The Engineering Society For Advancing Mobility Land Sea Air and Space®
INTERNATIONAL
400 Commonwealth Drive, Warrendale, PA 15096-0001

AEROSPACE RECOMMENDED PRACTICE

SAE ARP4754

Issued 1996-11

Submitted for recognition as an American National Standard

CERTIFICATION CONSIDERATIONS FOR HIGHLY-INTEGRATED OR COMPLEX AIRCRAFT SYSTEMS

INTRODUCTION

Established industry practices and associated regulatory requirements have developed over many years to ensure that safety standards are maintained in civil aircraft. The increasing integration and complexity of aircraft electronic systems has led to a need to review existing procedures and provide additional guidance to ensure that the proper operation and safety of future systems and system changes can be assured.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

1. SCOPE:

This document discusses the certification aspects of highly-integrated or complex systems installed on aircraft, taking into account the overall aircraft operating environment and functions. The term “highly-integrated” refers to systems that perform or contribute to multiple aircraft-level functions. The term “complex” refers to systems whose safety cannot be shown solely by test and whose logic is difficult to comprehend without the aid of analytical tools.

The guidance material in this document was developed in the context of Federal Aviation Regulations (FAR) and Joint Airworthiness Requirements (JAR) Part 25. It may be applicable to other regulations, such as Parts 23, 27, 29 and 33. In general, this material is also applicable to engine systems and related equipment. Final regulatory approval of all systems is assumed to be accomplished in conjunction with an aircraft certification.

This document has been prepared primarily for electronic systems which, by their nature, may be complex and are readily adaptable to high levels of integration. However, the guidance provided in this document may be considered for other aircraft systems.

This document addresses the total life cycle for systems that implement aircraft-level functions. It excludes specific coverage of detailed systems, software and hardware design processes beyond those of significance in establishing the safety of the implemented system. More detailed coverage of the software aspects of design are dealt with in RTCA document DO-178B and its EUROCAE counterpart, ED-12B. Coverage of complex hardware aspects of design are dealt with in RTCA document DO-xxx, (working title: “Design Assurance Guidance for Airborne Electronic Hardware,”) currently under development by RTCA special committee SC-180. Methodologies for safety assessment processes are outlined in ARP4761. Figure 1 outlines the relationships between the various documents which provide guidance for system development, safety assessment, and the hardware and software life-cycle processes.

This document is intended to be a guide for both the certification authorities and applicants for certification of highly-integrated or complex systems, particularly those with significant software elements. As such, the focus is toward ensuring that safety is adequately assured through the development process and substantiating the safety of the implemented system. Specific guidance on how to do the substantiation work is beyond the scope of this document, though references are provided where applicable.

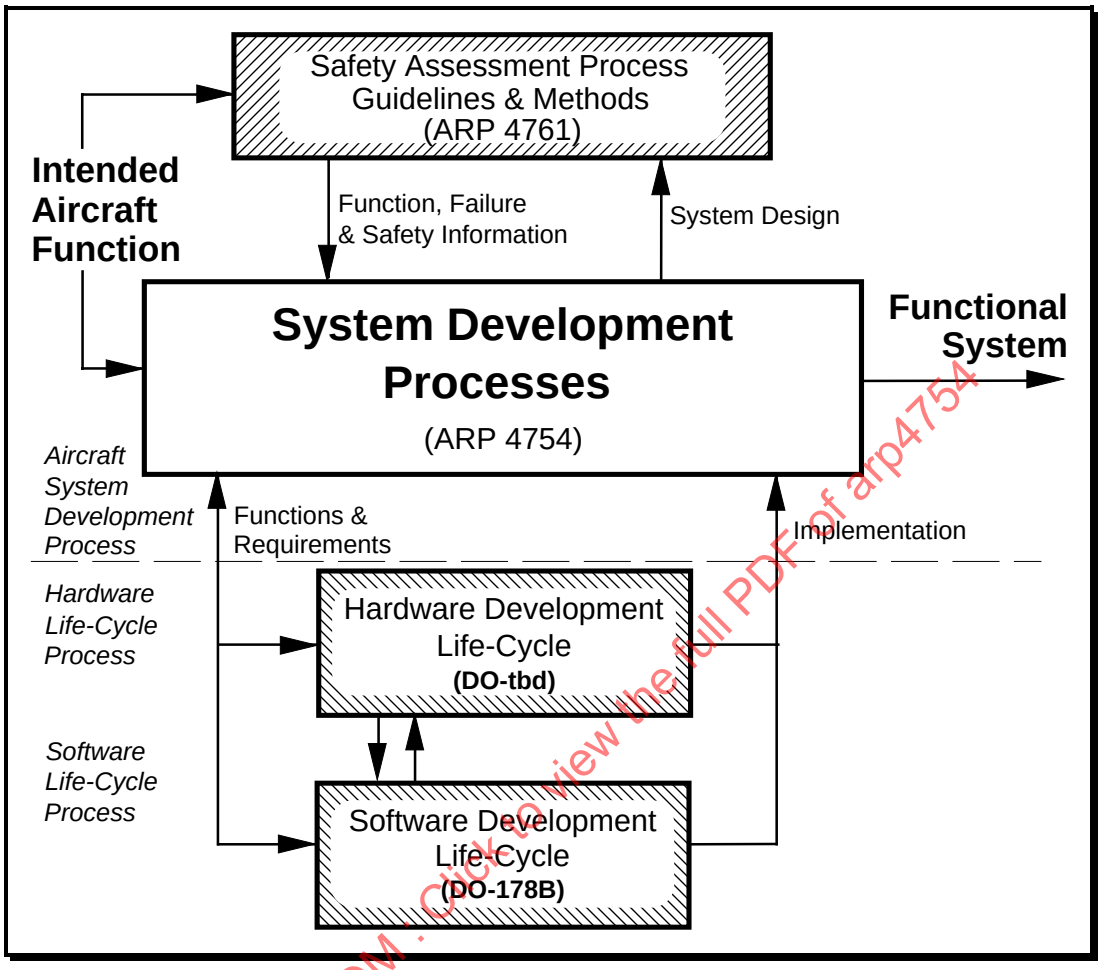

FIGURE 1 - Certification Guidance Documents Covering System, Safety, Software, and Hardware Processes

1. (Continued):

This document is intended to cover the needs of current technology and, as far as possible, emerging technology. It is anticipated that this document will be revised periodically to incorporate future technological changes and engineering process improvements.

1.1 Purpose:

These guidelines are intended to provide designers, manufacturers, installers, and certification authorities a common international basis for demonstrating compliance with airworthiness requirements applicable to highly-integrated or complex systems. The guidelines are primarily directed toward systems that integrate multiple aircraft-level functions and have failure modes with the potential to result in unsafe aircraft operating conditions. Typically, these systems are software based and involve significant interactions with other systems in a larger integrated environment. Frequently significant elements of these system are developed by separate individuals, groups or organizations. Highly-integrated or complex systems require added design discipline and development structure to ensure that safety and operational requirements can be fully realized and substantiated. While these guidelines could be applied to the development of simpler systems, the formality of the development structure, processes, and documentation should be reduced substantially.

Since this document is intended to provide a common basis for certification, the guidelines concentrate primarily on safety requirements associated with JAR/FAR 25.1309. Other requirements that determine the basis for satisfactory functionality, such as JAR/FAR 25.1301, can be addressed using this same guidance or simple extensions.

Much of the material covered in this document is not new and, where relevant, references are included to related documents. Many of processes referred to in this document are undergoing rapid evolutionary development. Moreover, the extent to which different systems can be classified as complex or highly-integrated is subject to wide variation. By providing a single publication that addresses the generic, high-level aspects of certification of highly-integrated or complex systems, it is believed that a broader understanding of the fundamental issues will develop. This, in turn, should aid both the applicant and the certification authorities in reaching agreement on the detailed system certification process for a particular aircraft model.

This document does not provide guidelines concerning the structure of the applicant's organization nor how the responsibilities for certification activities are divided. Neither should any such guidance be inferred from the descriptions provided.

1.2 Document Organization:

Table 1 outlines the content and purpose of each section of this document.

TABLE 1 - Content and Purpose of the Document Sections

Section	Content and Purpose
2	List of references.
3	Outlines the significant characteristics of highly-integrated or complex systems and introduces the concept of system development assurance as a means of certification.
4	Summarizes guidelines for the selection of certification data along with the planning and coordination of the certification program.
5	Describes means used to determine and allocate system requirements, including those requirements derived from system architecture, with specific emphasis given to the identification of development assurance levels.
6	Describes safety assessment activities that support the design process and lead to development of the recommended certification substantiation information.
7	Describes validation activities that substantiate the correctness and completeness of requirements.
8	Describes verification activities that substantiate correct design implementation.
9	Outlines elements of configuration management which support substantiation of development assurance requirements.
10	Outlines elements of process assurance which support substantiation of development assurance.
11	Explains the applicability of this document to modifications of existing aircraft or existing systems.
Appendix A	Describes a generic approach to systems development as an aid to understanding the systems concepts and process terminology used in this document.
Appendix B	A list of definitions, abbreviations and acronyms used in this document.
Appendix C	Index to descriptions and applications of concepts, tasks, and functions particularly significant to this document.
Appendix D	Outlines the importance of system architecture as a means of managing risk in highly-integrated or complex systems.

1.3 Document Conventions:

This document contains concepts and guidelines collected from representatives of the civil avionics, airframe, engine and regulatory communities. The contents are recommendations and are not mandated by law. For these reasons, the use of words such as “shall” and “must” is avoided. It is recognized that alternative methods to the processes described or referenced in this document may be available to an organization desiring to obtain certification of a highly-integrated or complex aircraft system.

The terms function and system can be applied at many different levels. Since the terms system and function are used at all levels of the development process, they create many opportunities for miscommunication. Any development program should identify the intended scope of these terms when they are used.

The term item is used in this document to describe any equipment, line replaceable unit, or line replaceable module. All items are characterized by a hardware definition and, where appropriate, a software definition. Components and software that are included in equipment, LRUs, or LRMs, and are not required to be controlled by part number at the aircraft level, are not items for the purpose of this document.

In this document, system generally means a combination of interrelated items arranged to implement a specific aircraft-level function or group of functions. A typical system will include such items as: power sources, sensors, control processing, indications and functional outputs. This is a broader meaning than the typical ATA 100 system designation.

The term partition is used in this document to describe the mechanism used to separate portions of a system or an item with sufficient independence such that a specific development assurance level can be substantiated within the partitioned portion.

1.4 Document Background:

During development of Revision B to RTCA document DO-178, it became apparent that system-level information would be required as input to the software development process. Since many system-level decisions are fundamental to the safety and functional aspects of aircraft systems, regulatory involvement in the processes and results relating to such decisions is both necessary and appropriate.

This document was developed in response to a request from the FAA to SAE. The FAA requested that SAE define the appropriate nature and scope of system-level information for demonstrating regulatory compliance for highly-integrated or complex avionic systems. The Systems Integration Requirements Task group (SIRT) was formed to develop an ARP that would address this need.

1.4 (Continued):

The initial members of SIRT recognized that harmonization of international understanding in this undertaking was highly desirable and encouraged participation by both Federal Aviation Administration (FAA) and Joint Aviation Authorities (JAA) representatives. A companion working group was formed under EUROCAE, WG-42, to coordinate European input to the SIRT group. The task group included people with direct experience in design and support of large commercial aircraft, commuter aircraft, commercial and general aviation avionics, jet engines, and engine controls. Regulatory personnel with a variety of backgrounds and interests participated in the work of the task group. Both formal and informal links with RTCA special committees (SC-167 and SC-180) and SAE committee (S-18) were established and maintained. Communication with the harmonization working group addressing FAR/JAR 25.1309 was maintained throughout development of this document.

Throughout development of this document, discussion return repeatedly to the issue of guidance specificity. Strong arguments were presented in favor of providing a list of very specific certification steps—a checklist. Equally strong arguments were made that the guidance should focus on fundamental issues, allowing the applicant and the certification authority to tailor details to the specific system. It was recognized that in either case, certification of all but the most idealized systems will require significant engineering judgment by both parties. The quality of those judgments is served best by a common understanding of, and attention to, fundamental principles. The decision to follow this course was supported by several other factors; the variety of potential systems applications, the rapid development of systems engineering, and industry experience with the evolving guidance contained in DO-178, DO-178A, and DO-178B being particularly significant.

The generic systems development road map presented in Appendix A, together with the detailed definitions contained in Appendix B.1, provides additional insight into systems development that can be of assistance to those not directly familiar with this type of work. The concept, task, and function index in Appendix C is intended to help users of the document quickly find specific types of information. The risk allocation discussion in Appendix D clarifies the relationship between this document and DO-178B.

2. REFERENCES:

2.1 Applicable Documents:

The following publications form a part of this document to the extent specified herein. The latest issue of SAE publications shall apply. The applicable issue of other publications shall be the issue in effect on the date of the purchase order. In the event of conflict between the text of this document and references cited herein, the text of this document takes precedence. Nothing in this document, however, supersedes applicable laws and regulations unless a specific exemption has been obtained.

2.1.1 SAE Publications: Available from SAE, 400 Commonwealth Drive, Warrendale, PA 15096-0001.

ARP4761	Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems

2.1.2 FAA Publications: Available from Federal Aviation Administration, 800 Independence Avenue, SW, Washington, DC 20591.

AC 25.1309-1A	System Design and Analysis, Advisory Circular

2.1.3 JAA Publications: Available from JAA Headquarters, Saturnusstraat 10, P.O. Box 3000, 2130 KA Hoofddorp, Netherlands.

AMJ 25.1309	System Design and Analysis, Advisory Material Joint

2.1.4 ATA Publications: Available from Airline Transport Association of America, 1709 New York Avenue, NW, Washington, DC 20006.

ATA-100	"ATA Specification for Manufacturer's Technical Data", Rev. 30, Oct. 31, 1991

2.1.5 RTCA Publications: Available from RTCA Inc., 1140 Connecticut Avenue, NW, Suite 1020, Washington, DC 20036.

DO-178B	Software Considerations in Airborne Systems and Equipment Certification
DO-xxx	Document under development by SC-180 Working title: "Design Assurance Guidance for Airborne Electronic Hardware"

2.2 Definitions

See Appendix B.1

2.3 Abbreviations and Acronyms:

See Appendix B.2

3. SYSTEM DEVELOPMENT:

This section outlines an overall system development process model that provides a structure for the remaining sections in this document. The concept of development assurance is introduced.

