Guidelines on

Certification of Software for Computer Based Control Systems

2019

IRCLASS
Indian Register of Shipping

Index

Topic

Section 1

Introduction

1.1 General

1.1.1 These Guidelines provide requirements for quality assurance certification of computer based control system software for shipboard applications. The requirements focus on the functionality of the software.

1.1.2 The procedures and criteria indicated in these Guidelines are intended for various stakeholders involved in design, development and maintenance of software. The stakeholders include Systems Providers, System Integrators or the Shipyards, Owners, Sub-suppliers, and Subcontractors involved in integrated system software development, implementation, operation, and maintenance.

1.2. Scope

1.2.1 The main objective of these Guidelines is to reduce the software related incidents, which if not addressed can affect the safety of the vessel.

1.2.2 The Guidelines are applicable to software installed in a standalone or integrated computer-based control systems installed on ships or offshore units. Computer-based systems can be associated with a control system of any level of complexity, including those used for propulsion and integrated navigation systems. The complexity and criticality of the software would depend on the possible extent of damage which can be caused by single failure within the system.

1.2.3 The Guidelines provide a framework for procedures and documentation which when implemented would enable various stake holders to have confidence in the developed software product.

1.2.4 It is intended that through the use of the Guidelines the developers consider various scenarios that are normally encountered during installation and operation phase. Examples of such scenarios could be, but not limited to

- ♦ Software compatibility with the installed hardware;
- ♦ Integration with other software;
- ♦ Malfunctions by upgrade/update software;
- ♦ Recurrence of problems due to inadequate feedback and recording;
- ♦ Software updates;
- ♦ Incorrect and/or incomplete manuals;
- ♦ Inadequate instructions

1.3 References

- IRS Rules and Regulations for the Construction and Classification of Steel Ships
- IEC 12207 Systems and software engineering -- Software life cycle processes
- IEC 60092-504
- BS EN ISO 9001: 2000 Quality systems – Model for quality assurance in design, development, production, installation and servicing.

- ISO 90003: Guidelines for the application of ISO 9001 to the development, supply and maintenance of software. Or equivalent
- Industry standard software maintenance of ship board equipment -BIMCO

1.4 Definitions

1.4.1 ***Life cycle model***: The manufacturer's record of processes, activities and tasks involved in the development, operation and maintenance of a product.

1.4.2 ***Major revision***: A software change which affects the functioning of the product and requires significant testing to prove its operation.

1.4.3 ***Module***: A part of software. A series of logical instructions to a computer or similar device written to perform the specific function or sub-functions of the overall system.

1.4.4 ***Verification***: is the process of checking that the software meets the specification.

1.5 Certification

1.5.1 When requested by the software developers, IRS will assess the quality of software in all phases of software life cycle; as an independent verifier and issue software quality assurance certificate (SQAC).

Section 2

Documentation

2.1 General

2.1.1 The procedure for quality assurance certification of Control system software involves the following:

a) Functional design appraisal;

b) Review of software developer quality system documentation;

c) Assessment of implemented quality systems towards software design, development and testing, at development centre;

d) Verification of developed control system software at development centre;

e) Issuance of certificate indicating that the developed control system software is suitable for on board application.

2.1.2 Towards above procedures the developer is to submit various documents indicated in this section for review.

2.2 Quality System

2.2.1 The developer is to submit a Software Quality system plan for review. The plan is to include software life cycle approach. The developing organization is to have valid Quality Certificate confirming ISO 9001 or equivalent. The document is to contains details of procedures adopted towards quality control in design and development of software, internal quality assurance methods for testing, replication, maintenance and configuration management.

2.3 Product Documentation

2.3.1 Following documents are to be submitted towards review of developed software:

a) *Functional description document*:

A brief description of the control system software clearly defining the scope of applicability and specific operational features for which the software is designed.

b) *Software details*:

Following information is to be included in the document

- Description of the basic and communication software. For integrated systems the details of basic and communication software installed in each hardware unit is to be provided.

- Description of application software (not program listings).

- Description of functions, performance, constraints and dependencies between modules or other components.
- A brief description of the minimum hardware configuration required to run the software including interface requirements.

c) Function block diagram showing inputs and outputs from each module. For integrated systems the relation between various modules is to be shown.

2.4 Network Data Information

2.4.1 Following information /data is to be documented:

i) Data and commands controlled by the control system software.

ii) Communication protocol for the control system interfaces. For integrated system, if interfaced with systems with different protocols, the same are to be identified.

iii) Topology drawing showing networked and serial connected equipment, including interface to outside ship connectivity.

