DO Qualification Kit

User's Guide

R2012a

MATLAB®
&SIMULINK®

MathWorks®

How to Contact MathWorks

DO Qualification Kit User's Guide

Trademarks

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

Patents

MathWorks products are protected by one or more U.S. patents. Please see www.mathworks.com/patents for more information.

Contents

Getting Started with DO Qualification Kit

1

Tool Qualification

2

Access and Manage Qualification Artifacts

3

Index

Getting Started with DO Qualification Kit

Product Description

Qualify Simulink® and Polyspace® verification tools to DO-178

DO Qualification Kit provides documentation, test cases, and procedures that let you qualify Simulink or Polyspace software verification tools for projects based on the DO-178 standard. The kit contains tool qualification plans, tool operational requirements, and additional materials required for qualifying software verification tools. With the kit, you can streamline certification of your embedded systems developed using Simulink or Polyspace products.

Specific versions of both the Simulink verification, validation, and test tools and the Polyspace code verification tools have qualification kits for the DO-178 standard. For each version of tools supported, DO Qualification Kit contains artifacts needed to qualify the tools with the standard. You can modify these artifacts as needed for your project and submit them to your certification authority as part of your DO-178 certification package.

Key Features

- Tool Qualification Plan and Tool Operational Requirements documents

- Test case models and code, test procedures, and expected results

- Traceability tables mapping test cases to requirements

- Qualification materials for Simulink verification, validation, and test tools

- Qualification materials for Polyspace code verification tools

What Is a Verification Tool?

Verification tools automate verification, validation, and testing activities to achieve efficiency and detect certain errors. A verification tool does not introduce errors, but might fail to detect errors that already exist in a product. (See DO-178B Section 12.2 and DO-254 Section 11.4.)

What Is Software Verification Tool Qualification for DO-178B?

Software that is included in avionic systems must be certified by meeting objectives of the DO-178B standard. To achieve credit for the use of a tool in the certification of an avionic system, you must qualify that tool if the output of the tool is not verified. The DO Qualification Kit product provides resources for qualifying MathWorks® software verification tools. The tool user (not the tool vendor) is responsible for the qualification.

For guidance on how to use the DO Qualification Kit product to qualify MathWorks software verification tools, see "Qualification Process" on page 2-2.

For information on criteria used to determine whether a tool needs to be qualified, see the "Requirement for Qualification" section of the Tool Qualification Plan for the relevant MathWorks verification tool.

Can You Apply the DO Qualification Kit Product to Other DO Standards?

In this section...
"DO-178C" on page 1-5
"DO-254" on page 1-5
"DO-278" on page 1-7

DO-178C

The publication of DO-178C, *Software Considerations in Airborne Systems and Equipment Certification*, is scheduled for 2012. MathWorks has been an active participant in the development of this standard for several years. Many aspects of the DO Qualification Kit, including the Model-Based Design for DO-178 workflow, can be applied to DO-178C. Additional support will be provided in a future release. To learn more about DO-178C and MathWorks tool support, visit Technical Solution 1-G355V1.

DO-254

You can use the DO Qualification Kit product to qualify verification tools for projects that follow DO-254, *Design Assurance Guidance for Airborne Electronic Hardware*. The process for tool qualification is described in Section 11.4 of DO-254. The following verification tool capabilities are applicable to DO-254 projects:

- Simulink Verification and Validation™ — DO-178B checks

- Simulink Verification and Validation — Model coverage

- Simulink Report Generator™ — System Design Description report

- SystemTest™ — Limit Check element

To take credit for DO-254 for the verification tool capabilities that are applicable to your project, modify all references to DO-178B to refer to the appropriate sections of DO-254. For example, modify references to DO-178B Section 12.2 to refer to DO-254 Section 11.4. The following tables list the items to modify and what files correspond to these items.

Simulink Verification and Validation Verification Tool

Item to Modify	File

Document templates	`qualkitdo_slvnv_tqp.rtf`
	`qualkitdo_slvnv_tor.rtf`
	`qualkitdo_slvnv_tcpr.rtf`

Test cases, procedures, and results	`qualkitdo_slvnv_tcpr1.xls*`
	`qualkitdo_slvnv_tcpr2.xls`
	`qualkitdo_slvnv_tcpr3.xls`
	`qualkitdo_slvnv_tcpr4.xls`
	`qualkitdo_slvnv_tcpr5.xls`
	`qualkitdo_slvnv_tcpr6.xls`

*The Model Advisor checks **Check safety-related optimization settings** and **Check safety-related code generation settings** are not applicable to DO-254. In `qualkitdo_slvnv_tcpr1.xls`, delete the rows corresponding to these checks. After deleting the rows, when you generate the report using `qualkitdo_slvnv_tcpr1.rpt`, the report does not include these two checks. Therefore, the report you generate does not match the example report `qualkitdo_slvnv_qualificationreport1.html`.

Simulink Report Generator Verification Tool

Item to Modify	File

Document templates	`qualkitdo_rptgenext_tqp.rtf`
	`qualkitdo_rptgenext_tor.rtf`
	`qualkitdo_rptgenext_tcpr.rtf`

SystemTest Verification Tool

Item to Modify	File

Document templates	`qualkitdo_systemtest_tqp.rtf`
	`qualkitdo_systemtest_tor.rtf`
	`qualkitdo_systemtest_tcpr.rtf`

Test cases, procedures, and results	`qualkitdo_systemtest_tcpr.test`

DO-278

DO-278, *Guidelines for Communications, Navigation, Surveillance, and Air Traffic Management (CNS/ATM) Systems Software Integrity Assurance*, is closely related to DO-178, and identical for tool qualification. Many aspects of the DO Qualification Kit, including the Model-Based Design for DO-178 workflow, can be used to meet the objectives of the DO-278 standard. To learn more about DO-278 and MathWorks tool support, visit Technical Solution 1-G355VC.

