

LDRA Productivity Package for Automotive

LDRA Productivity Package for Automotive

The LDRA Productivity Package for Automotive is a set of targeted capabilities drawn from the LDRA tool suite enabling customers to achieve ISO 26262 compliance through to Automotive Safety Integrity Level (ASIL) D, the highest level of assurance for automotive software. The productivity package also supports the development of software in accordance with Automotive SPICE® (Software Process Improvement Capability dEtermination) processes and practices.



The package includes LDRA's best in class capabilities for the development and verification of automotives software. The package's flexibility and scalability addresses the changing needs of the most demanding projects and enhances collaboration across large and small teams, making it ideal for use at any point in the software development lifecycle. The breadth of capabilities included in the package make it suitable for use by automotive OEMs and suppliers developing automotive software to varying levels of functional safety and security assurance.

The LDRA Productivity Package for Automotive embraces both static and dynamic analysis techniques to encompass a wide range of essential features designed to ensure compliance with functional safety and security standards.

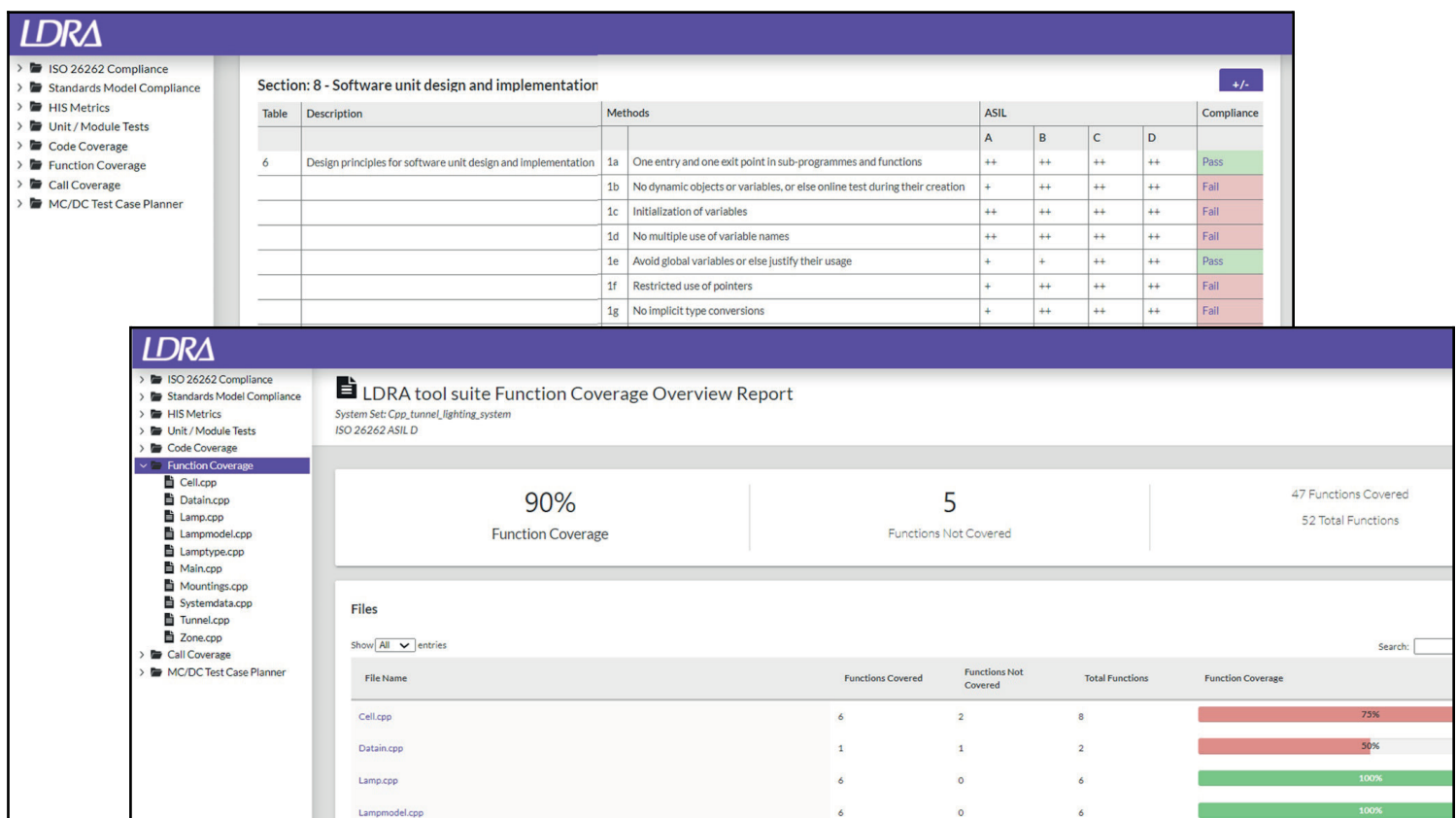
The screenshot displays the LDRA Productivity Package for Automotive 10.2.0 interface. The File Explorer on the left shows the project structure for 'Cpp_tunnel_lighting_system'. The Results View on the right shows the analysis results for 'Cpp_tunnel_lighting_system' under the 'MISRA-C++:2008' model. The results are categorized by 'Number Violations', 'Level of Violations', 'Phase Code', and 'Standard Code'. The Code Coverage results show the percentage of code covered for various components, including 'Combined Coverage Run', 'Statement Coverage', 'Branch Coverage', and 'Modified Condition / Decision Coverage'.

