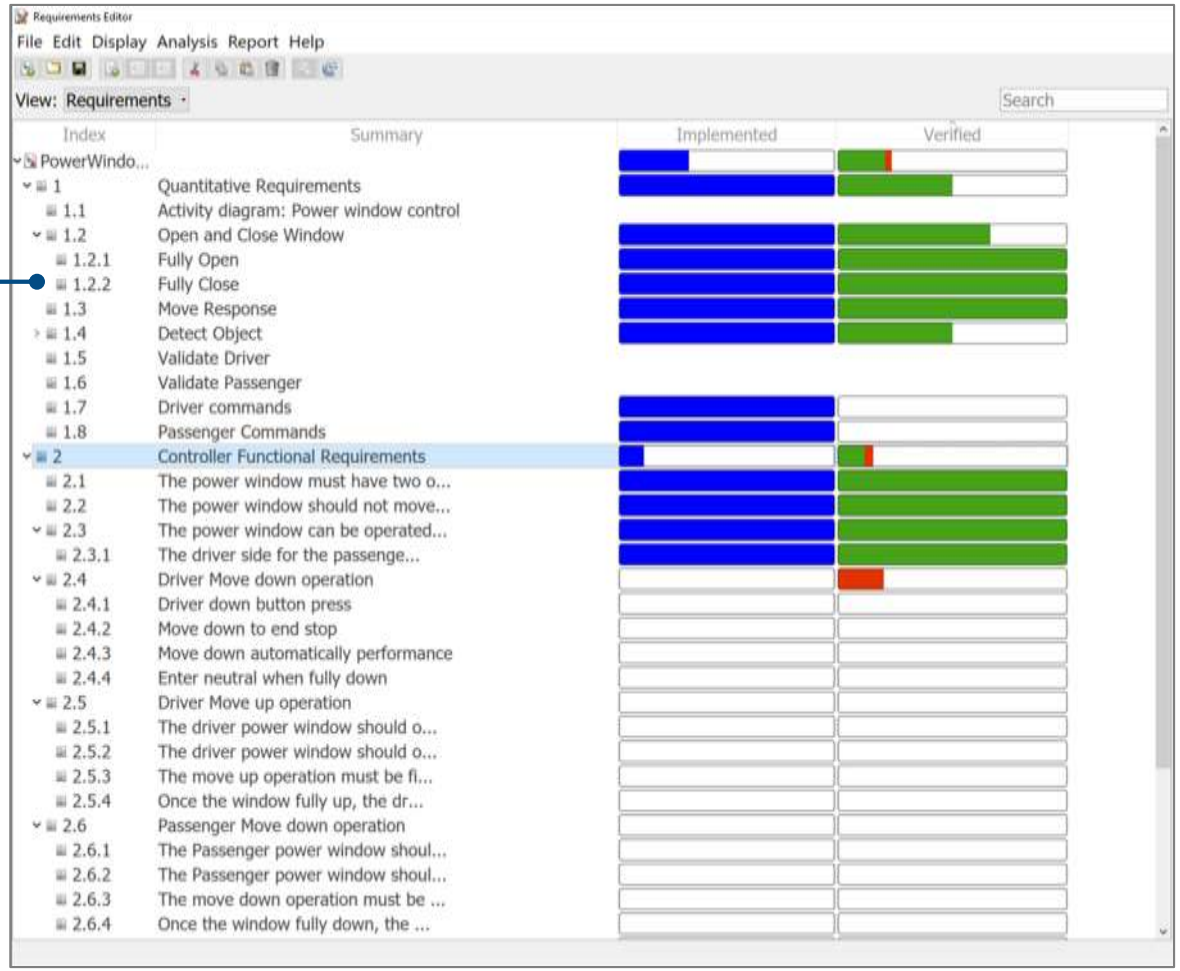


需求捕获

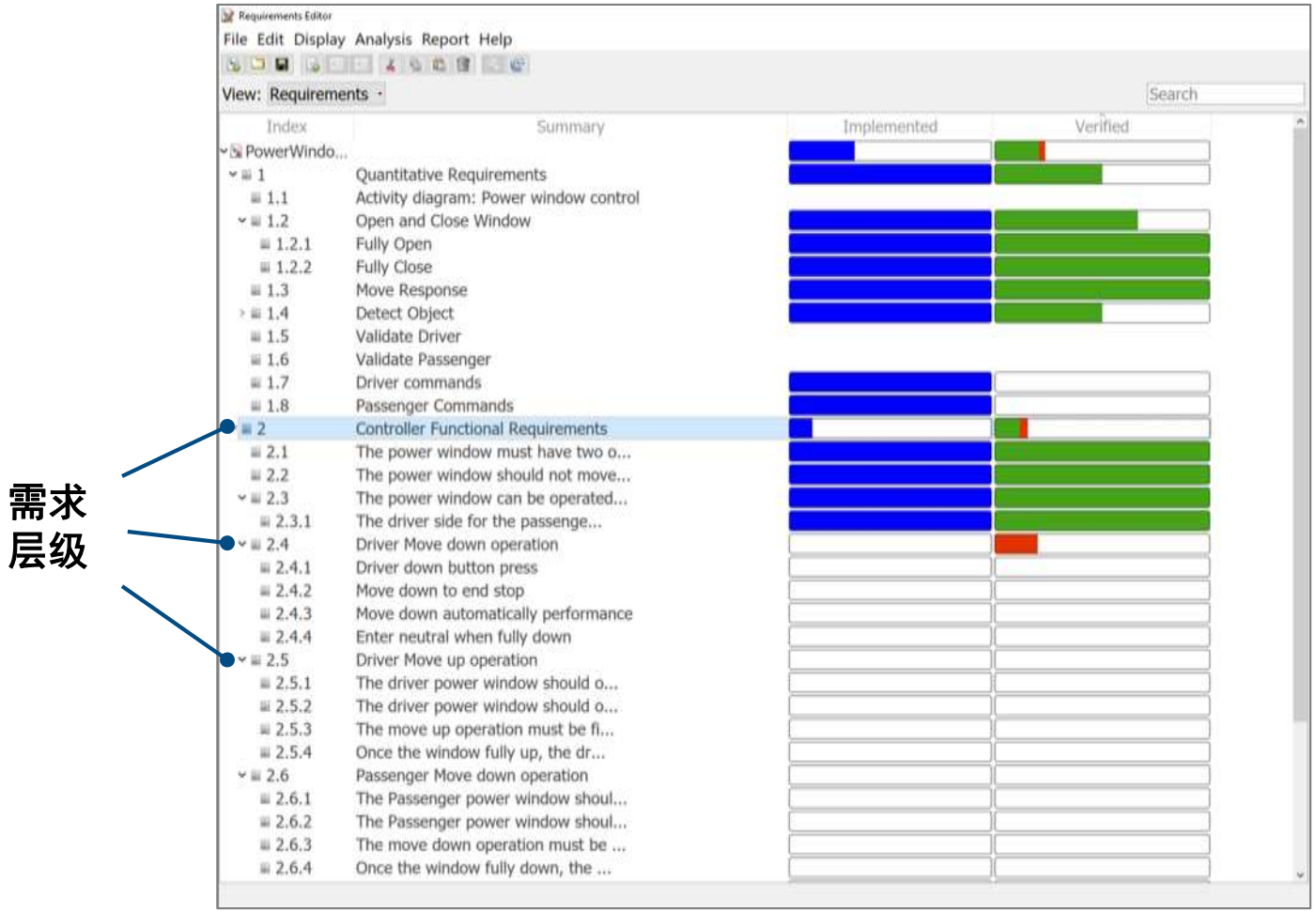

File Edit Display Analysis Report Help
View: Requirements
Search
Index
Summary
Implemented
Verified
PowerWindo...
1 Quantitative Requirements
1.1 Activity diagram: Power window control
1.2 Open and Close Window
1.2.1 Fully Open
1.2.2 Fully Close
1.3 Move Response
1.4 Detect Object
1.5 Validate Driver
1.6 Validate Passenger
1.7 Driver commands
1.8 Passenger Commands
2 Controller Functional Requirements
2.1 The power window must have two o...
2.2 The power window should not move...
2.3 The power window can be operated...
2.3.1 The driver side for the passenge...
2.4 Driver Move down operation
2.4.1 Driver down button press
2.4.2 Move down to end stop
2.4.3 Move down automatically performance
2.4.4 Enter neutral when fully down
2.5 Driver Move up operation
2.5.1 The driver power window should o...
2.5.2 The driver power window should o...
2.5.3 The move up operation must be fi...
2.5.4 Once the window fully up, the dr...
2.6 Passenger Move down operation
2.6.1 The Passenger power window shoul...
2.6.2 The Passenger power window shoul...
2.6.3 The move down operation must be ...
2.6.4 Once the window fully down, the ...
Properties
Type: Container
Index: 2
Custom ID: REQ 2
Summary: Controller Functional Requirements
Description
Rationale
Both the driver and passenger can send commands to the window to move it up and down. The controller infers the correct command to send to the window actuator (e.g., the driver command has priority over the passenger command). In addition, diagram monitors the state of the window system to establish when the window is fully opened and closed and to detect if there is an object between the window and frame.
top
obstacle position
bottom
Keywords:
Revision information:
Links
Implemented by:
power_window_control_system

详情查看

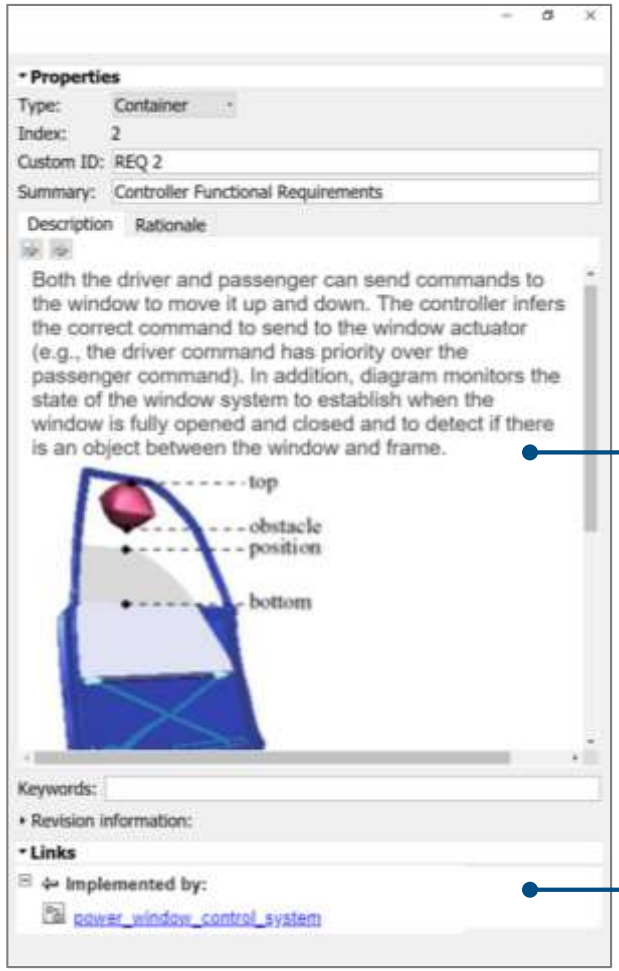
需求
详情


Requirements Editor
File Edit Display Analysis Report Help
View: Requirements
Search
Index
Summary
Implemented
Verified
PowerWindo...
1 Quantitative Requirements
1.1 Activity diagram: Power window control
1.2 Open and Close Window
1.2.1 Fully Open
1.2.2 Fully Close
1.3 Move Response
1.4 Detect Object
1.5 Validate Driver
1.6 Validate Passenger
1.7 Driver commands
1.8 Passenger Commands
2 Controller Functional Requirements
2.1 The power window must have two o...
2.2 The power window should not move...
2.3 The power window can be operated...
2.3.1 The driver side for the passenge...
2.4 Driver Move down operation
2.4.1 Driver down button press
2.4.2 Move down to end stop
2.4.3 Move down automatically performance
2.4.4 Enter neutral when fully down
2.5 Driver Move up operation
2.5.1 The driver power window should o...
2.5.2 The driver power window should o...
2.5.3 The move up operation must be fi...
2.5.4 Once the window fully up, the dr...
2.6 Passenger Move down operation
2.6.1 The Passenger power window shoul...
2.6.2 The Passenger power window shoul...
2.6.3 The move down operation must be ...
2.6.4 Once the window fully down, the ...

组织并创建需求层级


需求
层级
Requirements Editor
File Edit Display Analysis Report Help
View: Requirements
Search
Index
Summary
Implemented
Verified
PowerWindo...
1 Quantitative Requirements
1.1 Activity diagram: Power window control
1.2 Open and Close Window
1.2.1 Fully Open
1.2.2 Fully Close
1.3 Move Response
1.4 Detect Object
1.5 Validate Driver
1.6 Validate Passenger
1.7 Driver commands
1.8 Passenger Commands
2 Controller Functional Requirements
2.1 The power window must have two o...
2.2 The power window should not move...
2.3 The power window can be operated...
2.3.1 The driver side for the passenge...
2.4 Driver Move down operation
2.4.1 Driver down button press
2.4.2 Move down to end stop
2.4.3 Move down automatically performance
2.4.4 Enter neutral when fully down
2.5 Driver Move up operation
2.5.1 The driver power window should o...
2.5.2 The driver power window should o...
2.5.3 The move up operation must be fi...
2.5.4 Once the window fully up, the dr...
2.6 Passenger Move down operation
2.6.1 The Passenger power window shoul...
2.6.2 The Passenger power window shoul...
2.6.3 The move down operation must be ...
2.6.4 Once the window fully down, the ...