DO Qualification Kit Components

In this section...
"Model-Based Design Workflow for DO-178B" on page 1-8
"Supported MathWorks Software Verification Tools" on page 1-9
"Simulink® Verification and Validation™ Product" on page 1-10
"Simulink® Report Generator™ Product" on page 1-10
"Simulink® Code Inspector™ Product" on page 1-11
"Polyspace Products for C/C++" on page 1-12
"SystemTest Product" on page 1-13

To view the document templates, test cases, and test procedures that are part of the DO Qualification Kit product, use the Certification Artifacts Explorer. Artifacts included in the kit are not accessible from the MathWorks web site. For more information, see Chapter 3, "Access and Manage Qualification Artifacts".

Model-Based Design Workflow for DO-178B

The DO-178B software life cycle consists of objectives that must be met for each of the life cycle stages. In Appendix A of DO-178B, these objectives are summarized in tables. The *DO Qualification Kit Model-Based Design Workflow for DO-178B* document summarizes those tables and provides recommendations on meeting the objectives using a Model-Based Design process.

The workflow document is in the following folder:

> *matlabroot*/toolbox/qualkits/do/

The following table lists available workflow documents.

Workflow Document	File
Model-Based Design Workflow for DO-178B	qualkitdo_workflow.pdf

Supported MathWorks Software Verification Tools

The DO Qualification Kit product includes document templates, test cases, and test procedures that you need to qualify the following MathWorks software verification tools:

- Simulink Verification and Validation (DO-178B checks; model coverage)
- Simulink Report Generator (System Design Description report)
- Simulink Code Inspector™ (Code inspector report)
- Polyspace Client™ for C/C++; Polyspace Server™ for C/C++
- SystemTest (Limit Check element)

Note By qualifying the model coverage capability of the Simulink Verification and Validation product, you can claim credit for demonstrating completeness and correctness of the test cases generated by Simulink Design Verifier™. Use the model coverage capability instead of reviewing the Simulink Design Verifier report. You can demonstrate completeness and correctness by running the generated test cases against the model, measuring the model coverage, and reviewing the model coverage report. You can run the test cases on the executable object code and compare the actual results to the simulation results. The model coverage capability does not assess the correctness of the simulation results, only the test case coverage of the model. Therefore, you must verify that the expected results from the model are correct.

In addition to this documentation, the DO Qualification Kit product consists of the qualification documents described in the following sections. The qualification documents are located in your MATLAB® installation folder at *matlabroot*/toolbox/qualkits/do. Although the kit contains the documents and tests that you need to qualify the supported tools, you must determine how to apply the documents and tests to your specific project or application.

Refer to the PDF version of a qualification document to become familiar with the qualification process and the required content. To customize the qualification documents for your specific project or application, modify the RTF versions.

Simulink Verification and Validation Product

The DO Qualification Kit product supports qualifying the following versions and features of the Simulink Verification and Validation verification tool:

- Version 3.3 (R2012a): DO-178B checks
- Version 3.3 (R2012a): Model coverage

Document templates, test cases, and test procedures are in the following folder:

matlabroot/toolbox/qualkits/do/slvnv/r2012a/

The following table lists qualification documents, and the primary files associated with the qualification documents.

Qualification Documents	Files
Tool Qualification Plan (TQP)	qualkitdo_slvnv_tqp.pdf qualkitdo_slvnv_tqp.rtf
Tool Operational Requirements (TOR) and supporting documentation	qualkitdo_slvnv_tor.pdf qualkitdo_slvnv_tor.rtf qualkitdo_slvnv_ma_ui.pdf qualkitdo_slvnv_mc_ui.pdf
Test Cases, Procedures, and Results (TCPR), and supporting documentation	qualkitdo_slvnv_tcpr.pdf qualkitdo_slvnv_tcpr.rtf qualkitdo_slvnv_tcpr*.rpt qualkitdo_slvnv_tcpr*.xls tests/*.mdl outputs/ qualkitdo_slvnv_qualificationreport*.html

Simulink Report Generator Product

The DO Qualification Kit product supports qualifying the following version and feature of the Simulink Report Generator verification tool:

Version 3.12 (R2012a): System Design Description report

Document templates, test cases, and test procedures are in the following folder:

matlabroot/toolbox/qualkits/do/rptgenext/r2012a/

The following table lists qualification documents, and the primary files associated with the qualification documents.

