



QKIT Overview

Compiler Qualification Kit

 Safety Multicore Development Suite



QKIT

TOOL QUALIFICATION



Toolchain Standard Compliant Qualification. Automated.

HighTec Core Products

Development & Consulting



COMPILER

BUILD TOOL



PXROS-HR

REAL-TIME OS



QKIT

TOOL QUALIFICATION



CONSULTING

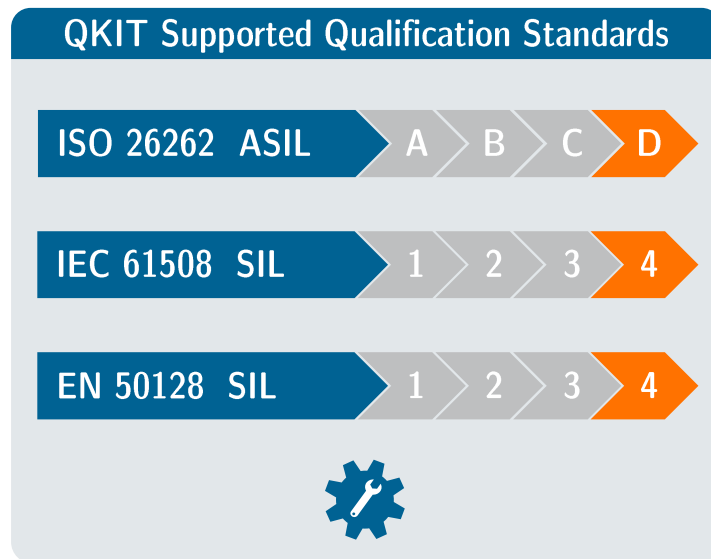
DESIGN & SUPPORT



Safety Multicore Development Suite

Compiler Qualification Kit Overview

For Safety Critical Applications

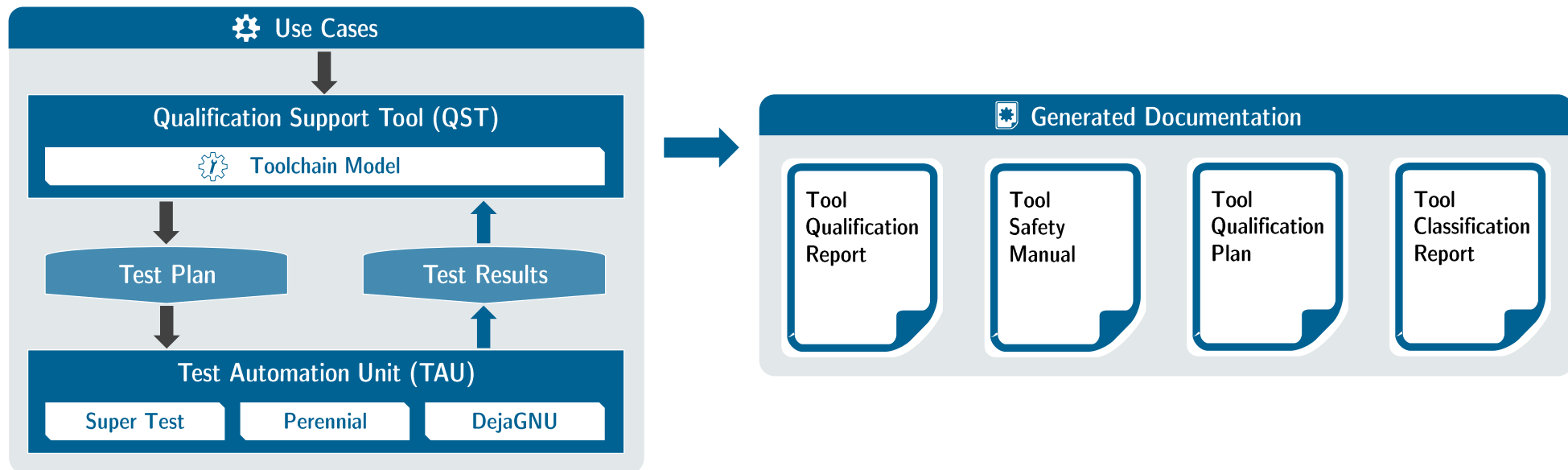


- Enables to perform standard compliant qualification in a simple way
- Significantly reduces effort for test and document generation process
- Transparent and traceable validation flow
- Flexible and extensible due to model-based approach