3.1 A Conceptual System Development Process:

Figure 2 shows an aircraft function implementation process model. The model includes multiple system development processes. Each system development process can consist of multiple item development processes. Nested within each item development process there can be multiple hardware life cycles as well as multiple software life cycles. There are certain supporting processes that take place repetitively during each of the development activities. Figure 3 illustrates a generic systems development process showing the scope of this document. A description of a generic system development process is provided in Appendix A.1. This description is not intended as a preferred method, nor does it imply a specific organizational structure, but is intended to facilitate understanding. The interface between the system development process and the hardware and software life-cycles is described in Appendix A.2.

Most actual systems development processes involve many iterative cycles, making the experience appear more cyclic than sequential. The entry point for aircraft function implementation may occur at any point in the cycle. For a new aircraft-level function, the process begins with the top-level definition of requirements. For adding functions to an aircraft, the entry point may occur in the context of changes to a particular piece of equipment. However, regardless of the entry point, an assessment of the impact of the new or modified function on other aircraft-level functions and their supporting requirements is necessary. In practice many of the development activities shown in Figure 3 are concurrent and may involve interactive dependencies that lead to alteration of previously established requirements.

Highly-integrated and complex systems present greater opportunities for development error (requirements determination and design errors) and undesirable, unintended effects. At the same time it is generally not practical (and may not even be possible) to develop a finite test suite for highly-integrated and complex systems which conclusively demonstrates that there is no development error residue. Since these errors are generally not deterministic and suitable numerical methods for characterizing them are not available, other qualitative means should be used to establish that the system can satisfy safety objectives. Development assurance (as defined below and in Appendix B.1) establishes confidence that system development has been accomplished in a sufficiently disciplined manner to limit the likelihood of development errors that could impact aircraft safety.

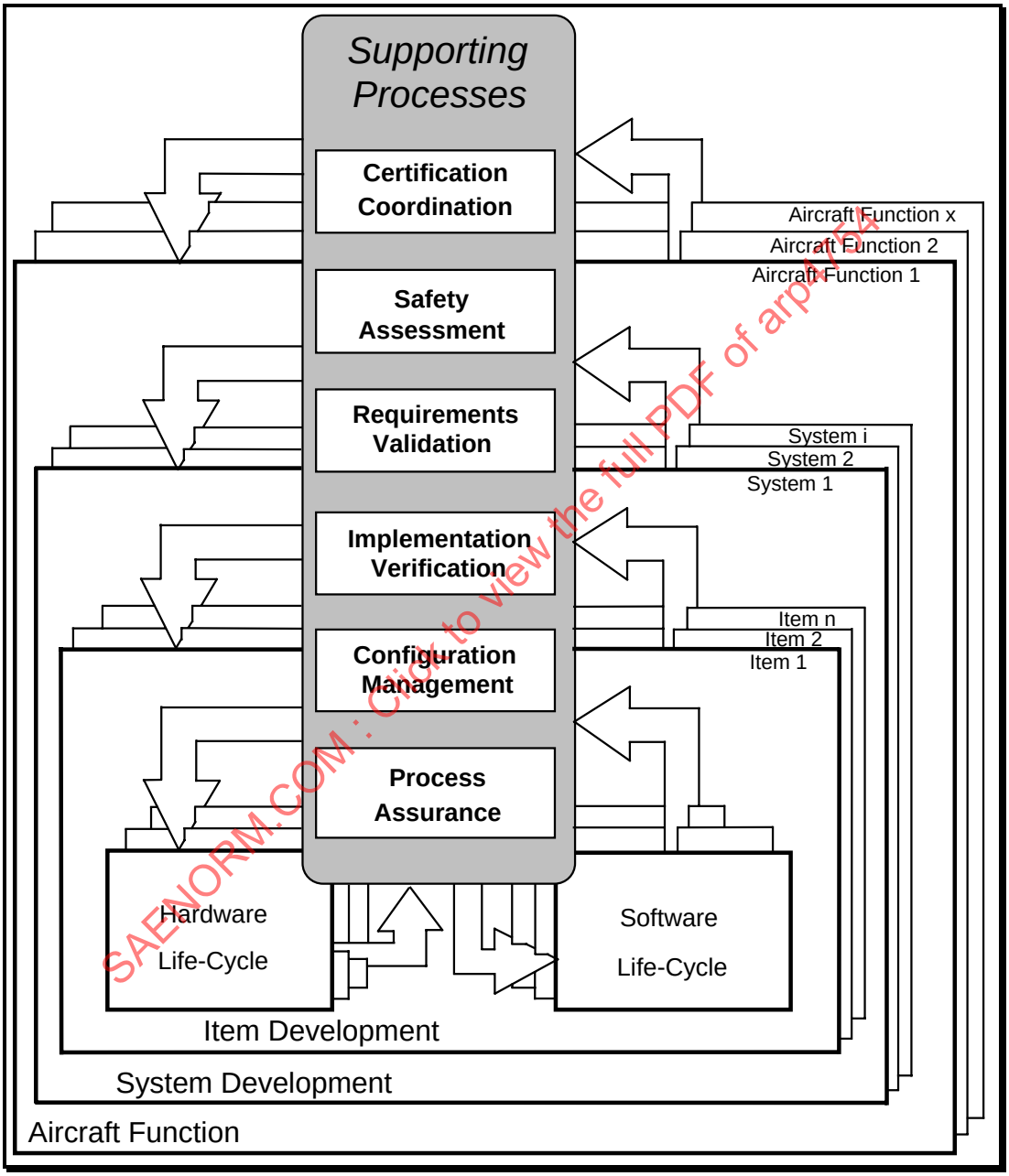

FIGURE 2 - Aircraft Function Implementation Process

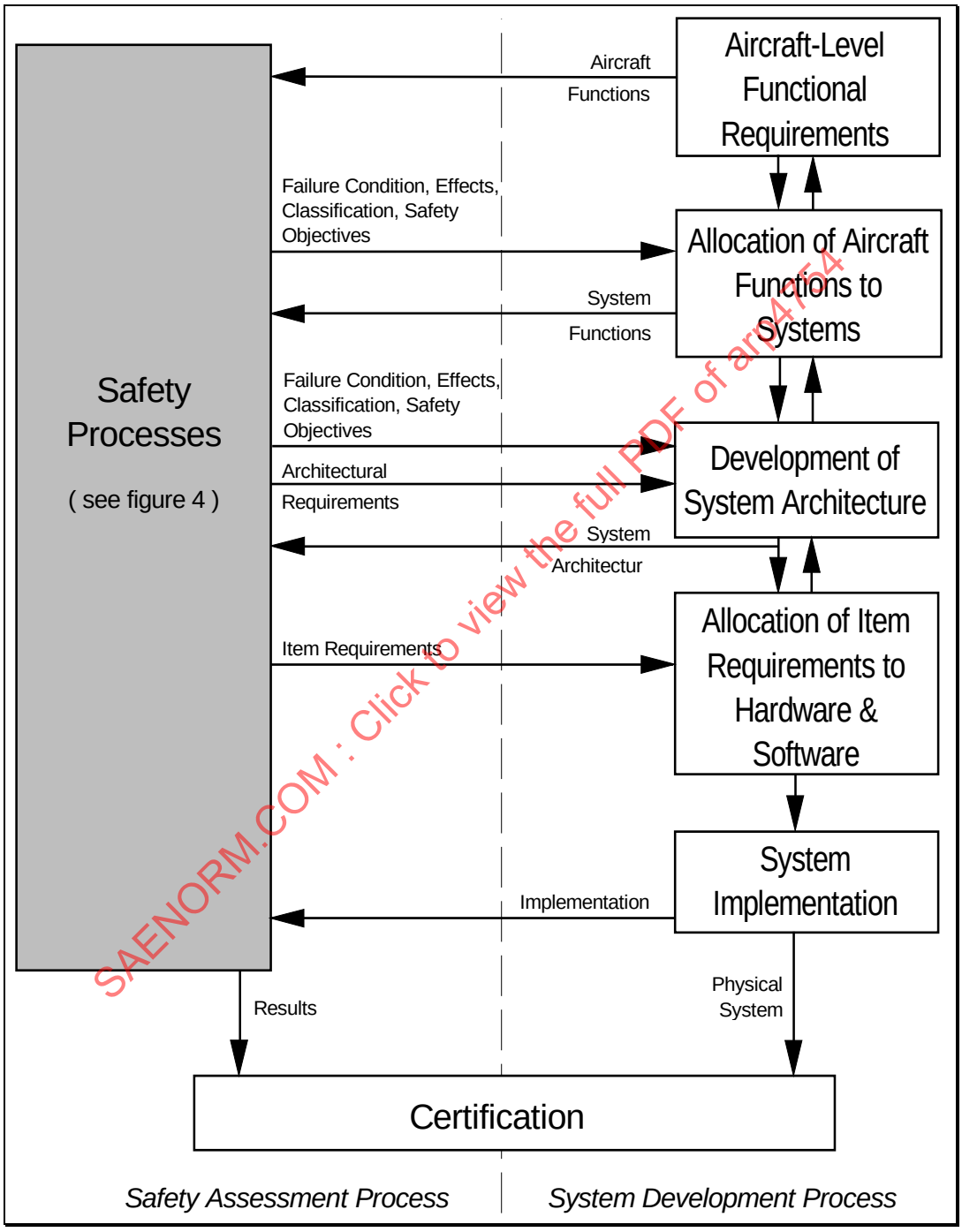

FIGURE 3 - System Development Process Model

3.2 Development Assurance:

Development assurance is a process involving specific planned and systematic actions that together provide confidence that errors or omissions in requirements or design have been identified and corrected to the degree that the system, as implemented, satisfies applicable certification requirements.

3.2.1 Development Assurance Activities: The development assurance activities are the supporting processes shown in Figure 2. In Section 5 of this document, systems and items are assigned "development assurance levels" based on failure condition classifications associated with aircraft-level functions implemented in the systems and items. The rigor and discipline needed in performing the supporting processes will vary corresponding to the assigned development assurance level.

Sections 4 through 10 give insight into the correspondence between development assurance level and the recommended activities contained within the supporting processes. The development assurance level determines the necessary software and hardware design levels of DO-178B and DO-xxx.

3.2.2 Development Assurance Substantiation: There are two key elements in the process of substantiating development assurance. One involves timely visibility of the activities that make up the supporting processes along with applicable results. The other illuminates the manner in which the safety assessment process interacts with the other supporting processes at key points in the development program. For most highly integrated or complex systems, development assurance substantiation extends throughout the majority of the development period. This makes it convenient to view the certification process itself as a supporting process.

3.3 Safety Directed Development Concept:

As an alternative to the system development assurance methods in this document, use of a safety directed development concept may provide an alternate means for showing compliance with FAR/JAR 25.1309. The safety directed development concept shifts the certification compliance emphasis from a general effort to eliminate errors (only some of which are safety related) to an effort focusing on potential system safety related errors. Design features are specifically employed to preclude or contain the safety consequences of these errors. The safety directed development concept employs safety, validation and verification processes that are similar in many respects to the methods outlined in this document.

4. CERTIFICATION PROCESS AND COORDINATION:

The objective of the certification process is to substantiate that the aircraft and its systems comply with applicable airworthiness requirements. In most situations the aircraft certification is accomplished through compliance with a series of system certification plans. Planning and coordination are vital to establish effective communications between the applicant and the certification authority and to reach agreement on the intended means of showing that the aircraft and its systems comply with airworthiness requirements.

One of the characteristics of highly-integrated or complex systems is the need to use development assurance methods as part of the evidence supporting system certification.

Careful attention to system architecture, item architecture, and component selection may simplify the development assurance methods and may limit the range of systems or items to which this certification strategy applies.

4.1 Certification Planning:

Aircraft tend to have a high degree of variance in both system complexity and integration. The certification process, therefore, should be flexible. Certification planning provides applicants with that flexibility, while at the same time providing both the applicant and certification authority a high level of confidence in the certification approach.

Certification planning separates the regulatory aspects of the overall development process into manageable tasks that can be accomplished in a logical and sequential manner. A top-level plan describes the certification basis (applicable regulations and special conditions) and outlines the means by which the applicant expects to demonstrate compliance. Early certification planning can minimize the effects of misinterpretations of the regulations and advisory material.

The certification plan for a complex or highly-integrated system defines the product and installation to be certified, outlines the product development processes to be used for development assurance, and identifies the proposed means of compliance with the regulations. Since many of the development assurance activities occur well before an implementation is available, early coordination with the certification authorities is recommended. An early certification plan may be missing significant detail, necessitating subsequent updates. Even so, early coordination of the plan is strongly encouraged.

4.2 Agreement on the Proposed Means of Compliance:

The applicant proposes a means of compliance that defines how the development of the system or equipment will satisfy the applicable certification basis. The applicant should:

a. Submit plans to the certification authority for review before relevant development activities occur. If additional substantiating data concerning the plans or means of compliance is requested, it should be submitted in time to allow meaningful review.

b. Resolve issues identified by the certification authority concerning the means by which the system will be shown to comply with airworthiness requirements.

c. Obtain agreement with the certification authority on the plans.

4.3 Compliance Substantiation:

The certification data is the evidence that the system satisfies airworthiness requirements. This includes both the data that is submitted and the data that is required to be retained. The certification authority determines the adequacy of the data for showing regulatory compliance. The applicant should develop a certification summary to describe how it was determined that the system, as installed on the aircraft, complies with the agreed certification plan.

The certification summary should provide an outline of the results of the activities established in the certification plan. Any deviation from the agreed plan should be described together with rationale to substantiate the deviation. In addition to addressing each of the content items in the certification plan, the certification summary should include:

a. A statement of compliance to the airworthiness requirements.
b. An outline of any open problem reports that impact functionality or safety.

4.4 Certification Data:

The range of possible certification data listed in Table 2 includes a cross reference to sections of this document where related information, including information about the development assurance level, can be found.

TABLE 2 - Certification Data

System Certification Data	Description
Certification Plan	4.4.1
Development Plan	4.4.3
Architecture and Design	4.4.4
Requirements	5.2 and 5.3
Validation Plan	7.7.1
Verification Plan	8.5.1
Configuration Management Plan	9.0
Process Assurance Plan	10.2
Configuration Index	4.4.2
Functional Hazard Assessment	6.1
Preliminary System Safety Assessment	6.2
System Safety Assessment	6.3
Common Cause Analysis	6.4
Validation Data	7.7
Verification Data	8.5
Evidence of Configuration Management	9.2
Evidence of Process Assurance	10.4
Certification Summary	4.3

NOTE: Shading indicates minimum certification data.

There is no need to submit certification data beyond that specifically negotiated with the certification authority. The minimum certification data to be submitted to the certification authority includes the certification plan, certification summary, and configuration index.

The certification data described in this document does not imply a requirement for specific arrangement or grouping of data or delivery format (such as paper, computer files, or remote terminal displays.) Whatever form is selected by the applicant should provide for efficient retrieval and review by the certification authority as required by regulations and laws in effect governing in-service aircraft.