Qualification Documents	Files
Tool Qualification Plan (TQP)	qualkitdo_rptgenext_tqp.pdf qualkitdo_rptgenext_tqp.rtf
Tool Operational Requirements (TOR) and supporting documentation	qualkitdo_rptgenext_tor.pdf qualkitdo_rptgenext_tor.rtf qualkitdo_rptgenext_sdd_ui.pdf
Test Cases, Procedures, and Results (TCPR), and supporting documentation	qualkitdo_rptgenext_tcpr.pdf qualkitdo_rptgenext_tcpr.rtf qualkitdo_rptgenext_tcpr.rpt outputs/ qualkitdo_rptgenext_qualificationreport.pdf qualkitdo_rptgenext_qualificationreport_*.html qualkitdo_rptgenext_qualificationreport_*.doc qualkitdo_rptgenext_qualificationreport_*.rtf qualkitdo_rptgenext_qualificationreport_*.pdf

Simulink Code Inspector Product

The DO Qualification Kit product supports qualifying the following version and feature of the Simulink Code Inspector verification tool:

Version 1.1 (R2012a): Code inspection report

Document templates, test cases, and test procedures are in the following folder:

matlabroot/toolbox/qualkits/do/slci/r2012a/

The following table lists qualification documents, and the primary files associated with the qualification documents.

Qualification Documents	Files
Tool Qualification Plan (TQP)	`qualkitdo_slci_tqp.pdf` `qualkitdo_slci_tqp.rtf`

Tool Operational Requirements (TOR) and supporting documentation	`qualkitdo_slci_tor.pdf`
	`qualkitdo_slci_tor.rtf`
	`qualkitdo_slci_ui.pdf`

Test Cases, Procedures, and Results (TCPR), and supporting documentation	`qualkitdo_slci_tcpr.pdf`
	`qualkitdo_slci_tcpr.rtf`
	`qualkitdoSlciTraceMatrix.xlsx`
	`qualkitdoSlciRunTests.m`
	`qualkitdoSlciRunTests.rpt`
	`qualkitdoSlciRunTests.xlsx`
	`tests/`
	` *.mdl`
	` *.m`
	` *.mat`
	`outputs/`
	` qualkitdoSlciQualificationReport_*.html`

Polyspace Products for C/C++

The DO Qualification Kit product supports qualifying the following version and feature of Polyspace code verification tools:

Version 8.3 (R2012a): Polyspace Client for C/C++; Polyspace Server for C/C++

Document templates, test cases, and test procedures are in the following folder:

matlabroot/toolbox/qualkits/do/polyspace/r2012a/

The following table lists qualification documents, and the primary files associated with the qualification documents.

Qualification Documents	Files
Theoretical Foundation (TF)	`qualkitdo_polyspace_tf.pdf` `qualkitdo_polyspace_tf.rtf`
Tool Qualification Plan (TQP)	`qualkitdo_polyspace_tqp.pdf` `qualkitdo_polyspace_tqp.rtf`
Tool Operational Requirements (TOR)	`qualkitdo_polyspace_tor.pdf` `qualkitdo_polyspace_tor.rtf`

Tests Cases, Procedures, and Results (TCPR), and supporting documentation	`qualkitdo_polyspace_tcpr.pdf`
	`qualkitdo_polyspace_tcpr.rtf`
	`Matrix_*.txt`
	`tests/qualkitdo_polyspace_tcpr.*`
	`tests/tor-checks/tor-checks.txt`
	`tests/*/*_rules.txt`
	`tests/*/*.c`
	`tests/*/*.cpp`
	`outputs/`
	`  qualkitdo_polyspace_qualificationreport_*.txt`
	`  qualkitdo_polyspace_qualificationreport_summary.pdf`

SystemTest Product

The DO Qualification Kit product supports qualifying the following version and feature of the SystemTest verification tool:

Version 2.6.3 (R2012a): Limit Check element

Document templates, test cases, and test procedures are in the following folder:

matlabroot`/toolbox/qualkits/do/systemtest/r2012a/`

The following table lists qualification documents, and the primary files associated with the qualification documents.

Qualification Documents	Files
Tool Qualification Plan (TQP)	`qualkitdo_systemtest_tqp.pdf` `qualkitdo_systemtest_tqp.rtf`

Tool Operational Requirements (TOR) and supporting documentation	`qualkitdo_systemtest_tor.pdf`
	`qualkitdo_systemtest_tor.rtf`
	`qualkitdo_systemtest_lc_ui.pdf`

Test Cases, Procedures, and Results (TCPR), and supporting documentation	`qualkitdo_systemtest_tcpr.pdf`
	`qualkitdo_systemtest_tcpr.rtf`
	`qualkitdo_systemtest_tcpr.test`
	`outputs/qualkitdo_systemtest_tcpr_report/`
	` qualkitdo_systemtest_qualificationreport.html`

What Certification Credit Can Be Achieved?

For a list of areas of the DO-178B standard for which some certification credit (meet or replace DO-178B objectives) can be achieved by qualifying MathWorks verification tools, see the "Certification Credit" section of the Tool Qualification Plan for the relevant tool. The amount of credit that can be achieved is dependent upon the development processes used and how verification tools are applied within the processes.

Required Knowledge

Before using the DO Qualification Kit product, make sure that you have:

- Knowledge about developing safety-critical software.

- Knowledge of DO-178B, Software Considerations in Airborne Systems and Equipment Certification, or DO-254, Design Assurance Guidance for Airborne Electronic Hardware.

- Experience using features of MathWorks verification tools that you want to qualify (and related required software, such as MATLAB and Simulink):

  - Simulink Verification and Validation (DO-178B checks, model coverage)

  - Simulink Report Generator (System Design Description report)

  - Simulink Code Inspector (Code inspection report)

  - Polyspace Client for C/C++, Polyspace Server for C/C++

  - SystemTest (Limit Check element)

Also, become familiar with the following information:

- "Modeling Guidelines for High-Integrity Systems" in the Simulink documentation.

- If you have a Embedded Coder™ license, "Develop Models and Code That Comply with the DO-178B Standard" in the Embedded Coder documentation.

