A Process Model for the Development of Airborne Electronic Equipment

A DISSERTATION PRESENTED TO

THE SCHOOL FOR COMPUTER AND ELECTRONIC ENGINEERING

NORTH-WEST UNIVERSITY

POTCHEFSTROOM CAMPUS

As part of the fulfilment of the requirements for the degree

Magister Ingeneriae

in Computer and Electronic Engineering

by

D.A. Viljoen

supervised by

Prof. J.E.W. (Johann) Holm

November 2008

Acknowledgements

The author wishes to acknowledge contributions in terms of understanding and refining concepts described in this dissertation from the following persons: from Denel Aviation, messrs. Abhi Raghu, Andrew Douglas, Andries Jansen Van Rensburg, Anton Jacobs, Bernhard Meier, Chris Versluis, Danie Dreyer, Dewald Steyn, Dougie Lawson, Garth Tolmie, Jan van Niekerk, Johan (JC) Botha, Johan Zietsman, Jorge Pinto, Jimmy Nel, Justin Shulman, Kevin Ward, Kobus Pieters, Luke Sibisi, Nic du Plessis, Phil Smalman, Philip van Rooyen, Pieter Gerber, and Pieter Booyse; from Saab AB (Sweden), messrs. Carl Stocklassa, Rikard Johanssen, Lars-Olof Ohberg, Kjell Alm, and Anders Petterson; from Armscor, Mr Andre Kok and Mrs Madalein Young and from the South African Air Force, Mr. Philip Nell, Lt Col Hannes Oosthuizen and Lt Col Willie Möller.

Other mentors with whom I have had the privilege to discuss topics addressed in this study are Dr Jerry Lake, Prof Johann Kruger, and Prof Ad Sparrius.

I am indebted to my study leader, Prof Johann Holm for his guidance, enthusiasm, patience and motivation.

And finally, thank you to my wife Thelma and my family for their support in more ways that can be listed here.

Summary

Developments in systems engineering concepts and in the regulatory environment necessitated improvements to the processes used by Denel Aviation for the development of electronic equipment and software for use on board aircraft. A project was undertaken to improve the existing systems and software development processes.

Shortcomings in the existing development processes used by the organisation were identified.

A set of process requirements was determined, referring to general characteristics of airborne electronic equipment and to regulatory standards.

A process model, the Airborne Electronic System Development Process (AESDP) to be used by Denel Aviation, was developed. This proposed process model was designed to support incremental and iterative development. The process employs a strict requirements based development methodology in accordance with the standards and recommended best practises.

Important aspects of the proposed model include the following: (i) verification of requirement implementation commences in the definition stages of the project and (ii) a parallel (in time) breadboarding process is used to validate requirements and test implementation strategies and trade-offs - this is done without using the rigorous configuration management and other control process applicable to the life cycle data of the item under development.

The process model represents hierarchical development, i.e. system (or product), software and hardware layers. The organisational context of the model was delineated, project stages and decision gates were identified and different development “threads” and associated activities were described. Requirements for detailed methods and procedures associated with theses activities were identified.

The process model that resulted from this work, was approved by Denel and SAAB and is currently being applied to manage the upgrade of the Oryx helicopter fleet of the South African air force.

Opsomming

Ontwikkelinge op die gebied van stelselingenieurswese, asook veranderinge in die toepasike lugvaartregulasies, het Denel Lugvaart genoodsaak om hulle prosesse vir die ontwikkeling van elektroniese toerusting en verwante sagteware, vir gebruik aan boord vliegtuie en helikopters, te verbeter. ‘n Projek is van stapel gestuur om die bestaande prosesse vir die ontwikkeling van aan-boord stelsels en sagteware te verbeter.

Tekortkominge in die bestaande ontwikkelingsprosesse wat die maatskappy gebruik, is geïdentifiseer.

‘n Stel vereistes vir ‘n ontwikkelingsproses is bepaal, deur te verwys na eienskappe van elektroniese stelsels wat aan boord vliegtuie gebruik word, en na die toepaslike regulasies en standaarde.

‘n Prosesmodel (genoem AESDP, afgelei van Airborne Electronic System Development Process) om deur Denel Lugvaart gebruik te word, is ontwikkel. Hierdie model is ontwerp om inkrementele en iteratiewe ontwikkeling te steun. Die proses is op ‘n streng behoeftestellinggebasserde (requirements based) metodologie gebaseer, in ooreenstemming met aanvaarde praktyk en voorgeskrewe regulasies.

Belangrike aspekte van die proses sluit die volgende in: (i) verifikasie van implementering van behoeftes begin reeds in die konsepsuele stadiums van die projek en (ii) ‘n parallele subproses word bedryf waar behoeftestellings gevalideer kan word en waar konsepte getoets kan word sonder om die streng konfigurasie- en ander beheerprosesse te gebruik wat vir die item onder ontwikkeling gebruik word (“breadboarding”).

Die prosesmodel implementeer hiërargiese ontwikkeling, onderskeibaar tussen die vlakke van ontwikkeling van die stelsel, sagteteware, en hardeware. Die organisatoriese konteks van die proses is geïsoleer, projekfases en besluitnemingsafsnypunte is geïdentifiseer en verskeie parallelopende ontwikkelings fokusfunksies (“threads”) en gepaardgaande aktiwiteite is beskryf. Die vereistes vir gedetaileerde definisies van metodes en prosedures is geïdentifiseer.

Die prosesmodel, wat gevolg het as ‘n resultaat van hierdie werk, is goedgekeur deur Denel en SAAB en word tans aangewend om die opgradering van die Oryx helikopter vloot van die Suid-Afrikaanse lugmag mee te bestuur.

Abbreviations

AB	Aktie Bolag (Swedish – share company)
AESDP	Airborne Electronic System Development Process)
AFCS	Automatic Flight Control System
ARINC	Aeronautical Radio Incorporated
ARP	Aerospace Recommended Practice
AS	Aerospace Standard
ATP	Acceptance Test Procedure
CCB	Configuration / Change Control Board
CDR	Critical Design Review
CFT	Certificate for Flight Trials
CMMI	Capability Maturity Model Integration
COSPAS	Cosmicheskaya Sistyema Poiska Avariynich Sudov (Russian - Space System for the Search of Vessels in Distress)
CRT	Cathode Ray Tube
DDP	Declaration of Design and Performance
DEF STAN	Defence Standard (British)
DSI	Directorate System Integrity
EIA	Electronic Industries Alliance
FAA	Federal Aviation Authority
FAR	Federal Aviation Regulation
FCA	Functional Configuration Audit
FHA	Functional Hazard Assessment
FMECA	Failure Mode, Effect and Criticality Assessment

GPS	Global Positioning Satellite / System
HF	High Frequency
HMI	Human - Machine Interface
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronic Engineers
INCOSE	International Council on Systems Engineering
ISO	International Standards Organisation
JAR	Joint Aviation Regulation
LCD	Liquid Crystal Display
MAB	Military Airworthiness Board
MOC	Means of Compliance
MRI	Master Record Index
OCD	Operational Concept Definition
OT&E	Operational Test and Evaluation
PCA	Physical Configuration Audit
PRACAS	Problem Reporting and Corrective Action System
PSAC	Plan for Software Aspects of Certification
PSSA	Preliminary System Safety Assessment
Req	Requirement
RF	Radio Frequency
RFC	Request For Change
RSA	Republic of South Africa
RTCA	Radio Technical Commission for Aeronautics
SAAF	South African Air Force
SAE	Society of Automotive Engineers

SARSAT	Search and Rescue Satellite
SEMP	Systems Engineering Management Plan
SSA	System Safety Assessment
STD	Standard
TEMP	Test and Evaluation Master Plan
TSO	Technical Standard Order
UHF	Ultra High Frequency
URS	Users Requirements Specification
USB	Universal Serial Bus
VHF	Very High Frequency

Table of contents

Acknowledgements ..... **ii**

Summary ..... **iii**

Opsomming ..... **iv**

Abbreviations ..... **v**

Chapter 1: Introduction **1**

1.1 Introduction 1

1.2 Background 1

1.2.1 Modern avionics equipment 1

1.2.2 Airworthiness 3

1.2.3 The engineering of complex systems 4

1.2.4 Engineering processes 5

1.3 Purpose of this study 5

1.4 Summary 7

Chapter 2: Literature study and shortcomings **8**

2.1 Introduction 8

2.2 Regulatory aspects 8

2.2.1 Airworthiness 8

2.2.2 Quality system 8

2.3 Development process detailed guidelines 9

2.4 Systems engineering principles 9

2.5 Specification practises 10

2.6 Compatibility with modern development tools 10
2.7 Organisational misalignment with development objectives 11
2.8 Iterative and incremental development 11
2.9 Process model selection considerations 12
2.10 Metrics 12
2.11 Summary of subject literature and normative standards 12
2.11.1 EIA-632 13
2.11.2 SAE ARP 7454 15
2.11.3 SAE AS9100 16
2.11.4 ISO/IEC 15288 17
2.11.5 IEEE 1220 18
2.11.6 INCOSE handbook 19
2.11.7 ISO/IEC 12207 21
2.11.8 RTCA/DO-178B 21
2.11.9 RTCA/DO254 22
2.12 Listed shortcomings 22
2.13 Summary 23
Chapter 3: Process requirements 24
3.1 Introduction 24
3.2 Process constraints 24
3.2.1 Characteristics of airborne electronic systems 24
3.2.2 Airworthiness 29
3.3 Process context, architecture and components 30

3.3.1 Development process context 30
3.3.2 Iterative and incremental development 30
3.3.3 Development stages 32
3.3.4 Development layers 32
3.4 Process main activities 32
3.4.1 Development planning 33
3.4.2 Development process control 34
3.4.3 Requirements driven development 37
3.4.4 System safety process and airworthiness certification 39
3.4.5 End product realisation 42
3.4.6 Verification 45
3.4.7 Infrastructure management 49
3.4.8 Prototyping 50
3.5 Selection of appropriate methods 52
3.6 Quality management system assessment 52
3.7 Summary 52
Chapter 4: Process description 59
4.1 Introduction 59
4.2 Process context 59
4.3 Process architecture 60
4.4 Process layers 61
4.5 Development life cycle stages 61
4.5.1 Concept stage 61

4.5.2 Definition stage 62
4.5.3 Design and development stage 62
4.5.4 Industrialisation stage 63
4.5.5 Production and system utilisation stages 63
4.5.6 Transition management 63
4.6 Developmental threads 63
4.7 Planning and control thread 63
4.7.1 Concept stage 64
4.7.2 Definition stage 66
4.7.3 Design and development stage 67
4.7.4 Industrialisation stage 71
4.7.5 Production and system utilisation stage 72
4.8 Requirements thread 73
4.8.1 Concept stage 73
4.8.2 Definition stage 74
4.8.3 Design and development stage 76
4.8.4 Industrialisation stage 76
4.8.5 Production stage and system utilisation stages 77
4.9 System safety / airworthiness / certification thread 77
4.9.1 Concept stage 77
4.9.2 Definition stage 78
4.9.3 Design and development stage 79
4.9.4 Industrialisation stage 79
4.9.5 Production stage and system utilisation stages 80

4.10 End product realisation thread 80
4.10.1 Concept stage 80
4.10.2 Definition stage 81
4.10.3 Design and development stage 81
4.10.4 Industrialisation stage 83
4.10.5 Production and system utilisation stage 83
4.11 Verification thread 83
4.11.1 Concept stage 84
4.11.2 Definition stage 84
4.11.3 Design and development stage 84
4.11.4 Industrialisation stage 86
4.11.5 Production stage and system utilisation stages 86
4.12 Infrastructure thread 86
4.12.1 Concept stage 86
4.12.2 Definition stage 87
4.12.3 Design and development stage 87
4.12.4 Industrialisation stage 88
4.12.5 Production and system utilisation stage 88
4.13 Breadboarding 93
4.14 Design guidelines for future detail design 93
4.14.1 Planning and control thread design guidelines 93
4.14.2 Requirements thread design guidelines 96
4.14.3 System safety/airworthiness/certification thread design guidelines 97
4.14.4 End product realisation thread design guidelines 99

4.14.5 Verification thread design guidelines 100
4.14.6 Infrastructure thread design guidelines 101
4.15 Derived Statement of Work (SOW) 101
4.16 Summary 102
Chapter 5: Conclusion and recommendations 103
5.1 Conclusion 103
5.2 Recommendations 103
References 104
Appendix A: Allocation of high-level requirements 106
Appendix B: Allocation of process characteristics 110
Appendix C: Statement of work 115
Appendix D: Glossary of systems engineering terms 143

Chapter 1: Introduction

1.1 Introduction

As a component of its business, Denel Aviation develops and modifies electronic equipment for use on board aircraft. Developments in systems engineering concepts necessitated improvements to the processes used by the company for the development, or changing, of these systems. A project was undertaken in cooperation with Saab AB (Sweden), (manufacturer of the Gripen fighter aircraft) to improve the existing systems development process. This project formed the basis for the work described in this study.

This work describes the development of a process model framework that shall be used as the process baseline for implementation over the following years. The aim is to provide a high-level (preliminary, not detailed) process architecture to be used in the improvement of systems engineering policies and procedures in the Denel Aviation Quality Management System. All detail work shall form part of further study as the process is developed in the future.

This chapter provides general background on modern avionics equipment, airworthiness, the engineering of complex systems and processes, as well as the purpose and scope of this study.

1.2 Background

1.2.1 Modern avionics equipment

Modern airborne electronic equipment include radio communication and navigation devices, automatic flight control systems, health and usage monitoring equipment, mission and flight management computers, inertial-, satellite- and Doppler navigation systems, radio altimeters, air data computers, heading and attitude reference systems, electronic warfare systems and electronically managed weapons systems.

Until the advent of practical embedded computers (early 1980s) electronic systems on board aircraft mostly operated independently of one another. A system with a particular function was essentially self contained and comprised its own input devices or sensors, output devices or actuators, processors, and human-machine interface (HMI) devices. A diagram of a typical system is shown in Figure 1. The inputs could be RF signals from an antenna (radio), rates and attitudes from gyroscopes (flight instruments and autopilots) or temperature and air pressures (bombing computer). The outputs could be audio signals, flight control commands to servos or

driver signals to e.g. seven segment displays. The HMI devices could be indicator lamps, dials, switches, and control knobs. The processing block would perform the required functions to convert the information from the inputs and input HMI devices to output voltages or currents and output display indications.

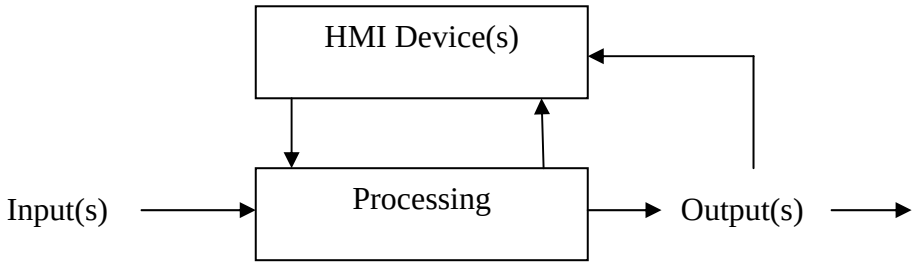

Figure 1: Stand-alone Airborne Electronic System

The development of embedded digital computers introduced intelligence and the capability to share data and other resources (e.g. HMI displays) between systems, thereby enhancing functional capabilities and improving effectiveness of aircrews, specifically under high workload conditions. This sharing of data and resources has lead to the notion of integrated (avionics) systems. A generalised schematic diagram for an integrated avionics system is shown in Figure 2. Flight data sensors include equipment for the measurement of aircraft attitudes, rates and accelerations, air pressures and temperature. Navigation sensors provide information about the geographical position of the aircraft. Health monitoring sensors include pressure sensors, thermocouples, flow rate meters, and other transducers, mounted on the engines, transmission devices, hydraulic-, fuel- and electrical systems.

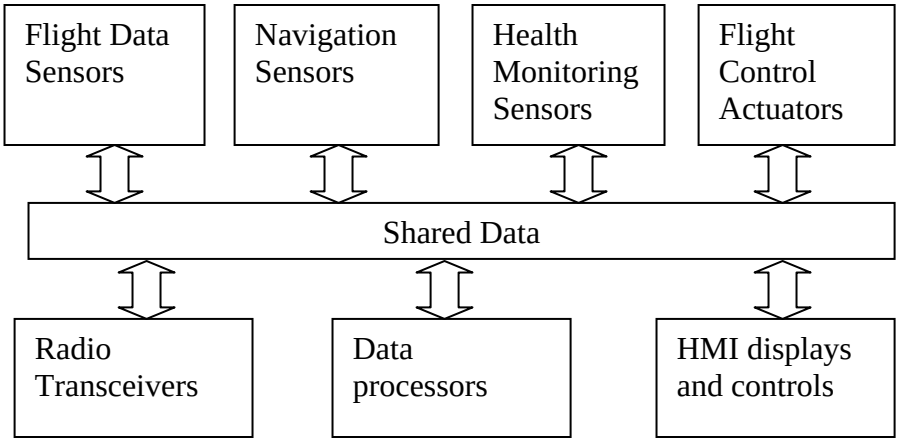

Figure 2: Integrated Avionics System

Flight control actuators typically employ servo devices to drive flight control surfaces or rotor system controls. Radio transceivers for airborne use cover HF, VHF and UHF ranges. The data processors are typically single task embedded computers. The HMI devices include multi-function displays, employing CRT or LCD technologies, keyboards and a variety of switches. Associated with these advanced capabilities is an increase in system complexity. A comprehensive description of avionics architectures is described by Spitzer [1].

Performance and *airworthiness requirements* became linked to the integrated system as a whole, and airworthiness could not be granted for the individual sub systems anymore, without taking cognisance of interactions between them.

The organisation who is contracted for the integration of the system on the airborne platform also assumes the responsibility for the airworthiness of the integrated and installed system. In many instances, this integrator is responsible for the specification of the sub systems as well as for the embedded software where the software is affected by the specifics of the aircraft. Typical examples are aircraft dynamics which have a bearing on the parameters of the automatic flight control system (AFCS) and the specific aircraft missions that determine the mission computer requirements.

1.2.2 Airworthiness

Airworthiness is managed worldwide by statutory organisations, such as the Federal Aviation Administration (FAA) in the United States, the Joint Aviation Authorities (JAR) in Europe and

the Civilian Aviation Authority (CAA) in South Africa. The prime objective of the airworthiness process is to ensure that aircraft are safe for use.

A manufacturer of aircraft or airborne equipment needs to demonstrate compliance with prescribed standards to the applicable authority for the specific country, (e.g. Federal Aviation Regulations of FARs published by the FAA in the USA) which then issues a Type Certificate for the aircraft type to the specific manufacturer. This Type Certificate permits the aircraft to be operated in civilian airspace and for commercial gain.

Materials, parts, processes and equipment not directly associated with a specific aircraft type, e.g. VHF radios and cockpit instruments, are granted a Technical Standard Order (TSO) approval, and may be used on type certificated aircraft. The standards (FARs) typically require proof of integrity of the design, adequate performance with respect to safe operation, acceptable handling qualities of the aircraft throughout the permissible operating envelope and reliability and redundancy of safety-critical sub-systems.

For military aircraft the process is essentially the same, but certification is normally granted by a military airworthiness authority.

It is important to note that the *engineering processes* at an organisation engaged in the development of airborne products need to be aligned with these *airworthiness requirements*.

1.2.3 The engineering of complex systems

The complexity of engineered systems has increased significantly since the end of the Second World War. Direct consequences of this increase in complexity were that cost overruns were often experienced, and in many instances the performance or functionality of systems did not meet expectations. A number of examples are cited by Bar-Yam [2]. It was realised that fundamental engineering approaches had to be improved to measure up to the challenges posed by emerging technologies and increased sophistication of user's requirements which encompass functional performance, safety, reliability, supportability and affordability facets of systems.

In addition to a proliferation of scientific literature on the subject, many military and industrial standards were developed in attempts to systematise methodologies with the objective of improving the efficiency by which complex systems could be developed, with varying degrees of success. A selection of these standards is discussed in the following two chapters.

Any organisation involved in the development of equipment utilising sophisticated technologies should stay abreast of developments in the discipline of systems engineering, and *ensure that the processes applied by the organisation* remain commensurate with these developments.

1.2.4 Engineering processes

The capability of any organisation to produce goods and services is dependent on the skills of its human resources, its infrastructure and the processes it employs. In particular, these processes should contain the combined learned experience of the organisation and should be adequately formulated and documented so that the optimum use can be made of the acquired collective knowledge. Major studies are undertaken to support the development of these capabilities, notably efforts from the Software Engineering Institute at the Carnegie Mellon University, culminating in the well known CMMI [3].

These processes are typically collected and managed in the company quality system, as required by international quality assurance authorities and described in quality system management standards such as ISO 9000 and AS 9100 [4]. In addition to providing visibility for use, a well described process facilitates efficient process improvement.

An organisation engaged in the development of complex systems should take particular care in ensuring that its process definitions adequately record outcomes of lessons learned and should make certain that the engineering processes are aligned with the environment wherein it operates.

1.3 Purpose of this study

Denel Aviation, (formerly Atlas Aircraft Corporation) has been engaged in the specification and development of airborne electronic equipment for use on military aircraft for more than two decades.

The author was the team leader for the development of a digital automatic flight control system (AFCS) for the Rooivalk attack helicopter developed for use by the South African Air Force (SAAF). This project involved the specification of hardware as well as the specification, development, verification and validation of embedded software. Experience gained from this project needs to be captured and formalised to improve associated processes in the organisation.

A number of factors (described in the next section) lead to the need to review the systems engineering methods employed at the organisation.

Most importantly, no generic systems development process model could be identified that was sufficiently applicable to the specific needs associated with the efficient development of equipment for airborne use.

A project was undertaken in co-operation with Saab AB of Sweden, starting in 2003, to address this situation. The results of this effort formed a basic framework to support further development of a comprehensive development process model.

This framework was also used for the planning of a program for the upgrading of the communications and navigation systems of the Oryx medium transport helicopters used by the SAAF.

In order to address the need for a customized process model (and associated support models), it was thus necessary to develop a model for the development of equipment for airborne use.

Therefore, the purpose of this study is to validate aspects of the development process model resulting from co-operation with Saab, to document relevant experience from other development projects, and to determine requirements for further improvement to the development process, ultimately leading to an improved process model definition.

The scope of the study is limited to the development process up to release of the system to service, or to updates of in-service equipment – *specifically related to the development of electronic systems and software for use on board aircraft*. Long term aspects of the system life cycle, such as obsolescence management and system disposal are not considered in this work, but do form part of the overall process. In addition, it is assumed that the very important business processes that precede and follow the development process are addressed outside the scope of this work.

A very important consideration is that this work is the result of the inputs obtained from a number of sources and people over a number of years. It is also the first baseline and iteration of an ongoing process and is thus a working document to be used as a management tool for process improvement.

Shortcomings of processes that existed at the time of commencement of the improvement exercise are discussed in Chapter 2.

The requirements for a process model suitable for the development of airborne equipment containing embedded software are considered in Chapter 3. The process model will be referred to as the Airborne Electronic System Development Process (AESDP) in the remainder of the text.

The description of a process model commensurate with the requirements is presented in Chapter 4.

1.4 Summary

In this chapter, the attributes of modern avionics equipment were described and essential concepts related to airworthiness of systems were introduced. Problems related to the development of complex systems were briefly mentioned and the significance of the use of well defined processes to direct the engineering effort was indicated. The purpose and scope of the study and a summary of the contents of the document were presented.

Chapter 2: Literature study and shortcomings

2.1 Introduction

This chapter presents an assessment of the engineering methods that were in use at Denel Aviation, a summary of existing subject literature and normative standards, a summary of requirements that follow from the literature and standards, and finally an identification of shortcomings of the existing Denel development processes.

2.2 Regulatory aspects

2.2.1 Airworthiness

Developers of airborne equipment need to demonstrate compliance with specific military or civilian standards, such as the Federal Aviation Regulations, produced by the US Federal Aviation Administration or the British DEF-STAN set of standards applicable to military aircraft.

Up to around 1994, military standards were used prevalently by the international military aerospace industry. Due to a significant drive in the USA to reduce the number of military standards, the industry at that time started to replace military standards with civilian aerospace standards, to which compliance had to be demonstrated.

As a rule, military standards were prescriptive in terms of processes to be followed, e.g. MIL-STD-498 [5], which describes comprehensive processes for the development of software. The modern civilian standards however, such as EIA-632 [6], IEEE 12207 [7], IEEE 1220 [8], ISO/IEC 15288 [9], and RTCA/DO-178B [10], specify the requirements to which development processes need to comply, rather than prescribing specific development methods. As a consequence, the organisation has to design its own development processes and then show compliance with the requirements of the standards.

The development processes at Denel Aviation were essentially in line with the military standards and needed to be changed so that they comply with the requirements stated by modern commercial aviation industry standards.

2.2.2 Quality system

The quality management system at Denel Aviation is aligned with SAE AS9100 [4] which is the most widely used quality standard in the international aerospace industry (it is the aerospace

equivalent of ISO 9001). The company is audited against this standard annually by an internationally recognised auditing organisation e.g. Bureau Veritas. Accreditation to this standard is required from the company's clients before work is authorised by them.

However, it was found that although a comprehensive set of procedures for system development exists, the existing quality management system in Denel did not provide users with an adequately encompassing view which facilitated, for example, project planning. Specific procedures were prepared to meet with the requirements of particular tasks when needed, but no overall relationship between tasks and associated procedures was defined. Many procedures directly associated with the creation of life cycle data as part of a systems engineering effort were also found to be inadequate, or there was a lack of synergy between procedures.

It was therefore required to develop a framework for a process which identified activities, required outcomes and associations between activities. This framework is to be used for the identification and indexing of required quality procedures, for project planning and for process improvement. In other words, this framework should provide the "big picture".

2.3 Development process detailed guidelines

A significant portion of the effort associated with the development of a certifiable airborne system is devoted to the preparation of life cycle data, such as development plans, specifications and test procedures. Documentation templates, guidelines, checklists and examples are required to aid workers in the development of these documents.

The support environment which existed in the company for the preparing of life cycle data in a consistent and interrelated approach was found to be insufficient and not adequately developed.

In addition, it was found that procedures for reviews, inspections and other tasks associated with the development process needed updating in order to be compatible with the processes described by newer standards.

2.4 Systems engineering principles

An objective of the systems engineering activities on a development program is to determine the functional, performance, safety, reliability and other operational requirements of the system, and then to ensure that these requirements are met in the system design and implementation at an acceptable level of integrity within practical constraints. Although systems engineering paradigms are well developed and a vast associated body of knowledge exists, as e.g. collated by Blanchard

and Fabrycky [11] and in the INCOSE Systems Engineering Handbook [12], as well as the technical standards referenced in 2.2.1, the organisation needs to refine and describe the detailed approach that suits its specific needs. This includes the *definition of project boundaries and interfaces, including development organisational layers, development life cycle stages and baselines and project decision making gates, transition criteria, process workflow descriptions and specific engineering methods*.

Significant emphasis is placed on the demonstration of compliance with functional, performance, safety and environmental requirements in the execution of military and aviation development contracts. The management of these requirements in terms of traceability to higher level requirements and the qualification status of the requirements need to be visible to all stakeholders.

Personal experience has shown that the methods used within Denel Aviation for requirements management did not always meet with customer expectations.

It was required to *determine the set of necessary and sufficient tasks to be performed to realise a qualified electronic system for airborne use, commensurate with user's requirements.* It was a further requirement that lessons learned from previous experiences should be addressed.

2.5 Specification practises

The company (and other parties in the defence industry) adhered to the specification practises described by MIL-STD-490 [13], which classified specifications according to the layer of hierarchy at which the product exists. The modern approach, advocated by e.g. EIA-632 [6], proposes a building block approach, where the format of requirements specifications are generically the same for each item developed under a program, on different product hierarchies. This means that the same type of documents are produced on system, sub system and enabling product layers, in stead of the previously used A-, B- and C-specifications.

The specification practises at Denel Aviation needed to be brought in line with modern approaches, and needed to be formalised in the company quality system.

2.6 Compatibility with modern development tools

The tools available to support the development of airborne equipment have improved significantly and keep on evolving. Improvements in software design methods, requirements management, software and modelling and simulation methods have rendered older development

approaches obsolete. As an example, the use of databases to manage traceability, e.g. DOORS®, can be cited.

The policies and procedures within Denel Aviation were in many instances not aligned with these developments.

A study was required to determine how the processes need to be structured to make optimal use of modern development aids.

2.7 Organisational misalignment with development objectives

The project organisation used to be structured according to different disciplines emphasising different objectives, namely systems engineering, quality assurance, configuration management, system safety, logistic support analysis and others. These different disciplines worked in parallel on projects, producing life cycle data according to the needs of the discipline, rather than to the needs of the project.

These parallel focuses were found to be unsupportive of an integrated approach which requires the synergistic utilisation of resources, e.g. time, manpower, and tools. This “parallel objectives” approach lead to duplication of effort in some cases, and oversights in others.

The identified process shortcoming in this instance is a lack of a common development process reference structure, leading to a decrease in the efficiency of the overall effort due to the development organisation not being aligned with process objectives.

2.8 Iterative and incremental development

The generic classical systems development model typically follows the path of requirements definition, system design, design realisation, integration and verification. As the main focus of the company was on the development of an air vehicle where this approach is essentially valid, it was expected from the developers of sub-systems to abide by the same principles. Although generally applicable, experience indicated that, for example, some requirements were only better understood after integration and flight testing, requiring updates to life cycle data developed earlier in the program. This then required re-engineering and re-testing, resulting in an unintended iterative process.

The process for development of equipment containing embedded software needed to provide more support for the development of requirements in the initial stages of the project.

Another consequence of the strict waterfall approach was that a substantive amount of work was produced before the verification activities were initiated, leading to very large amounts of source code to be inspected at a time as an example, which proved very difficult.

This problem can be alleviated by breaking the development effort up into smaller manageable "blocks" of functionality that strictly follow an iterative and incremental philosophy, where each new incremental block adds to the total functionality. Each block is fully qualified, i.e. verified and validated, before the next block is added. This concept of iterative and incremental development is described by Larman and Basili [32].

It is quite important to note that iterative and incremental development should not be confused with classic spiral models or perpetual development models, where baselines are not managed in a hard fashion. Iterative development, in Denel's environment, refers to added functionality at different layers, building iteratively from the bottom up (after a top-down design approach, however) to minimize lower-layer risk when upper layers are being developed.

2.9 Process model selection considerations

A number of models for the process to develop of a system can be applied. The selection of the model is dependent on the type of business in which that the enterprise is engaged. For an organization engaged in the development of equipment that needs to be certified for airborne use, the choice of model is limited to the set which is commensurate with the applicable governing regulations. This, by definition, rules out development approaches based on methods where traceability and evidence of requirement verification are not supported (e.g. Agile development).

2.10 Metrics

In order to determine the effectiveness of a development process, workers need to be able to measure the quality of the outcomes of the process and the efficiency of the associated effort.

No effective method for determining this effectiveness was employed at Denel Aviation.

Specific metrics needed to be identified and methods for collecting these needed to be implemented.

2.11 Summary of subject literature and normative standards

A summary of process descriptions found in the subject literature and industry standards are briefly summarised in the sub-sections below.

2.11.1 EIA-632

EIA-632 [6] (Clause 5) defines 33 requirements for system realisation, as summarised below.

