

 TARGET SOURCE CODE MISRA-C APPLICATION REPORT

Altair Embed® 2025.2

Programming Standards Model MISRA-C:2023

Intellectual Property Rights Notice:

Copyright ©1986-2025 Altair Engineering Inc. All Rights Reserved.

This Intellectual Property Rights Notice is exemplary, and therefore not exhaustive, of the intellectual property rights held by Altair Engineering Inc. or its affiliates. Software, other products, and materials of Altair Engineering Inc. or its affiliates are protected under laws of the United States and laws of other jurisdictions.

In addition to intellectual property rights indicated herein, such software, other products, and materials of Altair Engineering Inc. or its affiliates may be further protected by patents, additional copyrights, additional trademarks, trade secrets, and additional other intellectual property rights. For avoidance of doubt, copyright notice does not imply publication. Copyrights in the below are held by Altair Engineering Inc. or its affiliates. Additionally, all non-Altair marks are the property of their respective owners. If you have any questions regarding trademarks or registrations, please contact marketing and legal.

This Intellectual Property Rights Notice does not give you any right to any product, such as software, or underlying intellectual property rights of Altair Engineering Inc. or its affiliates. Usage, for example, of software of Altair Engineering Inc. or its affiliates is governed by and dependent on a valid license agreement.

Altair® HyperWorks®, a Design & Simulation Platform

Altair® AcuSolve® ©1997-2025

Altair® Activate® ©1989-2025

Altair® Automated Reporting Director™ ©2008-2022

Altair® Battery Damage Identifier™ ©2019-2025

Altair® CFD™ ©1990-2025

Altair Compose® ©2007-2025

Altair® ConnectMe™ ©2014-2025

Altair® DesignAI™ ©2022-2025

Altair® DSim® ©2024-2025

Altair® DSim® Cloud ©2024-2025

Altair® DSim® Cloud CLI ©2024-2025

Altair® DSim® Studio ©2024-2025

Altair® EDEM™ ©2005-2025

Altair® EEvision™ ©2018-2025

Altair® ElectroFlo™ ©1992-2025

Altair Embed® ©1989-2025

Altair Embed® SE ©1989-2025

Altair Embed®/Digital Power Designer ©2012-2025

Altair Embed®/eDrives ©2012-2025

Altair Embed® Viewer ©1996-2025

Altair® e-Motor Director™ ©2019-2025

Altair® ESAComp® ©1992-2025

Altair® expertAI™ ©2020-2025

Altair® Feko® ©1999-2025

Altair® FlightStream® ©2017-2025

Altair® Flow Simulator™ ©2016-2025

Altair® Flux® ©1983-2025

Altair® FluxMotor® ©2017-2025

Altair® GateVision PRO™ ©2002-2025

Altair® Geomechanics Director™ ©2011-2022

Altair® HyperCrash® ©2001-2023

Altair® HyperGraph® ©1995-2025

Altair® HyperLife® ©1990-2025

Altair® HyperMesh® ©1990-2025

Altair® HyperMesh® CFD ©1990-2025

Altair® HyperMesh® NVH ©1990-2025

Altair® HyperSpice™ ©2017-2025

Altair® HyperStudy® ©1999-2025

Altair® HyperView® ©1999-2025

Altair® HyperView Player® ©2022-2025
Altair® HyperWorks® ©1990-2025
Altair® HyperWorks® Design Explorer ©1990-2025
Altair® HyperXtrude® ©1999-2025
Altair® Impact Simulation Director™ ©2010-2022
Altair® Inspire™ ©2009-2025
Altair® Inspire™ Cast ©2011-2025
Altair® Inspire™ Extrude Metal ©1996-2025
Altair® Inspire™ Extrude Polymer ©1996-2025
Altair® Inspire™ Form ©1998-2025
Altair® Inspire™ Mold ©2009-2025
Altair® Inspire™ PolyFoam ©2009-2025
Altair® Inspire™ Print3D ©2021-2025
Altair® Inspire™ Render ©1993-2025
Altair® Inspire™ Studio ©1993-2025
Altair® Material Data Center™ ©2019-2025
Altair® Material Modeler™ ©2019-2025
Altair® Model Mesher Director™ ©2010-2025
Altair® MotionSolve® ©2002-2025
Altair® MotionView® ©1993-2025
Altair® Multi-Disciplinary Optimization Director™ ©2012-2025
Altair® Multiscale Designer® ©2011-2025
Altair® newFASANT™ ©2010-2020
Altair® nanoFluidX® ©2013-2025
Altair® NLView™ ©2018-2025
Altair® NVH Director™ ©2010-2025
Altair® NVH Full Vehicle™ ©2022-2025
Altair® NVH Standard™ ©2022-2025
Altair® OmniV™ ©2015-2025
Altair® OptiStruct® ©1996-2025
Altair® PhysicsAI™ ©2021-2025
Altair® PolEx™ ©2003-2025
Altair® PolEx™ for ECAD ©2003-2025
Altair® PSIM™ ©1994-2025
Altair® Pulse™ ©2020-2025
Altair® Radioss® ©1986-2025
Altair® romAI™ ©2022-2025
Altair® RTLvision PRO™ ©2002-2025
Altair® S-CALC™ ©1995-2025
Altair® S-CONCRETE™ ©1995-2025
Altair® S-FRAME® ©1995-2025
Altair® S-FOUNDATION™ ©1995-2025
Altair® S-LINE™ ©1995-2025
Altair® S-PAD™ © 1995-2025
Altair® S-STEEL™ ©1995-2025
Altair® S-TIMBER™ ©1995-2025
Altair® S-VIEW™ ©1995-2025
Altair® SEAM® ©1985-2025
Altair® shapeAI™ ©2021-2025
Altair® signalAI™ ©2020-2025
Altair® Silicon Debug Tools™ ©2018-2025
Altair® SimLab® ©2004-2025
Altair® SimLab® ST ©2019-2025
Altair® SimSolid® ©2015-2025
Altair® SpiceVision PRO™ ©2002-2025
Altair® Squeak and Rattle Director™ ©2012-2025
Altair® StarVision PRO™ ©2002-2025
Altair® Structural Office™ ©2022-2025