Component	Statement Coverage	Branch Coverage	Modified Condition / Decision Coverage
Cpp_tunnel_lighting_system	84	82	38
Zone.cpp	88	90	90
TunnelData::SystemData::SystemData (Code Coverage results from Zone.cpp)	0	0	0
TunnelData::SystemData::SystemData (Code Coverage results from Zone.cpp)	0	0	0

Static code analysis is used to evaluate code quality and gather critical metrics such as those identified in the Hersteller Initiative Software metrics (HIS). It enables checking against the rules defined by coding standards such as MISRA, CERT, CWE and AUTOSAR in addition to user configurable rules sets.

Dynamic analysis capabilities include unit, integration and system level testing on simulators and target hardware alike. Structural coverage analysis including MC/DC and function and call coverage enables the assessment of coverage statistics and the implementation of requirements-based and on-target testing to ensure that code is implemented in accordance with its developer's intentions.

Reporting of both static and dynamic analyses is tuned to industry requirements and formatted to reflect the objectives of standards such as ISO 26262. Optional add-on capabilities including requirements traceability, worst-case execution time analysis, and web-based project reporting of testing and analysis results. These capabilities complement the base productivity package to provide a comprehensive toolkit for ensuring the highest level of assurance in the development of automotive software, meeting the rigorous standards of the automotive industry.



The LDRA Productivity Package for Automotive at a glance:

- Functional safety and security standards compliance with ISO 26262 and optionally with ISO/SAE 21434
- Static code analysis including key quality, complexity and testability metrics
- Coding standards compliance including MISRA, CERT, CWE and AUTOSAR for both C and C++, along with many other commercially available coding standards
- The ability to define your own coding standards
- Dynamic structural coverage analysis including MC/DC, function coverage and call coverage
- Unit and system level testing on target hardware
- Requirements-based and on-target testing
- Automatic test case generation, test harness generation, and stub generation
- Security analysis and reporting, including taint analysis
- Data coupling and control coupling analysis
- Automatic generation of ISO 26262 Compliance Reports, Function Coverage, and Call Coverage Reports
- Can be extended through add-on capabilities including requirements traceability, web-based project-level reporting, and worst-case execution time analysis
- Breadth of capabilities meets the needs of most commercial and defense software development and verification projects
- Flexibility and scalability to support projects at any point in the lifecycle

TASKING®

EMPOWERING YOUR CODE, PULSE BY PULSE.



COMPILE | DEBUG | TEST

TASKING Germany GmbH



Streitfeldstrasse 19, 81673 Munich
Germany



+49 (0)89 2620 10080



www.tasking.com