Acquisition and Supply

- Supply Processes
  - Req. 1 – Product Supply
- Acquisition Process
  - Req. 2 – Product Acquisition
  - Req. 3 – Supplier Performance

Technical Management

- Planning Process
  - Req. 4 – Process Implementation Strategy
  - Req. 5 – Technical Effort Definition
  - Req. 6 – Schedule and Organisation
  - Req. 7 – Technical Plans
  - Req. 8 – Work Directives
- Assessment Process
  - Req. 9 – Progress Against Plans and Schedules
  - Req. 10 – Progress Against Requirements
  - Req. 11 – Technical Reviews
- Control Process
  - Req. 12 – Outcomes Management
  - Req. 13 – Information Dissemination

System Design

- Requirements Definition Process
  - Req. 14 – Acquirer Requirements
  - Req. 15 – Other Stakeholder Requirements
  - Req. 16 – System Technical Requirements
- Solution Definition Process
  - Req. 17 – Logical Solution Representations
  - Req. 18 – Physical Solution Representations
  - Req. 19 – Specified Requirements

Product Realisation

Implementation Process

Req. 20 – Implementation

Transition to use process

Req. 21 – Transition to Use

Technical Evaluation

Systems Analysis Process

Req. 22 – Effectiveness Analysis

Req. 23 – Trade-off Analysis

Req. 24 – Risk Analysis

Requirements Validation Process

Req. 25 – Requirements Statements Validation

Req. 26 – Acquirer Requirements Validation

Req. 27 – Other Stakeholder Requirements Validation

Req. 28 – System Technical Requirements Validation

Req. 29 – Logical Solution Representation Validation

System Verification Process

Req. 30 – Design Solution Verification

Req. 31 – End Product Verification

Req. 32 – Enabling Product Readiness

End Products Validation Process

Req. 33 – End Products Validation

This standard describes the application context as follows:

External environment, encompassing laws and regulations, legal liabilities, social responsibilities, technology base, labour pool, competing products, standards and specifications (national/international) and a public culture.

Enterprise environment, comprising policies and procedures, standards and specifications (corporate), guidelines, domain technologies and local culture.

Project environment, consisting of directives and procedures, plans, tools, project reviews and metrics. Within the environment for a specific project, project support processes, namely *project management* and *agreement support*, and process groups for engineering systems, comprising *acquisition and supply*, *technical management*, *system design*, *product realisation* and *technical*

evaluation are identified. The project environment is supported by enterprise support processes, including investment decisions, external agreements, infrastructure support, resource management, process management, production and field support.

2.11.2 SAE ARP 7454

SAE ARP 4754 [18] (section 3) describes an aircraft function implementation process model. A salient aspect of this process model is its promotion of an iterative development cycle. A simplified presentation of this process model is shown in Figure 3.

Appendix A1 of this standard presents an overview of a generic approach to aircraft systems development, under the following topics:

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 and Build

g. Hardware/Software Integration

h. System Integration

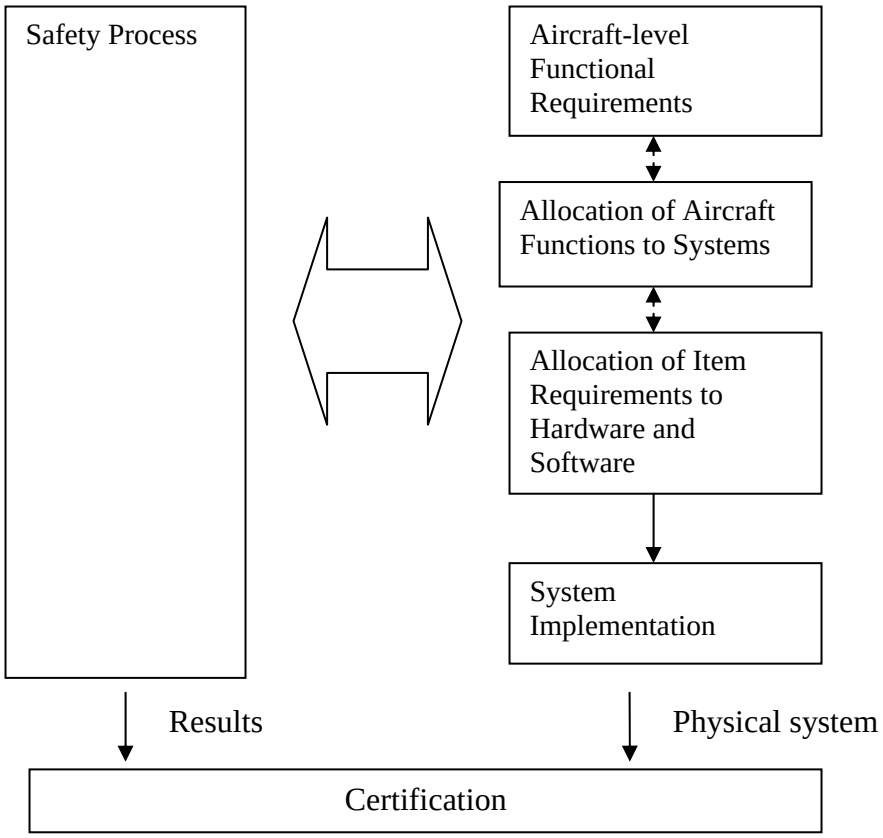

Figure 3: System Development Process Model (Adapted from SAE ARP4754)

Note: This standard is of particular significance in specifying processes for the development of airborne equipment.

2.11.3 SAE AS9100

Although SAE AS9100 [4] specifies the requirements for a quality management system, aspects from this standard apply to the development processes. Clause 7 of this standard deals with *product realisation*, identifying the following processes:

7.1 Planning of Product Realisation

7.2 Customer-Related Processes:

7.2.1 Determine Requirements

7.2.2 Review Requirements

7.2.3 Customer Communication

7.3 Design and Development

7.3.1 Design and Development (and control)

7.3.2 Design and Development Inputs

7.3.3 Design and Development Outputs

7.3.4 Design and Development Review

7.3.5 Design and Development Verification

7.3.6 Design and Development Validation

7.3.7 Control of Design and Development Changes

7.4 Purchasing

7.4.1 Purchasing Process

7.4.2 Purchasing Information

7.4.3 Verification of Purchased Product

7.5 Production and Service Provision (not part of the scope of the study).

It is required that each development procedure in the quality managements system is referenced to at least one of these clauses.

2.11.4 ISO/IEC 15288

ISO/IEC 15288 [9] presents a “common framework for describing the life cycle of systems created by humans”. The standard also presents “processes that support the definition, control and improvement of the life cycle processes …” It “does not detail the life cycle processes in terms of methods or procedures required to meet the requirements and outcomes of a process”.

The life cycle processes are described in clause 5 of the standard.

Four process groups are identified, namely, agreement processes, enterprise processes, project processes and technical processes.

The following processes are applicable to the AESDP:

5.3 Enterprise Processes

5.3.4 System Life Cycle Management Process

5.4 Project Processes

5.4.2 Project Planning Process

5.4.3 Project Assessment Process

5.4.4 Project Control Process

5.4.5 Decision-making Process

5.4.6 Risk Management Process

5.4.7 Configuration Management Process

5.4.8 Information Management Process

5.5 Technical Processes

5.5.2 Stakeholder Requirements Definition Processes

5.5.3 Requirements Analysis Process

5.5.4 Architectural Design Process

5.5.5 Implementation Process

5.5.6 Integration Process

5.5.7 Verification Process

5.5.8 Transition Process

5.5.9 Validation Process

Clause 6 specifies the need for, and Annex B identifies life cycle stages. Six stages are identified, i.e. Concept Stage, Development Stage, Production Stage, Utilisation Stage, Support Stage and Retirement Stage. Of these, only the Concept and Development stages are of interest to this study.

A useful diagrammatic representation of the ISO/IEC 15288 and ISO 12207 life cycle processes is presented in Annex C (not reproduced here).

2.11.5 IEEE 1220

IEEE 1220 [8] does not present a system life cycle process framework as in ISO/IEC 15288, but rather an approach for systems definition and management [op. cit.].

The standard identifies the systems engineering context in Annex A. This context definition is similar to the definition found in EIA-632 [6], and is summarised below:

External environment: laws, standards and regulations, natural constraints, induced constraints, technology base and competitive products.

Enterprise environment: policies and procedures, standards and general specifications and guidelines, resources and domain technologies.

Project environment: plans, teams, tools, controls, metrics. Within the environment for a specific project, a systems engineering process and manufacturing and test processes are identified. The systems engineering process is applied recursively and concurrently to development, manufacturing, verification, deployment, operations, support, training and disposal. Manufacturing and test processes for models, prototypes and final products include facilities; equipment and tools, procurement, fabrication/production (assembly and integration), test/verification, by-product disposal and packaging.

2.11.6 INCOSE handbook

The INCOSE handbook [12] is consistent with ISO/IEC 15288.

The life cycle processes and its context are presented in [op. cit] figure 1.1. This figure is redrawn in Figure 4 for ease of reference.

In terms of the AESDP, the System Life Cycle Processes Management process from the Enterprise Processes, the Project Processes, the Stakeholder Requirements Definition, Requirements Analysis, Architectural Design, Implementation, Integration, Verification, Transition and Validation processes from the Technical Processes are of interest.

For each of these processes, the handbook specifies inputs, activities, outputs, controls and enablers.

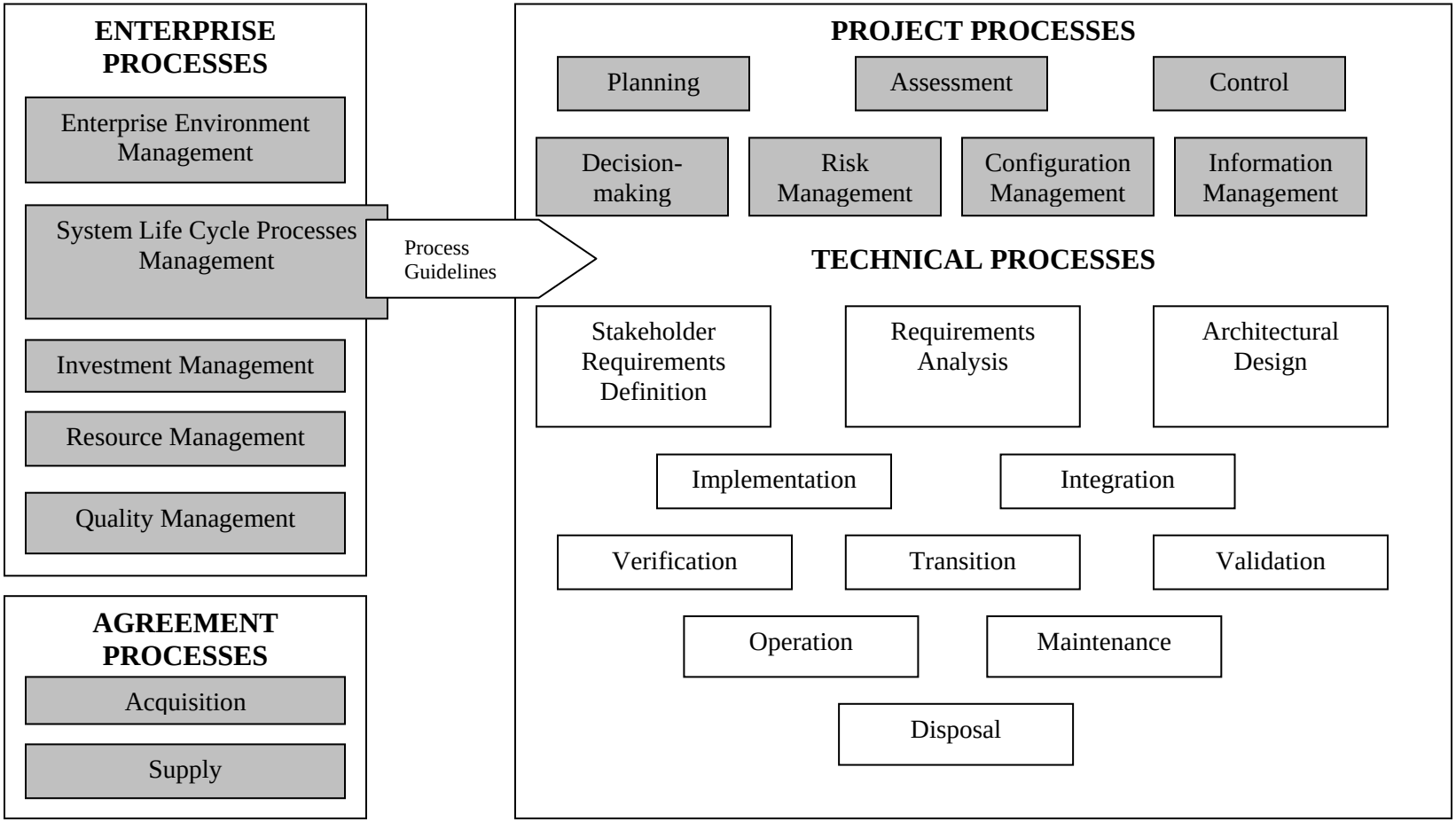

Figure 4: System Life Cycle Process Overview per ISO/IEC 15288, redrawn from INCOSE Systems Engineering Handbook v. 3 (figure 1.1)

2.11.7 ISO/IEC 12207

ISO/IEC 12207 [7] (Annex C) presents the software development processes in terms of a contract view, management view, operating view, engineering view and supporting view. For the purpose of this dissertation, only the engineering view is considered. This view is shown schematically in Figure 5.

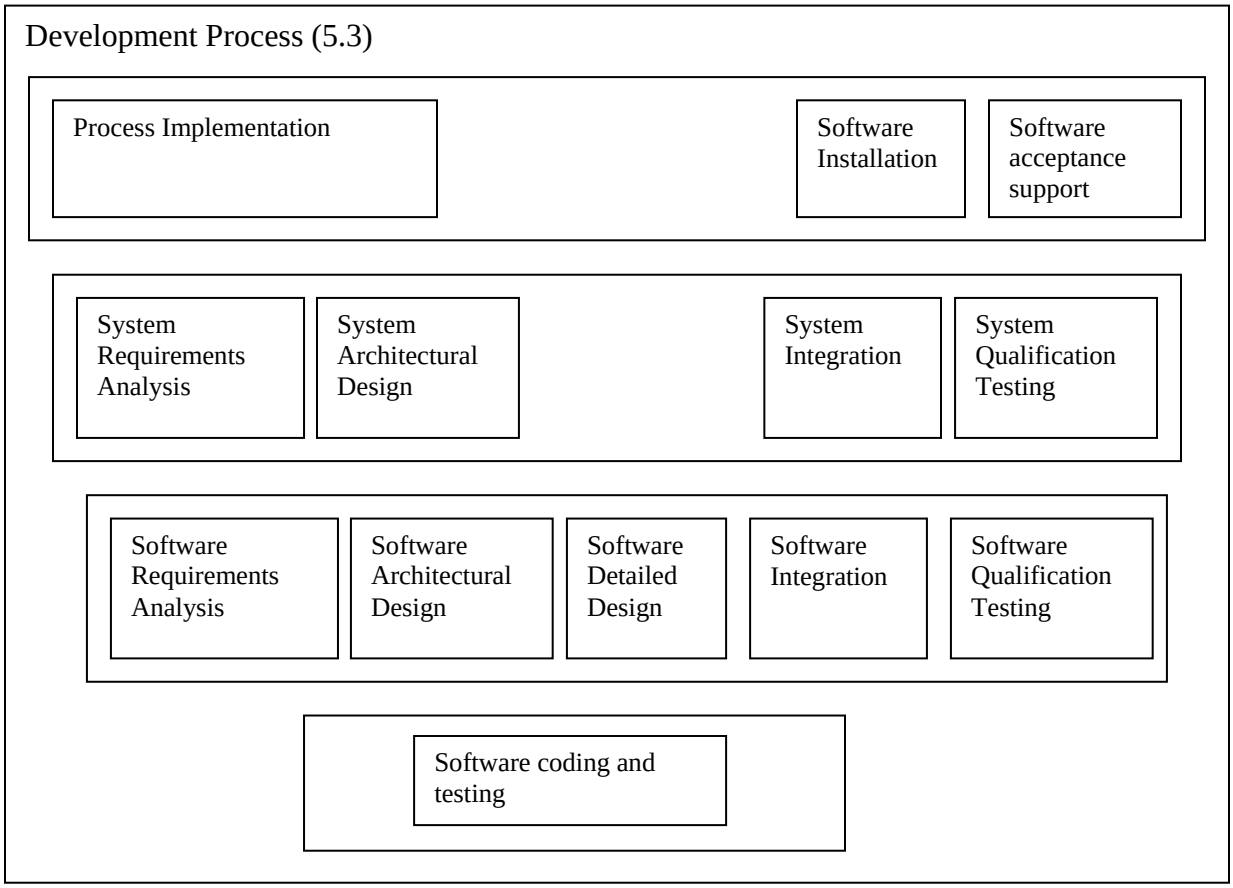

Figure 5: ISO/IEC 12207 presentation of software development process

2.11.8 RTCA/DO-178B

RTCA/DO-178B [10] deals with software development where the software is intended for use on airborne systems. This standard identifies three types of life cycle processes:

The *software planning process*, where the activities of the software development and integral processes are defined;

The *software development process,* which produces the software product. The standard identifies the software requirements process, the software design process, the coding process and the integration process as sub processes.

The *integral processes*, "that ensure correctness, control, and confidence of the software life cycle processes and their outputs". These include software verification, configuration management, and quality assurance and certification liaison. The standard also emphasises that these processes are performed concurrently with the development processes.

2.11.9 RTCA/DO254

RTCA/DO-254 [33] provides design assurance guidance for airborne electronic hardware, including hardware that contains complex electronic devices such as programmable logic devices.

The standard supports an iterative development process and identifies the following elements of the hardware design life cycle: Requirements Capture, Conceptual Design, Detailed Design, Implementation and Production Transition.

2.12 Listed shortcomings

The following shortfalls were identified during an analysis of the existing Denel process model. These are:

a. The military aircraft industry adopted civilian airworthiness standards in parallel with military standards introduced inconsistencies in methods used for specification and development of hardware and software;

b. The existing systems development process framework in the company was not defined to a sufficient level of detail to fully support detailed planning of, and execution of a development project, and not adequately visible to workers;

c. The specification practises at Denel Aviation are to be aligned with modern approaches;

d. The requirements and baseline management processes in cases fell short of customer expectations;

e. The process did not sufficiently support requirements development activities;

f. The process did not provide for a structured iterative development practice;

g. The guidelines and standards for preparation of life cycle data, e.g. style guides, documentation templates and worked examples, and checklists for use at reviews, are not sufficient, or are not in line with modern developments;

h. No effective method for measuring process effectiveness existed. This made it difficult to implement quality management.

2.13 Summary

The chapter commenced with a discussion of changes in the regulations that govern the development of airborne equipment for use on board military aircraft. The inadequacy of the policies and procedures and appropriate guidelines in the existing Denel Aviation Quality Management System were highlighted and the importance of following modern systems engineering concepts during system development was emphasized. The modern approach towards the development of system specifications was described briefly. Other considerations related to development models were also studied and summarized. A summary of process descriptions from normative industry standards were presented and the chapter was concluded with a summary of shortcomings of the previously used development processes.

Chapter 3: Process requirements

3.1 Introduction

Up to this point, all relevant requirements documents have been identified. What remains to be done, is to group and identify all relevant requirements.

The ***first*** part of this chapter deals with process constraints and aspects to consider when the requirements for the design of the AESDP model are determined.

In the ***second*** part of this chapter, the requirements for a development process model are then described in terms of:

1. Process context;
2. Process organisation and structure;
3. System requirements management;
4. Requirements implementation;
5. Verification and validation;
6. Prototyping;
7. Process efficiency measurement.

Note that the terms project and program are used interchangeably in the context to follow.

Also, where applicable, process requirements derived from an associated context are presented as uniquely identified requirement statements - these requirements are clearly stated and shown in underlined and bold text. All these requirements shall then be addressed in the following chapter.

3.2 Process constraints

3.2.1 Characteristics of airborne electronic systems

Before determining a definition of a development process model for airborne equipment, the general characteristics of such equipment must be understood. The characteristics listed below are not exhaustive and is drawn from personal experience by the author.

3.2.1.1 Life cycle properties

Characteristic 1: Incremental development: Airborne systems such as autopilots and mission computers are typically released into service in “blocks”, where functionality is added or changed

incrementally during the life cycle of the aircraft, e.g. to accommodate a new weapon or integrate to improved navigation equipment. This agrees well with the principles of incremental and iterative development (IID) [32]. Due to budgetary constraints, changes in operational requirements or the development of associated systems, the system may undergo scheduled updates or modifications to in-service baselines. This requires a long term stable support infrastructure, based on a well established development process model.

Requirement 1.01: *The process model shall support an incremental development process, where a given increment forms a complete end product.*

Characteristic 2: High level requirements are normally well defined: In many instances, the system under development represents an improvement of a similar system, using improved technologies, or refinement of an existing system. The applied areas of expertise, e.g. vehicle navigation, flight control or ballistics, utilise well known principles and rules and the system dynamic behaviour models are known in general terms. This means that a significant number of high-level requirements can be established early in the development program. This will lead to the definition of lower-level requirements as derived from fixed high-level requirements.

Requirement 1.02: *The requirements capture process should be initiated in the initial phases of the project.*

Requirement 1.03: *The process model shall be structured such that the development of the end product and enabling products take place in parallel with prototyping activities.*

Characteristic 3: Established technologies are implemented: Due to the risks associated with airworthiness certification of systems, technologies are generally employed in airborne applications once they are well proven, especially with regards to safety. This implies that design constraints can also be determined early in the development program.

Requirement 1.04: *The process model shall be based on the premise that only established technologies shall be employed in the product realisation process.*

Note: The above requirements implies that, if it is identified that technology needs to be developed on a project in order to meet with the stakeholder requirements, the development process described in this dissertation is not valid, or needs to be tailored to specifically make provision for this additional development.

Characteristic 4: Extended development life cycles: The development of a system takes place over years rather than months. This requires a stable development environment and associated

development process to allow for personnel turnaround. Thus, not only from a business continuity perspective, it is important to ensure project continuity by means of a controlled environment.

Requirement 1.05: *In order for new members of the development team to perform their allocated tasks with a minimum of coaching, processes and methods that are well established in the industry shall be implemented.*

Characteristic 5: Service life varies between 15 and 30 years: Contemporary military aircraft are designed with minimum expected service in the excess of 30 years, some aircraft types, e.g. Boeing B52 and Lockheed C130 have already been in service for longer than 40 years where the retirement of these types is still not planned. Due to advances in applicable technologies as well as obsolescence due to changes in the technology, airborne electronic systems are typically updated more than once during the lifecycle of an aircraft type.

Requirement 1.06: *The process model shall make provision for the handling of legacy or precedented systems.*

Characteristic 6: Serviceable items are subjected to strict storage and handling rules: In order to prevent non airworthy items and software from inadvertently being installed onto serviceable aircraft, specific practises are mandatory throughout the aviation industry. These logistic planning and processes are to be integrated into the development process.

Requirement 1.07: *The process model shall show interfaces to the logistics processes*.

Characteristic 7: Economy of scale: Production runs for the type of systems under consideration are typically around 10 to 100 units, and seldom exceed 1000 units. This fact has a direct bearing on the manufacturing infrastructure that will be employed, which in turn is linked to the development infrastructure and processes.

Requirement 1.08: *The process model shall indicate interfaces with the manufacturing and product support processes.*

Note: Due to the economy of scale, development personnel typically perform some product support functions, using development environment infrastructure.

Characteristic 8: Interrelationships: The systems form part of a larger integrated physical system which forms part of the air vehicle. The interaction between the development of a subsystem and that of the larger system needs to be understood and managed.

Requirement 1.09: *The process model shall indicate interfaces with the air vehicle layer in the product hierarchy*.

Characteristic 9: Security: Access to information pertaining to the system and distribution of life cycle data is normally restricted, especially concerning military applications which impose specific constraints on the developer.

Requirement 1.10: *The process model shall show how project data security is ensured.*

Characteristic 10: Version control and configuration management: For some systems that were in use for a substantial period or supplied to multiple clients, it may happen that more than one version may be in service at a given time, placing additional requirements on version control procedures and support infrastructure.

Requirement 1.11: *The configuration management system employed by the enterprise shall have the ability to manage different operationally deployed versions of a particular system.*

3.2.1.2 System safety aspects

Characteristic 11: Safety and reliability: The consequences of hardware failures or erroneous behaviour of software are significant in terms of safety and mission criticality. These considerations normally play a more important role than lifecycle costs in deciding implementation technologies. The development processes must clearly indicate how the outcomes of system safety processes lead to safe system designs and system usage instructions. Reliability analyses form an essential part of the system safety process and has a direct bearing on the system architecture.

Requirement 1.12: *The handling of system safety and reliability aspects shall be detailed in the process model.*

Characteristic 12: Regulatory requirements: The system is subject to certification by a statutory body, i.e. compliance to airworthiness standards and requirements need to be demonstrated. Liability considerations also need to be considered. In order to meet with airworthiness requirements, evidence of verification and validation activities must also be presented to the certification authorities. This means that the process needs to indicate how verification evidence is to be obtained, recorded and made available to the authorities. Specific regulatory standards include RTCA/DO-178B [10] for software certification, FAR part 21 [17] for certification of aircraft sub-systems, SAE ARP 4754 [18] for certification of complex systems, RTCA/DO-160 [19] for the environmental qualification of airborne systems and FAR 25 [20] and FAR 29 [21] for the certification of transport category aircraft and rotorcraft respectively.

Requirement 1.13: *The process model shall identify the activities required to support the system certification process.*

3.2.1.3 Technology considerations

Characteristic 13: Time-critical embedded software: In most applications the system contains embedded software which processes data in near real time. That is, data from sensors is passed through algorithms where the outputs are used within milliseconds e.g. to yield flight control commands, graphical information display images or navigation solutions. Time delays can normally not be tolerated. This constraint has a bearing on the selection of hardware and software architectures.

Requirement 1.14: *The process model shall allow for mechanisms to facilitate the selection of appropriate hardware and software architectures, early in the development life cycle.*

Characteristic 14: Particular data transfer protocols: Specific data transfer protocols and interfaces are defined for airborne use, e.g. ARINC 429 [14] and MIL-STD-1553B [15], requiring specialised knowledge and development tools. (Note: Although these two standards are still dominant in the industry, the use of Ethernet, USB, IEEE 422 [16] and other well known data transfer protocols are introduced more frequently in modern systems.)

Requirement 1.15: *The development process model shall identify infrastructure requirements that will allow the support of test and integration using dedicated avionics data interface protocols.*

Characteristic 15: Computing platforms: The rapid development of microprocessor technology means that almost as soon as a system is fully qualified, the processors on which the design is based are almost obsolete. Due to the relatively small volumes and long service life of avionics and airborne weapons computers, this poses a specific predicament to developers. The trends of the electronics industry should be understood by developers when making design decisions.

Requirement 1.16: *The process model shall address obsolescence management.*

Characteristic 16: Navigation infrastructure: A worldwide infrastructure for the support of navigational operations, e.g. GPS and COSPAS/SARSAT has evolved, and is continually improved. Systems should be developed such that updates due to development of this infrastructure can be incorporated with minimum effort.

Characteristic 17: Hardware: Mechanical gyroscopic devices have by and large been replaced by solid state devices and sophisticated signal processing. Displays have evolved from

electromechanical devices to high resolution liquid crystal colour display panels. Different engineering techniques are required when implementing these into systems. These techniques are to be identified and mastered by the enterprise.

Characteristic 18: Software tools: Computer aided system and software development relies on a considerable number of tools that represent a significant investment in terms of capital and training on the part of the enterprise. Decisions on the types of tools to acquire and implement need to be taken prudently.

Requirement 1.17: *The process model shall indicate requirements for development infrastructure establishment*.

3.2.2 Airworthiness

3.2.2.1 Certification

In South Africa, final airworthiness approval for a system intended for use in a military aircraft is granted by the Military Airworthiness Board (MAB), under auspices of the Directorate for System Integrity (DSI) of the South African Air Force (SAAF). Airworthiness approval is required before a system can be released for service. In order to obtain this approval, the contractor has to agree on the certification requirements with the MAB, and provide them with the evidence that the requirements were met. A mechanism is required for the management of this process, by the contractor.

For all contracts involving military aircraft, the SAAF issues a Users Requirement Statement (URS) to the military contracting agency (Armscor) who, after a normal commercial process enters in an agreement with the contractor. The contracting agency ensures that the terms of the contract are met, requiring formalised interfaces with the contractor on the systems engineering level.

The development process model should take cognisance of the processes used by the SAAF and Armscor and ensure that adequate provision is made for the interfaces mentioned above, and be structured such that changes in the processes at the SAAF and Armscor can be accommodated.

Requirement 2.01: *The process model shall indicate the detail of the responsibilities of the enterprise with respect to the certification process.*

3.2.2.2 Continued airworthiness

When a system is employed on an aircraft meeting with airworthiness requirements, it is a prerequisite that sufficient resources exist within the supplying organisation to investigate any operational or safety problem that may arise as a consequence of the operational use of the system. This means that the developer of a product for airborne use should ensure that an adequate infrastructure is established during system development to meet with this requirement, and that product knowledge gained during the development process is adequately disseminated so that it will be available when required, throughout the life cycle of the product.

Requirement 2.02: *The process model shall indicate the mechanisms by which continued airworthiness shall be ensured.*

3.3 Process context, architecture and components

3.3.1 Development process context

In order to address the Denel need, a development process needs to be derived and defined in terms of the context in which it is to operate - this is done to segregate the activities required for system realisation. As a result, all activities *directly related* to the *development* processes need to be distinguished from activities related to *support* or *management* functions. It is also required to determine interactions (interfaces) with other processes within the overall aircraft development process, the organisation, and wider environment.

The process context definitions and logical groupings of related activities to form sub-processes, as well as major activities constituting these sub-processes, presented in selected referenced standards, are summarised below. In most cases a listing of requirements is provided since the detailed contents are simply too overwhelming – and, outside the scope of this work - to include in this thesis.

Requirement 3.01: The process model shall indicate the organisational context of the development process, referring to the referenced standards.

3.3.2 Iterative and incremental development

As already stated in section 2.8, for large programs, the development project should be structured to allow iterative and incremental development. The relationship between the requirements, design and verification life cycle data for a particular development increment can be presented as shown schematically in Figure 6.

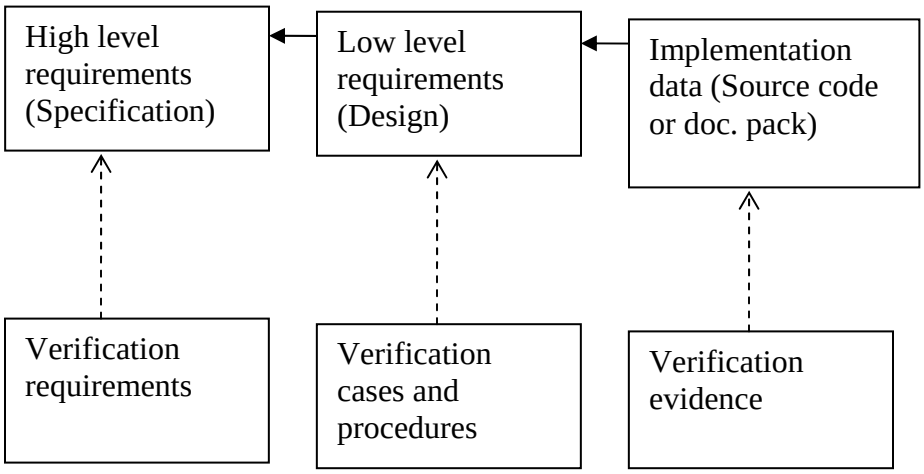

Figure 6: Product life cycle data for a particular development increment

Note that *these elements, together with the definition of the associated development and verification environment completely specify the end product for the development increment*. On the software layer, for example, these elements can be understood to be the Software Specification, together with a description of the method by which each requirement shall be verified; the Software Design (document), together with the suite of test cases by which the design can be tested; and the source code and associated unit (module) tests. These notions can readily be extended to the system and hardware layers.