Altair® Sulis™ ©2018-2025
Altair® Twin Activate® ©1989-2025
Altair® UDE™ ©2015-2025
Altair® ultraFluidX® ©2010-2025
Altair® Virtual Gauge Director™ ©2012-2025
Altair® Virtual Wind Tunnel™ ©2012-2025
Altair® Weight Analytics™ ©2013-2022
Altair® Weld Certification Director™ ©2014-2025
Altair® WinProp™ ©2000-2025
Altair® WRAP™ ©1998-2025

Altair® HPCWorks®, a HPC & Cloud Platform

Altair® Allocator™ ©1995-2025
Altair® Access™ ©2008-2025
Altair® Accelerator™ ©1995-2025
Altair® Accelerator™ Plus ©1995-2025
Altair® Breeze™ ©2022-2025
Altair® Cassini™ ©2015-2025
Altair® Control™ ©2008-2025
Altair® Desktop Software Usage Analytics™ (DSUA) ©2022-2025
Altair® FlowTracer™ ©1995-2025
Altair® Grid Engine® ©2001, 2011-2025
Altair® InsightPro™ ©2023-2025
Altair® InsightPro™ for License Analytics ©2023-2025
Altair® Hero™ ©1995-2025
Altair® Liquid Scheduling™ ©2023-2025
Altair® Mistral™ ©2022-2025
Altair® Monitor™ ©1995-2025
Altair® NavOps® ©2022-2025
Altair® PBS Professional® ©1994-2025
Altair® PBS Works™ ©2022-2025
Altair® Simulation Cloud Suite (SCS) ©2024-2025
Altair® Software Asset Optimization (SAO) ©2007-2025
Altair® Unlimited™ ©2022-2025
Altair® Unlimited Data Analytics Appliance™ ©2022-2025
Altair® Unlimited Virtual Appliance™ ©2022-2025

Altair® RapidMiner®, a Data Analytics & AI Platform

Altair® AI Hub ©2023-2025
Altair® AI Edge™ ©2023-2025
Altair® AI Cloud ©2022-2025
Altair® AI Studio ©2023-2025
Altair® Analytics Workbench™ ©2002-2025
Altair® Graph Lakehouse™ ©2013-2025
Altair® Graph Studio™ ©2007-2025
Altair® Knowledge Hub™ ©2017-2025
Altair® Knowledge Studio® ©1994-2025
Altair® Knowledge Studio® for Apache Spark ©1994-2025
Altair® Knowledge Seeker™ ©1994-2025
Altair® IoT Studio™ ©2002-2025
Altair® Monarch® ©1996-2025
Altair® Monarch® Classic ©1996-2025
Altair® Monarch® Complete™ ©1996-2025
Altair® Monarch® Data Prep Studio ©2015-2025
Altair® Monarch Server™ ©1996-2025
Altair® Panopticon™ ©2004-2025



Altair® Panopticon™ BI ©2011-2025
Altair® SLC™ ©2002-2025
Altair® SLC Hub™ ©2002-2025
Altair® SmartWorks™ ©2002-2025
Altair® RapidMiner® ©2001-2025

Altair One® ©1994-2025
Altair® CoPilot™ ©2023-2025
Altair® Drive™ ©2023-2025
Altair® License Utility™ ©2010-2025
Altair® TheaRender® ©2010-2025
OpenMatrix™ ©2007-2025
OpenPBS® ©1994-2025
OpenRadioss™ ©1986-2025

Contents

Introduction	1
Altair Embed MISRA-C 2023 Policy	2
MISRA-C:2023 Standards Model for C / C++	2
MISRA-C 2023 Implemented Guidelines	3
MISRA-C 2023 Partially-Checked Guidelines	13
MISRA-C 2023 Blanket Deviations	14
MISRA-C 2023 Case-by-Case Deviations	17
References	26

Introduction

The MISRA-C standard is a set of coding guidelines intended to improve the safety, security, portability, and reliability of software written in the C programming language. These guidelines:

- Advise developers to refrain from using the undefined and unspecified aspects of the C language.
- Minimize risks if developers misunderstand or misuse the language specifications.
- Identify potential runtime errors that are not flagged by the compiler.

MISRA-C standard is developed by the Motor Industry Software Reliability Association (MISRA). The standard is adopted for embedded software across automotive, industrial, medical, aerospace, and other safety-critical fields. There are four MISRA-C versions to date: MISRA-C:1998, MISRA-C:2004, MISRA-C:2012 and MISRA-C:2023. MISRA-C:2023 supports both C99 and C90 and C11/C18. It adds 200 new rules and 21 directives, with 221 guidelines total. In this application report, MISRA-C refers to MISRA-C:2023 version.

A guideline is classified as either a rule or a directive. The classification is denoted by the letter R or D in the corresponding guideline number. A rule requires only the source code to verify compliance, while a directive requires the source code and may also require additional inputs, like the design documentation, to verify compliance.

There are three guideline categories: mandatory, required, and advisory. The mandatory guidelines must always be followed for a project to meet MISRA-C compliance. The required guidelines should be complied with unless the developer documents a formal justification. The advisory guidelines are suggestions to be implemented if practical, and deviations must be documented.

This report lists the MISRA-C guidelines adhered to by MISRA-C compliant Altair Embed software. It also details the justifications for approved deviations from the MISRA-C 2023 standard. This report also covers Embed code generation, MCU tested legacy source code, and associated header files.