4.4.1 Certification Plan: The certification plan for a highly-integrated or complex system should address both the system and the aircraft environment within which the system will be used. The amount of detail contained in the plan should vary depending on the classification of the associated aircraft hazard(s). Each plan should include:

a. A functional and operational description of the system and the aircraft on which the system will be installed. A description of the system elements including hardware and software. This description should establish the functional, physical, and information relationship between the system and other aircraft systems and functions.

b. A statement of the relationship of this certification plan to any other relevant system certification plan(s.)

c. A summary of the functional hazard assessment (aircraft hazards, failure conditions, and classification).

d. A summary of the preliminary system safety assessment (system safety objectives and preliminary system development assurance levels.)

e. A description of any novel or unique design features that are planned to be used in meeting the safety objectives.

f. A description of the new technologies or new technology applications to be implemented.

g. The system certification basis including any special conditions.

h. The proposed methods of showing compliance with the certification basis, including an outline of the anticipated development assurance processes (safety assessment, validation, verification, configuration management, and process assurance.)

i. A list of the data to be submitted and the data to be retained under configuration control, along with a description or sample of data formats.

j. The approximate sequence and schedule for certification events.

k. Identification of the personnel or specific organization responsible for certification coordination.

4.4.2 Configuration Index: The system configuration index identifies all of the physical elements that, together, comprise the system. In addition the configuration index identifies procedures and limitations that are integral to system safety. Any system design features or capabilities provided in excess of those required to establish system safety under the applicable regulations should be identified.

A typical system configuration index will include the following information:

a. Configuration identification of each system item
b. Associated item software
c. Interconnection of items
d. Required interfaces with other systems
e. Safety-related operational or maintenance procedures and limitations

When applicable, information describing permissible interchangeability of alternate items within the system should be included.

4.4.3 Development Plan: While there is no specific recommended process for systems development, a generic development model is described in Appendix A to assist in establishing common terminology and understanding. The specific development process selected should be described in sufficient detail to achieve mutual understanding of the key elements and their relationships.

The development plan should identify the top-level processes planned for use, the key events that mark the planned development cycle, and the organizational structure and key individual responsibilities supporting the development. The processes and events should be described in sufficient detail to establish their relative significance to the system development, their relative timing and interdependencies, and the nature of the results expected at event or process completion.

4.4.4 Architecture and Design: The architecture and design description should be based on a common understanding of: the intended aircraft-level functionality provided or supported by the system, the anticipated system operating environment, and the specific capabilities of the system as installed on the aircraft. Sufficient system architectural and design detail should be provided to establish how the system will achieve the intended functionality. The description should also identify primary fault or failure containment means. New or novel design elements should be identified; along with specific architectural features and design elements that perform a specific role in establishing or maintaining system safety.

5. REQUIREMENTS DETERMINATION AND ASSIGNMENT OF DEVELOPMENT ASSURANCE LEVEL:

This section discusses the determination of requirements, from aircraft function identification, through derived requirements emanating from hardware and software development. The relationships among functions, related failure condition classifications, system and item requirements, and the corresponding assignment of development assurance levels is shown. Architectural alternatives are examined and examples provided that illustrate the effect of architecture on the assignment of item development assurance level.

The assignment of numerical risk requirements for deterministic hardware implementations and the potential for mitigating some failure conditions through human interaction are discussed.

5.1 Requirements Capture:

Requirements, together with related hazards, provide the common basis for the supporting processes. Because the hazards may have different levels of importance, the allocation of requirements, through system architecture, has significant impact on the ease of substantiating system certification.

The top level process in the aircraft development cycle includes the identification of aircraft functions and the requirements associated with these functions. The aircraft functions, including functional interfaces and corresponding safety requirements, form the basis for establishing the system architecture. Selection of the architecture establishes additional requirements necessary to implement that architecture. At each phase of the requirements identification and allocation process (i.e., system, item and hardware/software) both additional detail for existing requirements and new derived requirements are identified. Choices made and problems encountered during implementation are a primary source for derived requirements and may lead to identification of new system safety requirements.

5.2 Types of Requirements:

The requirements associated with a given function define the way the function acts in its environment and include the definition of the user/machine interface. The types of requirements detailed below should be considered at various phases of the development activities (i.e., function, system, item and hardware/software). There may be requirements that address strictly business or economic issues and do not impact safety or certification requirements.

5.2.1 Safety Requirements: The safety requirements for aircraft and system-level functions include minimum performance constraints for both availability (continuity of function) and integrity (correctness of behavior) of the function. These safety requirements should be determined by conducting a functional hazard assessment consistent with the processes in 6.1.

Safety requirements for aircraft and system functions are determined by identifying and classifying associated functional failure conditions. All functions have associated failure modes and associated aircraft effects, even if the classification is “No safety effect.” Safety related functional failure modes may have either contributory or direct effects upon aircraft safety.

Requirements that are defined to prevent failure conditions or to provide safety related functions should be traceable through the levels of development at least to the point of allocation to hardware and software. This will ensure visibility of the safety requirements at the software and hardware design level.

5.2.2 Functional Requirements: Functional requirements are those necessary to obtain the desired performance of the system under the conditions specified. They are a combination of customer desires, operational constraints, regulatory restrictions, and implementation realities. These requirements define all significant aspects of the system under consideration. Regardless of the original source, all functions should be evaluated for their safety related attributes.

5.2.2.1 Customer Requirements: Customer requirements will vary with the type of aircraft, the specific function or the type of system under consideration. Requirements may include those associated with the operator's intended payload, route system, operating practices, maintenance concepts, and desired features.

5.2.2.2 Operational Requirements: Operational requirements define the interfaces between the flight crew and each functional system, the maintenance crew and each aircraft system, and various other aircraft support people and related functions or equipment. Actions, decisions, information requirements and timing constitute the bulk of the operational requirements. Both normal and nonnormal circumstances need to be considered when defining operational requirements.

5.2.2.3 Performance Requirements: Performance requirements define those attributes of the function or system that make it useful to the aircraft and customer. In addition to defining the type of performance expected, performance requirements include function specifics such as: accuracy, fidelity, range, resolution, speed, and response times.

5.2.2.4 Physical and Installation Requirements: Physical and installation requirements relate the physical attributes of the system to the aircraft environment. They may include: size, mounting provisions, power, cooling, environmental restrictions, visibility, access, adjustment, handling, and storage. Production constraints may also play a role in establishing these requirements.

5.2.2.5 Maintainability Requirements: Maintainability requirements include scheduled and unscheduled maintenance requirements and any links to specific safety-related functions. Factors such as the percent of failure detection or the percent of fault isolation may also be important. Provisions for external test equipment signals and connections should be defined in these requirements.

5.2.2.6 Interface Requirements: Interface requirements include the physical system and item interconnections along with the relevant characteristics of the specific information communicated. The interfaces should be defined with all inputs having a source and all output destinations defined.

5.2.3 Additional Certification Requirements: Additional functions, functional attributes, or implementations may be required by airworthiness regulations or may be necessary to show compliance with airworthiness regulations. Requirements of this type should be defined and agreed upon with the appropriate certification authorities.

5.3 Derived Requirements:

At each phase of the development activity, decisions are made as to how particular requirements or groups of requirements are to be met. The consequences of these design choices become requirements for the next phase of the development. Since these requirements result from the design process itself, they may not be uniquely related to a higher-level requirement and are referred to as derived requirements.

Derived requirements should be examined to determine which aircraft-level function (or functions) they support so that the appropriate failure condition classification can be assigned and the requirement validated. While most such requirements will not impact the higher-level requirements, some may have implications at higher levels. Derived requirements should be reviewed at progressively higher system levels until it is determined that no further impact is propagated.

For example, derived requirements may result from the decision to select a separate power supply for equipment performing a specific function. The requirements for the power supply, including the safety requirements, are derived requirements. The hazard resulting from the fault or failure of the function supported by the power supply determines the necessary development assurance level.

Derived requirements may also result from architecture choices. For example, selecting a triplex architecture for achieving a high integrity functional objective would have different consequences and different derived requirements from selection of a dual monitored architecture for achievement of the same objective.

5.3 (Continued):

Derived requirements may result from a design decision to isolate function implementations having more severe failure condition classifications from the malfunction or failure effects of systems having less severe failure condition classifications.

Derived requirements also include those defining the hardware-software interface. Some of these requirements may be significant at the system level. The remainder, dealing with detailed aspects of the hardware-software interface, may be handled under the guidance of DO-178B and DO-xxx.

Derived requirements should be captured and treated in a manner consistent with other requirements applicable at that development phase.

5.4 Assignment of Development Assurance Levels:

The system development assurance level is assigned based on the most severe failure condition classification associated with the applicable aircraft-level function(s) (see Table 3). This table departs slightly from AC 25.1309-1A and AMJ 25.1309 by establishing level E as “no safety effect”.

If the Preliminary System Safety Assessment (PSSA - see 6.2) shows that the system architecture provides containment for the effects of design errors, so that the aircraft-level effects of such errors are sufficiently benign, then development assurance activities can be conducted at a reduced level of process rigor for the system items wholly within the architectural containment boundary.

If a system has multiple categories of failure conditions associated with its different functions, architectural means may be used to limit the interaction between items. This may allow the separate items to be developed at different assurance levels.

Item development assurance level is based on the overall system architecture through allocation of risk determined using the PSSA. For items that support multiple aircraft functions, the applicable safety requirement should be based on the most severe of the effects resulting from failure or malfunction of any supported aircraft function or any combination of supported aircraft functions.

TABLE 3 - System Development Assurance Level Assignment

Failure Condition Classification	System Development Assurance Level
Catastrophic	A
Hazardous/Severe Major	B
Major	C
Minor	D
No Safety Effect	E

5.4.1 Architecture Considerations: System architectural features, such as redundancy, monitoring, or partitioning, may be used to eliminate or contain the degree to which an item contributes to a specific failure condition. System architecture may reduce the complexity of the various items and their interfaces and thereby allow simplification or reduction of the necessary assurance activity. If architectural means are employed in a manner that permits a lower assurance level for an item within the architecture, substantiation of that architecture design should be carried out at the assurance level appropriate to the top-level hazard. This does not preclude assignment of item levels that are lower than the level associated with the top-level hazard; but assurance that the item level assignments and their independence are acceptable should be validated at the higher level and verified by the System Safety Assessment (SSA - see 6.3).

Redundancy is a technique for providing multiple implementations of a function — either as multiple items, or multiple lanes within an item. It is a design technique based on the assumption that a given set of faults with the same system effect will not occur simultaneously in two or more independent elements. Redundancy is required to provide fail-safe design protection from catastrophic failure conditions and may be necessary to meet the requirements associated with the more severe failure condition classifications. The redundant elements may be parallel or backup, and their designs may be similar or dissimilar.

For all but the simplest systems, it is practically impossible to guarantee the correctness and completeness of requirements or the correctness of all necessary assumptions. An architectural strategy incorporating dissimilarity can be a powerful means of reducing the potential for errors in requirements or in design implementation to cause serious effects. The nature and probability of such errors are, in general, not accurately or reliably predictable. Hence the effectiveness of particular architectural strategies, introduced to allow the allocation of a lower item risk level, generally can not be quantified. In consequence, the justification to support such allocation necessarily involves some degree of engineering judgment by the applicant and the certification authorities. Timely discussions, adequately documented, can reduce the likelihood of late design changes resulting from unresolved disagreements about such judgments.

5.4.1 (Continued):

When dissimilarity is used as a means of design error containment, the degree of credit should be related to the type and scope of design errors shown to be covered by the dissimilarity method used. For example, dissimilar design implementations of the same function can provide containment coverage for some types of implementation errors but not for function requirements errors. Assuming adequate independence can be shown, dissimilar design implementations of dissimilar functions can provide containment coverage for both implementation and function requirements errors. It should be noted that architectural dissimilarity impacts both integrity and availability. Since an increase in integrity may be associated with a reduction in availability, and vice-versa, the specific application should be analyzed from both perspectives to ensure its suitability.

The principles guiding systems development assurance are, in some ways, similar to those developed to cope with software assurance. However, there are significant differences between software assurance and systems development assurance.[1]

Table 4 and 5.4.1.1 through 5.4.1.5 present a series of example architectures and illustrate the effect those architectures may have on item development assurance level. Actual aircraft systems may involve combinations of these architectures or alternative architectures that do not conform to any one of these examples. If architecture is used as a means of coping with generic design error, it will be necessary to complement quantitative assessments of the architecture with qualitative assessments. Issues of design information source commonality, design methodology commonality, technology maturity, and application understanding, among others, are often dealt with qualitatively. The Table 4 examples, together with experience gained from successful systems, can augment the process of translating a specific system architecture into development assurance level requirements for each system item.

5.4.1.1 Partitioned Design: Partitioning is a design technique for providing isolation to contain and/or isolate faults and to potentially reduce the effort necessary for the system verification processes. Where an item contributes to aircraft functions of different criticality, a partition can be used to limit the cross-functional effect of system design errors in separate parts of the item.

The design that provides the partition should be developed at the development assurance level corresponding to the highest applicable function failure condition classification.

The design within a partitioned portion can be developed to the development assurance level corresponding to the most severe failure condition classification for that portion (see Table 4, architecture 1).

[1] The guidance material on risk allocation contained in ARP4754 is more detailed and definitive than the summary material presented in DO-178B. Both documents stress the use of the system safety assessment process as the means of capturing and justifying the development assurance level. Appendix D provides additional background on considerations associated with the allocation of risk in airplane systems development.

TABLE 4 - Examples of Architecturally Derived Assurance Levels and Constraints

	Architecture (see note 1)	Failure Condition Classification Catastrophic	Failure Condition Classification - Severe-Major/Hazardous
1	Partitioned Design (Multiple Failure Categories)	Level A for the system including establishment of the partition	Level B for the system including establishment of the partition
	Within each partitioned portion	Level corresponding to the most severe failure condition classification within that partitioned portion	Level corresponding to the most severe failure condition classification within that partitioned portion
2	Dissimilar, Independent Designs Implementing an Aircraft-Level Function (notes 2 and 3)	Level A for the system including establishment of dissimilarity and independence	Level B for the system including establishment of dissimilarity and independence
	Portions (note 4)	Level B (note 5 and 5.4.1.2)	Level C (note 5 and 5.4.1.2)
3	Dissimilar Designs Implementing an Aircraft Level Function (note 2)	Level A for the system including establishment of partition between the portions	Level B for the system including establishment of partition between the portions
	Primary Portion	Level A	Level B
	Secondary Portion	Level B (note 5 and 5.4.1.3)	Level C (note 5 and 5.4.1.3)
4	Active/Monitor Parallel Design (note 2)	Level A for the system	Level B for the system
	Active and Monitor Portions	At least one portion to Level A; the other portion to at least Level C (notes 5 and 6)	At least one portion to Level B; the other portion to at least Level C (notes 5 and 6)
5	Backup Parallel Design (note 2)	Level A for the system	Level B for the system
	Primary Portion	Level A	Level B
	Backup Portion	Level C (note 5)	Level D (note 5)

Note 1: These architectures illustrate specific development assurance situations; practical systems may employ a wide range of alternative architectures.
Note 2: The logic to determine switching/voting/fault detection between elements should be developed to the highest level applicable.
Note 3: It is especially important to obtain the agreement of the certification authority as outlined in 4.1 before adopting this method.
Note 4: Portions can refer to an item, a group of items or an entire subsystem in this case.
Note 5: Availability requirements must be satisfied and the constraints of the applicable paragraph followed.
Note 6: The development assurance level is dependent on the classification of any failure condition not constrained by the monitor.