With reference to Figure 6: solid arrows indicate that traceability is to be demonstrated, dotted line arrows indicate coverage is to be demonstrated.

As an example of IID, an autopilot could be developed incrementally in the following manner:

1. Iteration 1: Develop abstraction layer – that is, the layer between the actual sensors, actuators etc and the operator. This incremental development iteration is subjected to all the steps in Figure 6;
2. Iteration 2: Develop basic flight capability – that is, implement the functionality that provides stability augmentation, and other primary modes that allow safe flight. This iteration is also subjected to all steps in Figure 6 and follows sequentially on iteration 1;
3. Iteration 3: Add upper modes – this is where coupled modes, e.g. hover hold and navigation modes are implemented.

See requirement 1.01 in section 3.2.1.1.

3.3.3 Development stages

Decision gates are required to facilitate adequate control of the development process. A number of different development stages are described in the references. A definition of development stages that suits the particular requirements of developers of airborne electronic equipment is to be developed, where activities from different sub-processes are synchronised to produce the outcomes required at each decision gate or development stage formal closure.

3.3.4 Development layers

An item such as an autopilot or navigation computer forms part of the air vehicle system, and is finally qualified as such. In the development process, it will be handled by different teams and verification will take place in the software, system, integrated system (tested on a test rig) and eventually aircraft environments. In order to structure the management of the requirements, traceability and test coverage, it is a requirement to define the boundaries where hand-over of data and responsibilities occur.

Requirement 3.02: *The process model shall provide guidance for the division of the development process into layers of hierarchy*.

Requirement 3.03: *The process model shall identify a set of main project stages (decision gates) for a development increment*.

Requirement 3.04: *The process model shall define sub-processes that constitute the development process, using guidance from the referenced standards.*

Requirement 3.05: *For each development stage and sub-process, the major activities constituting these shall be identified*.

3.4 Process main activities

The requirements specified below were developed as a result of co-operation with Saab AB on the definition of a formalised development process, from personal experience gained during development of airborne electronic equipment within Denel Aviation, and from the literature cited in section 2.11.

3.4.1 Development planning

3.4.1.1 Development planning objectives

The development process should be managed by means of clearly defined plans that outline what has to be done, what procedures will be used (describing how things should be done), what resources are required, and against which standards outcomes will be measured.

The planning process should not end at the end of the initial planning phase. Plans should be updated regularly as the program progresses, to reflect changes that may be required due to e.g. improvements of the development environment or updates to relevant standards, or to capture process improvement recommendations. During quality assurance audits, it shall be verified that work on the program is executed in accordance with the plans and associated procedures.

Requirement 4.01: *The process description shall indicate which generic development and management plans and standards are required for a typical project.*

3.4.1.2 Development planning references

AS9100 [4] clause 7.1 – Planning of Product Realisation and 7.3.1 Design and Development Planning, identify the activities against which the enterprise will be audited in order to accredited to perform work in the aerospace sector.

EIA-632 [6] clause 4.2.1 – Planning Process, requirements 4, 5 and 7, Table C.4, C.5 and C.7 and planning documents in Annex D2 deal particularly with the management of the technical effort, the other requirements in this clause handle about aspects that should be described in a project management process model.

ISO/IEC 12207 [7] clause 5.2.4 – Planning, presents guidelines for the planning of software development activities and clause 5.3.1 – Process Implementation, describes additional aspects related to the planning of the software development effort.

IEEE 1220 [8] clause 4.2 – Policies and procedures for systems engineering and 4.3 – Planning the technical effort describe aspects related to planning of the technical effort and the systems engineering management plan is described in Annex B.

ISO/IEC 15288 [9] clause 5.4.2 – Project Planning Process, describes both project management and technical management activities.

RTCA/DO-178B [10] clause 4 – Software Planning Process describes requirements for the planning of software development where the software will be used in airborne applications, and

clause 11 – Software life cycle data, sections 11.1 to 11.8 defines the associated required planning documents.

Blanchard and Fabrycky [11] chapter 18 describes the planning and organisation of a systems engineering endeavour.

INCOSE handbook [12] chapter 5.2 describes project planning in a broader perspective than the other references mentioned here.

Of particular interest to the systems engineering effort is the systems engineering management plan (SEMP) which is discussed in Blanchard and Fabrycky chapter 18 and IEEE 1220 Annex B. Currently Denel Aviation uses the SEMP pro forma as defined in RSA-MIL-STD-182 [19]. This pro forma is to be updated to be in line with the international approach.

SAE ARP4754 [18] clause 4.1 and RTCA/DO-178B clause 4.3 address airworthiness certification planning in particular.

3.4.2 Development process control

3.4.2.1 Development process objectives

The technical effort needs to be monitored and directed to ensure that the project objectives will be achieved within the project constraints. In the context of the AESDP, process control includes the management of the technical effort, assessing technical progress against requirements, controlling of life cycle data and information, management of the system configuration, tracking and management of problems, and the identification and mitigation of risks. Methods may include technical workgroups (e.g. configuration control boards), formal reviews and project (technical) audits.

The development process interfaces with the other project and enterprise processes at this level.

In the course of execution of the project, aspects may be noticed where process features may have impacted negatively on project objectives, requiring modification or improvement of the process model. This information should be made available to the quality assurance processes.

Many examples are mentioned in subject literature where project costs overran budgets by considerable margins. This is due to the difficulty of estimating the required effort, even in cases where the enterprise has executed similar projects before.

A need is identified, to be able to measure the efficiency at which the outcomes of development activities can be measured, to support the management of the technical effort and to corroborate task size estimates.

Note related to risk management: Virtually all aspects of the definition of a development process contribute to risk abatement. The better every element of the process is defined and understood, the better the associated risks can be quantified and controlled or mitigated. Every project should however have its own risk management plan, in which these are addressed explicitly.

Requirement 4.02: *Reviews of the technical effort, at the different project stages, shall be defined in the process model.*

Requirement 4.03: *The activities required to control life cycle data and project information, shall be indicated in the process model.*

Requirement 4.04: *The activities required to achieve effective configuration management, shall be indicated in the process model.*

Requirement 4.05: *The activities required to identify, manage and track problems, shall be indicated in the process model.*

Requirement 4.06: *Interfaces (e.g. risks identified, work scope identified, resource requirements identified, etc.) with other project and enterprise processes need to be identified and documented in the process model.*

Requirement 4.07: *The process model shall indicate how process improvement feedback is provided to the quality assurance processes.*

Requirement 4.08: *The process model shall indicate process metrics to facilitate measurement of progress of the technical effort.*

3.4.2.2 Development process control references

AS9100 [4] clause 7.3.4 describes review activities, to be performed at various stages of development. Control of design and development changes is addressed in clause 7.3.7 and configuration control (identification and traceability) is described in clause 7.5.3.

The technical assessment process is described in EIA-632 [6] requirements 9, 10 and 11 and Table C.9, C.10 and C.11. The control process is described in requirements 12 and 13 and Table C.12 and C.13.Technical reviews are described in Annex E.

ISO/IEC 12207 [7] clause 5.2.5 describes project execution and control, clause 5.2.6 defines review and evaluation activities, 5.2.7 defines delivery and completion tasks, clause 6.2 describes the configuration management process, clause 6.6 describes joint review processes, clause 6.7 audit processes, clause 6.8 the problem resolution process, and clause 7.1 describes the management processes. The (software) life cycle improvement process is described in clause 7.3.

IEEE 1220 [8] clause 6.8 deals with control under the following headings: 6.8.1 Technical management; 6.8.1.1 Data management; 6.8.1.2 Configuration management; 6.8.1.3 Interface management; 6.8.1.4 Risk management; 6.8.1.5 Performance-based progress measurement; 6.8.2 Track systems analysis and test data; 6.8.3 Track requirement and design data; 6.8.4 Track progress against project plans; 6.8.5 Track progress against engineering plans; 6.8.6 Track product and process metrics; 6.8.7 Update specifications and configuration baselines; 6.8.8 Update requirements views and architectures; 6.8.9 Update engineering plans; 6.8.10 Update technical plans.

ISO/IEC 15288 [9] clause 5.3.4 defines the system life cycle processes management process. This standard should also be used to identify interfaces to other project and enterprise processes.

SAE ARP4754 [18] clause 9 describes the configuration management process and clause 10 describes the process assurance process.

Blanchard/Fabrycky [11] describes the program management, control and evaluation process in chapter 19.

The INCOSE handbook [12] describes the project assessment process in chapter 5.3, the process control process in chapter 5.4, the decision-making process in chapter 5.5, risk and opportunity management in chapter 5.6, configuration management in chapter 5.7 and information management in chapter 5.8.

Use of MIL-STD-1521B [23] is firmly entrenched in the South African defence industry, for the use of conducting formal reviews. Cognisance should be taken of this standard when defining review processes in the process model.

Life cycle stages and baselines are defined in RSA-MIL-STD 3 [24]. Current contracts with ARMSCOR invoke the use of this standard for baseline referencing.

3.4.3 Requirements driven development

3.4.3.1 Requirements process objectives

The requirements process entails the gathering and refinement of requirements, the approval by key stakeholders, the management of traceability between layers of requirements and of verification and validation of the requirements. The engineering baselines of a system under development are linked to the requirements management process.

The requirements statements also form the primary basis against which the system shall be qualified.

Note that the requirements process extends throughout the life cycle of the product. Requirements may be added or modified. This implies that a formal change process should be used to manage requirements databases.

In addition, handling of interfaces between requirements management processes on aircraft, product and software layers, and interfaces between requirements processes and other processes, such as the system safety process need to be addressed, as these are peculiar to the systems discussed in this dissertation.

A requirement typically has the following life cycle: Requirement identified, requirement validated, requirement verification requirements specified, requirement verification method specified, requirement implemented, requirement verified, requirement closed.

Please note that the above is the author's interpretation, based on EIA-632 [6] and ISO 15288 [9].

Kindly also take note that it will in some cases only be possible to perform validation after operational tests since the actual required operational performance only becomes apparent after operational testing.

From the above, it is evident that the entire development process pivots around a set of well defined and controlled requirements.

Requirement 4.09: *The process model shall indicate the major activities and their expected outcomes, for the requirements management process*.

Requirement 4.10: *The process model shall indicate how requirements traceability to requirements originating at higher layers of system hierarchy shall be achieved*.

Requirement 4.11: *The process model shall indicate how traceability to lower level requirements (design or requirements implementation strategies) shall be achieved.*

Requirement 4.12: *The process model shall indicate how the status of a requirement shall be tracked.*

Requirement 4.13: *The process model shall indicate how requirements shall be associated with requirements verification requirements.*

Requirement 4.14: *The process model shall indicate how every requirement shall be linked with its associated verification evidence.*

Requirement 4.15: *The process model shall indicate how every requirement shall be linked with its associated validation evidence.*

3.4.3.2 Requirements process references

SAE AS9100 [4] clause 7.2.1 describes the process of determining requirements related to the product.

EIA-632 [6] requirement 14 states: The developer shall define a validated set of acquirer requirements for the system, or portion thereof; requirement 15: The developer shall define a validated set of other stakeholder requirements for the system, or portion thereof; and requirement 16: The developer shall define a validated set of technical requirements. Further detail is specified in tables C.14, C.15 and C.16. Requirements 25 to 29 and table C.25 to C.29 deals with requirements validation.

ISO/IEC 12207 [7] clause 5.3.2 defines the system requirements analysis activity and clause 5.3.4 the software requirements analysis activity.

IEEE 1220 [8] clause 6 describes the requirements process under the following headings: 6.1 – Requirements analysis; 6.1.1 – Define stakeholder expectations; 6.1.2 – Define project and enterprise constraints; 6.1.3 – Define external constraints; 6.1.4 – Define operational scenarios; 6.1.5 – Define measures of effectiveness; 6.1.6 – Define system boundaries; 6.1.7 – Define interfaces; 6.1.8 – Define utilisation environments; 6.1.9 – Define life cycle process concepts (Under this heading the following are described: manpower, personnel, training, human engineering and safety); 6.1.10 – Define functional requirements; 6.1.11 – Define performance requirements; 6.1.12 – Define modes of operation; 6.1.13 – Define technical performance measures; 6.1.14 – Define design characteristics; 6.1.15 – Define human factors; 6.1.16 – Establish requirements baseline; 6.2 – Requirements validation; 6.2.1 – Compare to stakeholder expectations; 6.2.2 – Compare to enterprise and project constraints; 6.2.3 – Compare to external constraints; 6.2.4 – Identify variances and conflicts; 6.2.5 – Establish validated requirements

baseline; 6.3 – Functional analysis; 6.3.1 – Functional context analysis; 6.3.1.1 – Analyse functional behaviours; 6.3.1.2 – Define functional interfaces; 6.3.1.3 – Allocate performance requirements; 6.3.2 – Functional decomposition; 6.3.2.1 – Define sub-functions; 6.3.2.2 – Define sub-function states and modes; 6.2.3.3 – Define functional time line; 6.3.2.4 – Define data and control flows; 6.3.2.5 – Define functional failure modes and effects; 6.3.2.6 – Define safety-monitoring functions; 6.3.3 – Establish functional architecture.

ISO/IEC 15288 [9] clause 5.5.2 defines a stakeholder requirements definition process and clause 5.5.3 the requirements analysis process. Clause 5.5.9 describes the validation process.

RTCA/DO-178B [10] clause 5.1 describes the software requirements process, clause 5.5 provides traceability guidance.

Blanchard/Fabrycky [11] describes a requirements process in chapter 3 under the heading "Conceptual System Design".

INCOSE handbook [12] chapter 7.2 discusses requirements management.

RTCA/DO-254 [33] explains the requirements capture process in section 5.1 and the validation of derived requirements allocated to hardware, in section 6.1.

SAE ARP4754 [18] clause 5 describes the requirements process under the following headings: 5.1 – Requirements Capture; 5.2 – Types of Requirements; 5.3 – Derived Requirements; 5.4 – Assignment of Development Assurance Levels; 5.5 – Failure Condition Risk Assessment. Clause 7 describes requirements validation under the headings: 7.1 – Validation Process Objectives; 7.2 – Validation Process Model; 7.3 – Completeness Checks; 7.4 – Correctness Checks; 7.5 – Validation of Assumptions; 7.6 – Validation Rigor; 7.7 – Validation Data.

3.4.4 System safety process and airworthiness certification

3.4.4.1 System safety and certification process objectives

Due to the interrelationships between system safety and airworthiness certification, the topics are discussed under a single heading.

The objectives of a system safety process are to identify possible hazards and the severity thereof, to institute adequate mitigation actions and to control these actions.

SAE ARP4761 [26] describes a system safety process that starts with a Functional Hazard Assessment (FHA), where all the failures associated with all defined functions for a project are assessed and the severity of the consequences of the failures are classed as minor, major,

hazardous and catastrophic. Outputs from this process are used in the architectural design process, to e.g. provide additional redundancy to reduce the probability of a failure occurring. When the preliminary design is completed, a Preliminary System Safety Assessment (PSSA) is conducted, to confirm that the design adequately mitigates the identified hazards. The PSSA typically consists of the performing of a Failure Mode, Effects and Criticality Analysis (FMECA) and associated tasks and reviews. For a particular development increment, the system safety process concludes with a System Safety Assessment (SSA), where verification results are used to substantiate the design integrity. Note that these three outcomes remain active throughout the life cycle of the system, to handle newly identified hazards in a formal manner. These outcomes form the basis of the certification data pack. This process is also summarised in SAE ARP4754 [18] clause 6.

The safety aspects of airborne software are dealt with extensively in RTCA/DO-178B [10] and are summarised below.

Safety Aspects: For a project, a safety assessment process is conducted on system level. It determines the criticality of failures of system functions and components. The severity of a failure for hardware is mitigated by designing the components to minimise the probability of failure in accordance with the failure severity, and during qualification testing evidence is collected to substantiate the design. Software does not fail, it contains undetected errors. It cannot be designed to have a certain probability of failure. The potential for introduction of errors is consequently minimised by means of controls on the development process.

Anomalous behaviour of functions allocated to software are categorised by the safety assessment process in terms of its effect:

Level E: The consequence of a failure has no effect on the operational capability of the aircraft or increase in workload;

Level D: The consequence of a failure leads to a slight increase in pilot workload, and no significant reduction of aircraft safety;

Level C: The consequence of a failure leads to a significant increase in workload, and a significant reduction in safety margins or functional capabilities;

Level B: The consequence of a failure leads to a severe increase in workload, and a large reduction in safety margins or functional capabilities;

Level A: The consequence of a failure leads to a failure of system function that results in conditions that would prevent continued safe flight and landing.

The Standard prescribes the minimum requirements for the processes to produce software that will meet with the required level of safety.

The certification process deals with the aspects of liaison with airworthiness authorities, and with the provision of evidence that certification requirements, which are also dealt with in the mentioned standards, were met.

Requirement 4.16: *The process model shall indicate the necessary system safety activities and expected outcomes.*

Requirement 4.17: *The method by which hazards are identified and tracked shall be described in the process model.*

Requirement 4.18: *Criteria for assessing hazard severity shall be captured in the process model.*

Requirement 4.19: *Methods for assessing system reliability shall be identified.*

Requirement 4.20: *The process model shall indicate the activities and expected outcomes to support the certification process.*

Requirement 4.21: *The process model shall indicate how continued airworthiness of a system shall be ensured.*

3.4.4.2 System safety and certification process references

SAE ARP4761 [26] deals with the system safety processes for aircraft and aircraft systems.

RTCA/DO-178B [10] clause 2.2 describes the relationship between the software development processes and the system safety processes.

RTCA/DO-254 [33] section 2.2 – System Safety Assessment Process and section 2.3 – Hardware Safety Assessment Considerations describe the systems safety processes particularly applicable to airborne electronic hardware, including complex hardware.

Blanchard/Fabrycky [11] section 12.4 deals with reliability analysis and section 14.4.4 describes safety and hazard analysis.

INCOSE handbook [12] chapter 9 – Speciality Engineering Activities, section 9.7 discusses safety and health analyses.

SAE ARP4754 [18] clause 3.2 refers to a process called Development Assurance, which "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 particular certification requirements." This standard identifies the following development assurance activities: Certification Co-ordination; Safety Assessment; Requirements Validation; Implementation Verification; Configuration Management and Process Assurance, and it identifies the requirement for Development Assurance Substantiation.

FAR/JAR 25.1309 [20] states that "airplane systems and associated equipment, considered separately and in relation to other systems, must be designed so that –

(1) The occurrence of any failure condition which would prevent the continued safe flight and landing of the airplane is extremely improbable, and

(2) The occurrence of any other failure conditions which would reduce the capability of the airplane or the ability of the crew to cope with adverse operating conditions is improbable.

The standard also prescribes other safety requirements pertaining to aircraft systems.

3.4.5 End product realisation

3.4.5.1 End product realisation objectives

Hardware and software are developed to produce systems that perform the functionality defined by the requirements process. This process typically starts with the definition of hardware and software architectures, interfaces, sub-systems and enabling system requirements, and is followed by the detailed design, implementation and integration activities.

The design process also involves the integration of outcomes from speciality engineering activities, e.g. electromagnetic compatibility analysis and human factors engineering.

During the design process, requirements for sub-systems and enabling products become more evident and clear, and these should be captured. Note that it can be viewed that supporting documentation forms a specific form of enabling product.

Specifications for items to be procured are also established as part of the design process.

Outputs from the design processes need to be captured for use by other stakeholders for configuration management purposes. The formats of this life cycle data (e.g. documentation templates and guidelines) need to be defined.

It is imperative to demonstrate traceability between designs and requirements, and, in the case of software, between the source code and the software design.

For airborne electronic equipment, the requirements on change and configuration management are exacting, necessitating stringent process control. Personal experience from involvement in a number of projects has indicated that the management of the integration process can be particularly daunting, as integration normally involves a number of changes resulting from oversights in preceding processes, however well these may have been performed. Due to the strict configuration management policies, design changes due to integration problems can be difficult to accomplish. The process model should take cognisance of this reality. See also section 3.4.6.1 (informal testing).

Requirement 4.22: *The process model shall indicate a comprehensive set of end product realisation activities and expected outcomes.*

Requirement 4.23: *The process model shall indicate methods for the integration of engineering specialities.*

Requirement 4.24: *The process model shall indicate the points in the project workflow where sub-system development requirements shall be captured.*

Requirement 4.25: *The process model shall indicate the points in the project workflow where enabling product requirements shall be captured.*

Requirement 4.26: *The process model shall indicate the points in the project workflow where supporting documentation requirements shall be captured.*

Requirement 4.27: *The process model shall indicate the points in the project workflow where sub-system procurement requirements shall be captured.*

Requirement 4.28: *The process model shall provide guidelines for the preparation of end product realisation life cycle data.*

Requirement 4.29: *The process model shall indicate how traceability between requirements and design or requirements implementation strategies shall be achieved.*

Requirement 4.30: *The process model shall indicate how the integrity of the design life cycle data will be ensured.*

Requirement 4.31: *The process model shall make provision for methods to perform design changes during the integration process without affecting the integrity of the design.*

3.4.5.2 End product realisation references

AS9100 [4] clause 7.2.1 describes the process of determining requirements related to the product.

EIA-632 [6] clause 4.3.2 “Solution Definition Process” requirement 17 states “the developer shall define one or more validated sets of logical solutions representations that conform with the technical requirements of the system”; requirement 18 states “the developer shall define a preferred set of physical solution representations that agrees with the assigned logical solution representations, derived technical requirements, and system technical requirements; requirement 19 states “the developer shall specify requirements for the design solution.

ISO/IEC 12207 [7] clauses 5.3.3 and 5.3.5 describes the system and software architectural design processes respectively, clause 5.3.6. describes the software detailed design process, clause 5.3.7 describes software coding and testing, clause 5.3.8 describes software integration.

IEEE 1220 [8] clause 6.5 describes the synthesis process under the following headings: 6.5.1 – Group and allocate functions; 6.5.2 – Identify design solution alternatives; 6.5.3 – Assess safety and environment hazards; 6.5.4 – Assess life cycle quality factors; 6.5.5 – Assess technology requirements; 6.5.6 – Define design and performance characteristics; 6.5.7 – Define physical interfaces; 6.5.8 – Identify standardisation opportunities; 6.5.9 – Identify off-the-shelf availability; 6.5.10 – Identify make-or-buy alternatives; 6.5.11 – Develop models and prototypes; 6.5.12 – Assess failure modes, effects and criticality; 6.5.13 – Assess testability needs; 6.5.14 – Assess design capacity to evolve; 6.5.15 – Finalise design; 6.5.16 – Initiate evolutionary development; 6.5.17 – Produce integrated data package; 6.5.18 – Establish design architecture. Clause 6.7 – describes the systems analysis process under the following headings: 6.7.1 – Assess requirements conflicts; 6.7.2 – Assess functional alternatives; 6.7.3 – Assess design alternatives; 6.7.4 – Identify risk factors; 6.7.5 – Define trade-off analysis scope; 6.7.5.1 – Select methodology and success criteria; 6.7.5.2 – Identify alternatives; 6.7.5.3 - Establish trade-study environment; 6.7.6 – Conduct trade-off analysis; 6.7.6.1 – Analyse life cycle costs; 6.7.6.2 – Analyse system and cost effectiveness; 6.7.6.3 – Analyse safety and environmental impacts; 6.7.6.4 – Quantify risk factors; 6.7.7 – Select risk-handling options; 6.7.8 – Select alternative recommendation; 6.7.9 – Trade-offs and impacts; 6.7.10 – Design effective assessment.

ISO/IEC 15288 [9] clause 5.5.4 – Architectural Design Process; clause 5.5.5 – Implementation Process and clause 5.5.6 – Integration Process applies to system realisation.

RTCA/DO-178B [10] clause 5.2 – Software Design Process; clause 5.3 – Software Coding Process and clause 5.4 – Integration Process, describe the software realisation process.

RTCA/DO-254 [33] sections 5.2 – Conceptual Design Process; 5.3 – Detailed Design Process; 5.4 – Implementation Process and 5.5 – Production Transition Process, describe the hardware realisation processes.

Blanchard and Fabrycky [11] chapters 3, 4 and 5 relate to realisation aspects.

INCOSE handbook [12] chapter 8.2 – Architectural Design and chapter 9 – Speciality Engineering Activities, are relevant.

3.4.6 Verification

3.4.6.1 Verification process objectives

In the context of this study, the terms verification and validation will be interpreted as defined in RTCA/DO-178B [7]: Verification shall imply "the evaluation of the results of a process to ensure correctness and consistency with respect to the inputs and standards provided to that process" and validation shall imply "the process of determining that the requirements are the correct requirements and that they are complete". This can be stated in simpler terms: the verification process confirms that requirements were implemented correctly; the validation process confirms that the requirements were correct.

Verification of correct statement of requirements and validation of requirements are discussed in section 3.4.3 of this dissertation. This section deals with the verification that requirements were implemented correctly.

SAE ARP4754 [18] clause 8.1 summarises the verification process as follows:

"The verification process:

a. Confirms that the intended functions have been correctly implemented.

b. (Confirms that) the requirements have been satisfied.

c. Ensures that the safety analysis remains valid for the system as implemented."

According to RTCA/DO-178B [7], verification can be achieved by three methods, i.e. reviews, analyses and tests. This standard also provides guidelines with respect to the aspects to be subjected to reviews and analyses.

Reviews involve examination of life cycle data to obtain a peer opinion. Note that these reviews are different from the project management reviews. These technical reviews are more commonly referred to as formal inspections, as described by Fagan [27].

Analyses provide evidence of correctness that can be repeated by other investigators using the same methods. Note that analyses include traceability and coverage analyses.

Tests are used to satisfy two objectives, i.e. demonstrate that the system meets its requirements, and demonstrate that errors that may lead to an unsafe situation were isolated and removed. For airborne systems, three levels of tests are identified, i.e. laboratory tests, ground tests (aircraft engines not operating) and flight tests.

The following elements of the verification planning process are identified:

i) Requirements implementation verification planning: The means to show compliance with a specific requirement after implementation of the requirement are determined. This part of the verification process should take place during the requirements analysis phase, where it should be ensured that every requirement can be verified, the verification requirements are identified and that a specific method for verification of the requirement can be identified.

ii) Review planning: Requirements where reviews (inspections) were determined to be sufficient as a means of compliance are to be identified and appropriate review methods are to be specified.

iii) Analyses planning: For the requirements where analyses are required as a means of compliance as well as other analyses requirements, e.g. coverage and traceability analyses, analysis techniques and formats of analysis outcomes need to be determined.

iv) Test case planning: The detailed tests are to be specified for those requirements, where tests were identified as the means of compliance.

It is not always obvious where the boundaries between the system, software and hardware levels are. This compounds the definition of the hand-over interfaces between the levels during formal qualification tests, where qualification may be required at lower layers prior to integration into higher layer assemblies.

RTCA/DO-178B [7], and other standards, emphasise that some verification efforts need to be performed with independence. The accepted practise is that the test procedure is executed by a person or persons other than those who wrote the test procedure or participated in the development of the system.

It is required that the system under test need to be subjected to the specified tests, prior to freezing of the baseline, to ensure that the system will pass the test when it is performed by an

independent party, and that the test documentation is correct. The process needs to accommodate these “informal” and “formal” tests. Note that the results of informal tests can not be used as certification evidence.

It is normally expected from the development team to produce test specifications and procedures for testing production units. These tests should prove conformance of the production units to the design, rather than demonstrate compliance with specifications.

In the military environment, it is common practise that the user will subject the delivered system to operational test and evaluation. Outcomes of these tests need to be fed back to the developer as further substantiating verification evidence or to correct problems, where identified. Note that these outcomes are also used in validating stakeholder requirements.

Requirement 4.32: *The process model shall indicate the required set of verification activities and expected outcomes.*

Requirement 4.33: *The process model shall indicate methods for specifying requirements implementation verification requirements.*

Requirement 4.34: *The process model shall indicate methods for reviewing and approving the implementation of a requirement where it was determined that it shall be verified by review only.*

Requirement 4.35: *The process model shall indicate methods for analysing and approving the implementation of a requirement where it was determined that it shall be verified by analysis.*

Requirement 4.36: *The process model shall indicate methods for testing and approving the implementation of a requirement where it was determined that it shall be verified by test.*

Requirement 4.37: *The process model shall present guidelines for the preparation and handling of verification evidence.*

Requirement 4.38: *The process model shall indicate hand-over interfaces between the system, software and hardware levels.*

Requirement 4.39: *The process model shall indicate how verification independence shall be achieved.*

Requirement 4.40: *The process model shall indicate points in the project workflow where informal testing is feasible.*

Requirement 4.41: *The process model shall present guidelines for the preparation of production acceptance tests.*

Requirement 4.42: *The process model shall indicate interfaces with the operational test and evaluation process.*

3.4.6.2 Verification process references

AS9100 [4] clause 7.2.1 describes the process of determining requirements related to the product.

EIA-632 [6] clause 4.5.3, requirements 30 to 32 and tables C.30 to 32 define the verification activities of a project.

ISO/IEC 12207 [7] clause 5.3.9 – Software qualification testing and clause 5.3.11 – System qualification testing describes test aspects. However, other verification activities are described throughout the standard.

IEEE 1220 [8] clause 6.4 describes functional verification under the following headings: 6.4.1 – Define verification procedures; 6.4.2 – Conduct verification evaluation; 6.4.2.1 – Verify architecture completeness; 6.4.2.2 – Verify functional and performance measures; 6.4.2.3 – Verify satisfaction of constraints; 6.4.3 – Identify variances and conflicts; 6.4.4 – Establish verified functional architecture. Clause 6.6 describes design verification under the headings: 6.6.1 – Select verification approach; 6.6.1.1 – Define inspection, analysis, demonstration, or test requirements; 6.6.1.2 – Define verification procedures; 6.6.1.3 – Establish verification environment; 6.6.2 – Conduct verification evaluation; 6.6.2.1 – Verify architecture completeness; 6.6.2.2 – Verify functional and performance measures; 6.6.2.3 – Verify satisfaction of constraints; 6.6.3 – Identify variances and conflicts; 6.6.4 – Verified design architecture; 6.6.5 – Verified design architectures of the life cycle process; 6.6.6 – Verified system architecture; 6.6.7 – Establish specifications and configuration baselines; 6.6.8 – Develop system breakdown structure (SBS).

ISO/IEC 15288 [9] clause 5.5.7 describes the verification process.

RTCA/DO-178B [10] clause 6 describes the software verification process.

Blanchard and Fabrycky [11] chapter 6 describes system test, evaluation and validation.

RTCA/DO-254 [33] explains the verification of the correct implementationof the requirements allocated to hardware.

INCOSE handbook [12] chapter 4.7 describes the verification process.