Note: Reporting Scope covers 160 C files, which have approximately 145K lines of code and 161 C-code-generatable Embed blocks. In addition to the static file check, we tested multiple Embed application diagrams, which cover inline code generation and source code VSRTOS library links. The following is a partial list of Embed application diagrams that have more than 500 Embed blocks:

- HVAC rooftop control and equipment models and conditioned space models (17K Embed blocks)
- FOC sensor and sensor control of PMSM motor (1086 Embed blocks)
- Kalman filter to estimate acceleration Q matrix tuning (1197 Embed Blocks)
- PACE test diagram – Hitachi v0p138 (13K Embed Blocks)
- Quadcopter diagram with XYZ pitch-roll-yaw controllers and MPC (829 Embed Blocks)
- VisSimEKF with t-vector adjustment (1857 Embed Blocks)
- AFC – master diagram (1191 Embed Blocks)
- Exercise machine diagram (4668 Embed Blocks)

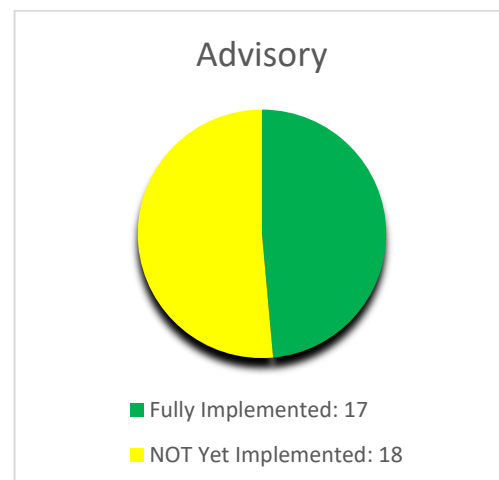
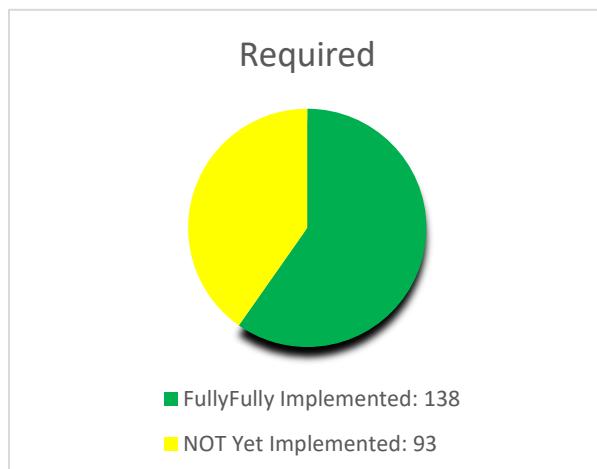
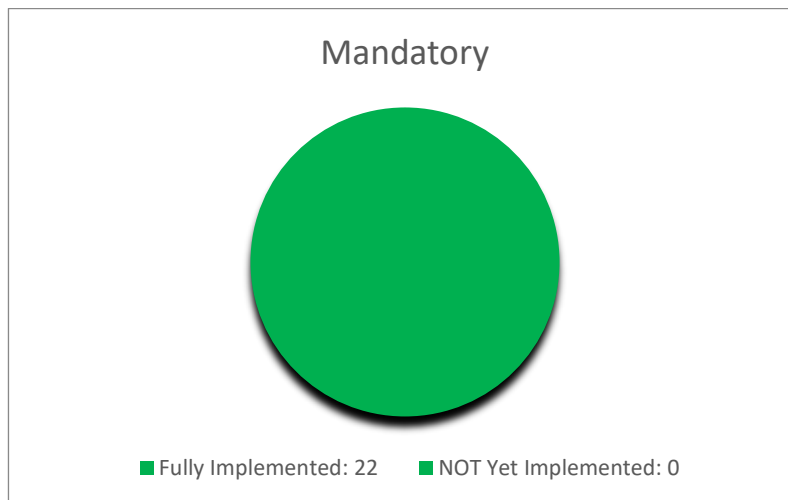
Detailed reports are available on demand.

Altair Embed MISRA-C 2023 Policy

MISRA-C:2023 Standards Model for C / C++

LDRA Version	Embed Version	Report Produced Date	Reporting Scope
10.2.1	2024	April 2024	Combined test of all the VSRTOS source code and large beefy diagrams with many blocks.

Classification	Enhanced Enforcement	Fully Implemented	Partially Implemented	Not Yet Implemented	Not Statically Checkable	Total
Mandatory	0	22	0	0	0	22
Required	0	138	1	93	0	232
Advisory	0	17	0	18	0	35
Total	0	177	1	111	0	289



The policy is divided into four sections:

- Implemented guidelines.
- Partially checked guidelines.
- Blanket deviations
- Case-by-case deviations

The MISRA Implemented guidelines are compliant to by Embed. The partially checked guidelines, blanket deviations, and case-by-case deviations provide justification for deviations.

MISRA-C 2023 Implemented Guidelines

Table 1 lists the MISRA-C guidelines that are Implemented to by Embed. If the static analysis reports indicate violations to any of these guidelines, then the code is rewritten, and static analysis is repeated until the violations are removed. As previously mentioned, mandatory guidelines cannot be violated in MISRA-C compliant software. The list of Implemented guidelines includes all the guidelines categorized as mandatory by MISRA-C, as well as the majority of required and advisory guidelines.

Table 1. Implemented Guidelines

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Status
MISRA C:2023 R.17.4	36 S	Function has no return statement.	Mandatory	DONE
MISRA C:2023 R.17.4	66 S	Function with empty return expression.	Mandatory	DONE
MISRA C:2023 R.12.5	401 S	Use of sizeof on an array parameter.	Mandatory	DONE
MISRA C:2023 R.22.2	407 S	free used on string.	Mandatory	DONE
MISRA C:2023 R.19.1	480 S	String function params access same variable.	Mandatory	DONE
MISRA C:2023 R.22.2	483 S	Freed parameter is not heap item.	Mandatory	DONE
MISRA C:2023 R.22.2	484 S	Attempt to use already freed object.	Mandatory	DONE
MISRA C:2023 R.1.3,R.21.17,R.21.18	489 S	Insufficient space for operation.	Mandatory	DONE
MISRA C:2023 R.17.3	496 S	Function call with no prior declaration.	Mandatory	DONE