2.5 Risk Assessment

2.5.1 A detailed risk assessment is to be carried out where a software is intended to be used for the following purposes:

a. Where computer based systems share common resources, for example, data communication link with any control, alarm or safety system for essential or safety critical systems

b. Where it is intended that computer based systems implement emergency stop or safety critical functions

c. When alternative operation for essential services and safety critical equipment depend on any computer based system.

2.5.2 The risk assessment is to include analysis for safety related functions for standalone systems and the same is to be submitted for review.

2.5.3 For integrated system software, in addition; a detailed failure analysis covering various interfaces and communication failures is to be carried out and the same is to be submitted for review.

2.5.4 The assessment is to demonstrate that for single failures, systems will fail to safety and that systems in operation will not be lost or degraded beyond acceptable performance criteria.

2.6 Software Test Plan

2.6.1 The Test plan is to list communication protocols between various components for an integrated system.

2.6.2 The test plan document as minimum is to contain the following:

- Procedures adopted during software development for testing at module level.
- Test cases are to be documented to include inputs, expected outputs and acceptance/rejection.
- The test plan is to include criteria for determining whether testing as a whole passes or fails.
- Test documents are to define the hardware and software configuration to be used for testing including test tools.
- Any test tools used are to be shown suitable for use.
- For integrated systems, the test program is to include verification methodology for data interfaces with integrated systems

2.7 Documentation and Attendance during Testing

2.7.1 The documentation to be submitted to IRS for information and approval and the required attendance during the testing are listed in Table 2.7.1.

Table 2.7.1 Documentation to be submitted to IRS and test attendance

Sr No.	Documentation/ Tests	Status
1	Quality plan	A
2	Risk assessment report	I
3	Software module functional description	I
4	Procedures of verification of software code , submodule and module levels	I
5	Risk assessment report when software is used for safety critical applications and for integrated systems	A
6	Test program for simulation tests	A
7	Simulation test	W
8	Test program for unit certification where software is verified at developers place	A
9	Unit certification as per test program at Sr. No.8	W
10	- List and version of software installed in system - Functional description of software - User manual including instructions during software maintenance - List of interfaces	I
11	Updated software registry (when revised software is tested at developer place)	I

Note: A- Submitted for Approval; I – Submitted for Information; W – Witness

1. Additional documentation may be required to be submitted upon request.

2. When software along with hardware are tested at developer premises, test reports for hardware as per IRS Classification Notes "*Type approval of electrical equipment used for control, monitoring, alarm and protection systems for use in ships*" are to be submitted for verification to attending surveyor.

Section 3

Software Quality Assurance

3.1 Software Development Life Cycle

3.1.1 A global top to bottom approach is to be followed during design and development of software. The software lifecycle activities are to be defined along with relevant procedures, responsibilities and system documentation, including configuration management.

3.2 Quality System

3.2.1 The Producer of software is to have a quality assurance (QA) system for software lifecycle activities, which documents relevant procedures, responsibilities and configuration management, including deliveries from sub-suppliers, also taking into account cyber-security considerations. A software quality plan giving details as above is to be submitted for review.

3.2.2 The manufacturer of the system is to be able to demonstrate that they have suitable accreditation for their overall quality assurance, to a recognized standard such as ISO 9001. Satisfaction of this requirement may be demonstrated either by:

a) The quality system being certified as compliant to the recognized standard by an organization with accreditation under a national accreditation scheme; or

b) IRS confirming compliance to the standards through a specific assessment

3.2.3 The quality system is to include the following:

a) Procedures regarding responsibilities, system documentation, configuration management and competent staff

b) Procedures regarding software lifecycle:

i. for acquisition of software (Ex. Firmware etc.) from suppliers
ii. for software code writing and verification
iii. for developed software
iv. for software modification and installation on board

3.3 Project Management

3.3.1 The manufacturer of a product is to nominate a designated person, to be wholly responsible for that product until its use has been terminated. The roles and responsibilities of the designated person are to be clearly defined. Interaction with stake holders, working groups during design and development, project review at predefined internals is to be carried out. It is recommended that any feedback received during execution of similar projects during development or in operations be suitably addressed during execution of the current project.

3.4 Approval of Quality System

3.4.1 The developer's quality system will be assessed through review of documentation submitted by the manufacturer and onsite assessment of following requirements as minimum.

