|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | |
|  | | **Document Ref:** |  | | | | |
| **Issue No. :** | <00X> | | **Issue Date :** | | <DD/MM/YYYY> |
| **Copy No. :** | 01 02 03 04 05 | | **No. of Pages :** | | < total no .of pages > |
| **Group :** | < Group Name> | | | | |
| **Document Classification :** | | 🞎Secret 🞎Confidential  🞎Restricted 🞎Unrestricted | | | |
| **Title:** | | | | | | **Project/System :** | |
| **Software Verification Plan**  **of <LRU/Module Name>**  **for <Platform name>** | | | | | | **<System/Project Name>** | |
| **Software Criticality Level :** | |
| DO-178B Level <A/B/C/D> | |
|  | **Name & Designation** | | | | | **Signature** | |
| Prepared By | <Design Rep Name>, < Designation> | | | | |  | |
| Reviewed By | <Project Leader Name>, <Designation>  <AWG/QA HOD Name>, <Designation>  <RCMA Rep Name>, <Designation> | | | | |  | |
| Approved By | < Design Head name>,<Designation>  <RD,RCMA Name>, <Designation> | | | | |  | |
| **<Design Firm Name & Address>** | | | | | | | |

**Document Copy Distribution List**

|  |  |  |
| --- | --- | --- |
| **Copy No.** | **DESIGNATION OF THE COPY HOLDER** | **ORGANISATION/DEPARTMENT** |
| 01 | Design Group Head | Design Agency |
| 02 | RD | RCMA, Hyderabad |
| 03 | HOD | QC/ AWG, Design agency |
| 04 | Coordinator | Customer |

**Amendment History**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Issue**  **No.** | **Issue Date** | **Brief Description of Amendment** | **Change Request Ref.** | **Affected**  **Pages** | **Affected Section** | **Change Effective From**  **(Version/ Date)** |
| 001 |  | Initial Issue | NA | NA | NA | Initial |
| 002 |  |  |  |  |  |  |

**Table of Contents**

[1.0 INTRODUCTION 4](#_Toc445817474)

[1.1 Purpose 4](#_Toc445817475)

[1.2 Scope 4](#_Toc445817476)

[1.3 Acronyms and Abbreviations 4](#_Toc445817477)

[1.4 External Documents 4](#_Toc445817478)

[1.5 Internal Documents 4](#_Toc445817479)

[2.0 Organization: 4](#_Toc445817480)

[3.0 Independence 4](#_Toc445817481)

[4.0 Verification methods 4](#_Toc445817482)

[4.1 Software Planning Process Verification methods 4](#_Toc445817483)

[4.2 Software Requirement Process Verification methods 4](#_Toc445817484)

[4.3 Software Design Process Verification methods 4](#_Toc445817485)

[4.4 Software Coding Process Verification methods 5](#_Toc445817486)

[4.5 Software Testing Process Verification methods 5](#_Toc445817487)

[5.0 Verification environment 5](#_Toc445817489)

[6.0 Transition criteria 5](#_Toc445817490)

[7.0 Partitioning considerations 5](#_Toc445817491)

[8.0 Compiler assumptions 5](#_Toc445817492)

[9.0 Re-verification 5](#_Toc445817493)

[10.0 Previously developed software 6](#_Toc445817494)

[11.0 Multiple-version dissimilar software 6](#_Toc445817495)

# INTRODUCTION

## Purpose

## Scope

## Acronyms and Abbreviations

## External Documents

Include references to standards, manuals, OEM documents etc.

## Internal Documents

# Organization:

Organizational responsibilities within the software verification process and interfaces with the other software life cycle processes.

# Independence

A description of the methods for establishing verification independence, when required.

# Verification methods

A description of the verification methods to be used for each activity of the software verification process.

# Software Planning Process Verification methods

Description of documents/ data that are reviewed and analysed. Cover PSAC, SVP, SCMP, SDP, SQAP. Completeness of the plans and consistency among the plans. Include Checklist(s) for the activity.

# Software Requirement Process Verification methods

Description of documents/ data that are reviewed and analysed. Cover High level requirement correctness and completeness checks, traceability to system requirements, consistency within the requirements etc. Include Checklist(s) for the activity.

# Software Design Process Verification methods

Description of documents/ data that are reviewed and analysed. Cover software architecture, Low level requirement correctness and completeness checks, traceability to High level requirements, algorithm verification methods, etc. Include Checklist(s) for the activity.

# Software Coding Process Verification methods

Description of code that is reviewed and analysed. Cover code walkthrough, traceability to low level requirements, control & data flow checks for determinism, coding standard checks, robustness/ testability/ readability/ maintainability checks, no undocumented function implemented, stack usage, fixed point arithmetic overflow and resolution, resource contention, worst-case execution timing, exception handling, use of uninitialized variables or constants, unused variables or constants, and data corruption due to task or interrupt conflicts etc. Include Checklist(s) for the activity.

# Software Testing Process Verification methods

## Description of test case preparation rationale, regression tests selection method, requirement based coverage, normal and abnormal conditions, external/ internal failure simulations, structural coverage, coverage analysis methods, test data to be produced for low level testing, integration testing, HSI testing, rig testing, aircraft testing etc. Include Checklist(s) for the activity.

# Verification environment

A description of the equipment for testing, the testing and analysis tools, and the guidelines for applying these tools and hardware test equipment, indicating target computer and simulator differences.

# Transition criteria

The transition criteria for entering the software verification process defined in this plan.

# Partitioning considerations

If partitioning is used, the methods used to verify the integrity of the partitioning.

# Compiler assumptions

A description of the assumptions made by the designer about the correctness of the compiler, linkage editor or loader.

# Re-verification

For software modification, a description of the methods for identifying the affected areas of the software and the changed parts of the Executable Object Code. The re-verification should ensure that previously reported errors or classes of errors have been eliminated.

# Previously developed software

For previously developed software, if the initial compliance baseline for the verification process does not comply with this document, a description of the methods to satisfy the objectives of this document.

# Multiple-version dissimilar software

If multiple-version dissimilar software is used, a description of the software verification process activities.