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Status
MISRA C:2023 R.19.1	545 S	Assignment of overlapping storage.	Mandatory	DONE
MISRA C:2023 R.22.5	591 S	Inappropriate use of file pointer.	Mandatory	DONE
MISRA C:2023 R.21.17	600 S	Argument of strlen is unterminated.	Mandatory	DONE
MISRA C:2023 R.17.6	614 S	Use of static keyword in array parameter.	Mandatory	DONE
MISRA C:2023 R.2.1,R.9.1	631 S	Declaration not reachable.	Mandatory	DONE
MISRA C:2023 R.22.2	644 S	realloc ptr does not originate from allocation function.	Mandatory	DONE
MISRA C:2023 R.19.1	647 S	Overlapping data items in memcpy.	Mandatory	DONE
MISRA C:2023 R.9.1	652 S	Object created by malloc used before initialisation.	Mandatory	DONE
MISRA C:2023 R.21.13	663 S	Invalid value may be passed to function in <ctype.h>.	Mandatory	DONE
MISRA C:2023 R.7.5	711 S	Inappropriate argument to integer-constant macro.	Mandatory	DONE
MISRA C:2023 R.17.9	34 V	Function declared with _Noreturn has return statement.	Mandatory	DONE
MISRA C:2023 R.17.9	35 V	Function declared with _Noreturn may return on some paths.	Mandatory	DONE
MISRA C:2023 R.18.10	54 V	Parameter is pointer to a variably-modified array type.	Mandatory	DONE
MISRA C:2023 R.5.8,R.5.9	1 S	Procedure name reused.	Required	DONE
MISRA C:2023 R.8.1	20 S	Parameter not declared explicitly.	Required	DONE
MISRA C:2023 R.1.1,R.1.3	21 S	Number of parameters does not match.	Required	DONE
MISRA C:2023 R.21.4	43 S	Use of setjmp/longjmp.	Required	DONE

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Status
MISRA C:2023 R.18.1	47 S	Array bound exceeded.	Required	DONE
MISRA C:2023 R.13.6	54 S	Sizeof operator with side effects.	Required	DONE
MISRA C:2023 R.2.2	57 S	Statement with no side effect.	Required	DONE
MISRA C:2023 R.16.6	60 S	Empty switch statement.	Required	DONE
MISRA C:2023 R.16.6	61 S	Switch contains default only.	Required	DONE
MISRA C:2023 R.1.3	64 S	Void procedure used in expression.	Required	DONE
MISRA C:2023 R.1.3	65 S	Void variable passed as parameter.	Required	DONE
MISRA C:2023 R.6.2	72 S	Signed bit field less than 2 bits wide.	Required	DONE
MISRA C:2023 R.6.1	73 S	Bit field not signed or unsigned int.	Required	DONE
MISRA C:2023 R.7.1	83 S	Octal number found.	Required	DONE
MISRA C:2023 R.20.4,R.21.1	86 S	Attempt to define reserved word.	Required	DONE
MISRA C:2023 D.4.3	88 S	Procedure is not pure assembler.	Required	DONE
MISRA C:2023 R.5.3	92 S	Duplicate use of a name in an enumeration.	Required	DONE
MISRA C:2023 R.20.2	100 S	#include filename is non conformant.	Required	DONE
MISRA C:2023 R.8.4	102 S	Function and prototype return inconsistent (MR).	Required	DONE
MISRA C:2023 R.8.4	103 S	Function and prototype param inconsistent (MR).	Required	DONE
MISRA C:2023 R.5.6	112 S	Typedef name redeclared.	Required	DONE

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Status
MISRA C:2023 R.1.3	113 S	Non standard character in source.	Required	DONE
MISRA C:2023 R.16.7	121 S	Use of boolean expression in switch.	Required	DONE
MISRA C:2023 R.21.8	122 S	Use of abort, exit, etc.	Required	DONE
MISRA C:2023 R.20.14	126 S	A #if has no #endif in the same file.	Required	DONE
MISRA C:2023 R.8.11,R.9.5	127 S	Array has no bounds specified.	Required	DONE
MISRA C:2023 R.5.3	128 S	Parameter has same name as global variable.	Required	DONE
MISRA C:2023 R.14.3	140 S	Infeasible loop condition found.	Required	DONE
MISRA C:2023 R.1.1	145 S	#if has invalid expression.	Required	DONE
MISRA C:2023 R.20.13	147 S	Spurious characters after preprocessor directive.	Required	DONE
MISRA C:2023 R.21.1	156 S	Use of 'defined' keyword in macro body.	Required	DONE
MISRA C:2023 R.1.3,R.4.1	176 S	Non standard escape sequence in source.	Required	DONE
MISRA C:2023 R.11.8	203 S	Cast on a constant value.	Required	DONE
MISRA C:2023 R.16.2	245 S	Case statement in nested block.	Required	DONE
MISRA C:2023 D.4.1,D.4.14	248 S	Divide by zero in preprocessor directive.	Required	DONE
MISRA C:2023 R.10.1	249 S	Shift operator with boolean operand.	Required	DONE
MISRA C:2023 R.7.3	252 S	Lower case suffix to literal number.	Required	DONE
MISRA C:2023 R.10.3	276 S	Case is not part of switch enumeration.	Required	DONE

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Status
MISRA C:2023 R.1.3	296 S	Function declared at block scope.	Required	DONE
MISRA C:2023 R.16.5	322 S	Default is not last case of switch.	Required	DONE
MISRA C:2023 R.1.1	323 S	Switch has more than one default case.	Required	DONE
MISRA C:2023 R.5.7	325 S	Inconsistent use of tag.	Required	DONE
MISRA C:2023 R.8.1	326 S	Declaration is missing type.	Required	DONE
MISRA C:2023 R.10.8	333 S	Widening cast on complex float expression.	Required	DONE
MISRA C:2023 R.1.3	335 S	Operator defined contains illegal items.	Required	DONE
MISRA C:2023 R.1.3	336 S	#if expansion contains define operator.	Required	DONE
MISRA C:2023 R.20.6	341 S	Preprocessor construct as macro parameter.	Required	DONE
MISRA C:2023 R.20.13	342 S	Extra chars after preprocessor directive.	Required	DONE
MISRA C:2023 R.20.14	343 S	#else has no #if, etc in the same file.	Required	DONE
MISRA C:2023 R.1.1,R.10.1	345 S	Bit operator with floating point operand.	Required	DONE
MISRA C:2023 R.5.6	374 S	Name conflict with typedef.	Required	DONE
MISRA C:2023 R.5.5	383 S	Identifier name matches macro name.	Required	DONE
MISRA C:2023 R.5.4,R.5.5	384 S	Identifier matches macro name up to max significant chars.	Required	DONE
MISRA C:2023 R.16.1	385 S	MISRA switch statement syntax violation.	Required	DONE
MISRA C:2023 R.1.1	387 S	Enum init not integer-constant-expression.	Required	DONE

