



LDRA
Software Technology

Product Brochure

Automating Software Verification,
Requirements Traceability and
Standards Compliance

LDRA customers utilise the *LDRA tool suite*[®] to produce safe, secure, and mission-critical software in an accelerated, cost-effective and requirements-driven process.

***LDRA Certification Services*[®] complement the tool suite and assures certification or qualification in the most stringent industry-regulated development environments. In association with this, *LDRA's Compliance Management System*[®] (LCMS) assures safety standards compliance with safety experts and Level A DERs providing documentation templates, process checklists and packaged certification.**

LDRA solutions deliver increased developer productivity and software quality while reducing overall project schedule and cost.

The *LDRA tool suite* is a completely integrated yet open and extensible solution enabling customers to build quality into their software from requirements through to deployment. The tool suite has enabled critical software development programs to achieve certification or approval under rigorous standards in:

- Aerospace and Defence (DO-178B/C, DO-254, DO-278, Def Stan 00-56)
- Automotive (ISO 26262)
- Industrial and Energy (IEC 61508)
- Medical (IEC 62304)
- Rail Transportation (EN 50128)

Technologies and services offered by LDRA have been successfully deployed across hundreds of projects expediting development, verification, and certification of critical software. The *LDRA tool suite* accelerates productivity throughout the entire software project team.

Daily users include project management, systems engineers, developers, QA managers and test and maintenance engineers.

LDRA's unrivalled pedigree draws on over 40 years of success in the software quality market. As leaders in the critical software verification and validation market, LDRA consistently contributes to advancing state-of-the-art software engineering practices to help ensure the delivery of high quality safety-, security-, and mission-critical systems.

LDRA participates in numerous standards bodies such as the MISRA C and C++ committees, the SC-205/WG-71 (DO-178C) committee and the SC22/WG14 C language committee, including the working group developing the "C secure coding rules".



LDRA Lifecycle Automated Activities

LDRA Tool Suite – Build Quality into Your Software Development Lifecycle

The *LDRA tool suite* supports you throughout your software development lifecycle. Our software standards compliance, testing and verification tools are based on industry best practices to help you develop high quality safety- and security-critical products. Many users of the *LDRA tool suite* are required to certify or qualify their software. The tool suite's open and extensible platform is unique in its integration of software lifecycle traceability, static and dynamic analysis and unit test and system-level testing on virtually any host or target platform.

LDRA's software testing tools expedite your certification/approval process through:

- The provision of traceability for all requirements, design and verification artifacts throughout the software development lifecycle
- Enforcing compliance with coding standards (industry- and user-defined)
- Automating unit test and system-level test
- Performing and reporting coverage analysis on all code down to the target level
- Support for planning and execution of requirements-based testing
- Tool chain integration
- Automating the generation of software certification and approval evidence

The right software testing tools and services for the most challenging safety-critical, security-critical and business-critical projects:

- DO-178B/C Level A (Aerospace)
- IEC 61508 SIL 4 (Industrial)
- ISO 26262 ASIL A – ASIL D (Automotive)
- EN 50128 SIL 3/4 (Rail Transportation)
- IEC 62304 (Medical Devices, Class II and III)

The *LDRA tool suite* is ideal for development teams who are moving to a more formal development process to improve code quality and maintainability, by utilising static and dynamic analysis, coding standards compliance, and unit, integration and system testing.

Lifecycle Best Practices Enabled with the *LDRA Tool Suite*

1. Requirements Traceability

If you are developing a safety-critical or security-critical application, chances are good that your process will need to be audited to meet certification or qualification requirements. Bi-directional requirements traceability—from requirements, to design, to implementation and verification activities and artifacts—differentiates your organisation from the competition and ensures your shortest path to device approval. Even if your process and resulting code does not need to meet stringent certification demands, requirements traceability and management increases code quality and the overall safety, security and effectiveness of the application.