No shade: applies to the system as a whole.
Shaded: applies to portions of the system, where system architecture permits

5.4.1.2 Dissimilar, Independent Designs Implementing An Aircraft-Level Function: A parallel, dissimilar, multiple channel architecture may provide protection from both random physical failures and anomalies due to design errors.[2] In these cases, it may be possible for the development assurance levels for the individual channels to be selected below that associated with the top level failure condition classification (see Table 4, architecture 2). An analysis should substantiate the dissimilarity and independence of: implementation, requirements, algorithms, data, environment, and other potential sources of design error.

To be considered within this category, there must be substantial differences between the designs in terms of the means of preventing the top level failure condition(s), the methodology by which the designs are created, the technology through which the designs are implemented, and the operations through which the function is used. Validation of any assumptions of independence is of particular importance in demonstrating compliance using this design strategy. Regulatory acceptance of any specific system architecture developed under this paragraph will be aided by early and frequent discussions that proceed in parallel with system development.

---

[2] Systems guidance in DO-178B recommends that for parallel architectures at least one software component must have the software level associated with the most severe failure condition classification for the system function. Such an architecture would fall into the category described in 5.4.1.3. In ARP4754 parallel architectures are further subdivided by noting that there are some system architectures where the dissimilarly and independence of subsystems may be sufficiently clear as to warrant a reduction in level for each subsystem.

One such example encompasses the aircraft-level ground deceleration function. On most current commercial jet aircraft this function can be accomplished using either wheel brakes or reverse thrust. In a similar fashion radar vectors from an Air Traffic Control controller using the communications radios provide safe and reliable navigation and guidance in a manner that is independent of and dissimilar to the navigation radio-based VOR/DME system. Properly executed, these combinations could satisfy aircraft-level functional integrity requirements one level greater than the integrity of either element alone. In taking credit for such combinations, it is necessary to show the dissimilarity and independence of the system designs at the development assurance level associated with the aircraft-level function (ground deceleration and navigation, respectively in these cases.)

An example at a more detailed level within an autopilot serves to illustrate how this concept can be applied within a system. Assume that the autopilot design calls for a high integrity, high availability monitor. The means chosen for accomplishing this involves two dissimilar, independent techniques; e.g., one using control surface motion, and the other using airplane motion. In this example, the requirement specification for the two monitors will be essentially dissimilar and independent, although deriving the appropriate monitor parameter threshold settings may involve some common assumptions and the use of common models. It is necessary that any such assumptions are straightforward to validate and that they are validated at the development assurance level associated with the top level failure condition classification. This being true, it is possible that the system safety assurance process will support a development assurance level for both monitors one level lower than the top level monitor requirement.

5.4.1.3 Dissimilar Designs Implementing an Aircraft Level Function: If multiple portions cannot be shown to be dissimilar under 5.4.1.2, then the primary portion should be developed to the development assurance level associated with the most severe failure condition classification of the function. In this case, the secondary portion(s) provides the function after a random hardware failure of the primary portion. Both primary and secondary portions can execute full-time, or the secondary portion can be a “hot spare” that is reverted to after failure of the primary portion.

The secondary portion(s) can be assigned a development assurance level one below that of the primary portion, but not less than Level C, if:

a. The primary portion has a random hardware failure rate for loss of function less than $1.0x10^{-5}$ for catastrophic failure conditions, or less than $1.0x10^{-4}$ for hazardous failure conditions; and

b. The primary portion is always used unless it has failed; and

c. The secondary portion does not contribute to fault detection, and can not cause the primary portion to fail.

5.4.1.4 Active-Monitor Parallel Design: The active-monitor parallel architecture represents the situation where both the active and monitor portions are necessary to meet the integrity requirements. This architecture provides detection of random physical failures and with sufficient independence may detect anomalies due to design error. The most severe failure condition classification establishes the development assurance level necessary for at least one channel and the channel independence. The other channel may have a lower level of development assurance, as necessary to meet availability requirements.

5.4.1.5 Backup Parallel Design: There may be items in a system that function as a backup to other items. That is, they are required to operate only after the other system items have failed. If the primary system satisfies the integrity requirements without the backup and the probability of the random failure rate for loss of function of the primary items is very low ($1.0x10^{-7}$, if the failure condition is catastrophic, or $1.0x10^{-5}$, if the failure condition is hazardous), then the development assurance level of the backup may be assigned up to two levels below that of the top-level hazard, but not less than level D. This assignment is subject to the agreement of the certification authority. Development assurance of the system architecture leading to the determination of the availability requirement of the backup should be at the level of the most critical system failure condition.

5.4.2 Implementation Error Management: The methods of 5.4.1 apply to both top-level functional requirements and requirements resulting from implementation choices. The chosen implementation architecture and technology can play significant roles in determining the system effects of design errors and failures. The use of techniques, such as proper architecture or implementation technology selection, when specified by the PSSA, may mitigate or eliminate the effects of certain types of design errors and failures. These and other types of development techniques may reduce, or avoid, the need to meet the some of the objectives which would otherwise have to be satisfied for a given development assurance level. Any reduction or elimination of development assurance objectives should be supported by analysis including the PSSA. These techniques may be applied to hardware or software implementations.

5.4.3 Software Level Assignment: The software levels and processes for compliance as defined in RTCA/DO-178B are related to the failure condition classifications and may be assigned taking account of the item development assurance level defined in 5.4 and Tables 3 and 4.

5.4.4 Hardware Level Assignment: The substantiation of hardware item designs can be accomplished through rigorous testing and/or analysis unless a specific hardware component(s) is too complex to be analyzed by deterministic techniques. The necessary degree of rigor in the testing or analysis is determined by the failure condition classification.

When deterministic techniques are not used or are determined to be insufficient, the development assurance assignment methods of this section can be applied. The five levels of system development assurance listed in Table 3 correspond to five specific hardware levels. (The methods of substantiating hardware development assurance for each level are to be defined by RTCA SC-180 and documented in DO-xxx.) The requirements validation and design verification processes at system and item level should ensure that the appropriate hardware design requirements, and development assurance expectations, are available for the hardware portion of the design.

5.5 Failure Condition Risk Assignment:

Risks that are attributable to hardware failures may be assigned to items based on the PSSA. The assigned numerical risk becomes an item hardware requirement or can be further allocated to item hardware architecture requirements. Item hardware requirements should also contain applicable qualitative requirements, such as “no single failure”.

Some failure conditions can be mitigated through human interaction. Recognizing that incorrect human interactions could exacerbate, rather than mitigate, the situation, it is essential to examine the type and independence of the support provided to ensure the correctness of such interaction. Support should be provided for both human recognition of the system or item failure condition and human action to mitigate the failure effects. Where such support substantiates the potential for appropriate human action, the system or item may be assigned a lower risk. The substantiation should consider, at least:

5.5 (Continued):

a. The failure recognition provided
b. The type and timeliness of required failure response
c. The priority relative to other crew tasks
d. The total crew workload
e. The probability of occurrence of the failure
f. The independence of the human recognition and action support provided

The development process should assign risk to items in a rational manner which results in compatible levels of development assurance and reliability for each item.

6. SAFETY ASSESSMENT PROCESS:

The safety assessment process provides analytic evidence showing compliance with airworthiness requirements. The process includes specific assessments conducted and updated during system development and interacts with the other system development supporting processes. The primary safety assessment processes are listed below.

a. Functional Hazard Assessment (FHA): Examines aircraft and system functions to identify potential functional failures and classifies the hazards associated with specific failure conditions. The FHA is developed early in the development process and is updated as new functions or fault conditions are identified.

b. Preliminary System Safety Assessment (PSSA): Establishes specific system and item safety requirements and provides preliminary indication that the anticipated system architecture can meet those safety requirements. The PSSA is updated throughout the system development process.

c. System Safety Assessment (SSA): Collects, analyzes, and documents verification that the system, as implemented, meets the system safety requirements established by the FHA and the PSSA.

d. Common Cause Analysis (CCA): Establishes and validates physical and functional separation and isolation requirements between systems and verifies that these requirements have been met.

A summary of the four primary safety assessments is included in this section. The process for determining safety-related maintenance tasks and task intervals for support of all four assessments is explained.

Figure 4 shows the fundamental relationships between these four specific assessments and the system development processes. In reality, there are many feedback loops within and among these relationships, though they have been omitted from the figure for clarity.

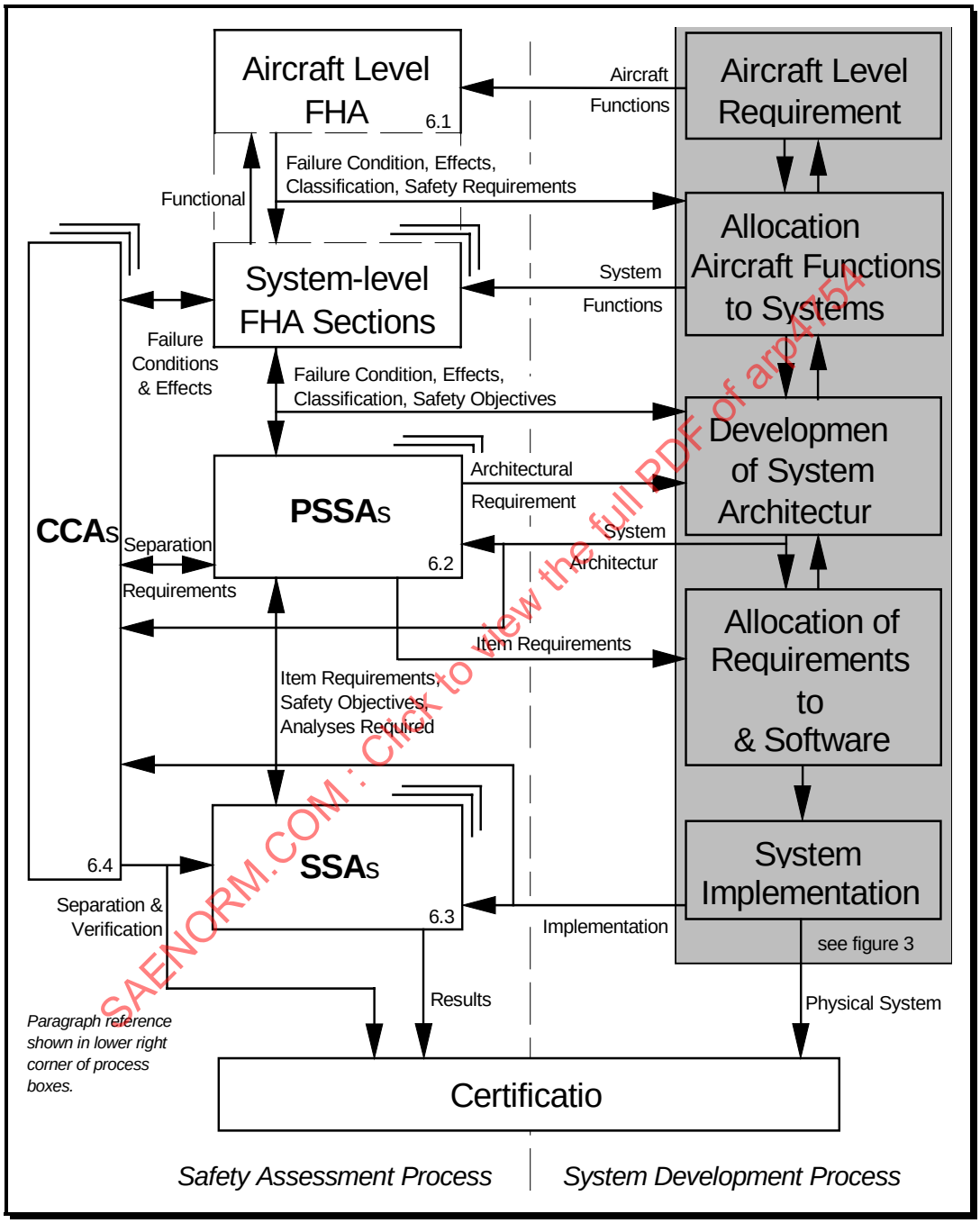

FIGURE 4 - Safety Assessment Process Model

6. (Continued):

This section of the document provides guidance and recommendations covering:

a. What safety analyses are most appropriate for each failure condition classification.

b. How to apply the results of the various safety assessment processes at each stage of system development. This includes identifying functional safety requirements and applicable derived safety requirements.

The level of detail needed for the various safety assessment activities is dependent on the aircraft-level failure condition classification, the degree of integration, and the complexity of the system implementation. In particular, the safety assessment process should take into account any interdependencies that arise due to common architecture or use of common complex components across systems or within system integration. The safety assessment process should be planned and managed so as to provide the necessary assurance that all relevant failure conditions have been identified, and that all significant combinations of failures that could cause those failure conditions have been considered. The safety assessment process is of fundamental importance in establishing appropriate safety objectives for the system and determining that the implementation satisfies these objectives.

The safety assessment activities and their objectives for each Failure Condition Classification are described in the subsequent sections of this document as referenced in Table 5. (Note: Safety-related requirements at the item level are derived during the PSSA process.) Detailed guidelines and methods for conducting the various assessments are described in ARP4761.

TABLE 5

Derived from FHA (see 6.1) Failure Condition Classification	Derived from FHA (see 6.1) Development Assurance Level	Safety Objectives Fail-Safe	Safety Objectives Quantitative Requirement (Note 1)	Safety Analyses PSSA	Safety Analyses SSA	Safety Analyses Common Cause
Catastrophic	A	Required, {5.4}	P $<10^{-9}$	{6.2, 6.5}	{6.3, 6.5}	{6.4}
Hazardous/ Severe-Major	B	May be needed, {5.4}	P $<10^{-7}$	{6.2, 6.5}	{6.3, 6.5}	{6.4}
Major	C	May be needed, {5.4}	P $<10^{-5}$	{6.2}	{6.3}	May be needed, {6.4}
Minor	D	No	None	(Note 2)	(Note 2)	None
No Safety Effect	E	No	None	(Note 3)	None	None

Paragraph references shown in { }.
Note 1: According to AC/AMJ 25.1309; rate shown per flight hour
Note 2: According to AC/AMJ 25.1309 par 8.2 ... an analysis of physical and functional isolation from other functions and systems may be required.
Note 3: Required to the level necessary to establish that no safety effect exists.