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Status
MISRA C:2023 R.10.1	389 S	Bool value incremented/decremented.	Required	DONE
MISRA C:2023 R.1.1	404 S	Array initialisation has too many items.	Required	DONE
MISRA C:2023 R.16.4	410 S	Switch empty default has no comment (MR).	Required	DONE
MISRA C:2023 R.10.3	411 S	Inappropriate value assigned to enum.	Required	DONE
MISRA C:2023 R.1.3	412 S	Undefined behaviour, \ before E-O-F.	Required	DONE
MISRA C:2023 R.20.3	427 S	Filename in #include not in < > or " ".	Required	DONE
MISRA C:2023 R.15.6	428 S	No {} for switch.	Required	DONE
MISRA C:2023 R.14.2	429 S	Empty middle expression in for loop.	Required	DONE
MISRA C:2023 R.18.3	437 S	< > <= >= used on different object pointers.	Required	DONE
MISRA C:2023 R.18.2	438 S	Pointer subtraction not addressing one array.	Required	DONE
MISRA C:2023 R.1.3	450 S	Wide string and string concatenated.	Required	DONE
MISRA C:2023 R.8.8	461 S	Identifier with ambiguous linkage.	Required	DONE
MISRA C:2023 R.1.3	465 S	Struct/union not completely specified.	Required	DONE
MISRA C:2023 R.15.7	477 S	Empty else clause following else if.	Required	DONE
MISRA C:2023 R.1.1,R.18.7	481 S	Array with no bounds in struct.	Required	DONE
MISRA C:2023 R.1.3	482 S	Incomplete structure referenced.	Required	DONE
MISRA C:2023 R.1.3	486 S	Incorrect number of formats in output function.	Required	DONE

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Status
MISRA C:2023 R.1.3	487 S	Insufficient space allocated.	Required	DONE
MISRA C:2023 D.4.1,D.4.14,R.12.4	493 S	Numeric overflow.	Required	DONE
MISRA C:2023 R.1.3	497 S	Type is incomplete in translation unit.	Required	DONE
MISRA C:2023 R.10.1	506 S	Use of boolean with relational operator.	Required	DONE
MISRA C:2023 R.15.2	509 S	goto label is backwards.	Required	DONE
MISRA C:2023 R.15.3	511 S	Jump into nested block.	Required	DONE
MISRA C:2023 R.8.8	553 S	Function and proto should both be static.	Required	DONE
MISRA C:2023 R.18.6	565 S	Assignment to wider scope.	Required	DONE
MISRA C:2023 R.1.3	573 S	Macro concatenation of uni char names.	Required	DONE
MISRA C:2023 R.1.5,R.8.8	575 S	Linkage differs from previous declaration.	Required	DONE
MISRA C:2023 R.1.3	576 S	Function pointer is of wrong type.	Required	DONE
MISRA C:2023 R.1.3	582 S	const object reassigned.	Required	DONE
MISRA C:2023 R.1.3	587 S	Const local variable not immediately initialised.	Required	DONE
MISRA C:2023 R.21.21	588 S	Use of system function.	Required	DONE
MISRA C:2023 R.1.3	589 S	Format is not appropriate type.	Required	DONE
MISRA C:2023 R.1.3	590 S	Mode fault in fopen.	Required	DONE
MISRA C:2023 R.11.1	606 S	Cast involving function pointer.	Required	DONE

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Status
MISRA C:2023 R.1.3	608 S	Use of explicitly undefined language feature.	Required	DONE
MISRA C:2023 R.3.2	611 S	Line splice used in // comment.	Required	DONE
MISRA C:2023 R.8.10	612 S	inline function should be declared static.	Required	DONE
MISRA C:2023 R.8.14	613 S	Use of restrict keyword.	Required	DONE
MISRA C:2023 R.1.1	615 S	Conditional operator has incompatible types.	Required	DONE
MISRA C:2023 R.20.8	616 S	Preprocessor result not 0 or 1.	Required	DONE
MISRA C:2023 R.21.16	618 S	Use of memcmp between structures.	Required	DONE
MISRA C:2023 R.9.4	620 S	Initialisation designator duplicated.	Required	DONE
MISRA C:2023 R.18.8	621 S	Variable-length array declared.	Required	DONE
MISRA C:2023 R.5.4	622 S	Macro parameters are not unique within limits.	Required	DONE
MISRA C:2023 R.7.4	623 S	String assigned to non const object.	Required	DONE
MISRA C:2023 R.20.4	626 S	#define of keyword.	Required	DONE
MISRA C:2023 R.9.4	627 S	Initialiser both positional and designational.	Required	DONE
MISRA C:2023 R.8.12	630 S	Duplicated enumeration value.	Required	DONE
MISRA C:2023 R.11.6,R.11.7	635 S	Cast from pointer to float type.	Required	DONE
MISRA C:2023 R.11.6,R.11.7	636 S	Cast from float type to pointer.	Required	DONE
MISRA C:2023 R.20.11	637 S	# operand followed by ##.	Required	DONE

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Status
MISRA C:2023 R.1.3	642 S	Function return type with array field.	Required	DONE
MISRA C:2023 R.1.3	645 S	realloc ptr type does not match target type.	Required	DONE
MISRA C:2023 R.1.1	646 S	Struct initialisation has too many items.	Required	DONE
MISRA C:2023 R.1.3	649 S	Use of unallocated flexible array.	Required	DONE
MISRA C:2023 R.1.3	650 S	Flexible array copy ignores last member.	Required	DONE
MISRA C:2023 R.21.14	661 S	memcmp used to compare null terminated strings.	Required	DONE
MISRA C:2023 R.22.7	662 S	EOF compared with char.	Required	DONE
MISRA C:2023 R.1.5	678 S	Type specifier before non-type specifier.	Required	DONE
MISRA C:2023 R.10.1	706 S	++ or -- with _Complex operand.	Required	DONE
MISRA C:2023 R.10.1	707 S	Relational operator with _Complex operand.	Required	DONE
MISRA C:2023 R.10.3	709 S	_Complex assigned to float without cast.	Required	DONE
MISRA C:2023 R.6.3,R.10.1	710 S	Use of bit field in union declaration.	Required	DONE
MISRA C:2023 R.7.6	712 S	Size of integer-constant macro is less than integer.	Required	DONE
MISRA C:2023 R.2.1	3 J	All internal linkage calls unreachable.	Required	DONE
MISRA C:2023 R.13.1,R.13.2,R.13.5	1 Q	Call has execution order dependant side effects.	Required	DONE
MISRA C:2023 R.17.2	1 U	Inter-file recursion found.	Required	DONE
MISRA C:2023 R.1.4	15 V	Use of banned keyword.	Required	DONE