Section 4

Software Design

4.1 Software Requirements Specification

4.1.1 The Software Requirements Specification (SRS) document is to:

a. fully specify, either directly or by reference to other submitted documents, all external interfaces between the software product and other software or hardware;

b. state and justify the properties of external devices or software that are relied upon for correct operation;

c. provide means to demonstrate traceability to requirements to ensure that all such requirements can be shown to have been implemented and tested.

4.1.2 The requirements are to be capable of implementation within the stated performance constraints.

4.2 Design and Coding

4.2.1 The software is to be designed and coded in accordance with procedures/standards specified in quality plan.

4.2.2 The software design is to be documented and is to be free from ambiguities, contradictions and other internal inconsistencies.

4.2.3 The software can be divided into hierarchy of modules and other components or sub-assemblies which are functionally distinct and are restricted in size and complexity.

4.2.4 The software design is to describe the functions, performance constraints and the interfaces and dependencies between modules, other components or sub-assemblies.

4.2.5 The software design is to describe the algorithms and data structures used by the modules, other components or sub-assemblies to achieve their function within the stated performance constraints.

4.2.6 The design is to be traceable to the software requirement specification.

4.2.7 It is to be possible to test the design.

4.2.8 The design is to be capable of implementation as specified within the applicable constraints.

4.2.9 The use of any proprietary software is to be identified and documented.

4.2.10 Any design tools, coding tools, compilers and links used are to be shown suitable for use.

Section 5

Software Verification

5.1 General

5.1.1 Verification is concerned with testing the system's functionality and demonstrating that functions meet the specification as detailed in the control systems functional description document.

5.1.2 The verification is to be carried out at software development Centre. The objective of the verification is to assert that the software is designed, developed and tested to through a structured quality system. Successful verification will, lead to issuance of certification.

5.1.3 On successful testing the software version number with its unique abbreviation would be included in the SQAC.

5.2 Verification Process

5.2.1 The functions of all modules, other software components are to be tested at various levels of software development cycle.

5.2.2 The software is to be tested after completion of each of the following stages of development life cycle

- Sub module level
- Module level
- System Level

5.2.3 Sub module level testing is the first level of testing and is often performed by the developers at sub module level by themselves. It is the process of ensuring individual components of a piece of software at the code level are functional and work as they were designed to.

5.2.4 After each sub module is thoroughly tested, it is to be integrated with other units to create modules or components that are designed to perform specific tasks or activities. These are then to be tested at module level.
The above two types of testing are to be carried out by the software developer internal quality team.

5.2.5 For Category II and III systems (as defined in Pt.4, Ch.7 of the *Rules and Regulations for the Construction and Classification of Steel Ships*) a testing team separate from the development team is recommended. The test results are to be documented and to be submitted to attending surveyor when requested.

5.2.6 System testing is a testing method used to evaluate the completed and integrated system, as a whole, to ensure it meets specified requirements. The objective of the system testing is to conduct an end to end functionality test of the developed software. The test is to be carried out at software development center and is to be witnessed by IRS surveyor.

5.3 Verification Test Requirements

5.3.1 The functions of all modules, other software components and sub-assemblies are to be tested.

5.3.2 The performance and timing constraints of each module, other software component or sub-assembly is to be tested.

5.3.3 All interfaces between modules, other software components or sub-assemblies are to be tested.

5.3.4 Tests are to be traceable to the requirements, design and code.

5.3.5 Test cases are to be documented to include inputs, expected outputs and pass/fail criteria.

5.3.6 The modules, other software components and sub-assemblies tested are to be traceable to the software under assessment.

5.3.7 All performance and timing requirements of the software are to be tested. This is to include all modes of operation of the software and its behavior under failure conditions.

5.3.8 Testing is to address normal and extreme operating conditions like normal, zero and saturation loading, and boundary data values.

5.3.9 The features specified in the User documentation/ manuals are to be tested.

5.4 Test Methods

5.4.1 The software developer is to identify one of the following types of verification methods depending on the criticality of the software and owners' specific requirements

a) *Software in Loop* (SIL) testing: for software controlling critical systems Cat II and Cat III systems

b) *Hardware in Loop testing* (HIL): recommended for integrated systems

Note: Where the developed software is tested on the hardware specific to a project as part of factory acceptance test, testing through simulation will not be required. (Example: software installed in main engine, auxiliary engines control panel, etc.).

Where testing cannot be performed on the target operational environment, testing is to be done in an environment that has been shown to be equivalent, and then it is to be supported by a demonstration of successful operation in the target operational environment.