Main Components of the QKIT

QKIT Architecture

- › Qualification Support Tool QST contains toolchain model
- › Test Automation Unit TAU integrates different test suites



Qualification Support Tool

Toolchain Model and Test Generator



- › Contains an extensive model of the tool
 - › Toolchain structure, its tools, artifacts and features
 - › Potential errors, known bugs and mitigation measures
 - › Test cases for errors with not known mitigation measures
- › For each specified Use Case QST computes
 - › List of measures to mitigate potential errors
 - › List of tests proving that no non-mitigable errors can occur
 - › Resulting Tool Confidence Level (TCL)

Test Automation Unit

Test Suites Integration



- › Integrates different standard test suites
 - › SuperTest Rembrandt Release for massive testing against different language standards
 - › Perennial Validation Suite
 - › The DejaGNU test suite for GNU GCC
- › Supports easy extension for other customized tests

Tool Safety Manual

Generated Documents



- › Description of methods
- › Requirements tracing to standards
- › Tool, usage and operation dependent safety guidelines
- › List of measures to mitigate potential errors of the selected Use Case

Tool Classification Report

Generated Documents



- Resulting Tool Confidence Level (TCL)
- Description of the TCL derivation method
- List of TCL for each component of the toolchain
- Determination of TCL for each use case and for each of the components (Compiler, Linker...)

Tool Qualification Plan

Generated Documents



- Use Cases and features with qualification needs
- Validation goals, requirements of standards and how they are to be satisfied
- Qualification environment
- Planned qualification process

Tool Qualification Report

Generated Documents



- Use Cases and features that have been qualified
- Test environment
- Test results and analysis for test cases with not PASS result
- Executed qualification process

QKIT GUI

Use Case Selection and Modification

Qualification Summary for Qualification of GCC
Summarizes the qualification that can be started now by creating the qualification documents of GCC in Compile Assemble Link With Optimization

Target Directory: C:\QST\HDPQKit\workspace\Qualification

Names of Use Cases: Compile Assemble Link With Optimization

Number of Features: 8

Number of selected Checks: 21

Number of selected Restrictions: 2

Number of Tests: 12

Name:	Path:
Tool Classification Report	C:\QST\HDPQKit\workspace\Qualification\Validation\Documentation\TC
Tool Qualification Plan	C:\QST\HDPQKit\workspace\Qualification\Validation\Documentation\TC
Tool Safety Manual	C:\QST\HDPQKit\workspace\Qualification\Validation\Documentation\TS
Tool Test Path	C:\QST\HDPQKit\workspace\Qualification\Validation\TestPath.txt
Tool Test Plan	C:\QST\HDPQKit\workspace\Qualification\Validation\TestExecution.txt
Toolchain Model	C:\QST\HDPQKit\workspace\Qualification\QKit\Model\Model.tca

Buttons: ? Save < Back Next > Finish Cancel

Feature Selection
Please select for each use case which features it uses

Tool GCC (TCL3)
Use Case Compile Assemble Link With Optimization (TCL3)

- Feature Merge Constants
- Feature Merge Jumping To A Jump Instru
- Feature Omit Frame Pointer
- Feature Optimization Level
 - Feature Optimization Level 0
 - Feature Optimization Level 1
 - Feature Optimization Level 2
 - Feature Optimization Level 3
- Feature Optimize For Size
- Feature Optimize Even Against IEEE and A

Compile Assemble Link With Optimization

Description:
gcc -Wall -Werror -O2 -mcpu=tc1796

Buttons: ? Save < Back Next > Finish Cancel

Questions?

Let Us Hear You!



www.hightec-rt.com



support@hightec-rt.com



sales@hightec-rt.com

Thank You