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Status
MISRA C:2023 R.17.10	32 V	Function declared _Noreturn does not have void return type.	Required	DONE
MISRA C:2023 R.17.13	39 V	Function type is type qualified.	Required	DONE
MISRA C:2023 R.8.15	40 V	Conflicting alignment specifiers in variable declarations.	Required	DONE
MISRA C:2023 R.23.2	42 V	Control expression in generic selection has side effects.	Required	DONE
MISRA C:2023 R.23.8	44 V	Default association not first or last in generic selection.	Required	DONE
MISRA C:2023 R.23.4	45 V	Generic association is not an appropriate type.	Required	DONE
MISRA C:2023 R.11.10	47 V	_Atomic qualifier applied to an incomplete type void.	Required	DONE
MISRA C:2023 D.5.3	48 V	Thread created in a thread entry point.	Required	DONE
MISRA C:2023 R.22.20	49 V	Thread-specific storage pointer created in thread entry.	Required	DONE
MISRA C:2023 R.22.13	50 V	Thread related objects have automatic storage duration.	Required	DONE
MISRA C:2023 R.9.7	51 V	Atomic object not initialised with atomic_init.	Required	DONE
MISRA C:2023 R.12.6	52 V	Direct access of a member of an atomic structure or union.	Required	DONE
MISRA C:2023 R.22.12	53 V	Direct assignment or check of thread type object.	Required	DONE
MISRA C:2023 R.21.26	55 V	Call mtx_timedlock for mutex that does not support timeout.	Required	DONE
MISRA C:2023 R.22.15	56 V	Dependant thread objects destroyed in thread entry points.	Required	DONE
MISRA C:2023 R.20.1	75 S	Executable code before an included file.	Advisory	DONE
MISRA C:2023 R.18.5	80 S	Pointer indirection exceeds 2 levels.	Advisory	DONE

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Status
MISRA C:2023 R.4.2	81 S	Use of trigraph.	Advisory	DONE
MISRA C:2023 R.1.2	110 S	Use of single line comment(s) //.	Advisory	DONE
MISRA C:2023 R.1.2	143 S	Curly brackets used in expression.	Advisory	DONE
MISRA C:2023 R.1.2	293 S	Non ANSI/ISO construct used.	Advisory	DONE
MISRA C:2023 R.20.5	426 S	#undef used in a block.	Advisory	DONE
MISRA C:2023 R.2.6	610 S	Label is unused.	Advisory	DONE
MISRA C:2023 R.1.2	632 S	Use of // comment in pre-processor directive or macro defn.	Advisory	DONE
MISRA C:2023 R.10.5	695 S	Char expression cast to non-char type.	Advisory	DONE
MISRA C:2023 R.17.11	33 V	Function that does not return requires _Noreturn specifier.	Advisory	DONE
MISRA C:2023 R.17.12	36 V	Function identifier used without & or parameter list.	Advisory	DONE
MISRA C:2023 R.8.16	37 V	Variable declaration has alignment specifier of 0.	Advisory	DONE
MISRA C:2023 R.8.17	38 V	Variable declaration has multiple alignment specifiers.	Advisory	DONE
MISRA C:2023 R.23.1	41 V	Use of generic selection applied directly.	Advisory	DONE
MISRA C:2023 R.23.3	43 V	Generic selection contains no non-default association.	Advisory	DONE
MISRA C:2023 R.23.5	46 V	Generic selection depends on implicit ptr type conversion.	Advisory	DONE

MISRA-C 2023 Partially-Checked Guidelines

Table 2 lists the guidelines of MISRA-C that cannot be completely verified through static analysis. Due to tool limitations, the source code is only partially checked for compliance to these guidelines.

Table 2. Partially-Checked Guidelines

MISRA-C:2023 Code Guide Line	LDRA Code	Guideline Description	Category	Reasoning
MISRA C:2023 D.4.1,D.4.14	45 D	Pointer not checked for null before use.	Required	45 D is waived for global pointers and linker-defined pointers, where pointer NULL check is performed once in the beginning of the function instead of multiple checks throughout. NOT Yet Implemented

MISRA-C 2023 Blanket Deviations

Table 3 lists the guidelines of MISRA-C that are always waived. The source code is not checked for compliance to these guidelines. The static analysis tool is configured such that violations to these guidelines are not reported. All of these guidelines are categorized as advisory by the MISRA-C standard.

Table 3. Blanket Deviations

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Reasoning
MISRA C:2023 R.15.1	13 S	goto detected.	Advisory	Hardware Tested legacy code base. This guideline is disabled in the tool. NOT Yet implemented
MISRA C:2023 R.12.1	49 S	Logical conjunctions need brackets.	Advisory	Hardware Tested legacy code base. This guideline is disabled in the tool. NOT Yet implemented
MISRA C:2023 R.12.3	53 S	Use of comma operator.	Advisory	Hardware Tested legacy code base. This guideline is disabled in the tool. NOT Yet implemented
MISRA C:2023 R.19.2	74 S	Union declared.	Advisory	Hardware Tested legacy code base. This guideline is disabled in the tool.