Tool Qualification

- "Qualification Process" on page 2-2
- "Propose Tool Qualification Plan to Certification Authorities" on page 2-4
- "Document Tool Operational Requirements" on page 2-6
- "Verify That the Tool Satisfies Tool Operational Requirements" on page 2-7
- "Provide Certification Authorities with Tool Qualification Results" on page 2-9

Qualification Process

1 Provide certification authorities with a tool qualification plan.

2 Document tool operational requirements.

3 Verify that the tool satisfies tool operational requirements and define limitations and restrictions of the tool.

4 Provide certification authorities with tool qualification results, including limitations and restrictions.
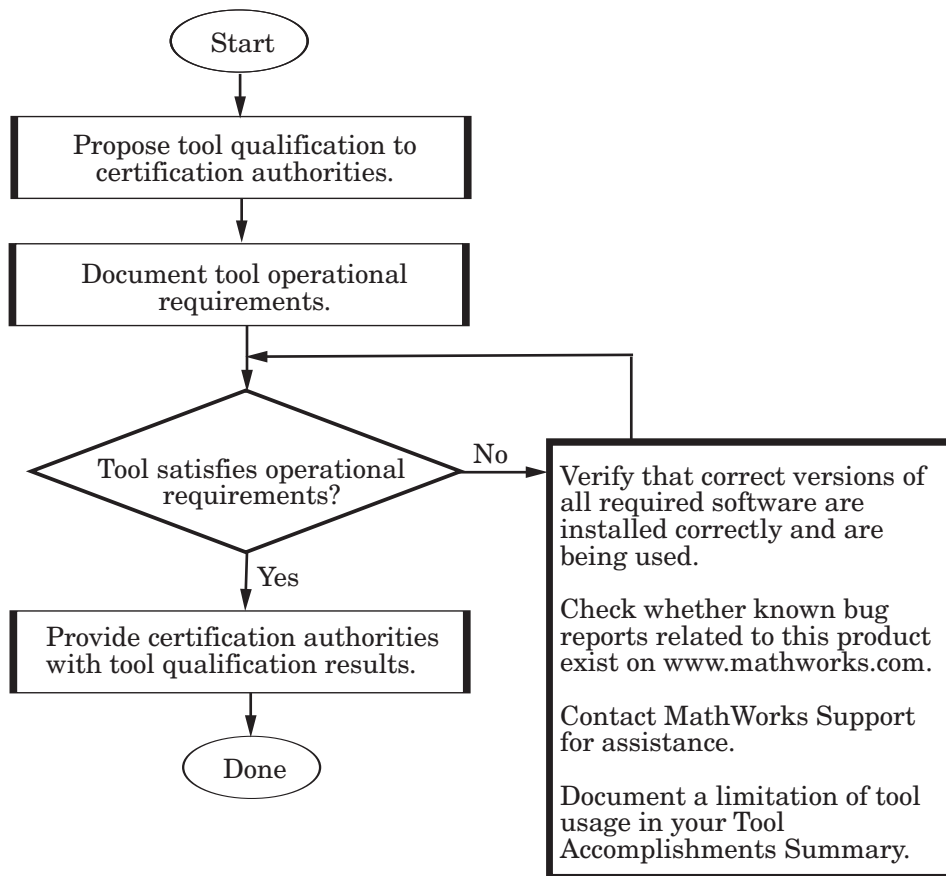
Note The `rights.txt` file, located at *matlabroot*/`toolbox/qualkits/do`, describes allowed uses of the MathWorks DO Qualification Kit product.

Propose Tool Qualification Plan to Certification Authorities

The first step of the qualification process is to propose a Tool Qualification Plan to the DO-178B certification authorities. To propose a Tool Qualification Plan, complete the following steps for each tool that you plan to qualify:

1 Identify a MathWorks software verification tool that you plan to qualify.

2 Locate the PDF version of the Tool Qualification Plan document provided for the tool in the DO Qualification Kit product. Review the document for applicability to your project.

3 Locate and make a copy of the `.rtf` Tool Qualification Plan template file for that tool.

The Tool Qualification Plan provides the following information:

- Introduction

- Title and version of the Tool Operational Requirements (TOR)

- Certification considerations — basis for qualifying the tool and objectives for which you are seeking certification credit

- Tool development lifecycle — planning, requirement documents and activity, verification documents and activity

- Tool lifecycle data — mapping of application documents to the DO-178B process lifecycle

- Qualification schedule

4 Edit the template, as necessary, for your application and make the document available for review. This typically includes adding the following information:

- In the table located in "Tool Lifecycle Data", where indicated, references to the documents:

  - Plan for Software Aspects of Certification (PSAC)

  - Software Accomplishment Summary (SAS)

  - Tool Qualification Accomplishment Summary (TQAS)

- Tool qualification schedule located in a section titled "Schedule"

5 Add a reference to the tool that you are qualifying and justification why the tool requires qualification to the application PSAC, indicating your intent to use the verification tool.

6 Submit the PSAC to the certification authorities. If you want, you can also submit the Tool Qualification Plan with the PSAC. These documents inform the certification authorities of your intent to use a tool and initiates a dialog regarding your plans for certification.

After you complete this procedure, document the Tool Operational Requirements, as described in "Document Tool Operational Requirements" on page 2-6.