详细需求定义



描述 / 解释

链接
看板

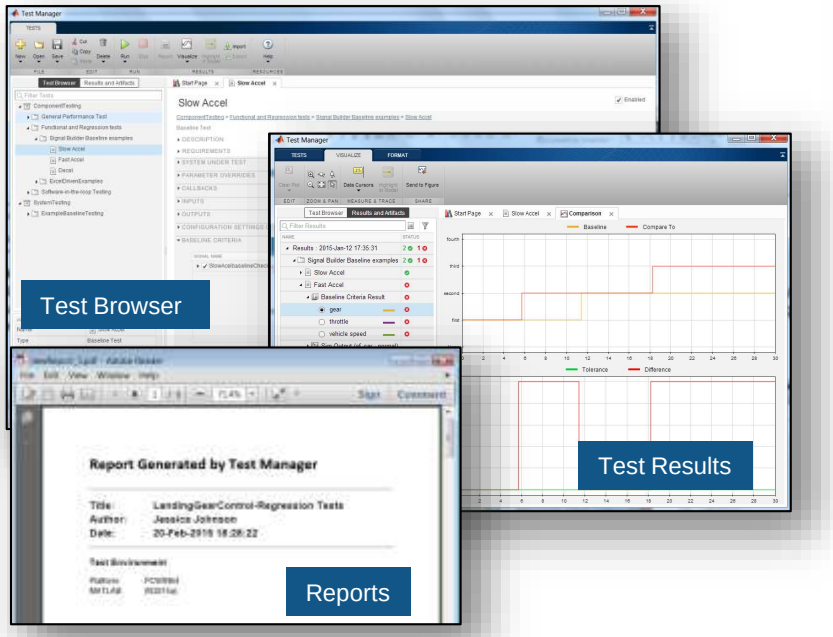
1. **创建测试**
2. 实施最少的设计以通过测试
3. 重构

创建, 管理, 并执行基于仿真的测试

Simulink Test

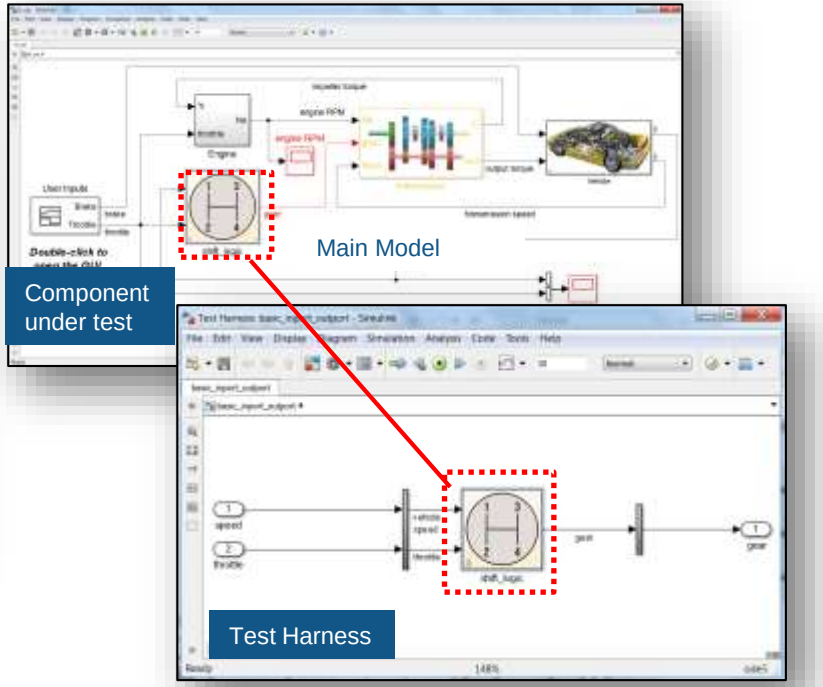
Test Manager

- 创建, 管理, 组织测试



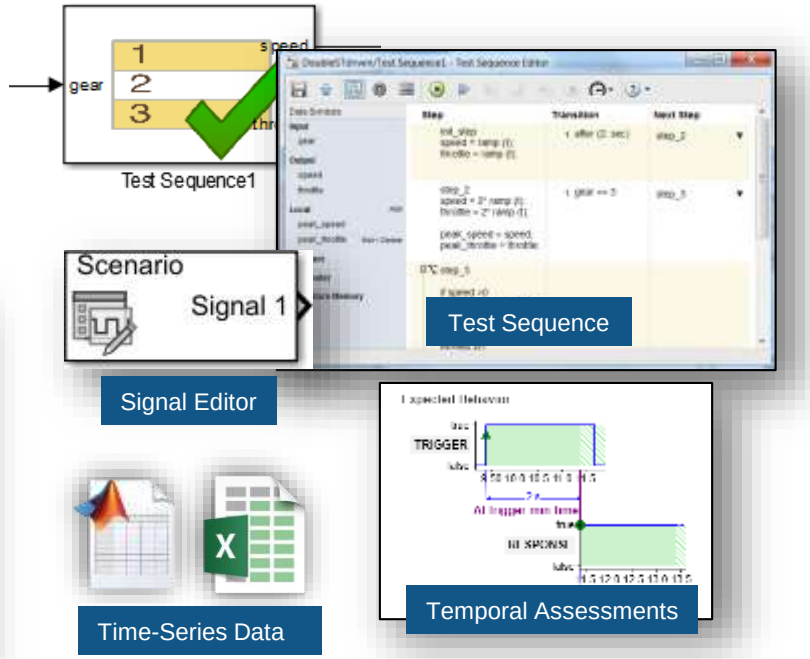
Test Harnesses

- 识别测试组件

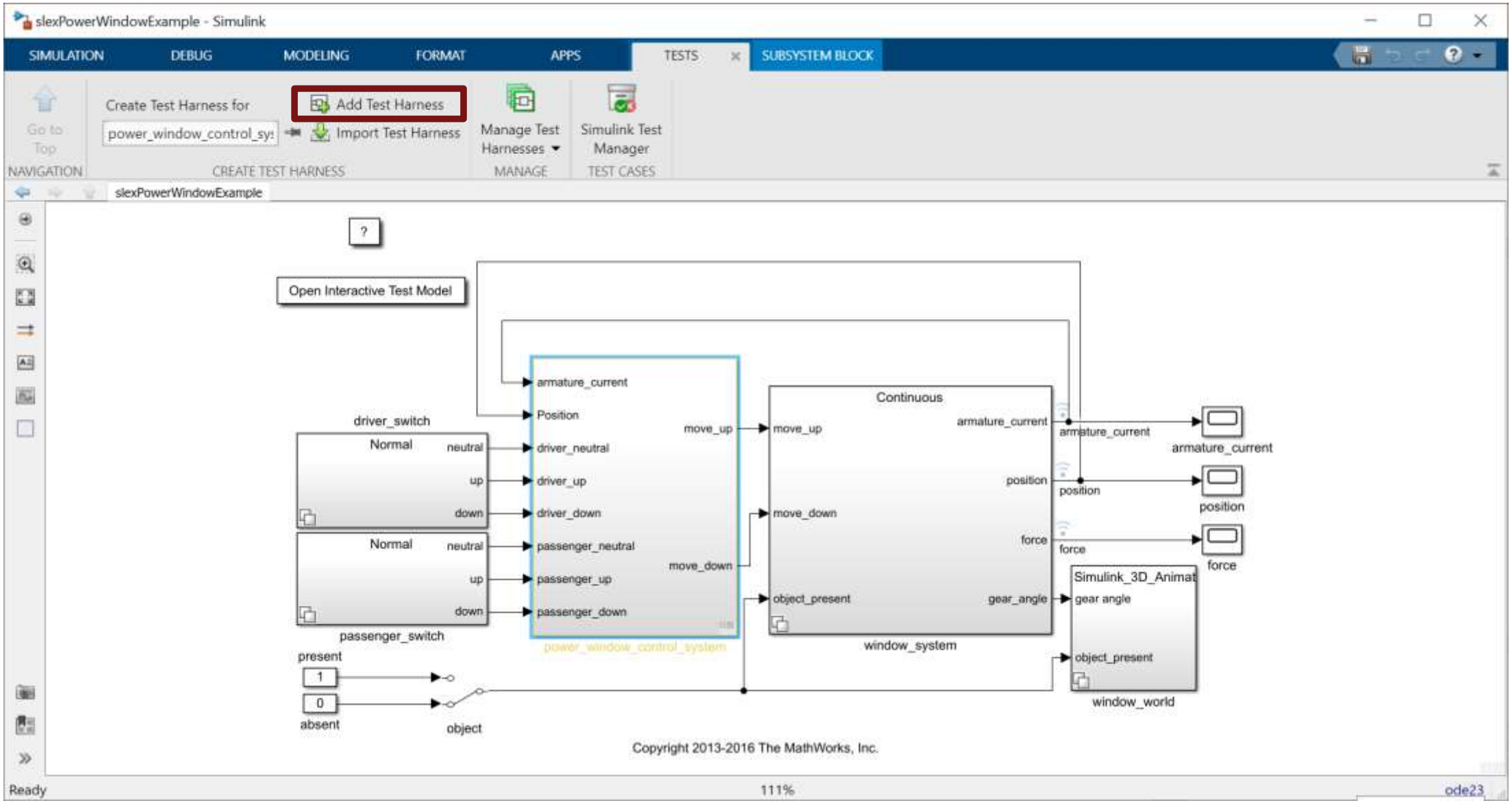


Test Authoring

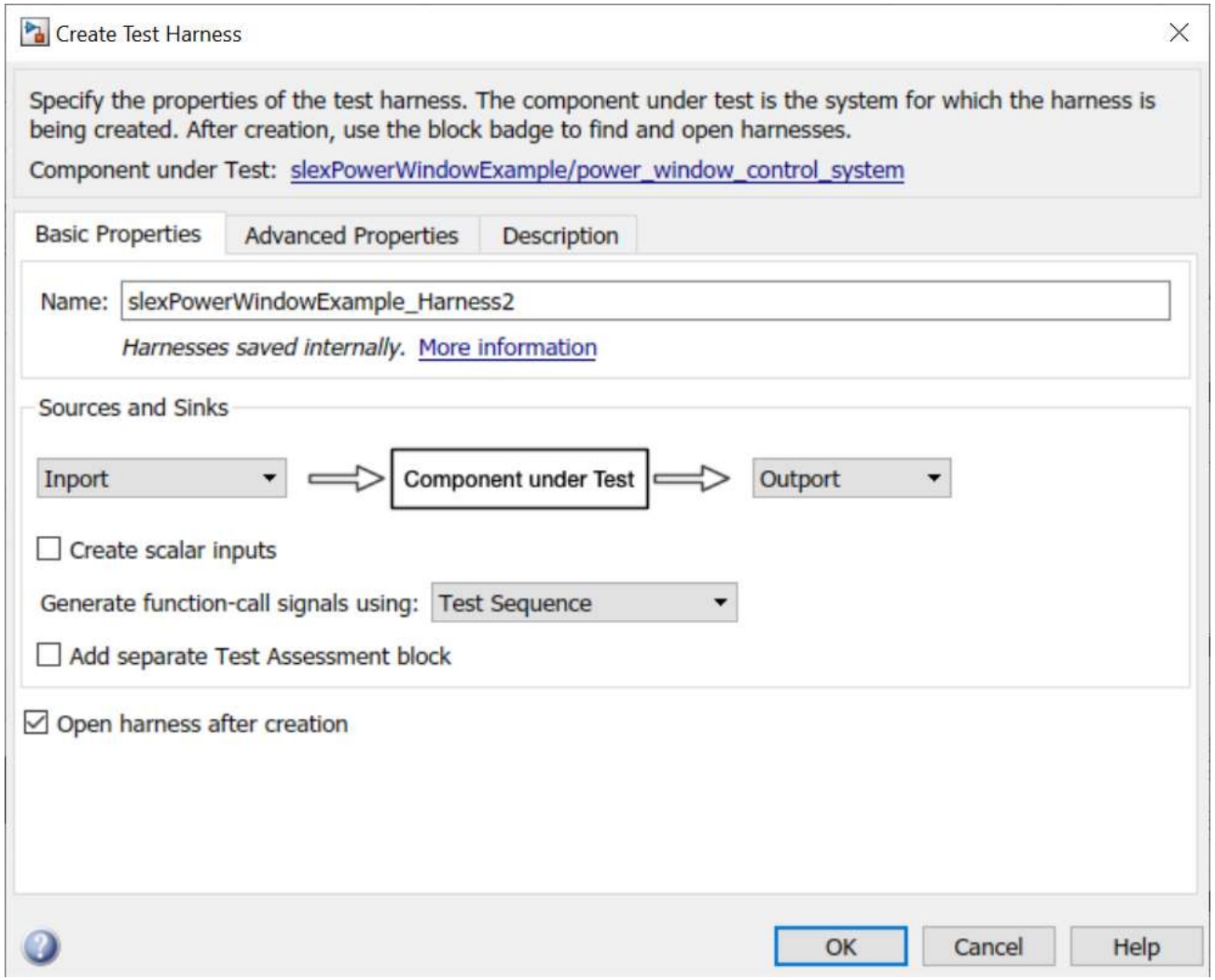
- 定义测试的输入、期望输出和偏差容限



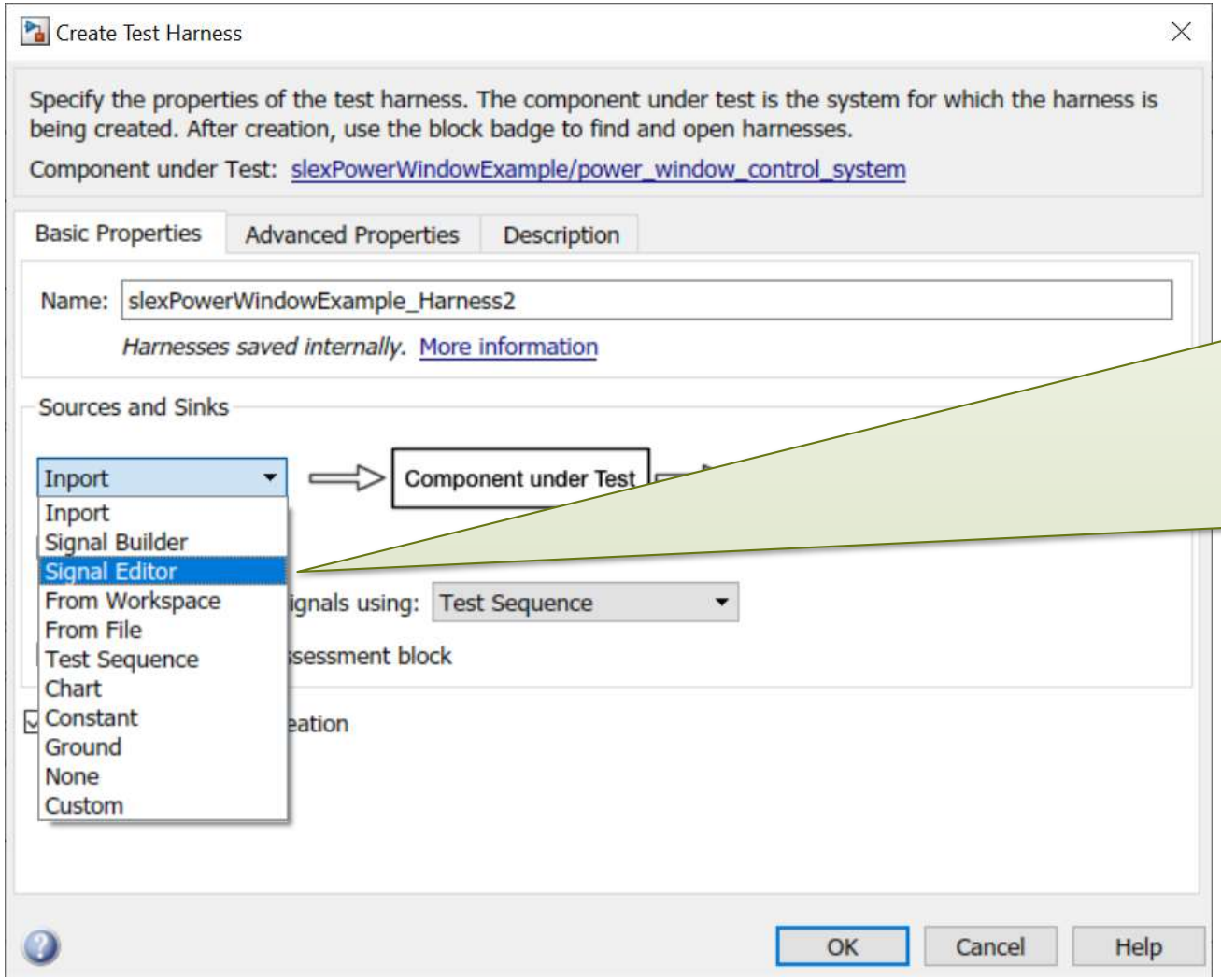
创建测试框架（Test Harness）以识别待测试的组件

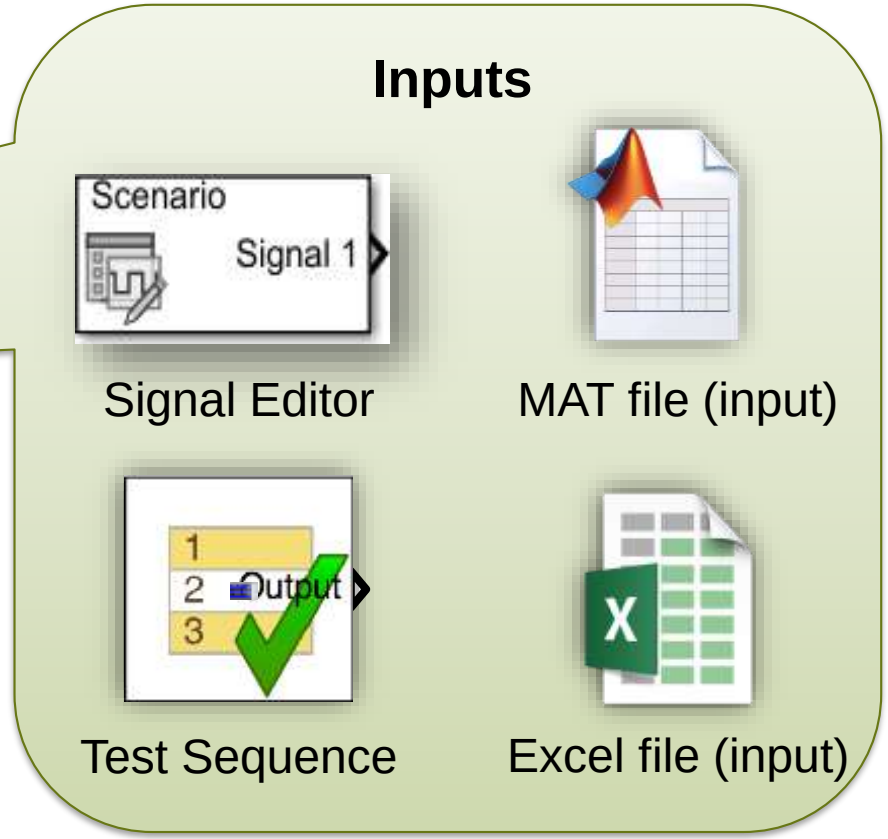


定义测试框架属性

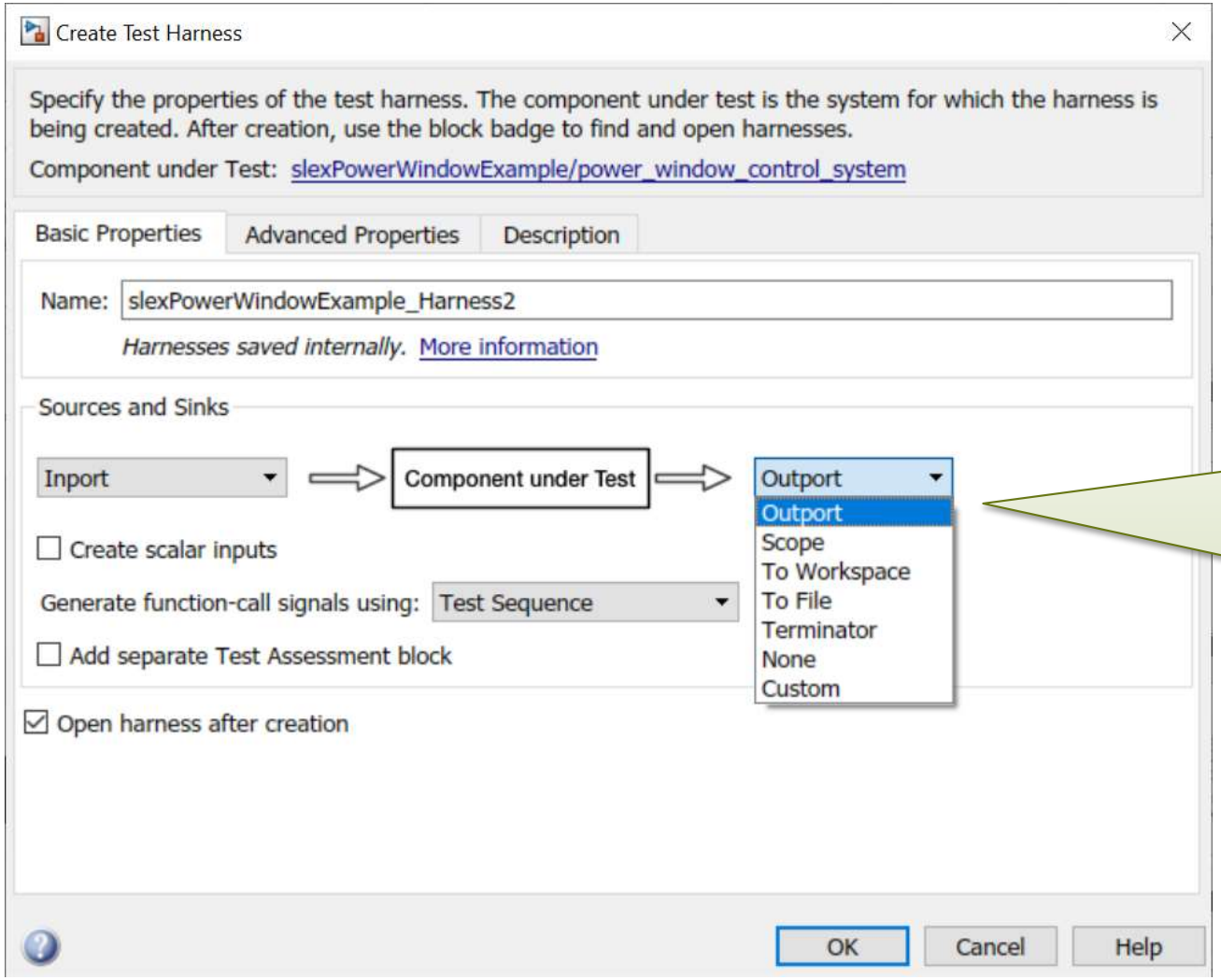

Create Test Harness
Specify the properties of the test harness. The component under test is the system for which the harness is being created. After creation, use the block badge to find and open harnesses.
Component under Test: slexPowerWindowExample/power_window_control_system
Basic Properties
Advanced Properties
Description
Name: slexPowerWindowExample_Harness2
Harnesses saved internally. More information
Sources and Sinks
Inport
Component under Test
Outport
Create scalar inputs
Generate function-call signals using: Test Sequence
Add separate Test Assessment block
Open harness after creation
OK
Cancel
Help

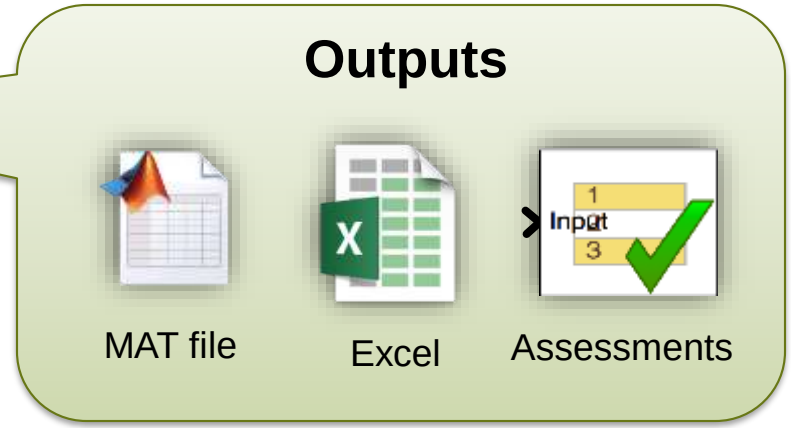
定义输入


Create Test Harness
Specify the properties of the test harness. The component under test is the system for which the harness is being created. After creation, use the block badge to find and open harnesses.