LDRA provides a comprehensive, role-based approach to traceability. Requirements and verification tasks can be assigned to team members and all resulting artifacts can be aggregated and linked. The result is a completely traceable process throughout the software development lifecycle so you can ensure that any changes to requirements, design, or source code are easily understood, verified and traced.

LDRA tools help you detect changes in requirements or the developed software and easily organise and re-run appropriate tests against any affected components. We provide an optimal environment for determining impact analysis either upstream to requirements or downstream to design, implementation and verification.

- *TBmanager*[®] is a role-based requirements management tool that lets teams work on individual activities and link code and verification artifacts back to higher-level objectives.
- *TBreq*[®] provides traceability throughout the workflow and quickly identifies broken links.

2. Test Management

With today's complex embedded applications, you need to develop and execute effective tests against your software. Equally important is knowing when to re-run those tests as your requirements or the software changes and to ensure that your code's performance or functionality has not degraded.

LDRA provides an automated approach to detecting requirements or software changes and then re-running the appropriate tests against the affected components. Verification activities can be easily assigned and tracked to team members, and the resulting artifacts are aggregated into a clean set of reports.

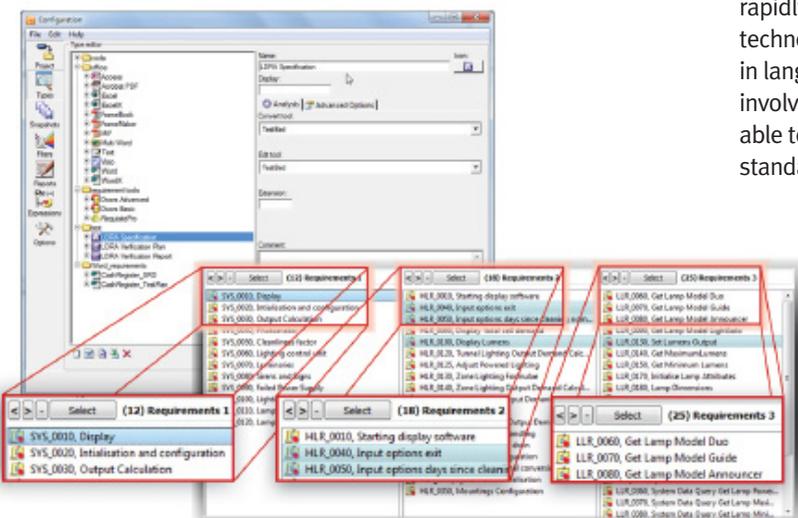
- *TBmanager*[®] lets you assign tasks to individuals on the team based on requirements.
- *TBevolve*[®] is an optional plug-in that lets project teams monitor the impact of code changes on their testing process.

3. Coding Standards Compliance

A common way to improve the consistency, maintainability and overall quality of code is to assess the code against industry- or company-defined standards so that deficiencies can be quickly identified and corrected. While this is often achieved through manual peer reviews, the process can be painful and time-consuming, especially in large projects. If your team is struggling to maintain code quality and consistency, it might be time to take a more rigorous yet automated approach.

LDRA automates static analysis and rules checking with tools that support almost any standard or standards you define. With LDRA's code visualisation, you can identify exactly where the source code deviates from the standard so it can be rapidly addressed. We have complete control over our parsing technology, which means we can quickly adapt to variations in languages and language constructs. And with our long involvement in standards organisations such as MISRA, we are able to respond rapidly with updated rules checking as new standards are released.

Our coding standards compliance tools let you combine standards and define appropriate rule subsets, select individual rules, and add your own. Within the tool, you can easily check for coding standards compliance to any single standard or combination of standards or subsets. And you can check compliance of a single code base against multiple standards to compare how the code fulfills each one and see what would be required to adapt the code to conform to it.