Document Tool Operational Requirements

After you propose a Tool Qualification Plan to certification authorities, develop a Tool Operational Requirements document. To do this, complete the following steps for each tool that you plan to qualify:

1 Locate the PDF version of the Tool Operational Requirements document provided for the tool in the DO Qualification Kit product, and review the document for applicability to your project.

The Tool Operational Requirements document provides the following information:

- Introduction
- Operational requirements — description of the functionality of the tool and how it is used
- Installation information
- Operational environment

2 Install and configure the required software in your project configuration management system, as cited in the Tool Operational Requirements document.

3 Locate and make a copy of the `.rtf` Tool Operational Requirements document template file for the tool.

4 Edit the template, as necessary for your application, and make the document available for review by certification authorities.

After you complete this procedure, verify that the tool satisfies Tool Operational Requirements, as described in "Verify That the Tool Satisfies Tool Operational Requirements" on page 2-7.

Verify That the Tool Satisfies Tool Operational Requirements

After you document the Tool Operational Requirements, verify that the tool satisfies the Tool Operational Requirements. To do this, complete the following steps for each tool that you plan to qualify:

1 Locate the PDF version of the Test Cases, Procedures, and Results (TCPR) document provided for the tool in the DO Qualification Kit product, and review the test cases and procedures described in the document for applicability to your project.

The Test Cases, Procedures, and Results document provides the following information:

- Introduction

- Description of test cases and procedures

- Traceability matrix between operational requirements and test cases and procedures

- Procedure for executing the test cases and procedures

- Test results

2 Locate and make a copy of the `.rtf` Test Cases, Procedures, and Results document template file for the tool.

3 Edit the test cases and procedures sections of the template, as necessary for your application, and finalize the document. For example, you might need to modify content in the Expected Result column of the table in the traceability matrix.

4 Locate and configure the test cases and procedures in your project configuration management system.

5 Execute the test cases and procedures in the installed and configured environment. For more information, see "Executing the Test Cases and Procedures" in the Test Cases, Procedures, and Results document for the verification tool you are qualifying.

6 Review the test results. If the results are not correct:

- Verify that correct versions of all required MathWorks software are installed correctly and being used.

- Check whether known bug reports related to this product exist on the MathWorks Web site (`www.mathworks.com`).

- Contact MathWorks support for assistance.

- Document a limitation of the tool usage in your Tool Accomplishments Summary if you cannot resolve the test failure.

7 Configure the test results in your project configuration management system.

After you complete this procedure, document and provide the results to certification authorities, as described in "Provide Certification Authorities with Tool Qualification Results" on page 2-9.

Provide Certification Authorities with Tool Qualification Results

After you verify whether the tool satisfies tool operational requirements, complete the following steps for each tool that you are qualifying.

1 Document the test results in a Tool Accomplishment Summary.

2 Add an entry that summarizes the results of the tool qualification to the Software Accomplishment Summary (SAS). You can also submit the Tool Accomplishment Summary with the SAS.

3 Submit the SAS to the certification authorities so that they can approve the results. The data can provide a record of the qualification status of the tool.

Access and Manage Qualification Artifacts

- "Access Artifacts Using the Certification Artifacts Explorer" on page 3-2
- "Manage Artifacts Using the Certification Artifacts Explorer" on page 3-4
- "Delete Certification Packages from the Certification Artifacts Explorer" on page 3-6
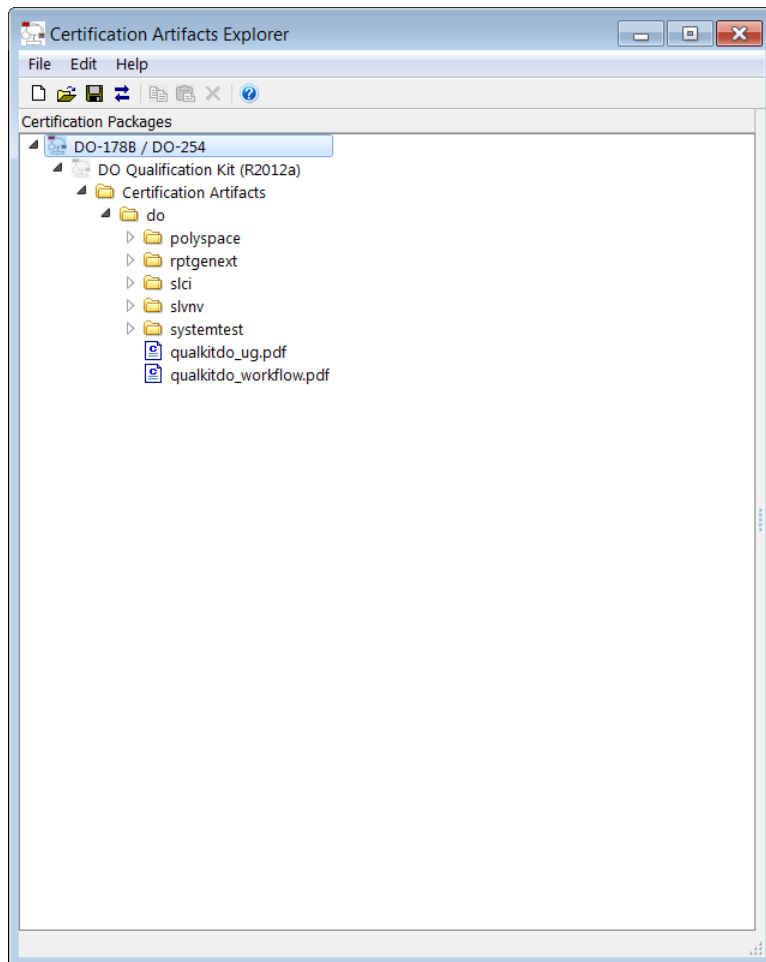- "Certification Artifacts Explorer Limitations" on page 3-7

Access Artifacts Using the Certification Artifacts Explorer

You can use the Certification Artifacts Explorer to access qualification artifacts. To start the Certification Artifacts Explorer, use one of the following methods.