Component under Test: slexPowerWindowExample/power_window_control_system
Basic Properties
Advanced Properties
Description
Name: slexPowerWindowExample_Harness2
Harnesses saved internally. More information
Sources and Sinks
Inport
Component under Test
Inport
Signal Builder
Signal Editor
From Workspace
From File
Test Sequence
Chart
Constant
Ground
None
Custom
ignals using: Test Sequence
sessment block
eation
OK
Cancel
Help

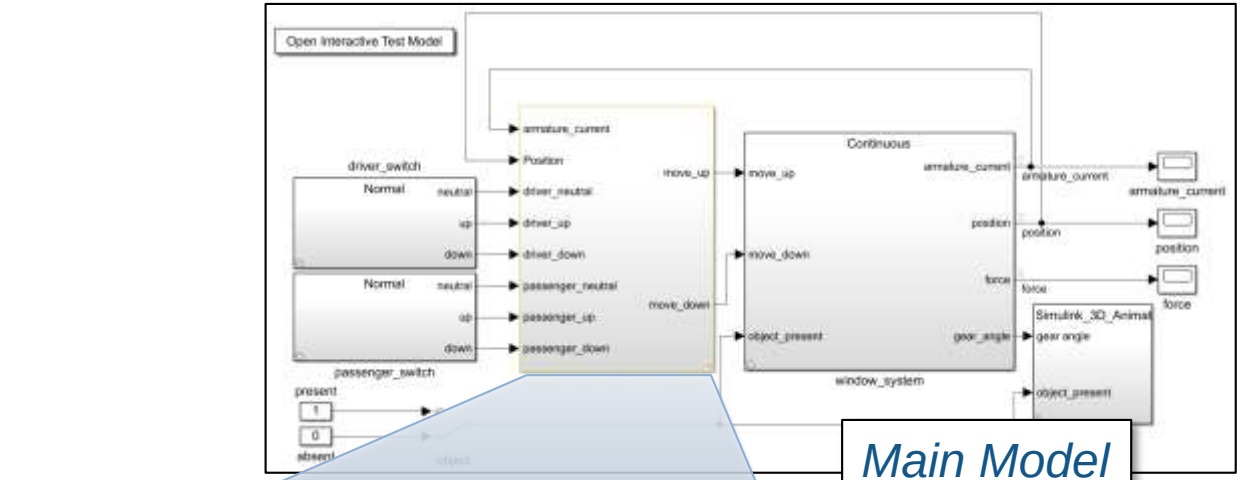

Inputs
Scenario
Signal 1
Signal Editor
MAT file (input)
1
2
3
Output
Test Sequence
Excel file (input)

定义输出

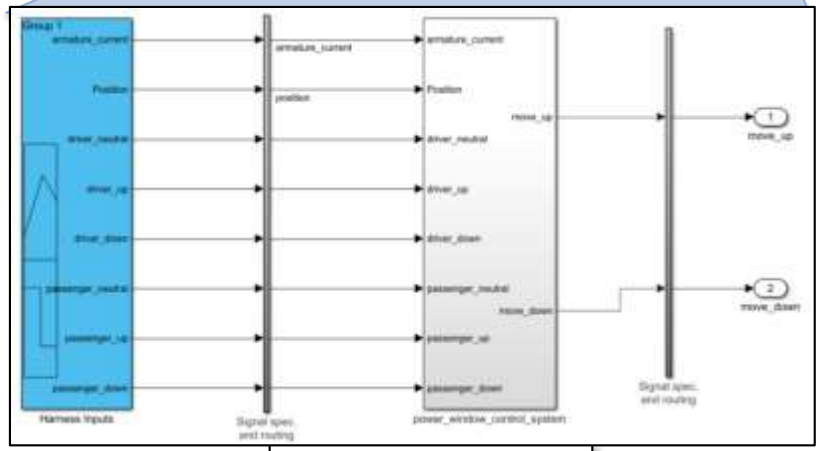

Create Test Harness
Specify the properties of the test harness. The component under test is the system for which the harness is being created. After creation, use the block badge to find and open harnesses.
Component under Test: slexPowerWindowExample/power_window_control_system
Basic Properties
Advanced Properties
Description
Name: slexPowerWindowExample_Harness2
Harnesses saved internally. More information
Sources and Sinks
Inport
Component under Test
Outport
Outport
Scope
To Workspace
To File
Terminator
None
Custom
Create scalar inputs
Generate function-call signals using: Test Sequence
Add separate Test Assessment block
Open harness after creation
OK
Cancel
Help