6.1 Functional Hazard Assessment:

The functional hazard assessment (FHA) should provide the following information relative to each aircraft-level function and combination of aircraft-level functions:

a. Identification of related failure condition(s).

b. Identification of the effects of the failure condition(s).

c. Classification of each failure condition based on the identified effects (i.e., Catastrophic, Hazardous/Severe-Major, Major, Minor, or No Safety Effect) and assignment of the necessary safety objectives, as defined in AC 25.1309-1A and AMJ 25.1309 extended to include the No Safety Effect classification.

d. Identification of the required system development assurance level.

e. A statement outlining what was considered and what assumptions were made when evaluating each failure condition (e.g., adverse operational or environmental conditions and phase of flight).

The goal in conducting this step is to clearly identify the circumstances and severity of each failure condition along with the rationale for its classification.

6.1 (Continued):

Since it is possible that use of common architectures or complex components in separate systems could introduce additional aircraft-level failure conditions involving multiple functions, the FHA should identify and classify these new failure conditions. When aircraft-level functions are integrated by a system or combination of systems, the FHA should be re-evaluated to identify and classify failure conditions involving multiple functions. If the FHA is constructed in system-oriented sections, traceability of hazards and failure conditions between the aircraft-level and system-level is necessary.

Implementation choices made during development may introduce common causes for multiple aircraft-level failure conditions or interactions between systems resulting in malfunction. These common causes could cross system or function boundaries. A review of the implementations of systems should be performed to determine if there are such conditions and if they should be added to the aircraft-level FHA (see 6.4).

6.2 Preliminary System Safety Assessment:

A PSSA is used to ensure completeness of the failure conditions list from the FHA and complete the safety requirements. It is also used to demonstrate how the system will meet the qualitative and quantitative requirements for the various hazards identified. The PSSA processes identify derived system safety requirements and may identify the need for alternative protective strategies (e.g., partitioning, built-in-test, dissimilarity, monitoring, and safety maintenance task intervals, etc.). The PSSA outputs should be used as an input to the SSA and other documents, including, but not limited to, system requirements, software requirements, and hardware requirements.

The PSSA is an iterative analysis embedded within the overall development. This is an ongoing process starting in the early phases of the development with the allocation of aircraft functions and their requirements to the system level. System-level requirements are then allocated to items, and, finally, item requirements are allocated to hardware and software (see Appendix A) Common Cause Analysis should determine the applicable separation and isolation requirements to be included in the PSSA.

The PSSA should identify failures and combinations of failures contributing to the failure conditions identified in the FHA. Possible contributing factors leading to failure conditions can be identified by using Fault Tree Analyses (FTA), Dependence Diagrams, Markov analysis, or other analysis methods. Hardware faults and possible software/hardware errors, as well as faults arising from common causes, should be included in the analysis to show their contribution and to derive what system, item, hardware, and software safety requirements are needed. This process is the basis for the assignment of numerical risk budgets to random hardware failures of items (see 5.5).

6.2 (Continued):

Calculating the event probability of a failure condition should include the time during which a latent failure can persist. The time exposure during which backups and/or protective mechanisms could fail and/or remain failed prior to repair should be considered. In many cases, the failures are detected by normal flight crew observation, or during periodic power-up or self test routines. In some cases, the detection of latent failures is associated with the interval between equipment shop tests or specific aircraft maintenance tests. These tasks and time intervals are identified during the PSSA and verified in the SSA by the use of FTA, Dependence Diagrams, Markov Analyses, or other similar analyses.

The inclusion of hardware and software errors, plus operational errors, in a qualitative manner in this analysis shows their contribution and can provide valuable information on deriving the needed development assurance levels (see 5.4). Also the PSSA can be used to identify specific safety-related requirements for hardware and software, such as containment boundary definitions, partitioning strategies, and specific verification strategies. The existence of such requirements should be captured as part of the system requirements.

All safety requirements should be traceable from the PSSA to the system requirements.

6.3 System Safety Assessment:

A system safety assessment (SSA) is a systematic, comprehensive evaluation of the implemented system functions to show that relevant safety requirements are met. The analysis process is similar to the activities of the PSSA, but different in intent. The PSSA is used to derive system and item safety requirements, whereas the SSA is used to verify that the implemented design meets those safety requirements.

The SSA combines the results of the various analyses to verify the safety of the overall system and to cover all of the specific safety considerations identified in the PSSA. The SSA process data includes results of the relevant analyses and substantiation. A typical SSA might include:

a. The list of agreed-to external event probabilities

b. A system description including contained functions and interfaces

c. The list of failure conditions (FHA, PSSA)

d. The classification of each failure condition (FHA, PSSA)

e. The results of qualitative analyses for failure conditions (FTA, FMES, Markov analysis, dependence diagrams)

f. Confirmation that any hazards resulting from the implementation of this system in combination with other systems implementations have been addressed

6.3 (Continued):

g. The results of quantitative analyses for failure conditions (FTA, FMES, Markov analysis, dependence diagrams)

h. The results obtained from Common Cause Analyses

i. A list of safety maintenance tasks and intervals (FTA, FMES, Markov analysis, dependence diagrams)

6.4 Common Cause Analysis:

Advisory material relating to systems establishes the need to address common cause faults (AC/AMJ 25.1309). The potential for this type of fault exists in any system architecture that relies on redundancy or uses components or software that are also used by other systems. The need to provide a fail-safe design will serve to separate a function from its applicable backups and/or protective mechanisms, or may separate redundant backups and/or protective mechanisms from each other. Once the applicable separation and isolation requirements have been identified, the common cause analysis should proceed to address the common cause fault potential across each boundary, and should identify the fault containment strategies to be used, along with the rationale supporting the fault coverage provided.

This type of fault can also be caused by common development errors.

Common-cause fault sources often fall into one of the following categories:

a. Software design error
b. Software coding error
c. Requirements error
d. Repair process error
e. Environmental factors
f. Hardware failure
g. Hardware design error
h. Compiler error
i. Production process error
j. Installation error
k. Operational error
l. Cascading failures

Common Cause Analysis is sub-divided into the following areas of study to aid in the assessment:

a. Zonal Safety Analysis
b. Particular Risks Assessment
c. Common Mode Analysis

6.4 (Continued):

These analyses may be performed at any stage of the design process. Obviously, the most cost-effective time is early in the design process because of the potential influence on system architecture. However, confirmation may not always be feasible until implementation is complete.

A Zonal Safety Analysis should examine each physical zone of the aircraft to ensure that equipment installation and potential physical interference with adjacent systems do not violate the independence requirements of the systems.

A Particular Risk Assessment should examine those common events or influences that are outside the system(s) concerned but which may violate independence requirements. These particular risks may also influence several zones at the same time, whereas zonal safety analysis is restricted to each specific zone. Some of these risks may also be the subject of specific airworthiness requirements.

The Common Mode Analysis provides evidence that the failures assumed to be independent are truly independent. The analysis also covers the effects of design, manufacturing, and maintenance errors and the effects of common component failures.

6.5 Safety-Related Flight Operations or Maintenance Tasks:

The functions allocated to aircraft operations and maintenance personnel result in tasks and procedures that may have an associated safety requirement. Safety-effects of identified failure conditions may be resolved through assignment of specific tasks or identification of specific limitations to these personnel. Where safety-related tasks or limitations form part of the certification substantiation, they should be identified and recorded in the certification data (see 4.1.2). For certification maintenance requirements, see AC 25.XX.

7. VALIDATION OF REQUIREMENTS:

Validation of requirements and specific assumptions is the process of ensuring that the specified requirements are sufficiently correct and complete so that the product will meet applicable airworthiness requirements. Validation is a combination of objective and subjective processes.

In showing compliance with JAR/FAR 25.1301 and JAR/FAR 25.1309, the validation process supports the development of requirements from functional needs and safety considerations. This development should generate a complete set of requirements. The validation process addresses each of these requirements. While the format is left to the developer's definition, a structured process should be defined in the validation plan (see 7.7.1).

Ideally from the point of view of facilitating a smooth development process, requirements should be validated before design implementation commences. However, in practice, particularly for complex and integrated systems, the necessary visibility of the whole set of consequences that flow from the requirements may not be obtainable until the system implementation itself is available and can be tested in its operational context. In consequence, validation is normally a staged process continuing through the development cycle. At each stage the validation activity provides increasing confidence in the correctness and completeness of the requirements.

7. (Continued):

The validation process at each level of the requirements hierarchy should involve all relevant technical disciplines, including the safety assessment process. Experience indicates that careful attention to requirements development and validation can identify subtle errors or omissions early in the development cycle and reduce exposure to subsequent redesign or inadequate system performance.

Individual tests may simultaneously serve the purposes of verification as well as validation when the system implementation is used as part of the requirements validation process. One purpose of this activity is to check that the requirements are met by the implemented system, while a separate purpose is checking that the requirements are appropriate to the context in which the system is operating. Such dual purposes should be reflected by coordination of the verification and validation plans.

7.1 Validation Process Objectives:

Ensuring correctness and completeness of requirements are the objectives of the validation process. Errors in the definition of system requirements can arise from three primary causes: (1) ambiguity, (2) incorrect statements, or (3) incomplete statements (i.e., omissions). The validation process should adequately cover all of these potential deficiencies. Examination of requirements to ensure they are both necessary and sufficient is a key aspect of validation. A further objective of the validation process is to limit the potential for unintended functions in the system or for unintended functions to be induced in interfacing systems.

For the purpose of this document, correctness and completeness are defined as follows:

a. Correctness of a requirement statement means the absence of ambiguity or error in its attributes.

b. Completeness of a requirement statement means that no attributes have been omitted and that those stated are essential.

7.2 Validation Process Model:

Requirements and assumptions should be validated at each hierarchical level of requirements definition. This includes validation of requirements at the aircraft function, system and item levels as well as validation of the FHA. Generally, validation of requirements and assumptions at higher levels serves as a basis for validation at lower levels.

The relationship of validation to system development is shown in Figure 2. An expanded process model is shown in Figure 5. Inputs to the validation process may include a description of the system (including the operating environment), the system requirements, a definition of system architecture, and the development assurance level.

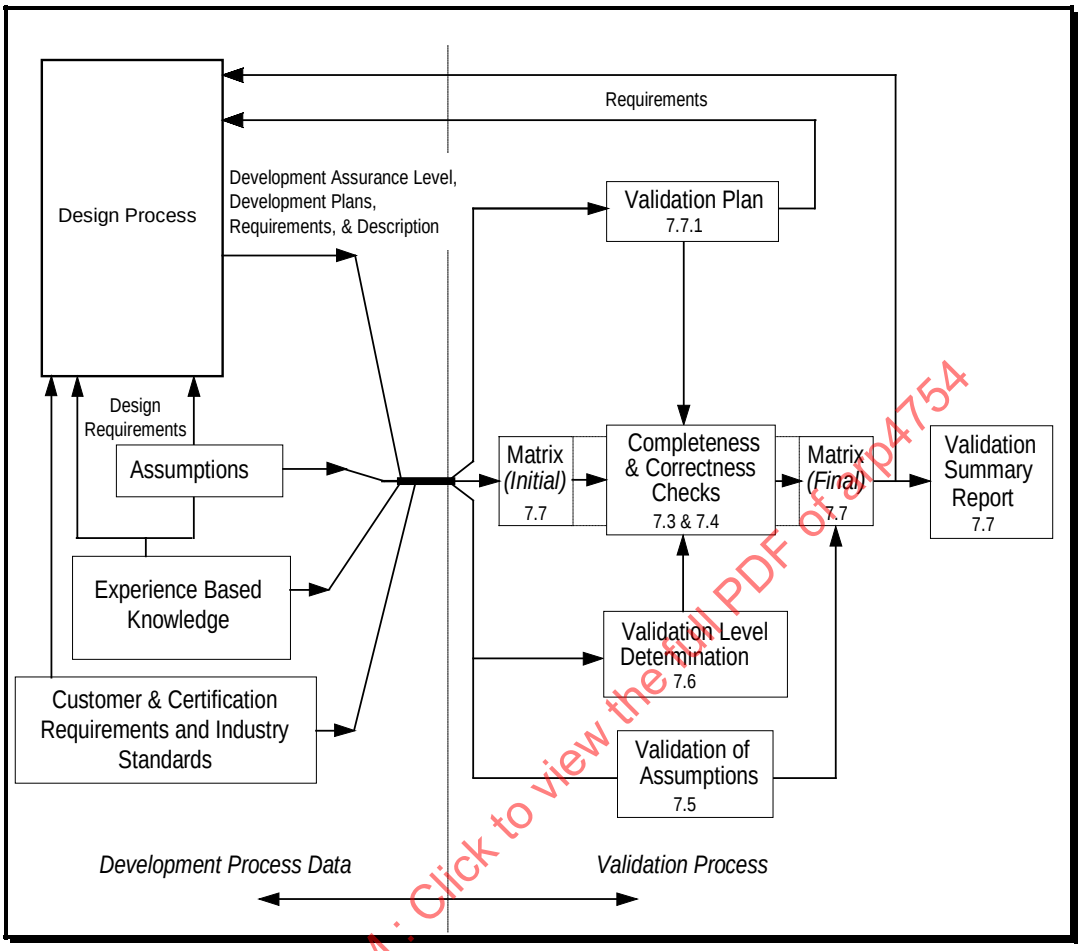

FIGURE 5 - Validation Process Model

7.2 (Continued):

An overview of the requirements and assumptions validation process is outlined below. These processes may be used for validation at the various hierarchical levels. These processes, or suitable alternatives, may be used to support certification.

a. Validation Plan:

The validation plan should define the specific methods to be used for validation of system requirements and assumptions. (Additional information on validation planning is provided in 7.7.1.)