To start the Certification Artifacts Explorer...	Do this:
From the MATLAB **Start** menu	Select **Start > Simulink > DO Qualification Kit > Certification Artifacts Explorer**.
From the MATLAB command line	Enter `qualkitdo`.

The Certification Artifacts Explorer window displays the qualification artifacts that are available with the DO Qualification Kit product. If the DO Qualification Kit product contains artifacts for more than one release, the Certification Artifacts Explorer lists the artifacts for each release. As you select folders and files, relevant information about the current selection is dynamically displayed in the status bar. Additionally,

- To display the properties of a certification package, right-click the package name and select **Properties**.

- To open an artifact, right-click the artifact and select **Open File**.

- Other right-click actions might be available depending on the type or state of an artifact, including **Copy**, **Paste**, **Delete**, **Open Folder**, and **Generate Traceability Matrix**.

Manage Artifacts Using the Certification Artifacts Explorer

To manage tool qualification artifacts using the Certification Artifacts Explorer:

1 Create a new certification package.

Tip Most certification package actions can be initiated in multiple ways. For example, to create a new package, you can:

- Select the menu item **File > New**.
- Click the toolbar icon **Create new certification package**.
- Right-click an existing package and select **New**.

---

2 Name the certification package.

3 Define the location where the Certification Artifacts Explorer stores the certification package (use right-click **> Properties**).

4 Save the certification package. The saved package has a KIT extension.

5 Copy the tool qualification artifacts for the product of interest into the certification package.

6 Delete qualification artifacts that are not required for your project.

7 Optionally, add related files to the certification package using a file browser such as Microsoft® Windows® Explorer.

Tip When you add files, to refresh the file list, use **File > Refresh**.

---

8 Use the Certification Artifacts Explorer to access tool qualification artifacts.

When you create and save new certification packages, the Certification Artifacts Explorer displays them. The certification packages that are listed

remain visible unless you delete them from the Certification Artifacts
Explorer.

Delete Certification Packages from the Certification Artifacts Explorer

The Certification Artifacts Explorer displays certification packages that you create or open. If you delete a certification package from the Certification Artifacts Explorer, the files associated with the package are still available on your computer. To delete the files, use a file browser such as Windows Explorer.

Certification Artifacts Explorer Limitations

The Certification Artifacts Explorer has the following limitations:

- The Certification Artifacts Explorer works on Microsoft Windows platforms only.

- For optimal performance, ensure that Microsoft Internet Explorer® is available on your machine. Internet Explorer does not have to be your default web browser.

Traceability Matrix Creation

Generate a Traceability Matrix

In this section...
"About Traceability Matrices" on page 4-2
"Prerequisites for Generating a Traceability Matrix" on page 4-3
"Generate a Traceability Matrix" on page 4-4

About Traceability Matrices

When you use Model-Based Design and production code generation to develop application software components, you can generate a *traceability matrix*. The traceability matrix provides traceability among model objects, generated code, and model requirements. You can add comments to the generated traceability matrix. If you change the model and regenerate the traceability matrix, the software retains your comments.

For a given model, the generated traceability matrix can provide information about:

- Model objects that are traceable between the model and generated code, such as Simulink blocks, Stateflow® objects, and MATLAB functions.

- Model objects that are untraceable between the model and generated code, such as eliminated and virtual blocks.

- Requirements documents that you link to model objects using the Simulink Verification and Validation Requirements Management Interface (RMI).

Generate the traceability matrix using either the do178b.ExportTraceReport function from the MATLAB Command Window or the **Generate Traceability Matrix** button in the generated HTML code generation report for your model. Either method creates an XLS file that contains the following worksheets:

- **Model Information** — Summary of the model configuration and checksum. The summary includes the model name, version, author, creation date, last saved by, last updated date, checksum, and the selection of **Traceability Report Contents** parameters.

- **Code Interface** — Information about the generated code interface, such as function prototype and timing information for the model initialize and step functions.

- **Code Files** — File folders and names of the generated code files.

- **Report** — Traceability information for each model object, including model, generated code, and requirements. Each row in the worksheet pertains to a single occurrence of a model object. The information for a model object is in more than one row if the object:

  - Appears more than once in the generated code.

  - Links to more than one requirement.

Prerequisites for Generating a Traceability Matrix

Before generating a traceability matrix for model objects, generated code, and model requirements, perform the following steps:

1 Optionally, attach requirements documents. For more information, see "Requirements Traceability" in the Simulink Verification and Validation documentation.

2 In the Configuration Parameters dialog box, on the **Code Generation > Report** pane, select:

a "**Create code generation report**"

b At least one of the following **Traceability Report Contents** parameters:

- "**Eliminated / virtual blocks**"

- "**Traceable Simulink blocks**"

- "**Traceable Stateflow objects**"

- "**Traceable MATLAB functions**"

Tip If you want to generate the traceability matrix directly from the code generation report, select "**Open report automatically**".