Outputs
MAT file
Excel
Assessments

创建测试框架以识别待测试组件

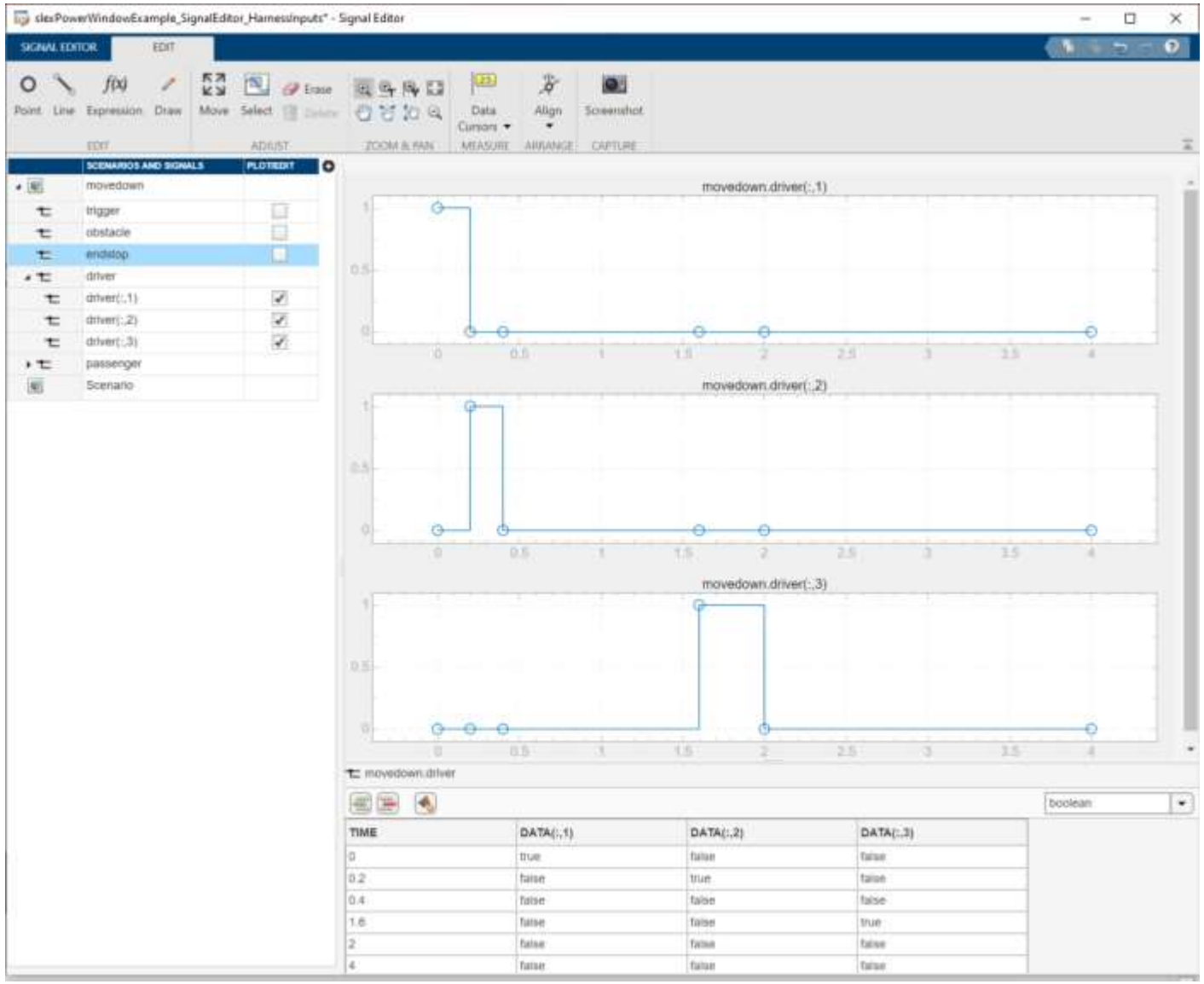


Main Model

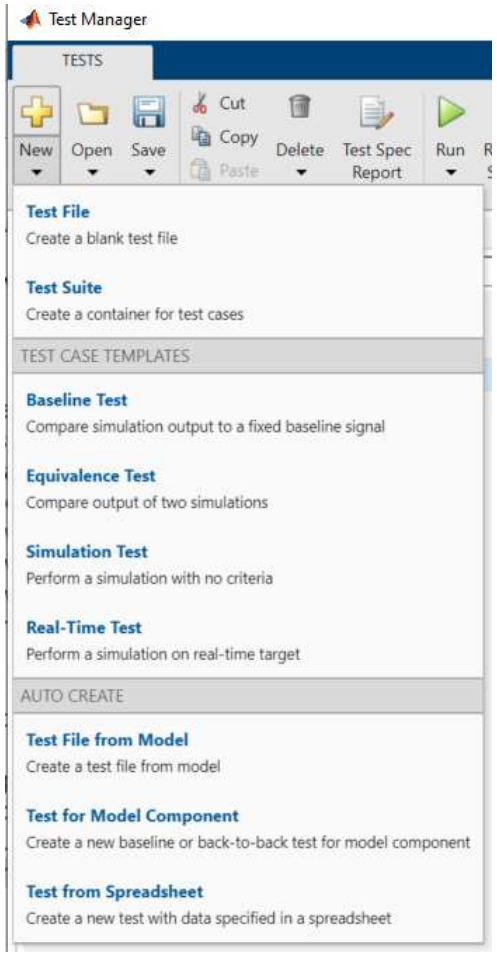


Test Harness

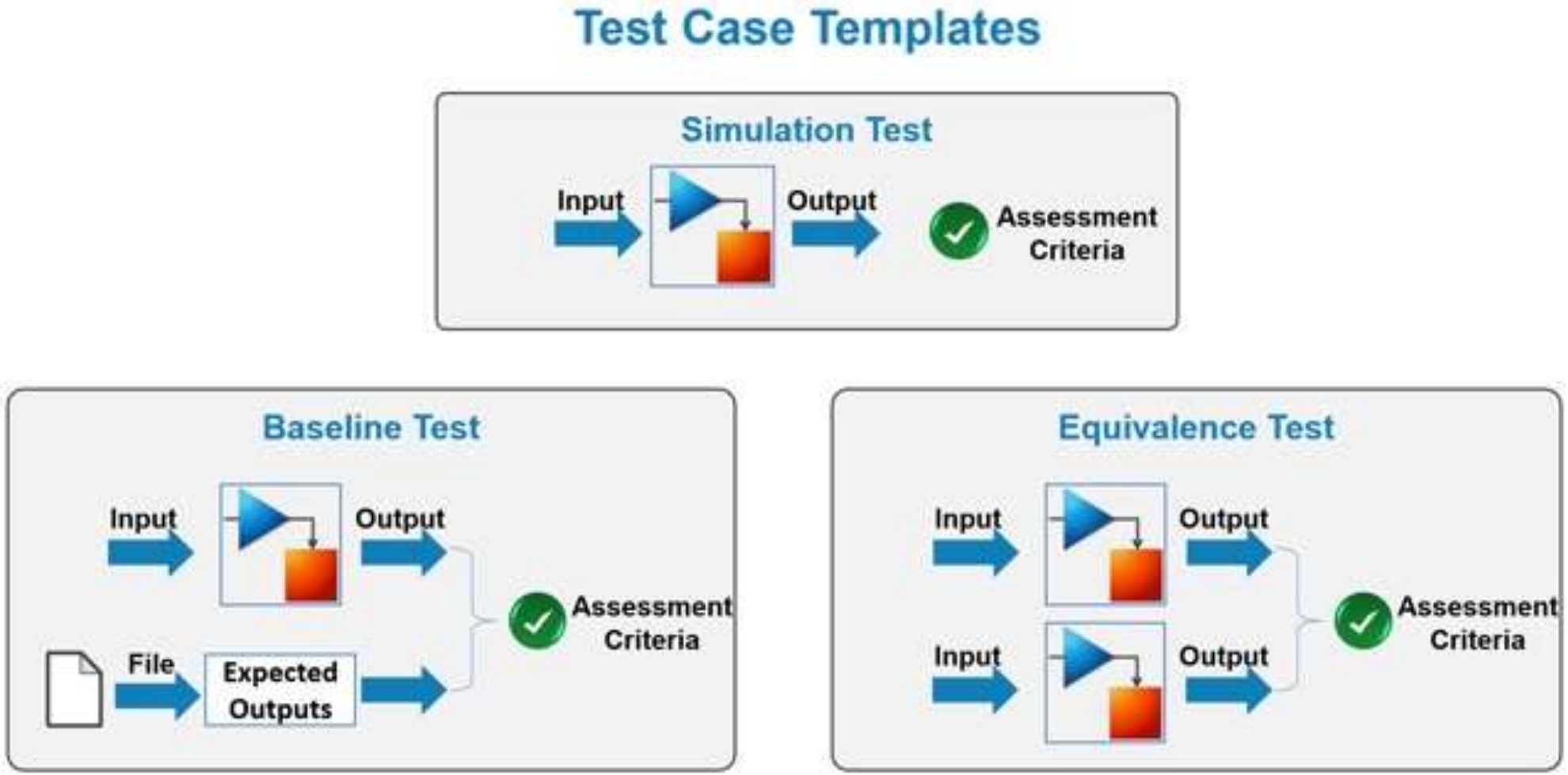
利用Signal Editor编写测试输入



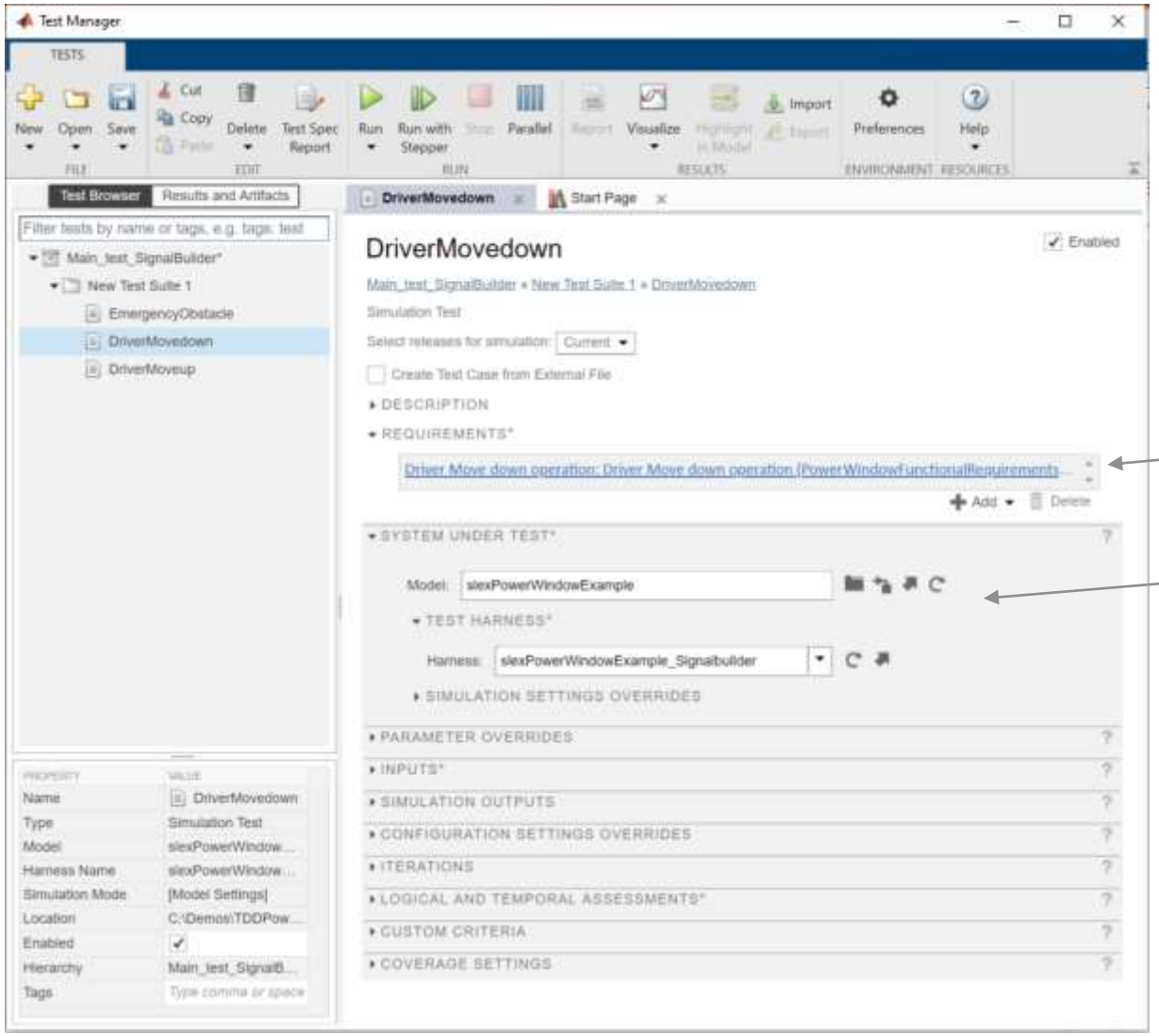
使用模板和向导（wizards）自动创建测试用例



使用模板和向导（wizards）自动创建测试用例



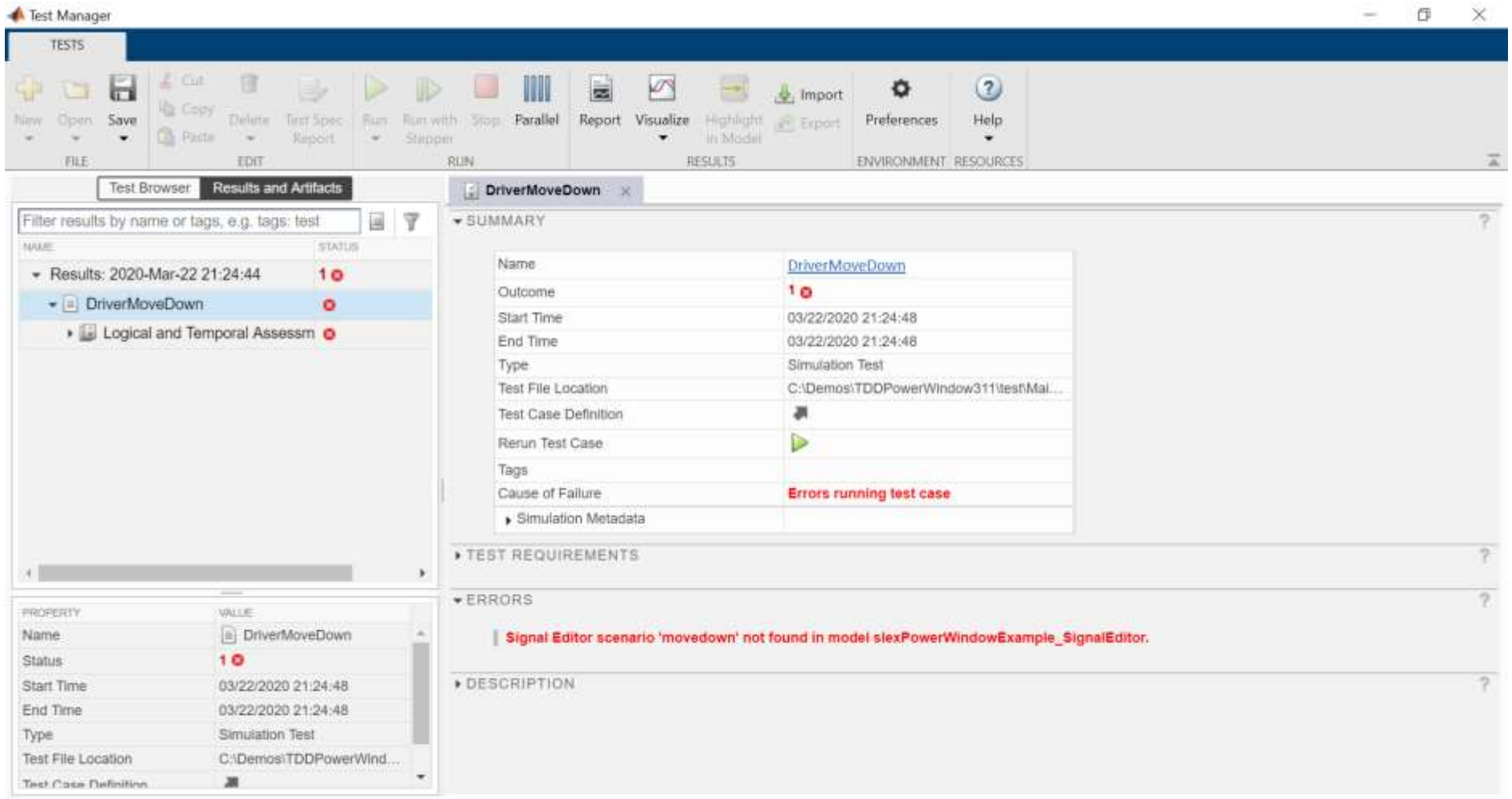
创建测试用例并关联到需求



关联到需求

定义要测试的模型

编译错误导致的测试失败

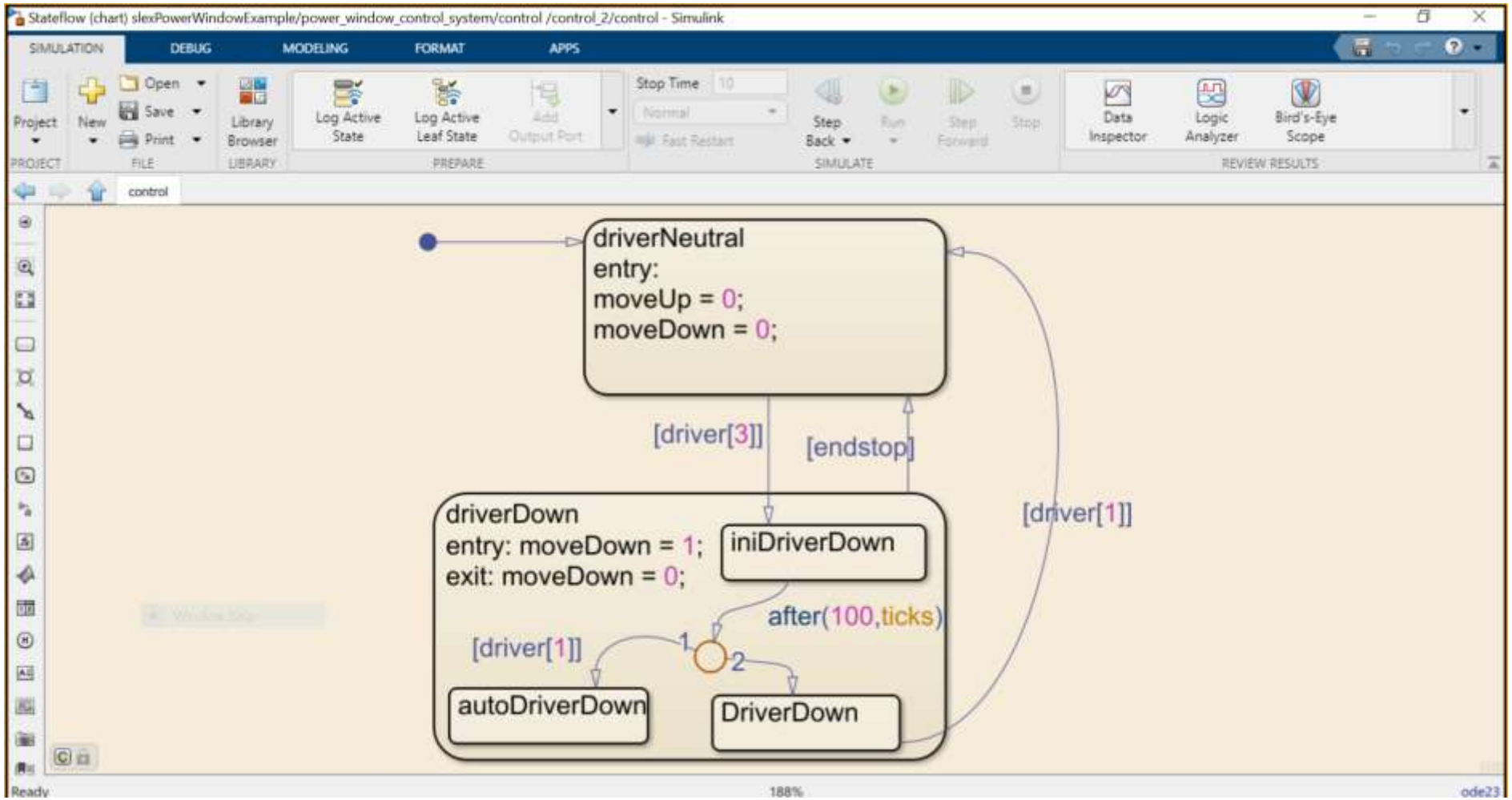


1. 创建测试

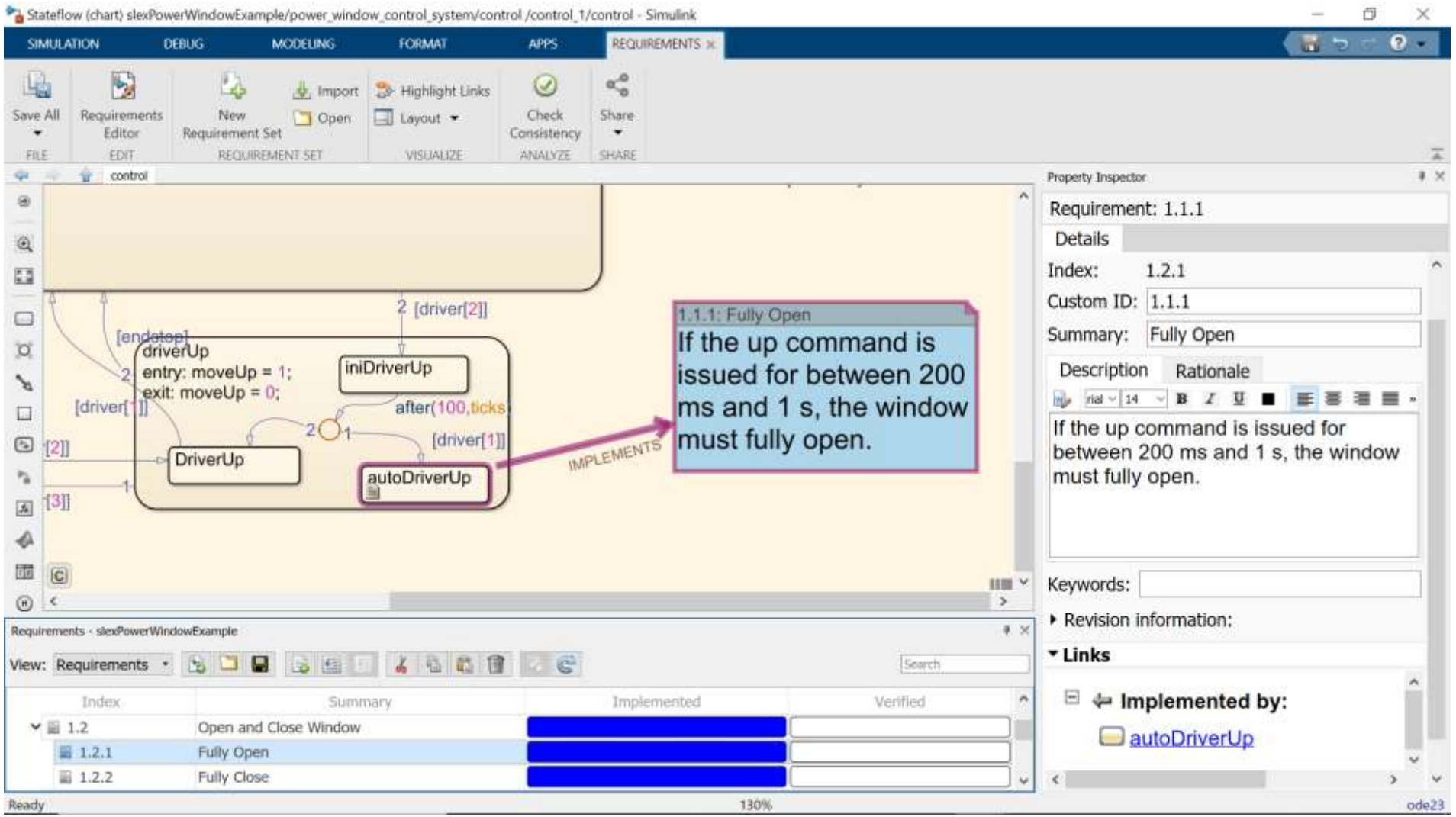
2. **实施最少的设计以通过测试**

3. 重构

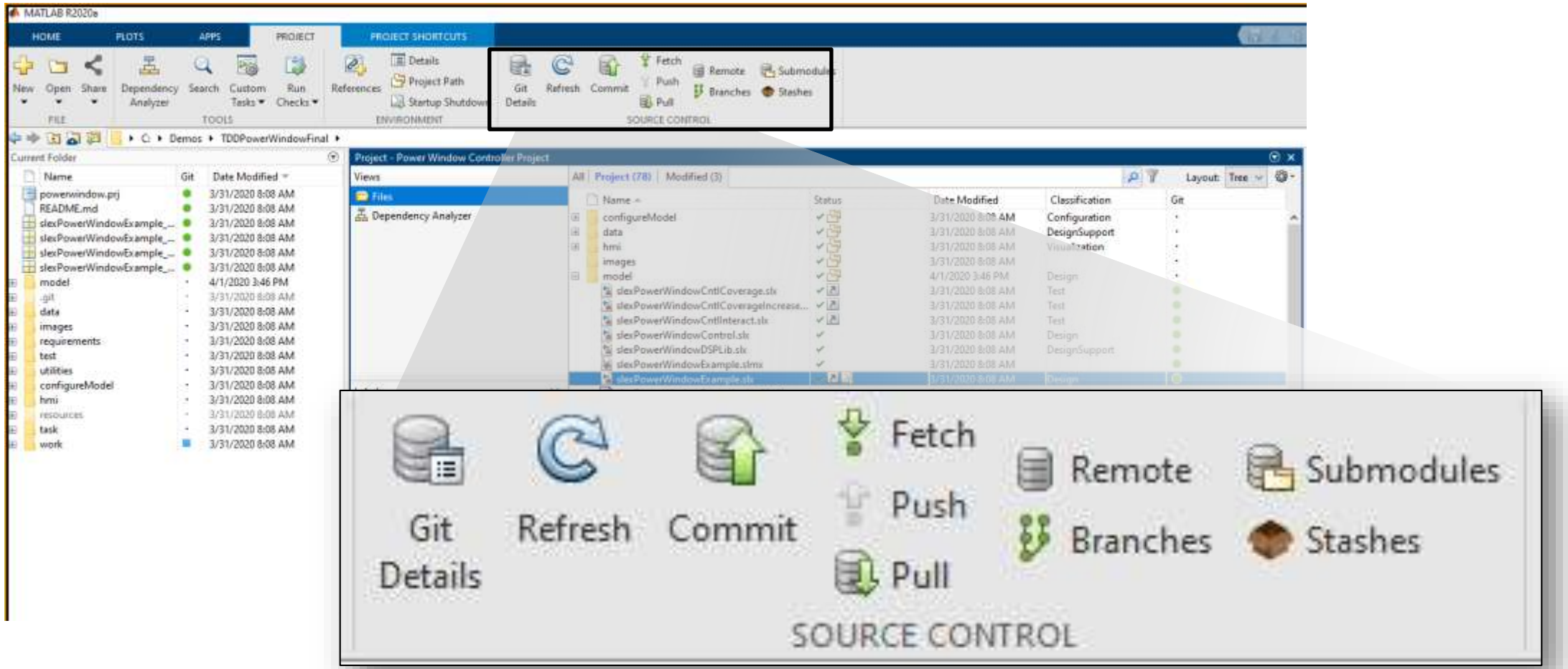
给出仅足以通过测试的实现



将实现关联到需求

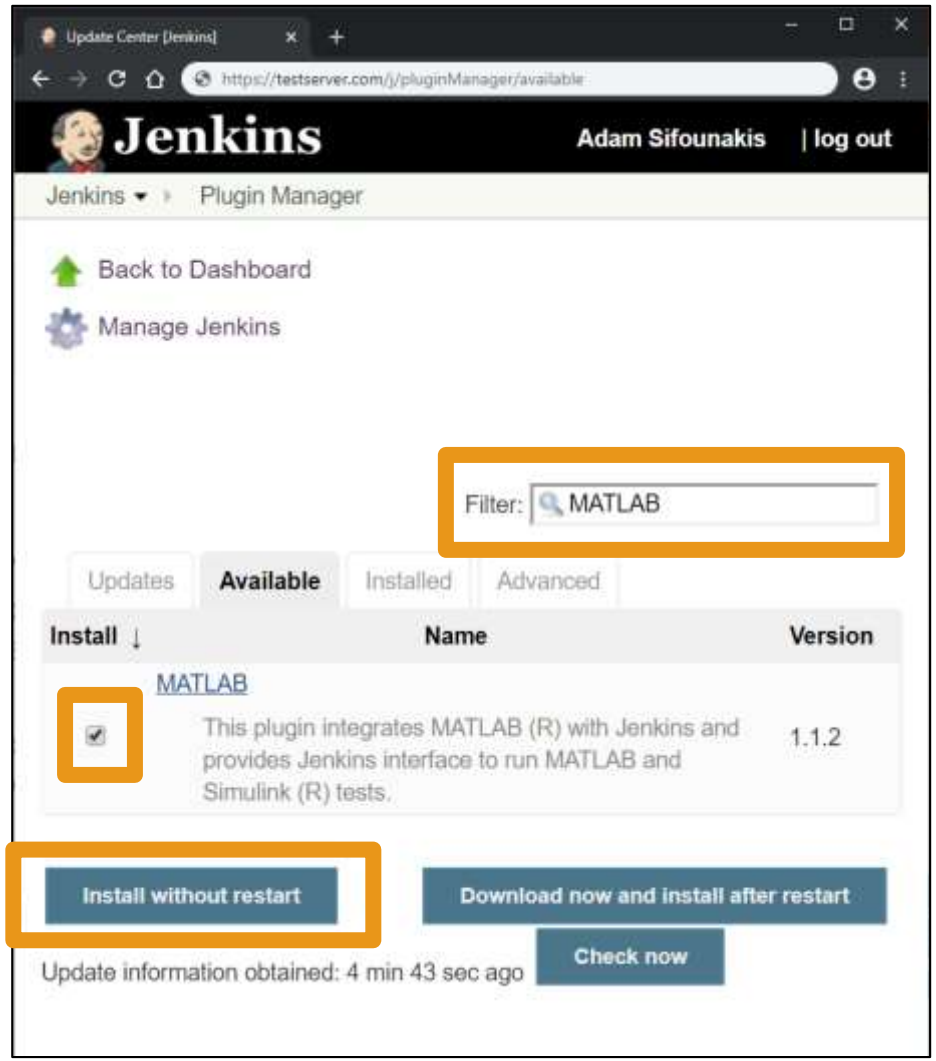

Stateflow (chart) slexPowerWindowExample/power_window_control_system/control /control_1/control - Simulink
SIMULATION
DEBUG
MODELING
FORMAT
APPS
REQUIREMENTS
Save All
Requirements Editor
New Requirement Set
Import
Open
Highlight Links
Layout
Check Consistency
Share
FILE
EDIT
REQUIREMENT SET
VISUALIZE
ANALYZE
SHARE
control
2 [driver[2]]
[endstep]
driverUp
entry: moveUp = 1;
exit: moveUp = 0;
iniDriverUp
after(100,ticks)
[driver[1]]
[2]]
DriverUp
autoDriverUp
[3]]
IMPLEMENTS
1.1.1: Fully Open
If the up command is issued for between 200 ms and 1 s, the window must fully open.
Property Inspector
Requirement: 1.1.1
Details
Index: 1.2.1
Custom ID: 1.1.1
Summary: Fully Open
Description
Rationale
If the up command is issued for between 200 ms and 1 s, the window must fully open.
Keywords:
Revision information:
Links
Implemented by:
autoDriverUp
Requirements - slexPowerWindowExample
View: Requirements
Search
Index
Summary
Implemented
Verified
1.2
Open and Close Window
1.2.1
Fully Open
1.2.2
Fully Close
Ready
130%
ode23