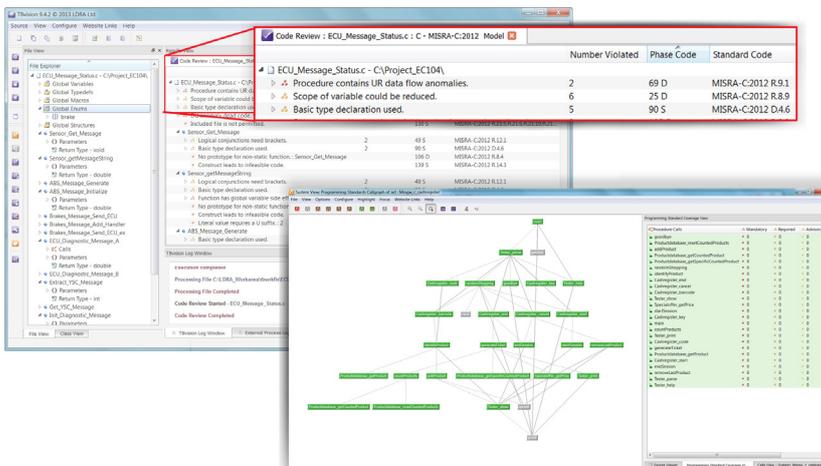


Requirements Traceability - With TBreq, you can see relationships between levels of requirement to quickly highlight the requirement's impact upstream and downstream of the selection.

- **LDRA Testbed®/TBvision®** is the static analysis engine and code visualisation capability that helps enforce compliance with coding standards and clearly indicates software flaws that might otherwise pass through the standard build and test process.
- **LDRARules®** is a cost-effective, stand-alone rules checker that does not require investment in the complete tool chain.
- **TBevolve®** is an optional plug-in that lets project teams monitor code changes with respect to coding standards compliance.

Our tools support C, C++, Java, Ada and Assemblers, running on a broad range of target platforms—from powerful 64-bit microprocessors to highly constrained 8- or 16-bit microcontrollers. LDRA tools can automatically generate test cases that typically provide 50-80% of coverage and our intuitive test case building environment lets developers quickly augment those test cases to increase their coverage if necessary.

- **TBrun®** with the **TBextreme®** module provides automated unit testing, test case generation and execution, and code coverage analysis.



Coding standards compliance is displayed inline with file/function name to show which aspects of the system do not comply with the standard. Programming standards callgraphs shows a high-level, colour-coded view of coding standards compliance of the system.

4. Code Coverage Analysis

Many regulatory authorities now require evidence of structural coverage analysis to prove that the code has been adequately tested and to measure software test effectiveness. For safety-critical software development, certification bodies need reports that detail the extent of code coverage. And of course, you need to be able to provide all of that efficiently, so you do not slow development.

LDRA's coverage analysis tools address the most stringent requirements in the safety- and security-critical markets. We are the only company that offers coverage analysis at both the source code and object code levels to help you meet the most stringent coverage requirements. LDRA tools automatically generate test cases, execute those test cases and visually report levels of coverage analysis, such as statement, branch/decision, procedure/function call, MC/DC, dynamic data flow and more.

- The **TBobjectbox®** module adds object code verification.
- The **Tool Qualification Support Pack (TQSP)** provides documentation on our own tools and qualification guidance and artifacts to meet certification requirements.
- **LDRAunit®** is a stand-alone tool for automated unit test generation and management.
- **LDRAcover®** is a stand-alone code coverage tool to help you meet structural coverage analysis objectives up to RTCA DO-178B/C Level A.

5. Automated Code Review

With today's complex systems, manual peer reviews are time-consuming and costly. You need to raise the level of abstraction to quickly gain insight into the architecture and behaviour, control flow, and data flow of your code in order to highlight and correct potential problem areas and produce software quality metrics. If you are developing to comply with stringent standards—such as the latest version of DO-178C in the aerospace market—you may be required to demonstrate analysis of data flow and control flow coupling for software certification.

LDRA provides detailed insight into your code through graphical code visualisation tools, including callgraphs, procedure flowgraphs, and data flow analysis reports. These are powerful components in the software debug and test environment that are not supported in most other software development tool environments. With these easy-to-use tools, your team can more quickly and easily identify bottlenecks such as logic flaws or race conditions.