3 Generate code for the model.

Tip You do not have to build an executable to generate a traceability matrix. To generate code only, on the **Code Generation > General** pane, select **Generate code only**.

Generate a Traceability Matrix

To generate a traceability matrix:

1 Open the model if it is not already open.

2 Ensure that you have completed the "Prerequisites for Generating a Traceability Matrix" on page 4-3.

3 Generate the traceability matrix using one of the following methods:

- In the MATLAB Command Window, enter the following command, where *model_name* is the name of the model:

  ```
  do178b.ExportTraceReport('model_name')
  ```

  The software generates the traceability matrix.

- Open the code generation report for the model if it is not already open. Go to the **Traceability Report** section and click the **Generate Traceability Matrix** button. For example:
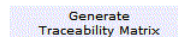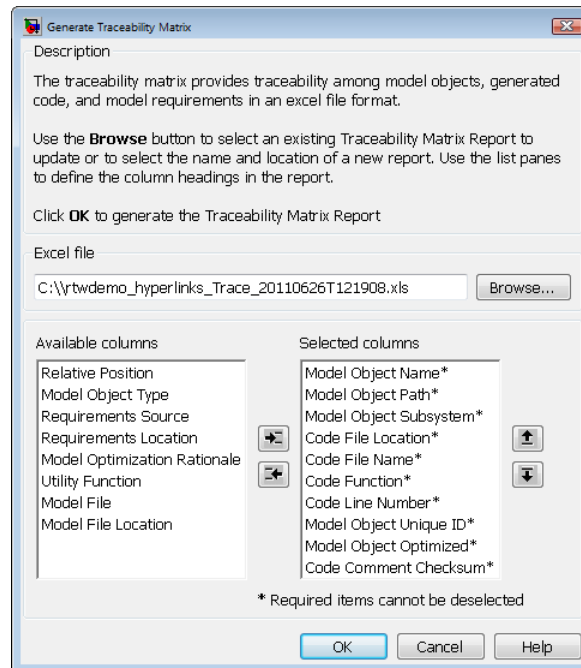
Traceability Report for rtwdemo_hyperlinks

Generate Traceability Matrix

Table of Contents

1. Eliminated / Virtual Blocks
2. Traceable Simulink Blocks / Stateflow Objects / MATLAB Functions
   - rtwdemo_hyperlinks
   - rtwdemo_hyperlinks/Chart
   - rtwdemo_hyperlinks/Chart:43

When you click the button, the Generate Traceability Matrix dialog box appears.

You can use this dialog box to browse to an existing matrix file to update or specify a new matrix file to create. Optionally, you can also use this dialog box to select and order the columns that appear in the generated matrix. Click **OK** to update or create the specified report.

4 Review the traceability matrix and add comments in new columns. For more information, see "Add Comments to a Traceability Matrix" on page 4-6.

Add Comments to a Traceability Matrix

In this section...
"Requirements for Adding Comments to a Traceability Matrix" on page 4-6
"Retain Comments" on page 4-7

Requirements for Adding Comments to a Traceability Matrix

You can add comments to the traceability matrix that you generated using the `do178b.ExportTraceReport` function.

To add comments to the traceability matrix, you must:

- Create new columns for your comments.

- Use unique column headings. All columns that you add must have headings.

- Add at least one entry to the column, other than the column heading.

- Retain the following columns:
  - Model Object Name
  - Model Object Path
  - Model Object Subsystem
  - Code File Location
  - Code File Name
  - Code Function
  - Code Line Number
  - Model Object Unique ID
  - Model Object Optimized
  - Code Comment Checksum

> **Note** All comments must resolve to a text string. For example, a link to an image resolves to a text string, but a copy of the image does not.

Retain Comments

To regenerate a traceability matrix and retain your comments:

1 Navigate to the working folder of the model.

2 Optionally, regenerate code for your model. Regenerating code before generating the traceability matrix ensures that you have the latest model-to-code traceability information.

3 In the MATLAB Command Window, enter the following command. *file_name* is the name of the existing traceability matrix that you are regenerating. If the existing traceability matrix is in a different folder, include the full path to that folder in *path*.

```
do178b.ExportTraceReport('model_name', 'file_name', 'path')
```

The traceability matrix regenerates.

Traceability Matrix Limitations

The traceability matrix generation capability has the following limitations:

- Does not support generating a traceability matrix for referenced models. When you generate a traceability matrix for a model that contains referenced models, the traceability matrix contains information about the Model block only. The traceability matrix does not contain information about the contents of the referenced model. If your model contains referenced models, generate a traceability matrix for the top-level model and each referenced model separately.

- Works with the Microsoft Windows platform only.

- In most cases, identifies comments that you add to the traceability matrix, but when comments cannot be identified, the traceability matrix includes the text:

    Row is not unique: *comment*

- Does not support information stored in external `.req` files. For example, when you generate a traceability matrix for a model with externally stored requirements information, the traceability matrix does not include the requirements information.

5

Function Reference

Qualification Artifacts Management

qualkitdo   Open Certification Artifacts Explorer

Qualification-Related Development Activities

do178b.ExportTraceReport   Generate XLS file that contains
                                  traceability matrix

Functions — Alphabetical List

do178b.ExportTraceReport

Purpose   Generate XLS file that contains traceability matrix

Syntax   `do178b.ExportTraceReport('`*`model_name`*`')`
`do178b.ExportTraceReport('`*`model_name`*`', '`*`file_name`*`')`
`do178b.ExportTraceReport('`*`model_name`*`', '`*`file_name`*`', '`*`path`*`')`

Description   `do178b.ExportTraceReport('`*`model_name`*`')` generates an XLS file
that contains a "Traceability Matrix" on page 6-3. *model_name* is the
name of the model.