SAE ARP4754 [18] clause 8 describes the implementation verification process under the following headings: 8.1 – Verification Process Objectives; 8.2 – Verification Process Model; 8.3

– Verification Planning; 8.4 – Verification Methods, under the sub-headings 8.4.1 Inspection and Review; 8.4.2 Analysis, under the sub-headings 8.4.2.1 Modelling; 8.4.3.3 Coverage Analysis; 8.4.3 – Testing; 8.4.4 – Similarity/Service Experience; 8.4.5 – Recommended Verification Activities; 8.5 Verification Data, under the headings 8.5.1 – Verification Plan; 8.5.2 – Verification Procedures and Results; 8.5.3 Verification Matrix; 8.5.4 Verification Summary.

3.4.7 Infrastructure management

3.4.7.1 Infrastructure management objectives

Infrastructure requirements and configuration is addressed as a component of the development process model as the configuration of the infrastructure required to produce a particular baseline forms part of the definition of that baseline.

Aspects of the infrastructure to be addressed are the requirements management database, development tools, policies and procedures, verification tools, configuration management tools, etc.

Note: Resource requirements excluded in this model are time, personnel and funding. These shall be addressed by the project management process.

Requirement 4.43: *The process model shall indicate the required set of infrastructure management activities and expected outcomes.*

Requirement 4.44: *The process model shall describe the interfaces to the project management processes, indicating the development infrastructure requirements.*

Requirement 4.45: *The process model shall indicate methods for controlling the configuration of the development infrastructure, associated with given development increment.*

Requirement 4.46: *The process model shall indicate methods for controlling the configuration of the policies and procedures, associated with given development increment.*

3.4.7.2 Infrastructure management references

AS9100 [4] clause 6.3 describes the responsibilities of the enterprise with respect to development infrastructure.

ISO/IEC 12207 [7] clause 7.2 – Infrastructure process, describes process implementation, establishment, and maintenance of the infrastructure.

IEEE 1220 [8] clauses 4.6 and 6.8.1 describes an integrated repository for the capturing of all pertinent design data.

ISO/IEC 15288 [9] clause 5.3.2 – Enterprise Environment Management Process focuses on the applicable policies and procedures.

RTCA/DO-178B [10] clause 4.4 – Software Life Cycle Environment Planning describes the software development environment and software verification environment, and clause 11.15 presents requirements for a "Software Life Cycle Environment Index".

Blanchard and Fabrycky [11] chapter 5.4 Gives an overview of the use of design tools and aids.

INCOSE handbook [12] chapter 6 – Resource Management Process, provides guidance on the management of all project resources, i.e. materials, services, facilities and personnel.

3.4.8 Prototyping

3.4.8.1 Prototyping objectives

In the classical systems acquisition model, prototypes were developed to explore concepts and to demonstrate capability before full scale development commenced, as shown in Figure 7. This process typically utilised a typical waterfall development model, which extended over a significant period of time.

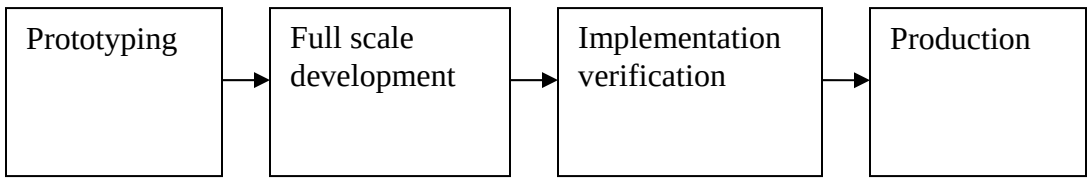

Figure 7: Traditional development process model

A problem that used to face developers of airborne electronic equipment was the fact that prototype equipment normally does not meet airworthiness requirements, necessitating special procedures to allow flight tests of the equipment. Some equipment, such as an autopilot, could only be set up adequately (adjustment of control system gains) when installed in a representative aircraft.

Considerable advances in modelling and simulation techniques, including graphic imaging, has lead to concepts of rapid prototyping, which enabled developers to produce concept demonstration equipment at comparatively low costs, early in the development cycle.

These prototypes should be used for the refinement of requirements and to test implementation trade-offs.

Given the fact (see 3.2.1.1) that requirements are normally at a state of maturity at early stages of the project, although breadboarding activities still form a significant part of the development undertaking, it should place in parallel to the process that produce the end product. This is represented in Figure 8.

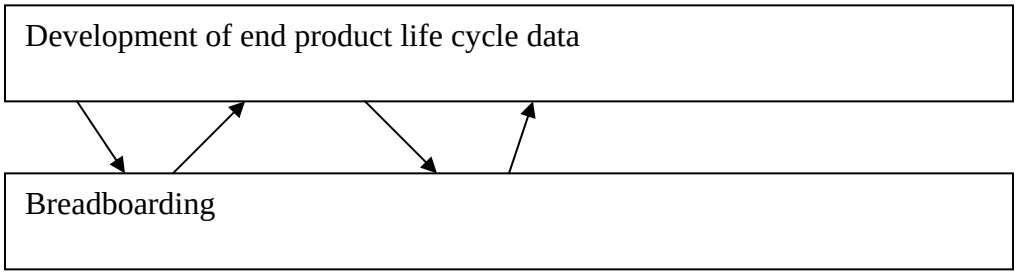

Figure 8: Use of breadboarding in proposed development process model

A particular aspect of this rapid prototyping model is that the embedded software developed should be discarded after use. This is equivalent to bread-boarding of an electronic circuit – after the requirements are adequately captured, the software should be developed from the requirements, using a development process that complies with RTCA/DO-178B [7].

Requirement 4.47: *The process model shall indicate the breadboarding process in the context of the development life cycle, for a generic project.*

Requirement 4.48: *The process model shall indicate interfaces between breadboarding activities and other life cycle processes.*

Requirement 4.49: *The process model shall present guidelines for the preparation and handling of breadboarding process outcomes.*

3.4.8.2 Prototyping references

INCOSE handbook [12] chapter 9.6 – Modelling, Simulation and Prototyping, describe these processes as speciality engineering activities. This presents a comprehensive discussion of the breadboarding principles, supporting modern systems engineering paradigms.

3.5 Selection of appropriate methods

There are numerous techniques available by which the outcome of each activity in the process can be achieved, in other words, the detailed "how" (which is the subject of further study). Due to the long life cycles associated with the systems described here, selecting an inappropriate technique or technology can have severe implications on the performance of the project. Examples of methods include techniques for requirements analyses (UML, safety assessments), programming languages, etc.

Requirement 5.01: *The process model shall provide a means where requirements for the establishment of specific methodologies are indicated to the quality management system.*

3.6 Quality management system assessment

The preceding sections of this chapter described requirements for a process model. The description of this model shall be captured and managed in the quality assurance management system of the enterprise.

The quality management system should be assessed against ISO/IEC15504-1 / 2 / 3 / 4: 2004: Information technology – process assessment [29]; SAE AS9101: Quality Management System Assessment [30] and the CMMI [3].

The process of maintaining the processes within the quality management system lies outside of the scope of this study since it is an external management function (that is, external to the development process itself). However, this will form part of further study.

3.7 Summary

In this chapter, most significant requirements were identified at a high level. These shall be used as input requirements to the definition of a process architecture in Chapter 4. A list of the requirements is provided in Table 1. An allocation matrix, allocating the higher level requirements to these requirements, is presented in Appendix A.

Table 1: Requirements listing

Number	Requirement statement
1.01	The process model shall support an incremental development process, where a given increment forms a complete end product.
1.02	The requirements capture process should be initiated in the initial phases of the project.
1.03	The process model shall be structured such that the development of the end product and enabling products take place in parallel with prototyping activities.
1.04	The process model shall be based on the premise that only established technologies shall be employed in the product realisation process.
1.05	In order for new members of the development team to perform their allocated tasks with a minimum of coaching, processes and methods that are well established in the industry shall be implemented.
1.06	The process model shall make provision for the handling of legacy or precedented systems.
1.07	The process model shall show interfaces to the logistics processes.
1.08	The process model shall indicate interfaces with the manufacturing and product support processes.
1.09	The process model shall indicate interfaces with the air vehicle layer in the product hierarchy.
1.10	The process model shall show how project data security is ensured.
1.11	The configuration management system employed by the enterprise shall have the ability to manage different operationally deployed versions of a particular system.
1.12	The handling of system safety and reliability aspects shall be detailed in the process model.
1.13	The process model shall identify the activities required to support the system certification process.

Number	Requirement statement
1.14	The process model shall allow for mechanisms to facilitate the selection of appropriate hardware and software architectures, early in the development life cycle.
1.15	The development process model shall identify infrastructure requirements that will allow the support of test and integration using dedicated avionics data interface protocols.
1.16	The process model shall address obsolescence management.
1.17	The process model shall indicate requirements for development infrastructure establishment.
2.01	The process model shall indicate the detail of the responsibilities of the enterprise with respect to the certification process.
2.02	The process model shall indicate the mechanisms by which continued airworthiness shall be ensured.
3.01	The process model shall indicate the organisational context of the development process, referring to the referenced standards.
3.02	The process model shall provide guidance for the division of the development process into layers of hierarchy.
3.03	The process model shall identify a set of main project stages (decision gates) for a development increment.
3.04	The process model shall define sub-processes that constitute the development process, using guidance from the referenced standards.
3.05	For each development stage and sub-process, the major activities constituting these shall be identified.
4.01	The process description shall indicate which generic development and management plans and standards are required for a typical project.
4.02	Reviews of the technical effort, at the different project stages, shall be defined in the process model.

Number	Requirement statement
4.03	The activities required to control life cycle data and project information, shall be indicated in the process model.
4.04	The activities required to achieve effective configuration management, shall be indicated in the process model.
4.05	The activities required to identify, manage and track problems, shall be indicated in the process model.
4.06	Interfaces (e.g. risks identified, work scope identified, resource requirements identified, etc.) with other project and enterprise processes need to be identified and documented in the process model.
4.07	The process model shall indicate how process improvement feedback is provided to the quality assurance processes.
4.08	The process model shall indicate process metrics to facilitate measurement of progress of the technical effort.
4.09	The process model shall indicate the major activities and their expected outcomes, for the requirements management process.
4.10	The process model shall indicate how requirements traceability to requirements originating at higher layers of system hierarchy shall be achieved.
4.11	The process model shall indicate how traceability to lower level requirements (design or requirements implementation strategies) shall be achieved.
4.12	The process model shall indicate how the status of a requirement shall be tracked.
4.13	The process model shall indicate how requirements shall be associated with requirements verification requirements.
4.14	The process model shall indicate how every requirement shall be linked with its associated verification evidence.
4.15	The process model shall indicate how every requirement shall be linked with its associated validation evidence.

Number	Requirement statement
4.16	The process model shall indicate the necessary system safety activities and expected outcomes.
4.17	The method by which hazards are identified and tracked shall be described in the process model.
4.18	Criteria for assessing hazard severity shall be captured in the process model.
4.19	Methods for assessing system reliability shall be identified.
4.20	The process model shall indicate the activities and expected outcomes to support the certification process.
4.21	The process model shall indicate how continued airworthiness of a system shall be ensured.
4.22	The process model shall indicate a comprehensive set of end product realisation activities and expected outcomes.
4.23	The process model shall indicate methods for the integration of engineering specialities.
4.24	The process model shall indicate the points in the project workflow where sub-system development requirements shall be captured.
4.25	The process model shall indicate the points in the project workflow where enabling product requirements shall be captured.
4.26	The process model shall indicate the points in the project workflow where supporting documentation requirements shall be captured.
4.27	The process model shall indicate the points in the project workflow where sub-system procurement requirements shall be captured.
4.28	The process model shall provide guidelines for the preparation of end product realisation life cycle data.
4.29	The process model shall indicate how traceability between requirements and design or requirements implementation strategies shall be achieved.

Number	Requirement statement
4.30	The process model shall indicate how the integrity of the design life cycle data will be ensured.
4.31	The process model shall make provision for methods to perform design changes during the integration process without affecting the integrity of the design.
4.32	The process model shall indicate the required set of verification activities and expected outcomes.
4.33	The process model shall indicate methods for specifying requirements implementation verification requirements.
4.34	The process model shall indicate methods for reviewing and approving the implementation of a requirement where it was determined that it shall be verified by review only.
4.35	The process model shall indicate methods for analysing and approving the implementation of a requirement where it was determined that it shall be verified by analysis.
4.36	The process model shall indicate methods for testing and approving the implementation of a requirement where it was determined that it shall be verified by test.
4.37	The process model shall present guidelines for the preparation and handling of verification evidence.
4.38	The process model shall indicate hand-over interfaces between the system, software and hardware levels.
4.39	The process model shall indicate how verification independence shall be achieved.
4.40	The process model shall indicate points in the project workflow where informal testing is feasible.
4.41	The process model shall present guidelines for the preparation of production acceptance tests.

Number	Requirement statement
4.42	The process model shall indicate interfaces with the operational test and evaluation process.
4.43	The process model shall indicate the required set of infrastructure management activities and expected outcomes.
4.44	The process model shall describe the interfaces to the project management processes, indicating the development infrastructure requirements.
4.45	The process model shall indicate methods for controlling the configuration of the development infrastructure, associated with given development increment.
4.46	The process model shall indicate methods for controlling the configuration of the policies and procedures, associated with given development increment.
4.47	The process model shall indicate the breadboarding process in the context of the development life cycle, for a generic project.
4.48	The process model shall indicate interfaces between breadboarding activities and other life cycle processes.
4.49	The process model shall present guidelines for the preparation and handling of breadboarding process outcomes.
5.01	The process model shall provide a means where requirements for the establishment of specific methodologies are indicated to the quality management system.

Chapter 4: Process description

4.1 Introduction

A framework for the process model is presented in this chapter. The organisational context of the development process is identified and a process architecture in terms of process layers, development life cycle stages and developmental threads is described. The context of breadboarding activities is identified and design guidelines for future detailed design of process elements are presented. A matrix indicating allocation of the process characteristics to the requirements derived in Chapter 3: is presented in Appendix B.

4.2 Process context

A process context definition is required to delineate the scope of the process, i.e. to specify which of the activities related to the development of a system need to be described in the process model description and to indicate interfaces to other relevant processes.

The process for the development of airborne electronic equipment exists within the greater organisation and interacts with other processes, as discussed in section 3.3. The process model described below describes the activities related to the realisation of the product. A schematic representation of the context of the process for the development of the product is shown in Figure 9

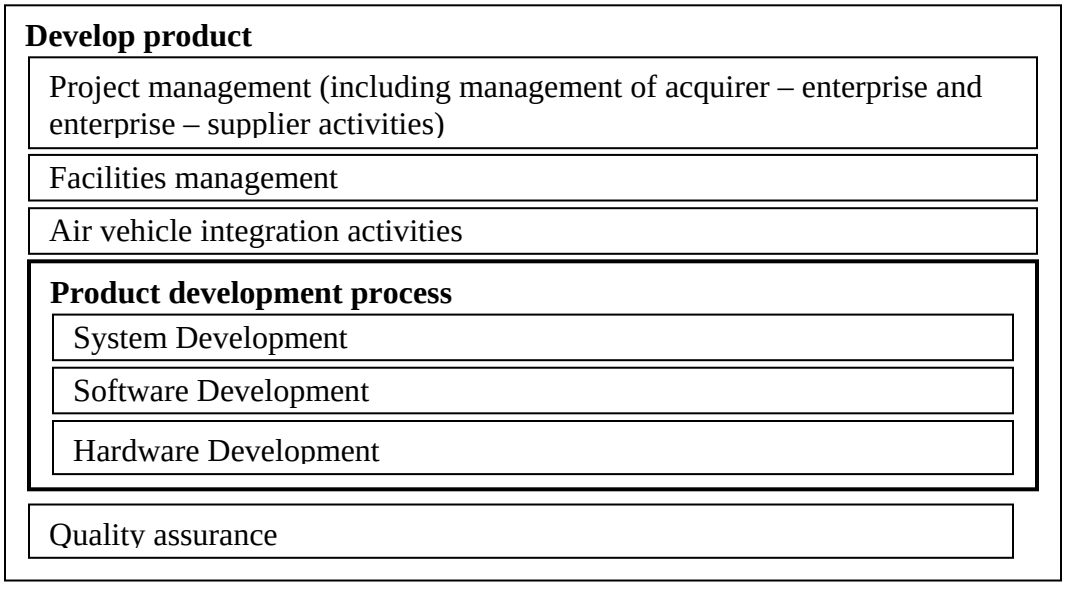

Figure 9: Product development and support concept contexts

Interfaces between the development process and other processes include traceability between the contract and the requirements specifications, allocation of resources and progress reports.

4.3 Process architecture

A design was made to address the requirements of Chapter 3. An allocation matrix is provided at the end of this chapter (Chapter 4) to provide traceability between the input requirements (the formal listing provided in Chapter 3) and the relevant process elements as shown in figures 10 and 11.

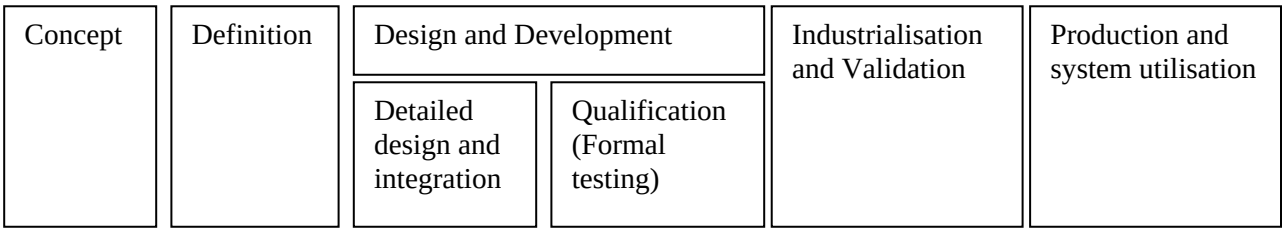

Figure 10: Project stages, from development point of view

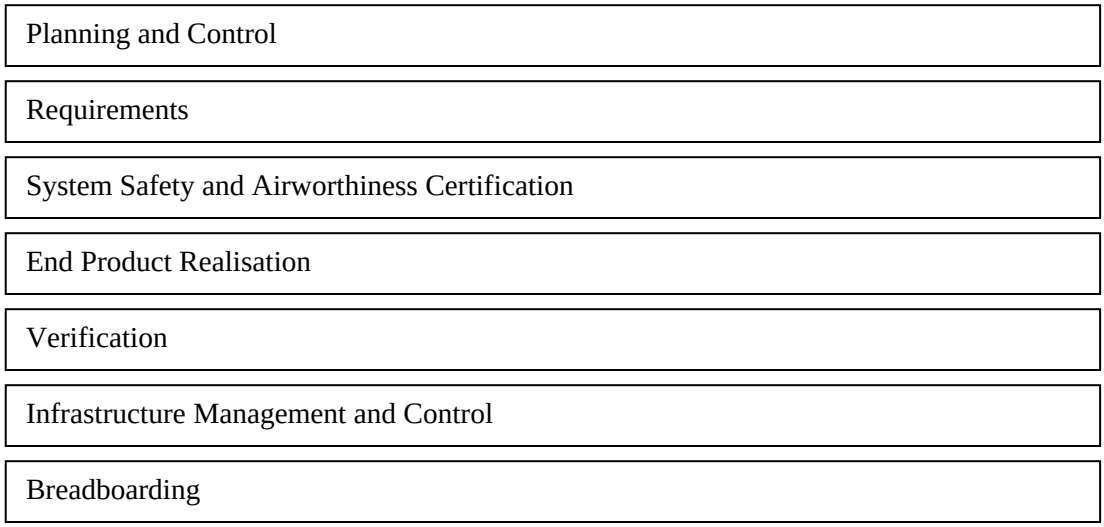

Figure 11: AESDP developmental threads

This process structure supports the concepts of iterative and incremental development. The following sub-sections describe this architecture in more detail.

4.4 Process layers

The development process for an airborne electronic system is divided into a system, software and hardware layer, for organisational purposes and to simplify specification tree structures.

The architecture of the process within each layer is generically similar; sub-system processes may exist within each layer, also having similar process architectures.

4.5 Development life cycle stages

Life cycle is defined in RTCA/DO-178B as: (1) An ordered collection of processes determined by an organisation to be sufficient and adequate to produce a product. (2) The period of time that begins with the decision to produce or modify a product and ends when the product is retired from service.

The life cycle is divided into stages to assist in the structuring of the development effort. The development life cycle phases described in this section apply to all process layers and sub-system development processes within each layer.

The life cycle selected for this process model is tailored from RSA MIL STD 3, [24] as the primary customer for Denel Aviation is the South African Air Force.

A baseline is established at the end of each stage, which co-insides with a formal review. The purpose of the reviews at the end of each development stage is to confirm that the objectives of the stage were achieved and to provide visibility of progress to all stakeholders. These reviews are currently based on MIL-STD-1521B [23].

A development program shall be executed according to the life cycle stages shown diagrammatically in Figure 10.

4.5.1 Concept stage

The objectives of this stage are to understand the *scope of the project* and to *identify sources of requirements*.

The project is planned in detail, a conceptual design is made to indicate the major functions and a functional hazard assessment is performed to determine the high level safety requirements. The stage is concluded with a planning review, a concept evaluation review and a system safety review. The functional baseline is established when all the outcomes from the activities in this stage are documented and approved.

4.5.2 Definition stage

During this stage all *requirements* related to the project (new equipment or equipment modification) should be finalised.

The requirements are analysed in detail and each user requirement and derived requirement is recorded in the System Requirements Specification and in the requirements management database tool. Human-machine interfaces and crew operating definitions are determined and recorded. A means of compliance for every requirement is established and recorded. A preliminary design is made, to confirm that the proposed system architecture shall be able to meet with the requirements. The stage is concluded with a preliminary design review and the allocated baseline is established when all the outcomes from the activities in the definition stage are documented and approved.

4.5.3 Design and development stage

In this stage the requirements are *implemented* in hardware and software.

This stage is sub-divided into a detailed design and integration sub-stage and a qualification sub-stage. This sub stage structure is required in order to align the life cycle with the life cycle model presented in MIL STD 3 [24]. During the detailed design and integration sub-stage the detailed design is completed, all test cases are specified, system integration is performed and first article components are manufactured and procured. Critical design reviews are conducted on sub-systems when suitable maturity of design is reached. Upon completion of the critical design reviews a design baseline is established. When the system integration is completed, the configurations of all configuration items as well as that of the test and development environments are frozen. A test readiness review is conducted and the formal test baseline is established. During the qualification sub-stage, all formal tests, including flight tests, are conducted. The stage is concluded with the formal qualification review, and the product baseline is established.

4.5.4 Industrialisation stage

During the industrialisation stage, preparations are made for the *production* of the system. This aspect of the life cycle is excluded from the scope of the study. However, verification and validation activities are performed during this stage which is of significance with respect to the process model.

In particular, operational test and evaluation (OT&E) tests are performed by the user, and the logistics process associated with the project is evaluated.

4.5.5 Production and system utilisation stages

The details of the production stage and subsequent life cycle stages are excluded from the scope of this study, except where changes to the requirements and design are concerned.

4.5.6 Transition management

Transition between stages shall be managed via the formal reviews described in the next section.

4.6 Developmental threads

A number of “threads” can be identified in the development of a system, as shown in Figure 11.

These threads consist of series of activities, where successive activities with related objectives build on outcomes from previous activities in the thread. Interfaces exist between different threads.

In the following sub-sections, the different activities are described with reference to the relevant thread to which it belongs. Each activity is stated in bold and underlined (as with the requirements in Chapter 3) so as to stay with a convention.

4.7 Planning and control thread

This thread includes the processes to manage the change, keep track of the configuration status and ensuring the quality of work.

4.7.1 Concept stage

Activity 1-1-1: Development management

Description:

The development methods, environments and standards are identified and collated in the development plans.

Skill and resource requirements are determined and indicated to project management to contract these and assign roles and responsibilities.

All other interfaces, e.g. to the air vehicle layer of the hierarchy and the logistics support development process, shall be planned and the mechanisms identified. Important interfaces include the management of risk and obsolescence.

As part of the development management process, the applicable metrics by which engineering performance shall be measured are determined, and the metrics associated with the concept stage activities are collated and indicated to project management and quality assurance processes.

Expected outcomes:

a. Development standards

b. Development plans

c. Collated metrics

Activity 1-1-2: Change control

Description:

The change is initiated by means of a change request document which is to be reviewed by a Configuration Control Board (CCB) specifically constituted for the project and approved by the designated authority, to provide traceability in terms of airworthiness requirements. At this first meeting of the CCB related to the development increment, the feasibility of the proposed design change is considered, and approval is granted for the preparation of a change authorisation document. A typical industry standard change request document is the Request for Change (RFC).

A Master Record Index (MRI) is established that keeps track of all the life cycle data associated with the development increment.

The functional baseline is established after all life cycle data are approved.

Expected outcomes:

a. RFC approved

b. MRI preliminary issue, containing documentation family tree framework, issued.

c. Functional baseline established

Activity 1-1-3: Technical pre-study

Description:

The project scope is determined, resource requirements identified, timescales are estimated and risks are identified. These are recorded in a technical pre-study report. Note that this report shall remain a company confidential document with restricted distribution due to the nature of the information contained therein.

This document is also an input to the project management processes.

Expected outcomes:

a. Technical Pre-study Report

Activity 1-1-4: Planning review

Description:

The plans from activity 1-1-1 are reviewed at the system, software and hardware layers, and formally approved by the project sponsor.

Expected outcomes:

a. Development plans formally accepted

b. Minutes of planning review

Activity 1-1-5: Concept evaluation review

Description:

The operational concept definition and requirements lists (from activity 2-1-1 in the requirements thread) are reviewed and formally accepted by the project sponsor.

Expected outcomes:

a. Concepts as put forward in OCD formally accepted

4.7.2 Definition stage

Activity 1-2-1: Development management

Description:

The development plans are updated to reflect changes resulting from activities in the concept stage. The updated plans are published and stakeholders informed.

Metrics associated with the concept stage activities are collated and indicated to project management and quality assurance processes.

Expected outcomes:

a. Updated development plans

b. Collated metrics

Activity 1-2-2: Change control

Description:

The change is quantified in a change authorisation document which is to be reviewed by the Configuration Control Board (CCB) and approved by the designated authority. At this second meeting of the CCB, the proposed design increment is approved. A typical industry standard change authorisation document is the Engineering Change Proposal (ECP).

The MRI is updated to incorporate references to life cycle data produced during the definition stage.

The allocated baseline is established.

Expected outcomes:

a. ECP approved

b. Allocated baseline established

Activity 1-2-3: Preliminary design review

Description:

The system, software and hardware requirements specifications, the proposed methods for demonstrating compliance with the requirements and the proposed architecture by which requirements will be implemented, are reviewed and formally accepted by the project sponsor.

Expected outcomes:

a. Requirements formally accepted

b. Proposed architecture formally accepted

c. Proposed means of compliance (MoC) formally accepted

4.7.3 Design and development stage

4.7.3.1 Detailed design, implementation and integration

Activity 1-3-1: Development management

Description:

The development plans are updated to reflect changes resulting from activities in the definition stage. The updated plans are published and stakeholders informed.

Metrics associated with the detailed design stage activities are collated and indicated to project management and quality assurance processes.

Expected outcomes:

a. Updated development plans

b. Collated metrics

Activity 1-3-2: Change control

Description:

The MRI is updated to incorporate references to life cycle data produced during the detailed design, implementation and integration stage.

The product baseline is established.

Expected outcomes:

a. MRI updated

b. Product baseline established

Activity 1-3-3: Problem reporting and corrective action system

Description:

The requirements for a problem reporting and corrective action system (PRACAS) for use on the product and manufacturing baselines baseline are determined and recorded. This process shall be linked to the formal change process.

An insular debugging process control mechanism is established to control changes during the implementation and integration process. The objective of this process is to ensure that problems are tracked and that the life cycle data accurately reflects the status of the design at all times. Note that this process is not required for activities within the breadboarding thread.

Expected outcomes:

a. PRACAS requirements captured

b. Insular debugging process activated

Activity 1-3-4: Critical design review

Description:

The objective of the CDR is to conclude the design stage of development and to ensure that the activities constituting this stage are completed and that their objectives were met.

Expected outcomes:

a. Detailed design stage life cycle data formally accepted

Activity 1-3-5: Test readiness review

Description:

Although this review is only defined for software in MIL-STD-1521 [23], it is proposed for use on all layers. It shall be used to confirm that all the pre-requisites for formal qualification tests were met and that the configuration management audit prior to formal testing was completed successfully.

It shall also be verified that the test environments are configured and that all instruments used are calibrated as required.

Expected outcomes:

a. Test documentation formally accepted

b. Approval granted for formal testing

Activity 1-3-6: Formal test preparation

Description:

Ensure that all life cycle data required for formal testing (including flight testing is complete and that no alterations can be performed anymore, as the test results from the following activities can only be considered valid if the status of the life cycle data and test objects remain unchanged. All configuration and quality assurance audits must be completed.

Expected outcomes:

a. Life cycle data for build increment “frozen” for formal tests, FCA, PCA

4.7.3.2 Qualification

Activity 1-4-1: Development management

Description:

The development plans are finally updated to reflect changes resulting from activities in the previous stages. The updated plans are now linked to the build increment as they form part of the information set that shall be used to re-produce the build.

Metrics associated with the qualification stage activities are collated and indicated to project management and quality assurance processes.

Expected outcomes:

a. Updated development plans linked to build increment

b. Collated metrics

Activity 1-4-2: Change control

Description:

No changes are permitted during this stage. The MRI shall be updated to incorporate reference to test reports.

Expected outcomes:

a. MRI updated

Activity 1-4-3: Problem reporting and corrective action (PRACAS)

Description:

The PRACAS (from activity 1-3-3) is implemented and available to record and track any problem identified during the formal tests.

Problems shall be recorded (in problem reports) and indicated to the formal change process, i.e. any problem identified shall only be addressed after approval of an RFC raised in response to a problem report.

Expected outcomes:

a. PRACAS established

b. Problem reports

Activity 1-4-4: Formal qualification review

Description:

This review shall be conducted when all the test results and verification evidence is available. The objective of this review is to verify that the completed system complies with the requirements as accepted at the PDR and that all aspects agreed to with the certification authorities as presented in the Certification Plan (see 4.14.1.1.2) were met.

Expected outcomes:

a. Requirements verification evidence formally accepted

Activity 1-4-5: Software conformity review

Description:

The software accomplishment summary is formally accepted at this review. The objective of this review is to verify that the software complies with the requirements as accepted at the PDR and that all aspects agreed to with the certification authorities as presented in the Plan for Software Aspects of certification (PSAC) (see 4.14.1.1.2) were met.

Expected outcomes:

a. Requirements verification evidence formally accepted

b. Software accomplishment summary formally accepted

Activity 1-4-6: Delivery preparation

Description:

All life cycle data associated with the build increment, including test results, is collated, configured and audited, and provided to the industrialisation planning team.

Expected outcomes:

a. Life cycle data for build increment configured and audited

4.7.4 Industrialisation stage

Activity 1-5-1: Development management

Description:

It is ensured that the development cycle for the current increment build is completed; supporting activities are co-ordinated.

Metrics associated with the industrialisation stage activities are collated and indicated to project management and quality assurance processes.

Expected outcomes:

a. Collated metrics

Activity 1-5-2: Change control

Description:

The Engineering Change Proposal (ECP) is closed, i.e. no further changes using the approved modification authorisation may be performed.