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Reasoning
				NOT Yet implemented
MISRA C:2023 R.18.4	87 S	Use of pointer arithmetic.	Advisory	Hardware Tested legacy code base. This guideline is disabled in the tool. NOT Yet implemented
MISRA C:2023 D.4.6	90 S	Basic type declaration used.	Advisory	Hardware Tested legacy code base. This guideline is disabled in the tool. NOT Yet implemented
MISRA C:2023 R.17.8	149 S	Reference parameter to procedure is reassigned.	Advisory	Hardware Tested legacy code base. This guideline is disabled in the tool. NOT Yet implemented
MISRA C:2023 D.4.5	217 S	Names only differ by case.	Advisory	Hardware Tested legacy code base. This guideline is disabled in the tool. NOT Yet implemented
MISRA C:2023 D.4.4	302 S	Comment possibly contains code.	Advisory	Hardware Tested legacy code base. This guideline is disabled in the tool. NOT Yet implemented
MISRA C:2023 R.20.1	338 S	#include preceded by non preproc directives.	Advisory	Hardware Tested legacy code base. This guideline is disabled in

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Reasoning
				the tool. NOT Yet implemented
MISRA C:2023 D.4.9	340 S	Use of function like macro. (845 additional violations not shown)	Advisory	Hardware Tested legacy code base. This guideline is disabled in the tool. NOT Yet implemented
MISRA C:2023 R.15.4	409 S	More than one break or goto statement in loop.	Advisory	Hardware Tested legacy code base. This guideline is disabled in the tool. NOT Yet implemented
MISRA C:2023 D.4.6	495 S	Typedef name has no size indication.	Advisory	Hardware Tested legacy code base. This guideline is disabled in the tool. NOT Yet implemented
MISRA C:2023 R.11.5	540 S	Cast from pointer to void to pointer.	Advisory	Hardware Tested legacy code base. This guideline is disabled in the tool. NOT Yet implemented
MISRA C:2023 R.2.5	628 S	Macro not used in translation unit.	Advisory	Hardware Tested legacy code base. This guideline is disabled in the tool. NOT Yet implemented
MISRA C:2023 R.10.5	696 S	Bool expression cast to non-bool type.	Advisory	Hardware Tested legacy code base. This guideline

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Reasoning
				is disabled in the tool. NOT Yet implemented
MISRA C:2023 R.8.9	25 D	Scope of variable could be reduced.	Advisory	Hardware Tested legacy code base. This guideline is disabled in the tool. NOT Yet implemented
MISRA C:2023 D.4.8	104 D	Structure implementation not hidden.	Advisory	Hardware Tested legacy code base. This guideline is disabled in the tool. NOT Yet implemented

MISRA-C 2023 Case-by-Case Deviations

Table 4 lists the rules of MISRA-C that are followed when reasonably practical but can be waived in approved situations. Each instance of a violation is reviewed by the software team. The developer determines whether the reasoning documented in this policy applies to the violation. If the deviation is approved by this policy, an inspected comment is placed directly above the violating line of code to suppress the violation.

Table 4. Case-by-Case Deviations

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Reasoning
MISRA C:2023 R.13.1,R.13.2,R.13.4	9 S	Assignment operation in expression.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.15.6	11 S	No brackets to loop body.	Required	NOT Yet implemented
MISRA C:2023 R.15.6	12 S	No brackets to then/else.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Reasoning
MISRA C:2023 R.13.1,R.13.2,R.13.3	30 S	Deprecated usage of ++ or -- operators found.	Required	NOT Yet implemented
MISRA C:2023 R.2.1	35 S	Static procedure is not explicitly called in code analysed.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.8.2	37 S	Procedure parameter has a type but no identifier.	Required	NOT Yet implemented
MISRA C:2023 R.14.1	39 S	Unsuitable type for loop variable.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 D.4.12,R.1.3,R.1.4,R.17.1,R.21.3,R.21.6,R.21.7,R.21.9,R.21.10,R.21.24	44 S	Use of banned function, type or variable.	Required	NOT Yet implemented
MISRA C:2023 R.16.4	48 S	No default case in switch statement.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.10.1	50 S	Use of shift operator on signed type.	Required	NOT Yet implemented
MISRA C:2023 R.12.2	51 S	Shifting value too far.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.10.1	52 S	Unsigned expression negated.	Required	NOT Yet implemented
MISRA C:2023 R.15.7	59 S	Else alternative missing in if.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.16.3	62 S	Switch case not terminated with break.	Required	NOT Yet implemented

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Reasoning
MISRA C:2023 R.1.5,R.8.2	63 S	Empty parameter list to procedure/function.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.1.5,R.20.5	68 S	#undef used.	Required	NOT Yet implemented
MISRA C:2023 R.18.6	71 S	Pointer assignment to wider scope.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.20.7	78 S	Macro parameter not in brackets. (1621 additional violations not shown)	Required	NOT Yet implemented
MISRA C:2023 R.10.3	101 S	Function return type inconsistent.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.10.3	104 S	Struct field initialisation incorrect.	Required	NOT Yet implemented
MISRA C:2023 R.9.2	105 S	Initialisation brace { } fault.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.10.4	107 S	Type mismatch in ternary expression.	Required	NOT Yet implemented
MISRA C:2023 R.10.1	109 S	Array subscript is not integral.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.10.1,R.14.4	114 S	Expression is not Boolean.	Required	NOT Yet implemented
MISRA C:2023 R.1.3	118 S	main must be int (void) or int (int,char*[]).	Required	Hardware Tested legacy code base. This guideline

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Reasoning
				is disabled in the tool.
MISRA C:2023 R.3.1	119 S	Nested comment found.	Required	NOT Yet implemented
MISRA C:2023 R.10.1	120 S	Use of bit operator on signed type.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.10.1,R.10.3,R.10.4,R.10.5	123 S	Use of underlying enum representation value.	Required	NOT Yet implemented
MISRA C:2023 R.20.10,R.20.12	125 S	Use of ## or # in a macro.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.1.4,R.17.1,R.21.5,R.21.6,R.21.10,R.21.11,R.21.12	130 S	Included file is not permitted.	Required	NOT Yet implemented
MISRA C:2023 R.5.3	131 S	Name reused in inner scope.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.13.1,R.13.4	132 S	Assignment operator in boolean expression.	Required	NOT Yet implemented
MISRA C:2023 R.13.1,R.13.2	134 S	Volatile variable in complex expression.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.1.5,R.8.1,R.8.2	135 S	Parameter list is KR.	Required	NOT Yet implemented
MISRA C:2023 R.10.1	136 S	Bit operator with boolean operand.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Reasoning
MISRA C:2023 R.14.3	139 S	Construct leads to infeasible code.	Required	NOT Yet implemented
MISRA C:2023 R.7.4	157 S	Modification of string literal.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.8.5	172 S	Variable declared multiply.	Required	NOT Yet implemented
MISRA C:2023 R.21.2	218 S	Name is used in standard libraries.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.21.1,R.21.2	219 S	User name starts with underscore. (846 additional violations not shown)	Required	NOT Yet implemented
MISRA C:2023 D.4.10	243 S	Included file not protected with #define.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.14.2	270 S	For loop initialisation is not simple.	Required	NOT Yet implemented
MISRA C:2023 R.14.2	271 S	For loop incrementation is not simple.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.1.3	324 S	Macro call has wrong number of parameters.	Required	NOT Yet implemented
MISRA C:2023 R.10.1,R.10.2	329 S	Operation not appropriate to plain char.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.10.3,R.10.4	331 S	Literal value requires a U suffix.	Required	NOT Yet implemented