- **LDRA Testbed®** and **TBvision®** provide a graphical static and dynamic analysis tool for both host and embedded software analysis.
- **Dynamic Data Flow Coverage (DDFC)** is an optional module that examines exactly which variables were used during runtime execution of the application.

6. Unit/Integration/Target Testing

Testing on the host platform enables functional testing without consideration for hardware timing or constraints. Often this is performed prior to target hardware availability. In some development environments, simulators extend this capability by giving software teams “virtual” targets and simulated connected peripherals to provide a jumpstart on verification activities early in the development cycle. Ideally, and in most certification environments, the same tests executed on the host must also be executed on the target hardware to ensure proper functional testing. Rather than writing your own code for test harnesses, vectors, and stubs (and then having to manage all of those both on the host platform, with simulators, and on the target), automated unit testing and target testing tools can save you time and headaches by doing all this for you.

With LDRA, you can quickly and easily generate and execute tests at the unit and integration levels both on the host (standalone or with target simulation) as well as on the target hardware. We provide test generation (test harness, test vectors, code stubs) and result-capture support for a wide range of host and target platforms. Our optimised instrumentation technology lets you pull test information even off highly constrained 8- and 16-bit microcontrollers, and through high-performance 32- and 64-bit processors.

With that range of support, your team has a common unit testing and integration testing environment for multiple projects with different target platforms.

- **TBrun**[®] provides automated unit test and system integration testing capabilities.
- **LDRAunit**[®] is a stand-alone tool that provides a complete integrated environment for automated unit test generation and management, without requiring investment in a complete tool chain.
- **TBxtreme**[®] (an optional module for either **TBrun**[®] or **LDRAunit**[®]) revolutionises unit testing by providing a totally automated solution for test vector generation.

7. Code Quality Review

Manual code inspections for today’s systems are no longer cost-effective due to the increasing size and complexity of modern applications. Instead, developers need to quickly and effectively assess the quality of their code using software metrics for testability, maintainability, and clarity. Each of these characteristics is fundamental to the overall assessment of code quality, and the process for determining them requires software quality metrics tools to apply efficient and accurate measurement.

LDRA automates code quality review by analysing the code and reporting on metrics for testability, maintainability, and clarity. Even beyond these metrics, LDRA helps you identify and visually highlight areas of code which may require refactoring. You can easily configure the parameters of the overall quality model to meet the needs of your organisation or application requirements.

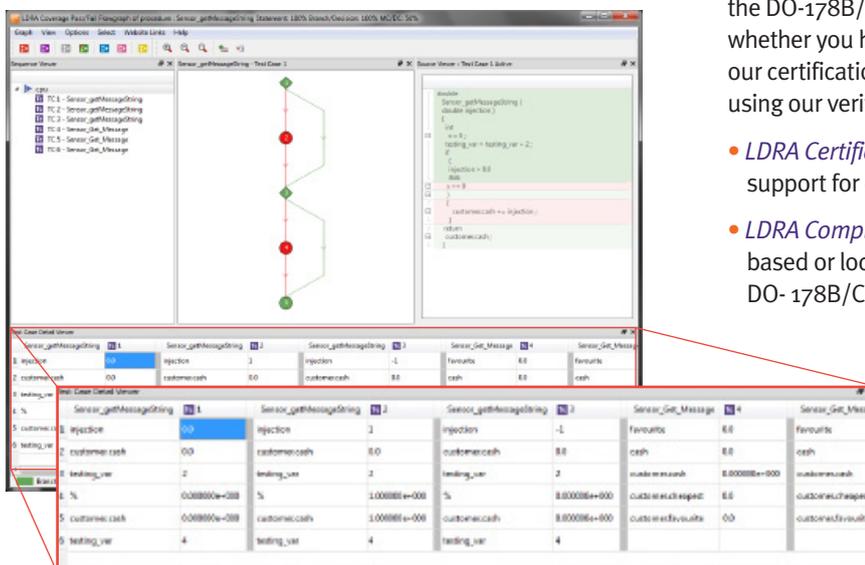
- **LDRA Testbed**[®] and **TBvision**[®] provide a graphical static and dynamic analysis tool for code quality metrics and review.
- **Dynamic Data Flow Coverage (DDFC)** is an optional module that examines exactly which variables were used during run-time execution of the application.