使用源代码管理直接从项目（Project）中管理工件



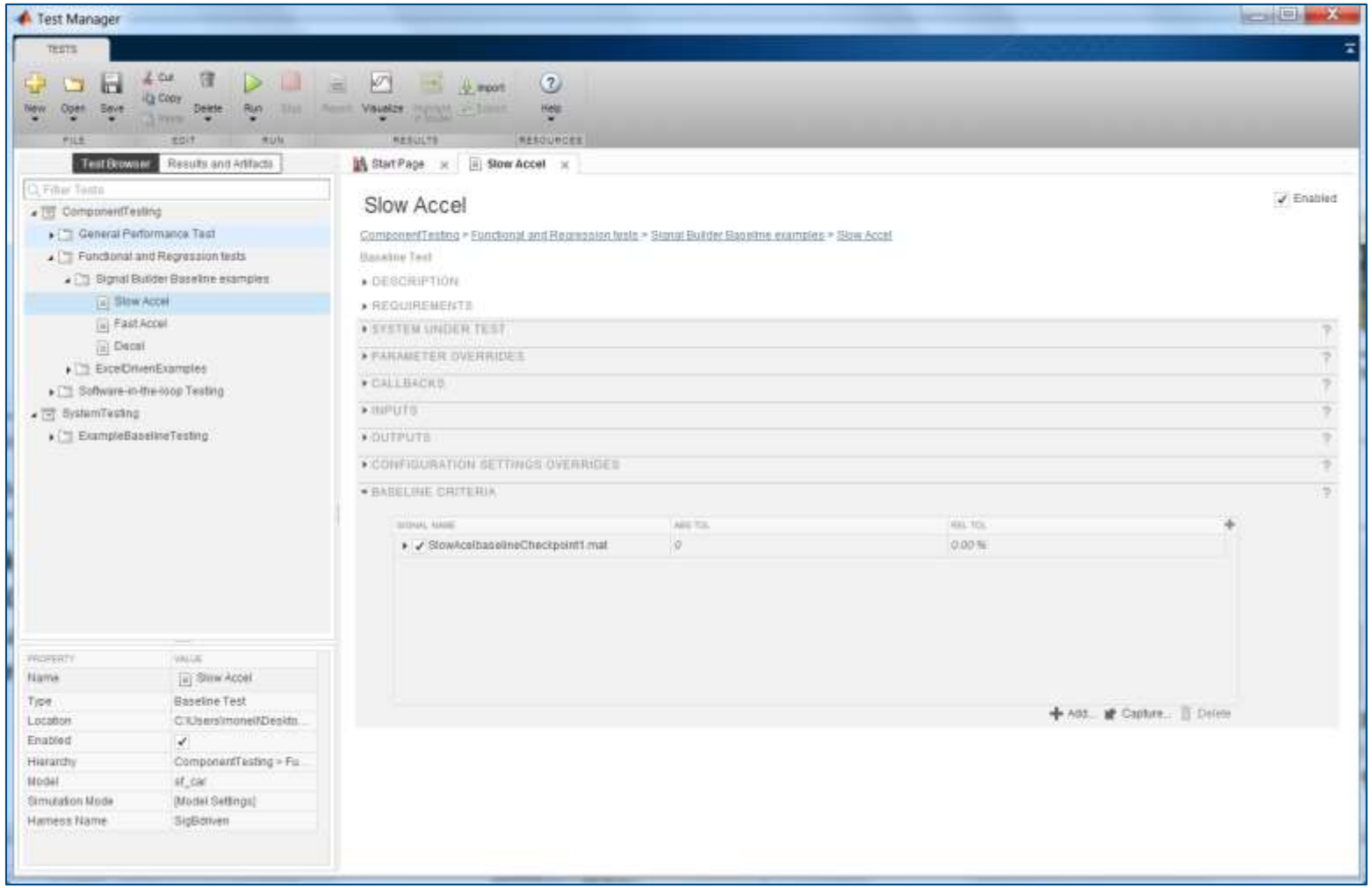
利用持续集成扩展测试并自动化测试

- 规划自动化的代码和模型测试
- 在Jenkins中安装MATLAB插件



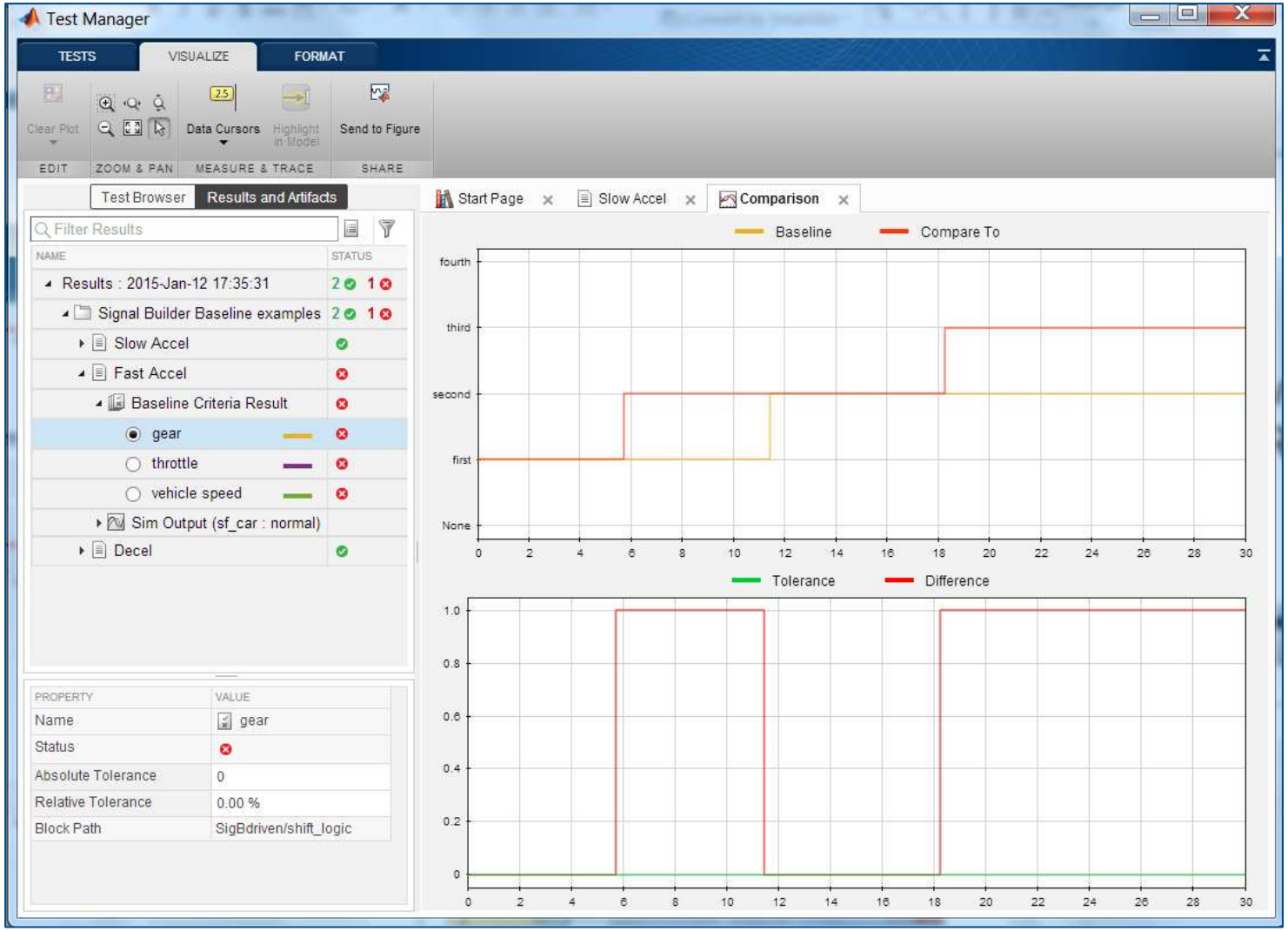
利用Test Manager执行测试

- 测试用例与文件分组
- 单独或批量执行



利用Test Manager分析和调试测试结果

- 结果摘要查看
- 使用Simulation Data Inspector排查调试
- 将结果归档，导出，或生成报告



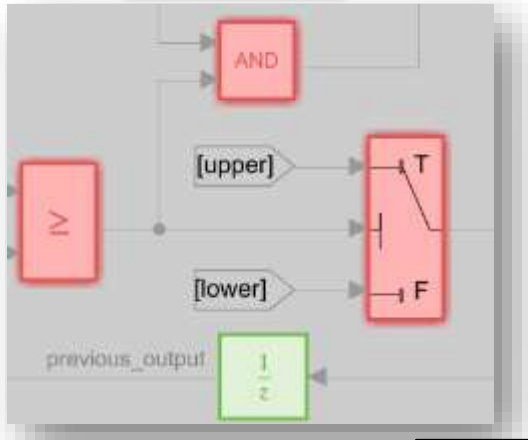
执行所有测试直至全部通过

PowerWindowControlUnitTest_sig	6 ✓
DriverMoveDown	✓
DriverMoveUp	✓
EmergencyObstacle	✓
Detect Object	✓
Fully Close Window	✓
PassengerMovedown	✓