7.2 (Continued):

b. Determination of the Level of Validation:

The necessary level of validation is determined by the development assurance level of the function addressed by the requirement (see 7.6).

c. Completeness and Correctness Checks:

The checks for requirements completeness and correctness may require engineering judgment, as well as analysis or test. The validity of such judgments will be established more easily if they and their supporting rationale are recorded at the time that the related requirement is developed. (Additional information on these checks is provided in 7.3 and 7.4.)

d. Validation of Assumptions:

The process of validation of assumptions focuses on ensuring that assumptions are explicitly stated, appropriately disseminated, and justified by supporting data (see 7.5).

e. Validation Matrix:

This process includes preparation of a validation matrix (see 7.7.3) that references requirements and validation results, including, as appropriate, those for hardware/software performance, derived requirements, environmental and operational considerations, assumptions and supporting data. The source of each requirement should be identifiable. This matrix should be updated regularly during the development and included in the validation summary.

f. Validation Summary:

Data describing the process, as well as the results, can be an effective means of ensuring that communication is based on a consistent and balanced understanding of the issues significant to the system design (see 7.7.4).

7.3 Completeness Checks:

The following is an example set of questions for assessing completeness at each hierarchical level of requirements. This list should be tailored for the specific application.

a. Do requirements trace to identified sources?

(1) Intended functions — aircraft-level, system-level
(2) All functions, hazards, and failure condition classifications identified in FHA
(3) All failure conditions incorporated in PSSA
(4) Derived requirements — design decisions or assumptions
(5) Applicable regulatory standards and guidelines
(6) Anticipated operating environment
(7) Established flight operations or maintenance procedures

b. Are constraints and assumptions adequately defined, substantiated and addressed?

(1) Market considerations
(2) Safety considerations (e.g., FHA, FMEAs, PSSAs)
(3) Environmental constraints
(4) Industry and company standards

c. Has the system implementation been adequately specified?

(1) All aircraft and system functions fully allocated
(2) All interfaces defined — internal, external, physical, functional, human
(3) System architecture defined and requirements allocated to hardware and software

d. Are prohibited behavior characteristics explicitly stated?

7.4 Correctness Checks:

During the validation process both the correctness of failure condition classification and the correctness of the stated requirements content should be reviewed and justified. Correctness checks should be carried out at each level of the requirements hierarchy. The following questions may help assess correctness of requirements. This list should be tailored and expanded for the specific application.

a. Are all requirements correctly stated?

(1) What is required (as opposed to how it should be designed)
(2) Unambiguous
(3) Statements leading to appropriate design
(4) Realizable and verifiable to the level of rigor appropriate to the system development assurance level
(5) Stated for all required environmental conditions
(6) Stated for degraded and normal modes
(7) Derived requirements are correct and supported by analysis
(8) Source(s) of each requirement identified

b. Are assumptions correct?

(1) Significant to/inherent in the requirements
(2) Documented
(3) Traced
(4) FHA failure condition classification assumptions confirmed

c. Do requirements correctly reflect the safety analyses?

(1) Appropriate safety analyses completed correctly
(2) All system hazards identified and classified correctly
(3) Impact of unsafe design or design errors
(4) Reliability, availability, and fault tolerance requirements

7.5 Validation of Assumptions:

In the majority of system development programs, a number of assumptions (or judgments) are made that are not directly provable at the time the information is needed. This is specifically true of the initial version of the FHA. The existence of such assumptions is not, by itself, a certification or a safety concern. However, the possibilities for miscommunication about the basis and scope of such assumptions are numerous and the related consequences can jeopardize satisfactory implementation of safety requirements. Thus, assumptions (either explicit or implicit) should be identified and their reasonableness and rationale established based on the specific system and its development assurance level.

7.5 (Continued):

Assumptions may be used early in the development process as a substitute for more explicit knowledge that will be available later. In these cases, validation consists of showing that the explicit knowledge or acceptable rationale was indeed obtained and that any inconsistencies between the explicit knowledge and the related assumption were resolved.

The process of validation of assumptions focuses on ensuring that assumptions are:

a. Explicitly stated
b. Appropriately disseminated
c. Justified by supporting data

The processes used to validate assumptions may include: reviews, analyses, and tests. Where the consequences of an erroneous assumption appear to have significant potential to reduce safety, one possible validation strategy consists of showing how the system design, in fact, limits or bounds the achievable consequences of an assumption error.

The remainder of this section provides guidance for identifying and judging the reasonableness of assumptions. To facilitate this purpose, assumptions are categorized as:

a. Operational / environmental
b. Design
c. Manufacturing (or producibility)
d. Serviceability
e. Installation

Within each category, examples of specific assumptions and related considerations are provided.

7.5.1 Operational and Environmental Assumptions: The operational assumptions include those associated with:

a. Air traffic
b. Maintenance
c. Cargo
d. Personnel
e. Flight dynamics
f. Air traffic control systems
g. Aircraft or engine performance
h. Operational procedures
i. Passengers
j. Policies and goals of the operator and applicable governmental organizations

7.5.1 (Continued):

Environmental assumptions involve the physical conditions in and around the aircraft or through which it is expected to operate. As a minimum these include: atmospheric, electromagnetic and lightning conditions; and hazardous objects and materials.

Some examples of operational assumptions include:

a. Exposure times
b. Traffic densities
c. Maintenance intervals
d. Performance limitations

Accepting assumptions in these categories may be accomplished by review of related engineering data, airline/operator experience, or similarity to other conditions or circumstances that are explicitly known.

7.5.2 Design Related Assumptions: Design related assumptions are grouped into crew interface, system interface, and reliability. Accepting assumptions in this area may be accomplished by review against existing industry experience and practice.

a. Crew Interface Assumptions:

The crew interface assumptions may include the interaction of the crew with the equipment and the operational environment under normal and emergency conditions, crew member performance characteristics (e.g., response times, display interpretation, physical limitations), and crew interaction. Some examples of assumptions about the crew interface are: crew response times to various types of messages, event recognition times (e.g., recognition of hardovers), decision making strategies, perceptual error rates, the discrimination accuracy on the basis of physical shape, visual form, color, or dynamic performance.

b. System Interface Assumptions:

System interface assumptions may address issues associated with the meaning or logical interpretation of the data exchanged (e.g., format, integrity, latency, resolution) or they may focus on the physical characteristics of the data signal (e.g., voltage levels, impedance, signal-to-noise ratio).

Some examples of assumptions about the system interface would be the probability of misreads of data bus information, correct processing of fault data by all related interfacing systems, fault containment, and immunity to external faults.

7.5.2 (Continued):

c. Reliability Assumptions:

Reliability topics for which assumptions are often made may include:

(1) the adequacy of failure rate modeling over the life cycle

(2) dispatch inoperative considerations

(3) the adequacy of scheduled maintenance tasks and their frequency

(4) the adequacy of parts derating

(5) consideration of potential failure latency and exposure periods

(6) the completeness of the failure modes analysis

(7) the adequacy of test data to establish or demonstrate MTBF predictions

(8) the applicability of in-service proven parts

7.5.3 Manufacturing and Producibility Assumptions: Manufacturing and producibility assumptions include the effectiveness of inspection and production test. Acceptance of these assumptions may be accomplished by review against company standards and practices.

a. Inspection Assumptions:

The engineering analyses usually assume that the inspection system adheres to company and relevant standards (i.e., FAA/JAA).

b. Production Test Assumptions:

Production test may be assumed to adequately verify that the equipment manufacturing process will maintain the equipment compliance with the specification (operation, environment and safety wise) during the equipment production life. Production test may also address the performance defects that cannot be easily detected by normal functional testing, (e.g., functionality of protection devices).

Typical assumptions may include:

(1) factory test tolerances ensure serviceability

(2) test tolerances do not degrade safety

(3) special manufacturing tests are defined to detect errors that would otherwise remain undiscovered

7.5.4 Serviceability Assumptions: It is usually assumed that provisions for service and repair do not degrade safety. This assumption may be validated by review of service and maintenance procedures, and associated equipment.

7.5.5 Installation Assumptions: Typical assumptions about installations may involve separation, isolation, cable binding, wire sizing, environment, power hook-up, circuit breaker sizing, ventilation, drainage, sources of contamination, mount integrity, grounding and shielding. Validating assumptions in this area may be accomplished by review against industry standards and practice, selective testing and/or inspections of mockup, prototype, or production drawings/hardware.

7.6 Validation Rigor:

The level of rigor of validation is determined by the development assurance level(s) of the system, item, or partitioned portion addressed by the requirement. (see 5.4). Each development assurance level and the basis for it should be validated. Validation methods are identified in 7.6.1 and their acceptable use is described in 7.6.2.

7.6.1 Validation Methods: Several methods may be needed to support validation. These methods include: traceability, analysis, modeling, test, similarity, and engineering judgment. Validation should consider both intended and unintended functions. Intended function requirements validation involves evaluation against objective pass/fail criteria. Vigilance during all analysis and testing can be used to identify unintended system/item operations or side-effects. While the absence of unintended functions can not be validated directly, ad hoc testing and targeted analyses can be used to reduce the probability of their presence.

a. Traceability:

Traceability is an essential component of all validation. The requirement should either be traceable to a parent requirement, or by identification of the specific design decision or data from which the requirement was derived. An assumption should be traceable to a standard, practice, analysis, or test.

b. Analysis:

A wide range of analysis methods and techniques may be used to determine requirements acceptability. Several specific safety-related analysis methods are described in ARP4761. Early discussion with regulatory authorities on the acceptability of the FHA and PSSAs will assist in the validation of the safety-related requirements.

c. Modeling:

Modeling of complex systems may be used to validate the requirements.

7.6.1 (Continued):

d. Test:

Special tests, simulations, or demonstrations may be used to validate requirements. These activities may occur at anytime during development based on availability of mock-ups, prototypes, simulations or end-item equipment. Care should be exercised to ensure any simulation is sufficiently representative of the actual system, its interfaces, and the installation environment.

e. Similarity (Service Experience)

This method allows validation of a requirement by comparison to the requirements of similar in-service certified systems. The similarity argument gains strength as the period of experience with the in-service system increases. Arguments of similarity should not be used until there is adequate confidence that the period of experience is satisfactory and any safety-related problems of the in-service system are understood and resolved. Service experience can be classified into the following two types:

(1) Directly Applicable Similarity – Can be claimed if the two systems/items have the same function and failure condition classification, and operate in the same environment with similar usage.

(2) Applicable Similarity – Can be claimed if the two systems/items perform similar functions in equivalent environments.

f. Engineering judgment

Application of personal experience through reviews, inspections and demonstrations can support determination of completeness and correctness. The properly justified engineering judgment (i.e., the rationale/logic used) should be traceable.

7.6.2 Recommended Methods: Table 6 identifies validation methods and data as a function of the allocated development assurance level A-E. For example, to validate requirements to level A or B, analysis, tests of intended function, and directly applicable similarity may be used to establish correctness and completeness. Validation of some requirements may use one method for correctness checks and another method for completeness checks.

TABLE 6 - Requirements Validation Methods and Data

Methods and Data (see 7.6.1.a-f and 7.7)	Development Assurance Level - A and B	Development Assurance Level - C	Development Assurance Level - D	Development Assurance Level - E
PSSA (see 6.2)	R	R	A	A
Validation Plan	R	R	A	N
Validation Matrix	R	R	A	N
Validation Summary	R	R	A	N
Requirements Traceability	R	A	A	N
Analysis, Modeling, or Test	R	One recommended	A	N
Similarity (Service Experience)	A	One recommended	A	N
Engineering Judgment	A	One recommended	A	N
Cross System Implementation Effects	R	A	A	N

R - Recommended for certification
A - As negotiated for certification
N - Not required for certification

For each requirement, a combination of the recommended and allowable methods, necessary to establish the required confidence in the validation of that requirement, should be identified and then applied.

7.7 Validation Data:

7.7.1 Validation Plan: A requirements validation plan should be in place throughout the development process. This plan should outline how the requirements and assumptions will be shown to be complete and correct. The plan should include descriptions of:

a. The methods to be used.
b. The data to be gathered or generated.

7.7.1 (Continued):

c. What should be recorded (such as: summaries, reviews, or investigations).
d. The means for timely access to requirements validation information.
e. How the status of validation will be maintained, or managed, when changes are made to requirements.
f. Roles and responsibilities associated with the validation.
g. A schedule of key validation activities.

Aspects of the validation process that may also serve as part of verification should be coordinated with the verification plan.

7.7.2 Supporting Data and Records: Data and records need to meet the following criteria if they are to be used to support certification:

a. The data and records should be retrievable for later reference.

b. The source of the data generated, such as by analysis or test, and the methods used, should be sufficiently controlled so as to allow regeneration of the same data.

This provides archived evidence for future enhancements, problem resolution, and review by certification authorities.

7.7.3 Validation Tracking: A validation matrix or other adequate approach is desirable to track the status of the requirements validation process. The level of detail should depend upon the development assurance level of the function addressed by the requirement and should be described in the validation plan. It is recommended that a preliminary tracking process be described in the certification plan, and that it be updated as required. The final data should be included in the validation summary. The specific format is up to the applicant, but it should at least contain:

a. Requirement or Assumption
b. Source of the Requirement or Basis for the Assumption
c. Associated Function(s)
d. Development Assurance Level
e. Validation Method(s) Applied
f. Validation Conclusion (Valid/Not valid)

7.7.4 Validation Summary: The validation summary should provide assurance that the requirements were properly validated. The summary should include:

a. A reference to the validation plan and a description of any significant deviations from the plan.
b. The validation matrix, as described in 7.7.3.
c. Identification of supporting data or data sources (see 7.7.2).

8. IMPLEMENTATION VERIFICATION:

The purpose of verification is to ascertain that each level of the implementation meets its specified requirements.

The verification process ensures that the system implementation satisfies the validated requirements. Verification consists of inspections, reviews, analyses, tests, and service experience applied in accordance with a verification plan. These activities are described in the paragraphs that follow.

8.1 Verification Process Objectives:

The verification process:

a. Confirms that the intended functions have been correctly implemented.
b. The requirements have been satisfied.
c. Ensures that the safety analysis remains valid for the system as implemented.

8.2 Verification Process Model:

Figure 6 shows an overview of a generic process model for verification at each level of system implementation.

The verification process is composed of three distinct activities described as follows:

a. Planning: Includes planning for the resources required, the sequence of activities, the data to be produced, collation of required information, selection of specific activities and assessment criteria, and generation of verification-specific hardware or software (see 8.3).

b. Activities: Includes the activity in which the verification methods are employed (see 8.4).

c. Data: Includes evidence of the results developed in the process (see 8.5).

Level of verification is determined by the development assurance level of the system or item (see 8.4.5).

The inputs to the verification process include the set of documented requirements for the implemented system or item and a complete description of the system or item to be verified.

More than one verification method may be necessary to substantiate compliance with the requirements. For example, an analysis may be required in conjunction with a physical test to assure that worst case issues have been covered.

During the process of verifying intended functions any anomalies recognized (such as an unintended function or incorrect performance) should be reported so that they can be reviewed.