8. Certification and Regulatory Support

LDRA offers the first comprehensive and fully compliant FAA/EASA certification services. These services are performed by our on-staff team of certification industry experts, including Level A DERs, who are credentialed across all aviation disciplines.

The **LDRA Certification Services (LCS)** team is also aligned with development and verification specialists who use the LDRA tool suite and other complementary tools to accomplish the activities that get your products ready for certification. In addition, the **LDRA Compliance Management System (LCMS)** helps you develop the infrastructure to become compliant to the DO-178B/C, DO-278A, DO-254 and ISO 26262 standards, whether you have a DER on staff, a consulting DER, or leverage our certification services. Although there is unique value in using our verification products with LCMS, it is not required.

- **LDRA Certification Services**[®] (**LCS**) provide comprehensive support for certification to aviation standards.
- **LDRA Compliance Management System**[®] (**LCMS**) is a cloud-based or locally hosted compliance management server for DO-178B/C, DO-278A, DO-254 and ISO 26262 applicants.



Interactive inspection of test cases displaying all the test inputs and outputs along with a graphical and textual code representation with coverage results overlaid.

LDRA Tool Suite

LDRA Testbed®

LDRA Testbed provides the core static and dynamic analysis engines for the LDRA suite of tools. This technology analyses both host and embedded software in a rigorous and systematic manner. *LDRA Testbed* is the proven technology foundation for supporting all software development efforts requiring certification or formal regulatory approvals.

TBvision®

TBvision presents code standard violations and software flaws in the context of the original source code. The interactive environment, enabling the execution of both static and dynamic analysis on a user-defined scope, allows switching between reported violations, the original source code and any of the *LDRA Testbed* supported coding standards. In so doing *TBvision* clarifies why an issue is being reported and what remedial action is required. Software integrity can also be measured and reported in terms of quality and security, or simply the presence of defects (including dynamic memory errors). *TBvision* identifies the issues that need to be addressed to ensure that a software project meets its objectives.

TBrun®

TBrun provides a GUI-driven interface for creating unit and module test cases for either host or target-based software. Test harnesses are automatically generated and *TBrun*, also supports the creation of stubs for code that is outside the scope of the tests. Structural coverage analysis support can also be included in the test harness, helping to ensure that the defined test cases execute 100% of the code under test. Formal test reports are produced and the generated test harnesses serve as a regression test environment for future development phases.

TBreq®/TBmanager®

TBreq supports the tracing of requirements through the software development lifecycle. *TBmanager* adds a task-oriented interface providing a common user experience across all LDRA tool suite modules and across the development team. High-level requirements captured in standard office documents or third party RTM solutions, such as DOORS or Requisite Pro can be traced through lower level documents to the code itself. The end result is a Requirements Traceability Matrix (RTM) that captures how each high level requirement was traced to the final software product, including the associated verification activities and results.

Standalone Products

LDRARules®

LDRARules leverages *LDRA Testbed* to enforce compliance with industry or user-defined coding standards and provides clear visibility of software flaws that might typically pass through the build and test process to become latent problems. Most industry standard rule sets are supported off-the-shelf with additional compliance checks made available on an as needed basis.

LDRACover®

LDRACover utilises *LDRA Testbed* technology to provide extensive test effectiveness feedback generated through structural coverage analysis reporting facilities. *LDRACover* addresses the rigorous structural coverage objectives of standards such as DO-178B/C (Avionics) up to and including Level A.