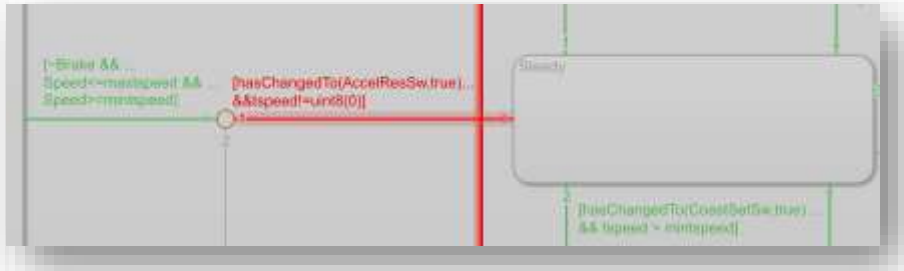
使用覆盖度衡量测试完备性

- 测试缺陷
- 需求缺失
- 非预期功能
- 设计错误

Simulink

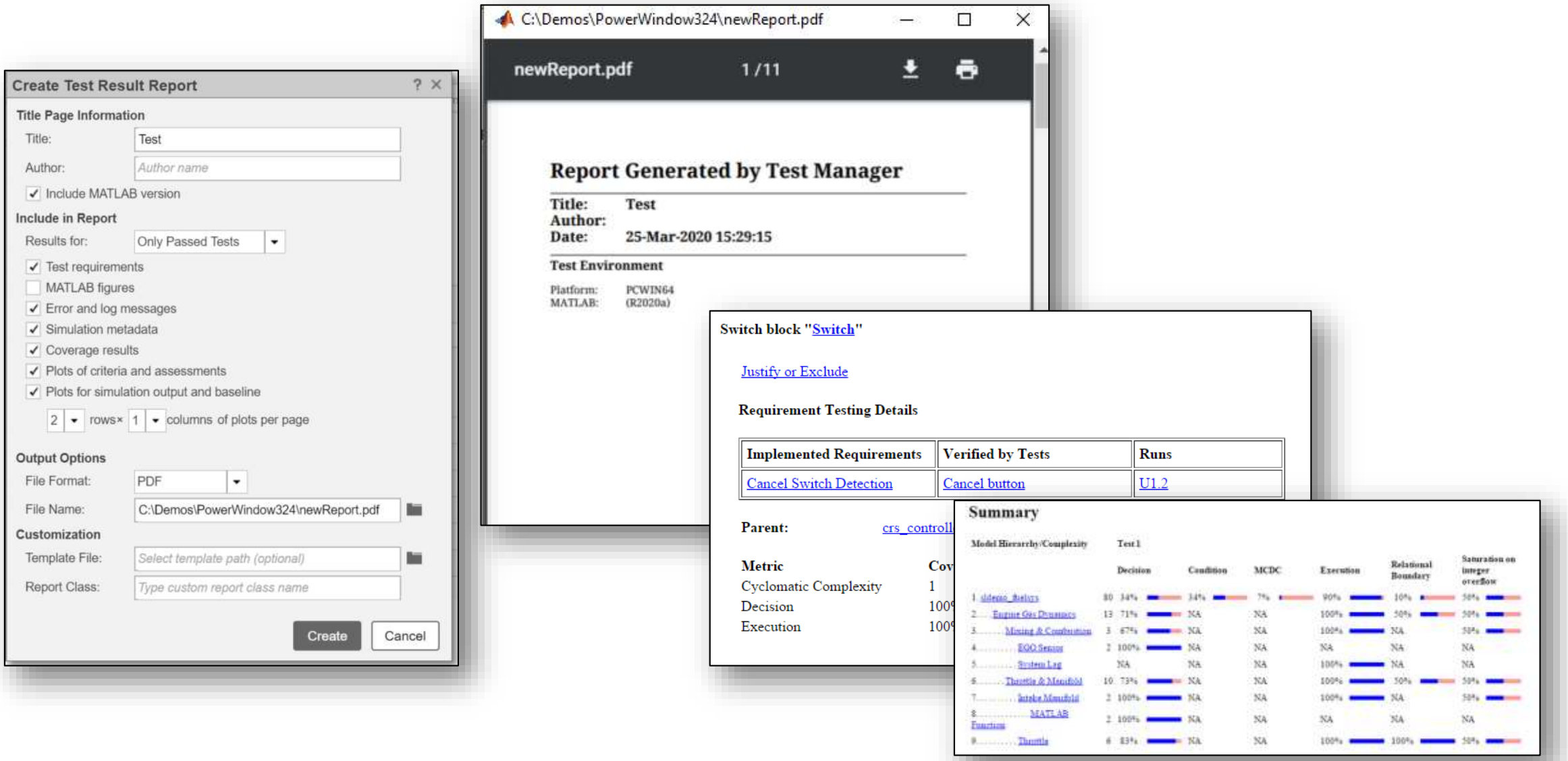


Stateflow



Code

生成测试报告以用于评审



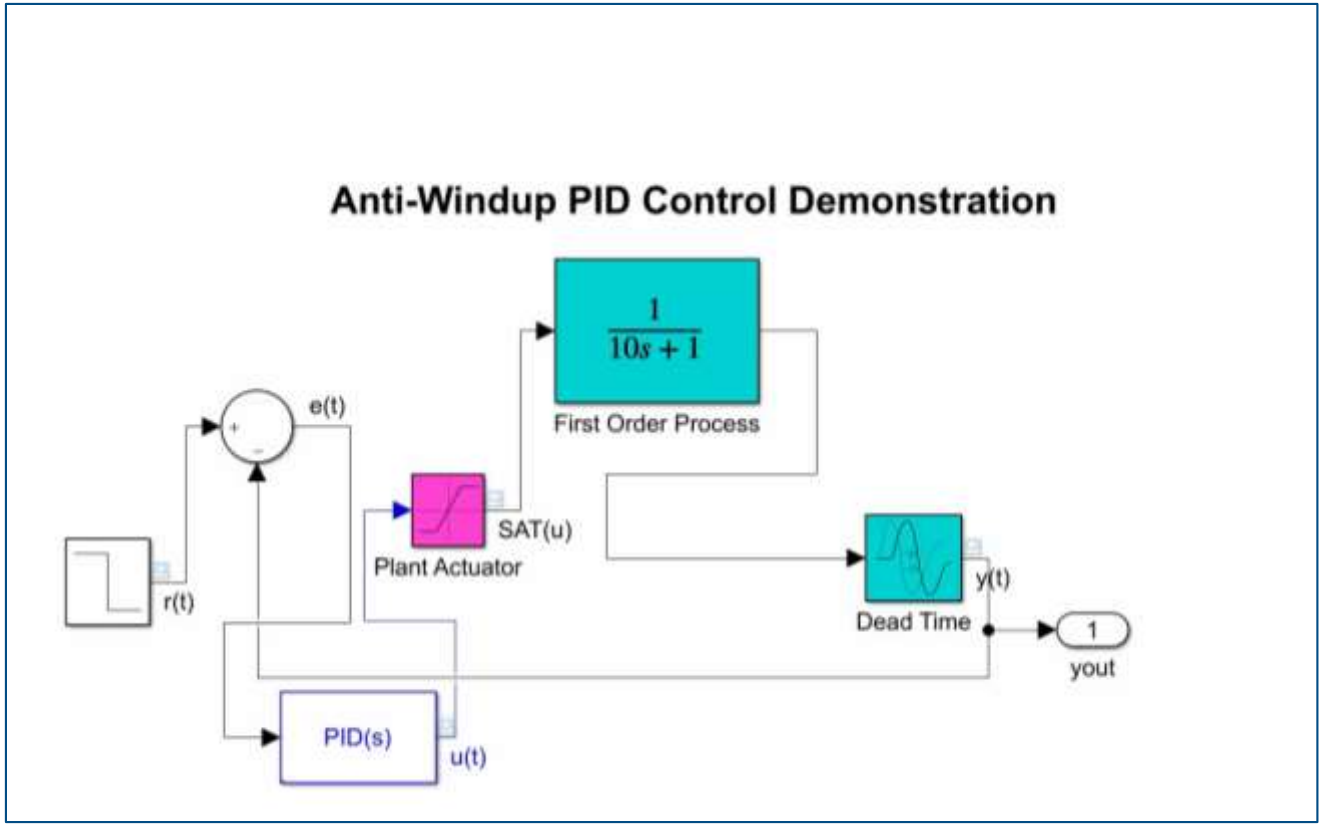
1. 创建测试
2. 实施最少的设计以通过测试
3. **重构**

重构

- 重构是一种仅改善代码内部结构而不改变其外部行为的软件更改过程

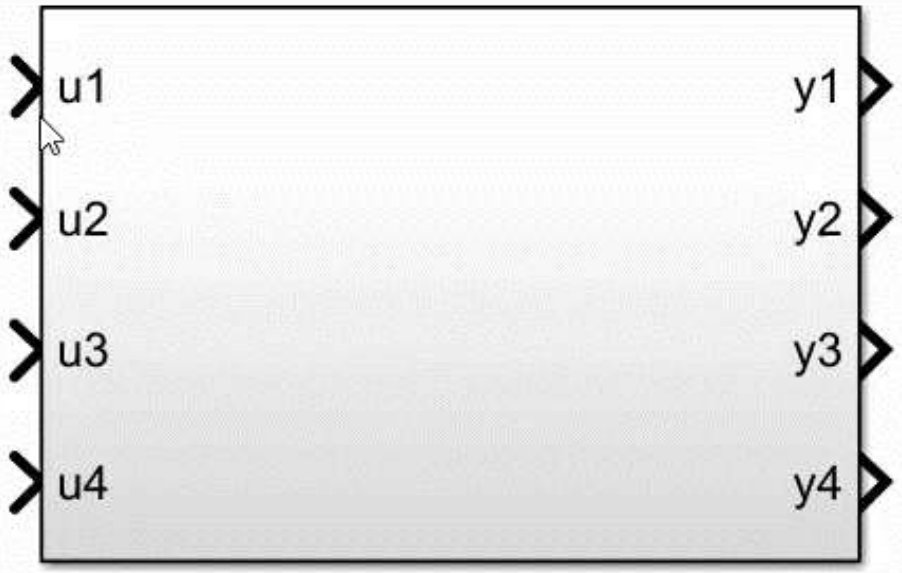
多种方式方法实现重构

- 重排布局



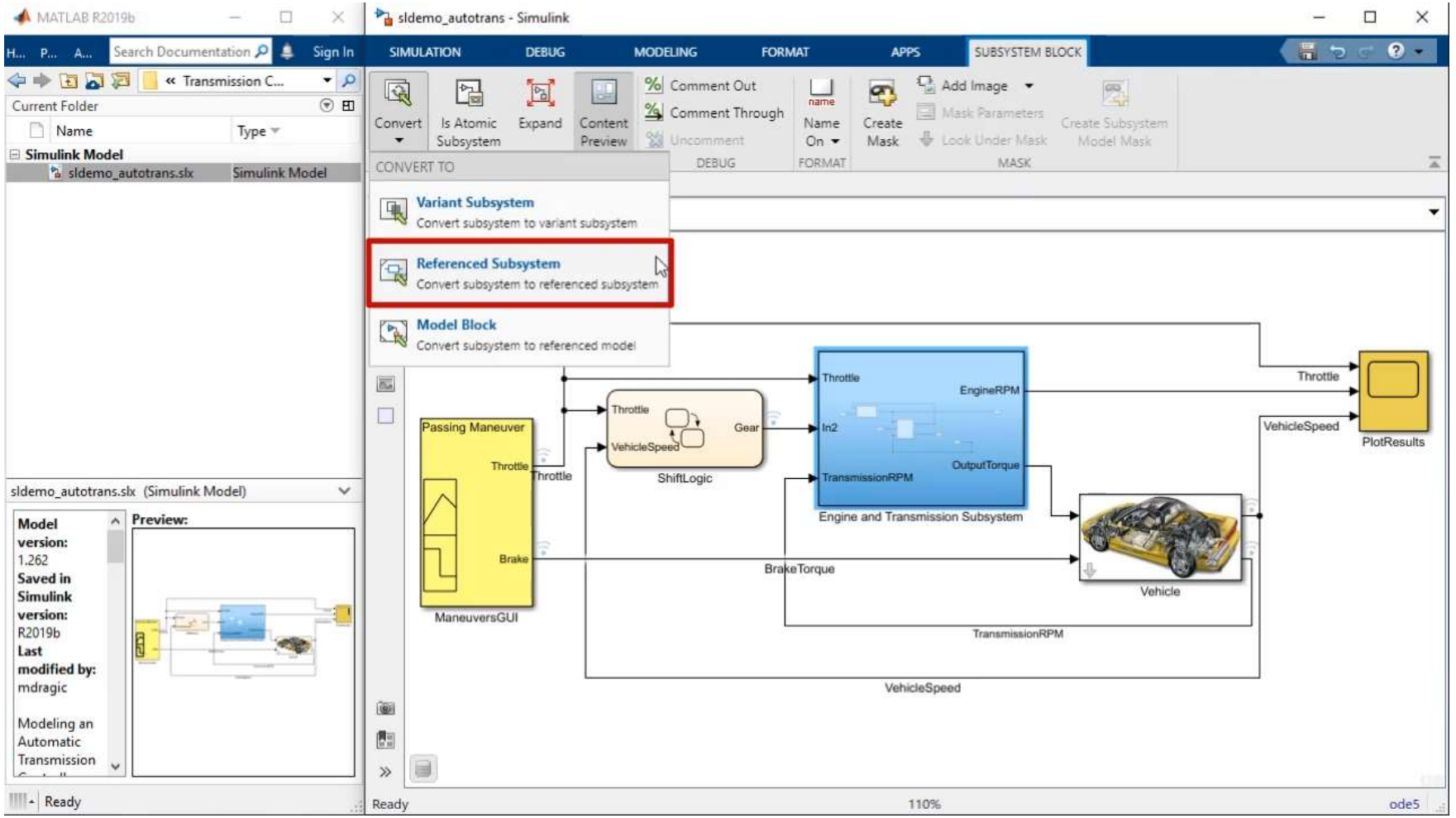
多种方式方法实现重构

- 重排布局



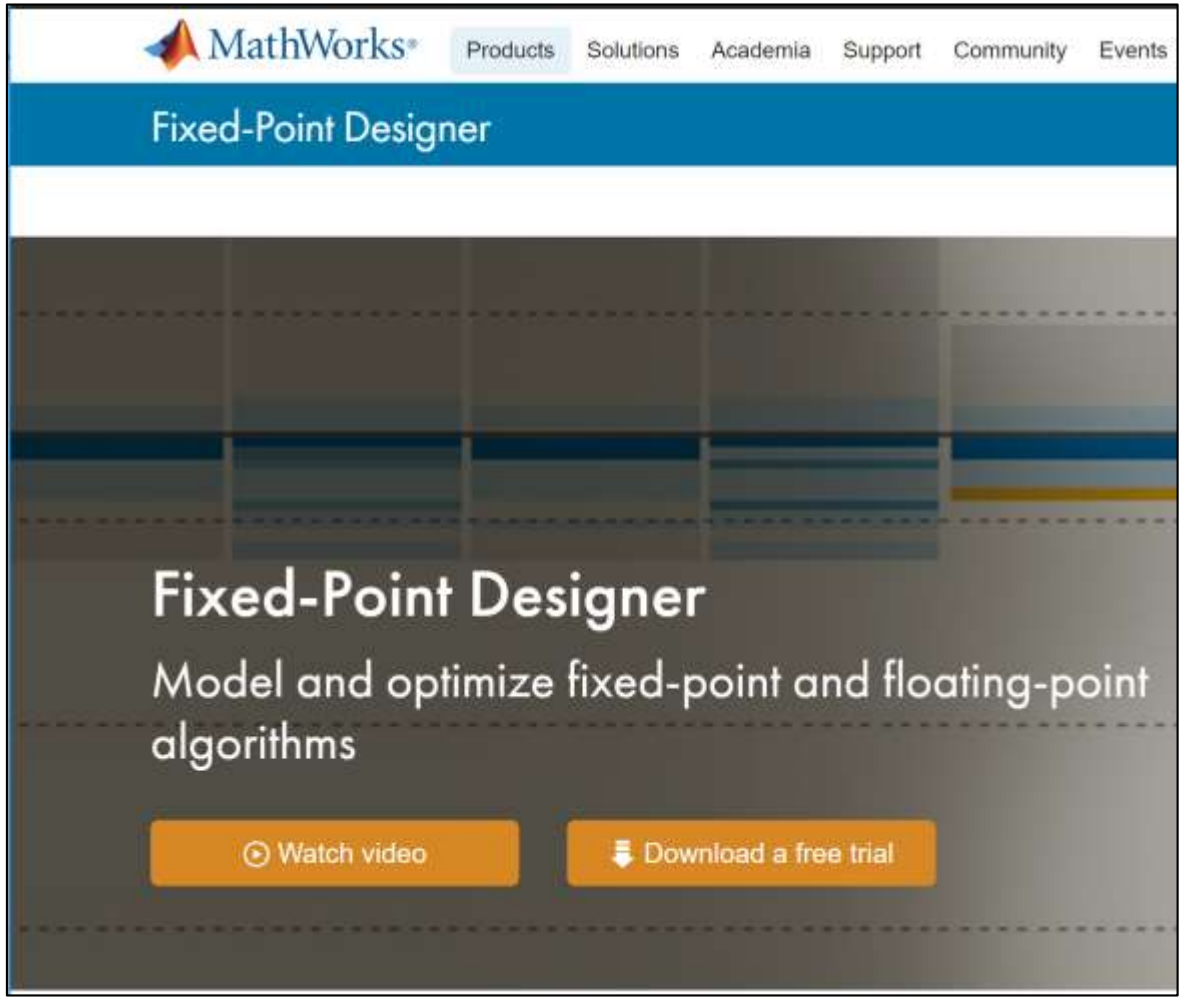
多种方式方法实现重构

- 重排布局
- 重构层次



多种方式方法实现重构

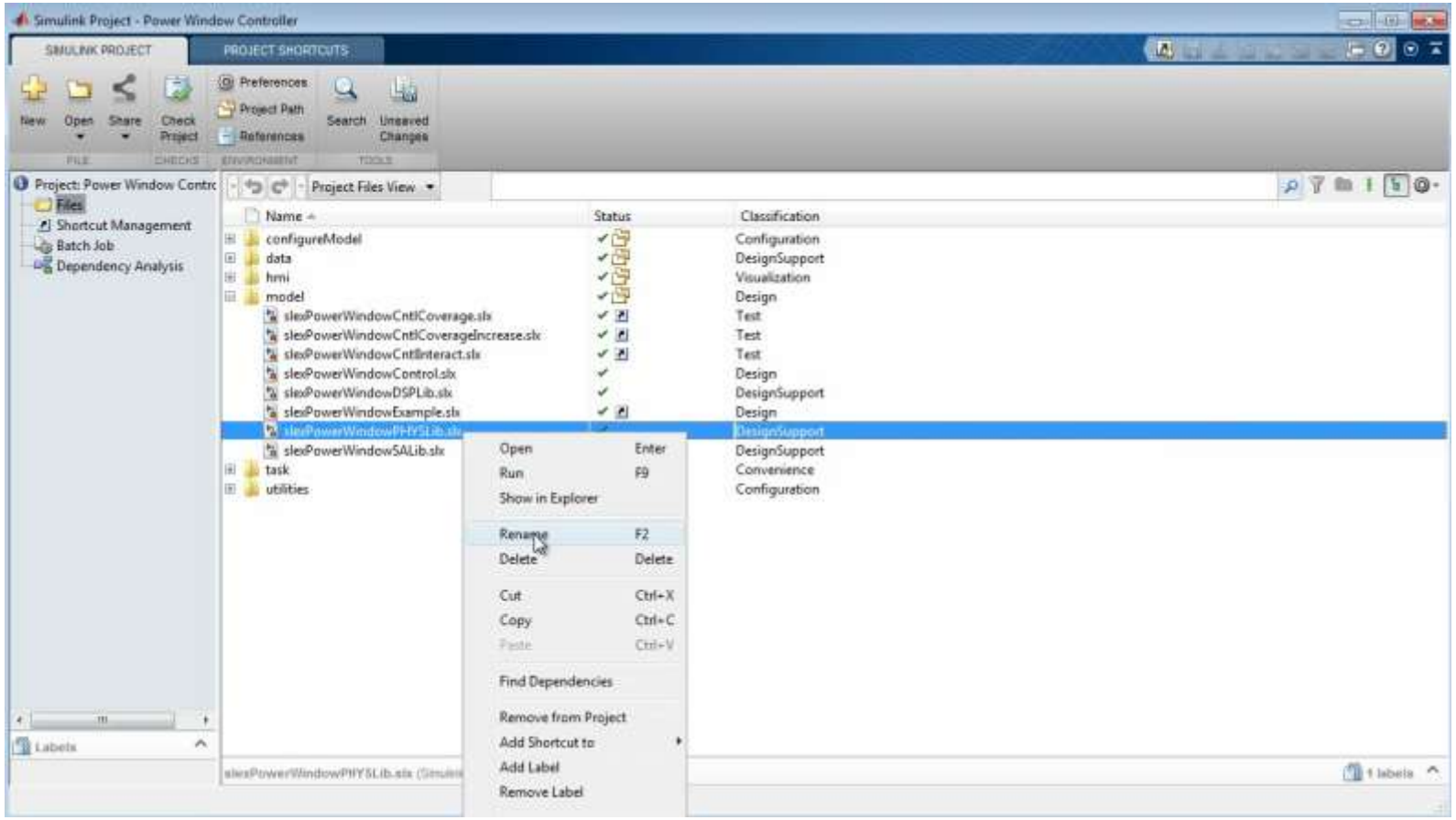
- 重排布局
- 重构层次
- 优化实现



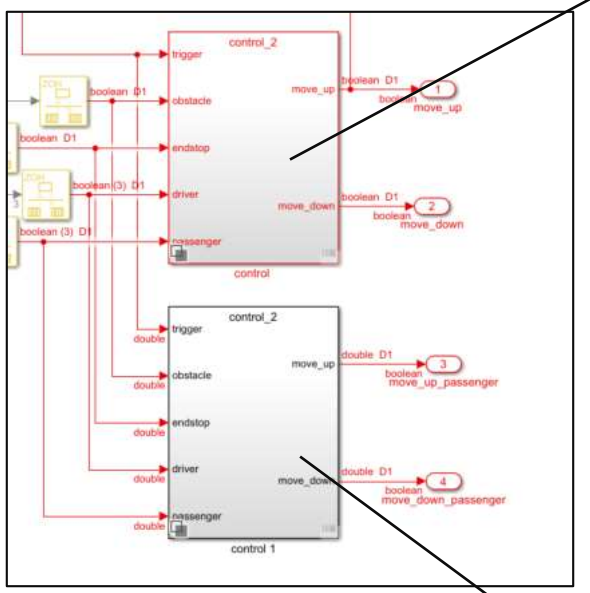
多种方式方法实现重构

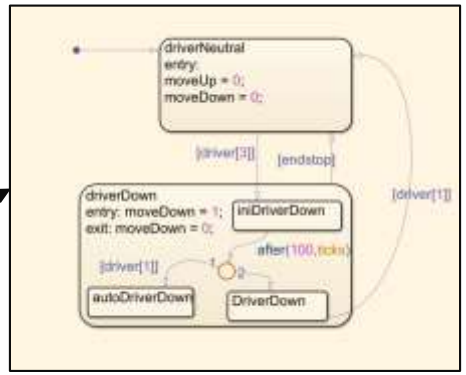
- 重排布局
- 重构层次
- 优化实现
- 工程项目范围内的重命名

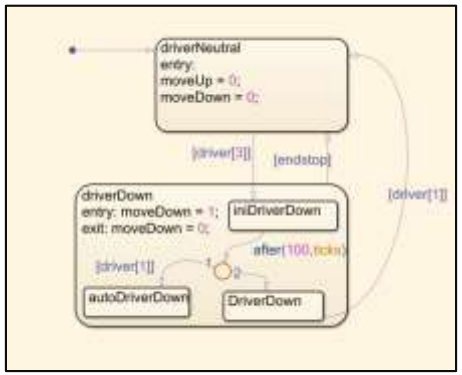
…. 更多!



通过整合冗余的Stateflow Chart实现重构

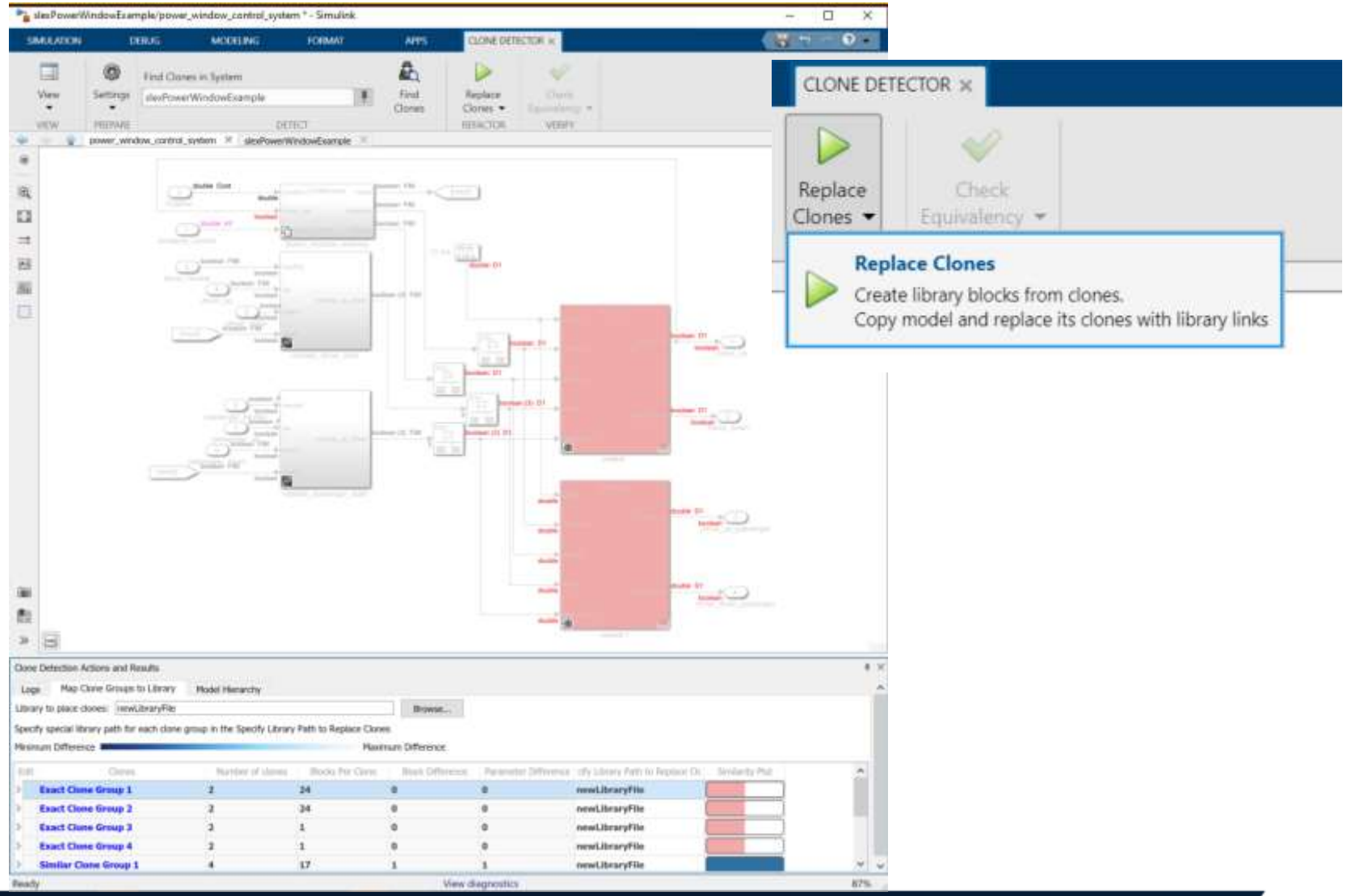