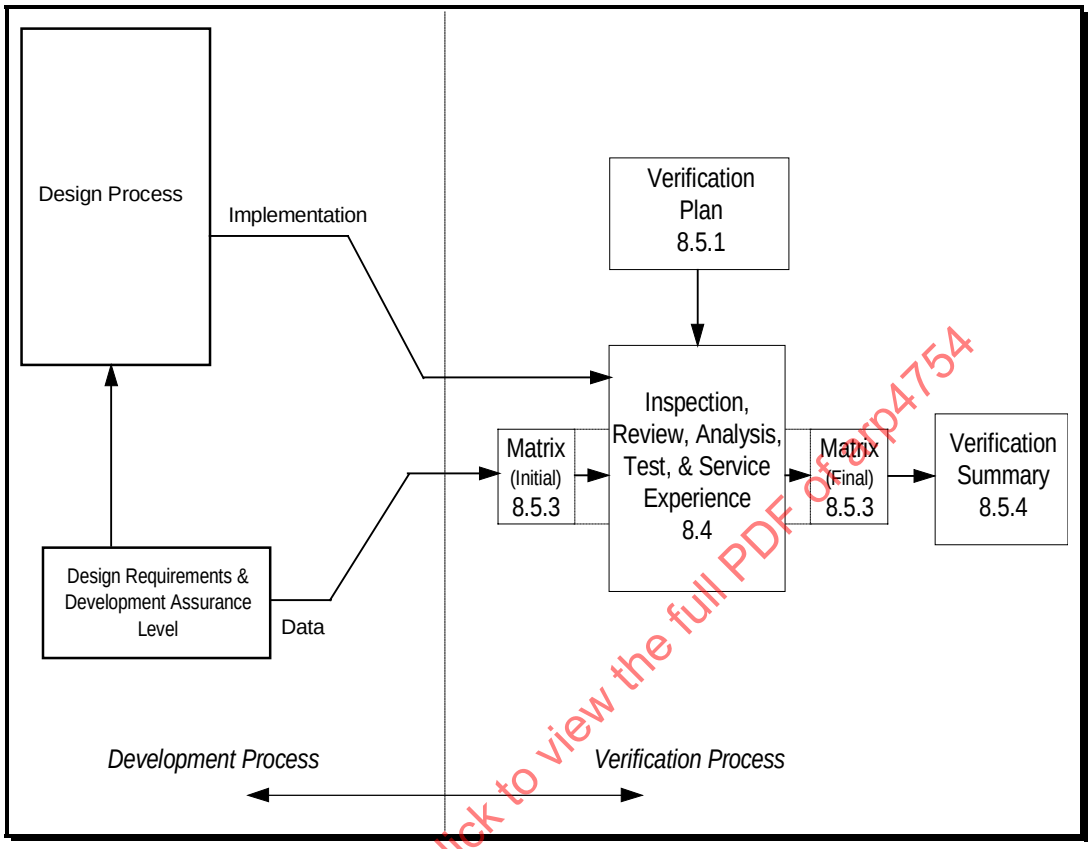

FIGURE 6 - Verification Process Model

8.3 Verification Planning:

The purpose of this phase is to define the processes and criteria to be applied to the verification of each requirement to achieve the verification objectives. The following activities should be performed during the planning phase:

a. Identification of the system or item configuration, including the definition of any special test equipment, facilities, and any special hardware or software features to be verified.

b. Collation of all requirements appropriate to the level under consideration, including derived requirements and their traceability

c. Definition of the specific verification methods to be employed to show compliance with each requirement, based on the development assurance level.

8.3 (Continued):

d. Definition of the criteria to be used to assess the evidence resulting from each verification method applied.

e. Identification of system verification credit taken for hardware or software verification activities.

8.4 Verification Methods:

The purpose of these activities is to verify that the implemented system satisfies its functional and design requirements including the intended operating environment. Four basic methods may be employed in the verification of any system or item:

a. Inspection and Review
b. Analysis
c. Test
d. Service Experience

Each of these methods is discussed in the paragraphs that follow.

8.4.1 Inspection and Review: Inspection and review is performed to demonstrate adequate consensus that the product complies with its requirements. Generally, a checklist or similar aid is used. The typical types of reviews performed are as follows:

a. Inspection that the system or item meets established physical implementation and workmanship requirements.

b. Design reviews showing how the system or item is expected to perform in normal and nonnormal conditions.

c. Test reviews establishing the applicability of test cases to system or item requirements.

8.4.2 Analysis: An analysis provides evidence of compliance by performing a detailed examination (e.g., functionality and performance) of a system or item. Analysis methods include, but are not limited to, those described in the following paragraphs.

8.4.2.1 Modeling: Modeling of complex, deterministic systems may be entirely computational or a combination of computation and test. Modeling may be used for system parameter evaluation, to provide early system information, or other purposes.

8.4.2.2 Coverage Analysis: Coverage analysis is performed to determine the degree to which the requirements are addressed throughout the development and verification activities. This is typically implemented using some form of traceability.

8.4.3 Testing: Testing provides repeatable evidence of correctness by exercising a system or item to verify that the requirements are satisfied. Testing has the following two objectives:

a. To demonstrate that the system or item implementation performs its intended functions. Testing an intended function involves evaluation against objective pass/fail criteria established by the safety requirements.

b. To provide confidence that the implemented system does not perform unintended functions (i.e., not consciously part of the design) that impact safety. Ad hoc testing, and special vigilance during normal testing, may be used to identify unintended system or item operation or side-effects. It should be noted that complete absence of unintended function can never be established by test.

Tests are performed on all or part of the physical system or item or an appropriate validated model using procedures documented in sufficient detail so that a second party could reproduce the test results. Problems uncovered during testing should be reported, corrective action tracked, and the modified item retested.

For each test, the following should be specified:

a. Required inputs – variability should be considered in setting the test criteria.
b. Actions required.
c. Expected results and the tolerances associated with those results.

Test result data should contain, as a minimum, the following:

a. The version of the test specification used.
b. The version of the system or item being tested.
c. The version or reference standard for tools and equipment used, together with applicable calibration data.
d. The results of each test including a PASS or FAIL declaration.
e. The discrepancy between expected and actual results.
f. A statement of success or failure of the testing process including its relationship to the verification program.

8.4.4 Similarity / Service Experience: Verification credit may be derived from design and installation appraisals and evidence of satisfactory service experience on other aircraft using the same or other systems that are similar in their relevant attributes. This method should use documented experience along with engineering and operational judgment to demonstrate that no significant failures remain unresolved in these installations. See 11.4.2 for more detail.

8.4.5 Recommended Verification Activities: Table 7 lists a variety of recommended and allowable verification methods and data as a function of the development assurance level. The necessary scope and coverage associated with these methods and data also depends on the development assurance level and, if known, may be further influenced by the specific related fault condition.

For example, an implementation being verified to level A or level B may involve inspection or review and analysis, and should involve some form of test. The extent to which each method needs to be applied or data developed will be the result of agreement with the certification authorities, based on the specific system to be certified.

TABLE 7 - Verification Methods and Data

Methods and Data (see 8.4 and 8.5)	Development Assurance Level A and B	Development Assurance Level C	Development Assurance Level D	Development Assurance Level E
Verification Matrix	R	R	A	N
Verification Plan	R	R	A	N
Verification Procedures	R	A	A	N
Verification Summary	R	R	A	N
SSA (see 6.3)	R	R	N	N
Inspection, Review, Analysis, or Test (note 1)	Test and one or more of others	One or more	A	N (note 2)
Test, unintended function	R	A	A	N
Service Experience	A	A	A	A

R - Recommended.
A - As negotiated.
N - Not required for certification.

Note 1: These activities provide similar degrees of verification. The selection of which activities will be most useful may depend on the specific system architecture or the specific function(s) implemented.
Note 2: Installation and environmental compatibility should be shown.

8.5 Verification Data:

The purpose of verification data is to provide evidence that the verification process was conducted as planned. This evidence may be required for compliance substantiation and to support certification data requirements described in 4.3 and 4.4. A reasonable approach is to maintain a verification matrix during development and to produce a verification summary report.

Requirements for software verification are included in DO-178B and for hardware verification in DO-xxx, a summary of software and hardware verification should be included in the verification data of the system in which it is embedded.

8.5.1 Verification Plan: The verification plan establishes the strategies to show how the implementation satisfies its requirements. A typical verification plan might include:

a. Roles and responsibilities associated with conducting the verification activities.
b. A description of the degree of independence of the design and verification activities.
c. Application of verification method(s).
d. Data to be produced.
e. Sequence of dependent activities.
f. A schedule of key verification activities.

Some aspects of the verification process may also support validation of specific requirements and should be coordinated with the validation plan.

8.5.2 Verification Procedures and Results: Data describing the verification procedures together with the results achieved provides the evidence necessary to establish the appropriateness of the verification effort.

8.5.3 Verification Matrix: A verification matrix or an equivalent tracking document should be produced to track the status of the verification process. The level of detail of this matrix should depend on the development assurance level of the system or item being verified. While the specific format may be determined by the applicant, it should contain, at least:

a. Requirement
b. Associated Function
c. Development Assurance Level
d. Verification Method(s) Applied
e. Verification Conclusion (Pass or Fail)
f. Verification Coverage Summary (relating procedures and results to system or item requirements)

8.5.4 Verification Summary: The verification summary provides visibility for the evidence used to show that the system or item implementation satisfies its requirements. The summary should include:

a. A reference to the verification plan and a description of any significant deviations from the plan.

8.5.4 (Continued):

b. The verification matrix as described in 8.5.3.

c. A reference to the problem report system (as appropriate for the development assurance level).

d. A description of any open problem reports and an assessment of the related impact on safety (as appropriate for the development assurance level).

e. Identification of supporting data or data sources (see 7.7.2 for supporting data criteria) (as appropriate for the development assurance level).

9. CONFIGURATION MANAGEMENT:

This section discusses the objectives and activities of the system configuration management process. It is applicable to the system, item(s) that make up the system, certain facilities and tools, and the required certification data. The Configuration Management Plan should include the methods used to show the objectives of the configuration management process are satisfied.

The existence of an independent entity or organization to perform the configuration management activities should not be implied by the title or content of this section.

9.1 Configuration Management Process Objectives:

The objectives of the configuration management process are to provide:

a. Technical and administrative control of the configuration of:

(1) System requirements

(2) Items that implement the system

(3) Applicable certification data (Table 2)

(4) Facilities and tools, where configuration is essential to establishing development assurance certification compliance

b. Control of changes to the system configuration or certification data by:

(1) Providing a known point for review – identifying modification status and change control by control of system configuration in relation to a configuration baseline.

(2) Providing controls that ensure that identified problems and their resolution are recorded, approved, and implemented.

c. Assurance that physical archiving, recovery, and control are maintained for relevant system data.

9.1 (Continued):

d. Traceability of system compliance with requirements.

Configuration management is both a system development and a certification activity. In terms of certification schedule, a configuration baseline, along with appropriate change control procedures, should be established by the point in the system development where certification credit is first desired. The traceability of the final proposed configuration to that configuration baseline is a necessary element of demonstrating development assurance.

9.2 Configuration Management Process Activities:

The configuration management process includes configuration identification, problem reporting/change control, and archiving/retrieval activities. Continuity of these activities significantly enhances their effectiveness and the credibility of the overall configuration management process. For certification purposes, evidence of a continuous configuration management process may include, but is not limited to, historical records or successive reports from these activities.

9.2.1 Configuration Identification: The objective of the configuration identification activity is to label unambiguously each item and certification data (and their successive versions) so that a basis is established for their control and reference. If applicable, each separately controlled component or software version of an item should be labeled. The format for the labeling is the responsibility of the applicant.

9.2.2 Problem Reporting and Change Control: The objective of certification-related problem reporting is to record certification-related problems and their resolutions. Change control provides a means for the evaluation and approval of certification-related changes. The following guidelines highlight the aspects of problem reporting and change control that are significant in demonstrating development assurance:

a. Means should be established to document the nature and resolution of problems and changes to an item, its data, or certification data. The data should include any changes resulting from derived requirements.

b. Change control should preserve the integrity of the items and certification data by providing protection against unauthorized change.

c. Change control should ensure that any change to an item or certification data is appropriately identified by a change to its configuration identification.

d. Change control should ensure that changes to an item require applicable changes to the data associated with the item.

9.2.3 Archive and Retrieval: The objective of the archive and retrieval activity is to ensure that the certification data associated with the system and related items can be retrieved. Data retention procedures should be established to satisfy airworthiness requirements. The following guidance is provided:

a. Data associated with the system or item should be retrievable from a controlled source (for example, the organization or company that developed the system).

b. Procedures should be established to ensure the integrity of the stored data for as long as may be required by the certification authority. These procedures should include:

(1) Ensuring that no unauthorized changes can be made.

(2) Selecting storage media that minimize regeneration errors or deterioration.

(3) Exercising and/or refreshing archived data at a frequency compatible with the storage life of the medium.

(4) Storing duplicate copies in physically separate archives that minimize the risk of loss in the event of a disaster.

10. PROCESS ASSURANCE:

This section describes the activities that ensure the system development and supporting processes are appropriate, maintained, and followed. The assurance activities described are not intended to imply or impose specific organizational structures or responsibilities.

10.1 Objectives of Process Assurance:

The objectives of the process assurance activities are:

a. To ensure the necessary plans are in place or developed, and then maintained for all aspects of system development.

b. To ensure development activities and processes are conducted in accordance with those plans.

c. To ensure evidence is available to show the implementation of the plans.

10.2 Process Assurance Plan:

The Process Assurance Plan describes the means to assure that the practices and procedures to be applied during system development are followed. Particular emphasis should be placed on the certification-related activities. The following issues should be considered when producing the Process Assurance Plan:

a. The scope and content of the other project plans (development, certification, validation, verification, and configuration management) are consistent with the development assurance level of the system or item.

b. Project communications, coordination and sequencing, and progress monitoring mechanisms are defined.

c. Change control, operational, and maintenance procedures are defined.

d. Sufficient project review opportunities are defined to best achieve the timely detection of development errors.

e. Sufficient coordination with the certification authorities is planned.

10.3 Project Plan Reviews:

The following issues should be considered when assessing the project plans:

a. Applicable procedures and practices are documented.

b. Defined communication practices ensure the timely exchange of information between the applicable processes and affected personnel.

c. Procedures for plan updates due to process, schedule, or technical changes are defined.

d. Plan updates are appropriately tracked and controlled.

10.4 Evidence of Process Assurance:

Evidence of conformance with the project plans can include:

a. Dated and approved project plans.

b. Reports, metrics, and summaries of reviews, as required by the plans.

c. Actual data developed from design, verification, validation, configuration management, and certification activities.

d. Confirmation (e.g., completed checklists and meeting minutes) of timely process assurance reviews.