The manufacturing baseline is established.

Expected outcomes:

a. ECP closed

b. Manufacturing baseline established.

Activity 1-5-3: Problem reporting and corrective action (PRACAS)

Description:

The PRACAS (from activity 1-3-3) is used to record and track any problem identified during the industrialisation and validation stage.

Problems shall be recorded (in problem reports) and indicated to the formal change process, i.e. any problem identified shall only be addressed after approval of an RFC raised in response to a problem report.

Expected outcomes:

a. Problem reports

4.7.5 Production and system utilisation stage

Activity 1-6-1: Development management

Description:

Outstanding actions associated with the build increment are resolved, corrective action with respect to unresolved matters is planned.

A close out report for the build increment is prepared, to be provided to the project management for project finalisation. This report should also include a summary of lessons learned, accompanied by recommendations as to how the development process should be improved.

Metrics associated with the production and system utilisation stage activities are collated and indicated to project management and quality assurance processes.

Expected outcomes:

a. Outstanding actions from current build increment finalised

b. Development increment close-out report (lessons learned recorded)

c. Collated metrics

Activity 1-6-2: Change control

Description:

The Request for Change (RFC) is closed, signifying the completion of all activities related to the build increment.

Expected outcomes:

a. RFC closed

Activity 1-6-3: Problem reporting and corrective action (PRACAS)

Description:

The PRACAS (from activity 1-3-3) is used to record and track any problem identified during the production and system utilisation stage.

Problems shall be recorded (in problem reports) and indicated to the formal change process, i.e. any problem identified shall only be addressed after approval of an RFC raised in response to a problem report.

Problem reports shall also be indicated to the continued airworthiness thread.

Expected outcomes:

a. Problem reports addressed

4.8 Requirements thread

The requirements thread is the “core” thread for the development of a system.

The primary objective of the systems engineering effort is to identify and clearly describe all the requirements that the intended system needs to fulfil, to ensure that ensuing design process outcomes are commensurate with the requirements and to verify that the realised system complies with the requirements.

The requirement thread shall therefore consist of activities which ensure the adequate management of requirements, and the tracking of the implementation status of the requirements.

The requirements thread activities and their outcomes, associated with the development stages, are the following:

4.8.1 Concept stage

Activity 2-1-1: Requirements capture

Description:

Requirements statements from all stakeholders are collated. The objective is to ensure that as many possible requirements are captured, to reduce the susceptibility for requirements to be added in due course, which add to costs and cause delays.

Expected outcomes:

a. Requirements lists

b. Stakeholders lists

Activity 2-1-2: Concept definition

Description:

A document shall be produced to provide visibility on the proposed system or modification to all stakeholders. This document should contain a high level abstraction of system functions and the allocation of these to software and hardware. It should describe possible scenarios and implementation propositions. These propositions should serve only to illuminate conceptual ideas and should not pre-empt a design.

Typical aspects to be addressed include:

- Trade-off studies
- Identification of appropriate technologies
- Identification of technical risks
- Functions the intended system is to fulfil

Expected outcomes:

a. Operational Concept Definition

4.8.2 Definition stage

Activity 2-2-1: Requirements analysis

Description:

The requirements emanating from the concept stage are studied. The amassed requirements are analysed, possible implementations are investigated, and technical and commercial risks are identified.

Sources of requirements include the following:

- URS;
- Contract;
- Standards;
- HMI workgroup outcomes;
- System safety process;
- Existing avionics architectures;

- New equipment specifications;
- Standard operating procedures and doctrines.

This list is not exhaustive.

Derived requirements are developed.

Use should be made of modelling, simulation and prototyping where feasible to improve understanding of the requirements. Other requirements analysis techniques, such as the development of scenarios, should also be employed.

The task culminates in a System or Software Requirements and in a populated requirements tracking database.

Requirements shall typically have the following attributes:

- Type
- Implementation status
- Means of compliance
- Verification status

Expected outcomes:

a. System requirements specification

b. Requirements database populated

Activity 2-2-2: Human-machine interface (HMI) definition

Description:

Most airborne electronic equipment has an interaction with the crew. The detail of these interactions as well as associated anthropometric requirements are identified and documented, in conjunction with aircrew with relevant experience.

Expected outcomes:

a. HMI descriptions

4.8.3 Design and development stage

4.8.3.1 Detailed design, implementation and integration

Activity 2-3-1: Requirements implementation corroboration

Description:

Traceability between designs and requirements, and test coverage are analysed and recorded in the requirements tracking database.

Expected outcomes:

a. Requirements allocated

b. Requirements database updated

4.8.3.2 Qualification

Activity 2-4-1: Requirements verification confirmation

Description:

The test results, analysis reports and inspection reports from activities 5-4-1, 5-4-2 and 5-4-3 are reviewed and cross referred in the requirements tracking database.

Expected outcomes:

a. Requirements verified

b. Requirements database updated

4.8.4 Industrialisation stage

Activity 2-5-1: Requirements validation confirmation

Description:

Outcomes from the OT&E activities are used to verify and validate requirements. Cross references are recorded in the requirements tracking database.

Expected outcomes:

a. Requirements validated

b. Requirements database updated

4.8.5 Production stage and system utilisation stages

Activity 2-6-1: Requirements updating

Description:

Issues arising during operational use of the system are captured in the requirements tracking database.

Expected outcomes:

a. New requirements captured in register

4.9 System safety / airworthiness / certification thread

The systems safety / airworthiness / certification thread deals with activities which ensure that all safety related requirements are identified and that adequate steps are taken in the design and use of the system to ensure that the safety requirements are met. Airworthiness requirements are identified and managed, and the processes required for the system under development to obtain statutory approval are executed.

4.9.1 Concept stage

Activity 3-1-1: Functional hazard assessment (FHA)

Description:

A functional hazard assessment (FHA) uses the outcomes from the concept stage requirements activities, to determine the criticality of functional failures associated with each requirement. This information is required to determine reliability requirements in the design, and the software development level, as described in RTCA/DO-178B [3].

Consideration should be given to using a database for the management of hazards identified by the FHA.

The FHA shall be maintained during the entire development cycle, and shall be updated when new requirements are added, when a requirement changes or when a requirement is better understood.

Expected outcomes:

a. Functional hazard assessment report

b. Preliminary software criticality assessment report

Activity 3-1-2: Certification preparation

Description:

The liaison process with the certification authorities is established. Agreement shall be reached on the basis on which final airworthiness approval will be granted for the system. This includes a selection of the standards against which system characteristics will be measured.

Expected outcomes:

a. Certification basis agreement

4.9.2 Definition stage

Activity 3-2-1: Hazard tracking

Description:

After the issue of the system and software requirements specifications, the requirements are analysed to determine if new or derived requirements introduced a change in the criticality of functional failures.

Expected outcomes:

a. Updated FHA report

Activity 3-2-2: Preliminary system safety assessment (PSSA)

Description:

In performing the preliminary system safety assessment (PSSA), the proposed system architecture is analysed to determine how the failures identified in the FHA can be caused, and inputs are provided to the design process in terms of aspects that can implemented in order to meet with the system safety requirements.

Expected outcomes:

a. PSSA report

b. Software criticality assessment report

4.9.3 Design and development stage

4.9.3.1 Detailed design, implementation and integration

Activity 3-3-1: Flight airworthiness preparation

Description:

The documentation required to allow flight testing of the system under development is collated and reviewed. Traceability to source documents needs to be demonstrated.

Expected outcomes:

a. Preliminary declaration of design and performance (DDP)

b. Certificate for flight trials (CFT)

4.9.3.2 Qualification

Activity 3-4-1: Operational airworthiness preparation

Description:

The documentation required for airworthiness approval of the system under development is collated and reviewed. Traceability to source documents needs to be demonstrated.

Expected outcomes:

a. Declaration of design and performance (DDP)

4.9.4 Industrialisation stage

Activity 3-5-1: Airworthiness approval

Description:

All the formalities required for the certification authority to grant airworthiness approval of the system under development is completed.

Expected outcomes:

a. Application for statutory approval

Activity 3-5-2: System safety assessment (SSA)

Description:

The system safety assessment (SSA) evaluates the implemented system in terms of the safety objectives from the FHA and PSSA.

All the system safety documents remain active, and are re-visited whenever a requirement is added or changed as a consequence of activities from the industrialisation stage, to assess and analyse the safety implications of the change.

Expected outcomes:

a. SSA report

4.9.5 Production stage and system utilisation stages

Activity 3-6-1: Continued airworthiness

Description:

The system safety documents remain active, and are re-visited whenever a requirement is added or changed as a result of occurrences during the operational use of the system, to assess and analyse the safety implications of the change.

Expected outcomes:

a. Incident investigation reports

b. Failure investigation reports

c. Service bulletins

4.10 End product realisation thread

The end product realisation thread consists of the activities that produce a tangible end product and enabling products, as well as the life cycle data required to reproduce these, such as design descriptions, source code and drawings.

Note that all life cycle data produced as outcomes of development activities shall be subjected to strict quality assurance and configuration management practises.

4.10.1 Concept stage

Activity 4-1-1: Development preparation

Description:

The life cycle data and associated formats requirements, to be produced during the development of the build increment are identified and indicated to the development management activity (1-1-1).

Expected outcomes:

a. Life cycle data formats selection

4.10.2 Definition stage

Activity 4-2-1: Preliminary analysis

Description:

The system architecture is defined and high level internal and external interface requirements are determined.

Sub systems and enabling products are identified. For each sub-system and enabling product, a development process should be planned as described in section 4.3. If any new technology is to be developed in support of the project, the requirements shall also be identified during this activity. Note that the process model for technology development is different to the AESDP, due to the nature of the work.

Requirements for supporting documentation, such as user manuals, are identified.

Items to be procured are identified and detailed specifications for these are prepared.

Expected outcomes:

a. Architecture design description

b. Interface requirements specifications

c. Sub systems and enabling products requirements

d. Supporting documentation requirements

e. Specifications for items to be procured

4.10.3 Design and development stage

4.10.3.1 Detailed design, implementation and integration

Activity 4-3-1: Detailed design

Description:

The detailed design is produced on the system, software and hardware layers.

Expected outcomes:

a. Design descriptions

b. Interface control documents (ICDs)

Activity 4-3-2: Design implementation (SW layer)

Description:

The detailed design is implemented in source code, and in the case of databases, the databases are set up.

Expected outcomes:

a. Source code

b. Databases

Activity 4-3-3: Hardware procurement (HW layer)

Description:

Hardware (see 4-2-1) is procured and inspected.

Expected outcomes:

a. Hardware items delivered

b. Vendor documentation

Activity 4-3-4: Integration

Description:

Software – software, software – hardware and hardware – hardware integration tasks are performed. Changes are introduced to design documents and ICDs as a consequence of the integration process.

Expected outcomes:

a. Updated life cycle data

Activity 4-3-5: Informal tests

Description:

Informal tests are employed to verify that the completed product will pass formal tests, as well as to verify the correctness of the test specifications. Note that the results of informal tests can not be used as certification evidence. Outputs from the informal tests are fed to the product definition.

Expected outcomes:

a. Draft test reports

b. Correctness of test documentation and maturity of test subjects confirmed

4.10.3.2 Qualification

Activity 4-4-1: Preparation of support documentation

Description:

Supporting documentation (see 4-2-1) are prepared, inspected and approved. This task is typically sub-contracted to a technical publications organisation.

Expected outcomes:

a. Approved supporting documentation

Activity 4-4-2: Design increment closure

Description:

It is ensured that all the design documentation associated with the build increment is correct, complete and approved.

Expected outcomes:

a. Life cycle data for the increment finalised

4.10.4 Industrialisation stage

There are no end product realisation activities in the industrialisation stage. (Note that any change required to the product as a consequence of other activities during this stage requires a new build increment).

4.10.5 Production and system utilisation stage

As for the industrialisation stage.

4.11 Verification thread

The verification thread consists of all the activities required to verify that the requirements are satisfied.

4.11.1 Concept stage

Activity 5-1-1: Verification preparation

Description:

Verification methods, standards against which verification shall be performed and the means of demonstrating compliance, resources required in support of verification efforts and the verification schedule shall be planned. See also section 4.4.5.1.

Expected outcomes:

a. Applicable standards selected

4.11.2 Definition stage

Activity 5-2-1: Verification planning

Description:

The requirements verification requirements, and consequently the means by which it will be demonstrated that a requirement was satisfied in the implemented system, referred to as the means of compliance (MoC), shall be determined for every requirement, and agreed to by the certification authority. The MoC is captured in the requirements database.

This sub-process shall also provide for the planning of verification of legacy or precedented systems.

Expected outcomes:

a. Requirements verification requirements (captured in requirements database)

b. Means of compliance specifications (captured in requirements database)

4.11.3 Design and development stage

4.11.3.1 Detailed design, implementation and integration

Activity 5-3-1: Verification specification

Description:

The detailed test cases by which compliance shall be proved are developed and documented. These test cases should typically be collated in the following documents:

- Lab test specifications
- Ground test specifications

- Flight test specifications

Analyses and inspections (reviews) required as part of the verification effort are identified and tasks allocated.

Expected outcomes:

a. Test cases (collated in rig and ground test specifications)

b. Flight test request

c. Analyses requirements

d. Inspection requirements

4.11.3.2 Qualification stage

Activity 5-4-1: Requirements and design verification

Description:

Formal test are performed by the assigned quality assurance representatives, and in the case of qualification flight tests, by the assigned flight test organisation. This ensures verification independence with respect to testing.

If a discrepancy is reported in this stage, it requires a new development increment to be handled via the formal change process, to address the problem.

Tests as identified in 5-3-1, as well performed and results are documented.

Expected outcomes:

a. Test reports

Activity 5-4-2: Analysis

Description:

Analyses as identified in 5-3-1, as well as traceability and coverage analyses are performed and documented.

Expected outcomes:

a. Engineering reports

Activity 5-4-3: Inspections

Description:

Inspections as identified in 5-3-1 are performed and documented.

Expected outcomes:

a. Inspection reports

4.11.4 Industrialisation stage

Activity 5-5-1: Production acceptance preparation

Description:

The acceptance test procedure (ATP) for the acceptance of production units is prepared. This ATP confirms that the production units conform to the qualified design.

Expected outcomes:

a. Production acceptance test procedures

Activity 5-5-2: Operational test and evaluation (OT&E)

Description:

The operational test and evaluation tests are performed by the user. These tests serve to validate the user's requirements.

Expected outcomes:

a. Test and evaluation reports (from user)

4.11.5 Production stage and system utilisation stages

No verification activities identified.

4.12 Infrastructure thread

4.12.1 Concept stage

Activity 6-1-1: Infrastructure preparation

Description:

The infrastructure requirements for the development increment is identified and indicated to the planning process.

The requirements database for use by the requirements management process is set up at this stage, to support the requirements capture processes in the concept stage.

Expected outcomes:

a. Development environment requirements

b. Requirements database established

4.12.2 Definition stage

Activity 6-2-1: Infrastructure management

Description:

The development environment for the build increment is set up. It shall be ensured that the description of the environment is sufficiently documented to repeat the build at any future stage if required.

Expected outcomes:

a. Development environment established

4.12.3 Design and development stage

4.12.3.1 Detailed design, implementation and integration

Activity 6-3-1: Infrastructure management

Description:

The verification environment for the build increment is set up. It shall be ensured that the description of the environment is sufficiently documented to repeat the build at any future stage if required.

Expected outcomes:

Verification environment established

4.12.3.2 Qualification

Activity 6-4-1: Infrastructure management

Description:

The status of the development and verification environment used for the build increment is captured. Note that this includes the issue status of all relevant plans, as well as referenced quality

system policies and procedures. This information forms part of the life cycle data defining the build.

Expected outcomes:

a. Development and verification environment configured for build increment

4.12.4 Industrialisation stage

No infrastructure related activities identified.

4.12.5 Production and system utilisation stage

No infrastructure related activities identified.

The development processes on the system software and hardware layers are shown in **Error! Reference source not found.**, **Error! Reference source not found.**, and **Error! Reference source not found.**.

PHASE / THREAD	CONCEPT	DEFINITION	DESIGN AND DEVELOPMENT		INDUSTRIALISATION (DELIVERY AND VALIDATION)	PRODUCTION AND SYSTEM UTILISATION
			DETAIL DESIGN, IMPLEMENTATION AND INTEGRATION	QUALIFICATION (FORMAL TEST)		
PLANNING AND CONTROL	1-1-1 DEVELOPMENT MANAGEMENT *-Development plans* *-Collated metrics* 1-1-2 CHANGE CONTROL *-Functional baseline established* *-MRI prelim issue established* *-RFC approved* 1-1-3 TECHNICAL PRE-STUDY *-Technical pre-study report* 1-1-4 PLANNING REVIEW *-Plans formally accepted* 1-1-5 CONCEPT EVALUATION REVIEW *-Concepts formally accepted*	1-2-1 DEVELOPMENT MANAGEMENT *-Updated development plans* *-Collated metrics* 1-2-2 CHANGE CONTROL *-Allocated baseline established* *-ECP approved* 1-2-3 PRELIMINARY DESIGN REVIEW *-Requirements formally accepted* *-Proposed means of compliance formally accepted* *-Proposed architecture formally accepted*	1-3-1 DEVELOPMENT MANAGEMENT *-Updated development plans* *-Collated metrics* 1-3-2 CHANGE CONTROL *-Product baseline established* *-MRI updated* 1-3-3 PRACAS *-PRACAS requirements captured* *-Insular debugging process activated* 1-3-4 CRITICAL DESIGN REVIEW *-Detailed design phase life cycle data formally accepted* 1-3-5 TEST READINESS REVIEW *-Test documentation formally accepted* *-Approval granted for formal testing* 1-3-6 FORMAL TEST PREPARATION *-Life cycle data for build increment frozen for formal tests*	1-4-1 DEVELOPMENT MANAGEMENT *-Updated development plans* *-Collated metrics* 1-4-2 CHANGE CONTROL *-MRI updated* 1-4-3 PRACAS *-PRACAS established* *-Problem reports addressed* 1-4-6 FORMAL QUALIFICATION REVIEW *-Requirements verification evidence formally accepted* 1-4-8 DELIVERY PREPARATION *-Life cycle data for build increment, including test results, configured and audited*	1-5-1 DEVELOPMENT MANAGEMENT *-Collated metrics* 1-5-2 CHANGE CONTROL *-Manufacturing baseline established* *-ECP closed* 1-5-3 PRACAS *-PRACAS utilised* *-Problem reports addressed*	1-6-1 DEVELOPMENT MANAGEMENT *-Outstanding actions from current build increment finalised* *-Development increment close-out report (lessons learned recorded)* *-Collated metrics* 1-6-2 CHANGE CONTROL *-RFC closed* 1-6-3 PRACAS *-PRACAS utilised* *-Problem reports addressed*
REQUIREMENTS	2-1-1 REQUIREMENTS CAPTURE *-Requirements lists* *-Stakeholders lists* 2-1-2 CONCEPT DEFINITION *-Operational concept definition*	2-2-1 REQUIREMENTS ANALYSIS *-System requirements specification* *-Requirements database populated* 2-2-2 HMI DEFINITION *-HMI descriptions*	2-3-1 REQUIREMENTS IMPLEMENTATION CORROBORATION *-Requirements allocated* *-Requirements database updated*	2-4-1 REQUIREMENTS VERIFICATION CONFIRMATION *-Requirements verified* *-Requirements database updated*	2-5-1 REQUIREMENTS VALIDATION CONFIRMATION *-Requirements validated* *-Requirements database updated*	2-6-1 REQUIREMENTS UPDATING *-New requirements captured in register*
SYSTEM SAFETY / AIRWORTHINESS / CERTIFICATION	3-1-1 FUNCTIONAL HAZARD ASSESSMENT *-Functional hazard assessment report* *-Prelim. software criticality assessment report* 3-1-2 CERTIFICATION PREPARATION *-Certification basis agreement*	3-2-1 HAZARD TRACKING *-Updated FHA report* 3-2-2 PRELIMINARY SSA *-PSSA report* *-Software criticality assessment report*	3-3-1 FLIGHT TEST AIRWORTHINESS PREPARATION *-Preliminary declaration of design and performance issued* *-CFT issued*	3-4-1 OPERATIONAL AIRWORTHINESS PREPARATION *-Declaration of design and performance issued*	3-5-1 AIRWORTHINESS APPROVAL *-Application of statutory approval* 3-5-2 SYSTEM SAFETY ASSESSMENT *-System safety assessment report*	3-6-1 CONTINUED AIRWORTHINESS *-Incident investigation reports* *-Failure investigation reports* *-Service bulletins*
END PRODUCT REALISATION	4-1-1 DEVELOPMENT PREPARATION *-Life cycle data formats selected*	4-2-1 PRELIMINARY DESIGN *-Architecture design description* *-Interface requirements specifications* *-Sub systems and enabling products requirements* *-Supporting documentation requirements* *-Specifications for items to be procured*	4-3-1 DETAILED DESIGN *-Design descriptions* *-ICDs* 4-3-4 INTEGRATION *-Updated life cycle data* 4-3-5 INFORMAL TESTS *-Draft test reports* *-Correctness of test documentation and maturity of test subjects confirmed*	4-4-1 PREPARATION OF SUPPORT DOCUMENTATION *-Approved supporting documentation* 4-4-2 DESIGN INCREMENT CLOSURE *-Life cycle data for the build increment finalised*		
VERIFICATION	5-1-1 VERIFICATION PREPARATION *-Applicable standards selected*	5-2-1 VERIFICATION PLANNING *-Means of compliance specifications (captured in requirements database)*	5-3-1 VERIFICATION SPECIFICATION *-Test cases (collated in rig and ground test specifications)* *-Flight test request* *-Analyses requirements* *-Inspection requirements*	5-4-1 REQUIREMENTS AND DESIGN VERIFICATION TESTS *-Test reports* 5-4-2 ANALYSES *-Engineering reports* 5-4-3 INSPECTIONS *-Inspection reports* *-PCA report* *-FCA report*	5-5-1 PRODUCTION ACCEPTANCE PREPARATION *-Production acceptance test procedures* 5-5-2 OPERATIONAL TEST AND EVALUATION *-Test and evaluation reports*	
INFRASTRUCTURE	6-1-1 INFRASTRUCTURE PREPARATION *-Development environment requirements* *-Requirements database established*	6-2-1 INFRASTRUCTURE MANAGEMENT *-Development environment established*	6-3-1 INFRASTRUCTURE MANAGEMENT *-Verification environment established*	6-4-1 INFRASTRUCTURE MANAGEMENT *-Development and verification environment configured for build increment*		

Figure 12: System (product) layer

PHASE / THREAD	CONCEPT	DEFINITION	DESIGN AND DEVELOPMENT		INDUSTRIALISATION (DELIVERY AND VALIDATION)	PRODUCTION AND SYSTEM UTILISATION
			DETAIL DESIGN, IMPLEMENTATION AND INTEGRATION	QUALIFICATION (FORMAL TEST)		
PLANNING AND CONTROL	1-1-1 DEVELOPMENT MANAGEMENT *-Development plans* *-Software accomplishment summary* *-Collated metrics* 1-1-2 CHANGE CONTROL *-Functional baseline established* *-Software configuration index preliminary issue established* *-RFC approved* 1-1-4 PLANNING REVIEW *-Plans formally accepted*	1-2-1 DEVELOPMENT MANAGEMENT *-Updated development plans* *-Collated metrics* *-Software accomplishment summary* 1-2-2 CHANGE CONTROL *-Allocated baseline established* *-ECP approved* 1-2-3 PRELIMINARY DESIGN REVIEW *-Requirements and MOCs formally accepted*	1-3-1 DEVELOPMENT MANAGEMENT *-Updated development plans* *-Collated metrics* *-Software accomplishment summary* 1-3-2 CHANGE CONTROL *-Product baseline established* *-Software configuration index updated* 1-3-3 PRACAS *-PRACAS requirements captured* *-Insular debugging process control* 1-3-4 CRITICAL DESIGN REVIEW *-Design life cycle data formally accepted* 1-3-5 TEST READINESS REVIEW *-Test documentation formally accepted* 1-3-6 FORMAL TEST PREPARATION *-Life cycle data for build increment configured*	1-4-1 DEVELOPMENT MANAGEMENT *-Updated development plans* *-Collated metrics* *-Software accomplishment summary (final for increment)* 1-4-2 CHANGE CONTROL *-Software configuration index updated* 1-4-7 SOFTWARE CONFORMITY REVIEW *-Requirements verification results formally accepted* 1-4-8 DELIVERY PREPARATION *-Life cycle data for build increment, including test results, configured.*	1-5-1 DEVELOPMENT MANAGEMENT *-Collated metrics* 1-5-2 CHANGE CONTROL *-Manufacturing baseline established* *-ECP closed* 1-5-3 PRACAS *-PRACAS utilised* *-Problem reports addressed*	1-6-1 DEVELOPMENT MANAGEMENT *-Outstanding actions from current build increment finalised* *-Development increment close-out report (lessons learned recorded)* *-Collated metrics* 1-6-2 CHANGE CONTROL *-RFC closed* 1-6-3 PRACAS *-PRACAS utilised* *-Problem reports addressed*
REQUIREMENTS	2-1-1 REQUIREMENTS CAPTURE *-Requirements list*	2-2-1 REQUIREMENTS ANALYSIS *-Software requirements specification* *-Requirements database populated*	2-3-1 REQUIREMENTS IMPLEMENTATION *-Software requirements allocated* *-Software requirements database updated*	2-4-1 REQUIREMENTS VERIFICATION CONFIRMATION *-Requirements verified* *-Requirements database updated*	2-5-1 REQUIREMENTS VALIDATION CONFIRMATION *-Requirements validated* *-Requirements database updated*	2-6-1 REQUIREMENTS UPDATING *-Requirements capture emanating from PRACAS*
END PRODUCT REALISATION	4-1-1 DEVELOPMENT PREPARATION *-Identification of life cycle data formats*	4-2-1 PRELIMINARY DESIGN *-Software architecture design description* *-Software interface requirements specifications* *-Computer resource utilisation - determined* *-Sub systems and enabling products requirements* *-Supporting documentation requirements*	4-3-1 DETAILED DESIGN *-Software design descriptions* *-Database design* *-Software ICDs* 4-3-2 DESIGN IMPLEMENTATION *-Source code* *-Databases* 4-3-4 INTEGRATION (software/software integration) *-Executable object code* *-Updated life cycle data* 4-3-5 INFORMAL TESTS *-Draft test report* *-Correctness of test documentation and maturity of test subjects confirmed*	4-4-1 PREPARATION OF SUPPORT DOCUMENTATION *-Approved supporting documentation* 4-4-2 DESIGN INCREMENT CLOSURE *-Life cycle data for the build increment finalised*		
VERIFICATION	5-1-1 VERIFICATION PREPARATION *-Identification of applicable standards*	5-2-1 VERIFICATION PLANNING *-Means of compliance specifications (captured in requirements database)* *-Tests, analyses and inspections to be performed on system layer indicated to system layer.*	5-3-1 VERIFICATION SPECIFICATION *-Test cases* *-Flight test request* *-Analyses requirements* *-Inspection requirements*	5-4-1 REQUIREMENTS AND DESIGN VERIFICATION TESTS *-Test reports* 5-4-2 ANALYSES *-Unit test harness developed* *-Unit test results* *-Engineering reports* 5-4-3 INSPECTIONS *-Inspection reports*	5-5-2 OPERATIONAL TEST AND EVALUATION *-Test and evaluation reports*	
INFRASTRUCTURE	6-1-1 INFRASTRUCTURE PREPARATION *-Development environment requirements* *-Requirements database established*	6-2-1 INFRASTRUCTURE MANAGEMENT *-Development environment established* *-Software life cycle environment configuration index*	6-3-1 INFRASTRUCTURE MANAGEMENT *-Verification environment established* *-Software life cycle environment configuration index updated*	6-4-1 INFRASTRUCTURE MANAGEMENT *-Development and verification environment configured for build increment*		

Figure 13: Software layer

PHASE / THREAD	CONCEPT	DEFINITION	DESIGN AND DEVELOPMENT		INDUSTRIALISATION (DELIVERY AND VALIDATION)	PRODUCTION AND SYSTEM UTILISATION
			DETAIL DESIGN, IMPLEMENTATION AND INTEGRATION	QUALIFICATION (FORMAL TEST)		

PLANNING AND CONTROL	1-1-1 DEVELOPMENT MANAGEMENT *-Development plans* *-Collated metrics* 1-1-2 CHANGE CONTROL *-Functional baseline established* *-MRI prelim issue established* *-RFC approved* 1-1-3 PLANNING REVIEW *-Plans formally accepted*	1-2-1 DEVELOPMENT MANAGEMENT *-Updated development plans* *-Collated metrics* 1-2-2 CHANGE CONTROL *-Allocated baseline established* *-ECP approved* 1-2-3 PRELIMINARY DESIGN REVIEW *-Requirements and MOCs formally accepted*	1-3-1 DEVELOPMENT MANAGEMENT *-Updated development plans* *-Collated metrics* 1-3-2 CHANGE CONTROL *-Product baseline established* *-MRI updated* 1-3-3 PRACAS *-PRACAS requirements captured* *-Insular debugging process control* 1-3-4 CRITICAL DESIGN REVIEW *-Design life cycle data formally accepted* 1-3-5 TEST READINESS REVIEW *-Test documentation formally accepted* 1-3-6 FORMAL TEST PREPARATION *-Test documentation and life cycle data for build increment configured*	1-4-1 DEVELOPMENT MANAGEMENT *-Updated development plans* *-Collated metrics* 1-4-2 CHANGE CONTROL *-MRI updated* 1-4-3 PRACAS *-PRACAS established* *-Problem reports addressed* 1-4-6 FORMAL QUALIFICATION REVIEW *-Requirements verification evidence formally accepted* 1-4-8 DELIVERY PREPARATION *-Life cycle data for build increment, including test results, configured*	1-5-1 DEVELOPMENT MANAGEMENT *-Collated metrics* 1-5-2 CHANGE CONTROL *-Manufacturing baseline established* *-ECP closed* 1-5-3 PRACAS *-PRACAS utilised* *-Problem reports addressed*	1-6-1 DEVELOPMENT MANAGEMENT *-Outstanding actions from current build increment finalised* *-Development increment close-out report (lessons learned recorded)* *-Collated metrics* 1-6-2 CHANGE CONTROL *-RFC closed* 1-6-3 PRACAS *-PRACAS utilised* *-Problem reports addressed*
REQUIREMENTS	2-1-1 REQUIREMENTS CAPTURE *-Requirements list*	2-2-1 REQUIREMENTS ANALYSIS *-Hardware requirements specification*	2-3-1 REQUIREMENTS IMPLEMENTATION *-Hardware requirements allocated* *-Hardware requirements database updated*	2-4-1 REQUIREMENTS VERIFICATION CONFIRMATION *-Requirements verified* *-Requirements database updated*	2-5-1 REQUIREMENTS VALIDATION CONFIRMATION *-Requirements validated* *-Requirements database updated*	2-6-1 REQUIREMENTS UPDATING *-Requirements capture emanating from PRACAS*
END PRODUCT REALISATION	4-1-1 DEVELOPMENT PREPARATION *-Identification of life cycle data formats*	4-2-1 PRELIMINARY DESIGN *-Hardware architecture design description* *-Hardware interface requirements specifications* *-Sub systems and enabling products requirements* *-Supporting documentation requirements* *-Specifications for items to be procured*	4-3-1 DETAILED DESIGN *-Hardware design descriptions* *-ICDs* 4-3-3 HARDWARE PROCUREMENT *-Vendor documentation* 4-3-4 INTEGRATION *-Updated life cycle data* 4-3-5 INFORMAL TESTS *-Correctness of test documentation and maturity of test subjects confirmed*	4-4-1 PREPARATION OF SUPPORT DOCUMENTATION *-Approved supporting documentation* 4-4-2 DESIGN INCREMENT CLOSURE *-Life cycle data for the build increment finalised*		
VERIFICATION	5-1-1 VERIFICATION PREPARATION *-Identification of applicable standards*	5-2-1 VERIFICATION PLANNING *-Means of compliance specifications (captured in requirements database)* *-Tests, analyses and inspections to be performed on system layer indicated to system layer*	5-3-1 VERIFICATION SPECIFICATION *-Test cases compiled* *-Environmental tests specified* *-Flight test request* *-Analyses requirements* *-Inspection requirements*	5-4-1 REQUIREMENTS AND DESIGN VERIFICATION TESTS *-Test reports* 5-4-2 ANALYSES *-Engineering reports* 5-4-3 INSPECTIONS *-Inspection reports*	5-5-1 PRODUCTION ACCEPTANCE PREPARATION *-Production acceptance test procedures* 5-5-2 OPERATIONAL TEST AND EVALUATION *-Test and evaluation reports*	
INFRASTRUCTURE	6-1-1 INFRASTRUCTURE PREPARATION *-Development environment requirements* *-Requirements database established*	6-2-1 INFRASTRUCTURE MANAGEMENT *-Development environment established*	6-3-1 INFRASTRUCTURE MANAGEMENT *-Verification environment established*	6-4-1 INFRASTRUCTURE MANAGEMENT *-Development and verification environment configured for build increment*		

Figure 14: Hardware layer

4.13 Breadboarding

The development process shall be supported by an “informal” process where rapid prototyping, simulations and other experimental work is conducted. The objective is to improve understanding of requirements, to evaluate implementation strategies, demonstrate concepts such as HMI arrangements, and perform other tasks to support decision making at the other threads.