LDRAunit®

LDRAunit extends *LDRA Testbed* in a standalone environment by supporting the generation of unit and module test cases for host or target-based software. In addition, users can ensure that the defined test cases execute 100% of the code under test through the application of structural coverage analysis. Formal test reports can be produced and the generated test harnesses serve as a regression test environment for future development phases.

Additional Options

TBmisra®

TBmisra provides coding standards checking against MISRA C:2012, MISRA C:2004, MISRA C:1998, MISRA C++:2008, JSF++ AV, High Integrity C++, DERA and IPA/SEC C.

TBsecure®

The *TBvision* plug-in, *TBsecure* provides secure code standards checking against the Carnegie Mellon Software Engineering Institute (SEI) CERT C secure coding standard and the Common Weakness Enumeration (CWE) dictionary.

TBsafe®

TBsafe incorporates an additional set of high-integrity analysis tools to help rigorously test code to exacting standards such as those required for DO-178B/C, Def Stan 00-56 and IEC 61508. Features include Information Flow Analysis, Dynamic Data Flow Coverage, Modified Condition/Decision Coverage (MC/DC) for DO-178B/C Level A and Exact Semantic Analysis.

TBeXtreme®

TBeXtreme revolutionises unit testing by using the information gathered by *LDRA Testbed* to provide a totally automated solution for test vector generation. *TBeXtreme* eliminates the traditional time and resource problems associated with bottom-up testing.

TBobjectBox®

TBobjectBox provides an Object Code Verification (OCV) capability, as described in DO-178B/C, offering the only direct way to relate code coverage at the source code level with that achieved at the object code level. The tool also provides the mechanism to extend, where necessary, the code coverage at the assembler level.

TBevolve®

The *TBvision* plug-in, *TBevolve*, enables project teams to monitor the impact of code changes on their testing process. As the source code changes *TBevolve* will compare a baseline copy of a system with new versions and will highlight changed source code lines and report on untested source code which affects the overall code coverage analysis.

TBpublish®

TBpublish captures the analysis and test results from the *LDRA tool suite* and publishes the results, via an HTML index, into a self-contained directory for easy navigation and collaborative reference.

TBaudit®

TBaudit offers development and test managers an automatically generated, user-configurable Microsoft Word report which contains the results of the review and testing activities carried out by the development, QA and testing teams using the LDRA tool suite. *TBpublish* is a prerequisite for *TBaudit*.

Target Licence Package (TLP)

The *Target Licence Package (TLP)* provides the right to use and to receive support for the *LDRA tool suite* when it is used for target testing. The *TLP* allows the user to receive assistance from LDRA in configuring the *LDRA tool suite* to interface with a specified embedded tool chain.

Dynamic Data Flow Coverage

An optional module that examines exactly which variables were used during run-time execution of the application.

Tool Qualification Support Pack (TQSP)

Tool qualification is becoming increasingly important if not a requirement in some safety- and security-critical markets. The *TQSP* supports clients through the process of qualifying the *LDRA tool suite* for use as a verification tool in their project environments.

Client Testimonials

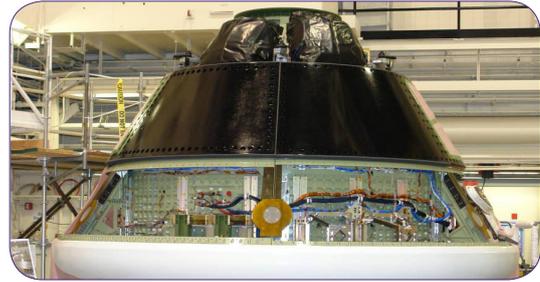
F-35 Lightning II



‘LDRA has proven they will support us in any way to get the job done especially in meeting demanding milestones. They provided outstanding support for several F-35 teammates: Lockheed Martin (Fort Worth), BAE (Warton), Northrop Grumman (El Segundo), Seaweed, and Honeywell which directly contributed to a successful first flight of the AA-1 aircraft. We continue to work closely with LDRA to develop the needed automated process support to ensure that our software meets program cost, schedule, and quality targets.’