11. MODIFIED AIRCRAFT:

Many of the processes in this document rely on product and process information obtained during aircraft and system development. Such data may not always be available for existing aircraft. In such cases, alternative means may be necessary to satisfy the certification requirements. The objective of this section is to describe how the material in this document could be applied when modifying an aircraft or system that was not developed in accordance with these guidelines.

11.1 Certification Basis:

The certification basis defines the applicable regulations to be met by the applicant and includes any special conditions required to supplement the published regulations. When introducing a system or aircraft modification, the certification authority considers the impact the modification has on the existing aircraft certification. In some cases, a supplement to the existing certification basis may need to be added (for example, when a radically different technology is employed).

11.2 Means of Compliance:

The applicant proposes a means of compliance that defines how it will be demonstrated that the system satisfies the certification basis. Whether or not the certification basis is changed, it will be necessary to assess the anticipated means of showing compliance to ensure compatibility with the agreed certification basis.

11.3 Considerations for Modification:

Modifications to existing aircraft may take various forms, including:

a. Introducing a new aircraft-level function.
b. Replacing one system with another on an existing aircraft.
c. Adapting an existing system to a different aircraft type.
d. Altering an existing system on an existing aircraft.

Since most highly-integrated or complex systems implement multiple functions, it is likely that a specific modification for such a system would involve more than one of these forms.

11.3.1 Introducing a New Aircraft-Level Function: A new aircraft-level function can be introduced into an aircraft type by modification of an existing system previously installed on the aircraft or by installing a new system. Considerations for addressing the introduction of a new aircraft-level function include:

a. The applicant should develop the new aircraft function in accordance with Sections 3 through 10 of this document. Emphasis should be given to the following:

(1) The Functional Hazard Assessment should address the failure conditions and associated hazards for the new function and identify the safety objectives for the systems to be modified.

(2) The FHA should also identify and substantiate the manner in which other functions and systems are affected by the introduction of the new aircraft function. This may be achieved by conducting analysis on functional interactions and interdependencies, and by determining the degree to which the aircraft function is integrated with other aircraft functions.

b. If credit is sought for development assurance activities performed on a previously certificated “baseline” aircraft or system, the proposed system or item and its certification data should be traceable to that baseline.

c. If certification data does not exist or is not available to the applicant for unchanged areas of the aircraft on which the aircraft function is reliant, the applicant should identify and substantiate the assumptions that were made about those areas to support the results of the safety assessment process.

11.3.2 Replacing One System With Another on an Existing Aircraft: Installing a replacement system in a previously certified aircraft type may change the implementation of an aircraft function or functions without adding a new aircraft-level function. If a new aircraft-level function is added, see 11.3.1. The replacement system may be installed for a number of reasons including: replacement of obsolescent equipment, improvement of reliability or integrity, or in compliance with a regulatory change. Considerations for addressing the installation of replacement systems in previously certified aircraft type include:

a. The applicant should review the results of the existing safety assessment process considering: (if not available, see c.)

(1) Installation of the replacement system.
(2) Areas on which the replacement system is reliant.
(3) Availability of certification data for the system being replaced.
(4) Certification basis for the aircraft.

It may be necessary to develop safety assessment data to ensure that the safety objectives for the replacement system are correct and complete and to ensure that the aircraft-level safety objectives have been satisfied.

11.3.2 (Continued):

b. If credit is sought for development assurance activities performed on a previously certified "baseline" aircraft or system, the proposed system or item and its certification data should be traceable to that baseline.

c. If certification data is not available for unchanged areas of the aircraft on which the system is reliant, the applicant should identify and substantiate the assumptions that were made about those areas to support the results of the safety assessment process.

d. The applicant should develop the replacement system in accordance with Sections 3 through 10 of this document. Emphasis should be given to the following:

(1) The Functional Hazard Assessment should address the failure conditions and associated hazards for the replacement system and identify the safety objectives for the systems to be modified.

(2) The FHA should also identify and substantiate the manner in which other functions and systems are affected by the introduction of the replacement system. This may be achieved by conducting analysis on functional interactions and interdependencies, and by determining the degree to which the aircraft function(s) performed by the replacement system is integrated with other aircraft functions.

11.3.3 Adapting an Existing System to a Different Aircraft Type: Systems previously approved for operation in one aircraft type must be reexamined for use in a different aircraft. When the applicant chooses to seek credit from the previous certification, the certification authority will require evidence that the design, installation, and application are similar. When this evidence is not available, the relevant parts of Sections 3 through 10 may be applied as necessary to provide evidence that the system to be installed satisfies the safety objectives. Considerations for addressing installation in a different aircraft include the following:

a. The applicant should review the results of the existing safety assessment process considering:

(1) Similarity of system installation and operation on both the existing and proposed aircraft.
(2) Similarity of functions on which the transferred system is reliant.
(3) Impact of the transferred system on other systems in the proposed aircraft.
(4) Certification basis.
(5) Adequacy of the certification data available from the previous installation.

If the safety objectives are the same for the proposed installation as they were in the previous installation and provided that an appropriate level of aircraft similarity is established, no additional effort will be required.

11.3.3 (Continued):

Otherwise, the safety assessment process should identify and substantiate the old functions affected by the new installation. The assessment should address unchanged functionality on which the new installation relies. This may be achieved by conducting analyses of the interaction and interdependencies between the new system and other aircraft systems.

b. If credit is sought for development assurance activities performed on a previously certified "baseline" aircraft or system, the proposed system or item and its certification data should be traceable to that baseline.

c. If certification data is not available for unchanged functions of the aircraft on which the system is reliant, the applicant should identify and substantiate the assumptions that were made about those functions to support the results of the safety assessment process.

d. The applicant should supplement the existing certification data in accordance with the guidelines of 11.4.1 under the following conditions:

(1) The certification data from the previous installation is not available to substantiate that the new installation satisfies the safety objectives of the proposed installation.

(2) The certification data from the previous installation is inadequate to define and substantiate the affected area.

e. Any requirements impacted by the new installation should be validated and the new installation verified in accordance with the guidance provided in Sections 7 and 8.

11.3.4 Altering a System on an Existing Aircraft: An alteration to a previously approved system may change the implementation of an aircraft function without adding a new aircraft-level function. If a new aircraft-level function is added, see 11.3.1. An alteration may result from a change in requirements or desired performance, correction of an implementation error, or an enhancement to equipment reliability. Considerations for addressing a system alteration on a previously certificated aircraft include:

a. The applicant should review the results of the safety assessment process considering the impact of the alteration on:

(1) Areas on which the system is reliant.
(2) Availability of certification data.
(3) Certification basis for the aircraft.
(4) Existing requirements (including the impact on unchanged requirements).
(5) System architecture.

The safety assessment process should identify and substantiate the area affected by the alteration. This may be achieved by conducting, in accordance with the relevant parts of Section 5, an analysis of the interaction of the altered system with other aircraft systems.

11.3.4 (Continued):

b. If credit is sought for development assurance activities performed on a previously certified "baseline" aircraft or system, the proposed system or item alteration and its certification data should be traceable to that baseline.

c. If certification data is not available for unchanged areas of the aircraft on which the system alteration is reliant, the applicant should identify and substantiate the assumptions that were made about those areas to support the results of the safety assessment.

d. The applicant should supplement the existing certification data in accordance with the guidelines of 11.4.1 under the following conditions:

(1) The certification data from the previous installation is not available to substantiate that the altered system satisfies the safety objectives of the proposed installation.

(2) The certification data from the previous development is inadequate to define and substantiate area(s) affected by the system alteration.

e. The requirements impacted by the system alteration should be validated and the altered system implementation verified in accordance with the guidance provided in Sections 7 and 8.

11.4 Additional Considerations:

The guidelines contained in this document presume that evidence of an appropriately structured development and safety assessment process will be used to substantiate any development assurance requirements. If credit is sought for development assurance activities performed on a previously certified "baseline" aircraft or system, the proposed system or item and its certification data should be traceable to that baseline. In some cases, such evidence may not be adequate. The following paragraphs provide guidance for:

a. Supplementing the existing certification data to support certification of modifications to existing systems or new installations of previously certified systems.

b. Using system service history obtained from an installation on one aircraft type to support the certification of that system on a different aircraft type or a similar system on the same aircraft type.

11.4.1 Supplementing Existing Certification Data: To supplement existing certification data, the applicant may:

a. Evaluate the data available from the previous certification effort to determine which objectives of this document are satisfied for the new application and which objectives require additional consideration.

b. Use reverse engineering to develop certification data necessary to satisfy the objectives of this document.

c. Use service history in accordance with the guidelines of 11.4.2 to satisfy the objectives of this document.

d. Specify the strategy for accomplishing compliance with this document in the Certification Plan.

11.4.2 Use of Service History: Service history may be used to support certification of a new or modified system if an analysis shows the history to be applicable and changes to the referenced system configuration have been appropriately controlled and documented. This method allows validation of a requirement by comparison to the requirements of similar in-service certified systems. The similarity argument gains strength as the applicable period of system service experience increases. Arguments of similarity should not be used until any significant system problems experienced in service have been understood and resolved.

Considerations for the use of service history include:

a. The applicant should propose, in the Certification Plan, how service history will be used (e.g., the amount of service experience available and a description of how the service data will be analyzed).

b. The applicant should conduct an analysis to determine the extent to which the service history is applicable. Such an analysis should show that:

(1) Problem reporting procedures during the period of applicable service history were sufficient to provide an appropriate cross-section of in service problems.

(2) Changes to the referenced system during the service history period did not materially alter the safety or performance of the system.

11.4.2 (Continued):

(3) Actual usage of the referenced system during the service history period was consistent with the intended usage for the new or modified system. If the operational environments of the existing and proposed applications differ, additional validation and verification activities related to the differences should be conducted in accordance with Sections 7 and 8.

c. The applicant should analyze any reported safety-related problems, together with their causes and corrective actions, and establish whether or not they are relevant to the proposed system, system modification, or system application.

PREPARED BY SYSTEMS INTEGRATION SUBCOMMITTEE (FAA SYSTEMS INTEGRATION REQUIREMENTS TG (SIRT) OF COMMITTEE AS-1, AVIONICS/ARMAMENT INTEGRATION

APPENDIX A
AN OVERVIEW OF A GENERIC APPROACH TO AIRCRAFT SYSTEMS DEVELOPMENT

This section provides an overview of a generic approach for developing aircraft systems from conceptual definition of the desired functionality to certification. The purpose of this appendix is to establish common terminology and expectations associated with common processes and their interrelationships in order to understand the intent and applicability of the substantiating material. The generic process outlined in this section is intended to assist in establishing a framework for that understanding. This section does not imply a preferred method or process; nor does it imply a specific organizational structure.

A top-down sequence for developing a specific system implementation from knowledge of an intended aircraft function provides a convenient conceptual model for the system development process. Typical system development progresses in an iterative and concurrent fashion using both top-down and bottom-up strategies. In this document, primary attention is focused on the top-down aspect since it provides the necessary links between system development and aircraft safety for the purpose of substantiating system airworthiness compliance.

The following is a list of generic systems development process activities:

a. Identification of Aircraft-Level Functions, Functional Requirements, and Functional Interfaces
b. Determination of Functional Failure Consequences and Implications
c. Allocation of Functions to Systems and People
d. Design of System Architecture and Allocation of Requirements to Items
e. Allocation of Item Requirements to Hardware and Software
f. Hardware and Software Design/Build
g. Hardware/Software Integration
h. System Integration

Because of the iterative nature of all but the simplest development programs, the systems development process should be thought of as cyclic rather than sequential. Furthermore, the entry point for any given function may occur at any point in the cycle. For a new aircraft-level function, the process begins with the top level definition of requirements. For functional additions to an aircraft, the entry point may occur in the context of changes to a particular item. However, regardless of the entry point, an assessment of the impact of the new or modified function on other aircraft-level functions and their supporting requirements is necessary.

In practice, many of the development activities are concurrent and may involve interactive dependencies which lead to alteration of previously established requirements. Derived requirements can arise at any level and may alter or constrain design decisions associated with higher-level requirements (see 5.3).

A.1 GENERIC SYSTEMS DEVELOPMENT PROCESSES:

A.1.1 Identification of Aircraft-Level Functions, Functional Requirements and Functional Interfaces:

This activity begins with the establishment of basic aircraft performance and operational requirements. From these basic requirements, the functional requirements can be established and the functional interfaces with the external physical and operational environment identified.

Aircraft-level functions are high-level activities and are not necessarily associated with a single, physical system implementation. Typical aircraft functions include:

a. Flight Control
b. Ground Steering
c. Aircraft Aspects of ATC
d. Automatic Flight Control
e. Cargo Handling
f. Engine Control
g. Collision Avoidance
h. Ground Deceleration
i. Environmental Control
j. Passenger Comfort
k. Communication
l. Guidance
m. Navigation
n. Passenger Safety

The output of this activity is a list of aircraft-level functions and the associated functional requirements and interfaces.

A.1.2 Determination of Functional Failure Consequences and Implications:

A Functional Hazard Assessment at the aircraft level is performed to determine the classification of the failure conditions associated with each function. Guidelines for assigning failure condition classifications are contained in 6.1.

The output of this activity is the association of each aircraft function (see A.1.1) with specific failure conditions and the associated failure condition classifications based on hazard severity.

A.1.3 Allocation of Functions To Systems and People:

The next level of activity consists of establishing the appropriate grouping of aircraft functions and the allocation of the related requirements to people or systems. The process for selecting the appropriate functional grouping out of the range of possible groupings is often complex and controversial. No specific recommendations for accomplishing the grouping activity are provided in this document. However, careful attention to the basis for the selection decisions including related assumptions is fundamental to the success of subsequent processes. The functional groupings interact with the aircraft architecture and are the basis for system architecture development. While it is not necessary to know in detail how a system will be implemented to accomplish the necessary functional groupings, implementation constraints, failure effects, and life-cycle support may all play a significant role in selecting the most appropriate groupings. Assumptions that are made in the course of this process become a vital part of the overall system requirements package and are subject to the same validation activity as are other requirements.

The specific nature of each functional grouping plays a significant role in influencing which elements of the functions will be allocated to people and which will be allocated to machines. In either case the allocation should define inputs, processes performed, and outputs. Both operational and support aspects should be considered.

From the function allocations and the associated failure consequences, further specific system requirements necessary to achieve the safety objectives are determined. Derived requirements and additional assumptions will emerge during this phase as the consequences of the various combinations of functions, allocations to systems and to people are considered. These, in turn, may alter the aircraft-level function requirements.

The output of this activity is a set of requirements for each human activity and aircraft system together with associated interfaces. The interfaces should be defined with all inputs having a source and all outputs having destination(s), either human or another system.