`do178b.ExportTraceReport('`*`model_name`*`', '`*`file_name`*`')` generates
an XLS file that contains a "Traceability Matrix" on page 6-3.
file_name is a string that specifies the name of the file. The first time
that you call `do178b.ExportTraceReport`, *file_name* is optional. If
you do not provide *file_name*, the function names the file using the
following convention. *modelUpdate* is the date and time that you last
updated the model:

    *model_name*_Trace_*modelUpdate*.xls

To regenerate the traceability matrix, you must specify *file_name*.

`do178b.ExportTraceReport('`*`model_name`*`', '`*`file_name`*`', '`*`path`*`')`
generates an XLS file that contains a "Traceability Matrix" on page 6-3.
path is an optional string that specifies the full path to the location
where you want the software to save the file.

Tips
- The `do178b.ExportTraceReport` function works in Microsoft
  Windows platforms only.

- To include requirements documentation in the traceability
  matrix, attach requirements documents to the model before using
  `do178b.ExportTraceReport`.

- You must generate a code generation traceability report (requires
  an Embedded Coder license) for your model before using
  `do178b.ExportTraceReport`.

- The `do178b.ExportTraceReport` function does not support
  generating a traceability matrix for referenced models. When you

generate a traceability matrix for a model that contains referenced models, the traceability matrix contains information about the Model block only. The traceability matrix does not contain information about the contents of the referenced model. If your model contains referenced models, generate a traceability matrix for the top-level model and each referenced model separately.

- In most cases, the `do178b.ExportTraceReport` function identifies comments that you add to the traceability matrix. When the function cannot identify comments, the traceability matrix includes the text:

    Row is not unique: *comment*

For more information, see "Prerequisites for Generating a Traceability Matrix" on page 4-3.

Definitions   **Traceability Matrix**

A traceability matrix provides traceability among model objects, generated code, and model requirements. You can add comments to the generated traceability matrix. If you change the model and regenerate the traceability matrix, the software retains your comments.

Examples   Generate a traceability matrix with traceability between model objects and generated code for the rtwdemo_hyperlinks model:

Note This example requires an Embedded Coder license.

```
% Open the model.
open_system('rtwdemo_hyperlinks');
% Generate code only.
set_param('rtwdemo_hyperlinks', 'GenCodeOnly', 'on');
% Initiate the build process.
rtwbuild('rtwdemo_hyperlinks');
% Generate a traceability matrix.
do178b.ExportTraceReport('rtwdemo_hyperlinks');
```

Generate a traceability matrix with traceability among model objects, generated code, and model requirements for the slvnvdemo_fuelsys_docreq model:

Note This example requires a Simulink Verification and Validation license.

```
% Open the model.
open_system('slvnvdemo_fuelsys_docreq');
% Select the code generation report and traceability report parameters.
set_param('slvnvdemo_fuelsys_docreq', 'GenerateReport', 'on');
set_param('slvnvdemo_fuelsys_docreq', 'GenerateTraceReport', 'on');
set_param('slvnvdemo_fuelsys_docreq', 'GenerateTraceReportSl', 'on');
set_param('slvnvdemo_fuelsys_docreq', 'GenerateTraceReportSf', 'on');
set_param('slvnvdemo_fuelsys_docreq', 'GenerateTraceReportEml', 'on');
% Generate code only.
set_param('slvnvdemo_fuelsys_docreq', 'GenCodeOnly', 'on');
% Initiate the build process.
rtwbuild('slvnvdemo_fuelsys_docreq');
% Generate a traceability matrix.
do178b.ExportTraceReport('slvnvdemo_fuelsys_docreq');
```

Alternatives   You can generate a traceability matrix directly from the code generation report for your model. Go to the **Traceability Report** section and click the **Generate Traceability Matrix** button.

How To

- "Generate a Traceability Matrix" on page 4-2
- "Add Comments to a Traceability Matrix" on page 4-6
- "Code Tracing"
- "Requirements Traceability"

Purpose   Open Certification Artifacts Explorer

Syntax   `qualkitdo`

Description   `qualkitdo` opens the Certification Artifacts Explorer and displays qualification artifacts.

Tips
- The `qualkitdo` function works in Microsoft Windows platforms only.

- For optimal performance, ensure that Microsoft Internet Explorer is available on your machine. Internet Explorer does not have to be your default web browser.

Alternatives Open the Certification Artifacts Explorer by selecting **Start > Simulink > DO Qualification Kit > Certification Artifacts Explorer**.

How To
- Chapter 3, "Access and Manage Qualification Artifacts"

- Chapter 2, "Tool Qualification"

qualkitdo

A

References

Normative References

DO-178B, *Software Considerations in Airborne Systems and Equipment Certification.* RTCA, Inc., 1992.

DO-178C, *Software Considerations in Airborne Systems and Equipment Certification.* In preparation.

DO-254, *Design Assurance Guidance for Airborne Electronic Hardware.* RTCA, Inc., 2000.

DO-278, *Guidelines for Communication, Navigation, Surveillance, and Air Traffic Management (CNA/ATM) Systems Software Integrity Assurance.* RTCA, Inc., 2002.

Index