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Reasoning
MISRA C:2023 R.10.8	332 S	Widening cast on complex integer expression.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.20.9	337 S	Undefined macro variable in #if.	Required	NOT Yet implemented
MISRA C:2023 R.11.8	344 S	Cast on volatile value.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.12.1,R.20.7	361 S	Expression needs brackets.	Required	NOT Yet implemented
MISRA C:2023 R.17.7	382 S	(void) missing for discarded return value.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.9.3	397 S	Array initialisation has insufficient items.	Required	NOT Yet implemented
MISRA C:2023 R.10.1,R.12.2	403 S	Negative (or potentially negative) shift.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.13.5	406 S	Use of ++ or -- on RHS of && or operator.	Required	NOT Yet implemented
MISRA C:2023 R.13.5	408 S	Volatile variable accessed on RHS of && or .	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.2.3,R.2.4,R.2.8	413 S	User type declared but not used in code analysed.	Required	NOT Yet implemented
MISRA C:2023 R.14.2	430 S	Inconsistent usage of loop control variable.	Required	Hardware Tested legacy code base. This guideline is disabled in

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Reasoning
				the tool.
MISRA C:2023 R.10.2,R.10.3,R.10.4	433 S	Type conversion without cast.	Required	NOT Yet implemented
MISRA C:2023 R.10.3,R.10.4	434 S	Signed/unsigned conversion without cast.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.10.3,R.10.4	435 S	Float/integer conversion without cast.	Required	NOT Yet implemented
MISRA C:2023 R.18.1	436 S	Declaration does not specify an array.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.11.2,R.11.4,R.11.6,R.11.7	439 S	Cast from pointer to integral type.	Required	NOT Yet implemented
MISRA C:2023 R.11.2,R.11.4,R.11.6,R.11.7	440 S	Cast from integral type to pointer.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.10.5,R.10.8	441 S	Float cast to non-float.	Required	NOT Yet implemented
MISRA C:2023 R.10.5,R.10.8	442 S	Signed integral type cast to unsigned.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.10.5,R.10.8	443 S	Unsigned integral type cast to signed.	Required	NOT Yet implemented
MISRA C:2023 R.10.5,R.10.8	444 S	Integral type cast to non-integral.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Reasoning
MISRA C:2023 R.10.3	445 S	Narrower float conversion without cast.	Required	NOT Yet implemented
MISRA C:2023 R.10.3	446 S	Narrower int conversion without cast.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.10.6,R.10.7	451 S	No cast for widening complex float expression (MR).	Required	NOT Yet implemented
MISRA C:2023 R.10.6,R.10.7	452 S	No cast for widening complex int expression (MR).	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.10.3	458 S	Implicit conversion: actual to formal param (MR).	Required	NOT Yet implemented
MISRA C:2023 D.4.14,R.10.3,R.10.4,R.12.4	488 S	Value outside range of underlying type.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.6.1	520 S	Bit field is not bool or explicit integral.	Required	NOT Yet implemented
MISRA C:2023 R.11.9	531 S	Literal zero used in pointer context.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.7.2	550 S	Unsuffix hex or octal is unsigned, add U.	Required	NOT Yet implemented
MISRA C:2023 R.11.2,R.11.3	554 S	Cast to an unrelated type.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.18.1,R.18.4	567 S	Pointer arithmetic is not on array.	Required	NOT Yet implemented

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Reasoning
MISRA C:2023 R.1.1,R.20.4	580 S	Macro redefinition without using #undef.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.14.2	581 S	Loop conditions are independent.	Required	NOT Yet implemented
MISRA C:2023 D.4.1,D.4.14	629 S	Divide by zero found.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.21.15	655 S	Standard library copy/compare objects have different types.	Required	NOT Yet implemented
MISRA C:2023 R.18.1	692 S	Array index is negative.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.1.3,R.11.5	697 S	Pointer converted to different pointer type.	Required	NOT Yet implemented
MISRA C:2023 R.10.1	708 S	Equality operator with floating point operand.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.2.1	76 D	Procedure is not called or referenced in code analysed.	Required	NOT Yet implemented
MISRA C:2023 R.2.2	105 D	DU anomaly dead code, var value is unused on all paths.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.8.4	106 D	No prototype for non-static function.	Required	NOT Yet implemented
MISRA C:2023 D.4.1,D.4.14	128 D	Global pointer not checked within this procedure.	Required	Hardware Tested legacy code base. This guideline is disabled in

MISRA-C:2023 Code Guideline	LDRA Code	Guideline Description	Category	Reasoning
				the tool.
MISRA C:2023 R.2.2	151 D	DD data flow anomaly found.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool. NOT Yet implemented
MISRA C:2023 R.2.1	1 J	Unreachable Code found.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool.
MISRA C:2023 R.1.3	5 Q	File does not end with new line.	Required	NOT Yet implemented
MISRA C:2023 R.5.8,R.5.9	30 X	Identifier reuse: persistent var vs proc param.	Required	Hardware Tested legacy code base. This guideline is disabled in the tool. NOT Yet implemented

References

Liverpool Data Research Associates, *MISRA-C:2023 Standards Model for C/C++*, LDRA Tool Suite Installation Download.