John H. Robb, Air Vehicle Software Senior Manager, LMCO

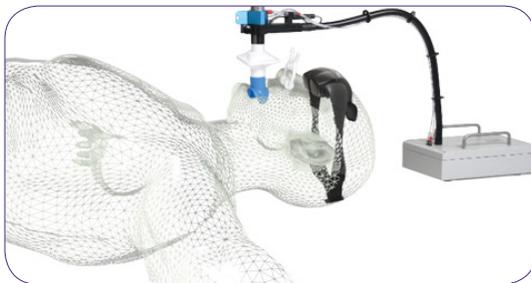
Orion Crew Exploration Vehicle



‘Once the LDRA tool suite has analysed the code, it knows the inputs and outputs, expected returns and types, and every function. So, instead of spending months hand-coding and qualifying a test harness, the tool can use this information to automatically generate a test harness that will call every function, feed data in and out, and test every path of the program.’

Trevor Tidwell, Software Development Engineer, USA

Synchro Module: Radiation therapy system for cancer treatment



‘We configured the LDRA tool suite to meet the company’s specific standard requirements as well as those of IEC 62304. The LDRA tool suite ensures that our products are compliant and that security is worth a premium for customers. Customers can be rest assured that any code we provide has been rigorously reviewed, achieves standards’ compliance, and will be easier to maintain, reuse and add functionality to.’

Frédéric Rabouin, Embedded Software Developer, Esprit Lean

Lexus, LS Sport, Sedan



‘LDRA has the ability to work with limited target hardware which is important in the automotive sector in order to meet the demands for cost reduction and downsizing. We use the LDRA tool suite as a benchmark for other third-party and similar software platform products.’

Akihito Iwai, Project Manager DENSO Japan

Languages & Platforms

The LDRA tool suite is available for the following source code languages and host/target platforms.

Languages

Ada 83	Atmel Assemblers
Ada 95	Freescale Assemblers
C/C++	Intel Assemblers
Java	Texas Instruments Assemblers
	ARM Assemblers

Languages shown in orange signify TBRUN availability

Host Platforms

Windows 7/8/XP	RHEL 6 64 bit
Solaris	Ubuntu 10.10 32 bit
Linux	Ubuntu 12.04 64 bit
RHEL 5 32 bit	

Target Platforms

IDE:	iSYSTEM	Processor:
Analog Devices	Keil	ARM
AONIX	LynuxWorks	Freescale
ARM	Microchip MPLAB X	Infineon
Cosmic	QNX	Intel
Eclipse	TI	Microchip
Freescale	Renesas	MIPS
GNU	TASKING	PowerPC
Green Hills MULTI	Wind River	Renesas
IAR		TI
		Xilinx

Note: This is not an exhaustive list as other languages and host / target platforms are available. Please contact LDRA for more information.



All brand names and product names mentioned herein are trademarks or registered trademarks of their respective companies.
Picture acknowledgements: Laerdal, Sellafield, Lockheed Martin, DENSO Airbus, NASA, Toyota, Esprit Lean, United Space Alliance & SNC

LDRA Ltd. reserves the right to change any specifications contained within this literature without prior notice.

Designed by Young Greenwood Design (01260) 226541.

www.ldra.com

LDRA

LDRA UK & Worldwide

Portside, Monks Ferry, Wirral, CH41 5LH

Tel: +44 (0)151 649 9300

e-mail: info@ldra.com



LDRA Technology, Inc.

2540 King Arthur Blvd, Suite #228

Lewisville Texas 75056

Tel: +1 (855) 855 5372

e-mail: info@ldra.com

LDRA Technology Pvt. Ltd.

Unit B-3, Third floor Tower B, Golden Enclave

HAL Airport Road, Bengaluru 560017

Tel: +91 80 4080 8707

e-mail: india@ldra.com