5.4.1.1 Software-In-the-Loop (SIL) testing

- The term software-in-the-loop testing, (SIL) is used to describe a test methodology where executable code such as algorithms (or even an entire controller strategy), usually written for a particular mechatronic/control system, is tested within a modelling environment that can help prove or test the software.

- SIL test is to verify the software runs without errors. It will not verify network connections and input /output hardware connections.

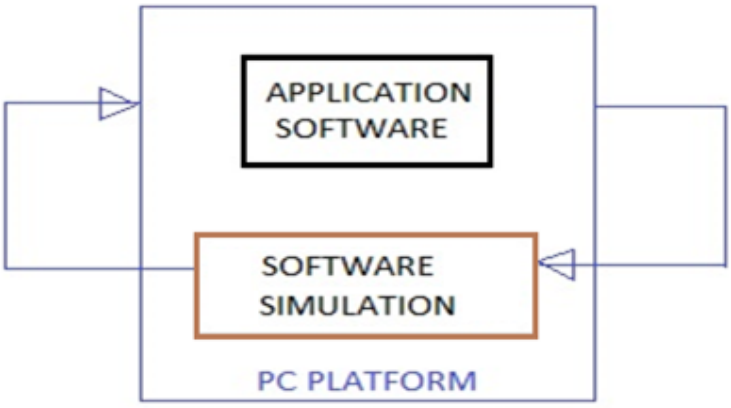

Fig. 5.4.1.1 Software in Loop Testing

5.4.1.2 Hardware-in-the loop testing

- In HIL testing technique a real-time computer is used as a virtual representation of plant model and a real (native) version of controller with software.

- In real-time control system, machine or physical part of the system (the plant) is connected with the control system, through actuators and sensors. With HIL simulation the plant is replaced by a simulation of the plant (the HIL simulator) and will be running on a PC (simulator Computer) and will be physically connected to application computer system (could be a PC/PLC etc). If the HIL simulator is designed well; it will accurately mimic the real plant, and can be used to test the control system.

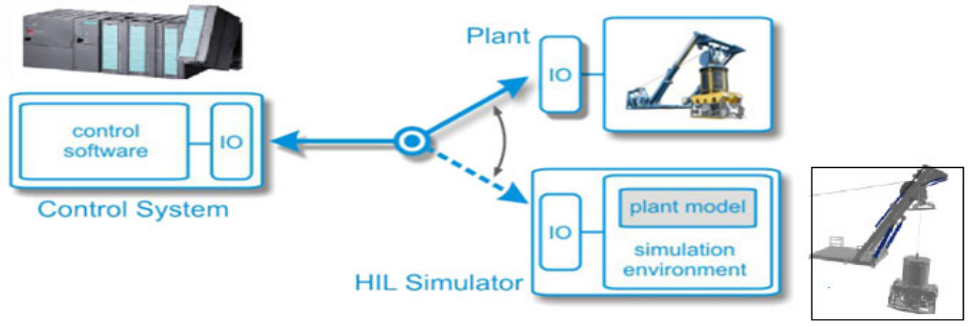

Fig. 5.4.1.2 Hardware in Loop Testing

5.5 Cyber Security Testing of Software

5.5.1 Where cybersecurity testing is required to be demonstrated or security and/or quality of the system against specific threats is to be proved, appropriate tests are to be included to complement the functional verification methods above. Security testing intends to find vulnerabilities of the system and determine that its data and resources are protected from possible intruders. It ensures an integrated software system protects data and maintains functionality as intended.

5.5.2 When cyber security testing is an identified phase within software development lifecycle, it should is to be followed throughout the development process

5.5.3 The software may be tested for various security levels based on manufacturer requirements. The specific test requirements for each level and for various system configurations are to be identified for each product

5.5.4 Following features towards cyber risk management of an integrated system software are to be tested as minimum:

a) Authentication
b) Authorization

5.5.5 Additional features where identified as below may require to be tested

a) cryptography
b) data storage security etc

5.5.7 Test requirements are to be identified based on risk analysis and the test protocol is to be submitted to IRS for approval prior to commencement of testing.

5.6 Proprietary Operating System Software

5.6.1 Proprietary software is to be controlled in the same manner as the manufacturer's development software.

5.6.2 Where the product may be based on generally available proprietary software, the designer is to:

a) ensure prevention of unauthorised tampering;
b) ensure prevention of unauthorised software being loaded in the product;
c) provide the proprietary software at hand-over to enable re-installation in the event of a failure;
d) ensure the full software details are recorded in the ship's asset registry;
e) not supply service packs to the vessel owner/operator unless it is necessary due to system failure.