No strict configuration management or other means of process control is enforced. Where outcomes of work performed in the breadboarding thread are required at the other threads, these shall be presented in engineering reports. This breadboarding process shall take place in parallel with the processes described in sections 4.5 and 4.6.

Planning of resources and infrastructure required for breadboarding shall be captured in the development plans described in 1-1-1.

No work performed within this thread can be used as verification evidence for airworthiness purposes.

4.14 Design guidelines for future detail design

The Denel Aviation quality system needs to be updated to make provision for quality policies and procedures associated with particular activities as shown below.

It should be noted that in many cases procedures associated with specific activities exist within the present quality system, these should be identified and reviewed within the context of the model defined in the preceding sections.

Recommended references to sources which contain applicable guidance information with respect to the required procedure or process are indicated where applicable and available, and should be used wherever possible.

4.14.1 Planning and control thread design guidelines

4.14.1.1 Development management

Development management is addressed in activities 1-1-1; 1-2-1; 1-3-1; 1-4-1; 1-5-1; 1-6-1.

4.14.1.1.1 Development standards

Generic versions of the following set of development standards should be developed, to be tailored on a project to project basis.

a. System Requirements Standard;

b. System Design Standard;

c. Software Requirements Standard;

d. Software Design Standard;

e. Software Coding Standard;

f. Hardware Requirements Standard;

g. Hardware Design Standard;

These standards shall describe the methods and rules to be used for specifying and tracking requirements, for designing systems, software and hardware and for software coding.

4.14.1.1.2 Development plans

Templates and procedures are required for the preparation of development plans.

System layer:

a. Systems Engineering Management Plan;

b. Test and Evaluation Master Plan;

c. System Configuration Management Plan;

d. System Quality Assurance Plan;

e. System Safety Plan;

Software layer:

f. Plan for Software Aspects of Certification;

g. Software Development Plan;

h. Software Verification Plan;

i. Software Configuration Management Plan;

j. Software Quality Assurance Plan;

k. Software Accomplishment Summary (planning stage);

l. Software Life Cycle Environment Index;

Hardware layer:

m. Hardware Development Plan;

n. Hardware Qualification Plan;

o. Hardware Configuration Management Plan;

p. Hardware Quality Assurance Plan.

4.14.1.1.3 Development process metrics

Benchmarks for the execution of tasks associated with all the listed activities need to be determined and methods for measuring performance against these benchmarks is to be established.

4.14.1.1.4 Close out report

A procedure for the preparation of a close-out report is to be produced. This procedure should incorporate the required pro forma and a checklist to ensure that all aspects were addressed.

4.14.1.2 Change control

Refer to activities 1-1-2; 1-2-2; 1-3-2; 1-4-2; 1-5-2.

A change control process shall be described to address the particular aspects related to life cycle data associated with the development of airborne electronic equipment. Specific procedures required include:

a. Change process and baseline management;

b. Configuration control board constitution and meetings;

c. Preparation, control and closure of Request for Change (RFC);

d. Preparation, control and closure of Engineering Change Proposal (ECP);

e. Master Record Index (MRI) procedure;

f. Appointment of Technical Authority (system and sub-system layers).

4.14.1.3 Technical pre-study

Refer to activity 1-1-3

A procedure for the preparation of a technical pre-study report is to be produced. This procedure should incorporate the required pro forma and a checklist to ensure that all aspects were addressed.

4.14.1.4 Formal reviews

Refer to activities 1-1-4, 1-1-5, 1-3-4, 1-3-5, 1-4-4, 1-4-5.

Procedures, including checklists, are required for the preparation and conducting of formal reviews.

4.14.1.5 Problem reporting and corrective action system (PRACAS)

Refer to activities 1-3-3; 1-4-3; 1-5-3; 1-6-3.

A process for the formal recording of problems and the management of corrective action is to be defined. This process should make provision for the handling of problems identified after establishment of the product baseline and to provide input to the formal change process.

A process for handling problems identified prior to establishment of the formal baseline, e.g. during integration, allowing changes to design life cycle data at a lower level of decision making than which is required for changes to baselined designs, shall also be defined.

4.14.1.6 Formal test preparation

Refer to activity 1-3-6.

A checklist should be developed to support this activity.

4.14.1.7 Delivery preparation

Refer to activity 1-4-6.

A checklist should be developed to support this activity.

4.14.2 Requirements thread design guidelines

4.14.2.1 Requirements management

Requirements management refers to activities 2-1-1, 2-2-1, 2-3-1, 2-4-1, 2-5-1, 2-6-1.

Modern requirements management processes use database oriented systems for requirements management, e.g. DOORS™.

A generic methodology for recording and identifying requirements, managing traceability and tracking requirements implementation, verification and validation is to be provided in the quality system. This methodology should be selected from the significant number of practises described in the literature.

Procedures for the preparation of system and software requirements specifications are to be produced. These procedures should incorporate the required pro formas and checklists to ensure that all aspects were addressed.

See also 4.14.1.1.1, a, c and f (requirements standards).

4.14.2.2 Concept definition

Refer to activity 2-1-2.

A procedure for the preparation of a technical pre-study report is to be produced. This procedure should incorporate the required pro forma and a checklist to ensure that all aspects were addressed.

4.14.2.3 Human-machine interface (HMI) definition

Refer to activity 2-2-2.

Human-machine interface definitions differ from project to project. No specific methodology for the preparing of HMI definition documents is proposed.

4.14.3 System safety/airworthiness/certification thread design guidelines

4.14.3.1 Functional hazard assessment (FHA)

Refer to activity 3-1-1.

a. A procedure for the preparation of a FHA report is to be produced. This procedure should incorporate the required pro forma and a checklist to ensure that all aspects were addressed.

b. A system for the recording and tracking of hazards (hazard list) should be defined, and a procedure for the management of this system should be prepared.

4.14.3.2 Hazard tracking

Refer to activity 3-2-1.

The system mentioned in 4.14.3.2 b shall be used to record changes to the hazard characterisation of the system, which came about as a consequence of the requirements analysis process.

4.14.3.3 Preliminary system safety assessment (PSSA)

Refer to activity 3-2-2.

A procedure for the preparation of a PSSA report is to be produced. This procedure should incorporate the required pro forma and a checklist to ensure that all aspects were addressed.

4.14.3.4 Flight test airworthiness preparation

Refer to activity 3-3-1.

a. A procedure for the preparation of a Declaration of Design and Performance for use by the Denel Aviation flight safety authorities is to be produced. This procedure should

incorporate the required pro forma and a checklist to ensure that all aspects were addressed;

b. It is required to have a single document that contains references to all the aspects required to ensure that flight testing can be conducted safely. Denel Aviation used to use a Certificate for Flight Trials (CFT) for this purpose. A procedure for the preparation of a CPT is to be produced. This procedure should incorporate the required pro forma and a checklist to ensure that all aspects were addressed.

4.14.3.5 Operational airworthiness preparation

Refer to activity 3-4-1.

A procedure for the preparation of a Declaration of Design and Performance for issuing to the certification authorities is to be produced. This procedure should incorporate the required pro forma and a checklist to ensure that all aspects were addressed.

4.14.3.6 Airworthiness approval

Refer to activity 3-5-1.

The process describing all pre-requisites for airworthiness approval should be maintained in the quality system.

4.14.3.7 System safety assessment (SSA)

Refer to activity 3-5-2.

A procedure for the preparation of a SSA report is to be produced. This procedure should incorporate the required pro forma and a checklist to ensure that all aspects were addressed.

4.14.3.8 Continued airworthiness

Refer to activity 3-6-1.

a. A policy is to be defined, outlining duties, roles and responsibilities with respect to continued airworthiness of an airborne electronic system produced by Denel Aviation;

b. Procedures are required for the preparation of Incident Investigation Reports;

c. Procedures are required for the preparation of Failure Investigation Reports;

d. Procedures are required for the preparation of Service Bulletins.

4.14.4 End product realisation thread design guidelines

4.14.4.1 Development preparation

Refer to activity 4-1-1.

A library of templates for documentation required for capturing of design information need to be set up.

4.14.4.2 Preliminary design

Refer to activity 4-2-1.

The work performed within the definition of this activity is dependent on the specific project. Detailed requirements with respect to the outcomes of this activity are to be captured in the development plans and standards as described in section 4.14.1.1.

4.14.4.3 Detailed design

Refer to activity 4-3-1.

As for 4.14.4.3.

4.14.4.4 Design implementation (SW layer)

Refer to activity 4-3-2.

As for 4.14.4.3.

4.14.4.5 Hardware procurement (HW layer)

Refer to activity 4-3-3.

The procurement of project related assets need to be formulated in a policy and captured in the quality system.

4.14.4.6 Integration

Refer to activity 4-3-4.

As for 4.14.4.3.

4.14.4.7 Informal tests

Refer to activity 4-3-5.

Informal testing is not subjected to quality assurance audits. However, it shall be ensured that no unauthorised changes are made to configured life cycle data or test objects when used for informal testing.

4.14.4.8 Preparation of support documentation

Refer to activity 4-4-1.

The outsourcing and acceptance of supporting documentation should be addressed in the development plans as described in section 4.14.1.1.

4.14.4.9 Design increment closure

Refer to activity 4-4-2.

A generic checklist for use to ensure that all the life cycle data for a design increment is complete is to be prepared and captured in the quality system.

4.14.5 Verification thread design guidelines

4.14.5.1 Verification preparation

Refer to activity 5-1-1.

A checklist should be developed to support this activity.

4.14.5.2 Verification planning

Refer to activity 5-2-1.

A checklist should be developed to support this activity.

4.14.5.3 Verification specification

Refer to activity 5-3-1.

Pro formas and checklists need to be developed for:

a. Test specifications

b. Flight test request

4.14.5.4 Requirements and design verification tests

Refer to activity 5-4-1.

Pro formas and checklists need to be developed for test reports.

4.14.5.5 Analyses

Refer to activity 5-4-2.

Analysis results should be captured in engineering reports.

A procedure for the preparation and handling of engineering reports is to be created and captured in the quality system.

4.14.5.6 Inspections

Refer to activity 5-4-3.

A procedure for the conducting of inspections and reporting of results is to be created and captured in the quality system.

4.14.5.7 Production acceptance preparation

Refer to activity 5-5-1.

A checklist should be developed to support this activity.

4.14.5.8 OT&E

Refer to activity 5-5-2.

A checklist should be developed to support this activity.

4.14.6 Infrastructure thread design guidelines

4.14.6.1 Infrastructure preparation

Refer to activity 6-1-1.

A checklist should be developed to support this activity.

4.14.6.2 Infrastructure management

Refer to activity 6-2-1; 6-3-1; 6-4-1.

A checklist should be developed to support this activity.

4.15 Derived Statement of Work (SOW)

The process model described in this work was reviewed and authenticated (accepted) by Denel and SAAB quality assurance and was subsequently used to derive a Statement of Work (SOW) for the upgrade of the Oryx helicopter fleet of the South African air force. This SOW is attached to this document in Appendix C for the sake of completeness.

4.16 Summary

A process model to be used by Denel Aviation, for the development of airborne electronic equipment, was presented in this chapter. The following are important salient aspects of the proposed model:

- The process facilitates incremental and iterative development;
- A requirements-based development methodology is used;
- Verification of requirement implementation commences in the definition stages of the project;
- The process supports a layered hierarchy of development;
- A parallel breadboarding process is used to validate requirements, test implementation strategies and trade-offs and demonstrate concepts to stakeholders.

The organisational context of the model was outlined, project stages and decision gates were identified and the development threads and associated activities were described.

A description of each identified activity, with associated outcomes, was presented.

Requirements for detailed methods and procedures associated with theses activities were identified.

The allocation of process characteristics defined in this chapter to the requirements derived in chapter 3 is presented schematically in Appendix B.

The process model was approved for use by Denel and Saab. A statement of work was prepared as a result of this work and is available in Appendix C.

Chapter 5: Conclusion and recommendations

5.1 Conclusion

In this work, shortcomings of the processes used by Denel Aviation for the development of airborne electronic equipment were identified. Existing standards (normative) and other requirements were analyzed and 74 requirements for a new process model were identified and documented. These requirements were allocated to the relevant standards for the purpose of traceability. Note that this list of requirements is not exhaustive or static, and new requirements shall be added as and when required. Again, the process of requirements management (change control) shall be followed when these high-level requirements are subject to change.

A set of necessary and sufficient engineering activities required for the realisation of an airborne electronic system was identified and contextualized within the scope of work of a development program. A framework for the logical organisation of these activities was developed. This framework follows a requirements-based development paradigm, as advocated by the consulted normative standards. Detailed methods and procedures, required to perform the tasks associated with each activity, were identified and allocated to the list of 74 requirements. This model allows for incremental development and verification of system functionality and the same process framework can be applied to the development of an end product, enabling product and sub-systems. A “breadboarding” process, executed in parallel with the development of the end product, to support technical decision making and minimize risk, is encouraged.

The process model in this work can form the baseline for the definition of engineering policies and procedures in the Denel Aviation Quality Management System and can be refined and extended by using feedback from projects where the model is implemented.

5.2 Recommendations

The following recommendations are made, namely: (i) the framework should be used to organise the airborne electronic systems development section of the Denel Aviation Quality System; (ii) procedures and work methods, as proposed herein, should be updated or developed in the quality system where applicable; (iii) this document should be updated regularly to refine the process requirements and to record additional requirements and process design aspects in order for it to be used as a management tool for the control of the improvement of the quality system; (iv) the implementation of the process described in this document in a practical development program should be evaluated in order to validate the AESDP process model.

References

1. Digital Avionics Systems Principles and Practise 2nd edition Gary R Spitzer Blackburn Press.
2. Large Scale Engineering and Evolutionary Change : Useful Concepts for Implementation of FORCEnet. Yaneer Bar-Yam, New England Complex Systems Institute, September 2, 2002.
3. Guideline for identification of process components CMMI-SE/SW, v1.1 Continuous Representation.
4. SAE AS9100 (Revision B): Quality Management Systems – Aerospace – Requirements Revision B Revised 2004-01.
5. MIL-STD-498: Software development and documentation (Cancelled).
6. ANSI/EIA-632-1999: Processes for Engineering a System. (Published by Government Electronics and Information Technology Association (GEIA); January 1999).
7. IEEE/EIA 12207.0, Standard for Information Technology-Software Life Cycle Processes.
8. IEEE 1220–2005: IEEE Standard for Application and Management of the Systems Engineering Process.
9. ISO/IEC 15288:2002 : Systems and software engineering – System life cycle processes (superseded by ISO/IEC 15288:2008).
10. RTCA/DO-178B: Software Considerations in Airborne Systems and Equipment Certification, December 1, 1992.
11. Blanchard, Fabrycky: Systems Engineering and Analysis (Fourth Edition) Prentice Hall.
12. Systems Engineering Handbook – A Guide for System Life Cycle Processes and Activities INCOSE June 2006.
13. MIL-STD-490A: Specification Practises (Cancelled).
14. Mark 33 Digital Information Transfer System (DITS) ARINC SPECIFICATION 429-14: March 10, 1993.
15. MIL-STD-1553B: Digital Time Division Command/Response Multiplex Data Bus.
16. ANSI/TIA/EIA-422-B. Electrical Characteristics of Balanced Voltage Digital Interface Circuits.
17. FAR Part 21: Certification procedures for products and parts.
18. SAE ARP 4754: Certification Considerations for Highly – Integrated or Complex Aircraft Systems.

19. RTCA/DO-160E: Environmental Conditions and. Test Procedures for Airborne Equipment, December 9, 2004.
20. FAR Part 25: Airworthiness standards: Transport category airplanes.
21. FAR Part 29: Airworthiness standards: Transport category rotorcraft.
22. RMSS Document; RSA-MIL-STD-182: Systems Engineering Management Plan (SEMP), Preparation of.
23. MIL-STD-1521B: Technical Reviews and Audits for Systems, Equipments, and Computer Software.
24. RMSS Document; Document Issue 4, RSA-MIL-STD 3.
25. http//af.wikipedia.org/wiki/Requirement: accessed 17/07/2008.
26. SAE ARP4761: Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems And Equipment.
27. Fagan, M.E., Design and Code inspections to reduce errors in program development, 1976, IBM Systems Journal, Vol. 15, No 3, Page 182-211.
28. MIL-STD-499A: Engineering Management / Systems Engineering (Cancelled).
29. ISO/IEC15504-1;2 ;3 ;4: 2004: Information technology – process assessment. First edition 2004.
30. SAE AS9101: Quality Management System Assessment.
31. RMSS Document; RSA-MIL-STD-257: Test And Evaluation Master Plan (TEMP), Preparation of.
32. Larman, Basili, Computer, June 2003 p47 to 56: Iterative and Incremental Development: A Brief History.
33. RTCA/DO-254: Design Assurance Guidance for Airborne Electronic Hardware, April 19, 2000.

Appendix A: Allocation of high-level requirements

Normative input requirements are listed in the rows and derived requirements are listed in the columns. An “x” in the matrix denotes applicability.

	1.01	1.02	1.03	1.04	1.05	1.06	1.07	1.08	1.09	1.10	1.11	1.12	1.13	1.14	1.15	1.16	1.17	2.01	2.02	3.01	3.02	3.03	3.04	3.05
Other references																								
Denel / Saab	x	x	x		x		x			x							x							
System characteristics	x	x		x		x				x	x					x								
Normative references																								
AS9100		x						x	x					x								x	x	x
EIA-632		x						x	x					x						x	x	x	x	x
ISO/IEC 12207		x						x	x					x	x					x	x	x	x	x
IEEE 1220		x						x	x					x	x					x	x	x	x	x
ISO/IEC 15288		x						x	x					x	x					x	x	x	x	x
RTCA/DO-178B		x										x	x	x				x	x				x	x
RTCA/DO-254		x										x	x	x				x	x				x	x
RSA-MIL-STD-182																								
MIL-STD-1521B																						x		
RSA-MIL-STD 3																							x	x
SAE ARP4754		x						x	x			x	x	x				x	x					
SAE ARP4761												x												

	4.01	4.02	4.03	4.04	4.05	4.06	4.07	4.08	4.09	4.10	4.11	4.12	4.13	4.14	4.15	4.16	4.17	4.18	4.19	4.20	4.21	4.22	4.23	4.24
Other references																								
Denel / Saab																								
System characteristics																								
Normative references																								
AS9100		x	x	x	x		x	x	x												x	x		
EIA-632	x	x	x	x	x	x	x		x	x	x	x	x	x	x				x			x	x	x
ISO/IEC 12207	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x							x		
IEEE 1220	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x							x	x	
ISO/IEC 15288	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x							x	x	
RTCA/DO-178B	x	x	x	x	x	x			x	x	x	x	x	x	x			x		x	x	x		
RSA-MIL-STD-182	x	x	x	x																				
MIL-STD-1521B	x	x	x	x																				
RSA-MIL-STD 3	x	x	x	x																				
SAE ARP4754	x	x	x	x	x									x	x	x	x	x	x	x	x			
SAE ARP4761	x															x	x	x	x					

	4.25	4.26	4.27	4.28	4.29	4.30	4.31	4.32	4.33	4.34	4.35	4.36	4.37	4.38	4.39	4.40	4.41	4.42	4.43	4.44	4.45	4.46	4.47	4.48	4.49	5.01
Other references																										
Denel / Saab										x	x	x		x		x	x	x	x	x			x	x	x	
System characteristics																x		x								
Normative references																										
AS9100						x	x														x	x				x
EIA-632	x	x	x	x	x	x	x	x	x												x	x				
ISO/IEC 12207		x		x	x	x	x	x	x												x	x				
IEEE 1220			x	x	x	x	x	x	x												x	x				
ISO/IEC 15288			x	x	x	x	x	x	x												x	x				
RTCA/DO-178B				x	x	x	x	x	x	x	x	x	x		x						x	x				
RSA-MIL-STD-182																										
MIL-STD-1521B																										
RSA-MIL-STD 3							x																			
SAE ARP4754						x	x						x								x	x				x
SAE ARP4761																										

Appendix B: Allocation of process characteristics

All high-level requirements are listed in the rows and process characteristics are listed in the columns. An “x” in the matrix denotes applicability.

	1.01	1.02	1.03	1.04	1.05	1.06	1.07	1.08	1.09	1.10	1.11	1.12	1.13	1.14	1.15	1.16	1.17	2.01	2.02	3.01
P. Char: Section 4.2									x											x
P. Char: Section 4.3	x																			
P. Char: Section 4.4																				
P. Char: Section 4.5																				
P. Char: Section 4.6			x																	
P. Char: Section 4.13																				
P. Char: Activity 1-1-1																x				
P. Char: Activity 1-1-2																				
P. Char: Activity 1-1-3							x													
P. Char: Activity 1-1-4																				
P. Char: Activity 1-1-5																				
P. Char: Activity 1-2-1																				
P. Char: Activity 1-2-2																				
P. Char: Activity 1-2-3																				
P. Char: Activity 1-3-1																				
P. Char: Activity 1-3-2																				
P. Char: Activity 1-3-3																				
P. Char: Activity 1-3-4																				
P. Char: Activity 1-3-5																				
P. Char: Activity 1-3-6																				
P. Char: Activity 1-4-1																				
P. Char: Activity 1-4-2																				
P. Char: Activity 1-4-3																				
P. Char: Activity 1-4-4																				
P. Char: Activity 1-4-5																				
P. Char: Activity 1-4-6																				
P. Char: Activity 1-4-7																				
P. Char: Activity 1-4-8																				
P. Char: Activity 1-5-1																				
P. Char: Activity 1-5-2																				
P. Char: Activity 1-5-3																				
P. Char: Activity 1-6-1																				
P. Char: Activity 1-6-2																				
P. Char: Activity 1-6-3								x												

	3.02	3.03	3.04	3.05	4.01	4.02	4.03	4.04	4.05	4.06	4.07	4.08	4.09	4.10	4.11	4.12	4.13	4.14	4.15	4.16
P. Char: Section 4.2	x																			
P. Char: Section 4.3			x																	
P. Char: Section 4.4		x																		
P. Char: Section 4.5																				
P. Char: Section 4.6				x																
P. Char: Section 4.13																				
P. Char: Activity 1-1-1					x															
P. Char: Activity 1-1-2							x	x		x		x								
P. Char: Activity 1-1-3																				
P. Char: Activity 1-1-4						x														
P. Char: Activity 1-1-5						x														
P. Char: Activity 1-2-1																				
P. Char: Activity 1-2-2							x	x												
P. Char: Activity 1-2-3						x														
P. Char: Activity 1-3-1																				
P. Char: Activity 1-3-2							x	x												
P. Char: Activity 1-3-3									x											
P. Char: Activity 1-3-4						x														
P. Char: Activity 1-3-5						x														
P. Char: Activity 1-3-6																				
P. Char: Activity 1-4-1																				
P. Char: Activity 1-4-2							x	x												
P. Char: Activity 1-4-3									x											
P. Char: Activity 1-4-4																				
P. Char: Activity 1-4-5																				
P. Char: Activity 1-4-6																				
P. Char: Activity 1-4-7																				
P. Char: Activity 1-4-8																				
P. Char: Activity 1-5-1																				
P. Char: Activity 1-5-2							x	x												
P. Char: Activity 1-5-3									x											
P. Char: Activity 1-6-1											x									
P. Char: Activity 1-6-2							x	x												
P. Char: Activity 1-6-3									x											

	4.17	4.18	4.19	4.20	4.21	4.22	4.23	4.24	4.25	4.26	4.27	4.28	4.29	4.30	4.31	4.32	4.33	4.34	4.35	4.36
P. Char: Section 4.2																				
P. Char: Section 4.3																				
P. Char: Section 4.4																				
P. Char: Section 4.5																				
P. Char: Section 4.6																				
P. Char: Section 4.13																				
P. Char: Activity 1-1-1							x													
P. Char: Activity 1-1-2																				
P. Char: Activity 1-1-3																				
P. Char: Activity 1-1-4																				
P. Char: Activity 1-1-5																				
P. Char: Activity 1-2-1																				
P. Char: Activity 1-2-2																				
P. Char: Activity 1-2-3																				
P. Char: Activity 1-3-1																				
P. Char: Activity 1-3-2																				
P. Char: Activity 1-3-3																				
P. Char: Activity 1-3-4																				
P. Char: Activity 1-3-5																				
P. Char: Activity 1-3-6																				
P. Char: Activity 1-4-1																				
P. Char: Activity 1-4-2																				
P. Char: Activity 1-4-3																				
P. Char: Activity 1-4-4																				
P. Char: Activity 1-4-5																				
P. Char: Activity 1-4-6																				
P. Char: Activity 1-4-7																				
P. Char: Activity 1-4-8							x													
P. Char: Activity 1-5-1																				
P. Char: Activity 1-5-2																				
P. Char: Activity 1-5-3																				
P. Char: Activity 1-6-1																				
P. Char: Activity 1-6-2																				
P. Char: Activity 1-6-3																				

	4.37	4.38	4.39	4.40	4.41	4.42	4.43	4.44	4.45	4.46	4.47	4.48	4.49	5.01
P. Char: Section 4.2														
P. Char: Section 4.3														
P. Char: Section 4.4														
P. Char: Section 4.5														
P. Char: Section 4.6														
P. Char: Section 4.13											x	x	x	
P. Char: Activity 1-1-1		x												
P. Char: Activity 1-1-2														
P. Char: Activity 1-1-3														
P. Char: Activity 1-1-4														
P. Char: Activity 1-1-5														
P. Char: Activity 1-2-1														
P. Char: Activity 1-2-2														
P. Char: Activity 1-2-3														
P. Char: Activity 1-3-1														
P. Char: Activity 1-3-2														
P. Char: Activity 1-3-3														
P. Char: Activity 1-3-4														
P. Char: Activity 1-3-5														
P. Char: Activity 1-3-6														
P. Char: Activity 1-4-1														
P. Char: Activity 1-4-2														
P. Char: Activity 1-4-3														
P. Char: Activity 1-4-4														
P. Char: Activity 1-4-5														
P. Char: Activity 1-4-6														
P. Char: Activity 1-4-7														
P. Char: Activity 1-4-8														
P. Char: Activity 1-5-1														
P. Char: Activity 1-5-2														
P. Char: Activity 1-5-3														
P. Char: Activity 1-6-1														x
P. Char: Activity 1-6-2														
P. Char: Activity 1-6-3														

Appendix C: Statement of work

Navigation System Obsolescence Mitigation

WBS 210 – Concept and Planning Stage

WBS 211 – Product Layer Development Planning

Inputs

a. SAAF User Requirements
b. Oryx Design Specification (Reference 330S05-2030 and 332A05-0008)
c. DEF STAN 00-0970 Design and Airworthiness Requirements for Service Aircraft VOL 2 - Rotorcraft
d. Aviation Legislation in South Africa: South African Civil Aviation d. Technical Standards Volume 3
e. Applicable Industry Standards
f. Systems engineering policies and procedures as per the Company Quality System
g. Minutes of discussion with SAAF, Armscor and Civil Aviation technical authorities

Work Description

a. Define the working methods and development environment for the development of the navigation system obsolescence mitigation program on the product layer
b. Define the detailed configuration management process to be followed by the project
c. Identify the quality assurance activities for the project
d. Define and authorise the development team roles and responsibilities
e. Determine the detailed system qualification and certification requirements and obtain formal approval
f. Determine and refine the development standards
g. Define the modelling, simulation and prototyping environment and methods
h. Define the system safety process

Outcomes

a. System Development and Verification Plan
b. Role Assignment Document (Product layer)
c. System Configuration Management Plan
d. System Quality Assurance Plan
e. Requirements Standard
f. Design Standard
g. System Modelling and Prototyping Plan
h. System Safety Plan

Deliverables

a. System Development and Verification Plan
b. System Configuration Management Plan
c. System Quality Assurance Plan
d. System Safety Plan

WBS 212 – Concept Definition and Requirements Capture

Inputs

a. SAAF User Requirements
b. Systems engineering policies and procedures as per the Company Quality System
c. Technical data from potential suppliers
d. Obsolescence Study Report
e. ROTERS reports

Work Description

a. Develop conceptual definition and capture requirements
b. Capture initial requirements
c. Determine stakeholders

Outcomes

a. Operational Concept Definition
b. Requirements lists
c. Stakeholders lists

Deliverables

None of the outcomes from this activity are intended to be formal project deliverables; however, they form authoritative inputs to the downstream activities.

WBS 213 – Functional Hazard Assessment (FHA)

Inputs

a. Operational Concept Definition
b. Requirements Lists
c. System Safety Plan
d. Company procedure for performing a FHA.

Work Description

a. Develop Hazard List (i.e. list of unwanted events)
b. Perform Functional Hazard Analysis
c. Determine software criticality level
d. Conduct FHA workgroup meetings
d. Perform formal inspections of relevant output documents
e. Perform engineering studies as required for system verification purposes

Outcomes

a. Hazard List
b. Functional Hazard Assessment Report
c. Engineering report (software criticality assessment)

Deliverables

a. Functional Hazard Assessment Report
b. Engineering report (software criticality assessment)

WBS 214 – Concept Evaluation Review

Inputs

a. Functional Hazard Assessment Report
b. Operational Concept Definition
c. Requirements lists
d. Stakeholders lists
e. Company procedure for conducting formal reviews

Work Description

a. Plan, co-ordinate and execute the Concept Evaluation Review

Outcomes

a. Minutes of the Concept Evaluation Review

Deliverables

a. Approved minutes of the Concept Evaluation Review

WBS 215 – Planning Review

Inputs

a. Development plans
b. Planning review agenda
c. Company procedure for conducting formal reviews

Work Description

a. Present content of development plans to stakeholders.
b. Capture review comments in review minutes.