驾驶员和乘客的控制逻辑一致

使用Clone Detector App检查模型克隆（冗余）

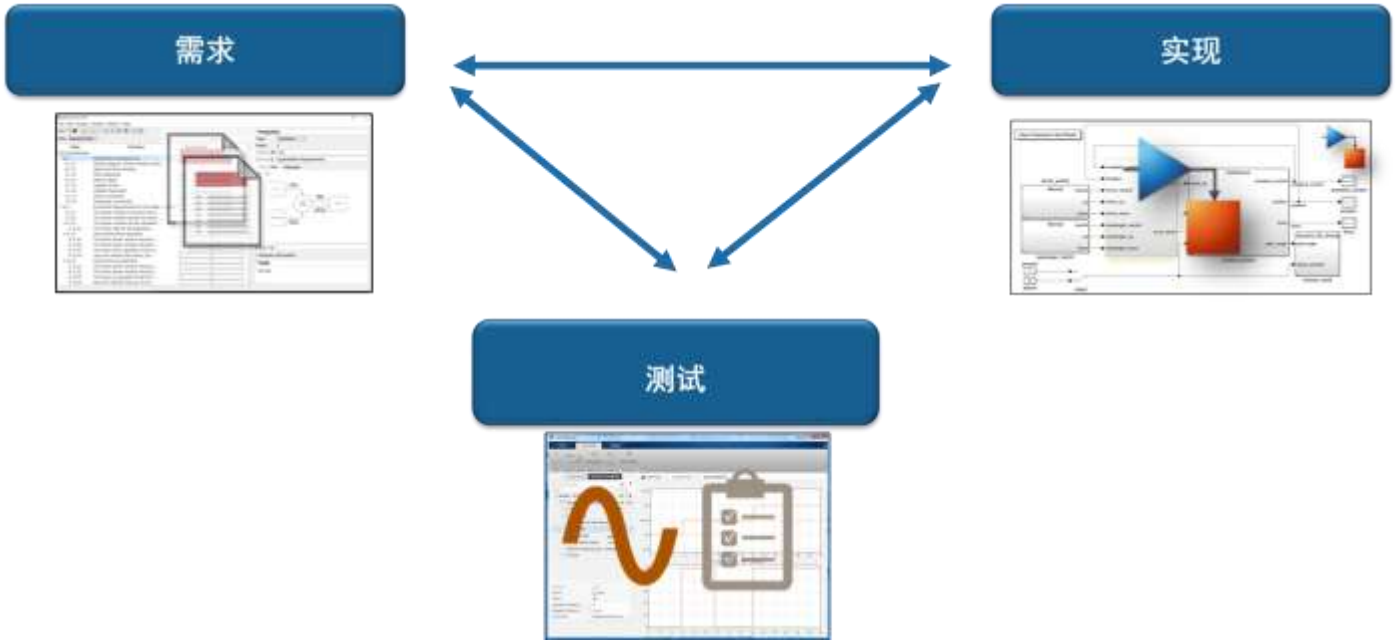


测试驱动的开发流程

1. 创建测试
2. 实施最少的设计以通过测试
3. 重构

结论&要点

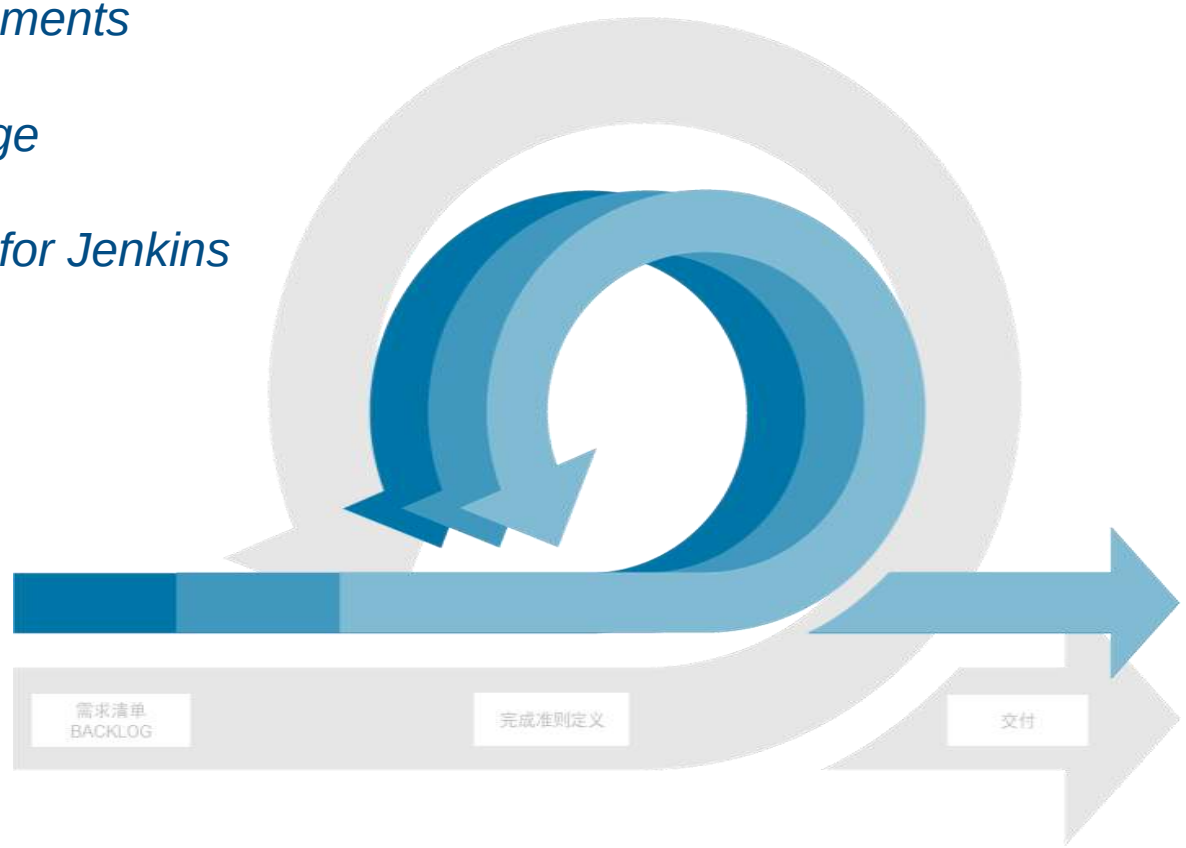
- Simulink为TDD提供集成框架
- 系统性地验证需求
- 自动化测试以更快交付可用系统



测试驱动开发

MATLAB和Simulink提供助力

- **基于模型的设计**
  - *Simulink and Stateflow*
- **需求管理**
  - *Simulink Requirements*
- **创建并执行测试**
  - *Simulink Test*
- **测试完备度评估**
  - *Simulink Coverage*
- **重构和合规性验证**
  - *Simulink Check*
- **持续集成**
  - *MATLAB Plug in for Jenkins*
- **组织，管理和共享**
  - *Projects*



了解更多

- Agile System Development with Model-Based Design
- Agile Model-Based Design: Accelerating Simulink Simulations in Continuous Integration Workflows
- Verification, Validation, and Test Solution Page
- Continuous Integration Solution Page