5.7 Test Records and Results

5.7.1 All specified test cases are to be executed and their results recorded (including whether or not they pass or fail).

5.7.2 The overall testing acceptance/ rejection criteria are to be shown to be satisfactory.

5.7.3 Any outstanding software faults are to be fully documented in the User manual (or some other User accessible documentation).

5.7.4 Test results are to be recorded as defined in the relevant test specification and in a form that permits verification.

5.7.5 The SQAC will be issued on successful completion of testing.

5.8 Unit Certification

5.8.1 Unit certification of software along with hardware would be required for Cat II and Cat III systems. The objective is to check that software functions are properly executed, that the software and the hardware it controls interact and function properly together and that software systems react properly in case of failures. Faults are to be simulated as realistically as possible to demonstrate appropriate system fault detection and system response. The test program is to be submitted to IRS for approval prior to commencement of testing.

Note: Where the software assessment is carried out on project specific hardware, separate unit certification would not be required. During the FAT the hardware type tests reports confirming their suitability for on board installation as per IRS Classification Notes "*Type approval of electrical equipment used for control, monitoring, alarm and protection systems for use in ships*", and project specific and owners' requirements are to be submitted to the attending surveyor for review.

5.8.2 Test programs and procedures for functional and failure tests would be witnessed by IRS. Failure modes as identified in the risk assessment are to be demonstrated.

Section 6

Configuration and Maintenance Management

6.1 Configuration Management

6.1.1 The software product is to be under effective configuration management to support satisfactory replication and delivery. A systematic and disciplined approach to maintenance is to be demonstrated which is to be built on an effective change control system.
Configuration management is the identification, control and tracking the versions of each identifiable part of the software throughout the development and up to termination

6.1.2 A configuration management plan indicating following details /information is to be submitted to IRS. These are to consist of the following, as a minimum:

a. Unique identification of the version of each software item;
b. version details of each software item which constitute a specific version of the complete software product;
c. current build status of the software product;
d. demonstrate traceability of a software item or product to software requirements;
e. modified parts of software items resulting from a change request

6.1.3 The following are to be identified for each version of a software item:

a. the functional and technical specifications;
b. all development tools affecting the functional and technical specifications;
c. all interfaces to other software items and hardware;
d. all documents and computer files related to the software item.

6.1.4 Before a change is accepted, its validity is to be confirmed and the effects on other items are to be identified and examined before the change is authorized. Any additional hardware requirements are to be identified.

6.1.5 Records and summary reports are to be maintained on the status of:

a. software items;
b. change requests;
c. the implementation of approved changes.

6.1.6 Any configuration management tools used are to be shown suitable for use.

6.2 Training

6.2.1 Successful and safe operation of the software is dependent on the training imparted and competence of the user. Depending on the complexity of the software the manufacturer is to provide adequate training to user. This could be in form of videos, instruction for simple software to class room training for advanced integrated software.

6.2.2 The manufacturer is to provide appropriate training for all upgrades and modifications, subsequently carried out on the delivered system.

6.3 Maintenance

6.3.1 The manufacturer is to define and document a maintenance plan for the developed software. The maintenance plan is to define or identify:

a. the scope of the maintenance;
b. the initial status of the software;
c. the facilities and resources for maintenance;
d. maintenance activities;
e. the format and contents of maintenance records and reports.

6.3.2 The initial status of the released software is to be defined and documented.

6.3.3 During maintenance all documentation is to be kept consistent and at the same standard as for the initial release of the software.

6.3.4 Maintenance records for each software item being maintained are to include:

a. the list of problem reports received and their current status;
b. the organization responsible for responding to the reports or implementing corrective action;
c. the priorities assigned to corrective actions;
d. the results of corrective actions;
e. statistical data on failure occurrences and maintenance activities.

6.3.5 Measurements made are to represent reported field failures and defects from the customer's viewpoint.

6.3.6 Maintenance procedures are to include:

a. requirements for records indicating the release status of the software and at what sites the release has been implemented;
b. rules to determine whether localized 'patches' or a complete update of the software is required;
c. the types or classes of release including the version identification policy with respect to minor or major upgrades;
d. methods by which purchasers will be advised of current or planned future changes;
e. methods to confirm that changes implemented will not introduce other problems;