Outcomes

a. Planning review minutes

Deliverables

a. Planning review minutes

WBS 220 – Definition Stage
(Requirements Analysis and Verification Planning)

Note: the work specified in this and consequent WBS descriptions are to be performed for each development increment.

WBS 221 - System Requirements Analysis

Inputs

a. System Development and Verification Plan
b. System Configuration Management Plan
c. System Quality Assurance Plan
d. SAAF User Requirements
e. Oryx Obsolescence Study Report
f. Operational Concept Definition
g. Requirements lists
h. Stakeholders lists
i. Systems engineering policies and procedures as per the Company Quality System

Work Description

a. Analyse and refine the requirements for the navigation system obsolescence mitigation program (including RAMS analysis) by means of studies and reviews/discussions with SAAF and Armscor relevant authorities
b. Conduct navigation system requirements refinement and design workgroups
c. Develop the System Specification
d. Perform formal inspections of relevant output documents
e. Perform engineering studies as required for system verification purposes

Outcomes

a. System Requirement Specification
b. Minutes of workgroup meetings
c. Inspection records
d. Engineering reports

Deliverables

a. System Requirement Specification

WBS 222 – Human - Machine Interface (HMI) Analysis

Inputs

a. System Development and Verification Plan
b. System Configuration Management Plan
c. System Quality Assurance Plan
d. SAAF User Requirements
e. Oryx Obsolescence Study Report
f. Operational Concept Definition
g. Requirements lists
h. Stakeholders lists
i. Systems engineering policies and procedures as per the Company Quality System

Work Description

a. Analyse and refine the requirements for the navigation system HMI by means of studies and reviews/discussions with SAAF and Armscor relevant authorities
b. Conduct navigation system HMI requirements refinement and design workgroups
c. Develop the HMI Definition (Navigation System Sections)
d. Perform formal inspections of relevant output documents
e. Perform engineering studies as required for system verification purposes

Outcomes

a. System Requirement Specification
b. HMI Definition (Navigation System Sections)
c. Minutes of workgroup meetings
d. Inspection records
e. Engineering reports

Deliverables

a. HMI Definition (Navigation System Sections)

WBS 223 - System Verification Analysis

Inputs

a. System Development and Verification Plan
b. System Configuration Management Plan
c. System Quality Assurance Plan
d. System Requirement Specification
e. HMI Definition (Navigation System Sections)
f. Systems engineering policies and procedures as per the Company Quality System
g. Contracted Verification Matrix

Work Description

a. Determine and document the appropriate method for demonstrating compliance w.r.t each requirement stated in the System Specification
b. Perform formal inspections of relevant output documents
c. Perform engineering studies as required for system verification purposes

Outcomes

a. System Requirements Verification Specification
b. Inspection records
c. Engineering reports

Deliverables

a. System Requirements Verification Specification

WBS 224 – Product Layer Architecture Design

Inputs

a. System Development and Verification Plan
b. System Configuration Management Plan
c. System Quality Assurance Plan
d. System Specification
e. HMI Definition (Navigation System Sections)
f. Systems engineering policies and procedures as per the Company Quality System
g. Existing subsystems design and technical documentation (specifications, ICDs)

Work Description

a. Perform and document the product layer architectural design of the system, identifying all major components and interfaces
b. Identify all hardware to form part of the system
c. Perform formal inspections of relevant output documents
d. Perform engineering studies as required for system verification purposes

Outcomes

a. System architecture diagram, including the product breakdown structure, captured in a preliminary issue of the System Design Description
b. System Interface Requirements Description
c. Inspection records
d. Engineering reports

Deliverables

a. System Design Description (preliminary issue)
b. System Interface Requirements Description

WBS 226 – Establish Product Layer PRACAS

Inputs

a. System Quality Assurance Plan

Work Description

a. Establish a failure reporting and corrective action system for the upgraded system
b. Perform formal inspections of relevant output documents
c. Perform engineering studies as required for system verification purposes, e.g. test coverage analysis

Outcomes

a. PRACAS definition

Deliverables

a. PRACAS reports

WBS 227 – Preliminary Safety Assessment (PSA)

Inputs

a. System Development and Verification Plan
b. System Configuration Management Plan
c. System Quality Assurance Plan
d. System Requirement Specification
e. HMI Definition (Navigation System Sections)
f. System Description (Architecture description)
g. FMEA report
h. FHA report
i. Systems engineering policies and procedures as per the Company Quality System

Work Description

a. Perform Preliminary Safety Assessment (PSA) study
b. Conduct PSA reviews as per the System Development and Verification Plan
c. Ensure that the findings of the PSA are communicated to all stakeholders and that the failure effect mitigation is addressed in downstream engineering activities
d. Perform formal inspections of relevant output documents
e. Perform engineering studies as required for system verification purposes

Outcomes

a. Preliminary Safety Assessment Report
b. Minutes of PSA reviews
c. Inspection records
d. Engineering reports

Deliverables

a. Preliminary Safety Assessment Report

WBS 228 - Preliminary Design Review

Inputs

a. System Development and Verification Plan
b. System Configuration Management Plan
c. System Quality Assurance Plan
d. Oryx User Requirement
e. System Requirement Specification
f. System Design Description (preliminary issue, describing architecture)
g. System Interface Requirements Description
h. HMI Definition (Navigation System Sections)
i. Logistics Requirements Specification
j. Maintenance Policy
k. FMEA report
l. MTBF Breakdown report
m. Preliminary Maintenance Level Definition
n. Maintenance Safety Assessment
o. Preliminary Maintenance Manual
p. Preliminary Aircrew Manual
q. Preliminary IPC
r. PSA Report
s. Company procedure for conducting formal reviews

Work Description

a. Plan, co-ordinate and execute the Preliminary Design Review

Outcomes

a. Minutes of the Preliminary Design Review
b. Functional Baseline

Deliverables

a. Approved minutes of the Preliminary Design Review
b. Functional Baseline MRI

WBS 230 – Design and Development Stage (Requirements Implementation and Verification)

WBS 231 – System Detailed Design

Inputs

a. System Development and Verification Plan
b. System Configuration Management Plan
c. System Quality Assurance Plan
d. System Specification
e. HMI Definition (Navigation System Sections)
f. Systems engineering policies and procedures as per the Company Quality System
g. Existing subsystems design and technical documentation (specifications, ICDs)

Work Description

a. Perform the product layer detailed design of the Navigation System.
b. Analyse the product layer functions and allocate functions to hardware and software
c. Perform formal inspections of relevant output documents
d. Perform engineering studies as required for system verification purposes

Outcomes

a. System Design Description
b. System Interface Design Description
c. Inspection records
d. Engineering reports

Deliverables

a. System Design Description
b. System Interface Design Description

WBS 232 – System Verification Specification

Inputs

a. System Development and Verification Plan
b. System Configuration Management Plan
c. System Quality Assurance Plan
d. System Specification
e. HMI Definition (Navigation System Sections)
f. Systems engineering policies and procedures as per the Company Quality System
g. System Design Description
h. System Interface Design Description
i. System Requirement Verification Specification

Work Description

a. Specify the suite of test cases required to test the implemented design
b. Specify the qualification flight test requirements
c. Perform formal inspections of relevant output documents
d. Perform engineering studies as required for system verification purposes, e.g. test coverage analysis

Outcomes

a. System Test Description
b. System Verification Flight Test Request
c. Inspection records
d. Engineering reports

Deliverables

a. System Test Description
b. System Verification Flight Test Request

WBS 234 - Critical Design Review

Inputs

a. System Development and Verification Plan
b. System Configuration Management Plan
c. System Quality Assurance Plan
d. System Design Description
e. System Interface Design Description
f. System Test Description
g. System Flight Test Request
h. MSG3 Report
i. Parts lists
j. Ground Support and Test Equipment Requirement Specification
k. Scheduled Maintenance Specification
l. Logistics Support Analysis Report (LSAR)
m. Training Plan
n. Company procedure for conducting formal reviews

Work Description

a. Plan, co-ordinate and execute the Critical Design Review

Outcomes

a. Minutes of the Critical Design Review
b. Allocated Baseline

Deliverables

a. Approved minutes of the Critical Design Review
b. Allocated Baseline MRI

WBS 235 – Informal Integration

Inputs

a. System Development and Verification Plan
b. System Configuration Management Plan
c. System Quality Assurance Plan
d. System Test Description
e. System Interface Design Description

Work Description

a. Perform product layer integration tasks on the HeliLab simulator
b. Support aircraft layer integration on the prototype aircraft
c. Perform "dry runs" of system tests to ensure that the test documentation is correct; the test environment is sufficient and the system under test is mature. This is to be performed in the HeliLab simulator, the prototype aircraft using ground power, and on the prototype aircraft using aircraft power.
d. Use the insular debugging process to record problems for feedback to. upstream development activities

Outcomes

a. System Test Description (final issue for this development increment)
b. System Test Report (not for issuing)
c. Data required for CFTs

Deliverables

a. No formal project deliveries

WBS 238 – Formal Test Baseline Preparation

Inputs

a. System Development and Verification Plan
b. System Configuration Management Plan
c. System Quality Assurance Plan
d. System Requirement Specification
e. System Interface Requirements Description
f. HMI Definition (Navigation System Sections)
g. Preliminary Safety Assessment Report
h. System Verification specification
i. System Design Description
j. System Interface Design Description
k. System Test Description
l. System Verification Flight Test Request
m. All verification evidence collated during the development increment

Work Description

a. Ensure completeness of life cycle data
b. Ensure all configuration identification requirements are met

Outcomes

a. System configuration management records
b. Quality assurance records

Deliverables

Nil

WBS 237 - Test Readiness Review

Inputs

a. System Development and Verification Plan
b. System Configuration Management Plan
c. System Quality Assurance Plan
d. System Requirements Verification Specification
e. System Test Description (final issue for this development increment)
f. Flight Test Request
g. Flight Test Plan
h. System Test Report (preliminary issue)
i. Company procedure for conducting formal reviews

Work Description

a. Plan, co-ordinate and execute the Test Readiness Review.

Outcomes

a. Minutes of the Test Readiness Review.
b. Formal Test Baseline

Deliverables

a. Approved minutes of the Test Readiness Review.

WBS 250 – Delivery and Validation

WBS 251 – Formal System Integration Tests

Inputs

a. System Development and Verification Plan
b. System Configuration Management Plan
c. System Quality Assurance Plan
d. System Test Description
e. System Interface Design Description

Work Description

a. Perform formal product layer laboratory tests

Outcomes

a. Navigation System Test Report

Deliverables

a. Navigation System Test Report

WBS 252 – System Safety Assessment

Inputs

a. System Development and Verification Plan
b. System Configuration Management Plan
c. System Quality Assurance Plan
d. System Design Description
e. System Test Description
f. HMI Definition (Navigation System Sections)
g. PSA Report
h. Systems engineering policies and procedures as per the Company Quality System

Work Description

a. Perform System Safety Assessment (SSA) study
b. Conduct SSA reviews as per the System Development and Verification Plan
c. Perform formal inspections of relevant output documents
d. Perform engineering studies as required for system verification purposes

Outcomes

a. System Safety Assessment Report
b. Minutes of SSA reviews
c. Inspection records
d. Engineering reports

Deliverables

a. System Safety Assessment Report

WBS 253 – Formal Test Verification

Inputs

a. System Development and Verification Plan
b. System Configuration Management Plan
c. System Quality Assurance Plan
d. System Test Report
e. System data pack from formal test preparation activity
f. DDPs or equivalent documentation from suppliers

Work Description

a. Ensure completeness of life cycle data
b. Ensure all configuration identification requirements are met
c. Prepare the System Declaration of Design and Performance (DDP) as an input to the aircraft layer flight safety documentation
d. Review the DDP

Outcomes

a. Configuration management records
b. Quality assurance records
c. System Declaration of Design and Performance (DDP)
d. Verification Report

Deliverables

a. System Declaration of Design and Performance (DDP)
b. Verification Report

WBS 254 – Product Layer Flight Test Support

Inputs

a. System Development and Verification Plan
b. System data pack
c. System Verification Flight Test Request
d. Flight Test Plan

Work Description

a. Assist in preparation of CFT
b. Participate in flight test planning workgroups
c. Attend flight test briefings and de-briefings
d. Review after flight data and pilot's daily reports
e. Perform product layer engineering support w.r.t. "system under test"
f. Participate in flight test execution as required
g. Participate in Flight Test Report review workgroups
h. Flight test data analysis

Outcomes

a. Flight test dossier documentation allocated to product layer team
b. Flight Test Report (provided by DAFT)

Deliverables

a. Nil

WBS 255 – System Release Preparation

Inputs

a. System Development and Verification Plan
b. System Configuration Management Plan
c. System Quality Assurance Plan
d. System Test Report
e. System data pack from formal test preparation activity
f. Flight Test Report

Work Description

a. Ensure completeness of life cycle data, after completion of flight tests
b. Ensure all configuration identification requirements are met, after completion of flight tests
c. Update the System Declaration of Design and Performance (DDP) to include references to the Flight Test Report
d. Review the DDP

Outcomes

a. Navigation System configuration management records
b. Quality assurance records
c. Verification Report
d. Navigation System Declaration of Design and Performance (DDP)

Deliverables

a. Verification Report
b. Navigation System Declaration of Design and Performance (DDP)

WBS 256 – Production Support

Inputs

a. System Development and Verification Plan
b. System Configuration Management Plan
c. System Quality Assurance Plan
d. System data pack

Work Description

a. Resolve queries arising during the production phase
b. Ensure all issues related to the upgraded navigation system identified during OT & E are recorded and addressed

Outcomes

a. Product layer FRACAS reports
b. Quality assurance records
c. Configuration management records

Deliverables

a. Nil

WBS 257 – Operations Support

Inputs

a. System Development and Verification Plan
b. System Configuration Management Plan
c. System Quality Assurance Plan
d. System data pack

Work Description

a. Resolve queries arising during operations
b. Ensure all issues related to the upgraded navigation system identified during operations are recorded and addressed

Outcomes

a. Product layer FRACAS reports
b. Quality assurance records
c. Configuration management records

Deliverables

a. Nil

WBS 280 - Supporting activities

WBS 281 – Modelling, simulation and rapid prototyping

(Note: outputs from these activities may not be used as verification evidence, and equipment produced from these activities may not be installed on airworthy aircraft)

Inputs

a. Requirements, engineering reports and other data required.

Work Description

a. Perform modelling, simulation and rapid prototyping tasks as required

Outcomes

a. Engineering reports

Deliverables

a. Engineering reports

WBS 282 – Quality Assurance

Inputs

a. System Quality Assurance Plan

Work Description

a. Perform QA activities as specified in the System QA Plan
b. Ensure all issues are recorded and addressed

Outcomes

a. Product layer QA reports
b. Quality assurance records
c. QA Audit Reports

Deliverables

a. QA Audit Reports

WBS 283 – Configuration Management

Inputs

a. System Configuration Management Plan

Work Description

a. Perform CM activities as specified in the System CM Plan
b. Ensure all issues are recorded and addressed

Outcomes

a. Product layer CM reports
b. Configuration management records

Deliverables

a. As per CM Plan

Appendix D: Glossary of systems engineering terms

Term	Definition	Reference
Acceptance	Acknowledgement by the certification authority that a submission of data, argument, or claim of equivalence satisfies applicable requirements.	SAE ARP4754 (Appendix B)
Acquirer	The stakeholder that acquires or procures a product or service from a supplier. An enterprise, organisation, or individual that obtains a product (good or service) from a supplier. Note 1: The acquirer can be a customer or user of a desired system product, or can be a developer obtaining a lower layer product in the system hierarchy from another vendor or developer in the role of supplier. Note 2: An acquirer is a type of stakeholder. An organisation that acquires or procures a system, software product or service from a supplier.	ISO/IEC 15288:2002(E) EIA-632 (Annex A) ISO/IEC 12207:1995
Acquisition	The process of obtaining a system, software product or service.	ISO/IEC 12207:1995
Activity	Set of actions that consume time and resources and whose performance is necessary to achieve or contribute to the realisation of one or more outcomes.	ISO/IEC 15288:2002(E)
Agreement	The mutual acknowledgement of terms and conditions under which a working relationship is conducted. Acknowledgement by the certification authority that a plan or proposal relating to, or supporting, an application for approval of a system or equipment, is an acceptable statement of intent with respect to applicable requirements. An arrangement, not necessary contractual, between two parties (an acquirer and a supplier) that defines the tasks to be performed, the items to be delivered, the acceptance criteria to be applied to delivered systems, and other requirements affecting the development or procurement of system products. The definition of terms and conditions under which a working relationship will be conducted.	ISO/IEC 12588:2002(E) SAE ARP4754 (Appendix B) IEEE Std 1220-2005 ISO/IEC 12207:1995

Term	Definition	Reference
Airworthiness	The condition of an item (aircraft, aircraft system, or part) in which that item operates in a safe manner to accomplish its intended function. Airworthiness encompasses the activities required to ensure that the aircraft is fit to fly, including assurances that the design is safe, that reasonable testing was performed against accepted performance and safety requirements and that potential hazards are identified and managed.	SAE ARP4754 (Appendix B) Denel Aviation
Algorithm	A finite set of well defined rules that give a sequence of operations for performing a specific task.	RTCA/DO-178B (glossary)
Allocation	The decision to assign a function or decision to hardware, software, or humans. Note: Allocation may be made entirely to one of these three system element types or to some combination to be resolved upon further functional decomposition. (see also assign)	IEEE Std 1220-2005
Analysis	An evaluation based on decomposition into simple elements. An examination of data that should lead to repeatable results.	SAE ARP4754 (Appendix B) RTCA/DO-178B para 6.3
Anomalous behaviour	Behaviour that is inconsistent with specified requirements.	RTCA/DO-178B (glossary)
Approval	The agreement that an item is complete and suitable for its intended use. The act of formal sanction of an implementation by a certification authority. The act or instance of expressing a favourable opinion or giving formal or official sanction	EIA-649 (Definitions, #3) SAE ARP4754 (Appendix B) RTCA/DO-178B (glossary)
Approved	Accepted by the certification authority as suitable for a particular purpose.	SAE ARP4754 (Appendix B)
Assembly	A number of parts, subassemblies, or any combination thereof, joined together to perform a specific function and which can be disassembled without destruction of designed use.	SAE ARP4754 (Appendix B)
Assign	Designate a function, product, process, or other item as accountable for a particular purpose. Note 1: The terms *allocate* and *partition* are used in some domains to denote this concept. Note 2: The “assign” relationship can be in various forms: a) requirement to function, b) requirement to product or process, c) requirement to interface, d) function to product or process, e) function to external entity (e.g. operator), or f) requirement to external entity (e.g., external system).	EIA-632 (Annex A)

Term	Definition	Reference
Associated processes	Processes that enable one or more end products to be put into service, maintained in service, or retired from service.	EIA-632 (Annex A)
Assurance	The planned and systematic actions necessary to provide adequate confidence and evidence that a product or process satisfies given requirements.	RTCA/DO-178B (glossary) (also quoted by SAE ARP4754)
Audit	An independent examination of the software life cycle processes and their outputs to confirm required attributes. Conducted by an authorised person for the purpose of providing an independent assessment of software products and processes in order to assess compliance with requirements.	RTCA/DO-178B (glossary) ISO/IEC 12207:1995
Authority	The organisation or person responsible within the State (country) concerned with the certification of compliance with applicable requirements.	SAE ARP4754 (Appendix B)
Availability	Probability that an item is in a functioning state at a given point in time.	SAE ARP4754 (Appendix B)
Baseline	A specification or product that has been formally reviewed and agreed upon, that thereafter serves as the basis for further development, and that can be changed only through formal change control procedures. The approved, recorded configuration of one or more configuration items, that thereafter serves as the basis for further development, and that is changed only through change control procedures. A formally approved version of a configuration item, regardless of media, formally designated and fixed at a specific time during the configuration item's life cycle. A set of documents formally established and issued to reflect a particular milestone during a product's life cycle. Baselines, plus approved changes from those baselines, constitute the current approved product configuration through change control procedures.	ISO/IEC 15288:2002(E) RTCA/DO-178B (glossary) ISO/IEC 12207:1995 Ref? (internet)
Baseline: Product baseline (PBL):	The documentation describing all functional, physical and interface characteristics necessary for production acceptance and product support.	Denel definition
Baseline: Allocated baseline (ABL):	The documentation defining a subsystem's functional and interface characteristics in relation to the overall system requirements, including design constraints and verification methods required to demonstrate that requirements have been met.	Denel definition

Term	Definition	Reference
Baseline: Functional baseline (FBL):	The documentation describing sub-systems functional characteristics (requirements) and the method used to demonstrate that those requirements have been met.	Denel definition
Baseline: Operational Baseline (OBL) (New):	The documentation describing all operational, maintenance and logistic characteristics for operational deployment and support.	Denel definition
Building block	A representation of the conceptual framework of a system that is used for organising the requirements, work, and other information associated with engineering a system. An element in the structured decomposition of a system.	EIA-632 (Annex A)
Cascading failure	A failure for which the probability of occurrence is substantially increased by the existence of a previous failure.	SAE ARP4754 (Appendix B)
Certification	Legal recognition by the certification authority that a product complies with the requirements. Such certification comprises the activity of technically checking the product and the formal recognition of compliance with the applicable requirements by issuing an approval (document) as required by national laws and procedures. In particular, certification of a product involves: (a) the process of assessing the design of a product to ensure that it complies with a set of standards applicable to that type of product so as to demonstrate an acceptable level of safety; (b) the process of assessing an individual product to ensure that it conforms with the certified type design; (c) the issuance of a certificate required by national laws to declare that compliance or conformity has been found with standard in accordance with items (a) or (b) above.	RTCA/DO-178B (glossary) (condensed) (also quoted almost verbally by SAE ARP4754)
Certification authority	Organisation or person responsible for granting approval on behalf of the nation of manufacture.	SAE ARP4754 (Appendix B)
Certification credit	Acceptance by the certification authority that a process, product or demonstration satisfies a certification requirement.	RTCA/DO-178B (glossary)
Certification Evidence	Documented proof of the outcome of a verification or validation activity.	Denel definition
Change	Updating and implementing a set of requirements and its associated life cycle data and end product.	Denel definition
Change authorisation	Control mechanism that allows changes to product artefacts.	Denel definition

Term	Definition	Reference
Change control / configuration control	(1) The process of recording, evaluating, approving or disapproving and coordinating changes to configuration items after formal establishment of their configuration identification or to baselines after their establishment. (2) The systematic evaluation, coordination, approval or disapproval and implementation of approved changes in the configuration of a configuration item after formal establishment of its configuration identification or to baselines after their establishment.	RTCA/DO-178B (glossary) (also quoted almost verbally by SAE ARP4754)
Code	The implementation of particular data or a particular computer program in a symbolic form, such as source code, object code or machine code.	RTCA/DO-178B (glossary)
Common cause analysis	Generic term encompassing zonal analysis, particular risk analysis, and common mode analysis.	SAE ARP4754 (Appendix B)
Common mode failure	An event which simultaneously affects a number of elements otherwise considered to be independent.	SAE ARP4754 (Appendix B)
Complexity	Complexity is an attribute of systems or items which makes their operation difficult to comprehend. Increased system complexity is often caused by such items as sophisticated components and multiple interrelationships.	SAE ARP4754 (Appendix B)
Compliance	Successful performance of all mandatory activities: agreement between the expected or specified result and the actual result.	SAE ARP4754 (Appendix B)
Component	A self contained part, combination of parts, sub-assemblies or units, which performs a distinct function of a system, necessary to the operation of the system.	Combined from RTCA/DO-178B (glossary) / SAE ARP4754 (Appendix B)
Condition	A Boolean expression containing no Boolean operators.	RTCA/DO-178B (glossary)
Condition/Decision Coverage	Every point of entry and exit in the program has been invoked at least once, every condition in a decision in the program has taken on all possible outcomes at least once, and every decision in the program has taken on all possible outcomes at least once.	RTCA/DO-178B (glossary)
Configuration baseline	A system configuration first established no later than that point in the development where credit for development assurance activities is first desired. Documented change control procedures should be followed subsequently.	SAE ARP4754 (Appendix B)
Configuration identification	(1) The process of identifying and defining the configuration items in a system and recording their characteristics (2) The approved documentation that defines a configuration item.	RTCA/DO-178B (glossary)

Term	Definition	Reference
Configuration item	(1) One or more hardware or software components treated as a unit for configuration management purposes (2) Software life cycle data treated as a unit for configuration management purposes. An entity within a configuration that satisfies an end use function and that can be uniquely identified at a given reference point.	RTCA/DO-178B (glossary) ISO/IEC 12207:1995
Configuration management	(1) The process of identifying and defining the configuration items of a system, controlling the release and change of these systems throughout the software life cycle, recording and reporting the status of configuration items and change requests and verifying the completeness and correctness of configuration items.(2) A discipline applying technical and administrative direction and surveillance to (a) identify and record the functional and physical characteristics of a configuration item, (b) control changes to those characteristics, and (c) record and report change control processing and implementation status. A management process for establishing and maintaining consistency of a product's performance, functional, and physical attributes with its requirements, design, and operational management information throughout its life. (ref ANSI/EIA-649).	RTCA/DO-178B (glossary) (also quoted almost verbally by SAE ARP4754) EIA-632 (Annex A)
Configuration status accounting	The recording and reporting of the information necessary to manage a configuration effectively, including a listing of the approved configuration identification, the status of proposed changes to the configuration and the implementation status of approved changes.	RTCA/DO-178B (glossary)
Conformance	Established as correct with reference to a standard, specification or drawing.	SAE ARP4754 (Appendix B)
Constraint	A limitation or implied requirement that constrains the design solution or implementation of the SEP and is not changeable by the enterprise. Note: A constraint is generally nonallocable. (1) A restriction, limit, or regulation imposed on product, project or process. (2) A type of requirement or design feature that cannot be traded off.	IEEE Std 1220-2005 EIA-632 (Annex A)
Control coupling	The manner or degree by which one software component influences the execution of programs in a computer system.	RTCA/DO-178B (glossary)

Term	Definition	Reference
Coverage analysis	The process of determining the degree to which a proposed software verification process activity satisfies its objective.	RTCA/DO-178B (glossary)
Criticality	Indication of the hazard level associated with a function, hardware, software, etc., considering abnormal behaviour (of this function, hardware, software etc) alone or in combination with external events.	SAE ARP4754 (Appendix B)
Defect	State of an item consisting of the non-performance of specified requirements by a characteristic of the item. A defect may, but need not, lead to a failure.	SAE ARP4754 (Appendix B)
Deliverable	An item agreed to be delivered to an acquirer as specified in an agreement. This item can be a document, a hardware item, a software item,, a service, or any type of work product.	EIA-632 (Annex A)
Demonstration	A method of proof of performance by observation (e.g. of a test or system operation).	SAE ARP4754 (Appendix B)
Derivative system	A special type of precedented system derived from a previously operational system through the use of major elements, but whose requirements have been modified to meet new objectives.	EIA-632 (Annex A)
Derived requirements	Additional requirements resulting from the design or implementation decisions during the development process. Derived requirements are not directly traceable to higher level requirements: though derived requirements can influence higher-level requirements. (1) A requirement that is further refined from a primary source requirement or a higher-level derived requirement. (2) A requirement that results from a design decision for a logical or physical solution representation.	SAE ARP4754 (Appendix B) EIA-632 (Annex A)
Design architecture	An arrangement of design elements that provides the design solution for a product or life cycle process intended to satisfy the functional architecture and the requirements baseline.	IEEE Std 1220-2005
Design characteristic	The design attributes or distinguishing features that pertain to a measurable description of a product or process.	IEEE Std 1220-2005
Design error	A mistake in the design process resulting from incorrect methods or incorrect application of methods or knowledge.	SAE ARP4754 (Appendix B)
Design process	The process of creating a system or an item from a set of requirements.	SAE ARP4754 (Appendix B)
Developer	An organisation that performs development activities (including requirements analysis, design, testing through acceptance) during the software life cycle process.	ISO/IEC 12207:1995
Development	The action by which a set of requirements is translated into a solution definition for a set of products that satisfy stakeholders.	EIA-632 (Annex A)

Term	Definition	Reference
Development assurance	All of those planned and systematic actions used to substantiate, at an adequate level of confidence, that development errors have been identified and corrected such that the system satisfies the applicable certification basis.	SAE ARP4754 (Appendix B)
Development error	A mistake in requirements determination or design.	SAE ARP4754 (Appendix B)
Document	A collection of data, regardless of the medium on which it is recorded, that generally has permanence and can be read by humans or machines.	EIA-632 (Annex A)
Effectiveness analysis	An analysis of how well a design solution will perform or operate given anticipated operational scenarios. An assessment of how well a product associated with an alternative logical, physical, or design solution is expected to perform or operate, given an anticipated usage scenario.	IEEE Std 1220-2005 EIA-632 (Annex A)
Effectiveness assessment	The evaluation of the design solution with respect to manufacturing, test, distribution, operations, support, training, environmental impact, cost effectiveness, and life cycle cost.	IEEE Std 1220-2005
Enabling product	Item that provides the means for a) getting an end product into service, b) keeping it in service, or c) ending its service.	EIA-632 (Annex A)
Enabling system	A system that complements a system-of-interest during its life cycle stages but does not necessarily contribute directly to its function during operations. Note 1: For example, when a system-of-interest enters the production stage, an enabling production is required. Note 2: Each enabling system has a life cycle of its own. This International Standard (ISO/IEC 15288) is applicable to each enabling system when, in its own right, it is treated as a system-of-interest.	ISO/IEC 15288:2002(E)
End item	An entity (hardware equipment, software equipment, data, facilities, material, services, and/or techniques) identified with an element of the SBS.	IEEE Std 1220-2005
End product	The portion of a system that performs the operational functions and is delivered to an acquirer.	EIA-632 (Annex A)

Term	Definition	Reference
End product validation	Confirmation by examination and provision of objective evidence that the specific intended use of an end product (developed or purchased), or an aggregation of end products, is accomplished in an intended usage environment. Note 1: The key difference between end product validation and end product verification is that end product validation answers the question: Does the delivered end product conform to the validated input acquirer requirements, certification criteria, or acceptance criteria, as applicable? End product verification answers the question: Does the output end product comply with the output specified requirements from which the end products were built, coded, procured, or assembled and integrated? Note 2: End product validation is used to demonstrate that the product developed or purchased satisfies the validated acquirer requirements in the context of its intended use. Note 3: Validation against other stakeholder requirements, generally, is not required. These requirements generally act as constraints on either the solution or the process by which a solution is generated. Constraints on solutions will show up in specifications to which an end product is built, coded, or assembled, and then verified against. Process constraints will be evaluated during management reviews or in management reports. Note 4: Validated is used to designate the corresponding status.	EIA-632 (Annex A)
End product verification	Confirmation by examination and provision of objective evidence that the specified requirements to which and end product is built, coded or assembled have been fulfilled. Note 1: End product verification is used to demonstrate that the specified requirements (specifications) generated by the developer and used to build, code, or assemble the end product have been satisfied. Note 2: Verified is used to designate the corresponding status.	EIA-632 (Annex A)
Engineering life cycle	A sequence of phases that evolves an instance of a system from a concept to a set of products consistent with the exit criteria established for an enterprise-based life cycle phase.	EIA-632 (Annex A)
Engineering plan	The plan for implementing the process for engineering a system. The engineering plan reflects an integrated technical effort that balances all factors associated with meeting its life cycle requirements.	EIA-632 (Annex A)

Term	Definition	Reference
Enterprise	The organisation that performs specified tasks. The entity that has governance over a set of projects, or over organisations in which projects are carried out.	IEEE Std 1220-2005 EIA-632 (Annex A)
Enterprise-based life cycle	The incremental progress of a system from conception through disposal, marked by management established milestones with assigned exit criteria.	EIA-632 (Annex A)
Environment	(1) The natural conditions (weather, climate, ocean conditions, terrain, vegetation, dust, etc.) and induced conditions (electromagnetic interference, heat, vibration, etc.) that constrain the design definition for end products and their enabling products. (2) External factors affecting an enterprise or project. (3) External factors affecting development tools, methods, or processes.	EIA-632 (Annex A)
Environmental (Qualification) testing	Tests performed to demonstrate that the system will function to specification in the intended environment. The standard used by the aviation industry is RTCA/DO-160; issue E is the current version at the date of issue of the dissertation.	Denel definition
Error	An occurrence arising as a result of an incorrect action or decision by personnel operating or maintaining a system. With respect to software, a mistake in requirements, design or code.	SAE ARP4754 (Appendix B) RTCA/DO-178B (glossary)
Evaluation	A systematic determination of the extent to which any entity meets its specified criteria.	ISO/IEC 12207:1995
Facility	The physical means or equipment for facilitating the performance of an action, e.g. buildings, instruments, tools.	ISO/IEC 15288:2002(E)
Failure	The inability of a system or system component to perform a required function within specified limits. A failure may be produced when a fault is encountered.	RTCA/DO-178B (glossary)
Failure condition	The effect on the aircraft and its occupants both direct and consequential, caused or contributed to by one or more failures, considering relevant adverse operational and environmental conditions. A failure condition is classified according to the severity of its effect as defined in FAA AC 25.1309-1A or JAA AMJ 25.1309.	RTCA/DO-178B (glossary) (also quoted almost verbally by SAE ARP4754)
Failure effect	A description of the operation of an item as the result of a failure.	SAE ARP4754 (Appendix B)
Failure mode	The way in which the failure of an item occurs.	SAE ARP4754 (Appendix B)
Failure rate	The performance figures within a hardware item population, calculated by dividing the number of failures by the total unit operating hours.	SAE ARP4754 (Appendix B)

Term	Definition	Reference
Fault	A manifestation of an error in software. A fault, if it occurs, may cause a failure.	RTCA/DO-178B (glossary)
Firmware	The combination of a hardware device and computer instructions or computer data that reside as read-only software on the hardware device. The software cannot be readily modified under program control.	ISO/IEC 12207:1995
Formal inspection	A set of procedures used to verify outputs of an activity by means of peer review, also known as Fagan Inspection (Ref [27]).	Denel definition
Formal methods	Descriptive notations and analytical methods used to construct, develop and reason about mathematical models of system behaviour.	RTCA/DO-178B (glossary)
Formal test	Test performed on an item after the associated change authorisation has been closed, normally performed or witnessed by person not associated with the development team.	Denel definition
Function	A task, action, or activity performed to achieve a desired outcome.	EIA-632 (Annex A)
Functional architecture	An arrangement of functions and their sub-functions and interfaces (internal and external) that defines the execution sequencing, conditions for data flow, and the performance requirements to satisfy the requirements baseline.	IEEE Std 1220-2005
Functional Hazard Assessment (FHA)	A systematic, comprehensive examination of aircraft functions to identify and classify Failure Conditions of those functions according to their severity.	SAE ARP4754 (Appendix B)
Functional requirement	A statement that identifies what a product or process must accomplish to provide required behaviour and/or results. A requirement that defines what system products must do and their desired behaviour in terms of an effect produced, or an action or service to be performed. Note 1: An example of a behaviour is “system switches from standby mode to run mode;” an example of an effect produced is “cause an alert signal;” an example of an action or service to be performed is “signal opens valve.” Note 2: A functional requirement can include the actor that is to perform the function, the function to be performed, and, if appropriate, the object acted upon. In addition, this information can be complemented by a statement of the environment within which the function is performed, the conditions that cause the function to start, the performance requirements associated with that function, and the conditions that cause the function to terminate.	IEEE Std 1220-2005 EIA-632 (Annex A)

Term	Definition	Reference
Guidelines	Recommended procedures for complying with regulations.	SAE ARP4754 (Appendix B)
Hardware	An item that has physical being. Generally refers to such items as line replaceable units or modules, circuit cards, and power supplies.	SAE ARP4754 (Appendix B)
Hardware/software integration	The process of combining the software into the target computer.	RTCA/DO-178B (glossary)
Hazard	A potentially unsafe condition resulting from failures, malfunctions, external events, errors, or combinations thereof.	SAE ARP4754 (Appendix B)
High level requirements	Software requirements developed from analysis of system requirements, safety related requirements, and system architecture.	RTCA/DO-178B (glossary)
Host computer	The computer on which the software is developed.	RTCA/DO-178B (glossary)
Human systems engineering	The activities involved throughout the system life cycle that address the human element of system design (including usability, measures of effectiveness, measures of performance, and total ownership cost) and that include the definition and synthesis of manpower, personnel, training, human engineering, health hazards, and safety issues.	IEEE Std 1220-2005
Implementation	The act of creating a physical reality from a specification.	SAE ARP4754 (Appendix B)
Increment	An addition to a closed set of product artefacts.	
Independence	1. A design concept that ensures that the failure of one item does not cause a failure of another item. (derived from JAR AMJ 25.1309). 2. Separation of responsibilities that assures the accomplishment of objective evaluation. Separation of responsibilities which ensures the accomplishment of objective evaluation. (1) For software verification process activities, independence is achieved when the verification activity is performed by a person(s) other than the developer of the item being verified, and a tool(s) may be used to achieve an equivalence to the human verification activity. (2) For the software quality assurance process, independence also includes the authority to ensure corrective action.	SAE ARP4754 (Appendix B) RTCA/DO-178B (glossary)
Informal test	Testing performed during the integration phase, to verify correctness of the test documentation and maturity of the item under test. Test results can not be used as certification evidence.	Denel definition

Term	Definition	Reference
Information database	A repository that provides a capacity to maintain work products and outcomes from implementation of the processes for engineering a system in a controlled manner. Note: This database provides the basis for controlled maintenance of the information needed by the multidisciplinary teams and management to efficiently and effectively accomplish their assigned tasks. It typically contains the requirements, configurations of a system (past, current, and planned), and all analyses and test results. This database allows for traceability, supports the validation and verification tasks, is essential for change management, and provides information to support decision making. See also integrated repository.	EIA-632 (Annex A)
Inspection	An examination of an item against a specific standard.	SAE ARP4754 (Appendix B)
Integral process	A process which assists the software development process and other integral processes and, therefore, remains active throughout the software life cycle. The integral processes are the software verification process, the software quality assurance process, the software configuration management process and the certification liaison process.	RTCA/DO-178B (glossary)
Integrated repository	A repository for storing all information pertinent to the SEP to include all data, schema, models, tools, technical management decisions, process analysis information, requirement changes, process and product metrics, and trade-offs.	IEEE Std 1220-2005
Integration	1. The act of causing elements of a system to function together. 2. The act of gathering a number of separate functions within a single implementation.	SAE ARP4754 (Appendix B)
Integrity	Attribute of a system or an item indicating that it can be relied upon to work correctly on demand.	SAE ARP4754 (Appendix B)
Interchangeability	The ability to substitute one item for another within a system and have the system perform to its specification.	SAE ARP4754 (Appendix B)

Term	Definition	Reference
Interface requirement	A requirement that defines the conditions of interaction between items. Note 1: Interface requirements include both logical and physical interfaces. They include, as necessary, physical measurements, definitions of sequences of energy or information transfer, and all other significant interactions between items. Note 2: There are interfaces between a system and things external to the system. The latter include, but are not limited to, interfaces between the end products and their operators or maintainers, the interfaces between items that make up an end product, an interfaces between an end product and enabling products of the associated processes. Note 3: For example, communication interfaces involve the movement and transfer of data and information within the system, and between the system and its environment. Proper evaluation of communications requirements involves definition of both the structural components of communications (e.g. bandwidth, data rate, distribution, etc.) and content requirements (what data information is being communicated, why it is being moved among the system components, and the criticality of this information to system functionality).	EIA-632 (Annex A)
Interface specification	The description of essential functional, performance, and design requirements and constraints at a common boundary between two or more system elements. Note: This includes interfaces between humans and hardware or software, as well as interfaces between humans themselves.	IEEE Std 1220-2005
Layer of development	(1) A level of abstraction as it relates to the system structure made up of building blocks. (2) A level of system decomposition.	EIA-632 (Annex A)
Life cycle	The system or product evolution initiated by a perceived stakeholder need through the disposal of the products. (1) An ordered collection of processes determined by an organisation to be sufficient and adequate to produce a product. (2) The period of time that begins with the decision to produce or modify a product and ends when the product is retired from service.	IEEE Std 1220-2005 (Adapted from RTCA/DO-178B)

Term	Definition	Reference
Life cycle model	A framework of processes and activities concerned with the life cycle, which also acts as a common reference for communication and understanding. A framework containing the processes, activities, and tasks involved in the development, operation, and maintenance of a software product, spanning the life of the system from the definition of its requirements to the termination of its use.	ISO/IEC 15288:2002(E) ISO/IEC 12207:1995
Low-level requirements	Software requirements derived from high level requirements, derived requirements, and design constraints from which source code can be directly implemented without further information.	RTCA/DO-178B (glossary)
Malfunction	The occurrence of a condition whereby the operation is outside specified limits.	SAE ARP4754 (Appendix B)
Mean Time Between Failures (MTBF)	Mathematical expectation of the time interval between two consecutive failures of a hardware item Note: The definition of this statistic has meaning only for repairable items. For non-repairable items, the term "mean life" is used.	SAE ARP4754 (Appendix B)
Means of compliance	The intended method(s) to be used by the applicant to satisfy the requirements stated in the certification basis for aircraft or engine. Examples include statements, drawings, analyses, calculations, testing, simulation, inspection, and environmental qualification. Advisory material issued by the certification authority is used if appropriate.	RTCA/DO-178B (glossary)
Measure of effectiveness (MOE)	The metrics by which and acquirer will measure satisfaction with product produced by the technical effort.	IEEE Std 1220-2005
Measure of performance (MOP)	An engineering performance measure that provides design requirements that are necessary to satisfy a MOE.	IEEE Std 1220-2005
Method	Techniques that support implementation of process tasks. Note: A method is the "how" of each task. Methods have the following attributes: a) thought patterns or approaches; b) knowledge base; c) rules and heuristics; d) structure and order; and e) notation.	EIA-632 (Annex A)
Operator	An individual who, or an organisation that, contributes to the functionality of a system and draws on knowledge, skills and procedures to contribute the function. Note 1: The role of the operator and role of the user may be vested, simultaneously or sequentially, in the same individual or organisation. Note 2: An individual operator combined with knowledge, skills and procedures may be considered as an element of the system.	ISO/IEC 15288:2002(E)

Term	Definition	Reference
Operational scenario	A sequence of events expected during operation of system products. Includes the environmental conditions and usage rates as well as expected stimuli (inputs) and responses (outputs).	EIA-632 (Annex A)
Organisation	A group of people and facilities with an arrangement of responsibilities, authorities and relationships.	ISO/IEC 15288:2002(E)
Performance requirement	The measurable criteria that identifies a quality attribute of a function or how well a functional requirement must be accomplished. A requirement that defines how well the system products are required to perform a function, along with the conditions under which the function is performed.	IEEE Std 1220-2005 EIA-632 (Annex A)
Precedented	An end product that is a legacy product undergoing modification or a product that the enterprise both has expertise to make and has similar products already in the market place.	EIA-632 (Annex A)
Preliminary System Safety Assessment (PSSA)	A systematic evaluation of a proposed system architecture and its implementation, based on the Functional Hazard Assessment and failure condition classification, to determine safety requirements for all items in the architecture.	SAE ARP4754 (Appendix B)
Procedure	A detailed description of the method by which an objective shall be achieved, i.e. a procedure describes how a specific task shall be performed. A procedure typically consists of an instruction, one or more templates and associated checklists.	Denel definition
Process	Set of interrelated or interacting activities that transform inputs into outputs. A collection of activities performed in the software life cycle to produce a definable output or product.	ISO/IEC 15288:2002(E) RTCA/DO-178B (glossary)
Product	Hardware, software, item or system generated in response to a defined set of requirements. (1) An item that consists of one or more of the following: hardware, software, firmware, facilities, data, materials, personnel, services, techniques, and procedures. (2) A constituent part of a system.	SAE ARP4754 (Appendix B) EIA-632 (Annex A)

Term	Definition	Reference
Project	An endeavour with start and finish dates undertaken to create a product or service in accordance with specified resources and requirements. A development effort consisting of both technical and management activities for the purpose of engineering a system. Note: For the purpose of this standard (EIA-632), project and program is synonymous.	ISO/IEC 15288:2002(E) EIA-632 (Annex A)
Prototype	A model (physical, electronic, digital, analytical, etc.) of a product built for the purpose of a) assessing the feasibility of a new or unfamiliar technology; b) assessing or mitigating technical risk; c) validating requirements; d) demonstrating critical features; e) verifying a product; f) validating a product; g) determining enabling product readiness; h) characterising performance or product features; or i) discovering physical principles.	EIA-632 (Annex A)
Qualification	The process of demonstrating whether an entity is capable of fulfilling specified requirements. See also verification and validation.	ISO/IEC 12207:1995
Qualification requirement	A set of criteria or conditions that have to be met in order to qualify a software product as complying with its specifications and being ready for use in its target environment.	ISO/IEC 12207:1995
Qualification testing	Testing, conducted by the developer and witnessed by the acquirer (as appropriate), to demonstrate that a software product meets its specifications and is ready for use in its target environment.	ISO/IEC 12207:1995
Quality assurance	All the planned and systematic activities implemented within the quality system, and demonstrated as needed, to provide adequate confidence that an entity will fulfil requirements for quality. Note 1: There are both internal and external purposes for quality assurance: a) Internal quality assurance: within an organisation, quality assurance provides confidence to management; a) External quality assurance: in contractual situations, quality assurance provides confidence to the customer or others.	ISO/IEC 12207:1995

Term	Definition	Reference
Redundant	Multiple independent means incorporated to accomplish a given function. Notes: 1. Distinction is made between the following redundant architecture principles: - similar redundancy (the multiple means are identical), - dissimilar redundancy 9the multiple means are of different types), - temporal redundancy (redundancy given by repetition of the operation). 2. The operation of redundant architectures may be classified as follows: - active redundancy (multiple means are routinely in operation and participating in carrying out the task), - passive redundancy (the additional means participate in carrying out task only in case of malfunction or failure), - warm passive redundancy (the additional means are always switched on), - cold passive redundancy (the additional means are switched on only in case of malfunction or failure).	SAE ARP4754 (Appendix B)
Release	The act of formally making available and authorising the use of a retrievable configuration item. A particular version of a configuration item that is made available for a specific purpose (for example, test release).	RTCA/DO-178B (glossary) ISO/IEC 12207:1995
Reliability	The probability that an item will perform a required function under specified conditions, without failure, for a specified period of time.	SAE ARP4754 (Appendix B)
Requirement	A statement that identifies a product or process operational, functional, or design characteristic or constraint, which is unambiguous, testable or measurable, and necessary for product or process acceptability (by consumers or internal quality assurance guidelines). An identifiable element of a function specification that can be validated and against which an implementation can be verified. (1) Something that governs what, how well, and under what conditions a product will achieve a given purpose.	IEEE Std 1220-2005 SAE ARP4754 (Appendix B) EIA-632 (Annex A)
Requirements validation	Confirmation by examination that requirements (individually and as a set) are well formulated and are useable for intended use.	EIA-632 (Annex A)

Term	Definition	Reference
Resource	An asset that is utilised or consumed during the execution of a process. Note 1: Resources may include diverse entities such as personnel, facilities, capital equipment, tools, and utilities such as power, water, fuel and communication infrastructures. Note 2: Resources may be reusable, renewable or consumable.	ISO/IEC 15288:2002(E)
Retirement	Withdrawal of active support by the operation and maintenance organisation, partial or total replacement by a new system, or installation of an upgraded system.	ISO/IEC 12207:1995
Reverse engineering	The method of extracting software design information from the source code.	RTCA/DO-178B (glossary)
Review	See technical review and formal inspection. The term review is used in the literature to denote either concept.	
Risk	The frequency (probability) of an occurrence and the associated level of hazard. (1) A measure combining the uncertainty of reaching a goal with the consequences of failing to reach the goal. (2) The probability of suffering injury or loss.	SAE ARP4754 (Appendix B) EIA-632 (Annex A)
Risk aversion	The act of avoiding risk. Averting risk can be through various means: mitigation, avoidance, transfer, or acceptance.	EIA-632 (Annex A)
Risk management	An organised process for identifying and assessing risks, and for implementing means to avoid them or mitigate their effect if they occur.	EIA-632 (Annex A)
Robustness	The extent to which software can continue to operate correctly despite invalid inputs.	RTCA/DO-178B (glossary)
Security	The protection of information and data so that unauthorised persons or systems are not denied access to them.	ISO/IEC 12207:1995
Safety	The state in which risk is lower than the boundary risk. The boundary risk is the upper limit of the acceptable risk. It is specified for a technical process or state.	SAE ARP4754 (Appendix B)
Similarity (as a certification strategy)	Applicable to systems similar in characteristics and usage to systems used on previously certified airplanes. In principle, there are no parts of the subject system more at risk (due to environment or installation) and that operational stresses are no more severe than on the previously certified system. If no identified failure condition of the subject system results in a more severe effect on the subject aircraft than the similar failure condition of the similar system on the reference aircraft, the prove reliability record of the similar system can be used to show compliance with the intent of JAR 25.1309 / FAR 25.1309.	SAE ARP4754 (Appendix B)

Term	Definition	Reference
Software	Computer programs, procedures, rules and any associated documentation pertaining to the operation of a computer system. Computer programs and, possibly, associated documentation and data pertaining to the operation of a computer system.	SAE ARP4754 (Appendix B) RTCA/DO-178B (glossary)
Software architecture	The structure of the software selected to implement the software requirements.	RTCA/DO-178B (glossary)
Software change	A modification in source code, object code, executable object code, or its related documentation from its baseline.	RTCA/DO-178B (glossary)
Software integration	The process of combining code components	RTCA/DO-178B (glossary)
Software library	A controlled repository containing a collection of software and related data and documents designed to aid in software development, use or modification. Examples include software development library, master library, production library, program library and software repository.	RTCA/DO-178B (glossary)
Software life cycle	(1) An ordered collection of processes determined by an organisation to be sufficient and adequate to produce a software product. (2)The period of time that begins with the decision to produce or modify a software product and ends when the product is retired from service.	RTCA/DO-178B (glossary)
Software partitioning	The process of separating, usually with the express purpose of isolating one or more attributes of the software, to prevent specific interactions and cross coupling interference.	RTCA/DO-178B (glossary)
Software product	The set of computer programs, and associated documentation and data, designated for delivery to a user. In the context of (RTCA/DO-178B) this term refers to software intended for use in airborne applications and the associated software life cycle data.	RTCA/DO-178B (glossary)
Software requirement	A description of what is to be produced by the software given the inputs and constraints. Software requirements include both high level requirements and low level requirements.	RTCA/DO-178B (glossary)
Software tool	A computer program used to help develop, test, analyse, produce or modify another program or its documentation. Examples are an automated design tool, a compiler, test tools and modification tools.	RTCA/DO-178B (glossary)
Source code	Code written in source languages, such as assembly language and/or high level language, in a machine-readable form for input to an assembler or a compiler.	RTCA/DO-178B (glossary)

Term	Definition	Reference
Specification	A document that fully describes a design element or its interfaces in terms of requirements (functional, performance, constraints, and design characteristics) and the qualification conditions and procedures for each requirement. A collection of requirements which, when taken together, constitute the criteria that define the functions and attributes of a system, or an item. A document that contains specified requirements for a product and the means to be used to determine that the product satisfies these requirements.	IEEE Std 1220-2005 SAE ARP4754 (Appendix B) EIA-632 (Annex A)
Specification tree	A hierarchy of specification elements and their interface specifications that identify the elements and the specifications related to design elements of the system configuration that are to be controlled.	IEEE Std 1220-2005
Stage	A period within the life cycle of that relates to the state of the system description or the system itself. Note 1: Stages relate to major progress and achievement milestones of the system through its life cycle. Note 2: Stages may be overlapping.	ISO/IEC 15288:2002(E)
Stakeholder	A party having a right, share or claim in a system or in its possession of characteristics that meet that party's needs and expectations. An enterprise, organisation, or individual having an interest or a stake in the outcome of the engineering of a system.	ISO/IEC 15288:2002(E) EIA-632 (Annex A)
Standard	A rule or basis of comparison used to provide both guidance in and assessment of the performance of a given activity or the content of a specified data item. A document that establishes engineering and technical requirements for products, processes, procedures, practises, and methods that have been decreed by authority or adopted by consensus.	RTCA/DO-178B (glossary) EIA-632 (Annex A)
State	A condition that characterises the behaviour of a function/sub function or element at a point in time.	IEEE Std 1220-2005
Statement of work	A document used by the acquirer as the means to describe and specify the tasks to be performed under the agreement.	Adapted from ISO/IEC 12207:1995
Structure	A specified arrangement or interrelationship of parts to form a whole.	RTCA/DO-178B (glossary)
Subsystem	A grouping of items that perform a set of functions within a particular end product.	EIA-632 (Annex A)

Term	Definition	Reference
Supplier	An organisation or an individual that enters into an agreement with the acquirer for the supply of a product or service. Provides a product (either end products, enabling products, or both) or a group of products to an acquirer. The supplier (external or internal to the acquirers organisation) can be a vendor that has a product that does not need development, or a developer that must develop the desired system product or products.	ISO/IEC 15288:2002(E) EIA-632 (Annex A)
System	A set or arrangement of elements [people, product (hardware and software) and processes (facilities, equipment, material and procedures)] that are related, and whose behaviour satisfies operational needs and provides for the life cycle sustainment of the products. A combination of interacting elements organised to achieve one or more stated purposes. Note 1: A system may be considered as a product or as the services it provides. Note 2: In practise, the interpretation of its meaning is frequently clarified by the use of an associated noun, e.g. aircraft system. Alternatively the word system may be substituted simply by a context dependent synonym, e.g. aircraft, though this may then obscure a system principles perspective. A combination of inter-related items arranged to perform a specific function. A collection of hardware and software components organised to accomplish a specific function or set of functions. An aggregation of end products and enabling products to achieve a given purpose. An integrated composite that consists of one or more of the processes, hardware, software, facilities and people, that provides a capability to satisfy a stated need or objective.	IEEE Std 1220-2005 ISO/IEC 15288:2002(E) SAE ARP4754 (Appendix B) RTCA/DO-178B (glossary) EIA-632 (Annex A) ISO/IEC 12207:1995
System architecture	The composite of the design architectures for products and their life cycle processes. The structure of the hardware and the software selected to implement the system requirements.	IEEE Std 1220-2005 RTCA/DO-178B (glossary)

Term	Definition	Reference
System breakdown structure (SBS)	A hierarchy of elements, related life cycle processes, and personnel used to assign development teams, conduct technical reviews, and to partition out the assigned work and associated resource allocation to each of the tasks necessary to accomplish the objectives of the project.	IEEE Std 1220-2005
System effectiveness	A measurement of the ability of a system to satisfy its intended operational uses as a function of how the system performs under anticipated environmental conditions, and the ability to produce, test, distribute, operate, support, train, and dispose of the system throughout its life cycle.	IEEE Std 1220-2005
System element	A member of a set of elements that constitutes a system. Note: A system element is a discrete part of a system that can be implemented to fulfil specified requirements.	ISO/IEC 15288:2002(E)
System life cycle	The evolution with time of a system-of-interest from conception through to retirement.	ISO/IEC 15288:2002(E)
System Safety Assessment (SSA)	A systematic, comprehensive evaluation of the implemented system to show that the relevant safety requirements are met. An ongoing, systematic, comprehensive evaluation of the proposed system to show that relevant safety-related requirements are satisfied.	SAE ARP4754 (Appendix B) RTCA/DO-178B (glossary)
System safety assessment process	Those activities which demonstrate compliance with airworthiness requirements and associated guidance material, such as, JAA AMJ/FAA AC25.1309. The major activities within this process include: functional hazard assessment, preliminary safety assessment, and system safety assessment. The rigor of the activities will depend on the criticality, complexity, novelty, and relevant service experience of the system concerned.	RTCA/DO-178B (glossary)
System-of-interest	The system whose life cycle is under consideration in the context of this International Standard (ISO/IEC 15288).	ISO/IEC 15288:2002(E)
System technical requirement	A requirement derived from one or more stakeholder requirements and stated in technical terms.	EIA-632 (Annex A)
Systems Engineering	Systems Engineering is an interdisciplinary approach and means to enable the realisation of successful systems. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, and then proceeding with design synthesis and system validation while considering the complete problem. Systems Engineering considers both business and the technical needs of all customers with the goal of providing a quality product that meets the user needs.	INCOSE Systems Engineering Handbook v.3

Term	Definition	Reference
Technical performance measurement (TPM)	The technique of predicting the future value of a key technical parameter of the higher-level end product under development, based on current assessment of products lower in the system structure. Note 1: Involves the continuing verification of the degree of anticipated and actual achievement for technical parameters. Confirms progress and identifies variances that that might jeopardise meeting a higher-level end product requirement. Assessed values falling outside established tolerances indicate a need for evaluation and corrective action. Note 2: Key characteristics of TPM are: a) *Achievement to Date* – present achieved value of the technical parameter based on estimates or actual measurement; b) *Current Estimate* – the value of the technical parameter predicted to be achieved by the end of the technical effort with remaining resources (including schedule and budget); c) *Technical Milestone* – a point where TPM evaluation is accomplished or reported; d) *Planned Value Profile* – the projected time-phased achievement projected for the technical parameter from the beginning of the development or as re-planned as a result of corrective projection; e) *Tolerance Band* – an envelope containing the Planned Value Profile and indicating the allowed variation and projected estimation error; f) *Objective* – the goal or desired value at the end of the technical effort; g) *Threshold* – the limiting acceptable value that, if not met, would jeopardise the project; h) *Variation* – the difference between the planned value and the achievement-to-date value.	EIA-632 (Annex A)
Technical review	An event at which the progress of the technical effort is assessed relative to its governing plans and technical requirements.	EIA-632 (Annex A)
Test	A quantitative procedure to prove performance using stated objective criteria with pass or fail results.	SAE ARP4754 (Appendix B)
Test article	An item built, constructed, coded, or otherwise implemented, for checking conformance to specified requirements or for checking validation against acquirer requirements for the item.	EIA-632 (Annex A)
Test case	A set of test inputs, execution conditions, and expected results developed for a particular objective, such as to exercise a particular program path or to verify compliance with a specific requirement.	RTCA/DO-178B (glossary)
Test coverage	The extent to which the test cases test the requirements for the system or software product.	ISO/IEC 12207:1995

Term	Definition	Reference
Testability	The extent to which an objective and feasible test can be designed to determine whether a requirement is met.	ISO/IEC 12207:1995
Test procedure	Detailed instructions for the set-up and execution of a given set of test cases, and instructions for the evaluation of results of executing the test cases.	RTCA/DO-178B (glossary)
Testing	The process of exercising a system or system component to verify that it satisfies specified requirements and to detect errors.	RTCA/DO-178B (glossary)
Time limited despatch	A means of allowing aircraft dispatch for a limited time with certain faults present.	SAE ARP4754 (Appendix B)
Tool qualification	The process necessary to obtain certification credit for a software tool within the context of a specific airborne system.	RTCA/DO-178B (glossary)
Traceability	The characteristic by which requirements at one level of a design may be related to requirements at another level. The evidence of an association between items, such as between process outputs, between an output and its originating process, or between a requirement and its implementation. The ability to identify the relationships between various artefacts of the development process, i.e., the lineage of requirements, the relationship between a design decision and the affected requirements and design features, the assignment of requirements to design features, the relationship of test results to the original source of requirements.	SAE ARP4754 (Appendix B) RTCA/DO-178B (glossary) EIA-632 (Annex A)
Trade-off	Decision-making actions that select from various requirements and alternative solutions on the basis of net benefit to the stakeholders.	ISO/IEC 15288:2002(E)
Transition criteria	The minimum conditions, as defined by the software planning process, to be satisfied to enter a process.	RTCA/DO-178B (glossary)
Unintended function	A function that is visible at the airplane level and was neither intended nor a predicted fault condition in the PSSA. Only when an unintended function leads to an airplane –level hazard or a degradation of an intended function is it considered significant relative to certification.	SAE ARP4754 (Appendix B)
Unit Testing		
Unprecedented	A specific end product that is not known *a priori*, or the enterprise has limited experience in developing this type of system.	EIA-632 (Annex A)
User	Individual or group that benefits from a system during its utilisation. Note: The role of the user and the role of the operator may be vested, simultaneously or sequentially, in the same individual or organisation.	ISO/IEC 15288:2002(E)

Term	Definition	Reference
Validation	Confirmation, through the provision of objective evidence, that the requirements for a specific intended use or application have been fulfilled. Note: Validation in a system life cycle context is the set of activities ensuring and gaining confidence that a system is able to accomplish its intended use, goals and objective.	ISO/IEC 15288:2002(E)
	The determination that the requirements for a product are sufficiently correct and complete.	SAE ARP4754 (Appendix B)
	The process of determining that the requirements are the correct requirements and that they are complete. The system life cycle process may use software requirements and derived requirements in system validation.	RTCA/DO-178B (glossary)
	Confirmation by examination and provision of objective evidence that the particular requirements for a specific intended use are fulfilled. Note 1: In design and development, validation concerns the process of examining a product to determine conformity with user needs. Note 2: Validation is normally performed on the final product under defined operating conditions. It may be necessary in earlier stages. Note 3: “Validated” is used to designate the corresponding status. Note 4: Multiple validations may be carried out if there are different intended uses. See also end product validation and requirements validation.	ISO/IEC 12207:1995

Term	Definition	Reference
Verification	Confirmation, through the provision of objective evidence, that specified requirements have been fulfilled. Note: Verification in a system life cycle context is a set of activities that compares a product of the system life cycle against the required characteristics for that product. This may include, but is not limited to, specified requirements, design description and the system itself. The evaluation of an implementation of requirements to determine they have been met. The evaluation of the results of a process to ensure correctness and consistency with respect to the inputs and standards provided to that process. Confirmation by examination and provision of objective evidence that specified requirements have been fulfilled. Note 1: In design and development, verification concerns the process of examining the result of a given activity to determine conformity with the stated requirement for that activity. Note 2: Verified is used to designate the corresponding status. See also end product verification.	ISO/IEC 15288:2002(E) SAE ARP4754 (Appendix B) RTCA/DO-178B (glossary) ISO/IEC 12207:1995
Version	An identified instance of an item.	
Workflow	A workflow is a logical concatenation of activities. Note: Modification to a version of a software product, resulting in a new version, requires configuration management action.	ISO/IEC 12207:1995
Zonal safety	The safety standard with respect to installation, interference between systems, and potential maintenance errors.	SAE ARP4754 (Appendix B)